UNI-T®

Instruments.uni-trend.com

Programming Manual

Spectrum/Signal Analyzer Programming Manual

This document applies to the following models:

UTS5000A series

UTS3000B/T series

UTS3000T+ series

UTS3000A series

UTS1000B/T series

Warranty and Statement

Copyright

Brand Information

UNI-T is the registered trademark of Uni-Trend Technology (China) Co., Ltd.

File Number

2024.09.14

Software Version

V1.03.0009

Software upgrade may have some change and add more function, please subscribe to the **UNI-T** website to get the new version or contact UNI-T.

Statement

- UNI-T products are protected by patents (including obtained and pending) in China and other countries and regions.
- UNI-T reserves the right to change specifications and prices.
- The information provided in this manual supersedes all previous publications.
- The information provided in this manual is subject to change without notice.
- UNI-T shall not be liable for any errors that may be contained in this manual. For any incidental or consequential damages arising out of the use or the information and deductive functions provided in this manual.
- No part of this manual shall be photocopied, reproduced, or adapted without the prior written permission of UNI-T.

Product Certification

UNI-T has certified that the product conforms to China national product standard and industry product standard as well as ISO9001:2015 standard and ISO14001:2015 standard. UNI-T will go further to certificate product to meet the standard of other member of the international standards organization.

Contact Us

If you have any questions or problems, please contact **UNI-T**.

Website: https://www.uni-trend.com

SCPI

SCPI（Standard Commands for Programmable Instruments）is a standardized instrument programming language that builds on existing standards IEEE 488.1 and IEEE 488.2 and follows the floating-point rules of IEEE 754 standard, ISO 646 message exchange 7-bit encoding notation (equivalent to ASCII programming) and many other standards.

This section introduces the format, symbols, parameters, and abbreviations of the SCPI command.

Instruction Format

The SCPI command is a tree-like hierarchy consisting of multiple subsystems, each subsystem consisting of a root keyword and one or more hierarchical key words. **The command line usually begins with a colon ":"; Keywords are separated by the colon ":", followed by optional parameter settings. The command keyword is separated by spaces from the first parameter. The command string must end with a newline <NL> character. Add the question mark "? " after the command line. It is usually indicated that this feature is being queried.**

Symbol Functional Descriptionn

The following four symbols are not part of the SCPI command. They cannot be sent with the command, but it is commonly used for supplementary specification.

- **Braces { }**

  It usually contains multiple optional parameters, one of which must be selected when sending a command.

  For example, the command :DISPlay:GRID:MODE　{ FULL | GRID | CROSS | NONE}

- **Vertical Bar |**

  It is used to separate multiple parameters, one of which must be selected when sending a command.

  For example, the command :DISPlay:GRID:MODE　{ FULL | GRID | CROSS | NONE}

- **Square Brackets [ ]**

  The contents in square brackets (command keywords) can be omitted. If the parameter is omitted, the instrument will set the parameter to the default value.

  For example, the command :MEASure:NDUTy? [<source>], [<source>] represents the current

channel.

- **Angle Braces < >**

  The parameter enclosed in the angle brackets must be replaced by an effective value.

  For example, use the command :DISPlay:GRID:BRIGhtness 30 to send the command :DISPlay:GRID:BRIGhtness <count>

Parameter Functional Description

The parameters in this manual can be divided into five types: Boolean, Integer, Real, Discrete, and ASCII string.

- **Boolean**

  The available value for the parameter is "ON" (1) or "OFF" (0).

  For example, :SYSTem:LOCK {{1 | ON} | {0 | OFF}}.

- **Integer**

  Unless otherwise specified, the parameter can be any integer within the effective value range.

  Notice: Do not set the parameter to a decimal or in scientific notation, otherwise, errors will occur.

  For example, :DISPlay:GRID:BRIGhtness <count>, <count> can take integer form 0-100.

- **Real**

  Unless otherwise specified, the parameter can be any real value (in decimal form or in scientific notation) within the effective value range.

  For example, for CH1, <offset> in the command :CHANnel1:OFFSet <offset> can take value as real

- **Discrete**

  The parameter can only take a few specified numbers or characters.

  For example, the parameter in the command :DISPlay:GRID:MODE { FULL | GRID | CROSS | NONE}, it can only be FULL, GRID, CROSS or NONE.

- **ASCII**

  Character string parameters can contain all ASCII sets. Character string must begin and end with paired quotes; it can use single or double quotation marks. The quotation and delimiter can also be part of a string by typing it twice and not adding any characters.

  For example, set IP: SYST:COMM:LAN:IPAD "192.168.1.10"

Abbreviations

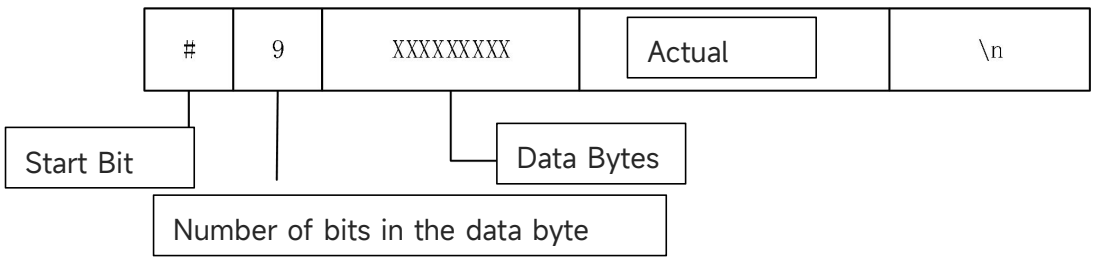
All the commands are case-insensitive. The commands can be all input in uppercase letters or in lowercase letters. For abbreviations, it should enter all the uppercase letters that exist in the command.

Data Return

The data return is divided into single data and batch data returns and must end with a newline “<NL>” character. If the data is of string type, return the string; if the data is of integer type, return the integer; if the data is of real numeric type, return it in scientific notation, with the part before 'e' retaining the actual significant digits after the decimal point, and the part after 'e' retaining three digits. The batch data return format is "data block header + data block", where the data block header has the following format: the first digit (9) after “#” indicates the remaining number of digits in the data block header; the remaining digits in the data block header indicate the number of bytes of data to be transmitted this time (if less than 9 digits, pad with zeros in front). For example, the header for sending a 1000-byte data block is “#9000001000”.

Data Block Format

DATA is a data stream, while others are in ASCII strings, represented as “<#9XXXXXXXXX + DATA + \n>”, as shown in the following figure.



Note: When returning invalid data, use “*” to indicate ASCII type; for real numeric type, return the maximum value that can be represented.

SCPI Explanation

IEEE488.2 General Command

*CLS

- **Command Format**

  *CLS

- **Functional Description**

  Clear the values of all event register and status byte register.

*ESE

- **Command Format**

  *ESE <integer>

  *ESE?

- **Functional Description**

  Set the enable value for standard event register. The decimal sum of the register bits, the default is 0. For example, to enable 2-bit (value: 4), 3-bit (value: 8), and bit-7 (value: 128), so the decimal sum should be 140 (4+8+128). 1-bit and 6-bit of the standard event register are not enabled.

  <integer>: An integer value

- **Return Format**

  The query returns the enable value of the standard event register as an integer value.

- **For Example**

*ESE 16	Enable the fourth bit for the standard event register.
*ESE?	The query returns 16.

*ESR?

- **Command Format**

  *ESR?

- **Functional Description**

  Query the standard event register. The event register is read-only register, lock the event from the condition register.

- **Return Format**

  The query returns the event value of the standard event register as an integer value.

- **For Example**

  *ESR?

  The query returns 24, with 3-bit and 4-bit enabled.

***IDN?**

- **Command Format**

  *IDN?

- **Functional Description**

  Query for manufacture name, product model, product serial number, and software version number.

- **Return Format**

  The query returns the manufacture name, product model, and product serial number. The software version is separated by commas.

  Notice: The return model number should be consistent with the nameplate.

- **For Example**

  UNI-T Technologies, UTS3036B, 000000001, 00.00.01

***OPC**

- **Command Format**

  *OPC

  *OPC?

- **Functional Description**

  Stores the standard event status register to 1 after the current operation is finished.

- **Return Format**

  The query returns whether the current operation is finished. 0 represents that the current operation is unfinished. 1 represents that the current operation is unfinished.

- **For Example**

  *OPC Set the standard event status byte register to 1.

  *OPC?

  The query returns 1. If the current operation is finished; otherwise, the query returns 0.

***RCL**

- **Command Format**

  *RCL <integer>

- **Functional Description**

Loading the storage state of the instrument, which is equivalent to the command :MMEMory:LOAD:STATe "STATE_n.state".

<integer>: The range is from 1 to 16.

- **Return Format**

  No return value.

- **For Example**

  *RCL 1 Load the state file of 1 (STATE_1.state).

***RST**

- **Command Format**

  *RST

- **Functional Description**

  Restore the instrument to its factory default settings, clear all the error messages and send and receive queue buffers.

***SAV**

- **Command Format**

  *SAV <integer>

- **Functional Description**

  Save the storage state of the instrument, which is equivalent to the command :MMEMory:STORe:STATe "STATE_n.state".

- **Return Format**

  No return value.

- **For Example**

  *SAV 1 Save the state file of 1 (STATE_1.state)

***SRE**

- **Command Format**

  *SRE <integer>

  *SRE?

- **Functional Description**

  Sets the enable value of status byte register. Service Request Enable is the bit in the enable register of the status register. The enable register defines which bit in the event register will be reported to the “status byte” register group. The enable register is readable and writeable.

  <integer>: An integer value

- **Return Format**

  The query returns the enable value of status byte register and it is integer value.

- **For Example**

*SRE 24	Enable 3-bit and 4-bit for the enable register.
*SRE?	Query returns 24.

***STB?**

- **Command Format**

  *STB?

- **Functional Description**

  Query the status value of status byte register.

- **Return Format**

  The query returns the status value of status byte register and it is integer value.

- **For Example**

*STB?	40, set 3-bit and 5-bit of the status byte register.

***TRG**

- **Command Format**

  *TRG

- **Functional Description**

  Initiate an immediate scan.

- **Return Format**

  No return value.

- **For Example**

*TRG	Initiate an immediate scan.

Instrument Command

The command can be in common use for full function module.

INSTrument Command

:INSTrument:DEFault

- **Command Format**

  :INSTrument:DEFault

- **Functional Description**

Restore the current measurement function module.

- **Return Format**

  No return value.

- **For Example**

:INSTrument:DEFault	Restore the mode.

:INSTrument[:SELect]

- **Command Format**

  :INSTrument[:SELect] {SA|EMI|AD}

  :INSTrument[:SELect]?

- **Functional Description**

  Select the measurement function module.

  SA: Spectrum sweep

  EMI: Electromagnetic interference

  AD: Analog demodulation

- **Return Format**

  The query returns the current measurement function module, SA, EMI, or AD.

- **For Example**

:INSTrument:SELect SA	Select spectrum sweep.
:INSTrument:SELect?	The query returns SA.

SYSTem Command

This command is used to set the basic operations of the spectrum analyzer, including the full QWERTY lock and system data settings.

:SYSTem:COMMunicate:KEYBoard:STATe?

- **Command Format**

  :SYSTem:COMMunicate:KEYBoard:STATe?

- **Functional Description**

  Query the state of keyboard.

- **Return Format**

  The query returns the state of keyboard. 0 represents that the keyboard is not inserted. 1 represents that the keyboard is inserted.

- **For Example**

:SYSTem:COMMunicate:KEYBoard:STATe?	The query returns 1.

:SYSTem:COMMunicate:LAN:IPV4:ADAPter

- **Command Format**

  :SYSTem:COMMunicate:LAN:IPV4:ADAPter {{1|ON} | {0|OFF}}

  :SYSTem:COMMunicate:LAN:IPV4:ADAPter?

- **Functional Description**

  The switching of the network adapter.

- **Return Format**

  The query returns the switch status of network adapter: 0 or 1.

- **For Example**

:SYSTem:COMMunicate:LAN:IPV4:ADAPter ON	Turn on network adapter.
:SYSTem:COMMunicate:LAN:IPV4:ADAPter?	The query returns 1.

:SYSTem:COMMunicate:LAN:IPV4:CONFIG

- **Command Format**

  :SYSTem:COMMunicate:LAN:IPV4:CONFIG <ip>

  :SYSTem:COMMunicate:LAN:IPV4:CONFIG?

- **Functional Description**

  Set the IP address.

  <ip>: Four fields in dotted decimal format, represented as xxx.xxx.xxx.xxx.

- **Return Format**

  The query returns the current IP address in the format xxx.xxx.xxx.xxx.

- **For Example**

:SYSTem:COMMunicate:LAN:IPV4:CONFIG “192.168.20.111”	Set the IP address.
:SYSTem:COMMunicate:LAN:IPV4:CONFIG?	It returns 192.168.20.111.

:SYSTem:COMMunicate:LAN:IPV4:DHCP

- **Command Format**

  :SYSTem:COMMunicate:LAN:IPV4:DHCP {{1|ON} | {0|OFF}}

  :SYSTem:COMMunicate:LAN:IPV4:DHCP?

- **Functional Description**

  Set the DHCP switch.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of DHCP: 0 or 1.

- **For Example**

  :SYSTem:COMMunicate:LAN:IPV4:DHCP ON Turn on DHCP.

  :SYSTem:COMMunicate:LAN:IPV4:DHCP? The query returns 1.

:SYSTem:COMMunicate:LAN:IPV4:GATEway

- **Command Format**

  :SYSTem:COMMunicate:LAN:IPV4:GATEway <gateway>

  :SYSTem:COMMunicate:LAN:IPV4:GATEway?

- **Functional Description**

  Sets the gateway of the internet.

  <gateway>: Four fields in dotted decimal format, represented as xxx.xxx.xxx.xxx.

- **Return Format**

  The query returns the gateway data in the format xxx.xxx.xxx.xxx.

- **For Example**

  :SYSTem:COMMunicate:LAN:IPV4:GATEway “192.168.20.1”

  Sets the gateway to 192.168.20.1.

  :SYSTem:COMMunicate:LAN:IPV4:GATEway? The query returns 192.168.20.1.

:SYSTem:COMMunicate:LAN:IPV4:MASK

- **Command Format**

  :SYSTem:COMMunicate:LAN:IPV4:MASK <mask>

  :SYSTem:COMMunicate:LAN:IPV4:MASK?

- **Functional Description**

  Set the mask of the internet.

  <mask>: Four fields in dotted decimal format, represented as xxx.xxx.xxx.xxx.

- **Return Format**

  The query returns the mask data in the format xxx.xxx.xxx.xxx.

- **For Example**

  :SYSTem:COMMunicate:LAN:IPV4:MASK “255.255.255.0” Set the mask to 255.255.255.0.

  :SYSTem:COMMunicate:LAN:IPV4:MASK? The query returns 255.255.255.0.

:SYSTem:COMMunicate:LAN:STATe?

- **Command Format**

:SYSTem:COMMunicate:LAN:STATe?

- **Functional Description**

  Query the connecting state of device's network.

- **Return Format**

  The query returns the connecting state of device's network. 0 represents that the network is not connected. 1 represents that the network is connected.

- **For Example**

  :SYSTem:COMMunicate:LAN:STATe? The query returns 1.

:SYSTem:COMMunicate:MAC?

- **Command Format**

  :SYSTem:COMMunicate:MAC?

- **Functional Description**

  Query the MAC address of device.

- **Return Format**

  The query returns the MAC address of the device in character.

- **For Example**

  :SYSTem:COMMunicate:MAC? The query returns 3E:A4:20:3D:82:83.

:SYSTem:COMMunicate:MOUSe:STATe?

- **Command Format**

  :SYSTem:COMMunicate:MOUSe:STATe?

- **Functional Description**

  Query the mouse status.

- **Return Format**

  The query returns the mouse status. 0 represents that the mouse is not inserted. 1 represents that the mouse is inserted.

- **For Example,**

  :SYSTem:COMMunicate:MOUSe:STATe? The query returns 1.

:SYSTem:COMMunicate:USB:STATe?

- **Command Format**

  :SYSTem:COMMunicate:USB:STATe?

- **Functional Description**

  Query the status of USB flash drive.

- **Return Format**

  The query returns the status of USB flash drive. 0 represents that the mouse is not inserted. 1 represents that the mouse is inserted.

- **For Example**

  :SYSTem:COMMunicate:USB:STATe? The query returns 1.

:SYSTem:CONFigure

- **Command Format**

  :SYSTem:CONFigure <filename>

  :SYSTem:CONFigure?

- **Functional Description**

  Write/read the configuration file. Send the instruction first, and then send the configuration file data to the spectrum analyzer.

  <filename>: The filename of configuration file.

- **Return Format**

  The query returns the current configuration file data of the spectrum analyzer. Its binary system.

- **For Example**

  :SYSTem:CONFigure test

  Write the configuration file data into the spectrum analyzer and load it.

  :SYSTem:CONFigure?

  The query returns the current configuration file data of the spectrum analyzer. Its binary system.

:SYSTem:DATE

- **Command Format**

  :SYSTem:DATE <yyyymmdd>

  :SYSTem:DATE?

- **Functional Description**

  Set the date. The format is the year, month, day with no separating mark. For example: set the date to 2000, January 1st, the parameter is 20000101.

- **Return Format**

  The query returns the date.

- **For Example**

  :SYSTem:DATE? The query returns 20000101.

  (2000, January 1st)

:SYSTem:DEFault

- **Command Format**

  :SYSTem:DEFault

- **Functional Description**

  Restore to the factory settings.

- **Return Format**

  No return value.

- **For Example,**

  :SYSTem:DEFault Restore to the factory settings.

:SYSTem:DISPlay:BACKlight:INTensity

- **Command Format**

  :SYSTem:DISPlay:BACKlight:INTensity <integer>

  :SYSTem:DISPlay:BACKlight:INTensity?

- **Functional Description**

  Controls the level of backlight brightness of the system.

  <integer>: A continuous integer, with the default unit in %. The value range is from 0 to 100.

- **Return Format**

  The query returns the level of backlight brightness of the system. The unit is %.

- **For Example**

  :SYSTem:DISPlay:BACKlight:INTensity 80

  Set the level of backlight brightness of the system to 80%.

  :SYSTem:DISPlay:BACKlight:INTensity? The query returns 80.

:SYSTem:DISPlay:CFORmat

- **Command Format**

  :SYSTem:DISPlay:CFORmat {HR12|HR24}

  :SYSTem:DISPlay:CFORmat?

- **Functional Description**

  Set the time format of the system.

  HR12: 12-hour system

  HR24: 24-hour system

- **Return Format**

  The query returns the time format of the system: HR12 or HR24.

- **For Example**

:SYSTem:DISPlay:CFORmat HR24	Set the time format to HR24.
:SYSTem:DISPlay:CFORmat?	The query returns HR24.

:SYSTem:DISPlay:LANGuage

- **Command Format**

  :SYSTem:DISPlay:LANGuage {CHINese|ENGLish|GERMan}

  :SYSTem:DISPlay:LANGuage?

- **Functional Description**

  Set the system language.

  CHINese: Chinese

  ENGLish: English

  GERMan: GERMan

- **Return Format**

  The query returns the system language: CHINese, ENGLish, or GERMan.

- **For Example**

:SYSTem:DISPlay:LANGuage CHINese	Set the system language to CHINese.
:SYSTem:DISPlay:LANGuage?	The query returns CHINese.

:SYSTem:LOCK

- **Command Format**

  :SYSTem:LOCK {{1|ON} | {0|OFF}}

  :SYSTem:LOCK?

- **Functional Description**

  Lock or unlock the keyboard and touchscreen.

  1|ON: Lock

  0|OFF: Unlock

- **Return Format**

  The query returns the lock status of keyboard and touchscreen. 0 represents that the keyboard is unlocked. 1 represents that the keyboard is unlocked.

- **For Example**

:SYSTem:LOCK ON	The keyboard and touchscreen are locked.
:SYSTem:LOCK OFF	The keyboard and touchscreen are unlocked.
:SYSTem:LOCK?	The query returns 0, it represents that the keyboard is unlocked.

:SYSTem:OUTPut:HDMI

- **Command Format**

  :SYSTem:OUTPut:HDMI {{1|ON} | {0|OFF}}

  :SYSTem:OUTPut:HDMI?

- **Functional Description**

  Set the HDMI output switch.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of HDMI output: 0 or 1.

- **For Example**

:SYSTem:OUTPut:HDMI ON	Turns on HDMI output.
:SYSTem:OUTPut:HDMI?	The query returns 1.

:SYSTem:PICture:FORMat

- **Command Format**

  :SYSTem:PICture:FORMat {BMP|PNG}

  :SYSTem:PICture:FORMat?

- **Functional Description**

  Select the image format of the system.

- **Return Format**

  The query returns the image format of the system: BMP or PNG.

- **For Example**

:SYSTem:PICture:FORMat BMP	The image format of the system is BMP.
:SYSTem:PICture:FORMat?	The query returns in BMP format.

:SYSTem:PON:TYPE

- **Command Format**

  :SYSTem:PON:TYPE {MODE|LAST|USER}

  :SYSTem:PON:TYPE?

- **Functional Description**

  Set the power on parameter of the instrument.

  MODE: The default parameter

  LAST: The newest parameter

  USER: User's preset parameter

- **Return Format**

  The query returns the power-on parameter: MODE, LAST, or USER.

- **For Example**

  :SYSTem:PON:TYPE MODE The default parameter.

  :SYSTem:PON:TYPE? The query returns MODE.

:SYSTem:TIME

- **Command Format**

  :SYSTem:TIME <hhmmss>

  :SYSTem:TIME?

- **Functional Description**

  Set the time and the format is 24 hours with no colon mark. For example, set the time to 12 (hours) 12 (minutes) 12 (seconds), the parameter is 121212.

- **Return Format**

  The query returns the time.

- **For Example**

  :SYSTem:TIME? The query returns 1.30101.

  (13:1:1)

:SYSTem:VOLume

- **Command Format**

  :SYSTem:VOLume <integer>

  :SYSTem:VOLume?

- **Functional Description**

  Controls the level of system's volume.

  <integer>: A continuous integer, with the default unit in %. The value range is from 0 to 100.

- **Return Format**

  The query returns the level of system's volume. The unit is %.

- **For Example**

  :SYSTem:VOLume 30 Set the level of system's volume to 30%.

  :SYSTem:VOLume? The query returns 30.

KEY Command

:KEY:<key>

- **Command Format**

  :KEY:<key>

  :KEY:<key>:LOCK { {1 | ON} | {0 | OFF} }

  :KEY:<key>:LOCK?

  :KEY:<key>:LED?

- **Functional Description**

  Sets key function and its lock function. The definition and Functional Description refer to Appendix 1: <key> Table.

- **Return Format**

  The query returns the key lock status or LED status.

  LED status: 0 represents LED is extinguished, 1 represents LED is illuminated (green).

- **For Example**

  :KEY:FREQ Enter frequency setting menu.

  :KEY:LED? The query returns LED status, 0 represents that LED is extinguished.

:KEY:LED?

- **Command Format**

  :KEY:LED?

- **Functional Description**

  Query the status of all keys with indicator light.

- **Return Format**

  The query returns the status of all keys with indicator light. The query returns character sequence, every character represents the status of one key. Illuminated: ASCll '1'; Extinguished: ASCll '0'. There are three keys with indicator light, it is TG key, Single key, and Touch/Lock key in sequence. The query returns a 3-bit string which consists of '1' or '0'.

- **For Example,**

  The query returns ASCll 100 when TG key is illuminated, single key and Touch/Lock key is extinguished.

:KEY:LOCK?

- **Command Format**

  :KEY:LOCK?

- **Functional Description**

  Query the lock status of all keys.

- **Return Format**

  The query returns the lock status of all keys. The query returns character sequence, every character represents the lock status of one key. Lock: ASCII '1'; Unlock: ASCII '0'. Return the lock status according to the sequence of Appendix 1: <key> Table.

- **For Example**

  Total 38 keys, only lock the fourth, fifth key, the query returns ASCII,

  00011000000000000000000000000000000000

Functional Module Command

This command is used to set the spectrum analysis, EMI, and analog demodulation.

Spectrum analysis

CALCulate Command

:CALCulate:LLINe:ALL:DELete

- **Command Format**

  :CALCulate:LLINe:ALL:DELete

- **Functional Description**

  Delete all the limit value data.

- **Return Format**

  No return value.

- **For Example,**

  :CALCulate:LLINe:ALL:DELete Delete all the data of the limit value.

:CALCulate:LLINe:SELect

- **Command Format**

  :CALCulate:LLINe:SELect <integer>

  :CALCulate:LLINe:SELect?

- **Functional Description**

  Select a currently limit value from the sequence of the limit value.

  <integer>: Serial number of the limit value, a continuous integer with a value range of 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to

Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the current serial number of limit value.

- **For Example**

:CALCulate:LLINe:SELect 1	Select limit value 1 as the current limit value.
:CALCulate:LLINe:SELect?	The query returns 1.

:CALCulate:LLINe<n>:BUILd

- **Command Format**

  :CALCulate:LLINe<n>:BUILd {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6}

- **Functional Description**

  Derive the trace from the specified limited data.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value.

  TRACE1 - TRACE6: Corresponding to trace 1 to trace 6.

  The limit values and trace values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:LLINe1:BUILd TRACE1	The limit value 1 data is derived from trace 1.

:CALCulate:LLINe<n>:COPY

- **Command Format**

  :CALCulate:LLINe<n>:COPY {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6}

- **Functional Description**

  Copy the specified limit value to a different limit value. The limit value cannot be copied to itself.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value.

  LLINE1 - LLINE6: Corresponding to limit value 1 to limit value 6.

  The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:LLINe2:COPY LLINE1	Copy the data of limit value 1 to limit value 2.

:CALCulate:LLINe<n>:DATA

- **Command Format**

  :CALCulate:LLINe<n>:DATA {<freq>,<ampl>,<connect>,<freq>,<ampl>,<connect>,...}

  :CALCulate:LLINe<n>:DATA?

- **Functional Description**

  Edit the specified limit data, with {frequency, amplitude, connection attribute} as the basic unit to editing.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <freq>: A continuous real number with the unit in Hz.

  <ampl>: A continuous real number with the default unit in dBm.

  <connect>: Takes a value of 0 or 1. If the value is 1, it indicates that the current point is connected to the previous point to determine the limit line; if the value is 0, it indicates that the current point is not connected to the previous point (cut-off). The <connect> value of the first point is ignored; either 0 or 1 is acceptable, and the return value defaults to 1.

- **Return Format**

  The query returns the specified limit data, with {frequency, amplitude, connection attribute} as the basic unit for newline. The unit of frequency is Hz. The unit of amplitude is dBm. The connection attribute is 0 or 1.

- **For Example,**

  :CALCulate:LLINe1:DATA 10000000,-50,0,100000000,-60,1 Set the data for Limit 1.

  :CALCulate:LLINe1:DATA? The query returns limit 1 data.

:CALCulate:LLINe<n>:DELete

- **Command Format**

  :CALCulate:LLINe<n>:DELete

- **Functional Description**

  Delete the specified limit data.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:LLINe1:DELete Delete the data for Limit 1.

:CALCulate:LLINe<n>:DISPlay

- **Command Format**

  :CALCulate:LLINe<n>:DISPlay {{1|ON} | {0|OFF}}

  :CALCulate:LLINe<n>:DISPlay?

- **Functional Description**

  Set the display switch for the limit value to on or off.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  1|ON: The display is opened.

  0|OFF: The display is closed.

- **Return Format**

  The query returns the display switch status of the limit value: 0 or 1.

- **For Example,**

  :CALCulate:LLINe1:DISPlay ON Limit 1 is displayed.

  :CALCulate:LLINe1:DISPlay? The query returns 1.

:CALCulate:LLINe<n>:MARGin

- **Command Format**

  :CALCulate:LLINe<n>:MARGin <real>

  :CALCulate:LLINe<n>:MARGin?

- **Functional Description**

  Set the margin for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <real>: A continuous real number with the default unit in dB. The range is from -40 dB to 40 dB.

- **Return Format**

  The query returns the margin of the specified limit value in scientific notation, with the unit in dB.

- **For Example**

  :CALCulate:LLINe1:MARGin 20 Set the margin for limit value 1 to 20 dB.

  :CALCulate:LLINe1:MARGin? The query returns 2.000000e+01.

:CALCulate:LLINe<n>:MARGin:STATe

- **Command Format**

  :CALCulate:LLINe<n>:MARGin:STATe {{1|ON} | {0|OFF}}

  :CALCulate:LLINe<n>:MARGin:STATe?

- **Functional Description**

  Set the margin switch for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the margin switch status of the specified limit value: 0 or 1.

- **For Example**

:CALCulate:LLINe1:MARGin:STATe ON	Turn on the margin for Limit 1.
:CALCulate:LLINe1:MARGin:STATe?	The query returns 1.

:CALCulate:LLINe<n>:OFFSet:UPDate

- **Command Format**

  :CALCulate:LLINe<n>:OFFSet:UPDate

- **Functional Description**

  Set the application offset for the limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:LLINe1:OFFSet:UPDate Set the application offset for Limit 1.

:CALCulate:LLINe<n>:OFFSet:X

- **Command Format**

  :CALCulate:LLINe<n>:OFFSet:X <freq>

  :CALCulate:LLINe<n>:OFFSet:X?

- **Functional Description**

Set the X-axis offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: A continuous real number, with the default unit in Hz.

- **Return Format**

The query returns the X-axis offset of the specified limit value in scientific notation, with the default unit in Hz.

- **For Example**

:CALCulate:LLINe1:OFFSet:X 10000000	Set the X-axis offset for limit 1 to 10 MHz.
:CALCulate:LLINe1:OFFSet:X?	The query returns 1.000000e+07.

:CALCulate:LLINe<n>:OFFSet:Y

- **Command Format**

:CALCulate:LLINe<n>:OFFSet:Y <real>

:CALCulate:LLINe<n>:OFFSet:Y?

- **Functional Description**

Set the Y-axis offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers: Refer to Appendix 3 Parameter Table for Each Model.

<real>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the Y-axis offset of the specified limit value in scientific notation, with the unit in dB.

- **For Example,**

:CALCulate:LLINe1:OFFSet:Y 5	Set the Y-axis offset for the limit value to 5 dB.
:CALCulate:LLINe1:OFFSet:Y?	The query returns 5.000000e+00.

:CALCulate:LLINe<n>:TRACe

- **Command Format**

:CALCulate:LLINe<n>:TRACe <integer>

:CALCulate:LLINe<n>:TRACe?

- **Functional Description**

Select the test trace for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value.

<integer>: Trace serial number, a continuous integer with a value range of 1 to the maximum trace value.

The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the current serial number of the limit value.

- **For Example**

:CALCulate:LLINe1:TRACe 1	Select trace 1 as the test trace for limit 1.
:CALCulate:LLINe1:TRACe?	The query returns 1.

:CALCulate:LLINe<n>:TYPE

- **Command Format**

:CALCulate:LLINe<n>:TYPE {UPPer|LOWer}

:CALCulate:LLINe<n>:TYPE?

- **Functional Description**

Select the type for the limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

UPPer: Higher

LOWer: Lower

- **Return Format**

The query returns the specified type of the limit value: UPPer or LOWer.

- **For Example,**

:CALCulate:LLINe1:TYPE UPPer	Set the type for limit 1 to UPPer.
:CALCulate:LLINe1:TYPE?	The query returns UPPer.

:CALCulate:LIMit<n>:CLEar

- **Command Format**

:CALCulate:LIMit<n>:CLEar

- **Functional Description**

Clear the data of the specified limit value.

- **Return Format**

No return value.

- **For Example**

:CALCulate:LIMit1:CLEar Clear the data of limit 1.

:CALCulate:LIMit<n>:CONTrol[:DATA]

- **Command Format**

:CALCulate:LIMit<n>:CONTrol[:DATA] <freq>,<freq>,<freq>,...

:CALCulate:LIMit<n>:CONTrol[:DATA]?

- **Functional Description**

Set the X-axis data to the specified limit.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: Frequency value with the default unit in Hz.

- **Return Format**

The query returns the X-axis data in scientific notation for the specified limit, with the unit in Hz.

- **For Example**

:CALC:LIMit1:CONT 10,20,30,40 Set the X-axis data for Limit 1.

:CALC:LIMit1:CONT? The query returns the X-axis data of Limit 1.

:CALCulate:LIMit<n>:CONTrol:POINts

- **Command Format**

:CALCulate:LIMit<n>:CONTrol:POINts?

- **Functional Description**

Set the X-axis data number for the specified limit.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the X-axis data number of the specified limit.

- **For Example**

:CALC:LIMit1:CONT:POIN? The query returns 4.

:CALCulate:LIMit<n>:LOWer[:DATA]

- **Command Format**

:CALCulate:LIMit<n>:LOWer[:DATA] <ampl>,<ampl>,<ampl>,...

:CALCulate:LIMit<n>:LOWer[:DATA]?

- **Functional Description**

  Set the Y-axis data of the lower limit type for the specified limit.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <ampl>: Amplitude value with the default unit in dBm.

- **Return Format**

  The query returns the Y-axis data in scientific notation for the lower limit type at the specified limit. The unit is dBm.

- **For Example,**

  :CALC:LIMit1:LOW -10,-10,-10,-10 Set the Y-axis data for Limit 1.

  :CALC:LIMit1:LOW? The query returns the Y-axis data of the lower type for Limit 1.

:CALCulate:LIMit<n>:LOWer:POINts

- **Command Format**

  :CALCulate:LIMit<n>:LOWer:POINts?

- **Functional Description**

  Set the Y-axis point of the lower limit type for the specified limit.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the Y-axis point of the lower limit type for the specified limit.

- **For Example**

  :CALC:LIMit1:LOW:POIN? The query returns 4.

:CALCulate:LIMit<n>:UPPer[:DATA]

- **Command Format**

  :CALCulate:LIMit<n>:UPPer[:DATA] <ampl>,<ampl>,<ampl>,...

  :CALCulate:LIMit<n>:UPPer[:DATA]?

- **Functional Description**

  Set the Y-axis data of the upper limit type for the specified limit.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter

Table for Each Model.

<ampl>: Amplitude value with the default unit in dBm.

- **Return Format**

The query returns the Y-axis data in scientific notation for the upper limit type at the specified limit. The unit is dBm.

- **For Example**

:CALC:LIMit1:UPP -10,-10,-10,-10 Set the X-axis data for Limit 1.

:CALC:LIMit1:UPP? The query returns the Y-axis data of the upper limit type for Limit 1.

:CALCulate:LIMit<n>:UPPer:POINts

- **Command Format**

:CALCulate:LIMit<n>:UPPer:POINts?

- **Functional Description**

Set the Y-axis point of the upper limit type for the specified limit.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the Y-axis point of the upper limit type for the specified limit.

- **For Example,**

:CALC:LIMit1:UPP:POIN? The query returns 4.

:CALCulate:MARKer:AOFF

- **Command Format**

:CALCulate:MARKer:AOFF

- **Functional Description**

Turn off all the markers.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer:AOFF Turn off all the markers.

:CALCulate:MARKer:FCOunt:GATetime

- **Command Format**

:CALCulate:MARKer:FCOunt:GATetime <time>

:CALCulate:MARKer:FCOunt:GATetime?

- **Functional Description**

  Set the gate time of frequency meter.

  <time>: A continuous real number. The default unit is seconds (s), with a value range of 1 μs to 500 ms.

- **Return Format**

  The query returns the gate time of frequency meter in scientific notation, with the unit in second(s).

- **For Example**

  :CALCulate:MARKer:FCOunt:GATetime 0.1 Set the gate time of frequency meter to 100 ms.

  :CALCulate:MARKer:FCOunt:GATetime? The query returns 1.000000e-01.

:CALCulate:MARKer:FCOunt:GATetime:AUTO

- **Command Format**

  :CALCulate:MARKer:FCOunt:GATetime:AUTO {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:FCOunt:GATetime:AUTO?

- **Functional Description**

  Switch the gate time of frequency meter to automatic or manual mode.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of gate time in automatic mode: 0 or 1.

- **For Example**

  :CALCulate:MARKer:FCOunt:GATetime:AUTO ON

  Switch the gate time of frequency meter to automatic (AUTO) mode.

  :CALCulate:MARKer:FCOunt:GATetime:AUTO? The query returns 1.

:CALCulate:MARKer:FCOunt[:STATe]

- **Command Format**

  :CALCulate:MARKer:FCOunt[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:FCOunt[:STATe]?

- **Functional Description**

  The switch of frequency meter.

- **Return Format**

  The query returns the switch status of frequency meter: 0 or 1.

- **For Example**

:CALCulate:MARKer:FCOunt ON	Turn on frequency meter.
:CALCulate:MARKer:FCOunt?	The query returns 1.

:CALCulate:MARKer:FCOunt:X

- **Command Format**

  :CALCulate:MARKer:FCOunt:X?

- **Functional Description**

  Query the measured result of frequency meter.

- **Return Format**

  The query returns frequency meter in scientific notation, with the unit in Hz.

- **For Example**

:CALCulate:MARKer:FCOunt:X?	The query returns 1.000000e+09.

:CALCulate:MARKer:PEAK:EXCursion

- **Command Format**

  :CALCulate:MARKer:PEAK:EXCursion <ampl>

  :CALCulate:MARKer:PEAK:EXCursion?

- **Functional Description**

  Set the excursion of peak.

  <ampl>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the peak shift in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:MARKer:PEAK:EXCursion 20	Set the peak shift to 20 dB.
:CALCulate:MARKer:PEAK:EXCursion?	The query returns 2.000000e+01.

:CALCulate:MARKer:PEAK:EXCursion:STATe

- **Command Format**

  :CALCulate:MARKer:PEAK:EXCursion:STATe {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:EXCursion:STATe?

- **Functional Description**

  Automatically/manually switch peak shift.

  1|ON: Automatically switch the peak shift.

  0|OFF: Manually witch the peak shift.

- **Return Format**

  The query returns the status of automatic switch the peak shift: 0 or 1.

- **For Example,**

  :CALCulate:MARKer:PEAK:EXCursion:STATe ON Select automatic switch the peak shift.

  :CALCulate:MARKer:PEAK:EXCursion:STATe? The query returns 1.

:CALCulate:MARKer:PEAK:TABLe:STATe

- **Command Format**

  :CALCulate:MARKer:PEAK:TABLe:STATe {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:TABLe:STATe?

- **Functional Description**

  Set the display switch for the peak list.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display switch status of peak list: 0 or 1.

- **For Example**

  :CALCulate:MARKer:PEAK:TABLe:STATe ON The peak list is displayed.

  :CALCulate:MARKer:PEAK:TABLe:STATe? The query returns 1.

:CALCulate:MARKer:PEAK:THReshold

- **Command Format**

  :CALCulate:MARKer:PEAK:THReshold <ampl>

  :CALCulate:MARKer:PEAK:THReshold?

- **Functional Description**

  Set the threshold for the peak.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the threshold of peak in scientific notation, with the unit in dBm.

- **For Example**

  :CALCulate:MARKer:PEAK:THReshold -20 Set the threshold gate for the peak to -20 dBm.

  :CALCulate:MARKer:PEAK:THReshold? The query returns -2.000000e+01.

:CALCulate:MARKer:PEAK:THReshold:LINE[:STATe]

- **Command Format**

:CALCulate:MARKer:PEAK:THReshold:LINE[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer:PEAK:THReshold:LINE[:STATe]?

- **Functional Description**

Set the display switch of threshold line for the peak.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the display switch of threshold line: 0 or 1.

- **For Example**

:CALCulate:MARKer:PEAK:THReshold:LINE ON The threshold line of the peak is displayed.

:CALCulate:MARKer:PEAK:THReshold:LINE? The query returns 1.

:CALCulate:MARKer:PEAK:THReshold:STATe

- **Command Format**

:CALCulate:MARKer:PEAK:THReshold:STATe {{1|ON} | {0|OFF}}

:CALCulate:MARKer:PEAK:THReshold:STATe?

- **Functional Description**

Automatically/manually switch the threshold of peak

1|ON: Automatically switch the threshold of peak.

0|OFF: Manually witch the threshold of peak.

- **Return Format**

The query returns the status of automatic switch the threshold of peak: 0 or 1.

- **For Example**

:CALCulate:MARKer:PEAK:THReshold:STATe ON

Automatically switch the threshold of peak.

:CALCulate:MARKer:PEAK:THReshold:STATe? The query returns 1.

:CALCulate:MARKer:SELect

- **Command Format**

:CALCulate:MARKer:SELect <integer>

:CALCulate:MARKer:SELect?

- **Functional Description**

Select a marker as the current marker from the sequence of the marker.

<integer>: The marker sequence, a continuous integer with a value range of 1 to 10.

- **Return Format**

The query returns the current serial number of the marker. The value range is 1 to 10.

- **For Example**

:CALCulate:MARKer:SELect 1	Select marker 1 as the current marker.
:CALCulate:MARKer:SELect?	The query returns 1.

:CALCulate:MARKer:TABLe[:STATe]

- **Command Format**

:CALCulate:MARKer:TABLe[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer:TABLe[:STATe]?

- **Functional Description**

The display switch of mark list.

- **Return Format**

The query returns the display switch status of mark list, 0 or 1.

- **For Example**

:CALCulate:MARKer:TABLe ON	Display the mark list.
:CALCulate:MARKer:TABLe?	The query returns 1.

:CALCulate:MARKer<n>:BANDwidth[:STATe]

- **Command Format**

:CALCulate:MARKer<n>:BANDwidth[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer<n>:BANDwidth[:STATe]?

- **Functional Description**

NDB bandwidth switch of the specified marker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

The query returns NDB bandwidth switch status of the specified marker: 0 or 1.

- **For Example**

:CALCulate:MARKer1:BANDwidth ON	Turn on NDB bandwidth of marker 1.
:CALCulate:MARKer1:BANDwidth?	The query returns 1.

:CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB

- **Command Format**

:CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB <real>

:CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB?

- **Functional Description**

Set NDB point of the specified marker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

<real>: A continuous real number with the default unit in dB. The value range is from -0.01 to -140.

- **Return Format**

The query returns NDB value of the specified marker in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:MARKer1:BANDwidth:NDB -3	NDB bandwidth of marker 1 is -3 dB.
:CALCulate:MARKer1:BANDwidth:NDB?	The query returns -3.000000e+00.

:CALCulate:MARKer<n>:BANDwidth|BWIDth:RESult?

- **Command Format**

:CALCulate:MARKer<n>:BANDwidth|BWIDth:RESult?

- **Functional Description**

Query the measured result of NDB bandwidth for the specified marker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

The query returns the measured result of NDB bandwidth for the specified marker, with the unit in Hz.

- **For Example**

:CALCulate:MARKer1:BANDwidth:RESult?

The query returns NDB measured result of marker 1.

:CALCulate:MARKer<n>:CPSearch[:STATe]

- **Command Format**

:CALCulate:MARKer<n>:CPSearch[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer<n>:CPSearch[:STATe]?

- **Functional Description**

Switch the continuous peak search for the specified marker to on or off.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the switch status of continuous peak search for the specified marker: 0 or 1.

- **For Example**

:CALCulate:MARKer1:CPSearch ON	Turn on continuous peak search for Marker 1.
:CALCulate:MARKer1:CPSearch?	The query returns 1.

:CALCulate:MARKer<n>:FUNCtion

- **Command Format**

  :CALCulate:MARKer<n>:FUNCtion {OFF|NOISe|BPOWer|BDENsity}

  :CALCulate:MARKer<n>:FUNCtion?

- **Functional Description**

  Select the marker function of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  NOISe: Mark noise

  BPOWer: in-band power

  BDENsity: in-band density

- **Return Format**

  The query returns the marker function of the specified marker: OFF, NOISe, BPOWer, or BDENsity.

- **For Example**

:CALCulate:MARKer1:FUNCtion NOISe	Select the marker function of marker 1 to NOISe.
:CALCulate:MARKer1:FUNCtion?	The query returns NOISe.

:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT

- **Command Format**

  :CALCulate:MARKer<n>:FUNCtion:BAND:LEFT <freq>|<time>

  :CALCulate:MARKer<n>:FUNCtion:BAND:LEFT?

- **Functional Description**

  Set the left-edge bandwidth for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  <freq>: Frequency can be set when X-axis scale type is frequency or reversing time. The default unit is Hz.

  <time>: Time can be set when X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

  The query returns the left-edge bandwidth of the specified marker in scientific notation.

- **For Example**

  :CALCulate:MARKer1:FUNCtion:BAND:LEFT 5MHz

Set the left-edge bandwidth of marker 1 to 5 MHz.

:CALCulate:MARKer1:FUNCtion:BAND:LEFT? The query returns 5.000000e+06.

:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt

- **Command Format**

  :CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt <freq>|<time>

  :CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt?

- **Functional Description**

  Set the right-edge bandwidth for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  <freq>: Frequency can be set when X-axis scale type is frequency or reversing time. The default unit is Hz.

  <time>: Time can be set when X-axis scale type is X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

  The query returns the right-edge bandwidth of the specified marker in scientific notation.

- **For Example**

  :CALCulate:MARKer1:FUNCtion:BAND:RIGHt 5MHz

  Set the right-edge bandwidth of marker 1 to 5 MHz.

  :CALCulate:MARKer1:FUNCtion:BAND:RIGHt? The query returns 5.000000e+06.

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN

- **Command Format**

  :CALCulate:MARKer<n>:FUNCtion:BAND:SPAN <freq>|<time>

  :CALCulate:MARKer<n>:FUNCtion:BAND:SPAN?

- **Functional Description**

  Set the marker bandwidth for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  <freq>: Frequency can be set when X-axis scale type is frequency or reversing time, the default unit is Hz.

  <time>: Time can be set when X-axis scale type is X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

  The query returns the marker bandwidth of the specified marker in scientific notation.

- **For Example**

:CALCulate:MARKer1:FUNCtion:BAND:SPAN 10MHz

Set the marker bandwidth of marker 1 to 10 MHz.

:CALCulate:MARKer1:FUNCtion:BAND:SPAN? The query returns 1.000000e+07.

:CALCulate:MARKer<n>:LINes[:STATe]

- **Command Format**

  :CALCulate:MARKer<n>:LINes[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:LINes[:STATe]?

- **Functional Description**

  Set the marker line switch for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  The query returns the marker line switch status of the specified marker: 0 or 1.

- **For Example**

  :CALCulate:MARKer1:LINes ON Enable the marker line for Marker 1.

  :CALCulate:MARKer1:LINes? The query returns 1.

:CALCulate:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum:LEFT

  Execute a next-peak search to the left side for Marker 1.

:CALCulate:MARKer<n>:MAXimum[:MAX]

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum[:MAX]

- **Functional Description**

  Execute a peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum Execute peak search for Marker 1.

:CALCulate:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum:NEXT Execute a next-peak search for Marker 1.

:CALCulate:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum:RIGHt

  Execute a next-peak search to the right side for Marker 1.

:CALCulate:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:MARKer<n>:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:MINimum Execute a minimum peak search to the left side for Marker 1.

:CALCulate:MARKer<n>:MODE

- **Command Format**

:CALCulate:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

:CALCulate:MARKer<n>:MODE?

- **Functional Description**

Select the mode for the specified maker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

OFF: Disable the marker function

POSition: Normal mode

DELTa: Difference mode

FIXed: Fixed mode

- **Return Format**

The query returns the specified maker, <time>: Time OFF, POSition, DELTa, or FIXed.

- **For Example**

:CALCulate:MARKer1:MODE POSition Sets marker 1 to POSition.

:CALCulate:MARKer1:MODE? The query returns POSition.

:CALCulate:MARKer<n>:PTPeak

- **Command Format**

:CALCulate:MARKer<n>:PTPeak

- **Functional Description**

Execute a peak-to-peak search for the specified marker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:PTPeak Execute a peak-to-peak search for Marker 1.

:CALCulate:MARKer<n>:REFerence

- **Command Format**

:CALCulate:MARKer<n>:REFerence <integer>

:CALCulate:MARKer<n>:REFerence?

- **Functional Description**

  Select the reference marker for the specified marker. The reference marker cannot be set as its own reference.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  <integer>: Serial number of another marker, excluding the current one. The value range is 1 to 10.

- **Return Format**

  The query returns the reference marker of the specified marker.

- **For Example**

  :CALCulate:MARKer1:REFerence 2 Set mark 2 as the reference marker for Marker 1.

  :CALCulate:MARKer1:REFerence? The query returns 2.

:CALCulate:MARKer<n>[:SET]:CENTer

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:CENTer

- **Functional Description**

  Set the center frequency to the frequency of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:CENTer Set the center frequency to the frequency of marker 1.

:CALCulate:MARKer<n>[:SET]:RLEVel

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:RLEVel

- **Functional Description**

  Set the reference level to the amplitude of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:RLEVel Set the reference level to the amplitude of marker 1.

:CALCulate:MARKer<n>[:SET]:STARt

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:STARt

- **Functional Description**

  Set the start frequency to the frequency of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:STARt Set the start frequency to the frequency of marker 1.

:CALCulate:MARKer<n>[:SET]:STEP

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:STEP

- **Functional Description**

  Set the step frequency to the frequency of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:STEP Set the step frequency to the frequency of marker 1.

:CALCulate:MARKer<n>[:SET]:STOP

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:STOP

- **Functional Description**

  Set the stop frequency to the frequency of the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:STOP Set the stop frequency to the frequency of marker 1.

:CALCulate:MARKer<n>:TRACe

- **Command Format**

:CALCulate:MARKer<n>:TRACe <integer>

:CALCulate:MARKer<n>:TRACe?

- **Functional Description**

Select the trace for the specified maker.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

<integer>: Serial number of the trace, continuous integer, the value range is from 1 to trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the serial number of the trace for the specified maker.

- **For Example**

:CALCulate:MARKer1:TRACe 1	Set marker 1 to correspond to the trace.
:CALCulate:MARKer1:TRACe?	The query returns 1.

:CALCulate:MARKer<n>:X

- **Command Format**

:CALCulate:MARKer<n>:X <freq>|<time>

:CALCulate:MARKer<n>:X?

- **Functional Description**

Set the coordinate value of the X-axis for the specified maker. The data type is set according to the X-axis scale type.

<n>: Marker serial number, a continuous integer with a value range of 1 to 10.

<freq>: Frequency can be set when X-axis scale type is frequency or reversing time. The default unit is Hz.

<time>: Time can be set when X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

The query returns the coordinate value of the X-axis for the specified maker in scientific notation. When X-axis scale type is frequency or reversing time, the unit is Hz; when X-axis scale type is period or time, the unit is second (s).

- **For Example**

:CALCulate:MARKer1:X 1GHz	Set the coordinate value of the X-axis for marker 1 to 1 GHz.
:CALCulate:MARKer1:X?	The query returns 1.000000e+09.

:CALCulate:MARKer<n>:X:READout

- **Command Format**

  :CALCulate:MARKer<n>:X:READout {FREQuency|PERiod|TIME|ITIMe}

  :CALCulate:MARKer<n>:X:READout?

- **Functional Description**

  Set the X-axis scale type for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  FREQuency: Frequency

  PERiod: Period

  TIME: Time

  ITIMe: Reversing time

- **Return Format**

  The query returns the X-axis scale type of the specified marker: FREQuency, PERiod, TIME, or ITIMe.

- **For Example**

  :CALCulate:MARKer1:X:READout FREQuency

  Set the X-axis scale type of marker 1 to FREQuency.

  :CALCulate:MARKer1:X:READout? The query returns FREQuency.

:CALCulate:MARKer<n>:X:READout:AUTO

- **Command Format**

  :CALCulate:MARKer<n>:X:READout:AUTO {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:X:READout:AUTO?

- **Functional Description**

  Switch the X-axis scale type of the specified marker to automatic or manual mode.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of X-axis scale type of the specified marker in automatic mode: 0 or 1.

- **For Example**

  :CALCulate:MARKer1:X:READout:AUTO ON

  Switch the X-axis scale type of marker 1 to automatic (AUTO) mode.

  :CALCulate:MARKer1:X:READout:AUTO? The query returns 1.

:CALCulate:MARKer<n>:Y

- **Command Format**

  :CALCulate:MARKer<n>:Y <ampl>

  :CALCulate:MARKer<n>:Y?

- **Functional Description**

  Set the amplitude value for the specified maker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 10.

  <ampl>: Amplitude value of the marker, the default unit is dBm.

  When the marker function is disabled, the command :CALCulate:MARKer<n>:Y? query returns the amplitude value of the specified maker; When enabled, the command :CALCulate:MARKer<n>:Y? query returns the measured results of marker.

- **Return Format**

  The query returns the amplitude the specified marker and the measured results of marker in scientific notation. The unit of amplitude value is dBm; the unit of noise marker is dBm/Hz; the unit of in-band power is dBm; the unit of in-band is density dBm/Hz.

- **For Example**

:CALCulate:MARKer1:Y -50	Set the amplitude value of marker 1 to -50 dBm.
:CALCulate:MARKer1:Y?	The query returns -5.000000e+01.

:CALCulate:MATH

- **Command Format**

  :CALCulate:MATH {<trace>,<type>,<op1>,<op2>, <real>}

- **Functional Description**

  Set the mathematical operation for the trace.

  <trace>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6, to specify the mathematical operation for the trace.

  <type>: OFF|PDIFference|PSUM|LDIFference|LOFFset, OFF represents the mathematical operation is turned off; PDIFference represents power difference; PSUM represents power sum; LDIFference represents logarithm difference; LOFFset represents logarithm offset.

  <op1>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6, to specify the trace of operand A.

  <op2>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6, to specify the trace of operand B.

  <real>: Mathematical operation offset, the unit is dB, the value range is from -100 dB to 100 dB. It only valid when the mathematical operation is LDIFference and LOFFset.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MATH TRACE1,PDIFference,TRACE2,TRACE3,0

  Set power difference operation for trace 1, operand 1 represents Trace 2, operand 2 represents Trace 3, and the offset is 0 dB.

:CALCulate:NTData[:STATe]

- **Command Format**

  :CALCulate:NTData[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:NTData[:STATe]?

- **Functional Description**

  Set the switch of normalization for the track source.

  1|ON: Enable the normalization function

  0|OFF: Disable the normalization function

- **Return Format**

  The query returns the normalization switch status for track source: 0 or 1.

- **For Example**

  :CALCulate:NTData ON Turn on normalization.

  :CALCulate:NTData? The query returns 1.

:CALCulate:RMCal

- **Command Format**

  :CALCulate:RMCal

- **Functional Description**

  Calibrate the reflectance measurement.

  Return Format

  No return value.

- **For Example**

  :CALCulate:RMCal Calibrate the reflectance measurement.

:CALCulate:RMData[:STATe]

- **Command Format**

  :CALCulate:RMData[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:RMData[:STATe]?

- **Functional Descriptio**

  The switch of the reflectance measurement.

1|ON: ON

0|OFF: OFF

Return Format

The query returns the switch status of reflectance measurement: 0 or 1.

- **For Example**

:CALCulate:RMData ON	Turn on the reflectance measurement.
:CALCulate:RMData?	The query returns 1.

CALibration Command

:CALibration[:ALL]

- **Command Format**

  :CALibration[:ALL]

- **Functional Description**

  Apply the calibration function immediately.

- **Return Format**

  No return value.

- **For Example**

:CALibration	Apply the calibration function immediately.

:CALibration:AUTO

- **Command Format**

  :CALibration:AUTO {{1|ON} | {0|OFF}}

  :CALibration:AUTO?

- **Functional Description**

  Control the automatic calibration switch.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of automatic calibration: 0 or 1.

- **For Example**

:CALibration:AUTO ON	Turn on automatic calibration.
:CALibration:AUTO?	The query returns 1.

CONFigure Command

:CONFigure:ACPower

- **Command Format**

  :CONFigure:ACPower

- **Functional Description**

  Enter adjacent channel power measurement and preset measurement. The measurement parameter of adjacent channel power will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:ACPower Enter adjacent channel power.

:CONFigure:ACPower:NDEFault

- **Command Format**

  :CONFigure:ACPower:NDEFault

- **Functional Description**

  Enter adjacent channel power measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:ACPower:NDEFault Enter adjacent channel power.

:CONFigure:CHPower

- **Command Format**

  :CONFigure:CHPower

- **Functional Description**

  Enter channel power measurement and preset measurement. The measurement parameter of the channel power will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:CHPower Enter the channel power.

:CONFigure:CHPower:NDEFault

- **Command Format**

  :CONFigure:CHPower:NDEFault

- **Functional Description**

  Enter the channel power measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:CHPower:NDEFault Enter the channel power measurement.

:CONFigure:CNRatio

- **Command Format**

  :CONFigure:CNRatio

- **Functional Description**

  Enter the carrier to noise ratio test and preset measurement. The measurement parameter of carrier to noise ratio will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:CNRatio Enter the carrier to noise ratio measurement.

:CONFigure:CNRatio:NDEFault

- **Command Format**

  :CONFigure:CNRatio:NDEFault

- **Functional Description**

  Enter the carrier to noise ratio measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:CNRatio:NDEFault Enter the carrier to noise ratio measurement.

:CONFigure:HARMonics

- **Command Format**

  :CONFigure:HARMonics

- **Functional Description**

Enter the harmonic wave measurement and preset measurement. The measurement parameter of harmonic wave test will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:HARMonics Enter the harmonic wave measurement.

:CONFigure:HARMonics:NDEFault

- **Command Format**

  :CONFigure:HARMonics:NDEFault

- **Functional Description**

  Enter the harmonic wave measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:HARMonics:NDEFault Enter the harmonic wave measurement.

:CONFigure:OBWidth

- **Command Format**

  :CONFigure:OBWidth

- **Functional Description**

  Enter the occupied bandwidth measurement and preset measurement. The measurement parameter of the occupied bandwidth will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:OBWidth Enter the occupied bandwidth measurement.

:CONFigure:OBWidth:NDEFault

- **Command Format**

  :CONFigure:OBWidth:NDEFault

- **Functional Description**

  Enter the occupied bandwidth measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:OBWidth:NDEFault Enter the occupied bandwidth measurement.

:CONFigure:SANalyzer

- **Command Format**

  :CONFigure:SANalyzer

- **Functional Description**

  Enter the spectrum analysis function.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:SANalyzer Enter the spectrum analysis function.

:CONFigure:SPECtrogram

- **Command Format**

  :CONFigure:SPECtrogram

- **Functional Description**

  Enter the spectrum monitoring measurement and preset measurement. The measurement parameter of spectrum monitoring will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:SPECtrogram Enter the spectrum monitoring measurement.

:CONFigure:SPECtrogram:NDEFault

- **Command Format**

  :CONFigure:SPECtrogram:NDEFault

- **Functional Description**

  Enter the spectrum monitoring measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:SPECtrogram:NDEFault Enter the spectrum monitoring measurement.

:CONFigure:TOI

- **Command Format**

  :CONFigure:TOI

- **Functional Description**

  Enter the third-order intercept point measurement and preset measurement. The measurement parameter of third-order intercept point will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:TOI Enter the third-order intercept point measurement.

:CONFigure:TOI:NDEFault

- **Command Format**

  :CONFigure:TOI:DEFault

- **Functional Description**

  Enter the third-order intercept point measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:TOI:DEFault Enter the third-order intercept point measurement.

:CONFigure:TPOWer

- **Command Format**

  :CONFigure:TPOWer

- **Functional Description**

  Enter the time domain power measurement and preset measurement. The measurement parameter of time domain power will restore to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:TPOWer Enter the time domain power measurement.

:CONFigure:TPOWer:NDEFault

- **Command Format**

  :CONFigure:TPOWer:NDEFault

- **Functional Description**

  Enter the time-domain measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:TPOWer:NDEFault Enter the time-domain measurement.

DISPlay Command

:DISPlay:DATA?

- **Command Format**

  :DISPlay:DATA?

- **Functional Description**

  Acquire the screen image.

- **Return Format**

  The query returns the screen image.

- **For Example**

  :DISPlay:DATA? Acquire the screen image.

:DISPlay:WINDow:TRACe:Y:DLINe

- **Command Format**

  :DISPlay:WINDow:TRACe:Y:DLINe <ampl>

  :DISPlay:WINDow:TRACe:Y:DLINe?

- **Functional Description**

  Set the amplitude value for display line.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the amplitude value of display line in scientific notation, with the unit in dBm.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:DLINe -40 Set the amplitude value of display line to -40 dBm.

  :DISPlay:WINDow:TRACe:Y:DLINe? The query returns -4.000000e+01.

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

- **Command Format**

  :DISPlay:WINDow:TRACe:Y:DLINe:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

  Switch the display line to on or off.

  1|ON: Display

  1|ON: Not display

- **Return Format**

  The query returns the switch status of display line: 0 or 1.

- **For Example**

:DISPlay:WINDow:TRACe:Y:DLINe:STATe ON	Turn on the display line.
:DISPlay:WINDow:TRACe:Y:DLINe:STATe?	The query returns 1.

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

- **Command Format**

  :DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel {ampl}

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

- **Functional Description**

  Set the reference level of normalization for trace source.

  <ampl>: A continuous integer with the unit in dB.

- **Return Format**

  The query returns the reference level of normalization for trace source, with the unit in dB.

- **For Example**

:DISPlay:WINDow:TRACe:Y:NRLevel 10	Set the reference level of normalization to 10 dB.
:DISPlay:WINDow:TRACe:Y:NRLevel?	The query returns 10.

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

- **Command Format**

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

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

- **Functional Description**

  Set the scale level for Y-axis.

  <real>: A discrete real number with a default unit in dB. The value range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the scale e level of Y-axis in scientific notation, with the unit in dB.

- **For Example**

:DISPlay:WINDow:TRACe:Y:PDIVision 1	Set the scale level of Y-axis to 1 dB.
:DISPlay:WINDow:TRACe:Y:PDIVision?	The query returns 1.000000e+00.

:DISPlay:WINDow[1]:TRACe:Y[:SCALe]:REFLevel

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:REFLevel {ampl}

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:REFLevel?

- **Functional Description**

  The reference level of reflectance measurement.

  <ampl>: A continuous integer with the unit in dB.

- **Return Format**

  The query returns the reference level of reflectance measurement, with the unit in dB.

- **For Example**

:DISPlay:WINDow:TRACe:Y:REFLevel 10	The reference level of reflectance is 10 dB.
:DISPlay:WINDow:TRACe:Y:REFLevel?	The query returns 10.

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

- **Command Format**

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

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

- **Functional Description**

  Set the reference level.

  <real>: A continuous real number. The default unit is dBm.

- **Return Format**

  The query returns the reference level in scientific notation, with the unit in dBm.

- **For Example**

:DISPlay:WINDow:TRACe:Y:RLEVel -10	Set the reference level to -10 dBm.
:DISPlay:WINDow:TRACe:Y:RLEVel?	The query returns -1.000000e+01.

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

- **Command Format**

  :DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet <real>

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

- **Functional Description**

Set the offset of the reference level.

<real>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the offset of the reference level in scientific notation, with the unit in dB.

- **For Example**

:DISPlay:WINDow:TRACe:Y:RLEVel:OFFSet 5	Set the offset of the reference level to 5 dB.
:DISPlay:WINDow:TRACe:Y:RLEVel:OFFSet?	The query returns 5.000000e+00.

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

- **Command Format**

:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing {LOGarithmic|LINear}

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

- **Functional Description**

Select the display scale type for the Y-axis.

LOGarithmic: Logarithmic

LINear: Linear

- **Return Format**

The query returns the display scale type of Y-axis: LOGarithmic or LINear.

- **For Example**

:DISPlay:WINDow:TRACe:Y:SPACing LOGarithmic

The display scale type of Y-axis is LOGarithmic.

:DISPlay:WINDow:TRACe:Y:SPACing?	The query returns LOGarithmic.

FETCh Command

:FETCh:ACPower<n>?

- **Command Format**

:FETCh:ACPower<n>?

- **Functional Description**

Query the results or the trace data for the ACPR (Adjacent Channel Power Ratio) measurement.

<n>: When n is 1, the query returns three ACPR (Adjacent Channel Power Ratio) measurement results: the primary channel power, the previous channel power relative to the primary channel power, and the next channel power relative to the primary channel power. The unit for channel power is dBm, and the unit for relative channel power is dBc.

When n is 2, the query returns five ACPR (Adjacent Channel Power Ratio) measurement results:

the primary channel power, the previous channel power relative to the primary channel power, the previous channel power, the next channel power relative to the primary channel power, and the next channel power. The unit for channel power is dBm, and the unit for relative channel power is dBc.

When n is 4, the query returns the trace data of ACPR (Adjacent Channel Power Ratio). The data consists of power values in dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the results or the trace data of the ACPR (Adjacent Channel Power Ratio) measurement in scientific notation.

- **For Example**

  :FETCh:ACPower1?

  The query returns the measured results of the ACPR (Adjacent Channel Power Ratio).

  :FETCh:ACPower2?

  The query returns the measured results of the ACPR (Adjacent Channel Power Ratio).

  :FETCh:ACPower4?

  The query returns the trace data of the ACPR (Adjacent Channel Power Ratio).

:FETCh:ACPower:LOWer?

- **Command Format**

  :FETCh:ACPower:LOWer?

- **Functional Description**

  Query the previous channel power and the relative power of the previous channel of adjacent channel power measurement. The unit of the main channel power is dBm. The unit of relative power is dBc.

- **Return Format**

  The query returns the previous channel power in scientific notation.

- **For Example**

  :FETCh:ACPower:LOWer? The query returns the previous channel power.

:FETCh:ACPower:MAIN?

- **Command Format**

  :FETCh:ACPower:MAIN?

- **Functional Description**

  Query the main channel power of all adjacent channel power measurements.

- **Return Format**

  The query returns the main channel power in scientific notation.

- **For Example**

  :FETCh:ACPower:MAIN? The query returns the main channel power.

:FETCh:ACPower:UPPer?

- **Command Format**

  :FETCh:ACPower:UPPer?

- **Functional Description**

  Query the last channel power and the relative power of the last channel of adjacent channel power measurement. The unit of the main channel power is dBm. The unit of relative power is dBc.

- **Return Format**

  The query returns the last channel power in scientific notation.

- **For Example**

  :FETCh:ACPower:UPPer? The query returns the last channel power.

:FETCh:CHPower<n>?

- **Command Format**

  :FETCh:CHPower<n>?

- **Functional Description**

  Query the results or the trace data for the channel power measurement.

  <n>: When n is 1, the query returns the channel power in unit dBm and channel power density in unit dBm/Hz.

  When n is 2, the query returns the trace data of the channel power measurement. The data consists of power value in unit dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the results or the trace data of the channel power measurement in scientific notation.

- **For Example**

  :FETCh:CHPower1? The query returns the results of the channel power measurement.

  :FETCh:CHPower2? The query returns the trace data of the channel power measurement.

:FETCh:CHPower:CHPower?

- **Command Format**

  :FETCh:CHPower:CHPower?

- **Functional Description**

  Query the channel power for channel power measurement.

- **Return Format**

  The query returns the channel power in scientific notation, with the unit in dBm.

- **For Example**

  :FETCh:CHPower:CHPower? The query returns the channel power.

:FETCh:CHPower:DENSity?

- **Command Format**

  :FETCh:CHPower:DENSity?

- **Functional Description**

  Query the channel power density for channel power measurement.

- **Return Format**

  The query returns the channel power density in scientific notation, with the unit in dBm/Hz.

- **For Example**

  :FETCh:CHPower:DENSity? The query returns the channel power density.

:FETCh:CNRatio<n>?

- **Command Format**

  :FETCh:CNRatio<n>?

- **Functional Description**

  Query the results or the trace data for the CNR (Carrier to Noise Ratio) measurement.

  <n>:

  When n is 1, the query returns the measured results of the CNR: the load power, noise power, and CNR. The unit for the load power is dBm, and the unit for CNR (Carrier to Noise Ratio) is dB.

  When n is 2, the query returns the trace data of the CNR measurement. The data consists of power value in unit dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the results or the trace data for the CNR measurement in scientific notation.

- **For Example**

  :FETCh:CNRatio11? The query returns the results of the CNR measurement.

:FETCh:CNRatio12? The query returns the trace data of the CNR measurement.

:FETCh:CNRatio:CARRier?

- **Command Format**

  :FETCh:CNRatio:CARRier?

- **Functional Description**

  Query the load power of carrier to noise measurement.

- **Return Format**

  The query returns the load power of carrier to noise test in scientific notation, with the unit in dBm.

- **For Example**

  :FETCh:CNRatio:CARRier?

  The query returns the load power of carrier to noise measurement.

:FETCh:CNRatio:CNRatio?

- **Command Format**

  :FETCh:CNRatio:CNRatio?

- **Functional Description**

  Query the carrier to noise of carrier to noise measurement.

- **Return Format**

  The query returns the carrier to noise in scientific notation, with the unit in dB.

- **For Example**

  :FETCh:CNRatio:CNRatio? The query returns the carrier to noise.

:FETCh:CNRatio:NOISe?

- **Command Format**

  :FETCh:CNRatio:NOISe?

- **Functional Description**

  Query the noise power of carrier to noise measurement.

- **Return Format**

  The query returns the noise power of carrier to noise test in scientific notation, with the unit in dBm.

- **For Example**

  :FETCh:CNRatio:NOISe?

  The query returns the noise power of carrier to noise measurement.

:FETCh:HARMonics<n>?

- **Command Format**

  :FETCh:HARMonics<n>?

- **Functional Description**

  Query the results for the harmonic distortion measurement.

  <n>: When n is 1, the query returns the percentage value of the harmonic distortion.

  When n is 2, the query returns the dBc value of the harmonic distortion.

- **Return Format**

  The query returns the results of the harmonic distortion measurement in scientific notation.

- **For Example**

  :FETCh:HARMonics1?

  The query returns the percentage value of the harmonic distortion from the harmonic measurement.

  :FETCh:HARMonics2?

  The query returns the dBc value of the harmonic distortion from the harmonic measurement.

:FETCh:HARMonics:AMPLitude:ALL?

- **Command Format**

  :FETCh:HARMonics:AMPLitude:ALL?

- **Functional Description**

  Query the amplitude value of the previous 10th harmonic in the harmonic wave test, the harmonics are arranged in order from the fundamental wave to the 10th harmonic. The amplitude unit for the fundamental wave is dBm, while the amplitude unit for the Nth harmonic is dBc.

- **Return Format**

  The query returns all the amplitude value of harmonic wave in scientific notation.

- **For Example**

  :FETCh:HARMonics:AMPLitude:ALL?

  The query returns all the amplitude value of harmonic wave.

:FETCh:HARMonics:AMPLitude<n>?

- **Command Format**

  :FETCh:HARMonics:AMPLitude<n>?

- **Functional Description**

  Query the harmonic wave's amplitude value of the specified number of times for harmonic wave

test. The amplitude unit of the fundamental wave is dBm, other amplitude unit of Nth harmonic is dBc.

<n>: Harmonic number, an integer with a value range of 1 to 10. The 1st harmonic is the fundamental harmonic.

- **Return Format**

  The query returns the harmonic wave's amplitude value of the specified number of times in scientific notation.

- **For Example**

  :FETCh:HARMonics:AMPLitude2? The query returns the 2nd harmonic amplitude.

:FETCh:HARMonics:DISTortion?

- **Command Format**

  :FETCh:HARMonics:DISTortion?

- **Functional Description**

  Query the results of harmonic distortion for the harmonic measurement, including the percentage of harmonic distortion and the relative power of harmonic distortion. The unit of harmonic distortion is %. The unit of the relative power of harmonic distortion is dBc.

- **Return Format**

  The query returns the result of harmonic distortion in scientific notation.

- **For Example**

  :FETCh:HARMonics:DISTortion?

  The query returns the measured results of harmonic distortion.

:FETCh:HARMonics:FREQuency:ALL?

- **Command Format**

  :FETCh:HARMonics:FREQuency:ALL?

- **Functional Description**

  Query the harmonic wave's frequency value of the previous 10th harmonic for the harmonic wave test. They are arranged in order from the fundamental wave to the 10th harmonic. The unit is Hz.

- **Return Format**

  The query returns the specified number of times of harmonic wave's frequency value in scientific notation.

- **For Example**

  :FETCh:HARMonics:FREQuency:ALL?

Query all the frequency of the previous 10th harmonic.

:FETCh:HARMonics:FREQuency<n>?

- **Command Format**

  :FETCh:HARMonics:FREQuency<n>?

- **Functional Description**

  Query the specified number of times of harmonic wave's frequency for the harmonic wave test. The unit is Hz.

  <n>: Harmonic number, an integer with a value range of 1 to 10. The 1st harmonic is the fundamental harmonic.

- **Return Format**

  The query returns the specified number of times of harmonic wave's frequency in scientific notation.

- **For Example**

  :FETCh:HARMonics:FREQuency3? The query returns 3rd harmonic wave's frequency.

:FETCh:OBWidth<n>?

- **Command Format**

  :FETCh:OBWidth<n>?

- **Functional Description**

  Query the results or the trace data of the occupied bandwidth measurement.

  <n>: When n is 1, the query returns the results of the occupied bandwidth measurement, including occupied bandwidth frequency, total power, test set bandwidth, sweep point, resolution bandwidth, transmitted frequency error, and XdB. The power unit is dBm, and the units for frequency and sweep width are Hz.

  When n is 2, the query returns the trace data of the occupied bandwidth measurement. The data consists of power value in unit dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the results or the trace data of the occupied bandwidth measurement in scientific notation.

- **For Example**

  :FETCh:OBWidth1?

  The query returns the results of the occupied bandwidth measurement.

  :FETCh:OBWidth2?

The query returns the trace data of the occupied bandwidth measurement.

:FETCh:OBWidth:APOWr

- **Command Format**

  :FETCh:OBWidth:APOWr?

- **Functional Description**

  Query the total power for occupied bandwidth test.

- **Return Format**

  The query returns the total power of the occupied bandwidth in scientific notation, with the unit in dBm.

- **For Example**

  :FETCh:OBWidth:APOWr? The query returns the total power of the occupied bandwidth.

:FETCh:OBWidth:FERRor?

- **Command Format**

  :FETCh:OBWidth:FERRor?

- **Functional Description**

  Query the error of transmission frequency of the occupied bandwidth measurement.

- **Return Format**

  The query returns the error of transmission frequency in scientific notation, with the unit in Hz.

- **For Example**

  :FETCh:OBWidth:FERRor? The query returns the error of transmission frequency.

:FETCh:OBWidth:OBWidth?

- **Command Format**

  :FETCh:OBWidth:OBWidth?

- **Functional Description**

  Query the occupied bandwidth of the occupied bandwidth measurement.

- **Return Format**

  The query returns the occupied bandwidth in scientific notation, with the unit in Hz.

- **For Example**

  :FETCh:OBWidth:OBWidth? The query returns the occupied bandwidth.

:FETCh:OBWidth:XDB?

- **Command Format**

:FETCh:OBWidth:XDB?

- **Functional Description**

  Query the xdB bandwidth of the occupied bandwidth measurement.

- **Return Format**

  The query returns the xdB bandwidth of the occupied bandwidth measurement in scientific notation, with the unit in Hz.

- **For Example**

  :FETCh:OBWidth:XDB?

  The query returns the xdB bandwidth of the occupied bandwidth measurement.

:FETCh:PTABle?

- **Command Format**

  :FETCh:PTABle?

- **Functional Description**

  Query the data in the peak list under the frequency spectrum mode.

- **Return Format**

  The query returns the data in the peak list under the frequency spectrum mode in scientific notation. The peak list contains the serial number of peaks, the peak frequency in Hz and the peak amplitude in dBm.

- **For Example**

  :FETCh:PTABle? The query returns the data in the peak list.

:FETCh:TOI<n>?

- **Command Format**

  :FETCh:TOI<n>?

- **Functional Description**

  Query the results and and the trace data of the third-order intermodulation measurement.

  <n>: When n is 0, the query returns the trace data of the third-order intermodulation measurement. The data units are based on frequency and power, with frequency in Hz and power in dBm. The number of data units corresponds to the set sweep points, and the data are separated by commas.

  When n is 1, the query returns the six test results of the third-order intermodulation, including the worst output intercept power, the worst output intercept point frequency, the low output intercept power, the low output intercept point frequency, the high output intercept power, and the high output intercept point frequency. The power unit is dBm, and the frequency unit is Hz.

When n is 2, the query returns the thirteen test results of the third-order intermodulation, including the worst output intercept point frequency, the worst output intercept point power, the worst output intercept power, the low base frequency, the low base power, the high base frequency, the high base power, the low output intercept point frequency, the low output intercept point power, the low output intercept power, the high output intercept point power frequency, the high output intercept point power, and high output intercept power. The power unit is dBm, and the frequency unit is Hz.

- **Return Format**

  The query returns the results of the third-order intermodulation in scientific notation.

- **For Example**

  :FETCh:TOI0?

  The query returns the trace data of the third-order intermodulation measurement.

  :FETCh:TOI1?

  The query returns the results of the third-order intermodulation measurement.

  :FETCh:TOI2?

  The query returns the results of the third-order intermodulation measurement.

:FETCh:TOI:IP3?

- **Command Format**

  :FETCh:TOI:IP3?

- **Functional Description**

  Query the results of the third-order intermodulation measurement, including the coefficient of third-order intercept in dBc and the third-order intercept point in dBm.

- **Return Format**

  The query returns the measured results of third-order intercept in scientific notation.

- **For Example**

  :FETCh:TOI:IP3? The query returns the measured results of third-order intercept.

:FETCh:TOI:LOW:TONE

- **Command Format**

  :FETCh:TOI:LOW:TONE?

- **Functional Description**

  Query the frequency and amplitude of low fundamental frequency for third-order intermodulation measurement. The unit of frequency is Hz. The unit of amplitude is dBm.

- **Return Format**

The query returns the frequency and amplitude of low fundamental frequency in scientific notation.

- **For Example**

  :FETCh:TOI:LOW:TONE?

  The query returns the frequency and amplitude of low fundamental frequency.

:FETCh:TOI:LOW:TRD

- **Command Format**

  :FETCh:TOI:LOW:TRD?

  Functional Description

  Query the low third-order intercept result of third-order intermodulation measurement, which including frequency, the unit of frequency is Hz; amplitude, the unit of amplitude is dBm; the coefficient of low third-order intercept, the unit dBc; low third-order intercept point, the unit is dBm.

- **Return Format**

  The query returns the result of low third-order intercept in scientific notation.

- **For Example**

  :FETCh:TOI:LOW:TRD? The query returns the result of third-order intercept.

:FETCh:TOI:UP:TONE

- **Command Format**

  :FETCh:TOI:UP:TONE?

- **Functional Description**

  Query the frequency and amplitude of high fundamental frequency for third-order intermodulation measurement. The unit of frequency is Hz. The unit of amplitude is dBm.

- **Return Format**

  The query returns the frequency and amplitude of high fundamental frequency for third-order intermodulation measurement in scientific notation.

- **For Example**

  :FETCh:TOI:UP:TONE?

  The query returns the frequency and amplitude of high fundamental frequency.

:FETCh:TOI:UP:TRD

- **Command Format**

  :FETCh:TOI:UP:TRD?

- **Functional Description**

  Query the high third-order intercept result of third-order intermodulation measurement, which including frequency, the unit of frequency is Hz; amplitude, the unit of amplitude is dBm; the coefficient of high third-order intercept, the unit dBc; high third-order intercept point, the unit is dBm.

- **Return Format**

  The query returns the result of high third-order intercept in scientific notation.

- **For Example**

  :FETCh:TOI:UP:TRD? The query returns high third-order intercept.

FETCh:TPower<n>?

- **Command Format**

  :FETCh:TPower<n>?

- **Functional Description**

  Query the total power or the trace data for the time domain power measurement.

  <n>:

  When n is 1, the query returns the total power of the time domain power measurement in unit dBm.

  When n is 2, the query returns the trace data of the time domain power measurement. The data consists of power value in unit dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the total power or the trace data of the time domain power measurement in scientific notation.

- **For Example**

  :FETCh:TPower1? The query returns the results of the time domain power measurement.

  :FETCh:TPower2? The query returns he trace data of the time domain power measurement.

FORMat Command

:FORMat[:TRACe][:DATA]

- **Command Format**

  :FORMat[:TRACe][:DATA] {ASCII|REAL32|REAL64}

  :FORMat[:TRACe][:DATA]?

- **Functional Description**

Set the return format of trace data. The default format is ASCII.

ASCII: Character

REAL32: 32-bit binary system integer

REAL64: 64-bit binary system integer

- **Return Format**

The query returns the return format of trace data, ASC8, REAL32, or REAL64.

- **For Example**

:FORMat ASCII	Set the return format of trace data to ASCII.
:FORMat?	The query returns ASC8.

INITiate

:INITiate:CONTinuous

- **Command Format**

:INITiate:CONTinuous {{1|ON} | {0|OFF}}

:INITiate:CONTinuous?

- **Functional Description**

Switch between single sweep and continuous sweep mode.

1|ON: Continuous sweep mode

0|OFF: Single sweep mode

- **Return Format**

The query returns whether the mode is continuous sweep: 0 or 1.

- **For Example**

:INITiate:CONTinuous ON	Continuous sweep mode.
:INITiate:CONTinuous?	The query returns 1.

INPut Command

:INPut:MIXer?

- **Command Format**

:INPut:MIXer?

- **Functional Description**

Query the frequency reference of device, EXTernal or INTernal.

- **Return Format**

The query returns the frequency reference of device: EXTernal or INTernal.

- **For Example**

:INPut:MIXer?	The query returns INTernal.

MMEMory Command

:MMEMory:LOAD:CORRection

- **Command Format**

  :MMEMory:LOAD:CORRection {<integer>,<filename>}

- **Functional Description**

  Specify the corrections needed for loading the data.

  <integer>: Serial number of the correction; a continuous integer with a value range of 1 to 10.

  <filename>: The file name and the file suffix is ".corr".

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:CORRection 1, "test.corr" The correction 1 loads the data file test ".corr".

:MMEMory:LOAD:LIMit

- **Command Format**

  :MMEMory:LOAD:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

  Specify the limit values needed for loading the data.

  LLINE1-LLINE6: Corresponding to limit value 1 to limit value 6.

  <filename>: The file name and the file suffix is ".limit".

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:LIMit LLINE1,"test.limit" Limit value 1 loads the data file test ".limit".

:MMEMory:LOAD:STATe

- **Command Format**

  :MMEMory:LOAD:STATe <filename>

- **Functional Description**

  Load the status data of the register.

  <filename>: The file name and the file suffix is ".state".

- **Return Format**

No return value.

- **For Example**

:MMEMory:LOAD:STATe "test.state" Load the status data file of the register test ".state".

:MMEMory:LOAD:TRACe

- **Command Format**

:MMEMory:LOAD:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

Specific the trace to load the data.

TRACE1-TRACE6: Corresponding to trace 1 to trace 6.

<filename>: The file name and the file suffix is ".trace".

- **Return Format**

No return value.

- **For Example**

:MMEMory:LOAD:TRACe TRACE1,"test.trace" Trace 1 loads the data file test ".trace".

:MMEMory:STORe:CORRection

- **Command Format**

:MMEMory:STORe:CORRection {<integer>,<filename>}

- **Functional Description**

Save the specified correction data into the default catalogue.

<integer>: Serial number of the correction; a continuous integer with a value range of 1 to 10.

<filename>: The file name and the file suffix is ".corr"..

- **Return Format**

No return value.

- **For Example**

:MMEMory:STORe:CORRection 1,"test.corr"

Save the correction 1 data into the file test ".corr".

:MMEMory:STORe:LIMit

- **Command Format**

:MMEMory:STORe:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

Save the specified limit value data into the default catalogue.

LLINE1-LLINE6: Corresponding to limit value 1 to limit value 6.

<filename>: The file name and the file suffix is ".limit".

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:LIMit LLINE1,"test.limit"

  Save the data of limit value 1 data to the file test ".limit".

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

  Save the register status to the file.

  <filename>: The file name and the file suffix is ".state".

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:STATe "test.state" Save the register status to the file test ".state".

:MMEMory:STORe:TRACe

- **Command Format**

  :MMEMory:STORe:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Save the specified trace data into the file.

  TRACE1-TRACE6: Corresponding to trace 1 to trace 6.

  <filename>: The file name and the file suffix is ".trace"..

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:TRACe TRACE1,"test.trace"

  Save the data of trace 1 to the file test ".trace".

OUTPut Command

:OUTPut[:EXTernal][:STATe]

- **Command Format**

:OUTPut[:EXTernal][:STATe] {{1|ON} | {0|OFF}}

:OUTPut[:EXTernal][:STATe]?

- **Functional Description**

  The switch of trace source.

- **Return Format**

  The query returns the status of trace source: 0 or 1.

- **For Example**

  :OUTPut ON Enable the trace source.

  :OUTPut? The query returns 1.

READ Command

The subsystem commands :READ and :FETCh are used to acquire measured results. The key difference between these commands is that the :FETCh command immediately retrieves the measured result, while the :READ command initiates a measurement, waits for it to complete, and then returns the result. If the measurement time exceeds the timeout period, the :READ command may fail to return the results due to a timeout.

:READ:ACPower<n>?

- **Command Format**

  :READ:ACPower<n>?

- **Functional Description**

  Execute an ACPR (Adjacent Channel Power Ratio) measurement, and the query returns the measured results or the trace data.

  <n>: When n is 1, the query returns three ACPR (Adjacent Channel Power Ratio) measurement results: the primary channel power, the previous channel power relative to the primary channel power, and the next channel power relative to the primary channel power. The unit for channel power is dBm, and the unit for relative channel power is dBc.

  When n is 2, the query returns five ACPR (Adjacent Channel Power Ratio) measurement results: the primary channel power, the previous channel power relative to the primary channel power, the previous channel power, the next channel power relative to the primary channel power, and the next channel power. The unit for channel power is dBm, and the unit for relative channel power is dBc.

  When n is 4, The query returns the trace data for ACPR (Adjacent Channel Power Ratio). The data consists of power values in dBm, with the number of data points corresponding to the set

sweep points, separated by commas.

- **Return Format**

  The query returns the results or the trace data for the ACPR (Adjacent Channel Power Ratio) measurement in scientific notation.

- **For Example**

  :READ:ACPower1?

  The query returns the measured results of the ACPR (Adjacent Channel Power Ratio) measurement.

  :READ:ACPower2?

  The query returns the measured results of the ACPR (Adjacent Channel Power Ratio) measurement.

  :READ:ACPower4?

  The query returns the trace data of the ACPR (Adjacent Channel Power Ratio) measurement.

:READ:ACPower:MAIN?

- **Command Format**

  :READ:ACPower:MAIN?

- **Functional Description**

  Execute an adjacent channel power measurement and return the main channel power.

- **Return Format**

  The query returns the main channel power in scientific notation.

- **For Example**

  :READ:ACPower:MAIN? The query returns the main channel power.

:READ:ACPower:LOWer?

- **Command Format**

  :READ:ACPower:LOWer?

- **Functional Description**

  Execute an adjacent channel power measurement and return the previous channel power and the relative power of the previous channel. The unit of the main channel power is dBm. The unit of relative power is dBc.

- **Return Format**

  The query returns the previous channel power in scientific notation.

- **For Example**

  :READ:ACPower:LOWer? The query returns the previous channel power.

:READ:ACPower:UPPer?

- **Command Format**

  :READ:ACPower:UPPer?

- **Functional Description**

  Execute a adjacent channel power measurement and return the last channel power and the relative power of the last channel. The unit of the main channel power is dBm. The unit of relative power is dBc.

- **Return Format**

  The query returns the last channel power in scientific notation.

- **For Example**

  :READ:ACPower:UPPer? The query returns the last channel power.

:READ:CHPower<n>?

- **Command Format**

  :READ:CHPower<n>?

- **Functional Description**

  Execute a channel power measurement, and the query returns the measured results or the trace data.

  <n>: When n is 1, the query returns the channel power in dBm and the channel power density in dBm/Hz.

  When n is 2, the query returns the trace data of the channel power. The data consists of power values in dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the measured results or the trace data of the channel power measurement in scientific notation.

- **For Example**

  :READ:CHPower1? The query returns the measured results of the channel power.

  :READ:CHPower2? The query returns the trace data of the channel power.

:READ:CHPower:CHPower?

- **Command Format**

  :READ:CHPower:CHPower?

- **Functional Description**

  Execute a channel power measurement and return the channel power of the channel power

measurement. The unit is dBm.

- **Return Format**

  The query returns the channel power in scientific notation.

- **For Example**

  :READ:CHPower:CHPower? The query returns the channel power.

:READ:CHPower:DENSity?

- **Command Format**

  :READ:CHPower:DENSity?

- **Functional Description**

  Execute a channel power measurement and return the channel power density of the channel power measurement. The unit is dBm/Hz.

- **Return Format**

  The query returns the channel power density in scientific notation.

- **For Example**

  :READ:CHPower:DENSity? The query returns the channel power density.

:READ:CNRatio<n>?

- **Command Format**

  :READ:CNRatio<n>?

- **Functional Description**

  Execute a CNR (Carrier to Noise Ratio) measurement, and the query returns the measured results or the trace data.

  <n>: When n is 1, the query returns the results of CNR (Carrier to Noise Ratio) measurement, including the load power, noise power, and CNR. The unit for load power and noise power is dBm, and the unit for CNR is dB.

  When n is 2, the query returns the trace data of CNR (Carrier to Noise Ratio). The data consists of power values in dBm, with the number of data points corresponding to the set sweep points, separated by commas.

- **Return Format**

  The query returns the measured results or the trace data of the CNR (Carrier to Noise Ratio) measurement in scientific notation.

- **For Example**

  :READ:CNRatio1?

  The query returns the measured results of the CNR (Carrier to Noise Ratio).

:READ:CNRatio2?

The query returns the trace data of the CNR (Carrier to Noise Ratio).

:READ:CNRatio:CARRier?

- **Command Format**

  :READ:CNRatio:CARRier?

- **Functional Description**

  Execute a carrier to noise test and return the load power.

- **Return Format**

  The query returns the load power of the carrier to noise in scientific notation, with the unit in dBm.

- **For Example**

  :READ:CNRatio:CARRier? The query returns the load power of carrier to noise test.

:READ:CNRatio:CNRatio?

- **Command Format**

  :READ:CNRatio:CNRatio?

- **Functional Description**

  Execute a carrier to noise test and return the carrier to noise.

- **Return Format**

  The query returns the carrier to noise in scientific notation, with the unit in dB.

- **For Example**

  :READ:CNRatio:CNRatio? The query returns the carrier to noise.

:READ:CNRatio:NOISe?

- **Command Format**

  :READ:CNRatio:NOISe?

- **Functional Description**

  Execute a carrier to noise test and return the noise power.

- **Return Format**

  The query returns the noise power of the carrier to noise in scientific notation, with the unit in dBm.

- **For Example**

  :READ:CNRatio:NOISe? The query returns the noise power of the carrier to noise test.

:READ:HARMonics<n>?

- **Command Format**

  :READ:HARMonics<n>?

- **Functional Description**

  Execute a harmonic measurement, and the query returns the measured results of the harmonic distortion.

  <n>: When n is 1, the query returns the percentage of the harmonic distortion.

  When n is 2, the query returns the dBc of the harmonic distortion.

- **Return Format**

  The query returns the harmonic distortion results of the harmonic measurement in scientific notation.

- **For Example**

  :READ:HARMonics1?

  The query returns the percentage of the harmonic distortion for the harmonic measurement.

  :READ:HARMonics2?

  The query returns the dBc of the harmonic distortion for the harmonic measurement.

:READ:HARMonics:AMPLitude:ALL?

- **Command Format**

  :READ:HARMonics:AMPLitude:ALL?

- **Functional Description**

  Execute a harmonic wave test and return the amplitude value of the previous 10th harmonic. They are arranged in order from the fundamental wave to the 10th harmonic. The amplitude unit of the fundamental wave is dBm, other amplitude unit of Nth harmonic is dBc.

- **Return Format**

  The query returns all the amplitude value of harmonic wave in scientific notation.

- **For Example**

  :READ:HARMonics:AMPLitude:ALL?

  The query returns all the amplitude value of harmonic wave.

:READ:HARMonics:AMPLitude<n>?

- **Command Format**

  :READ:HARMonics:AMPLitude<n>?

- **Functional Description**

  Execute a harmonic wave test and return the harmonic wave's amplitude value of the specified

number of times. The amplitude unit of the fundamental wave is dBm, other amplitude unit of Nth harmonic is dBc.

<n>: Harmonic number, an integer with a value range of 1 to 10. The 1st harmonic is the fundamental harmonic.

- **Return Format**

  The query returns the harmonic wave's amplitude value of the specified number of times in scientific notation.

- **For Example**

  :READ:HARMonics:AMPLitude2? The query returns 2.th harmonic wave's amplitude value.

:READ:HARMonics:DISTortion?

- **Command Format**

  :READ:HARMonics:DISTortion?

- **Functional Description**

  Execute a harmonic wave test and return the result of harmonic distortion, which including the percentage of harmonic distortion and the relative power of harmonic distortion. The unit of harmonic distortion is %. The unit of the relative power of harmonic distortion is dBc.

- **Return Format**

  The query returns the result of harmonic distortion in scientific notation.

- **For Example**

  :READ:HARMonics:DISTortion? The query returns the result of harmonic distortion.

:READ:HARMonics:FREQuency:ALL?

- **Command Format**

  :READ:HARMonics:FREQuency:ALL?

- **Functional Description**

  Execute a harmonic wave test and return the harmonic wave's frequency value of the previous 10th harmonic. They are arranged in order from the fundamental wave to the 10th harmonic. The unit is Hz.

- **Return Format**

  The query returns the specified number of times of harmonic wave's frequency value in scientific notation.

- **For Example**

  :READ:HARMonics:FREQuency:ALL? Query all the frequency of the previous 10th harmonic.

:READ:HARMonics:FREQuency<n>?

- **Command Format**

  :READ:HARMonics:FREQuency<n>?

- **Functional Description**

  Execute a harmonic wave test and return the specified number of times of harmonic wave's frequency.

  <n>: Harmonic number, an integer with a value range of 1 to 10. The 1st harmonic is the fundamental harmonic.

- **Return Format**

  The query returns the specified number of times of harmonic wave's frequency in scientific notation, with the unit in Hz.

- **For Example**

  :READ:HARMonics:FREQuency3? The query returns 3rd harmonic wave's frequency.

:READ:OBWidth<n>?

- **Command Format**

  :READ:OBWidth<n>?

- **Functional Description**

  Excute a OBW (occupied bandwidth) measurement, and the query returns the measured results or the trace data.

  <n>: When n is 1, the query returns the measured results of OBW (occupied bandwidth), including the OBW frequency, total power, bandwidth set to test, sweep point, resolution bandwidth, transmission frequency error, and XdB bandwidth. The power unit is dBm, and the unit for frequency and sweep width is Hz.

  When n is 2, the query returns the trace data of OBW (occupied bandwidth). The number of data corresponds to the set sweep points, and the data is separated by commas.

- **Return Format**

  The query returns the measured results of OBW (occupied bandwidth) or the trace data in scientific notation.

- **For Example**

  :READ:OBWidth1? The query returns the measured results of OBW (occupied bandwidth).

  :READ:OBWidth2? The query returns the trace data of OBW (occupied bandwidth).

:READ:OBWidth:APOWr

- **Command Format**

:READ:OBWidth:APOWr?

- **Functional Description**

  Execute an occupied bandwidth test and return the total power of the occupied bandwidth measurement.

- **Return Format**

  The query returns the total power of the occupied bandwidth in scientific notation, with the unit in dBm.

- **For Example**

  :READ:OBWidth:APOWr? The query returns the total power of the occupied bandwidth.

:READ:OBWidth:FERRor?

- **Command Format**

  :READ:OBWidth:FERRor?

- **Functional Description**

  Execute an occupied bandwidth test and return the error of transmission frequency of the occupied bandwidth measurement.

- **Return Format**

  The query returns the error of transmission frequency in scientific notation, with the unit in Hz.

- **For Example**

  :READ:OBWidth:FERRor? The query returns the error of transmission frequency.

:READ:OBWidth:OBWidth?

- **Command Format**

  :READ:OBWidth:OBWidth?

- **Functional Description**

  Execute an occupied bandwidth test and return the occupied bandwidth of the occupied bandwidth measurement.

- **Return Format**

  The query returns the occupied bandwidth in scientific notation, with the default unit in Hz.

- **For Example**

  :READ:OBWidth:OBWidth? The query returns the occupied bandwidth.

:READ:OBWidth:XDB?

- **Command Format**

  :READ:OBWidth:XDB?

- **Functional Description**

  Execute an occupied bandwidth test and return the xdB bandwidth of the occupied bandwidth measurement.

- **Return Format**

  The query returns the xdB bandwidth of the occupied bandwidth measurement in scientific notation, with the unit in Hz.

- **For Example**

  :READ:OBWidth:XDB?

  The query returns the xdB bandwidth of the occupied bandwidth measurement.

:READ:TOI<n>?

- **Command Format**

  :READ:TOI<n>?

- **Functional Description**

  Execute a third-order intermodulation measurement, and the query returns the measured results or the trace data.

  <n>: When n is 0, the query returns the trace data for the third-order intermodulation. The data units are based on frequency and power, with frequency in Hz and power in dBm. The number of data units corresponds to the set sweep points, and the data is separated by commas.

  When n is 1, the query returns the six test results of the third-order intermodulation, including the worst output intercept power, the worst output intercept point frequency, the low output intercept power, the low output intercept point frequency, the high output intercept power, and the high output intercept point frequency. The power unit is dBm, and the frequency unit is Hz.

  When n is 2, the query returns the thirteen test results of the third-order intermodulation, including the worst output intercept point frequency, the worst output intercept point power, the worst output intercept power, the low base frequency, the low base power, the high base frequency, the high base power, the low output intercept point frequency, the low output intercept point power, the low output intercept power, the high output intercept point power frequency, the high output intercept point power, and high output intercept power. The power unit is dBm, and the frequency unit is Hz.

- **Return Format**

  The query returns the measured results or the trace data of the third-order intermodulation.

- **For Example**

  :READ:TOI0? The query returns the trace data of the third-order intermodulation.

  :READ:TOI1? The query returns the measured results of third-order intermodulation.

:READ:TOI2? The query returns the measured results of third-order intermodulation.

:READ:TOI:IP3?

- **Command Format**

  :READ:TOI:IP3?

- **Functional Description**

  Execute a third-order intermodulation measurement and return the result, which including the coefficient of third-order intercept, the unit is dBc; third-order intercept point, the unit is dBm.

- **Return Format**

  The query returns the result of third-order intermodulation measurement in scientific notation.

- **For Example**

  :READ:TOI:IP3? The query returns the result of third-order intermodulation measurement.

:READ:TOI:LOW:TONE

- **Command Format**

  :READ:TOI:LOW:TONE?

- **Functional Description**

  Execute a third-order intermodulation measurement and return the frequency and amplitude of low fundamental frequency for third-order intermodulation measurement. The unit of frequency is Hz. The unit of amplitude is dBm.

- **Return Format**

  The query returns the frequency and amplitude of low fundamental frequency in scientific notation.

- **For Example**

  :READ:TOI:LOW:TONE?

  The query returns the frequency and amplitude of low fundamental frequency.

:READ:TOI:LOW:TRD

- **Command Format**

  :READ:TOI:LOW:TRD?

- **Functional Description**

  Execute a third-order intermodulation measurement and return the result of low third-order intercept, which including frequency, the unit of frequency is Hz; amplitude, the unit of amplitude is dBm; the coefficient of low third-order intercept, the unit dBc; low third-order intercept point, the unit is dBm.

- **Return Format**

  The query returns the result of low third-order intercept in scientific notation.

- **For Example**

  :READ:TOI:LOW:TRD? The query returns the result of low third-order intercept.

:READ:TOI:UP:TONE

- **Command Format**

  :READ:TOI:UP:TONE?

- **Functional Description**

  Execute a third-order intermodulation measurement and return the frequency and amplitude of high fundamental frequency for third-order intermodulation measurement. The unit of frequency is Hz. The unit of amplitude is dBm.

- **Return Format**

  The query returns the frequency and amplitude of high fundamental frequency in scientific notation.

- **For Example**

  :READ:TOI:UP:TONE?

  The query returns the frequency and amplitude of high fundamental frequency.

:READ:TOI:UP:TRD

- **Command Format**

  :READ:TOI:UP:TRD?

- **Functional Description**

  Execute a third-order intermodulation measurement and return the result of high third-order intercept, which including frequency, the unit of frequency is Hz; amplitude, the unit of amplitude is dBm; the coefficient of high third-order intercept, the unit dBc; high third-order intercept point, the unit is dBm.

- **Return Format**

  The query returns the result of high third-order intercept in scientific notation.

- **For Example**

  :READ:TOI:UP:TRD? The query returns the result of high third-order intercept.

:READ:TPower<n>?

- **Command Format**

  :READ:TPower?

- **Functional Description**

  Execute a time domain power measurement and return the total power or the trace data.

  <n>: When n is 1, the query returns the total power of the time domain power measurement in dBm.

  When n is 2, the query returns the trace data of the time domain power measurement. The number of data corresponds to the set sweep points, and the data is separated by commas.

- **Return Format**

  The query returns the total power or the trace data of the time domain power measurement in scientific notation.

- **For Example**

  :READ:TPower1?

  The query returns the total power of the time domain power measurement.

  :READ:TPower2?

  The query returns the trace data of the time domain power measurement.

SENSe Command

[:SENSe]:ACPower:AVERage:COUNt

- **Command Format**

  [:SENSe]:ACPower:AVERage:COUNt <integer>

  [:SENSe]:ACPower:AVERage:COUNt?

- **Functional Description**

  Set the average number for the adjacent channel power.

  <integer>: An integer, with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number of the adjacent channel power. The value range is from 1 to 10000.

- **For Example**

  :ACPower:AVERage:COUNt 10

  Set the average number of the adjacent channel power to 10.

  :ACPower:AVERage:COUNt? The query returns 10.

[:SENSe]:ACPower:AVERage[:STATe]

- **Command Format**

  [:SENSe]:ACPower:AVERage[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:ACPower:AVERage[:STATe]?

- **Functional Description**

The average switch of the adjacent channel power.

- **Return Format**

The query returns the average switch status of the adjacent channel power: 0 or 1.

- **For Example**

:ACPower:AVERage ON Turn on averaged adjacent channel power.

:ACPower:AVERage? The query returns 1.

[:SENSe]:ACPower:AVERage:TCONtrol

- **Command Format**

[:SENSe]:ACPower:AVERage:TCONtrol {EXPonential|REPeat}

[:SENSe]:ACPower:AVERage:TCONtrol?

- **Functional Description**

Select the average mode for the adjacent channel power.

EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of the adjacent channel power: EXPonential or REPeat.

- **For Example**

:ACPower:AVERage:TCONtrol EXPonential Set the average mode to EXPonential.

:ACPower:AVERage:TCONtrol? The query returns EXPonential.

[:SENSe]:ACPower:AVERage:TYPE

- **Command Format**

[:SENSe]:ACPower:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:ACPower:AVERage:TYPE?

- **Functional Description**

Set the average type for the adjacent channel power.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

The query returns the average type of the adjacent channel power: VOLTage, POWEr, or LOG.

- **For Example**

:ACPower:AVERage:TYPE POWEr	Set the average mode to POWEr.
:ACPower:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:ACPower:BANDwidth|BWIDth:INTegration

- **Command Format**

  [:SENSe]:ACPower:BANDwidth|BWIDth:INTegration <freq>

  [:SENSe]:ACPower:BANDwidth|BWIDth:INTegration?

- **Functional Description**

  Set the noise bandwidth for the adjacent channel power.

  <freq>: Frequency value with the unit in Hz.

- **Return Format**

  The query returns the noise bandwidth of the adjacent channel power in scientific notation, with the unit in Hz.

- **For Example**

  :ACPower:BANDwidth:INTegration 10MHz

  Set the noise bandwidth of the adjacent channel power to 10 MHz.

:ACPower:BANDwidth:INTegration?	The query returns 1.000000e+07.

[:SENSe]:ACPower:CARRier:LIST:BANDwidth|BWIDth[:INTegration]

- **Command Format**

  [:SENSe]:ACPower:CARRier:LIST:BANDwidth|BWIDth[:INTegration] <freq>

  [:SENSe]:ACPower:CARRier:LIST:BANDwidth|BWIDth[:INTegration]?

- **Functional Description**

  Set the integral bandwidth for the adjacent channel power.

  <freq>: A continuous real number with the unit in Hz.

- **Return Format**

  The query returns the integral bandwidth of the adjacent channel power in scientific notation, with the unit in Hz.

- **For Example**

  :ACPower:CARRier:LIST:BANDwidth 10MHz

  Set the integral bandwidth of the adjacent channel power to 10 MHz.

:ACPower:CARRier:LIST:BANDwidth?	The query returns 1.000000e+07.

[:SENSe]:ACPower:DETector[:FUNCtion]

- **Command Format**

[:SENSe]:ACPower:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

[:SENSe]:ACPower:DETector[:FUNCtion]?

- **Functional Description**

Select the detector for the adjacent channel power.

SAMPle: Sampling detection

POSitive: Peak detection

NEGative: Negative peak detection

NORMal: Normal detection

AVERage: Averaged detection

- **Return Format**

The query returns the detector of the adjacent channel power, SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:ACPower:DETector POSitive

Select the detector of the adjacent channel power to POSitive.

:ACPower:DETector? The query returns POSitive.

[:SENSe]:ACPower:OFFSet:LIST[:FREQuency]

- **Command Format**

[:SENSe]:ACPower:OFFSet:LIST[:FREQuency] <freq>

[:SENSe]:ACPower:OFFSet:LIST[:FREQuency]?

- **Functional Description**

Set the frequency offset for the adjacent channel power.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

The query returns the frequency offset of the adjacent channel power in scientific notation, with the unit in Hz.

- **For Example**

:ACPower:OFFSet:LIST 1MHz

Set the frequency offset of the adjacent channel power to 1 MHz.

:ACPower:OFFSet:LIST? The query returns 1.000000e+06.

[:SENSe]:ACPower:OFFSet[:OUTer]:LIST:SIDE

- **Command Format**

[:SENSe]:ACPower:OFFSet[:OUTer]:LIST:SIDE {BOTH|NEGative|POSitive}

[:SENSe]:ACPower:OFFSet[:OUTer]:LIST:SIDE?

- **Functional Description**

Set the shift side for the adjacent channel power.

BOTH: Both side

NEGative: Negative

POSitive: Positive

- **Return Format**

The query returns the shift side of the adjacent channel power: BOTH, NEGative or POSitive.

- **For Example**

:ACPower:OFFSet:LIST:SIDE BOTH

Set the shift side of the adjacent channel power to SIDE BOTH.

:ACPower:OFFSet:LIST:SIDE? The query returns BOTH.

[:SENSe]:AVERage:COUNt

- **Command Format**

[:SENSe]:AVERage:COUNt <integer>

[:SENSe]:AVERage:COUNt?

- **Functional Description**

Set the average number.

<integer>: An integer with a value range of 1 to 10000.

- **Return Format**

The query returns the average number, with a value range of 1 to 10000.

- **For Example**

:AVERage:COUNt 10 Set the average number to 10.

:AVERage:COUNt? The query returns 10.

[:SENSe]:AVERage:TYPE

- **Command Format**

[:SENSe]:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:AVERage:TYPE?

- **Functional Description**

Set the average type.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

  The query returns the average type: VOLTage, POWEr, or LOG.

- **For Example**

:AVERage:TYPE VOLTage	Set the average type to VOLTage.
:AVERage:TYPE?	The query returns VOLTage.

[:SENSe]:BANDwidth|BWIDth[:RESolution]

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth[:RESolution] <freq>

  [:SENSe]:BANDwidth|BWIDth[:RESolution]?

- **Functional Description**

  Set the resolution bandwidth.

  <freq>: A discrete real number with a default unit of Hz. The value ranges from 1 Hz to the maximum of the resolution bandwidth, stepped in a 1-3-10 sequence.

  The maximum resolution bandwidth varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the resolution bandwidth in scientific notation, with the unit in Hz.

- **For Example**

:BANDwidth 1MHz	Set the resolution bandwidth to 1 MHz.
:BANDwidth?	The query returns 1.000000e+06.

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO?

- **Functional Description**

  Automatically/manually switch the resolution bandwidth.

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the resolution bandwidth in automatic mode: 0 or 1.

- **For Example**

:BANDwidth:AUTO ON	Switch the resolution bandwidth automatic (AUTO) mode.
:BANDwidth:AUTO?	The query returns 1.

[:SENSe]:BANDwidth|BWIDth:SHAPe

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth:SHAPe {GAUSsian|FLATtop}

  [:SENSe]:BANDwidth|BWIDth:SHAPe?

- **Functional Description**

  Select the filter.

  GAUSsian: Gaussian filter

  FLATtop: Flat top filter

- **Return Format**

  The query returns the filter type: GAUSsian or FLATtop.

- **For Example**

  :BANDwidth:SHAPe GAUSsian Select Gaussian filter.

  :BANDwidth:SHAPe? The query returns GAUSsian.

[:SENSe]:BANDwidth|BWIDth:VIDeo

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth:VIDeo <freq>

  [:SENSe]:BANDwidth|BWIDth:VIDeo?

- **Functional Description**

  Set the video bandwidth.

  <freq>: A discrete real number with a default unit of Hz. The value ranges from 1 Hz to the maximum of the resolution bandwidth, stepped in a 1-3-10 sequence.

  The maximum video bandwidth varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the status of the video bandwidth in scientific notation, with the unit in Hz.

- **For Example**

  :BANDwidth:VIDeo 1MHz Set the video bandwidth to 1 MHz.

  :BANDwidth:VIDeo? The query returns 1.000000e+06.

[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO?

- **Functional Description**

Automatically/manually switch the video bandwidth.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of video bandwidth in automatic mode: 0 or 1.

- **For Example**

:BANDwidth:VIDeo:AUTO ON	Switch the video bandwidth to automatic mode (AUTO).
:BANDwidth:VIDeo:AUTO?	The query returns 1.

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio

- **Command Format**

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio <ratio>

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio?

- **Functional Description**

Set the ratio for the video bandwidth and resolution bandwidth.

<ratio>: Discrete real number.

The value range of UTS1015B and UTS1032B is from 0.000001 to 1000000. The maximum video bandwidth of UTS3036B and UTS3084B is from 0.0000003 to 3000000. The maximum video bandwidth of UTS5026A is from 0.0000001 to 8000000.

- **Return Format**

The query returns the ratio in scientific notation.

- **For Example**

:BANDwidth:VIDeo:RATio 0.1	Set the ratio to 0.1.
:BANDwidth:VIDeo:RATio?	The query returns 1.000000e-01.

[:SENSe]:CHPower:AVERage:COUNt

- **Command Format**

[:SENSe]:CHPower:AVERage:COUNt <integer>

[:SENSe]:CHPower:AVERage:COUNt?

- **Functional Description**

Set the average number of the channel power.

<integer>: An integer, with a value range of 1 to 10000.

- **Return Format**

The query returns the average number of the channel power. The value range is from 1 to 10000.

- **For Example**

:CHPower:AVERage:COUNt 10	Set the average number of the channel power to 10.
:CHPower:AVERage:COUNt?	The query returns 10.

[:SENSe]:CHPower:AVERage[:STATe]

- **Command Format**

[:SENSe]:CHPower:AVERage[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:CHPower:AVERage[:STATe]?

- **Functional Description**

The average switch of the channel power.

- **Return Format**

The query returns the status of the switch of the channel power: 0 or 1.

- **For Example**

:CHPower:AVERage ON	Turn on the average switch of the channel power.
:CHPower:AVERage?	The query returns 1.

[:SENSe]:CHPower:AVERage:TCONtrol

- **Command Format**

[:SENSe]:CHPower:AVERage:TCONtrol {EXPonential|REPeat}

[:SENSe]:CHPower:AVERage:TCONtrol?

- **Functional Description**

Set the average mode of the channel power.

EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of the channel power: EXPonential or REPeat.

- **For Example**

:CHPower:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:CHPower:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:CHPower:AVERage:TYPE

- **Command Format**

[:SENSe]:CHPower:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:CHPower:AVERage:TYPE?

- **Functional Description**

Set the average mode for the channel power.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

The query returns the average mode of the channel power: VOLTage, POWEr, or LOG.

- **For Example**

:CHPower:AVERage:TYPE POWEr	Set the average mode to POWEr.
:CHPower:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:CHPower:BANDwidth:INTegration

- **Command Format**

[:SENSe]:CHPower:BANDwidth:INTegration <freq>

[:SENSe]:CHPower:BANDwidth:INTegration?

- **Functional Description**

Set the integral bandwidth for the channel power.

<freq>: Frequency value with the unit in Hz.

- **Return Format**

The query returns the average mode of the channel power in scientific notation, with the unit in Hz.

- **For Example**

:CHPower:BANDwidth:INTegration 2MHz

Set the integral bandwidth of the channel power to 2 MHz.

:CHPower:BANDwidth:INTegration?	The query returns 2.000000e+06.

[:SENSe]:CHPower:DETector[:FUNCtion]

- **Command Format**

[:SENSe]:CHPower:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

[:SENSe]:CHPower:DETector[:FUNCtion]?

- **Functional Description**

Select the detector for the channel power trace.

SAMPle: Sampling detection

POSitive: Peak detection

NEGative: Negative peak detection

NORMal: Normal detection

AVERage: Averaged detection

- **Return Format**

  The query returns the detector for the channel power trace, SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:CHPower:DETector POSitive	Select the detector of the channel power trace to POSitive.
:CHPower:DETector?	The query returns POSitive.

[:SENSe]:CNRatio:AVERage:COUNt

- **Command Format**

  [:SENSe]:CNRatio:AVERage:COUNt <integer>

  [:SENSe]:CNRatio:AVERage:COUNt?

- **Functional Description**

  Set the average number for the carrier to noise.

  <integer>: A continuous integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number of the carrier to noise. The value range is from 1 to 10000.

- **For Example**

:CNRatio:AVERage:COUNt 10	Set the average number of the carrier to noise to 10.
:CNRatio:AVERage:COUNt?	The query returns 10.

[:SENSe]:CNRatio:AVERage[:STATe]

- **Command Format**

  [:SENSe]:CNRatio:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:CNRatio:AVERage[:STATe]?

- **Functional Description**

  Set the average switch of the carrier to noise to on or off.

  1|ON: OFF

  0|OFF: OFF

- **Return Format**

  The query returns the average switch status of the carrier to noise: 0 or 1.

- **For Example**

:CNRatio:AVERage ON	Turn on averaged carrier to noise.
:CNRatio:AVERage?	The query returns 1.

[:SENSe]:CNRatio:AVERage:TCONtrol

- **Command Format**

  [:SENSe]:CNRatio:AVERage:TCONtrol {EXPonential|REPeat}

  [:SENSe]:CNRatio:AVERage:TCONtrol?

- **Functional Description**

  Select the average mode for the carrier to noise.

  EXPonential: Exponential

  REPeat: Repeat

- **Return Format**

  The query returns the average mode of the carrier to noise: EXPonential to REPeat.

- **For Example**

  :CNRatio:AVERage:TCONtrol EXPonential Set the average mode to EXPonential.

  :CNRatio:AVERage:TCONtrol? The query returns EXPonential.

[:SENSe]:CNRatio:AVERage:TYPE

- **Command Format**

  [:SENSe]:CNRatio:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:CNRatio:AVERage:TYPE?

- **Functional Description**

  Set the average type for the carrier to noise.

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average type of the carrier to noise, VOLTage, POWEr, or LOG.

- **For Example**

  :CNRatio:AVERage:TYPE POWEr Set the average type to POWEr.

  :CNRatio:AVERage:TYPE? The query returns POWEr.

[:SENSe]:CNRatio:BANDwidth:INTegration

- **Command Format**

  [:SENSe]:CNRatio:BANDwidth:INTegration <freq>

  [:SENSe]:CNRatio:BANDwidth:INTegration?

- **Functional Description**

  Adjust carrier to noise ratio carrier bandwidth.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

  Query returns carrier to noise ratio carrier bandwidth in scientific notation, with the unit in Hz.

- **For Example**

  :CNRatio:BANDwidth:INTegration 10MHz

  Set carrier to noise ratio carrier bandwidth of 10 MHz

  :CNRatio:BANDwidth:INTegration? The query returns 1.000000e+07.

[:SENSe]:CNRatio:BANDwidth:NOISe

- **Command Format**

  [:SENSe]:CNRatio:BANDwidth:NOISe <freq>

  [:SENSe]:CNRatio:BANDwidth:NOISe?

- **Functional Description**

  Set the noise bandwidth for the carrier to noise.

  <freq>: A continuous real number with the unit in Hz.

- **Return Format**

  The query returns the noise bandwidth of the carrier to noise in scientific notation, with the unit in Hz.

- **For Example**

  :CNRatio:BANDwidth:NOISe 10MHz

  Set the noise bandwidth of the carrier to noise to 10 MHz.

  :CNRatio:BANDwidth:NOISe? The query returns 1.000000e+07.

[:SENSe]:CNRatio:OFFSet

- **Command Format**

  [:SENSe]:CNRatio:OFFSet <freq>

  [:SENSe]:CNRatio:OFFSet?

- **Functional Description**

  Set the frequency offset for the carrier to noise.

  <freq>: A continuous real number with the unit in Hz.

- **Return Format**

  The query returns the frequency offset of the carrier to noise in scientific notation, with the unit in Hz.

- **For Example**

  :CNRatio:OFFSet 10MHz Set the frequency offset of the carrier to noise to 10 MHz.

:CNRatio:OFFSet?	The query returns 1.000000e+07.

[:SENSe]:CNRatio:DETector[:FUNCtion]

- **Command Format**

  [:SENSe]:CNRatio:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

  [:SENSe]:CNRatio:DETector[:FUNCtion]?

- **Functional Description**

  Select the detector for carrier to noise.

  SAMPle: Sampling detection

  POSitive: Peak detection

  NEGative: Negative peak detection

  NORMal: Normal detection

  AVERage: Averaged detection

- **Return Format**

  The query returns the detector of carrier to noise, SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:CNRatio:DETector POSitive	Select the detector of carrier to noise to POSitive.
:CNRatio:DETector?	The query returns POSitive.

[:SENSe]:CORRection:CSET:ALL:DELete

- **Command Format**

  [:SENSe]:CORRection:CSET:ALL:DELete

- **Functional Description**

  Delete all the correction data.

- **Return Format**

  No return value.

- **For Example**

:CORRection:CSET:ALL:DELete	Delete all the correction data.

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

- **Command Format**

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

- **Functional Description**

  Turn off all the corrections.

- **Return Format**

  No return value.

- **For Example**

  :CORRection:CSET:ALL Turn off all the corrections.

[:SENSe]:CORRection:CSET<n>:DATA

- **Command Format**

  [:SENSe]:CORRection:CSET<n>:DATA {<freq>,<ampl>,<freq>,<ampl>,...}

  [:SENSe]:CORRection:CSET<n>:DATA?

- **Functional Description**

  Edit the specified correction data.

  <n>: Serial number of the correction, with a value range of 1 to 10.

  <freq>: Frequency of the correction with the unit in Hz.

  <ampl>: Amplitude of the correction with the unit in dB.

- **Return Format**

  The query returns the specified correction data in scientific notation. The return data follows the structure {frequency, amplitude, frequency, amplitude, ...}, with frequency in Hz and amplitude in dB.

- **For Example**

  :CORRection:CSET1:DATA 10000000,5 Edit the correction 1 (10000000,5).

  :CORRection:CSET1:DATA? The query returns 1.000000e+07., 5.000000e+00.

[:SENSe]:CORRection:CSET<n>:DELete

- **Command Format**

  [:SENSe]:CORRection:CSET<n>:DELete

- **Functional Description**

  Delete the specified correction data.

  <n>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CORRection:CSET1:DELete Delete the correction 1 data.

[:SENSe]:CORRection:CSET<n>[:STATe]

- **Command Format**

[:SENSe]:CORRection:CSET<n>[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:CORRection:CSET<n>[:STATe]?

- **Functional Description**

Switch the specified correction to on or off.

<n>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

The query returns the status of the specified correction: 0 or 1.

- **For Example**

:CORRection:CSET1 ON Turn on the correction 1.

:CORRection:CSET1? The query returns 1.

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

- **Command Format**

[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] {50|75}

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

- **Functional Description**

Select the input resistance, 50 Ω or 75 Ω.

- **Return Format**

The query returns the input resistance: 50 Ω or 75 Ω.

- **For Example**

:CORRection:IMPedance 50 Set the input resistance to 50 Ω.

:CORRection:IMPedance? The query returns 50.

[:SENSe]:CORRection:SELect

- **Command Format**

[:SENSe]:CORRection:SELect <integer>

[:SENSe]:CORRection:SELect?

- **Functional Description**

Select a current correction from the serial number of the correction.

<integer>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

The query returns the serial number of the current correction, with a value range of 1 to 10.

- **For Example**

:CORRection:SELect 2 Select the correction 2.

:CORRection:SELect? The query returns 2.

[:SENSe]:DEMod

- **Command Format**

  [:SENSe]:DEMod {OFF|FM|AM}

  [:SENSe]:DEMod?

- **Functional Description**

  Select the demodulation type.

  OFF: Disable the demodulation

  FM: Frequency demodulation

  AM: Amplitude demodulation

- **Return Format**

  The query returns the demodulation type: OFF, FM, or AM.

- **For Example**

  :DEMod FM Select FM.

  :DEMod? The query returns FM.

[:SENSe]:DETector[:FUNCtion]

- **Command Format**

  [:SENSe]:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

  [:SENSe]:DETector[:FUNCtion]?

- **Functional Description**

  Set the detector of all trace.

  SAMPle: Sampling detection

  POSitive: Peak detection

  NEGative: Negative peak detection

  NORMal: Normal detection

  AVERage: Averaged detection

- **Return Format**

  The query returns the detector of the first trace: SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

  :DETector POSitive Set the detector of all trace to POSitive.

  :DETector? The query returns POSitive.

[:SENSe]:DETector:TRACe<n>

- **Command Format**

[:SENSe]:DETector:TRACe<n> {SAMPle|POSitive|NEGative|NORMal|AVERage}

[:SENSe]:DETector:TRACe<n>?

- **Functional Description**

Select the specified trace detection.

<n>: Serial number of the trace, the value range is from 1 to trace value.

The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

SAMPle: Sampling detection

POSitive: Peak detection

NEGative: Negative peak detection

NORMal: Normal detection

AVERage: Averaged detection

- **Return Format**

The query returns the specified trace detection: SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:DETector:TRACe1 POSitive	Select the trace 1 detection to POSitive.
:DETector:TRACe1?	The query returns POSitive.

[:SENSe]:DETector:TRACe<n>:AUTO

- **Command Format**

[:SENSe]:DETector:TRACe<n>:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:DETector:TRACe<n>:AUTO?

- **Functional Description**

The automatic detector switch of the specified trace.

<n>: Trace serial number, with a value range of 1 to the maximum trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the automatic detector switch of the specified trace: 0 or 1.

- **For Example**

:DETector:TRACe1:AUTO ON	Turn on automatic detector switch of trace 1.
:DETector:TRACe1:AUTO?	The query returns 1.

[:SENSe]:FEED:AREFerence

- **Command Format**

  [:SENSe]:FEED:AREFerence {{1|ON} | {0|OFF}}

  [:SENSe]:FEED:AREFerence?

- **Functional Description**

  Switch the RF calibration signal to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch state of RF calibration signal: 0 or 1.

- **For Example**

:FEED:AREFerence ON	Turn on RF calibrating signal.
:FEED:AREFerence?	The query returns 0.

[:SENSe]:FREQuency:CENTer

- **Command Format**

  [:SENSe]:FREQuency:CENTer <freq>

  [:SENSe]:FREQuency:CENTer?

- **Functional Description**

  Set the center frequency for the sweep.

  <freq>: A continuous real number. The default unit is Hz, with a frequency range of 50 Hz to the maximum frequency -50 Hz.

  The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the center frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz	Set the center frequency of the sweep to 1 GHz.
:FREQuency:CENTer?	The query returns 1.000000e+09.

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

- **Command Format**

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

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

- **Functional Description**

Automatically/manually switch the center frequency.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of stepped center frequency in automatic mode: 0 or 1.

- **For Example**

:FREQuency:CENTer:STEP:AUTO ON

Switch the stepped center frequency to automatic mode (AUTO).

:FREQuency:CENTer:STEP:AUTO? The query returns 1.

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

- **Command Format**

[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>

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

- **Functional Description**

Set the stepped center frequency.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

The query returns in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer:STEP 10MHz Set the stepped center frequency to 10 MHz.

:FREQuency:CENTer:STEP? The query returns 1.000000e+07.

[:SENSe]:FREQuency:OFFSet

- **Command Format**

[:SENSe]:FREQuency:OFFSet <freq>

[:SENSe]:FREQuency:OFFSet?

- **Functional Description**

Set the frequency offset.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

The query returns the frequency offset in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:OFFSet 10MHz Set the frequency offset to 10 MHz.

:FREQuency:OFFSet? The query returns 1.000000e+07.

[:SENSe]:FREQuency:SPAN

- **Command Format**

  [:SENSe]:FREQuency:SPAN <freq>

  [:SENSe]:FREQuency:SPAN?

- **Functional Description**

  Set the sweep bandwidth.

  <freq>: A continuous real number with the unit in Hz.

  The sweep bandwidth range is from 100 Hz to the maximum frequency.

  The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the sweep bandwidth in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:SPAN 1GHz Set the sweep bandwidth to 1 GHz.

  :FREQuency:SPAN? The query returns 1.000000e+09.

[:SENSe]:FREQuency:SPAN:FULL

- **Command Format**

  [:SENSe]:FREQuency:SPAN:FULL

- **Functional Description**

  Set the sweep bandwidth to full sweep bandwidth (the maximum sweep bandwidth).

  The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

  :FREQuency:SPAN:FULL Set the sweep bandwidth to full sweep bandwidth.

[:SENSe]:FREQuency:SPAN:PREVious

- **Command Format**

  [:SENSe]:FREQuency:SPAN:PREVious

- **Functional Description**

  Set the sweep bandwidth to the last sweep bandwidth.

- **Return Format**

  No return value.

- **For Example**

  :FREQuency:SPAN:PREVious Set the sweep bandwidth to the last sweep bandwidth.

[:SENSe]:FREQuency:SPAN:ZERO

- **Command Format**

  [:SENSe]:FREQuency:SPAN:ZERO

- **Functional Description**

  Set the sweep bandwidth to zero sweep bandwidth.

- **Return Format**

  No return value.

- **For Example**

  :FREQuency:SPAN:ZERO Set the sweep bandwidth to zero sweep bandwidth.

[:SENSe]:FREQuency:STARt

- **Command Format**

  [:SENSe]:FREQuency:STARt <freq>

  [:SENSe]:FREQuency:STARt?

- **Functional Description**

  Set the start frequency for the sweep.

  <freq>: A continuous real number with the unit in Hz.

  Start frequency range is from 0 Hz to the maximum frequency -100 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.100 Hz represents the minimum sweep bandwidth.

- **Return Format**

  The query returns the start frequency value in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:STARt 10MHz Set the start frequency of the sweep to 10 MHz.

  :FREQuency:STARt? The query returns 1.000000e+07.

[:SENSe]:FREQuency:STOP

- **Command Format**

  [:SENSe]:FREQuency:STOP <freq>

  [:SENSe]:FREQuency:STOP?

- **Functional Description**

  Set the stop frequency for the sweep.

<freq>: A continuous real number with the unit in Hz.

The stop frequency range is from 100 Hz to the maximum frequency. 100 Hz represents the minimum sweep bandwidth. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the stop frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:STOP 1GHz	Set the stop frequency of the sweep to 1 GHz.
:FREQuency:STOP?	The query returns 1.000000e+09.

[:SENSe]:FREQuency:TUNE:IMMediate

- **Command Format**

[:SENSe]:FREQuency:TUNE:IMMediate

- **Functional Description**

Set the automatic tuning.

- **Return Format**

No return value.

- **For Example**

:FREQuency:TUNE:IMMediate	Set the automatic tuning.

[:SENSe]:FREQuency:ZOOM:IN

- **Command Format**

[:SENSe]:FREQuency:ZOOM:IN

- **Functional Description**

Zooming in, which expands the sweep bandwidth, makes the signal difficult to identify.

- **Return Format**

No return value.

- **For Example**

:FREQuency:ZOOM:IN	The signal is zoomed in.

[:SENSe]:FREQuency:ZOOM:OUT

- **Command Format**

[:SENSe]:FREQuency:ZOOM:OUT

- **Functional Description**

Zooming out, which reduces the sweep bandwidth, makes the signal more prominent.

- **Return Format**

  No return value.

- **For Example**

  :FREQuency:ZOOM:OUT The signal is zoomed out.

[:SENSe]:HARMonics:AVERage:COUNt

- **Command Format**

  [:SENSe]:HARMonics:AVERage:COUNt <integer>

  [:SENSe]:HARMonics:AVERage:COUNt?

- **Functional Description**

  Set the average number for the harmonic wave.

  <integer>: A continuous integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number of the harmonic wave. The value range is from 1 to 10000.

- **For Example**

  :HARMonics:AVERage:COUNt 10 Set the average number of the harmonic wave to 10.

  :HARMonics:AVERage:COUNt? The query returns 10.

[:SENSe]:HARMonics:AVERage[:STATe]

- **Command Format**

  [:SENSe]:HARMonics:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:HARMonics:AVERage[:STATe]?

- **Functional Description**

  The average switch of the harmonic wave.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the average switch status of the harmonic wave: 0 or 1.

- **For Example**

  :HARMonics:AVERage ON Turn on averaged harmonic wave.

  :HARMonics:AVERage? The query returns 1.

[:SENSe]:HARMonics:AVERage:TCONtrol

- **Command Format**

[:SENSe]:HARMonics:AVERage:TCONtrol {EXPonential|REPeat}

[:SENSe]:HARMonics:AVERage:TCONtrol?

- **Functional Description**

Select the average mode for the harmonic wave.

EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of the harmonic wave: EXPonential or REPeat.

- **For Example**

:HARMonics:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:HARMonics:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:HARMonics:AVERage:TYPE

- **Command Format**

[:SENSe]:HARMonics:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:HARMonics:AVERage:TYPE?

- **Functional Description**

Set the average type for the harmonic wave.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

The query returns the average type of harmonic wave: VOLTage, POWEr, or LOG.

- **For Example**

:HARMonics:AVERage:TYPE POWEr	Set the average type to POWEr.
:HARMonics:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:HARMonics:FREQuency:FUNDamental

- **Command Format**

[:SENSe]:HARMonics:FREQuency:FUNDamental <freq>

[:SENSe]:HARMonics:FREQuency:FUNDamental?

- **Functional Description**

Set the fundamental harmonic frequency of harmonics.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

The query returns the fundamental harmonic frequency of harmonics in scientific notation, with the unit in Hz.

- **For Example**

:HARMonics:FREQuency:FUNDamental 100MHz

Set harmonic basic harmonic frequency of 100 MHz.

:HARMonics:FREQuency:FUNDamental? The query returns 1.000000e+08.

[:SENSe]:HARMonics:NUMBer

- **Command Format**

[:SENSe]:HARMonics:NUMBer <integer>

[:SENSe]:HARMonics:NUMBer?

- **Functional Description**

Set the harmonic wave order.

<integer>: The harmonic wave order, a continuous integer with a value range of 2 to 10.

- **Return Format**

The query returns the harmonic wave order, with a value range of 2 to 10.

- **For Example**

:HARMonics:NUMBer 4 Set the number time of the harmonic wave to 4.

:HARMonics:NUMBer? The query returns 4.

[:SENSe]:HARMonics:SWEeptime

- **Command Format**

[:SENSe]:HARMonics:SWEeptime <time>

[:SENSe]:HARMonics:SWEeptime?

- **Functional Description**

Set the dwell time for the harmonic wave.

<time>: Dwell time, a continuous integer with a value range of 1 ms to 100 s. The default unit is seconds (s).

- **Return Format**

The query returns the dwell time of the harmonic wave in scientific notation, with the unit in second (s).

- **For Example**

:HARMonics:SWEeptime 0.1 Set the dwell time of the harmonic wave to 100 ms.

:HARMonics:SWEeptime? The query returns 1.000000e-01.

[:SENSe]:OBWidth:AVERage:COUNt

- **Command Format**

  [:SENSe]:OBWidth:AVERage:COUNt <integer>

  [:SENSe]:OBWidth:AVERage:COUNt?

- **Functional Description**

  Set the average number for the occupied bandwidth.

  <integer>: A continuous integer, with a value range of from 1 to 10000.

- **Return Format**

  The query returns the average number of the occupied bandwidth. The value range is from 1 to 10000.

- **For Example**

  :OBWidth:AVERage:COUNt 10 Set the average number of the occupied bandwidth to 10.

  :OBWidth:AVERage:COUNt? The query returns 10.

[:SENSe]:OBWidth:AVERage[:STATe]

- **Command Format**

  [:SENSe]:OBWidth:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:OBWidth:AVERage[:STATe]?

- **Functional Description**

  The average switch for the occupied bandwidth.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the average switch status of the occupied bandwidth: 0 or 1.

- **For Example**

  :OBWidth:AVERage ON Turn on AVERage.

  :OBWidth:AVERage? The query returns 1.

[:SENSe]:OBWidth:AVERage:TCONtrol

- **Command Format**

  [:SENSe]:OBWidth:AVERage:TCONtrol {EXPonential|REPeat}

  [:SENSe]:OBWidth:AVERage:TCONtrol?

- **Functional Description**

  Set the average mode for the occupied bandwidth.

  EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of the occupied bandwidth: EXPonential or REPeat.

- **For Example**

:OBWidth:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:OBWidth:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:OBWidth:AVERage:TYPE

- **Command Format**

[:SENSe]:OBWidth:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:OBWidth:AVERage:TYPE?

- **Functional Description**

Set the average type of the occupied bandwidth.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

The query returns the average type of the occupied bandwidth: VOLTage, POWEr, or LOG.

- **For Example**

:OBWidth:AVERage:TYPE POWEr	Set the average type to POWEr.
:OBWidth:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:OBWidth:DETector[:FUNCtion]

- **Command Format**

[:SENSe]:OBWidth:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

[:SENSe]:OBWidth:DETector[:FUNCtion]?

- **Functional Description**

Select the detector for the occupied bandwidth.

SAMPle: Sampling detection

POSitive: Peak detection

NEGative: Negative peak detection

NORMal: Normal detection

AVERage: Averaged detection

- **Return Format**

The query returns the detector of the occupied bandwidth: SAMPle, POSitive, NEGative,

NORMal, or AVERage.

- **For Example**

:OBWidth:DETector POSitive Select the detector of the occupied bandwidth to POSitive.

:OBWidth:DETector? The query returns POSitive.

[:SENSe]:OBWidth:PERCent

- **Command Format**

[:SENSe]:OBWidth:PERCent <ratio>

[:SENSe]:OBWidth:PERCent?

- **Functional Description**

Set the power ratio for the occupied bandwidth.

<ratio>: A continuous real number of floating-point type, with a value range of 0 to 1.

- **Return Format**

The query returns the power ratio of the occupied bandwidth in scientific notation.

- **For Example**

:OBWidth:PERCent 0.5 Set the power ratio of the occupied bandwidth to 0.5.

:OBWidth:PERCent? The query returns 5.000000e-01.

[:SENSe]:OBWidth:XDB

- **Command Format**

[:SENSe]:OBWidth:XDB <real>

[:SENSe]:OBWidth:XDB?

- **Functional Description**

Set the xdb value for the occupied bandwidth.

<real>: A continuous real number with the unit in dB. The value range is from -100 dB to -0.1 dB.

- **Return Format**

The query returns the xdb value of the occupied bandwidth in scientific notation, with the unit in dB.

- **For Example**

:OBWidth:XDB -10 Set the xdb value of the occupied bandwidth to -10 dB.

:OBWidth:XDB? The query returns -1.000000e+01.

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

- **Command Format**

[:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

Set the input attenuation.

<ampl>: Continuous integer, the default unit is dB. The value range is from 0 dB to 51 dB.

- **Return Format**

The query returns the input attenuation, with the unit in dB.

- **For Example**

:POWer:ATTenuation 10	Set the input attenuation to 10 dB.
:POWer:ATTenuation?	The query returns 10.

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

- **Command Format**

[:SENSe]:POWer[:RF]:ATTenuation:AUTO {{1|ON} | {0|OFF}}

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

- **Functional Description**

Switch the input attenuation by manual/automatic.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of input attenuation in automatic mode: 0 or 1.

- **For Example**

:POWer:ATTenuation:AUTO ON	Switch the input attenuation automatic (AUTO) mode.
:POWer:ATTenuation:AUTO?	The query returns 1.

[:SENSe]:POWer[:RF]:GAIN:BAND

- **Command Format**

[:SENSe]:POWer[:RF]:GAIN:BAND {LOW|FULL}

[:SENSe]:POWer[:RF]:GAIN:BAND?

- **Functional Description**

Select the range of pre-amplifier.

LOW: Low frequency band

FULL: High frequency band

- **Return Format**

The query returns the range of pre-amplifier: LOW or FULL.

- **For Example**

:POWer:GAIN:BAND LOW	Select the range of pre-amplifier to LOW.
.:POWer:GAIN:BAND?	The query returns LOW.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

  The switch of pre-amplifier.

- **Return Format**

  The query returns the switch status of pre-amplifier: 0 or 1.

- **For Example**

:POWer:GAIN:STATe ON	Turn on pre-amplifier.
:POWer:GAIN:STATe?	The query returns 1.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

  [:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

  [:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

  Set the frequency reference.

  INTernal: Internal

  EXTernal: External

- **Return Format**

  The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

[:SENSe]:SPECtrogram:AVERage:COUNt

- **Command Format**

  [:SENSe]:SPECtrogram:AVERage:COUNt <integer>

  [:SENSe]:SPECtrogram:AVERage:COUNt?

- **Functional Description**

Set the average number for frequency monitoring.

<integer>: A continuous integer with a value range of 1 to 10000.

- **Return Format**

The query returns the average number of frequency monitoring. The value range is from 1 to 10000.

- **For Example**

:SPECtrogram:AVERage:COUNt 10

Set the average number of frequency monitoring to 10.

:SPECtrogram:AVERage:COUNt? The query returns 10.

[:SENSe]:SPECtrogram:AVERage[:STATe]

- **Command Format**

[:SENSe]:SPECtrogram:AVERage[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:SPECtrogram:AVERage[:STATe]?

- **Functional Description**

The average switch of frequency monitoring.

- **Return Format**

The query returns the average switch status of frequency monitoring: 0 or 1.

- **For Example**

:SPECtrogram:AVERage ON Turn on averaged frequency monitoring.

:SPECtrogram:AVERage? The query returns 1.

[:SENSe]:SPECtrogram:AVERage:TCONtrol

- **Command Format**

[:SENSe]:SPECtrogram:AVERage:TCONtrol {EXPonential|REPeat}

[:SENSe]:SPECtrogram:AVERage:TCONtrol?

- **Functional Description**

Select the average mode for frequency monitoring.

EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of frequency monitoring: EXPonential or REPeat.

- **For Example**

:SPECtrogram:AVERage:TCONtrol EXPonential Set the average mode to EXPonential.

:SPECtrogram:AVERage:TCONtrol? The query r eturns EXPonential.

[:SENSe]:SPECtrogram:AVERage:TYPE

- **Command Format**

  [:SENSe]:SPECtrogram:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:SPECtrogram:AVERage:TYPE?

- **Functional Description**

  Set the average type for frequency monitoring.

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average type of frequency monitoring, VOLTage, POWEr, or LOG.

- **For Example**

  :SPECtrogram:AVERage:TYPE POWEr Set the average type to POWEr.

  :SPECtrogram:AVERage:TYPE? The query returns POWEr.

[:SENSe]:SPECtrogram:DETector[:FUNCtion]

- **Command Format**

  [:SENSe]:SPECtrogram:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

  [:SENSe]:SPECtrogram:DETector[:FUNCtion]?

- **Functional Description**

  Select the detector for the frequency spectrum monitoring.

  SAMPle: Sampling detection

  POSitive: Peak detection

  NEGative: Negative peak detection

  NORMal: Normal detection

  AVERage: Averaged detection

- **Return Format**

  The query returns the detector of the frequency spectrum monitoring: SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

  :SPECtrogram:DETector POSitive

  Select the detector for the frequency spectrum monitoring to POSitive.

  :SPECtrogram:DETector? The query returns POSitive.

[:SENSe]:SPECtrogram:TRACe

- **Command Format**

  [:SENSe]:SPECtrogram:TRACe <integer>

  [:SENSe]:SPECtrogram:TRACe?

- **Functional Description**

  Set the history trace for the frequency spectrum monitoring.

  <integer>: A continuous integer, with a value range of 1 to 300.

- **Return Format**

  The query returns the historical trace of the frequency spectrum monitoring, with a value range of 1 to 300.

- **For Example**

:SPECtrogram:TRACe 100	Set the historical trace to 100.
:SPECtrogram:TRACe?	The query returns 100.

[:SENSe]:SWEep:POINts

- **Command Format**

  [:SENSe]:SWEep:POINts <integer>

  [:SENSe]:SWEep:POINts?

- **Functional Description**

  Set the sweep points.

  <integer>: The point is an integer with a value range of 11 to the maximum point.

  The maximum points vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the sweep points.

- **For Example**

:SWEep:POINts 1001	Set the sweep points to 1001.
:SWEep:POINts?	The query returns 1001.

[:SENSe]:SWEep:TIME

- **Command Format**

  [:SENSe]:SWEep:TIME <time>

  [:SENSe]:SWEep:TIME?

- **Functional Description**

  Set the sweep time.

<time>: Time value with the unit in second (s). The zero span range is from 1 μs to 4 ks.

The non-zero span range is from 1 ms to 4 ks.

- **Return Format**

The query returns the sweep time in scientific notation, with the unit in second (s).

- **For Example**

:SWEep:TIME 0.1	Set the sweep time to 100 ms.
:SWEep:TIME?	The query returns 1.000000e-01.

[:SENSe]:SWEep:TIME:AUTO

- **Command Format**

[:SENSe]:SWEep:TIME:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:SWEep:TIME:AUTO?

- **Functional Description**

Automatically/manually switch the sweep time.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of the sweep time: 0 or 1.

- **For Example**

:SWEep:TIME:AUTO ON	Switch the sweep time to automatic (AUTO) mode.
:SWEep:TIME:AUTO?	The query returns 1.

[:SENSe]:SWEep:TIME:AUTO:RULes

- **Command Format**

[:SENSe]:SWEep:TIME:AUTO:RULes {NORMal|ACCuracy}

[:SENSe]:SWEep:TIME:AUTO:RULes?

- **Functional Description**

Select the ruler for the sweep time.

NORMal: Normal

ACCuracy: Accuracy

- **Return Format**

The query returns the ruler of the sweep time, NORMal, or ACCuracy.

- **For Example**

:SWEep:TIME:AUTO:RULes ACCuracy	Select ACCuracy.
:SWEep:TIME:AUTO:RULes?	The query returns ACCuracy.

[:SENSe]:SWEep:TYPE

- **Command Format**

  [:SENSe]:SWEep:TYPE {SWEep|FFT}

  [:SENSe]:SWEep:TYPE?

- **Functional Description**

  Select the sweep mode.

  SWEep: Sweep

  FFT: FFT

- **Return Format**

  The query returns the sweep mode, SWEep, or FFT.

- **For Example**

:SWEep:TYPE SWEep	Select the sweep mode.
:SWEep:TYPE?	The query returns SWEep.

[:SENSe]:SWEep:TYPE:AUTO

- **Command Format**

  [:SENSe]:SWEep:TYPE:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:SWEep:TYPE:AUTO?

- **Functional Description**

  Automatically/manually switch the sweep mode.

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the sweep mode: 0 or 1.

- **For Example**

:SWEep:TYPE:AUTO ON	Switch the sweep mode to automatic (AUTO) mode.
:SWEep:TYPE:AUTO?	The query returns 1.

[:SENSe]:TOI:AVERage:COUNt

- **Command Format**

  [:SENSe]:TOI:AVERage:COUNt <integer>

  [:SENSe]:TOI:AVERage:COUNt?

- **Functional Description**

  Set the average number for third-order intercepts.

  <integer>: A continuous integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number of third-order intercepts. The value range is from 1 to 10000.

- **For Example**

  :TOI:AVERage:COUNt 10 Set the average number of third-order intercepts to 10.

  :TOI:AVERage:COUNt? The query returns 10.

[:SENSe]:TOI:AVERage[:STATe]

- **Command Format**

  [:SENSe]:TOI:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:TOI:AVERage[:STATe]?

- **Functional Description**

  Set the average switch for the third-order intercept.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the average switch status of third-order intercept: 0 or 1.

- **For Example**

  :TOI:AVERage ON Turn on averaged third-order intercept.

  :TOI:AVERage? The query returns 1.

[:SENSe]:TOI:AVERage:TCONtrol

- **Command Format**

  [:SENSe]:TOI:AVERage:TCONtrol {EXPonential|REPeat}

  [:SENSe]:TOI:AVERage:TCONtrol?

- **Functional Description**

  Select the average mode for third-order intercept.

  EXPonential: Exponential

  REPeat: Repeat

- **Return Format**

  The query returns the average mode of third-order intercept: EXPonential or REPeat.

- **For Example**

  :TOI:AVERage:TCONtrol EXPonential Set the average mode to EXPonential.

  :TOI:AVERage:TCONtrol? The query returns EXPonential.

[:SENSe]:TOI:AVERage:TYPE

- **Command Format**

  [:SENSe]:TOI:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:TOI:AVERage:TYPE?

- **Functional Description**

  Set the average type for third-order intercept.

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average type of third-order intercept, VOLTage, POWEr, or LOG.

- **For Example**

:TOI:AVERage:TYPE POWEr	Set the average type to POWEr.
:TOI:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:TOI:DETector[:FUNCtion]

- **Command Format**

  [:SENSe]:TOI:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

  [:SENSe]:TOI:DETector[:FUNCtion]?

- **Functional Description**

  Select the detector for third-order intercept.

  SAMPle: Sampling detection

  POSitive: Peak detection

  NEGative: Negative peak detection

  NORMal: Normal detection

  AVERage: Averaged detection

- **Return Format**

  The query returns the detector of third-order intercept: SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:TOI:DETector POSitive	Select the detector of third-order intercept to POSitive.
:TOI:DETector?	The query returns POSitive.

[:SENSe]:TPOWer:AVERage:COUNt

- **Command Format**

[:SENSe]:TPOWer:AVERage:COUNt <integer>

[:SENSe]:TPOWer:AVERage:COUNt?

- **Functional Description**

Set the average number for the time domain power.

<integer>: An integer, with a value range of 1 to 10000.

- **Return Format**

The query returns the average number of the time domain power. The value range is from 1 to 10000.

- **For Example**

:TPOWer:AVERage:COUNt 10 Set the average number of time domain power to 10.

:TPOWer:AVERage:COUNt? The query returns 10.

[:SENSe]:TPOWer:AVERage[:STATe]

- **Command Format**

[:SENSe]:TPOWer:AVERage[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:TPOWer:AVERage[:STATe]?

- **Functional Description**

Set the average switch for the time domain power.

- **Return Format**

The query returns the the average switch status of time domain power: 0 or 1.

- **For Example**

:TPOWer:AVERage ON Turn on averaged time domain power.

:TPOWer:AVERage? The query returns 1.

[:SENSe]:TPOWer:AVERage:TCONtrol

- **Command Format**

[:SENSe]:TPOWer:AVERage:TCONtrol {EXPonential|REPeat}

[:SENSe]:TPOWer:AVERage:TCONtrol?

- **Functional Description**

Set the average mode for the time domain power.

EXPonential: Exponential

REPeat: Repeat

- **Return Format**

The query returns the average mode of the time domain power: EXPonential or REPeat.

- **For Example**

:TPOWer:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:TPOWer:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:TPOWer:AVERage:TYPE

- **Command Format**

  [:SENSe]:TPOWer:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:TPOWer:AVERage:TYPE?

- **Functional Description**

  Set the average mode for time domain power.

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average mode of time domain power: VOLTage, POWEr, or LOG.

- **For Example**

:TPOWer:AVERage:TYPE POWEr	Set the average mode to POWEr.
:TPOWer:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:TPOWer:DETector[:FUNCtion]

- **Command Format**

  [:SENSe]:TPOWer:DETector[:FUNCtion] {SAMPle|POSitive|NEGative|NORMal|AVERage}

  [:SENSe]:TPOWer:DETector[:FUNCtion]?

- **Functional Description**

  Select the detector for time domain power.

  SAMPle: Sampling detection

  POSitive: Peak detection

  NEGative: Negative peak detection

  NORMal: Normal detection

  AVERage: Averaged detection

- **Return Format**

  The query returns the detector of time domain power, SAMPle, POSitive, NEGative, NORMal, or AVERage.

- **For Example**

:TPOWer:DETector POSitive	Select the detector of time domain power to POSitive.
:TPOWer:DETector?	The query returns POSitive.

[:SENSe]:TPOWer:LLIMit

- **Command Format**

  [:SENSe]:TPOWer:LLIMit <time>

  [:SENSe]:TPOWer:LLIMit?

- **Functional Description**

  Set the start time for time domain power.

  <time>: Time value with the unit in second (s). The value range is from 0 to 10 ms.

- **Return Format**

  The query returns the start time of time domain power in scientific notation, with the unit in second (s).

- **For Example**

  :TPOWer:LLIMit 5ms Set the start time of time domain power to 5 ms.

  :TPOWer:LLIMit? The query returns 5.000000e-03.

[:SENSe]:TPOWer:RLIMit

- **Command Format**

  [:SENSe]:TPOWer:RLIMit <time>

  [:SENSe]:TPOWer:RLIMit?

- **Functional Description**

  Adjust the time domain power end time.

  <time>: A continuous real number with the unit in second (s). The value range is from 0 to 10 ms.

- **Return Format**

  The query returns time domain power end time in scientific notation, with the unit in Hz.

- **For Example**

  :TPOWer:RLIMit 5ms Set the time domain power end time to 5 ms.

  :TPOWer:RLIMit? The query returns 5.000000e-03.

SOURce Command

:SOURce:CORRection:OFFSet

- **Command Format**

  :SOURce:CORRection:OFFSet <real>

  :SOURce:CORRection:OFFSet?

- **Functional Description**

Set the amplitude offset for trace source.

<real>: A continuous real number with the default unit in dB. The value range is from -200 dB to 200 dB.

- **Return Format**

The query returns the amplitude offset of trace source in scientific notation, with the unit in dB.

- **For Example**

:SOURce:CORRection:OFFSet 10	Set the amplitude offset of trace source to 10 dB.
:SOURce:CORRection:OFFSet?	The query returns 1.000000e+01.

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

- **Command Format**

:SOURce[:EXTernal]:POWer[:LEVel][:IMMediate][:AMPLitude] <ampl>

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

- **Functional Description**

Set the amplitude for trace source.

<ampl>: A continous real number value with the default unit in dBm.

- **Return Format**

The query returns the amplitude of trace source in scientific notation, with the unit in dBm.

- **For Example**

:SOURce:POWer -40	Set the amplitude of trace source to -40 dBm.
:SOURce:POWer?	The query returns -4.000000e+01.

:SOURce:TRACe:REFerence:STATe

- **Command Format**

:SOURce:TRACe:REFerence:STATe {{1|ON} | {0|OFF}}

:SOURce:TRACe:REFerence:STATe?

- **Functional Description**

The switch of the reference trace of trace source.

1|ON: Turn on the reference trace

0|OFF: Disable the reference trace

- **Return Format**

The query returns the switch status of the reference trace: 0 or 1.

- **For Example**

:SOURce:TRACe:REFerence:STATe ON	Enable the reference trace.
:SOURce:TRACe:REFerence:STATe?	The query returns 1.

TRACe Command

:TRACe[:DATA]

- **Command Format**

  :TRACe[:DATA] TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<data>

  :TRACe[:DATA]? TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6

- **Functional Description**

  Set and acquire the trace data block. The number of data points corresponds to the sweep points, with each data value representing the power at a frequency point between the start frequency and the stop frequency. The data format is set using the command ":FORMat[:TRACe][:DATA]", and can be ASCII characters, 32-bit, or 64-bit binary real numbers. The default is ASCII, with data values separated by commas. Binary real numbers are converted from floating point data according to the IEEE 754 standard.

  TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6: Trace sequence character, indicating the trace1|2|3|4|5|6.

  The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns trace amplitude data. The ASCII format represents data in scientific notation and returns it as a character stream, while the binary format returns data in the corresponding bit pattern, with data returned in Data Block Format. The default unit is dBm.

- **For Example**

  Set the data for trace 1.

  :TRACe:DATA TRACE1,-59,-77,-60,-98,-59,-59,-77,-60,-98,-59,-59

  :TRACe:DATA? TRACE1

  The query returns

  -5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01

:TRACe:SELect

- **Command Format**

  :TRACe:SELect <integer>

  :TRACe:SELect?

- **Functional Description**

  Select a current trace from the sequence of the trace.

<integer>: Serial number of the trace, integer, the value range is from 1 to trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the current serial number of the trace.

- **For Example**

:TRACe:SELect 1	Set trace 1 to the current trace.
:TRACe:SELect?	The query returns 1.

:TRACe:TYPE

- **Command Format**

:TRACe:TYPE {WRITe|AVERage|MAXHold|MINHold}

:TRACe:TYPE?

- **Functional Description**

Set the trace type for the current trace, for the advanced measurement.

WRITe: Refresh

AVERage: Averaged trace

MAXHold: The maximum hold

MINHold: The minimum hold

- **Return Format**

The query returns the trace type of the current trace: WRITe, AVERage, MAXHold, or MINHold.

- **For Example**

:TRACe:TYPE AVERage	Set the trace type to AVERage.
:TRACe:TYPE?	The query returns AVERage.

:TRACe<n>:DISPlay[:STATe]

- **Command Format**

:TRACe<n>:DISPlay[:STATe] {{1|ON} | {0|OFF}}

:TRACe<n>:DISPlay[:STATe]?

- **Functional Description**

Set the display of the specified trace to on or off.

<n>: Serial number of the trace; an integer with a value range of 1 to the maximum trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the display switch status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:DISPlay ON Trace 1 is displayed.

:TRACe1:DISPlay? The query returns 1.

:TRACe<n>:TYPE

- **Command Format**

:TRACe<n>:TYPE <WRITe|AVERage|MAXHold|MINHold>

:TRACe<n>:TYPE?

- **Functional Description**

Select the trace type for the specified trace.

<n>: Serial number of the trace; an integer with a value range of 1 to the maximum trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

WRITe: Refresh

AVERage: Averaged trace

MAXHold: The maximum hold

MINHold: The minimum hold

- **Return Format**

The query returns the specified trace type: WRITe, AVERage, MAXHold, or MINHold.

- **For Example**

:TRACe1:TYPE AVERage Set trace 1 to AVERage.

:TRACe1:TYPE? The query returns AVERage.

:TRACe<n>:UPDate:STATe

- **Command Format**

:TRACe<n>:UPDate:STATe {{1|ON} | {0|OFF}}

:TRACe<n>:UPDate:STATe?

- **Functional Description**

The refresh switch of the specified trace, after turning on the refresh switch, the trace will keep refreshing.

<n>: Serial number of the trace; an integer with a value range of 1 to the maximum trace value. The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the refresh switch status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:UPDate:STATe ON Turn on the refresh switch of trace 1.

:TRACe1:UPDate:STATe? The query returns 1.

TRIGger Command

:TRIGger[:SEQuence]:EXTernal:DELay

- **Command Format**

:TRIGger[:SEQuence]:EXTernal:DELay <time>

:TRIGger[:SEQuence]:EXTernal:DELay?

- **Functional Description**

Set the trigger delay for the external trigger.

<time>: A continuous positive number with the default unit in s. The value range is from 1000 ps to 500 ms.

- **Return Format**

The query returns the trigger delay of the external trigger in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:EXTernal:DELay 0.01 Set the trigger delay of the external trigger to 10 ms.

:TRIGger:EXTernal:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal:SLOPe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal:SLOPe {POSitive|NEGative}

:TRIGger[:SEQuence]:EXTernal:SLOPe?

- **Functional Description**

Select the trigger edge for the external trigger.

POSitive: Rising edge

POSitive: Falling edge

- **Return Format**

The query returns the trigger edge: external trigger: POSitive, or NEGative.

- **For Example**

:TRIGger:EXTernal:SLOPe POSitive

Select the trigger edge for the external trigger to POSitive.

:TRIGger:EXTernal:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:EXTernal1:DELay

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:DELay <time>

  :TRIGger[:SEQuence]:EXTernal1:DELay?

- **Functional Description**

  Set the trigger delay of the external trigger 1.

  <time>: A continuous positive number with the default unit in s. The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the external trigger 1 in scientific notation, with the unit in second (s).

- **For Example**

  :TRIGger:EXTernal1:DELay 0.01

  Set the trigger delay for the external trigger from 1 to 10 ms.

  :TRIGger:EXTernal1:DELay? Query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:EXTernal1:DELay:STATe?

- **Functional Description**

  Control the trigger delay switch for the external trigger 1, setting it to on or off.

  1|ON: ON

  1|ON: OFF

- **Return Format**

  The query returns the switch state of trigger delay of the external trigger 1: 0 or 1.

- **For Example**

  :TRIGger:EXTernal1:DELay:STATe OFF Disable the trigger delay of the external trigger 1.

  :TRIGger:EXTernal1:DELay:STATe? The query returns 0.

:TRIGger[:SEQuence]:EXTernal1:LEVel

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:LEVel <voltage>

:TRIGger[:SEQuence]:EXTernal1:LEVel?

- **Functional Description**

  Set the trigger level for the external trigger 1. Currently, only the UTS5026A supports this feature.

  <voltage>: Voltage value, with a default unit of volts (V). The range is from -3.3 V to 3.3 V.

- **Return Format**

  The query returns the trigger level of the external trigger 1 in scientific notation.

- **For Example**

  :TRIGger:EXTernal:LEVel 1 Set the trigger level for the external trigger to 1 V.

  :TRIGger:EXTernal:LEVel? The query returns 1.000000e+00.

:TRIGger[:SEQuence]:EXTernal1:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:EXTernal1:SLOPe?

- **Functional Description**

  Select the trigger edge for the external trigger 1.

  POSitive: Positive edge

  NEGative: Negative edge

- **Return Format**

  The query returns the trigger edge of the external trigger 1, POSitive or NEGative.

- **For Example**

  :TRIGger:EXTernal1:SLOPe POSitive

  Select the trigger edge of the external trigger 1 to POSitive.

  :TRIGger:EXTernal1:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:EXTernal2:DELay

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal2:DELay <time>

  :TRIGger[:SEQuence]:EXTernal2:DELay?

- **Functional Description**

  Set the trigger delay for the external trigger 2.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

The query returns the trigger delay of the external trigger 2 in scientific notation, with the unit in second(s).

- **For Example**

:TRIGger:EXTernal2:DELay 0.01	Set the trigger delay of the external trigger 2 to 10 ms.
:TRIGger:EXTernal2:DELay?	The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe {{1|ON} | {0|OFF}}

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe?

- **Functional Description**

Control the trigger delay switch for the external trigger 2, setting it to on or off.

1|ON: ON

1|ON: OFF

- **Return Format**

The query returns the trigger delay switch status of the external trigger 2: 0 or 1.

- **For Example**

:TRIGger:EXTernal2:DELay:STATe OFF	Disable the trigger delay of the external trigger 2.
:TRIGger:EXTernal2:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:EXTernal2:LEVel

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:LEVel <voltage>

:TRIGger[:SEQuence]:EXTernal2:LEVel?

- **Functional Description**

Set the trigger level for the external trigger 2. Currently, only the UTS5026A supports this feature.

<voltage>: Voltage value, with a default unit of volts (V). The range is from -3.3 V to 3.3 V.

- **Return Format**

The query returns the trigger level of the external trigger 2 in scientific notation.

- **For Example**

:TRIGger:EXTernal2:LEVel 1	Set the trigger level of the external trigger 2 to 1 V.
:TRIGger:EXTernal2:LEVel?	The query returns 1.000000e+00.

:TRIGger[:SEQuence]:EXTernal2:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal2:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:EXTernal2:SLOPe?

- **Functional Description**

  Select the trigger edge for the external trigger 2.

  POSitive: Rising edge

  NEGative: Falling edge

- **Return Format**

  The query returns the trigger edge of the external trigger 2, POSitive or NEGative.

- **For Example**

  :TRIGger:EXTernal2:SLOPe POSitive

  Select the trigger edge of the external trigger 2 to POSitive.

  :TRIGger:EXTernal2:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:FRAMe:DELay

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:DELay <time>

  :TRIGger[:SEQuence]:FRAMe:DELay?

- **Functional Description**

  Set the trigger delay for the period trigger. Currently, only the UTS5026A supports this feature.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

  :TRIGger:FRAMe:DELay 0.01 Set the trigger delay of the period trigger to 10 ms.

  :TRIGger:FRAMe:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:FRAMe:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:FRAMe:DELay:STATe?

- **Functional Description**

Control the trigger delay switch for the period trigger, setting it to on or off. Currently, only the UTS5026A supports this feature.

1|ON: ON

1|ON: OFF

- **Return Format**

The query returns the switch state of trigger delay of the period trigger: 0 or 1.

- **For Example**

:TRIGger:FRAMe:DELay:STATe OFF	Disable the trigger delay of the period trigger.
:TRIGger:FRAMe:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:FRAMe:OFFSet

- **Command Format**

:TRIGger[:SEQuence]:FRAMe:OFFSet <time>

:TRIGger[:SEQuence]:FRAMe:OFFSet?

- **Functional Description**

Set the offset for the period trigger. Currently, only the UTS5026A supports this feature.

<time>: A continuous positive number with the default unit in s. The range is 0s to 10s.

- **Return Format**

The query returns the offset of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:FRAMe:OFFSet 1	Set the offset of the period trigger to 1s.
:TRIGger:FRAMe:OFFSet?	The query returns 1.000000e+00.

:TRIGger[:SEQuence]:FRAMe:PERiod

- **Command Format**

:TRIGger[:SEQuence]:FRAMe:PERiod <time>

:TRIGger[:SEQuence]:FRAMe:PERiod?

- **Functional Description**

Set the period for the period trigger. Currently, only the UTS5026A supports this feature.

<time>: A continuous positive number with the default unit in second (s). The range is from100 ns to 559 ms.

- **Return Format**

The query returns the period of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:FRAMe:PERiod 0.03 Set the period of the period trigger to 30 ms.

:TRIGger:FRAMe:PERiod? The query returns 3.000000e-02.

:TRIGger[:SEQuence]:FRAMe:SYNC

- **Command Format**

:TRIGger[:SEQuence]:FRAMe:SYNC {OFF|EXTernal1|EXTernal2}

:TRIGger[:SEQuence]:FRAMe:SYNC?

- **Functional Description**

Set the synchronous source for the period trigger. Currently, only the UTS5026A supports this feature.

OFF

EXTernal1: External 1

EXTernal2: External 2

- **Return Format**

The query returns the synchronous source of the period trigger: OFF, EXTernal1, or EXTernal2.

- **For Example**

:TRIGger:FRAMe:SYNC EXTernal1

Set the synchronous source of the period trigger to EXTernal1.

:TRIGger:FRAMe:SYNC? The query returns “EXTernal1”.

:TRIGger[:SEQuence]:SOURce

- **Command Format**

:TRIGger[:SEQuence]:SOURce { IMMediate|VIDeo|EXTernal1|EXTernal2|FRAMe }

:TRIGger[:SEQuence]:SOURce?

- **Functional Description**

Select the trigger mode.

IMMediate: Free trigger

VIDeo: Video trigger

EXTernal1: External trigger 1

EXTernal2: External trigger 2, currently supported only by UTS5026A.

FRAMe: Period trigger currently supported only by UTS5026A.

- **Return Format**

The query returns the trigger mode, IMMediate, VIDeo, EXTernal1, EXTernal2, or FRAMe.

- **For Example**

:TRIGger:SOURce IMMediate	Select the trigger mode to IMMediate.
:TRIGger:SOURce?	The query returns IMMediate.

:TRIGger[:SEQuence]:VIDeo:DELay

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:DELay <time>

  :TRIGger[:SEQuence]:VIDeo:DELay?

- **Functional Description**

  Set the trigger delay for the video trigger.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the video trigger in scientific notation, with the unit in seconds (s).

- **For Example**

:TRIGger:VIDeo:DELay 0.01	Set the trigger delay for the video trigger to 10 ms.
:TRIGger:VIDeo:DELay?	The query returns 1.000000e-02.

:TRIGger[:SEQuence]:VIDeo:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:VIDeo:DELay:STATe?

- **Functional Description**

  Set the trigger delay switch for external trigger 2.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the trigger delay switch status of the external trigger 2: 0 or 1.

- **For Example**

:TRIGger:VIDeo:DELay:STATe OFF	Disable the trigger delay for the video trigger.
:TRIGger:VIDeo:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:VIDeo:LEVel

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:LEVel <ampl>

:TRIGger[:SEQuence]:VIDeo:LEVel?

- **Functional Description**

  Set the trigger level for the video trigger.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the trigger edge of the external trigger in scientific notation, with the unit in dBm.

- **For Example**

:TRIGger:VIDeo:LEVel -50	Set the trigger level of video trigger to -50dBm.
:TRIGger:VIDeo:LEVel?	The query returns -5.000000e+01.

:TRIGger[:SEQuence]:VIDeo:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:VIDeo:SLOPe?

- **Functional Description**

  Set the trigger edge for the video trigger. Currently, only the UTS5026A supports this feature.

  POSitive: Rising edge

  POSitive: Falling edge

- **Return Format**

  The query returns the trigger edge of the video trigger: POSitive or NEGative.

- **For Example**

:TRIGger:VIDeo:SLOPe POSitive	Set the trigger edge of the video trigger to POSitive.
:TRIGger:VIDeo:SLOPe?	The query returns POSitive.

UNIT Command

:UNIT:POWer

- **Command Format**

  :UNIT:POWer {DBM|DBMV|DBUV|V|W}

  :UNIT:POWer?

- **Functional Description**

  Select the scale unit for Y-axis.

- **Return Format**

  The query returns the scale unit of Y-axis, DBM, DBMV, DBUV, V, or W.

- **For Example**

:UNIT:POWer DBM	Select the scale unit of Y-axis to DBM.
:UNIT:POWer?	The query returns DBM.

EMI

CALCulate Command

:CALCulate:FSCan:LLINe<n>:BUILd

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:BUILd {TRACE1|TRACE2|TRACE3}

- **Functional Description**

  Derive the specified limit value from the specified trace.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  TRACE1 – TRACE3: Corresponds trace 1 to trace 3.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:FSCan:LLINe1:BUILd TRACE1	Derive limit value 1 from trace 1.

:CALCulate:FSCan:LLINe<n>:CONTrol[:DATA]

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:CONTrol[:DATA] <freq>,<freq>,<freq>,…

  :CALCulate:FSCan:LLINe<n>:CONTrol[:DATA]?

- **Functional Description**

  Set the X-axis data for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <freq>: Frequency value with the default unit in Hz.

- **Return Format**

  The query returns the X-axis data of the specified limit value in scientific notation, with the unit in Hz.

- **For Example**

:CALC:FSC:LLINe1:CONT 10,20,30,40	Set the X-axis data for limit 1.
:CALC:FSC:LLINe1:CONT?	The query returns the X-axis data of limit 1.

:CALCulate:FSCan:LLINe<n>:CONTrol:POINts

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:CONTrol:POINts?

- **Functional Description**

  Set the X-axis point for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the X-axis point of the specified limit value.

- **For Example**

:CALC:FSC:LLINe1:CONT:POIN?	The query returns 4.

:CALCulate:FSCan:LLINe<n>:COPY

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:COPY {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6}

- **Functional Description**

  Copy the specified limit value to a different limit value. The limit value cannot be copied to itself.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value.

  LLINE1 - LLINE6: Corresponding to limit value 1 to limit value 6.

  The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:FSCan:LLINe2:COPY LLINE1	Copy the data of limit value 1 to the limit value 2.

:CALCulate:FSCan:LLINe<n>:DATA

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:DATA {<freq>,<ampl>,<connect>,<freq>,<ampl>,<connect>,...}

  :CALCulate:FSCan:LLINe<n>:DATA?

- **Functional Description**

Edit the specified limit data, using {frequency, amplitude, connection attribute} as the basic unit for editing.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: A continuous real number with the unit in Hz.

<ampl>: A continuous real number with the default unit in dBm.

<connect>: Takes a value of 0 or 1. If the value is 1, it indicates that the current point is connected to the previous point to determine the limit line; if the value is 0, it indicates that the current point is not connected to the previous point (cut-off). The <connect> value of the first point is 0.

- **Return Format**

The query returns the specified limit data, with {frequency, amplitude, connection attribute} as the basic unit for newline. The unit of frequency is Hz. The unit of amplitude is dBm. The connection attribute is 0 or 1.

- **For Example**

:CALCulate:FSCan:LLINe1:DATA 10000000,-50,0,100000000,-60,1 Set the data for Limit 1.

:CALCulate:FSCan:LLINe1:DATA? The query returns limit 1 data.

:CALCulate:FSCan:LLINe<n>:DELete

- **Command Format**

:CALCulate:FSCan:LLINe<n>:DELete

- **Functional Description**

Delete the specified limit data.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

No return value.

- **For Example**

:CALCulate:FSCan:LLINe1:DELete Delete the data for Limit 1.

:CALCulate:FSCan:LLINe<n>:DISPlay

- **Command Format**

:CALCulate:FSCan:LLINe<n>:DISPlay {{1|ON} | {0|OFF}}

:CALCulate:FSCan:LLINe<n>:DISPlay?

- **Functional Description**

  Set the display switch for the limit value to on or off.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the display switch status of the limit value: 0 or 1.

- **For Example**

:CALCulate:FSCan:LLINe1:DISPlay ON	Limit 1 is displayed.
:CALCulate:FSCan:LLINe1:DISPlay?	The query returns 1.

:CALCulate:FSCan:LLINe<n>:MARGin

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:MARGin <real>

  :CALCulate:FSCan:LLINe<n>:MARGin?

- **Functional Description**

  Set the margin for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <real>: A continuous real number with the default unit in dB. The range is from -40 dB to 40 dB.

- **Return Format**

  The query returns the margin of the specified limit value in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:FSCan:LLINe1:MARGin 20	Set the margin for limit 1 to 20 dB.
:CALCulate:FSCan:LLINe1:MARGin?	The query returns 2.000000e+01.

:CALCulate:FSCan:LLINe<n>:MARGin:STATe

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:MARGin:STATe {{1|ON} | {0|OFF}}

  :CALCulate:FSCan:LLINe<n>:MARGin:STATe?

- **Functional Description**

  Set the margin switch for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the margin switch status of the specified limit value: 0 or 1.

- **For Example**

:CALCulate:FSCan:LLINe1:MARGin:STATe ON	Turn on the margin for limit 1.
:CALCulate:FSCan:LLINe1:MARGin:STATe?	The query returns 1.

:CALCulate:FSCan:LLINe<n>:OFFSet:UPDate

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:OFFSet:UPDate

- **Functional Description**

  Set the application offset for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:FSCan:LLINe1:OFFSet:UPDate	Set the application offset for limit 1.

:CALCulate:FSCan:LLINe<n>:OFFSet:X

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:OFFSet:X <freq>

  :CALCulate:FSCan:LLINe<n>:OFFSet:X?

- **Functional Description**

  Set the X-axis offset for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the X-axis offset of the specified limit value in scientific notation, with the default unit in Hz.

- **For Example**

:CALCulate:FSCan:LLINe1:OFFSet:X 10000000	Set the X-axis offset for limit 1 to 10 MHz.
:CALCulate:FSCan:LLINe1:OFFSet:X?	The query returns 1.000000e+07.

:CALCulate:FSCan:LLINe<n>:OFFSet:Y

- **Command Format**

:CALCulate:FSCan:LLINe<n>:OFFSet:Y <real>

:CALCulate:FSCan:LLINe<n>:OFFSet:Y?

- **Functional Description**

Set the Y-axis offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the Y-axis offset of the specified limit value in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:FSCan:LLINe1:OFFSet:Y 5	Set the Y-axis offset for limit 1 to 5 dB.
:CALCulate:FSCan:LLINe1:OFFSet:Y?	The query returns 5.000000e+00.

:CALCulate:FSCan:LLINe<n>:TRACe

- **Command Format**

:CALCulate:FSCan:LLINe<n>:TRACe <integer>

:CALCulate:FSCan:LLINe<n>:TRACe?

- **Functional Description**

Select the test trace for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<integer>: Trace serial number, a continuous integer with a value range of 1 to 3.

- **Return Format**

The query returns the current serial number of the limit value.

- **For Example**

:CALCulate:FSCan:LLINe1:TRACe 1	Select trace 1 as the test trace for limit 1.

:CALCulate:FSCan:LLINe1:TRACe? The query returns 1.

:CALCulate:FSCan:LLINe<n>:UPPer[:DATA]

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:UPPer[:DATA] <ampl>,<ampl>,<ampl>,…

  :CALCulate:FSCan:LLINe<n>:UPPer[:DATA]?

- **Functional Description**

  Set the Y-axis data for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <ampl>: Amplitude value with the default unit in dBm.

- **Return Format**

  The query returns the Y-axis data of the specified limit value in scientific notation, with the unit in dBm.

- **For Example**

  :CALC:FSC:LLINe1:UPP -10,-10,-10,-10 Set the Y-axis data for limit 1.

  :CALC:FSC:LLINe1:UPP? The query returns the Y-axis data of limit 1.

:CALCulate:FSCan:LLINe<n>:UPPer:POINts

- **Command Format**

  :CALCulate:FSCan:LLINe<n>:UPPer:POINts?

- **Functional Description**

  Set the Y-axis point for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the Y-axis point of the specified limit value.

- **For Example**

  :CALC:FSC:LLINe1:UPP:POIN? The query returns 4.

:CALCulate:FSCan:MAMarker[:SET]:SLISt

- **Command Format**

  :CALCulate:FSCan:MAMarker[:SET]:SLISt

- **Functional Description**

  Add the measured result of a marker to the signal list. If the marker is not tested, it will not be added to the signal list.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MAMarker:SET:SLISt

  Add the measured result of a marker to the signal list.

:CALCulate:FSCan:MARKer:AOFF

- **Command Format**

  :CALCulate:FSCan:MARKer:AOFF

- **Functional Description**

  Turn off all the markers.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer:AOFF Turn off all the markers.

:CALCulate:FSCan:MARker:FUNCtion:MAMarker?

- **Command Format**

  :CALCulate:FSCan:MARker:FUNCtion:MAMarker?

- **Functional Description**

  Enable the marker measurement and return the measured results.

- **Return Format**

  The query returns the result of marker measurement, which is the marker frequency with the default unit of Hz; the value of detector 1/2/3 with the default unit of dBm; the difference value of the value of detector 1/2/3 and the limit value, the default unit is dBm. The data is separated by comma mark, all values are expressed in scientific notation. 9.91e+37 represents the invalid value.

- **For Example**

  :CALCulate:FSCan:MARker:FUNCtion:MAMarker?

  Enable the marker measurement and return the measured results.

  5.150339e+08,-5.842101e+01,-6.528230e+01,-7.108303e+01,9.910000e+37,9.910000e+37,9.910000e+37

:CALCulate:FSCan:MARKer:PEAK:EXCursion

- **Command Format**

  :CALCulate:FSCan:MARKer:PEAK:EXCursion <ampl>

  :CALCulate:FSCan:MARKer:PEAK:EXCursion?

- **Functional Description**

  Set the peak offset.

  <ampl>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the peak offset in scientific notation, with the unit in dB.

- **For Example**

  :CALCulate:FSCan:MARKer:PEAK:EXCursion 20 Set the peak offset to 20 dB.

  :CALCulate:FSCan:MARKer:PEAK:EXCursion? The query returns 2.000000e+01.

:CALCulate:FSCan:MARKer:PEAK:EXCursion:STATe

- **Command Format**

  :CALCulate:FSCan:MARKer:PEAK:EXCursion:STATe {{1|ON} | {0|OFF}}

  :CALCulate:FSCan:MARKer:PEAK:EXCursion:STATe?

- **Functional Description**

  Automatically/manually switch the peak offset.

  1|ON: Automatically switch the peak offset.

  1|OFF: Manually switch the peak offset.

- **Return Format**

  The query returns the status of the peak offset in automatic mode: 0 or 1.

- **For Example**

  :CALCulate:FSCan:MARKer:PEAK:EXCursion:STATe ON Select the peak offset to ON.

  :CALCulate:FSCan:MARKer:PEAK:EXCursion:STATe? The query returns 1.

:CALCulate:FSCan:MARKer:PEAK:THReshold

- **Command Format**

  :CALCulate:FSCan:MARKer:PEAK:THReshold <ampl>

  :CALCulate:FSCan:MARKer:PEAK:THReshold?

- **Functional Description**

  Set the threshold of peak.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

The query returns the threshold of peak in scientific notation, with the unit in dBm.

- **For Example**

:CALCulate:FSCan:MARKer:PEAK:THReshold -20 Set the threshold of peak to -20 dBm.

:CALCulate:FSCan:MARKer:PEAK:THReshold? The query returns -2.000000e+01.

:CALCulate:FSCan:MARKer:PEAK:THReshold:LINE[:STATe]

- **Command Format**

:CALCulate:FSCan:MARKer:PEAK:THReshold:LINE[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:FSCan:MARKer:PEAK:THReshold:LINE[:STATe]?

- **Functional Description**

The display switch of threshold line for peak.

1|ON: Display

1|ON: Not display

- **Return Format**

The query returns the display status of the peak threshold line.

- **For Example**

:CALCulate:FSCan:MARKer:PEAK:THReshold:LINE ON

To display threshold line of peak.

:CALCulate:FSCan:MARKer:PEAK:THReshold:LINE? The query returns 1.

:CALCulate:FSCan:MARKer:PEAK:THReshold:STATe

- **Command Format**

:CALCulate:FSCan:MARKer:PEAK:THReshold:STATe {{1|ON} | {0|OFF}}

:CALCulate:FSCan:MARKer:PEAK:THReshold:STATe?

- **Functional Description**

Automatically/manually switch the threshold of peak.

1|ON: Automatically switch the threshold of peak.

1|OFF: Manually switch the threshold of peak.

- **Return Format**

The query returns the status of the threshold of peak in automatic mode: 0 or 1.

- **For Example**

:CALCulate:FSCan:MARKer:PEAK:THReshold:STATe ON

Automatically switch the threshold of peak.

:CALCulate:FSCan:MARKer:PEAK:THReshold:STATe? The query returns 1.

:CALCulate:FSCan:MARKer:SELect

- **Command Format**

  :CALCulate:FSCan:MARKer:SELect <integer>

  :CALCulate:FSCan:MARKer:SELect?

- **Functional Description**

  Select a current marker from the sequence of the marker.

  <integer>: Serial number of the marker, a continuous integer with a value range of 1-6.

- **Return Format**

  The query returns the serial number of the current marker. The value range is integer from 1 to 6.

- **For Example**

:CALCulate:FSCan:MARKer:SELect 1	Select marker 1 as the current marker.
:CALCulate:FSCan:MARKer:SELect?	The query returns 1.

:CALCulate:FSCan:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:MAXimum:LEFT

  Execute a next-peak search to the left side for Marker 1.

:CALCulate:FSCan:MARKer<n>:MAXimum[:MAX]

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MAXimum[:MAX]

- **Functional Description**

  Execute a peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:FSCan:MARKer1:MAXimum Execute peak search of marker 1.

:CALCulate:FSCan:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:MAXimum:NEXT Execute a next-peak search for Marker 1.

:CALCulate:FSCan:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:MAXimum:RIGHt

  Execute a next-peak search to the right side for Marker 1.

:CALCulate:FSCan:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:MINimum Execute the minimum peak search for Marker 1.

:CALCulate:FSCan:MARKer<n>:MODE

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

  :CALCulate:FSCan:MARKer<n>:MODE?

- **Functional Description**

  Select the marker mode for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  OFF: Disable the marker function

  POSition: Normal mode

  DELTa: Difference mode

  FIXed: Fixed mode

- **Return Format**

  The query returns the marker mode of the specified marker <time>: Time OFF, POSition, DELTa, or FIXed.

- **For Example**

  :CALCulate:FSCan:MARKer1:MODE POSition Set the mode of marker 1 to POSition.

  :CALCulate:FSCan:MARKer1:MODE? The query returns POSition.

:CALCulate:FSCan:MARKer<n>:REFerence

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:REFerence <integer>

  :CALCulate:FSCan:MARKer<n>:REFerence?

- **Functional Description**

  Select the reference marker for the specified marker. The reference marker cannot be set as its own reference.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  <integer>: Serial number of another marker, excluding the current one. The value range is 1 to 6.

- **Return Format**

  The query returns the reference marker of the specified marker.

- **For Example**

  :CALCulate:FSCan:MARKer1:REFerence 2 Set mark 2 as the reference marker for Marker 1.

  :CALCulate:FSCan:MARKer1:REFerence? The query returns 2.

:CALCulate:FSCan:MARKer<n>[:SET]:METer

- **Command Format**

  :CALCulate:FSCan:MARKer<n>[:SET]:METer

- **Functional Description**

  Set the specified marker's frequency to match the meter's frequency.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:SET:METer

  Set the marker 1 frequency to match the meter's frequency.

:CALCulate:FSCan:MARKer<n>[:SET]:SLISt

- **Command Format**

  :CALCulate:FSCan:MARKer<n>[:SET]:SLISt

- **Functional Description**

  Add a signal to the signal list. The frequency is the specified frequency.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:SET:SLISt

  Add a signal to the signal list. The frequency is the specified frequency.

:CALCulate:FSCan:MARKer<n>:TO:METer

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:TO:METer

- **Functional Description**

  Set the meter's frequency to the specified marker's frequency.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:MARKer1:TO:METer

  Set the frequency of meter to the frequency of marker 1.

:CALCulate:FSCan:MARKer<n>:TRACe

- **Command Format**

  :CALCulate:FSCan:MARKer<n>:TRACe <integer>

  :CALCulate:FSCan:MARKer<n>:TRACe?

- **Functional Description**

  Select the trace for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  <integer>: Serial number of the trace, a continuous integer with a value range is 1 to 3.

- **Return Format**

  The query returns the serial number of traces for the specified marker.

- **For Example**

  :CALCulate:FSCan:MARKer1:TRACe 1 Set marker 1 to correspond to trace 1.

  :CALCulate:FSCan:MARKer1:TRACe? The query returns 1.

:CALCulate:FSCan:SLISt:APPend:METer

- **Command Format**

  :CALCulate:FSCan:SLISt:APPend:METer

- **Functional Description**

  Add a signal to the signal list. The frequency is the meter frequency.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:APPend:METer

  Add a signal to the signal list. The frequency is the meter frequency.

:CALCulate:FSCan:SLISt:DELete:ALL

- **Command Format**

  :CALCulate:FSCan:SLISt:DELete:ALL

- **Functional Description**

  Delete all signals from the signal table.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:DELete:ALL Delete all signals from the signal table.

:CALCulate:FSCan:SLISt:DELete:MARKed

- **Command Format**

  :CALCulate:FSCan:SLISt:DELete:MARKed

- **Functional Description**

  Delete marked signals from the signal table.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:DELete:MARKed Delete marked signals from the signal table.

:CALCulate:FSCan:SLISt:DELete:SIGNal

- **Command Format**

  :CALCulate:FSCan:SLISt:DELete:SIGNal <integer>

- **Functional Description**

  Delete the specified signal from the signal list.

  <integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:DELete:SIGNal 1 Delete the first signal from the signal list.

:CALCulate:FSCan:SLISt:DELete:UNMarked

- **Command Format**

  :CALCulate:FSCan:SLISt:DELete:UNMarked

- **Functional Description**

  Delete the unmarked signal from the signal list.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:DELete:UNMarked Delete the unmarked signal from the signal list.

:CALCulate:FSCan:SLISt:MARK:ALL

- **Command Format**

  :CALCulate:FSCan:SLISt:MARK:ALL

- **Functional Description**

Mark all the signals in the marker list.

- **Return Format**

No return value.

- **For Example**

:CALCulate:FSCan:SLISt:MARK:ALL Mark all the signals in the marker list.

:CALCulate:FSCan:SLISt:MARK:CLEar:ALL

- **Command Format**

:CALCulate:FSCan:SLISt:MARK:CLEar:ALL

- **Functional Description**

Clear the marker from the signal list.

- **Return Format**

No return value.

- **For Example**

:CALCulate:FSCan:SLISt:MARK:CLEar:ALL Clear the marker from the signal list.

:CALCulate:FSCan:SLISt:MARK:CLEar:SIGNal

- **Command Format**

:CALCulate:FSCan:SLISt:MARK:CLEar:SIGNal <integer>

- **Functional Description**

Clear the selected marker from the signal list.

<integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

No return value.

- **For Example**

:CALCulate:FSCan:SLISt:MARK:CLEar:SIGNal 1 Clear the marker of signal 1 from the signal list.

:CALCulate:FSCan:SLISt:MARK:SIGNal

- **Command Format**

:CALCulate:FSCan:SLISt:MARK:SIGNal <integer>

- **Functional Description**

Mark the selected signal in the signal list.

<integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:MARK:SIGNal 1 Mark signal 1 in the signal list.

:CALCulate:FSCan:SLISt:REPLace:METer

- **Command Format**

  :CALCulate:FSCan:SLISt:REPLace:METer <integer>

- **Functional Description**

  Replace the specified signal's frequency with the meter's frequency.

  <integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1000.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:REPLace:METer 1

  Replace the frequency of signal 1 to the frequency of meter.

:CALCulate:FSCan:SLISt:SET:METer

- **Command Format**

  :CALCulate:FSCan:SLISt:SET:METer <integer>

- **Functional Description**

  Set the frequency meter to the frequency of the specified signal.

  <integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:FSCan:SLISt:SET:METer 1 Set the frequency meter to the frequency of signal 1.

:CALCulate:FSCan:SLISt:SORT:ORDer

- **Command Format**

  :CALCulate:FSCan:SLISt:SORT:ORDer {ASCending|DESCending}

  :CALCulate:FSCan:SLISt:SORT:ORDer?

- **Functional Description**

  Set the sort order for the signal list. Set the sorting type with the common signal.

  ASCending: Rising order

  DESCending: Decreasing order

- **Return Format**

The query returns the sort order of the signal list.

- **For Example**

:CALCulate:FSCan:SLISt:SORT:ORDer ASCending Set the signal list in ASCending.

:CALCulate:FSCan:SLISt:SORT:ORDer? The query returns ASCending.

:CALCulate:FSCan:SLISt:SORT:TYPE

- **Command Format**

:CALCulate:FSCan:SLISt:SORT:TYPE

{FREQuency|DAMPlitude1|DAMPlitude2|DAMPlitude3|DLDelta1|DLDelta2|DLDelta3}

:CALCulate:FSCan:SLISt:SORT:TYPE?

- **Functional Description**

Set the sorting type for the signal list.

FREQuency: Frequency value

DAMPlitude1: The measured value of detector 1.

DAMPlitude2: The measured value of detector 2.

DAMPlitude3: The measured value of detector 3.

DLDelta1: The measured value and the corresponding limited difference of detector 1.

DLDelta2: The measured value and the corresponding limited difference of detector 2.

DLDelta3: The measured value and the corresponding limited difference of detector 3.

- **Return Format**

The query returns the sorting type of the signal list.

- **For Example**

:CALCulate:FSCan:SLISt:SORT:TYPE FREQuency

The signal list is sorted in frequency size.

:CALCulate:FSCan:SLISt:SORT:TYPE? The query returns FREQuency.

:CALCulate:FSCan:SLISt:ZOOM

- **Command Format**

:CALCulate:FSCan:SLISt:ZOOM <integer>

- **Functional Description**

Enlarge the display of the specified signal in the signal list by reducing the bandwidth, centering on the signal.

<integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

No return value.

- **For Example**

:CALCulate:FSCan:SLISt:ZOOM	Enlarge the display of the current signal.
:CALCulate:FSCan:SLISt:ZOOM 1	Enlarge the display of signal 1.

:CALCulate:FSCan:SLISt:ZOOM:OUT

- **Command Format**

  :CALCulate:FSCan:SLISt:ZOOM:OUT <integer>

- **Functional Description**

  Reduce the display of the specified signal in the signal list by enlarging the bandwidth, centering on the signal.

  <integer>: Serial number of the signal, a continuous integer with a value range of 1 to 1001.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:FSCan:SLISt:ZOOM:OUT	Reduce the display of the current signal.
:CALCulate:FSCan:SLISt:ZOOM:OUT 1	Reduce the display of signal 1.

:CALCulate:LLINe:ALL:DELete

- **Command Format**

  :CALCulate:LLINe:ALL:DELete

- **Functional Description**

  Delete all the limit value data.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:LLINe:ALL:DELete Delete all the limit value data.

:CALCulate:LLINe:SELect

- **Command Format**

  :CALCulate:LLINe:SELect <integer>

  :CALCulate:LLINe:SELect?

- **Functional Description**

  Select a currently limit value from the sequence of the limit value.

  <integer>: Serial number of the signal, a continuous integer. The value range is from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to

Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the current serial number of limit value.

- **For Example**

  :CALCulate:LLINe:SELect 1 Select limit value 1 as the current limit value.

  :CALCulate:LLINe:SELect? The query returns 1.

:CALCulate:LLINe<n>:BUILd

- **Command Format**

  :CALCulate:LLINe<n>:BUILd {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6}

- **Functional Description**

  Derive the trace from the specified limited data.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  TRACE1 – TRACE3: Corresponding to trace 1 to trace 3.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:LLINe1:BUILd TRACE1 The limit value 1 data is derived from trace 1.

:CALCulate:LLINe<n>:CONTrol[:DATA]

- **Command Format**

  :CALCulate:LLINe<n>:CONTrol[:DATA] <freq>,<freq>,<freq>,...

  :CALCulate:LLINe<n>:CONTrol[:DATA]?

- **Functional Description**

  Set the X-axis data for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <freq>: Frequency value with the default unit in Hz.

- **Return Format**

  The query returns the X-axis data of the specified limit value in scientific notation, with the unit in Hz.

- **For Example**

:CALC:LLINe1:CONT 10,20,30,40	Set the X-axis data for limit 1.
:CALC:LLINe1:CONT?	The query returns the X-axis data of limit 1.

:CALCulate:LLINe<n>:CONTrol:POINts

- **Command Format**

  :CALCulate:LLINe<n>:CONTrol:POINts?

- **Functional Description**

  Set the X-axis point for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the X-axis point of the specified limit value.

- **For Example**

:CALC:LLINe1:CONT:POIN?	The query returns 4.

:CALCulate:LLINe<n>:COPY

- **Command Format**

  :CALCulate:LLINe<n>: COPY {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6}

- **Functional Description**

  Copy the specified limit value to a different limit value. The limit value cannot be copied to itself.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value.

  LLINE1 - LLINE6: The corresponding limit value 1 to limit value 6.

  The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:LLINe2:COPY LLINE1	Copy the data of limit value 1 to the limit value 2.

:CALCulate:LLINe<n>:DATA

- **Command Format**

  :CALCulate:LLINe<n>:DATA {<freq>,<ampl>,<connect>,<freq>,<ampl>,<connect>,...}

  :CALCulate:LLINe<n>:DATA?

- **Functional Description**

Edit the specified limit data, using {frequency, amplitude, connection attribute} as the basic unit for editing.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: A continuous real number with the unit in Hz.

<ampl>: A continuous real number with the default unit in dBm.

<connect>: Takes a value of 0 or 1. If the value is 1, it indicates that the current point is connected to the previous point to determine the limit line; if the value is 0, it indicates that the current point is not connected to the previous point (cut-off). The <connect> value of the first point is 0.

- **Return Format**

The query returns the specified limit data, with {frequency, amplitude, connection attribute} as the basic unit for newline. The unit of frequency is Hz. The unit of amplitude is dBm. The connection attribute is 0 or 1.

- **For Example**

:CALCulate:LLINe1:DATA 10000000,-50,0,100000000,-60,1 Set the data for Limit 1.

:CALCulate:LLINe1:DATA? The query returns limit 1 data.

:CALCulate:LLINe<n>:DELete

- **Command Format**

:CALCulate:LLINe<n>:DELete

- **Functional Description**

Delete the specified limit data.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

No return value.

- **For Example**

:CALCulate:LLINe1:DELete Delete limit value 1 data.

:CALCulate:LLINe<n>:DISPlay

- **Command Format**

:CALCulate:LLINe<n>:DISPlay {{1|ON} | {0|OFF}}

:CALCulate:LLINe<n>:DISPlay?

- **Functional Description**

  Set the display switch for the limit value to on or off.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the display switch status of the limit value: 0 or 1.

- **For Example**

  :CALCulate:LLINe1:DISPlay ON Limit 1 is displayed.

  :CALCulate:LLINe1:DISPlay? The query returns 1.

:CALCulate:LLINe<n>:MARGin

- **Command Format**

  :CALCulate:LLINe<n>:MARGin <real>

  :CALCulate:LLINe<n>:MARGin?

- **Functional Description**

  Set the margin for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <real>: A continuous real number with the default unit in dB. The range is from -40 dB to 40 dB.

- **Return Format**

  The query returns the margin of the specified limit value in scientific notation, with the unit in dB.

- **For Example**

  :CALCulate:LLINe1:MARGin 20 Set the margin for limit value 1 to 20 dB.

  :CALCulate:LLINe1:MARGin? The query returns 2.000000e+01.

:CALCulate:LLINe<n>:MARGin:STATe

- **Command Format**

  :CALCulate:LLINe<n>:MARGin:STATe {{1|ON} | {0|OFF}}

  :CALCulate:LLINe<n>:MARGin:STATe?

- **Functional Description**

  Set the margin switch for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the margin switch status of the specified limit value: 0 or 1.

- **For Example**

:CALCulate:LLINe1:MARGin:STATe ON	Turn on the margin for Limit 1.
:CALCulate:LLINe1:MARGin:STATe?	The query returns 1.

:CALCulate:LLINe<n>:OFFSet:UPDate

- **Command Format**

:CALCulate:LLINe<n>:OFFSet:UPDate

- **Functional Description**

Set the application offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

No return value.

- **For Example**

:CALCulate:LLINe1:OFFSet:UPDate	Set the application offset for Limit 1.

:CALCulate:LLINe<n>:OFFSet:X

- **Command Format**

:CALCulate:LLINe<n>:OFFSet:X <freq>

:CALCulate:LLINe<n>:OFFSet:X?

- **Functional Description**

Set the X-axis offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<freq>: A continuous real number with the unit in Hz.

- **Return Format**

The query returns the X-axis offset of the specified limit value in scientific notation, with the default unit in Hz.

- **For Example**

:CALCulate:LLINe1:OFFSet:X 10000000	Set the X-axis offset for limit 1 to 10 MHz.
:CALCulate:LLINe1:OFFSet:X?	The query returns 1.000000e+07.

:CALCulate:LLINe<n>:OFFSet:Y

- **Command Format**

:CALCulate:LLINe<n>:OFFSet:Y <real>

:CALCulate:LLINe<n>:OFFSet:Y?

- **Functional Description**

Set the Y-axis offset for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<real>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the Y-axis offset of the specified limit value in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:LLINe1:OFFSet:Y 5	Set the Y-axis offset for limit 1 to 5 dB.
:CALCulate:LLINe1:OFFSet:Y?	The query returns 5.000000e+00.

:CALCulate:LLINe<n>:TRACe

- **Command Format**

:CALCulate:LLINe<n>:TRACe <integer>

:CALCulate:LLINe<n>:TRACe?

- **Functional Description**

Select the test trace for the specified limit value.

<n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

<integer>: Trace serial number, a continuous integer with a value range of 1 to 3.

- **Return Format**

The query returns the current serial number of limit value.

- **For Example**

:CALCulate:LLINe1:TRACe 1	Select trace 1 as the test trace for limit 1.

:CALCulate:LLINe1:TRACe?	The query returns 1.

:CALCulate:LLINe<n>:UPPer[:DATA]

- **Command Format**

  :CALCulate:LLINe<n>:UPPer[:DATA] <ampl>,<ampl>,<ampl>,...

  :CALCulate:LLINe<n>:UPPer[:DATA]?

- **Functional Description**

  Set the Y-axis data for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

  <ampl>: Amplitude value with the default unit in dBm.

- **Return Format**

  The query returns the Y-axis data of the specified limit value in scientific notation, with the unit in dBm.

- **For Example**

:CALC:LLINe1:UPP -10,-10,-10,-10	Set the Y-axis data for Limit 1.
:CALC:LLINe1:UPP?	The query returns the Y-axis data of Limit 1.

:CALCulate:LLINe<n>:UPPer:POINts

- **Command Format**

  :CALCulate:LLINe<n>:UPPer:POINts?

- **Functional Description**

  Set the Y-axis point for the specified limit value.

  <n>: Limit serial number, a continuous integer ranging from 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the Y-axis point of the specified limit value.

- **For Example**

:CALC:LLINe1:UPP:POIN?	The query returns 4.

:CALCulate:MARKer:COUPle:METer

- **Command Format**

  :CALCulate:MARKer:COUPle:METer {{1|ON} | {0|OFF}}

:CALCulate:MARKer:COUPle:METer?

- **Functional Description**

  The meter is coupled to the marker switch.

- **Return Format**

  The query returns the switch status of the meter coupled to the marker: 0 or 1.

- **For Example**

  :CALCulate:MARKer:COUPle:METer ON

  Turn on the switch to allow the meter list to couple to the marker.

  :CALCulate:MARKer:COUPle:METer? The query returns 1.

:CALCulate:MARKer<n>:LINes[:STATe]

- **Command Format**

  :CALCulate:MARKer<n>:LINes[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:LINes[:STATe]?

- **Functional Description**

  Set the marker line switch for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  The query returns the marker line switch status of the specified marker: 0 or 1.

- **For Example**

  :CALCulate:MARKer1:LINes ON Enable the marker line for Marker 1.

  :CALCulate:MARKer1:LINes? The query returns 1.

:CALCulate:MARKer<n>:X

- **Command Format**

  :CALCulate:MARKer<n>:X <freq>

  :CALCulate:MARKer<n>:X?

- **Functional Description**

  Set the coordinate value of the X-axis for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  <freq>: Frequency value with the default unit in Hz.

- **Return Format**

  The query returns the coordinate value of the X-axis for the specified marker in scientific notation, with the unit in Hz.

- **For Example**

:CALCulate:MARKer1:X 1GHz	Set the coordinate value of marker 1 to 1 GHz.
:CALCulate:MARKer1:X?	The query returns 1.000000e+09.

:CALCulate:MARKer<n>:Y

- **Command Format**

  :CALCulate:MARKer<n>:Y <ampl>

  :CALCulate:MARKer<n>:Y?

- **Functional Description**

  Set the amplitude value for the specified marker.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  <ampl>: The amplitude value of the marker, with the default unit in dBm.

- **Return Format**

  The query returns the amplitude value of the specified marker in scientific notation, with the unit in dBm.

- **For Example**

:CALCulate:MARKer1:Y -50	Set the amplitude value of marker 1 to -50 dBm.
:CALCulate:MARKer1:Y?	The query returns -5.000000e+01.

:CALCulate:METer:POWer[:CURRent]?

- **Command Format**

  :CALCulate:METer:POWer[:CURRent]?

- **Functional Description**

  Acquire the current amplitude of the meter.

- **Return Format**

  The query returns the current amplitude of the meter in scientific notation. It returns the current amplitude of 3 meters, the unit is dBm. If the meter is disabled, it returns the invalid value 9.91e+37.

- **For Example**

:CALCulate:METer:POWer?	The query returns the current amplitude of the meter.

:CALCulate:METer:POWer:PEAK?

- **Command Format**

  :CALCulate:METer:POWer:PEAK?

- **Functional Description**

  Acquire the maximum amplitude of the meter.

- **Return Format**

  The query returns the maximum amplitude of the meter in scientific notation. It returns the maximum amplitude of 3 meters, the unit is dBm. If the meter is disabled, it returns the invalid value 9.91e+37.

- **For Example**

  :CALCulate:METer:POWer:PEAK?

  The query returns the maximum amplitude of the meter.

:CALCulate:METer<n>:LIMit[:DATA]

- **Command Format**

  :CALCulate:METer<n>:LIMit[:DATA] <ampl>

  :CALCulate:METer<n>:LIMit[:DATA]?

- **Functional Description**

  Set the limit value of the specified meter.

  <ampl>: Amplitude of limit value, a continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the limited amplitude value of the specified meter in scientific notation, with the unit in dBm.

- **For Example**

  :CALCulate:METer1:LIMit:DATA -50dBm

  Set the limited amplitude value of meter 1 to -50 dBm.

  :CALCulate:METer1:LIMit:DATA? The query returns -5.000000e+01.

:CALCulate:METer<n>:LIMit:STATe

- **Command Format**

  :CALCulate:METer<n>:LIMit:STATe {{1|ON} | {0|OFF}}

  :CALCulate:METer<n>:LIMit:STATe?

- **Functional Description**

  Specific the limited switch for the meter.

- **Return Format**

  The query returns the limited switch of the meter: 0 or 1.

- **For Example**

  :CALCulate:METer1:LIMit:STATe ON Turn on the limit value for the meter list 1.

  :CALCulate:METer1:LIMit:STATe? The query returns 1.

CONFigure Command

:CONFigure:COUPle

- **Command Format**

  :CONFigure:COUPle

- **Functional Description**

  Automatic coupling.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:COUPle Automatic coupling.

:CONFigure:MEASure:DEFAult

- **Command Format**

  :CONFigure:MEASure:DEFAult

- **Functional Description**

  Restore the measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:MEASure:DEFAult Restore the measurement.

DISPlay Command

:DISPlay:DATA?

- **Command Format**

  :DISPlay:DATA?

- **Functional Description**

  Acquire the screen image.

- **Return Format**

  The query returns the screen image.

- **For Example**

  :DISPlay:DATA? Acquire the screen image.

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

- **Command Format**

:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:PDIVision <real>

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

- **Functional Description**

  Set the scale for the Y-axis.

  <real>: A discrete real number, with the default unit in dB. The value range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the scale value of the Y-axis in scientific notation, with the unit in dB.

- **For Example**

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y:PDIVision 1

  Set the scale value for the Y-axis to 1 dB.

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y:PDIVision? The query returns 1.000000e+00.

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

- **Command Format**

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:RLEVel <real>

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

- **Functional Description**

  Set the reference level.

  <real>: A continuous real number. The default unit is the Y-axis scale unit. Use the command :UNIT:POWer? to query.

- **Return Format**

  The query returns the reference level in scientific notation. The unit is the Y-axis scale unit.

- **For Example**

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y:SCALe:RLEVel -10

  Set the reference level to -10 dBm.

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y:SCALe:RLEVel?

  If the scale unit for Y-axis is dBm, the query returns “-1.000000e+01”.

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

- **Command Format**

  :DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet <real>

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

- **Functional Description**

  Set the offset for the reference level.

<real>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the offset of the reference level in scientific notation, with the unit in dB.

- **For Example**

:DISPlay:FSCan:VIEW:WINDow:TRACe:Y:RLEVel:OFFSet 5

Set the offset for the reference level to 5 dB.

:DISPlay:FSCan:VIEW:WINDow:TRACe:Y:RLEVel:OFFSet?

The query returns 5.000000e+00.

:DISPlay:FSCan:WINDow:MAMarker[:STATe]

- **Command Format**

:DISPlay:FSCan:WINDow:MAMarker[:STATe] {{1|ON}|{0|OFF}}

:DISPlay:FSCan:WINDow:MAMarker[:STATe]?

- **Functional Description**

The display switch of marker measurement window.

1|ON: Display

0|OFF: Close

- **Return Format**

The query returns the display switch status of marker measurement window, 0 or 1.

- **For Example**

:DISPlay:FSCan:WINDow:MAMarker:STATe ON

To display marker measurement window.

:DISPlay:FSCan:WINDow:MAMarker:STATe? The query returns 1.

:DISPlay:METer<n>[:STATe]

- **Command Format**

:DISPlay:METer<n>[:STATe] {{1|ON}|{0|OFF}}

:DISPlay:METer<n>[:STATe]?

- **Functional Description**

Set the display switch for the specified meter.

1|ON: Display

0|OFF: Close

- **Return Format**

The query returns the display switch status of the specified meter: 0 or 1.

- **For Example**

:DISPlay:METer1:STATe ON	Meter 1 is displayed.
:DISPlay:METer1:STATe?	The query returns 1.

:DISPlay:VIEW:WINDow:TRACe:X:SPACing

- **Command Format**

  :DISPlay:VIEW:WINDow:TRACe:X:SPACing {LINear|LOGarithmic}

  :DISPlay:VIEW:WINDow:TRACe:X:SPACing?

- **Functional Description**

  Select the scale level of X-axis.

  LINear: Linear

  LOGarithmic: Logarithmic

- **Return Format**

  The query returns the scale level of X-axis, LINear or LOGarithmic.

- **For Example**

  :DISPlay:VIEW:WINDow:TRACe:X:SPACing LOGarithmic

  Select the scale level of X-axis to LOGarithmic.

  :DISPlay:VIEW:WINDow:TRACe:X:SPACing?

  The query returns LOGarithmic.

FETCh Command

FETCh:FSCan<n>?

- **Command Format**

  :FETCh:FSCan<n>?

- **Functional Description**

  Query the measured results or the trace data for scanning measurement, and the query returns the signal data table.

  <n>: When n is 1, the query returns the signal data table, including the signal number, each signal's serial number, signal's trace frequency, the amplitude of three detectors, the difference between the amplitude and the limit value. The unit for signal frequency is Hz, the unit for amplitude is dBm, and the unit for amplitude difference is dB. The data is separated by commas. When n is 2, 3, or 4, the query returns the trace 1, 2, or 3 data of EMI (Electromagnetic Interference). The data unit is based on the frequency and the relative power. The frequency unit is Hz, and the amplitude unit is dBm. The number of data is the sum of the sweep points for all enabled frequency bands in the sweep list. The data is separated by commas.

- **Return Format**

  The query returns the signal data table or the trace data in scientific notation.

- **For Example**

:FETCh:FSCan1?	The query returns the signal data.
:FETCh:FSCan2?	The query returns the trace 1 data.
:FETCh:FSCan3?	The query returns the trace 2 data.
:FETCh:FSCan4?	The query returns the trace 3 data.

FORMat Command

:FORMat[:TRACe][:DATA]

- **Command Format**

  :FORMat[:TRACe][:DATA] {ASCII|REAL32|REAL64}

  :FORMat[:TRACe][:DATA]?

- **Functional Description**

  Set the return format of trace data, the default format is ASCII.

  ASCII: Character

  REAL32: 32-bit binary system integer

  REAL64: 64-bit binary system integer

- **Return Format**

  The query returns the return format of trace data, ASC8, REAL32, or REAL64.

- **For Example**

:FORMat ASCII	Set the return format of trace data to ASCII.
:FORMat?	The query returns ASC8.

INITiate Command

:INITiate:FSCan:CLEar:IMMediate

- **Command Format**

  :INITiate:FSCan:CLEar:IMMediate

- **Functional Description**

  Clear the list and start.

- **Return Format**

  No return value.

- **For Example**

:INITiate:FSCan:CLEar:IMMediate	Clear the list and start.

:INITiate:IMMediate

- **Command Format**

  :INITiate:IMMediate

- **Functional Description**

  Start the sweep.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:IMMediate Start the sweep.

:INITiate:PAUSe

- **Command Format**

  :INITiate:PAUSe

- **Functional Description**

  Pause the sweep.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:PAUSe Pause the sweep.

:INITiate:RESTart

- **Command Format**

  :INITiate:RESTart

- **Functional Description**

  Restart the sweep.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:RESTart Restart the sweep.

:INITiate:RESume

- **Command Format**

  :INITiate:RESume

- **Functional Description**

  Resume the sweep.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:RESume Resume the sweep.

:INITiate:STOP

- **Command Format**

  :INITiate:STOP

- **Functional Description**

  Stop the sweep.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:STOP Stop the sweep.

:INITiate1:CONTinuous

- **Command Format**

  :INITiate1:CONTinuous {{1|ON} | {0|OFF}}

- **Functional Description**

  Switch between single sweep and continuous sweep mode for the meter.

  1|ON: Continuous sweep mode

  0|OFF: Single sweep mode

- **Return Format**

  No return value.

- **For Example**

  :INITiate1:CONTinuous ON Turn on continuous sweep mode for the meter.

  :INITiate1:CONTinuous OFF Turn on single sweep mode for the meter.

:INITiate2:CONTinuous

- **Command Format**

  :INITiate2:CONTinuous {{1|ON} | {0|OFF}}

- **Functional Description**

  Switch between single sweep and continuous sweep mode.

  1|ON: Continuous sweep mode

  0|OFF: Single sweep mode

- **Return Format**

  No return value.

- **For Example**

  :INITiate2:CONTinuous ON Continuous sweep mode.

INPut Command

:INPut:MIXer?

- **Command Format**

  :INPut:MIXer?

- **Functional Description**

  Query the frequency reference of device, EXTernal or INTernal.

- **Return Format**

  The query returns the frequency reference of device, EXTernal or INTernal.

- **For Example**

  :INPut:MIXer? The query returns INTernal.

MMEMory Command

:MMEMory:LOAD:CORRection

- **Command Format**

  :MMEMory:LOAD:CORRection {<integer>,<filename>}

- **Functional Description**

  Load the file from the default directory to the specified limit value.

  <integer>: Serial number of the correction; a continuous integer, with a value range of 1 to 10.

  <filename>: The file name and the file suffix is ".corr". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:CORRection 1,"emi.corr" Correction 1 loads the data file "emi.corr".

:MMEMory:LOAD:LIMit

- **Command Format**

  :MMEMory:LOAD:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

Load the file from the default catalogue to the specified limit value.

LLINE1-LLINE6: Corresponding to limit value 1 to limit value 6.

<filename>: The file name and the file suffix is ".limit". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:LIMit LLINE1,"emi.limit" Limit value 1 loads the data file "emi.limit".

:MMEMory:LOAD:SCAN

- **Command Format**

  :MMEMory:LOAD:SCAN <filename>

- **Functional Description**

  Load the sweep list file from the default catalogue.

  <filename>: The file name and the file suffix is ".csv". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:SCAN "scan.csv" load the sweep list file "scan.csv".

:MMEMory:LOAD:SLISt

- **Command Format**

  :MMEMory:LOAD:SLISt <filename>

- **Functional Description**

  Load the signal list file from the default catalogue.

  <filename>: The file name and the file suffix are ".csv". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:SLISt "slist.csv" Load the signal list file "list.csv".

:MMEMory:LOAD:STATe

- **Command Format**

:MMEMory:LOAD:STATe <filename>

- **Functional Description**

  To load the register status file in the default catalogue.

  <filename>: The file name and the file suffix is ".state". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:STATe "emi.state" To load the register status file "emi.state".

:MMEMory:LOAD:TRACe

- **Command Format**

  :MMEMory:LOAD:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Load the file from the default catalogue to the specified trace.

  TRACE1-TRACE6: Corresponding to trace 1 to trace 6.

  <filename>: The file name and the file suffix is ".trace". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:TRACe TRACE1,"emi.trace" Trace 1 loads the data file "emi.trace".

:MMEMory:STORe:CORRection

- **Command Format**

  :MMEMory:STORe:CORRection {<integer>,<filename>}

- **Functional Description**

  Save the specified correction data in the specified file format to the default catalog.

  <integer>: Serial number of the correction; a continuous integer, with a value range of 1 to 10.

  <filename>: The file name of the correction and the file suffix is ".corr". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:CORRection 1,"emi.corr" Save the correction 1 data to the file "emi.corr".

:MMEMory:STORe:LIMit

- **Command Format**

  :MMEMory:STORe:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

  Save the specified limit value data in the specified file format to the default catalog.

  LLINE1-LLINE6: Corresponding to limit value 1 to limit value 6.

  <filename>: The file name of the limit value and the file suffix is ".limit". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:LIMit LLINE1,"emi.limit" Save limit value 1 data to the file "emi.limit".

:MMEMory:STORe:SCAN

- **Command Format**

  :MMEMory:STORe:SCAN <filename>

- **Functional Description**

  Save the specified sweep list data in the specified file format to the default catalog.

  <filename>: The file name of sweep list and the file suffix is ".csv". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:SCAN "scan.csv" Save trace 1 data to the file "scan.csv".

:MMEMory:STORe:SLISt

- **Command Format**

  :MMEMory:STORe:SLISt <filename>

- **Functional Description**

  Save the specified signal list data in the specified file format to the default catalog.

  <filename>: The file name of signal list and the file suffix is ".csv". The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

:MMEMory:STORe:SLIST “slist.csv” Save trace 1 data to the file to the file “slist.csv”.

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

  Save the specified register status in the specified file format to the default catalog.

  <filename>: The file name of the state and the file suffix is “.state”. The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:STATe “emi.state” Save the register status to the file “emi.state”.

:MMEMory:STORe:TRACe

- **Command Format**

  :MMEMory:STORe:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Save the specified trace data in the specified file format to the default catalog.

  TRACE1-TRACE6: Corresponding to trace 1 to trace 6.

  <filename>: The file name of the trace and the file suffix is “.trace”. The whole file name is a character string, it needs to be wrapped in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:TRACe TRACE1,“emi.trace” Save trace 1 data to the file “emi.trace”.

SENSe Command

[:SENSe]:AVERage:COUNt

- **Command Format**

  [:SENSe]:AVERage:COUNt <integer>

  [:SENSe]:AVERage:COUNt?

- **Functional Description**

  Set the average number.

<integer>: An integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number, with a value range of 1 to 10000.

- **For Example**

  :AVERage:COUNt 10 Set the average number to 10.

  :AVERage:COUNt? The query returns 10.

[:SENSe]:AVERage:TYPE

- **Command Format**

  [:SENSe]:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:AVERage:TYPE?

- **Functional Description**

  Set the average type

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average mode: VOLTage, POWEr, or LOG.

- **For Example**

  :AVERage:TYPE VOLTage Set the average mode to VOLTage.

  :AVERage:TYPE? The query returns VOLTage.

[:SENSe]:BANDwidth|BWIDth[:RESolution]

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth[:RESolution] <freq>

  [:SENSe]:BANDwidth|BWIDth[:RESolution]?

- **Functional Description**

  Set the resolution bandwidth for the meter.

  <freq>: A discrete real number with a default unit of Hz. When the EMI measurement standard is CISPR, the resolution bandwidth is 200 Hz/9 kHz/120 kHz/1 MHz; When the EMI measurement standard is None, the value range is from 1kHz to the maximum of the resolution bandwidth, it stepped by order 1-3-10. The maximum resolution bandwidth varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the resolution bandwidth value of the meter in scientific notation, with the

unit in Hz.

- **For Example**

:BANDwidth 1MHz Set the resolution bandwidth of the meter to 1 MHz.

:BANDwidth? The query returns 1.000000e+06.

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO

- **Command Format**

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO?

- **Functional Description**

Automatically/manually switch the resolution bandwidth of the meter.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of resolution bandwidth of the meter in automatic mode: 0 or 1.

- **For Example**

:BANDwidth:AUTO ON

Switch the resolution bandwidth of the meter to automatic (AUTO) mode.

:BANDwidth:AUTO? The query returns 1.

[:SENSe]:CORRection:CSET:ALL:DELete

- **Command Format**

[:SENSe]:CORRection:CSET:ALL:DELete

- **Functional Description**

Delete all the correction data.

- **Return Format**

No return value.

- **For Example**

:CORRection:CSET:ALL:DELete Delete all the correction data.

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

- **Command Format**

[:SENSe]:CORRection:CSET:ALL[:STATe] {{1|ON} | {0|OFF}}

- **Functional Description**

Turn on/off all the correction.

1|ON: ON

1|OFF: OFF

- **Return Format**

No return value.

- **For Example**

:CORRection:CSET:ALL OFF Turn off all the corrections.

[:SENSe]:CORRection:CSET<n>:DATA

- **Command Format**

[:SENSe]:CORRection:CSET<n>:DATA {<freq>,<ampl>,<freq>,<ampl>,...}

[:SENSe]:CORRection:CSET<n>:DATA?

- **Functional Description**

Edit all the specified correction data.

<n>: Serial number of the correction, with a value range of 1 to 10.

<freq>: Frequency of the correction, with the unit in Hz.

<ampl>: Amplitude of the correction, with the unit in dB.

- **Return Format**

The query returns all the specified correction data in scientific notation. The returned data follows the structure {frequency, amplitude, frequency, amplitude, ...}, with frequency in Hz and amplitude in dB.

- **For Example**

:CORRection:CSET1:DATA 10000000,5 Edit correction 1 (10000000,5).

:CORRection:CSET1:DATA? The query returns 1.000000e+07.,”5.000000e+00”.

[:SENSe]:CORRection:CSET<n>:DELete

- **Command Format**

[:SENSe]:CORRection:CSET<n>:DELete

- **Functional Description**

Delete the specified correction data.

<n>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CORRection:CSET1:DELete Delete correction 1 data.

[:SENSe]:CORRection:CSET<n>[:STATe]

- **Command Format**

  [:SENSe]:CORRection:CSET<n>[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:CORRection:CSET<n>[:STATe]?

- **Functional Description**

  Switch the specified correction to on or off.

  <n>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

  The query returns the switch status of the specified correction: 0 or 1.

- **For Example**

  :CORRection:CSET1 ON Turn on the correction 1.

  :CORRection:CSET1? The query returns 1.

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

- **Command Format**

  [:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] {50|75}

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

- **Functional Description**

  Select the input resistance: 50 Ω or 75 Ω.

- **Return Format**

  The query returns the input resistance value: 50 or 75, with the unit in Ω.

- **For Example**

  :CORRection:IMPedance 50 Set the input resistance to 50 Ω.

  :CORRection:IMPedance? The query returns 50.

[:SENSe]:CORRection:SELect

- **Command Format**

  [:SENSe]:CORRection:SELect <integer>

  [:SENSe]:CORRection:SELect?

- **Functional Description**

  Select a current correction from the sequence of the correction.

  <integer>: Serial number of the correction, with a value range of 1 to 10.

- **Return Format**

  The query returns the current serial number of the correction, with a value range of 1 to 10.

- **For Example**

:CORRection:SELect 2	Select the correction 2.
:CORRection:SELect?	The query returns 2.

[:SENSe]:EMC:STANdard[:SELect]

- **Command Format**

  [:SENSe]:EMC:STANdard[:SELect] {NONE|CISPr}

  [:SENSe]:EMC:STANdard[:SELect]?

- **Functional Description**

  Select the standard of EMI.

  NONE: None

  CISPr: International Special Committee on Radio Interference, it is responsible for the preparation of standard specifications for EMI radio signal protection testing for all types of appliances >9 kHz.

- **Return Format**

  The query returns the average mode: NONE or CISPr.

- **For Example**

:EMC:STANdard CISPr	Set the standard of to CISPR.
:EMC:STANdard?	The query returns CISPr.

[:SENSe]:FREQuency:CENTer

- **Command Format**

  [:SENSe]:FREQuency:CENTer <freq>

  [:SENSe]:FREQuency:CENTer?

- **Functional Description**

  Set the frequency for the meter.

  <freq>: A continuous real number with the unit in Hz.

  The frequency range is from 0 to the maximum frequency. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the frequency value of the meter in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz	Set the frequency of the meter to 1 GHz.
:FREQuency:CENTer?	The query returns 1.000000e+09.

[:SENSe]:FREQuency:MIDSpan

- **Command Format**

  [:SENSe]:FREQuency:MIDSpan <freq>

  [:SENSe]:FREQuency:MIDSpan?

- **Functional Description**

  Set the center frequency for the sweep frequency function.

  <freq>: A continuous real number. The default unit is Hz, with a frequency range of 50 Hz to the maximum frequency -50 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the center frequency value in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:MIDSpan 1GHz Set the center frequency of the sweep frequency to 1 GHz.

  :FREQuency:MIDSpan? The query returns 1.000000e+09.

[:SENSe]:FREQuency:SPAN

- **Command Format**

  [:SENSe]:FREQuency:SPAN <freq>

  [:SENSe]:FREQuency:SPAN?

- **Functional Description**

  Set the sweep bandwidth.

  <freq>: A continuous real number with the unit in Hz.

  The sweep bandwidth range is from 100 Hz to the maximum frequency. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the sweep bandwidth value in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:SPAN 1GHz Set the sweep bandwidth to 1 GHz.

  :FREQuency:SPAN? The query returns 1.000000e+09.

[:SENSe]:FREQuency:STARt

- **Command Format**

  [:SENSe]:FREQuency:STARt <freq>

  [:SENSe]:FREQuency:STARt?

- **Functional Description**

  Set the start frequency for the sweep.

  <freq>: A continuous real number with the unit in Hz.

  Start frequency range is from 0 Hz to the maximum frequency -100 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.100 Hz is minimum sweep bandwidth.

- **Return Format**

  The query returns the start frequency value in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:STARt 10MHz Set the start frequency of the sweep to 10 MHz.

  :FREQuency:STARt? The query returns 1.000000e+07.

[:SENSe]:FREQuency:STARt:AUTO

- **Command Format**

  [:SENSe]:FREQuency:STARt:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:FREQuency:STARt:AUTO?

- **Functional Description**

  Automatically/manually switch the start frequency.

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the start frequency in automatic mode: 0 or 1.

- **For Example**

  :FREQuency:STARt:AUTO ON Switch the start frequency to automatic (AUTO) mode.

  :FREQuency:STARt:AUTO? The query returns 1.

[:SENSe]:FREQuency:STOP

- **Command Format**

  [:SENSe]:FREQuency:STOP <freq>

  [:SENSe]:FREQuency:STOP?

- **Functional Description**

  Set the stop frequency for the sweep.

  <freq>: A continuous real number with the unit in Hz.

  The stop frequency range is from 100 Hz to the maximum frequency. 100 Hz represents the minimum sweep bandwidth. The maximum frequency varies for different models of spectrum

analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the stop frequency value in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:STOP 1GHz	Set the stop frequency of the sweep to 1 GHz.
:FREQuency:STOP?	The query returns 1.000000e+09.

[:SENSe]:FREQuency:STOP:AUTO

- **Command Format**

  [:SENSe]:FREQuency:STOP:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:FREQuency:STOP:AUTO?

- **Functional Description**

  Automatically/manually switch the stop frequency.

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the stop frequency in automatic mode: 0 or 1.

- **For Example**

:FREQuency:STOP:AUTO ON	Switch the stop frequency to automatic (AUTO) mode.
:FREQuency:STOP:AUTO?	The query returns 1.

[:SENSe]:FSCan:DETector:TRACe<n>

- **Command Format**

  [:SENSe]:FSCan:DETector:TRACe<n> {POSitive|QPEak|CAVerage|AVERage|NEGative}

  [:SENSe]:FSCan:DETector:TRACe<n>?

- **Functional Description**

  Select the detector type for the specified trace.

  <n>: Serial number of the trace, the value range is 1-3. EMI only have 3 traces.

  POSitive: Peak detection

  QPEak: Quasi-peak detection

  CAVerage: EMI averaged detection

  AVERage: Averaged detection

  NEGative: Negative peak detection

- **Return Format**

  The query returns the current detector type of the specified trace, POSitive, QPEak, CAVerage,

AVERage, or NEGative.

- **For Example**

:FSCan:DETector:TRACe1 POSitive Select detector type of trace 1 to POSitive.

:FSCan:DETector:TRACe1? The query returns POSitive.

[:SENSe]:FSCan:DETector:TRACe<n>:AUTO

- **Command Format**

[:SENSe]:FSCan:DETector:TRACe<n>:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:FSCan:DETector:TRACe<n>:AUTO?

- **Functional Description**

The automatic detector switch of the specified trace.

<n>: Serial number of the trace, the value range is 1-3. EMI only have 3 traces.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the automatic detector switch of the specified trace: 0 or 1.

- **For Example**

:FSCan:DETector:TRACe1:AUTO ON Turn on automatic detector switch of trace 1.

:FSCan:DETector:TRACe1:AUTO? The query returns 1.

[:SENSe]:FSCan:FINal:DETector<n>

- **Command Format**

[:SENSe]:FSCan:FINal:DETector<n> {POSitive|QPEak|CAVerage|AVERage|NEGative}

[:SENSe]:FSCan:FINal:DETector<n>?

- **Functional Description**

Set the detector type for the specified signal measuring detector.

<n>: The serial number of the signal measuring detector, a continuous integer. The value range is 1 to 3.

POSitive: Peak detection

QPEak: Quasi-peak detection

CAVerage: EMI averaged detection

AVERage: Averaged detection

NEGative: Negative peak detection

- **Return Format**

The query returns the detector type for the specified signal measuring detector, POSitive,

QPEak, CAVerage, AVERage, or NEGative.

- **For Example**

:FSCan:FINal:DETector1 AVERage Set signal measuring detector 1 to AVERage.

:FSCan:FINal:DETector1? The query returns AVERage.

[:SENSe]:FSCan:FINal:DETector<n>:DWELl

- **Command Format**

[:SENSe]:FSCan:FINal:DETector<n>:DWELl <time>

[:SENSe]:FSCan:FINal:DETector<n>:DWELl?

- **Functional Description**

Set dwell time for the specified signal measuring detector.

<n>: The serial number of the signal measuring detector, a continuous integer. The value range is 1 to 3.

<time>: Measured dwell time, a continuous real number; the default unit is seconds (s). The value range is from 1 ms to 60 s.

- **Return Format**

The query returns the dwell time of the specified signal measuring detector in scientific notation, with the unit in second (s).

- **For Example**

:FSCan:FINal:DETector1:DWELl 100ms

Set the dwell time of signal measuring detector 1 to 100 ms.

:FSCan:FINal:DETector1:DWELl? The query returns “1.000000e-01”.

[:SENSe]:FSCan:FINal:DETector<n>:LDELta

- **Command Format**

[:SENSe]:FSCan:FINal:DETector<n>:LDELta <integer>

[:SENSe]:FSCan:FINal:DETector<n>:LDELta?

- **Functional Description**

Select the limit value for the specified signal measuring detector.

<n>: The serial number of the signal measuring detector, a continuous integer. The value range is from 1 to 3.

<integer>: Serial number of the limit value, a continuous integer with a value range of 1 to the maximum limit value. The limit values vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the limit value of the specified signal measuring detector.

- **For Example**

:FSCan:FINal:DETector1:LDELta 2

Select the limit value of signal measuring detector 1 to limit value 2.

:FSCan:FINal:DETector1:LDELta? The query returns 2.

[:SENSe]:FSCan:RANGe<n>:PRESet

- **Command Format**

[:SENSe]:FSCan:RANGe<n>:PRESet {CISA|CISB|CISC|CISD|CISCD|CISE}

- **Functional Description**

The preset frequency band parameter for the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

CISA: Preset frequency band A, 9 kHz to 150 kHz

CISB: Preset frequency band B, 150 kHz to 30 MHz

CISC: Preset frequency band C, 30 MHz to 300 MHz

CISD: Preset frequency band D, 300 MHz to 1 GHz

CISCD: Preset frequency band CD, 30 MHz to 1 GHz

CISCE: Preset frequency band E, 1 GHz to the maximum frequency. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

No return value.

- **For Example**

:FSCan:RANGe5:PRESet CISD

Preset frequency band E for frequency band 5 in the sweep list, 300 MHz to 1 GHz.

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution]

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution] <freq>

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution]?

- **Functional Description**

Set the resolution bandwidth for the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

<freq>: A discrete real number with the default unit in Hz. When the EMI measurement standard is CISPR, the resolution bandwidth is 200 Hz/9 kHz/120 kHz/1 MHz; When the EMI measurement standard is None, the value range is from 1 kHz to the maximum of the resolution bandwidth, it stepped by order 1-3-10. The maximum resolution bandwidth varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the resolution bandwidth of the specified frequency band in the sweep list in scientific notation, with the unit in Hz.

- **For Example**

:FSCan:SCAN5:BANDwidth:RESolution 1MHz

Set the resolution bandwidth of frequency band 5 in the sweep list to 1 MHz.

:FSCan:SCAN5:BANDwidth:RESolution? The query returns 1.000000e+06.

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution]:AUTO

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution]:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:FSCan:SCAN<n>:BANDwidth[:RESolution]:AUTO?

- **Functional Description**

Automatically/manually switch the resolution bandwidth of the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the automatic/manual status of the resolution bandwidth of the specified frequency band in the sweep list: 0 or 1.

- **For Example**

:FSCan:SCAN5:BANDwidth:AUTO ON

The automatic resolution bandwidth of frequency band 5 in the sweep list.

:FSCan:SCAN5:BANDwidth:AUTO? The query returns 1.

[:SENSe]:FSCan:SCAN<n>:INPut:ATTenuation

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:INPut:ATTenuation <integer>

[:SENSe]:FSCan:SCAN<n>:INPut:ATTenuation?

- **Functional Description**

  Set the input attenuation value for the specified frequency band in the sweep list.

  <n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

  <integer>: The input attenuation value of the frequency band is even number; the default unit is dB. The value range is from 0 dB to 50 dB.

- **Return Format**

  The query returns the input attenuation value of the specified frequency band in the sweep list, with the unit in dB.

- **For Example**

  :FSCan:SCAN5:INPut:ATTenuation 6dB

  Set the input attenuation value of frequency band 5 in the sweep list to 6 dB.

  :FSCan:SCAN5:INPut:ATTenuation? The query returns 6.

[:SENSe]:FSCan:SCAN<n>:POINts

- **Command Format**

  [:SENSe]:FSCan:SCAN<n>:POINts <integer>

  [:SENSe]:FSCan:SCAN<n>:POINts?

- **Functional Description**

  Set the sweep's point for the specified frequency band in the sweep list.

  <n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

  <integer>: The sweep's point of frequency band, a continuous integer. The value range is from 11 to the maximum sweep's point. The maximum sweep points vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns in the sweep's point of the specified frequency band in the sweep list.

- **For Example**

  :FSCan:SCAN5:POINts 2000

  Set the sweep's point of frequency band 5 in the sweep list to 2000.

  :FSCan:SCAN5:POINts? The query returns 2.000.

[:SENSe]:FSCan:SCAN<n>:POWer:GAIN[:STATe]

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:POWer:GAIN[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:FSCan:SCAN<n>:POWer:GAIN[:STATe]?

- **Functional Description**

The pre-amplifier switch of the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

- **Return Format**

The query returns the pre-amplifier switch status of the specified frequency band in the sweep list: 0 or 1.

- **For Example**

:FSCan:SCAN5:POWer:GAIN:STATe ON

Enable the pre-amplifier switch of frequency band 5 in the sweep list.

:FSCan:SCAN5:POWer:GAIN:STATe? The query returns 1.

[:SENSe]:FSCan:SCAN<n>:STARt

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:STARt <freq>

[:SENSe]:FSCan:SCAN<n>:STARt?

- **Functional Description**

Set the start frequency for the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

<freq>: The start frequency of the frequency band in the sweep list, a continuous real number; the default unit is Hz. The start frequency range is from 0 Hz to the maximum frequency -100 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the start frequency of the specified frequency band in the sweep list in scientific notation, with the unit in Hz.

- **For Example**

:FSCan:SCAN5:STARt 100MHz

Set the start frequency of frequency band 5 in the sweep list to 100 MHz.

:FSCan:SCAN5:STARt? The query returns 1.000000e+08.

[:SENSe]:FSCan:SCAN<n>:STATe

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:STATe {{1|ON} | {0|OFF}}

[:SENSe]:FSCan:SCAN<n>:STATe?

- **Functional Description**

Specific the frequency band in enable sweep list.

<n>: The serial number of frequency band for the sweep list, a continuous integer. The value range is from 1 to 10.

- **Return Format**

The query returns the enable status of the specified frequency band in sweep list: 0 or 1.

- **For Example**

:FSCan:SCAN5:STATe ON	Frequency band 5 in enable sweep list.
:FSCan:SCAN5:STATe?	The query returns 1.

[:SENSe]:FSCan:SCAN<n>:STOP

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:STOP <freq>

[:SENSe]:FSCan:SCAN<n>:STOP?

- **Functional Description**

Set the stop frequency for the specified frequency band in the sweep list.

<n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

<freq>: The stop frequency of frequency band in the sweep list, a continuous real number; the default unit is Hz. The stop frequency range is from 100 Hz to the maximum frequency. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the stop frequency of the specified frequency band in the sweep list in scientific notation, with the unit in Hz.

- **For Example**

:FSCan:SCAN5:STOP 1GHz

Set the stop frequency of frequency band 5 in the sweep list to 1 GHz.

:FSCan:SCAN5:STOP?	The query returns 1.000000e+09.

[:SENSe]:FSCan:SCAN<n>:TIME

- **Command Format**

[:SENSe]:FSCan:SCAN<n>:TIME <time>

[:SENSe]:FSCan:SCAN<n>:TIME?

- **Functional Description**

  Set the sweep time for the specified frequency band in the sweep list.

  <n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

  < time >: The sweep time of the frequency band in the sweep list, continuous real number. The value range is from 1ms to 4ks.

- **Return Format**

  The query returns the sweep time of the specified frequency band in the sweep list in scientific notation, with the unit in second (s).

- **For Example**

  :FSCan:SCAN5:TIME 10 ms

  Set the sweep time of frequency band 5 in the sweep list to 10 ms.

  :FSCan:SCAN5:TIME? The query returns 1.000000e-02.

[:SENSe]:FSCan:SCAN<n>:TIME:AUTO

- **Command Format**

  [:SENSe]:FSCan:SCAN<n>:TIME:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:FSCan:SCAN<n>:TIME:AUTO?

- **Functional Description**

  Automatically/manually switch the sweep time of the specified frequency band in the sweep list.

  <n>: Serial number of the frequency band in the sweep list; a continuous integer with a value range of 1 to 10.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the automatic/manual status of the sweep time of the specified frequency band in the sweep list.

- **For Example**

  :FSCan:SCAN5:TIME:AUTO ON

  The automatic sweep time of frequency band 5 in the sweep list.

  :FSCan:SCAN5:TIME:AUTO? The query returns 1.

[:SENSe]:FSCan:SEARch:MODE

- **Command Format**

  [:SENSe]:FSCan:SEARch:MODE {PONLy|PLIMits|SLIMits}

  [:SENSe]:FSCan:SEARch:MODE?

- **Functional Description**

  Select the mode for signal search.

  PONLy: Peak

  PLIMits: Peak and limit value

  SLIMits: Segment and limit value

- **Return Format**

  The query returns the signal search mode: PONLy, PLIMits, or SLIMits.

- **For Example**

  :FSCan:SEARch:MODE PONLy Select the mode to “PONLy”.

  :FSCan:SEARch:MODE? The query returns “PONLy”.

[:SENSe]:FSCan:SEARch:PEAK:COUNt

- **Command Format**

  [:SENSe]:FSCan:SEARch:PEAK:COUNt <integer>

  [:SENSe]:FSCan:SEARch:PEAK:COUNt?

- **Functional Description**

  Set the count of peaks for signal search.

  <integer>: Peak count, a continuous integer. The value range is 1 to 50.

- **Return Format**

  The query returns the count of peak for signal search.

- **For Example**

  :FSCan:SEARch:PEAK:COUNt 10 Set the count of peak for signal search to 10.

  :FSCan:SEARch:MODE? The query returns 10.

[:SENSe]:FSCan:SEARch:SUBRange:COUNt

- **Command Format**

  [:SENSe]:FSCan:SEARch:SUBRange:COUNt <integer>

  [:SENSe]:FSCan:SEARch:SUBRange:COUNt?

- **Functional Description**

  Set the count of frequency band for signal search.

  <integer> : The count of frequency band, a continuous integer. The value range is from 1 to 50.

- **Return Format**

  The query returns the count of frequency band for signal search.

- **For Example**

  :FSCan:SEARch:SUBRange:COUNt 10

  Set the count of frequency band for signal search to 10.

  :FSCan:SEARch:SUBRange:COUNt? The query returns 10.

[:SENSe]:FSCan:SEQuence

- **Command Format**

  [:SENSe]:FSCan:SEQuence {SCAN|SEARch|SSAMeasure|SASearch|SAMeasure|REMeasure}

  [:SENSe]:FSCan:SEQuence?

- **Functional Description**

  Select the spectrum sweep mode for EMI.

  SCAN: Scan only

  SEARch: Search only

  SSAMeasure: Scan search measurement

  SASearch: Scan search

  SAMeasure: Search measurement

  REMeasure: Measurement

- **Return Format**

  The query returns the spectrum sweep mode: EMI, SCAN, SEARch, SSAMeasure, SASearch, SAMeasure, or REMeasure.

- **For Example**

  :FSCan:SEQuence SCAN Set the spectrum sweep mode of EMI to SCAN.

  :FSCan:SEQuence? The query returns SCAN.

[:SENSe]:FSCan:SEQuence:REMeasure

- **Command Format**

  [:SENSe]:FSCan:SEQuence:REMeasure {CURRent|ALL|MARKed}

  [:SENSe]:FSCan:SEQuence:REMeasure?

- **Functional Description**

  Select the mode of the measurement.

  CURRent: The current signal

  MARKed: The marked signal

  ALL: All the signal

- **Return Format**

  The query returns the mode of the measurement: CURRent, MARKed, or ALL.

- **For Example**

  :FSCan:SEQuence:REMeasure MARKed Set the mode of the measurement to MARKed.

  :FSCan:SEQuence:REMeasure? The query returns MARKed.

[:SENSe]:FSCan:SLISt:COUPle:METer

- **Command Format**

  [:SENSe]:FSCan:SLISt:COUPle:METer {{1|ON} | {0|OFF}}

  [:SENSe]:FSCan:SLISt:COUPle:METer?

- **Functional Description**

  The meter is coupled to the switch of the signal list.

- **Return Format**

  The query returns the switch status of the meter is coupled to the signal list: 0 or 1.

- **For Example**

  :FSCan:SLISt:COUPle:METer ON

  Turn on the switch that allows the meter to couple with the signal list.

  :FSCan:SLISt:COUPle:METer? The query returns 1.

[:SENSe]:METer:DETector:DWELl

- **Command Format**

  [:SENSe]:METer:DETector:DWELl <time>

  [:SENSe]:METer:DETector:DWELl?

- **Functional Description**

  Set the dwell time for the meter.

  <time>: Dwell time with the default unit in second (s). The value range is from 1 ms to 100s.

- **Return Format**

  The query returns the dwell time of the meter in scientific notation, with the unit in second (s).

- **For Example**

  :METer:DETector:DWELl 0.5 Set the dwell time of the meter to 500 ms.

  :METer:DETector:DWELl? The query returns 5.000000e-01.

[:SENSe]:METer:PHOLd:ADJustable

- **Command Format**

  [:SENSe]:METer:PHOLd:ADJustable <time>

[:SENSe]:METer:PHOLd:ADJustable?

- **Functional Description**

  Set the maximum hold time for the meter.

  <time>: The maximum hold time with the default unit in second (s). The value range is from 500 to 100s.

- **Return Format**

  The query returns the maximum hold time of the meter in scientific notation, with the unit in second (s).

- **For Example**

:METer:PHOLd:ADJustable 1	Set the maximum hold time of the meter to 1s.
:METer:PHOLd:ADJustable?	The query returns 1.000000e+00.

[:SENSe]:METer:PHOLd:RESet

- **Command Format**

  [:SENSe]:METer:PHOLd:RESet

- **Functional Description**

  Reset the maximum hold for the meter.

- **Return Format**

  No return value.

- **For Example**

:METer:PHOLd:RESet	Reset the maximum hold for the meter.

[:SENSe]:METer:PHOLd:TYPE

- **Command Format**

  [:SENSe]:METer:PHOLd:TYPE { ADJustable| INFinite }

  [:SENSe]:METer:PHOLd:TYPE?

- **Functional Description**

  Select the maximum hold type for the meter.

  INFinite: Infinite, the periodicity refreshes the maximum value.

  ADJustable: Adjustable, the maximum hold time is interval, the periodicity refreshes the maximum value.

- **Return Format**

  The query returns the maximum hold type of the meter, ADJustable or INFinite.

- **For Example**

  :METer:PHOLd:TYPE ADJustable

Select the maximu hold type of the meter to ADJustable.

:METer:PHOLd:TYPE? The query returns ADJustable.

[:SENSe]:METer<n>:DETector

- **Command Format**

  [:SENSe]:METer<n>:DETector {POSitive|QPEak|CAVerage|AVERage|NEGative}

  [:SENSe]:METer<n>:DETector?

- **Functional Description**

  Select the detector type for the specified meter.

  <n>: Serial number of the specified meter, continuous integer, the value range is 1-3.

  POSitive: Peak detection

  QPEak: Quasi-peak detection

  CAVerage: EMI Averaged detection

  AVERage: Averaged detection

  NEGative: Negative peak detection

- **Return Format**

  The query returns the detector type of the specified meter: POSitive, QPEak, CAVerage, AVERage, or NEGative.

- **For Example**

  :METer1:DETector POSitive Select the detector type of the meter 1 to POSitive.

  :METer1:DETector? The query returns POSitive.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

  Set the input attenuation for the meter.

  <ampl>: Continuous integer, the default unit is dB. The value range is from 0 dB to 51 dB.

- **Return Format**

  The query returns input attenuation of the meter value. The unit is dB.

- **For Example**

  :POWer:ATTenuation 10 Set input attenuation of the meter to 10 dB.

  :POWer:ATTenuation? The query returns 10.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

  The pre-amplifier switch of the meter.

- **Return Format**

  The query returns the status of pre-amplifier switch of the meter: or 1.

- **For Example**

  :POWer:GAIN:STATe ON Turn on the pre-amplifier of the meter.

  :POWer:GAIN:STATe? The query returns 1.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

  [:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

  [:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

  Set the frequency reference.

  INTernal: Internal

  EXTernal: External

- **Return Format**

  The query returns the frequency reference.

- **For Example**

  :ROSCillator:SOURce:TYPE INTernal Set the frequency reference to INTernal.

  :ROSCillator:SOURce:TYPE? The query returns INTernal.

[:SENSe]:SLISt:COUPle:METer

- **Command Format**

  [:SENSe]:SLISt:COUPle:METer {{1|ON} | {0|OFF}}

  [:SENSe]:SLISt:COUPle:METer?

- **Functional Description**

  Set whether the meter couples with the signal list switch.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

The query returns the status of whether the meter is coupled to the signal list switch: 0 or 1.

- **For Example**

:SLISt:COUPle:METer ON	Set the meter to couple with the signal list switch.
:SLISt:COUPle:METer?	The query returns 1.

TRACe Command

:TRACe[:DATA]

- **Command Format**

  :TRACe[:DATA] TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<data>

  :TRACe[:DATA]? TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6

- **Functional Description**

  Set and acquire the trace data block. The number of data points corresponds to the sweep points, with each data value representing the power at a frequency point between the start frequency and the stop frequency. The data format is set using the command ":FORMat[:TRACe][:DATA]", and can be ASCII characters, 32-bit, or 64-bit binary real numbers. The default is ASCII, with data values separated by commas. Binary real numbers are converted from floating point data according to the IEEE 754 standard.

  TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6: Trace sequence character, indicating the trace 1|2|3|4|5|6.

  The trace value varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns trace amplitude data. The ASCII format represents data in scientific notation and returns it as a character stream, while the binary format returns data in the corresponding bit pattern, with data returned in Data Block Format. The default unit is dBm.

- **For Example**

  Set the data for trace 1.

  :TRACe:DATA TRACE1,-59,-77,-60,-98,-59,-59,-77,-60,-98,-59,-59

  :TRACe:DATA? TRACE1

  The query returns

  -5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01

:TRACe:FSCan:SELect

- **Command Format**

  :TRACe:FSCan:SELect <integer>

  :TRACe:FSCan:SELect?

- **Functional Description**

  Select a current trace from the sequence of the trace.

  <integer>: Serial number of the trace, integer, the value range is 1-3.

- **Return Format**

  The query returns the current serial number of the trace.

- **For Example**

  :TRACe:FSCan:SELect 1 Set trace 1 to the current trace.

  :TRACe:FSCan:SELect? The query returns 1.

:TRACe<n>:FSCan:DISPlay[:STATe]

- **Command Format**

  :TRACe<n>:FSCan:DISPlay[:STATe] {{1|ON} | {0|OFF}}

  :TRACe<n>:FSCan:DISPlay[:STATe]?

- **Functional Description**

  Set the display switch for the specified trace.

  <n>: Serial number of the trace, which is the integer. The value range is 1-3.

- **Return Format**

  The query returns the display switch status of the specified trace, 0 or 1.

- **For Example**

  :TRACe1:FSCan:DISPlay ON Trace 1 is displayed.

  :TRACe1:FSCan:DISPlay? The query returns 1.

:TRACe<n>:FSCan:TYPE

- **Command Format**

  :TRACe<n>:FSCan:TYPE <WRITe|AVERage|MAXHold|MINHold>

  :TRACe<n>:FSCan:TYPE?

- **Functional Description**

  Select the trace type for the specified trace.

  <n>: Serial number of the trace, which is the integer. The value range is 1-3.

  WRITe: Refresh

  AVERage: Averaged trace

MAXHold: The maximum hold

MINHold: The minimum hold

- **Return Format**

The query returns the specified trace type: WRITe, AVERage, MAXHold, or MINHold.

- **For Example**

:TRACe1:FSCan:TYPE AVERage	Set trace 1 to AVERage.
:TRACe1:FSCan:TYPE?	The query returns AVERage.

:TRACe<n>:FSCan:UPDate:STATe

- **Command Format**

:TRACe<n>:FSCan:UPDate:STATe {{1|ON} | {0|OFF}}

:TRACe<n>:FSCan:UPDate:STATe?

- **Functional Description**

The refresh switch of the specified trace, after turning on the refresh switch, the trace will keep refreshing.

<n>: Serial number of the trace, which is the integer. The value range is 1-3.

- **Return Format**

The query returns the refresh switch status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:FSCan:UPDate:STATe ON	Turn on the refresh switch of trace 1.
:TRACe1:FSCan:UPDate:STATe?	The query returns 1.

TRIGger Command

:TRIGger:FSCan[:SEQuence]:SOURce

- **Command Format**

:TRIGger:FSCan[:SEQuence]:SOURce {IMMediate|VIDeo|EXTernal1|EXTernal2|FRAMe}

:TRIGger:FSCan[:SEQuence]:SOURce?

- **Functional Description**

Select the trigger mode

IMMediate: Free to trigger

VIDeo: Video trigger

EXTernal1: External trigger 1

EXTernal2: External trigger 2; currently, only the UTS5026A supports this feature.

FRAMe: Period trigger; currently, only the UTS5026A supports this feature.

- **Return Format**

  The query returns the trigger mode: IMMediate, VIDeo, EXTernal1, EXTernal2, or FRAMe.

- **For Example**

  :TRIGger:FSCan:SOURce IMMediate Select the trigger mode toIMMediate.

  :TRIGger:FSCan:SOURce? The query returns IMMediate.

:TRIGger[:SEQuence]:EXTernal:DELay

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal:DELay <time>

  :TRIGger[:SEQuence]:EXTernal:DELay?

- **Functional Description**

  Set the trigger delay for the external trigger.

  <time>: A continuous positive number with the default unit in s. The value range is from 1000 ps to 500 ms.

- **Return Format**

  The query returns the trigger delay of the external trigger in scientific notation, with the unit in second (s).

- **For Example**

  :TRIGger:EXTernal:DELay 0.01 Set the trigger delay of the external trigger to 10 ms.

  :TRIGger:EXTernal:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:EXTernal:SLOPe?

- **Functional Description**

  Select the trigger edge for the external trigger.

  POSitive: Rising edge

  POSitive: Falling edge

- **Return Format**

  The query returns the trigger edge: external trigger, POSitive, or NEGative.

- **For Example**

  :TRIGger:EXTernal:SLOPe POSitive

  Select the trigger edge for the external trigger to POSitive.

  :TRIGger:EXTernal:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:EXTernal1:DELay

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:DELay <time>

  :TRIGger[:SEQuence]:EXTernal1:DELay?

- **Functional Description**

  Set the trigger delay for the external trigger 1.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the external trigger 1 in scientific notation, with the unit in second (s).

- **For Example**

  :TRIGger:EXTernal1:DELay 0.01 Set the trigger delay for the external trigger from 1 to 10 ms.

  :TRIGger:EXTernal1:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:EXTernal1:DELay:STATe?

- **Functional Description**

  Control the trigger delay switch for the external trigger 1, setting it to on or off.

  1|ON: ON

  1|ON: OFF

- **Return Format**

  The query returns the trigger delay switch status of the external trigger 1: 0 or 1.

- **For Example**

  :TRIGger:EXTernal1:DELay:STATe OFF

  Disable the switch of trigger delay of the external trigger 1.

  :TRIGger:EXTernal1:DELay:STATe? The query returns 0.

:TRIGger[:SEQuence]:EXTernal1:LEVel

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:LEVel <voltage>

  :TRIGger[:SEQuence]:EXTernal1:LEVel?

- **Functional Description**

Set the trigger level for the external trigger 1. Currently, only the UTS5026A supports this feature.

<voltage>: Voltage value, the default unit is V. The range is -3.3 V to 3.3 V.

- **Return Format**

The query returns the trigger level of the external trigger 1 in scientific notation

- **For Example**

:TRIGger:EXTernal:LEVel 1	Set the trigger level of the external trigger 1 to 1 V.
:TRIGger:EXTernal:LEVel?	The query returns 1.000000e+00.

:TRIGger[:SEQuence]:EXTernal1:SLOPe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:SLOPe {POSitive|NEGative}

:TRIGger[:SEQuence]:EXTernal1:SLOPe?

- **Functional Description**

Select the trigger edge of the external trigger 1.

POSitive: Rising edge

POSitive: Falling edge

- **Return Format**

The query returns the trigger edge of the external trigger 1, POSitive or NEGative.

- **For Example**

:TRIGger:EXTernal1:SLOPe POSitive

Select the trigger edge for the external trigger 1 to POSitive.

:TRIGger:EXTernal1:SLOPe?	The query returns POSitive.

:TRIGger[:SEQuence]:EXTernal2:DELay

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:DELay <time>

:TRIGger[:SEQuence]:EXTernal2:DELay?

- **Functional Description**

Set the trigger delay for the external trigger 2.

<time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

The query returns the trigger delay of the external trigger 2 in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:EXTernal2:DELay 0.01	Set the trigger delay of the external trigger 2 to 10 ms.
:TRIGger:EXTernal2:DELay?	The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe {{1|ON} | {0|OFF}}

:TRIGger[:SEQuence]:EXTernal2:DELay:STATe?

- **Functional Description**

Control the trigger delay switch for external trigger 2, setting it to on or off.

1|ON: ON

1|ON: OFF

- **Return Format**

The query returns the trigger delay switch status of the external trigger 2: 0 or 1.

- **For Example**

:TRIGger:EXTernal2:DELay:STATe OFF	Disable the trigger delay of the external trigger 2.
:TRIGger:EXTernal2:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:EXTernal2:LEVel

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:LEVel <voltage>

:TRIGger[:SEQuence]:EXTernal2:LEVel?

- **Functional Description**

Set the trigger level for the external trigger 2. Currently, only the UTS5026A supports this feature.

<voltage>: Voltage value, the default unit is V. The range is -3.3 V to 3.3 V.

- **Return Format**

The query returns the trigger level of the external trigger 2 in scientific notation.

- **For Example**

:TRIGger:EXTernal2:LEVel 1	Set the trigger level of the external trigger 2 to 1 V.
:TRIGger:EXTernal2:LEVel?	The query returns 1.000000e+00.

:TRIGger[:SEQuence]:EXTernal2:SLOPe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal2:SLOPe {POSitive|NEGative}

:TRIGger[:SEQuence]:EXTernal2:SLOPe?

- **Functional Description**

  Select the trigger edge of the external trigger 2.

  POSitive: Rising edge

  POSitive: Falling edge

- **Return Format**

  The query returns the trigger edge of the external trigger 2: POSitive, or NEGative.

- **For Example**

  :TRIGger:EXTernal2:SLOPe POSitive

  Select the trigger edge of the external trigger 2 to POSitive.

  :TRIGger:EXTernal2:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:FRAMe:DELay

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:DELay <time>

  :TRIGger[:SEQuence]:FRAMe:DELay?

- **Functional Description**

  Set the trigger delay for the period trigger. Currently, only the UTS5026A supports this feature.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

  :TRIGger:FRAMe:DELay 0.01 Set trigger delay of the period trigger to 10 ms.

  :TRIGger:FRAMe:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:FRAMe:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:FRAMe:DELay:STATe?

- **Functional Description**

  Control the trigger delay switch for the period trigger, setting it to on or off. Currently, only the UTS5026A supports this feature.

  1|ON: ON

1|ON: OFF

- **Return Format**

  The query returns the switch state of trigger delay of the period trigger: 0 or 1.

- **For Example**

:TRIGger:FRAMe:DELay:STATe OFF	Disable the trigger delay of the period trigger.
:TRIGger:FRAMe:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:FRAMe:OFFSet

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:OFFSet <time>

  :TRIGger[:SEQuence]:FRAMe:OFFSet?

- **Functional Description**

  Set the offset for the period trigger. Currently, only the UTS5026A supports this feature.

  <time>: A continuous positive number with the default unit in s. The range is 0s to 10s.

- **Return Format**

  The query returns the offset of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:FRAMe:OFFSet 1	Set the offset of the period trigger to 1s.
:TRIGger:FRAMe:OFFSet?	The query returns 1.000000e+00.

:TRIGger[:SEQuence]:FRAMe:PERiod

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:PERiod <time>

  :TRIGger[:SEQuence]:FRAMe:PERiod?

- **Functional Description**

  Set the period for the period trigger. Currently, only the UTS5026A supports this feature.

  <time>: A continuous positive number with the default unit in s. The range is 100 ns to 559 ms.

- **Return Format**

  The query returns the period of the period trigger in scientific notation, with the unit in second (s).

- **For Example**

:TRIGger:FRAMe:PERiod 0.03	Set the period of the period trigger to 30 ms.
:TRIGger:FRAMe:PERiod?	The query returns 3.000000e-02.

:TRIGger[:SEQuence]:FRAMe:SYNC

- **Command Format**

  :TRIGger[:SEQuence]:FRAMe:SYNC {OFF|EXTernal1|EXTernal2}

  :TRIGger[:SEQuence]:FRAMe:SYNC?

- **Functional Description**

  Select the synchronous source of the period trigger. Currently, only the UTS5026A supports this feature.

  OFF

  EXTernal1: External 1

  EXTernal2: External 2

- **Return Format**

  The query returns the synchronous source of the period trigger: OFF, EXTernal1, or EXTernal2.

- **For Example**

  :TRIGger:FRAMe:SYNC EXTernal1

  Select the synchronous source of the period trigger to EXTernal1.

  :TRIGger:FRAMe:SYNC? The query returns "EXTernal1".

:TRIGger[:SEQuence]:VIDeo:DELay

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:DELay <time>

  :TRIGger[:SEQuence]:VIDeo:DELay?

- **Functional Description**

  Set the trigger delay for the video trigger.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the video trigger in scientific notation, with the unit in seconds (s).

- **For Example**

  :TRIGger:VIDeo:DELay 0.01 Set the trigger delay of the video trigger to 10 ms.

  :TRIGger:VIDeo:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:VIDeo:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:DELay:STATe {{1|ON} | {0|OFF}}

:TRIGger[:SEQuence]:VIDeo:DELay:STATe?

- **Functional Description**

  Set the trigger delay switch for the external trigger 2.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the trigger delay switch status of the external trigger 2: 0 or 1.

- **For Example**

:TRIGger:VIDeo:DELay:STATe OFF	Disable the trigger delay for the video trigger.
:TRIGger:VIDeo:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:VIDeo:LEVel

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:LEVel <ampl>

  :TRIGger[:SEQuence]:VIDeo:LEVel?

- **Functional Description**

  Set the trigger level for the video trigger.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the trigger edge of the external trigger in scientific notation, with the unit in dBm.

- **For Example**

:TRIGger:VIDeo:LEVel -50	Set the trigger level of video trigger to -50 dBm.
:TRIGger:VIDeo:LEVel?	The query returns -5.000000e+01.

:TRIGger[:SEQuence]:VIDeo:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:VIDeo:SLOPe?

- **Functional Description**

  Select the trigger edge for the video trigger. Currently, only the UTS5026A supports this feature.

  POSitive: Rising edge

  POSitive: Falling edge

- **Return Format**

The query returns the trigger edge of the video trigger: POSitive or NEGative.

- **For Example**

:TRIGger:VIDeo:SLOPe POSitive

Select the trigger edge for the video trigger to POSitive.

:TRIGger:VIDeo:SLOPe? The query returns POSitive.

UNIT Command

:UNIT:POWer

- **Command Format**

:UNIT:POWer {DBM|DBMV|DBUV|V|W}

:UNIT:POWer?

- **Functional Description**

Select the scale unit for Y-axis.

- **Return Format**

The query returns the scale unit of Y-axis, DBM, DBMV, DBUV, V, or W.

- **For Example**

:UNIT:POWer DBM Select the scale unit of Y-axis to DBM.

:UNIT:POWer? The query returns DBM.

Analog Demodulation

CALCulate Command

:CALCulate:AM|FM|PM:MARKer:AOFF

- **Command Format**

:CALCulate:AM|FM|PM:MARKer:AOFF

- **Functional Description**

Disable all the markers in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

- **Return Format**

No return value.

- **For Example**

:CALCulate:AM:MARKer:AOFF

Disable all the markers in different analog demodulation mode.

:CALCulate:AM|FM|PM:MARKer:SELect

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer:SELect <integer>

  :CALCulate:AM|FM|PM:MARKer:SELect?

- **Functional Description**

  Select a marker as the current marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <integer>: Serial number of markers, a continuous integer with a value range of 1 to 6.

- **Return Format**

  The query returns the serial number of the current marker, with a value range of 1 to 6.

- **For Example**

  :CALCulate:AM:MARKer:SELect 1

  Select marker 1 to the current marker in amplitude modulation mode.

  :CALCulate:AM:MARKer:SELect? The query returns 1.

:CALCulate:AM|FM|PM:MARKer:TABLe[:STATe]

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer:TABLe[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:AM|FM|PM:MARKer:TABLe[:STATe]?

- **Functional Description**

  Select to display the marker list switch in different analog demodulation modes.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  1|ON: Display

  1|ON: Not display

- **Return Format**

  The query returns the display status of the marker list: 0 or 1.

- **For Example**

  :CALCulate:AM:MARKer:TABLe:STATe ON

  Display the marker list in amplitude modulation mode.

:CALCulate:AM:MARKer:TABLe:STATe? The query returns 1.

:CALCulate:AM|FM|PM:MARKer<n>:MAXimum

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:MAXimum

- **Functional Description**

  Execute a peak search for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <n>: Serial number of the marker, a continuous integer. The value range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:AM:MARKer1:MAXimum

  Execute a peak search for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:AM:MARKer1:MAXimum:LEFT

  Execute a next-peak search to the left side for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next-peak search for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:AM:MARKer1:MAXimum:NEXT

  Execute a next-peak search for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:AM:MARKer1:MAXimum:RIGHt

  Execute a next-peak search to the right side for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:MINimum

- **Functional Description**

  Execute the minimum peak search for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

No return value.

- **For Example**

:CALCulate:AM:MARKer1:MINimum

Execute the minimum peak search for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:MODE

- **Command Format**

:CALCulate:AM|FM|PM:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

:CALCulate:AM|FM|PM:MARKer<n>:MODE?

- **Functional Description**

Select the mode for the specified marker in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

OFF: Disable the marker function

POSition: Normal mode

DELTa: Difference mode

FIXed: Fixed mode

- **Return Format**

The query returns the mode of the specified marker: <time>: Time OFF, POSition, DELTa, or FIXed.

- **For Example**

:CALCulate:AM:MARKer1:MODE POSition

Select marker 1 in amplitude modulation mode to POSition.

:CALCulate:AM:MARKer1:MODE? The query returns POSition.

:CALCulate:AM|FM|PM:MARKer<n>:PTPeak

- **Command Format**

:CALCulate:AM|FM|PM:MARKer<n>:PTPeak

- **Functional Description**

Execute peak-to-peak search for the specified marker in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

- **Return Format**

No return value.

- **For Example**

:CALCulate:AM:MARKer1:PTPeak

Execute peak-to-peak search for marker 1 in amplitude modulation mode.

:CALCulate:AM|FM|PM:MARKer<n>:REFerence

- **Command Format**

:CALCulate:AM|FM|PM:MARKer<n>:REFerence <integer>

:CALCulate:AM|FM|PM:MARKer<n>:REFerence?

- **Functional Description**

Select the reference marker for the specified marker in different analog demodulation mode.

The reference marker cannot be set as its own reference.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

<integer>: Serial number of another marker, excluding the current one. The value range is 1 to 6.

- **Return Format**

The query returns the reference marker of the specified marker.

- **For Example**

:CALCulate:AM:MARKer1:REFerence 2

Set mark 2 in amplitude modulation mode as the reference marker for Marker 1.

:CALCulate:AM:MARKer1:REFerence? The query returns 2.

:CALCulate:AM|FM|PM:MARKer<n>:TRACe

- **Command Format**

:CALCulate:AM|FM|PM:MARKer<n>:TRACe

{RFSPectrum|DEMod|DAVerage|DMAXimum|DMINimum|AFSPectrum}

:CALCulate:AM|FM|PM:MARKer<n>:TRACe?

- **Functional Description**

Select the trace for the specified marker in different analog demodulation mode.

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

RFSPectrum: Radio-frequency spectrum trace

DEMod: Demodulation trace

DAVerage: Demodulation averaged trace

DMAXimum: The maximum hold trace of demodulation

DMINimum: The minimum hold trace of demodulation

AFSPectrum: Modulated signal spectrum trace

- **Return Format**

The query returns the trace of the specified marker:

RFSPectrum, DEMod, DAVerage, DMAXimum, DMINimum, or AFSPectrum.

- **For Example**

:CALCulate:AM:MARKer1:TRACe DEMod

Select marker 1 in amplitude modulation mode to DEMod.

:CALCulate:AM:MARKer1:TRACe? The query returns DEMod.

:CALCulate:AM|FM|PM:MARKer<n>:X

- **Command Format**

:CALCulate:AM|FM|PM:MARKer<n>:X <freq>|<time>

:CALCulate:AM|FM|PM:MARKer<n>:X?

- **Functional Description**

Set the reading of X-axis for the specified marker in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<n>: Marker serial number, a continuous integer with a value range of 1 to 6.

<freq>|<time>: The X-axis reading represents either frequency or time, depending on the trace corresponding to the specified marker. When the corresponding trace of the specified marker is the RF spectrum or the modulated signal spectrum, the X-axis reading is a frequency value, with the default unit in Hz. When the corresponding trace is demodulation, demodulation

average, maximum hold of demodulation, or minimum hold of demodulation, the X-axis reading is a time value, with the default unit in seconds (s).

- **Return Format**

  The query returns the coordinate value of X-axis for the specified marker in scientific notation. When the corresponding trace of the specified marker is the RF spectrum or the modulated signal spectrum, the X-axis reading is a frequency value, with the default unit in Hz. When the corresponding trace is demodulation, demodulation average, maximum hold of demodulation, or minimum hold of demodulation, the X-axis reading is a time value, with the default unit in seconds (s).

- **For Example**

  :CALCulate:AM:MARKer1:X 60MHz

  Select X-axis of marker 1 in amplitude modulation mode to 60 MHz.

  :CALCulate:AM:MARKer1:X? The query returns 6.000000e+07.

:CALCulate:AM|FM|PM:MARKer<n>:Y

- **Command Format**

  :CALCulate:AM|FM|PM:MARKer<n>:Y <ampl>|<freq>|<depth>

  :CALCulate:AM|FM|PM:MARKer<n>:Y?

- **Functional Description**

  Set the Y-axis reading for the specified marker in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <n>: Marker serial number, a continuous integer with a value range of 1 to 6.

  <ampl>|<freq>|<depth>: Depending on the demodulation mode, the marker corresponds to different traces. The Y-axis reading can be amplitude, frequency, or modulation depth percentage. When the marker's trace is the RF spectrum, in amplitude modulation and frequency modulation mode, the Y-axis reading is an amplitude value with a default unit of dBm. When the marker's trace is another type, in amplitude modulation mode, the Y-axis reading is the modulated depth as a percentage, with no unit (where 1 corresponds to 100%); in frequency modulation mode, the Y-axis reading is a frequency value, with a default unit of Hz.

- **Return Format**

  The query returns the Y-axis reading in scientific notation. When the marker's trace is the RF spectrum, in amplitude modulation and frequency modulation mode, the Y-axis reading is an amplitude value with a default unit of dBm. When the marker's trace is another type, in

amplitude modulation mode, the Y-axis reading is the modulated depth as a percentage, with no unit (where 1 corresponds to 100%); in frequency modulation mode, the Y-axis reading is a frequency value, with a default unit of Hz.

- **For Example**

  :CALCulate:AM:MARKer1:Y -50dBm

  Select the amplitude of marker 1 in amplitude modulation mode to -50 dBm.

  :CALCulate:AM:MARKer1:Y? The query returns -5.000000e+01.

CONFigure Command

:CONFigure

- **Command Format**

  :CONFigure?

- **Functional Description**

  Query the current demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

- **Return Format**

  The query returns the current demodulation mode, AM or FM.

- **For Example**

  :CONFigure? The query returns the current demodulation mode.

:CONFigure:AM

- **Command Format**

  :CONFigure:AM

- **Functional Description**

  Switch the demodulation mode to amplitude modulation. Restore the measurement parameter to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:AM Switch the demodulation mode to amplitude modulation.

:CONFigure:AM:NDEFault

- **Command Format**

:CONFigure:AM:NDEFault

- **Functional Description**

  Switch the demodulation mode to amplitude modulation.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:AM:NDEFault Switch the demodulation mode to amplitude modulation.

:CONFigure:FM

- **Command Format**

  :CONFigure:FM

- **Functional Description**

  Switch the demodulation mode to frequency modulation. Restore the measurement parameter to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:FM Switch the demodulation mode to frequency modulation.

:CONFigure:FM:NDEFault

- **Command Format**

  :CONFigure:FM:NDEFault

- **Functional Description**

  Switch the demodulation mode to frequency modulation.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:FM:NDEFault Switch the demodulation mode to frequency modulation.

:CONFigure:PM

- **Command Format**

  :CONFigure:PM

- **Functional Description**

  Switch the demodulation mode to phase modulation. Restore the measurement parameter to the default value.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:PM Switch the demodulation mode to phase modulation.

:CONFigure:PM:NDEFault

- **Command Format**

  :CONFigure:PM:NDEFault

- **Functional Description**

  Switch the demodulation mode to phase modulation.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:PM:NDEFault Switch the demodulation mode to phase modulation.

DISPlay Command

:DISPlay:DATA?

- **Command Format**

  :DISPlay:DATA?

- **Functional Description**

  Acquire the screen image.

- **Return Format**

  The query returns the screen image.

- **For Example**

  :DISPlay:DATA? Acquire the screen image.

:DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Set the scale of Y-axis for each measurement window in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

PM: Phase modulation

[1]: RF spectrum window, <real> is relative value, the default unit is dB.

2: Demodulation wave window, in amplitude modulation mode, <real> is modulated depth, the percentage value, no unit. 1 corresponds to 100%; in frequency modulation mode, <real> is frequency value, the unit is Hz; in phase modulation mode, <real> is radian value, the unit is rad.

3: Modulation frequency spectrum window, <real> is relative value, the default unit is dB.

- **Return Format**

The query returns the scale of Y-axis in scientific notation.

- **For Example**

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

Set the scale of Y-axis in RF spectrum window to 10 dB.

:DISPlay:AM:WINDow:TRACe:Y:PDIVision? The query returns 1.000000e+01.

:DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

:DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:RLEVel <real>

:DISPlay:AM|FM|PM:WINDow[1]|2|3:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

Set the reference level for each measurement window in a different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

[1]: RF spectrum window, <real> is amplitude value, the unit is dBm.

2: Demodulation wave window, in amplitude modulation mode, <real> is modulated depth, the percentage value, no unit. 1 corresponds to 100%; in frequency modulation mode, <real> is frequency value, the unit is Hz; in phase modulation mode, <real> is radian value, the unit is rad.

3: Modulation frequency spectrum window, in amplitude modulation mode, <real> is modulated depth, the percentage value, no unit. 1 corresponds to 100%; in frequency modulation mode, <real> is frequency value, the unit is Hz; in phase modulation mode, <real> is radian value, the unit is rad.

- **Return Format**

The query returns the reference level value in scientific notation.

- **For Example**

:DISPlay:AM:WINDow:TRACe:Y:RLEVel -10dBm

Set the reference level for RF spectrum window to -10 dBm.

:DISPlay:AM:WINDow:TRACe:Y:RLEVel? The query returns -1.000000e+01.

:DISPlay:AM|FM|PM:WINDow3:TRACe:Y[:SCALe]:SPACing

- **Command Format**

  :DISPlay:AM|FM|PM:WINDow3:TRACe:Y[:SCALe]:SPACing {LOGarithmic|LINear}

  :DISPlay:AM|FM|PM:WINDow3:TRACe:Y[:SCALe]:SPACing?

- **Functional Description**

  Set the scale mode of Y-axis for each measurement window in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  LOGarithmic: Logarithmic

  LINear: Linear

- **Return Format**

  The query returns the scale mode of Y-axis: LOGarithmic or LINear.

- **For Example**

  :DISPlay:AM:WINDow3:TRACe:Y:SPACing LOGarithmic

  Set the scale mode of Y-axis for RF spectrum window to LOGarithmic.

  :DISPlay:AM:WINDow3:TRACe:Y:SPACing? The query returns LOGarithmic.

FETCh Command

:FETCh:AM<n>?

- **Command Format**

  :FETCh:AM<n>?

- **Functional Description**

  Query trace data of analog demodulation AM.

  n: different trace data.

  0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

  1, Demodulation results including RF spectrum center frequency, carrier power, 0, modulation frequency, SNR (signal-to-noise ratio), modulation distortion, THD (total harmonic distortion), AM depth (positive peak), AM modulation depth (negative peak),

AM modulation depth ((Pk-Pk)/2), AM modulation depth (root mean square), AM modulation depth (positive peak) the maximum hold, AM modulation depth (negative peak) the maximum hold, AM modulation depth ((Pk-Pk)/2) the maximum hold and AM modulation depth (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents modulation depth in %.

- **Return Format**

The query returns analog demodulation AM trace data in scientific notation.

- **For Example**

:FETCh:AM0? AM RF spectrum trace data.

:FETCh:FM<n>?

- **Command Format**

:FETCh:FM<n>?

- **Functional Description**

Query trace data of analog demodulation FM.

n: different trace data.

0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

1, Demodulation results including RF spectrum center frequency, carrier power, carrier frequency offset, 0, modulation frequency, SNR (signal-to-noise ratio), modulation

distortion, THD (total harmonic distortion), FM frequency offset (positive peak), FM frequency offset (negative peak), FM frequency offset ((Pk-Pk)/2), FM frequency offset (root mean square), FM frequency offset (positive peak) the maximum hold, FM frequency offset (negative peak) the maximum hold, FM frequency offset ((Pk-Pk)/2) the maximum hold and FM frequency offset (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. The X-axis represents frequency in Hz. The Y-axis represents frequency in Hz.

- **Return Format**

The query returns analog demodulation FM trace data, with the data presented in scientific notation.

- **For Example**

:FETCh:FM0? FM RF spectrum trace data.

:FETCh:PM<n>?

- **Command Format**

:FETCh:PM<n>?

- **Functional Description**

Query trace data of analog demodulation PM.

n: different trace data.

0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

1, Demodulation results including RF spectrum center frequency, carrier power, carrier frequency offset, 0, modulation frequency, SNR (signal-to-noise ratio), modulation distortion, THD (total harmonic distortion), PM radian (positive peak), PM radian (negative peak), PM radian ((Pk-Pk)/2), PM radian (root mean square), PM radian (positive peak) the maximum hold, PM radian (negative peak) the maximum hold, PM radian ((Pk-Pk)/2) the maximum hold and PM radian (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. X-axis represents frequency and the unit is Hz. Y-axis represents radian and the unit is rad.

- **Return Format**

  The query returns analog demodulation PM trace data, with the data presented in scientific notation.

- **For Example**

  :FETCh:PM0? PM RF spectrum trace data.

INITiate Command

:INITiate:CONTinuous

- **Command Format**

  :INITiate:CONTinuous {{1|ON} | {0|OFF}}

  :INITiate:CONTinuous?

- **Functional Description**

Switch between single sweep and continuous sweep mode.

1|ON: Continuous sweep

1|ON: Single sweep

- **Return Format**

The query returns whether the mode is continuous sweep: 0 or 1.

- **For Example**

:INITiate:CONTinuous ON	Continuous sweep mode.
:INITiate:CONTinuous?	The query returns 1.

INPut Command

:INPut:MIXer?

- **Command Format**

:INPut:MIXer?

- **Functional Description**

Query the frequency reference of device, EXTernal or INTernal.

- **Return Format**

The query returns the frequency reference of device, EXTernal or INTernal.

- **For Example**

:INPut:MIXer?	Query returns INTernal.

MMEMory Command

:MMEMory:LOAD:STATe

- **Command Format**

:MMEMory:LOAD:STATe <filename>

- **Functional Description**

Load the register status data.

<filename>: The file name and the file suffix is “.state”.

- **Return Format**

No return value.

For Example

:MMEMory:LOAD:STATe “test.state”

Load the register status data from the file “test.state”.

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

  Save the register status to the file.

  <filename>: The file name and the file suffix is ".state".

- **Return Format**

  No return value.

  For Example

  :MMEMory:STORe:STATe "test.state" Save the register status to the file "test.state".

READ Command

The subsystem commands :READ and :FETCh are used to acquire measured results. The key difference between these commands is that the :FETCh command immediately retrieves the measured result, while the :READ command initiates a measurement, waits for it to complete, and then returns the result. If the measurement time exceeds the timeout period, the :READ command may fail to return the results due to a timeout.

:READ:AM<n>?

- **Command Format**

  :READ:AM<n>?

- **Functional Description**

  Query trace data of analog demodulation AM.

  n: different trace data.

  0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

  1, Demodulation results including RF spectrum center frequency, carrier power, 0, modulation frequency, SNR (signal-to-noise ratio), modulation distortion, THD (total harmonic distortion), AM depth (positive peak), AM modulation depth (negative peak), AM modulation depth ((Pk-Pk)/2), AM modulation depth (root mean square), AM modulation depth (positive peak) the maximum hold, AM modulation depth (negative peak) the maximum hold, AM modulation depth ((Pk-Pk)/2) the maximum hold and AM modulation depth (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time with the unit in second (s). Y-axis represents AM demodulation depth in %.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

- **Return Format**

The query returns analog demodulation AM trace data, and the data is returned in scientific notation.

- **For Example**

:READ:AM0? Query and return AM RF spectrum trace data.

:READ:FM<n>?

- **Command Format**

:READ:FM<n>?

- **Functional Description**

Query trace data of analog demodulation FM.

n: different trace data.

0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

1, Demodulation results including RF spectrum center frequency, carrier power, carrier frequency offset, 0, modulation frequency, SNR (signal-to-noise ratio), modulation distortion, THD (total harmonic distortion), FM frequency offset (positive peak), FM frequency offset (negative peak), FM frequency offset ((Pk-Pk)/2), FM frequency offset (root mean square), FM frequency offset (positive peak) the maximum hold, FM

frequency offset (negative peak) the maximum hold, FM frequency offset ((Pk-Pk)/2) the maximum hold and FM frequency offset (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). The Y-axis represents FM frequency offset in Hz.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. X-axis represents frequency and the unit is Hz. The Y-axis represents frequency in Hz.

- **Return Format**

The query returns analog demodulation FM trace data, and the data is returned in scientific notation.

- **For Example**

:READ:FM0? Query and return FM RF spectrum trace data.

:READ:PM<n>?

- **Command Format**

:READ:PM<n>?

- **Functional Description**

Query trace data of analog demodulation PM.

n: different trace data.

0, RF spectrum trace, data is represented as a list of points on the X and Y coordinates. X-axis represents frequency with the unit in Hz. Y-axis represents amplitude with the unit in dBm.

1, Demodulation results including RF spectrum center frequency, carrier power, carrier frequency offset, 0, modulation frequency, SNR (signal-to-noise ratio), modulation distortion, THD (total harmonic distortion), PM radian (positive peak), PM radian

(negative peak), PM radian ((Pk-Pk)/2), PM radian (root mean square), PM radian (positive peak) the maximum hold, PM radian (negative peak) the maximum hold, PM radian ((Pk-Pk)/2) the maximum hold and PM radian (root mean square) the maximum hold.

2, The minimum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

3, The maximum hold trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

4, Trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

5, Average trace data of demodulation, data is represented as a list of points on the X and Y coordinates. X-axis represents time and the unit is second (s). Y-axis represents PM radian and the unit is rad.

6, Trace data of modulation signal spectrum (AF Spectrum), data is represented as a list of points on the X and Y coordinates. X-axis represents frequency and the unit is Hz. Y-axis represents PM radian and the unit is rad.

- **Return Format**

The query returns analog demodulation PM trace data, and the data is returned in scientific notation.

- **For Example**

:READ:PM0? Query and return PM RF spectrum trace data.

SENSe Command

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth

- **Command Format**

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth <freq>

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth?

- **Functional Description**

Set the resolution bandwidth for modulation spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<freq>: Discrete real number the default unit is Hz, the value range is from 100 Hz to 1 MHz, it stepped with order 1-1.5-2-3-5-7.5-10.

- **Return Format**

The query returns the resolution bandwidth value in scientific notation, with the unit in Hz.

- **For Example**

:AM:AFSPectrum:BANDwidth 1MHz

Set the resolution bandwidth for modulation spectrum to 1 MHz.

:AM:AFSPectrum:BANDwidth? The query returns 1.000000e+06.

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth:AUTO

- **Command Format**

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:AM|FM|PM:AFSPectrum:BANDwidth:AUTO?

- **Functional Description**

Automatically/manually switch the resolution bandwidth for modulation spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of the resolution bandwidth, 0 or 1.

- **For Example**

:AM:AFSPectrum:BANDwidth:AUTO ON

Set the resolution bandwidth for modulation spectrum in amplitude modulation mode to AUTO.

:AM:AFSPectrum:BANDwidth:AUTO? The query returns 1.

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STARt

- **Command Format**

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STARt <freq>

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STARt?

- **Functional Description**

Set the start frequency for the modulation spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<freq>: A continuous real number with the unit in Hz. The start frequency range is from 0 Hz to 100 MHz-100 Hz. 100 Hz is minimum sweep bandwidth.

- **Return Format**

The query returns the start frequency value in scientific notation, with the unit in Hz.

- **For Example**

:AM:AFSPectrum:FREQuency:STARt 10MHz

Set the start frequency for the modulation spectrum in amplitude modulation mode to 10 MHz.

:AM:AFSPectrum:FREQuency:STARt? The query returns 1.000000e+07.

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STOP

- **Command Format**

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STOP <freq>

[:SENSe]:AM|FM|PM:AFSPectrum:FREQuency:STOP?

- **Functional Description**

Set the stop frequency for the modulation spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<freq>: A continuous real number with the unit in Hz. The stop frequency range is from 100 Hz to 100 MHz.

- **Return Format**

The query returns stop frequency value in scientific notation, with the unit in Hz.

- **For Example**

:AM:AFSPectrum:FREQuency:STOP 20MHz

Set the stop frequency for the modulation spectrum in amplitude modulation mode to 20 MHz.

:AM:AFSPectrum:FREQuency:STOP? The query returns 2.000000e+07.

[:SENSe]:AM|FM|PM:AVERage:COUNt

- **Command Format**

[:SENSe]:AM|FM|PM:AVERage:COUNt <integer>

[:SENSe]:AM|FM|PM:AVERage:COUNt?

- **Functional Description**

Set the average number in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

< integer > : The average number, continuous integer The value range is from 1 to 10000.

- **Return Format**

The query returns the average number.

- **For Example**

:AM:AVERage:COUNt 100	Set the average in amplitude modulation mode to 100.
:AM:AVERage:COUNt?	The query returns 100.

[:SENSe]:AM|FM|PM:AVERage[:STATe]

- **Command Format**

[:SENSe]:AM|FM|PM:AVERage[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:AM|FM|PM:AVERage[:STATe]?

- **Functional Description**

The average switch in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

1|ON: OFF

0|OFF: OFF

- **Return Format**

The query returns the status of the sweep time, 0 or 1.

- **For Example**

:AM:AVERage ON	Turn on the average in amplitude modulation mode.
:AM:AVERage?	The query returns 1.

[:SENSe]:AM|FM|PM:BANDwidth:CHANnel

- **Command Format**

[:SENSe]:AM|FM|PM:BANDwidth:CHANnel <freq>

[:SENSe]:AM|FM|PM:BANDwidth:CHANnel?

- **Functional Description**

Set the demodulated bandwidth in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<freq>: Discrete real number, the default unit is Hz, the value range is from 400 Hz to 25 MHz, 400 Hz to 10 MHz. 400 Hz to 25 MHz is stepped with order 1-1.5-2-3-5-7.5-10. 400 Hz to 10 MHz is stepped with order 5 MHz.

- **Return Format**

The query returns demodulated bandwidth value in scientific notation, with the unit in Hz.

- **For Example**

:AM:BANDwidth:CHANnel 1MHz

Set the demodulated bandwidth amplitude modulation mode to 1 MHz.

:AM:BANDwidth:CHANnel? The query returns 1.000000e+06.

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution]

- **Command Format**

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution] <freq>

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution]?

- **Functional Description**

Set the resolution bandwidth for RF spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

<freq>: Discrete real number, the default unit is Hz, the value range is from 100 Hz to 1 MHz, it stepped with order 1-1.5-2-3-5-7.5-10.

- **Return Format**

The query returns the resolution bandwidth value in scientific notation, with the unit in Hz.

- **For Example**

:AM:BANDwidth 1MHz

Set the resolution bandwidth in amplitude modulation mode to 1 MHz.

:AM:BANDwidth? The query returns 1.000000e+06.

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution]:AUTO

- **Command Format**

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution]:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:AM|FM|PM:BANDwidth[:RESolution]:AUTO?

- **Functional Description**

Automatically/manually switch the resolution bandwidth for RF spectrum in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of the resolution bandwidth, 0 or 1.

- **For Example**

:AM:BANDwidth:AUTO ON

Set the resolution bandwidth for RF spectrum in amplitude modulation mode to AUTO.

:AM:BANDwidth:AUTO? The query returns 1.

[:SENSe]:AM|FM|PM:BPFilter

- **Command Format**

[:SENSe]:AM|FM|PM:BPFilter {OFF|AWEighting|CWEighting|CCIR1k|CCIR2k|CUNWeighting}

[:SENSe]:AM|FM|PM:BPFilter?

- **Functional Description**

Select the band-pass filter in different analog demodulation mode.

AM: Amplitude modulation

FM: Frequency modulation

PM: Phase modulation

OFF: OFF

AWEighting: A weighting

CWEighting: C weighting

CCIR1k: CCIR-1 weighting

CCIR2k: CCIR-2 weighting

CUNWeighting: CCIR weighting

- **Return Format**

The query returns the type of band-pass filter, OFF, AWEighting, CWEighting, CCIR1k, CCIR2k, or CUNWeighting.

- **For Example**

:AM:BPFilter AWEighting

Select AWEighting band-pass filter in amplitude modulation mode.

:AM:BPFilter? The query returns AWEighting.

[:SENSe]:AM|FM|PM:DEMod:TIME

- **Command Format**

  [:SENSe]:AM|FM|PM:DEMod:TIME <time>

  [:SENSe]:AM|FM|PM:DEMod:TIME?

- **Functional Description**

  Set the demodulation time in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <time>: Time value with the unit in second (s). The value range is from 1 μs to 500 ms.

- **Return Format**

  The query returns the demodulation time in scientific notation, with the unit in second (s).

- **For Example**

  :AM:DEMod:TIME 0.1 Set the demodulation time in amplitude modulation mode to 100 ms.

  :AM:DEMod:TIME? The query returns 1.000000e-01.

[:SENSe]:AM|FM|PM:DEMod:TIME:AUTO

- **Command Format**

  [:SENSe]:AM|FM|PM:DEMod:TIME:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:AM|FM|PM:DEMod:TIME:AUTO?

- **Functional Description**

  Automatically/manually switch the demodulation time in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the sweep time: 0 or 1.

- **For Example**

  :AM:DEMod:TIME:AUTO ON Automatic demodulation time in amplitude modulation mode.

  :AM:DEMod:TIME:AUTO? The query returns 1.

[:SENSe]:AM|FM|PM:DWSWeep:TIME

- **Command Format**

  [:SENSe]:AM|FM|PM:DWSWeep:TIME <time>

  [:SENSe]:AM|FM|PM:DWSWeep:TIME?

- **Functional Description**

  Set the sweep time for demodulated window in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <time>: Time value with the unit in second (s). The value range is from 1 μs to 500 ms,

- **Return Format**

  The query returns the sweep time for demodulated window in scientific notation, with the unit in second (s).

- **For Example**

  :AM:DWSWeep:TIME 0.1

  Set the sweep time for demodulated window in amplitude modulation mode to100 ms.

  :AM:DWSWeep:TIME? The query returns 1.000000e-01.

[:SENSe]:AM|FM|PM:FREQuency:SPAN

- **Command Format**

  [:SENSe]:AM|FM|PM:FREQuency:SPAN <freq>

  [:SENSe]:AM|FM|PM:FREQuency:SPAN?

- **Functional Description**

  Set sweep bandwidth for the RF spectrum in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  <freq>: A continuous real number with the unit in Hz. The sweep bandwidth range is from 100 Hz to 10MHz.

- **Return Format**

  The query returns sweep bandwidth value in scientific notation, with the unit in Hz.

- **For Example**

  :AM:FREQuency:SPAN 1MHz

  Set the sweep bandwidth for RF spectrum in amplitude modulation mode to 1 MHz.

  :AM:FREQuency:SPAN? The query returns 1.000000e+06.

[:SENSe]:AM|FM|PM:HPFilter

- **Command Format**

  [:SENSe]:AM|FM|PM:HPFilter {OFF|HPF20|HPF50|HPF300|HPF400}

  [:SENSe]:AM|FM|PM:HPFilter?

- **Functional Description**

  Select the high-pass filter in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  OFF: OFF

  HPF20: 20 Hz

  HPF50: 50 Hz

  HPF300: 300 Hz

  HPF400: 400 Hz

- **Return Format**

  The query returns the type of high-pass filter, OFF, HPF20, HPF50, HPF300 or HPF400.

- **For Example**

  :AM:HPFilter HPF50 Select HPF50 high-pass filter in amplitude modulation mode.

  :AM:HPFilter? The query returns HPF50.

[:SENSe]:AM|FM|PM:LPFilter

- **Command Format**

  [:SENSe]:AM|FM|PM:LPFilter {OFF|LPF300|LPF3K|LPF15K|LPF30K|LPF80K|LPF100K|LPF300K}

  [:SENSe]:AM|FM|PM:LPFilter?

- **Functional Description**

  Select the low-pass filter in different analog demodulation mode.

  AM: Amplitude modulation

  FM: Frequency modulation

  PM: Phase modulation

  OFF: OFF

  LPF300: 300 Hz

  LPF3K: 3 kHz

  LPF15K: 15 kHz

  LPF30K: 30 kHz

  LPF80K: 80 kHz

LPF100K: 100 kHz

LPF300K: 300 kHz

- **Return Format**

The query returns the type of low-pass filter:

OFF, LPF300, LPF3K, LPF15K, LPF30K, LPF80K, LPF100K, or LPF300K.

- **For Example**

:AM:LPFilter LPF3K	Select 3 kHz low-pass filter in amplitude modulation mode.
:AM:LPFilter?	The query returns LPF3K.

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

- **Command Format**

[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] {50|75}

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

- **Functional Description**

Select the input resistance, 50 Ω or 75 Ω.

- **Return Format**

The query returns the input resistance value: 50 or 75.

- **For Example**

:CORRection:IMPedance 50	Select the input resistance to 50 Ω.
:CORRection:IMPedance?	The query returns 50.

[:SENSe]:FREQuency:CENTer

- **Command Format**

[:SENSe]:FREQuency:CENTer <freq>

[:SENSe]:FREQuency:CENTer?

- **Functional Description**

Set the center frequency of the sweep in analog demodulation.

<freq>: A continuous real number. The default unit is Hz, with a frequency range of 50 Hz to the maximum frequency -50 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the center frequency value in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz	Set the center frequency of the sweep to 1 GHz.
:FREQuency:CENTer?	The query returns 1.000000e+09.

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

- **Command Format**

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

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

- **Functional Description**

  Automatically/manually switch the stepped center frequency in analog demodulation.

  1|ON: Automatic (AUTO)

  1|OFF: Manual

- **Return Format**

  The query returns the status of the stepped center frequency: 0 or 1.

- **For Example**

  :FREQuency:CENTer:STEP:AUTO ON Set the stepped center frequency to AUTO.

  :FREQuency:CENTer:STEP:AUTO? The query returns 1.

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

- **Command Format**

  [:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>

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

- **Functional Description**

  Set the stepped center frequency of the sweep in analog demodulation.

  <freq>: A continuous real number with the unit in Hz.

- **Return Format**

  The query returns the stepped center frequency value in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:CENTer:STEP 10MHz Set the stepped center frequency to 10 MHz.

  :FREQuency:CENTer:STEP? The query returns 1.000000e+07.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

  Set input attenuation.

  <ampl>: Continuous integer, the default unit is dB. The value range is from 0 dB to 51 dB.

- **Return Format**

The query returns input attenuation, the unit is dB.

- **For Example**

:POWer:ATTenuation 10	Set input attenuation value to 10 dB.
:POWer:ATTenuation?	The query returns 10.

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

- **Command Format**

[:SENSe]:POWer[:RF]:ATTenuation:AUTO {{1|ON} | {0|OFF}}

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

- **Functional Description**

Automatically/manually switch input attenuation.

1|ON: Automatic (AUTO)

1|OFF: Manual

- **Return Format**

The query returns the status of input attenuation: 0 or 1.

- **For Example**

:POWer:ATTenuation:AUTO ON	Set input attenuation value to AUTO.
:POWer:ATTenuation:AUTO?	The query returns 1.

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

- **Command Format**

[:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

The pre-amplifier switch.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the switch status of the pre-amplifier: 0 or 1.

- **For Example**

:POWer:GAIN:STATe ON	Turn on the pre-amplifier.
:POWer:GAIN:STATe?	The query returns 1.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

[:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

[:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

Set the frequency reference.

INTernal: Internal

EXTernal: External

- **Return Format**

The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

[:SENSe]:SPEaker[:STATe]

- **Command Format**

[:SENSe]:SPEaker[:STATe] {{1|ON} | {0|OFF}}

[:SENSe]:SPEaker[:STATe]?

- **Functional Description**

Set the audio switch.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the switch status of audio: 0 or 1.

- **For Example**

:SPEaker ON	Enable the audio.
:SPEaker?	The query returns 1.

IQ

CALCulate Command

:CALCulate:SPECtrum:MARKer:AOFF

- **Command Format**

:CALCulate:SPECtrum:MARKer:AOFF

- **Functional Description**

Disable all markers for the complex spectrum.

- **Return Format**

No return value.

- **For Example**

:CALCulate:SPECtrum:MARKer:AOFF Disable all markers.

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion

- **Command Format**

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion <ampl>

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion?

- **Functional Description**

Set the peak offset for the complex spectrum.

<ampl>: A continuous real number with the default unit in dB.

- **Return Format**

The query returns the peak offset in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion 20 Set the peak offset to 20 dB.

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion? The query returns 2.000000e+01.

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion:STATe

- **Command Format**

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion:STATe {{1|ON} | {0|OFF}}

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion:STATe?

- **Functional Description**

Automatically/manually switch the peak offset for the complex spectrum.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the peak offset status: 0 or 1.

- **For Example**

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion:STATe ON

Switch the peak offset switch to automatic (AUTO) mode.

:CALCulate:SPECtrum:MARKer:PEAK:EXCursion:STATe? The query returns 1.

:CALCulate:SPECtrum:MARKer:PEAK:THReshold

- **Command Format**

:CALCulate:SPECtrum:MARKer:PEAK:THReshold <ampl>

:CALCulate:SPECtrum:MARKer:PEAK:THReshold?

- **Functional Description**

  Set the peak threshold for the complex spectrum.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the peak threshold in scientific notation, with the unit in dBm.

- **For Example**

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold -20 Set the peak threshold to -20 dBm.

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold? The query returns -2.000000e+01.

:CALCulate:SPECtrum:MARKer:PEAK:THReshold:LINE[:STATe]

- **Command Format**

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:LINE[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:LINE[:STATe]?

- **Functional Description**

  Set the peak threshold line for the complex spectrum.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns status of the peak threshold line: 0 or 1.

- **For Example**

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:LINE ON

  The peak threshold line is displayed.

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:LINE? The query returns 1.

:CALCulate:SPECtrum:MARKer:PEAK:THReshold:STATe

- **Command Format**

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:STATe {{1|ON} | {0|OFF}}

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:STATe?

- **Functional Description**

  Automatically/manually switch the peak threshold for the complex spectrum.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the peak threshold: 0 or 1.

- **For Example**

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:STATe ON

  Switch the peak threshold to automatic (AUTO) mode.

  :CALCulate:SPECtrum:MARKer:PEAK:THReshold:STATe? The query returns 1.

:CALCulate:SPECtrum:MARKer:SELect

- **Command Format**

  :CALCulate:SPECtrum:MARKer:SELect <integer>

  :CALCulate:SPECtrum:MARKer:SELect?

- **Functional Description**

  Select a marker by its serial number to be the current marker for the complex spectrum.

  <integer>: Marker serial number; a continuous integer with a value range of 1 to 4.

- **Return Format**

  The query returns the marker serial number, which ranges from 1 to 4.

- **For Example**

  :CALCulate:SPECtrum:MARKer:SELect 1 Select marker 1 as the current marker.

  :CALCulate:SPECtrum:MARKer:SELect? The query returns 1.

:CALCulate:SPECtrum:MARKer<n>:CPSearch[:STATe]

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:CPSearch[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:SPECtrum:MARKer<n>:CPSearch[:STATe]?

- **Functional Description**

  Switch the continuous peak search for the specified marker in the complex spectrum to on or off.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of the continuous peak search for the specified marker: 0 or 1.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:CPSearch ON

  Turn on the continuous peak search for Marker 1.

  :CALCulate:SPECtrum:MARKer1:CPSearch? The query returns 1.

:CALCulate:SPECtrum:MARKer<n>:FUNCtion

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion {OFF|NOISe|BPOWer|BDENsity}

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion?

- **Functional Description**

  Select the marker function for the specified marker in the complex spectrum. This setting is only valid for traces in the frequency domain.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  OFF: Enable the marker function

  NOISe: Noise marker

  BPOWer: in-band power

  BDENsity: in-band density

- **Return Format**

  The query returns the marker function for the specified marker: OFF, NOISe, BPOWer, or BDENsity.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:FUNCtion NOISe

  Set the marker function of marker 1 to NOISe.

  :CALCulate:SPECtrum:MARKer1:FUNCtion? The query returns NOISe.

:CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:LEFT

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:LEFT <freq>

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:LEFT?

- **Functional Description**

  Adjust the left-edge frequency of the marker measurement bandwidth for the specified marker in the complex spectrum. This setting is only valid for traces in the frequency domain.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

- **Return Format**

  The query returns the left-edge frequency of the marker measurement bandwidth for the specified marker in scientific notation, with the unit in Hz.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:LEFT 5MHz

Set the left-edge frequency of the marker 1 bandwidth to 5 MHz.

:CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:LEFT? The query returns 5.000000e+06.

:CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:RIGHt

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:RIGHt <freq>

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:RIGHt?

- **Functional Description**

  Adjust the right-edge frequency of the marker measurement bandwidth for the specified marker in the complex spectrum. This setting is only valid for traces in the frequency domain.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

- **Return Format**

  The query returns the right-edge frequency of the marker measurement bandwidth for the specified marker in scientific notation. The unit is seconds (s).

- **For Example**

  :CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:RIGHt 5MHz

  Set the right-edge frequency of the marker 1 bandwidth to 5 MHz.

  :CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:RIGHt? The query returns 5.000000e+06.

:CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:SPAN

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:SPAN <freq>

  :CALCulate:SPECtrum:MARKer<n>:FUNCtion:BAND:SPAN?

- **Functional Description**

  Adjust the marker measurement bandwidth for the specified marker in the complex spectrum. This setting is only valid for traces in the frequency domain.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

- **Return Format**

  The query returns the marker measurement bandwidth of the specified marker in scientific notation. The unit is seconds (s).

- **For Example**

:CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:SPAN 10MHz

Set the marker bandwidth for marker 1 to 10 MHz.

:CALCulate:SPECtrum:MARKer1:FUNCtion:BAND:SPAN?

The query returns 1.000000e+07.

:CALCulate:SPECtrum:MARKer<n>:MAXimum

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MAXimum

- **Functional Description**

  Execute a peak search for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:MAXimum Execute a peak search for Marker 1.

:CALCulate:SPECtrum:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next-peak search for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:MAXimum:NEXT Execute a next-peak search for Marker 1.

:CALCulate:SPECtrum:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:MAXimum:LEFT

  Execute a next-peak search to the left side for Marker 1.

:CALCulate:SPECtrum:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search on the right side for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:MAXimum:RIGHt

  Execute a next-peak search on the right side for Marker 1.

:CALCulate:SPECtrum:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:MINimum Execute a minimum peak search for Marker 1.

:CALCulate:SPECtrum:MARKer<n>:MODE

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

  :CALCulate:SPECtrum:MARKer<n>:MODE?

- **Functional Description**

  Select the specified marker's mode in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  OFF: Turn off the marker function

POSition: Normal mode

DELTa: Difference mode

FIXed: Fixed mode

- **Return Format**

The query returns the specified marker's mode: OFF, POSition, DELTa, or FIXed.

- **For Example**

:CALCulate:SPECtrum:MARKer1:MODE POSition Set marker 1 mode to POSition.

:CALCulate:SPECtrum:MARKer1:MODE? The query returns POSition.

:CALCulate:SPECtrum:MARKer<n>:PTPeak

- **Command Format**

:CALCulate:SPECtrum:MARKer<n>:PTPeak

- **Functional Description**

Execute a peak-to-peak search for the specified marker in the complex spectrum.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

No return value.

- **For Example**

:CALCulate:SPECtrum:MARKer1:PTPeak Marker 1 performs a peak-to-peak search.

:CALCulate:SPECtrum:MARKer<n>:REFerence

- **Command Format**

:CALCulate:SPECtrum:MARKer<n>:REFerence <integer>

:CALCulate:SPECtrum:MARKer<n>:REFerence?

- **Functional Description**

Select the reference marker for the specified marker in the complex spectrum. The reference marker cannot be set as its own reference.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

<integer>: Serial number of another marker, excluding the current one. The value range is 1 to 4.

- **Return Format**

The query returns the reference marker of the specified marker.

- **For Example**

:CALCulate:SPECtrum:MARKer1:REFerence 2

Set marker 2 as the reference marker for Marker 1.

:CALCulate:SPECtrum:MARKer1:REFerence? The query returns 2.

:CALCulate:SPECtrum:MARKer<n>[:SET]:CENTer

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>[:SET]:CENTer

- **Functional Description**

  Set the center frequency to the specified marker frequency in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:CENTer Set the center frequency to marker 1 frequency.

:CALCulate:SPECtrum:MARKer<n>[:SET]:RLEVel

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>[:SET]:RLEVel

- **Functional Description**

  Set the reference level to the specified marker amplitude in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:RLEVel Set the reference level to marker 1 amplitude.

:CALCulate:SPECtrum:MARKer<n>:TRACe

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:TRACe { SPECtrum|ASPectrum|I|Q }

  :CALCulate:SPECtrum:MARKer<n>:TRACe?

- **Functional Description**

  Set the trace for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  SPECtrum: Spectrum trace

  ASPectrum: Average spectrum trace

  I: Real part trace

  Q: Imaginary trace

- **Return Format**

  The query returns the trace of the specified marker.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:TRACe SPECtrum

  Set marker 1 to mark the spectrum trace.

  :CALCulate:SPECtrum:MARKer1:TRACe? The query returns SPECtrum.

:CALCulate:SPECtrum:MARKer<n>:X

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:X <freq>|<time>

  :CALCulate:SPECtrum:MARKer<n>:X?

- **Functional Description**

  Set the coordinate value of the X-axis for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <freq>: The frequency can be set when the marker is in the frequency domain spectrum. The default unit is Hz.

  <time>: The time can be set when the marker is in the time domain spectrum. The default unit is seconds (s).

- **Return Format**

  The query returns the coordinate value of the X-axis for the specified marker in scientific notation, with the unit in Hz or seconds (s).

- **For Example**

  :CALCulate:SPECtrum:MARKer1:X 50MHz

  Set the coordinate value of the X-axis for marker 1 to 50 MHz.

  :CALCulate:SPECtrum:MARKer1:X? The query returns 1.000000e+09.

:CALCulate:SPECtrum:MARKer<n>:X:POSition

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:X:POSition <integer>

  :CALCulate:SPECtrum:MARKer<n>:X:POSition?

- **Functional Description**

  Set the position of the trace data for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <integer>: An integer value without a unit, ranging from 0 to the maximum trace value. The sweep points vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter

Table for Each Model.

- **Return Format**

  The query returns the position of the trace data for the specified marker.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:X:POSition 100

  Set marker 1 at the 100th point of the trace data.

  :CALCulate:SPECtrum:MARKer1:X:POSition? The query returns 100.

:CALCulate:SPECtrum:MARKer<n>:Y

- **Command Format**

  :CALCulate:SPECtrum:MARKer<n>:Y?

- **Functional Description**

  Set the coordinate value of the Y-axis for the specified marker in the complex spectrum.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  When the marker function is disabled, the query returns the amplitude of the specified marker.

  When enabled, the query returns the measured results of the specified marker.

- **Return Format**

  The query returns the amplitude the specified marker and the measured results of the marker function in scientific notation. The unit of amplitude for the marker in frequency domain is dBm; the unit of amplitude for the marker in time domain is V; the unit of noise marker is dBm/Hz; the unit of in-band power is dBm; the unit of in-band is density dBm/Hz.

- **For Example**

  :CALCulate:SPECtrum:MARKer1:Y?

  The query returns the coordinate value of the Y-axis for marker 1.

:CALCulate:WAVeform:MARKer:AOFF

- **Command Format**

  :CALCulate:WAVeform:MARKer:AOFF

- **Functional Description**

  Turn off all the markers in the IQ waveform.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer:AOFF Turn off all the markers.

:CALCulate:WAVeform:MARKer:PEAK:EXCursion

- **Command Format**

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion <ampl>

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion?

- **Functional Description**

  Adjust the peak offset for the IQ waveform.

  <ampl>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the peak offset in scientific notation, with the unit in dB.

- **For Example**

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion 20 Set the peak offset to 20 dB.

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion? The query returns 2.000000e+01.

:CALCulate:WAVeform:MARKer:PEAK:EXCursion:STATe

- **Command Format**

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion:STATe {{1|ON} | {0|OFF}}

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion:STATe?

- **Functional Description**

  Automatically/manually switch the peak offset for the IQ waveform.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the switch status of the peak offset: 0 or 1.

- **For Example**

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion:STATe ON

  Switch the peak offset to automatic (AUTO) mode.

  :CALCulate:WAVeform:MARKer:PEAK:EXCursion:STATe? The query returns 1.

:CALCulate:WAVeform:MARKer:PEAK:THReshold

- **Command Format**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold <ampl>

  :CALCulate:WAVeform:MARKer:PEAK:THReshold?

- **Functional Description**

  Set the peak threshold for the IQ waveform.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the peak threshold in scientific notation, with the unit in dBm.

- **For Example**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold -20 Set the peak threshold to -20 dBm.

  :CALCulate:WAVeform:MARKer:PEAK:THReshold? The query returns -2.000000e+01.

:CALCulate:WAVeform:MARKer:PEAK:THReshold:LINE[:STATe]

- **Command Format**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:LINE[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:LINE[:STATe]?

- **Functional Description**

  Set the peak threshold line switch for the IQ waveform.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display status of the threshold line: 0 or 1.

- **For Example**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:LINE ON

  Display the peak threshold line.

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:LINE? The query returns 1.

:CALCulate:WAVeform:MARKer:PEAK:THReshold:STATe

- **Command Format**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:STATe {{1|ON} | {0|OFF}}

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:STATe?

- **Functional Description**

  Automatically/manually switch the peak threshold for the IQ waveform.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the peak threshold: 0 or 1.

- **For Example**

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:STATe ON

  Switch the peak threshold to automatic (AUTO) mode.

  :CALCulate:WAVeform:MARKer:PEAK:THReshold:STATe? The query returns 1.

:CALCulate:WAVeform:MARKer:SELect

- **Command Format**

  :CALCulate:WAVeform:MARKer:SELect <integer>

  :CALCulate:WAVeform:MARKer:SELect?

- **Functional Description**

  Select a marker by its serial number to be the current marker for the IQ waveform.

  <integer>: Marker serial number; a continuous integer with a value range of 1 to 4.

- **Return Format**

  The query returns the marker serial number, which ranges from 1 to 4.

- **For Example**

  :CALCulate:WAVeform:MARKer:SELect 1 Select marker 1 as the current marker.

  :CALCulate:WAVeform:MARKer:SELect? The query returns 1.

:CALCulate:WAVeform:MARKer<n>:CPSearch[:STATe]

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:CPSearch[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:WAVeform:MARKer<n>:CPSearch[:STATe]?

- **Functional Description**

  Switch the continuous peak search for the specified marker in the IQ waveform to on or off.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of the continuous peak search for the specified marker: 0 or 1.

- **For Example**

  :CALCulate:WAVeform:MARKer1:CPSearch ON

  Turn on the continuous peak search for trace 1.

  :CALCulate:WAVeform:MARKer1:CPSearch? The query returns 1.

:CALCulate:WAVeform:MARKer<n>:FUNCtion

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:FUNCtion {OFF|NOISe|BPOWer|BDENsity}

  :CALCulate:WAVeform:MARKer<n>:FUNCtion?

- **Functional Description**

Select the marker function for the specified marker in the IQ waveform. This setting is only valid for traces in the frequency domain.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

OFF: Enable the marker function

NOISe: Noise marker

BPOWer: in-band power

BDENsity: in-band density

- **Return Format**

The query returns the marker function for the specified marker: OFF, NOISe, BPOWer, or BDENsity.

- **For Example**

:CALCulate:WAVeform:MARKer1:FUNCtion NOISe

Set the marker function for marker 1 to NOISe.

:CALCulate:WAVeform:MARKer1:FUNCtion? The query returns NOISe.

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:LEFT

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:LEFT <time>

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:LEFT?

- **Functional Description**

Adjust the left-edge time of the marker measurement bandwidth for the specified marker in the IQ waveform.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

<time>: Time value with the default unit in seconds (s).

- **Return Format**

The query returns the left-edge time of the marker measurement bandwidth for the specified marker in scientific notation. The unit is seconds (s).

- **For Example**

:CALCulate:WAVeform:MARKer1:FUNCtion:BAND:LEFT 1μs

Set the left-edge time for the marker 1 bandwidth to 1 μs.

:CALCulate:WAVeform:MARKer1:FUNCtion:BAND:LEFT?

The query returns 1.000000e-06.

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:RIGHt

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:RIGHt <time>

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:RIGHt?

- **Functional Description**

  Adjust the right-edge time of the marker measurement bandwidth for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <time>: Time value with the default unit in seconds (s).

- **Return Format**

  The query returns the right-edge time of the marker measurement bandwidth for the specified marker in scientific notation. The unit is seconds (s).

- **For Example**

  :CALCulate:WAVeform:MARKer1:FUNCtion:BAND:RIGHt 10us

  Set the right-edge time for the marker 1 bandwidth to 10 μs.

  :CALCulate:WAVeform:MARKer1:FUNCtion:BAND:RIGHt?

  The query returns 1.000000e-05.

:CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:SPAN

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:SPAN <time>

  :CALCulate:WAVeform:MARKer<n>:FUNCtion:BAND:SPAN?

- **Functional Description**

  Adjust the marker measurement bandwidth for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <time>: Time value with the default unit in seconds (s).

- **Return Format**

  The query returns the marker measurement bandwidth of the specified marker in scientific notation. The unit is seconds (s).

- **For Example**

  :CALCulate:WAVeform:MARKer1:FUNCtion:BAND:SPAN 10us

  Set the marker bandwidth for the marker 1 to 10 μs.

  :CALCulate:WAVeform:MARKer1:FUNCtion:BAND:SPAN?

  The query returns 1.000000e-05.

:CALCulate:WAVeform:MARKer<n>:MAXimum

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:MAXimum

- **Functional Description**

  Execute a peak search for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MAXimum Marker 1 performs a peak search.

:CALCulate:WAVeform:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MAXimum:NEXT Marker 1 performs a next-peak search.

:CALCulate:WAVeform:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MAXimum:LEFT

  Marker 1 performs a next-peak search to the left side.

:CALCulate:WAVeform:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MAXimum:RIGHt

  Marker 1 performs a next-peak search on the right side.

:CALCulate:WAVeform:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MINimum Marker 1 performs a minimum peak search.

:CALCulate:WAVeform:MARKer<n>:MODE

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

  :CALCulate:WAVeform:MARKer<n>:MODE?

- **Functional Description**

  Select the marker mode for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  OFF: Turn off the marker function

  POSition: Normal mode

  DELTa: Difference mode

  FIXed: Fixed mode

- **Return Format**

  The query returns the specified marker's mode: OFF, POSition, DELTa, or FIXed.

- **For Example**

  :CALCulate:WAVeform:MARKer1:MODE POSition Set marker 1 mode to POSition.

:CALCulate:WAVeform:MARKer1:MODE? The query returns POSition.

:CALCulate:WAVeform:MARKer<n>:PTPeak

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:PTPeak

- **Functional Description**

  Execute a peak-to-peak search for the specified marker in the IQ waveform.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:WAVeform:MARKer1:PTPeak Marker 1 performs a peak-to-peak search.

:CALCulate:WAVeform:MARKer<n>:REFerence

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:REFerence <integer>

  :CALCulate:WAVeform:MARKer<n>:REFerence?

- **Functional Description**

  Select the reference marker for the specified marker in the IQ waveform. The reference marker cannot be set as its own reference.

  <n>: Marker serial number, a continuous integer with a value range of 1 to 4.

  <integer>: Serial number of another marker, excluding the current one. The value range is 1 to 4.

- **Return Format**

  The query returns the reference marker of the specified marker.

- **For Example**

  :CALCulate:WAVeform:MARKer1:REFerence 2

  Set marker 2 as the reference marker for Marker 1.

  :CALCulate:WAVeform:MARKer1:REFerence? The query returns 2.

:CALCulate:WAVeform:MARKer<n>:TRACe

- **Command Format**

  :CALCulate:WAVeform:MARKer<n>:TRACe { RFENvelope|I|Q }

  :CALCulate:WAVeform:MARKer<n>:TRACe?

- **Functional Description**

Select the trace for the specified marker in the IQ waveform.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

RFENvelope: Rf pulse envelope trace

I: Real-part trace

Q: Imaginary part trace

- **Return Format**

The query returns the trace for the specified marker.

- **For Example**

:CALCulate:WAVeform:MARKer1:TRACe RFENvelope

Set marker 1 to mark the spectrum trace.

:CALCulate:WAVeform:MARKer1:TRACe? The query returns RFENvelope.

:CALCulate:WAVeform:MARKer<n>:X

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:X <time>

:CALCulate:WAVeform:MARKer<n>:X?

- **Functional Description**

Adjust the coordinate value of the X-axis for the specified marker in the IQ waveform.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

<time>: Time can be set when the marker is in the time domain spectrum. The default unit is seconds (s).

- **Return Format**

The query returns the coordinate value of the X-axis for the specified marker in scientific notation, with the unit in seconds (s).

- **For Example**

:CALCulate:WAVeform:MARKer1:X 10us

Set the coordinate value of the X-axis for marker 1 to 10 μs.

:CALCulate:WAVeform:MARKer1:X? The query returns 1.000000e-05.

:CALCulate:WAVeform:MARKer<n>:X:POSition

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:X:POSition <integer>

:CALCulate:WAVeform:MARKer<n>:X:POSition?

- **Functional Description**

Set the position of the trace data for the specified marker in the IQ waveform.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

<integer>: An integer value without a unit, ranging from 0 to the maximum trace value. The sweep points vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the position of the trace data for the specified marker.

- **For Example**

:CALCulate:WAVeform:MARKer1:X:POSition 100

Set marker 1 at the 100th point of the trace data.

:CALCulate:WAVeform:MARKer1:X:POSition? The query returns 100.

:CALCulate:WAVeform:MARKer<n>:Y

- **Command Format**

:CALCulate:WAVeform:MARKer<n>:Y?

- **Functional Description**

Query the coordinate value of the Y-axis for the specified marker in the IQ waveform.

<n>: Marker serial number, a continuous integer with a value range of 1 to 4.

When the marker function is disabled, the query returns the amplitude of the specified marker. When enabled, the query returns the measured results of the specified marker.

- **Return Format**

The query returns the amplitude of the specified marker and the measured results of the marker function in scientific notation. The unit of amplitude for the marker in the frequency domain is dBm; the unit of the noise marker is dBm/Hz; the unit of in-band power is dBm; the unit of in-band density is dBm/Hz.

- **For Example**

:CALCulate:WAVeform:MARKer1:Y?

The query returns the coordinate value of the Y-axis for marker 1.

CONFigure Command

:CONFigure

- **Command Format**

:CONFigure?

- **Functional Description**

Query the measurement type for the IQ mode.

- **Return Format**

  The query returns the measurement type of the IQ mode: SPECtrum or WAVeform.

- **For Example**

  :CONFigure? Query the measurement type for the IQ mode.

:CONFigure:SPECtrum

- **Command Format**

  :CONFigure:SPECtrum

- **Functional Description**

  Set the IQ mode to enter the complex spectrum measurement and restore the measurement/setting parameters to their preset values.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:SPECtrum Enter the complex spectrum measurement.

:CONFigure:SPECtrum:NDEFault

- **Command Format**

  :CONFigure:SPECtrum:NDEFault

- **Functional Description**

  Set the IQ mode to enter the complex spectrum measurement without changing the measurement/setting parameters.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:SPECtrum:NDEFault Enter the complex spectrum measurement.

:CONFigure:WAVeform

- **Command Format**

  :CONFigure:WAVeform

- **Functional Description**

  Set the IQ mode to enter the IQ waveform measurement and restore the measurement/setting parameters to their preset values.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:WAVeform Enter the IQ waveform measurement.

:CONFigure:WAVeform:NDEFault

- **Command Format**

  :CONFigure:WAVeform:NDEFault

- **Functional Description**

  Set the IQ mode to enter the IQ waveform measurement without changing the measurement/setting parameters.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:WAVeform:NDEFault Enter the IQ waveform measurement.

DISPlay Command

:DISPlay:DATA?

- **Command Format**

  :DISPlay:DATA?

- **Functional Description**

  Acquire the screen image.

- **Return Format**

  The query returns the screen image data.

- **For Example**

  :DISPlay:DATA? Acquire the screen image.

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale in the frequency spectrum window for the complex spectrum.

  <real>: A discrete real number with the default unit in dB. The range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the Y-axis scale in the frequency spectrum window in scientific notation, with

the unit in dB.

- **For Example**

:DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:PDIVision 1 Set the Y-axis scale to 1 dB.

:DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:PDIVision?

The query returns 1.000000e+00.

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel <real>

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

Adjust the reference level of the Y-axis in the frequency spectrum window for the complex spectrum.

<real>: A continuous real number with the default unit in dBm.

- **Return Format**

The query returns the reference level of the Y-axis in the frequency spectrum window in scientific notation, with the unit in dBm.

- **For Example**

:DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:RLEVel -10

Set the reference level of the Y-axis to -10 dBm.

:DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:RLEVel? The query returns -1.000000e+01.

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition

- **Command Format**

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition {TOP|CENTer|BOTTom}

:DISPlay:SPECtrum:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition?

- **Functional Description**

Adjust the reference position of the Y-axis in the frequency spectrum window for the complex spectrum.

TOP: TOP

CENTer: Center

BOTTom: Bottom

- **Return Format**

The query returns the Y-axis scale value in the frequency spectrum window: TOP, CENTer, or BOTTom.

- **For Example**

  :DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:RPOSition TOP

  Set the reference position for Y-axis to TOP.

  :DISPlay:SPECtrum:VIEW:WINDow:TRACe:Y:RPOSition? The query returns TOP.

:DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale in the IQ window for the complex spectrum.

  <real>: A discrete real number with the default unit in volts (V).

- **Return Format**

  The query returns the Y-axis scale in the IQ window in scientific notation, with the unit in volts (V).

- **For Example**

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:PDIVision 1mV Set the Y-axis scale to 1 mV.

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:PDIVision? The query returns 1.000000e-03.

:DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RLEVel <real>

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

  Adjust the reference level of the Y-axis in the IQ window for the complex spectrum.

  <real>: A continuous real number with the default unit in volts (V).

- **Return Format**

  The query returns the reference level of the Y-axis in the IQ window in scientific notation, with the unit in volts (V).

- **For Example**

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:RLEVel 0.1

  Set the reference level of the Y-axis to 0.1 V.

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:RLEVel? The query returns 1.000000e-01.

:DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RPOSition

- **Command Format**

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RPOSition {TOP|CENTer|BOTTom}

  :DISPlay:SPECtrum:VIEW[1]:WINDow2:TRACe:Y[:SCALe]:RPOSition?

- **Functional Description**

  Adjust the reference position of the Y-axis in the IQ window for the complex spectrum.

  TOP: TOP

  CENTer: Center

  BOTTom: Bottom

- **Return Format**

  The query returns the reference position of the Y-axis in the IQ window: TOP, CENTer, or BOTTom.

- **For Example**

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:RPOSition TOP

  Set the reference position for Y-axis to TOP.

  :DISPlay:SPECtrum:VIEW:WINDow2:TRACe:Y:RPOSition? The query returns TOP.

:DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale in the IQ waveform window for the IQ waveform.

  <real>: A discrete real number with the default unit in dB. The range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the Y-axis scale in the IQ waveform in scientific notation, with the unit in dB.

- **For Example**

  :DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:PDIVision 1 Set the Y-axis scale to 1 dB.

  :DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:PDIVision? The query returns 1.000000e+00.

:DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

  :DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel <real>

  :DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

Adjust the reference level of the Y-axis scale in the IQ waveform window for the IQ waveform.

<real>: A continuous real number with the default unit in dBm.

- **Return Format**

The query returns the reference level of the Y-axis scale in the IQ waveform window in scientific notation, with the unit in dBm.

- **For Example**

:DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:RLEVel -10 Set the reference level to -10 dBm.

:DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:RLEVel? The query returns -1.000000e+01.

:DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition

- **Command Format**

:DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition {TOP|CENTer|BOTTom}

:DISPlay:WAVeform:VIEW[1]:WINDow[1]:TRACe:Y[:SCALe]:RPOSition?

- **Functional Description**

Select the reference position of the Y-axis scale in the IQ waveform window for the IQ waveform.

TOP: TOP

CENTer: Center

BOTTom: Bottom

- **Return Format**

The query returns the reference position of the Y-axis scale in the IQ waveform window: TOP, CENTer, or BOTTom.

- **For Example**

:DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:RPOSition TOP

Set the reference position for Y-axis to TOP.

:DISPlay:WAVeform:VIEW:WINDow:TRACe:Y:RPOSition? The query returns TOP.

FETCh Command

:FETCh:SPECtrum<n>?

- **Command Format**

:FETCh:SPECtrum<n>?

- **Functional Description**

Query the measured results for the complex spectrum.

<n>: When n is 0, the query returns the IQ trace data of the time domain spectrum. The data

consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sweep points of the time domain spectrum.

When n is 1, the query returns a set of results for the frequency spectrum, including: peak amplitude, peak frequency, point number, start frequency, and the frequency step.

Additionally, it returns the following for the time domain spectrum: point number, start time, time step, IQ results (I is the real part, Q is the imaginary part. The query returns 1 if the value is complex and 0 if the value is real), and the total scanning time of the time domain spectrum (IQ).

Average time for the current data processing:

When n is 2, the query returns the trace logarithmic power of the time domain spectrum.

When n is 3, the query returns the IQ trace data of the time domain spectrum. The data consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sweep points of the time domain spectrum.

When n is 4, the query returns the trace logarithmic power of the frequency domain spectrum.

When n is 5, the query returns the trace logarithmic power of the time domain spectrum.

When n is 7, the query returns the average trace logarithmic power of the frequency domain spectrum.

- **Return Format**

The query returns the trace data of the complex spectrum in scientific notation.

- **For Example**

:FETCh:SPECtrum1? Query a group of the measured results for the complex spectrum.

:FETCh:WAVeform<n>?

- **Command Format**

:FETCh:WAVeform<n>?

- **Functional Description**

Query the measured results of the IQ waveform.

<n>: When n is 0, the query returns the unprocessed I/Q data. The data consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sampling points.

When n is 1, the query returns a set of results, including: sampling time of single point, active power, average active power (if the average function is enabled and the average time is greater than 1, it represents N active power; otherwise, it is equivalent to the active power), sampling

point, the ratio of maximum power to active power, maximum power, and the latest power. When n is 2, the query returns the trace data of the entire envelope. The number of groups corresponds to the sampling points. The interval is the sampling time of single point.

- **Return Format**

  The query returns the measured results of the IQ waveform in scientific notation.

- **For Example**

  :FETCh:WAVeform1? Query a group of the measured results for the IQ waveform.

INITiate Command

:INITiate:CONTinuous

- **Command Format**

  :INITiate:CONTinuous {{1|ON} | {0|OFF}}

  :INITiate:CONTinuous?

- **Functional Description**

  Switch between single sweep mode and continuous sweep mode.

  1|ON: Continuous sweep mode

  0|OFF: Single sweep mode

- **Return Format**

  The query returns the sweep mode: 0 or 1.

- **For Example**

  :INITiate:CONTinuous ON Continuous sweep mode.

  :INITiate:CONTinuous? The query returns 1.

MMEMory Command

:MMEMory:LOAD:STATe

- **Command Format**

  :MMEMory:LOAD:STATe <filename>

- **Functional Description**

  Load the register status data.

  <filename>: Filename with the ".state" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:STATe "test.state" Load the register status data from the file "test.state".

:MMEMory:LOAD:TRACe

- **Command Format**

  :MMEMory:LOAD:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Load the data from the specified trace.

  TRACE1-TRACE6: Corresponds to trace 1 to trace 6

  <filename>: Filename with the ".trace" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:TRACe TRACE1,"test.trace" Trace 1 loads the data from the file "test.trace".

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

  Save the register status to the specified file.

  <filename>: Filename with the ".state" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:STATe "test.state" Save the register status to the file "test.state".

:MMEMory:STORe:TRACe

- **Command Format**

  :MMEMory:STORe:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Save the specified trace data to the file.

  TRACE1-TRACE6: Corresponds to trace 1 to trace 6

  <filename>: Filename with the ".trace" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:TRACe TRACE1,"test.trace" Save trace 1 data to the file "test.trace".

READ Command

:READ:SPECtrum<n>?

- **Command Format**

  :READ:SPECtrum<n>?

- **Functional Description**

  Query the measured results for the complex spectrum.

  <n>: When n is 0, the query returns the IQ trace data of the time domain spectrum. The data consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sweep points of the time domain spectrum.

  When n is 1, the query returns a set of results for the frequency spectrum, including: peak amplitude, peak frequency, point number, start frequency, and the frequency step.

  Additionally, it returns the following for the time domain spectrum: point number, start time, time step, IQ results (I is the real part, Q is the imaginary part. The query returns 1 if the value is complex and 0 if the value is real), and the total scanning time of the time domain spectrum (IQ).

  When n is 2, the query returns the trace logarithmic power of the time domain spectrum.

  When n is 3, the query returns the IQ trace data of the time domain spectrum. The data consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sweep points of the time domain spectrum.

  When n is 4, the query returns the trace logarithmic power of the frequency domain spectrum.

  When n is 5, the query returns the trace logarithmic power of the time domain spectrum.

  When n is 7, the query returns the average trace logarithmic power of the frequency domain spectrum.

- **Return Format**

  The query returns the trace data of the complex spectrum in scientific notation.

- **For Example**

  :READ:SPECtrum<n>? Query a group of the measured results for the complex spectrum.

:READ:WAVeform<n>?

- **Command Format**

  :READ:WAVeform<n>?

- **Functional Description**

  Query the measured results of the IQ waveform.

<n>: When n is 0, the query returns the unprocessed I/Q data. The data consists of power values in volts (V). The data is organized into groups, each containing an I (real part) and a Q (imaginary part). The number of groups corresponds to the sampling points.

When n is 1, the query returns a set of results, including: sampling time of single point, active power, average active power (if the average function is enabled and the average time is greater than 1, it represents N active power; otherwise, it is equivalent to the active power), sampling point, the ratio of maximum power to active power, maximum power, and the latest power.

When n is 2, the query returns the trace data of the entire envelope. The number of groups corresponds to the sampling points. The interval is the sampling time of single point.

- **Return Format**

The query returns the measured results of the IQ waveform in scientific notation.

- **For Example**

:READ:WAVeform1? Query a group of the measured results for the IQ waveform.

SENSe Command

[:SENSe]:FREQuency:CENTer

- **Command Format**

[:SENSe]:FREQuency:CENTer <freq>

[:SENSe]:FREQuency:CENTer?

- **Functional Description**

Set the center frequency for the sweep frequency function.

<freq>: A continuous real number. The default unit is Hz, with a frequency range of 50 Hz to the maximum frequency -50 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the center frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz Set the center frequency of the sweep frequency to 1 GHz.

:FREQuency:CENTer? The query returns 1.000000e+09.

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

- **Command Format**

[:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

Adjust the input attenuation.

<ampl>: A continuous real number. The default unit is dB, with an amplitude range of 0 dB to 51 dB.

- **Return Format**

The query returns the input attenuation in dB.

- **For Example**

:POWer:ATTenuation 10	Adjust the input attenuation to 10 dB.
:POWer:ATTenuation?	The query returns 10.

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

- **Command Format**

[:SENSe]:POWer[:RF]:ATTenuation:AUTO {{1|ON} | {0|OFF}}

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

- **Functional Description**

Automatically/manually switch the input attenuation.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the input attenuation status: 0 or 1.

- **For Example**

:POWer:ATTenuation:AUTO ON	Switch input attenuation to automatic (AUTO) mode.
:POWer:ATTenuation:AUTO?	The query returns 1.

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

- **Command Format**

[:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

Switch the preamplifier to on or off.

- **Return Format**

The query returns the preamplifier status: 0 or 1.

- **For Example**

:POWer:GAIN:STATe ON	Turn on the preamplifier.
:POWer:GAIN:STATe?	The query returns 1.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

  [:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

  [:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

  Set the frequency reference.

  INTernal: Internal

  EXTernal: External

- **Return Format**

  The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

[:SENSe]:SPECtrum:AVERage[:STATe]

- **Command Format**

  [:SENSe]:SPECtrum:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:SPECtrum:AVERage[:STATe]?

- **Functional Description**

  Switch the average function of the complex spectrum to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the average function status: 0 or 1.

- **For Example**

:SPECtrum:AVERage ON	Turn on the average function.
:SPECtrum:AVERage?	The query returns 1.

[:SENSe]:SPECtrum:AVERage:COUNt

- **Command Format**

  [:SENSe]:SPECtrum:AVERage:COUNt <integer>

  [:SENSe]:SPECtrum:AVERage:COUNt?

- **Functional Description**

  Adjust the average time for the complex spectrum.

  <integer>: An integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average time, which ranges from 1 to 10000.

- **For Example**

:SPECtrum:AVERage:COUNt 10	Set the average time to 10.
:SPECtrum:AVERage:COUNt?	The query returns 10.

[:SENSe]:SPECtrum:AVERage:TCONtrol

- **Command Format**

  [:SENSe]:SPECtrum:AVERage:TCONtrol {EXPonential|REPeat}

  [:SENSe]:SPECtrum:AVERage:TCONtrol?

- **Functional Description**

  Set the average mode for the complex spectrum.

  EXPonential: Exponential

  REPeat: Repeat

- **Return Format**

  The query returns the average mode: EXPonential or REPeat.

- **For Example**

:SPECtrum:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:SPECtrum:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:SPECtrum:AVERage:TYPE

- **Command Format**

  [:SENSe]:SPECtrum:AVERage:TYPE {VOLTage|POWEr|LOG}

  [:SENSe]:SPECtrum:AVERage:TYPE?

- **Functional Description**

  Set the average type for the complex spectrum.

  VOLTage: Voltage average

  POWEr: Power average

  LOG: Logarithmic average

- **Return Format**

  The query returns the average type: VOLTage, POWEr, or LOG.

- **For Example**

:SPECtrum:AVERage:TYPE POWEr	Set the average type to POWEr.
:SPECtrum:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:SPECtrum:BANDwidth[:RESolution]

- **Command Format**

  [:SENSe]:SPECtrum:BANDwidth[:RESolution] <freq>

  [:SENSe]:SPECtrum:BANDwidth[:RESolution]?

- **Functional Description**

  Adjust the resolution bandwidth for the complex spectrum.

  <freq>: A discrete real number with a default unit of Hz. The value ranges from 1 Hz to the maximum of the resolution bandwidth, stepped in a 1-3-10 sequence. The maximum resolution bandwidth varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the resolution bandwidth in scientific notation, with the unit in Hz.

- **For Example**

:SPECtrum:BANDwidth 1MHz	Set the resolution bandwidth to 1 MHz.
:SPECtrum:BANDwidth?	The query returns 1.000000e+06.

[:SENSe]:SPECtrum:BANDwidth[:RESolution]:AUTO

- **Command Format**

  [:SENSe]:SPECtrum:BANDwidth[:RESolution]:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:SPECtrum:BANDwidth[:RESolution]:AUTO?

- **Functional Description**

  Automatically/manually switch the resolution bandwidth for the complex spectrum.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the switch status of the resolution bandwidth: 0 or 1.

- **For Example**

  :SPECtrum:BANDwidth:AUTO ON

  Switch the resolution bandwidth to automatic (AUTO) mode.

:SPECtrum:BANDwidth:AUTO?	The query returns 1.

[:SENSe]:SPECtrum:FFT:WINDow[:TYPE]

- **Command Format**

  [:SENSe]:SPECtrum:FFT:WINDow[:TYPE] {HANNing|FLATtop|GAUSsian|BLACkman|BH4Tap}

  [:SENSe]:SPECtrum:FFT:WINDow[:TYPE]?

- **Functional Description**

  Select the FFT window type in IQ mode for the complex spectrum.

  HANNing: Hanning

  FLATtop: Flat top

  GAUSsian: Gaussian

  BLACkman: Blackman

  BH4Tap: Blackman-harris

- **Return Format**

  The query returns FFT window type: HANNing, FLATtop, GAUSsian, BLACkman, or BH4Tap.

- **For Example**

:SPECtrum:FFT:WINDow GAUSsian	Select GAUSsian.
:SPECtrum:FFT:WINDow?	The query returns GAUSsian.

[:SENSe]:SPECtrum:FREQuency:SPAN

- **Command Format**

  [:SENSe]:SPECtrum:FREQuency:SPAN <freq>

  [:SENSe]:SPECtrum:FREQuency:SPAN?

- **Functional Description**

  Set the span for the complex spectrum.

  <freq>: A continuous real number with the default unit in Hz.

  The span range is from 100 Hz to 10 MHz.

- **Return Format**

  The query returns the span in scientific notation, with the unit in Hz.

- **For Example**

:SPECtrum:FREQuency:SPAN 1MHz	Set the span to 1 MHz.
:SPECtrum:FREQuency:SPAN?	The query returns 1.0000000000e+06.

[:SENSe]:WAVeform:AVERage[:STATe]

- **Command Format**

  [:SENSe]:WAVeform:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:WAVeform:AVERage[:STATe]?

- **Functional Description**

  Switch the average function of the IQ waveform to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the average status: 0 or 1.

- **For Example**

:WAVeform:AVERage ON	Turn on the average function.
:WAVeform:AVERage?	The query returns 1.

[:SENSe]:WAVeform:AVERage:COUNt

- **Command Format**

  [:SENSe]:WAVeform:AVERage:COUNt <integer>

  [:SENSe]:WAVeform:AVERage:COUNt?

- **Functional Description**

  Adjust the average time for the IQ waveform.

  <integer>: An integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average time, which ranges from 1 to 10000.

- **For Example**

:WAVeform:AVERage:COUNt 10	Set the average time to 10.
:WAVeform:AVERage:COUNt?	The query returns 10.

[:SENSe]:WAVeform:AVERage:TCONtrol

- **Command Format**

  [:SENSe]:WAVeform:AVERage:TCONtrol {EXPonential|REPeat}

  [:SENSe]:WAVeform:AVERage:TCONtrol?

- **Functional Description**

  Set the average mode for the IQ waveform.

  EXPonential: Exponential

  REPeat: Repeat

- **Return Format**

  The query returns the average mode: EXPonential or REPeat.

- **For Example**

:WAVeform:AVERage:TCONtrol EXPonential	Set the average mode to EXPonential.
:WAVeform:AVERage:TCONtrol?	The query returns EXPonential.

[:SENSe]:WAVeform:AVERage:TYPE

- **Command Format**

[:SENSe]:WAVeform:AVERage:TYPE {VOLTage|POWEr|LOG}

[:SENSe]:WAVeform:AVERage:TYPE?

- **Functional Description**

Set the average type for the IQ waveform.

VOLTage: Voltage average

POWEr: Power average

LOG: Logarithmic average

- **Return Format**

The query returns the average type: VOLTage, POWEr, or LOG.

- **For Example**

:WAVeform:AVERage:TYPE POWEr	Set the average type to POWEr.
:WAVeform:AVERage:TYPE?	The query returns POWEr.

[:SENSe]:WAVeform:DIF:BANDwidth

- **Command Format**

[:SENSe]:WAVeform:DIF:BANDwidth <freq>

[:SENSe]:WAVeform:DIF:BANDwidth?

- **Functional Description**

Adjust the resolution bandwidth for the IQ waveform.

<freq>: A discrete real number with a default unit of Hz. The value ranges from 1 Hz to the maximum of the resolution bandwidth, stepped in a 1-3-10 sequence. The maximum resolution bandwidth vary for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

The query returns the resolution bandwidth in scientific notation, with the unit in Hz.

- **For Example**

:WAVeform:DIF:BANDwidth 1MHz	Set the resolution bandwidth to 1 MHz.
:WAVeform:DIF:BANDwidth?	The query returns 1.000000e+06.

[:SENSe]:WAVeform:DIF:BANDwidth:AUTO

- **Command Format**

[:SENSe]:WAVeform:DIF:BANDwidth:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:WAVeform:DIF:BANDwidth:AUTO?

- **Functional Description**

Automatically/manually switch the resolution bandwidth for the IQ waveform.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the switch status of the resolution bandwidth: 0 or 1.

- **For Example**

:WAVeform:DIF:BANDwidth:AUTO ON

Switch the resolution bandwidth to automatic (AUTO) mode.

:WAVeform:DIF:BANDwidth:AUTO? The query returns 1.

[:SENSe]:WAVeform:DIF:FILTer:TYPE

- **Command Format**

[:SENSe]:WAVeform:DIF:FILTer:TYPE {HANNing|FLATtop|GAUSsian|BLACkman|BH4Tap}

[:SENSe]:WAVeform:DIF:FILTer:TYPE?

- **Functional Description**

Select the filter type for the IQ waveform.

HANNing: Hanning

FLATtop: Flat top

GAUSsian: Gaussian

BLACkman: Blackman

BH4Tap: Blackman-harris

- **Return Format**

The query returns the filter type for the IQ waveform: HANNing, FLATtop, GAUSsian, BLACkman, or BH4Tap.

- **For Example**

:WAVeform:DIF:FILTer:TYPE GAUSsian Select GAUSsian.

:WAVeform:DIF:FILTer:TYPE? The query returns GAUSsian.

TRACe Command

:TRACe:SPECtrum:STRace[:STATe]

- **Command Format**

:TRACe:SPECtrum:STRace[:STATe] {{1|ON} | {0|OFF}}

:TRACe:SPECtrum:STRace[:STATe]?

- **Functional Description**

Set display switch of spectrum trace in the frequency domain for the complex spectrum. IQ

frequency domain has two spectrum traces: spectrum trace and average spectrum. This command specifically controls the display of the spectrum trace, not the average spectrum.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the display status of spectrum trace line in the frequency domain: 0 or 1.

- **For Example**

:TRACe:SPECtrum:STRace OFF

The spectrum trace in the frequency domain is not displayed.

:TRACe:SPECtrum:STRace? The query returns 0.

TRIGger Command

:TRIGger[:SEQuence]:EXTernal1:DELay

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:DELay <time>

:TRIGger[:SEQuence]:EXTernal1:DELay?

- **Functional Description**

Set the trigger delay for the external trigger 1.

<time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

The query returns the trigger delay of the external trigger 1 in scientific notation, with the unit in seconds (s).

- **For Example**

:TRIGger:EXTernal1:DELay 0.01 Set the trigger delay for external trigger 1 to 10 ms.

:TRIGger:EXTernal1:DELay? The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe {{1|ON} | {0|OFF}}

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe?

- **Functional Description**

Switch the trigger delay switch of the external trigger 1 to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

  The query returns the trigger delay switch status of the external trigger 1: 0 or 1.

- **For Example**

  :TRIGger:EXTernal1:DELay:STATe OFF

  Turn off the trigger delay switch of the external trigger 1.

  :TRIGger:EXTernal1:DELay:STATe? The query returns 0.

:TRIGger[:SEQuence]:EXTernal1:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:EXTernal1:SLOPe?

- **Functional Description**

  Select the trigger edge for the external trigger 1.

  POSitive: Rising edge

  NEGative: Negative edge

- **Return Format**

  The query returns the trigger edge of the external trigger 1: POSitive or NEGative.

- **For Example**

  :TRIGger:EXTernal1:SLOPe POSitive

  Select the trigger edge for the external trigger 1 to POSitive.

  :TRIGger:EXTernal1:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:SOURce

- **Command Format**

  :TRIGger[:SEQuence]:SOURce {IMMediate|EXTernal1}

  :TRIGger[:SEQuence]:SOURce?

- **Functional Description**

  Select the trigger type.

  IMMediate: Free trigger

  EXTernal1: External trigger 1

- **Return Format**

  The query returns the trigger type: IMMediate, VIDeo, EXTernal1, EXTernal2, or FRAMe.

- **For Example**

  :TRIGger:SOURce IMMediate Select the trigger type to IMMediate.

:TRIGger:SOURce?	The query returns IMMediate.

Real-time Spectrum Analysis

CALCulate Command

:CALCulate:MARKer:AOFF

- **Command Format**

  :CALCulate:MARKer:AOFF

- **Functional Description**

  Turn off all the markers.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:MARKer:AOFF	Turn off all the markers.

:CALCulate:MARKer:PEAK:EXCursion

- **Command Format**

  :CALCulate:MARKer:PEAK:EXCursion <ampl>

  :CALCulate:MARKer:PEAK:EXCursion?

- **Functional Description**

  Adjust the peak offset.

  <ampl>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the peak offset in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:MARKer:PEAK:EXCursion 20	Set the peak offset to 20 dB.
:CALCulate:MARKer:PEAK:EXCursion?	The query returns 2.000000e+01.

:CALCulate:MARKer:PEAK:EXCursion:STATe

- **Command Format**

  :CALCulate:MARKer:PEAK:EXCursion:STATe {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:EXCursion:STATe?

- **Functional Description**

  Automatically/manually switch the peak offset.

  1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

  The query returns the switch status of the peak offset: 0 or 1.

- **For Example**

  :CALCulate:MARKer:PEAK:EXCursion:STATe ON

  Switch the peak offset to automatic (AUTO) mode.

  :CALCulate:MARKer:PEAK:EXCursion:STATe? The query returns 1.

:CALCulate:MARKer:PEAK:TABLe[:STATe]

- **Command Format**

  :CALCulate:MARKer:PEAK:TABLe[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:TABLe[:STATe]?

- **Functional Description**

  Set the display switch of the peak list to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display status of the peak list: 0 or 1.

- **For Example**

  :CALCulate:MARKer:PEAK:TABLe ON Display the peak list.

  :CALCulate:MARKer:PEAK:TABLe? The query returns 1.

:CALCulate:MARKer:PEAK:THReshold

- **Command Format**

  :CALCulate:MARKer:PEAK:THReshold <ampl>

  :CALCulate:MARKer:PEAK:THReshold?

- **Functional Description**

  Set the peak threshold.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the peak threshold in scientific notation, with the unit in dBm.

- **For Example**

  :CALCulate:MARKer:PEAK:THReshold -20 Set the peak threshold to -20 dBm.

  :CALCulate:MARKer:PEAK:THReshold? The query returns -2.000000e+01.

:CALCulate:MARKer:PEAK:THReshold:LINE[:STATe]

- **Command Format**

  :CALCulate:MARKer:PEAK:THReshold:LINE[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:THReshold:LINE[:STATe]?

- **Functional Description**

  Set the peak threshold line switch.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display status of the threshold line: 0 or 1.

- **For Example**

  :CALCulate:MARKer:PEAK:THReshold:LINE ON Display the peak threshold line.

  :CALCulate:MARKer:PEAK:THReshold:LINE? The query returns 1.

:CALCulate:MARKer:PEAK:THReshold:STATe

- **Command Format**

  :CALCulate:MARKer:PEAK:THReshold:STATe {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:PEAK:THReshold:STATe?

- **Functional Description**

  Automatically/manually switch the peak threshold.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the peak threshold: 0 or 1.

- **For Example**

  :CALCulate:MARKer:PEAK:THReshold:STATe ON

  Switch the peak threshold to automatic (AUTO) mode.

  :CALCulate:MARKer:PEAK:THReshold:STATe? The query returns 1.

:CALCulate:MARKer:SELect

- **Command Format**

  :CALCulate:MARKer:SELect <integer>

  :CALCulate:MARKer:SELect?

- **Functional Description**

  Select a marker by its serial number to be the current marker for the complex spectrum.

<integer>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  The query returns the marker serial number, which ranges from 1 to 10.

- **For Example**

:CALCulate:MARKer:SELect 1	Select marker 1 as the current marker.
:CALCulate:MARKer:SELect?	The query returns 1.

:CALCulate:MARKer:TABLe[:STATe]

- **Command Format**

  :CALCulate:MARKer:TABLe[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:TABLe[:STATe]?

- **Functional Description**

  Set the display switch of the marker list to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display status of the marker list: 0 or 1.

- **For Example**

:CALCulate:MARKer:TABLe ON	Display the peak list.
:CALCulate:MARKer:TABLe?	The query returns 1.

:CALCulate:MARKer<n>:BANDwidth|BWIDth[:STATe]

- **Command Format**

  :CALCulate:MARKer<n>:BANDwidth|BWIDth[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:BANDwidth|BWIDth[:STATe]?

- **Functional Description**

  Set the NDB switch for the specified marker. The marker of time domain trace is not supported.

  1|ON: ON

  0|OFF: OFF

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  The query returns the NDB switch status for the specified marker: 0 or 1.

- **For Example**

:CALCulate:MARKer1:BANDwidth ON	Turn on the NDB for marker 1.
:CALCulate:MARKer1:BANDwidth?	The query returns 1.

:CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB

- **Command Format**

  :CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB <real>

  :CALCulate:MARKer<n>:BANDwidth|BWIDth:NDB?

- **Functional Description**

  Set the NDB value for the specified marker. The marker of time domain trace is not support.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  <real>: A continuous real number; the default unit is dB, ranging from -0.01 to -140.

- **Return Format**

  The query returns the NDB value of the specified marker in scientific notation, with the unit in dB.

- **For Example**

:CALCulate:MARKer1:BANDwidth:NDB -3	Set the NDB value for marker 1 to -3 dB.
:CALCulate:MARKer1:BANDwidth:NDB?	The query returns -3.000000e+00.

:CALCulate:MARKer<n>:BANDwidth|BWIDth:RESult?

- **Command Format**

  :CALCulate:MARKer<n>:BANDwidth|BWIDth:RESult?

- **Functional Description**

  Set the measured NDB value for the specified marker. The marker of time domain trace is not support.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  The query returns the measured NDB value of the specified marker in scientific notation. The NDB unit is Hz.

- **For Example**

  :CALCulate:MARKer1:BANDwidth:RESult?

  The query returns the measured NDB value of marker 1.

:CALCulate:MARKer<n>:CPSearch[:STATe]

- **Command Format**

  :CALCulate:MARKer<n>:CPSearch[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:CPSearch[:STATe]?

- **Functional Description**

  Switch the continuous peak search for the specified marker to on or off.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the switch status of the continuous peak search for the specified marker: 0 or 1.

- **For Example**

:CALCulate:MARKer1:CPSearch ON	Turn on the continuous peak search for Marker 1.
:CALCulate:MARKer1:CPSearch?	The query returns 1.

:CALCulate:MARKer<n>:FUNCtion

- **Command Format**

:CALCulate:MARKer<n>:FUNCtion {OFF|NOISe|BPOWer|BDENsity}

:CALCulate:MARKer<n>:FUNCtion?

- **Functional Description**

Select the marker function for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

OFF: Enable the marker function

NOISe: Noise marker

BPOWer: in-band power

BDENsity: in-band density

- **Return Format**

The query returns the marker function for the specified marker: OFF, NOISe, BPOWer, or BDENsity.

- **For Example**

:CALCulate:MARKer1:FUNCtion NOISe	Set the marker function for marker 1 to NOISe.
:CALCulate:MARKer1:FUNCtion?	The query returns NOISe.

:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT

- **Command Format**

:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT <freq>|<time>

:CALCulate:MARKer<n>:FUNCtion:BAND:LEFT?

- **Functional Description**

Adjust the left-edge bandwidth for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

<time>: The time can be set when the X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

The query returns the left-edge bandwidth of the specified marker in scientific notation. The unit is Hz when the X-axis scale type is frequency or inverse time. The unit is seconds (s) when the X-axis scale type is period or time.

- **For Example**

:CALCulate:MARKer1:FUNCtion:BAND:LEFT 5MHz

Set the left-edge bandwidth for marker 1 to 5 MHz.

:CALCulate:MARKer1:FUNCtion:BAND:LEFT? The query returns 5.000000e+06.

:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt

- **Command Format**

:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt <freq>|<time>

:CALCulate:MARKer<n>:FUNCtion:BAND:RIGHt?

- **Functional Description**

Adjust the right-edge bandwidth for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

<time>: The time can be set when the X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

The query returns the right-edge bandwidth of the specified marker in scientific notation. The unit is Hz when the X-axis scale type is frequency or inverse time. The unit is seconds (s) when the X-axis scale type is period or time.

- **For Example**

:CALCulate:MARKer1:FUNCtion:BAND:RIGHt 5MHz

Set the right-edge bandwidth for marker 1 to 5 MHz.

:CALCulate:MARKer1:FUNCtion:BAND:RIGHt? The query returns 5.000000e+06.

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN

- **Command Format**

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN <freq>|<time>

:CALCulate:MARKer<n>:FUNCtion:BAND:SPAN?

- **Functional Description**

Adjust the marker bandwidth for the specified marker. The time domain window (PvT) does not support this setting.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<freq>: The frequency can be set when the X-axis scale type is either frequency or reversed. The default unit is Hz.

<time>: The time can be set when the X-axis scale type is period or time. The default unit is seconds (s).

- **Return Format**

The query returns the marker bandwidth of the specified marker in scientific notation. The unit is Hz when the X-axis scale type is frequency or inverse time. The unit is seconds (s) when the X-axis scale type is period or time.

- **For Example**

:CALCulate:MARKer1:FUNCtion:BAND:SPAN 10MHz

Set the marker bandwidth for marker 1 to 10 MHz.

:CALCulate:MARKer1:FUNCtion:BAND:SPAN? The query returns 1.000000e+07.

:CALCulate:MARKer<n>:LINes[:STATe]

- **Command Format**

:CALCulate:MARKer<n>:LINes[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer<n>:LINes[:STATe]?

- **Functional Description**

Set the switch for the specified marker line.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

The query returns the switch status of the specified marker line 0 or 1.

- **For Example**

:CALCulate:MARKer1:LINes ON Turn on the marker line for marker 1.

:CALCulate:MARKer1:LINes? The query returns 1.

:CALCulate:MARKer<n>:MAXimum[:MAX]

- **Command Format**

:CALCulate:MARKer<n>:MAXimum[:MAX]

- **Functional Description**

  Execute a peak search for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum Marker 1 performs a peak search.

:CALCulate:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum:NEXT Marker 1 performs a next-peak search.

:CALCulate:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:MAXimum:LEFT Marker 1 performs a next-peak search to the left side.

:CALCulate:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search on the right side for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:MAXimum:RIGHt

Marker 1 performs a next-peak search on the right side.

:CALCulate:MARKer<n>:MINimum

- **Command Format**

:CALCulate:MARKer<n>:MINimum

- **Functional Description**

Execute a minimum peak search for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:MINimum Marker 1 performs a minimum peak search.

:CALCulate:MARKer<n>:MODE

- **Command Format**

:CALCulate:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

:CALCulate:MARKer<n>:MODE?

- **Functional Description**

Select the marker mode for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

OFF: Turn off the marker function

POSition: Normal mode

DELTa: Difference mode

FIXed: Fixed mode

- **Return Format**

The query returns the specified marker's mode: OFF, POSition, DELTa, or FIXed.

- **For Example**

:CALCulate:MARKer1:MODE POSition Set marker 1 mode to POSition.

:CALCulate:MARKer1:MODE? The query returns POSition.

:CALCulate:MARKer<n>:PTPeak

- **Command Format**

  :CALCulate:MARKer<n>:PTPeak

- **Functional Description**

  Execute a peak-to-peak search for the specified marker

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:PTPeak Marker 1 performs a peak-to-peak search.

:CALCulate:MARKer<n>:REFerence

- **Command Format**

  :CALCulate:MARKer<n>:REFerence <integer>

  :CALCulate:MARKer<n>:REFerence?

- **Functional Description**

  Select the reference marker for the specified marker. The reference marker cannot be set as its own reference.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  <integer>: Serial number of another marker, excluding the current one. The value range is 1 to 10.

- **Return Format**

  The query returns the reference marker of the specified marker.

- **For Example**

  :CALCulate:MARKer1:REFerence 2 Set marker 2 as the reference marker for Marker 1.

  :CALCulate:MARKer1:REFerence? The query returns 2.

:CALCulate:MARKer<n>[:SET]:CENTer

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:CENTer

- **Functional Description**

  Set the center frequency to the frequency of the specified marker. Markers corresponding to the time domain trace are not supported.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:CENTer Set the center frequency to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:RLEVel

- **Command Format**

:CALCulate:MARKer<n>[:SET]:RLEVel

- **Functional Description**

Set the reference level to the amplitude of the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:RLEVel Set the reference level to marker 1 amplitude.

:CALCulate:MARKer<n>[:SET]:STEP

- **Command Format**

:CALCulate:MARKer<n>[:SET]:STEP

- **Functional Description**

Set the center frequency step to the frequency of the specified marker. Markers corresponding to the time domain trace are not supported.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:STEP Set the center frequency step to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:STARt

- **Command Format**

:CALCulate:MARKer<n>[:SET]:STARt

- **Functional Description**

Set the start frequency to the frequency of the specified marker. Markers corresponding to the time domain trace are not supported.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:STARt Set the start frequency to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:STOP

- **Command Format**

:CALCulate:MARKer<n>[:SET]:STOP

- **Functional Description**

Set the stop frequency to the frequency of the specified marker. Markers corresponding to the time domain trace are not supported.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:STOP Set the stop frequency to marker 1 frequency.

:CALCulate:MARKer<n>:TRACe

- **Command Format**

:CALCulate:MARKer<n>:TRACe <integer>

:CALCulate:MARKer<n>:TRACe?

- **Functional Description**

Select the trace for the specified marker.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<integer>: Trace sequence number, an integer ranging from 1 to 12. Traces 1 to 6 correspond to frequency domain traces, while traces 7 to 12 correspond to time domain traces. These traces correspond to the trace data within the frequency domain window and the time domain window (PvT), respectively.

- **Return Format**

The query returns the trace serial number of the specified marker.

- **For Example**

:CALCulate:MARKer1:TRACe 1 Set Marker 1 to correspond with Trace 1.

:CALCulate:MARKer1:TRACe? The query returns 1.

:CALCulate:MARKer<n>:X

- **Command Format**

:CALCulate:MARKer<n>:X <freq>|<time>

:CALCulate:MARKer<n>:X?

- **Functional Description**

Adjust the coordinate value of the X-axis for the specified marker and set the corresponding data type based on the X-axis scale type.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<freq>: Frequency can be set when the X-axis scale in the frequency domain trace is set to frequency or inverse time. The default unit is Hz. The marker in the time domain trace is not supported.

<time>: Time can be set when the X-axis scale in the frequency domain trace is set to period or time. The default unit is seconds (s). The time domain trace can only be set to time.

- **Return Format**

The query returns the coordinate value of the X-axis for the specified marker in scientific notation. The unit is Hz if the X-axis scale is set to frequency or inverse time. The unit is seconds (s) if the X-axis scale is set to period or time.

- **For Example**

:CALCulate:MARKer1:X 1GHz	Set the coordinate value of the X-axis for marker 1 to 1 GHz.
:CALCulate:MARKer1:X?	The query returns 1.000000e+09.

:CALCulate:MARKer<n>:X:READout

- **Command Format**

:CALCulate:MARKer<n>:X:READout {FREQuency|PERiod|TIME|ITIMe}

:CALCulate:MARKer<n>:X:READout?

- **Functional Description**

Select the X-axis scale type for the specified marker. Markers corresponding to the time domain trace are not supported.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

FREQuency: Frequency

PERiod: Period

TIME: Time

ITIMe: Inverse time

- **Return Format**

The query returns the X-axis scale type of the specified marker: FREQuency, PERiod, TIME, or ITIMe.

- **For Example**

:CALCulate:MARKer1:X:READout FREQuency

Select the X-axis scale type for marker 1 to FREQuency.

:CALCulate:MARKer1:X:READout? The query returns FREQuency.

:CALCulate:MARKer<n>:X:READout:AUTO

- **Command Format**

  :CALCulate:MARKer<n>:X:READout:AUTO {{1|ON} | {0|OFF}}

  :CALCulate:MARKer<n>:X:READout:AUTO?

- **Functional Description**

  Automatically/manually switch the X-axis scale type for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the X-axis scale type of the specified marker: 0 or 1.

- **For Example**

  :CALCulate:MARKer1:X:READout:AUTO ON

  Switch the X-axis scale type of marker 1 to automatic (AUTO) mode.

  :CALCulate:MARKer1:X:READout:AUTO? The query returns 1.

:CALCulate:MARKer<n>:Y

- **Command Format**

  :CALCulate:MARKer<n>:Y <ampl>

  :CALCulate:MARKer<n>:Y?

- **Functional Description**

  Adjust the amplitude for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  <ampl>: Amplitude of the marker, with the default unit in dBm.

  When the marker function is disabled, the command ":CALCulate:MARKer<n>:Y?" query returns the amplitude of the specified marker. When enabled, the command ":CALCulate:MARKer<n>:Y?" query returns the measured results of the specified marker.

- **Return Format**

  The query returns the amplitude of the specified marker and the measured results of the marker function in scientific notation. The unit of amplitude for the marker in the frequency domain is dBm; the unit of the noise marker is dBm/Hz; the unit of in-band power is dBm; the unit of

in-band density is dBm/Hz.

- **For Example**

:CALCulate:MARKer1:Y -50	Set the amplitude of marker 1 to -50 dBm.
:CALCulate:MARKer1:Y?	The query returns -5.000000e+01.

:CALCulate:MARKer<n>:Z:POSition

- **Command Format**

:CALCulate:MARKer<n>:Z:POSition <integer>

:CALCulate:MARKer<n>:Z:POSition?

- **Functional Description**

Set the trace serial number for the specified cursor in the spectrum window.

<integer>: Trace serial number; a continuous integer. The range is from 1 to 10000

- **Return Format**

The query returns the trace serial number for the specified cursor in the spectrum window.

- **For Example**

:CALCulate:MARKer1:Z:POSition 10	Set the trace for trace 1 in the spectrum window to 10.
:CALCulate:MARKer1:Z:POSition?	The query returns 10.

:CALCulate:MATH

- **Command Format**

:CALCulate:MATH {<trace>,<type>,<op1>,<op2>,<real>}

- **Functional Description**

Set the mathematical operation on the trace.

<trace>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6; Set the mathematical operation on the specified trace.

<type>: OFF|PDIFference|PSUM|LDIFference|LOFFset

OFF: Turn off the mathematical operation

PDIFference: Power difference

PSUM: Power sum

LDIFference: Logarithmic difference

LOFFset: Logarithmic offset

<op1>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6. Performs the operand A on the trace.

<op2>: TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6, Performs the operand B on the trace.

<real>: Mathematical operation offset in unit dB, ranging from -100 dB to 100 dB. This is only valid when the operation type is logarithmic difference and logarithmic offset.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MATH TRACE1,PDIFference,TRACE2,TRACE3,0

  Trace 1 performs a power difference operation; operand A represents Trace 2, operand B represents Trace 3 and offset is 0 dB.

DISPlay Command

:DISPlay:VIEW:DENSity:AADJust

- **Command Format**

  :DISPlay:VIEW:DENSity:AADJust

- **Functional Description**

  Automatically correct the hue for the density spectrum.

- **Return Format**

  No return value.

- **For Example**

  :DISPlay:VIEW:DENSity:AADJust Automatically correct the hue.

:DISPlay:VIEW:DENSity:CNONlinear

- **Command Format**

  :DISPlay:VIEW:DENSity:CNONlinear <real>

  :DISPlay:VIEW:DENSity:CNONlinear?

- **Functional Description**

  Set the hue surface curvature for the density spectrum.

  <real>: A continuous real number, with a value range of -100 to 100.

- **Return Format**

  The query returns the hue surface curvature of the density spectrum in scientific notation.

- **For Example**

  :DISPlay:VIEW:DENSity:CNONlinear 10

  Set the hue surface curvature for the density spectrum 10.

  :DISPlay:VIEW:DENSity:CNONlinear? The query returns 1.000000e+01.

:DISPlay:VIEW:DENSity:CPALettes

- **Command Format**

  :DISPlay:VIEW:DENSity:CPALettes {COOL|WARM|RADar|FIRE|FROSt}

  :DISPlay:VIEW:DENSity:CPALettes?

- **Functional Description**

  Select the display hue of the color palette for the density spectrum.

  COOL: Cool color

  WARM: Warm color

  RADar: Radar color

  FIRE: Fire color

  FROSt: Frost color

- **Return Format**

  The query returns the display hue of the color palette for the density spectrum: COOL, WARM, RADar, FIRE, or FROSt.

- **For Example**

  :DISPlay:VIEW:DENSity:CPALettes WARM

  Select the display color of the color palette for the density spectrum to WARM.

  :DISPlay:VIEW:DENSity:CPALettes? The query returns WARM.

:DISPlay:VIEW:DENSity:HDHue

- **Command Format**

  :DISPlay:VIEW:DENSity:HDHue <real>

  :DISPlay:VIEW:DENSity:HDHue?

- **Functional Description**

  Set the highest probability value for the density spectrum.

  <real>: A continuous real number, with a value range of 0.1 to 100.

- **Return Format**

  The query returns the highest probability value of the density spectrum in scientific notation.

- **For Example**

  :DISPlay:VIEW:DENSity:HDHue 100 Set the highest probability value to 100%.

  :DISPlay:VIEW:DENSity:HDHue? The query returns 1.000000e+00.

:DISPlay:VIEW:DENSity:LDHue

- **Command Format**

  :DISPlay:VIEW:DENSity:LDHue <real>

:DISPlay:VIEW:DENSity:LDHue?

- **Functional Description**

  Set the lowest probability value for the density spectrum.

  <real>: A continuous real number, with a value range of 0 to 99.9.

- **Return Format**

  The query returns the lowest probability value of the density spectrum in scientific notation.

- **For Example**

:DISPlay:VIEW:DENSity:LDHue 0	Set the lowest probability value to 0.
:DISPlay:VIEW:DENSity:LDHue?	The query returns 0.000000e+00.

:DISPlay:VIEW:DENSity:PERSistence

- **Command Format**

  :DISPlay:VIEW:DENSity:PERSistence <time>

  :DISPlay:VIEW:DENSity:PERSistence?

- **Functional Description**

  Set the persistence time for the density spectrum.

  <time>: A continuous real number with the default unit in seconds (s).

- **Return Format**

  The query returns the persistence time of the density spectrum in scientific notation, with the unit in seconds (s).

- **For Example**

:DISPlay:VIEW:DENSity:PERSistence 100ms	Set the persistence time to 100 ms.
:DISPlay:VIEW:DENSity:PERSistence?	The query returns 1.000000e-01.

:DISPlay:VIEW:DENSity:PERSistence:INFinite

- **Command Format**

  :DISPlay:VIEW:DENSity:PERSistence:INFinite {{1|ON} | {0|OFF}}

  :DISPlay:VIEW:DENSity:PERSistence:INFinite?

- **Functional Description**

  Control the infinite switch of the persistence time for the density spectrum.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the infinite switch status of the persistence time for the density spectrum: 0 or 1.

- **For Example**

:DISPlay:VIEW:DENSity:PERSistence:INFinite ON	Infinite persistence time is turned on.
:DISPlay:VIEW:DENSity:PERSistence:INFinite?	The query returns 1.

:DISPlay:VIEW:DENSity:TRUNcate:STATe

- **Command Format**

:DISPlay:VIEW:DENSity:TRUNcate:STATe {{1|ON} | {0|OFF}}

:DISPlay:VIEW:DENSity:TRUNcate:STATe?

- **Functional Description**

Control the hue cutoff switch for the density spectrum.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the hue cutoff switch status of the density spectrum: 0 or 1.

- **For Example**

:DISPlay:VIEW:DENSity:TRUNcate:STATe ON	Hue cutoff is turned on.
:DISPlay:VIEW:DENSity:TRUNcate:STATe?	The query returns 1.

:DISPlay:VIEW[:SELect]

- **Command Format**

:DISPlay:VIEW[:SELect]

{NORMal|DENSity|SPECtrogram|DSGRam|PVTime|PSPectrum|PSGRam|POWergram|POSPectro}

:DISPlay:VIEW[:SELect]?

- **Functional Description**

Select the current displayed window.

NORMal: Normal

DENSity: Density spectrum

SPECtrogram: Spectrum

DSGRam: Density spectrum

PVTime: Power time

PSPectrum: Power time frequency spectrum

PSGRam: Power time spectrum

POWergram: Power spectrum

POSPectro: Power spectrum + Spectrum

- **Return Format**

  The query returns the current displayed window: NORMal, DENSity, SPECtrogram, DSGRam, PVTime, PSPectrum, PSGRam, POWergram, or POSPectro.

- **For Example**

  :DISPlay:VIEW PVTime Select PVTime as the current displayed window.

  :DISPlay:VIEW? The query returns PVTime.

:DISPlay:WINDow[1]:TRACe:Y:DLINe

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y:DLINe <ampl>

  :DISPlay:WINDow[1]:TRACe:Y:DLINe?

- **Functional Description**

  Adjust the amplitude of the displayed trace.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the amplitude of the displayed trace in scientific notation, with the unit in dBm.

- **For Example**

  :DISPlay:WINDow[1]:TRACe:Y:DLINe -40

  Set the amplitude of the displayed trace to -40 dBm.

  :DISPlay:WINDow[1]:TRACe:Y:DLINe? The query returns -4.000000e+01.

:DISPlay:WINDow[1]:TRACe:Y:DLINe:STATe

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y:DLINe:STATe {{1|ON} | {0|OFF}}

  :DISPlay:WINDow[1]:TRACe:Y:DLINe:STATe?

- **Functional Description**

  Control the displayed trace switch.

  1|ON: Display

  0|OFF: Not display

- **Return Format**

  The query returns the status of the displayed trace: 0 or 1.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:DLINe:STATe ON The displayed line is displayed.

  :DISPlay:WINDow:TRACe:Y:DLINe:STATe? The query returns 1.

:DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel <real>

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

  Adjust the reference level for the Y-axis.

  <real>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the reference level of the Y-axis in scientific notation, with the unit in dBm.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:RLEVel -10 Set the reference level for the Y-axis to -10 dBm.

  :DISPlay:WINDow:TRACe:Y:RLEVel? The query returns -1.000000e+01.

:DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel:OFFSet

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel:OFFSet <real>

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:RLEVel:OFFSet?

- **Functional Description**

  Adjust the reference level offset for the Y-axis.

  <freq>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the reference level offset of the Y-axis in scientific notation, with the unit in dB.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:RLEVel:OFFSet 5

  Set the reference level offset for the Y-axis to 5 dB.

  :DISPlay:WINDow:TRACe:Y:RLEVel:OFFSet? The query returns 5.000000e+00.

:DISPlay:WINDow[1]:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale.

  <real>: A discrete real number with the default unit in dB. The range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the Y-axis scale in scientific notation, with the unit in dB.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:PDIVision 1 Set the Y-axis scale to 1 dB.

  :DISPlay:WINDow:TRACe:Y:PDIVision? The query returns 1.000000e+00.

:DISPlay:WINDow[1]:TRACe:Y[:SCALe]:SPACing

- **Command Format**

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:SPACing {LOGarithmic|LINear}

  :DISPlay:WINDow[1]:TRACe:Y[:SCALe]:SPACing?

- **Functional Description**

  Select the scale type for the Y-axis. This setting is only valid in the normal, power time, and power time spectrum windows.

  LOGarithmic: Logarithmic

  LINear: Linear

- **Return Format**

  The query returns the scale type of the Y-axis: LOGarithmic or LINear.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:SPACing LOGarithmic

  Select the scale type for the Y-axis to LOGarithmic.

  :DISPlay:WINDow:TRACe:Y:SPACing? The query returns LOGarithmic.

:DISPlay:WINDow4:AADJust

- **Command Format**

  :DISPlay:WINDow4:AADJust

- **Functional Description**

  Automatically correct the hue for the spectrum.

- **Return Format**

  No return value.

- **For Example**

  :DISPlay:WINDow4:AADJust Automatically correct the hue for the spectrum.

:DISPlay:WINDow4:BOTTom

- **Command Format**

  :DISPlay:WINDow4:BOTTom <integer>

:DISPlay:WINDow4:BOTTom?

- **Functional Description**

Adjust the bottom color position for the spectrum.

<integer>: Bottom color position; an integer value, with a value range of 0 to 90.

- **Return Format**

The query returns the bottom color position of the spectrum.

- **For Example**

:DISPlay:WINDow4:BOTTom 20	Set the bottom color position for the spectrum to 20.
:DISPlay:WINDow4:BOTTom?	The query returns 20.

:DISPlay:WINDow4:HUE

- **Command Format**

:DISPlay:WINDow4:HUE <real>

:DISPlay:WINDow4:HUE?

- **Functional Description**

Set the reference hue for the spectrum.

<real>: Reference hue; a continuous real value.

- **Return Format**

The query returns the reference hue of the spectrum in scientific notation

- **For Example**

:DISPlay:WINDow4:HUE 10	Set the reference hue for the spectrum to 10.
:DISPlay:WINDow4:HUE?	The query returns 1.000000e+01.

:DISPlay:WINDow4:REFerence

- **Command Format**

:DISPlay:WINDow4:REFerence <integer>

:DISPlay:WINDow4:REFerence?

- **Functional Description**

Adjust the reference hue position for the spectrum.

<integer>: Reference hue position; a continuous real value, with a value range of 10 to 100.

- **Return Format**

The query returns the reference hue position.

- **For Example**

:DISPlay:WINDow4:REFerence 50	Set the reference hue position to 50.
:DISPlay:WINDow4:REFerence?	The query returns 50.

:DISPlay:WINDow4:TRACe:COUPle

- **Command Format**

  :DISPlay:WINDow4:TRACe:COUPle {{1|ON} | {0|OFF}}

  :DISPlay:WINDow4:TRACe:COUPle?

- **Functional Description**

  Control the cursor coupling trace switch for the spectrum.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of the cursor coupling trace for the spectrum: 0 or 1.

- **For Example**

  :DISPlay:WINDow4:TRACe:COUPle ON

  The cursor coupling trace of the spectrum is turned on.

  :DISPlay:WINDow4:TRACe:COUPle? The query returns 40.

:DISPlay:WINDow4:TRACe:POSition

- **Command Format**

  :DISPlay:WINDow4:TRACe:POSition <integer>

  :DISPlay:WINDow4:TRACe:POSition?

- **Functional Description**

  Adjust the serial number of the displayed traces for the spectrum. The trace serial number and start time can be directly converted into each other. Within the serial trace range, the final setting value is consistent with the imported setting values.

  <integer>: Trace serial number; a continuous integer value, with a value range of 1 to 10000.

- **Return Format**

  The query returns the serial number of the displayed traces for the spectrum.

- **For Example**

  :DISPlay:WINDow4:TRACe:POSition 40

  Set the displayed trace serial number to 40 for the spectrum.

  :DISPlay:WINDow4:TRACe:POSition? The query returns 40.

:DISPlay:WINDow4:TRACe:TIME

- **Command Format**

  :DISPlay:WINDow4:TRACe:TIME <time>

  :DISPlay:WINDow4:TRACe:TIME?

- **Functional Description**

  Adjust the start time in the displayed traces for the spectrum. The trace serial number and start time can be directly converted into each other. Within the start time range, the final setting value may closely match the imported setting values.

  <time>: Trace time, a continuous real number with a default unit of seconds (s). The range is from 32 ms to 320 s.

- **Return Format**

  The query returns the start time in the displayed traces for the spectrum in scientific notation, with the unit in seconds (s).

- **For Example**

  :DISPlay:WINDow4:TRACe:TIME 1s

  Set the start time in the displayed traces to 1 s for the spectrum.

  :DISPlay:WINDow4:TRACe:TIME? The query returns 1.000000e+00.

:DISPlay:WINDow4:TRACe:TYPE

- **Command Format**

  :DISPlay:WINDow4:TRACe:TYPE {TIME|TNUMber}

  :DISPlay:WINDow4:TRACe:TYPE?

- **Functional Description**

  Select the displayed trace type for the spectrum.

  TIME: Start time

  TNUMber: Serial number

- **Return Format**

  The query returns the displayed trace type of the spectrum: TIME or TNUMber.

- **For Example**

  :DISPlay:WINDow4:TRACe:TYPE TIME

  Select the displayed trace type to TIME for the spectrum.

  :DISPlay:WINDow4:TRACe:TYPE? The query returns TIME.

:DISPlay:WINDow8:TRACe:X[:SCALe]:AUTO

- **Command Format**

  :DISPlay:WINDow8:TRACe:X[:SCALe]:AUTO {{1|ON} | {0|OFF}}

  :DISPlay:WINDow8:TRACe:X[:SCALe]:AUTO?

- **Functional Description**

  Automatically switch the X-axis in the time domain (PvT) window.

1|ON: Display

0|OFF: Not display

- **Return Format**

The query returns the automatic switch status of the X-axis in the time domain (PvT) window: 0 or 1.

- **For Example**

:DISPlay:WINDow8:TRACe:X:AUTO ON

Automatically switch X-axis function in the time domain (PvT) window is turned on.

:DISPlay:WINDow8:TRACe:X:AUTO? The query returns 1.

:DISPlay:WINDow8:TRACe:X[:SCALe]:PDIVision

- **Command Format**

:DISPlay:WINDow8:TRACe:X[:SCALe]:PDIVision <time>

:DISPlay:WINDow8:TRACe:X[:SCALe]:PDIVision?

- **Functional Description**

Set the X-axis scale in the time domain (PvT) window.

<time>: The X-axis scale is time when in the time domain (PvT) window. The default unit is seconds (s).

- **Return Format**

The query returns the X-axis scale in the time domain (PvT) window in scientific notation, with the unit in seconds (s).

- **For Example**

:DISPlay:WINDow8:TRACe:X:PDIVision 1

Set the X-axis scale in the time domain (PvT) window to 1 s.

:DISPlay:WINDow8:TRACe:X:PDIVision? The query returns 1.000000e+00.

:DISPlay:WINDow8:TRACe:X[:SCALe]:RLEVel

- **Command Format**

:DISPlay:WINDow8:TRACe:X[:SCALe]:RLEVel <time>

:DISPlay:WINDow8:TRACe:X[:SCALe]:RLEVel?

- **Functional Description**

Set the reference time of the X-axis scale in the time domain (PvT) window.

<time>: The default unit of the reference time is seconds (s).

- **Return Format**

The query returns the reference time of the X-axis scale in the time domain (PvT) window in

scientific notation, with the unit in seconds (s).

- **For Example**

  :DISPlay:WINDow8:TRACe:X:RLEVel 1

  Set the reference time of the X-axis scale in the time domain (PvT) window to 1 s.

  :DISPlay:WINDow8:TRACe:X:RLEVel? The query returns 1.000000e+00.

:DISPlay:WINDow8:TRACe:X[:SCALe]:RPOSition

- **Command Format**

  :DISPlay:WINDow8:TRACe:X[:SCALe]:RPOSition { LEFT|CENTer|RIGHt }

  :DISPlay:WINDow8:TRACe:X[:SCALe]:RPOSition?

- **Functional Description**

  Set the reference time position of the X-axis scale in the time domain (PvT) window.

  LEFT: Left

  CENTer: Center

  RIGHt: Right

- **Return Format**

  The query returns the reference time position of the X-axis scale in the time domain (PvT) window: LEFT, CENTer, or RIGHt.

- **For Example**

  :DISPlay:WINDow8:TRACe:X:RPOSition LEFT

  Set the reference time position of the X-axis scale to LEFT in the time domain (PvT) window.

  :DISPlay:WINDow8:TRACe:X:RPOSition? The query returns LEFT.

:DISPlay:WINDow8:TRACe:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:WINDow8:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:WINDow8:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale for the time domain (PvT) window.

  <real>: A discrete real number with the default unit in dB. The range is from 0.1 dB to 20 dB.

- **Return Format**

  The query returns the Y-axis scale of the time domain (PvT) window in scientific notation, with the unit in dB.

- **For Example**

  :DISPlay:WINDow8:TRACe:Y:PDIVision 1 Set the Y-axis scale to 1 dB.

:DISPlay:WINDow8:TRACe:Y:PDIVision?	The query returns 1.000000e+00.

:DISPlay:WINDow8:TRACe:Y[:SCALe]:RLEVel

- **Command Format**

  :DISPlay:WINDow8:TRACe:Y[:SCALe]:RLEVel <real>

  :DISPlay:WINDow8:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

  Adjust the reference level for the time domain (PvT) window.

  <real>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the reference level of the time domain (PvT) window in scientific notation, with the unit in dBm.

- **For Example**

:DISPlay:WINDow8:TRACe:Y:RLEVel -10	Set the reference level to -10 dBm.
:DISPlay:WINDow8:TRACe:Y:RLEVel?	The query returns -1.000000e+01.

:DISPlay:WINDow9:AADJust

- **Command Format**

  :DISPlay:WINDow9:AADJust

- **Functional Description**

  Automatically correct the color for the power spectrum.

- **Return Format**

  No return value.

- **For Example**

:DISPlay:WINDow9:AADJust	Automatically correct the color for the power spectrum.

:DISPlay:WINDow9:BOTTom

- **Command Format**

  :DISPlay:WINDow9:BOTTom <integer>

  :DISPlay:WINDow9:BOTTom?

- **Functional Description**

  Adjust the bottom hue position for the power spectrum.

  <integer>: Bottom hue position; an integer value, with a value range of 0 to 90.

- **Return Format**

  The query returns the bottom hue position of the power spectrum.

➢ **For Example**

:DISPlay:WINDow9:BOTTom 20

Adjust the bottom hue position for the power spectrum to 20.

:DISPlay:WINDow9:BOTTom? The query returns 20.

:DISPlay:WINDow9:HUE

➢ **Command Format**

:DISPlay:WINDow9:HUE <real>

:DISPlay:WINDow9:HUE?

➢ **Functional Description**

Set the reference hue for the power spectrum.

<real>: Reference hue; a continuous real value.

➢ **Return Format**

The query returns the reference hue of the power spectrum in scientific notation

➢ **For Example**

:DISPlay:WINDow9:HUE 10 Set the reference hue for the power spectrum to 10.

:DISPlay:WINDow9:HUE? The query returns 1.000000e+01.

:DISPlay:WINDow9:REFerence

➢ **Command Format**

:DISPlay:WINDow9:REFerence <integer>

:DISPlay:WINDow9:REFerence?

➢ **Functional Description**

Adjust the reference hue position for the power spectrum.

<integer>: Reference hue position; a continuous integer value, with a value range of 10 to 100.

➢ **Return Format**

The query returns the reference hue position of the power spectrum.

➢ **For Example**

:DISPlay:WINDow4:REFerence 50

Set the reference hue position for the power spectrum to 50.

:DISPlay:WINDow4:REFerence? The query returns 50.

:DISPlay:WINDow9:TRACe:COUPle

➢ **Command Format**

:DISPlay:WINDow9:TRACe:COUPle {{1|ON} | {0|OFF}}

:DISPlay:WINDow9:TRACe:COUPle?

- **Functional Description**

  Control the cursor coupling trace switch for the power spectrum.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of the cursor coupling trace for the power spectrum: 0 or 1.

- **For Example**

  :DISPlay:WINDow9:TRACe:COUPle ON

  The cursor coupling trace of the power spectrum is turned on.

  :DISPlay:WINDow9:TRACe:COUPle? The query returns 40.

:DISPlay:WINDow9:TRACe:POSition

- **Command Format**

  :DISPlay:WINDow9:TRACe:POSition <integer>

  :DISPlay:WINDow9:TRACe:POSition?

- **Functional Description**

  Adjust the serial number of the displayed traces for the power spectrum. The trace serial number and start time can be directly converted into each other.

  <integer>: Trace serial number; a continuous integer value, with a value range of 1 to 10000.

- **Return Format**

  The query returns the displayed traces position of the power spectrum.

- **For Example**

  :DISPlay:WINDow9:TRACe:POSition 40

  Set the displayed trace serial number to 40 for the power spectrum.

  :DISPlay:WINDow9:TRACe:POSition? The query returns 40.

:DISPlay:WINDow9:TRACe:TIME

- **Command Format**

  :DISPlay:WINDow9:TRACe:TIME <time>

  :DISPlay:WINDow9:TRACe:TIME?

- **Functional Description**

  Adjust the start time of the displayed traces for the power spectrum. The trace serial number and start time can be directly converted into each other.

<time>: Trace time, a continuous real number with a default unit of seconds (s). The range is from 32 ms to 320 s.

- **Return Format**

  The query returns the start time of the displayed traces for the power spectrum in scientific notation, with the unit in seconds (s).

- **For Example**

  :DISPlay:WINDow9:TRACe:TIME 1s

  Set the start time of the displayed traces to 1 s for the power spectrum.

  :DISPlay:WINDow9:TRACe:TIME? The query returns 1.000000e+00.

:DISPlay:WINDow9:TRACe:TYPE

- **Command Format**

  :DISPlay:WINDow9:TRACe:TYPE {TIME|TNUMber}

  :DISPlay:WINDow9:TRACe:TYPE?

- **Functional Description**

  Select the displayed trace type for the spectrum.

  TIME: Start time

  TNUMber: Serial number

- **Return Format**

  The query returns the displayed trace type of the spectrum: TIME or TNUMber.

- **For Example**

  :DISPlay:WINDow9:TRACe:TYPE TIME

  Set the displayed trace type for the spectrum to TIME.

  :DISPlay:WINDow9:TRACe:TYPE? The query returns TIME.

FETCh Command

:FETCh:RTSAnalyer<n>?

- **Command Format**

  :FETCh:RTSAnalyer<n>?

- **Functional Description**

  Query the result data for real-time spectrum analysis measurement. The data consists of 800 power values in dBm.

  <n>: The sequence number of the corresponding measurement trace, with a value range of 1 to 12. Traces 1 to 6 correspond to frequency-domain traces, while traces 7 to 12 correspond to

time-domain traces.

The data format is set using the command ":FORMat[:TRACe][:DATA]", and can be ASCII characters, 32-bit, or 64-bit binary real numbers. The default is ASCII, with data values separated by commas. Binary real numbers are converted from floating point data according to the IEEE 754 standard.

- **Return Format**

The query returns the result data of real-time spectrum analysis measurement. The ASCII format represents data in scientific notation and returns it as a character stream, while the binary format returns data in the corresponding bit pattern, with data returned in Data Block Format. The default unit is dBm.

- **For Example**

:FETCh:RTSAnalyer1?

The query returns trace 1 data in the frequency domain for real-time spectrum analysis.

FORMat Command

:FORMat[:TRACe][:DATA]

- **Command Format**

:FORMat[:TRACe][:DATA] {ASCii|REAL32|REAL64}

:FORMat[:TRACe][:DATA]?

- **Functional Description**

Set the return format for the trace data. The default format is ASCII.

ASCII: Character

REAL32: A 32-bit binary real number.

REAL64: A 64-bit binary real number.

- **Return Format**

The query returns the return format of the trace data: ASC8, REAL32, or REAL64.

For Example

:FORMat ASCii	Set the return format for the trace data to ASCII.
:FORMat?	The query returns ASC8.

INITiate Command

:INITiate:CONTinuous

- **Command Format**

:INITiate:CONTinuous {{1|ON} | {0|OFF}}

:INITiate:CONTinuous?

- **Functional Description**

  Switch between single sweep and continuous sweep mode.

  1|ON: Continuous sweep mode

  0|OFF: Single sweep mode

- **Return Format**

  The query returns whether the mode is continuous sweep: 0 or 1.

- **For Example**

:INITiate:CONTinuous ON	Continuous sweep mode.
:INITiate:CONTinuous?	The query returns 1.

:INITiate[:IMMediate]

- **Command Format**

  :INITiate[:IMMediate]

- **Functional Description**

  Start the sweep function.

- **Return Format**

  No return value.

- **For Example**

:INITiate:IMMediate	Start the sweep function.

:INITiate:PAUSe

- **Command Format**

  :INITiate:PAUSe

- **Functional Description**

  Pause the sweep function.

- **Return Format**

  No return value.

- **For Example**

:INITiate:PAUSe	Pause the sweep function.

:INITiate:RESTart

- **Command Format**

  :INITiate:RESTart

- **Functional Description**

Restart the sweep function.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:RESTart Restart the sweep function.

:INITiate:RESume

- **Command Format**

  :INITiate:RESume

- **Functional Description**

  Resume the sweep function.

- **Return Format**

  No return value.

- **For Example**

  :INITiate:RESume Resume the sweep function.

MMEMory Command

:MMEMory:LOAD:LIMit

- **Command Format**

  :MMEMory:LOAD:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

  Load the data corresponding to the specified limit value.

  LLINE1-LLINE6: Corresponds to limit 1 to limit 6

  <filename>: Filename with the ".limit" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:LIMit LLINE1,"test.limit" Limit 1 loads the data from the file "test.limit".

:MMEMory:LOAD:STATe

- **Command Format**

  :MMEMory:LOAD:STATe <filename>

- **Functional Description**

  Load the register status data.

<filename>: Filename with the ".state" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:STATe "test.state" Load the register status data from the file "test.state".

:MMEMory:LOAD:TRACe

- **Command Format**

  :MMEMory:LOAD:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Load the data from the specified trace.

  TRACE1-TRACE6: Corresponds to trace 1 to trace 6

  <filename>: Filename with the ".trace" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:TRACe TRACE1,"test.trace"

  Trace 1 loads the data from the file "test.trace".

:MMEMory:STORe:LIMit

- **Command Format**

  :MMEMory:STORe:LIMit {LLINE1|LLINE2|LLINE3|LLINE4|LLINE5|LLINE6,<filename>}

- **Functional Description**

  Save the specified limit data to the default catalogue.

  LLINE1-LLINE6: Corresponds to limit 1 to limit 6

  <filename>: Filename with the ".limit" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:LIMit LLINE1,"test.limit" Save limit 1 data to the file "test.limit".

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

Save the register status to the specified file.

<filename>: Filename with the ".state" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:STATe "test.state" Save the register status to the file "test.state".

:MMEMory:STORe:TRACe

- **Command Format**

  :MMEMory:STORe:TRACe {TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<filename>}

- **Functional Description**

  Save the specified trace data to the file.

  TRACE1-TRACE6: Corresponds to trace 1 to trace 6

  <filename>: Filename with the ".trace" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:STORe:TRACe TRACE1,"test.trace" Save trace 1 data to the file "test.trace".

READ Command

The subsystem commands :READ and :FETCh are used to acquire measured results. The key difference between these commands is that the :FETCh command immediately retrieves the measured result, while the :READ command initiates a measurement, waits for it to complete, and then returns the result. If the measurement time exceeds the timeout period, the :READ command may fail to return the results due to a timeout.

:READ:RTSAnalyer<n>?

- **Command Format**

  :READ:RTSAnalyer<n>?

- **Functional Description**

  Query the result data for real-time spectrum analysis measurements. The data consists of 800 power values in dBm.

  <n>: The sequence number of the corresponding measurement trace, with a value range of 1 to 12. Traces 1 to 6 correspond to frequency-domain traces, while traces 7 to 12 correspond to

time-domain traces.

The data format is set using the command ":FORMat[:TRACe][:DATA]", and can be ASCII characters, 32-bit, or 64-bit binary real numbers. The default is ASCII, with data values separated by commas. Binary real numbers are converted from floating point data according to the IEEE 754 standard.

- **Return Format**

The query returns the result data of real-time spectrum analysis measurement. The query returns trace amplitude data. The ASCII format represents data in scientific notation and returns it as a character stream, while the binary format returns data in the corresponding bit pattern, with data returned in Data Block Format. The default unit is dBm.

- **For Example**

:READ:RTSAnalyer1?

The query returns trace 1 data in the frequency domain for real-time spectrum analysis.

SENSe Command

[:SENSe]:ACQuisition:TIME

- **Command Format**

[:SENSe]:ACQuisition:TIME <time>

[:SENSe]:ACQuisition:TIME?

- **Functional Description**

Set the capture time for the frequency domain window.

<time>: A continuous real number with the default unit in seconds (s).

- **Return Format**

The query returns the capture time of the frequency domain window in scientific notation, with the unit in seconds (s).

- **For Example**

:ACQuisition:TIME 100ms

Set the capture time for the frequency domain window to 100 ms.

:ACQuisition:TIME? The query returns 1.000000e-01.

[:SENSe]:ACQuisition:TIME:AUTO

- **Command Format**

[:SENSe]:ACQuisition:TIME:AUTO {{1|ON} | {0|OFF}}

[:SENSe]:ACQuisition:TIME:AUTO?

- **Functional Description**

  Automatically/manually switch the capture time for the frequency domain window.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the switch status of the capture time: 0 or 1.

- **For Example**

:ACQuisition:TIME:AUTO ON	Set the capture time to automatic (AUTO) mode.
:ACQuisition:TIME:AUTO?	The query returns 1.

[:SENSe]:ACQuisition:TIME:PVTime

- **Command Format**

  [:SENSe]:ACQuisition:TIME:PVTime <time>

  [:SENSe]:ACQuisition:TIME:PVTime?

- **Functional Description**

  Set the capture time for the time domain (PvT) window.

  <time>: A continuous real number with the default unit in seconds (s).

- **Return Format**

  The query returns the capture time of the time domain (PvT) window in scientific notation, with the unit in seconds (s).

- **For Example**

:ACQuisition:TIME:PVTime 100ms	Set the capture time for the time domain to 100 ms.
:ACQuisition:TIME:PVTime?	The query returns 1.000000e-01.

[:SENSe]:ACQuisition:TIME:PVTime:AUTO

- **Command Format**

  [:SENSe]:ACQuisition:TIME:PVTime:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:ACQuisition:TIME:PVTime:AUTO?

- **Functional Description**

  Automatically/manually switch the capture time for the time domain (PvT) window.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the switch status of the capture time: 0 or 1.

- **For Example**

:ACQuisition:TIME:PVTime:AUTO ON	Set the capture time to automatic (AUTO) mode.
:ACQuisition:TIME:PVTime:AUTO?	The query returns 1.

[:SENSe]:AVERage:COUNt

- **Command Format**

  [:SENSe]:AVERage:COUNt <integer>

  [:SENSe]:AVERage:COUNt?

- **Functional Description**

  Set the average number.

  <integer>: An integer with a value range of 1 to 10000.

- **Return Format**

  The query returns the average number, with a value range of 1 to 10000.

- **For Example**

:AVERage:COUNt 10	Set the average number to 10.
:AVERage:COUNt?	The query returns 10.

[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect {RBW1|RBW2|RBW3|RBW4|RBW5|RBW6}

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect?

- **Functional Description**

  Select the resolution bandwidth. The time domain window (PvT) does not support this setting.

  RBW1|RBW2|RBW3|RBW4|RBW5|RBW6: Six resolution bandwidth options are available, with specific resolution bandwidth values varying with sweep width.

- **Return Format**

  The query returns the sequency of the resolution bandwidth: RBW1, RBW2, RBW3, RBW4, RBW5, or RBW6.

- **For Example**

:BANDwidth:SELect RBW1	Select RBW1.
:BANDwidth:SELect?	The query returns RBW1.

[:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO[:STATe]

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:BANDwidth|BWIDth[:RESolution]:SELect:AUTO[:STATe]?

- **Functional Description**

  Automatically/manually switch the resolution bandwidth. The time domain window (PvT) does not support this setting.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the switch status of the resolution bandwidth: 0 or 1.

- **For Example**

  :BANDwidth:SELect:AUTO ON

  Switch the resolution bandwidth to automatic (AUTO) mode.

  :BANDwidth:SELect:AUTO? The query returns 1.

[:SENSe]:BANDwidth|BWIDth:SHAPe

- **Command Format**

  [:SENSe]:BANDwidth|BWIDth:SHAPe

  {GAUSsian|FLATtop|BHARris|RECTangular|HANNing|KAISer}

  [:SENSe]:BANDwidth|BWIDth:SHAPe?

- **Functional Description**

  Select the filter. The time domain window (PvT) does not support this setting.

  GAUSsian: Gaussian filter

  FLATtop: Flat top filter

  BHARris: Blackman-harris filter

  RECTangular: Rectangular filter

  HANNing: Hanning filter

  KAISer: Kaiser filter

- **Return Format**

  The query returns the filter type: GAUSsian, FLATtop, BHARris, RECTangular, HANNing, or KAISer.

- **For Example**

  :BANDwidth:SHAPe GAUSsian Select Gaussian filter.

  :BANDwidth:SHAPe? The query returns GAUSsian.

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

- **Command Format**

  [:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] {50|75}

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

- **Functional Description**

  Select the input impedance 50 Ω or 75 Ω.

- **Return Format**

  The query returns the input impedance: 50 Ω or 75 Ω.

- **For Example**

:CORRection:IMPedance 50	Set the input impedance to 50 Ω.
:CORRection:IMPedance?	The query returns 50.

[:SENSe]:DETector:TRACe

- **Command Format**

  [:SENSe]:DETector:TRACe {SAMPle|POSitive|NEGative|AVERage}

  [:SENSe]:DETector:TRACe?

- **Functional Description**

  Select the detector type for all traces.

  SAMPle: Sample detector

  POSitive: Peak detector

  NEGative: Negative peak detector

  AVERage: Average detector

- **Return Format**

  The query returns the detector type of trace: SAMPle, POSitive, NEGative, or AVERage.

- **For Example**

:DETector:TRACe POSitive	Select the detector type for all traces to POSitive.
:DETector:TRACe?	The query returns POSitive.

[:SENSe]:FREQuency:CENTer

- **Command Format**

  [:SENSe]:FREQuency:CENTer <freq>

  [:SENSe]:FREQuency:CENTer?

- **Functional Description**

  Set the center frequency for the sweep frequency.

  <freq>: A continuous real number with the default unit in Hz.

  The frequency range is from 50 Hz to the maximum frequency -50 Hz.

  The maximum frequency of UTS3084B is 8.4 GHz.

- **Return Format**

The query returns the center frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz Set the center frequency for the sweep frequency to 1 GHz.

:FREQuency:CENTer? The query returns 1.000000e+09.

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

- **Command Format**

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

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

- **Functional Description**

Automatically/manually switch the center frequency step.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the status of the center frequency step: 0 or 1.

- **For Example**

:FREQuency:CENTer:STEP:AUTO ON

Set the center frequency step to automatic (AUTO) mode.

:FREQuency:CENTer:STEP:AUTO? The query returns 1.

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

- **Command Format**

[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>

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

- **Functional Description**

Adjust the center frequency step.

<freq>: A continuous real number with the default unit in Hz.

- **Return Format**

The query returns the center frequency step in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer:STEP 10MHz Set the center frequency step to 10 MHz.

:FREQuency:CENTer:STEP? The query returns 1.000000e+07.

[:SENSe]:FREQuency:OFFSet

- **Command Format**

[:SENSe]:FREQuency:OFFSet <freq>

[:SENSe]:FREQuency:OFFSet?

- **Functional Description**

Set the frequency offset.

<freq>: A continuous real number with the default unit in Hz.

- **Return Format**

The query returns the frequency offset in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:OFFSet 10MHz	Set the frequency offset to 10 MHz.
:FREQuency:CENTer?	The query returns 1.000000e+07.

[:SENSe]:FREQuency:SPAN

- **Command Format**

[:SENSe]:FREQuency:SPAN <freq>

[:SENSe]:FREQuency:SPAN?

- **Functional Description**

Set the span.

<freq>: A continuous real number with the default unit in Hz.

The span range is from 100 Hz to the maximum frequency.

- **Return Format**

The query returns the span in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:SPAN 10MHz	Set the span to 1 MHz.
:FREQuency:SPAN?	The query returns 1.000000e+07.

[:SENSe]:FREQuency:SPAN:FULL

- **Command Format**

[:SENSe]:FREQuency:SPAN:FULL

- **Functional Description**

Set the span to the full span (maximum span).

- **Return Format**

No return value.

- **For Example**

:FREQuency:SPAN:FULL	Set the span to the full span (maximum span).

[:SENSe]:FREQuency:SPAN:PREVious

- **Command Format**

  [:SENSe]:FREQuency:SPAN:PREVious

- **Functional Description**

  Set the span to the previous span.

- **Return Format**

  No return value.

- **For Example**

  :FREQuency:SPAN:PREVious Set the span to the previous span.

[:SENSe]:FREQuency:STARt

- **Command Format**

  [:SENSe]:FREQuency:STARt <freq>

  [:SENSe]:FREQuency:STARt?

- **Functional Description**

  Set the start frequency for the sweep frequency function.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the start frequency in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:STARt 10MHz Set the start frequency to 10 MHz.

  :FREQuency:STARt? The query returns 1.000000e+07.

[:SENSe]:FREQuency:STOP

- **Command Format**

  [:SENSe]:FREQuency:STOP <freq>

  [:SENSe]:FREQuency:STOP?

- **Functional Description**

  Set the stop frequency for the sweep frequency function.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the stop frequency in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:STOP 1GHz Set the stop frequency to 1 GHz.

  :FREQuency:STOP? The query returns 1.000000e+09.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

  Adjust the input attenuation.

  <ampl>: A continuous real number. The default unit is dB, with an amplitude range of 0 dB to 51 dB.

- **Return Format**

  The query returns the input attenuation in dB.

- **For Example**

:POWer:ATTenuation 10	Set the input attenuation to 10 dB.
:POWer:ATTenuation?	The query returns 10.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:ATTenuation:AUTO {{1|ON} | {0|OFF}}

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

- **Functional Description**

  Automatically/manually switch the input attenuation.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the input attenuation: 0 or 1.

- **For Example**

:POWer:ATTenuation:AUTO ON	Set the input attenuation to automatic (AUTO) mode.
:POWer:ATTenuation:AUTO?	The query returns 1.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

  Set the preamplifier switch to on or off.

- **Return Format**

The query returns the switch status of the preamplifier: 0 or 1.

- **For Example**

:POWer:GAIN:STATe ON	Turn on the preamplifier switch.
:POWer:GAIN:STATe?	The query returns 1

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

[:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

[:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

Set the frequency reference.

INTernal: Internal

EXTernal: External

- **Return Format**

The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

TRACe Command

:TRACe[:DATA]

- **Command Format**

:TRACe[:DATA] TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6,<data>

:TRACe[:DATA]? TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6

- **Functional Description**

Set and acquire the trace data block. The number of data points corresponds to the sweep points, with each data value representing the power at a frequency point between the start frequency and the stop frequency. The data format is set using the command ":FORMat[:TRACe][:DATA]", and can be ASCII characters, 32-bit, or 64-bit binary real numbers. The default is ASCII, with data values separated by commas. Binary real numbers are converted from floating point data according to the IEEE 754 standard.

TRACE1|TRACE2|TRACE3|TRACE4|TRACE5|TRACE6: Trace sequence character, indicating the trace 1|2|3|4|5|6.

The trace value varies for different models of spectrum analyzers. Refer to Appendix 3

Parameter Table for Each Model.

- **Return Format**

  The query returns trace amplitude data. The ASCII format represents data in scientific notation and returns it as a character stream, while the binary format returns data in the corresponding bit pattern, with data returned in Data Block Format. The default unit is dBm.

- **For Example**

  Set the data for trace 1.

  :TRACe:DATA TRACE1,-59,-77,-60,-98,-59,-59,-77,-60,-98,-59,-59

  :TRACe:DATA? TRACE1

  The query returns

  -5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01,-7.700000e+01,-6.000000e+01,-9.800000e+01,-5.900000e+01,-5.900000e+01

:TRACe<n>:DISPlay[:STATe]

- **Command Format**

  :TRACe<n>:DISPlay[:STATe] {{1|ON} | {0|OFF}}

  :TRACe<n>:DISPlay[:STATe]?

- **Functional Description**

  Set the display switch for the specified trace.

  <n>: Trace sequence number, an integer ranging from 1 to 12. Traces 1 to 6 correspond to frequency domain traces, while traces 7 to 12 correspond to time domain traces. These traces correspond to the trace data within the frequency domain window and the time domain window (PvT), respectively.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the display status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:DISPlay ON	Display trace 1.
:TRACe1:DISPlay?	The query returns 1.

:TRACe<n>:TYPE

- **Command Format**

  :TRACe<n>:TYPE <WRITe|AVERage|MAXHold|MINHold>

  :TRACe<n>:TYPE?

- **Functional Description**

  Select the trace type for the specified trace.

  <n>: Trace sequence number, an integer ranging from 1 to 12. Traces 1 to 6 correspond to frequency domain traces, while traces 7 to 12 correspond to time domain traces. These traces correspond to the trace data within the frequency domain window and the time domain window (PvT), respectively.

  WRITe: Refresh

  AVERage: Trace average

  MAXHold: Maximum hold

  MINHold: Minimum hold

- **Return Format**

  The query returns the trace type of the specified trace: WRITe, AVERage, MAXHold, or MINHold.

- **For Example**

:TRACe1:TYPE AVERage	Set trace 1 to AVERage.
:TRACe1:TYPE?	The query returns AVERage.

:TRACe<n>:UPDate:STATe

- **Command Format**

  :TRACe<n>:UPDate:STATe {{1|ON} | {0|OFF}}

  :TRACe<n>:UPDate:STATe?

- **Functional Description**

  Set the trace refresh switch. The trace will continue refreshing even when it is turned off.

  <n>: Trace sequence number, an integer ranging from 1 to 12. Traces 1 to 6 correspond to frequency domain traces, while traces 7 to 12 correspond to time domain traces. These traces correspond to the trace data within the frequency domain window and the time domain window (PvT), respectively.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the refresh status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:UPDate:STATe ON	Turn on the trace refresh for trace 1.
:TRACe1:UPDate:STATe?	The query returns 1

:TRACe:SELect

- **Command Format**

  :TRACe:SELect <integer>

  :TRACe:SELect?

- **Functional Description**

  Select a trace from all available traces to be the current trace.

  <integer>: Trace sequence number, an integer ranging from 1 to 12. Traces 1 to 6 correspond to frequency domain traces, while traces 7 to 12 correspond to time domain traces. These traces correspond to the trace data within the frequency domain window and the time domain window (PvT), respectively.

- **Return Format**

  The query returns the serial number of the current trace.

- **For Example**

:TRACe:SELect 1	Set trace 1 to the current trace.
:TRACe:SELect?	The query returns 1

TRIGger Command

:TRIGger[:SEQuence]:EXTernal1:DELay

- **Command Format**

  :TRIGger[:SEQuence]:EXTernal1:DELay <time>

  :TRIGger[:SEQuence]:EXTernal1:DELay?

- **Functional Description**

  Set the trigger delay for the external trigger 1.

  <time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

  The query returns the trigger delay of the external trigger 1 in scientific notation, with the unit in seconds (s).

- **For Example**

:TRIGger:EXTernal1:DELay 0.01	Set the trigger delay for external trigger 1 to 10 ms.
:TRIGger:EXTernal1:DELay?	The query returns 1.000000e-02.

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe {{1|ON} | {0|OFF}}

:TRIGger[:SEQuence]:EXTernal1:DELay:STATe?

- **Functional Description**

Switch the trigger delay switch of the external trigger 1 to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the trigger delay switch status of the external trigger 1: 0 or 1.

- **For Example**

:TRIGger:EXTernal1:DELay:STATe OFF

Turn off the trigger delay switch of the external trigger 1.

:TRIGger:EXTernal1:DELay:STATe? The query returns 0.

:TRIGger[:SEQuence]:EXTernal1:LEVel

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:LEVel <voltage>

:TRIGger[:SEQuence]:EXTernal1:LEVel?

- **Functional Description**

Set the trigger level for the external trigger 1. Currently, only UTS5026A is supported.

<voltage>: Voltage value with the default unit in volts (V). The range is from -3.3 V to 3.3 V.

- **Return Format**

The query returns the trigger level of the external trigger 1 in scientific notation.

- **For Example**

:TRIGger:EXTernal1:LEVel 1 Set the trigger level of the external trigger 1 to 1 V.

:TRIGger:EXTernal1:LEVel? The query returns 1.000000e+00.

:TRIGger[:SEQuence]:EXTernal1:SLOPe

- **Command Format**

:TRIGger[:SEQuence]:EXTernal1:SLOPe {POSitive|NEGative}

:TRIGger[:SEQuence]:EXTernal1:SLOPe?

- **Functional Description**

Set the trigger edge for the external trigger 1.

POSitive: Rising edge

NEGative: Negative edge

- **Return Format**

The query returns the trigger edge of the external trigger 1: POSitive or NEGative.

- **For Example**

:TRIGger:EXTernal1:SLOPe POSitive

Set the trigger edge for the external trigger 1 to POSitive.

:TRIGger:EXTernal1:SLOPe? The query returns POSitive.

:TRIGger[:SEQuence]:SOURce

- **Command Format**

:TRIGger[:SEQuence]:SOURce {IMMediate|VIDeo|EXTernal1|FMT}

:TRIGger[:SEQuence]:SOURce?

- **Functional Description**

Select the trigger type.

IMMediate: Free trigger

VIDeo: Medium frequency power trigger

EXTernal1: External trigger

FMT: Frequency mask trigger

- **Return Format**

The query returns the trigger type: IMMediate, VIDeo, EXTernal1, or FMT.

- **For Example**

:TRIGger:SOURce IMMediate Select the trigger type to IMMediate.

:TRIGger:SOURce? The query returns IMMediate.

:TRIGger[:SEQuence]:VIDeo:DELay

- **Command Format**

:TRIGger[:SEQuence]:VIDeo:DELay <time>

:TRIGger[:SEQuence]:VIDeo:DELay?

- **Functional Description**

Set the trigger delay for the video trigger.

<time>: A continuous positive number with the default unit in seconds (s). The range is from 0 s to 500 ms.

- **Return Format**

The query returns the trigger delay of the video trigger in scientific notation, with the unit in seconds (s).

- **For Example**

:TRIGger:VIDeo:DELay 0.01 Set the trigger delay of the video trigger to 10 ms.

:TRIGger:VIDeo:DELay?	The query returns 1.000000e-02.

:TRIGger[:SEQuence]:VIDeo:DELay:STATe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:DELay:STATe {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:VIDeo:DELay:STATe?

- **Functional Description**

  Set the trigger delay switch for the external trigger 2.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the trigger delay switch status of the external trigger 2: 0 or 1.

- **For Example**

:TRIGger:VIDeo:DELay:STATe OFF	Disable the trigger delay for the video trigger.
:TRIGger:VIDeo:DELay:STATe?	The query returns 0.

:TRIGger[:SEQuence]:VIDeo:LEVel

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:LEVel <ampl>

  :TRIGger[:SEQuence]:VIDeo:LEVel?

- **Functional Description**

  Set the trigger level for the video trigger.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the trigger edge of the external trigger in scientific notation, with the unit in dBm.

- **For Example**

:TRIGger:VIDeo:LEVel -50	Set the trigger level for the video trigger to -50 dBm.
:TRIGger:VIDeo:LEVel?	The query returns -5.000000e+01.

:TRIGger[:SEQuence]:VIDeo:SLOPe

- **Command Format**

  :TRIGger[:SEQuence]:VIDeo:SLOPe {POSitive|NEGative}

  :TRIGger[:SEQuence]:VIDeo:SLOPe?

- **Functional Description**

Select the trigger edge for the video trigger. Currently, only UTS5026A is supported.

POSitive: Rising edge

NEGative: Negative edge

- **Return Format**

The query returns the trigger edge of the video trigger: POSitive or NEGative.

- **For Example**

:TRIGger:VIDeo:SLOPe POSitive	Select the trigger edge for the video trigger to POSitive.
:TRIGger:VIDeo:SLOPe?	The query returns POSitive.

:TRIGger[:SEQuence]:FMT:MASK

- **Command Format**

:TRIGger[:SEQuence]:FMT:MASK {UPPer|LOWer|BOTH}

:TRIGger[:SEQuence]:FMT:MASK?

- **Functional Description**

Select the trigger edge for the frequency mask trigger.

UPPer: Upper mask

LOWer: Lower mask

BOTH: All mask

- **Return Format**

The query returns the trigger edge od the frequency mask trigger: UPPer, LOWer, or BOTH.

- **For Example**

:TRIGger:FMT:MASK UPPer	Select the trigger edge for the frequency mask trigger to UPPer.
:TRIGger:FMT:MASK?	The query returns UPPer.

:TRIGger[:SEQuence]:FMT:CRITeria

- **Command Format**

:TRIGger[:SEQuence]:FMT:CRITeria {ENTer|LEAVe|INSide|OUTSide|ELEave|LENTer}

:TRIGger[:SEQuence]:FMT:CRITeria?

- **Functional Description**

Select the trigger condition for the frequency mask trigger.

ENTer: Enter

LEAVe: Leave

INSide: Inside

OUTSide: Outside

ELEave: Enter-level

LENTer: Level-enter

- **Return Format**

  The query returns the trigger condition of the frequency mask trigge: ENTer, LEAVe, INSide, OUTSide, ELEave, or LENTer.

- **For Example**

  :TRIGger:FMT:CRITeria ENTer

  Select the trigger condition for the frequency mask trigger to ENTer.

  :TRIGger:FMT:CRITeria? The query returns ENTer.

:TRIGger[:SEQuence]:FMT:MASK:EDIT

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK:EDIT {UPPer|LOWer}

  :TRIGger[:SEQuence]:FMT:MASK:EDIT?

- **Functional Description**

  Set the frequency mask trigger for the current selected option.

  UPPer: Upper mask trigger

  LOWer: Lower mask trigger

- **Return Format**

  The query returns the frequency mask trigger for the current selected option: UPPer or LOWer.

- **For Example**

  :TRIGger:FMT:MASK:EDIT UPPer

  Set the frequency mask trigger for the current selected option to UPPer.

  :TRIGger:FMT:MASK:EDIT? The query returns UPPer.

:TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:FMT:MASK:RELative:FREQuency?

- **Functional Description**

  Set the reference switch of the frequency mask for the frequency mask trigger.

  1|ON: Center frequency of frequency reference

  0|OFF: Frequency fixed

- **Return Format**

  The query returns the reference switch status of the frequency mask trigger: 0 or 1.

- **For Example**

:TRIGger:FMT:MASK:RELative:FREQuency ON

Turn on the frequency mask reference for the frequency mask trigger.

:TRIGger:FMT:MASK:RELative:FREQuency? The query returns 1.

:TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude {{1|ON} | {0|OFF}}

  :TRIGger[:SEQuence]:FMT:MASK:RELative:AMPLitude?

- **Functional Description**

  Set the reference switch of the amplitude mask for the frequency mask trigger.

  1|ON: Amplitude reference level

  0|OFF: Amplitude fixed

- **Return Format**

  The query returns the reference switch status of the amplitude mask for the frequency mask trigger:0 or 1.

- **For Example**

  :TRIGger:FMT:MASK:RELative:AMPLitude ON

  Turn on the amplitude mask reference for the frequency mask trigger.

  :TRIGger:FMT:MASK:RELative:AMPLitude? The query returns 1.

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X <freq>

  :TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:X?

- **Functional Description**

  Set the X offset of the mask for the frequency mask trigger.

  <n>: Mask serial number, 1 represents upper mask, and 2 represents lower mask.

  <freq>: A continuous real number, with the default unit in Hz.

- **Return Format**

  The query returns the X offset of the mask for the frequency mask trigger in scientific notation, with the unit in Hz.

- **For Example**

  :TRIGger:FMT:MASK1:OFFSet:X 1MHz

  Set the X offset of the mask for the frequency mask trigger to 1 MHz.

  :TRIGger:FMT:MASK1:OFFSet:X? The query returns 1.000000e+06.

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y <ampl>

  :TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:Y?

- **Functional Description**

  Set the Y offset of the mask for the frequency mask trigger.

  <n>: Mask serial number, 1 represents upper mask, and 2 represents lower mask.

  <ampl>: A continuous real number with the default unit in dB.

- **Return Format**

  The query returns the Y offset of the mask for the frequency mask trigger in scientific notation, with the unit in dB.

- **For Example**

  :TRIGger:FMT:MASK1:OFFSet:Y 1

  Set the Y offset of the mask for the frequency mask trigger to 1 dB.

  :TRIGger:FMT:MASK1:OFFSet:Y? The query returns 1.000000e+00.

:TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:UPDate

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK<n>:OFFSet:UPDate

- **Functional Description**

  Apply the mask offset for the frequency mask trigger.

  <n>: Mask serial number, 1 represents upper mask, and 2 represents lower mask.

- **Return Format**

  No return value.

- **For Example**

  :TRIGger:FMT:MASK1:OFFSet:UPDate

  Apply the upper mask offset for the frequency mask trigger.

:TRIGger[:SEQuence]:FMT:MASK<n>:DELete

- **Command Format**

  :TRIGger[:SEQuence]:FMT:MASK<n>:DELete

- **Functional Description**

  Delete the mask data for the frequency mask trigger.

  <n>: Mask serial number, 1 represents upper mask, and 2 represents lower mask.

- **Return Format**

No return value.

- **For Example**

:TRIGger:FMT:MASK1:DELete Delete the upper mask data for the frequency mask trigger.

:TRIGger[:SEQuence]:FMT:MASK<n>:DATA

- **Command Format**

:TRIGger[:SEQuence]:FMT:MASK<n>:DATA {<freq>,<ampl>,<freq>,<ampl>,...}

:TRIGger[:SEQuence]:FMT:MASK<n>:DATA?

- **Functional Description**

Generate the mask data for the frequency mask trigger.

<n>: Mask serial number, 1 represents upper mask, and 2 represents lower mask.

- **Return Format**

The query returns the specified limit data, using {frequency, amplitude} as a basic unit for newline. The unit for frequency is Hz, and the unit for amplitude is dBm.

- **For Example**

:TRIGger:FMT:MASK1:DATA 1000,-10,1000000,-20

Generate the mask data for the frequency mask trigger.

:TRIGger:FMT:MASK1:DATA?

The query returns 1.000000000e+03,-1.000000e+01,1.000000000e+06,-2.000000e+01

UNIT Command

:UNIT:POWer

- **Command Format**

:UNIT:POWer {DBM|DBMV|DBUV|V|W}

:UNIT:POWer?

- **Functional Description**

Select the unit for the Y-axis scale.

- **Return Format**

The query returns the unit of the Y-axis scale: DBM, DBMV, DBUV, V, or W.

- **For Example**

:UNIT:POWer DBM Select the unit for the Y-axis scale to DBM.

:UNIT:POWer? The query returns DBM.

Vector Network Analysis

CALCulate Command

:CALCulate:MARKer:AOFF

- **Command Format**

  :CALCulate:MARKer:AOFF

- **Functional Description**

  Turn off all the markers.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:AOFF Turn off all the markers.

:CALCulate:MARKer:COFF

- **Command Format**

  :CALCulate:MARKer:COFF

- **Functional Description**

  Turn off all the markers on the traces.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:COFF Turn off all the markers on the traces.

:CALCulate:MARKer:NEXT

- **Command Format**

  :CALCulate:MARKer:NEXT

- **Functional Description**

  Move to the next marker. Only markers that are turned on can be selected. If the current marker is the last one, it will jump to the first marker.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:NEXT Move to the next marker.

:CALCulate:MARKer:CPSearch[:STATe]

- **Command Format**

  :CALCulate:MARKer:CPSearch[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:MARKer:CPSearch[:STATe]?

- **Functional Description**

  Switch the marker continuous peak search to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of marker continuous peak search: 0 or 1.

- **For Example**

:CALCulate:MARKer:CPSearch ON	Turn on the continuous peak search.
:CALCulate:MARKer:CPSearch?	The query returns 1.

:CALCulate:MARKer:MAXimum:LEFT

- **Command Format**

  :CALCulate:MARKer:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:MARKer:MAXimum:LEFT	Execute a next-peak search to the left side.

:CALCulate:MARKer:MAXimum[:MAX]

- **Command Format**

  :CALCulate:MARKer:MAXimum[:MAX]

- **Functional Description**

  Execute a peak search for the specified marker.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:MARKer:MAXimum	Execute a peak search.

:CALCulate:MARKer:MAXimum:NEXT

- **Command Format**

  :CALCulate:MARKer:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:MAXimum:NEXT Execute a next peak search.

:CALCulate:MARKer:MAXimum:RIGHt

- **Command Format**

  :CALCulate:MARKer:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:MAXimum:RIGHt Execute a next-peak search to the right side.

:CALCulate:MARKer:MINimum

- **Command Format**

  :CALCulate:MARKer:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer:MINimum Execute a minimum peak search.

:CALCulate:MARKer:SELect

- **Command Format**

  :CALCulate:MARKer:SELect <integer>

  :CALCulate:MARKer:SELect?

- **Functional Description**

Select a marker by its serial number to be the current marker for the complex spectrum.

<integer>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

The query returns the marker serial number, which ranges from 1 to 10.

- **For Example**

:CALCulate:MARKer:SELect 1	Select marker 1 as the current marker.
:CALCulate:MARKer:SELect?	The query returns 1.

:CALCulate:MARKer:TABLe[:STATe]

- **Command Format**

:CALCulate:MARKer:TABLe[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer:TABLe[:STATe]?

- **Functional Description**

Switch the marker list display to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the display status of the marker list: 0 or 1.

- **For Example**

:CALCulate:MARKer:TABLe ON	The marker list is displayed.
:CALCulate:MARKer:TABLe?	The query returns 1.

:CALCulate:MARKer<n>:LINes[:STATe]

- **Command Format**

:CALCulate:MARKer<n>:LINes[:STATe] {{1|ON} | {0|OFF}}

:CALCulate:MARKer<n>:LINes[:STATe]?

- **Functional Description**

Set the switch for the specified marker line.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

The query returns the switch status of the specified marker line 0 or 1.

- **For Example**

:CALCulate:MARKer1:LINes ON	Turn on the marker line for marker 1.
:CALCulate:MARKer1:LINes?	The query returns 1.

:CALCulate:MARKer<n>:MODE

- **Command Format**

  :CALCulate:MARKer<n>:MODE {OFF|POSition|DELTa }

  :CALCulate:MARKer<n>:MODE?

- **Functional Description**

  Select the marker mode for the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  OFF: Turn off the marker function

  POSition: Normal mode

  DELTa: Difference mode

- **Return Format**

  The query returns the specified marker mode: OFF, POSition, or DELTa.

- **For Example**

  :CALCulate:MARKer1:MODE POSition Set marker 1 mode to POSition.

  :CALCulate:MARKer1:MODE? The query returns POSition.

:CALCulate:MARKer<n>:REFerence

- **Command Format**

  :CALCulate:MARKer<n>:REFerence <integer>

  :CALCulate:MARKer<n>:REFerence?

- **Functional Description**

  Select the reference marker for the specified marker. The reference marker cannot be set as its own reference.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  <integer>: Serial number of another marker, excluding the current one. The value range is 1 to 10.

- **Return Format**

  The query returns the reference marker of the specified marker.

- **For Example**

  :CALCulate:MARKer1:REFerence 2 Set marker 2 as the reference marker for Marker 1.

  :CALCulate:MARKer1:REFerence? Return the reference marker 2 of marker 1.

:CALCulate:MARKer<n>[:SET]:CENTer

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:CENTer

- **Functional Description**

  Set the center frequency to the specified marker's frequency.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:CENTer Set the center frequency to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:RLEVel

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:RLEVel

- **Functional Description**

  Set the reference level to the specified marker amplitude, which is not available when the marker trace format is in Smith or polar coordinates.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:RLEVel Set the reference level to marker 1 amplitude.

:CALCulate:MARKer<n>[:SET]:STARt

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:STARt

- **Functional Description**

  Set the start frequency to the specified marker's frequency.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:MARKer1:STARt Set the start frequency to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:STEP

- **Command Format**

  :CALCulate:MARKer<n>[:SET]:STEP

- **Functional Description**

Set the center frequency step to the specified marker's frequency.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:STEP Set the center frequency step to marker 1 frequency.

:CALCulate:MARKer<n>[:SET]:STOP

- **Command Format**

:CALCulate:MARKer<n>[:SET]:STOP

- **Functional Description**

Set the stop frequency to the specified marker's frequency.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

No return value.

- **For Example**

:CALCulate:MARKer1:STOP Set the stop frequency to marker 1 frequency.

:CALCulate:MARKer<n>:TRACe

- **Command Format**

:CALCulate:MARKer<n>:TRACe <integer>

:CALCulate:MARKer<n>:TRACe?

- **Functional Description**

Set the marker for the specified trace.

<n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

<integer>: Trace serial number; a continuous integer. The range is from 1 to 4.

- **Return Format**

The query returns the trace serial number, which ranges from 1 to 4.

- **For Example**

:CALCulate:MARKer1:TRACe 1 Set marker 1 for trace 1.

:CALCulate:MARKer1:TRACe? The query returns 1.

:CALCulate:MARKer<n>:X

- **Command Format**

:CALCulate:MARKer<n>:X <freq>|<time>

:CALCulate:MARKer<n>:X?

- **Functional Description**

  Adjust the coordinate value of the X-axis for the specified marker and set the corresponding data type based on the X-axis scale type.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

  <freq>: The frequency can be set when the X-axis scale type is frequency or time. The default unit is Hz.

  <time>: The time can be set when the X-axis scale type is time. The default unit is seconds (s).

- **Return Format**

  The query returns the coordinate value of the X-axis for the specified marker in scientific notation. The unit is Hz when the X-axis scale type is frequency or time. The unit is seconds (s) when the X-axis scale type is period or time.

- **For Example**

:CALCulate:MARKer1:X 1GHz	Set the coordinate value of the X-axis for marker 1 to 1 GHz.
:CALCulate:MARKer1:X?	The query returns 1.000000e+09.

:CALCulate:MARKer<n>:Y

- **Command Format**

  :CALCulate:MARKer<n>:Y?

- **Functional Description**

  The query returns the amplitude of the specified marker.

  <n>: Marker serial number; a continuous integer, with a value range of 1 to 10.

- **Return Format**

  The query returns the amplitude of the specified marker in scientific notation

- **For Example**

:CALCulate:MARKer1:Y?	The query returns the amplitude of the specified marker.

:CALCulate:FORMat

- **Command Format**

  :CALCulate:FORMat { MLOGarithmic | PHASe | GDELay | SLINear | SLOGarithmic | SCOMplex | SMITh | SADMittance | PLINear | PLOGarithmic | POLar | MLINear | SWR | REAL | IMAGinary | DRETurnloss | UPHase | PPHase }

  :CALCulate:FORMat?

- **Functional Description**

  Select the trace format based on the trace serial number using the command :TRACe:SELect.

MLOGarithmic: Logarithmic amplitude

PHASe: Phase

GDELay: Group delay

SLINear: Smith (Linear/phase)

SLOGarithmic: Smith (Logarithmic/phase)

SCOMplex: Smith (Real part/ imaginary part)

SMITh: Smith (R+jX)

SADMittance: Smith (G+jB)

PLINear: Polar coordinate (Linear/phase)

PLOGarithmic: Polar coordinate (Logarithmic/phase)

POLar: Polar coordinate (Real part/ imaginary part)

MLINear: Linear amplitude

SWR: Standing wave ratio

REAL: Real part

IMAGinary: Imaginary part

UPHase: Phase extension

PPHase: Positive phase

In S11 mode, these trace format can be set: MLOGarithmic, PHASe, GDELay, SLINear, SLOGarithmic, SCOMplex, SMITh, SADMittance, PLINear, PLOGarithmic, POLar, MLINear, SWR, REAL, IMAGinary, DMLOgarithmic, UPHase, and PPHase.

In S21 mode, these trace format can be set: MLOGarithmic, MLINear, REAL, and IMAGinary.

- **Return Format**

The query returns the trace format for the specified trace: MLOG, PHAS, GDEL, SLIN, SLOG, SCOM, SMIT, SADM, PLIN, PLOG, POL, MLIN, SWR, REAL, IMAG, UPH, or PPH.

- **For Example**

:CALCulate:FORMat MLOGarithmic	Set the trace format to MLOGarithmic.
:CALCulate:FORMat?	The query returns MLOGarithmic.

:CALCulate:TRACe<n>:MATH

- **Command Format**

:CALCulate:TRACe<n>:MATH { DIVide | MULTiply | SUBTract | ADD | OFF }

:CALCulate:TRACe<n>:MATH?

- **Functional Description**

Set the mathematical operation type for the trace data. This can be used when the trace data are saved into the internal storage.

<n>: Trace serial number, an integer ranging from 1 to 4.

DIVide: Data/ internal storage

MULTiply: Data * internal storage

SUBTract: Data - internal storage

ADD: Data + internal storage

OFF: OFF

- **Return Format**

The query returns the mathematical operation type of the trace data: DIV, MULT, SUBT, ADD, or OFF.

- **For Example**

:CALCulate:TRACe1:MATH DIVide

Set the mathematical operation type to "Data/ internal storage" for trace 1.

:CALCulate:TRACe1:MATH?

The query returns the mathematical operation type "Data/ internal storage" of trace 1.

CONFigure Command

:CONFigure:MEAS[:MODE]

- **Command Format**

:CONFigure:MEAS[:MODE] { S11 | S21 }

:CONFigure:MEAS[:MODE]?

- **Functional Description**

Select the measurement mode.

S11: S11 measurement mode

S21: S21 measurement mode

- **Return Format**

The query returns measurement mode: S11 or S2.

- **For Example**

:CONFigure:MEAS S11	Enter the S11 measurement mod.
:CONFigure:MEAS?	The query returns S11.

:CONFigure:MEASure:DEFAult

- **Command Format**

:CONFigure:MEASure:DEFAult

- **Functional Description**

Reset the measurement.

- **Return Format**

  No return value.

- **For Example**

  :CONFigure:MEASure:DEFAult Reset the measurement.

DISPlay Command

:DISPlay:DATA?

- **Command Format**

  :DISPlay:DATA?

- **Functional Description**

  Acquire the screen image.

- **Return Format**

  The query returns the screen image.

- **For Example**

  :DISPlay:DATA? Acquire the screen image.

:DISPlay:WINDow:SPLit

- **Command Format**

  :DISPlay:WINDow:SPLit

  {ONEWindow|LRWindow|UDWindow|LRDWindow|UDRWindow|FOURwindow}

- **Functional Description**

  Set the window layout.

  ONEWindow: Single window

  LRWindow: Two windows, one on the left and one on the right

  UDWindow: Two windows, one on the up and one on the down

  LRDWindow: Three windows, one on the left, one on the right, and one on the down

  UDRWindow: Three windows, one on the up, one on the down, and one on the right

  FOURwindow: Four windows

- **Return Format**

  No return value.

- **For Example**

  :DISPlay:Window:SPLit FOURwindow Set the window layout to FOURwindow.

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

- **Command Format**

  :DISPlay:WINDow:TRACe:Y[:SCALe]:AUTO <integer>

- **Functional Description**

  Set the automatic scale for the window trace.

  <integer>: Window serial number or trace serial number. The window is bound to the trace, so it displays the fixed trace. Adjusting the trace will affect the content of the window. The range is from 1 to 4.

- **Return Format**

  No return value.

- **For Example**

  :DISPlay:WINDow:TRACe:Y:AUTO 1 Set the automatic scale for window trace 1.

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

- **Command Format**

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

- **Functional Description**

  Set the automatic scale for all window traces.

- **Return Format**

  No return value.

- **For Example**

  :DISPlay:WINDow:Y:AUTO Set the automatic scale for all window traces.

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

- **Command Format**

  :DISPlay:WINDow<n>:TRACe:Y[:SCALe]:PDIVision <real>

  :DISPlay:WINDow<n>:TRACe:Y[:SCALe]:PDIVision?

- **Functional Description**

  Adjust the Y-axis scale of the specified measurement window in VNA mode. The unit is determined by the displayed trace format in the window.

  <n>: Window serial number or trace serial number. The window is bound to the trace, so it displays the fixed trace. Adjusting the trace will affect the content of the window. The range is from 1 to 4.

  <real>: A continuous real number

  <real> unit is determined by the trace format:

MLOGarithmic: unit in dB

PHASe: unit in deg

GDELay: unit in ns

Other traces do not have a unit.

- **Return Format**

The query returns the Y-axis scale in scientific notation.

- **For Example**

:DISPlay:WINDow1:TRACe:Y:PDIVision 10dB	Set the Y-axis scale for window 1 to 10 dB.
:DISPlay:WINDow1:TRACe:Y:PDIVision?	The query returns 1.000000e+01.

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

- **Command Format**

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel <real>

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RLEVel?

- **Functional Description**

Adjust the reference level for each measurement window in VNA mode. Note that Smith and polar coordinate windows do not require adjustment of the reference level. The unit is determined by the trace format displayed in the window.

<n>: Window serial number or trace serial number. The window is bound to the trace, so it displays the fixed trace. Adjusting the trace will affect the content of the window. The range is from 1 to 4.

<real>: A continuous real number

<real> unit is determined by the trace format:

MLOGarithmic: unit in dB

PHASe: unit in deg

GDELay: unit in s

Other traces do not have a unit.

- **Return Format**

The query returns the reference level in scientific notation.

- **For Example**

:DISPlay:WINDow1:TRACe:Y:RLEVel -10dB	Set the reference level for window 1 to -10 dB.
:DISPlay:WINDow1:TRACe:Y:RLEVel?	The query returns -1.000000e+01.

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

- **Command Format**

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RPOSition <integer>

:DISPlay:WINDow<n>:TRACe:Y[:SCALe]:RPOSition?

- **Functional Description**

  Adjust the reference position for each measurement window in VNA mode. Note that Smith and polar coordinate windows do not require adjustment of the reference position.

  <n>: Window serial number or trace serial number. The window is bound to the trace, so it displays the fixed trace. Adjusting the trace will affect the content of the window. The range is from 1 to 4.

  <integer>: Reference position; an integer, with a value range of 0 to 10.

- **Return Format**

  The query returns the reference position in scientific notation.

- **For Example**

:DISPlay:WINDow1:TRACe:Y:RPOSition 5	Set the reference position for window 1 to 5.
:DISPlay:WINDow1:TRACe:Y:RPOSition?	The query returns 5.

FETCh Command

:FETCh: TRACe<n>?

- **Command Format**

  :FETCh: TRACe<n>?

- **Functional Description**

  Query the trace data.

  <n>: Trace serial number, ranging from 1 to 4.

  Trace data is arranged as a set of real and imaginary parts. When the trace is in Smith or Polar format, the real part corresponds to the real element of Smith or Polar, and the imaginary part corresponds to the imaginary element. For all other formats, the real part contains the data, while the imaginary part is zero.

- **Return Format**

  The query returns the trace data in scientific notation.

- **For Example**

:FETCh:TRACe1?	The query returns a set of measured results for trace 1.

INITiate Command

:INITiate:CONTinuous

- **Command Format**

:INITiate:CONTinuous {{1|ON} | {0|OFF}}

:INITiate:CONTinuous?

- **Functional Description**

Switch between single sweep and continuous sweep mode.

1|ON: Continuous sweep mode

0|OFF: Single sweep mode

- **Return Format**

The query returns whether the mode is continuous sweep: 0 or 1.

- **For Example**

:INITiate:CONTinuous ON	Continuous sweep mode.
:INITiate:CONTinuous?	The query returns 1.

:INITiate:SINGle

- **Command Format**

:INITiate:SINGle

- **Functional Description**

A single signal.

- **Return Format**

No return value.

- **For Example**

:INITiate:SINGle	A single signal.

INPut Command

:INPut:IMPedance

- **Command Format**

:INPut:IMPedance {50|75}

:INPut:IMPedance?

- **Functional Description**

Select the input impedance 50 Ω or 75 Ω.

- **Return Format**

The query returns the input impedance: 50 Ω or 75 Ω.

- **For Example**

:INPut:IMPedance 50	Set the input impedance to 50 Ω.
:INPut:IMPedance?	The query returns 50

MMEMory Command

:MMEMory:LOAD:STATe

- **Command Format**

  :MMEMory:LOAD:STATe <filename>

- **Functional Description**

  Load the register state file from the default catalogue.

  <filename>: Filename with the ".state" file extension. Provide the entire filename as a character string, and it should be enclosed in quotation marks.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:STATe "vna.state" Load the register state file "vna.state".

:MMEMory:LOAD:TRACe

- **Command Format**

  :MMEMory:LOAD:TRACe {TRACE1|TRACE2|TRACE3|TRACE4,<filename>}

- **Functional Description**

  Load the data from the specified trace.

  TRACE1-TRACE6: Corresponds to trace 1 to trace 4

  <filename>: Filename with the ".trace" file extension.

- **Return Format**

  No return value.

- **For Example**

  :MMEMory:LOAD:TRACe TRACE1,"test.trace"

  Trace 1 loads the data from the file "test.trace".

:MMEMory:STORe:STATe

- **Command Format**

  :MMEMory:STORe:STATe <filename>

- **Functional Description**

  Save the register status to the specified file.

  <filename>: Filename with the ".state" file extension.

- **Return Format**

  No return value.

➢ **For Example**

:MMEMory:STORe:STATe "test.state"

Save the register status to the file “test.state”.

:MMEMory:STORe:TRACe

➢ **Command Format**

:MMEMory:STORe:TRACe {TRACE1|TRACE2|TRACE3|TRACE4,<filename>}

➢ **Functional Description**

Save the specified trace data to the file.

TRACE1-TRACE4: Corresponds to trace 1 to trace 4

<filename>: Filename with the “.trace” file extension.

➢ **Return Format**

No return value.

➢ **For Example**

:MMEMory:STORe:TRACe TRACE1,"test.trace" Save trace 1 data to the file “test.trace”.

SENSe Command

[:SENSe]:AVERage:COUNt

➢ **Command Format**

[:SENSe]:AVERage:COUNt <integer>

[:SENSe]:AVERage:COUNt?

➢ **Functional Description**

Set the average number.

<integer>: An integer with a value range of 1 to 10000.

➢ **Return Format**

The query returns the average number, with a value range of 1 to 10000.

➢ **For Example**

:AVERage:COUNt 10 Set the average number to 10.

:AVERage:COUNt? The query returns 10.

[:SENSe]:BANDwidth|BWIDth[:RESolution]

➢ **Command Format**

[:SENSe]:BANDwidth|BWIDth[:RESolution] <freq>

[:SENSe]:BANDwidth|BWIDth[:RESolution]?

- **Functional Description**

  Adjust the intermediate frequency bandwidth.

  <freq>: A discrete real number; the default unit is Hz, with a value range of 100 Hz to 1MHz.

- **Return Format**

  The query returns intermediate frequency bandwidth in scientific notation, with the unit in Hz.

- **For Example**

:BANDwidth 1MHz	Set the bandwidth to 1 MHz.
:BANDwidth?	The query returns 1.000000e+06.

:SENSe:CORRection:COLLect

- **Command Format**

  :SENSe:CORRection:COLLect { OPEN | SHORt | LOAD | THRU | SAVE }

- **Functional Description**

  Set the calibration. In S11 mode, these setting can be set: open-circuit, short-circuit, and load. In S21 mode, only shoot through can be set.

  OPEN: Open-circuit

  SHORt: Short-circuit

  LOAD: Load

  THRU: Shoot through

  SAVE: Complete

- **Return Format**

  No return value.

- **For Example**

:SENSe:CORRection:COLLect OPEN	Set the calibration to open-circuit.

:SENSe:CORRection:COLLect:CKIT:ARBitrary

- **Command Format**

  :SENSe:CORRection:COLLect:CKIT:ARBitrary <resistance>

  :SENSe:CORRection:COLLect:CKIT:ARBitrary?

- **Functional Description**

  Set the characteristic impedance (Z0) for the standard element. The default resistance is 50 Ω.

  <resistance>: Resistance

- **Return Format**

  The query returns the characteristic impedance (Z0) of the standard element.

- **For Example**

:SENSe:CORRection:COLLect:CKIT:ARBitrary 50	Set R to 50 Ω.
:SENSe:CORRection:COLLect:CKIT:ARBitrary?	The query returns 5.000000e+01.

:SENSe:CORRection:COLLect:CKIT:C<n>

- **Command Format**

  :SENSe:CORRection:COLLect:CKIT:C<n> <real>

  :SENSe:CORRection:COLLect:CKIT:C<n>?

- **Functional Description**

  Set the capacitance parameters for the standard element: C0, C1, C2, and C3. The units are as follows: C0 is in fF, C1 is in fF/GHz, C2 is in fF/GHz^2, and C3 is in fF/GHz^3.

  <n>: An integer number, with a value range of 0 to 3.

  <real>: A continuous real number.

- **Return Format**

  The query returns the capacitance parameter of the standard element.

- **For Example**

:SENSe:CORRection:COLLect:CKIT:C0 -2.71	Set C0 to -2.71fF.
:SENSe:CORRection:COLLect:CKIT:C0?	The query returns -2.71.

:SENSe:CORRection:COLLect:CKIT:L<n>

- **Command Format**

  :SENSe:CORRection:COLLect:CKIT:L<n> <real>

  :SENSe:CORRection:COLLect:CKIT:L<n>?

- **Functional Description**

  Set the inductance parameters for the standard element: L0, L1, L2, and L3. The units are as follows: L0 is in pH, L1 is in pF/GHz, L2 is in pF/GHz^2, and L3 is in pF/GHz^3.

  <n>: An integer number, with a value range of 0 to 3.

  <real>: A continuous real number.

- **Return Format**

  The query returns the inductance parameters of the standard element.

- **For Example**

:SENSe:CORRection:COLLect:CKIT:L0 -18.17	Set L0 to -18.17fF.
:SENSe:CORRection:COLLect:CKIT:L0?	The query returns -18.17

:SENSe:CORRection:COLLect:CKIT:LABel

- **Command Format**

:SENSe:CORRection:COLLect:CKIT:LABel { 3009 | USR1 | USR2 }

:SENSe:CORRection:COLLect:CKIT:LABel?

- **Functional Description**

Select the standard element.

3009: Standard element 3009F/M

USR1: Custom element 1

USR2: Custom element 2

- **Return Format**

The query returns the current selected standard element: 3009, USR1, or USR2.

- **For Example**

:SENSe:CORRection:COLLect:CKIT:LABel 3009 Select the standard element 3009F/M.

:SENSe:CORRection:COLLect:CKIT:LABel? The query returns the standard element 3009.

:SENSe:CORRection:COLLect:CKIT:LOSS

- **Command Format**

:SENSe:CORRection:COLLect:CKIT:LOSS <real>

:SENSe:CORRection:COLLect:CKIT:LOSS?

- **Functional Description**

Set the offset loss for the standard element. The unit is dB/GHz.

<real>: A continuous real number.

- **Return Format**

The query returns the offset loss of the standard element.

- **For Example**

:SENSe:CORRection:COLLect:CKIT:LOSS 0 Set the offset loss for the standard element to 0.

:SENSe:CORRection:COLLect:CKIT:LOSS? The query returns 0.000000e+00.

:SENSe:CORRection:COLLect:CKIT:OFFSet

- **Command Format**

:SENSe:CORRection:COLLect:CKIT:OFFSet <real>

:SENSe:CORRection:COLLect:CKIT:OFFSet?

- **Functional Description**

Set the offset length for the standard element.

<real>:A continuous real number, with the unit in mm.

- **Return Format**

The query returns the offset length of the standard element.

- **For Example**

  :SENSe:CORRection:COLLect:CKIT:OFFSet 14.89mm

  Set the offset length for the standard element to 14.89 mm.

  :SENSe:CORRection:COLLect:CKIT:OFFSet? The query returns 1.489000e-02.

:SENSe:CORRection:COLLect:CKIT:ORDer

- **Command Format**

  :SENSe:CORRection:COLLect:CKIT:ORDer { OPEN | SHORt | LOAD | THRU }

  :SENSe:CORRection:COLLect:CKIT:ORDer?

- **Functional Description**

  Set the type for the standard element.

  OPEN: Open-circuit

  SHORt: Short-circuit

  LOAD: Load

  THRU: Shoot through

- **Return Format**

  The query returns the type of the standard element: OPEN, SHORt, LOAD, or THRU.

- **For Example**

  :SENSe:CORRection:COLLect:CKIT:ORDer OPEN

  Set the type for the standard element to OPEN.

  :SENSe:CORRection:COLLect:CKIT:ORDer? The query returns OPEN.

:SENSe:CORRection:COLLect:CLEar

- **Command Format**

  :SENSe:CORRection:COLLect:CLEar

- **Functional Description**

  Clear the calibration.

- **Return Format**

  No return value.

- **For Example**

  :SENSe:CORRection:COLLect:CLEar Clear the calibration.

[:SENSe]:FREQuency:CENTer

- **Command Format**

  [:SENSe]:FREQuency:CENTer <freq>

[:SENSe]:FREQuency:CENTer?

- **Functional Description**

Set the center frequency for the sweep frequency.

<freq>: A continuous real number with the default unit in Hz.

- **Return Format**

The query returns the center frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer 1GHz Set the center frequency for the sweep frequency to 1 GHz.

:FREQuency:CENTer? The query returns 1.000000e+09.

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

- **Command Format**

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

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

- **Functional Description**

Automatically/manually switch the center frequency step.

1|ON: Automatic (AUTO)

0|OFF: Manual

- **Return Format**

The query returns the status of the center frequency step: 0 or 1.

- **For Example**

:FREQuency:CENTer:STEP:AUTO ON

Set the center frequency step to automatic (AUTO) mode.

:FREQuency:CENTer:STEP:AUTO? The query returns 1.

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

- **Command Format**

[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>

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

- **Functional Description**

Adjust the center frequency step.

<freq>: A continuous real number with the default unit in Hz.

- **Return Format**

The query returns the center frequency step in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer:STEP 10MHz	Set the center frequency step to 10 MHz.
:FREQuency:CENTer:STEP?	The query returns 1.000000e+07.

[:SENSe]:FREQuency:SPAN

- **Command Format**

  [:SENSe]:FREQuency:SPAN <freq>

  [:SENSe]:FREQuency:SPAN?

- **Functional Description**

  Set the span.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the span in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:SPAN 1GHz	Set the span to 1 GHz.
:FREQuency:SPAN?	The query returns 1.000000e+09.

[:SENSe]:FREQuency:SPAN:FULL

- **Command Format**

  [:SENSe]:FREQuency:SPAN:FULL

- **Functional Description**

  Set the span to the full span (maximum span).

- **Return Format**

  No return value.

- **For Example**

:FREQuency:SPAN:FULL	Set the span to the full span (maximum span).

[:SENSe]:FREQuency:SPAN:PREVious

- **Command Format**

  [:SENSe]:FREQuency:SPAN:PREVious

- **Functional Description**

  Set the span to the previous span.

- **Return Format**

  No return value.

- **For Example**

:FREQuency:SPAN:PREVious	Set the span to the previous span.

[:SENSe]:FREQuency:STARt

- **Command Format**

  [:SENSe]:FREQuency:STARt <freq>

  [:SENSe]:FREQuency:STARt?

- **Functional Description**

  Set the start frequency for the sweep frequency function.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the start frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:STARt 10MHz	Set the start frequency to 10 MHz.
:FREQuency:STARt?	The query returns 1.000000e+07.

[:SENSe]:FREQuency:STOP

- **Command Format**

  [:SENSe]:FREQuency:STOP <freq>

  [:SENSe]:FREQuency:STOP?

- **Functional Description**

  Set the stop frequency for the sweep frequency function.

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the stop frequency in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:STOP 1GHz	Set the stop frequency to 1 GHz.
:FREQuency:STOP?	The query returns 1.000000e+09.

[:SENSe]:MEAS:APERture

- **Command Format**

  [:SENSe]:MEAS:APERture <integer>

  [:SENSe]:MEAS:APERture?

- **Functional Description**

  Set the measurement pore diameter. This setting is only valid when the measurement mode is S11.

  <integer>: An integer value, with a range of 1 to 20.

- **Return Format**

The query returns the measurement pore diameter, ranging from 1 to 20.

- **For Example**

:MEAS:APERture 10	Set the measurement pore diameter to 10.
:MEAS:APERture?	The query returns 10.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

[:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

[:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

Set the frequency reference.

INTernal: Internal

EXTernal: External

- **Return Format**

The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

[:SENSe]:SWEep:POINts

- **Command Format**

[:SENSe]:SWEep:POINts <integer>

[:SENSe]:SWEep:POINts?

- **Functional Description**

Adjust the sweep points.

<integer>: A point represented as an integer, ranging from 1 to 10001.

- **Return Format**

The query returns the sweep points.

- **For Example**

:SWEep:POINts 1001	Set the sweep points to 1001.
:SWEep:POINts?	The query returns 1001.

[:SENSe]:SWEep:TIME

- **Command Format**

[:SENSe]:SWEep:TIME <time>

[:SENSe]:SWEep:TIME?

- **Functional Description**

  Set the scanning time.

  <time>: Time value in unit s. The scanning time cannot be set when in the zero span mode. In non-zero span mode, minimum value is confirmed according to the number of scan points, maximum value is 4 ks.

- **Return Format**

  The query returns the scanning time in scientific notation. The unit is seconds (s).

- **For Example**

:SWEep:TIME 10s	Set the scanning time to 10 s.
:SWEep:TIME?	The query returns 1.000000e+01.

[:SENSe]:SWEep:TIME:AUTO

- **Command Format**

  [:SENSe]:SWEep:TIME:AUTO {{1|ON} | {0|OFF}}

  [:SENSe]:SWEep:TIME:AUTO?

- **Functional Description**

  Automatically/manually switch the scanning time.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the scanning time: 0 or 1.

- **For Example**

:SWEep:TIME:AUTO ON	Set the scanning time to automatic (AUTO) mode.
:SWEep:TIME:AUTO?	The query returns 1.

SOURce Command

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

- **Command Format**

  :SOURce[:EXTernal]:POWer[:LEVel][:IMMediate][:AMPLitude] <ampl>

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

- **Functional Description**

  Adjust the amplitude for the tracking generator.

  <ampl>: A continuous real number with the default unit in dBm.

- **Return Format**

  The query returns the amplitude of the tracking generator in scientific notation. The unit is dBm.

- **For Example**

:SOURce:POWer -40	Set the amplitude for the tracking generator to -40 dBm.
:SOURce:POWer?	The query returns -4.000000e+01.

TRACe Command

:TRACe:SELect

- **Command Format**

  :TRACe:SELect <integer>

  :TRACe:SELect?

- **Functional Description**

  Select a trace.

  <integer>: Trace serial number; a continuous integer. The range is from 1 to 4.

- **Return Format**

  The query returns the serial number of the current trace. The range is from 1 to 4.

- **For Example**

:TRACe:SELect 1	Select trace 1 to the current trace.
:TRACe1:SELect?	The query returns 1.

:TRACe:MEMorize

- **Command Format**

  :TRACe:MEMorize <integer>

- **Functional Description**

  Save the specified trace data to the internal storage.

  <integer>: Trace serial number, an integer with a range of 1 to 4.

- **Return Format**

  No return value.

- **For Example**

:TRACe:MEMorize 1	Save trace data 1 to the internal storage.

:TRACe:NEXT

- **Command Format**

  :TRACe:NEXT

- **Functional Description**

  Select the next trace. If the current trace is the last one, select the first trace.

- **Return Format**

  No return value.

- **For Example**

  :TRACe:NEXT Select the next trace.

:TRACe<n>:DISPlay

- **Command Format**

  :TRACe<n>:DISPlay { DATA | MEM | DATAmem | OFF }

  :TRACe<n>:DISPlay?

- **Functional Description**

  Select the trace display type for the specified trace.

  <n>: Trace serial number, an integer ranging from 1 to 4.

  DATA: Data

  MEM: Internal storage

  DATAmem: Data and internal storage

  OFF: OFF

- **Return Format**

  The query returns the trace display type of the specified trace: DATA, MEM, DATAmem, or OFF.

- **For Example**

  :TRACe1:DISPlay DATA Set the trace display type for trace 1 to DATA.

  :TRACe1:DISPlay? The query returns DATA.

:TRACe<n>:TYPE

- **Command Format**

  :TRACe<n>:TYPE { WRITe | AVERage | MAXHold | MINHold }

  :TRACe<n>:TYPE?

- **Functional Description**

  Select the trace type for the specified trace.

  <n>: Trace serial number, an integer ranging from 1 to 4.

  WRITe: Refresh

  AVERage: Trace average

  MAXHold: Maximum hold

  MINHold: Minimum hold

- **Return Format**

  The query returns the trace type of the specified trace: WRITe, AVERage, MAXHold, or MINHold.

- **For Example**

:TRACe1:TYPE AVERage	Set trace 1 to AVERage.
:TRACe1:TYPE?	The query returns AVERage

:TRACe<n>:UPDate:STATe

- **Command Format**

  :TRACe<n>:UPDate:STATe {{1|ON} | {0|OFF}}

  :TRACe<n>:UPDate:STATe?

- **Functional Description**

  Set the trace refresh switch. The trace will continue refreshing even when it is turned off.

  <n>: Trace serial number, an integer number with a value range of 1 to the maximum trace.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the refresh status of the specified trace: 0 or 1.

- **For Example**

:TRACe1:UPDate:STATe ON	Turn on the trace refresh for trace 1.
:TRACe1:UPDate:STATe?	The query returns 1.

Vector Signal Analysis

CALCulate Command

:CALCulate:DDEMod:MARKer:AOFF

- **Command Format**

  :CALCulate:DDEMod:MARKer:AOFF

- **Functional Description**

  Turn off all the markers.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:DDEMod:MARKer:AOFF	Turn off all the markers.

:CALCulate:DDEMod:MARKer:SELect

- **Command Format**

  :CALCulate:DDEMod:MARKer:SELect <integer>

  :CALCulate:DDEMod:MARKer:SELect?

- **Functional Description**

  Select a marker by its serial number to be the current marker for the complex spectrum.

  <integer>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  The query returns the marker serial number, which ranges from 1 to 6.

- **For Example**

:CALCulate:DDEMod:MARKer:SELect 1	Select marker 1 as the current marker.
:CALCulate:DDEMod:MARKer:SELect?	The query returns 1.

:CALCulate:DDEMod:MARKer:TABLe[:STATe]

- **Command Format**

  :CALCulate:DDEMod:MARKer:TABLe[:STATe] {{1|ON} | {0|OFF}}

  :CALCulate:DDEMod:MARKer:TABLe[:STATe]?

- **Functional Description**

  Switch the marker list to on or off.

  1|ON: ON

  0|OFF: OFF

- **Return Format**

  The query returns the switch status of the marker list: 0 or 1.

- **For Example**

:CALCulate:DDEMod:MARKer:TABLe ON	Turn on the marker list.
:CALCulate:DDEMod:MARKer:TABLe?	The query returns 1.

:CALCulate:DDEMod:MARKer<n>:MAXimum

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MAXimum

- **Functional Description**

  Execute a peak search for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:DDEMod:MARKer1:MAXimum Marker 1 performs a peak search.

:CALCulate:DDEMod:MARKer<n>:MAXimum:NEXT

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MAXimum:NEXT

- **Functional Description**

  Execute a next peak search for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:DDEMod:MARKer1:MAXimum:NEXT Marker 1 performs a next-peak search.

:CALCulate:DDEMod:MARKer<n>:MAXimum:LEFT

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MAXimum:LEFT

- **Functional Description**

  Execute a next-peak search to the left side for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:DDEMod:MARKer1:MAXimum:LEFT

  Marker 1 performs a next-peak search to the left side.

:CALCulate:DDEMod:MARKer<n>:MAXimum:RIGHt

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MAXimum:RIGHt

- **Functional Description**

  Execute a next-peak search to the right side for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

:CALCulate:DDEMod:MARKer1:MAXimum:RIGHt

Marker 1 performs a next-peak search on the right side.

:CALCulate:DDEMod:MARKer<n>:MINimum

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MINimum

- **Functional Description**

  Execute a minimum peak search for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

  No return value.

- **For Example**

  :CALCulate:DDEMod:MARKer1:MINimum Marker 1 performs a minimum peak search.

:CALCulate:DDEMod:MARKer<n>:MODE

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:MODE {OFF|POSition|DELTa|FIXed}

  :CALCulate:DDEMod:MARKer<n>:MODE?

- **Functional Description**

  Select the marker mode for the specified marker.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

  OFF: Turn off the marker function

  POSition: Normal mode

  DELTa: Difference mode

  FIXed: Fixed mode

- **Return Format**

  The query returns the specified marker's mode: OFF, POSition, DELTa, or FIXed.

- **For Example**

  :CALCulate:DDEMod:MARKer1:MODE POSition Set marker 1 mode to POSition.

  :CALCulate:DDEMod:MARKer1:MODE? The query returns POSition.

:CALCulate:DDEMod:MARKer<n>:PTPeak

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:PTPeak

- **Functional Description**

Execute a peak-to-peak search for the specified marker

<n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

No return value.

- **For Example**

:CALCulate:DDEMod:MARKer1:PTPeak Marker 1 performs a peak-to-peak search.

:CALCulate:DDEMod:MARKer<n>:REFerence

- **Command Format**

:CALCulate:DDEMod:MARKer<n>:REFerence <integer>

:CALCulate:DDEMod:MARKer<n>:REFerence?

- **Functional Description**

Select the reference marker for the specified marker. The reference marker cannot be set as its own reference.

<n>: Marker serial number; a continuous integer. The range is from 1 to 6.

<integer>: Serial number of another marker, excluding the current one. The value range is 1 to 6.

- **Return Format**

The query returns the reference marker of the specified marker.

- **For Example**

:CALCulate:DDEMod:MARKer1:REFerence 2

Set marker 2 as the reference marker for Marker 1.

:CALCulate:DDEMod:MARKer1:REFerence? The query returns 2.

:CALCulate:DDEMod:MARKer<n>:TRACe

- **Command Format**

:CALCulate:DDEMod:MARKer<n>:TRACe <integer>

:CALCulate:DDEMod:MARKer<n>:TRACe?

- **Functional Description**

Select the trace for the specified marker.

<n>: Marker serial number; a continuous integer. The range is from 1 to 6.

<integer>: Marker serial number; a continuous integer with a value range of 1 to 4.

- **Return Format**

The query returns the trace serial number of the specified marker.

- **For Example**

:CALCulate:DDEMod:MARKer1:TRACe 1	Set Marker 1 to correspond with Trace 1.
:CALCulate:DDEMod:MARKer1:TRACe?	The query returns 1.

:CALCulate:DDEMod:MARKer<n>:X

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:X <freq>|<time>|<symbol>

  :CALCulate:DDEMod:MARKer<n>:X?

- **Functional Description**

  Adjust the X-axis coordinate value for the specified marker and set the corresponding data type based on the X-axis scale type.

  <n>: Marker serial number; a continuous integer. The range is from 1 to 6.

  <freq>: The data source for the corresponding trace is frequency domain data from capture data, measurement reference data, or demodulation error data. This applies when the data format is not vector, constellation, or eye diagram. The X-axis value is expressed as frequency, with the default unit in Hz.

  <time>: The data source for the corresponding trace is time domain data from capture data. The X-axis value is expressed as time, with the default unit is seconds (s).

  <symbol>: For all cases not mentioned above, X-axis values are expressed as symbols with no units. For traces formatted as IQ vectors and constellations, the X-axis represents the real and imaginary parts of the IQ data, where X is perpendicular to the current 2D axis.

- **Return Format**

  The query returns the X-axis coordinate value of the specified marker in scientific notation.

- **For Example**

  :CALCulate:DDEMod:MARKer1:X 1GHz

  Set the coordinate value of the X-axis for marker 1 to 1 GHz.

  :CALCulate:DDEMod:MARKer1:X? The query returns 1.000000e+09.

:CALCulate:DDEMod:MARKer<n>:Y[:REAL]

- **Command Format**

  :CALCulate:DDEMod:MARKer<n>:Y[:REAL] <real>

  :CALCulate:DDEMod:MARKer<n>:Y[:REAL]?

- **Functional Description**

  Adjust the Y value for the specified marker. This setting is only effective when the marker type is fixed. For traces formatted as IQ vectors or constellations, the coordinate system represents IQ data with both real and imaginary parts as Y values. The horizontal coordinate corresponds

to the real part, while the vertical coordinate corresponds to the imaginary part. The current command changes the Y value, which only affects the real part, causing the marker to move horizontally. To adjust the imaginary part, use the :CALCulate:DDEMod:MARKer<n>:Y:IMAGinary. Note that the X value is perpendicular to the 2D coordinate system on the screen, and changing the X value does not move the marker.

<n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

The query returns the Y-axis coordinate value of the specified marker in scientific notation.

- **For Example**

:CALCulate:DDEMod:MARKer1:Y 1	Set the amplitude value of marker 1 to -50.
:CALCulate:DDEMod:MARKer1:Y?	The query returns 1.000000e+00.

:CALCulate:DDEMod:MARKer<n>:Y:IMAGinary

- **Command Format**

:CALCulate:DDEMod:MARKer<n>:Y:IMAGinary <real>

:CALCulate:DDEMod:MARKer<n>:Y:IMAGinary?

- **Functional Description**

Adjust the imaginary Y value for the specified marker. This setting is effective only when the marker type is fixed and the trace format is either IQ vector or constellation. Changing the imaginary part will move the marker vertically.

<n>: Marker serial number; a continuous integer. The range is from 1 to 6.

- **Return Format**

The query returns the Y-axis coordinate value of the specified marker in scientific notation.

- **For Example**

:CALCulate:DDEMod:MARKer1:Y:IMAGinary -1	Set the imaginary value of marker 1 to -1.
:CALCulate:DDEMod:MARKer1:Y:IMAGinary?	The query returns -1.000000e+00.

DISPlay Command

:DISPlay:DDEMod:TRACe:DDEMod:SYMBol:FORMat

- **Command Format**

:DISPlay:DDEMod:TRACe:DDEMod:SYMBol:FORMat {HEXadecimal|BINary}

:DISPlay:DDEMod:TRACe:DDEMod:SYMBol:FORMat?

- **Functional Description**

Select the trace symbol table format.

HEXadecimal: Hexadecimal system

BINary: Binary system

- **Return Format**

The query returns the trace symbol table format.

- **For Example**

:DISPlay:DDEMod:TRACe:DDEMod:SYMBol:FORMat BINary

Select the trace symbol table format to binary system.

:DISPlay:DDEMod:TRACe:DDEMod:SYMBol:FORMat? The query returns BINary.

:DISPlay:DDEMod:TRACe:SELect

- **Command Format**

:DISPlay:DDEMod:TRACe:SELect <integer>

:DISPlay:DDEMod:TRACe:SELect?

- **Functional Description**

Select a trace from all available traces to be the current trace.

<integer>: Trace serial number, an integer with a range of 1 to 4.

- **Return Format**

The query returns the serial number of the current trace.

- **For Example**

:DISPlay:DDEMod:TRACe:SELect 1 Set trace 1 to the current trace.

:DISPlay:DDEMod:TRACe:SELect? The query returns 1.

:DISPlay:DDEMod:TRACe<n>:FEED

- **Command Format**

:DISPlay:DDEMod:TRACe<n>:FEED

{Time1|Spectrum1|IQ Meas Time1|IQ Meas Spec1|IQ Ref Time1|IQ Ref Spec1|

Error Vector Time1|Error Vector Spec1|IQ Mag Error1|IQ Phase Error1|Syms/Errs1}

:DISPlay:DDEMod:TRACe<n>:FEED?

- **Functional Description**

Select the specified trace data source.

<n>: Trace serial number, an integer ranging from 1 to 4.

Time1: Captured time domain data

Spectrum1: Captured frequency domain data

IQ Meas Time1: Measured time domain data

IQ Meas Spec1: Measured frequency domain data

IQ Ref Time1: Reference time domain data

IQ Ref Spec1: Reference frequency domain data

Error Vector Time1: Error time domain vector

Error Vector Spec1: Error frequency domain vector

IQ Mag Error1: IQ magnitude error

IQ Phase Error1: IQ phase error

Syms/Errs1: Error summary

- **Return Format**

The query returns the specified trace data source.

- **For Example**

:DISPlay:DDEMod:TRACe1:FEED "IQ Meas Time1"

Select time domain measurement data for trace 1.

:DISPlay:DDEMod:TRACe1:FEED? The query returns IQ Meas Time1.

:DISPlay:DDEMod:TRACe<n>:FORMat

- **Command Format**

:DISPlay:DDEMod:TRACe<n>:FORMat

{MLOG|MLINe|REAL|IMAGinary|VECTor|CONStellation|IEYE|QEYE}

:DISPlay:DDEMod:TRACe<n>:FORMat?

- **Functional Description**

Select the specified trace format.

<n>: Trace serial number, an integer ranging from 1 to 4.

MLOG: Logarithmic amplitude

MLINe: Linear amplitude

REAL: Real part

IMAGinary: Imaginary part

VECTor: IQ vector

CONStellation: Constellation diagram

IEYE: I-eye diagram

QEYE: Q-eye diagram

- **Return Format**

The query returns the specified trace format.

- **For Example**

:DISPlay:DDEMod:TRACe1:FORMat CONStellation

Select the constellation diagram format for trace 1.

:DISPlay:DDEMod:TRACe1:FORMat? The query returns CONStellation.

:DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:RLEVel

- **Command Format**

  :DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:RLEVel <real>

  :DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:RLEVel?

- **Functional Description**

  Set the Y-axis reference for the specified trace.

  <n>: Trace serial number, an integer ranging from 1 to 4.

- **Return Format**

  The query returns the Y-axis reference of the specified trace in scientific notation.

- **For Example**

  :DISPlay:DDEMod:TRACe1:Y:RLEVel 10 Set the Y-axis reference for trace1 to 10.

  :DISPlay:DDEMod:TRACe1:Y:RLEVel ? The query returns 1.000000e+01.

:DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:PDIVision

- **Command Format**

  :DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:PDIVision <real>

  :DISPlay:DDEMod:TRACe<n>:Y[:SCALe]:PDIVision?

- **Functional Description**

  Set the Y-axis scale for the specified trace.

  <n>: Trace serial number, an integer ranging from 1 to 4.

- **Return Format**

  The query returns the Y-axis scale of the specified trace in scientific notation.

- **For Example**

  :DISPlay:DDEMod:TRACe1:Y:PDIVision 10 Set the Y-axis scale for trace1 to 10.

  :DISPlay:DDEMod:TRACe1:Y:PDIVision? The query returns 1.000000e+01.

FETCh Command

:FETCh:DDEMod[1]?

- **Command Format**

  :FETCh:DDEMod[1]?

- **Functional Description**

  Query the result data for the vector signal analysis measurement. The data consists of the

following numbers, arranged in sequence:

1. Root Mean Square (RMS) of Error Vector Magnitude (EVM): percentage data, with the unit %rms. This data is invalid under frequency shift keying (FSK) modulation.
2. Peak of EVM: percentage data, with the unit %pk. This data is invalid under FSK modulation.
3. Peak Position of EVM. This data is invalid under FSK modulation.
4. RMS of EVM offset: percentage data, with the unit %rms. This data is only valid under offset quadrature phase shift keying (OQPSK) modulation.
5. Peak of EVM offset: percentage data, with the unit %pk. This data is only valid under offset quadrature phase shift keying (OQPSK) modulation.
6. Peak Position of EVM offset. This data is only valid under OQPSK.
7. RMS of Error for FSK modulation: percentage data, with the unit %rms. This data is only valid under FSK modulation.
8. Peak of Error for FSK modulation: percentage data, with the unit %pk. This data is only valid under FSK modulation.
9. Peak Position of Error for FSK modulation. This data is only valid under FSK modulation.
10. RMS of Magnitude Error: percentage data: percentage data, with the unit %rms.
11. Peak of Magnitude Error: percentage data: percentage data, with the unit %pk.
12. Peak Position of Magnitude Error.
13. RMS of Phase Error: percentage data, with the unit %rms. This data is invalid under FSK modulation.
14. Peak of Phase Error: percentage data, with the unit %pk. This data is invalid under FSK modulation.
15. Peak Position of Phase Error. This data is invalid under FSK modulation.
16. Frequency Error: frequency value, with the unit Hz. This data is invalid under FSK modulation.
17. Carrier Offset: frequency value, with the unit Hz. This data is only valid under FSK modulation.
18. Signal-to-Noise Ratio (SNR), with the unit dB. This data is valid under QPSK (Quadrature Phase Shift Keying), QAM (Quadrature Amplitude Modulation), APSK (Absolute Phase Shift Keying), and VSB (Vestigial Sideband) modulation.
19. Frequency Offset for FSK modulation: frequency value, with the unit Hz. This data is only valid under FSK modulation.
20. IQ Data Offset, with the unit dB. This data is invalid under FSK modulation.
21. Amplitude Droop, with the unit dB/sym. This data is invalid under QPSK (Quadrature

Phase Shift Keying), OQPSK (Offset Quadrature Phase Shift Keying), MSK1 (Minimum Shift Keying), and FSK (Frequency Shift Keying).

22. ρ (rho): This data is only valid under QPSK/OQPSK modulation.
23. Quadrature Error, with the unit degrees (deg). This data is invalid under BPSK (Binary Phase Shift Keying) and FSK modulation.
24. Gain Equalization: This data is invalid under BPSK and FSK modulation.
25. 2FSK Symbol Clock Error, with the unit ppm.
26. 2FSK Zero-crossing Error RMS, with the unit %rms.
27. 2FSK Zero-crossing Error peak, with the unit %pk.

- **Return Format**

The query returns the vector signal analysis measurement results in scientific notation.

- **For Example**

:FETCh:DDEMod? Query the result data for the vector signal analysis measurement.

INITiate Command

:INITiate:CONTinuous

- **Command Format**

:INITiate:CONTinuous {{1|ON} | {0|OFF}}

:INITiate:CONTinuous?

- **Functional Description**

Switch between single sweep and continuous sweep mode.

1|ON: Continuous sweep mode

0|OFF: Single sweep mode

- **Return Format**

The query returns whether the mode is continuous sweep: 0 or 1.

- **For Example**

:INITiate:CONTinuous ON Continuous sweep mode.

:INITiate:CONTinuous? The query returns 1.

READ Command

:READ:DDEMod[1]?

- **Command Format**

:READ:DDEMod[1]?

- **Functional Description**

Query the result data for the vector signal analysis measurement. The data consists of the following numbers, arranged in sequence:

1. Root Mean Square (RMS) of Error Vector Magnitude (EVM): percentage data, with the unit %rms. This data is invalid under frequency shift keying (FSK) modulation.
2. Peak of EVM: percentage data, with the unit %pk. This data is invalid under FSK modulation.
3. Peak Position of EVM. This data is invalid under FSK modulation.
4. RMS of EVM offset: percentage data, with the unit %rms. This data is only valid under offset quadrature phase shift keying (OQPSK) modulation.
5. Peak of EVM offset: percentage data, with the unit %pk. This data is only valid under offset quadrature phase shift keying (OQPSK) modulation.
6. Peak Position of EVM offset. This data is only valid under OQPSK.
7. RMS of Error for FSK modulation: percentage data, with the unit %rms. This data is only valid under FSK modulation.
8. Peak of Error for FSK modulation: percentage data, with the unit %pk. This data is only valid under FSK modulation.
9. Peak Position of Error for FSK modulation. This data is only valid under FSK modulation.
10. RMS of Magnitude Error: percentage data: percentage data, with the unit %rms.
11. Peak of Magnitude Error: percentage data: percentage data, with the unit %pk.
12. Peak Position of Magnitude Error.
13. RMS of Phase Error: percentage data, with the unit %rms. This data is invalid under FSK modulation.
14. Peak of Phase Error: percentage data, with the unit %pk. This data is invalid under FSK modulation.
15. Peak Position of Phase Error. This data is invalid under FSK modulation.
16. Frequency Error: frequency value, with the unit Hz. This data is invalid under FSK modulation.
17. Carrier Offset: frequency value, with the unit Hz. This data is only valid under FSK modulation.
18. Signal-to-Noise Ratio (SNR), with the unit dB. This data is valid under QPSK (Quadrature Phase Shift Keying), QAM (Quadrature Amplitude Modulation), APSK (Absolute Phase Shift Keying), and VSB (Vestigial Sideband) modulation.
19. Frequency Offset for FSK modulation: frequency value, with the unit Hz. This data is only valid under FSK modulation.
20. IQ Data Offset, with the unit dB. This data is invalid under FSK modulation.

21. Amplitude Droop, with the unit dB/sym. This data is invalid under QPSK (Quadrature Phase Shift Keying), OQPSK (Offset Quadrature Phase Shift Keying), MSK1 (Minimum Shift Keying), and FSK (Frequency Shift Keying).
22. ρ (rho): This data is only valid under QPSK/OQPSK modulation.
23. Quadrature Error, with the unit degrees (deg). This data is invalid under BPSK (Binary Phase Shift Keying) and FSK modulation.
24. Gain Equalization: This data is invalid under BPSK and FSK modulation.
25. 2FSK Symbol Clock Error, with the unit ppm.
26. 2FSK Zero-crossing Error RMS, with the unit %rms.
27. 2FSK Zero-crossing Error peak, with the unit %pk.

- **Return Format**

  The query returns the vector signal analysis measurement results in scientific notation.

- **For Example**

  :READ:DDEMod? Query the result data for the vector signal analysis measurement.

SENSe Command

[:SENSe]:DDEMod:ALPHa

- **Command Format**

  [:SENSe]:DDEMod:ALPHa <real>

  [:SENSe]:DDEMod:ALPHa?

- **Functional Description**

  Set the filter coefficient.

- **Return Format**

  The query returns he filter coefficient in scientific notation.

- **For Example**

  DDEMod:ALPHa 0.2 Set the filter coefficient to 0.2.

  DDEMod:ALPHa? The query returns 2.000000e-01.

[:SENSe]:DDEMod:AVERage[:STATe]

- **Command Format**

  [:SENSe]:DDEMod:AVERage[:STATe] {{1|ON} | {0|OFF}}

  [:SENSe]:DDEMod:AVERage[:STATe]?

- **Functional Description**

  Switch the average function to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the average status: 0 or 1.

- **For Example**

:DDEMod:AVERage ON	Turn on the average function.
:DDEMod:AVERage?	The query returns 1.

[:SENSe]:DDEMod:AVERage:COUNt

- **Command Format**

[:SENSe]:DDEMod:AVERage:COUNt <integer>

[:SENSe]:DDEMod:AVERage:COUNt?

- **Functional Description**

Adjust the average time.

<integer>: An integer with a value range of 1 to 10000.

- **Return Format**

The query returns the average time, which ranges from 1 to 10000.

- **For Example**

:DDEMod:AVERage:COUNt 100	Set the average time to 100.
:DDEMod:AVERage:COUNt?	The query returns 100.

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

- **Command Format**

[:SENSe]:DDEMod:FFT:WINDow[:TYPE] {HANNing|FLATtop|GAUSsian|BLACkman|BHARris}

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

- **Functional Description**

Select the measurement filter.

HANNing: Hanning

FLATtop: Flat top

GAUSsian: Gaussian

BLACkman: Blackman

BHARris: Blackman-harris

- **Return Format**

The query returns the measurement filter.

- **For Example**

:DDEMod:FFT:WINDow GAUSsian	Select the measurement filter to RRCosine.
:DDEMod:FFT:WINDow?	The query returns RRCosine.

[:SENSe]:DDEMod:FILTer:MEASurement

- **Command Format**

  [:SENSe]:DDEMod:FILTer:MEASurement {NONE|RRCosine|GAUSsian|EDGE|IS95EQ|RECTangle}

  [:SENSe]:DDEMod:FILTer:MEASurement?

- **Functional Description**

  Select the measurement filter.

  NONE: None

  RRCosine: Root Raised Cosine filter

  GAUSsian: Gaussian filter

  EDGE: Edge filter

  IS95EQ: CDMA (IS–95A Base EQ) filter

  RECTangle: Rectangle filter

- **Return Format**

  The query returns the measurement filter.

- **For Example**

:DDEMod:FILTer:MEASurement RRCosine	Select the measurement filter to RRCosine.
:DDEMod:FILTer:MEASurement?	The query returns RRCosine.

[:SENSe]:DDEMod:FILTer:REFerence

- **Command Format**

  [:SENSe]:DDEMod:FILTer:REFerence {RCOSine|RRCosine|GAUSsian|EDGE|IS95BB|RECTangle}

  [:SENSe]:DDEMod:FILTer:REFerence?

- **Functional Description**

  Select the reference filter.

  RCOSine: Raised Cosine filter

  RRCosine: Root Raised Cosine filter

  GAUSsian: Gaussian filter

  EDGE: Edge filter

  IS95EQ: CDMA (IS–95A Base EQ) filter

  RECTangle: Rectangle filter

- **Return Format**

  The query returns the reference filter.

➢ **For Example**

:DDEMod:FILTer:REFerence RCOSine	Select the reference filter to RCOSine.
:DDEMod:FILTer:REFerence?	The query returns RCOSine.

[:SENSe]:DDEMod:MODulation

➢ **Command Format**

[:SENSe]:DDEMod:MODulation

{FSK2|FSK4|FSK8|FSK16|BPSK|QPSK|PSK8|PI4DQPSK|OQPSK|DQPSK|QAM16|QAM32|QAM64|QAM128|QAM256|ASK2|ASK4|ASK8|ASK16|MSK1|MSK2}

[:SENSe]:DDEMod:MODulation?

➢ **Functional Description**

Select the modulation type.

FSK2: 2-frequency shift keying modulation

FSK4: 4-frequency shift keying modulation

FSK8: 8-frequency shift keying modulation

FSK16: 16-frequency shift keying modulation

BPSK: 2-phase shift keying modulation

QPSK: 4-phase shift keying modulation

PSK8: 8-phase shift keying modulation

OQPSK: Offset quadrature phase shift keying modulation

DQPSK: Differential quadrature phase shift keying modulation

PI4DQPSK: π/4-DQPSK

QAM16: 16-quadrature amplitude modulation

QAM32: 32-quadrature amplitude modulation

QAM64: 64-quadrature amplitude modulation

QAM128: 128-quadrature amplitude modulation

QAM256: 256-quadrature amplitude modulation

ASK2: 2-amplitude shift keying modulation

ASK4: 4-amplitude shift keying modulation

ASK8: 8-amplitude shift keying modulation

ASK16: 16-amplitude shift keying modulation

MSK1: Minimum frequency shift keying modulation 1

MSK2: Minimum frequency shift keying modulation 2

➢ **Return Format**

The query returns the modulation type.

- **For Example**

:DDEMod:MODulation QAM16	Select the modulation type QAM16.
:DDEMod:MODulation?	The query returns QAM16 QAM16.

[:SENSe]:DDEMod:PPSYmbol

- **Command Format**

  [:SENSe]:DDEMod:PPSYmbol <integer>

  [:SENSe]:DDEMod:PPSYmbol?

- **Functional Description**

  Set the sample point or code element.

  <integer>: An integer value, typically a power of 2.

- **Return Format**

  The query returns the sample point or code element.

- **For Example**

:DDEMod:PPSYmbol 16	Select the reference filter to RCOSine.
:DDEMod:PPSYmbol?	The query returns 16.

[:SENSe]:DDEMod:SRATe

- **Command Format**

  [:SENSe]:DDEMod:SRATe <freq>

  [:SENSe]:DDEMod:SRATe?

- **Functional Description**

  Set the code rate.

  <freq>: Frequency value with the default unit in Hz.

- **Return Format**

  The query returns the code rate in scientific notation.

- **For Example**

:DDEMod:SRATe 32kHz	Set the code rate to 32 kHz.
:DDEMod:SRATe?	The query returns 3.2000000000e+04.

[:SENSe]:DDEMod:STANdard:PRESet

- **Command Format**

  [:SENSe]:DDEMod:STANdard:PRESet

  {GSM|NADC|WCDMA|PDC|PHS|BLUETOOTH|WLAN11B|ZIGBEE868|ZIGBEE915|ZIGBEE2450|TETRA|DECT|APCO}

[:SENSe]:DDEMod:STANdard:PRESet?

- **Functional Description**

Select the preset standard.

GSM: Global System for Mobile Communications, the second-generation mobile communication technology standard.

NADC: North American Digital Cellular

WCDMA: Wideband Code Division Multiple Access is a standard developed by the International Telecommunication Union (ITU)

PDC: Personal Digital Cellular is a second-generation (2G) mobile communication standard developed and used in Japan.

PHS: Personal Handy-phone System, also known as "Little Smart".

BLUETOOTH: A wireless technology for short-range device communication (typically within 10 meters).

WLAN11B: Wireless Local Area Network communication standard

ZIGBEE868/ZIGBEE915/ZIGBEE2450: A low-speed, short-range wireless network protocol that uses license-free Industrial, Scientific, and Medical (ISM) bands, with frequencies: 868 MHz (Europe), 915 MHz (USA), and 2.4 GHz (global).

TETRA: Terrestrial Trunked Radio, a wireless trunked mobile communication system based on Time Division Multiple Access (TDMA) technology

DECT: Terrestrial Trunked Radio, a wireless trunked mobile communication system based on Time Division Multiple Access (TDMA) technology.

APCO: Association of Public-Safety Communications Officials, a U.S. radio communication standard for walkie-talkies.

- **Return Format**

The query returns the preset standard.

- **For Example**

:DDEMod:STANdard:PRESet 1000	Select the preset standard to WCDMA.
:DDEMod:STANdard:PRESet?	The query returns WCDMA.

[:SENSe]:DDEMod:SWEep:POINts

- **Command Format**

[:SENSe]:DDEMod:SWEep:POINts <integer>

[:SENSe]:DDEMod:SWEep:POINts?

- **Functional Description**

Set the measurement length.

<integer>: An integer value.

- **Return Format**

  The query returns the measurement length.

- **For Example**

  :DDEMod:SWEep:POINts 1000 Set the measurement length to 1000.

  :DDEMod:SWEep:POINts? The query returns 1000.

[:SENSe]:DDEMod:SYNC:BURSt:OFFSet

- **Command Format**

  [:SENSe]:DDEMod:SYNC:BURSt:OFFSet <integer>

  [:SENSe]:DDEMod:SYNC:BURSt:OFFSet?

- **Functional Description**

  Set the burst search offset.

  <integer>: An integer value.

- **Return Format**

  The query returns the burst search offset.

- **For Example**

  :DDEMod:SYNC:BURSt:OFFSet 64 Set the burst search offset to 64.

  :DDEMod:SYNC:BURSt:OFFSet? The query returns 64.

[:SENSe]:DDEMod:SYNC:BURSt:SLENgth

- **Command Format**

  [:SENSe]:DDEMod:SYNC:BURSt:SLENgth <integer>

  [:SENSe]:DDEMod:SYNC:BURSt:SLENgth?

- **Functional Description**

  Set the burst search length.

  <integer>: An integer value.

- **Return Format**

  The query returns the burst search length.

- **For Example**

  :DDEMod:SYNC:BURSt:SLENgth 64 Set the burst search length to 64.

  :DDEMod:SYNC:BURSt:SLENgth? The query returns 64.

[:SENSe]:DDEMod:SYNC:BURSt:STATe

- **Command Format**

[:SENSe]:DDEMod:SYNC:BURSt:STATe {{1|ON} | {0|OFF}}

[:SENSe]:DDEMod:SYNC:BURSt:STATe?

- **Functional Description**

Switch the burst search to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the burst search status: 0 or 1.

- **For Example**

:DDEMod:SYNC:BURSt:STATe ON Turn on the burst search.

:DDEMod:SYNC:BURSt:STATe? The query returns 1.

[:SENSe]:DDEMod:SYNC:SLENgth

- **Command Format**

[:SENSe]:DDEMod:SYNC:SLENgth <integer>

[:SENSe]:DDEMod:SYNC:SLENgth?

- **Functional Description**

Set the burst search length and synchronous length, as well as the burst search length and synchronous analysis length.

<integer>: An integer value.

- **Return Format**

The query returns the burst search length and synchronous length.

- **For Example**

:DDEMod:SYNC:SWORd:OFFSet 64

Set the burst search length and synchronous length to 64.

:DDEMod:SYNC:SWORd:OFFSet? The query returns 64.

[:SENSe]:DDEMod:SYNC:SWORd:OFFSet

- **Command Format**

[:SENSe]:DDEMod:SYNC:SWORd:OFFSet <integer>

[:SENSe]:DDEMod:SYNC:SWORd:OFFSet?

- **Functional Description**

Set the synchronous offset.

<integer>: An integer value.

- **Return Format**

The query returns the synchronous offset.

- **For Example**

:DDEMod:SYNC:SWORd:OFFSet 64	Set the synchronous offset to 64.
:DDEMod:SYNC:SWORd:OFFSet?	The query returns 64.

[:SENSe]:DDEMod:SYNC:SWORd:SLENgth

- **Command Format**

[:SENSe]:DDEMod:SYNC:SWORd:SLENgth <integer>

[:SENSe]:DDEMod:SYNC:SWORd:SLENgth?

- **Functional Description**

Set the synchronous analysis length.

<integer>: An integer value.

- **Return Format**

The query returns the synchronous analysis length.

- **For Example**

:DDEMod:SYNC:SWORd:SLENgth 64	Set the synchronous analysis length to 64.
:DDEMod:SYNC:SWORd:SLENgth?	The query returns 64.

[:SENSe]:DDEMod:SYNC:SWORd:STATe

- **Command Format**

[:SENSe]:DDEMod:SYNC:SWORd:STATe {{1|ON} | {0|OFF}}

[:SENSe]:DDEMod:SYNC:SWORd:STATe?

- **Functional Description**

Switch the synchronous search to on or off.

1|ON: ON

0|OFF: OFF

- **Return Format**

The query returns the synchronous search status: 0 or 1.

- **For Example**

:DDEMod:SYNC:SWORd:STATe ON	Turn on the synchronous search status.
:DDEMod:SYNC:SWORd:STATe?	The query returns 1.

[:SENSe]:FREQuency:CENTer

- **Command Format**

[:SENSe]:FREQuency:CENTer <freq>

[:SENSe]:FREQuency:CENTer?

- **Functional Description**

  Set the center frequency for the sweep frequency.

  <freq>: A continuous real number with the default unit in Hz.

  The frequency range is from 50 Hz to the maximum frequency -50 Hz. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the center frequency in scientific notation, with the unit in Hz.

- **For Example**

  :FREQuency:CENTer 1GHz Set the center frequency for the sweep frequency to 1 GHz.

  :FREQuency:CENTer? The query returns 1.000000e+09.

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

- **Command Format**

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

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

- **Functional Description**

  Automatically/manually switch the center frequency step.

  1|ON: Automatic (AUTO)

  0|OFF: Manual

- **Return Format**

  The query returns the status of the center frequency step: 0 or 1.

- **For Example**

  :FREQuency:CENTer:STEP:AUTO ON

  Set the center frequency step to automatic (AUTO) mode.

  :FREQuency:CENTer:STEP:AUTO? The query returns 1.

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

- **Command Format**

  [:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>

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

- **Functional Description**

  Adjust the center frequency step

  <freq>: A continuous real number with the default unit in Hz.

- **Return Format**

  The query returns the center frequency step in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:CENTer:STEP 10MHz	Set the center frequency step to 10 MHz.
:FREQuency:CENTer:STEP?	The query returns 1.000000e+07.

[:SENSe]:FREQuency:SPAN

- **Command Format**

  [:SENSe]:FREQuency:SPAN <freq>

  [:SENSe]:FREQuency:SPAN?

- **Functional Description**

  Set the span.

  <freq>: A continuous real number with the default unit in Hz.

  The span range is from 100 Hz to the maximum frequency. The maximum frequency varies for different models of spectrum analyzers. Refer to Appendix 3 Parameter Table for Each Model.

- **Return Format**

  The query returns the span in scientific notation, with the unit in Hz.

- **For Example**

:FREQuency:SPAN 1GHz	Set the span to 1 GHz.
:FREQuency:SPAN?	The query returns 1.000000e+09.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:ATTenuation <ampl>

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

- **Functional Description**

  Adjust the input attenuation.

  <ampl>: A continuous real number. The default unit is dB, with an amplitude range of 0 dB to 51 dB.

- **Return Format**

  The query returns the input attenuation in dB.

- **For Example**

:POWer:ATTenuation 10	Set the input attenuation to 10 dB.
:POWer:ATTenuation?	The query returns 10.

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

- **Command Format**

  [:SENSe]:POWer[:RF]:GAIN:STATe {{1|ON} | {0|OFF}}

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

- **Functional Description**

  Set the preamplifier switch to on or off.

- **Return Format**

  The query returns the switch status of the preamplifier: 0 or 1.

- **For Example**

:POWer:GAIN:STATe ON	Turn on the preamplifier switch.
:POWer:GAIN:STATe?	The query returns 1.

[:SENSe]:ROSCillator:SOURce:TYPE

- **Command Format**

  [:SENSe]:ROSCillator:SOURce:TYPE {INTernal|EXTernal}

  [:SENSe]:ROSCillator:SOURce:TYPE?

- **Functional Description**

  Set the frequency reference.

  INTernal: Internal

  EXTernal: External

- **Return Format**

  The query returns the frequency reference.

- **For Example**

:ROSCillator:SOURce:TYPE INTernal	Set the frequency reference to INTernal.
:ROSCillator:SOURce:TYPE?	The query returns INTernal.

Programming Explanation

This chapter describes troubleshooting during the programming process. If you encounter any of the following problems, please follow the related instructions.

Programming Preparation

Users can remotely control the spectrum analyzer via USB or LAN interfaces and integrate it with NI-VISA and programming languages. Programming is applicable only when using Visual Studio and LabVIEW development tools under the Windows operating system.

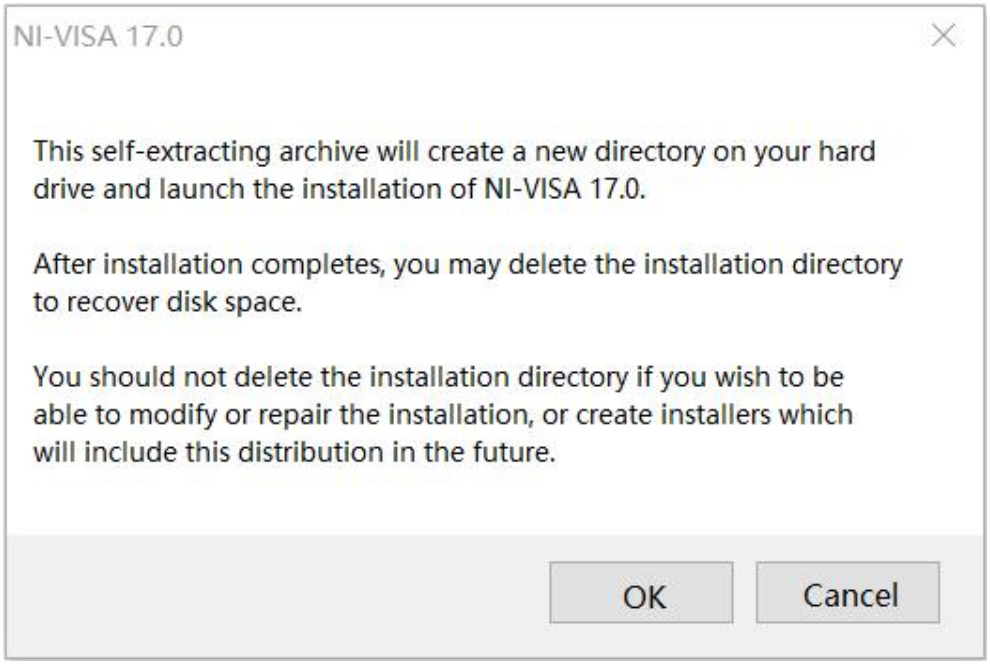
1. Setup Communication

NI-VISA is the communication library for communication between computers and devices. NI software has two valid VISA installation packages: Full version and the Run-Time Engine version. The full version includes the NI device driver and the NI MAX tool, NI MAX is the used to control user interface of the device. While the driver and NI MAX are useful, they are not used for remote control. Run-Time Engine is a smaller file than the full version, and it is primarily used for remote control. You can download the latest NI-VISA Run-Time Engine or the full version from the NI website. The installation steps are the same.

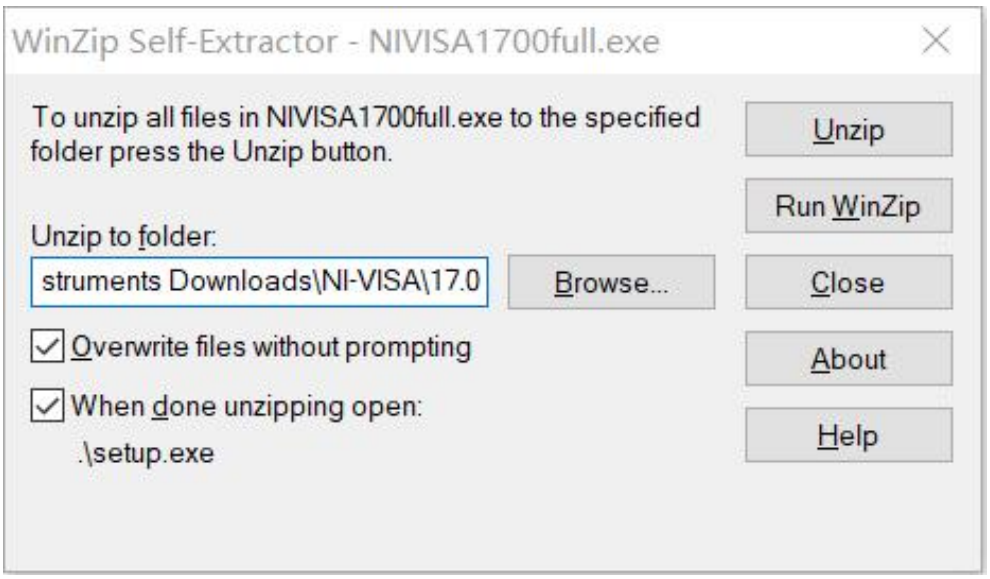
Follow these steps to install NI-VISA (take NI-VISA 17.0 full version as an example).

a. Download the proper version of NI-VISA.

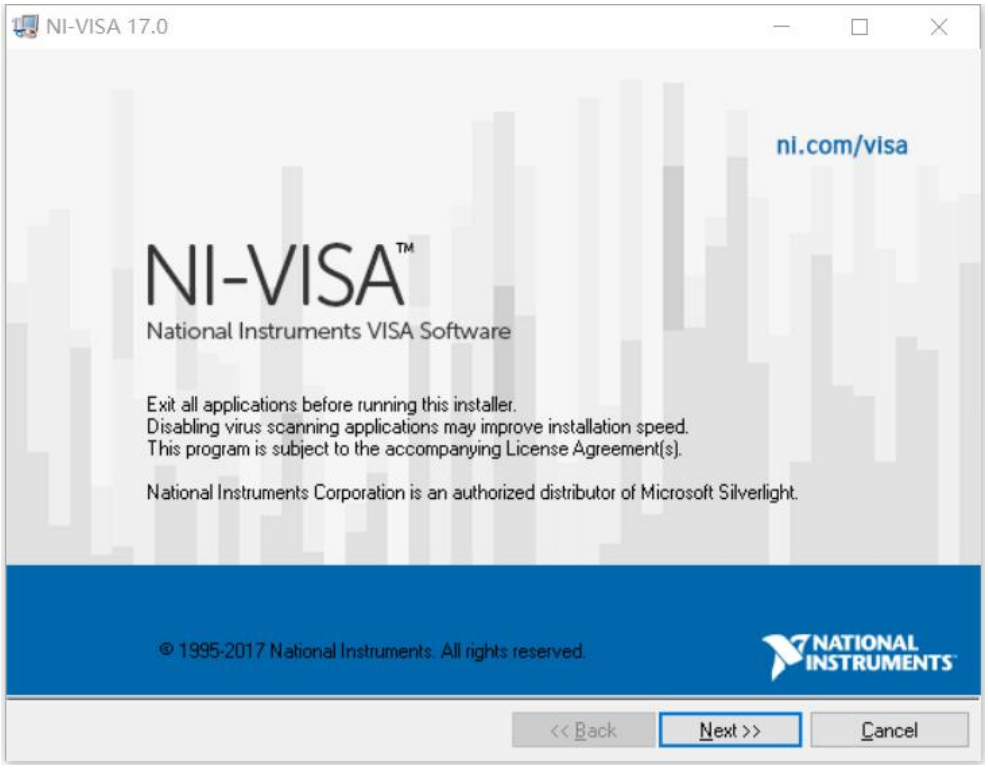
b. Double click NIVISA1700full.exe to pop out the dialogue.



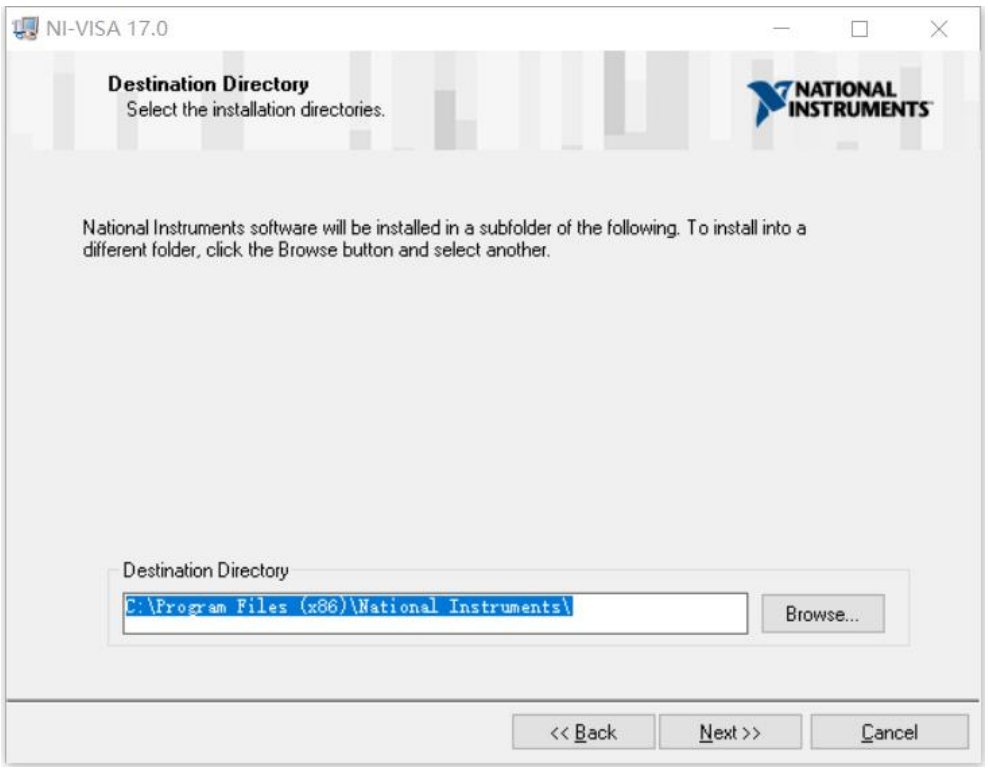
c. Click to confirm to pop-out the dialogue.



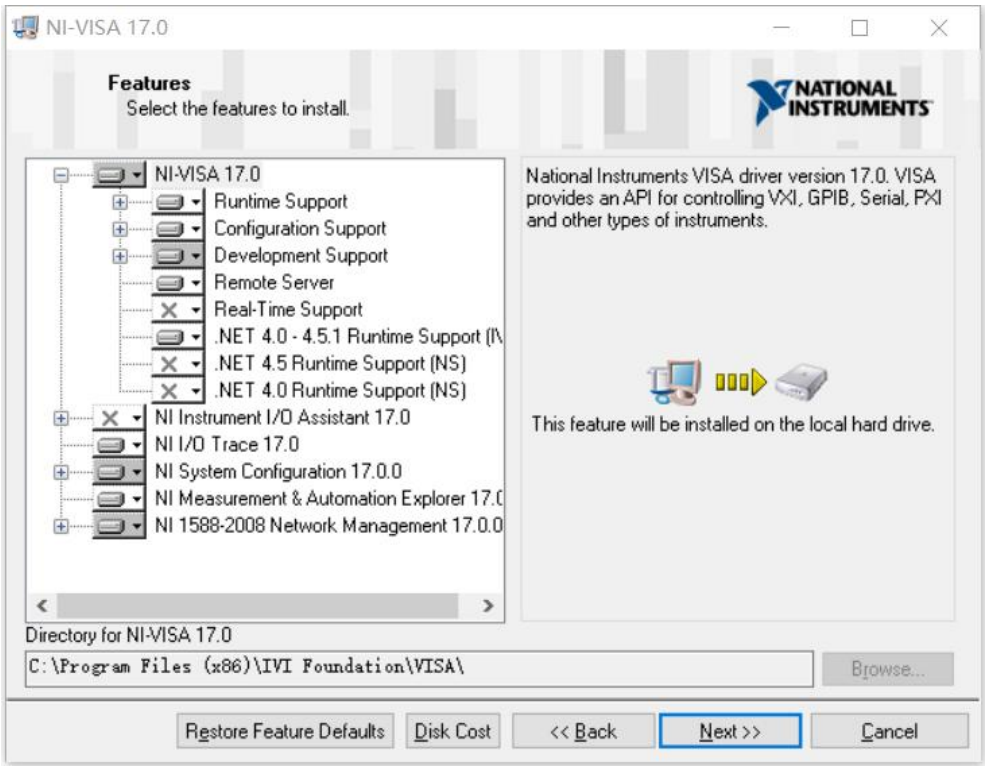
d. Click "Unzip" to decompress the files. After decompression is complete, the installation program will run automatically. If your computer requires .NET Framework 4, it will be installed automatically during the installation process.



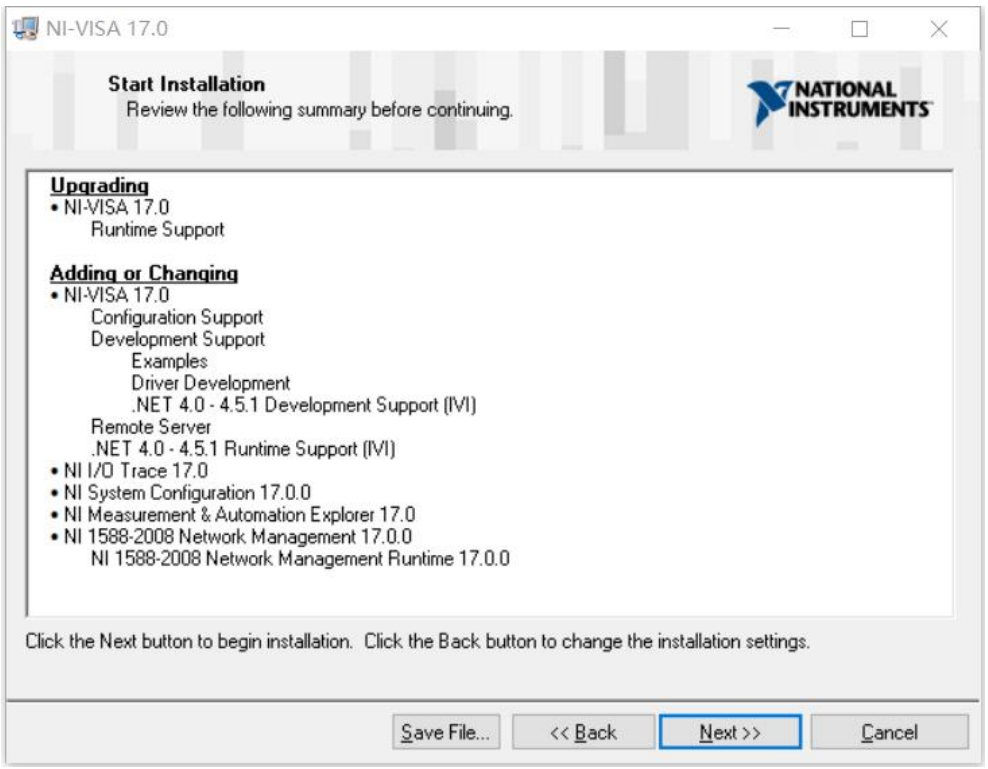
e. Installation dialogue of NI-VISA as shown in the figure above. Click Next to start the installation.



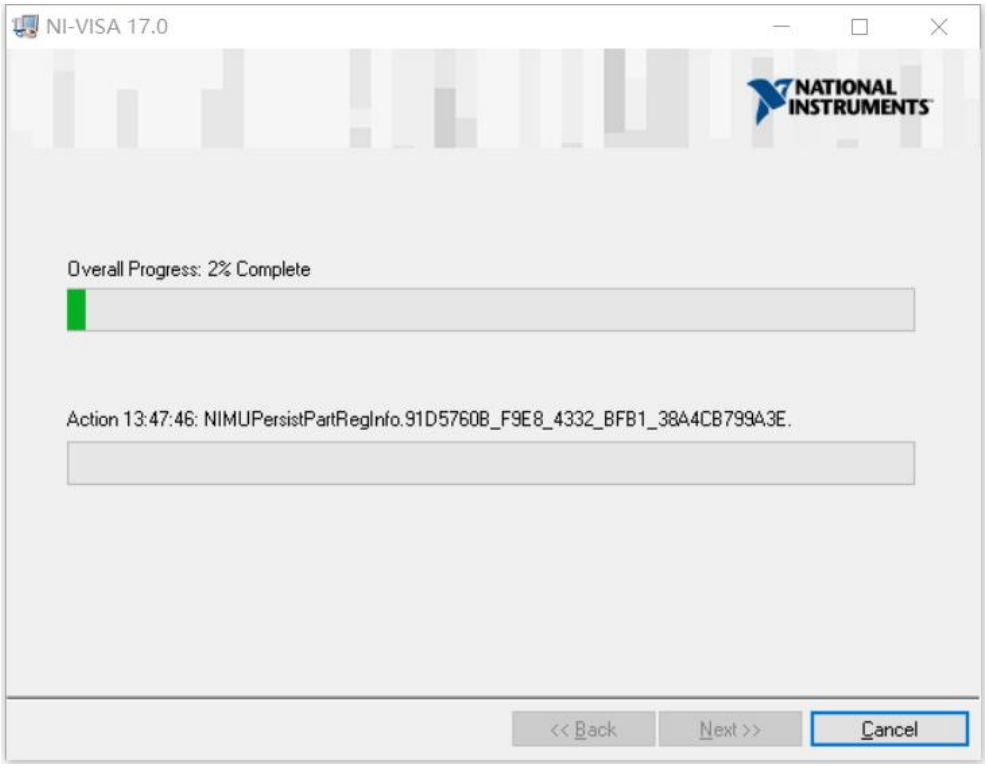
f. Set the installation path, the default path is "C:\Program Files (x86)\National Instruments\", or you can click Next to change the installation path, as shown in the following figure.



g. Double click Next in the license agreement dialogue and select "I accept the above 2 License Agreement(s)." and then click Next, the dialogue as shown in the following figure.



h. Click Next to start the installation.



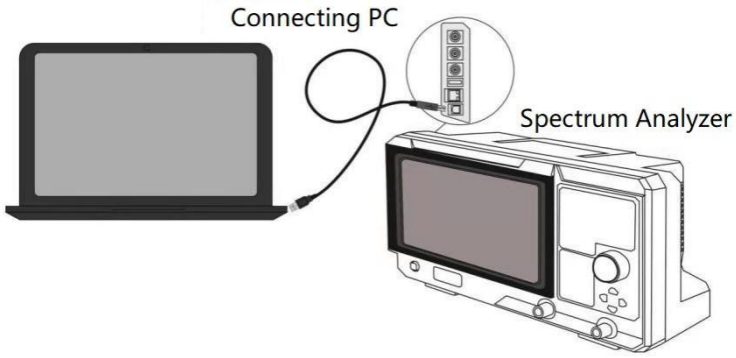
j. Restart the computer after the installation is completed.

2. Connecting Device

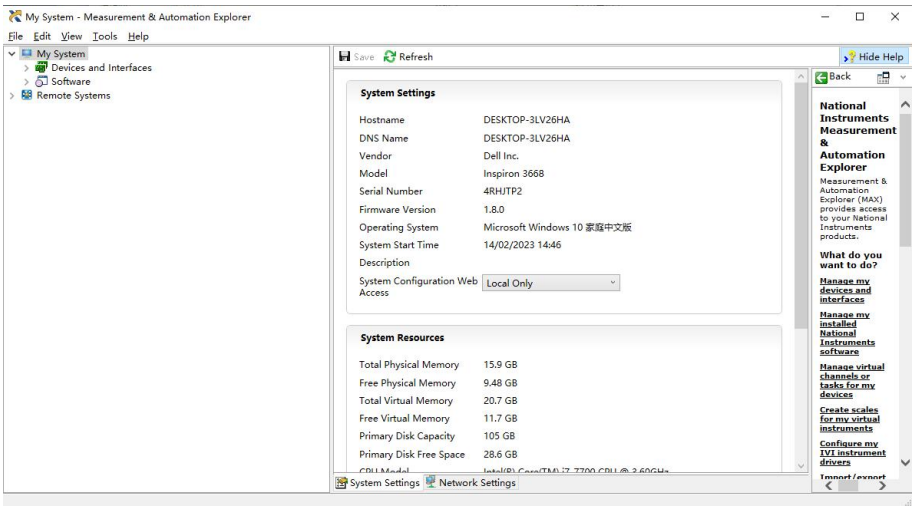
Use the USB method as an example to introduce the connection process.

a. Turn on the spectrum analyzer.

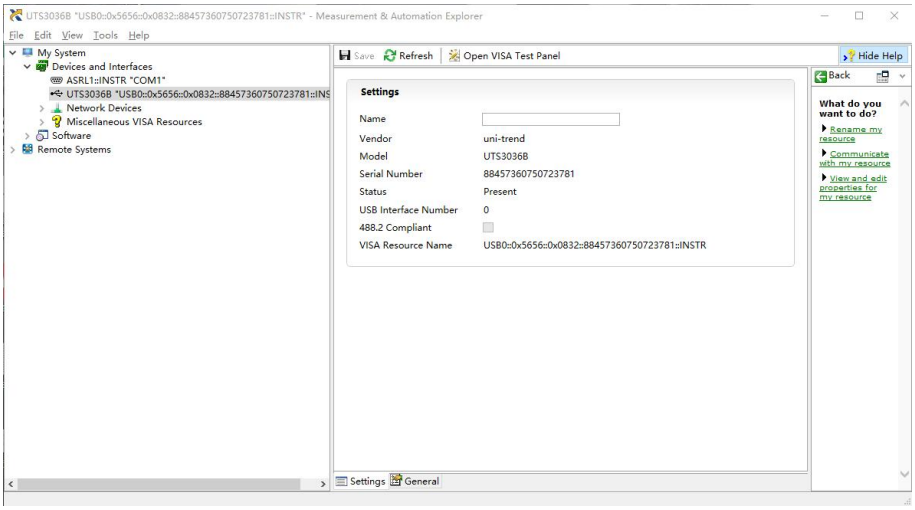
b. Use USB wire to connect the USB Device port of the spectrum analyzer with USB Host port of the computer, as shown in the following figure.



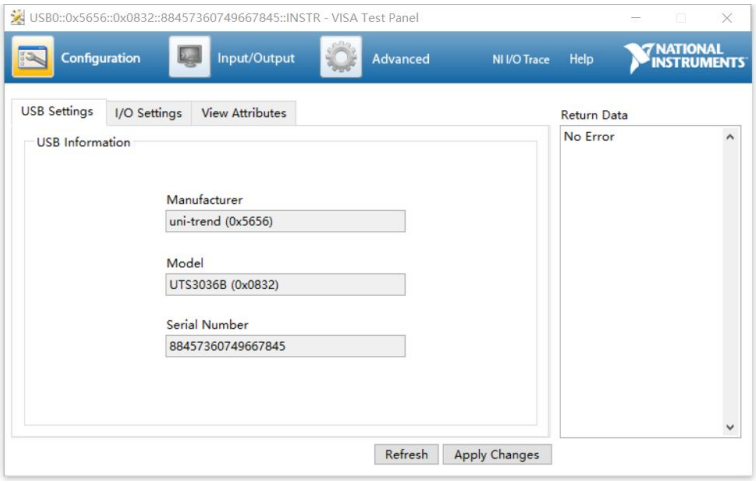
c. Turn on NI MAX on the computer. The dialogue as shown in the following figure.



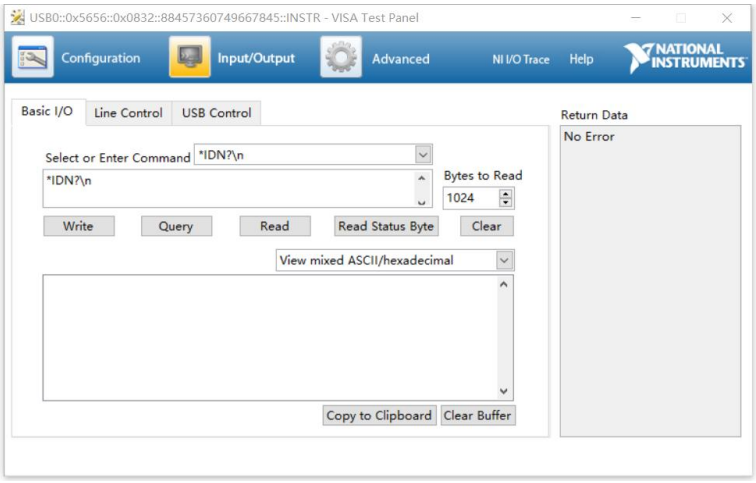
d. Turn on the device and scroll down the options Select the spectrum analyzer's drive, as shown in the following figure.



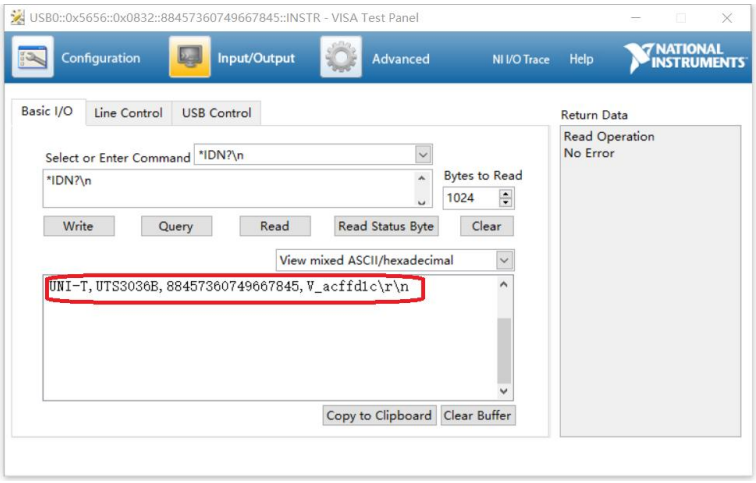
e. Use mouse to click VISA test panel to pop-out the dialogue as shown in the following figure.



f. Use mouse to click the option Input/Output, as shown in the following figure.



g. Use mouse to click Query IDN of the spectrum analyzer, the query result will display at the red area, as shown in the following figure.



h. If it can query the relevant information of the spectrum analyzer, which means of the spectrum analyzer is communication with the computer.

VISA Programming Example

This section contains examples. Through these examples, users can learn how to use VISA and combine it with the commands in the programming manual to control the instrument. With these examples, users can develop more applications.

VC++Example

- Environment: Window System, Visual Studio
- Description: Access the instrument via USBTMC and TCP/IP and send "*IDN?" command on NI-VISA Query the device information.
- **Steps**

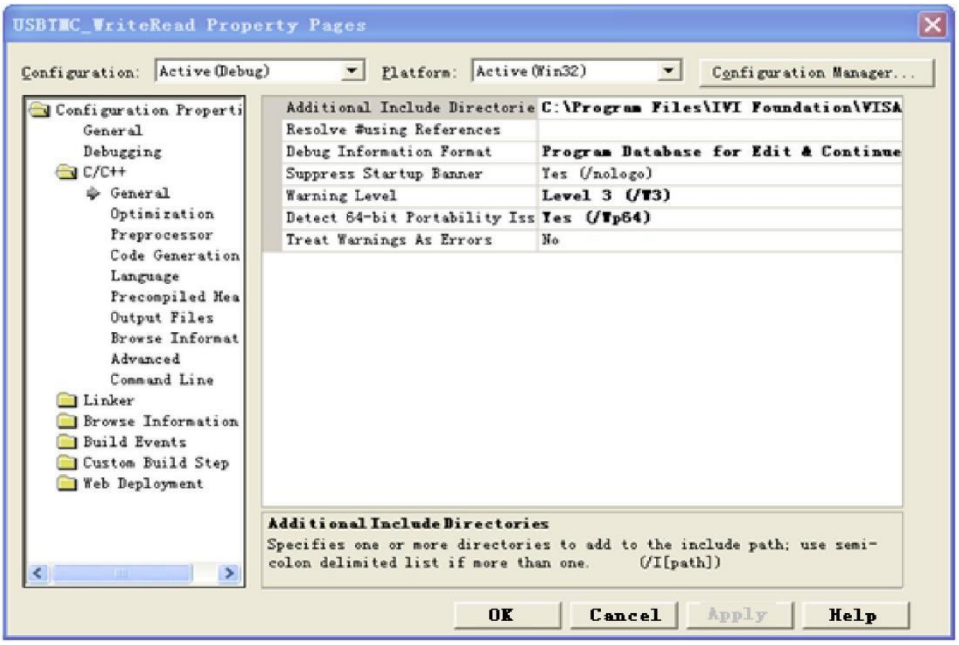
1. Open the Visual Studio software to create a new VC++ win32 console project.
2. Set the project environment that can adjust NI-VISA library, which includes both static and dynamic libraries.
3. Static library

   Locate the files “visa.h, visatype.h, and visa32.lib” in the NI-VISA installation path. Copy them to the root directory of the VC++ project and add them to the project. Add the following two lines of code to the projectname.cpp file:

   ```
   #include "visa.h"
   #pragma comment(lib,"visa32.lib")
   ```

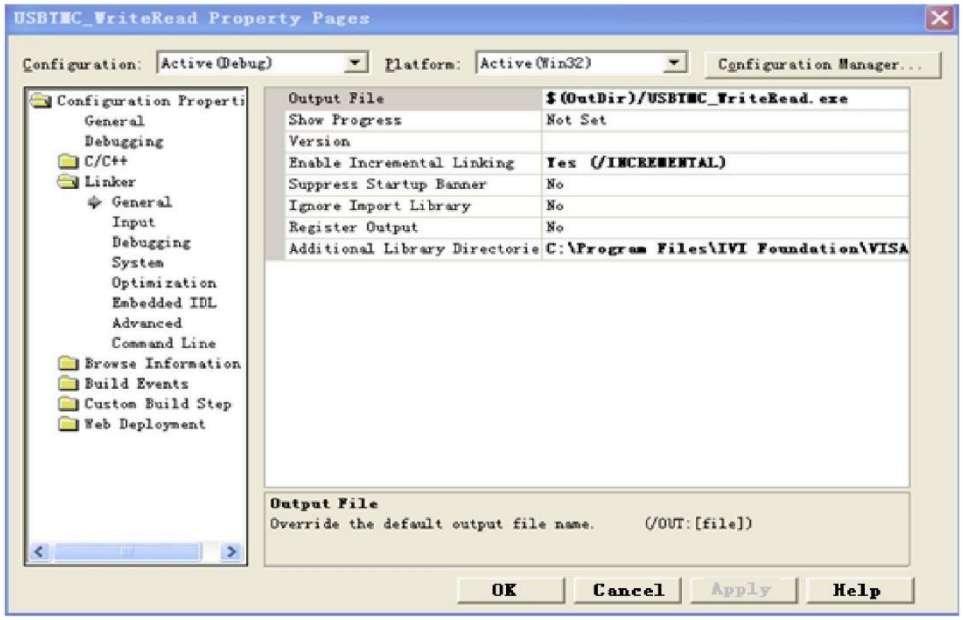
   a) Dynamic library

   Press “project>>properties”, select "c/c++---General" in the attribute dialog on the leftside, and set the value of "Additional Include Directories" to the installment path of NI-VIS (for example, C:\ProgramFiles\IVI Foundation\VISA\WinNT\include), as shown in the following figure.

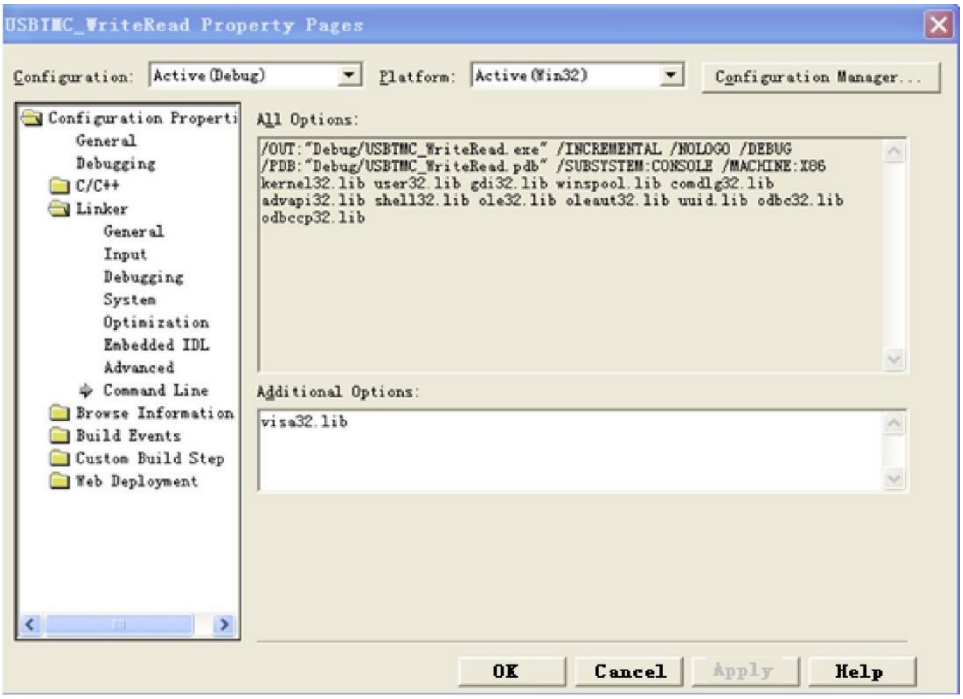


Select "Linker-General" in the attribute dialog on the left side, and set the value of "Additional Library Directories" as the installment path of NI-VIS

(for example, C:\Program Files\IVI Foundation\VISA\WinNT\include), as shown in the following figure.



Select "Linker-Command Line" in the attribute dialog on the left side, and set the value of "Additional" as visa32.lib, as shown in the following figure.



Add the file visa.h in the projectname.cpp file.

#include <visa.h>

1. Source code

a) USBTMC Example

```
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.
     * 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*/
     ViSession defaultRM;
     ViSession instr;
     ViUInt32 numInstrs;
     ViFindList findList;
     ViStatus status;
     char instrResourceString[VI_FIND_BUFLEN];
     unsigned char buffer[100];
     int i;
```

```
 status = viOpenDefaultRM(&defaultRM);
 if (status < VI_SUCCESS)
 {
       printf("Could not open a session to the VISA Resource Manager!\n");
       return status;
 }
/*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);
       return status;
 }
 /** 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;
}
```

b) TCP/IP Example

```
int tcp_ip_test(char *pIP)
{
    char outputBuffer[VI_FIND_BUFLEN];
    ViSession defaultRM, instr;
    ViStatus status;
    /* 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[] = "::inst0::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;
```

C# Example

- Environment: Window System, Visual Studio
- Description: Access the instrument via USBTMC and TCP/IP, and send "*IDN?" command on NI-VISA Query the device information.
- Steps
- Open the Visual Studio software and create a new C# console project.
- Add C# quote Ivi.Visa.dll and NationalInstruments.Visa.dll of VISA.
- Source code
- USBTMC Example

```
class Program
{
    void usbtmc_test()
    {
        using (var rmSession = new ResourceManager())
        {
            var resources = rmSession.Find("USB?*INSTR");
            foreach (string s in resources)
            {
                try
                {
                    var mbSession = (MessageBasedSession)rmSession.Open(s);
                    mbSession.RawIO.Write("*IDN?\n");
                    System.Console.WriteLine(mbSession.RawIO.ReadString());
                }
                catch (Exception ex)
```

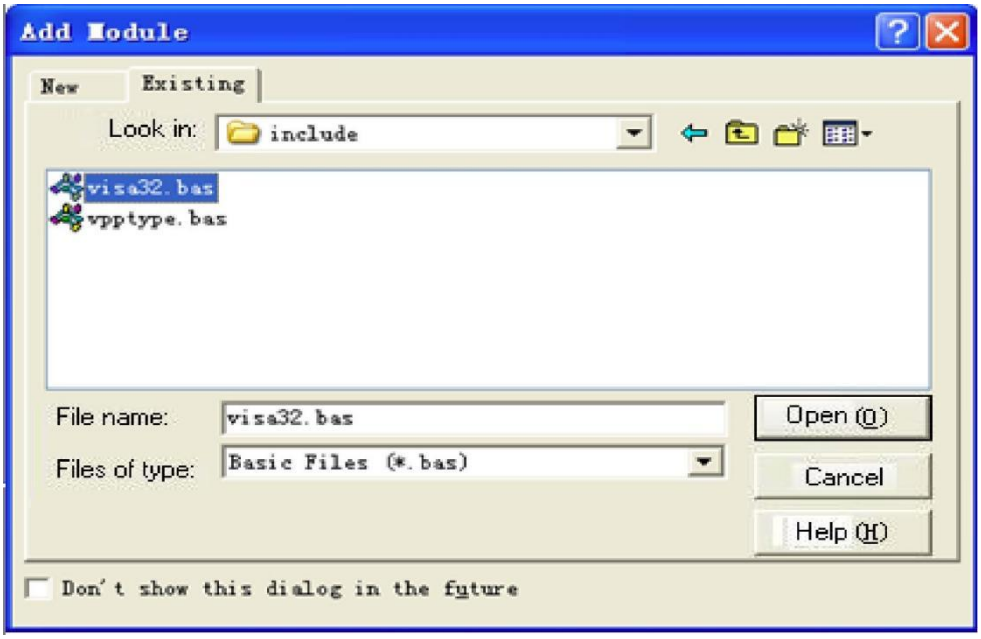
```
                {
                    System.Console.WriteLine(ex.Message);
                }
            }
        }

        void Main(string[] args)
        {
            usbtmc_test();
        }
}
```

a) TCP/IP Example

```
class Program
  {
        void tcp_ip_test(string ip)
        {
            using (var rmSession = new ResourceManager())
            {
                try
                {
                    var resource = string.Format("TCPIP0::{0}::inst0::INSTR", ip);
                    var mbSession = (MessageBasedSession)rmSession.Open(resource);
                    mbSession.RawIO.Write("*IDN?\n");
                    System.Console.WriteLine(mbSession.RawIO.ReadString());
                }
                catch (Exception ex)
                {
                    System.Console.WriteLine(ex.Message);
                }
            }
        }

        void Main(string[] args)
        {
            tcp_ip_test("192.168.20.11");
        }
```

VB Example

- Environment: Window System, Microsoft Visual Basic 6.0.
- Description: Access the instrument via USBTMC and TCP/IP, and send "*IDN?" command on NI-VISA to query the device information.
- Steps
- Open the Visual Basic software and create a new standard application program project.
- Set the project environment that can adjust NI-VISA library, press “Existing tab of Project>>Add Existing Item”, find the “visa32.bas file” in the "include" folder under the NI-VISA installation path, and add it, as shown in the following figure.



1. Source code
2. USBTMC Example

```
PrivateFunction usbtmc_test() AsLong
' 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 AsLong
Dim instrsesn AsLong
Dim numInstrs AsLong
Dim findList AsLong
Dim retCount AsLong
Dim status AsLong
Dim instrResourceString AsString *VI_FIND_BUFLEN
Dim Buffer AsString * MAX_CNT
Dim i AsInteger

' 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
ExitFunction
EndIf

' 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) Then
    resultTxt.Text = "An error occurred while finding resources."
    viClose(defaultRM)
    usbtmc_test = status
ExitFunction
EndIf

' 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.
```

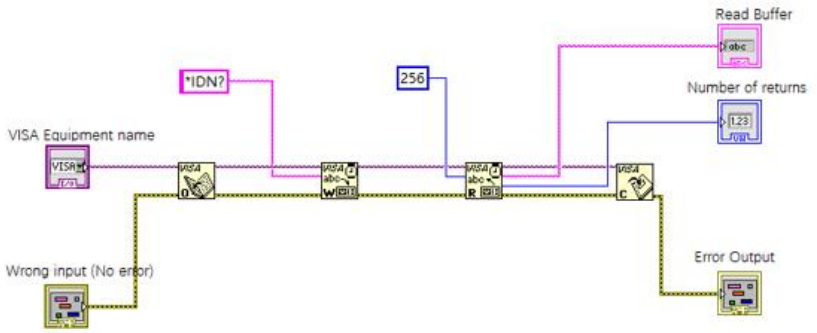
```
For i = 0 To numInstrs
If (i > 0) Then
    status = viFindNext(findList, instrResourceString)
EndIf
    status = viOpen(defaultRM, instrResourceString, VI_NULL, VI_NULL, instrsesn)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "Cannot open a session to the device " + CStr(i + 1)
GoTo NextFind
EndIf

' 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
EndIf

' 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
EndIf
    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
EndFunction
```

a) TCP/IP Example

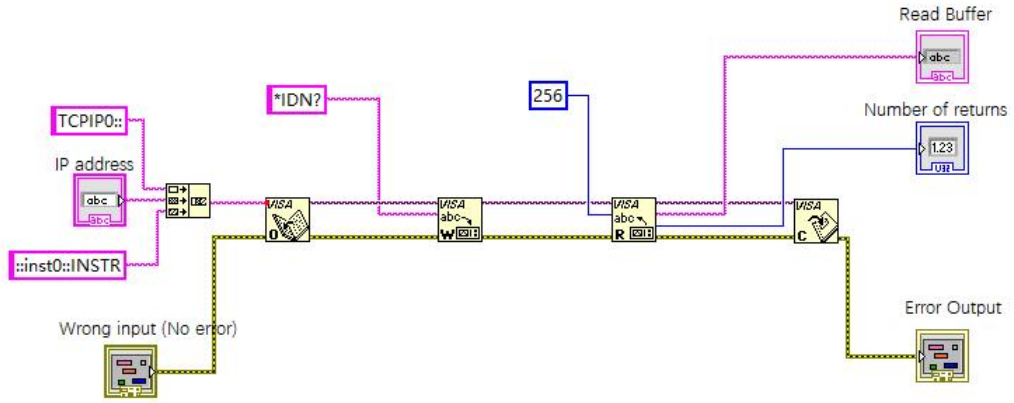
```
PrivateFunction tcp_ip_test(ByVal ip AsString) AsLong
Dim outputBuffer AsString * VI_FIND_BUFLEN
Dim defaultRM AsLong
Dim instrsesn AsLong
Dim status AsLong
Dim count AsLong

' 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
ExitFunction
EndIf

' Now we will open a session via TCP/IP device
status = viOpen(defaultRM, "TCPIP0::" + ip + "::inst0::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
ExitFunction
EndIf
status = viWrite(instrsesn, "*IDN?", 5, count)
If (status < VI_SUCCESS) Then
    resultTxt.Text = "Error writing to the device."
EndIf
    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
EndIf
    status = viClose(instrsesn)
    status = viClose(defaultRM)
    tcp_ip_test = 0
EndFunction
```

LabVIEW Example

- Environment: Window System, LabVIEW
- Description: Access the instrument via USBTMC and TCP/IP, and send "*IDN?" command on NI-VISA to query the device information.
- Steps
- Open the LabVIEW software and create a VI file.
- Add the control, click the front panel to select and add the VISA source name, error input, error output, and part of the indicator from the control column.
- Open the diagram, click the VISA source name, and then select and add these functions VISA Write, VISA Read, VISA Open, and VISA Close on the VISA menu.
- The VI opens a VISA session for a USBTMC device, writes the *IDN? command to the device, and reads back the response value. When all communication is complete, the VI closes the VISA session, as shown in the following figure.



1. Communication with the device via TCP/IP is similar to USBTMC. You need to set the VISA Write and Read functions to synchronous I/O, as LabVIEW uses asynchronous I/O by default. Right-click on the node and select “Synchronous I/O Mode >> Synchronous” from the shortcut menu to enable synchronous writing or reading of data, as shown in the following figure.



MATLAB Example

- Environment: Window System, MATLAB
- Description: Access the instrument via USBTMC and TCP/IP, and send "*IDN?" command on NI-VISA to query the device information.
- Steps
- Open the MATLAB software, click File>>New>>Script on Matlab interface to create an empty M file.
- Source code
- USBTMC Example

```
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::0x5345::0x1234::SN20220718::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
```

a) TCP/IP Example

```
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::','192.168.20.11','::inst0::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
```

Python Example

- Environment: Window System, Python3.8, PyVISA 1.11.0.
- Functional Description: Access the instrument via USBTMC and TCP/IP, and send "*IDN?" command on NI-VISA to query the device information.
- Steps
- Install python first, then open a Python script editor to create an empte test.py file.
- Use the command “pip install PyVISA” to install PyVISA. If the installation fails, refer to this link (https://pyvisa.readthedocs.io/en/latest/).
- Source code
- USBTMC Example

```
import pyvisa
    rm = pyvisa.ResourceManager()
    rm.list_resources()
    my_instrument = rm.open_resource('USB0::0x5345::0x1234::SN20220718::INSTR')
    print(my_instrument.query('*IDN?'))
```

a) TCP/TP Example

```
    import pyvisa
    rm = pyvisa.ResourceManager()
    rm.list_resources()
    my_instrument = rm.open_resource('TCPIP0::192.168.20.11::inst0::INSTR')
    print(my_instrument.query('*IDN?'))
```

Programming Application Example

This section will introduce how to make accurate measurements of stable signals in spectrum analysis mode by using SCPI commands.

1. Signal source

   The RF input port of the spectrum analyzer inputs a continuous sine wave signal with a signal frequency of 100 MHz and a signal power of -20 dBm.

2. Configure the spectrum analyzer:

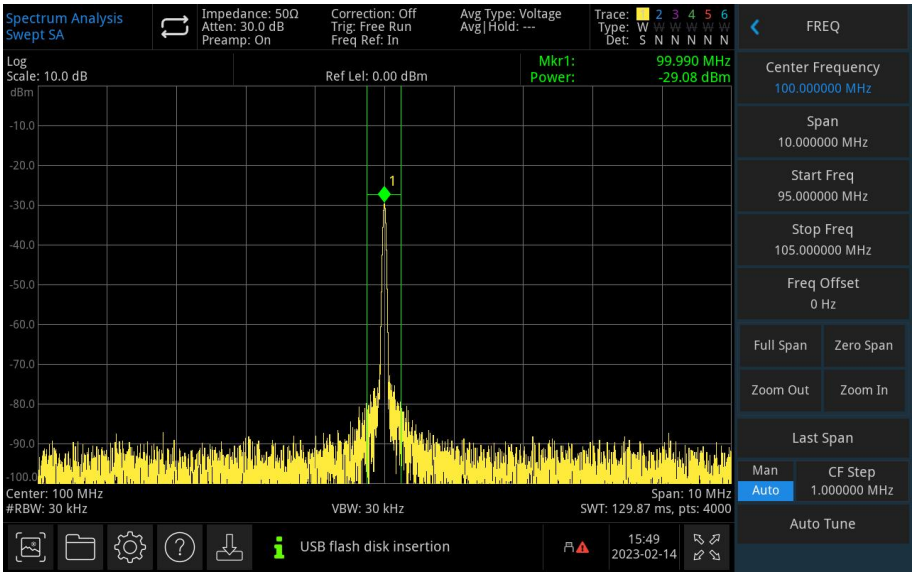
   Select the operating mode to spectrum analysis.

   Select Default to restore the parameter to the default setting.

   Adjust the center frequency, sweep bandwidth, resolution bandwidth, input attenuation, pre-amplifier, and display scale according to the input signal, ensuring the signal is visible and centered on the screen.

3. Adjust the following settings based on actual requirements to accurately display the signal: sweep time, sweep mode, sweep count, trace type, and trace detector type.

4. Finally, perform a peak value search and marker measurement on the signal.



The following commands can perform the operations as belonging to the above to obtain the accurate measurements, as shown in the figure above.

```
:instrument:select SA       // Select the operating mode to spectrum analysis
:key:default       // Restore to the default setting
:sense:frequency:center 100000000       // Set the center frequency to 100 MHz
:sense:frequency:span 10000000       // Set the sweep bandwidth to 10 MHz
:sense:bwidth:resolution 30000       // Set the resolution bandwidth to 30 kHz
```

:sense:bwidth:video 30000 // Set the video bandwidth to 30 kHz

:display:window:trace:y:scale:rlevel 0 // Set the reference level to 0

:sense:power:rf:attenuation:auto 1 // Automatically set input attenuation value

:sense:power:rf:gain:state 1 // Turn on the pre-amplifier

:display:window:trace:Y:scale:spacing LOG // Displays scale logarithm

:sense:sweep:time:auto 1 // Automatic sweep time

:sense:sweep:type:auto 1 // Automatic sweep mode

:sense:sweep:points 4000 // Set sweep count to 4000

:trace1:mode write // Refresh trace 1

:detector:trace1 sample // Trace 1 detector sampling

:calculate:marker1:maximum // Peak search

:calculate:marker1:function bpower // Turn on marker in-band power

:calculate:marker1:function:band:span 500000 // Set in-band width to 500 kHz. View the measurement results at the cursor measurement results area in the upper right corner of the spectrum graph, including the cursor frequency and in-band power.

Appendix 1 <key> Table

Key	Functional Description	LED
SETup	Measurement setting	
Meas	Measurement	
Mode	Module	
FREQ	Frequency	
AMPT	Amplitude	
BW	Bandwidth	
Sweep	Sweep	
Trace	Trace	
Marker	Marker	
Peak	Peak	
Auto	Automatic	
Save	Save	
System	System setting	
Default	Restore to the factory setting	
TG	Trace source	√
Single	Single	√
TLOCk	Screen-lock	√
UP	Direction key Up	
DOWN	Direction key Down	
LEFT	Direction key Left	
RIGHT	Direction key Right	
FKNLeft	Multipurpose left knob	
FKNRight	Multipurpose right knob	
NUM0	Numeric key 0	
NUM1	Numeric key 1	
NUM2	Numeric key 2	
NUM3	Numeric key 3	
NUM4	Numeric key 4	
NUM5	Numeric key 5	
NUM6	Numeric key 6	
NUM7	Numeric key 7	
NUM8	Numeric key 8	

NUM9	Numeric key 9	
DOT	Numeric key decimal point	
SYMBOL	Symbol of numeric key	
ESC	Exit	
BACKspace	Backspace	
Enter	Enter	

Appendix 2 Lock Status of Key

Bit Order	Key	Status
0	Setup	0 represents that the key is unlocked. 1 represents that the key is locked.
1	Meas	0 represents that the key is unlocked. 1 represents that the key is locked.
2	Mode	0 represents that the key is unlocked. 1 represents that the key is locked.
3	FREQ	0 represents that the key is unlocked. 1 represents that the key is locked.
4	AMPT	0 represents that the key is unlocked. 1 represents that the key is locked.
5	BW	0 represents that the key is unlocked. 1 represents that the key is locked.
6	Sweep	0 represents that the key is unlocked. 1 represents that the key is locked.
7	Trace	0 represents that the key is unlocked. 1 represents that the key is locked.
8	Marker	0 represents that the key is unlocked. 1 represents that the key is locked.
9	Peak	0 represents that the key is unlocked. 1 represents that the key is locked.
10	Auto	0 represents that the key is unlocked. 1 represents that the key is locked.
11	Save	0 represents that the key is unlocked. 1 represents that the key is locked.
12	System	0 represents that the key is unlocked. 1 represents that the key is locked.
13	Default	0 represents that the key is unlocked.

		1 represents that the key is locked.
14	TG	0 represents that the key is unlocked. 1 represents that the key is locked.
15	Single	0 represents that the key is unlocked. 1 represents that the key is locked.
16	TLOCk	0 represents that the key is unlocked. 1 represents that the key is locked.
17	UP	0 represents that the key is unlocked. 1 represents that the key is locked.
18	DOWN	0 represents that the key is unlocked. 1 represents that the key is locked.
19	LEFT	0 represents that the key is unlocked. 1 represents that the key is locked.
20	RIGHT	0 represents that the key is unlocked. 1 represents that the key is locked.
21	FKNLeft	0 represents that the key is unlocked. 1 represents that the key is locked.
22	FKNRight	0 represents that the key is unlocked. 1 represents that the key is locked.
23	NUM0	0 represents that the key is unlocked. 1 represents that the key is locked.
24	NUM1	0 represents that the key is unlocked. 1 represents that the key is locked.
25	NUM2	0 represents that the key is unlocked. 1 represents that the key is locked.
26	NUM3	0 represents that the key is unlocked. 1 represents that the key is locked.
27	NUM4	0 represents that the key is unlocked. 1 represents that the key is locked.
28	NUM5	0 represents that the key is unlocked. 1 represents that the key is locked.
29	NUM6	0 represents that the key is unlocked. 1 represents that the key is locked.
30	NUM7	0 represents that the key is unlocked. 1 represents that the key is locked.
31	NUM8	0 represents that the key is unlocked. 1 represents that the key is locked.
32	NUM9	0 represents that the key is unlocked. 1 represents that the key is locked.

33	DOT	0 represents that the key is unlocked. 1 represents that the key is locked.
34	SYMBOL	0 represents that the key is unlocked. 1 represents that the key is locked.
35	ESC	0 represents that the key is unlocked. 1 represents that the key is locked.
36	BACKspace	0 represents that the key is unlocked. 1 represents that the key is locked.
37	Enter	0 represents that the key is unlocked. 1 represents that the key is locked.

Appendix 3 Parameter Table for Each Model

Model	Maximum Frequency	Maximum RBW	Maximum Sweep Point	Limit Values	Trace Values
UTS1015B	1.5 GHz	1 MHz	10001	4	4
UTS1032B	3.2 GHz	1 MHz	10001	4	4
UTS1015T	1.5 GHz	1 MHz	10001	4	4
UTS1032T	3.2 GHz	1 MHz	10001	4	4
UTS3021B	2.1 GHz	3 MHz	40001	6	6
UTS3036B	3.6 GHz	3 MHz	40001	6	6
UTS3084B	8.4 GHz	3 MHz	40001	6	6
UTS3084T	8.4 GHz	3 MHz	40001	6	6
UTS3015T+	1.5 GHz	1 MHz	10001	4	4
UTS3032T+	3.2 GHz	1 MHz	10001	4	4
UTS3036T+	3.6 GHz	3 MHz	40001	6	6
UTS3084T+	8.4 GHz	3 MHz	40001	6	6
UTS5013A	13.6 GHz	8 MHz	100001	6	6
UTS5026A	26.5 GHz	8 MHz	100001	6	6
RSH5013A	13.6 GHz	8 MHz	100001	6	6
RSH5026A	26.5 GHz	8 MHz	100001	6	6