RIGOL

DS70000 Series

Digital Oscilloscope

Programming Guide

2021.Aug

Guaranty and Declaration

Copyright

Trademark Information

RIGOL®is the trademark of RIGOL TECHNOLOGIES CO., LTD.

Notices

- RIGOL products are covered by P.R.C. and foreign patents, issued and pending.
- RIGOL reserves the right to modify or change parts of or all the specifications and pricing policies at the company's sole decision.
- Information in this publication replaces all previously released materials.
- Information in this publication is subject to change without notice.
- RIGOL shall not be liable for either incidental or consequential losses in connection with the furnishing, use, or performance of this manual, as well as any information contained.
- Any part of this document is forbidden to be copied, photocopied, or rearranged without prior written approval of RIGOL.

Product Certification

RIGOL guarantees that this product conforms to the national and industrial standards in China as well as the ISO9001:2015 standard and the ISO14001:2015 standard. Other international standard conformance certifications are in progress.

Contact Us

If you have any problem or requirement when using our products or this manual, please contact RIGOL.

E-mail: service@rigol.com

Website: *http://www.rigol.com*

Section	Description	Page

List of Figures

1 Document Overview

This manual is your guide to programming DS70000 series digital oscilloscope by using SCPI commands through remote interface. DS70000 series can communicate with the PC via the USB, LAN, or GPIB (requiring to work with RIGOL's USB-GPIB interface converter) interface.

Tip

For the latest version of this manual, download it from the official website of RIGOL (*http://www.rigol.com*).

Publication Number

PGB28100-1110

Software Version

00.02.00

Software upgrade might change or add product features. Please acquire the latest version of the manual from RIGOL website or contact RIGOL to upgrade the software.

Format Conventions in this Manual

1 **Key**

The front panel key is denoted by the menu key icon. For example, Delfault indicates the "Default" key.

2 **Menu**

The menu item is denoted by the format of "Menu Name (Bold) + Character Shading" in the manual. For example, **Setup** indicates clicking or tapping the "Setup" sub-menu under the "Utility" function menu to view the basic setting configuration items.

3 **Operation Procedures**

The next step of the operation is denoted by ">" in the manual. For example, > **Storage** indicates that first clicking or tapping the icon , then clicking or tapping **Storage**.

Content Conventions in this Manual

DS70000 series includes the following models. Unless otherwise specified, this manual takes DS70504 as an example to illustrate the functions and operation methods of DS70000 series power supply.

Model	Max. Analog Bandwidth	Analog Channel
DS70504	5 GHz	4
DS70304	3 GHz	4

2 SCPI Command Overview

SCPI (Standard Commands for Programmable Instruments) is a standardized instrument programming language that is built upon the existing standard IEEE 488.1 and IEEE 488.2 and conforms to various standards, such as the floating point operation rule in IEEE 754 standard, ISO 646 7-bit coded character set for information interchange (equivalent to ASCII programming). The SCPI commands provide a hierarchical tree structure, and consist of multiple subsystems. Each command subsystem consists of one root keyword and one or more sub-keywords.

Syntax

The command line usually starts with a colon; the keywords are separated by colons, and following the keywords are the parameter settings available. The command ending with a quotation mark indicates querying a certain function and returns the query results. The keywords of the command and the first parameter are separated by a space.

For example,

`:ACQuire:TYPE` <*type*>

`:ACQuire:TYPE?`

`ACQuire` is the root keyword of the command, **`TYPE`** is the second-level keyword. The command line starts with a colon, and different levels of keywords are also separated by colons. <*type*> indicates a settable parameter. The command ending with a quotation mark indicates querying a function. The command keywords **`:ACQuire:TYPE`** and the parameter <*type*> are separated by a space.

In some commands with parameters, `","`is often used to separate multiple parameters. For example,

`:SYSTem:DATE` <*year*>,<*month*>,<*day*>

Symbol Description

The following symbols are not sent with the commands.

1 **Braces { }**

The contents in the braces can contain one or multiple parameters. These parameters can be omitted or used for several times. Parameters are usually separated by the vertical bar "|". When using the command, you must select one of the parameters.

2 **Vertical Bar |**

The vertical bar is used to separate multiple parameters. When using the command, you must select one of the parameters.

3 **Square Brackets [ ]**

The contents in the square brackets can be omitted.

4 Angle Brackets < >

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

Parameter Type

1 Bool

The parameter can be set to ON, OFF, 1, or 0. For example,

:SYSTem:BEEPer <*bool*>

:SYSTem:BEEPer?

Wherein, <*bool*> can be set to {{1|ON}|{0|OFF}}. The query returns 1 or 0.

2 Discrete

The parameter can be any of the values listed. For example,

:ACQuire:TYPE <*type*>

:ACQuire:TYPE?

Wherein,

- <*type*> can be set to NORMal|PEAK|AVERages|HRESolution.
- The query returns NORM, AVER, PEAK, or HRES.

3 Integer

Unless otherwise specified, the parameter can be any integer (NR1 format) within the effective value range.

Caution

Do not set the parameter to a decimal, otherwise, errors will occur.

For example,

:DISPlay:GBRightness <*brightness*>

:DISPlay:GBRightness?

Wherein, <*brightness*> can be set to an integer ranging from 1 to 100. The query returns an integer ranging from 1 to 100.

4 Real

The parameter can be any real number within the effective value range, and this command accepts parameter input in decimal (NR2 format) and scientific notation (NR3 format). For example,

:TRIGger:TIMeout:TIME <*time*>

:TRIGger:TIMeout:TIME?

Wherein, <*time*> can be set to any real number ranging from 1.6E-8 (i.g. 16 ns) to 1E+1 (i.g. 10 s). The query returns a real number in scientific notation.

5 ASCII String

The parameter can be the combinations of ASCII characters. For example,

:SYSTem:OPTion:INSTall <*license*>

Wherein, <*license*> can be set to

```
DS70000-
RTSA@724bf6bfd20ba23421bf1c98091cf7e87f5917401ce1a860fb1d975602096
27061dfb0fa414e71e178963cbd898dbd8e
```

Command Abbreviation

All the commands are case-insensitive. They can all be in upper case or in lower case. If abbreviation is used, you must input all the capital letters in the command. For example,

:DISPlay:GBRightness?

can be abbreviated as

:DISP:GBR?

3 Command System

This chapter introduces the syntax, functions, parameters, and usage of each DS70000 command.

Caution

1. **Unless otherwise specified, the descriptions in this manual all take DS70504 as an example.**
2. **For the parameter setting command (time, frequency, amplitude, etc.), the digital oscilloscope can only recognize the numbers, unable to recognize the unit sent together with them. The unit of the parameter is a default one. For the default units of various parameters, refer to the descriptions for the specified command.**

3.1 :AUToscale

Syntax

```
:AUToscale
```

Description

Enables the waveform auto setting function. The oscilloscope will automatically adjust the vertical scale, horizontal time base, and trigger mode according to the input signal to realize optimal waveform display. This command functions the same as clicking or tapping the **Auto** icon in the function navigation menu (clicking or tapping  at the lower-left part of the screen and then select the Auto icon).

Parameter

N/A

Remarks

- When the AUTO function is disabled, this command is invalid. For details, refer to *:SYSTem:AUToscale*.
- When the pass/fail test is enabled, the AUTO function runs normally, but the pass/fail test function is forced to be disabled.
- When the waveform recording function is enabled, the AUTO function runs normally, but the recording or playing function is forced to be disabled.

Return Format

N/A

Example

```
N/A
```

3.2 :CLEar

Syntax

:CLEar

Description

Clears all the waveforms on the screen. This command functions the same as clicking or tapping the **Clear** icon in the function navigation menu (clicking or tapping  at the lower-left part of the screen and then select the Clear icon).

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.3 :RUN

Syntax

:RUN

Description

The :RUN command starts running the oscilloscope This command functions the same as clicking or tapping the icon 

in the quick operation bar at the upper-right part of the screen.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.4 :STOP

Syntax

`:STOP`

Description

The :STOP command stops running the oscilloscope. This command functions the same as clicking or tapping the icon 

in the quick operation bar at the upper-right part of the screen.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.5 :SINGle

Syntax

`:SINGle`

Description

Single trigger mode. Sets the trigger mode of the oscilloscope to "Single". This command functions the same as sending the *:TRIGger:SWEep SINGle* command.

Parameter

N/A

Remarks

- In the single trigger mode, the oscilloscope performs a single trigger when the trigger conditions are met and then it stops.
- When the waveform recording function is enabled or the recorded waveforms are played back, this command is invalid.
- For the single trigger, you can use the *:TFORce* command to generate one trigger by force.

Return Format

N/A

Example

```
N/A
```

3.6 :TFORce

Syntax

```
:TFORce
```

Description

Generates a trigger signal forcefully. This command is only applicable to the normal and single trigger modes. Refer to the *:TRIGger:SWEep* command).

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.7 :ACQuire Commands

The **:ACQuire** commands are used to set the memory depth of the oscilloscope, the acquisition mode, the average times, as well as query the current sample rate.

3.7.1 :ACQuire:AVERages

Syntax

:ACQuire:AVERages <*count*>

:ACQuire:AVERages?

Description

Sets or queries the number of averages in the average acquisition mode.

Parameter

Name	Type	Range	Default
<count>	Integer	2^n (n is an integer, and its range is from 1 to 16).	2

Remarks

- You can send the *:ACQuire:TYPE* command to set the acquisition mode.
- In the average acquisition mode, greater number of averages can lower the noise and increase the vertical resolution; but will also slow the response of the displayed waveform to the waveform changes.

Return Format

The query returns an integer ranging from 2 to 65536.

Example

```
:ACQuire:AVERages 128    /*Sets the average times to 128.*/
:ACQuire:AVERages?  /*The query returns 128.*/
```

3.7.2 :ACQuire:BITS

Syntax

:ACQuire:BITS <*bit*>

:ACQuire:BITS?

Description

Sets or queries the resolution bits supported in high resolution mode.

Parameter

Name	Type	Range	Default
<bit>	Discrete	{9|10|11|12|14|16}	9

Remarks

This oscilloscope supports 9-16 bits in high resolution mode; 8 bits in other modes.

Return Format

The query returns the vertical resolution in integer. Its unit is bit.

Example

```
:ACQuire:BITS 10        /*Sets the resolution in high resolution mode
to 10 bits.*/
:ACQuire:BITS?          /*The query returns 10.*/
```

3.7.3 :ACQuire:MDEPth

Syntax

:ACQuire:MDEPth <*mdep*>

:ACQuire:MDEPth?

Compatible Command Syntax

:ACQuire:MEMDepth?

Description

Sets or queries the memory depth of the oscilloscope (i.e. the number of waveform points that can be stored through the sampling in a single trigger). The default unit is pts.

Parameter

Name	Type	Range	Default
<mdep>	Discrete	{AUTO|1k|10k|100k|1M|10M|100M|200M|500M|1G|2G|1000|10000|100000|1000000|10000000|100000000|200000000|500000000|1000000000|2000000000|1e3|1e4|1e5|1e6|1e7|1e8|2e8|5e8|1e9|2e9}	10k

Remarks

- When you select the "Auto" mode, the oscilloscope selects the memory depth automatically according to the current sample rate.
- For the specified memory depth:
  - When only one of the four channels (CH1, CH2, CH3, and CH4) is enabled, the max. memory depth is 2G (option).

- When any two or multiple channels are enabled, the max. memory depth is 1G.

Return Format

The query returns the memory depth in scientific notation.

Return Format of Compatible Command Syntax

The query returns the memory depth in strings.

Example

```
:ACQuire:MDEPth 1 M    /*Sets the memory depth to 1M.*/
:ACQuire:MDEPth?         /*The query returns 1.000E+6.*/
```

Compatible Command Example

```
:ACQuire:MEMDepth?   /*The query returns 1M.*/
```

3.7.4 :ACQuire:TYPE

Syntax

:ACQuire:TYPE <*type*>

:ACQuire:TYPE?

Description

Sets or queries the acquisition mode of the oscilloscope.

Parameter

Name	Type	Range	Default
<type>	Discrete	{NORMal|PEAK|AVERages|HRESolution}	NORMal

Remarks

- **NORMal:** In this mode, the oscilloscope samples the signal at a specified fixed time interval to rebuild the waveform. For most of the waveforms, using this mode can produce the optimal display effects.
- **AVERages:** In this mode, the oscilloscope averages the waveforms from multiple samples to reduce the random noise of the input signal and improve the vertical resolution. Greater number of averages can lower the noises and increase the vertical resolution, but will also slow the response of the displayed waveform to the waveform changes.

- **PEAK:** indicates the peak detection. In this mode, the oscilloscope samples the maximum and minimum value of the signal at the fixed sampling interval to acquire the signal envelope or the narrow pulses that might be lost. In this mode, signal aliasing can be prevented, but the noise displayed would be larger.
- **HRESolution:** indicates a high resolution. The oscilloscope will average the adjacent sample points of the sample waveform to lower the random noises of the input signals and display much more smoother waveforms. If the sample rate of the digital converter is greater than the storage rate of the acquisition memory, this mode is often adopted.

Return Format

The query returns NORM, PEAK, AVER, or HRES.

Example

```
:ACQuire:TYPE AVERages    /*Sets the acquisition mode to AVERages.*/
:ACQuire:TYPE?                /*The query returns AVER.*/
```

3.7.5 :ACQuire:SRATe?

Syntax

`:ACQuire:SRATe?`

Description

Queries the current sample rate. The default unit is Sa/s.

Parameter

N/A

Remarks

- Sample rate indicates the frequency of the signal sampling, i.e. the number of waveform points sampled per second.
- The formula below describes the relationship among sample rate, memory depth, and waveform length:

  memory depth = sample rate x waveform length

  Wherein, the memory depth is set by the *:ACQuire:MDEPth* command. The waveform length is obtained by multiplying the horizontal time base (set by the *:TIMebase[:MAIN]:SCALe* command) by the number of grids in the horizontal direction.

Return Format

The query returns the sample rate in scientific notation.

Example

```
:ACQuire:SRATe?   /*The query returns 2.500000E+9.*/
```

3.7.6 :ACQuire:AALias

Syntax

:ACQuire:AALias <*bool*>

:ACQuire:AALias?

Description

Enables or disables the anti-aliasing function of the oscilloscope; or queries the on/off status of the anti-aliasing function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:ACQuire:AALias ON    /*Enables the anti-aliasing function.*/
:ACQuire:AALias?       /*The query returns 1.*/
```

3.8 :BUS<n> Commands

The : BUS<n> commands are used to execute the decoding-related settings and operations.

3.8.1 :BUS<n>:MODE

Syntax

:BUS<*n*>**:MODE** <*mode*>

:BUS<*n*>**:MODE?**

Description

Sets or queries the decoding type of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<mode>	Discrete	{PARallel|RS232|SPI|IIC|IIS|LIN|CAN|FLEXray|M1553}	PARallel

Remarks

Except PARallel, all the other decodings are options. Only when the specified option is installed, can the command be available.

Return Format

The query returns PAR, RS232, SPI, IIC, IIS, LIN, CAN, FLEX, or M1553.

Example

```
:BUS1:MODE SPI     /*Sets the type of the decoding bus to SPI.*/
:BUS1:MODE?        /*The query returns SPI.*/
```

3.8.2 :BUS<n>:DISPlay

Syntax

:BUS<*n*>**:DISPlay** <*bool*>

:BUS<*n*>**:DISPlay?**

Description

Enables or disables the specified decoding bus; or queries the on/off display status of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:BUS1:DISPlay ON    /*Enables the decoding bus.*/
:BUS1:DISPlay?       /*The query returns 1.*/
```

3.8.3 :BUS<n>:FORMat

Syntax

:BUS<*n*>**:FORMat** <*format*>

:BUS<*n*>**:FORMat?**

Description

Sets or queries the display format of decoding data of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<format>	Discrete	{HEX|ASCii|DEC|BIN}	HEX

Remarks

- **HEX:** indicates Hexadecimal.
- **DEC:** indicates Decimal.
- **BIN:** indicates Binary.

Return Format

The query returns HEX, ASC, DEC, or BIN.

Example

```
:BUS1:FORMat HEX    /*Sets the display format of the bus to Hex.*/
:BUS1:FORMat?       /*The query returns HEX.*/
```

3.8.4 :BUS<n>:EVENt

Syntax

:BUS<*n*>**:EVENt** <*bool*>

:BUS<*n*>**:EVENt?**

Description

Enables or disables the event table of the specified decoding bus; or queries the on/off status of the specified decoding bus event table.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

Before using the command, enable the specified decoding bus.

Return Format

The query returns 1 or 0.

Example

```
:BUS1:EVENt ON     /*Enables the event table of the specified
decoding bus.*/
:BUS1:EVENt?       /*The query returns 1.*/
```

3.8.5 :BUS<n>:EVENt:VIEW

Syntax

:BUS<*n*>**:EVENt:VIEW** <*packet*>

:BUS<*n*>**:EVENt:VIEW?**

Description

Sets or queries the data page of the specified decoding bus event table.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<packet>	Discrete	{PACKets|DETails|PAYLoad}	PACKets

Remarks

- **PACKets:** displays time, data, and error information in the specified event table.
- **DETails:** displays the detailed data of the specified row in the event table.
- **PAYLoad:** displays all the data of the specified column in the event table.

When different views are selected, the export format of the data list will be changed accordingly.

Return Format

The query returns PACK, DET, or PAYL.

Example

```
:BUS1:EVENt:VIEW DETails        /*Sets the data page of the decoding
bus event table to DETails.*/
:BUS1:EVENt:VIEW?              /*The query returns DET.*/
```

3.8.6 :BUS<n>:LABel

Syntax

:BUS<*n*>**:LABel** <*bool*>

:BUS<*n*>**:LABel?**

Description

Enables or disables the label of the specified decoding bus; or queries the on/off display status of the label of the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

Before using the command, enable the specified decoding bus.

Return Format

The query returns 1 or 0.

Example

```
:BUS1:LABel ON  /*Enables the label of the specified decoding bus.*/
:BUS1:LABel?      /*The query returns 1.*/
```

3.8.7 :BUS<n>:DATA?

Syntax

:BUS<*n*>**:DATA?**

Description

Reads the data in the decoding event table.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Remarks

N/A

Return Format

The query returns the data in the decoding event table with the following formats.

```
#9000000086PARALLEL
Time,Data,
-2.47us,0,
-2.444us,1,
-1.448us,0,
-446ns,1,
551.6ns,0,
1.554us,1,
```

Wherein, "#9000000086" is the TMC data block header, which is followed by the data in the event table. The 9-digit data following #9 in the data block header indicates the number of bytes of the effective data. "PARALLEL" indicates the decoding type. The available decoding type can also be RS232, I2C, SPI, LIN, FlexRay, I2S, 1553B, or CAN. The data are separated by a comma, and will automatically switch to the next line according to the line length limit in the decoding list. The data value is related to the numeral system that you has set.

Caution

You can save all the data (except TMC data block header and decoding type, e.g. #9000000086PARALLEL) as the "*.csv" file and view the data in the form of a list.

Example

```
N/A
```

3.8.8 :BUS<n>:EEXPort

Syntax

:BUS<*n*>**:EEXPort** <*path*>

Description

Exports the decoding information in the specified decoding bus event table in CSV form.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<path>	ASCII String	Refer to *Remarks*	-

Remarks

- <path> includes the file storage location and the filename with a suffix. If the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The stored "*.csv" file can be opened and edited in Excel.

Return Format

N/A

Example

```
:BUS1:EEXPort C:/123.csv    /*Exports the decoding information in
the bus event table to the local Disk C, with the filename
123.csv.*/
```

3.8.9 :BUS<n>:POSition

Syntax

:BUS<*n*>**:POSition** <*pos*>

:BUS<*n*>**:POSition?**

Description

Sets or queries the vertical position of the bus on the screen.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<pos>	Integer	-250 to 250	0

Remarks

N/A

Return Format

The query returns an integer ranging from -250 to 250.

Example

```
:BUS1:POSition 200    /*Sets the vertical position of the bus to
200.*/
:BUS1:POSition?       /*The query returns 200.*/
```

3.8.10 :BUS<n>:THReshold

Syntax

:BUS<*n*>**:THReshold** <*value*>,<*type*>

:BUS<*n*>**:THReshold?** <*type*>

Description

Sets or queries the threshold of the specified decoding source.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<value>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0
<type>	Discrete	{PAL|TX|RX|SCL|SDA|CS|CLK|MISO|MOSI|LIN|CAN|CANSub1|FLEX|1553}	-

Remarks

- For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.
- For RS232 decoding, only when the Rx source is enabled, can you set the threshold.
- For SPI decoding, only when the MOSI source is enabled, can you set the threshold.

Return Format

The query returns the threshold of the specified decoding source in scientific notation.

Example

```
:BUS1:THReshold 2.4,PAL     /*Sets the threshold of the PAL
decoding source to 2.4 V.*/
:BUS1:THReshold? PAL       /*The query returns 2.400000E0.*/
```

3.8.11 :BUS<n>:PARallel:BUS

Syntax

:BUS<*n*>**:PARallel:BUS** <*source*>

:BUS<*n*>**:PARallel:BUS?**

Description

Sets or queries the current source of the Parallel decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|USER}	-

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or USER.

Example

```
:BUS1:PARallel:BUS CHANnel1    /*Sets the current source of the
Parallel decoding bus to CHANnel1.*/
:BUS1:PARallel:BUS?      /*The query returns CHAN1.*/
```

3.8.12 :BUS<n>:PARallel:CLK

Syntax

:BUS<*n*>**:PARallel:CLK** <*source*>

:BUS<*n*>**:PARallel:CLK?**

Description

Sets or queries the clock source of the Parallel decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|OFF}	OFF

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:PARallel:CLK CHANnel2    /*Sets the clock source of the
Parallel decoding to CHANnel2.*/
:BUS1:PARallel:CLK?      /*The query returns CHAN2.*/
```

3.8.13 :BUS<n>:PARallel:SLOPe

Syntax

:BUS<*n*>**:PARallel:SLOPe** <*slope*>

:BUS<*n*>**:PARallel:SLOPe?**

Description

Sets or queries the edge type of the clock channel when being sampled by Parallel decoding on the data channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<slope>	Discrete	{POSitive|NEGative|BOTH}	POSitive

Remarks

If no clock channel is selected, the instrument will sample when the channel data jumps during the decoding.

Return Format

The query returns POS, NEG, or BOTH.

Example

```
:BUS1:PARallel:SLOPe BOTH    /*Sets the Parallel decoding to sample
on any edge of the clock channel.*/
:BUS1:PARallel:SLOPe?         /*The query returns BOTH.*/
```

3.8.14 :BUS<n>:PARallel:WIDTh

Syntax

:BUS<*n*>**:PARallel:WIDTh** <*wid*>

:BUS<*n*>**:PARallel:WIDTh?**

Description

Sets or queries the data width of the parallel bus, i.e. the number of bits per frame.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<wid>	Integer	1 to 4	1

Remarks

- After you send this command to set the data width of the bus, send the *:BUS<n>:PARallel:BITX* and *:BUS<n>:PARallel:SOURce* command to select the bit respectively and set the channel source for the bit.
- Only when the bus source is set to User, can this command be valid.

Return Format

The query returns an integer ranging from 1 to 4.

Example

```
:BUS1:PARallel:WIDTh 4    /*Sets the data width of Parallel
decoding to 4.*/
:BUS1:PARallel:WIDTh?      /*The query returns 4.*/
```

3.8.15 :BUS<n>:PARallel:BITX

Syntax

:BUS<*n*>**:PARallel:BITX** <*bit*>

:BUS<*n*>**:PARallel:BITX?**

Description

Sets or queries the data bit that the parallel bus requires to set for the channel source.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bit>	Integer	0 to (data width - 1)	0

Remarks

- The data width is set by the *:BUS<n>:PARallel:WIDTh* command.
- After selecting the desired bit, send the *:BUS<n>:PARallel:SOURce* command to set the channel source for the bit.

Return Format

The query returns the current data bits in integer. Its unit is Hz.

Example

```
:BUS1:PARallel:BITX 2     /*Sets the current bit to 2.*/
:BUS1:PARallel:BITX?      /*The query returns 2.*/
```

3.8.16 :BUS<n>:PARallel:SOURce

Syntax

:BUS<*n*>**:PARallel:SOURce** <*src*>

:BUS<*n*>**:PARallel:SOURce?**

Description

Sets or queries the channel source of the currently selected data bit.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<src>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	Related to the selected bit

Remarks

Before sending this command, send the *:BUS<n>:PARallel:BITX* command to select the desired data bit.

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:PARallel:SOURce CHANnel2    /*Sets the channel source of the
current bit to CHANnel2.*/
:BUS1:PARallel:SOURce?             /*The query returns CHAN2.*/
```

3.8.17 :BUS<n>:PARallel:POLarity

Syntax

:BUS<*n*>**:PARallel:POLarity** <*pol*>

:BUS<*n*>**:PARallel:POLarity?**

Description

Sets or queries the data polarity of Parallel decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<pol>	Discrete	{NEGative|POSitive}	POSitive

Remarks

- **NEGative:** indicates negative polarity.
- **POSitive:** indicates positive polarity.

Return Format

The query returns NEG or POS.

Example

```
:BUS1:PARallel:POLarity NEGative    /*Sets the data polarity of
Parallel decoding to Negative.*/
:BUS1:PARallel:POLarity?            /*The query returns NEG.*/
```

3.8.18 :BUS<n>:RS232:TX (Option)

Syntax

:BUS<*n*>**:RS232:TX** <*source*>

:BUS<*n*>**:RS232:TX?**

Description

Sets or queries the TX channel source of RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|OFF}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:RS232:TX CHANnel2    /*Sets the TX channel source of RS232
decoding to CHANnel2.*/
:BUS1:RS232:TX?      /*The query returns CHAN2.*/
```

3.8.19 :BUS<n>:RS232:RX (Option)

Syntax

:BUS<*n*>**:RS232:RX** <*source*>

:BUS<*n*>**:RS232:RX?**

Description

Sets or queries the RX channel source of RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	OFF

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:RS232:RX CHANnel2    /*Sets the RX channel source of RS232
decoding to CHANnel2.*/
:BUS1:RS232:RX?      /*The query returns CHAN2.*/
```

3.8.20 :BUS<n>:RS232:POLarity (Option)

Syntax

:BUS<*n*>**:RS232:POLarity** <*pol*>

:BUS<*n*>**:RS232:POLarity?**

Description

Sets or queries the polarity of RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<pol>	Discrete	{POSitive|NEGative}	NEGative

Remarks

- The RS232 standard uses "Negative Logic", i.g. high level is Logic 0 and low level is Logic 1. Therefore, when the parameter <pol> is set to NEGative, it indicates that the polarity is set to negative logic, i.g. Normal polarity. When the parameter <pol> is set to POSitive, it indicates that the polarity is set to positive logic, i.g. Inverted polarity.

- In the RS232 decoding, the start bit of data packet indicates when to start data transmission. It is determined by the polarity. When <pol> is set to "POSitive", the start bit is 0; when "NEGative", the start bit is 1.

Return Format

The query returns POS or NEG.

Example

```
:BUS1:RS232:POLarity POSitive     /*Sets the polarity of RS232
decoding to POSitive.*/
:BUS1:RS232:POLarity?            /*The query returns POS.*/
```

3.8.21 :BUS<n>:RS232:ENDian (Option)

Syntax

:BUS<*n*>**:RS232:ENDian** <*endian*>

:BUS<*n*>**:RS232:ENDian?**

Description

Sets or queries the endian of data transmission in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<endian>	Discrete	{MSB|LSB}	LSB

Remarks

- **LSB:** indicates Least Significant Bit transmission sequence, i.g. the lowest bit of the data is transmitted first.
- **MSB:** indicates Most Significant Bit transmission sequence, i.g. the highest bit of the data is transmitted first.

Return Format

The query returns LSB or MSB.

Example

```
:BUS1:RS232:ENDian MSB        /*Sets the transmission order of
RS232 decoding to MSB.*/
:BUS1:RS232:ENDian?            /*The query returns MSB.*/
```

3.8.22 :BUS<n>:RS232:BAUD (Option)

Syntax

:BUS<*n*>**:RS232:BAUD** <*baud*>

:BUS<*n*>**:RS232:BAUD?**

Compatible Command Syntax

:BUS<*n*>**:RS232:BUSer** <*baud*>

:BUS<*n*>**:RS232:BUSer?**

Description

Sets or queries the baud rate of data transmission in RS232 decoding. The default unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<baud>	Integer	1 bps to 20 Mbps	9600 bps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 1 bps to 20 Mbps.

Example

```
:BUS1:RS232:BAUD 4800    /*Sets the baud rate of data transmission
in RS232 decoding to 4800 bps.*/
:BUS1:RS232:BAUD?              /*The query returns 4800.*/
```

Compatible Command Example

```
:BUS1:RS232:BUSer 4800        /*Sets the baud rate of data
transmission in RS232 decoding to 4800 bps.*/
:BUS1:RS232:BUSer?            /*The query returns 4800.*/
```

3.8.23 :BUS<n>:RS232:DBITs (Option)

Syntax

:BUS<*n*>**:RS232:DBITs** <*bits*>

:BUS<*n*>**:RS232:DBITs?**

Description

Sets or queries the data width of RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bits>	Discrete	{5|6|7|8|9}	8

Remarks

N/A

Return Format

The query returns 5, 6, 7, 8, or 9.

Example

```
:BUS1:RS232:DBITs 7   /*Sets the data width of RS232 decoding to
7.*/
:BUS1:RS232:DBITs?      /*The query returns 7.*/
```

3.8.24 :BUS<n>:RS232:SBITs (Option)

Syntax

:BUS<*n*>**:RS232:SBITs** <*stop bits*>

:BUS<*n*>**:RS232:SBITs?**

Description

Sets or queries the stop bits of each frame of data in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<stop bits>	Discrete	{1|1.5|2}	1

Remarks

N/A

Return Format

The query returns 1, 1.5, or 2.

Example

```
:BUS1:RS232:SBITs 2   /*Sets the stop bits of RS232 decoding to 2.*/
:BUS1:RS232:SBITs?   /*The query returns 2.*/
```

3.8.25 :BUS<n>:RS232:PARity (Option)

Syntax

:BUS<*n*>**:RS232:PARity** <*parity*>

:BUS<*n*>**:RS232:PARity?**

Description

Sets or queries the odd-even check mode of data transmission in RS232 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<parity>	Discrete	{NONE|ODD|EVEN}	NONE

Remarks

- **None:** indicates that there is no parity bit in data transmission.
- **ODD:** indicates the odd parity bit. The total count of occurrences of 1 in the data bit and check bit is an odd number. For example, if 0x55 (01010101) is transmitted, 1 shall be added to the check bit.
- **Even:** indicates the even parity bit. The total count of occurrences of 1 in the data bit and check bit is an even number. For example, if 0x55 (01010101) is transmitted, 0 shall be added to the check bit.

Return Format

The query returns NONE, ODD, or EVEN.

Example

```
:BUS1:RS232:PARity ODD        /*Sets the odd-even check mode of
data transmission of RS232 decoding to ODD.*/
:BUS1:RS232:PARity?            /*The query returns ODD.*/
```

3.8.26 :BUS<n>:IIC:SCLK:SOURce (Option)

Syntax

:BUS<*n*>**:IIC:SCLK:SOURce** <*source*>

:BUS<*n*>**:IIC:SCLK:SOURce?**

Description

Sets or queries the clock source of I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIC:SCLK:SOURce CHANnel2    /*Sets the clock source of I2C
decoding to CHANnel2.*/
:BUS1:IIC:SCLK:SOURce?      /*The query returns CHAN2.*/
```

3.8.27 :BUS<n>:IIC:SDA:SOURce (Option)

Syntax

:BUS<*n*>**:IIC:SDA:SOURce** <*source*>

:BUS<*n*>**:IIC:SDA:SOURce?**

Description

Sets or queries the data source of I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIC:SDA:SOURce CHANnel2    /*Sets the data source of I2C
decoding to CHANnel2.*/
:BUS1:IIC:SDA:SOURce?      /*The query returns CHAN2.*/
```

3.8.28 :BUS<n>:IIC:ADDRess (Option)

Syntax

:BUS<*n*>**:IIC:ADDRess** <*addr*>

:BUS<*n*>**:IIC:ADDRess?**

Description

Sets or queries the address mode of I2C decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<addr>	Discrete	{NORMal|RW}	NORMal

Remarks

- **NORMal:** indicates that the address width does not include the R/W bit.
- **RW:** indicates that the address width includes the R/W bit.

Return Format

The query returns NORM or RW.

Example

```
:BUS1:IIC:ADDRess RW    /*Sets the address of I2C decoding to
include the R/W bit.*/
:BUS1:IIC:ADDRess?       /*The query returns RW.*/
```

3.8.29 :BUS<n>:SPI:SCLK:SOURce (Option)

Syntax

:BUS<*n*>**:SPI:SCLK:SOURce** <*source*>

:BUS<*n*>**:SPI:SCLK:SOURce?**

Description

Sets or queries the clock source of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:SPI:SCLK:SOURce CHANnel2    /*Sets the clock source of SPI
decoding to CHANnel2.*/
:BUS1:SPI:SCLK:SOURce?      /*The query returns CHAN2.*/
```

3.8.30 :BUS<n>:SPI:SCLK:SLOPe (Option)

Syntax

:BUS<*n*>**:SPI:SCLK:SLOPe** <*slope*>

:BUS<*n*>**:SPI:SCLK:SLOPe?**

Description

Sets or queries the clock edge type of the SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:BUS1:SPI:SCLK:SLOPe NEGative    /*Sets the clock edge type of SPI
decoding to Negative.*/
:BUS1:SPI:SCLK:SLOPe?             /*The query returns NEG.*/
```

3.8.31 :BUS<n>:SPI:MISO:SOURce (Option)

Syntax

:BUS<*n*>**:SPI:MISO:SOURce** <*source*>

:BUS<*n*>**:SPI:MISO:SOURce?**

Description

Sets or queries the MISO data source of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|OFF}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:SPI:MISO:SOURce CHANnel2    /*Sets the MISO data source of
SPI decoding to CHANnel2.*/
:BUS1:SPI:MISO:SOURce?      /*The query returns CHAN2.*/
```

3.8.32 :BUS<n>:SPI:MISO:POLarity (Option)

Syntax

:BUS<*n*>**:SPI:MISO:POLarity** <*polarity*>

:BUS<*n*>**:SPI:MISO:POLarity?**

Description

Sets or queries the polarity of MISO data line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<polarity>	Discrete	{HIGH|LOW}	HIGH

Remarks

- **High:** indicates that low level is 1, and high level is 0.
- **Low:** indicates that high level is 1, and low level is 0.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:MISO:POLarity HIGH      /*Sets the polarity of MISO data
line to HIGH.*/
:BUS1:SPI:MISO:POLarity?          /*The query returns HIGH.*/
```

3.8.33 :BUS<n>:SPI:MOSI:SOURce (Option)

Syntax

:BUS<*n*>**:SPI:MOSI:SOURce** <*source*>

:BUS<*n*>**:SPI:MOSI:SOURce?**

Description

Sets or queries the MOSI data source of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|OFF}	OFF

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:BUS1:SPI:MOSI:SOURce CHANnel2    /*Sets the MOSI data source of
SPI decoding to CHANnel2.*/
:BUS1:SPI:MOSI:SOURce?      /*The query returns CHAN2.*/
```

3.8.34 :BUS<n>:SPI:MOSI:POLarity (Option)

Syntax

:BUS<*n*>**:SPI:MOSI:POLarity** <*polarity*>

:BUS<*n*>**:SPI:MOSI:POLarity?**

Description

Sets or queries the polarity of MOSI data line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<polarity>	Discrete	{HIGH|LOW}	HIGH

Remarks

- **High:** indicates that low level is 1, and high level is 0.
- **Low:** indicates that high level is 1, and low level is 0.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:MOSI:POLarity HIGH      /*Sets the polarity of MOSI data
line to HIGH.*/
:BUS1:SPI:MOSI:POLarity?          /*The query returns HIGH.*/
```

3.8.35 :BUS<n>:SPI:DBITs (Option)

Syntax

:BUS<*n*>**:SPI:DBITs** <*width*>

:BUS<*n*>**:SPI:DBITs?**

Description

Sets or queries the data width of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<width>	Integer	4 to 32	8

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:BUS1:SPI:DBITs 10      /*Sets the data width of SPI decoding to
10.*/
:BUS1:SPI:DBITs?      /*The query returns 10.*/
```

3.8.36 :BUS<n>:SPI:ENDian (Option)

Syntax

:BUS<*n*>**:SPI:ENDian** <*endian*>

:BUS<*n*>**:SPI:ENDian?**

Description

Sets or queries the endian of data transmission in SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<endian>	Discrete	{MSB|LSB}	MSB

Remarks

- **MSB:** indicates Most Significant Bit transmission sequence, i.e. the highest bit of the data is transmitted first.

- **LSB:** indicates Least Significant Bit transmission sequence, i.e. the lowest bit of the data is transmitted first.

Return Format

The query returns MSB or LSB.

Example

```
:BUS1:SPI:ENDian LSB          /*Sets the endian of data transmission
in SPI decoding to LSB.*/
:BUS1:SPI:ENDian?             /*The query returns LSB.*/
```

3.8.37 :BUS<n>:SPI:MODE (Option)

Syntax

:BUS<*n*>**:SPI:MODE** <*mode*>

:BUS<*n*>**:SPI:MODE?**

Description

Sets or queries the decode mode of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<mode>	Discrete	{CS|TIMeout}	TIMeout

Remarks

- **CS:** indicates chip select. CS: contains a chip select line (CS). You can perform frame synchronization according to CS.
- **TIMeout:** indicates timed out. You can perform frame synchronization according to the timeout.

  At this time, you can send the *:BUS<n>:SPI:TIMeout:TIME (Option)* command to set the timeout value.

Return Format

The query returns CS or TIM.

Example

```
:BUS1:SPI:MODE CS       /*Sets the decode mode of SPI decoding to
CS.*/
:BUS1:SPI:MODE?         /*The query returns CS.*/
```

3.8.38 :BUS<n>:SPI:TIMeout:TIME (Option)

Syntax

:BUS<*n*>**:SPI:TIMeout:TIME** <*time*>

:BUS<*n*>**:SPI:TIMeout:TIME?**

Description

Sets or queries the timeout value of SPI decoding. The default unit is s.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<time>	Real	8 ns to 10s	1 μs

Remarks

- The timeout must be greater than the maximum clock pulse width and less than the idle time between frames.
- This setting command is only valid in timeout mode.

Return Format

The query returns the timeout value in scientific notation.

Example

```
:BUS1:SPI:TIMeout:TIME 0.000005    /*Sets the timeout value to 5
us.*/
:BUS1:SPI:TIMeout:TIME?            /*The query returns
5.000000E-6.*/
```

3.8.39 :BUS<n>:SPI:SS:SOURce (Option)

Syntax

:BUS<*n*>**:SPI:SS:SOURce** <*source*>

:BUS<*n*>**:SPI:SS:SOURce?**

Description

Sets or queries the source channel of the CS line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Real	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel3

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:SPI:SS:SOURce CHANnel2    /*Sets the source channel of CS
line of SPI decoding to CHANnel2.*/
:BUS1:SPI:SS:SOURce?      /*The query returns CHAN2.*/
```

3.8.40 :BUS<n>:SPI:SS:POLarity (Option)

Syntax

:BUS<*n*>**:SPI:SS:POLarity** <*polarity*>

:BUS<*n*>**:SPI:SS:POLarity?**

Description

Sets or queries the polarity of the CS line of SPI decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<polarity>	Discrete	{HIGH|LOW}	LOW

Remarks

- **HIGH:** indicates that the oscilloscope samples data of the source channel of data line on the specified edge of the clock signal when the CS signal is high level.
- **LOW:** indicates that the oscilloscope samples data of the source channel of data line on the specified edge of the clock signal when the CS signal is low level.

Return Format

The query returns HIGH or LOW.

Example

```
:BUS1:SPI:SS:POLarity HIGH      /*Sets the polarity of CS line of
SPI decoding to HIGH.*/
:BUS1:SPI:SS:POLarity?          /*The query returns HIGH.*/
```

3.8.41 :BUS<n>:CAN:SOURce (Option)

Syntax

:BUS<*n*>**:CAN:SOURce** <*source*>

:BUS<*n*>**:CAN:SOURce?**

Description

Sets or queries the source channel of CAN decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:CAN:SOURce CHANnel2    /*Sets the source channel of CAN
decoding to CHANnel2.*/
:BUS1:CAN:SOURce?      /*The query returns CHAN2.*/
```

3.8.42 :BUS<n>:CAN:STYPe (Option)

Syntax

:BUS<*n*>**:CAN:STYPe** <*stype*>

:BUS<*n*>**:CAN:STYPe?**

Description

Sets or queries the signal type of CAN decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<stype>	Discrete	{TX|RX|CANH|CANL|DIFFerential}	CANL

Remarks

- **TX:** indicates the Transmit signal from the CAN bus transceiver.
- **RX:** indicates the Receive signal from the CAN bus transceiver.
- **CANH:** indicates the actual CAN_H differential bus signal.
- **CANL:** indicates the actual CAN_L differential bus signal.
- **DIFFerential:** indicates the CAN differential bus signal connected to an analog channel by using a differential probe. Connect the differential probe's positive lead to the CAN_H bus signal and connect the negative lead to the CAN_L bus signal..

Return Format

The query returns TX, RX, CANH, CANL, or DIFF.

Example

```
:BUS1:CAN:STYPe TX          /*Sets the signal type of CAN decoding to
TX.*/
:BUS1:CAN:STYPe?            /*The query returns TX.*/
```

3.8.43 :BUS<n>:CAN:BAUD (Option)

Syntax

:BUS<*n*>**:CAN:BAUD** <*baud*>

:BUS<*n*>**:CAN:BAUD?**

Compatible Command Syntax

:BUS<*n*>**:CAN:BUSer** <*baud*>

:BUS<*n*>**:CAN:BUSer?**

Description

Sets or queries the signal rate of CAN decoding. The unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<baud>	Integer	10 kbps to 5 Mbps	1 Mbps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 10 kbps to 5 Mbps.

Example

```
:BUS1:CAN:BAUD 120000    /*Sets the signal rate of CAN decoding to
120000 bps*/
:BUS1:CAN:BAUD?              /*The query returns 120000.*/
```

Compatible Command Example

```
:BUS1:CAN:BUSer 120000        /*Sets the user-defined signal rate
of CAN decoding to 125000 bps*/
:BUS1:CAN:BUSer?              /*The query returns 120000.*/
```

3.8.44 :BUS<n>:CAN:SPOint (Option)

Syntax

:BUS<*n*>**:CAN:SPOint** <*spoint*>

:BUS<*n*>**:CAN:SPOint?**

Description

Sets or queries the sample point position of CAN decoding (expressed in %).

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<spoint>	Integer	10 to 90	50

Remarks

The sample point is within the range of the bit time. The oscilloscope samples the bit level at the sample point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:BUS1:CAN:SPOint 70          /*Sets the sample point position of CAN
decoding to 70%.*/
:BUS1:CAN:SPOint?            /*The query returns 70.*/
```

3.8.45 :BUS<n>:CAN:FDBAUD (Option)

Syntax

:BUS<n>:CAN:FDBAUD <*baud*>

:BUS<n>:CAN:FDBAUD?

Compatible Command Syntax

:BUS<n>:CAN:FDBUSer <*baud*>

:BUS<n>:CAN:FDBUSer?

Description

Sets or queries the signal rate of CAN-FD decoding. The unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<baud>	Integer	0 to 200 Mbps	100 Mbps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 0 to 200000000.

Example

```
:BUS1:CAN:FDBAUD 120000     /*Sets the signal rate of CAN-FD
decoding to 120000 bps*/
:BUS1:CAN:FDBAUD?               /*The query returns 120000.*/
```

Compatible Command Example

```
:BUS1:CAN:FDBUSer 120000        /*Sets the user-defined signal rate
of CAN-FD decoding to 120000 bps*/
:BUS1:CAN:FDBUSer?              /*The query returns 120000.*/
```

3.8.46 :BUS<n>:CAN:FDSPoint (Option)

Syntax

:BUS<n>:CAN:FDSPoint <*spoint*>

:BUS<n>:CAN:FDSPoint?

Description

Sets or queries the sample point position of CAN-FD decoding (expressed in %).

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<spoint>	Integer	10 to 90	50

Remarks

The sample point is within the range of the bit time. The oscilloscope samples the bit level at the sample point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:BUS1:CAN:FDSPoint 70          /*Sets the sample point position of
CAN-FD decoding to 70%.*/
:BUS1:CAN:FDSPoint?            /*The query returns 70.*/
```

3.8.47 :BUS<n>:FLEXray:BAUD (Option)

Syntax

:BUS<*n*>**:FLEXray:BAUD** <*baud*>

:BUS<*n*>**:FLEXray:BAUD?**

Description

Sets or queries the signal rate of FlexRay decoding. The default unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<baud>	Discrete	{2500000|5000000|10000000}	10000000

Remarks

N/A

Return Format

The query returns 2500000, 5000000, or 10000000.

Example

```
:BUS1:FLEXray:BAUD 2500000          /*Sets the signal rate of
FlexRay decoding to 2500000 bps.*/
:BUS1:FLEXray:BAUD?                 /*The query returns 2500000.*/
```

3.8.48 :BUS<n>:FLEXray:SOURce (Option)

Syntax

:BUS<*n*>**:FLEXray:SOURce** <*source*>

:BUS<*n*>**:FLEXray:SOURce?**

Description

Sets or queries the source channel of FlexRay decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:FLEXray:SOURce CHANnel2    /*Sets the source channel of
FlexRay decoding to CHANnel2.*/
:BUS1:FLEXray:SOURce?      /*The query returns CHAN2.*/
```

3.8.49 :BUS<n>:FLEXray:SPOint (Option)

Syntax

:BUS<*n*>**:FLEXray:SPOint** <*spoint*>

:BUS<*n*>**:FLEXray:SPOint?**

Description

Sets or queries the sample point position of FlexRay decoding (expressed in %).

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<spoint>	Integer	10 to 90	50

Remarks

The sample point is within the range of the bit time. The oscilloscope samples the bit level at the sample point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:BUS1:FLEXray: SPOint 70          /*Sets the sample point position of
FlexRay decoding to 70%.*/
:BUS1:FLEXray: SPOint?            /*The query returns 70.*/
```

3.8.50 :BUS<n>:FLEXray:STYPe (Option)

Syntax

:BUS<*n*>**:FLEXray:STYPe** <*stype*>

:BUS<*n*>**:FLEXray:STYPe?**

Description

Sets or queries the signal type of FlexRay decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<stype>	Discrete	{BP|BM|RT}	BP

Remarks

N/A

Return Format

The query returns BP, BM, or RT.

Example

```
:BUS1:FLEXray:STYPe BM          /*Sets the signal type of FlexRay
decoding to BM.*/
:BUS1:FLEXray:STYPe?            /*The query returns BM.*/
```

3.8.51 :BUS<n>:LIN:BAUD (Option)

Syntax

:BUS<*n*>**:LIN:BAUD** <*baud*>

:BUS<*n*>**:LIN:BAUD?**

Compatible Command Syntax

:BUS<*n*>**:LIN:BUSer** <*baud*>

:BUS<*n*>**:LIN:BUSer?**

Description

Sets or queries the baud rate of LIN decoding. The default unit is bps.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<baud>	Integer	2.4 kbps to 20 Mbps	19200bps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 2.4 kbps to 20 Mbps.

Example

```
:BUS1:LIN:BAUD 9600    /*Sets the signal baud rate of LIN decoding
to 9600 bps*/
:BUS1:LIN:BAUD?              /*The query returns 9600.*/
```

Compatible Command Example

```
:BUS1:LIN:BUSer 9600        /*Sets the signal baud rate of LIN
decoding to 9600 bps*/
:BUS1:LIN:BUSer?              /*The query returns 9600.*/
```

3.8.52 :BUS<n>:LIN:POLarity (Option)

Syntax

:BUS<*n*>**:LIN:POLarity** <*bool*>

:BUS<*n*>**:LIN:POLarity?**

Description

Sets or queries the parity of LIN decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:BUS1:LIN:POLarity ON            /*Sets the parity bit to be
included in LIN decoding.*/
:BUS1:LIN:POLarity?               /*The query returns 1.*/
```

3.8.53 :BUS<n>:LIN:SOURce (Option)

Syntax

:BUS<*n*>**:LIN:SOURce** <*source*>

:BUS<*n*>**:LIN:SOURce?**

Description

Sets or queries the source of LIN bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:LIN:SOURce CHANnel2    /*Sets the source of LIN bus to
CHANnel2.*/
:BUS1:LIN:SOURce?      /*The query returns CHAN2.*/
```

3.8.54 :BUS<n>:LIN:STANdard (Option)

Syntax

:BUS<*n*>**:LIN:STANdard** <*value*>

:BUS<*n*>**:LIN:STANdard?**

Description

Sets or queries the version of LIN bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<value>	Discrete	{V1X|V2X|MIXed}	MIXed

Remarks

N/A

Return Format

The query returns V1X, V2X, or MIX.

Example

```
:BUS1:LIN:STANdard V2X                /*Sets the LIN bus version to
V2X.*/
:BUS1:LIN:STANdard?                   /*The query returns V2X.*/
```

3.8.55 :BUS<n>:IIS:SOURce:CLOCk (Option)

Syntax

:BUS<*n*>**:IIS:SOURce:CLOCk** <*source*>

:BUS<*n*>**:IIS:SOURce:CLOCk?**

Description

Sets or queries the clock source of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIS:SOURce:CLOCk CHANnel2        /*Sets the clock source of
the I2S decoding to CHANnel2.*/
:BUS1:IIS:SOURce:CLOCk?           /*The query returns CHAN2.*/
```

3.8.56 :BUS<n>:IIS:SOURce:DATA (Option)

Syntax

:BUS<*n*>**:IIS:SOURce:DATA** <*source*>

:BUS<*n*>**:IIS:SOURce:DATA?**

Description

Sets or queries the data source of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel3

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIS:SOURce:DATA CHANnel2        /*Sets the data source of the
I2S decoding to CHANnel2.*/
:BUS1:IIS:SOURce:DATA?           /*The query returns CHAN2.*/
```

3.8.57 :BUS<n>:IIS:SOURce:WSELect (Option)

Syntax

:BUS<*n*>**:IIS:SOURce:WSELect** <*source*>

:BUS<*n*>**:IIS:SOURce:WSELect?**

Description

Sets or queries the audio channel of the I2S trigger.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:IIS:SOURce:WSELect CHANnel2    /*Sets the audio channel to
CHANnel2.*/
:BUS1:IIS:SOURce:WSELect?            /*The query returns CHAN2.*/
```

3.8.58 :BUS<n>:IIS:ALIGnment (Option)

Syntax

:BUS<*n*>**:IIS:ALIGnment** <*align*>

:BUS<*n*>**:IIS:ALIGnment?**

Description

Sets or queries the alignment mode of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<align>	Discrete	{IIS|RJ|LJ}	IIS

Remarks

- **IIS:** first sends MSB of each sample data and then sends LSB.
- **RJ:** data transmission (MSB first) is right-justified to the WS transition.
- **LJ:** data transmission (MSB first) begins at the edge of the WS transition.

Return Format

The query returns IIS, RJ, or LJ.

Example

```
:BUS1:IIS:ALIGnment RJ           /*Sets the alignment mode of the
I2S decoding to RJ.*/
:BUS1:IIS:ALIGnment?             /*The query returns RJ.*/
```

3.8.59 :BUS<n>:IIS:CLOCk:SLOPe (Option)

Syntax

:BUS<*n*>**:IIS:CLOCk:SLOPe** <*slope*>

:BUS<*n*>**:IIS:CLOCk:SLOPe?**

Description

Sets or queries the clock edge type of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<slope>	Discrete	{NEGative|POSitive}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:BUS1:IIS:CLOCk:SLOPe NEGative    /*Sets the clock edge of I2S
decoding to NEGative.*/
:BUS1:IIS:CLOCk:SLOPe?            /*The query returns NEG.*/
```

3.8.60 :BUS<n>:IIS:RWIDth (Option)

Syntax

:BUS<*n*>**:IIS:RWIDth** <*val*>

:BUS<*n*>**:IIS:RWIDth?**

Description

Sets or queries the word size of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<val>	Integer	4 to 32	4

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:BUS1:IIS:RWIDth 5              /*Sets the of I2S decoding word size
to 5.*/
:BUS1:IIS:RWIDth?             /*The query returns 5.*/
```

3.8.61 :BUS<n>:IIS:RECEWIDth (Option)

Syntax

:BUS<n>:IIS:RECEWIDth <*val*>

:BUS<n>:IIS:RECEWIDth?

Description

Sets or queries the Receive width of the I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<val>	Integer	4 to 32	4

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:BUS1:IIS:RECEWIDth 5                /*Sets the Receive width of I2S
decoding to 5.*/
:BUS1:IIS:RECEWIDth?               /*The query returns 5.*/
```

3.8.62 :BUS<n>:IIS:WSLow (Option)

Syntax

:BUS<n>:IIS:WSLow <*bool*>

:BUS<n>:IIS:WSLow?

Description

Sets or queries the audio polarity of I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

- **0|OFF:** indicates negative polarity.
- **1|ON:** indicates positive polarity.

Return Format

The query returns 0 or 1.

Example

```
:BUS1:IIS:WSLow OFF    /*Sets the audio polarity of I2S decoding to
Negative.*/
:BUS1:IIS:WSLow?            /*The query returns 0.*/
```

3.8.63 :BUS<n>:IIS:ENDian (Option)

Syntax

:BUS<n>:IIS:ENDian <*bool*>

:BUS<n>:IIS:ENDian?

Description

Sets or queries the endian of I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

- **1|ON:** indicates Most Significant Bit transmission sequence, i.g. the highest bit of the data is transmitted first.
- **0|OFF:** indicates Least Significant Bit transmission sequence, i.g. the lowest bit of the data is transmitted first.

Return Format

The query returns 0 or 1.

Example

```
:BUS1:IIS:ENDian OFF       /*Sets the endian of I2S decoding to
LSB.*/
:BUS1:IIS:ENDian?          /*The query returns 0.*/
```

3.8.64 :BUS<n>:IIS:POLarity (Option)

Syntax

:BUS<n>:IIS:POLarity <*bool*>

:BUS<n>:IIS:POLarity?

Description

Sets or queries the data polarity of I2S decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Discrete	{{1|ON}|{0|OFF}}	1|ON

Remarks

- **1|ON:** indicates positive polarity.
- **0|OFF:** indicates negative polarity.

Return Format

The query returns 0 or 1.

Example

```
:BUS1:IIS:POLarity OFF    /*Sets the data polarity of I2S decoding
to Negative.*/
:BUS1:IIS:POLarity?            /*The query returns 0.*/
```

3.8.65 :BUS<n>:M1553:SOURce (Option)

Syntax

:BUS<*n*>**:M1553:SOURce** <*source*>

:BUS<*n*>**:M1553:SOURce**

Description

Sets or queries the source of the M1553 decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:M1553:SOURce CHANnel2    /*Sets the source of the M1553 decoding to CHANnel2.*/
:BUS1:M1553:SOURce?      /*The query returns CHAN2.*/
```

3.9 :CHANnel Commands

The **:CHANnel<n>** commands are used to set or query the bandwidth limit, coupling, vertical scale, vertical offset, and other vertical system parameters of the analog channel.

3.9.1 :CHANnel<n>:BWLimit

Syntax

:CHANnel<*n*>**:BWLimit** <*val*>

:CHANnel<*n*>**:BWLimit?**

Description

Sets or queries the bandwidth limit of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<val>	Discrete	Refer to *Remarks*	OFF

Remarks

- **1 MΩ:** {OFF|ON|20M|250M}, with the unit Hz
- **50 Ω:** {OFF|ON|20M|250M|1G|2G}, with the unit Hz

Return Format

- **1 MΩ:** The query returns 20M, 250M, or OFF.
- **50 Ω:** The query returns 20M, 250M, 1GHz, 2GHz, or OFF.

Example

```
:CHANnel1:BWLimit 20M     /*Enables the 20MHz bandwidth limit.*/
:CHANnel1:BWLimit?        /*The query returns 20M.*/
```

3.9.2 :CHANnel<n>:COUPling

Syntax

:CHANnel<*n*>**:COUPling** <*coupling*>

:CHANnel<*n*>**:COUPling?**

Description

Sets or queries the coupling mode of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<coupling>	Discrete	{AC|DC|GND}	DC

Description

- **AC:** the DC components of the signal under test are blocked.
- **DC:** both DC and AC components of the signal under test can pass through the channel.
- **GND:** both DC and AC components of the signal under test are blocked.

Return Format

The query returns AC, DC, or GND.

Example

```
:CHANnel1:COUPling AC     /*Selects the AC coupling mode.*/
:CHANnel1:COUPling?       /*The query returns AC.*/
```

3.9.3 :CHANnel<n>:DISPlay

Syntax

:CHANnel<*n*>**:DISPlay** <*bool*>

:CHANnel<*n*>**:DISPlay?**

Description

Turns on or off the specified channel; or queries the on/off status of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:DISPlay ON     /*Enables CH1.*/
:CHANnel1:DISPlay?       /*The query returns 1.*/
```

3.9.4 :CHANnel<n>:INVert

Syntax

:CHANnel<*n*>**:INVert** <*bool*>

:CHANnel<*n*>**:INVert?**

Description

Turns on or off the waveform invert for the specified channel; or queries the on/off status of the waveform invert for the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

When the waveform invert is turned off, the waveform is displayed normally; when the waveform invert is turned on, the voltage values of the displayed waveform are inverted.

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:INVert ON     /*Enables the waveform invert for CH1.*/
:CHANnel1:INVert?       /*The query returns 1.*/
```

3.9.5 :CHANnel<n>:OFFSet

Syntax

:CHANnel<*n*>**:OFFSet** <*offset*>

:CHANnel<*n*>**:OFFSet?**

Description

Sets or queries the vertical offset of the specified channel. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<offset>	Real	Related to vertical scale and input impedance when the input impedance is 1 MΩ: ±1 V (1 mV/div to 50 mV/div) ±30 V (51mV/div to 260mV/div) ±100 V (265 mV/div to 10 V/div) When the input impedance is 50 Ω: ±1 V (1 mV/div to 100 mV/div) ±4 V (102 mV/div to 1 V/div)	0 V

Remarks

N/A

Return Format

The query returns the vertical offset in scientific notation.

Example

```
:CHANnel1:OFFSet 0.01    /*Sets the vertical offset of CH1 to 10
mV.*/
:CHANnel1:OFFSet?        /*The query returns 1E-2.*/
```

3.9.6 :CHANnel<n>:TCALibrate

Syntax

:CHANnel<*n*>**:TCALibrate** <*val*>

:CHANnel<*n*>**:TCALibrate?**

Description

Sets or queries the delay calibration time (used to calibrate the zero offset of the corresponding channel) of the specified channel. The unit is s.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<val>	Real	-100 ns to 100 ns	0 s

Remarks

N/A

Return Format

The query returns the delay calibration time in scientific notation.

Example

```
:CHANnel1:TCALibrate 0.00000002      /*Sets the delay calibration
time to 20 ns.*/
:CHANnel1:TCALibrate?                /*The query returns
2.000000E-8.*/
```

3.9.7 :CHANnel<n>:SCALe

Syntax

:CHANnel<*n*>**:SCALe** <*scale*>

:CHANnel<*n*>**:SCALe?**

Description

Sets or queries the vertical scale of the specified channel. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<scale>	Real	Related to the current input impedance and the probe ratio: When the input impedance is 1 MΩ and the probe ratio is 1X: 1 mV to 10 V When the input impedance is 50 Ω and the probe ratio is 1X: 1 mV to 1 V	100 mV

Remarks

You can use the *:CHANnel<n>:VERNier* command to enable or disable the fine adjustment setting for the vertical scale of the specified channel. By default, the fine adjustment is Off. At this time, you can set the vertical scale at 1-2-5 step, i.g. 10 mV, 20 mV, 50 mV...100 V. When the fine adjustment is enabled, you can further adjust the vertical scale within a relatively smaller range to improve vertical resolution. To better observe the signal details, enable the fine adjustment function to improve the waveform display amplitude if the following conditions occur: the amplitude of the input waveform is a little bit greater than the full scale of the current scale; using the next scale for adjustment makes the amplitude a little bit lower than expected.

Return Format

The query returns the vertical scale in scientific notation.

Example

```
:CHANnel1:SCALe 1   /*Sets the vertical scale of CH1 to 1 V.*/
:CHANnel1:SCALe?   /*The query returns 1.000000E+00.*/
```

3.9.8 :CHANnel<n>:IMPedance

Syntax

:CHANnel<*n*>**:IMPedance** <*impedance*>

:CHANnel<*n*>**:IMPedance?**

Description

Sets or queries the input impedance of the specified analog channel. The default unit is Ω.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<impedance>	Discrete	{OMEG|FIFTy}	OMEG

Description

- **OMEG:** 1MΩ. This indicates that the input impedance of the oscilloscope is rather high, and the current flowing from the circuit under test to the oscilloscope can be ignored.
- **FIFTy:** 50 Ω. This indicates that the oscilloscope shall be matched with a device whose output impedance is 50 Ω.

Return Format

The query returns OMEG or FIFT.

Example

```
:CHANnel1:IMPedance FIFTy   /*Sets the input impedance of CH1 to 50 Ω.*/
:CHANnel1:IMPedance?       /*The query returns FIFT.*/
```

3.9.9 :CHANnel<n>:CSTart

Syntax

:CHANnel<*n*>**:CSTart**

Description

Starts calibration for the active probe currently connected to the specified analog channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Remarks

This command is only valid when an active probe is correctly connected.

Return Format

N/A

Example

```
N/A
```

3.9.10 :CHANnel<n>:PROBe

Syntax

:CHANnel<*n*>**:PROBe** <*atten*>

:CHANnel<*n*>**:PROBe?**

Description

Sets or queries the probe attenuation ratio of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<atten>	Discrete	{0.001|0.002|0.005|0.01|0.02|0.05|0.1|0.2|0.5|1|2|5|10|20|50|100|200|500|1000|2000|5000|10000|20000|50000}	1

Remarks

- Sets the probe ratio. That is, multiply the acquired signal by a specified number (not affect the actual amplitude of the signal).
- The set probe ratio affects the settable range of the current vertical scale.

Return Format

The query returns 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000, 20000, or 50000.

Example

```
:CHANnel1:PROBe 10   /*Sets the probe attenuation ratio of CH1 to
10X.*/
:CHANnel1:PROBe?    /*The query returns 10.*/
```

3.9.11 :CHANnel<n>:PROBe:BIAS

Syntax

`:CHANnel<n>:PROBe:BIAS <bias>`

`:CHANnel<n>:PROBe:BIAS?`

Description

Sets or queries the probe bias voltage for the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bias>	Real	Affected by the probe model.	0 V

Remarks

- This function is used to adjust the signal under test that exceeds the input dynamic range of the probe amplifier to an appropriate range to ensure the signal integrity.
- This command is only valid when an active probe is correctly connected.

Return Format

The query returns the probe bias voltage in scientific notation.

Example

```
N/A
```

3.9.12 :CHANnel<n>:PROBe:CALibration

Syntax

`:CHANnel<n>:PROBe:CALibration`

Description

Performs the calibration for the probe that is connected to the specified channel.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:CHANnel1:PROBe:CALibration /*Performs the calibration for the
probe that is connected to CH1.*/
```

3.9.13 :CHANnel<n>:PROBe:DELay

Syntax

:CHANnel<*n*>**:PROBe:DELay** <*delay*>

:CHANnel<*n*>**:PROBe:DELay?**

Description

Sets or queries the probe delay time of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<delay>	Real	-100 ns to 100 ns	0 s

Remarks

- To avoid measurement result errors arising from the transmission delay of the probe cable, the oscilloscope provides the probe delay adjustment function for the active probe.
- This command is only valid when an active probe is correctly connected.

Return Format

The query returns the probe delay time in scientific notation.

Example

```
:CHANnel1:PROBe:DELay 0.00000001   /*Sets the probe delay time of
CH1 to 10 ns.*/
:CHANnel1:PROBe:DELay?    /*The query returns 1E-8.*/
```

3.9.14 :CHANnel<n>:PROBe:DEMag

Syntax

:CHANnel<*n*>**:PROBe:DEMag**

Description

Starts to demagnetize the current probe that is connected to the specified channel.

Parameter

N/A

Remarks

This command is only valid when a current probe is correctly connected.

Return Format

N/A

Example

```
:CHANnel1:PROBe:DEMag /*Starts to demagnetize the current probe
that is connected to CH1.*/
```

3.9.15 :CHANnel<n>:UNITs

Syntax

:CHANnel<*n*>**:UNITs** <*units*>

:CHANnel<*n*>**:UNITs?**

Description

Sets or queries the amplitude display unit of the specified analog channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<units>	Discrete	{WATT|AMPere|VOLTage|UNKNown}	VOLTage

Remarks

N/A

Return Format

The query returns VOLT, WATT, AMP, or UNKN.

Example

```
:CHANnel1:UNITs VOLTage    /*Sets the amplitude display unit of CH1
to VOLTage.*/
:CHANnel1:UNITs?           /*The query returns VOLT.*/
```

3.9.16 :CHANnel<n>:VERNier

Syntax

:CHANnel<*n*>**:VERNier** <*bool*>

:CHANnel<*n*>**:VERNier?**

Description

Enables or disables the fine adjustment of the vertical scale of the specified channel; or queries the on/off status of the fine adjustment function of the vertical scale of the specified channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

By default, the fine adjustment is Off. At this time, you can set the vertical scale at 1-2-5 step, i.g. 10 mV, 20 mV, 50 mV, 100 mV...100 V. When the fine adjustment is enabled, you can further adjust the vertical scale within a relatively smaller range to improve vertical resolution. To better observe the signal details, enable the fine adjustment function to improve the waveform display amplitude if the following conditions occur: the amplitude of the input waveform is a little bit greater than the full scale of the current scale; using the next scale for adjustment makes the amplitude a little bit lower than expected.

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:VERNier ON    /*Enables the fine adjustment of the
vertical scale of CH1.*/
:CHANnel1:VERNier?       /*The query returns 1.*/
```

3.9.17 :CHANnel<n>:POSition

Syntax

:CHANnel<*n*>**:POSition** <*offset*>

:CHANnel<*n*>**:POSition?**

Description

Sets or queries the offset calibration voltage for calibrating the zero point of the specified analog channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<offset>	Real	1 MΩ: -30 V to 30 V 50 Ω: -4 V to 4 V	0

Remarks

N/A

Return Format

The query returns the offset calibration voltage for calibrating the zero point of the specified analog channel in scientific notation.

Example

```
:CHANnel1:POSition 10    /*Sets the offset calibration voltage for
calibrating the zero point of CH1 to 10 V.*/
:CHANnel1:POSition?       /*The query returns 1.000000E+01.*/
```

3.10 :COUNter Commands

3.10.1 :COUNter:CURRent?

Syntax

`:COUNter:CURRent?`

Description

Queries the measurement value of the frequency counter.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current measurement value of the frequency counter in scientific notation.

Example

```
N/A
```

3.10.2 :COUNter:ENABle

Syntax

:COUNter:ENABle <*bool*>

:COUNter:ENABle?

Description

Enables or disables the frequency counter; or queries the on/off status of the frequency counter.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:COUNter:ENABle ON     /*Enables the frequency counter.*/
:COUNter:ENABle?       /*The query returns 1.*/
```

3.10.3 :COUNter:SOURce

Syntax

:COUNter:SOURce <*source*>

:COUNter:SOURce?

Description

Sets or queries the source of the frequency counter.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|EXT}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or EXT.

Example

```
:COUNter:SOURce CHANnel2      /*Sets the source of the frequency
counter to CHANnel2.*/
:COUNter:SOURce?              /*The query returns CHAN2.*/
```

3.10.4 :COUNter:MODE

Syntax

:COUNter:MODE <*mode*>

:COUNter:MODE?

Description

Sets or queries the mode of the frequency counter.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{FREQuency|PERiod|TOTalize}	FREQuency

Remarks

N/A

Return Format

The query returns FREQ, PER, or TOT.

Example

```
:COUNter:MODE PERiod          /*Sets the mode of the frequency counter
to PERiod.*/
:COUNter:MODE?                /*The query returns PER.*/
```

3.10.5 :COUNter:NDIGits

Syntax

:COUNter:NDIGits <*val*>

:COUNter:NDIGits?

Description

Sets or queries the resolution of the frequency counter.

Parameter

Name	Type	Range	Default
<val>	Integer	3 to 8	5

Remarks

The resolution setting is only available for "Period" and "Frequency", and unavailable for "Totalize".

Return Format

The query returns an integer ranging from 3 to 8.

Example

```
:COUNter:NDIGits 4              /*Sets the resolution of the frequency
counter to 4.*/
:COUNter:NDIGits?               /*The query returns 4.*/
```

3.10.6 :COUNter:TOTalize:ENABle

Syntax

:COUNter:TOTalize:ENABle <*bool*>

:COUNter:TOTalize:ENABle?

Description

Enables or disables the statistical function of the frequency counter; or queries the on/off status of the statistical function of the frequency counter.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

The statistical function is available for "Period" and "Frequency", but unavailable for "Totalize".

Return Format

The query returns 1 or 0.

Example

```
:COUNter:TOTalize:ENABle ON     /*Enables the statistical function
of the frequency counter.*/
:COUNter:TOTalize:ENABle?       /*The query returns 1.*/
```

3.10.7 :COUNter:TOTalize:CLEar

Syntax

`:COUNter:TOTalize:CLEar`

Description

Clears the total count.

Parameter

N/A

Remarks

Available when "Totalize" is selected under "Measure".

Return Format

N/A

Example

```
N/A
```

3.11 :CURSor Commands

The **:CURSor** commands are used to measure the X axis values (e.g. Time) and Y axis values (e.g. Voltage) of the waveform on the screen.

3.11.1 :CURSor:MODE

Syntax

`:CURSor:MODE` <*mode*>

`:CURSor:MODE?`

Description

Sets or queries the mode of the cursor measurement.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{OFF|MANual|TRACk|XY}	OFF

Remarks

- **OFF:** disables the cursor measurement function.

- **MANual:** enables the manual mode of cursor measurement.
- **TRACk:** enables the track mode of cursor measurement.
- **XY:** enables the XY mode of cursor measurement. It is only valid when you select "XY" mode.

Return Format

The query returns OFF, MAN, TRAC, or XY.

Example

```
:CURSor:MODE MANual    /*Selects the manual mode of cursor
measurement.*/
:CURSor:MODE?           /*The query returns MAN.*/
```

3.11.2 :CURSor:MEASure:INDicator

Syntax

:CURSor:MEASure:INDicator <*bool*>

:CURSor:MEASure:INDicator?

Description

Sets or queries the on/off status of the indicator for the measurement function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:CURSor:MEASure:INDicator? ON   /*Sets the indicator for the
measurement function to ON.*/
:CURSor:MEASure:INDicator? /*The query returns 1.*/
```

3.11.3 :CURSor:MANual:TYPE

Syntax

:CURSor:MANual:TYPE <*type*>

:CURSor:MANual:TYPE?

Description

Sets or queries the cursor type in the manual mode of cursor measurement.

Parameter

Name	Type	Range	Default
<type>	Discrete	{TIME|AMPLitude}	TIME

Description

- **TIME:** indicates X cursor, which is often used to measure the time parameters.
- **AMPLitude:** indicates Y cursor, which is often used to measure the voltage parameters.

Return Format

The query returns TIME or AMPL.

Example

```
:CURSor:MANual:TYPE AMPLitude        /*Sets the cursor type to
AMPLitude.*/
:CURSor:MANual:TYPE?                 /*The query returns AMPL.*/
```

3.11.4 :CURSor:MANual:SOURce

Syntax

:CURSor:MANual:SOURce <*source*>

:CURSor:MANual:SOURce?

Description

Sets or queries the channel source of the manual mode of cursor measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|MATH1| MATH2|MATH3|MATH4|NONE}	CHANnel1

Remarks

Only the currently enabled channel can be selected as the channel source.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, MATH4, or NONE.

Example

```
:CURSor:MANual:SOURce CHANnel2    /*Sets the channel source to
CHANnel2.*/
:CURSor:MANual:SOURce?             /*The query returns CHAN2.*/
```

3.11.5 :CURSor:MANual:TUNit

Syntax

:CURSor:MANual:TUNit <*tunit*>

:CURSor:MANual:TUNit?

Description

Sets or queries the horizontal unit in the manual mode of cursor measurement.

Parameter

Name	Type	Range	Default
<tunit>	Discrete	{SECond}	SECond

Remarks

SECond: in the measurement results, AX, BX, and ΔX are expressed in "s"; 1/ΔX in "Hz".

Return Format

The query returns SEC.

Example

```
:CURSor:MANual:TUNit SECond    /*Sets the horizontal unit to
SECond.*/
:CURSor:MANual:TUNit?           /*The query returns SEC.*/
```

3.11.6 :CURSor:MANual:VUNit

Syntax

:CURSor:MANual:VUNit <*vunit*>

:CURSor:MANual:VUNit?

Description

Sets or queries the vertical unit in the manual mode of cursor measurement.

Parameter

Name	Type	Range	Default
<vunit>	Discrete	{SOURce}	SOUR

Remarks

SOURce: in the measurement results, the unit of AY, BY, and ΔY are automatically set to the unit of the current source.

Return Format

The query returns SOUR.

Example

```
:CURSor:MANual:VUNit SOURce /*Sets the unit of AY, BY, and ΔY to be
the unit of the current source.*/
:CURSor:MANual:VUNit?        /*The query returns SOUR.*/
```

3.11.7 :CURSor:MANual:CAX

Syntax

:CURSor:MANual:CAX <*ax*>

:CURSor:MANual:CAX?

Description

Sets or queries the horizontal position of Cursor A in the manual mode of cursor measurement.

Parameter

Name	Type	Range	Default
<ax>	Real	Refer to *Remarks*	-

Remarks

The range of the horizontal position of Cursor A is determined by the current horizontal scale and position.

Return Format

The query returns the horizontal position of Cursor A scientific notation. The unit is s.

Example

```
:CURSor:MANual:CAX 0.00000001    /*Sets the horizontal position of
Cursor A to 10 ns.*/
:CURSor:MANual:CAX?       /*The query returns 1.000000E-8.*/
```

3.11.8 :CURSor:MANual:CAY

Syntax

:CURSor:MANual:CAY <*ay*>

:CURSor:MANual:CAY?

Description

Sets or queries the vertical position of Cursor A in the manual mode of cursor measurement. The default unit is V.

Parameter

Name	Type	Range	Default
<ay>	Real	Refer to *Remarks*	-

Remarks

The range of the vertical position of Cursor A is determined by the current vertical scale and position.

Return Format

The query returns the vertical position of Cursor A in scientific notation.

Example

```
:CURSor:MANual:CAY 0.1     /*Sets the vertical position of Cursor A
to 0.1 V.*/
:CURSor:MANual:CAY?        /*The query returns 1.000000E-1.*/
```

3.11.9 :CURSor:MANual:CBX

Syntax

:CURSor:MANual:CBX <*bx*>

:CURSor:MANual:CBX?

Description

Sets or queries the horizontal position of Cursor B in the manual mode of cursor measurement. The default unit is s.

Parameter

Name	Type	Range	Default
<bx>	Real	Refer to *Remarks*	-

Remarks

The range of the horizontal position of Cursor B is determined by the current horizontal scale and position.

Return Format

The query returns the horizontal position of Cursor B in scientific notation.

Example

```
:CURSor:MANual:CBX 0.00000001    /*Sets the horizontal position of
Cursor B to 10 ns.*/
:CURSor:MANual:CBX?       /*The query returns 1.000000E-8.*/
```

3.11.10 :CURSor:MANual:CBY

Syntax

:CURSor:MANual:CBY <*by*>

:CURSor:MANual:CBY?

Description

Sets or queries the vertical position of Cursor B in the manual mode of cursor measurement. The unit is V.

Parameter

Name	Type	Range	Default
<by>	Real	Refer to *Remarks*	-

Remarks

The range of the vertical position of Cursor B is determined by the current vertical scale and position.

Return Format

The query returns the vertical position of Cursor B in scientific notation.

Example

```
:CURSor:MANual:CBY 0.1    /*Sets the vertical position of Cursor B
to 0.1 V.*/
:CURSor:MANual:CBY?       /*The query returns 1.000000E-1.*/
```

3.11.11 :CURSor:MANual:AXValue?

Syntax

:CURSor:MANual:AXValue?

Description

Queries the X value at Cursor A in the manual mode of cursor measurement. The unit is determined by the horizontal unit selected for the currently corresponding channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.12 :CURSor:MANual:AYValue?

Syntax

```
:CURSor:MANual:AYValue?
```

Description

Queries the Y value at Cursor A in the manual mode of cursor measurement. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

- The returned value is the same as the measurement value in the Cursor interface. Therefore, the unit is related to the vertical unit. When the vertical unit of cursor is set to Source, the unit of the returned value is the same as vertical unit of the channel.
- No value is returned when the cursor measurement value is invalid.

Return Format

The query returns the Y value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.13 :CURSor:MANual:BXValue?

Syntax

```
:CURSor:MANual:BXValue?
```

Description

Queries the X value at Cursor B in the manual mode of cursor measurement. The unit is determined by the currently selected horizontal unit.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor B in scientific notation.

Example

```
N/A
```

3.11.14 :CURSor:MANual:BYValue?

Syntax

```
:CURSor:MANual:BYValue?
```

Description

Queries the Y value at Cursor B in the manual mode of cursor measurement. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

- The returned value is the same as the measurement value in the Cursor interface. Therefore, the unit is related to the vertical unit. When the vertical unit of cursor is set to Source, the unit of the returned value is the same as vertical unit of the channel.
- No value is returned when the cursor measurement value is invalid.

Return Format

The query returns the Y value at Cursor B in scientific notation.

Example

```
N/A
```

3.11.15 :CURSor:MANual:XDELta?

Syntax

```
:CURSor:MANual:XDELta?
```

Description

Queries the difference (▵X) between the X value at Cursor A and the X value at Cursor B in the manual mode of cursor measurement. The unit is determined by the currently selected horizontal unit.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current difference in scientific notation.

Example

```
N/A
```

3.11.16 :CURSor:MANual:IXDelta?

Syntax

```
:CURSor:MANual:IXDelta?
```

Description

Queries the reciprocal (1/▵X) of the absolute difference between the X value at Cursor A and the X value at Cursor B in the manual mode of cursor measurement. The unit is determined by the currently selected horizontal unit.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 1/▵X in scientific notation.

Example

```
N/A
```

3.11.17 :CURSor:MANual:YDELta?

Syntax

:CURSor:MANual:YDELta?

Description

Queries the difference (▵Y) between the Y value at Cursor A and the Y value at Cursor B in the manual mode of cursor measurement. The unit is determined by the currently selected vertical unit.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current difference in scientific notation.

Example

```
N/A
```

3.11.18 :CURSor:TRACk:SOURce1

Syntax

:CURSor:TRACk:SOURce1 <*source*>

:CURSor:TRACk:SOURce1?

Description

Sets or queries the channel source of Cursor A in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4|NONE}	CHANnel1

Remarks

When no channel is enabled, sending this command will enable the corresponding channel.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, MATH4, or NONE.

Example

```
:CURSor:TRACk:SOURce1 CHANnel2    /*Sets the channel source to
CHANnel2.*/
:CURSor:TRACk:SOURce1?             /*The query returns CHAN2.*/
```

3.11.19 :CURSor:TRACk:SOURce2

Syntax

:CURSor:TRACk:SOURce2 <*source*>

:CURSor:TRACk:SOURce2?

Description

Sets or queries the channel source of Cursor B in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4|NONE}	CHANnel1

Remarks

When no channel is enabled, sending this command will enable the corresponding channel.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, MATH4, or NONE.

Example

```
:CURSor:TRACk:SOURce2 CHANnel2    /*Sets the channel source to
CHANnel2.*/
:CURSor:TRACk:SOURce2?             /*The query returns CHAN2.*/
```

3.11.20 :CURSor:TRACk:CAX

Syntax

:CURSor:TRACk:CAX <*ax*>

:CURSor:TRACk:CAX?

Description

Sets or queries the horizontal position of Cursor A in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<ax>	Real	Refer to *Remarks*	-

Remarks

The range of the horizontal position of Cursor A is determined by the current horizontal scale and position.

Return Format

The query returns the horizontal position of Cursor A scientific notation. The unit is s.

Example

```
:CURSor:TRACk:CAX 1.000000E-8    /*Sets the horizontal position of
Cursor A to 10 ns.*/
:CURSor:TRACk:CAX?       /*The query returns 1.000000E-8.*/
```

3.11.21 :CURSor:TRACk:CBX

Syntax

:CURSor:TRACk:CBX <*bx*>

:CURSor:TRACk:CBX?

Description

Sets or queries the horizontal position of Cursor B in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<bx>	Real	Refer to *Remarks*	-

Remarks

The range of the horizontal position of Cursor B is determined by the current horizontal scale and position.

Return Format

The query returns the horizontal position of Cursor B in scientific notation.

Example

```
:CURSor:TRACk:CBX 1.000000E -8    /*Sets the horizontal position of
Cursor B 10 ns.*/
:CURSor:TRACk:CBX?       /*The query returns 1.000000E-8.*/
```

3.11.22 :CURSor:TRACk:CAY

Syntax

:CURSor:TRACk:CAY <*ay*>

:CURSor:TRACk:CAY?

Description

Sets or queries the vertical position of Cursor A in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<ay>	Real	Refer to *Remarks*	-

Remarks

The range of the vertical position of Cursor A is determined by the current vertical scale and position.

Return Format

The query returns the vertical position of Cursor A in scientific notation.

Example

```
:CURSor:TRACk:CAY 0.1    /*Sets the vertical position of Cursor A
to 0.1 V.*/
:CURSor:TRACk:CAY?       /*The query returns 1.000000E-1.*/
```

3.11.23 :CURSor:TRACk:CBY

Syntax

:CURSor:TRACk:CBY <*by*>

:CURSor:TRACk:CBY?

Description

Sets or queries the vertical position of Cursor B in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<by>	Real	Refer to *Remarks*	-

Remarks

The range of the vertical position of Cursor B is determined by the current vertical scale and position.

Return Format

The query returns the vertical position of Cursor B in scientific notation.

Example

```
:CURSor:TRACk:CBY 0.1    /*Sets the vertical position of Cursor B
to 0.1 V.*/
:CURSor:TRACk:CBY?       /*The query returns 1.000000E-1.*/
```

3.11.24 :CURSor:TRACk:AXValue?

Syntax

:CURSor:TRACk:AXValue?

Description

Queries the X value at Cursor A in the track mode of cursor measurement. The unit is determined by the amplitude unit selected for the currently corresponding channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.25 :CURSor:TRACk:AYValue?

Syntax

```
:CURSor:TRACk:AYValue?
```

Description

Queries the Y value at Cursor A in the track mode of cursor measurement. The unit is the same as that selected for the current channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the Y value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.26 :CURSor:TRACk:BXValue?

Syntax

```
:CURSor:TRACk:BXValue?
```

Description

Queries the X value at Cursor B in the track mode of cursor measurement. The unit is determined by the amplitude unit selected for the currently corresponding channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor B in scientific notation.

Example

```
N/A
```

3.11.27 :CURSor:TRACk:BYValue?

Syntax

:CURSor:TRACk:BYValue?

Description

Queries the Y value at Cursor B in the track mode of cursor measurement. The unit is the same as that selected for the current channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the Y value at Cursor B in scientific notation.

Example

```
N/A
```

3.11.28 :CURSor:TRACk:XDELta?

Syntax

:CURSor:TRACk:XDELta?

Description

Queries the difference (▵X) between the X value at Cursor A and the X value at Cursor B in the track mode of cursor measurement.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current difference in scientific notation.

Example

```
N/A
```

3.11.29 :CURSor:TRACk:YDELta?

Syntax

:CURSor:TRACk:YDELta?

Description

Queries the difference (▵Y) between the Y value at Cursor A and the Y value at Cursor B in the track mode of cursor measurement. The unit is the same as that selected for the current channel.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current difference in scientific notation.

Example

```
N/A
```

3.11.30 :CURSor:TRACk:IXDelta?

Syntax

:CURSor:TRACk:IXDelta?

Description

Queries the reciprocal (1/▵X) of the absolute difference between the X value at Cursor A and the X value at Cursor B in the track mode of cursor measurement. The default unit is Hz.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 1/▵X in scientific notation.

Example

```
N/A
```

3.11.31 :CURSor:XY:AX

Syntax

:CURSor:XY:AX <*X*>

:CURSor:XY:AX?

Description

Sets or queries the horizontal position of Cursor A in the XY cursor measurement mode. The unit is V.

Parameter

Name	Type	Range	Default
<x>	Real	Related to the vertical scale and vertical offset of the oscilloscope	-

Remarks

N/A

Return Format

The query returns the horizontal position of Cursor A in scientific notation.

Example

```
:CURSor:XY:AX 0.1     /*Sets the horizontal position of Cursor A to
100 mV.*/
:CURSor:XY:AX?        /*The query returns 1.000000E-1.*/
```

3.11.32 :CURSor:XY:BX

Syntax

:CURSor:XY:BX <*X*>

:CURSor:XY:BX?

Description

Sets or queries the horizontal position of Cursor B in the XY cursor measurement mode. The unit is V.

Parameter

Name	Type	Range	Default
<x>	Real	Related to the vertical scale and vertical offset of the oscilloscope	-

Remarks

N/A

Return Format

The query returns the horizontal position of Cursor B in scientific notation.

Example

```
:CURSor:XY:BX 0.1     /*Sets the horizontal position of Cursor B to
100 mV.*/
:CURSor:XY:BX?        /*The query returns 1.000000E-1.*/
```

3.11.33 :CURSor:XY:AY

Syntax

:CURSor:XY:AY <*y*>

:CURSor:XY:AY?

Description

Sets or queries the vertical position of Cursor A in the XY cursor measurement mode. The unit is V.

Parameter

Name	Type	Range	Default
<y>	Real	Related to the vertical scale and vertical offset of the oscilloscope	-

Remarks

N/A

Return Format

The query returns the vertical position of Cursor A in scientific notation.

Example

```
:CURSor:XY:AY 0.1     /*Sets the vertical position of Cursor A to
100 mV.*/
:CURSor:XY:AY?        /*The query returns 1.000000E-1.*/
```

3.11.34 :CURSor:XY:BY

Syntax

:CURSor:XY:BY <*y*>

:CURSor:XY:BY?

Description

Sets or queries the vertical position of Cursor B in the XY cursor measurement mode. The unit is V.

Parameter

Name	Type	Range	Default
<y>	Real	Related to the vertical scale and vertical offset of the oscilloscope	-

Remarks

N/A

Return Format

The query returns the vertical position of Cursor B in scientific notation.

Example

```
:CURSor:XY:BY 0.1     /*Sets the vertical position of Cursor B to
100 mV.*/
:CURSor:XY:BY?        /*The query returns 1.000000E-1.*/
```

3.11.35 :CURSor:XY:AXValue?

Syntax

:CURSor:XY:AXValue?

Description

Queries the X value at Cursor A in the XY cursor measurement mode.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.36 :CURSor:XY:AYValue?

Syntax

```
:CURSor:XY:AYValue?
```

Description

Queries the X value at Cursor A in the XY cursor measurement mode.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the Y value at Cursor A in scientific notation.

Example

```
N/A
```

3.11.37 :CURSor:XY:BXValue?

Syntax

```
:CURSor:XY:BXValue?
```

Description

Queries the X value at Cursor B in the XY cursor measurement mode.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the X value at Cursor B in scientific notation.

Example

```
N/A
```

3.11.38 :CURSor:XY:BYValue?

Syntax

```
:CURSor:XY:BYValue?
```

Description

Queries the Y value at Cursor B in the XY cursor measurement mode.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the Y value at Cursor B in scientific notation.

Example

```
N/A
```

3.12 :DISPlay Commands

The **:DISPlay** commands can be used to set the displayed type of the waveform, persistence time, intensity, grid type, grid brightness, etc.

3.12.1 :DISPlay:CLEar

Syntax

```
:DISPlay:CLEar
```

Description

Clears all the waveforms on the screen.

Parameter

N/A

Remarks

- If the oscilloscope is in the "RUN" state, new waveforms will continue being displayed after being cleared.
- This command functions the same as tapping the **Clear** icon on the small screen. You can also send the *:CLEar* command to clear all the waveforms on the screen.

Return Format

N/A

Example

```
N/A
```

3.12.2 :DISPlay:TYPE

Syntax

:DISPlay:TYPE <*type*>

:DISPlay:TYPE?

Description

Sets or queries the display type of the waveforms on the screen.

Parameter

Name	Type	Range	Default
<type>	Discrete	{VECTors}	VECTors

Remarks

VECTors: The sample points are connected by lines and displayed. Normally, this mode can provide the most vivid waveform to view the steep edge of the waveform (such as square waveforms).

Return Format

The query returns VECT.

Example

```
:DISPlay:TYPE VECTors    /*Sets the display type to VECTors.*/
:DISPlay:TYPE?          /*The query returns VECT.*/
```

3.12.3 :DISPlay:GRADing:TIME

Syntax

:DISPlay:GRADing:TIME <*time*>

:DISPlay:GRADing:TIME?

Description

Sets or queries the persistence time. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Discrete	{MIN|0.1|0.2|0.5|1|2|5|10|INFinite}	MIN

Remarks

- **MIN:** sets the persistence time to its minimum value to view how the waveform changes at a high refresh rate.
- **specified value (e.g. 0.1, 0.2, 0.5, 1, 2, 5, 10):** sets the persistence time to any of the above specific value to observe glitch that changes relatively slowly or glitch with low occurrence probability.
- **INFinite:** In this mode, the oscilloscope displays the waveform newly acquired without clearing the waveforms acquired formerly. It can be used to measure noise and jitter and to capture incidental events.

Return Format

The query returns MIN, 0.1, 0.2, 0.5, 1, 2, 5, 10, or INF.

Example

```
:DISPlay:GRADing:TIME 0.1   /*Sets the persistence time to 100 ms.*/
:DISPlay:GRADing:TIME?     /*The query returns 0.1.*/
```

3.12.4 :DISPlay:WBRightness

Syntax

:DISPlay:WBRightness <*brightness*>

:DISPlay:WBRightness?

Description

Sets or queries the brightness of the waveform on the screen, expressed in percentage.

Parameter

Name	Type	Range	Default
<brightness>	Integer	1 to 100	50

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 100.

Example

```
:DISPlay:WBRightness 50     /*Sets the waveform brightness to 50%.*/
:DISPlay:WBRightness?       /*The query returns 50.*/
```

3.12.5 :DISPlay:GRID

Syntax

:DISPlay:GRID <*grid*>

:DISPlay:GRID?

Description

Sets or queries the display type of the screen grid.

Parameter

Name	Type	Range	Default
<grid>	Discrete	{FULL|HALF|NONE}	FULL

Remarks

- **FULL:** turns on the background grid and coordinates.
- **HALF:** turns the background grid off and turns the coordinate on.
- **NONE:** turns the background grid and coordinate off.

Return Format

The query returns FULL, HALF, or NONE.

Example

```
:DISPlay:GRID NONE    /*Turns off the background grid and
coordinates.*/
:DISPlay:GRID?         /*The query returns NONE.*/
```

3.12.6 :DISPlay:GBRightness

Syntax

:DISPlay:GBRightness <*brightness*>

:DISPlay:GBRightness?

Description

Sets or queries the brightness of the screen grid, expressed in percentage.

Parameter

Name	Type	Range	Default
<brightness>	Integer	0 to 100	50

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to 100.

Example

```
:DISPlay:GBRightness 60    /*Sets the screen grid brightness to
60%.*/
:DISPlay:GBRightness?      /*The query returns 60.*/
```

3.12.7 :DISPlay:DATA?

Syntax

:DISPlay:DATA?

Description

Queries the bitmap data stream of the currently displayed image.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the binary data stream of the screenshot in ".png" format.

Example

```
N/A
```

3.12.8 :DISPlay:RULers

Syntax

:DISPlay:RULers <*bool*>

:DISPlay:RULers?

Description

Enables or disables the ruler display; or queries the on/off status of the ruler.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:DISPlay:RULers ON    /*Enables the display of the ruler.*/
:DISPlay:RULers?       /*The query returns 1.*/
```

3.12.9 :DISPlay:COLor

Syntax

:DISPlay:COLor <*bool*>

:DISPlay:COLor?

Description

Enables or disables the color grade display; or queries the on/off status of the color grade display.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:DISPlay:COLor ON    /*Enables the color grade display.*/
:DISPlay:COLor?       /*The query returns 1.*/
```

3.13 :DVM Commands

3.13.1 :DVM:CURRent?

Syntax

`:DVM:CURRent?`

Description

Queries the current voltage value under test.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.13.2 :DVM:ENABle

Syntax

`:DVM:ENABle` <*bool*>

`:DVM:ENABle?`

Description

Enables or disables the digital voltmeter; or queries the on/off status of the digital voltmeter.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:DVM:ENABle ON     /*Enables the digital voltmeter.*/
:DVM:ENABle?       /*The query returns 1.*/
```

3.13.3 :DVM:SOURce

Syntax

:DVM:SOURce <*source*>

:DVM:SOURce?

Description

Sets or queries the source of the digital voltmeter.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:DVM:SOURce CHANnel1     /*Sets the source of DVM to CHANnel1.*/
:DVM:SOURce?             /*The query returns CHAN1.*/
```

3.13.4 :DVM:MODE

Syntax

:DVM:MODE <*mode*>

:DVM:MODE?

Description

Sets or queries the mode of the digital voltmeter.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{ACRMs|DC|DCRMs}	ACRMs

Remarks

- **ACRMs:** displays the root-mean-square value of the acquired data, with the DC component removed.
- **DC:** displays the root- average-square value of the acquired data.
- **DCRMs:** displays the root-mean-square value of the acquired data.

Return Format

The query returns ACRM, DC, or DCRM.

Example

```
:DVM:MODE DC          /*Sets the mode of the digital voltmeter to
DC.*/
:DVM:MODE?            /*The query returns DC.*/
```

3.14 :EYE Commands

3.14.1 :EYE:ENABle

Syntax

:EYE:ENABle <*bool*>

:EYE:ENABle?

Description

Enables or disables the eye analysis function; or queries the on/off status of the eye analysis function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:EYE:ENABle ON     /*Enables the eye analysis function.*/
:EYE:ENABle?       /*The query returns 1.*/
```

3.14.2 :EYE:SOURce

Syntax

:EYE:SOURce <*source*>

:EYE:SOURce?

Description

Sets or queries the source of the eye diagram.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:EYE:SOURce CHANnel3     /*Sets the source of eye diagram to
CHANnel3.*/
:EYE:SOURce?             /*The query returns CHAN3.*/
```

3.14.3 :EYE:MEASure:ENABle

Syntax

:EYE:MEASure:ENABle <*bool*>

:EYE:MEASure:ENABle?

Description

Enables or disables the eye measurement; or queries the on/off status of the eye measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:EYE:MEASure:ENABle ON     /*Enables the eye measurement.*/
:EYE:MEASure:ENABle?       /*The query returns 1.*/
```

3.14.4 :EYE:MEASure:ITEM

Syntax

:EYE:MEASure:ITEM? <*item*>

Description

Queries the measurements of an eye diagram.

Parameter

Name	Type	Range	Default
<item>	Discrete	{ONE|ZERO|WIDTh|HEIGth|AMP|CROSs|QFACtor}	-

Remarks

- **ONE:** indicates "1" level.

- **ZERO:** indicates "0" level.
- **WIDTh:** indicates the width of an eye diagram.
- **HEIGth:** indicates the height of an eye diagram.
- **AMP:** indicates the amplitude of an eye diagram.
- **CROSs:** indicates the crossing percentage of an eye diagram.
- **QFACtor:** indicates the Q factor.

Return Format

The query returns the measurements of an eye diagram in scientific notation.

Example

```
:EYE:MEASure:ITEM? AMP /*Queries the amplitude of an eye diagram
and returns 1.004000E0.*/
```

3.14.5 :EYE:OVERlap

Syntax

:EYE:OVERlap <*bool*>

:EYE:OVERlap?

Description

Enables or disables the overlap display of the eye diagram.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:EYE:OVERlap ON /*Enables the overlap of the eye diagram.*/
:EYE:OVERlap? /*The query returns 1.*/
```

3.14.6 :EYE:CLOCk:METHod

Syntax

:EYE:CLOCk:METHod <*type*>

:EYE:CLOCk:METHod?

Description

Sets or queries the clock recovery method of the eye diagram.

Parameter

Name	Type	Range	Default
<type>	Discrete	{CONStant|PLL|EXPLicit}	CONStant

Remarks

- **CONStant:** indicates the constant clock recovery.
- **PLL:** indicates the second-order PLL.
- **EXPLicit:** indicates the external clock recovery.

Return Format

The query returns CONS, PLL, or EXPL.

Example

```
:EYE:CLOCk:METHod EXPLicit   /*Sets the clock recovery method to
EXPLicit.*/
:EYE:CLOCk:METHod?      /*The query returns EXPL.*/
```

3.14.7 :EYE:CLOCk:TYPE

Syntax

:EYE:CLOCk:TYPE <*type*>

:EYE:CLOCk:TYPE?

Description

Sets or queries the type of the constant clock recovery method for the eye diagram.

Parameter

Name	Type	Range	Default
<type>	Discrete	{AUT|SEM|MAN}	AUT

Remarks

- **AUT:** recovers the clock based on the narrowest pulse of the signal.
- **SEM:** semi-auto. Recovers the clock by the manually preset data rate and the signal edge.
- **MAN:** manual mode. Recovers the clock by the manually input data rate.

Return Format

The query returns AUT, SEM, or MAN.

Example

```
:EYE:CLOCk:TYPE MAN /*Sets the type of the constant clock recovery
method for the eye diagram to MAN.*/
:EYE:CLOCk:TYPE? /*The query returns MAN.*/
```

3.14.8 :EYE:CLOCk:RATE

Syntax

:EYE:CLOCk:RATE <*val*>

:EYE:CLOCk:RATE?

Description

Sets or queries the manually set data rate for the eye diagram.

Parameter

Name	Type	Range	Default
<val>	Real	100 kH to 10 GHz	300 MHz

Remarks

N/A

Return Format

The query returns the manually set data rate for the eye diagram in scientific notation. The unit is Hz.

Example

```
:EYE:CLOCk:RATE 200000 /*Sets the manually set data rate for the
eye diagram to 200 kHz.*/
:EYE:CLOCk:RATE? /*The query returns 2.000000E+5.*/
```

3.14.9 :EYE:CLOCk:PLL:ORDer

Syntax

:EYE:CLOCk:PLL:ORDer <*ord*>

:EYE:CLOCk:PLL:ORDer?

Description

Sets or queries the PLL order of the eye diagram.

Parameter

Name	Type	Range	Default
<ord>	Discrete	{ONE|TWO}	ONE

Remarks

N/A

Return Format

The query returns ONE or TWO.

Example

```
:EYE:CLOCk:PLL:ORDer TWO        /*Sets the PLL order to TWO.*/
:EYE:CLOCk:PLL:ORDer?           /*The query returns TWO.*/
```

3.14.10 :EYE:CLOCk:PLL:BW

Syntax

:EYE:CLOCk:PLL:BW <*val*>

:EYE:CLOCk:PLL:BW?

Description

Sets or queries the loop bandwidth for the PLL clock recovery method of the eye diagram. The unit is Hz.

Parameter

Name	Type	Range	Default
<val>	Integer	Refer to *Remarks*	-

Remarks

The range of the PLL bandwidth of the eye diagram is determined by the value of the signal data rate. You can run the *:EYE:CLOCk:RATE* command to query or configure the data rate.

Return Format

The query returns the PLL bandwidth in scientific notation.

Example

```
:EYE:CLOCk:PLL:BW 200 /*Sets the PLL bandwidth of the eye diagram
to 200 Hz.*/
:EYE:CLOCk:PLL:BW? /*The query returns 2.000000E+2.*/
```

3.14.11 :EYE:CLOCk:PLL:DAMP

Syntax

:EYE:CLOCk:PLL:DAMP <*val*>

:EYE:CLOCk:PLL:DAMP?

Description

Sets or queries the PLL damp factor for the PLL clock recovery method of the eye diagram. The unit is U.

Parameter

Name	Type	Range	Default
<val>	Real	0 U to 1 U	0.707 U

Remarks

N/A

Return Format

The query returns a real number ranging from 0 to 1.

Example

```
:EYE:CLOCk:PLL:DAMP 0.5 /*Sets the PLL damp factor of the eye
diagram to 500 mU.*/
:EYE:CLOCk:PLL:DAMP? /*The query returns 5.000000E-1.*/
```

3.14.12 :EYE:CLOCk:EXTChan

Syntax

:EYE:CLOCk:EXTChan <*ch*>

:EYE:CLOCk:EXTChan?

Description

Sets or queries the external clock channel of the eye diagram.

Parameter

Name	Type	Range	Default
<ch>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel3

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:EYE:CLOCk:EXTChan CHANnel1   /*Sets the external clock channel of the eye diagram to CHANnel1.*/
:EYE:CLOCk:EXTChan?           /*The query returns CHAN1.*/
```

3.15 IEEE488.2 Common Commands

3.15.1 *IDN?

Syntax

***IDN?**

Description

Queries the ID string of the instrument.

Parameter

N/A

Remarks

N/A

Return Format

The query returns RIGOL TECHNOLOGIES,<model>,<serial number>,<software version>.

- **<model>:** indicates the model number of the instrument.
- **<serial number>:** indicates the serial number of the instrument.

- **<software version>:** indicates the software version of the instrument.

Example

```
N/A
```

3.15.2 *RST

Syntax

***RST**

Description

Restores the instrument to its factory default settings.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.15.3 *CLS

Syntax

***CLS**

Description

Clears all the event registers, and also clears the error queue.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.15.4 *ESE

Syntax

***ESE** <*maskargument*>

***ESE?**

Description

Sets or queries the enable register bit of the standard event register set.

Parameter

Name	Type	Range	Default
<maskargument>	Integer	0 to 255	0

Remarks

N/A

Return Format

The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register.

Example

```
*ESE 16    /*Enables Bit 4 (16 in decimal) in the register.*/
*ESE?      /*The query returns the enable value of the register
16.*/
```

3.15.5 *ESR?

Syntax

***ESR?**

Description

Queries and clears the event register of the standard event status register.

Parameter

N/A

Remarks

Bit 1 and Bit 6 in the standard event status register are not used and are always treated as 0; therefore, the range of the returned value is a decimal number corresponding to a binary number X0XXXX0X (X is 1 or 0).

Return Format

The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register.

Example

```
N/A
```

3.15.6 *OPC

Syntax

***OPC**

***OPC?**

Description

The *OPC command sets bit 0 (Operation Complete, OPC) in the standard event status register to 1 after the current operation is finished. The OPC? command queries whether the current operation is finished.

Parameter

N/A

Remarks

N/A

Return Format

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

Example

```
N/A
```

3.15.7 *RCL

Syntax

***RCL**

Description

Recalls the instrument state in the specified location.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.15.8 *SAV

Syntax

***SAV** <*value*>

Description

Saves the current instrument state to the selected register.

Parameter

Name	Type	Range	Default
<value>	Integer	0 to 49	0

Remarks

N/A

Return Format

N/A

Example

```
*SAV 1          /*Saves the current instrument state to Register 1.*/
```

3.15.9 *SRE

Syntax

***SRE** <*maskargument*>

***SRE**

Description

Sets or queries the enable register of the status byte register set.

Parameter

Name	Type	Range	Default
<maskargument>	Integer	0 to 255	0

Remarks

This command queries the standard event enable register value. Bit 3 and Bit 6 in the standard event status register are not used and are always treated as 0. Therefore, the range of <maskargument> is a decimal number corresponding to a binary number X0XX0XXX (X is 1 or 0).

Return Format

The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register.

Example

```
*SRE 16    /*Enables Bit 4 (16 in decimal) in the register.*/
*SRE?     /*The query returns the enable value of the register 16.*/
```

3.15.10 *STB?

Syntax

***STB?**

Description

Queries the event register for the status byte register. After executing the command, the value in the status byte register is cleared.

Parameter

N/A

Remarks

Bit 0 and Bit 1 in the status byte register are not used and are always treated as 0; therefore, the range of the returned value is a decimal number corresponding to a binary number X0XXXX0X (X is 1 or 0).

Return Format

The query returns an integer. The integer equals to the binary-weighted sum of all the bits set in the register.

Example

```
N/A
```

3.15.11 *WAI

Syntax

***WAI**

Description

Waits for all the pending operations to complete before executing any additional commands.

Parameter

N/A

Remarks

This operation command does not have any functions, only to be compatible with other devices.

Return Format

N/A

Example

```
N/A
```

3.15.12 *TST?

Syntax

`*TST?`

Description

Performs a self-test and queries the self-test result.

Parameter

N/A

Remarks

N/A

Return Format

The query returns a decimal integer.

Example

```
N/A
```

3.16 :JITTer Commands

3.16.1 :JITTer:ENABle

Syntax

`:JITTer:ENABle` <*bool*>

:JITTer:ENABle?

Description

Enables or disables the jitter analysis function; or queries the on/off status of the jitter analysis function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:JITTer:ENABle ON     /*Enables the jitter analysis function.*/
:JITTer:ENABle?       /*The query returns 1.*/
```

3.16.2 :JITTer:SOURce

Syntax

:JITTer:SOURce <*source*>

:JITTer:SOURce?

Description

Sets or queries the source of the jitter.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:JITTer:SOURce CHANnel3    /*Sets the source of the jitter to
CHANnel3.*/
:JITTer:SOURce?            /*The query returns CHAN3.*/
```

3.16.3 :JITTer:SETUp:HIGH

Syntax

:JITTer:SETUp:HIGH <*val*>

:JITTer:SETUp:HIGH?

Description

Sets or queries the high threshold of the jitter, expressed in %.

Parameter

Name	Type	Range	Default
<val>	Integer	Middle threshold of the jitter to 100	90

Remarks

you can run the *:JITTer:SETUp:MID* command to configure or query the middle threshold of the jitter.

Return Format

The query returns an integer ranging from the middle threshold of the jitter to 100.

Example

```
:JITTer:SETUp:HIGH 80    /*Sets the high threshold of the jitter to
80%*/
:JITTer:SETUp:HIGH?       /*The query returns 80.*/
```

3.16.4 :JITTer:SETUp:MID

Syntax

:JITTer:SETUp:MID <*val*>

:JITTer:SETUp:MID?

Description

Sets or queries the middle threshold of the jitter, expressed in %.

Parameter

Name	Type	Range	Default
<val>	Integer	Low threshold to high threshold of jitter	50

Remarks

The setting range of the middle threshold is between the high threshold and the low threshold.

You can run the *:JITTer:SETUp:HIGH* command to configure or query the high threshold of the jitter; run the *:JITTer:SETUp:LOW* command to configure or query the low threshold of the jitter.

Return Format

The query returns an integer ranging from low threshold to high threshold of the jitter.

Example

```
:JITTer:SETUp:MID 60    /*Sets the middle threshold of the jitter
to 60%.*/
:JITTer:SETUp:MID?       /*The query returns 60.*/
```

3.16.5 :JITTer:SETUp:LOW

Syntax

:JITTer:SETUp:LOW <*val*>

:JITTer:SETUp:LOW?

Description

Sets or queries the low threshold of the jitter, expressed in %.

Parameter

Name	Type	Range	Default
<val>	Integer	0 to the middle threshold of the jitter	10

Remarks

you can run the *:JITTer:SETUp:MID* command to configure or query the middle threshold of the jitter.

Return Format

The query returns an integer ranging from 0 to the middle threshold of the jitter.

Example

```
:JITTer:SETUp:LOW 40    /*Sets the low threshold of the jitter to
40%.*/
:JITTer:SETUp:LOW?       /*The query returns 40.*/
```

3.16.6 :JITTer:HISTogram:APPLy

Syntax

:JITTer:HISTogram:APPLy <*bool*>

:JITTer:HISTogram:APPLy?

Description

Enables or disables the jitter histogram; or queries the on/off status of the jitter histogram.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command is only valid when the jitter analysis function is enabled

(*:JITTer:ENABle*).

Return Format

The query returns 1 or 0.

Example

```
:JITTer:HISTogram:APPLy ON    /*Enables the jitter histogram.*/
:JITTer:HISTogram:APPLy?      /*The query returns 1.*/
```

3.16.7 :JITTer:SPECtrum:APPLy

Syntax

:JITTer:SPECtrum:APPLy <*bool*>

:JITTer:SPECtrum:APPLy?

Description

Enables or disables the spectrum graph of the jitter.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command is only valid when the jitter analysis function is enabled (*:JITTer:ENABle*).

Return Format

The query returns 1 or 0.

Example

```
:JITTer:SPECtrum:APPLy ON     /*Enables the spectrum graph of the
jitter.*/
:JITTer:SPECtrum:APPLy?       /*The query returns 1.*/
```

3.16.8 :JITTer:TRENd:APPLy

Syntax

:JITTer:TRENd:APPLy <*bool*>

:JITTer:TRENd:APPLy?

Description

Enables or disables the jitter trend graph; or queries the on/off status of the jitter trend graph.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command is only valid when the jitter analysis function is enabled (*:JITTer:ENABle*).

Return Format

The query returns 1 or 0.

Example

```
:JITTer:TRENd:APPLy ON     /*Enables the jitter trend graph.*/
:JITTer:TRENd:APPLy?       /*The query returns 1.*/
```

3.16.9 :JITTer:MEASure:ENABle

Syntax

:JITTer:MEASure:ENABle <*bool*>

:JITTer:MEASure:ENABle?

Description

Enables or disables the jitter measurement; or queries the on/off status of the jitter measurement.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:JITTer:MEASure:ENABle ON     /*Enables the jitter measurement.*/
:JITTer:MEASure:ENABle?       /*The query returns 1.*/
```

3.16.10 :JITTer:CLOCk:METHod

Syntax

:JITTer:CLOCk:METHod <*type*>

:JITTer:CLOCk:METHod?

Description

Sets or queries the clock recovery method of the jitter measurement.

Parameter

Name	Type	Range	Default
<type>	Discrete	{CONStant|PLL|EXPLicit}	CONStant

Remarks

- **CONStant:** indicates Constant clock recovery.

- **PLL:** indicates the PLL clock recovery.
- **EXPLicit:** indicates the external clock recovery.

Return Format

The query returns CONS, PLL, or EXPL.

Example

```
:JITTer:CLOCk:METHod EXPLicit   /*Sets the clock recovery method to
EXPLicit.*/
:JITTer:CLOCk:METHod?      /*The query returns EXPL.*/
```

3.16.11 :JITTer:CLOCk:TYPE

Syntax

:JITTer:CLOCk:TYPE <*type*>

:JITTer:CLOCk:TYPE?

Description

Sets or queries the type of the constant clock recovery method.

Parameter

Name	Type	Range	Default
<type>	Discrete	{AUT|SEM|MAN}	AUT

Remarks

- **AUT:** recovers the clock based on the narrowest pulse of the signal.
- **SEM:** semi-auto. Recovers the clock by the manually preset data rate and the signal edge.
- **MAN:** manual mode. Recovers the clock by the manually input data rate.

Return Format

The query returns AUT, SEM, or MAN.

Example

```
:JITTer:CLOCk:TYPE MAN /*Sets the type for the Constant recovery to
MAN.*/
:JITTer:CLOCk:TYPE?   /*The query returns MAN.*/
```

3.16.12 :JITTer:CLOCk:RATE

Syntax

:JITTer:CLOCk:RATE <*val*>

:JITTer:CLOCk:RATE?

Description

Sets or queries the manually set data rate of the jitter. The unit is Hz.

Parameter

Name	Type	Range	Default
<val>	Real	100 kHz to 10 GHz	50 MHz

Remarks

N/A

Return Format

The returns the data rate in scientific notation.

Example

```
:JITTer:CLOCk:RATE 100000   /*Sets the data rate to 100 kHz.*/
:JITTer:CLOCk:RATE?           /*The query returns 1.000000E+5.*/
```

3.16.13 :JITTer:CLOCk:PLL:ORDer

Syntax

:JITTer:CLOCk:PLL:ORDer <*ord*>

:JITTer:CLOCk:PLL:ORDer?

Description

Sets or queries the PLL order of the jitter.

Parameter

Name	Type	Range	Default
<ord>	Discrete	{ONE|TWO}	ONE

Remarks

N/A

Return Format

The query returns ONE or TWO.

Example

```
:JITTer:CLOCk:PLL:ORDer TWO      /*Sets the PLL order to TWO.*/
:JITTer:CLOCk:PLL:ORDer?         /*The query returns TWO.*/
```

3.16.14 :JITTer:CLOCk:PLL:BW

Syntax

:JITTer:CLOCk:PLL:BW <*val*>

:JITTer:CLOCk:PLL:BW?

Description

Sets or queries the loop bandwidth for the PLL clock recovery method of the jitter measurement. The unit is Hz.

Parameter

Name	Type	Range	Default
<val>	Real	Refer to *Remarks*	-

Remarks

The range of the PLL bandwidth of the jitter is determined by the value of the signal data rate. You can run the *:JITTer:CLOCk:RATE* command to configure or query the data rate.

Return Format

The query returns the PLL bandwidth in scientific notation.

Example

```
:JITTer:CLOCk:PLL:BW 100   /Sets the PLL bandwidth to 100 Hz.*/
:JITTer:CLOCk:PLL:BW?           /*The query returns 1.000000E+2.*/
```

3.16.15 :JITTer:CLOCk:PLL:DAMP

Syntax

:JITTer:CLOCk:PLL:DAMP <*val*>

:JITTer:CLOCk:PLL:DAMP?

Description

Sets or queries the PLL damp factor for the PLL clock recovery method of the jitter measurement. The unit is U.

Parameter

Name	Type	Range	Default
<val>	Real	0 U to 1 U	0.707 U

Remarks

N/A

Return Format

The query returns a real number ranging from 0 to 1.

Example

```
:JITTer:CLOCk:PLL:DAMP 0.5   /*Sets the damp factor to 0.5 U.*/
:JITTer:CLOCk:PLL:DAMP?           /*The query returns 5.000000E-1.*/
```

3.16.16 :JITTer:CLOCk:EXTChan

Syntax

:JITTer:CLOCk:EXTChan <*ch*>

:JITTer:CLOCk:EXTChan?

Description

Sets the external clock channel of the jitter measurement.

Parameter

Name	Type	Range	Default
<ch>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:JITTer:CLOCk:EXTChan CHANnel3   /*Sets the external clock channel
of the jitter measurement to CHANnel3.*/
:JITTer:CLOCk:EXTChan?           /*The query returns CHAN3.*/
```

3.16.17 :JITTer:RESult:RESst

Syntax

`:JITTer:RESult:RESst`

Description

Clears the jitter results.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:JITTer:RESult:RESst
```

3.16.18 :JITTer:RESult?

Syntax

`:JITTer:RESult?`

Description

Queries the jitter measurement result.

Parameter

N/A

Remarks

Only when the jitter measurement function is enabled, can you query the measurement results. You can run the *:JITTer:MEASure:ENABle* command to enable the jitter measurement.

Return Format

The query returns the jitter measurement results in strings.

Example

The query returns the jitter measurement results in the following formats.

```
Tj:152.5ps
Rj:9.461ps
Dj:19.43ps
```

```
Pj:11.52ps
DDj:7.909ps
DCD:7.909ps
ISI:0s
BR:200Mbit/s
```

3.17 :LAN Commands

The **:LAN** commands are used to set and query the LAN parameters.

3.17.1 :LAN:DHCP

Syntax

:LAN:DHCP <*bool*>

:LAN:DHCP?

Description

Turns on or off the DHCP configuration mode; or queries the on/off status of the current DHCP configuration mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

- When the three IP configuration types (DHCP, Auto IP, and Static IP) are all turned on, the priority of the parameter configuration from high to low is "DHCP", "Auto IP", and "Static IP". The three IP configuration types cannot be all turned off at the same time.
- When DHPC is valid, the DHCP server in the current network will assign the network parameters (such as the IP address) for the oscilloscope.
- After the *:LAN:APPLy* command is executed, the configuration type can take effect immediately.

Return Format

The query returns 1 or 0.

Example

```
:LAN:DHCP OFF       /*Disables DHCP configuration mode.*/
:LAN:DHCP?          /*The query returns 0.*/
```

3.17.2 :LAN:AUToip

Syntax

:LAN:AUToip <*bool*>

:LAN:AUToip?

Description

Turns on or off the Auto IP configuration mode; or queries the on/off status of the current Auto IP configuration mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

When the auto IP mode is valid, disable DHCP manually. You can self-define the gateway and DNS address for the oscilloscope.

Return Format

The query returns 1 or 0.

Example

```
:LAN:AUToip OFF          /*Disables the Auto IP configuration
mode.*/
:LAN:AUToip?             /*The query returns 0.*/
```

3.17.3 :LAN:GATeway

Syntax

:LAN:GATeway <*string*>

:LAN:GATeway?

Description

Sets or queries the default gateway.

Parameter

Name	Type	Range	Default
<string>	ASCII String	Refer to *Remarks*	-

Remarks

- The format of <string> is nnn.nnn.nnn.nnn. The range of the first section of "nnn" is from 0 to 223 (except 127), and the ranges of the other three sections of "nnn" are from 0 to 255.
- When you use this command, the IP configuration mode should be Auto IP or Static IP mode.

Return Format

The query returns the current gateway in strings.

Example

```
:LAN:GATeway 192.168.1.1          /*Sets the default gateway to
192.168.1.1.*/
:LAN:GATeway?                    /*The query returns the current
gateway.*/
```

3.17.4 :LAN:DNS

Syntax

:LAN:DNS <*string*>

:LAN:DNS?

Description

Sets or queries the DNS address.

Parameter

Name	Type	Range	Default
<string>	ASCII String	Refer to *Remarks*	-

Remarks

- The format of <string> is nnn.nnn.nnn.nnn. The range of the first section of "nnn" is from 0 to 223 (except 127), and the ranges of the other three sections of "nnn" are from 0 to 255.
- When you use this command, the IP configuration mode should be Auto IP or Static IP mode.

Return Format

The query returns the current DNS address in strings.

Example

```
:LAN:DNS 192.168.1.1          /*Sets the DNS address to
192.168.1.1*/
:LAN:DNS?                    /*The query returns the current DNS
address.*/
```

3.17.5 :LAN:MAC?

Syntax

:LAN:MAC?

Description

Queries the MAC address of the instrument.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the MAC address in strings. For example, 00:19:AF:00:11:22.

Example

```
N/A
```

3.17.6 :LAN:DSErver?

Syntax

:LAN:DSErver?

Description

Queries the address of the DHCP server.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the address of the DHCP server in strings.

Example

```
N/A
```

3.17.7 :LAN:MANual

Syntax

:LAN:MANual <*bool*>

:LAN:MANual?

Description

Turns on or off the static IP configuration mode; or queries the on/off status of the static IP configuration mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

When the static IP mode is valid, disable DHCP and Auto IP manually. You can self-define the network parameters of the oscilloscope, such as IP address, subnet mask, gateway, and DNS address. For the setting of the IP address, refer to the *:LAN:IPADdress* command. For the setting of the subnet mask, refer to the *:LAN:SMASk* command. For the setting of the gateway, refer to the *:LAN:GATeway* command. For the setting of DNS, refer to the *:LAN:DNS* command.

Return Format

The query returns 1 or 0.

Example

```
:LAN:MANual ON                    /*Enables the static IP
configuration mode.*/
:LAN:MANual?                      /*The query returns 1.*/
```

3.17.8 :LAN:IPADdress

Syntax

:LAN:IPADdress <*string*>

:LAN:IPADdress?

Description

Sets or queries the IP address of the instrument.

Parameter

Name	Type	Range	Default
<string>	ASCII String	Refer to *Remarks*	-

Remarks

- The format of <string> is nnn.nnn.nnn.nnn. The range of the first section of "nnn" is from 0 to 223 (except 127), and the ranges of the other three sections of "nnn" are from 0 to 255.
- When you use the command, the IP configuration mode should be static IP. Besides, the DHCP and auto IP should be disabled.

Return Format

The query returns the current IP address in strings.

Example

```
:LAN:IPADdress 192.168.1.10          /*Sets the IP address to
192.168.1.10.*/
:LAN:IPADdress?                     /*The query returns the current
IP address.*/
```

3.17.9 :LAN:SMASk

Syntax

:LAN:SMASk <*string*>

:LAN:SMASk?

Description

Sets or queries the subnet mask.

Parameter

Name	Type	Range	Default
<string>	ASCII String	Refer to *Remarks*	-

Remarks

- The format of <string> is nnn.nnn.nnn.nnn. The range of the section "nnn" is from 0 to 255.

- When you use the command, the IP configuration mode should be static IP.

  Besides, the DHCP and auto IP should be disabled.

Return Format

The query returns the current subnet mask in strings.

Example

```
:LAN:SMASk 255.255.255.0          /*Sets the subnet mask to
255,255,255.0*/
:LAN:SMASk?                      /*The query returns the current
subnet mask.*/
```

3.17.10 :LAN:STATus?

Syntax

`:LAN:STATus?`

Description

Queries the current network configuration status.

Parameter

N/A

Remarks

- **UNLINK:** not connected.
- **CONNECTED:** the network is successfully connected.
- **INIT:** the instrument is acquiring an IP address.
- **IPCONFLICT:** there is an IP address conflict.
- **BUSY:** please waiting...
- **CONFIGURED:** the network configuration has been successfully configured.
- **DHCPFAILED:** the DHCP configuration has failed.
- **INVALIDIP:** invalid IP.
- **IPLOSE:** IP lost.

Return Format

The query returns UNLINK, CONNECTED, INIT, IPCONFLICT, BUSY, CONFIGURED, DHCPFAILED, INVALIDIP, or IPLOSE.

Example

```
N/A
```

3.17.11 :LAN:VISA?

Syntax

:LAN:VISA? [<*type*>]

Description

Queries the VISA address of the instrument.

Parameter

Name	Type	Range	Default
<type>	Discrete	{USB|LXI|SOCKET}	-

Remarks

This command contains a parameter "type" and it is used to set or query the address type. By default, it returns the LXI address.

Return Format

The query returns the VISA address in strings.

Example

```
N/A
```

3.17.12 :LAN:MDNS

Syntax

:LAN:MDNS <*bool*>

:LAN:MDNS?

Description

Enables or disables MDNS; or queries the MDNS status.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:LAN:MDNS ON                    /*Enables MDNS.*/
:LAN:MDNS?                      /*The query returns 1.*/
```

3.17.13 :LAN:HOST:NAME

Syntax

:LAN:HOST:NAME <*name*>

:LAN:HOST:NAME?

Description

Sets or queries the host name.

Parameter

Name	Type	Range	Default
<name>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks

N/A

Return Format

The query returns the host name in ASCII strings.

Example

```
N/A
```

3.17.14 :LAN:DESCription

Syntax

:LAN:DESCription <*name*>

:LAN:DESCription?

Description

Sets or queries the description.

Parameter

Name	Type	Range	Default
<name>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks

N/A

Return Format

The query returns the description in ASCII strings.

Example

```
N/A
```

3.17.15 :LAN:APPLy

Syntax

:LAN:APPLy

Description

Applies the network configuration.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.18 :MASK Commands

3.18.1 :MASK:ENABle

Syntax

:MASK:ENABle <*bool*>

:MASK:ENABle?

Description

Enables or disables the pass/fail test function; or queries the on/off status of the pass/fail test function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

The pass/fail test function is invalid in the following conditions: when the horizontal time base mode is ROLL; when in the delayed sweep mode (Zoom).

Return Format

The query returns 1 or 0.

Example

```
:MASK:ENABle ON     /*Enables the pass/fail test function.*/
:MASK:ENABle?       /*The query returns 1.*/
```

3.18.2 :MASK:SOURce

Syntax

:MASK:SOURce <*source*>

:MASK:SOURce?

Description

Sets or queries the source of the pass/fail test.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

When you use the command to set the disabled channel, the disabled channel will be enabled automatically.

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MASK:SOURce CHANnel2    /*Sets the source of the pass/fail test to
CHANnel2.*/
:MASK:SOURce?             /*The query returns CHAN2.*/
```

3.18.3 :MASK:OPERate

Syntax

:MASK:OPERate <*oper*>

:MASK:OPERate?

Description

Runs or stops the pass/fail test; or queries the operating status of the pass/fail test.

Parameter

Name	Type	Range	Default
<oper>	Discrete	{RUN|STOP}	STOP

Remarks

Before running this command, send the *:MASK:ENABle* command to enable the pass/fail test function.

Return Format

The query returns RUN or STOP.

Example

```
:MASK:OPERate RUN    /*Runs the pass/fail test function.*/
:MASK:OPERate?        /*The query returns RUN.*/
```

3.18.4 :MASK:X

Syntax

:MASK:X <*x*>

:MASK:X?

Description

Sets or queries the horizontal adjustment parameter of the pass/fail test mask. The default unit is div.

Parameter

Name	Type	Range	Default
<x>	Real	0.01 div to 2 div	0.24 div

Remarks

N/A

Return Format

The query returns the current horizontal adjustment parameter in scientific notation.

Example

```
:MASK:X 0.28    /*Sets the horizontal adjustment parameter to 0.28
div.*/
:MASK:X?        /*The query returns 2.800000E-1.*/
```

3.18.5 :MASK:Y

Syntax

:MASK:Y <*y*>

:MASK:Y?

Description

Sets or queries the vertical adjustment parameter of the pass/fail test mask. The default unit is div.

Parameter

Name	Type	Range	Default
<y>	Real	0.04 div to 2 div	0.48 div

Remarks

N/A

Return Format

The query returns the current vertical adjustment parameter in scientific notation.

Example

```
:MASK:Y 0.36    /*Sets the vertical adjustment parameter to 0.36
div.*/
:MASK:Y?        /*The query returns 3.600000E-1.*/
```

3.18.6 :MASK:CREate

Syntax

```
:MASK:CREate
```

Description

Creates the pass/fail test mask with the currently set horizontal and vertical adjustment parameters.

Parameter

N/A

Remarks

This command is only valid when the pass/fail test function is enabled and not in the running state.

Return Format

N/A

Example

```
N/A
```

3.18.7 :MASK:RESet

Syntax

```
:MASK:RESet
```

Description

Resets the number of frames that passed and failed the pass/fail test, as well as the total number of frames.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.18.8 :MASK:FAILed?

Syntax

```
:MASK:FAILed?
```

Description

Queries the total number of frames in the pass/fail test results.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer.

Example

```
N/A
```

3.18.9 :MASK:PASSed?

Syntax

```
:MASK:PASSed?
```

Description

Queries the total number of passed frames in the pass/fail test results.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer.

Example

```
N/A
```

3.18.10 :MASK:TOTal?

Syntax

```
:MASK:TOTal?
```

Description

Queries the total number of frames in the pass/fail test results.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer.

Example

```
N/A
```

3.19 :MATH Commands

3.19.1 :MATH<n>:DISPlay

Syntax

:MATH<*n*>**:DISPlay** <*bool*>

:MATH<*n*>**:DISPlay?**

Description

Enables or disables the math operation function; or queries the on/off status of the math operation function.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:MATH1:DISPlay ON     /*Enables the operation function of Math1.*/
:MATH1:DISPlay?       /*The query returns 1.*/
```

3.19.2 :MATH<n>:OPERator

Syntax

:MATH<*n*>**:OPERator** <*opt*>

:MATH<*n*>**:OPERator?**

Description

Sets or queries the operator of math operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<opt>	Discrete	{ADD|SUBTract|MULTiply|DIVision|AND|OR|XOR|NOT|FFT|INTG|DIFF|SQRT|LOG|LN|EXP|ABS|LPASs|HPASs|BPASs|BSTop|AXB}	ADD

Remarks

N/A

Return Format

The query returns ADD, SUBT, MULT, DIV, AND, OR, XOR, NOT, FFT, INTG, DIFF, SQRT, LOG, LN, EXP, ABS, LPAS, HPAS, BPAS, BST, or AXB.

Example

```
:MATH1:OPERator INTG    /*Sets the math operator of Math1 to
Integrate.*/
:MATH1:OPERator?        /*The query returns INTG.*/
```

3.19.3 :MATH<n>:SOURce1

Syntax

:MATH<*n*>**:SOURce1** <*source*>

:MATH<*n*>**:SOURce1?**

Description

Sets or queries the source or Source A of arithmetic operation/function operation/filter operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|REF1| REF2|REF3|REF4|REF5|REF6|REF7| REF8|REF9|REF10|MATH1| MATH2|MATH3}	CHANnel1

Remarks

- For arithmetic operation, this command is used to set Source A.
- For function operation and filter operation, only use this command to set the source.

Return Format

The query returns CHAN2, CHAN3, CHAN4, REF1, REF2, REF3, REF4, REF5, REF6, REF7, REF8, REF9, REF10, MATH1, MATH2, MATH3, or MATH3.

Example

```
:MATH1:SOURce1 CHANnel3    /*Sets Source A of the arithmetic
operation to CHANnel3.*/
:MATH1:SOURce1?            /*The query returns CHAN3.*/
```

3.19.4 :MATH<n>:SOURce2

Syntax

:MATH<*n*>**:SOURce2** <*source*>

:MATH<*n*>**:SOURce2?**

Description

Sets or queries Source B of arithmetic operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|REF1| REF2|REF3|REF4|REF5|REF6|REF7|	CHANnel1

Name	Type	Range	Default
		REF8|REF9|REF10|MATH1| MATH2|MATH3}	

Remarks

This command is only available for arithmetic operation (containing two sources).

Return Format

The query returns CHAN2, CHAN3, CHAN4, REF1, REF2, REF3, REF4, REF5, REF6, REF7, REF8, REF9, REF10, MATH1, MATH2, MATH3, or MATH3.

Example

```
:MATH1:SOURce2 CHANnel3    /*Sets Source B of the arithmetic
operation to CHANnel3.*/
:MATH1:SOURce2?            /*The query returns CHAN3.*/
```

3.19.5 :MATH<n>:LSOUrce1

Syntax

:MATH<*n*>**:LSOUrce1** <*source*>

:MATH<*n*>**:LSOUrce1?**

Description

Sets or queries Source A of the logic operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

The logic operations include A&&B, A||B, A^B, and !A.

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MATH1:LSOUrce1 CHANnel3    /*Sets Source A of the logic operation
to CHANnel3.*/
:MATH1:LSOUrce1?            /*The query returns CHAN3.*/
```

3.19.6 :MATH<n>:LSOUrce2

Syntax

:MATH<*n*>**:LSOUrce2** <*source*>

:MATH<*n*>**:LSOUrce2?**

Description

Sets or queries Source B of the logic operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

- The logic operations include A&&B, A||B, A^B, and !A.
- This command is only available for the logic operation that contains two sources.

  It is used to set Source B.

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MATH1:LSOUrce2 CHANnel4    /*Sets Source B of the logic operation
to CHANnel4.*/
:MATH1:LSOUrce2?            /*The query returns CHAN4.*/
```

3.19.7 :MATH<n>:SCALe

Syntax

:MATH<*n*>**:SCALe** <*scale*>

:MATH<*n*>**:SCALe?**

Description

Sets or queries the vertical scale of the operation results. The unit is related to the currently selected operator and the unit selected by the source.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<scale>	Real	Refer to *Remarks*	-

Remarks

The setting range of vertical scale is related to the currently selected operator and the scale of the source channel. For integration and differentiation operations, the actual range of <scale> is also related to the current horizontal time base.

Return Format

The query returns the vertical scale of the current operation results in scientific notation.

Example

```
:MATH1:SCALe 0.2   /*Sets the vertical scale to 200mV.*/
:MATH1:SCALe?   /*The query returns 2.000000E-1.*/
```

3.19.8 :MATH<n>:OFFSet

Syntax

:MATH<*n*>**:OFFSet** <*offset*>

:MATH<*n*>**:OFFSet?**

Description

Sets or queries the vertical offset of the operation results. The unit is related to the currently selected operator and the unit selected by the source.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<offset>	Real	-1 GV to +1 GV	0.00 V

Remarks

N/A

Return Format

The query returns the vertical offset of the current operation results in scientific notation.

Example

```
:MATH1:OFFSet 8   /*Sets the vertical offset to 8 V.*/
:MATH1:OFFSet?      /*The query returns 8.000000E0.*/
```

3.19.9 :MATH<n>:INVert

Syntax

:MATH<*n*>**:INVert** <*bool*>

:MATH<*n*>**:INVert?**

Description

Enables or disables the inverted display of the operation results; or queries the on/off status of the inverted display of the operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command is invalid for FFT operation.

Return Format

The query returns 1 or 0.

Example

```
:MATH1:INVert ON    /*Enables the inverted display.*/
:MATH1:INVert?      /*The query returns 1.*/
```

3.19.10 :MATH<n>:RESet

Syntax

:MATH<*n*>**:RESet**

Description

After you send this command, the instrument will adjust the vertical scale of the operation results to an optimal value based on the currently selected operator and the horizontal time base of the source.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.19.11 :MATH<n>:EXPand

Syntax

:MATH<*n*>**:EXPand** <*exp*>

:MATH<*n*>**:EXPand?**

Description

Sets or queries the vertical expansion type of math operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<exp>	Discrete	{GND|CENTer}	GND

Remarks

- **CENTer:** when the vertical scale is changed, the waveform will be expanded or compressed around the screen center.
- **GND:** when the vertical scale is changed, the waveform will be expanded or compressed around the signal ground level position.

Return Format

The query returns GND or CENTer.

Example

```
:MATH1:EXPand CENTer /*Sets the vertical expansion type of math
operation of CH1 to CENTer.*/
:MATH1:EXPand? /*The query returns CENTer.*/
```

3.19.12 :MATH<n>:WAVetype

Syntax

:MATH<*n*>**:MATH<n>:WAVetype** <*type*>

:MATH<*n*>**:WAVetype?**

Description

Sets or queries the waveform type of math operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<type>	Discrete	{MAIN|ZOOM}	MAIN

Remarks

- **MAIN:** indicates the main time base region.
- **ZOOM:** indicates the zoomed time base region.

  Only when the zoom function is enabled, can the Zoom under the waveform type be enabled. Zoom is not supported for the FFT operation.

Return Format

The query returns MAIN or ZOOM.

Example

```
:MATH1:WAVetype ZOOM  /*Sets the waveform type of math operation to
ZOOM.*/
:MATH1:WAVetype?  /*The query returns ZOOM.*/
```

3.19.13 :MATH<n>:FFT:SOURce

Syntax

:MATH<*n*>**:FFT:SOURce** <*source*>

:MATH<*n*>**:FFT:SOURce?**

Description

Sets or queries the channel source of FFT operation/filter.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MATH1:FFT:SOURce CHANnel3    /*Sets the channel source of FFT
operation to CHANnel3.*/
:MATH1:FFT:SOURce?      /*The query returns CHAN3.*/
```

3.19.14 :MATH<n>:FFT:WINDow

Syntax

:MATH<*n*>**:FFT:WINDow** <*window*>

:MATH<*n*>**:FFT:WINDow?**

Description

Sets or queries the window function of FFT operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<window>	Discrete	{RECTangle|BLACkman| HANNing|HAMMing|FLATtop| TRIangle}	HANNing

Remarks

- The window function can effectively reduce the spectrum leakage effect.

- Different window functions are applicable to measurements of different waveforms. You need to select the window function according to the different waveforms to be measured and their characteristics.

Return Format

The query returns RECT, BLAC, HANN, HAMM, FLAT, or TRI.

Example

```
:MATH1:FFT:WINDow BLACkman    /*Sets the window function of FFT
operation to Blackman-Harris.*/
:MATH1:FFT:WINDow?             /*The query returns BLAC.*/
```

3.19.15 :MATH<n>:FFT:UNIT

Syntax

:MATH<*n*>**:FFT:UNIT** <*unit*>

:MATH<*n*>**:FFT:UNIT?**

Description

Sets or queries the vertical unit of FFT operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<unit>	Discrete	{VRMS|DB}	DB

Remarks

N/A

Return Format

The query returns VRMS or DB.

Example

```
:MATH1:FFT:UNIT VRMS      /*Sets the vertical unit of FFT operation
results to Vrms.*/
:MATH1:FFT:UNIT?           /*The query returns VRMS.*/
```

3.19.16 :MATH<n>:FFT:SCALe

Syntax

:MATH<*n*>**:FFT:SCALe** <*scale*>

:MATH<*n*>:FFT:SCALe?

Description

Sets or queries the vertical unit of FFT operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<scale>	Real	Refer to *Remarks*	-

Remarks

- When the unit is set to dB, the range of <scale> is from 1 ndB to 5 GdB. The default value is 20 dB.
- When the unit is set to V_{rms}, the range of <scale> is from 1 nV_{rms} to 5 GV_{rms}. The default value is 10 V_{rms}.

You can run the *:MATH<n>:FFT:UNIT* command to configure or query the current unit.

Return Format

The query returns the current vertical scale in scientific notation.

Example

```
:MATH1:FFT:SCALe 0.3    /*Sets the vertical scale of the FFT
operation results to 300 mdB.*/
:MATH1:FFT:SCALe?          /*The query returns 3.000000E-1.*/
```

3.19.17 :MATH<n>:FFT:OFFSet

Syntax

:MATH<*n*>**:FFT:OFFSet** <*offset*>

:MATH<*n*>**:FFT:OFFSet?**

Description

Sets or queries the vertical offset of FFT operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<offset>	Real	Refer to *Remarks*	0 dB

Remarks

- When the unit is set to dB, the range of <offset> is from -1 GdB to 1 GdB. The default value is 0 dB.
- When the unit is set to $V_{rms,}$the range of <offset> is from -1 GV_{rms} to 1 $GV_{rms,}$. The default value is 0 V_{rms}.

You can run the *:MATH<n>:FFT:UNIT* command to configure or query the current unit.

Return Format

The query returns the current vertical offset in scientific notation.

Example

```
:MATH1:FFT:OFFSet 0.3    /*Sets the vertical offset of the FFT
operation results to 300 mdB.*/
:MATH1:FFT:OFFSet?          /*The query returns 3.000000E-1.*/
```

3.19.18 :MATH<n>:FFT:HSCale

Syntax

:MATH<*n*>**:FFT:HSCale** <*hsc*>

:MATH<*n*>**:FFT:HSCale?**

Description

Sets or queries the frequency range of FFT operation results. The default unit is Hz.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<hsc>	Real	10 Hz to 5 GHz	10 MHz

Remarks

You can reduce the frequency range to observe the details of the spectrum.

Modifying the frequency range of the FFT operation results will affect the value of the center frequency. You can run the *:MATH<n>:FFT:HCENter* command to query or modify the center frequency.

Return Format

The query returns the current frequency range in scientific notation.

Example

```
:MATH1:FFT:HSCale 500000    /*Sets the frequency range of the FFT
operation results to 500 kHz.*/
:MATH1:FFT:HSCale?          /*The query returns 5.000000E+5.*/
```

3.19.19 :MATH<n>:FFT:HCENter

Syntax

:MATH<*n*>**:FFT:HCENter** <*cent*>

:MATH<*n*>**:FFT:HCENter?**

Description

Sets or queries the center frequency of FFT operation results, i.e. the frequency relative to the horizontal center of the screen.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<cent>	Real	5 Hz to 5 GHz	5 MHz

Remarks

Modifying the center frequency of the FFT operation results will affect the value of the frequency range. You can run the *:MATH<n>:FFT:HSCale* command to query or modify the frequency range.

Return Format

The query returns the current center frequency in scientific notation. The unit is Hz.

Example

```
:MATH1:FFT:HCENter 10000000    /*Sets the center frequency of the
FFT operation results to 10 MHz.*/
:MATH1:FFT:HCENter?          /*The query returns 1.000000E+7.*/
```

3.19.20 :MATH<n>:FFT:FREQuency:STARt

Syntax

:MATH<*n*>**:FFT:FREQuency:STARt** <*value*>

:MATH<*n*>**:FFT:FREQuency:STARt?**

Description

Sets or queries the start frequency of FFT operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<value>	Real	0 Hz to stop frequency	0 Hz

Remarks

The range of the start frequency of FFT operation is related to the stop frequency. You can run the *:MATH<n>:FFT:FREQuency:END* command to query or configure the stop frequency.

Return Format

The query returns the start frequency of the operation results in scientific notation. The unit is Hz.

Example

```
:MATH1:FFT:FREQuency:STARt 10000000   /*Sets the start frequency of
the FFT operation results to 10 Mhz.*/
:MATH1:FFT:FREQuency:STARt?           /*The query returns 1.000000E
+7.*/
```

3.19.21 :MATH<n>:FFT:FREQuency:END

Syntax

:MATH<*n*>**:FFT:FREQuency:END** <*value*>

:MATH<*n*>**:FFT:FREQuency:END?**

Description

Sets or queries the stop frequency of FFT operation results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<value>	Real	Start Freq to 5 GHz	10 MHz

Remarks

The range of the end frequency of FFT operation is related to the start frequency. You can run the *:MATH<n>:FFT:FREQuency:STARt* command to query or configure the start frequency.

Return Format

The query returns the stop frequency of the operation results in scientific notation. The unit is Hz.

Example

```
:MATH1:FFT:FREQuency:END 10000000    /*Sets the stop frequency of
the FFT operation results to 10 MHz.*/
:MATH1:FFT:FREQuency:END?          /*The query returns 1.000000E
+7.*/
```

3.19.22 :MATH<n>:FFT:SEARch:ENABle

Syntax

:MATH<*n*>**:FFT:SEARch:ENABle** <*bool*>

:MATH<*n*>**:FFT:SEARch:ENABle?**

Description

Enables or disables the FFT peak search; or queries the on/off status of the FFT peak search function.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:MATH1:FFT:SEARch:ENABle ON      /*Enables the FFT peak search.*/
:MATH1:FFT:SEARch:ENABle?           /*The query returns 1.*/
```

3.19.23 :MATH<n>:FFT:SEARch:NUM

Syntax

:MATH<*n*>**:FFT:SEARch:NUM** <*num*>

:MATH<*n*>**:FFT:SEARch:NUM?**

Description

Sets or queries the maximum number of the FFT peak search.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<num>	Integer	1 to 15	5

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 15.

Example

```
:MATH1:FFT:SEARch:NUM 10          /*Sets the maximum number of the
FFT peak search to 10.*/
:MATH1:FFT:SEARch:NUM?            /*The query returns 10.*/
```

3.19.24 :MATH<n>:FFT:SEARch:THReshold

Syntax

:MATH<*n*>**:FFT:SEARch:THReshold** <*thres*>

:MATH<*n*>**:FFT:SEARch:THReshold?**

Description

Sets or queries the threshold of the FFT peak search.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<thres>	Real	Related to the vertical scale and vertical offset of FFT operation	5.5 dB

Remarks

N/A

Return Format

The query returns the threshold in scientific notation.

Example

```
:MATH1:FFT:SEARch:THReshold 0.5          /*Sets the threshold of the
FFT peak search to 500 mdB.*/
:MATH1:FFT:SEARch:THReshold?            /*The query returns
5.000000E-1.*/
```

3.19.25 :MATH<n>:FFT:SEARch:EXCursion

Syntax

:MATH<*n*>**:FFT:SEARch:EXCursion** <*excur*>

:MATH<*n*>**:FFT:SEARch:EXCursion?**

Description

Sets or queries the excursion of the FFT peak search.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<excur>	Real	0 to (8×VerticalScale) (vertical scale of FFT)	1.8 dB

Remarks

VerticalScale indicates the vertical scale of the current source.

Return Format

The query returns the excursion in scientific notation.

Example

```
:MATH1:FFT:SEARch:EXCursion 0.5          /*Sets the excursion of the
FFT peak search to 500 mdB.*/
:MATH1:FFT:SEARch:EXCursion?            /*The query returns
5.000000E-1.*/
```

3.19.26 :MATH<n>:FFT:SEARch:ORDer

Syntax

:MATH<*n*>**:FFT:SEARch:ORDer** <*order*>

:MATH<*n*>**:FFT:SEARch:ORDer?**

Description

Sets or queries the sequence of the FFT peak search results.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<order>	Discrete	{AMPorder|FREQorder}	AMPorder

Remarks

N/A

Return Format

The query returns AMP or FREQ.

Example

```
:MATH1:FFT:SEARch:ORDer AMPorder  /*Sets the sequence of the FFT
peak search results to AMPorder.*/
:MATH1:FFT:SEARch:ORDer?           /*The query returns AMP.*/
```

3.19.27 :MATH<n>:FFT:SEARch:RES?

Syntax

:MATH<*n*>**:FFT:SEARch:RES?**

Description

Queries the peak search results table.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the peak search results table in strings.

Example

The query returns the peak search results table in the following formats.

```
1,2.50000MHz,-24.98dBV
2,3.50000MHz,-27.84dBV
3,4.50000MHz,-30.04dBV
4,5.50125MHz,-31.5dBV
5,6.50125MHz,-32.34dBV
```

3.19.28 :MATH<n>:FILTer:TYPE

Syntax

:MATH<*n*>**:FILTer:TYPE** <*type*>

:MATH<*n*>**:FILTer:TYPE?**

Description

Sets or queries the filter type.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<type>	Discrete	{LPASs|HPASs|BPASs|BSTop}	-

Remarks

DS70000provides 4 practical filters (Low Pass Filter, High Pass Filter, Band Pass Filter, and Band Stop Filter), which can filter the specified frequencies in the signal by setting the bandwidth. You can use the *:MATH<n>:FFT:SOURce* command to set or query the channel source of the filter.

- **LPASs:** indicates low pass filter, which only allows the signals whose frequencies are smaller than the current cut-off frequency to pass.
- **HPASs:** indicates high pass filter, which only allows the signals whose frequencies are greater than the current cut-off frequency to pass.
- **BPASs:** indicates band pass filter, which only allows the signals whose frequencies are greater than the current cut-off frequency 1 and smaller than the current cut-off frequency 2 to pass.

  Note: The cut-off frequency 1 must be smaller than the cut-off frequency 2.

- **BSTop:** indicates band stop filter, which only allows the signals whose frequencies are smaller than the current cut-off frequency 1 or greater than the current cut-off frequency 2 to pass.

  Note: The cut-off frequency 1 must be smaller than the cut-off frequency 2.

Return Format

The query returns LPAS, HPAS, BPAS, or BST.

Example

```
:MATH1:FILTer:TYPE LPASs     /*Sets the filter type to Low Pass
Filter.*/
:MATH1:FILTer:TYPE?          /*The query returns LPAS.*/
```

3.19.29 :MATH<n>:FILTer:W1

Syntax

:MATH<*n*>**:FILTer:W1** <*freq1*>

:MATH<*n*>**:FILTer:W1?**

Description

Sets or queries the cut-off frequency of Low Pass Filter/High Pass Filter; or the cut-off frequency 1 of Band Pass Filter/Band Stop Filter. The default unit is Hz.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<freq1>	Real	Refer to *Remarks*	Related to the filter type LPASs|BPASs|BSTop: 0.005×screen sample rate HPASs: 0.1×screen sample rate

Remarks

- When the filter type is set to LPASs (Low Pass Filter) or HPASs (High Pass Filter), you need to set one cut-off frequency. At this time, the range of <freq1> is from (0.005×screen sample rate) to (0.1×screen sample rate), at a step of

(0.005×screen sample rate). Wherein, screen sample rate = 100/horizontal time base

- When the filter type is set to BPASs (Band Pass Filter) or BSTop (Band Stop Filter), you need to set two cut-off frequencies. Run this command to set the cut-off frequency 1, and run the *:MATH<n>:FILTer:W2* command to set the cut-off frequency 2. At this time, the range of <freq1> is from (0.005×screen sample rate) to (0.095×screen sample rate), at a step of (0.005×screen sample rate). Wherein, screen sample rate = 100/horizontal time base

  Note: The cut-off frequency 1 must be smaller than the cut-off frequency 2.

Return Format

The query returns the current cut-off frequency or cut-off frequency 1 in scientific notation.

Example

```
:MATH1:FILTer:W1 1000000        /*Sets the cut-off frequency of Low
Pass Filter to 1 MHz.*/
:MATH1:FILTer:W1?               /*The query returns 1.000000E+6.*/
```

3.19.30 :MATH<n>:FILTer:W2

Syntax

:MATH<*n*>**:FILTer:W2** <*freq2*>

:MATH<*n*>**:FILTer:W2?**

Description

Sets or queries the cut-off frequency 2 of Band Pass Filter/Band Stop Filter. The default unit is Hz.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<freq2>	Real	Refer to *Remarks*	0.1×screen sample rate

Remarks

When the filter type is set to BPASs (Band Pass Filter) or BSTop (Band Stop Filter), you need to set two cut-off frequencies. Run the *:MATH<n>:FILTer:W1* command to set

the cut-off frequency 1, and run this command to set the cut-off frequency 2. At this time, the range of <freq2> is from (0.01×screen sample rate) to (0.1×screen sample rate), at a step of (0.005×screen sample rate). Wherein, screen sample rate = 100/ horizontal time base

Note: The cut-off frequency 2 must be greater than the cut-off frequency 1.

Return Format

The query returns the current cut-off frequency 2 in scientific notation.

Example

```
:MATH1:FILTer:W2 1500000      /*Sets the cut-off frequency 2 of
Band Pass Filter to 1.5 Mhz.*/
:MATH1:FILTer:W2?             /*The query returns 1.500000E+6.*/
```

3.19.31 :MATH<n>:SENSitivity

Syntax

:MATH<*n*>**:SENSitivity** <*sens*>

:MATH<*n*>**:SENSitivity?**

Description

Sets or queries the sensitivity of the logic operation. The default unit is div.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<sens>	Real	100 mdiv to 1 div	300 mdiv

Remarks

N/A

Return Format

The query returns the sensitivity of the logic operation in scientific notation.

Example

```
:MATH1:SENSitivity 0.2    /*Sets the sensitivity of the logic
operation to 0.2 div.*/
:MATH1:SENSitivity?     /*The query returns 2.000000E-1.*/
```

3.19.32 :MATH<n>:DISTance

Syntax

:MATH<*n*>**:DISTance** <*dist*>

:MATH<*n*>**:DISTance?**

Description

Sets or queries the smoothing window width of differential operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<dist>	Integer	5 to 10000	5

Remarks

N/A

Return Format

The query returns an integer ranging from 5 to 10000.

Example

```
:MATH1:DISTance 20    /*Sets the smoothing window width of
differential operation to 20.*/
:MATH1:DISTance?     /*The query returns 20.*/
```

3.19.33 :MATH<n>:THReshold1

Syntax

:MATH<*n*>**:THReshold1** <*thre*>

:MATH<*n*>**:THReshold1?**

Description

Sets or queries threshold level of Analog Channel 1 in the logic operation. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-

Name	Type	Range	Default
<thre>	Real	(-4 × VerticalScale - VerticalOffset) to (4 × VerticalScale - VerticalOffset)	0 V

Remarks

- This command is only available for the logic operations A&&B, A||B, A^B, and !A.
- VerticalScale indicates the vertical scale of Analog Channel 1. VerticalOffset indicates the vertical offset of Analog Channel 1. The step value is VerticalScale/10.

Return Format

The query returns the threshold level of of Analog Channel 1 in scientific notation.

Example

```
:MATH1:THReshold1 0.8   /*Sets the threshold level of Analog
Channel 1 in logic operation to 800 mV.*/
:MATH1:THReshold1?      /*The query returns 8.000000E-1.*/
```

3.19.34 :MATH<n>:THReshold2

Syntax

:MATH<*n*>**:THReshold2** <*thre*>

:MATH<*n*>**:THReshold2?**

Description

Sets or queries threshold level of Analog Channel 2 in the logic operation. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<thre>	Real	(-4 × VerticalScale - VerticalOffset) to	0 V

Name	Type	Range	Default
		(4 × VerticalScale - VerticalOffset)	

Remarks

- This command is only available for the logic operations A&&B, A||B, A^B, and !A.
- VerticalScale indicates the vertical scale of Analog Channel 2. VerticalOffset indicates the vertical offset of Analog Channel 2. The step value is VerticalScale/10.

Return Format

The query returns the threshold level of of Analog Channel 2 in scientific notation.

Example

```
:MATH1:THReshold2 0.8   /*Sets the threshold level of Analog
Channel 2 in logic operation to 800 mV.*/
:MATH1:THReshold2?      /*The query returns 8.000000E-1.*/
```

3.19.35 :MATH<n>:THReshold3

Syntax

:MATH<*n*>**:THReshold3** <*thre*>

:MATH<*n*>**:THReshold3?**

Description

Sets or queries threshold level of Analog Channel 3 in the logic operation. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<thre>	Real	(-4 × VerticalScale - VerticalOffset) to (4 × VerticalScale - VerticalOffset)	0 V

Remarks

- This command is only available for the logic operations A&&B, A||B, A^B, and !A.
- VerticalScale indicates the vertical scale of Analog Channel 3. VerticalOffset indicates the vertical offset of Analog Channel 3. The step value is VerticalScale/10.

Return Format

The query returns the threshold level of of Analog Channel 3 in scientific notation.

Example

```
:MATH1:THReshold3 0.8   /*Sets the threshold level of Analog
Channel 3 in logic operation to 800 mV.*/
:MATH1:THReshold3?      /*The query returns 8.000000E-1.*/
```

3.19.36 :MATH<n>:THReshold4

Syntax

:MATH<*n*>**:THReshold4** <*thre*>

:MATH<*n*>**:THReshold4?**

Description

Sets or queries threshold level of Analog Channel 4 in the logic operation. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<thre>	Real	(-4 × VerticalScale - VerticalOffset) to (4 × VerticalScale - VerticalOffset)	0 V

Remarks

- This command is only available for the logic operations A&&B, A||B, A^B, and !A.

- VerticalScale indicates the vertical scale of Analog Channel 4. VerticalOffset indicates the vertical offset of Analog Channel 4. The step value is VerticalScale/10.

Return Format

The query returns the threshold level of of Analog Channel 4 in scientific notation.

Example

```
:MATH1:THReshold4 0.8   /*Sets the threshold level of Analog
Channel 4 in logic operation to 800 mV.*/
:MATH1:THReshold4?      /*The query returns 8.000000E-1.*/
```

3.20 :MEASure Commands

3.20.1 :MEASure:SOURce

Syntax

:MEASure:SOURce <*source*>

:MEASure:SOURce?

Description

Sets or queries the channel source of the current measurement parameter.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SETup:DSA* and *:MEASure:SETup:PSA* commands.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:SOURce CHANnel2    /*Sets the channel source of the
measurement parameter to CHANnel2.*/
:MEASure:SOURce?             /*The query returns CHAN2.*/
```

3.20.2 :MEASure:ITEM

Syntax

:MEASure:ITEM <*item*>[,<*src*>[,<*src*>]]

:MEASure:ITEM? <*item*>[,<*src*>[,<*src*>]]

Description

Measures any waveform parameter of the specified source, or queries the statistical results of any waveform parameter of the specified source.

Parameter

Name	Type	Range	Default
<item>	Discrete	{VMAX|VMIN|VPP|VTOP|VBASe|VAMP|VAVG|VRMS|OVERshoot|PREShoot|MAREa|MPAREa|PERiod|FREQuency|RTIMe|FTIMe|PWIDth|NWIDth|PDUTy|NDUTy| TVMAX|TVMIN|PSLewrate|NSLewrate|VUPPer|VMID|VLOWer|VARiance|PVRMs|PPULses|NPULses|PEDGes|NEDGes|RRDelay|RFDelay|FRDelay|FFDelay|RRPHase|RFPHase|FRPHase|FFPHase}	-
<src>	Discrete	Refer to *Remarks*	-

Remarks

- The parameter [,<src>[,<src>]] is used to set the source of the parameter under measurement.
- If the parameter <item> is set to PERiod, FREQuency, PWIDth, NWIDth, PDUTy, NDUTy, RRDelay, RFDelay, FRDelay, FFDelay, RRPHase, RFPHase, FRPHase, or FFPHase, the range of the parameter <src> is any one of the values in {CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}.
- If the measurement parameter is a single source, you only need to set one source. If this parameter is omitted, then the source is by default, selected by the *:MEASure:SOURce* command.

- If the measurement parameter is a dual channel source, observe the following rules to determine the source that you've selected. That is, if the parameter <src> is omitted, the first source is, by default, the one that you've selected in the last sent command (*:MEASure:SOURce*, *:MEASure:SETup:PSA*, or *:MEASure:SETup:DSA*); the second source is, by default, the one that you've selected in the last sent command (*:MEASure:SETup:PSB* or *:MEASure:SETup:DSB*).

Return Format

The query returns the current measurement value in scientific notation.

Example

```
:MEASure:ITEM OVERshoot,CHANnel2   /*Enables the overshoot
measurement of CH2.*/
:MEASure:ITEM? OVERshoot,CHANnel2  /*The query returns
8.888889E-3.*/
```

3.20.3 :MEASure:CLEar

Syntax

`:MEASure:CLEar`

Description

Clears all the measurement items.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:MEASure:CLEar /*Clears all the measurement items.*/
```

3.20.4 :MEASure:AMSource

Syntax

`:MEASure:AMSource` <*chan*>

:MEASure:AMSource?

Description

Sets the source and displays all measurement values of the set source; or queries the channel source(s) of the all measurement function.

Parameter

Name	Type	Range	Default
<chan>	Discrete	{CHANnel1|CHANnel2|CHANnel3|HANnel4|OFF}	OFF

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:MEASure:AMSource CHANnel1    /*Sets the source to CHANnel1.*/
:MEASure:AMSource?             /*The query returns CHAN1.*/
```

3.20.5 :MEASure:STATistic:DISPlay

Syntax

:MEASure:STATistic:DISPlay <*bool*>

:MEASure:STATistic:DISPlay?

Description

Enables or disables the statistical function; or queries the status of the statistical function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

When the statistical function is enabled, the instrument makes statistics of the measurement results for at most 10 measurement items that are turned on last time and displays the statistical results.

Return Format

The query returns 1 or 0.

Example

```
:MEASure:STATistic:DISPlay ON    /*Enables the statistical
function.*/
:MEASure:STATistic:DISPlay?      /*The query returns 1.*/
```

3.20.6 :MEASure:STATistic:RESet

Syntax

:MEASure:STATistic:RESet

Description

Clears the history statistics data and makes statistics again.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.20.7 :MEASure:STATistic:ITEM

Syntax

:MEASure:STATistic:ITEM <*item*>[,<*src*>[,<*src*>]]

:MEASure:STATistic:ITEM?< <*type*>,<*item*>[,<*src*>[,<*src*>]]

Description

Enables the statistical function of any waveform parameter of the specified source, or queries the statistical results of any waveform parameter of the specified source.

Parameter

Name	Type	Range	Default
<item>	Discrete	{VMAX|VMIN|VPP|VTOP|VBASe| VAMP|VAVG|VRMS|OVERshoot|	-

Name	Type	Range	Default
		PREShoot|MAREa|MPAREa| PERiod|FREQuency|RTIMe| FTIMe|PWIDth|NWIDth|PDUTy| NDUTy|TVMAX|TVMIN| PSLewrate|NSLewrate|VUPPer| VMID|VLOWer|VARiance|PVRMs| PPULses|NPULses|PEDGes| NEDGes|RRDelay|RFDelay| FRDelay|FFDelay|RRPHase| RFPHase|FRPHase|FFPHase}	
<src>	Discrete	Refer to *Remarks*	-
<type>	Discrete	{MAXimum|MINimum|CURRent| AVERages|DEViation|CNT}	-

Remarks

- The parameter [,<src>[,<src>]] is used to set the source of the parameter under measurement.
- If the parameter <item> is set to PERiod, FREQuency, PWIDth, NWIDth, PDUTy, NDUTy, RRDelay, RFDelay, FRDelay, FFDelay, RRPHase, RFPHase, FRPHase, or FFPHase, the range of the parameter <src> is any one of the values in {CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}.
- If the measurement parameter is a single source, you only need to set one source. If this parameter is omitted, then the source is, by default, the one that you've selected in the last sent command (*:MEASure:SOURce*, *:MEASure:SETup:PSA*, or *:MEASure:SETup:DSA*).
- If the measurement parameter is a dual channel source, observe the following rules to determine the source that you've selected. That is, if the parameter <src> is omitted, the first source is, by default, the one that you've selected in the last sent command (*:MEASure:SOURce*, *:MEASure:SETup:PSA*, or *:MEASure:SETup:DSA*); the second source is, by default, the one that you've

selected in the last sent command (*:MEASure:SETup:PSB* or *:MEASure:SETup:DSB*).

Return Format

The query returns the statistical results in scientific notation.

Example

```
:MEASure:STATistic:ITEM VPP,CHANnel2   /*Enables the statistical
function of the peak-peak value of CH2.*/
:MEASure:STATistic:ITEM? MAXimum,VPP   /*The query returns
9.120000E-1.*/
```

3.20.8 :MEASure:SETup:MAX

Syntax

:MEASure:SETup:MAX <*value*>

:MEASure:SETup:MAX?

Description

Sets or queries the threshold level upper limit of the analog channel in auto measurement.

Parameter

Name	Type	Range	Default
<value>	Integer	Refer to *Remarks*	-

Remarks

- When the threshold type is percentage, its range is from 0% to 100%; when the threshold type is absolute, its range is from -100 V to 100 V.
- When the set upper limit is smaller than the current middle value, a prompt message "Set at lower limit" will be displayed, and the middle value will not be modified.

Return Format

The query returns an integer.

Example

```
:MEASure:SETup:MAX 95     /*Sets the upper limit of the threshold
level to 95%.*/
:MEASure:SETup:MAX?       /*The query returns 95.*/
```

3.20.9 :MEASure:SETup:MID

Syntax

:MEASure:SETup:MID <*value*>

:MEASure:SETup:MID?

Description

Sets or queries the threshold level middle value of the analog channel in auto measurement.

Parameter

Name	Type	Range	Default
<value>	Integer	Refer to *Remarks*	-

Remarks

- When the threshold type is percentage, its range is from 0% to 100%; when the threshold type is absolute, its range is from -20 V to 20 V.
- The set middle value must be smaller than the currently set upper limit and greater than the currently set lower limit.

Return Format

The query returns an integer.

Example

```
:MEASure:SETup:MID 89     /*Sets the middle value of the threshold
level to 89%.*/
:MEASure:SETup:MID?       /*The query returns 89.*/
```

3.20.10 :MEASure:SETup:MIN

Syntax

:MEASure:SETup:MIN <*value*>

:MEASure:SETup:MIN?

Description

Sets or queries the threshold level lower limit of the analog channel in auto measurement.

Parameter

Name	Type	Range	Default
<value>	Integer	Refer to *Remarks*	-

Remarks

- When the threshold type is percentage, its range is from 0% to 100%; when the threshold type is absolute, its range is from -100 V to 100 V.
- When the set lower limit is greater the current middle value, a prompt message "Set at upper limit" will be displayed, and the middle value will not be modified.

Return Format

The query returns an integer.

Example

```
:MEASure:SETup:MIN 53     /*Sets the lower limit of the threshold
level to 53%.*/
:MEASure:SETup:MIN?       /*The query returns 53.*/
```

3.20.11 :MEASure:SETup:PSA

Syntax

:MEASure:SETup:PSA <*source*>

:MEASure:SETup:PSA?

Description

Sets or queries Source A in the phase or delay measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|MATH1| MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SETup:DSA* and *:MEASure:SETup:DSA* commands.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:SETup:PSA CHANnel1    /*Sets Source A of the phase
measurement to CHANnel1.*/
:MEASure:SETup:PSA?             /*The query returns CHAN1.*/
```

3.20.12 :MEASure:SETup:PSB

Syntax

:MEASure:SETup:PSB <*source*>

:MEASure:SETup:PSB?

Description

Sets or queries Source B in the phase or delay measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SETup:DSB* command.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:SETup:PSB CHANnel2    /*Sets Source B of the phase
measurement to CHANnel2.*/
:MEASure:SETup:PSB?             /*The query returns CHAN2.*/
```

3.20.13 :MEASure:SETup:DSA

Syntax

:MEASure:SETup:DSA <*source*>

:MEASure:SETup:DSA?

Description

Sets or queries Source A in the phase or delay measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SETup:PSA* and *:MEASure:SETup:PSA* commands.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:SETup:DSA CHANnel1    /*Sets Source A of the delay
measurement to CHANnel1.*/
:MEASure:SETup:DSA?             /*The query returns CHAN1.*/
```

3.20.14 :MEASure:SETup:DSB

Syntax

:MEASure:SETup:DSB <*source*>

:MEASure:SETup:DSB

Description

Sets or queries Source B in the phase or delay measurement.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SETup:PSB* command.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:SETup:DSB CHANnel2    /*Sets Source B of the delay
measurement to CHANnel2.*/
:MEASure:SETup:DSB?             /*The query returns CHAN2.*/
```

3.20.15 :MEASure:THReshold:SOURce

Syntax

:MEASure:THReshold:SOURce <*source*>

:MEASure:THReshold:SOURce?

Description

Sets or queries the threshold source.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

Modifying the threshold will affect the measurement results of time, delay and phase parameters.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:MEASure:THReshold:SOURce CHANnel2    /*Sets the threshold source
to CHANnel2.*/
:MEASure:THReshold:SOURce?             /*The query returns CHAN2.*/
```

3.20.16 :MEASure:THReshold:TYPe

Syntax

:MEASure:THReshold:TYPe <*type*>

:MEASure:THReshold:TYPe?

Description

Sets or queries the measurement threshold type.

Parameter

Name	Type	Range	Default
<type>	Discrete	{PERCent|ABSolute}	PERCent

Remarks

N/A

Return Format

The query returns PERCent or ABSolute.

Example

```
:MEASure:THReshold:TYPe ABSolute    /*Sets the threshold type to
ABSolute.*/
:MEASure:THReshold: TYPe?             /*The query returns
ABSolute.*/
```

3.20.17 :MEASure:THReshold:DEFault

Syntax

`:MEASure:THReshold:DEFault`

Description

Sets the threshold level of the analog channel in auto measurement to a default value.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.20.18 :MEASure:MODE

Syntax

:MEASure:MODE <*mode*>

:MEASure:MODE?

Description

Sets or queries the measurement mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{NORMal|PRECision}	NORMal

Remarks

- **NORMal:** executes measurement of up to 1 Mpts.
- **PRECision:** executes measurement of up to 200 Mpts, improving the resolution of measurement results. Note, in this mode, the refresh rate of the waveforms may be reduced.

Return Format

The query returns NORM or PREC.

Example

```
:MEASure:MODE PRECision      /*Sets the measurement mode to
PRECision.*/
:MEASure:MODE?               /*The query returns PREC.*/
```

3.20.19 :MEASure:AREA

Syntax

:MEASure:AREA <*area*>

:MEASure:AREA?

Description

Sets or queries the type of the measurement range.

Parameter

Name	Type	Range	Default
<area>	Discrete	{MAIN|ZOOM}	MAIN

Remarks

- **MAIN:** indicates that the measurement range is within the main time base region.
- **ZOOM:** indicates that the measurement range is within the zoomed time base region. Note that only when you enable the delayed sweep function first, can "Zoom" be enabled.

Return Format

The query returns MAIN or ZOOM.

Example

```
:MEASure:AREA ZOOM              /*Sets the type of the measurement
range to ZOOM.*/
:MEASure:AREA?                    /*The query returns ZOOM.*/
```

3.20.20 :MEASure:INDicator

Syntax

:MEASure:INDicator <*bool*>

:MEASure:INDicator?

Description

Sets or queries the on/off status of the measurement auto cursor.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:MEASure:INDicator ON   /*Sets the measurement auto cursor to be
on.*/
:MEASure:INDicator? /*The query returns 1.*/
```

3.20.21 :MEASure:COUNter:ENABle

Syntax

:MEASure:COUNter:ENABle <*bool*>

:MEASure:COUNter:ENABle?

Description

Sets or queries the on/off status of the frequency counter.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:MEASure:COUNter:ENABle ON    /*Enables the frequency counter.*/
:MEASure:COUNter:ENABle?      /*The query returns 1.*/
```

3.20.22 :MEASure:COUNter:SOURce

Syntax

:MEASure:COUNter:SOURce <*source*>

:MEASure:COUNter:SOURce?

Description

Sets or queries the measurement source for the frequency counter.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|EXT}	CHANnel1

Remarks

The measurement source of the frequency counter is CH1-CH4 and EXT.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, or EXT.

Example

```
:MEASure:COUNter:SOURce CHANnel4    /*Sets the measurement source
for the frequency counter to CHANnel4.*/
:MEASure:COUNter:SOURce?             /*The query returns CHANnel4.*/
```

3.20.23 :MEASure:COUNter:VALue

Syntax

:MEASure:COUNter:VALue?

Description

Queries the measurement results of the frequency counter.

Parameter

N/A

Remarks

N/A

Return Format

The returns the measurement results of the frequency counter in scientific notation.

Example

```
:MEASure:COUNter:VALue?   /*The returns 9.999996E-04.*/
```

3.20.24 :MEASure:AMP:TYPe

Syntax

:MEASure:AMP:TYPe <*val*>

:MEASure:AMP:TYPe?

Description

Sets or queries the amplitude method.

Parameter

Name	Type	Range	Default
<val>	Discrete	{AUTo|MANual}	MANUal

Remarks

- **AUTo:** indicates the Auto method.
- **MANual:** indicates the Manual method.

Return Format

The query returns AUT or MAN.

Example

```
:MEASure:AMP:TYPe AUTo  /*Sets the amplitude method to AUTo.*/
:MEASure:AMP:TYPe?  /*The query returns AUT.*/
```

3.20.25 :MEASure:AMP:MANUal:TOP

Syntax

:MEASure:AMP:MANUal:TOP <*val*>

:MEASure:AMP:MANUal:TOP?

Description

Sets or queries the amplitude top value type for the manual amplitude method.

Parameter

Name	Type	Range	Default
<val>	Discrete	{HISTogram|MAXMin}	HISTogram

Remarks

- **HISTogram:** indicates the histogram type.
- **MAXMin:** indicates the Max-Min type.

Return Format

The query returns HISTogram or MAXMin.

Example

```
:MEASure:AMP:MANUal:TOP MAXMin   /*Sets the amplitude top value
type for the manual amplitude method to MAXMin.*/
:MEASure:AMP:MANUal:TOP?    /*The query returns MAXMin.*/
```

3.20.26 :MEASure:AMP:MANUal:BASe

Syntax

:MEASure:AMP:MANUal:BASe <*val*>

:MEASure:AMP:MANUal:BASe?

Description

Sets or queries the amplitude base value type for the manual amplitude method.

Parameter

Name	Type	Range	Default
<val>	Discrete	{HISTogram|MAXMin}	HISTogram

Remarks

- **HISTogram:** indicates the histogram type.
- **MAXMin:** indicates the Max-Min type.

Return Format

The query returns HISTogram or MAXMin.

Example

```
:MEASure:AMP:MANUal:BASe MAXMin   /*Sets the amplitude base value
type for the manual amplitude method to MAXMin.*/
:MEASure:AMP:MANUal:BASe?    /*The query returns MAXMin.*/
```

3.21 :QUICk Commands

3.21.1 :QUICk:OPERation

Syntax

:QUICk:OPERation <*type*>

:QUICk:OPERation?

Description

Sets or queries the type of the shortcut keys.

Parameter

Name	Type	Range	Default
<type>	Discrete	{SIMage|SWAVe|SSETup|AMEasure|SRESet}	SIMage

Remarks

- **SIMage:** indicates the screen image.

- **SWAVe:** indicates the waveform saving.
- **SSETup:** indicates the setup saving.
- **AMEasure:** indicates all measurement.
- **SRESet:** indicates statistics reset.

Return Format

The query returns SIM, SWAV, SSET, AME, or SRES.

Example

```
:QUICk:OPERation SWAVe        /*Sets the type of the shortcut key to
"save waveform".*/
:QUICk:OPERation?              /*The query returns SWAV.*/
```

3.22 :RECord Commands

3.22.1 :RECord:WRECord:ENABle

Syntax

:RECord:WRECord:ENABle <*bool*>

:RECord:WRECord:ENABle?

Compatible Command Syntax

:RECord:ENABle <*bool*>

:RECord:ENABle?

Description

Enables or disables the waveform recording function; or queries the on/off status of the waveform recording function.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Return Format

The query returns 0 or 1.

Example

```
:RECord:WRECord:ENABle ON     /*Enables the waveform recording
function.*/
:RECord:WRECord:ENABle?       /*The query returns 1.*/
```

Compatible Command Example

```
:RECord:ENABle ON     /*Enables the waveform recording function.*/
:RECord:ENABle?       /*The query returns 1.*/
```

3.22.2 :RECord:WRECord:OPERate

Syntax

:RECord:WRECord:OPERate <*operate*>

:RECord:WRECord:OPERate?

Compatible Command Syntax

:RECord:STARt <*bool*>

:RECord:STARt?

Description

Sets to start the waveform recording, or queries whether the waveform recording starts or stops.

Parameter

Name	Type	Range	Default
<operate>	Discrete	{RUN|STOP}	STOP
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns RUN or STOP.

Return Format of Compatible Command Syntax

The query returns 1 or 0.

Example

```
:RECord:WRECord:OPERate RUN    /*Sets to start recording
waveforms.*/
:RECord:WRECord:OPERate?      /*The query returns RUN.*/
```

Compatible Command Example

```
:RECord:STARt ON     /*Sets to start recording the waveforms.*/
:RECord:STARt?       /*The query returns 1.*/
```

3.22.3 :RECord:WRECord:FRAMes

Syntax

:RECord:WRECord:FRAMes <*value*>

:RECord:WRECord:FRAMes?

Compatible Command Syntax

:RECord:FRAMes <*value*>

:RECord:FRAMes?

Description

Sets or queries the number of frames for waveform recording.

Parameter

Name	Type	Range	Default
<value>	Integer	1 to the maximum number of frames that can be recorded currently	1000

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to the maximum number of frames that can be recorded currently.

Example

```
:RECord:WRECord:FRAMes 300     /*Sets the number of recorded frames
to 300.*/
:RECord:WRECord:FRAMes?        /*The query returns 300.*/
```

Example

```
:RECord:FRAMes 300     /*Sets the number of recorded frames to 300.*/
:RECord:FRAMes?        /*The query returns 300.*/
```

3.22.4 :RECord:WRECord:FRAMes:MAX

Syntax

:RECord:WRECord:FRAMes:MAX

Description

Sets the number of recorded frames to the maximum number of frames.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:RECord:WRECord:FRAMes:MAX   /*Sets the number of recorded frames
to the maximum number of frames.*/
```

3.22.5 :RECord:WRECord: FMAX?

Syntax

:RECord:WRECord: FMAX?

Description

Queries the maximum number of frames that can be recorded currently.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer. The maximum number of frames that can be recorded currently is determined by the current memory depth.

Example

```
N/A
```

3.22.6 :RECord:WRECord:FINTerval

Syntax

:RECord:WRECord:FINTerval <*interval*>

:RECord:WRECord:FINTerval?

Description

Sets or queries the time interval between frames in waveform recording.

Parameter

Name	Type	Range	Default
<interval>	Real	10 ns to 1 s	10 ns

Remarks

N/A

Return Format

The query returns the time interval in scientific notation.

Example

```
:RECord:WRECord:FINTerval 0.01 /*Sets the time interval between
frames in waveform recording to 0.01 s.*/
:RECord:WRECord:FINTerval? /*The query returns 1.000000E-2.*/
```

3.22.7 :RECord:WRECord:PROMpt

Syntax

:RECord:WRECord:PROMpt <*bool*>

:RECord:WRECord:PROMpt?

Description

Sets or queries the on/off status of the beeper when the recording is completed.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:RECord:WRECord:PROMpt ON    /*Enables the beeper when the
recording is completed.*/
:RECord:WRECord:PROMpt?       /*The query returns 1.*/
```

3.22.8 :RECord:WREPlay:FCURrent

Syntax

:RECord:WREPlay:FCURrent <*value*>

:RECord:WREPlay:FCURrent?

Compatible Command Syntax

:RECord:CURRent <*value*>

:RECord:CURRent?

Description

Sets or queries the current frame in waveform playing.

Parameter

Name	Type	Range	Default
<value>	Integer	1 to the maximum number of frames recorded	The maximum number of frames recorded

Remarks

N/A

Return Format

The query returns an integer.

Example

```
:RECord:WREPlay:FCURrent 300    /*Sets the current frame for
waveform playing to 300.*/
:RECord:WREPlay:FCURrent?      /*The query returns 300.*/
```

Compatible Command Example

```
:RECord:CURRent 300    /*Sets the current frame for waveform
playing to 300.*/
:RECord:CURRent?      /*The query returns 300.*/
```

3.22.9 :RECord:WREPlay:FCURrent:TIMe?

Syntax

:RECord:WREPlay:FCURrent:TIMe?

Description

Queries the time stamp of the current frame in waveform playing.

Parameter

N/A

Remarks

N/A

Return Format

Queries the time stamp of the current frame in strings in waveform playing.

Example

```
N/A
```

3.22.10 :RECord:WREPlay:FSTart

Syntax

:RECord:WREPlay:FSTart <*start*>

:RECord:WREPlay:FSTart?

Description

Sets or queries the start frame in waveform playback.

Parameter

Name	Type	Range	Default
<start>	Integer	1 to the maximum number of frames that can be played back currently	-

Remarks

N/A

Return Format

The query returns the start frame in integer.

Example

```
:RECord:WREPlay:FSTart 10     /*Sets the start frame in waveform
playing to 10.*/
:RECord:WREPlay:FSTart?       /*The query returns 10.*/
```

3.22.11 :RECord:WREPlay:FEND

Syntax

:RECord:WREPlay:FEND <*end*>

:RECord:WREPlay:FEND?

Description

Sets or queries the end frame in waveform playback.

Parameter

Name	Type	Range	Default
<end>	Integer	1 to the maximum number of frames recorded	-

Remarks

N/A

Return Format

The query returns the end frame in integer.

Example

```
:RECord:WREPlay:FEND 346    /*Sets the end frame for waveform
playing to 346.*/
:RECord:WREPlay:FEND?      /*The query returns 346.*/
```

3.22.12 :RECord:WREPlay:FMAX?

Syntax

:RECord:WREPlay:FMAX?

Description

Queries the maximum number of frames that can be played back currently.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to the number of frames that have been recorded currently.

Example

```
N/A
```

3.22.13 :RECord:WREPlay:FINTerval

Syntax

:RECord:WREPlay:FINTerval <*interval*>

:RECord:WREPlay:FINTerval?

Description

Sets or queries the time interval between frames in waveform playback.

Parameter

Name	Type	Range	Default
<interval>	Real	1 ms to 10s	-

Remarks

N/A

Return Format

The query returns the time interval in scientific notation.

Example

```
:RECord WREPlay:FINTerval 0.01 /*Sets the time interval between
frames in waveform playback to 0.01 s.*/
:RECord:WREPlay:FINTerval? /*The query returns 1.000000E-2.*/
```

3.22.14 :RECord:WREPlay:MODE

Syntax

:RECord:WREPlay:MODE <*mode*>

:RECord:WREPlay:MODE?

Description

Sets the waveform playback mode to Repeat or Single; queries the waveform playback mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{REPeat|SINGle}	SINGle

Remarks

N/A

Return Format

The query returns REPeat or SINGle.

Example

```
:RECord:WREPlay:MODE REPeat    /*Sets the playback mode to REPeat.*/
:RECord:WREPlay:MODE?      /*The query returns REPeat.*/
```

3.22.15 :RECord:WREPlay:DIRection

Syntax

:RECord:WREPlay:DIRection <*direction*>

:RECord:WREPlay:DIRection?

Description

Sets or queries the playback direction in waveform playing.

Parameter

Name	Type	Range	Default
<direction>	Discrete	{FORWard|BACKward}	FORWard

Remarks

N/A

Return Format

The query returns FORWard or BACKward.

Example

```
:RECord:WREPlay:DIRection BACKward  /*Sets the direction of
waveform playback to BACKward.*/
:RECord:WREPlay:DIRection?  /*The query returns BACKward.*/
```

3.22.16 :RECord:WREPlay:OPERate

Syntax

:RECord:WREPlay:OPERate <*operate*>

:RECord:WREPlay:OPERate?

Compatible Command Syntax

:RECord:PLAY <*bool*>

:RECord:PLAY?

Description

Enables or disables the waveform playing function; or queries the on/off status of the waveform playing function.

Parameter

Name	Type	Range	Default
<operate>	Discrete	{RUN|STOP}	STOP
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns RUN or STOP.

Return Format of Compatible Command Syntax

The query returns 1 or 0.

Example

```
:RECord:WREPlay:OPERate RUN    /*Sets to play the waveforms.*/
:RECord:WREPlay:OPERate?      /*The query returns RUN.*/
```

Compatible Command Example

```
:RECord:PLAY ON    /*Sets to play the waveforms.*/
:RECord:PLAY?      /*The query returns 1.*/
```

3.22.17 :RECord:WREPlay:BACK

Syntax

`:RECord:WREPlay:BACK`

Description

Plays back the previous frame of waveforms manually.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:RECord:WREPlay:BACK /*Plays back the previous frame of waveforms
manually.*/
```

3.22.18 :RECord:WREPlay:NEXT

Syntax

:RECord:WREPlay:NEXT

Description

Plays back the next frame of waveforms manually.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
:RECord:WREPlay:NEXT /*Plays back the next frame of waveforms
manually.*/
```

3.22.19 :RECord:WREPlay:PLAY

Syntax

:RECord:WREPlay:PLAY <*val*>

Description

Sets to play from the start frame or end frame manually.

Parameter

Name	Type	Range	Default
<val>	Discrete	{FFIRst|FEND}	FFIRst

Remarks

- **FFIRst:** starts from the start frame.
- **FEND:** starts from the end frame.

Return Format

The query returns FFIRst or FEND.

Example

```
:RECord:WREPlay: PLAY FEND /*Sets to play from end frame manually.*/
```

3.23 :REFerence Commands

3.23.1 :REFerence:SOURce

Syntax

:REFerence:SOURce <*ref*>,<*chan*>

:REFerence:SOURce? <*ref*>

Description

Sets or queries the source of the specified reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-
<chan>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4|MATH1| MATH2|MATH3|MATH4}	CHANnel1

Remarks

Only the currently enabled channel can be selected as the source of the specified reference channel.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:REFerence:SOURce 1,CHANnel1    /*Sets the source of the reference
channel 1 to CHANnel1.*/
:REFerence:SOURce? 1            /*The query returns CHAN1.*/
```

3.23.2 :REFerence:VSCale

Syntax

:REFerence:VSCale <*ref*>,<*scale*>

:REFerence:VSCale? <*ref*>

Description

Sets or queries the vertical scale of the specified reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-
<scale>	Real	Related to the probe ratio setting. For example, When the probe ratio is 1X: 1 mV to 10 V When the probe ratio is 10X: 10 mV to 100 V	1 V

Remarks

This command is only available when the specified reference channel has saved the reference waveforms.

Return Format

The query returns the vertical scale in scientific notation.

Example

```
:REFerence:VSCale 1,2    /*Sets the vertical scale of reference
channel 1 to 2 V.*/
:REFerence:VSCale? 1   /*The query returns 2.000000E0.*/
```

3.23.3 :REFerence:VOFFset

Syntax

:REFerence:VOFFset <*ref*>,<*offset*>

:REFerence:VOFFset? <*ref*>

Description

Sets or queries the vertical position of the specified reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-

Name	Type	Range	Default
<offset>	Real	(-10 × RefVerticalScale) to (10 × RefVerticalScale)	0 V

Remarks

RefVerticalScale indicates the vertical scale of the currently set reference channel.

Return Format

The query returns the vertical position in scientific notation.

Example

```
:REFerence:VOFFset 1,0.5    /*Sets the vertical offset of reference
channel 1 to 500 mV.*/
:REFerence:VOFFset? 1   /*The query returns 5.000000E-1.*/
```

3.23.4 :REFerence:RESet

Syntax

:REFerence:RESet <*ref*>

Description

Resets the vertical scale and vertical offset of the specified reference channel to the defaults.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.5 :REFerence:CURRent

Syntax

:REFerence:CURRent <*ref*>

Description

Sets the current reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	1

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.6 :REFerence:SAVE

Syntax

:REFerence:SAVE <*ref*>

Description

Saves the waveform of the specified reference channel to the internal memory as the reference waveform.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.7 :REFerence:COLor

Syntax

:REFerence:COLor <*ref*>, <*color*>

:REFerence:COLor? <*ref*>

Description

Sets or queries the color of the specified reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-
<color>	Discrete	{GRAY|GREen|BLUE|RED|ORANge}	-

Remarks

N/A

Return Format

The query returns GRAY, GRE, BLUE, RED, or ORAN.

Example

```
:REFerence:COLor 1,GREen   /*Sets the display color of the
reference channel 1 to GREen.*/
:REFerence:COLor? 1      /*The query returns GRE.*/
```

3.23.8 :REFerence:LABel:ENABle

Syntax

:REFerence:LABel:ENABle <*bool*>

:REFerence:LABel:ENABle?

Description

Enables or disables the display of all the reference channel labels; or queries the on/off display status of all the reference channel labels.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:REFerence:LABel:ENABle ON          /*Enables the display of all
the reference channel labels.*/
:REFerence:LABel:ENABle?            /*The query returns 1.*/
```

3.23.9 :REFerence:LABel:CONTent

Syntax

:REFerence:LABel:CONTent <*ref*>,<*str*>

:REFerence:LABel:CONTent? <*ref*>

Description

Sets or queries the label of the specified reference channel.

Parameter

Name	Type	Range	Default
<ref>	Discrete	{1|2|3|4|5|6|7|8|9|10}	-
<str>	ASCII String	The label can contain English letters and numbers, as well as some symbols.	-

Remarks

N/A

Return Format

The query returns the label of the specified reference channel in strings.

Example

```
:REFerence:LABel:CONTent 1,REF1        /*Sets the label of
reference channel 1 to REF1.*/
:REFerence:LABel:CONTent? 1            /*The query returns REF1.*/
```

3.24 :SAVE and :LOAD Commands

3.24.1 :SAVE:CSV

Syntax

:SAVE:CSV <*path*>

Description

Saves the waveform data displayed on the screen to the internal or external memory in "*.csv" format.

Parameter

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

- <path> includes the file storage location and the filename with a suffix. If the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The stored "*.csv" file can be opened and edited in Excel.

Return Format

N/A

Example

```
:SAVE:CSV D:\123.csv    /*Stores the waveform data displayed on the
screen into the external memory Disk D, with the filename 123.csv*/
```

3.24.2 :SAVE:CSV:LENGth

Syntax

:SAVE:CSV:LENGth <*len*>

:SAVE:CSV:LENGth?

Description

Sets or queries the data length type in saving the "*.csv" file.

Parameter

Name	Type	Range	Default
<len>	Discrete	{DISPlay|MAXimum}	DISPlay

Remarks

- **DISPlay:** only stores the points within the screen region, i.g. 1 kpts.
- **MAXimum:** stores all the points in the internal memory (equal to the current memory depth).

Return Format

The query returns DISP or MAX.

Example

```
:SAVE:CSV:LENGth MAXimum            /*Sets the data length type to
MAXimum.*/
:SAVE:CSV:LENGth?                    /*The query returns MAX.*/
```

3.24.3 :SAVE:IMAGe

Syntax

:SAVE:IMAGe <*path*>

Description

Stores the contents displayed on the screen into the internal or external memory in image format.

Parameter

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

<path> includes the file storage location and the filename with a suffix. The suffix of the filename can be .bmp, .png, or .jpg. If the specified storage location already contains a file with the same filename, the original file will be overwritten.

Return Format

N/A

Example

```
:SAVE:IMAGe D:\123.png    /*Stores the contents displayed on the
screen into the external memory Disk D, with the filename 123.png*/
```

3.24.4 :SAVE:IMAGe:TYPE

Syntax

:SAVE:IMAGe:TYPE < <*type*>

:SAVE:IMAGe:TYPE?

Description

Sets or queries in what format is the image saved.

Parameter

Name	Type	Range	Default
<type>	Discrete	{BMP24|PNG|JPEG}	PNG

Remarks

N/A

Return Format

The query returns BMP24, JPEG, or PNG.

Example

```
:SAVE:IMAGe:TYPE JPEG                  /*Sets the image storage format
to JPEG.*/
:SAVE:IMAGe:TYPE?                      /*The query returns JPEG.*/
```

3.24.5 :SAVE:IMAGe:INVert

Syntax

:SAVE:IMAGe:INVert <*bool*>

:SAVE:IMAGe:INVert?

Description

Enables or disables the invert function when saving the image; or queries whether the invert function is enabled when saving the image.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:SAVE:IMAGe:INVert ON              /*Enables the invert function when
saving the image.*/
:SAVE:IMAGe:INVert?                /*The query returns 1.*/
```

3.24.6 :SAVE:IMAGe:COLor

Syntax

:SAVE:IMAGe:COLor <*color*>

:SAVE:IMAGe:COLor?

Description

Sets the image color for image saving to Color or Gray; or queries image color for image saving.

Parameter

Name	Type	Range	Default
<color>	Discrete	{COLor|GRAY}	COLor

Remarks

N/A

Return Format

The query returns COL or GRAY.

Example

```
:SAVE:IMAGe:COLor GRAY                /*Sets the image color for image
saving to GRAY.*/
:SAVE:IMAGe:COLor?                     /*The query returns GRAY.*/
```

3.24.7 :SAVE:SETup

Syntax

:SAVE:SETup <*path*>

Description

Saves the current setup parameters of the oscilloscope to the internal or external memory as a file.

Parameter

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

- In the internal storage, if a file already exists in the specified storage location, the original file will be overwritten.

- In the external storage, <path> includes the file storage location and the filename with a suffix. If the specified storage location already contains a file with the same filename, the original file will be overwritten.

Return Format

N/A

Example

```
:SAVE:SETup D:\123.stp  /*Stores the current setup parameters of
the oscilloscope into the external memory Disk D, with the filename
123.stp*/
```

3.24.8 :SAVE:STATus?

Syntax

`:SAVE:STATus?`

Description

Queries the status of the memory.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 0 or 1 (when the saving operation is completed).

Example

```
N/A
```

3.24.9 :SAVE:IMAGe:DATA?

Syntax

`:SAVE:IMAGe:DATA?`

Description

Queries the bitmap data stream of the currently displayed image.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the binary data stream.

Example

```
:SAVE:IMAGe:DATA? /*The query returns the binary data stream.*/
```

3.24.10 :LOAD:SETup

Syntax

:LOAD:SETup <*path*>

Description

Loads the setup file of the oscilloscope from the specified path.

Parameter

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

<path> includes the file storage location and the filename with a suffix. By default, its suffix is in ".stp" format.

Return Format

N/A

Example

```
:LOAD:SETup D:\123.stp      /*Loads the setup file "123.stp" from
the external memory Disk D.*/
```

3.25 :SYSTem Commands

3.25.1 :SYSTem:AOUTput

Syntax

:SYSTem:AOUTput <*auxoutput*>

:SYSTem:AOUTput?

Description

Sets or queries the type of the signal output from the [Trigger Out] connector on the rear panel.

Parameter

Name	Type	Range	Default
<auxoutput>	Discrete	{TOUT|PFAil}	TOUT

Remarks

- **TOUT:** after you select this type, the oscilloscope initiates a trigger and then a signal which reflects the current capture rate of the oscilloscope can be output from the [TRIG OUT] connector.
- **PFAil:** after you select this type, a pulse signal will be output from the [TRIG OUT] connector once the oscilloscope detects a successful or failed event.

Return Format

The query returns TOUT or PFA.

Example

```
:SYSTem:AOUTput PFAil   /*Sets the signal type to PFAil.*/
:SYSTem:AOUTput?      /*The query returns PFA.*/
```

3.25.2 :SYSTem:BEEPer

Syntax

:SYSTem:BEEPer <*bool*>

:SYSTem:BEEPer?

Description

Turns on or off the beeper or queries the on/off status of the beeper.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:SYSTem:BEEPer ON    /*Enables the beeper.*/
:SYSTem:BEEPer?      /*The query returns 1.*/
```

3.25.3 :SYSTem:DATE

Syntax

:SYSTem:DATE <*year*>, <*month*>, <*day*>

:SYSTem:DATE?

Description

Sets or queries the system date.

Parameter

Name	Type	Range	Default
<year>	Integer	1900 to 2100	-
<month>	Integer	1 to 12	-
<day>	Integer	1 to 31 (28, 29, or 30)	-

Remarks

N/A

Return Format

The query returns the system date in strings. The year, month, and date are separated by commas.

Example

```
:SYSTem:DATE 2017,10,17        /*Sets the system date to
2017/10/17.*/
:SYSTem:DATE?                 /*The query returns 2017,10,17.*/
```

3.25.4 :SYSTem:ERRor[:NEXT]?

Syntax

:SYSTem:ERRor[:NEXT]?

Description

Queries and deletes the last system error message.

Parameter

N/A

Remarks

N/A

Return Format

The query is in <Message Number>,<Message Content> format. Wherein, <Message Number> is an integer;

<Message Content> is a double-quoted ASCII string. For example, -113,"Undefined header; command cannot be found".

Example

```
N/A
```

3.25.5 :SYSTem:RCLock

Syntax

:SYSTem:RCLock <*clock*>

:SYSTem:RCLock?

Description

Sets or queries the type of the 10M reference clock.

Parameter

Name	Type	Range	Default
<clock>	Discrete	{COUTput|CINPut}	COUTput

Remarks

- **COUTput:** indicates clock output. The oscilloscope outputs the internal 10M clock signal from the [10MHz In/Out] connector on the rear panel.
- **CINPut**: indicates clock input. The oscilloscope receives the external 10 MHz signal input from the [10MHz In/Out] connector on the rear panel as the clock signal.

Return Format

The query returns COUT or CINP.

Example

```
:SYSTem:RCLock CINPut /*Sets the type of the reference clock to
CINPut.*/
:SYSTem:RCLock? /*The query returns CINP.*/
```

3.25.6 :SYSTem:GPIB

Syntax

:SYSTem:GPIB <*adr*>

:SYSTem:GPIB?

Description

Sets or queries the GPIB address.

Parameter

Name	Type	Range	Default
<adr>	Integer	1 to 30	1

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 30.

Example

```
:SYSTem:GPIB 2              /*Sets the GPIB address to 2.*/
:SYSTem:GPIB?               /*The query returns 2.*/
```

3.25.7 :SYSTem:LANGuage

Syntax

:SYSTem:LANGuage <*language*>

:SYSTem:LANGuage?

Description

Sets or queries the system language.

Parameter

Name	Type	Range	Default
<language>	Discrete	{SCHinese|TCHinese|KORean| JAPanese|ENGLish|GERMan| PORTuguese|POLish|FRENch| RUSSian|SPAN|THAI| INDonesian}	SCHinese

Remarks

The language settings are not affected by factory default settings (send the **RST* command).

Return Format

The query returns SCH, TCH, KOR, JAP, ENGL, GERM, PORT, POL, FREN, RUSS, SPAN, THAI, or IND.

Example

```
:SYSTem:LANGuage ENGLish     /*Sets the system language to ENGLish.*/
:SYSTem:LANGuage?             /*The query returns ENGL.*/
```

3.25.8 :SYSTem:PON

Syntax

:SYSTem:PON <*power_on*>

:SYSTem:PON?

Description

Sets or queries the configuration type recalled by the oscilloscope when it is powered on again after power-off.

Parameter

Name	Type	Range	Default
<power_on>	Discrete	{LATest|DEFault}	DEFault

Remarks

N/A

Return Format

The query returns LAT or DEF.

Example

```
:SYSTem:PON LATest         /*Sets the oscilloscope to recall Last
value after it is powered on again.*/
:SYSTem:PON?              /*The query returns LAT.*/
```

3.25.9 :SYSTem:PSTatus

Syntax

:SYSTem:PSTatus <*sat*>

:SYSTem:PSTatus?

Description

Sets or queries the power status of the oscilloscope.

Parameter

Name	Type	Range	Default
<sat>	Discrete	{DEFault|OPEN}	OPEN

Remarks

- **DEFault:** after the oscilloscope is powered on, you need to press the Power key on the front panel to start the oscilloscope.
- **OPEN:** after the oscilloscope is powered on, it starts directly. You do not have to press the Power key.

Return Format

The query returns DEF or OPEN.

Example

```
:SYSTem:PSTatus DEFault          /*Sets the power status to DEFault.*/
:SYSTem:PSTatus?                 /*The query returns DEF.*/
```

3.25.10 :SYSTem:OPTion:INSTall

Syntax

:SYSTem:OPTion:INSTall <*license*>

Description

Installs an option.

Parameter

Name	Type	Range	Default
<license>	ASCII String	Refer to *Remarks*	-

Remarks

- To install the option, first purchase the required option to obtain the key, and then use the key to obtain the option license according to the following steps.
  - Log in to the **RIGOL** official website (*http://www.rigol.com*), click **License Activation** to enter the "Registered product license code" interface.
  - In the interface, input the correct key, serial number (click or tap > **Utility** > **About** to obtain the serial number of the instrument), and the verification code. Click **Generate** to obtain the download link of the option license file. If you need to use the option license file, click to download the file to the specified directory of the USB storage device.
- The license is a fixed length of strings. Each instrument has a unique license.

Return Format

N/A

Example

```
:SYSTem:OPTion:INSTall
DS70000-
RTSA@724bf6bfd20ba23421bf1c98091cf7e87f5917401ce1a860fb1d97560209627
061dfb0fa414e71e178963cbd898dbd8e
```

3.25.11 :SYSTem:OPTion:UNINstall

Syntax

`:SYSTem:OPTion:UNINstall`

Description

Uninstalls all the official options.

Parameter

Remarks

After the option has been uninstalled, you need to restart the instrument.

Return Format

N/A

Example

```
N/A
```

3.25.12 :SYSTem:OPTion:STATus?

Syntax

`:SYSTem:OPTion:STATus?` <*type*>

Compatible Command Syntax

`:SYSTem:OPTion:VALid?` <*type*>

Description

Queries whether an option is activated or not.

Parameter

Name	Type	Range	Default
<type>	Discrete	{RL2|COMP|EMBD|AUTO|FLEX|AUDio|AERO|JITTer|CMUSB}	-

Remarks

- **RL2:** 2G bandwidth.
- **COMP:** PC serial bus trigger and analysis option (RS232/UART).

- **EMBD:** embedded serial bus trigger and analysis option (I2C and SPI).
- **AUTO:** auto serial bus trigger and analysis option (CAN and LIN).
- **FLEX:** FlexRay serial bus trigger and analysis option (FlexRay).
- **AUDio:** audio serial bus trigger and analysis option (I2S).
- **AERO:** MIL-STD-1553 bus trigger and analysis option (MIL-STD-1553).
- **JITTer:** real-time eye diagram and jitter analysis.
- **CMUSB:** USB2.0 pre-compliance test.

Return Format

The query returns 0 or 1.

- 0: indicates that the option is not installed.
- 1: indicates that the official option has been installed.

Example

```
:SYSTem:OPTion:STATus? RL2   /*The query returns 0.*/
```

Compatible Command Example

```
:SYSTem:OPTion:VALid? RL2 /*The query returns 0.*/
```

3.25.13 :SYSTem:RAMount?

Syntax

`:SYSTem:RAMount?`

Description

Queries the number of analog channels of the current instrument.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the number of analog channels of the current instrument in integer. For this oscilloscope, the query returns a fixed value 4.

Example

```
N/A
```

3.25.14 :SYSTem:RESet

Syntax

`:SYSTem:RESet`

Description

Resets the system to power on.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.25.15 :SYSTem:SETup

Syntax

`:SYSTem:SETup` <*setup_data*>

`:SYSTem:SETup?`

Description

Sends or reads the data stream of the system setup file.

Parameter

Name	Type	Range	Default
<setup_data>	ASCII String	Refer to *Remarks*	-

Remarks

- <setup_data> is a binary data block, which consists of the TMC data block header and setup data.
  - The format of the TMC data block header is #NX...X. Wherein, # is the start identifier of the data stream; the N-digit data "X...X" (N≤9) following the start identifier indicate the length of the data stream (the number of bytes).

    For example, 9000002506

Wherein, N is 9, 000002506 following it represents that the data stream contains 2506 bytes of effective data.

- The setup data are expressed in ASCII format.

- When sending the command, directly place the data stream after the command string, then complete the whole sending process in one time. When reading the data stream, ensure that there is enough buffer space to receive the data stream; otherwise, errors might occur in reading the program.

Return Format

N/A

Example

```
N/A
```

3.25.16 :SYSTem:TIME

Syntax

:SYSTem:TIME <*hours*>,<*minutes*>,<*seconds*>

:SYSTem:TIME?

Description

Sets or queries the system time.

Parameter

Name	Type	Range	Default
<hours>	Integer	0 to 23	-
<minutes>	Integer	0 to 59	-
<seconds>	Integer	0 to 59	-

Remarks

There is a certain delay between the return time value and the set time value due to the command response time and other factors.

Return Format

The query returns the system time in strings.

Example

```
:SYSTem:TIME 16,10,17          /*Sets the system time to 16:10:17.*/
:SYSTem:TIME?                 /*The query returns 16:10:17.*/
```

3.25.17 :SYSTem:LOCKed

Syntax

:SYSTem:LOCKed <*bool*>

:SYSTem:LOCKed?

Description

Enables or disables the front-panel key operation and touch screen operation; queries whether the front-panel key operation and touch screen operation are locked.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:SYSTem:LOCKed ON           /*Disables the front-panel key operation
and touch screen operation.*/
:SYSTem:LOCKed?             /*The query returns 1.*/
```

3.25.18 :SYSTem:MODules?

Syntax

:SYSTem:MODules?

Description

Queries the hardware modules.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 1,0,0,0,0. The first figure indicates LA; the second figure indicates DG; and the others are not defined currently. 1 indicates not null, and 0 indicates null.

Example

```
N/A
```

3.25.19 :SYSTem:AUToscale

Syntax

:SYSTem:AUToscale <*bool*>

:SYSTem:AUToscale?

Description

Enables or disables the function of the Auto menu; or queries the on/off status of the Auto menu.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

- You can send this command or use the menu key to disable the function of the Auto menu.
- After the function of the Auto menu is disabled, you cannot perform the Auto Scale operation. The :AUToscale command is invalid.

Return Format

The query returns 1 or 0.

Example

```
:SYSTem: AUToscale ON    /*Enables the function of the Auto menu.*/
:SYSTem: AUToscale?      /*The query returns 1.*/
```

3.25.20 :SYSTem:GAMount?

Syntax

:SYSTem:GAMount?

Description

Queries the number of grids in the horizontal direction of the screen.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the number of grids in integer. For this oscilloscope, the query returns a fixed value 10.

Example

```
N/A
```

3.25.21 :SYSTem:STIMe

Syntax

:SYSTem:STIMe <*bool*>

:SYSTem:STIMe?

Description

Sets or queries whether to display the system time.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	1|ON

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:SYSTem: STIMe ON       /*Sets to display the system time.*/
:SYSTem: STIMe?               /*The query returns 1.*/
```

3.25.22 :SYSTem:VERSion?

Syntax

:SYSTem:VERSion?

Description

Queries the version number of the SCPI used by the system.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the SCPI version number in strings.

Example

```
:SYSTem:VERSion?   /*The query returns 3.0.*/
```

3.25.23 :SYSTem:KIMPedance

Syntax

:SYSTem:KIMPedance <*bool*>

:SYSTem:KIMPedance?

Description

Sets or queries whether to keep impedance when restoring to the default settings.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:SYSTem:KIMPedance ON  /*Sets to keep impedance when restoring to
the default settings.*/
:SYSTem:KIMPedance?   /*The query returns 1.*/
```

3.26 :TIMebase Commands

3.26.1 :TIMebase:DELay:ENABle

Syntax

:TIMebase:DELay:ENABle <*bool*>

:TIMebase:DELay:ENABle?

Description

Turns on or off the delayed sweep; or queries the on/off status of the delayed sweep.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

Delayed sweep can be used to enlarge a length of waveform horizontally to view waveform details.

Return Format

The query returns 1 or 0.

Example

```
:TIMebase:DELay:ENABle ON     /*Enables the delayed sweep.*/
:TIMebase:DELay:ENABle?       /*The query returns 1.*/
```

3.26.2 :TIMebase:DELay:OFFSet

Syntax

:TIMebase:DELay:OFFSet <*offset*>

:TIMebase:DELay:OFFSet?

Description

Sets or queries the offset of the delayed time base.

Parameter

Name	Type	Range	Default
<offset>	Real	-(LeftTime - DelayRange/2) to (RightTime - DelayRange/2)	0

Remarks

LeftTime = 5 × MainScale - MainOffset

RightTime = 5 × MainScale + MainOffset

DelayRange = 10 × DelayScale

Wherein, MainScale indicates the current main time base scale, MainOffset indicates the current main time base offset, and

DelayScale indicates the current delay time base scale.

Return Format

The query returns the offset of the delayed time base in scientific notation.

Example

```
:TIMebase:DELay:OFFSet 0.000002   /*Sets the offset of the delayed
time base to 2 μs.*/
:TIMebase:DELay:OFFSet?           /*The query returns 2.000000E-6.*/
```

3.26.3 :TIMebase:DELay:SCALe

Syntax

:TIMebase:DELay:SCALe <*scale*>

:TIMebase:DELay:SCALe?

Description

Sets or queries the scale of the delayed time base. The default unit is s/div.

Parameter

Name	Type	Range	Default
<scale>	Real	Refer to *Remarks*	2.5 ns/div

Remarks

- The maximum value of the parameter <scale> is the current main time base scale.
- The delayed time base scale can only be the maximum value or the values acquired by reducing form the maximum value at 1-2-5 step. If the minimum value calculated from the above formula is not one of the settable values, take the minimum settable value that is greater than the minimum value calculated.

Return Format

The query returns the scale of the delayed time base in scientific notation.

Example

```
:TIMebase:DELay:SCALe 0.00000005   /*Sets the scale of the delayed
time base to 50 ns/div.*/
:TIMebase:DELay:SCALe?           /*The query returns 5.000000E-8.*/
```

3.26.4 :TIMebase[:MAIN]:OFFSet

Syntax

:TIMebase[:MAIN]:OFFSet <*offset*>

:TIMebase[:MAIN]:OFFSet?

Description

Sets or queries the offset of the main time base. The default unit is s.

Parameter

Name	Type	Range	Default
<offset>	Real	Refer to *Remarks*	0

Remarks

The range of <offset> is related to the current horizontal time base mode and the operating status of the oscilloscope.

Return Format

The query returns the offset of the main time base in scientific notation.

Example

```
:TIMebase:MAIN:OFFSet 0.0002    /*Sets the offset of the main time
base to 200 μs.*/
:TIMebase:MAIN:OFFSet?          /*The query returns 2.000000E-4.*/
```

3.26.5 :TIMebase[:MAIN]:SCALe

Syntax

:TIMebase[:MAIN]:SCALe <*scale*>

:TIMebase[:MAIN]:SCALe?

Description

Sets or queries the scale of the main time base.

Parameter

Name	Type	Range	Default
<scale>	Real	50 ps/div to 1 ks/div	5 ns/div

Remarks

The range of <scale> is related to the current horizontal time base mode of the oscilloscope and its model.

Return Format

The query returns the main time base scale in scientific notation.

Example

```
:TIMebase:MAIN:SCALe 0.0002   /*Sets the main time base scale to
200 µs/div.*/
:TIMebase:MAIN:SCALe?           /*The query returns 2.000000E-4.*/
```

3.26.6 :TIMebase:MODE

Syntax

:TIMebase:MODE <*mode*>

:TIMebase:MODE?

Description

Sets or queries the horizontal time base mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{MAIN|XY|ROLL}	MAIN

Remarks

- **MAIN:** the current time base mode. When you send the command :TIMebase:MODE MAIN, the time base is set to YT mode.
- **XY:** XY mode. After sending the command :TIMebase:MODE XY, you can send the query command :TIMebase:MODE? and the query command returns MAIN.
- **ROLL:** When you send the command :TIMebase:MODE ROLL, the time base is set to Roll mode.

Return Format

The query returns MAIN or ROLL.

Example

```
:TIMebase:MODE ROLL    /*Sets the horizontal time base mode to
ROLL.*/
:TIMebase:MODE?      /*The query returns ROLL.*/
```

3.26.7 :TIMebase:HREFerence:MODE

Syntax

:TIMebase:HREFerence:MODE <*href*>

:TIMebase:HREFerence:MODE?

Description

Sets or queries the horizontal reference mode.

Parameter

Name	Type	Range	Default
<href>	Discrete	{CENTer|LB|RB|TRIG|USER}	CENTer

Remarks

- **CENTer:** when the horizontal time base is modified, the waveform displayed will be expanded or compressed horizontally relative to the screen center.
- **LB:** when the horizontal time base is modified, the waveform displayed will be expanded or compressed relative to the left border of the screen.
- **RB:** when the horizontal time base is modified, the waveform displayed will be expanded or compressed relative to the right border of the screen.
- **TRIG:** when the horizontal time base is modified, the waveform displayed will be expanded or compressed horizontally relative to the trigger position.
- **USER:** when the horizontal time base is modified, the waveform displayed will be expanded or compressed horizontally relative to the user-defined reference position.

Return Format

The query returns CENT, LB, RB, TRIG, or USER.

Example

```
:TIMebase: HREFerence:MODE TRIG          /*Sets the horizontal
reference mode to trigger position.*/
:TIMebase: HREFerence:MODE?              /*The query returns TRIG.*/
```

3.26.8 :TIMebase:HREFerence:POSition

Syntax

:TIMebase:HREFerence:POSition <*pos*>

:TIMebase:HREFerence:POSition?

Description

Sets or queries the user-defined reference position when the waveforms are expanded or compressed horizontally.

Parameter

Name	Type	Range	Default
<pos>	Discrete	-500 to 500	0

Remarks

N/A

Return Format

The query returns an integer ranging from -500 to 500.

Example

```
:TIMebase:HREFerence:POSition 60     /*Sets the user-defined
reference position to 60.*/
:TIMebase:HREFerence:POSition?               /*The query returns
60.*/
```

3.26.9 :TIMebase:VERNier

Syntax

`:TIMebase:VERNier` <*bool*>

`:TIMebase:VERNier?`

Description

Enables or disables the fine adjustment function of the horizontal scale; or queries the on/off status of the fine adjustment function of the horizontal scale.

Parameter

Name	Type	Range	Default
<bool>	Discrete	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:TIMebase:VERNier ON            /*Sets the fine adjustment function
of the horizontal scale to ON .*/
:TIMebase:VERNier?              /*The query returns 1.*/
```

3.26.10 :TIMebase:HOTKeys

Syntax

:TIMebase:HOTKeys <*action*>

Description

Sets the running status.

Parameter

Name	Type	Range	Default
<action>	Discrete	{STOP|RUN|SINGle}	-

Remarks

- **STOP:** stops the measurement.
- **RUN:** runs the measurement.
- **SINGle:** indicates the single trigger measurement.

Return Format

N/A

Example

```
:TIMebase:HOTKeys RUN  /*Sets the operating status to RUN.*/
```

3.26.11 :TIMebase:ROLL

Syntax

:TIMebase:ROLL <*value*>

:TIMebase:ROLL?

Description

Sets or queries the status of the ROLL time base mode.

Parameter

Name	Type	Range	Default
<value>	Integer	{0|1|2}	1

Remarks

- **0:** disables the ROLL mode

- **1|2:** enables the ROLL mode.

Return Format

The query returns 0 or 1.

Example

```
:TIMebase:ROLL 0    /*Disables the ROLL time base mode.*/
:TIMebase:ROLL?      /*The query returns 0.*/
```

3.26.12 :TIMebase:XY:ENABle

Syntax

:TIMebase:XY:ENABle <*bool*>

:TIMebase:XY:ENABle?

Description

Enables or disables the XY mode; or queries the on/off status of the XY mode.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:TIMebase:XY:ENABle OFF    /*Disables the XY mode.*/
:TIMebase:XY:ENABle?      /*The query returns 0.*/
```

3.26.13 :TIMebase:XY:X

Syntax

:TIMebase:XY:X <*s*>

:TIMebase:XY:X?

Description

Sets or queries the source channel of X coordinate when the horizontal time base mode is XY.

Parameter

Name	Type	Range	Default
<s>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TIMebase:XY:X CHANnel3          /*Sets the source channel of X
coordinate to CHANnel3.*/
:TIMebase:XY:X?                 /*The query returns CHAN3.*/
```

3.26.14 :TIMebase:XY:Y

Syntax

:TIMebase:XY:Y <*S*>

:TIMebase:XY:Y?

Description

Sets or queries the channel source of the Y coordinate when the horizontal time base mode is XY.

Parameter

Name	Type	Range	Default
<s>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TIMebase:XY:Y CHANnel3          /*Sets the channel source of the Y
coordinate to CHANnel3.*/
:TIMebase:XY:Y?                 /*The query returns CHAN3.*/
```

3.27 :TRIGger Commands

The **:TRIGger** commands are used to set the trigger source type, trigger input edge type and trigger delay as well as generate a trigger event.

3.27.1 :TRIGger:MODE

Syntax

:TRIGger:MODE <*mode*>

:TRIGger:MODE?

Description

Sets or queries the trigger type.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{EDGE|PULSe|SLOPe|VIDeo|PATTern|DURation|TIMeout|RUNT|WINDow|DELay|SETup|NEDGe|RS232|IIC|SPI|CAN|FLEXray|LIN|IIS|M1553}	EDGE

Remarks

N/A

Return Format

The query returns EDGE, PULS, SLOP, VID, PATT, DUR, TIM, RUNT, WIND, DEL, SET, NEDG, RS232, IIC, SPI, CAN, FLEX, LIN, IIS, or M1553.

Example

```
:TRIGger:MODE SLOPe     /*Selects the Slope trigger.*/
:TRIGger:MODE?          /*The query returns SLOP.*/
```

3.27.2 :TRIGger:COUPling

Syntax

:TRIGger:COUPling <*couple*>

:TRIGger:COUPling?

Description

Sets or queries the trigger coupling type.

Parameter

Name	Type	Range	Default
<couple>	Discrete	{AC|DC|LFReject|HFReject}	DC

Description

This command is only available for the Edge trigger in which the analog channel is selected as the source.

- **AC:** blocks any DC components.
- **DC:** allows DC and AC components to pass the trigger path.
- **LFReject:** blocks the DC components and rejects the low frequency components.
- **HFReject:** rejects the high frequency components.

Return Format

The query returns AC, DC, LFR, or HFR.

Example

```
:TRIGger:COUPling LFReject    /*Sets the trigger coupling type to
low frequency rejection.*/
:TRIGger:COUPling?           /*The query returns LFR.*/
```

3.27.3 :TRIGger:STATus?

Syntax

:TRIGger:STATus?

Description

Queries the current trigger status.

Remarks

N/A

Return Format

The query returns TD, WAIT, RUN, AUTO, or STOP.

3.27.4 :TRIGger:SWEep

Syntax

:TRIGger:SWEep <*sweep*>

:TRIGger:SWEep?

Description

Sets or queries the trigger mode.

Parameter

Name	Type	Range	Default
<sweep>	Discrete	{AUTO|NORMal|SINGle}	AUTO

Remarks

- **AUTO:** Auto trigger. The waveforms are displayed no matter whether the trigger conditions are met.
- **NORMal:** normal trigger. The waveforms are displayed when trigger conditions are met. If the trigger conditions are not met, the oscilloscope displays the original waveforms and waits for another trigger.
- **SINGle:** single trigger. The oscilloscope waits for a trigger, displays the waveforms when the trigger conditions are met, and then stops.

Return Format

The query returns AUTO, NORM, or SING.

Example

```
:TRIGger:SWEep NORMal    /*Selects the normal trigger mode.*/
:TRIGger:SWEep?          /*The query returns NORM.*/
```

3.27.5 :TRIGger:HOLDoff

Syntax

:TRIGger:HOLDoff <*value*>

:TRIGger:HOLDoff?

Description

Sets or queries the trigger holdoff time. The default unit is s.

Parameter

Name	Type	Range	Default
<value>	Real	8 ns to 10 s	8 ns

Remarks

- Trigger holdoff can be used to stably trigger complex waveforms (such as pulse waveform). Holdoff time indicates the time that the oscillscope waits for re-arming the trigger module. The oscilloscope will not trigger before the holdoff time expires.
- Holdoff time is no available for Video trigger, Timeout trigger, Setup&Hold trigger, Nth Edge trigger, RS232 trigger, I2C trigger, SPI trigger, CAN trigger, FlexRay trigger, LIN trigger, I2S trigger, or 1553B trigger.

Return Format

The query returns the trigger holdoff time in scientific notation.

Example

```
:TRIGger:HOLDoff 0.0000002    /*Sets the trigger holdoff time to 200
ns.*/
:TRIGger:HOLDoff?             /*The query returns 2.000000E-7.*/
```

3.27.6 :TRIGger:NREJect

Syntax

:TRIGger:NREJect <*bool*>

:TRIGger:NREJect?

Description

Turns on or off noise rejection; or queries the on/off status of noise rejection.

Parameter

Name	Type	Range	Default
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

- Noise rejection reduces the possibility of the Noise trigger.
- This command is only available when the source is an analog channel or EXT.

Return Format

The query returns 1 or 0.

Example

```
:TRIGger:NREJect ON     /*Enables the noise rejection function.*/
:TRIGger:NREJect?       /*The query returns 1.*/
```

3.27.7 :TRIGger:POSition?

Syntax

:TRIGger:POSition?

Description

Queries the waveform trigger position relative to the corresponding position in the internal memory.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the waveform trigger position relative to the corresponding position in the internal memory in scientific notation.

Example

```
:TRIGger:POSition?    /*The query returns 0.000E+00.*/
```

3.27.8 :TRIGger:EDGE:SOURce

Syntax

:TRIGger:EDGE:SOURce <*source*>

:TRIGger:EDGE:SOURce?

Description

Sets or queries the trigger source of Edge trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4| ACLine| EXT}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, ACL, or EXT.

Example

```
:TRIGger:EDGE:SOURce CHANnel1    /*Sets the trigger source to
CHANnel1.*/
:TRIGger:EDGE:SOURce?             /*The query returns CHAN1.*/
```

3.27.9 :TRIGger:EDGE:SLOPe

Syntax

:TRIGger:EDGE:SLOPe <*slope*>

:TRIGger:EDGE:SLOPe?

Description

Sets or queries the edge type of Edge trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative|RFALl}	POSitive

Remarks

- **POSitive:** indicates the rising edge.
- **NEGative:** indicates the falling edge.
- **RFALl:** indicates the rising or falling edge.

Return Format

The query returns POS, NEG, or RFAL.

Example

```
:TRIGger:EDGE:SLOPe NEGative    /*Sets the edge type to NEGative.*/
:TRIGger:EDGE:SLOPe?            /*The query returns NEG.*/
```

3.27.10 :TRIGger:EDGE:LEVel

Syntax

:TRIGger:EDGE:LEVel <*level*>

:TRIGger:EDGE:LEVel?

Description

Sets or queries the trigger level of Edge trigger. The unit is the same as that of current amplitude of the selected source.

Parameter

Name	Type	Range	Default
<level>	Real	analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0

Remarks

- For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.
- Only when the selected source is an analog channel or external trigger, can this setting command be valid.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:EDGE:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:EDGE:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.11 :TRIGger:PULSe:SOURce

Syntax

:TRIGger:PULSe:SOURce <*source*>

:TRIGger:PULSe:SOURce?

Description

Sets or queries the trigger source of Pulse trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:PULSe:SOURce CHANnel1    /*Sets the trigger source to
CHANnel1.*/
:TRIGger:PULSe:SOURce?             /*The query returns CHAN1.*/
```

3.27.12 :TRIGger:PULSe:POLarity

Syntax

:TRIGger:PULSe:POLarity <*polarity*>

:TRIGger:PULSe:POLarity?

Description

Sets or queries the polarity of Pulse trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:PULSe:POLarity NEGative  /*Sets the polarity of Pulse
trigger to NEGative.*/
:TRIGger:PULSe:POLarity?           /*The query returns NEG.*/
```

3.27.13 :TRIGger:PULSe:WHEN

Syntax

:TRIGger:PULSe:WHEN <*when*>

:TRIGger:PULSe:WHEN?

Description

Sets or queries the trigger condition of Pulse trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{GREater|LESS|GLESs}	GREater

Remarks

- **GREater:** triggers when the positive/negative pulse width of the input signal is greater than the specified pulse width.
- **LESS:** triggers when the positive/negative pulse width of the input signal is smaller than the specified pulse width.
- **GLESs:** triggers when the positive/negative pulse is greater than than the specified lower limit of pulse width and smaller than the specified upper limit of pulse width.

Return Format

The query returns GRE, LESS, or GLES.

Example

```
:TRIGger:PULSe:WHEN LESS    /*Sets the trigger condition to LESS.*/
:TRIGger:PULSe:WHEN?         /*The query returns LESS.*/
```

3.27.14 :TRIGger:PULSe:UWIDth

Syntax

:TRIGger:PULSe:UWIDth <*width*>

:TRIGger:PULSe:UWIDth?

Description

Sets or queries the pulse upper limit of the Pulse trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<width>	Real	Pulse lower limit to 10 s	2 us

Remarks

The set upper limit cannot be smaller than the lower limit, otherwise, the lower limit will be automatically changed.

Return Format

The query returns the upper limit of the pulse width in scientific notation.

Example

```
:TRIGger:PULSe:UWIDth 0.000003   /*Sets the pulse upper limit to 3
µs.*/
:TRIGger:PULSe:UWIDth?           /*The query returns 3.000000E-6.*/
```

3.27.15 :TRIGger:PULSe:LWIDth

Syntax

:TRIGger:PULSe:LWIDth <*width*>

:TRIGger:PULSe:LWIDth?

Description

Sets or queries the pulse lower limit of the Pulse trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<width>	Real	100 ps to upper limit	1us

Remarks

The set lower limit must be smaller than the upper limit, otherwise, the upper limit will be automatically changed.

Return Format

The query returns the lower limit of the pulse width in scientific notation.

Example

```
:TRIGger:PULSe:LWIDth 0.000003   /*Sets the lower limit of the
pulse width to 3 µs.*/
:TRIGger:PULSe:LWIDth?           /*The query returns 3.000000E-6.*/
```

3.27.16 :TRIGger:PULSe:LEVel

Syntax

:TRIGger:PULSe:LEVel <*level*>

:TRIGger:PULSe:LEVel?

Description

Sets or queries the trigger level of Pulse trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0

Remarks

- For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.
- Only when the selected source is analog channel or external trigger, can this setting command be valid.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:PULSe:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:PULSe:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.17 :TRIGger:SLOPe:SOURce

Syntax

:TRIGger:SLOPe:SOURce <*channel*>

:TRIGger:SLOPe:SOURce?

Description

Sets or queries the trigger source of Slope trigger.

Parameter

Name	Type	Range	Default
<channel>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SLOPe:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:SLOPe:SOURce?             /*The query returns CHAN2.*/
```

3.27.18 :TRIGger:SLOPe:POLarity

Syntax

:TRIGger:SLOPe:POLarity <*polarity*>

:TRIGger:SLOPe:POLarity?

Description

Sets or queries the edge type of Slope trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

- **POSitive:** triggers on the rising edge.
- **NEGative:** triggers on the falling edge.

Return Format

The query returns POS or NEG.

Example

```
:TRIGger: SLOPe:POLarity POSitive    /*Sets the polarity of Slope
trigger to POSitive.*/
:TRIGger: SLOPe:POLarity?           /*The query returns POS.*/
```

3.27.19 :TRIGger:SLOPe:WHEN

Syntax

:TRIGger:SLOPe:WHEN <*when*>

:TRIGger:SLOPe:WHEN?

Description

Sets or queries the trigger condition of Slope trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{GREater|LESS|GLESs}	GREater

Remarks

- **GREater:** the positive slope time of the input signal is greater than the specified time.
- **LESS:** the positive slope time of the input signal is smaller than the specified time.
- **GLESs:** the positive slope time of the input signal is greater than the specified lower time limit and smaller than the specified upper time limit.

Return Format

The query returns GRE, LESS, or GLES.

Example

```
:TRIGger:SLOPe:WHEN LESS     /*Sets the trigger condition to LESS*/
:TRIGger:SLOPe:WHEN?          /*The query returns LESS.*/
```

3.27.20 :TRIGger:SLOPe:TIME

Syntax

:TRIGger:SLOPe:TIME <*time*>

:TRIGger:SLOPe:TIME?

Description

Sets the current time of Slope trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to 10 s	1 μs

Remarks

N/A

Return Format

The query returns the time value in scientific notation.

Example

```
:TRIGger:SLOPe:TIME 10   /*Sets the current time value to 10 s.*/
:TRIGger:SLOPe:TIME?           /*The query returns 1.000000E+1.*/
```

3.27.21 :TRIGger:SLOPe:TUPPer

Syntax

:TRIGger:SLOPe:TUPPer <*time*>

:TRIGger:SLOPe:TUPPer?

Description

Sets or queries the upper time limit value of the Slope trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	Lower limit to 10 s	2 μs

Remarks

The set upper limit cannot be smaller than the lower limit, otherwise, the lower limit will be automatically changed.

Return Format

The query returns the upper time limit in scientific notation.

Example

```
:TRIGger:SLOPe:TUPPer 0.000003   /*Sets the upper time limit to 3
μs.*/
:TRIGger:SLOPe:TUPPer?           /*The query returns 3.000000E-6.*/
```

3.27.22 :TRIGger:SLOPe:TLOWer

Syntax

:TRIGger:SLOPe:TLOWer <*time*>

:TRIGger:SLOPe:TLOWer?

Description

Sets or queries the lower time limit value of the Slope trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to upper limit	1 μs

Remarks

The set lower limit must be smaller than the upper limit, otherwise, the upper limit will be automatically changed.

Return Format

The query returns the lower time limit in scientific notation.

Example

```
:TRIGger:SLOPe:TLOWer 0.000000020    /*Sets the lower time limit to
20 ns.*/
:TRIGger:SLOPe:TLOWer?                /*The query returns
2.000000E-8.*/
```

3.27.23 :TRIGger:SLOPe:WINDow

Syntax

:TRIGger:SLOPe:WINDow <*window*>

:TRIGger:SLOPe:WINDow?

Description

Sets or queries the vertical window type of Slope trigger.

Parameter

Name	Type	Range	Default
<window>	Discrete	{TA|TB|TAB}	TA

Remarks

- **TA:** only adjusts the upper limit of the trigger level.
- **TB:** only adjust the lower limit of the trigger level.
- **TAB:** adjusts the upper and lower limits of the trigger level at the same time.

Return Format

The query returns TA, TB, or TAB.

Example

```
:TRIGger:SLOPe:WINDow TB     /*Sets the vertical window type to TB.*/
:TRIGger:SLOPe:WINDow?       /*The query returns TB.*/
```

3.27.24 :TRIGger:SLOPe:ALEVel

Syntax

:TRIGger:SLOPe:ALEVel <*level*>

:TRIGger:SLOPe:ALEVel?

Description

Sets or queries the upper limit of the trigger level of Slope trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Lower limit of the trigger level to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the upper limit of the trigger level in scientific notation.

Example

```
:TRIGger:SLOPe:ALEVel 0.16   /*Sets the upper limit of the trigger
level to 160 mV.*/
:TRIGger:SLOPe:ALEVel?      /*The query returns 1.600000E-1.*/
```

3.27.25 :TRIGger:SLOPe:BLEVel

Syntax

:TRIGger:SLOPe:BLEVel <*level*>

:TRIGger:SLOPe:BLEVel?

Description

Sets or queries the lower limit of the trigger level of Slope trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to upper limit of the trigger level	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the lower limit of the trigger level in scientific notation.

Example

```
:TRIGger:SLOPe:BLEVel 0.16   /*Sets the lower limit of the trigger
level to 160 mV.*/
:TRIGger:SLOPe:BLEVel?      /*The query returns 1.600000E-1.*/
```

3.27.26 :TRIGger:VIDeo:SOURce

Syntax

:TRIGger:VIDeo:SOURce <*source*>

:TRIGger:VIDeo:SOURce?

Description

Sets or queries the trigger source of Video trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:VIDeo:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:VIDeo:SOURce?             /*The query returns CHAN2.*/
```

3.27.27 :TRIGger:VIDeo:POLarity

Syntax

:TRIGger:VIDeo:POLarity <*polarity*>

:TRIGger:VIDeo:POLarity?

Description

Sets or queries the video polarity of Video trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:VIDeo:POLarity NEGative    /*Sets the video polarity to
NEGative.*/
:TRIGger:VIDeo:POLarity?           /*The query returns NEG.*/
```

3.27.28 :TRIGger:VIDeo:MODE

Syntax

:TRIGger:VIDeo:MODE <*mode*>

:TRIGger:VIDeo:MODE?

Description

Sets or queries the sync type of Video trigger.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{ODDField|EVENfield|LINE|ALINes}	ALINes

Remarks

- **ODDField:** indicates the odd field. The oscilloscope triggers on the rising edge of the first ramp waveform in the odd field. It is available when the video standard is NTSC, PAL/SECAM, or 1080i.

- **EVENfield:** indicates the even field. The oscilloscope triggers on the rising edge of the first ramp waveform in the even field. It is available when the video standard is NTSC, PAL/SECAM, or 1080i.
- **LINE:** for NTSC and PAL/SECAM video standards, the oscilloscope triggers on the specified line in the odd or even field. For 480p, 576p, 720p, 480p and 1080i video standards, the oscilloscope triggers on the specified line.
- **ALINes:** triggers on all the horizontal sync pulses.

Return Format

The query returns ODDF, EVEN, LINE, or ALIN.

Example

```
:TRIGger:VIDeo:MODE ODDField    /*Sets the sync type to ODDField.*/
:TRIGger:VIDeo:MODE?            /*The query returns ODDF.*/
```

3.27.29 :TRIGger:VIDeo:LINE

Syntax

:TRIGger:VIDeo:LINE <*line*>

:TRIGger:VIDeo:LINE?

Description

Sets or queries the line number when the sync type of Video trigger is set to Line.

Parameter

Name	Type	Range	Default
<line>	Integer	Refer to *Remarks*	1

Remarks

- **PAL/SECAM:** 1 to 625
- **NTSC:** 1 to 525
- **480P:** 1 to 525
- **576P:** 1 to 625
- **720P60:** 1 to 750
- **720P50:** 1 to 750

- **720P30:** 1 to 750
- **720P25:** 1 to 750
- **720P24:** 1 to 750
- **1080P60:** 1 to 1125
- **1080P50:** 1 to 1125
- **1080P30:** 1 to 1125
- **1080P25:** 1 to 1125
- **1080P24:** 1 to 1125
- **1080I60:** 1 to 1125
- **1080I50:** 1 to 1125

Return Format

The query returns an integer.

Example

```
:TRIGger:VIDeo:LINE 100     /*Sets the line number to 100.*/
:TRIGger:VIDeo:LINE?        /*The query returns 100.*/
```

3.27.30 :TRIGger:VIDeo:STANdard

Syntax

:TRIGger:VIDeo:STANdard <*standard*>

:TRIGger:VIDeo:STANdard?

Description

Sets or queries the video standard of Video trigger.

Parameter

Name	Type	Range	Default
<standard>	Discrete	{PALSecam|NTSC|480P|576P| 720P60|720P50|720P30|720P25| 720P24|1080P60|1080P50| 1080P30|1080P25|1080P24| 1080I60|1080I50}	NTSC

Remarks

Video Standard	Frame Frequency (Frame)	Scan Type	TV Scan Line
PALSecam	25	Interlaced Scan	625
NTSC	30	Interlaced Scan	525
480P	60	Progressive Scan	525
576P	50	Progressive Scan	625
720P60	60	Progressive Scan	750
720P50	50	Progressive Scan	750
720P30	30	Progressive Scan	750
720P25	25	Progressive Scan	750
720P24	24	Progressive Scan	750
1080P60	60	Progressive Scan	1125
1080P50	50	Progressive Scan	1125
1080P30	30	Progressive Scan	1125
1080P25	25	Progressive Scan	1125
1080P24	24	Progressive Scan	1125
1080I60	60	Interlaced Scan	1125
1080I50	50	Interlaced Scan	1125

Return Format

The query returns PALS, NTSC, 480P, 576P, 720P60, 720P50, 720P30, 720P25, 720P24, 1080P60, 1080P50, 1080P30, 1080P25, 1080P24, 1080I60, or 1080I50.

Example

```
:TRIGger:VIDeo:STANdard NTSC    /*Selects NTSC as the video
standard.*/
:TRIGger:VIDeo:STANdard?         /*The query returns NTSC.*/
```

3.27.31 :TRIGger:VIDeo:LEVel

Syntax

:TRIGger:VIDeo:LEVel <*level*>

:TRIGger:VIDeo:LEVel?

Description

Sets or queries the trigger level of Video trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:VIDeo:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:VIDeo:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.32 :TRIGger:PATTern:PATTern

Syntax

:TRIGger:PATTern:PATTern <*pch1*>[,<*pch2*>[,<*pch3*>[,<*pch4*>]]]

:TRIGger:PATTern:PATTern?

Description

Sets or queries the channel pattern of Pattern trigger.

Parameter

Name	Type	Range	Default
<pch1>	Discrete	{H|L|X|R|F}	X
<pch2>	Discrete	{H|L|X|R|F}	X
<pch3>	Discrete	{H|L|X|R|F}	X
<pch4>	Discrete	{H|L|X|R|F}	X

Remarks

- The parameter "<pch1> to <pch4>" sets the patterns of the analog channels "CH1 to CH4".

- In the parameter range, H indicates high level (higher than the threshold level of the channel), L indicates low level (lower than the threshold level of the channel), and X indicates omitting the channel (This channel is not used as a part of the pattern. When all the channels are set to X, the oscilloscope will not trigger.) R indicates rising edge, and F indicates falling edge.
- In the pattern, you can only specify one edge (rising edge or falling edge). If one edge item is currently defined and then another edge item is defined in another channel in the pattern, then a prompt message "Invalid input" is displayed. Then, the latter defined edge item will be replaced by X.

Return Format

The query returns the currently set pattern of all the channels. The channels are separated by commas.

Example

```
:TRIGger:PATTern:PATTern H,R,L,X  /*Sets the patterns of "CH1 to
CH4" to H,R,L,X. Other channel pattern remains unchanged.*/
:TRIGger:PATTern:PATTern? /*The query returns H,R,L,X.*/
```

3.27.33 :TRIGger:PATTern:SOURce

Syntax

:TRIGger:PATTern:SOURce <*source*>

:TRIGger:PATTern:SOURce?

Description

Sets or queries the trigger source of Pattern trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:PATTern:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:PATTern:SOURce?             /*The query returns CHAN2.*/
```

3.27.34 :TRIGger:DURation:SOURce

Syntax

:TRIGger:DURation:SOURce <*source*>

:TRIGger:DURation:SOURce?

Description

Sets or queries the trigger source of Duration trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DURation:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:DURation:SOURce?             /*The query returns CHAN2.*/
```

3.27.35 :TRIGger:PATTern:LEVel

Syntax

:TRIGger:PATTern:LEVel <*source*>,<*level*>

:TRIGger:PATTern:LEVel? <*level*>

Description

Sets or queries the trigger level of the specified channel in Pattern trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1
<level>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:PATTern:LEVel CHANnel2,0.16  /*Sets the trigger level of
CH2 to 160 mV.*/
:TRIGger:PATTern:LEVel? CHANnel2     /*The query returns
1.600000E-1.*/
```

3.27.36 :TRIGger:DURation:TYPE

Syntax

:TRIGger:DURation:TYPE <*pch1*>[,<*pch2*>[,<*pch3*>[,<*pch4*>]]]]]

:TRIGger:DURation:TYPE?

Description

Sets or queries the channel pattern of Duration trigger.

Parameter

Name	Type	Range	Default
<pch1>	Discrete	{H|L|X}	X
<pch2>	Discrete	{H|L|X}	X
<pch3>	Discrete	{H|L|X}	X
<pch4>	Discrete	{H|L|X}	X

Remarks

- The parameter "<pch1> to <pch4>" sets the patterns of the analog channels "CH1 to CH4".
- In the parameter range, H indicates high level (higher than the threshold level of the channel), L indicates low level (lower than the threshold level of the channel), and X indicates omitting the channel (This channel is not used as a part of the pattern. When all the channels are set to X, the oscilloscope will not trigger.)

Return Format

The query returns the currently set pattern of all the channels. The channels are separated by commas.

Example

```
:TRIGger:DURation:TYPE L,X,H,L /*Sets the patterns of "CH1 to CH4"
to L,X,H,L.*/
:TRIGger:DURation:TYPE? /*The query returns L,X,H,L.*/
```

3.27.37 :TRIGger:DURation:WHEN

Syntax

:TRIGger:DURation:WHEN <*when*>

:TRIGger:DURation:WHEN?

Description

Sets or queries the trigger condition of Duration trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{GREater|LESS|GLESs|UNGLess}	GREater

Remarks

- **GREater:** triggers when the set duration time of the pattern is greater than the preset time.
- **LESS:** triggers when the set duration time of the pattern is smaller than the preset time.

- **GLESs:** triggers when the set duration time of the pattern is smaller than the preset upper time limit and greater than the preset lower time limit.
- **UNGLess:** triggers when the set duration time of the pattern is greater than the preset upper time limit and smaller than the preset lower time limit.

Return Format

The query returns GRE, LESS, GLES, or UNGL.

Example

```
:TRIGger:DURation:WHEN LESS     /*Sets the trigger condition to
LESS.*/
:TRIGger:DURation:WHEN?         /*The query returns LESS.*/
```

3.27.38 :TRIGger:DURation:TUPPer

Syntax

:TRIGger:DURation:TUPPer <*time*>

:TRIGger:DURation:TUPPer?

Description

Sets or queries the upper limit of the duration time of Duration trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to 10 s	1 μs

Remarks

This command is only available when the trigger conditions are LESS and GLESs.

Return Format

The query returns the upper limit of the duration time in scientific notation.

Example

```
:TRIGger:DURation:TUPPer 0.000003   /*Sets the upper limit of the
duration time to 3 μs.*/
:TRIGger:DURation:TUPPer?           /*The query returns
3.000000E-6.*/
```

3.27.39 :TRIGger:DURation:TLOWer

Syntax

:TRIGger:DURation:TLOWer <*time*>

:TRIGger:DURation:TLOWer?

Description

Sets or queries the lower limit of the duration time of Duration trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to 10s	1 μs

Remarks

This command is only available when the trigger conditions are GREater and GLESs.

Return Format

The query returns the lower limit of the duration time in scientific notation.

Example

```
:TRIGger:DURation:TLOWer 0.000003   /*Sets the lower limit of the
duration time to 3 μs.*/
:TRIGger:DURation:TLOWer?           /*The query returns
3.000000E-6.*/
```

3.27.40 :TRIGger:DURation:LEVel

Syntax

:TRIGger:DURation:LEVel <*source*>,<*level*>

:TRIGger:DURation:LEVel?<*source*>

Description

Sets or queries the trigger level of the specified channel in Duration trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:DURation:LEVel CHANnel2,0.16  /*Sets the trigger level of
CH2 to 160 mV.*/
:TRIGger:DURation:LEVel? CHANnel2     /*The query returns
1.600000E-1.*/
```

3.27.41 :TRIGger:TIMeout:SOURce

Syntax

:TRIGger:TIMeout:SOURce <*source*>

:TRIGger:TIMeout:SOURce?

Description

Sets or queries the trigger source of Timeout trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:TIMeout:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:TIMeout:SOURce?             /*The query returns CHAN2.*/
```

3.27.42 :TRIGger:TIMeout:SLOPe

Syntax

:TRIGger:TIMeout:SLOPe <*slope*>

:TRIGger:TIMeout:SLOPe?

Description

Sets or queries the edge type of Timeout trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative|RFALl}	POSitive

Remarks

- **POSitive:** starts timing when the rising edge of the input signal passes through the trigger level.
- **NEGative:** starts timing when the falling edge of the input signal passes through the trigger level.
- **RFALl:** starts timing when any edge of the input signal passes through the trigger level.

Return Format

The query returns POS, NEG, or RFAL.

Example

```
:TRIGger:TIMeout:SLOPe NEGative     /*Sets the edge type to
NEGative.*/
:TRIGger:TIMeout:SLOPe?             /*The query returns NEG.*/
```

3.27.43 :TRIGger:TIMeout:TIME

Syntax

:TRIGger:TIMeout:TIME <*time*>

:TRIGger:TIMeout:TIME?

Description

Sets or queries the timeout value of Timeout trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	16 ns to 10 s	1 μs

Remarks

N/A

Return Format

The query returns the timeout value in scientific notation.

Example

```
:TRIGger:TIMeout:TIME 0.002    /*Sets the timeout value to 2 ms.*/
:TRIGger:TIMeout:TIME?           /*The query returns 2.000000E-3.*/
```

3.27.44 :TRIGger:TIMeout:LEVel

Syntax

:TRIGger:TIMeout:LEVel <*level*>

:TRIGger:TIMeout:LEVel?

Description

Sets or queries the trigger level of Timeout trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:TIMeout:LEVel 0.16    /*Sets the trigger level to 160 mV.*/
:TRIGger:TIMeout:LEVel?        /*The query returns 1.600000E-1.*/
```

3.27.45 :TRIGger:RUNT:SOURce

Syntax

:TRIGger:RUNT:SOURce <*source*>

:TRIGger:RUNT:SOURce?

Description

Sets or queries the trigger source of Runt trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:RUNT:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:RUNT:SOURce?             /*The query returns CHAN2.*/
```

3.27.46 :TRIGger:RUNT:POLarity

Syntax

:TRIGger:RUNT:POLarity <*polarity*>

:TRIGger:RUNT:POLarity?

Description

Sets or queries the polarity of Runt trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

- **POSitive:** indicates the positive polarity. The oscilloscope triggers on the positive polarity of Runt trigger.
- **NEGative:** indicates the negative polarity. The oscilloscope triggers on the negative polarity of Runt trigger.

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:RUNT:POLarity NEGative     /*Sets the polarity of Runt
trigger to NEGative.*/
:TRIGger:RUNT:POLarity?             /*The query returns NEG.*/
```

3.27.47 :TRIGger:RUNT:WHEN

Syntax

:TRIGger:RUNT:WHEN <*when*>

:TRIGger:RUNT:WHEN?

Description

Sets or queries the trigger condition of Runt trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{NONE|GREater|LESS|GLESs}	NONE

Remarks

- **NONE:** indicates not setting the trigger condition of Runt trigger.
- **GREater:** triggers when the runt pulse width is greater than the lower limit of pulse width.
- **LESS:** triggers when the runt pulse width is smaller than the upper limit of pulse width.
- **GLESs:** triggers when the runt pulse width is greater than the lower limit and smaller than the upper limit of pulse width.
- The lower limit of the pulse width must be smaller than the upper limit.

Return Format

The query returns NONE, GRE, LESS, or GLES.

Example

```
:TRIGger:RUNT:WHEN LESS    /*Sets the trigger condition to LESS.*/
:TRIGger:RUNT:WHEN?        /*The query returns LESS.*/
```

3.27.48 :TRIGger:RUNT:WUPPer

Syntax

:TRIGger:RUNT:WUPPer <*width*>

:TRIGger:RUNT:WUPPer?

Description

Sets or queries the upper limit of the pulse width of Runt trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<width>	Real	100 ps to 10s	2 μs

Remarks

- This command is only available when the qualifier is LESS or GLESs.
- When the qualifier is set to "GLESs", the upper limit of the pulse width must be greater than the lower limit of the pulse width.

Return Format

The query returns the upper limit of the pulse width in scientific notation.

Example

```
:TRIGger:RUNT:WUPPer 0.02   /*Sets the upper limit of the pulse
width to 20 ms.*/
:TRIGger:RUNT:WUPPer?           /*The query returns 2.000000E-2.*/
```

3.27.49 :TRIGger:RUNT:WLOWer

Syntax

:TRIGger:RUNT:WLOWer <*width*>

:TRIGger:RUNT:WLOWer?

Description

Sets or queries the lower limit of the pulse width of Runt trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<width>	Real	8 ns to 100ps	1μs

Remarks

- This command is only available when the qualifier is GREater or GLESs.
- When the qualifier is set to "GLESs", the lower limit of the pulse width must be smaller than the upper limit of the pulse width.

Return Format

The query returns the lower limit of the pulse width in scientific notation.

Example

```
:TRIGger:RUNT:WLOWer 0.01   /*Sets the lower limit of the pulse
width to 10 ms.*/
:TRIGger:RUNT:WLOWer?           /*The query returns 1.000000E-2.*/
```

3.27.50 :TRIGger:RUNT:ALEVel

Syntax

`:TRIGger:RUNT:ALEVel` <*level*>

`:TRIGger:RUNT:ALEVel?`

Description

Sets or queries the upper limit of the trigger level of Runt trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Lower limit to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the upper limit of the trigger level in scientific notation.

Example

```
:TRIGger:RUNT:ALEVel 0.16   /*Sets the upper limit of the trigger
level to 160 mV.*/
:TRIGger:RUNT:ALEVel?      /*The query returns 1.600000E-1.*/
```

3.27.51 :TRIGger:RUNT:BLEVel

Syntax

:TRIGger:RUNT:BLEVel <*level*>

:TRIGger:RUNT:BLEVel?

Description

Sets or queries the lower limit of the trigger level of Runt trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to upper limit	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the lower limit of the trigger level in scientific notation.

Example

```
:TRIGger:RUNT:BLEVel 0.16   /*Sets the lower limit of the trigger
level to 160 mV.*/
:TRIGger:RUNT:BLEVel?      /*The query returns 1.600000E-1.*/
```

3.27.52 :TRIGger:WINDows:SOURce

Syntax

:TRIGger:WINDows:SOURce <*source*>

:TRIGger:WINDows:SOURce?

Description

Sets or queries the trigger source of Window trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:WINDows:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:WINDows:SOURce?             /*The query returns CHAN2.*/
```

3.27.53 :TRIGger:WINDows:SLOPe

Syntax

:TRIGger:WINDows:SLOPe <*type*>

:TRIGger:WINDows:SLOPe?

Description

Sets or queries the window type of Window trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	{POSitive|NEGative|RFALl}	POSitive

Remarks

- **POSitive:** triggers on the rising edge of the input signal when the voltage level is greater than the preset high trigger level.
- **NEGative:** triggers on the falling edge of the input signal when the voltage level is smaller than the preset low trigger level.
- **RFALl:** triggers on either the rising or falling edge of the input signal when the voltage level meets the preset trigger level.

Return Format

The query returns POS, NEG, or RFAL.

Example

```
:TRIGger:WINDows:SLOPe NEGative     /*Sets the window type to
NeGative.*/
:TRIGger:WINDows:SLOPe?             /*The query returns NEG.*/
```

3.27.54 :TRIGger:WINDows:POSition

Syntax

:TRIGger:WINDows:POSition <*pos*>

:TRIGger:WINDows:POSition?

Description

Sets or queries the trigger position of Window trigger.

Parameter

Name	Type	Range	Default
<pos>	Discrete	{EXIT|ENTer|TIME}	ENTer

Remarks

- **EXIT:** triggers when the input signal exits the specified trigger level range.
- **ENTer:** triggers when the input signal enters the specified trigger level range.
- **TIME:** triggers when the accumulated hold time after the trigger signal enters the specified trigger level range is equal to the window time.

Return Format

The query returns EXIT, ENT, or TIME.

Example

```
:TRIGger:WINDows:POSition ENTer     /*Sets the trigger position to
ENT.*/
:TRIGger:WINDows:POSition?           /*The query returns ENT.*/
```

3.27.55 :TRIGger:WINDows:TIME

Syntax

:TRIGger:WINDows:TIME <*time*>

:TRIGger:WINDows:TIME?

Description

Sets or queries the window time of Window trigger.

Parameter

Name	Type	Range	Default
<time>	Real	8 ns to 10s	1 μs

Remarks

N/A

Return Format

The query returns the window time in scientific notation.

Example

```
:TRIGger:WINDows:TIME 0.002    /*Sets the window time to 2 ms.*/
:TRIGger:WINDows:TIME?           /*The query returns 2.000000E-3.*/
```

3.27.56 :TRIGger:WINDows:ALEVel

Syntax

:TRIGger:WINDows:ALEVel <*level*>

:TRIGger:WINDows:ALEVel?

Description

Sets or queries the upper limit of the trigger level of Window trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Lower limit to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the upper limit of the trigger level in scientific notation.

Example

```
:TRIGger:WINDows:ALEVel 0.16   /*Sets the upper limit of the
trigger level to 160 mV.*/
:TRIGger:WINDows:ALEVel?      /*The query returns 1.600000E-1.*/
```

3.27.57 :TRIGger:WINDows:BLEVel

Syntax

:TRIGger:WINDows:BLEVel <*level*>

:TRIGger:WINDows:BLEVel?

Description

Sets or queries the lower limit of the trigger level of Window trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to upper limit	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the lower limit of the trigger level in scientific notation.

Example

```
:TRIGger:WINDows:BLEVel 0.05   /*Sets the lower limit of the
trigger level to 50 mV.*/
:TRIGger:WINDows:BLEVel?      /*The query returns 5.000000E-2.*/
```

3.27.58 :TRIGger:DELay:SA

Syntax

:TRIGger:DELay:SA <*source*>

:TRIGger:DELay:SA?

Description

Sets or queries the trigger source of Source A in Delay trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DELay:SA CHANnel2     /*Sets trigger source A to CHANnel2.*/
:TRIGger:DELay:SA?             /*The query returns CHAN2.*/
```

3.27.59 :TRIGger:DELay:SLOPA

Syntax

:TRIGger:DELay:SLOPA <*slope*>

:TRIGger:DELay:SLOPA?

Description

Sets or queries the edge type of Edge A in Delay trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:DELay:SLOPA NEGative     /*Sets the type of Edge A to
NEGative.*/
:TRIGger:DELay:SLOPA?             /*The query returns NEG.*/
```

3.27.60 :TRIGger:DELay:SB

Syntax

:TRIGger:DELay:SB <*source*>

:TRIGger:DELay:SB?

Description

Sets or queries the trigger source of Source B in Delay trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DELay:SB CHANnel4     /*Sets trigger source B to CHANnel4.*/
:TRIGger:DELay:SB?             /*The query returns CHAN4.*/
```

3.27.61 :TRIGger:DELay:SLOPB

Syntax

:TRIGger:DELay:SLOPB <*slope*>

:TRIGger:DELay:SLOPB?

Description

Sets or queries the edge type of Edge B in Delay trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:DELay:SLOPB NEGative     /*Sets the type of Edge B to
NEGative.*/
:TRIGger:DELay:SLOPB?             /*The query returns NEG.*/
```

3.27.62 :TRIGger:DELay:TYPE

Syntax

:TRIGger:DELay:TYPE <*type*>

:TRIGger:DELay:TYPE?

Description

Sets or queries the delay type of Delay trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	{GREater|LESS|GLESs|GOUT}	GREater

Remarks

- **GREater:** triggers when the time difference (ΔT) between the specified edges of Source A and Source B is greater than the preset time limit.
- **LESS:** triggers when the time difference (ΔT) between the specified edges of Source A and Source B is smaller than the preset time limit.
- **GLESs:** triggers when the time difference (ΔT) between the specified edges of Source A and Source B is greater than the lower limit of the preset time and smaller than the upper limit of the preset time.
- **GOUT:** triggers when the time difference (ΔT) between the specified edges of Source A and Source B is smaller than the lower limit of the preset time or greater than the upper limit of the preset time.

Return Format

The query returns GRE, LESS, GLES, or GOUT.

Example

```
:TRIGger:DELay:TYPe GOUT     /*Sets the delay type to GOUT.*/
:TRIGger:DELay:TYPe?          /*The query returns GOUT.*/
```

3.27.63 :TRIGger:DELay:TUPPer

Syntax

:TRIGger:DELay:TUPPer <*time*>

:TRIGger:DELay:TUPPer?

Description

Sets or queries the upper limit of delay time of the Delay trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to 10 s	2 μs

Remarks

This command is available when the delay type is LESS, GOUT, or GLESs.

Return Format

The query returns the upper limit of delay time in scientific notation.

Example

```
:TRIGger:DELay:TUPPer 0.002   /*Sets the upper limit of delay time
to 2 ms.*/
:TRIGger:DELay:TUPPer?        /*The query returns 2.000000E-3.*/
```

3.27.64 :TRIGger:DELay:TLOWer

Syntax

:TRIGger:DELay:TLOWer <*time*>

:TRIGger:DELay:TLOWer?

Description

Sets or queries the lower limit of delay time of the Delay trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	100 ps to 10 s	1μs

Remarks

This command is available when the delay type is GREater, GOUT, or GLESs.

Return Format

The query returns the lower limit of delay time in scientific notation.

Example

```
:TRIGger:DELay:TLOWer 0.002   /*Sets the lower limit of delay time
to 2 ms.*/
:TRIGger:DELay:TLOWer?        /*The query returns 2.000000E-3.*/
```

3.27.65 :TRIGger:DELay:ALEVel

Syntax

:TRIGger:DELay:ALEVel <*level*>

:TRIGger:DELay:ALEVel?

Description

Sets or queries the threshold level of Source A of Delay trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the threshold level of Source A in scientific notation.

Example

```
:TRIGger:DELay:ALEVel 0.16   /*Sets the threshold level of Source A
to 160 mV.*/
:TRIGger:DELay:ALEVel?      /*The query returns 1.600000E-1.*/
```

3.27.66 :TRIGger:DELay:BLEVel

Syntax

:TRIGger:DELay:BLEVel <*level*>

:TRIGger:DELay:BLEVel?

Description

Sets or queries the threshold level of Source B of Delay trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the threshold level of Source B in scientific notation.

Example

```
:TRIGger:DELay BLEVel 0.05   /*Sets the threshold level of Source B
to 50 mV.*/
:TRIGger:DELay:BLEVel?      /*The query returns 5.000000E-2.*/
```

3.27.67 :TRIGger:SHOLd:DSRC

Syntax

:TRIGger:SHOLd:DSRC <*source*>

:TRIGger:SHOLd:DSRC?

Description

Sets or queries the data source of Setup&Hold trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SHOLd:DSRC CHANnel1    /*Sets the data source to
CHANnel1.*/
:TRIGger:SHOLd:DSRC?             /*The query returns CHAN1.*/
```

3.27.68 :TRIGger:SHOLd:CSRC

Syntax

:TRIGger:SHOLd:CSRC <*source*>

:TRIGger:SHOLd:CSRC?

Description

Sets or queries the clock source of Setup&Hold trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SHOLd:CSRC CHANnel2    /*Sets the clock source to
CHANnel2.*/
:TRIGger:SHOLd:CSRC?             /*The query returns CHAN2.*/
```

3.27.69 :TRIGger:SHOLd:SLOPe

Syntax

:TRIGger:SHOLd:SLOPe <*slope*>

:TRIGger:SHOLd:SLOPe?

Description

Sets or queries the edge type of Setup&Hold trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:SHOLd:SLOPe NEGative    /*Sets the edge type to NEGative.*/
:TRIGger:SHOLd:SLOPe?            /*The query returns NEG.*/
```

3.27.70 :TRIGger:SHOLd:PATTern

Syntax

:TRIGger:SHOLd:PATTern <*pattern*>

:TRIGger:SHOLd:PATTern?

Description

Sets or queries the data type of Setup&Hold trigger.

Parameter

Name	Type	Range	Default
<pattern>	Discrete	{H|L}	H

Remarks

H: indicates high level.

L: indicates low level.

Return Format

The query returns H or L.

Example

```
:TRIGger:SHOLd:PATTern L    /*Sets data type to L.*/
:TRIGger:SHOLd:PATTern?    /*The query returns L.*/
```

3.27.71 :TRIGger:SHOLd:TYPE

Syntax

:TRIGger:SHOLd:TYPE <*type*>

:TRIGger:SHOLd:TYPE?

Description

Sets or queries the hold type of Setup&Hold trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	{SETup|HOLD|SETHold}	SETup

Remarks

- **SETup:** the oscilloscope triggers when the setup time is smaller than the specified setup time.
- **HOLd:** the oscilloscope triggers when the hold time is smaller than the specified hold time.
- **SETHold:** the oscilloscope triggers when the setup time or hold time is smaller than the specified time value.

Return Format

The query returns SET, HOLD, or SETH.

Example

```
:TRIGger:SHOLd:TYPE SETHOLd    /*Sets the hold type to SETHold.*/
:TRIGger:SHOLd:TYPE?           /*The query returns SETH.*/
```

3.27.72 :TRIGger:SHOLd:STIMe

Syntax

:TRIGger:SHOLd:STIMe <*time*>

:TRIGger:SHOLd:STIMe?

Description

Sets or queries the setup time of Setup&Hold trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	8 ns to 1s	1 μs

Remarks

- Setup time indicates the time that the data remain stable and unchanged before the clock signal of the trigger comes.
- This command is only available when the hold type is SETup or SETHOLd.

Return Format

The query returns the setup time value in scientific notation.

Example

```
:TRIGger:SHOLd:STIMe 0.002     /*Sets the setup time to 2 ms.*/
:TRIGger:SHOLd:STIMe?             /*The query returns 2.000000E-3.*/
```

3.27.73 :TRIGger:SHOLd:HTIMe

Syntax

:TRIGger:SHOLd:HTIMe <*time*>

:TRIGger:SHOLd:HTIMe?

Description

Sets or queries the hold time of Setup&Hold trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	8 ns to 1s	1 μs

Remarks

- Hold time indicates the time that the data remain stable and unchanged after the clock signal of the trigger comes.
- This command is only available when the hold type is HOLD or SETHOLd.

Return Format

The query returns the hold time value in scientific notation.

Example

```
:TRIGger:SHOLd:HTIMe 0.002    /*Sets the hold time to 2 ms.*/
:TRIGger:SHOLd:HTIMe?           /*The query returns 2.000000E-3.*/
```

3.27.74 :TRIGger:SHOLd:DLEVel

Syntax

:TRIGger:SHOLd:DLEVel <*level*>

:TRIGger:SHOLd:DLEVel?

Description

Sets or queries the trigger level of the data source. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level of the data source in scientific notation.

Example

```
:TRIGger:SHOLd:DLEVel 0.16    /*Sets the trigger level of the data
source to 160 mV.*/
:TRIGger:SHOLd:DLEVel?        /*The query returns 1.600000E-1.*/
```

3.27.75 :TRIGger:SHOLd:CLEVel

Syntax

:TRIGger:SHOLd:CLEVel<*level*>

:TRIGger:SHOLd:CLEVel?

Description

Sets or queries the trigger level of the clock source. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level of the clock source in scientific notation.

Example

```
:TRIGger:SHOLd:CLEVel 0.05    /*Sets the trigger level of the clock
source to 50 mV.*/
:TRIGger:SHOLd:CLEVel?        /*The query returns 5.000000E-2.*/
```

3.27.76 :TRIGger:NEDGe:SOURce

Syntax

:TRIGger:NEDGe:SOURce <*source*>

:TRIGger:NEDGe:SOURce?

Description

Sets or queries the trigger source of the Nth Edge trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:NEDGe:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:NEDGe:SOURce?             /*The query returns CHAN2.*/
```

3.27.77 :TRIGger:NEDGe:SLOPe

Syntax

:TRIGger:NEDGe:SLOPe <*slope*>

:TRIGger:NEDGe:SLOPe?

Description

Sets or queries the edge type of the Nth Edge trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

- **POSitive:** indicates that the oscilloscope triggers on the rising edge of the input signal when the voltage level meets the preset trigger level.
- **NEGative:** indicates that the oscilloscope triggers on the falling edge of the input signal when the voltage level meets the preset trigger level.

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:NEDGe:SLOPe NEGative     /*Sets the edge type to
NEGative .*/
:TRIGger:NEDGe:SLOPe?             /*The query returns NEG.*/
```

3.27.78 :TRIGger:NEDGe:IDLE

Syntax

:TRIGger:NEDGe:IDLE <*time*>

:TRIGger:NEDGe:IDLE?

Description

Sets or queries the idle time of the Nth Edge trigger. The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	16 ns to 10 s	1 μs

Remarks

N/A

Return Format

The query returns the idle time in scientific notation.

Example

```
:TRIGger:NEDGe:IDLE 0.002    /*Sets the idle time to 2 ms.*/
:TRIGger:NEDGe:IDLE?         /*The query returns 2.000000E-3.*/
```

3.27.79 :TRIGger:NEDGe:EDGE

Syntax

:TRIGger:NEDGe:EDGE <*edge*>

:TRIGger:NEDGe:EDGE?

Description

Sets or queries the number of edges of the Nth Edge trigger.

Parameter

Name	Type	Range	Default
<edge>	Integer	1 to 65535	1

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 65535.

Example

```
:TRIGger:NEDGe:EDGE 20     /*Sets the number of edges to 20.*/
:TRIGger:NEDGe:EDGE?       /*The query returns 20.*/
```

3.27.80 :TRIGger:NEDGe:LEVel

Syntax

:TRIGger:NEDGe:LEVel <*level*>

:TRIGger:NEDGe:LEVel?

Description

Sets or queries the trigger level of the Nth Edge trigger. The unit is the same as that of current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level of the Nth Edge trigger in scientific notation.

Example

```
:TRIGger:NEDGe:LEVel 0.16    /*Sets the trigger level to 160 mV.*/
:TRIGger:NEDGe:LEVel?        /*The query returns 1.600000E-1.*/
```

3.27.81 :TRIGger:RS232:SOURce (Option)

Syntax

:TRIGger:RS232:SOURce <*source*>

:TRIGger:RS232:SOURce?

Description

Sets or queries the trigger source of RS232 trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:RS232:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:RS232:SOURce?             /*The query returns CHAN2.*/
```

3.27.82 :TRIGger:RS232:LEVel (Option)

Syntax

:TRIGger:RS232:LEVel <*level*>

:TRIGger:RS232:LEVel?

Description

Sets or queries the trigger level of RS232 trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:RS232:LEVel 0.16    /*Sets the trigger level to 160 mV.*/
:TRIGger:RS232:LEVel?        /*The query returns 1.600000E-1.*/
```

3.27.83 :TRIGger:RS232:POLarity (Option)

Syntax

:TRIGger:RS232:POLarity <*polarity*>

:TRIGger:RS232:POLarity?

Description

Sets or queries the pulse polarity of RS232 trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:RS232:POLarity POSitive    /*Sets the pulse polarity of
RS232 trigger to POSitive.*/
:TRIGger:RS232:POLarity?           /*The query returns POS.*/
```

3.27.84 :TRIGger:RS232:WHEN (Option)

Syntax

`:TRIGger:RS232:WHEN` <*when*>

`:TRIGger:RS232:WHEN?`

Description

Sets or queries the trigger condition of RS232 trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{STARt|ERRor|CERRor|DATA}	STARt

Remarks

- **STARt:** triggers at the start of a frame.
- **ERRor:** triggers when an error frame is found.
- **CERRor:** triggers when a check error is found.
- **DATA:** triggers on the last bit of the preset data bits.

Return Format

The query returns STAR, ERR, CERR, or DATA.

Example

```
:TRIGger:RS232:WHEN ERRor    /*Sets the trigger condition to
ERRor.*/
:TRIGger:RS232:WHEN?         /*The query returns ERR.*/
```

3.27.85 :TRIGger:RS232:DATA (Option)

Syntax

:TRIGger:RS232:DATA <*data*>

:TRIGger:RS232:DATA?

Description

Sets or queries the data value of RS232 trigger when the trigger condition is "Data".

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 2^n-1	0

Remarks

In the expression 2^n-1, n indicates the current data width, and its available value can be 5, 6, 7, or 8.

Return Format

The query returns an integer.

Example

```
:TRIGger:RS232:DATA 10    /*Sets the data value to 10.*/
:TRIGger:RS232:DATA?      /*The query returns 10.*/
```

3.27.86 :TRIGger:RS232:BAUD (Option)

Syntax

:TRIGger:RS232:BAUD <*baud*>

:TRIGger:RS232:BAUD?

Compatible Command Syntax

:TRIGger:RS232:BUSer <*baud*>

:TRIGger:RS232:BUSer?

Description

Sets or queries the baud rate of RS232 trigger. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Integer	1 bps to 20 Mbps	9600 bps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 1 bps to 20 Mbps.

Example

```
:TRIGger:RS232:BAUD 4800     /*Sets the baud rate to 4800 bps.*/
:TRIGger:RS232:BAUD?         /*The query returns 4800.*/
```

Compatible Command Example

```
:TRIGger:RS232:BUSer 4800     /*Sets the baud rate to 4800 bps.*/
:TRIGger:RS232:BUSer?         /*The query returns 4800.*/
```

3.27.87 :TRIGger:RS232:WIDTh (Option)

Syntax

:TRIGger:RS232:WIDTh <*width*>

:TRIGger:RS232:WIDTh?

Description

Sets or queries the data width of RS232 trigger when the trigger condition is "Data".

Parameter

Name	Type	Range	Default
<width>	Discrete	{5|6|7|8}	8

Remarks

N/A

Return Format

The query returns 5, 6, 7, or 8.

Example

```
:TRIGger:RS232:WIDTh 6     /*Sets the data width to 6.*/
:TRIGger:RS232:WIDTh?       /*The query returns 6.*/
```

3.27.88 :TRIGger:RS232:STOP (Option)

Syntax

:TRIGger:RS232:STOP <*bit*>

:TRIGger:RS232:STOP?

Description

Sets or queries the stop bits of RS232 trigger.

Parameter

Name	Type	Range	Default
<bit>	Discrete	{1|1.5|2}	1

Remarks

N/A

Return Format

The query returns 1, 1.5, or 2.

Example

```
:TRIGger:RS232:STOP 2    /*Sets the stop bits to 2.*/
:TRIGger:RS232:STOP?    /*The query returns 2.*/
```

3.27.89 :TRIGger:RS232:PARity (Option)

Syntax

:TRIGger:RS232:PARity <*parity*>

:TRIGger:RS232:PARity?

Description

Sets or queries the check mode of RS232 trigger.

Parameter

Name	Type	Range	Default
<parity>	Discrete	{EVEN|ODD|NONE}	NONE

Remarks

N/A

Return Format

The query returns EVEN, ODD, or NONE.

Example

```
:TRIGger:RS232:PARity EVEN    /*Sets the check mode to EVEN.*/
:TRIGger:RS232:PARity?         /*The query returns EVEN.*/
```

3.27.90 :TRIGger:IIC:SCL (Option)

Syntax

:TRIGger:IIC:SCL <*source*>

:TRIGger:IIC:SCL?

Description

Sets or queries the source channel of the clock line of I2C trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIC:SCL CHANnel2    /*Sets the clock source to CHANnel2.*/
:TRIGger:IIC:SCL?             /*The query returns CHAN2.*/
```

3.27.91 :TRIGger:IIC:LEVel (Option)

Syntax

:TRIGger:IIC:CLEVel <*level*>

:TRIGger:IIC:CLEVel?

Description

Sets or queries the trigger level of the clock line in I2C trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:IIC:CLEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:IIC:CLEVel?         /*The query returns 1.600000E-1.*/
```

3.27.92 :TRIGger:IIC:SDA (Option)

Syntax

:TRIGger:IIC:SDA <*source*>

:TRIGger:IIC:SDA?

Description

Sets or queries the source channel of the data line of I2C trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIC:SDA CHANnel2    /*Sets the data source to CHANnel2.*/
:TRIGger:IIC:SDA?             /*The query returns CHAN2.*/
```

3.27.93 :TRIGger:IIC:DLEVel (Option)

Syntax

:TRIGger:IIC:DLEVel <*level*>

:TRIGger:IIC:DLEVel?

Description

Sets or queries the trigger level of the data line in I2C trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:IIC:DLEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:IIC:DLEVel?         /*The query returns 1.600000E-1.*/
```

3.27.94 :TRIGger:IIC:WHEN (Option)

Syntax

:TRIGger:IIC:WHEN <*when*>

:TRIGger:IIC:WHEN?

Description

Sets or queries the trigger condition of I2C trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{STARt|RESTart|STOP|NACKnowledge|ADDRess|DATA|ADATa}	STARt

Remarks

- **STARt:** indicates that the oscilloscope triggers when SCL is high level and SDA transitions from high level to low level.
- **RESTart:** indicates that the oscilloscope triggers when another start condition occurs before a stop condition.
- **STOP:** indicates that the oscilloscope triggers when SCL is high level and SDA transitions from low level to high level.
- **NACKnowledge:** indicates missing acknowledgment. The oscilloscope triggers when SDA is high level during the acknowledgment of the SCL bit.
- **ADDRess:** indicates that the oscilloscope searches for the specified address and triggers on the read/write bit.
- **DATA:** indicates that the oscilloscope searches for the specified data on the data line (SDA) and triggers on the clock line (SCL) of the jumping edge of the last bit of the data.
- **ADATa:** indicates that the oscilloscope searches for the specified address and data, and then triggers when both the address and data meet the conditions.

Return Format

The query returns STAR, REST,STOP, NACK, ADDR, DATA, or ADAT.

Example

```
:TRIGger:IIC:WHEN RESTart    /*Sets the trigger condition to
RESTart.*/
:TRIGger:IIC:WHEN?          /*The query returns REST.*/
```

3.27.95 :TRIGger:IIC:AWIDth (Option)

Syntax

:TRIGger:IIC:AWIDth <*bits*>

:TRIGger:IIC:AWIDth?

Description

Sets or queries the address width of I2C trigger when the trigger condition is "ADDRess" or "ADATa".

Parameter

Name	Type	Range	Default
<bits>	Discrete	{7|8|10}	7

Remarks

N/A

Return Format

The query returns 7, 8, or 10.

Example

```
:TRIGger:IIC:AWIDth 10     /*Sets the address width to 10.*/
:TRIGger:IIC:AWIDth?       /*The query returns 10.*/
```

3.27.96 :TRIGger:IIC:ADDRess (Option)

Syntax

:TRIGger:IIC:ADDRess <*address*>

:TRIGger:IIC:ADDRess?

Description

Sets or queries the address of I2C trigger when the trigger condition is "ADDRess" or "ADATa".

Parameter

Name	Type	Range	Default
<address>	Integer	0 to 2^n-1	0

Remarks

In the expression 2^n-1, n indicates the current address width. Its range is from 0 to 127, 0 to 255, or 0 to 1,023.

Return Format

The query returns the address in integer.

Example

```
:TRIGger:IIC:ADDRess 100     /*Sets the address to 100.*/
:TRIGger:IIC:ADDRess?        /*The query returns 100.*/
```

3.27.97 :TRIGger:IIC:DIRection (Option)

Syntax

:TRIGger:IIC:DIRection <*direction*>

:TRIGger:IIC:DIRection?

Description

Sets or queries the data direction of I2C trigger when the trigger condition is "ADDRess" or "ADATa".

Parameter

Name	Type	Range	Default
<dir>	Discrete	{READ|WRITe|RWRite}	WRITe

Remarks

This command is unavailable when the address width is set to 8.

Return Format

The query returns READ, WRIT, or RWR.

Example

```
:TRIGger:IIC:DIRection RWRite     /*Sets the data direction to
RWRite.*/
:TRIGger:IIC:DIRection?           /*The query returns RWR.*/
```

3.27.98 :TRIGger:IIC:DBYTes (Option)

Syntax

:TRIGger:IIC:DBYTes <*bytes*>

:TRIGger:IIC:DBYTes?

Description

Sets or queries of the data bytes of I2C trigger when the trigger condition is "DATA" or "ADATa".

Parameter

Name	Type	Range	Default
<bytes>	Real	1 to 5	1

Remarks

N/A

Return Format

The query returns the data bytes in scientific notation.

Example

```
:TRIGger:IIC:DBYTes 3    /*Sets the data bytes to 3 when the
trigger condition is "DATA" or "ADATa".*/
:TRIGger:IIC:DBYTes?        /*The query returns 3.*/
```

3.27.99 :TRIGger:IIC:DATA (Option)

Syntax

:TRIGger:IIC:DATA <*data*>

:TRIGger:IIC:DATA?

Description

Sets or queries the data value of I2C trigger when the trigger condition is "DATA" or "ADATa".

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 2^{40}-1	0

Remarks

The settable range of <data> is affected by the byte length. The maximum byte length can be set to 5, i.g. 40-bit binary data. Therefore, the range of <data> is from 0 to 2^{40}-1.

Return Format

The query returns an integer. (Option)

Example

```
:TRIGger:IIC:DATA 64    /*Sets the data value to 64.*/
:TRIGger:IIC:DATA?      /*The query returns 64.*/
```

3.27.100 :TRIGger:IIC:CURRbit (Option)

Syntax

:TRIGger:IIC:CURRbit <*currbit*>

:TRIGger:IIC:CURRbit?

Description

Sets or queries the current bit of the I2C trigger data.

Parameter

Name	Type	Range	Default
<currbit>	Integer	0 to 39	0

Remarks

After configuring the settings for this command, you can send the *:TRIGger:IIC:CODE (Option)* command to set or modify the set bit data.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:IIC:CURRbit 8    /*Sets the current bit of I2C trigger
data to 8.*/
:TRIGger:IIC:CURRbit?      /*The query returns 8.*/
```

3.27.101 :TRIGger:IIC:CODE (Option)

Syntax

:TRIGger:IIC:CODE <*code*>

:TRIGger:IIC:CODE?

Description

Sets or queries the data value of a certain bit of I2C trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

After sending the *:TRIGger:IIC:CURRbit (Option)* command to set the specified bit, you can send this command to query or modify the value of the specified data bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:IIC:CODE 0    /*Sets the data value to 0.*/
:TRIGger:IIC:CODE?      /*The query returns 0.*/
```

3.27.102 :TRIGger:SPI:CLK (Option)

Syntax

:TRIGger:SPI:CLK <*source*>

:TRIGger:SPI:CLK?

Compatible Command Syntax

:TRIGger:SPI:SCL <*source*>

:TRIGger:SPI:SCL?

Description

Sets or queries source channel the of data line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:CLK CHANnel3   /*Sets the source channel of the data
line of SPI trigger to CHANnel3.*/
:TRIGger:SPI:CLK?           /*The query returns CHAN3.*/
```

Compatible Command Example

```
:TRIGger:SPI:SCL CHANnel1   /*Sets the channel source of the clock
line to CHANnel1.*/
:TRIGger:SPI:SCL?           /*The query returns CHAN1.*/
```

3.27.103 :TRIGger:SPI:CLEVel (Option)

Syntax

:TRIGger:SPI:CLEVel <*level*>

:TRIGger:SPI:CLEVel?

Description

Sets or queries the trigger level of the clock channel of SPI trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:SPI:CLEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:SPI:CLEVel?         /*The query returns 1.600000E-1.*/
```

3.27.104 :TRIGger:SPI:SLOPe (Option)

Syntax

:TRIGger:SPI:SLOPe <*slope*>

:TRIGger:SPI:SLOPe?

Description

Sets or queries the type of the clock edge of SPI trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative}	POSitive

Remarks

- **POSitive:** samples the SDA data on the rising edge of the clock.
- **NEGative:** samples the SDA data on the falling edge of the clock.

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:SPI:SLOPe POSitive    /*Sets the clock edge to POSitive.*/
:TRIGger:SPI:SLOPe?           /*The query returns POS.*/
```

3.27.105 :TRIGger:SPI:MISO (Option)

Syntax

:TRIGger:SPI:MISO <*source*>

:TRIGger:SPI:MISO?

Compatible Command Syntax

:TRIGger:SPI:SDA <*source*>

:TRIGger:SPI:SDA?

Description

Sets or queries the source channel of the data line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, and CHAN4.

Example

```
:TRIGger:SPI:MISO CHANnel3   /*Sets the source channel of the data
line of SPI trigger to CHANnel3.*/
:TRIGger:SPI:MISO?           /*The query returns CHAN3.*/
```

Compatible Command Example

```
:TRIGger:SPI:SDA CHANnel2    /*Sets the channel source of the data
line to CHANnel2.*/
:TRIGger:SPI:SDA?            /*The query returns CHAN2.*/
```

3.27.106 :TRIGger:SPI:DLEVel (Option)

Syntax

:TRIGger:SPI:DLEVel <*level*>

:TRIGger:SPI:DLEVel?

Description

Sets or queries the trigger level of the data channel of SPI trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:SPI:DLEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:SPI:DLEVel?         /*The query returns 1.600000E-1.*/
```

3.27.107 :TRIGger:SPI:WHEN (Option)

Syntax

:TRIGger:SPI:WHEN <*when*>

:TRIGger:SPI:WHEN?

Description

Sets or queries the trigger condition of SPI trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{CS|TIMeout}	CS

Remarks

When the trigger condition is "TIMeout", you can run the *:TRIGger:SPI:TIMeout (Option)* command to set the timeout value.

Return Format

The query returns CS or TIM.

Example

```
:TRIGger:SPI:WHEN TIMeout    /*Sets the trigger condition to
TIMeout.*/
:TRIGger:SPI:WHEN?           /*The query returns TIM.*/
```

3.27.108 :TRIGger:SPI:CS (Option)

Syntax

:TRIGger:SPI:CS <*source*>

:TRIGger:SPI:CS?

Description

Sets or queries the source channel of the CS line when the trigger condition of SPI is set to CS.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel3

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:CS CHANnel2   /*Sets the source channel of the CS line
to CHANnel2 when the trigger condition of SPI is set to CS.*/
:TRIGger:SPI:CS?           /*The query returns CHAN2.*/
```

3.27.109 :TRIGger:SPI:SLEVel (Option)

Syntax

:TRIGger:SPI:SLEVel <*level*>

:TRIGger:SPI:SLEVel?

Description

Sets or queries the trigger level of the CS channel of SPI trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:SPI:SLEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:SPI:SLEVel?         /*The query returns 1.600000E-1.*/
```

3.27.110 :TRIGger:SPI:MODE (Option)

Syntax

:TRIGger:SPI:MODE <*mode*>

:TRIGger:SPI:MODE?

Description

Sets or queries the CS mode of SPI trigger when the trigger condition is "CS".

Parameter

Name	Type	Range	Default
<mode>	Discrete	{HIGH|LOW}	LOW

Remarks

N/A

Return Format

The query returns HIGH or LOW.

Example

```
:TRIGger:SPI:MODE LOW   /*Sets the CS mode to LOW.*/
:TRIGger:SPI:MODE?       /*The query returns LOW.*/
```

3.27.111 :TRIGger:SPI:TIMeout (Option)

Syntax

:TRIGger:SPI:TIMeout <*time*>

:TRIGger:SPI:TIMeout?

Description

Sets or queries the timeout value when the trigger condition of SPI trigger is "Timeout". The default unit is s.

Parameter

Name	Type	Range	Default
<time>	Real	8 ns to 10 s	1 μs

Remarks

N/A

Return Format

The query returns the timeout value in scientific notation.

Example

```
:TRIGger:SPI:TIMeout 0.001    /*Sets the timeout value to 1 ms.*/
:TRIGger:SPI:TIMeout?            /*The query returns 1.000000E-3.*/
```

3.27.112 :TRIGger:SPI:WIDTh (Option)

Syntax

:TRIGger:SPI:WIDTh <*width*>

:TRIGger:SPI:WIDTh?

Description

Sets or queries the data width of data channel in SPI trigger.

Parameter

Name	Type	Range	Default
<width>	Integer	4 to 32	8

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:TRIGger:SPI:WIDTh 10    /*Sets the data width to 10.*/
:TRIGger:SPI:WIDTh?      /*The query returns 10.*/
```

3.27.113 :TRIGger:SPI:DATA (Option)

Syntax

:TRIGger:SPI:DATA <*data*>

:TRIGger:SPI:DATA?

Description

Sets or queries the data value of SPI trigger.

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 2^{32}-1	0

Remarks

The range of the parameter <data> is related to the current data width. The available maximum data width is 32. Therefore, the range of <data> is from 0 to 2^{32}-1.

Return Format

The query returns an integer.

Example

```
:TRIGger:SPI:DATA 5    /*Sets the data value to 5.*/
:TRIGger:SPI:DATA?      /*The query returns 5.*/
```

3.27.114 :TRIGger:SPI:CURRbit (Option)

Syntax

:TRIGger:SPI:CURRbit <*currbit*>

:TRIGger:SPI:CURRbit?

Description

Sets or queries the current bit of the SPI trigger data.

Parameter

Name	Type	Range	Default
<currbit>	Integer	0 to 39	0

Remarks

After configuring the settings for this command, you can send the *:TRIGger:SPI:CODE (Option)* command to set or modify the set bit data.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:SPI:CURRbit 8    /*Sets the current bit of SPI trigger to
8.*/
:TRIGger:SPI:CURRbit?      /*The query returns 8.*/
```

3.27.115 :TRIGger:SPI:CODE (Option)

Syntax

:TRIGger:SPI:CODE <*code*>

:TRIGger:SPI:CODE?

Description

Sets or queries the data value of a certain bit of SPI trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

After sending the *:TRIGger:SPI:CURRbit (Option)* command to set the specified bit, you can send this command to query or modify the value of the specified data bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:SPI:CODE 0    /*Sets the data value to 0.*/
:TRIGger:SPI:CODE?      /*The query returns 0.*/
```

3.27.116 :TRIGger:CAN:BAUD (Option)

Syntax

:TRIGger:CAN:BAUD <*baud*>

:TRIGger:CAN:BAUD?

Description

Sets or queries the signal rate of CAN trigger. The unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Integer	10 kbps to 5 Mbps	1 Mbps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 10 kbps to 5 Mbps.

Example

```
:TRIGger:CAN:BAUD 125000    /*Sets the signal rate to 125000 bps.*/
:TRIGger:CAN:BAUD?        /*The query returns 125000.*/
```

3.27.117 :TRIGger:CAN:SOURce (Option)

Syntax

:TRIGger:CAN:SOURce <*source*>

:TRIGger:CAN:SOURce?

Description

Sets or queries the trigger source of CAN trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:CAN:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:CAN:SOURce?             /*The query returns CHAN2.*/
```

3.27.118 :TRIGger:CAN:STYPe (Option)

Syntax

:TRIGger:CAN:STYPe <*stype*>

:TRIGger:CAN:STYPe?

Description

Sets or queries the signal type of CAN trigger.

Parameter

Name	Type	Range	Default
<stype>	Discrete	{H|L|RXTX|DIFFerential}	H

Remarks

- **H:** indicates the actual CAN_H differential bus signal.
- **L:** indicates the actual CAN_L differential bus signal.
- **RXTX:** indicates the Receive or Transmit signal from the CAN bus transceiver.
- **DIFFerential:** indicates the CAN differential bus signal connected to an analog channel by using a differential probe. Connect the differential probe's positive lead to the CAN_H bus signal and connect the negative lead to the CAN_L bus signal..

Return Format

The query returns H, L, RXTX, or DIFF.

Example

```
:TRIGger:CAN:STYPe L      /*Sets the signal type to CAN_L
differential bus signal.*/
:TRIGger:CAN:STYPe?       /*The query returns L.*/
```

3.27.119 :TRIGger:CAN:WHEN (Option)

Syntax

:TRIGger:CAN:WHEN <*cond*>

:TRIGger:CAN:WHEN?

Description

Sets or queries the trigger condition of CAN trigger.

Parameter

Name	Type	Range	Default
<cond>	Discrete	{SOF|EOF|IDRemote|OVERload| IDFRame|DATaframe|IDData| ERFRame|ERANswer|ERCHeck| ERFormat|ERRandom|ERBit}	SOF

Remarks

- **SOF:** indicates start of frame. It indicates that the oscilloscope triggers at the start of a data frame.
- **EOF:** indicates end of frame. It indicates that the oscilloscope triggers at the end of a data frame.

Frame Type

- **IDRemote:** indicates remote ID. It indicates that the oscilloscope triggers on the remote frame with the specified ID.
- **OVERload:** indicates overload frame. It indicates that the oscilloscope triggers on the CAN overload frames.
- **IDFRame:** indicates frame ID. It indicates that the oscilloscope triggers on the data frames with the specified ID.

- **DATAframe:** indicates frame data: It indicates that the oscilloscope triggers on the data frames with specified data.
- **IDData:** indicates Data & ID. It indicates that the oscilloscope triggers on the data frames with the specified ID and data.

Frame Error

- **ERFRame:** indicates frame error. It indicates that the oscilloscope triggers on the error frame.
- **ERANswer:** indicates answer error. It indicates that the oscilloscope triggers on the answer error frame.
- **ERCHeck:** indicates check error. It indicates that the oscilloscope triggers on the check error frame.
- **ERFormat:** indicates format error. It indicates that the oscilloscope triggers on the format error frame.
- **ERRandom:** indicates random error. It indicates that the oscilloscope triggers on the random error frame, such as the format error frame, answer error frame, etc.
- **ERBit:** indicates bit fill. It indicates that the oscilloscope triggers on the error frame with the bit fill.

Return Format

The query returns SOF, EOF, IDR, OVER, IDFR, DAT, IDD, ERFR, ERAN, ERCH, ERF, ERR, or ERBit.

Example

```
:TRIGger:CAN:WHEN EOF    /*Sets the trigger condition to EOF.*/
:TRIGger:CAN:WHEN?         /*The query returns EOF.*/
```

3.27.120 :TRIGger:CAN:SPOint (Option)

Syntax

:TRIGger:CAN:SPOint <*spoint*>

:TRIGger:CAN:SPOint?

Description

Sets or queries the sample point position of CAN trigger (expressed in %).

Parameter

Name	Type	Range	Default
<spoint>	Integer	10 to 90	50

Remarks

The sample point is within the range of the bit time. The oscilloscope samples the bit level at the sample point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:TRIGger:CAN:SPoint 60          /*Sets the sample point position of
CAN trigger to 60%.*/
:TRIGger:CAN:SPoint?            /*The query returns 60.*/
```

3.27.121 :TRIGger:CAN:EXTended (Option)

Syntax

:TRIGger:CAN:EXTended <*bool*>

:TRIGger:CAN:EXTended?

Description

Enables or disables the extended ID when the trigger condition of CAN trigger is "Remote ID" or "Frame ID"; queries whether the extended ID is enabled when the trigger condition of CAN trigger is "Remote ID" or "Frame ID".

Parameter

Name	Type	Range	Default
<bool>	Bool	{false|true}	false

Remarks

- **false:** disables the extended ID.
- **true:** enables the extended ID.

To set or query the trigger condition of CAN trigger, send the *:TRIGger:CAN:WHEN (Option)* command.

Return Format

The query returns true or false.

Example

```
:TRIGger:CAN:EXTended true  /*Enables the extended ID.*/
:TRIGger:CAN:EXTended?   /*The query returns true.*/
```

3.27.122 :TRIGger:CAN:DEFine (Option)

Syntax

:TRIGger:CAN:DEFine <*bool*>

:TRIGger:CAN:DEFine?

Description

Sets Define to ID or Data when the trigger condition of CAN trigger is set to Data or ID; queries Define to ID or Data when the trigger condition of CAN trigger is set to Data or ID.

Parameter

Name	Type	Range	Default
<bool>	Bool	{false|true}	false

Remarks

- **false:** sets Define to Data.
- **true:** sets Define to ID.

To set or query the trigger condition of CAN trigger, send the *:TRIGger:CAN:WHEN (Option)* command.

Return Format

The query returns true or false.

Example

```
:TRIGger:CAN:DEFine true /*Sets Define to ID.*/
:TRIGger:CAN:DEFine? /*The query returns true.*/
```

3.27.123 :TRIGger:CAN:DWIDth (Option)

Syntax

:TRIGger:CAN:DWIDth <*data*>

:TRIGger:CAN:DWIDth?

Description

Sets or queries the data width of CAN trigger when the trigger condition is "DATaframe" or "IDData".

Parameter

Name	Type	Range	Default
<data>	Integer	1 to 8	1

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 8.

Example

```
:TRIGger:CAN:DWIDth 5    /*Sets the data width of CAN trigger to 5
when the trigger condition is "DATaframe" or "IDData".*/
:TRIGger:CAN:DWIDth?      /*The query returns 5.*/
```

3.27.124 :TRIGger:CAN:DATA (Option)

Syntax

:TRIGger:CAN:DATA <*data*>

:TRIGger:CAN:DATA?

Description

Sets or queries the data value of CAN trigger.

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 2^{40}-1	0

Remarks

The settable range of <data> is affected by the byte length. The maximum byte length can be set to 5, i.e. 40-bit binary data. Therefore, the range of <data> is from 0 to 2^{40}-1.

Return Format

The query returns an integer. (Option)

Example

```
:TRIGger:CAN:DATA 100    /*Sets the data value of CAN trigger to
100.*/
:TRIGger:CAN:DATA?      /*The query returns 100.*/
```

3.27.125 :TRIGger:CAN:CURRbit (Option)

Syntax

:TRIGger:CAN:CURRbit <*currbit*>

:TRIGger:CAN:CURRbit?

Description

Sets or queries the current bit of the CAN trigger data.

Parameter

Name	Type	Range	Default
<currbit>	Integer	0 to 39	0

Remarks

After configuring the settings for this command, you can send the *:TRIGger:CAN:CODE (Option)* command to set or modify the set bit data.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:CAN:CURRbit 8    /*Sets the current bit of CAN trigger to
8.*/
:TRIGger:CAN:CURRbit?      /*The query returns 8.*/
```

3.27.126 :TRIGger:CAN:CODE (Option)

Syntax

:TRIGger:CAN:CODE <*code*>

:TRIGger:CAN:CODE?

Description

Sets or queries the data value of a certain bit of CAN trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

After sending the *:TRIGger:CAN:CURRbit (Option)* command to set the specified bit, you can send this command to query or modify the value of the specified data bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:CAN:CODE 0     /*Sets the data value to 0.*/
:TRIGger:CAN:CODE?       /*The query returns 0.*/
```

3.27.127 :TRIGger:CAN:LEVel (Option)

Syntax

:TRIGger:CAN:LEVel <*level*>

:TRIGger:CAN:LEVel?

Description

Sets or queries the trigger level of CAN trigger. Its unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:CAN:LEVel 0.16    /*Sets the trigger level to 160 mV.*/
:TRIGger:CAN:LEVel?        /*The query returns 1.600000E-1.*/
```

3.27.128 :TRIGger:FLEXray:SOURce (Option)

Syntax

:TRIGger:FLEXray:SOURce <*source*>

:TRIGger:FLEXray:SOURce?

Description

Sets or queries the trigger source of FlexRay trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:FLEXray:SOURce CHANnel2    /*Sets the trigger source of
FlexRay trigger to CHANnel2.*/
:TRIGger:FLEXray:SOURce?             /*The query returns CHAN2.*/
```

3.27.129 :TRIGger:FLEXray:LEVel (Option)

Syntax

:TRIGger:FLEXray:LEVel <*level*>

:TRIGger:FLEXray:LEVel?

Description

Sets or queries the trigger level of FlexRay trigger. Its unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:FLEXray:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:FLEXray:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.130 :TRIGger:FLEXray:BAUD (Option)

Syntax

:TRIGger:FLEXray:BAUD <*baud*>

:TRIGger:FLEXray:BAUD?

Description

Sets or queries the signal rate of FlexRay trigger. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Discrete	{2500000|5000000|10000000}	10000000

Remarks

N/A

Return Format

The query returns 2500000, 5000000, or 10000000.

Example

```
:TRIGger:FLEXray:BAUD 5000000       /*Sets the signal rate to
5000000 bps.*/
:TRIGger:FLEXray:BAUD?               /*The query returns 5000000.*/
```

3.27.131 :TRIGger:FLEXray:CH (Option)

Syntax

:TRIGger:FLEXray:CH <*ch*>

:TRIGger:FLEXray:CH?

Description

Sets or queries the channel of FlexRay trigger.

Parameter

Name	Type	Range	Default
<ch>	Discrete	{A|B}	A

Remarks

N/A

Return Format

The query returns A or B.

Example

```
:TRIGger:FLEXray:CH B          /*Sets the channel of FlexRay trigger
to Channel B.*/
:TRIGger:FLEXray:CH?                    /*The query returns B.*/
```

3.27.132 :TRIGger:FLEXray:WHEN (Option)

Syntax

:TRIGger:FLEXray:WHEN <*cond*>

:TRIGger:FLEXray:WHEN?

Description

Sets or queries the trigger condition of FLEXray trigger.

Parameter

Name	Type	Range	Default
<cond>	Discrete	{FRAMe|SYMBol|ERRor|TSS}	FRAMe

Remarks

- **FRAMe:** triggers on the frame of the FlexRay bus.

- **SYMBol:** triggers on the Channel Idle Delimiter (CID), Collision Avoidance Symbol (CAS), Media Access Test Symbol (MTS), and Wakeup Pattern (WUP) of the FlexRay bus.
- **ERRor:** triggers when an error occurs to the FlexRay bus, including Head CRC Err, Tail CRC Err, Decode Err, and Random Err.
- **TSS:** triggers on the transmission start sequence of the FlexRay bus.

Return Format

The query returns FRAM, SYMB, ERR, or TSS.

Example

```
:TRIGger:FLEXray:WHEN TSS          /*Sets the trigger condition to
TSS.*/
:TRIGger:FLEXray:WHEN?                    /*The query returns TSS.*/
```

3.27.133 :TRIGger:FLEXray:POS (Option)

Syntax

:TRIGger:FLEXray:POS <*pos*>

:TRIGger:FLEXray:POS?

Description

Sets or queries the position of FlexRay trigger when the trigger condition is Position.

Parameter

Name	Type	Range	Default
<pos>	Discrete	{TSS|FSS|FES|DTS}	TSS

Remarks

- **TSS:** Transmit Start Sequence.
- **FSS:** Frame Start Sequence.
- **FES:** Frame End Sequence.
- **DTS:** Dynamic Trailing Sequence.

Return Format

The query returns TSS, FSS, FES, or DTS.

Example

```
:TRIGger:FLEXray:POS TSS        /*Sets the position of FlexRay
trigger to TSS when the trigger condition is Position.*/
:TRIGger:FLEXray:POS?                /*The query returns TSS.*/
```

3.27.134 :TRIGger:FLEXray:FRAMe (Option)

Syntax

:TRIGger:FLEXray:FRAMe? <*frame*>

:TRIGger:FLEXray:FRAMe?

Description

Sets or queries the frame type of FlexRay trigger.

Parameter

Name	Type	Range	Default
<frame>	Discrete	{NULL|SYNC|STAR|ANY}	NULL

Remarks

N/A

Return Format

The query returns NULL, SYNC, STAR, or ANY.

Example

```
:TRIGger:FLEXray:FRAMe SYNC        /*Sets the frame type of FlexRay
trigger to SYNC.*/
:TRIGger:FLEXray:FRAMe?                /*The query returns SYNC.*/
```

3.27.135 :TRIGger:FLEXray:SYMBol (Option)

Syntax

:TRIGger:FLEXray:SYMBol <*symbol*>

:TRIGger:FLEXray:SYMBol?

Description

Sets or queries the symbol type of FlexRay trigger when the trigger condition is set to Symbol.

Parameter

Name	Type	Range	Default
<symbol>	Integer	{CAS|WUS}	CAS

Remarks

N/A

Return Format

The query returns CAS or WUS.

Example

```
:TRIGger:FLEXray:SYMBol WUS        /*Sets the symbol type of
FlexRay trigger to WUS when the trigger condition is set to
Symbol.*/
:TRIGger:FLEXray:SYMBol?                  /*The query returns WUS.*/
```

3.27.136 :TRIGger:FLEXray:ERRor (Option)

Syntax

:TRIGger:FLEXray:ERRor <*err*>

:TRIGger:FLEXray:ERRor?

Description

Sets or queries the error type of FlexRay trigger when the trigger condition is Error.

Parameter

Name	Type	Range	Default
<err>	Discrete	{HEAD|TRAiler|DECode|ANY}	HEAD

Remarks

N/A

Return Format

The query returns HEAD, TRAiler, DECode, or ANY.

Example

```
:TRIGger:FLEXray:ERRor DECode        /*Sets the error type of
FlexRay trigger to DECode when the trigger condition is Error.*/
:TRIGger:FLEXray:ERRor?                 /*The query DECode.*/
```

3.27.137 :TRIGger:FLEXray:DEFine (Option)

Syntax

:TRIGger:FLEXray:DEFine? <*bool*>

:TRIGger:FLEXray:DEFine?

Description

Sets or queries to use ID or Cyc Count to define the trigger condition of FlexRay trigger when the trigger condition is set to Frame.

Parameter

Name	Type	Range	Default
<bool>	Bool	{false|true}	false

Remarks

- **true:** indicates Cyc Count.
- **false:** indicates ID.

Return Format

The query returns true or false.

Example

```
:TRIGger:FLEXray:DEFine? true     /*Sets to use Cyc Count to define
the trigger condition of FlexRay trigger when the trigger condition
is set to Frame.*/
:TRIGger:FLEXray:DEFine?                  /*The query returns true.*/
```

3.27.138 :TRIGger:FLEXray:CYCComp (Option)

Syntax

:TRIGger:FLEXray:CYCComp <*cycmax*>

:TRIGger:FLEXray:CYCComp?

Description

Sets or queries the CYC comparison condition of FlexRay trigger when the trigger condition is "FRAMe".

Parameter

Name	Type	Range	Default
<cycmax>	Discrete	{EQU|NOT|GRE|LESS|INR|OUTR}	EQU

Remarks

- **EQU:** =
- **NOT:** ≠
- **GRE:** >
- **LESS:** <
- **INR:** ><
- **OUTR:** <>

Return Format

The query returns EQU, NOT, GRE, LESS, INR, or OUTR.

Example

```
:TRIGger:FLEXray:CYCComp NOT        /*Sets the CYC comparison
condition of FlexRay trigger to NOT when the trigger condition is
"FRAMe".*/
:TRIGger:FLEXray:CYCComp?                  /*The query returns NOT.*/
```

3.27.139 :TRIGger:FLEXray:CYMAx (Option)

Syntax

:TRIGger:FLEXray:CYMAx <*cycmax*>

:TRIGger:FLEXray:CYMAx?

Description

Sets or queries the upper limit of cycle count of FlexRay trigger.

Parameter

Name	Type	Range	Default
<cycmax>	Integer	0 to 63	0

Remarks

The upper limit of cycle count of FlexRay trigger configured by this command should be greater than or equal to the cycle count lower limit configured by the *:TRIGger:FLEXray:CYMIn (Option)* command, otherwise, the lower limit will be changed with the modification of the upper limit.

Return Format

The query returns an integer ranging from 0 to 63.

Example

```
:TRIGger:FLEXray:CYMAx 10          /*Sets the upper limit of cycle
count of FlexRay trigger to 10.*/
:TRIGger:FLEXray:CYMAx?                  /*The query returns 10.*/
```

3.27.140 :TRIGger:FLEXray:CYMIn (Option)

Syntax

:TRIGger:FLEXray:CYMIn <*cycmin*>

:TRIGger:FLEXray:CYMIn?

Description

Sets or queries the lower limit of cycle count of FlexRay trigger.

Parameter

Name	Type	Range	Default
<cycmin>	Integer	0 to 63	0

Remarks

The lower limit of cycle count of FlexRay trigger configured by this command should be smaller than or equal to the cycle count upper limit configured by the *:TRIGger:FLEXray:CYMAx (Option)* command, otherwise, the upper limit will be changed with the modification of the lower limit.

Return Format

The query returns an integer ranging from 0 to 63.

Example

```
:TRIGger:FLEXray:CYMIn 10          /*Sets the lower limit of cycle
count of FlexRay trigger to 10.*/
:TRIGger:FLEXray:CYMIn?                  /*The query returns 10.*/
```

3.27.141 :TRIGger:FLEXray:IDCmp (Option)

Syntax

:TRIGger:FLEXray:IDCmp <*idcomp*>

:TRIGger:FLEXray:IDCmp?

Description

Sets or queries the ID comparison condition of FlexRay trigger when the trigger condition is "FRAMe" or "SYMBol".

Parameter

Name	Type	Range	Default
<idcomp>	Discrete	{EQUal|NOTequal|GREaterthan|LESSthan|INRange|OUTRange}	EQUal

Remarks

- **EQUal:** =
- **NOTequal:** ≠
- **GREaterthan:** >
- **LESSthan:** <
- **INRange:** ><
- **OUTRange:** <>

Return Format

The query returns EQU, NOT, GRE, LESS, INR, or OUTR.

Example

```
:TRIGger:FLEXray:IDCmp GREaterthan        /*Sets the ID comparison
condition of FlexRay trigger to GREaterthan when the trigger
condition is "FRAMe" or "SYMBol".*/
:TRIGger:FLEXray:IDCmp?                   /*The query returns GRE.*/
```

3.27.142 :TRIGger:FLEXray:IDMIn (Option)

Syntax

:TRIGger:FLEXray:IDMIn <*datamin*>

:TRIGger:FLEXray:IDMIn?

Description

Sets or queries the lower limit ID value of FlexRay trigger when the trigger condition is "FRAMe" or "SYMBol".

Parameter

Name	Type	Range	Default
<datamin>	Integer	0 to 1023	0

Remarks

N/A

Return Format

The returns an integer ranging from 0 to 1023.

Example

```
:TRIGger:FLEXray:IDMIn 10        /*Sets the lower limit ID value of
FlexRay trigger to 10 when the trigger condition is "FRAMe" or
"SYMBol".*/
:TRIGger:FLEXray:IDMIn?                  /*The query returns 10.*/
```

3.27.143 :TRIGger:FLEXray:IDMAx (Option)

Syntax

:TRIGger:FLEXray:IDMAx <*datamax*>

:TRIGger:FLEXray:IDMAx?

Description

Sets or queries the upper limit ID value of FlexRay trigger.

Parameter

Name	Type	Range	Default
<datamax>	Integer	0 to 1023	0

Remarks

N/A

Return Format

The returns an integer ranging from 0 to 1023.

Example

```
:TRIGger:FLEXray:IDMAx 100        /*Sets the upper limit ID value
of FlexRay trigger to 100.*/
:TRIGger:FLEXray:IDMAx?                  /*The query returns 10.*/
```

3.27.144 :TRIGger:IIS:SOURce:CLOCk (Option)

Syntax

:TRIGger:IIS:SOURce:CLOCk <*source*>

:TRIGger:IIS:SOURce:CLOCk?

Description

Sets or queries the clock source of the I2S trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIS:SOURce:CLOCk CHANnel2        /*Sets the clock source
to CHANnel2.*/
:TRIGger:IIS:SOURce:CLOCk?          /*The query returns CHAN2.*/
```

3.27.145 :TRIGger:IIS:CLEVel (Option)

Syntax

:TRIGger:IIS:CLEVel <*level*>

:TRIGger:IIS:CLEVel?

Description

Sets or queries the trigger level of clock line of I2S trigger.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 x VerticalScale - OFFSet) to (5 x VerticalScale - OFFSet)	0

Remarks

N/A

Return Format

The query returns the trigger level in scientific notation. The unit is V.

Example

```
:TRIGger:IIS:CLEVel 0.01  /*Sets the trigger level of clock line
source in I2S trigger to 0.01 V.*/
:TRIGger:IIS:CLEVel? /*The query returns 1.000000E-2.*/
```

3.27.146 :TRIGger:IIS:SOURce:WSELect (Option)

Syntax

:TRIGger:IIS:SOURce:WSELect <*source*>

:TRIGger:IIS:SOURce:WSELect?

Description

Sets or queries the audio channel of the I2S trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel2

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIS:SOURce:WSELect CHANnel2    /*Sets the audio channel to
CHANnel2.*/
:TRIGger:IIS:SOURce:WSELect?            /*The query returns CHAN2.*/
```

3.27.147 :TRIGger:IIS:SLEVel (Option)

Syntax

:TRIGger:IIS:SLEVel <*level*>

:TRIGger:IIS:SLEVel?

Description

Sets or queries the trigger level of frame clock line source in I2S trigger.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	-

Remarks

N/A

Return Format

The returns the trigger level in scientific notation.

Example

```
:TRIGger:IIS:SLEVel 0.01  /*Sets the trigger level of frame clock
line source in I2S trigger to 100 mV.*/
:TRIGger:IIS:SLEVel? /*The query returns 1.000000E-2.*/
```

3.27.148 :TRIGger:IIS:SOURce:DATA (Option)

Syntax

:TRIGger:IIS:SOURce:DATA <*source*>

:TRIGger:IIS:SOURce:DATA?

Description

Sets or queries the data source of the I2S trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel3

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIS:SOURce:DATA CHANnel2        /*Sets the data source to
CHANnel2.*/
:TRIGger:IIS:SOURce:DATA?          /*The query returns CHAN2.*/
```

3.27.149 :TRIGger:IIS:DLEVel (Option)

Syntax

:TRIGger:IIS:DLEVel <*level*>

:TRIGger:IIS:DLEVel?

Description

Sets or queries the trigger level of data line source of IIS trigger.

Parameter

Name	Type	Range	Default
<level>	Real	analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	-

Remarks

N/A

Return Format

The query returns the trigger level of data line source in scientific notation.

Example

```
:TRIGger:IIS DLEVel 0.16  /*Sets the trigger level of data line
source to 160 mV.*/
:TRIGger:IIS:DLEVel? /*The query returns 1.600000E-1.*/
```

3.27.150 :TRIGger:IIS:CLOCk:SLOPe (Option)

Syntax

:TRIGger:IIS:CLOCk:SLOPe <*slope*>

:TRIGger:IIS:CLOCk:SLOPe?

Description

Sets or queries the type of the clock edge of I2S trigger.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{NEGative|POSitive}	POSitive

Remarks

N/A

Return Format

The query returns NEG or POS.

Example

```
:TRIGger:IIS:CLOCk:SLOPe NEGative     /*Sets the clock edge to
NEGative.*/
:TRIGger:IIS:CLOCk:SLOPe?             /*The query returns NEG.*/
```

3.27.151 :TRIGger:IIS:AUDio (Option)

Syntax

:TRIGger:IIS:AUDio <*audio*>

:TRIGger:IIS:AUDio?

Description

Sets or queries the audio state of I2S trigger.

Parameter

Name	Type	Range	Default
<audio>	Discrete	{RIGHt|LEFT|EITHer}	LEFT

Remarks

- **RIGHt:** data of the right channel.
- **LEFT:** data of the left channel.
- **EITHer:** data of either of the channel.

Return Format

The query returns RIGH, LEFT, or EITH.

Example

```
:TRIGger:IIS:AUDio RIGHt            /*Sets the I2S audio to RIGHt.*/
:TRIGger:IIS:AUDio?                 /*The query returns RIGH.*/
```

3.27.152 :TRIGger:IIS:WHEN (Option)

Syntax

:TRIGger:IIS:WHEN <*operator*>

:TRIGger:IIS:WHEN?

Description

Sets or queries the trigger condition of I2S trigger.

Parameter

Name	Type	Range	Default
<operator>	Discrete	{EQUal|NOTequal|LESSthan| GREaterthan|INRange| OUTRange}	EQUal

Remarks

- **EQUal:** triggers when the channel's data value equals the set value.
- **NOTequal:** triggers when the channel's data value does not equal the set value.
- **LESSthan:** triggers when the channel's data value is smaller than the set value.
- **GREaterthan:** triggers when the channel's data value is greater than the set value.
- **INRange:** triggers when the channel's data value is smaller than the set upper limit value and greater than the set lower limit value.
- **OUTRange:** triggers when the channel's data value is greater than the set upper limit value or smaller than the set lower limit value.

Return Format

The query returns EQU, NOT, LESS, GRE, INR, or OUTR.

Example

```
:TRIGger:IIS:WHEN NOTequal        /*Sets the trigger condition of
I2S trigger to NOTequal.*/
:TRIGger:IIS:WHEN?                 /*The query returns NOT.*/
```

3.27.153 :TRIGger:IIS:UWIDth (Option)

Syntax

:TRIGger:IIS:UWIDth <*uwidth*>

:TRIGger:IIS:UWIDth?

Description

Sets or queries the user width of I2S trigger.

Parameter

Name	Type	Range	Default
<uwidth>	Integer	Refer to *Remarks*	-

Remarks

The user width ranges from 4 to width of I2S trigger. To set or query the width of I2S trigger, send the *:TRIGger:IIS:WIDTh (Option)* command.

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:TRIGger:IIS:UWIDth 10     /*Sets the user width of I2S trigger to
10.*/
:TRIGger:IIS:UWIDth?       /*The query returns 10.*/
```

3.27.154 :TRIGger:IIS:WIDTh (Option)

Syntax

:TRIGger:IIS:WIDTh <*uwidth*>

:TRIGger:IIS:WIDTh?

Description

Sets or queries the width of I2S trigger.

Parameter

Name	Type	Range	Default
<uwidth>	Integer	4 to 32	4

Remarks

N/A

Return Format

The query returns an integer ranging from 4 to 32.

Example

```
:TRIGger:IIS:WIDTh 10     /*Sets the width of I2S trigger to 10.*/
:TRIGger:IIS:WIDTh?       /*The query returns 10.*/
```

3.27.155 :TRIGger:IIS:ALIGnment (Option)

Syntax

:TRIGger:IIS:ALIGnment <*setting*>

:TRIGger:IIS:ALIGnment?

Description

Sets or queries the alignment mode of the I2S trigger.

Parameter

Name	Type	Range	Default
<setting>	Discrete	{LJ|RJ|IIS}	IIS

Remarks

- **LJ:** data transmission (MSB first) begins at the edge of the WS transition.
- **RJ:** data transmission (MSB first) is right-justified to the WS transition.
- **IIS:** MSB (Most Significant Bit) of data for each sample is sent first, and LSB (Least Significant Bit) is sent last. The MSB appears on the SDA line one bit clock after the edge of the WS transition.

Return Format

The query returns LJ, RJ, or IIS.

Example

```
:TRIGger:IIS:ALIGnment LJ          /*Sets the alignment mode of I2S
trigger to LJ.*/
:TRIGger:IIS:ALIGnment?            /*The query returns LJ.*/
```

3.27.156 :TRIGger:IIS:DATA (Option)

Syntax

:TRIGger:IIS:DATA <*data*>

:TRIGger:IIS:DATA?

Description

Sets or queries the data value of I2S trigger when the trigger condition is "=" or "≠".

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 2^{32}-1	0

Remarks

The settable range of <data> is affected by the byte length. The maximum byte length can be set to 4, i.g. 32-bit binary data. Therefore, the range of <data> is from 0 to 2^{32}-1.

Return Format

The query returns an integer ranging from 0 to 2^{32}-1.

Example

```
:TRIGger:IIS:DATA 10    /*Sets the data value of I2S trigger to 10
when the trigger condition is "=" or "≠".*/
:TRIGger:IIS:DATA?      /*The query returns 10.*/
```

3.27.157 :TRIGger:IIS:DMIN (Option)

Syntax

:TRIGger:IIS:DMIN <*datamin*>

:TRIGger:IIS:DMIN?

Description

Sets or queries the specified bit for the data min. of I2S trigger.

Parameter

Name	Type	Range	Default
<datamin>	Integer	Refer to *Remarks*	0

Remarks

The range of the parameter is related to the user width of I2S trigger set by the *:TRIGger:IIS:UWIDth (Option)* command.

After running this command, you can send the *:TRIGger:IIS:CODE (Option)* command to set or query the specified bit of the data min. value.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:IIS:DMIN 3  /*Sets the current bit for the data min. of
I2S trigger to 3.*/
:TRIGger:IIS:DMIN? /*The query returns 3.*/
```

3.27.158 :TRIGger:IIS:DMAX (Option)

Syntax

:TRIGger:IIS:DMAX <*datamax*>

:TRIGger:IIS:DMAX?

Description

Sets or queries the data max. value of the I2S trigger.

Parameter

Name	Type	Range	Default
<datamax>	Integer	Refer to *Remarks*	0

Remarks

The range of the parameter is related to the user width of I2S trigger set by the *:TRIGger:IIS:UWIDth (Option)* command.

After running this command, you can send the *:TRIGger:IIS:CODE (Option)* command to set or query the specified bit of the data max. value.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:IIS:DMAX 3    /*Sets the data max. value to 3.*/
:TRIGger:IIS:DMAX?      /*The query returns 3.*/
```

3.27.159 :TRIGger:IIS:CODE (Option)

Syntax

:TRIGger:IIS:CODE <*code*>

:TRIGger:IIS:CODE?

Description

Sets or queries the data value of a certain bit of I2S trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

After sending the *:TRIGger:IIS:DMAX (Option)* or *:TRIGger:IIS:DMIN (Option)* command to set the specified bit, you can send this command to query or modify the value of the specified data bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:IIS:CODE 0    /*Sets the data value to 0.*/
:TRIGger:IIS:CODE?      /*The query returns 0.*/
```

3.27.160 :TRIGger:LIN:SOURce (Option)

Syntax

:TRIGger:LIN:SOURce <*source*>

:TRIGger:LIN:SOURce?

Description

Sets or queries the trigger source of LIN trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:LIN:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:LIN:SOURce?             /*The query returns CHAN2.*/
```

3.27.161 :TRIGger:LIN:LEVel (Option)

Syntax

:TRIGger:LIN:LEVel <*level*>

:TRIGger:LIN:LEVel?

Description

Sets or queries the trigger level of LIN trigger. Its unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-5 × VerticalScale - OFFSet) to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the trigger level in scientific notation.

Example

```
:TRIGger:LIN:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:LIN:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.162 :TRIGger:LIN:STANdard (Option)

Syntax

:TRIGger:LIN:STANdard <*std*>

:TRIGger:LIN:STANdard?

Description

Sets or queries the protocol version of LIN trigger.

Parameter

Name	Type	Range	Default
<std>	Discrete	{1X|2X|BOTH}	BOTH

Remarks

N/A

Return Format

The query returns 1X, 2X, or BOTH.

Example

```
:TRIGger:LIN:STANdard 2X            /*Sets the protocol version of
LIN trigger to 2X.*/
:TRIGger:LIN:STANdard?              /*The query returns 2X.*/
```

3.27.163 :TRIGger:LIN:BAUD (Option)

Syntax

:TRIGger:LIN:BAUD <*baud*>

:TRIGger:LIN:BAUD?

Description

Sets or queries the baud rate of LIN trigger. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Integer	1 kbps to 20 Mbps	9600 bps

Remarks

If the baud rate is set to a value with "M", then "A" should be added at the end of the value. For example, if you send 5M, you need to send 5MA.

Return Format

The query returns an integer ranging from 1 kbps to 20 Mbps.

Example

```
:TRIGger:LIN:BAUD 19200         /*Sets the baud rate of LIN trigger
to 19.2 kbps.*/
:TRIGger:LIN:BAUD?               /*The query returns 19200.*/
```

3.27.164 :TRIGger:LIN:SAMPlepoint (Option)

Syntax

:TRIGger:LIN:SAMPlepoint <*value*>

:TRIGger:LIN:SAMPlepoint?

Description

Sets or queries the sample position of LIN trigger.

Parameter

Name	Type	Range	Default
<value>	Integer	10 to 90	50

Remarks

The sample position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

Return Format

The query returns an integer ranging from 10 to 90.

Example

```
:TRIGger:LIN:SAMPlepoint 40          /*Sets the sample point position
of LIN trigger to 40%.*/
:TRIGger:LIN:SAMPlepoint?            /*The query returns 40.*/
```

3.27.165 :TRIGger:LIN:WHEN (Option)

Syntax

`:TRIGger:LIN:WHEN` <*when*>

`:TRIGger:LIN:WHEN?`

Description

Sets or queries the trigger condition of LIN trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{SYNCbreak|ID|DATA|IDData|SLEep|WAKeup|ERRor}	SYNCbreak

Remarks

- **SYNCbreak:** triggers on the last bit of the sync field.
- **ID:** triggers when the frames with the specified ID are found.
- **DATA:** triggers when the data that meet the preset conditions are found.

- **IDData:** triggers when the frames with the specified ID and data that meet the preset conditions are both found.
- **SLEep:** triggers when the sleep frame is found.
- **WAKeup:** triggers when the wakeup frame is found.
- **ERRor:** triggers on the specified type of error frame.

Return Format

The query returns SYNC, ID, DATA, IDD, SLE, WAK, or ERR.

Example

```
:TRIGger:LIN:WHEN SYNCbreak          /*Sets the trigger condition to
SYNCbreak.*/
:TRIGger:LIN:WHEN?                   /*The query returns SYNC.*/
```

3.27.166 :TRIGger:LIN:ERRor (Option)

Syntax

:TRIGger:LIN:ERRor <*value*>

:TRIGger:LIN:ERRor?

Description

Sets or queries the error type of LIN trigger when the trigger condition is "Data".

Parameter

Name	Type	Range	Default
<value>	Discrete	{SYNC|ID|CHECK}	SYNC

Remarks

- **SYNC:** indicates Sync error.
- **ID:** indicates Even Odd error.
- **CHECK:** indicates Check Sum error.

Return Format

The query returns SYNC, ID, or CHECK.

Example

```
:TRIGger:LIN:ERRor CHECK          /*Sets the LIN trigger error type
to CHECK.*/
:TRIGger:LIN:ERRor?                /*The query returns CHECK.*/
```

3.27.167 :TRIGger:LIN:ID (Option)

Syntax

:TRIGger:LIN:ID <*id*>

:TRIGger:LIN:ID?

Description

Sets or queries the ID value of LIN trigger.

Parameter

Name	Type	Range	Default
<id>	Real	0 to 63	0

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to 63.

Example

```
:TRIGger:LIN:ID 4                /*Sets the ID value of LIN trigger to
4.*/
:TRIGger:LIN:ID?                /*The query returns 4.*/
```

3.27.168 :TRIGger:LIN:DATA (Option)

Syntax

:TRIGger:LIN:DATA <*data*>

:TRIGger:LIN:DATA?

Description

Sets or queries the data value of LIN trigger when the trigger condition is "Data".

Parameter

Name	Type	Range	Default
<data>	Integer	Refer to *Remarks*	0

Remarks

The range of the data value of LIN trigger is related to the value of data bytes. The maximum number of bytes can be set to 8, i.g. 64-bit binary data. Therefore, the range of <data> is from 0 to 2^{64}-1.

Return Format

The query returns an integer ranging from 0 to 2^{64}-1.

Example

```
:TRIGger:LIN:DATA 100    /*Sets the data value of LIN trigger to
100 when the trigger condition is "Data".*/
:TRIGger:LIN:DATA?      /*The query returns 100.*/
```

3.27.169 :TRIGger:LIN:CURRbit (Option)

Syntax

:TRIGger:LIN:CURRbit <*currbit*>

:TRIGger:LIN:CURRbit?

Description

Sets or queries the current bit of the LIN trigger data.

Parameter

Name	Type	Range	Default
<currbit>	Integer	0 to 39	0

Remarks

After configuring the settings for this command, you can send the *:TRIGger:LIN:CODE (Option)* command to set or modify the set bit data.

Return Format

The query returns an integer ranging from 0 to 39.

Example

```
:TRIGger:LIN:CURRbit 8    /*Sets the current bit of LIN trigger to
8.*/
:TRIGger:LIN:CURRbit?      /*The query returns 8.*/
```

3.27.170 :TRIGger:LIN:CODE (Option)

Syntax

:TRIGger:LIN:CODE <*code*>

:TRIGger:LIN:CODE?

Description

Sets or queries the data value of a certain bit of LIN trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

After sending the *:TRIGger:LIN:CURRbit (Option)* command to set the specified bit, you can send this command to query or modify the value of the specified data bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:LIN:CODE 0     /*Sets the data value to 0.*/
:TRIGger:LIN:CODE?       /*The query returns 0.*/
```

3.27.171 :TRIGger:M1553:SOURce (Option)

Syntax

:TRIGger:M1553:SOURce <*source*>

:TRIGger:M1553:SOURce?

Description

Sets or queries the trigger source of M1553 trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2| CHANnel3|CHANnel4}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:M1553:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:M1553:SOURce?             /*The query returns CHAN2.*/
```

3.27.172 :TRIGger:M1553:WINDow (Option)

Syntax

:TRIGger:M1553:WINDow <*window*>

:TRIGger:M1553:WINDow?

Description

Sets or queries the type of the trigger level of M1553 trigger.

Parameter

Name	Type	Range	Default
<window>	Discrete	{TA|TB|TAB}	TA

Remarks

- **TA:** Level A. Only adjusts the upper limit of the trigger level; the lower limit of the trigger level remains unchanged.
- **TB:** Level B. Only adjusts the lower limit of the trigger level; the upper limit of the trigger level remains unchanged.
- **TAB:** adjusts the upper and lower trigger level synchronously. The trigger level deviation (the difference between the upper limit and lower limit) remains unchanged.

Return Format

The query returns TA, TB, or TAB.

Example

```
:TRIGger:M1553:WINDow TAB /*Sets the type of the trigger level of
M1553 trigger to TAB.*/
:TRIGger:M1553:WINDow?   /*The query returns TAB.*/
```

3.27.173 :TRIGger:M1553:ALEVel (Option)

Syntax

:TRIGger:M1553:ALEVel <*level*>

:TRIGger:M1553:ALEVel?

Description

Sets or queries the trigger level of M1553 trigger. Its unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Lower limit to (5 × VerticalScale - OFFSet)	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the upper limit of the trigger level in scientific notation.

Example

```
:TRIGger:M1553:ALEVel 0.16     /*Sets the upper limit of the trigger
level to 160 mV.*/
:TRIGger:M1553:ALEVel?         /*The query returns 1.600000E-1.*/
```

3.27.174 :TRIGger:M1553:BLEVel (Option)

Syntax

:TRIGger:M1553:BLEVel <*level*>

:TRIGger:M1553:BLEVel?

Description

Sets or queries the lower limit of the trigger level of Delay trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	(-5 × VerticalScale - OFFSet) to upper limit	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

The query returns the lower limit of the trigger level in scientific notation.

Example

```
:TRIGger:M1553:BLEVel 0.05   /*Sets the lower limit of the trigger
level to 50 mV.*/
:TRIGger:M1553:BLEVel?      /*The query returns 5.000000E-2.*/
```

3.27.175 :TRIGger:M1553:POLarity (Option)

Syntax

:TRIGger:M1553:POLarity <*polarity*>

:TRIGger:M1553:POLarity?

Description

Sets or queries the polarity of M1553 trigger.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Return Format

The query returns POS or NEG.

Example

```
:TRIGger:M1553:POLarity POSitive    /*Sets the polarity of M1553
trigger to POSitive.*/
:TRIGger:M1553:POLarity?           /*The query returns POS.*/
```

3.27.176 :TRIGger:M1553:WHEN (Option)

Syntax

:TRIGger:M1553:WHEN <*when*>

:TRIGger:M1553:WHEN?

Description

Sets or queries the trigger condition of M1553 trigger.

Parameter

Name	Type	Range	Default
<when>	Discrete	{SYNCbreak|DATA|CMD|STATus|ERRor}	SYNCbreak

Remarks

- **SYNCbreak:** triggers on the specified sync type.
- **DATA:** triggers on the specified data word.
- **CMD:** triggers on the specified remote terminal address.
- **STATus:** triggers on the RTA and the remaining 11 bits.
- **ERRor:** triggers on the specified error type.

Return Format

The query returns SYNC, DATA, CMD, STAT, or ERR.

Example

```
:TRIGger:M1553:WHEN CMD        /*Sets the trigger condition to CMD.*/
:TRIGger:M1553:WHEN?            /*The query returns CMD.*/
```

3.27.177 :TRIGger:M1553:SYNC (Option)

Syntax

:TRIGger:M1553:SYNC <*sync*>

:TRIGger:M1553:SYNC?

Description

Sets or queries the sync type of M1553 trigger when the trigger condition is "SYNCbreak".

Parameter

Name	Type	Range	Default
<sync>	Discrete	{DATA|STATUS|ALL}	-

Remarks

N/A

- **DATA:** indicates data frame sync.

- **STATUS:** indicates command/status sync.
- **ALL:** indicates the all frame sync.

Return Format

The query returns DATA, STATUS, or ALL.

Example

```
:TRIGger:M1553:SYNC STATUS    /*Sets the sync type of M1553 trigger
to STATUS when the trigger condition is "SYNCbreak".*/
:TRIGger:M1553:SYNC?           /*The query returns STATUS.*/
```

3.27.178 :TRIGger:M1553:ERRor (Option)

Syntax

:TRIGger:M1553:ERRor <*err*>

:TRIGger:M1553:ERRor?

Description

Sets or queries the M1553 trigger error type.

Parameter

Name	Type	Range	Default
<err>	Discrete	{SYNC|ERR}	SYNC

Remarks

N/A

Return Format

The query returns SYNC or ERR.

Example

```
:TRIGger:M1553:ERRor ERR        /*Sets the M1553 trigger error type
to ERR.*/
:TRIGger:M1553:ERRor?              /*The query returns ERR.*/
```

3.27.179 :TRIGger:M1553:DATComp (Option)

Syntax

:TRIGger:M1553:DATComp <*datacomp*>

:TRIGger:M1553:DATComp?

Description

Sets or queries the comparison type of M1553 trigger when the trigger condition is "DATA".

Parameter

Name	Type	Range	Default
<datacomp>	Discrete	{EQUal|NOTequal|GREaterthan|LESSthan|INRange|OUTRange}	EQUal

Remarks

- **EQUal:** =
- **NOTequal:** ≠
- **GREaterthan:** >
- **LESSthan:** <
- **INRange:** ><
- **OUTRange:** <>

Return Format

The query returns EQU, NOT, GRE, LESS, INR, or OUTR.

Example

```
:TRIGger:M1553:DATComp NOTequal          /*Sets the comparison type
of M1553 trigger to NOTequal when the trigger condition is "DATA".*/
:TRIGger:M1553:DATComp?                   /*The query returns NOT.*/
```

3.27.180 :TRIGger:M1553:DATValue (Option)

Syntax

:TRIGger:M1553:DATValue <*data*>

:TRIGger:M1553:DATValue?

Description

Sets or queries the data value of 1553 trigger.

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 65535	0

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to 65535.

Example

```
:TRIGger:M1553:DATValue 100    /*Sets the data value of M1553
trigger to 100.*/
:TRIGger:M1553:DATValue?      /*The query returns 100.*/
```

3.27.181 :TRIGger:M1553:DMAX (Option)

Syntax

:TRIGger:M1553:DMAX <*datamax*>

:TRIGger:M1553:DMAX?

Description

Sets or queries the bit of data max. of M1553 trigger.

Parameter

Name	Type	Range	Default
<datamax>	Integer	0 to 19	0

Remarks

After configuring the specified bit, you can run the *:TRIGger:M1553:CODE (Option)* command to set or query the value of the specified bit.

Return Format

The query returns an integer ranging from 0 to 19.

Example

```
:TRIGger:M1553:DMAX 10    /*Sets the bit of data max. of M1553
trigger to 10.*/
:TRIGger:M1553:DMAX?      /*The query returns 10.*/
```

3.27.182 :TRIGger:M1553:DMIN (Option)

Syntax

:TRIGger:M1553:DMIN <*datamin*>

:TRIGger:M1553:DMIN?

Description

Sets or queries the bit of data min. of M1553 trigger.

Parameter

Name	Type	Range	Default
<datamin>	Integer	0 to 19	0

Remarks

The bit starts from the zero bit of the Binary format and ends at the last bit of the Hex format. The spaces between strings of Binary and Hex format are ignored.

After configuring the specified bit, you can run the *:TRIGger:M1553:CODE (Option)* command to set or query the value of the specified bit.

Return Format

The query returns an integer ranging from 0 to 19.

Example

```
:TRIGger:M1553:DMIN 10     /*Sets the bit of data min. of M1553
trigger to 10.*/
:TRIGger:M1553:DMIN?       /*The query returns 10.*/
```

3.27.183 :TRIGger:M1553:DRTA (Option)

Syntax

:TRIGger:M1553:DRTA <*data*>

:TRIGger:M1553:DRTA?

Description

Sets or queries the data value of M1553 trigger.

Parameter

Name	Type	Range	Default
<data>	Integer	0 to 6	0

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to 6.

Example

```
:TRIGger:M1553:DRTA 4    /*Sets the data value of M1553 trigger to
4.*/
:TRIGger:M1553:DRTA?      /*The query returns 4.*/
```

3.27.184 :TRIGger:M1553:DBIT (Option)

Syntax

:TRIGger:M1553:DBIT <*databit*>

:TRIGger:M1553:DBIT?

Description

Sets or queries the bit of bit time of M1553 trigger when the trigger condition is set to "RAT+11Bit".

Parameter

Name	Type	Range	Default
<databit>	Integer	0 to 13	0

Remarks

After configuring the specified bit, you can run the *:TRIGger:M1553:CODE (Option)* command to set or query the value of the specified bit.

Return Format

The query returns an integer ranging from 0 to 13.

Example

```
:TRIGger:M1553:DBIT 10    /*Sets the bit of the bit time of M1553
trigger to 10.*/
:TRIGger:M1553:DBIT?      /*The query returns 10.*/
```

3.27.185 :TRIGger:M1553:CODE (Option)

Syntax

:TRIGger:M1553:CODE <*code*>

:TRIGger:M1553:CODE?

Description

Sets or queries the data value of a certain bit of M1553 trigger.

Parameter

Name	Type	Range	Default
<code>	Discrete	{0|1|255}	255

Remarks

When <*code*> is set to 255, it indicates the data value can be any value.

- When you modify the data max. value of M1553 trigger, first send the *:TRIGger:M1553:DMAX (Option)* command to set the bit to be modified, and then send this command to modify the value of the bit.
- When you modify the data min. value of M1553 trigger, first send the *:TRIGger:M1553:DMIN (Option)* command to set the bit to be modified, and then send this command to modify the value of the bit.
- When you modify the bit time of M1553 trigger, first send the *:TRIGger:M1553:DBIT (Option)* command to set the bit to be modified, and then send this command to modify the value of the bit.

Return Format

The query returns 0, 1, or 255.

Example

```
:TRIGger:M1553:CODE 1     /*Sets the data value of a certain bit of
M1553 trigger to 1.*/
:TRIGger:M1553:CODE?       /*The query returns 1.*/
```

3.28 :WAVeform Commands

The :WAVeform commands are used to read waveform data and relevant settings. The *:WAVeform:MODE*command is used to set the reading mode of waveform data. In different modes, the definitions for the parameters are different, as shown in *Figure 3.1* and *Figure 3.2* .

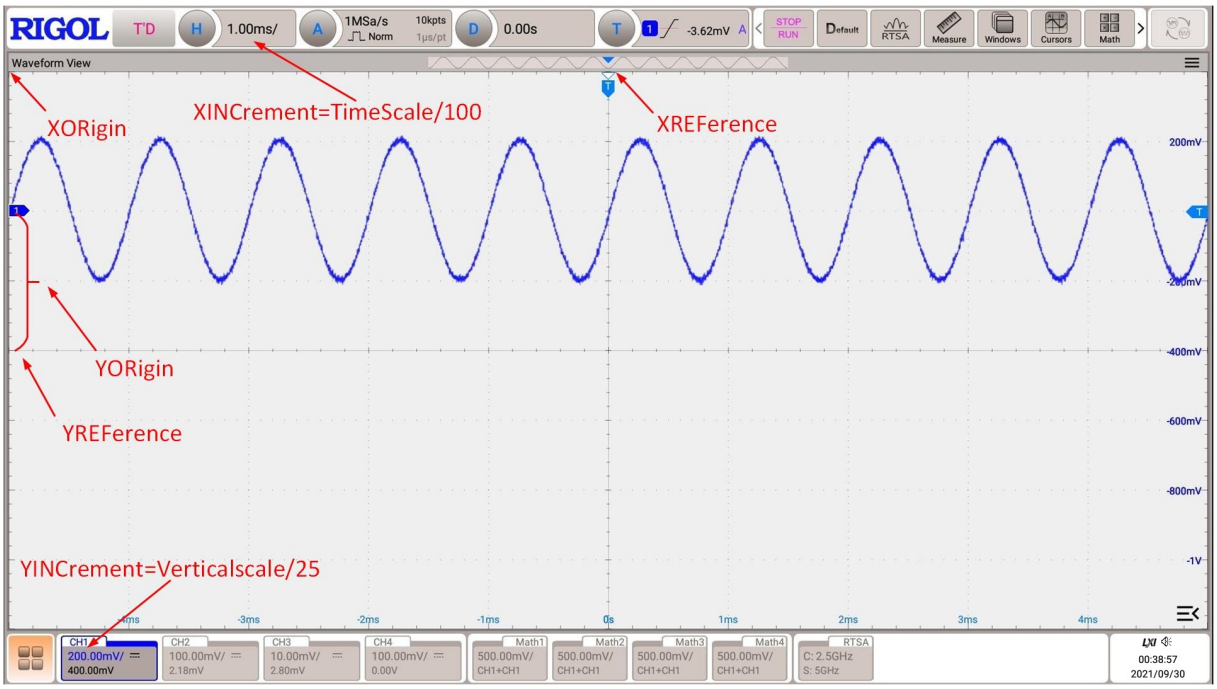


Figure 3.1 Parameter Definitions in NORMAL Mode

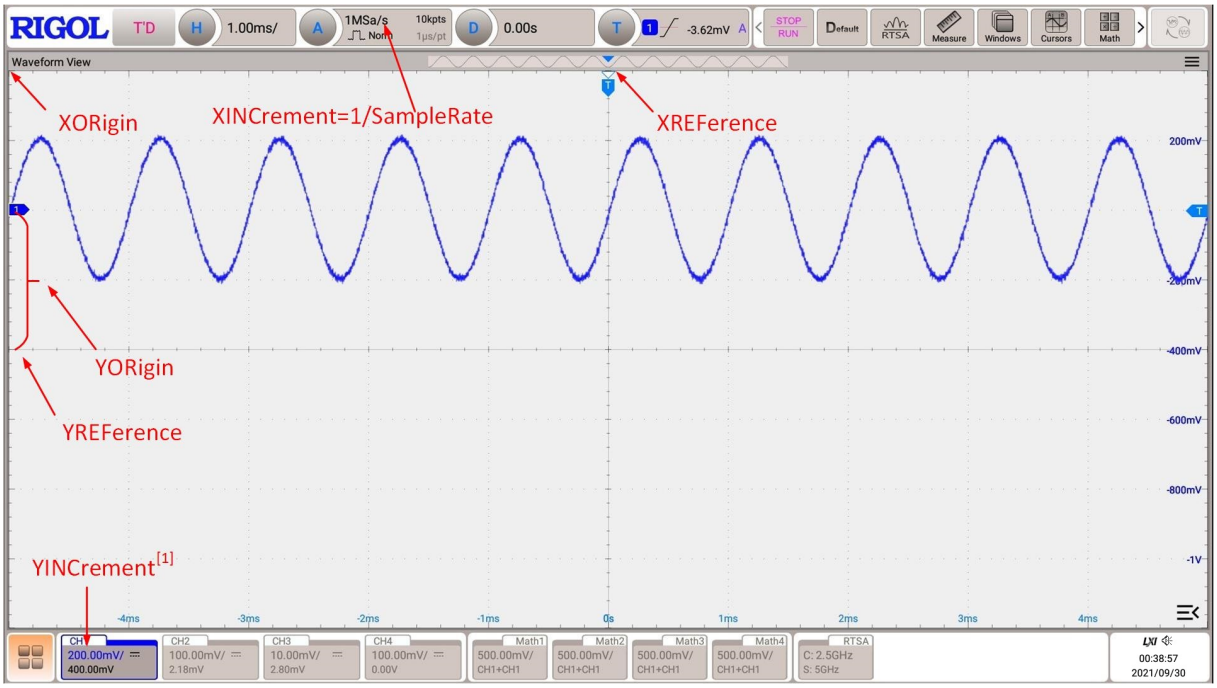


Figure 3.2 Parameter Definitions in RAW Mode

Note:[1]: In RAW mode, YINCrement and Verticalscale of the memory waveforms are related to the currently selected Verticalscale.

3.28.1 :WAVeform:SOURce

Syntax

:WAVeform:SOURce <*source*>

:WAVeform:SOURce?

Description

Sets or queries the source channel of waveform data reading.

Parameter

Name	Type	Range	Default
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

When the channel source is set to MATH1-MATH4, *:WAVeform:MODE* can only select the NORMal mode.

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:WAVeform:SOURce CHANnel2    /*Sets the channel source to
CHANnel2.*/
:WAVeform:SOURce?             /*The query returns CHAN2.*/
```

3.28.2 :WAVeform:MODE

Syntax

:WAVeform:MODE <*mode*>

:WAVeform:MODE?

Description

Sets or queries the mode of the *:WAVeform:DATA?* command in reading data.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{NORMal|MAXimum|RAW}	NORMal

Remarks

- **NORMal:** reads the waveform data currently displayed on the screen.
- **MAXimum:** reads the waveform data displayed on the screen when the oscilloscope is in the Run state; reads the waveform data from the internal memory when the oscilloscope is in the Stop state.
- **RAW:** reads the waveform data from the internal memory. Note: The data in the internal memory can only be read when the oscilloscope is in the Stop state. You are not allowed to operate the instrument when it is reading data.
- When the channel source is set to MATH, only the NORMal mode is valid.

Return Format

The query returns NORM, MAX, or RAW.

Example

```
:WAVeform:MODE RAW  /*Sets the reading mode of waveform data to
RAW.*/
:WAVeform:MODE?      /*The query returns RAW.*/
```

3.28.3 :WAVeform:FORMat

Syntax

:WAVeform:FORMat <*format*>

:WAVeform:FORMat?

Description

Sets or queries the return format of the waveform data.

Parameter

Name	Type	Range	Default
<format>	Discrete	{WORD|BYTE|ASCii}	BYTE

Remarks

- **WORD:** Each waveform point occupies two bytes (16 bits).
- **BYTE:** Each waveform point occupies one byte (8 bits).
- **ASCii:** The query returns the actual voltage value of each waveform point in scientific notation; and the voltage values are separated by commas.

Return Format

The query returns WORD, BYTE, or ASC.

Example

```
:WAVeform:FORMat WORD   /*Sets the returned format of waveform data
to WORD.*/
:WAVeform:FORMat?        /*The query returns WORD.*/
```

3.28.4 :WAVeform:POINts

Syntax

:WAVeform:POINts <*point*>

:WAVeform:POINts?

Description

Sets or queries the number of the waveform points to be read in the current mode.

Parameter

Name	Type	Range	Default
<point>	Integer	Refer to *Remarks*	-

Remarks

The range of <point> is related to the current reading mode of the waveform data.

- **NORMal:** 1 to 1000
- **RAW:** 1 to the current maximum memory depth
- **MAXimum:** 1 to the number of effective points on the current screen

Return Format

The query returns the number of waveform points in integer.

Example

```
N/A
```

3.28.5 :WAVeform:DATA?

Syntax

:WAVeform:DATA?

Description

Reads the waveform data.

Parameter

N/A

Remarks

Procedures of reading the waveform data on the screen:

```
:WAV:SOUR CHAN1    /*Sets the channel source to CH1.*/
:WAV:MODE NORMal   /*Sets the waveform reading mode to NORMal.*/
:WAV:FORM BYTE   /*Sets the return format of the waveform data to
BYTE.*/
:WAV:DATA?    /*Reads the waveform data on the screen.*/
```

Procedures of reading the waveform data in the internal memory:

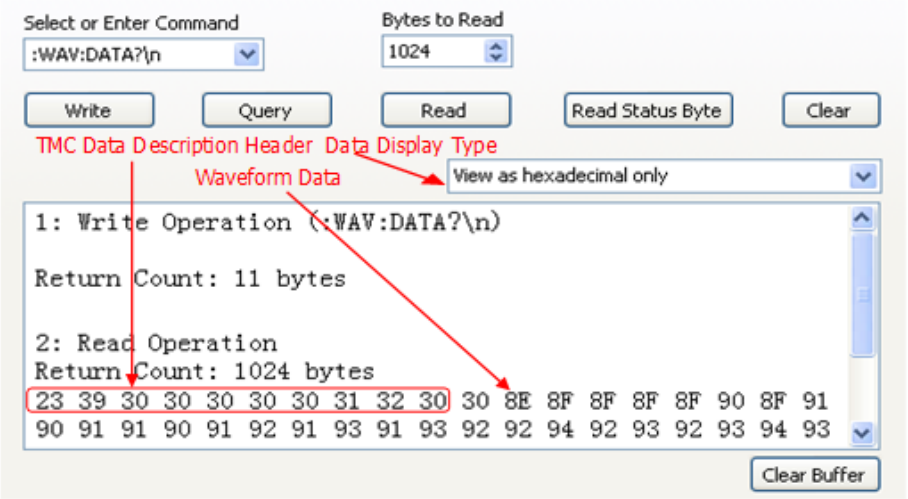
```
:STOP    /*Sets the instrument to STOP state (you can only read the
waveform data from the internal memory when the oscilloscope is in
STOP state).*/
:WAV:SOUR CHAN1   /*Sets the channel source to CH1.*/
:WAV:MODE RAW   /*Sets the waveform reading mode to RAW.*/
:WAV:FORM BYTE   /*Sets the return format of the waveform data to
BYTE.*/
:WAV:STAR 1     /*Sets the start point of waveform data reading to
the first waveform point.*/
:WAVeform:STOP 120000   /*Sets the stop point of waveform data
reading to the 120000th waveform point (last point).*/
```

Return Format

- The return format is related to the return format of the currently selected waveform data (*:WAVeform:FORMat*).

  WORD or BYTE format: The read data format is TMC header + waveform data points + end identifier. The TMC header is in #NXXXXXX format; wherein, # is the TMC header identifier; N following # represents the length of the waveform data; the length of the waveform data points is expressed in ASCII strings,and the terminator represents the ending of communication. For example, the data read for one time is #9000001000XXXX. It indicates that 9 bytes are used to describe the data length. 000001000 indicates the length of waveform data, i.g. 1,000 bytes.

  ASCii: The query returns the actual voltage value of each waveform point in scientific notation; and the voltage values are separated by commas.
- When the waveform data in the internal memory are read in batches, the waveform data returned each time might be the data in one area of the internal memory. In "WORD" or "BYTE" return format, each returned data in blocks contain the TMC data block header. Waveform data in two adjacent data blocks are consecutive.
- The figure below shows the waveform data that have been read (in BYTE format). First, select "View as hexadecimal only" from the drop-down list at the right side. Then, the waveform data that have been read are displayed in hexadecimal format. The first 11 bytes denote the "TMC data block header", and beginning from the 12th byte (8E) are the waveform data. You can convert the waveform data read to the voltage value of each point of the waveform by using the formula "(0x8E - YORigin - YREFerence) × YINCrement". For the definitions of the parameters in this formula, refer to *Related Command*.



- When the waveform data in the internal memory are read in batches, the waveform data returned each time might be the data in one area of the internal memory. In "WORD" or "BYTE" return format, each returned data in blocks contain the TMC data block header. Waveform data in two adjacent data blocks are consecutive.

Related Command

:WAVeform:MODE

:WAVeform:YINCrement?

:WAVeform:YINCrement?

:WAVeform:YORigin?

3.28.6 :WAVeform:XINCrement?

Syntax

`:WAVeform:XINCrement?`

Description

Queries the time interval between two neighboring points of the currently selected channel source in the X direction.

Parameter

N/A

Remarks

The returned value is related to the current data reading mode:

- In NORMal mode, XINCrement = TimeScale/100.
- In RAW mode, XINCrement = 1/SampleRate.

- In MAX mode, XINCrement = TimeScale/100 when the oscilloscope is in the Run state; XINCrement = 1/SampleRate when the oscilloscope is in the Stop state.

The unit is related to the current channel source:

Return Format

The query returns the time difference in scientific notation.

Example

```
N/A
```

3.28.7 :WAVeform:XORigin?

Syntax

```
:WAVeform:XORigin?
```

Description

Queries the start time of the waveform data of the currently selected channel source in the X direction.

Parameter

N/A

Remarks

The returned value is related to the current data reading mode:

- In NORMal mode, the query returns the start time of the waveform data displayed on the screen.
- In RAW mode, the query returns the start time of the waveform data in the internal memory.
- In MAX mode, the query returns the start time of the waveform data displayed on the screen when the instrument is in the RUN state; the query returns the start time of the waveform data in the internal memory when the instrument is in the Stop state.

The unit is related to the current channel source.

Return Format

The query returns the time value in scientific notation.

Example

```
N/A
```

3.28.8 :WAVeform:XREFerence?

Syntax

```
:WAVeform:XREFerence?
```

Description

Queries the reference time of the waveform points of the currently selected channel source in the X direction.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 0 (namely the first waveform point on the screen or in the internal memory).

Example

```
N/A
```

3.28.9 :WAVeform:YINCrement?

Syntax

```
:WAVeform:YINCrement?
```

Description

Queries the unit voltage value of the current source channel Y in the Y direction.

Parameter

N/A

Remarks

The returned value is related to the current data reading mode:

- In NORMal mode, YINCrement = VerticalScale/25.
- In RAW mode, YINCrement and Verticalscale of the memory waveforms are related to the currently selected Verticalscale.
- In MAX mode, YINCrement = VerticalScale/25 when the instrument is in the RUN state; YINCrement is related to the VerticalScale of the internal waveform and the currently selected VerticalScale when the instrument is in the Stop state.

Return Format

The query returns the unit voltage value in scientific notation.

Example

```
N/A
```

3.28.10 :WAVeform:YORigin?

Syntax

`:WAVeform:YORigin?`

Description

Queries the vertical offset relative to the vertical reference position of the currently selected channel source in the Y direction.

Parameter

N/A

Remarks

The returned value is related to the current data reading mode:

- In NORMal mode, YORigin = VerticalOffset/YINCrement.
- In RAW mode, YORigin is related to the VerticalScale of the memory waveforms and the currently selected VerticalScale.
- In MAX mode, YORigin = VerticalOffset/YINCrement when the instrument is in the RUN state; YORigin is related to the VerticalScale of the internal waveform and the currently selected VerticalScale when the instrument is in the Stop state.

Return Format

The query returns an integer.

Example

```
N/A
```

3.28.11 :WAVeform:YREFerence?

Syntax

`:WAVeform:YREFerence?`

Description

Queries the vertical reference position of the currently selected channel source in the Y direction.

Parameter

N/A

Remarks

The value of YREFerence is related to the configuration of the *:WAVeform:FORMat* command. The reference position is different for different return formats of waveform data.

Return Format

The query returns an integer.

Example

```
N/A
```

3.28.12 :WAVeform:STARt

Syntax

:WAVeform:STARt <*sta*>

:WAVeform:STARt?

Description

Sets or queries the start position of waveform data reading.

Parameter

Name	Type	Range	Default
<sta>	Integer	NORMal: 1 to 1000 MAX: 1 to the number of effective points on the current screen RAW: 1 to the current maximum memory depth	1

Remarks

When reading the waveform data from the internal memory, the actual settable ranges of the start point and stop point of a reading operation are related to the memory depth of the oscilloscope and the return format of the waveform data currently selected.

Return Format

The query returns an integer.

Example

```
:WAVeform:STARt 100   /*Sets the start point to 100.*/
:WAVeform:STARt?   /*The query returns 100.*/
```

3.28.13 :WAVeform:STOP

Syntax

:WAVeform:STOP <*stop*>

:WAVeform:STOP?

Description

Sets or queries the stop position of waveform data reading.

Parameter

Name	Type	Range	Default
<stop>	Integer	NORMal: 1 to 1000 MAX: 1 to the number of effective points on the current screen RAW: 1 to the current maximum memory depth	1,000

Remarks

When reading the waveform data in the internal memory, the actual settable ranges of the start point and stop point of a reading operation are related to the memory depth of the oscilloscope and the return format of the waveform data currently selected.

Return Format

The query returns an integer.

Example

```
:WAVeform:STOP 500   /*Sets the stop point to 500.*/
:WAVeform:STOP?   /*The query returns 500.*/
```

3.28.14 :WAVeform:PREamble?

Syntax

:WAVeform:PREamble?

Description

Queries all the waveform parameters.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 10 waveform parameters, separated by commas.

<format>,<type>,<points>,<count>,<xincrement>,<xorigin>,<xreference>,<yincrement>,<yorigin>,<yreference>

Wherein,

<format>: indicates 0 (BYTE), 1 (WORD), or 2 (ASC).

<type>: indicates 0 (NORMal), 1 (MAXimum), or 2 (RAW).

<points>: an integer ranging from 1 to 50,000,000.

<count>: indicates the number of averages in the average sample mode. The value of <count> parameter is 1 in other modes.

<xincrement>: indicates the time difference between two neighboring points in the X direction.

<xorigin>: indicates the start time of the waveform data in the X direction.

<xreference>: indicates the reference time of the waveform data in the X direction.

<yincrement>: indicates the step value of the waveforms in the Y direction.

<yorigin>: indicates the vertical offset relative to the "Vertical Reference Position" in the Y direction.

<yreference>: indicates the vertical reference position in the Y direction.

Example

```
:WAVeform:PREamble?/*The query returns
0,0,1000,1,1.000000E-8,-5.000000E-6,0.000000E-12,4.000000E-03,0,128.
*/
```

4 Programming Examples

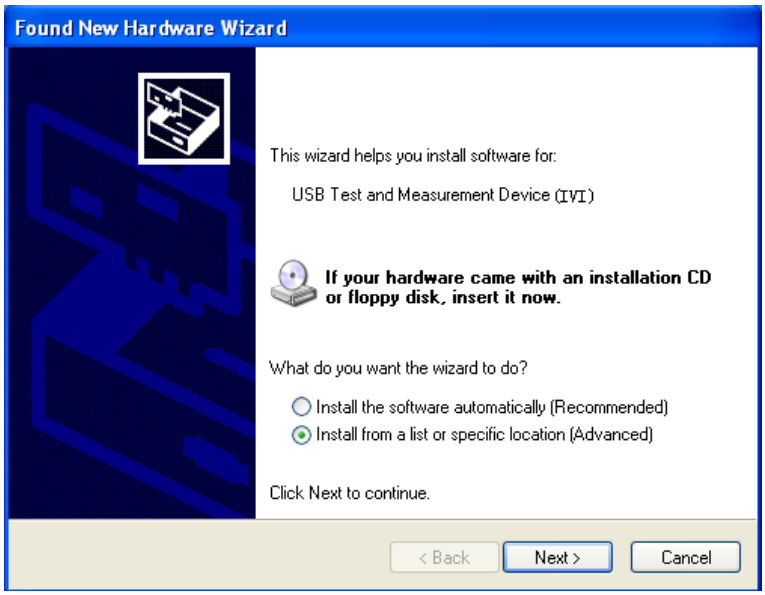
This chapter illustrates how to control the digital oscilloscope by programming in Excel, LabVIEW, Visual Basic, and Visual C++. These examples are programmed based on Virtual Instrument Software Architecture (VISA) library.

4.1 Programming Preparations

Before programming, you need to prepare the following tasks:

You can log in to the RIGOL official website (www.rigol.com) to download the software. Then install the software according to the installation wizard. After Ultra Sigma is installed successfully, NI-VISA library will be completely installed automatically. In this manual, the default installation path is C:\Program Files\IVI Foundation\VISA.

In the manual, the oscilloscope communicates with the PC via the USB interface. Connect the USB Device interface on the rear panel of the oscilloscope to the PC by using the USB cable. After the oscilloscope is properly connected to the PC, power on the instrument to start it. In this case, "Found New Hardware Wizard" dialog box appears on the PC. Please install "USB Test and Measurement Device (IVI)" according to the instructions.



By now, the programming preparations are complete. The following parts will make a detailed introduction about the programming examples in Excel, LabVIEW, Visual Basic, and Visual C++.

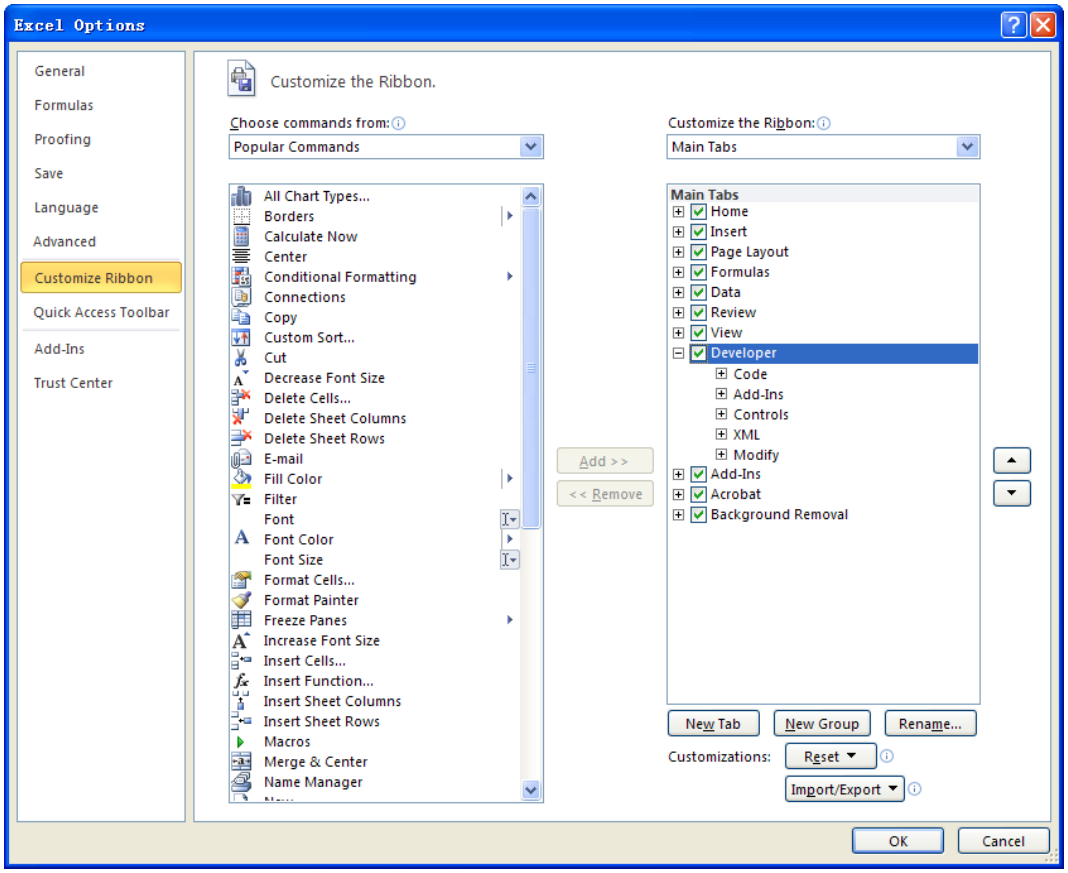
4.2 Excel Programming Examples

Program used in this instance: Microsoft Excel 2010

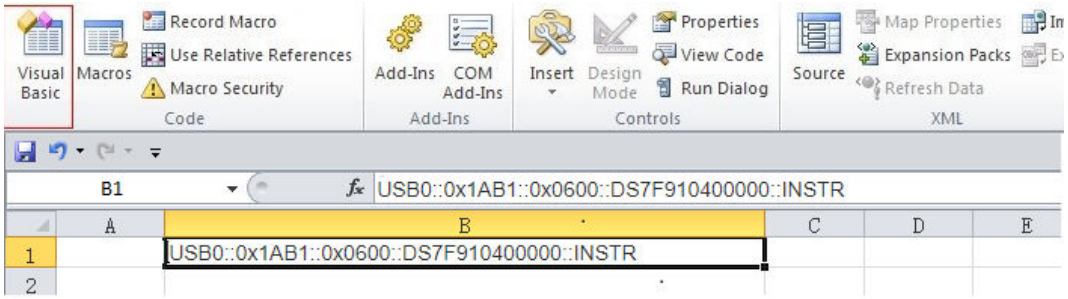
Function realized in this example: sending the *IDN? command and reading the instrument information.

1 Open a new Macro-enabled Excel file and name it "Demo_Excel.xlsm" in this example.

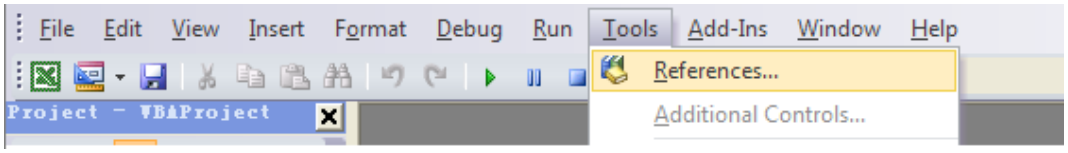
2 Run the Demo_Excel.xlsm file. Click File > Options at the upper-left corner of the Excel file to open the interface as shown in the figure below. Click Customize Ribbon at the left, check Developer and click OK. At this point, the Excel menu bar displays the Developer menu.



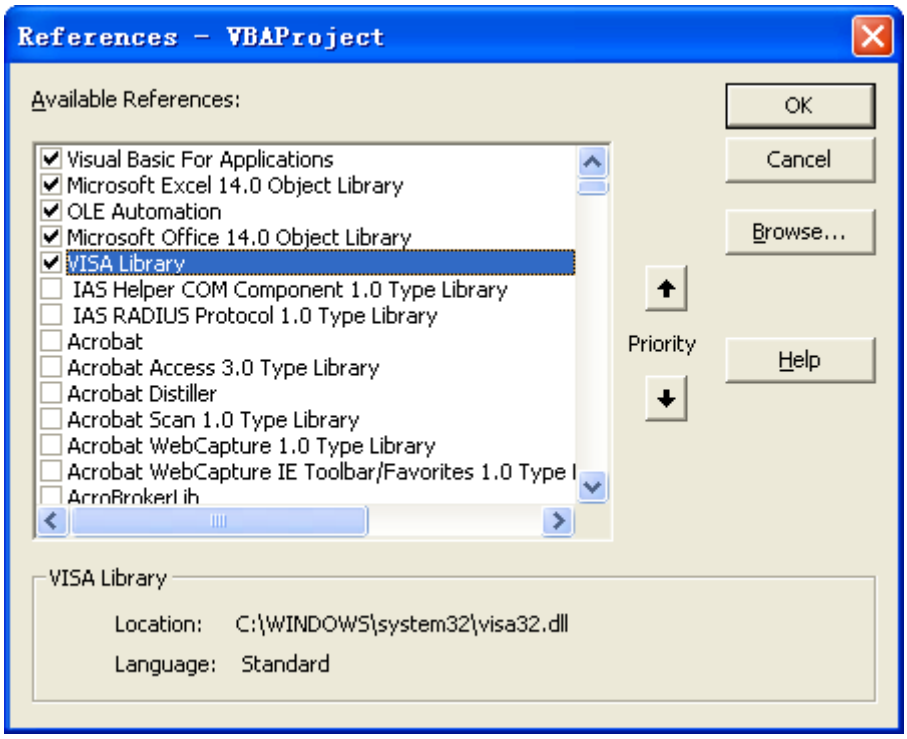
3 Enter a device resource descriptor into a cell of the file as shown in the figure below. For example, the device resource descriptor is USB0::0x1AB1::0x0600::DS7F910400000::INSTR. Input it into SHEET1.CELLS(1,2) (i.g. the B1 cell in Sheet1). Click the Developer menu and select the Visual Basic option to open the Microsoft Visual Basic.



4 Select Tools(T) in the Microsoft Visual Basic menu bar and click References.



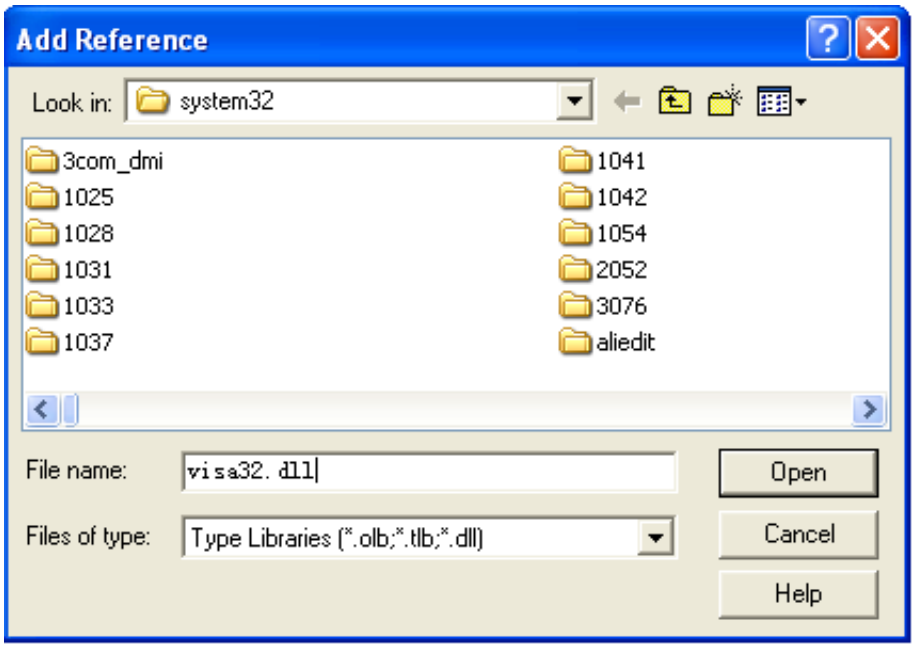
In the displayed dialog box, select VISA Library, and click OK to refer to VISA Library.



If you cannot find VISA Library in the left section of the above dialog box, please follow the method below to find it.

a Make sure that your PC has installed the NI-VISA library.

b Click Browse... at the right section to search visa32.dll from C:\WINDOWS \system32, as shown in the figure below.



5 Click View Code under Developer menu to enter the interface of Microsoft Visual Basic. Add the following codes and save it.

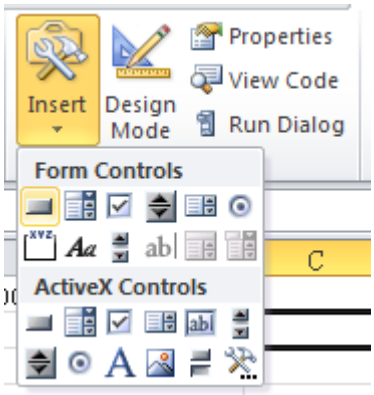
Caution

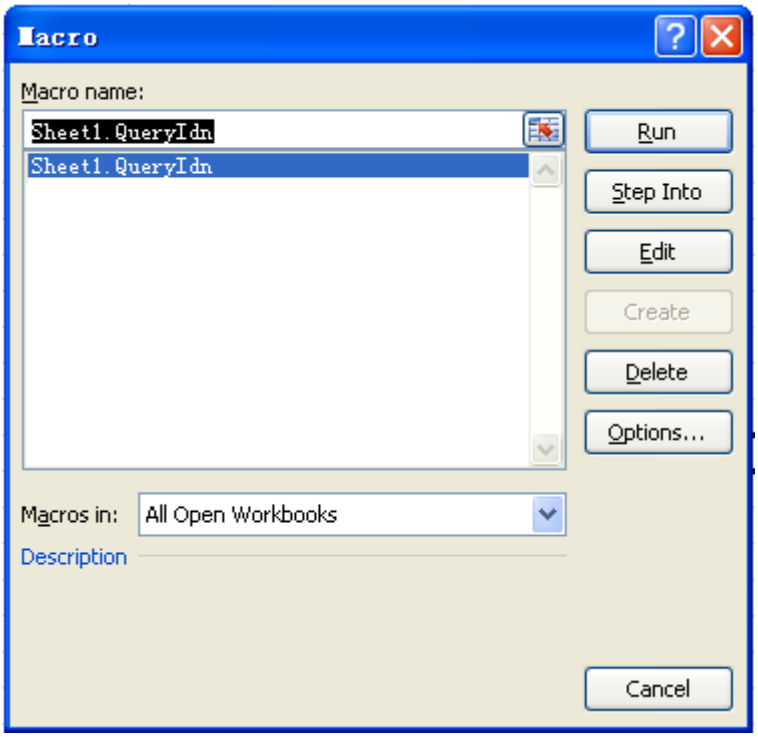
If the Excel file created in Step 2 does not enable the Macros, a prompt message "The following features cannot be saved in macro-free workbooks" will be displayed. In this case, please save the file as a macro-enabled file type (filename with a suffix of ".xlsm").

```
Sub QueryIdn()

    Dim viDefRm As Long
    Dim viDevice As Long
    Dim viErr As Long
    Dim cmdStr As String
    Dim idnStr As String * 128
    Dim ret As Long
    'Turn on the device, and the device resource descriptor is in
CELLS(1,2) of SHEET1'
    viErr = visa.viOpenDefaultRM(viDefRm)
    viErr = visa.viOpen(viDefRm, Sheet1.Cells(1, 2), 0, 5000,
viDevice)

   'Send request, read the data, and the return value is in
CELLS(2,2) of SHEET1'
    cmdStr = "*IDN?"
    viErr = visa.viWrite(viDevice, cmdStr, Len(cmdStr), ret)
    viErr = visa.viRead(viDevice, idnStr, 128, ret)
    Sheet1.Cells(2, 2) = idnStr

   'Turn off the device'
    visa.viClose (viDevice)
    visa.viClose (viDefRm)
```

```
End Sub
```

6 Add the button control. Click Insert under the Developer menu, and select a button control under the Form Controls menu item and put it into the Excel cell. At this time, the Assign Macro dialog box is displayed, select "Sheet1.QueryIdn" and click OK.





The default name of the button is "Button1". Right-click the button and select Edit Text in the pop-up menu to change the button name to "*IDN?".

7 Click the "*IDN?" button to send request and read data. The returned value is in CELLS(2,2) of SHEET1.

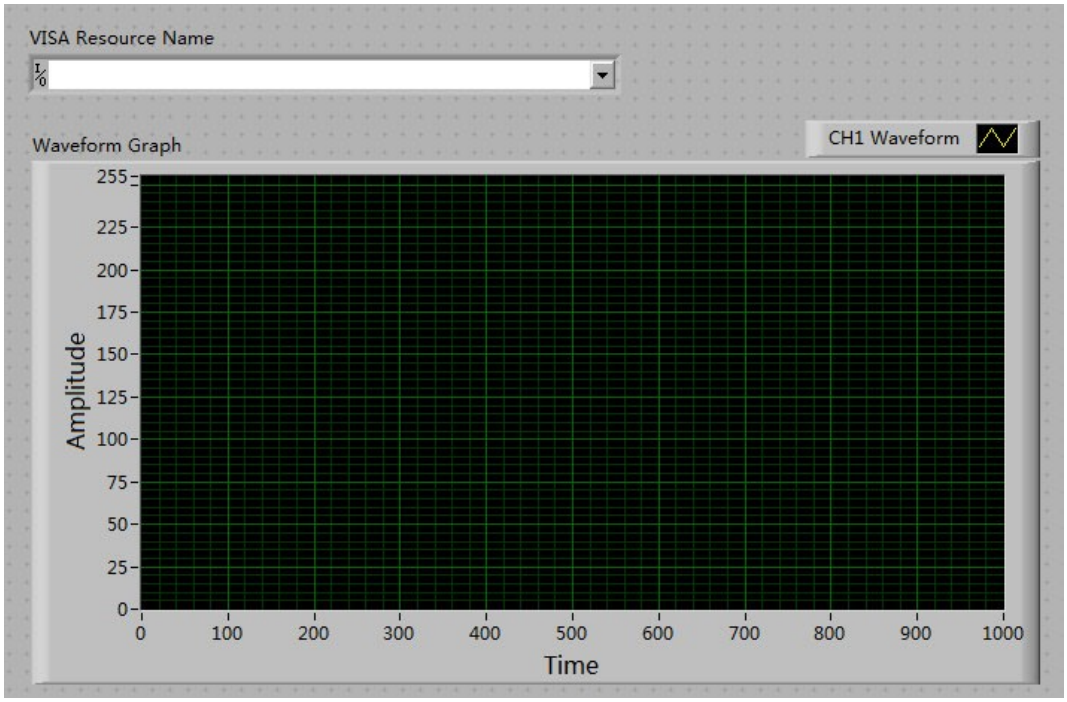
4.3 LabVIEW Programming Example

Program used in this example: LabVIEW2010

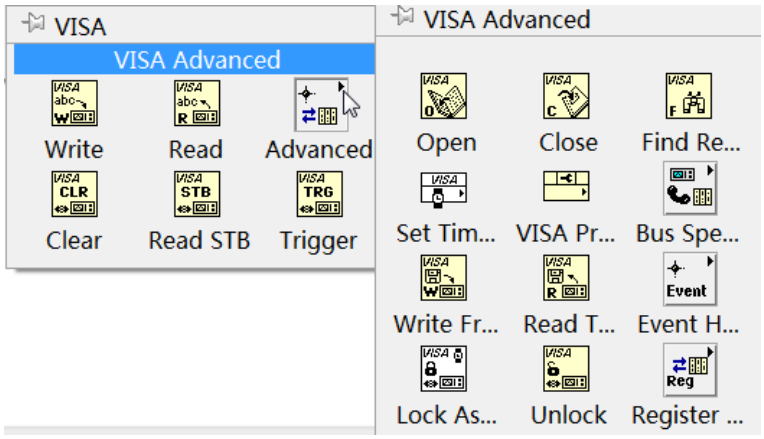
Function realized in this example: reading the waveform data of CH1 on the screen.

1 Run LabVIEW, and then create a VI file named LabVIEW_Demo.

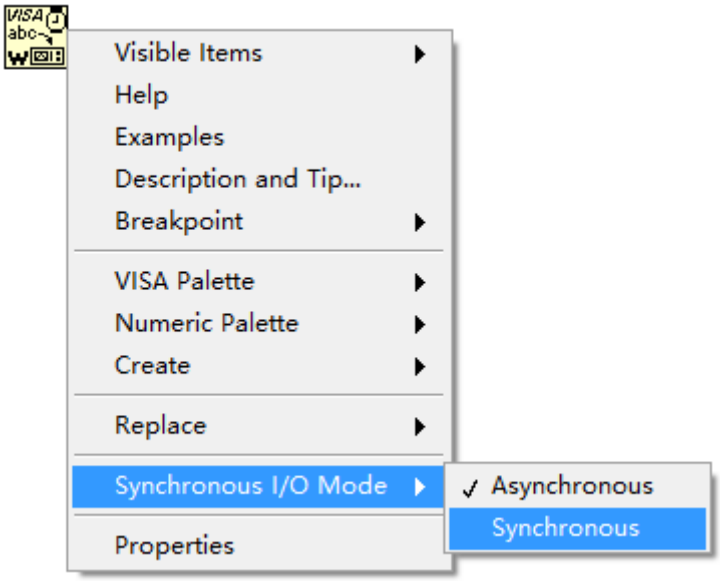
2 Add controls and create the front panel as shown in the figure below.



3 Open the Block Diagram panel. Click Instrument I/O > VISA. Add the following functions: VISA Open, VISA Read, VISA Write, and VISA Close.

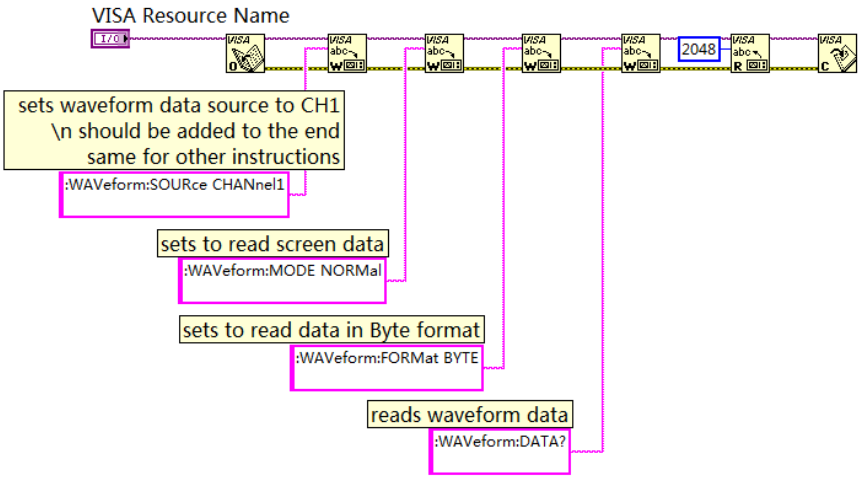


4 Change the default Asynchronous to Synchronous.

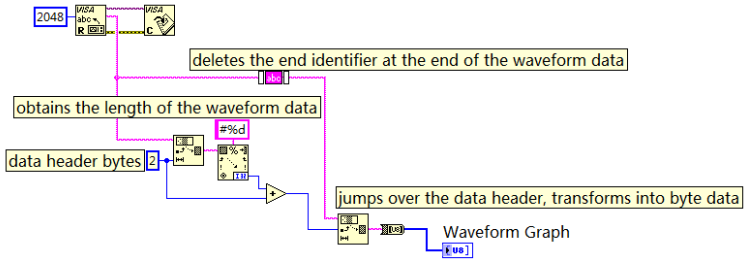


5 Connect the VISA resource name with the VISA Open. Then, connect the VISA resource name outputs of all the functions with the VISA resource name and connect the error output with the error input, as shown in the figure below.

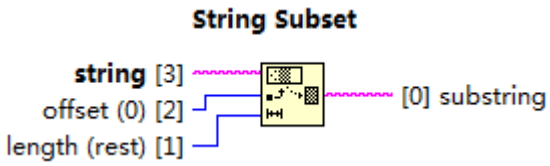
6 Add string constant in the write buffer areas of the VISA Write function and input the following instructions in the figure below. Waveform data is read through the VISA Read function which requires users to input the total number of bytes to be read. In this example, the total number of bytes of waveform data to be read is less than 2048. Use the VISA Close function to close the VISA resource after the VISA operation is finished.



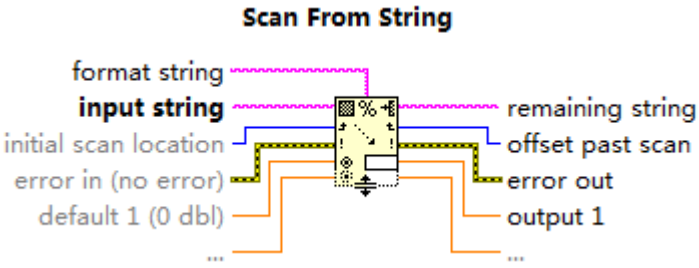
7 The data format is TMC data block header + waveform data + terminator. The TMC header is in #NXXXXXX format; wherein, # is the TMC header identifier; N following # represents the the length of the waveform data; the length of the waveform data points is expressed in ASCII strings,and the terminator represents the ending of communication. For example, the data read for one time is #9000001000XXXX. It indicates that 9 bytes are used to describe the data length. 000001000 indicates the length of waveform data, i.g. 1,000 bytes. Use the following block diagram to obtain the number of bytes that the TMC header occupies. Ignore the TMC header and delete the terminator at the end of the waveform data, and transfer the waveform data to the byte data and display it on the waveform diagram controls.



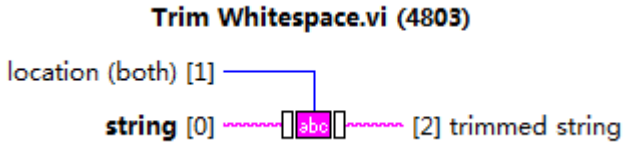
The available functions used in the above block diagram as as follows:



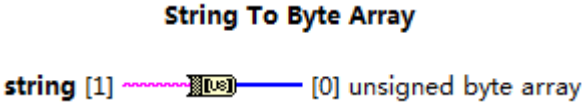
Used to obtain the TMC header"#N". After obtaining the number of bytes that the TMC header occupies, ignore the data header to obtain the waveform data strings



Used to obtain the waveform data length bytes

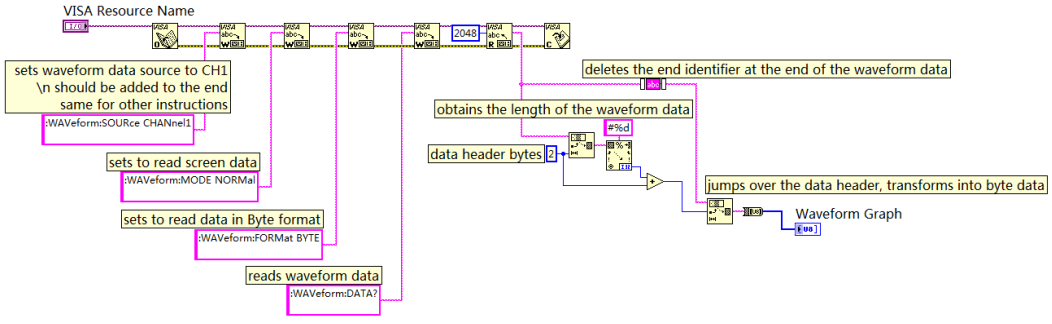


Used to delete the terminator at the end of the waveform data

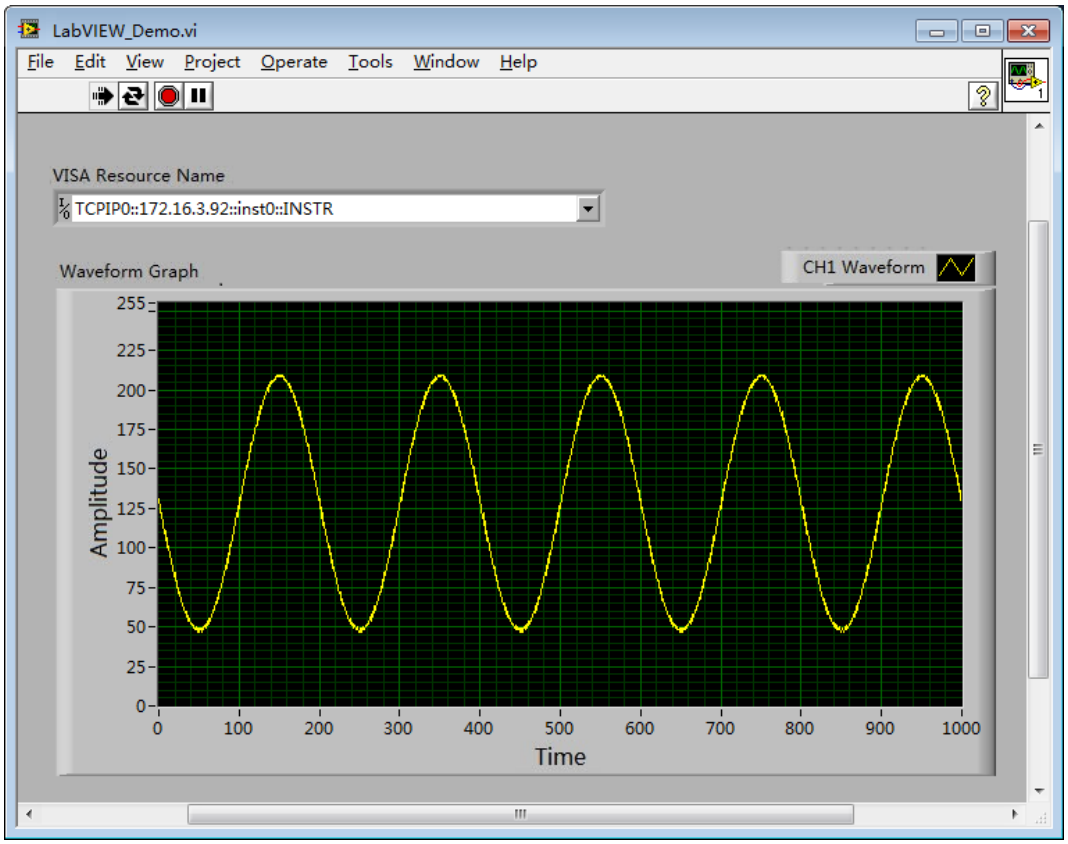


Used to transfer waveform data strings to the byte group

8 The complete program block diagram is as shown in the figure below:



9 Select the device resource from the VISA Resource Name drop-down list and run the program.



4.4 Visual Basic Programming Example

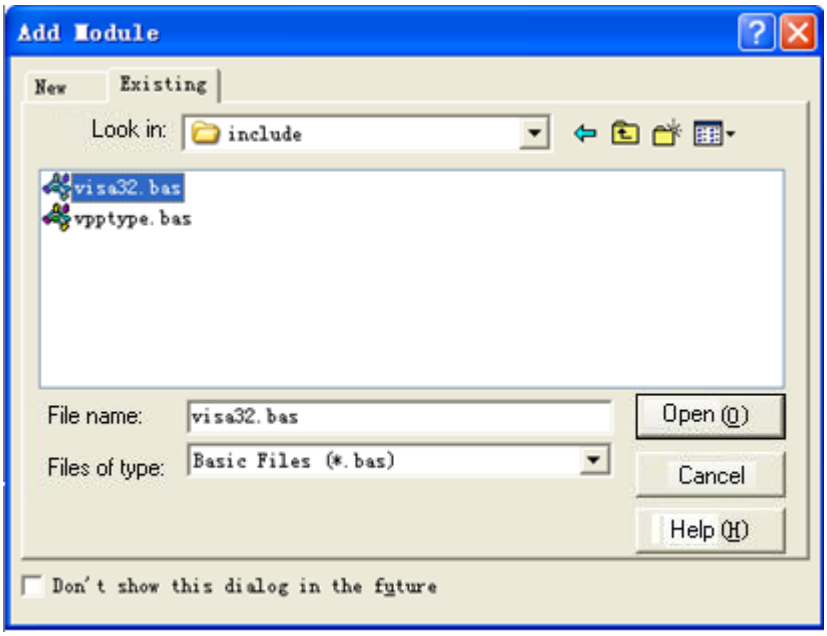
Program used in this example: Visual Basic 6.0

Function realized in this example: control the on/off state of any channel.

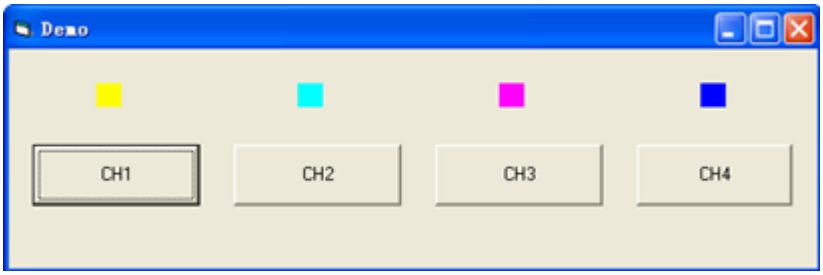
Enter the Visual Basic 6.0 programming environment, and perform the following procedures.

1 Build a standard application program project (Standard EXE), and name it "Demo".

2 Click **Project** > **Add Module** to open the Add Module dialog box. In the dialog box, click the Existing tab to search for the **visa32.bas** file in the include folder under the NI-VISA installation path and add the file.



3 In the Demo dialog box, add four buttons to represent CH1 to CH4 respectively. Add four Labels (Label1(0), Label1(1),Label1(2),Label1(3)) to represent the statuses of CH1 to CH4 respectively (when the channel is enabled, it displays the color of the channel; when the channel is disabled, it displays gray), as shown in the figure below.



4 Click**Project** > **Project1 Properties** to open the Project1 – Project Properties dialog box. In the dialog box, click on the General tab and select **Form1** from the drop-down list under Startup Object.

5 Double-click CH1 to enter the programming environment. Add the following codes to control CH1-CH4. The codes of CH1 are as shown below; the codes of the other channels are similar.

```
Dim defrm As Long
Dim vi As Long
Dim strRes As String * 200
Dim list As Long
Dim nmatches As Long
Dim matches As String * 200 'Reserve the obtained device number
Dim s32Disp As Integer
' Obtain the usb resource of visa
Call viOpenDefaultRM(defrm)
Call viFindRsrc(defrm, "USB?*", list, nmatches, matches)
' Turn on the instrument
Call viOpen(defrm, matches, 0, 0, vi)
```

```
' Send a command to query the status of CH1
Call viVPrintf(vi, ":CHAN1:DISP?" + Chr$(10), 0)
' Obtain the status of CH1
Call viVScanf(vi, "%t", strRes)
s32Disp = CInt(strRes)
If (s32Disp = 1) Then
' Send the setting command
Call viVPrintf(vi, ":CHAN1:DISP 0" + Chr$(10), 0)
Label1(0).ForeColor = &H808080 ''Gray
Else
Call viVPrintf(vi, ":CHAN1:DISP 1" + Chr$(10), 0)
Label1(0).ForeColor = &HFFFF& 'Yellow
End If
' Close the resource
Call viClose(vi)
Call viClose(defrm)
```

6 Save and run the project to obtain a single exe program for demo. When the oscilloscope is correctly connected to the PC, you can control the on/off status of any channel.

4.5 Visual C++ Programming Example

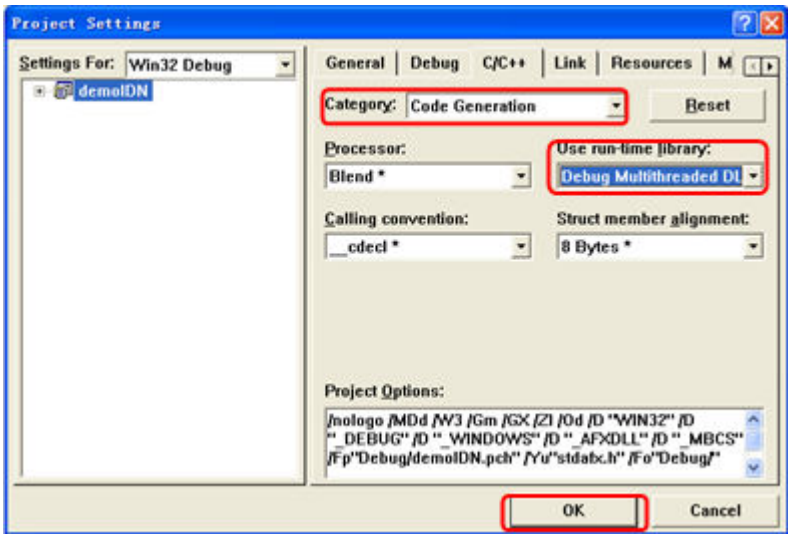
Program used in this example: Visual C++6.0

Function realized in this example:search for the instrument address, connect to the instrument, send commands, and read return values.

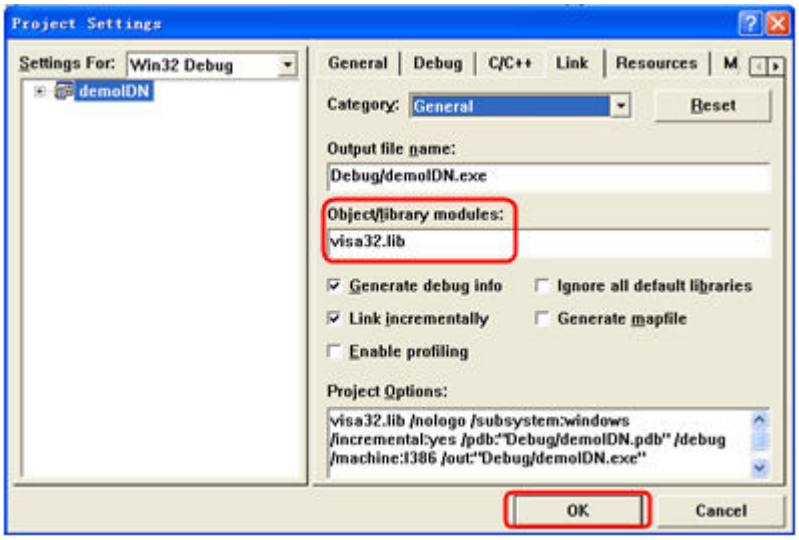
Enter the Visual C++6.0 programming environment, and perform the following procedures.

1 Create a MFC project based on a dialog box.

2 Click Project > Settings to open the Project Setting dialog box. In the dialog box, click the C/C++ tab, select Code Generation from the drop-down list under Category. Choose Debug Multithreaded DLL from the drop-down list under Use run-time library. Click OK to close the dialog box.



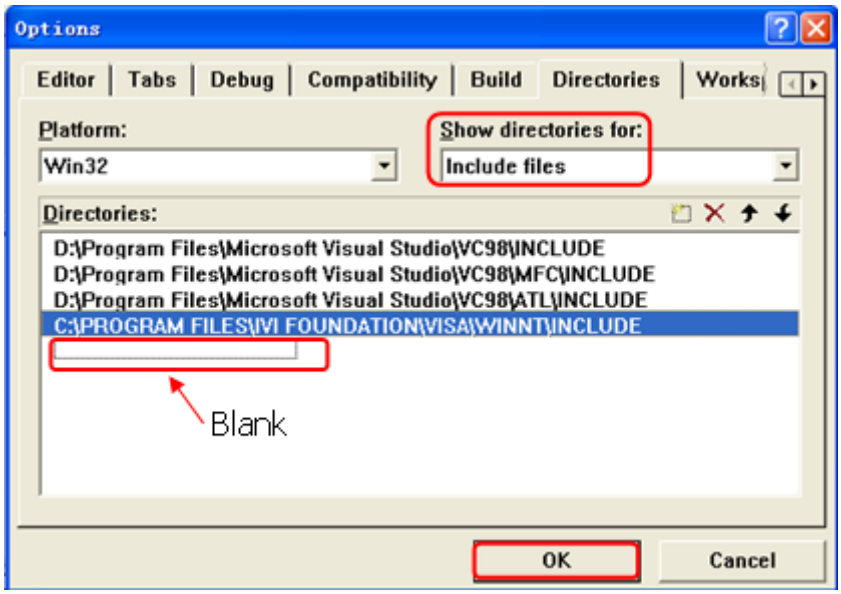
3 Click Project > Settings to open the Project Setting dialog box. In the dialog box, click the Link tab, add "visa32.lib" under Object/library modules, then click OK to close the dialog box.



4 Click Tools > Options to open the Options dialog box. Then click the Directories tab.

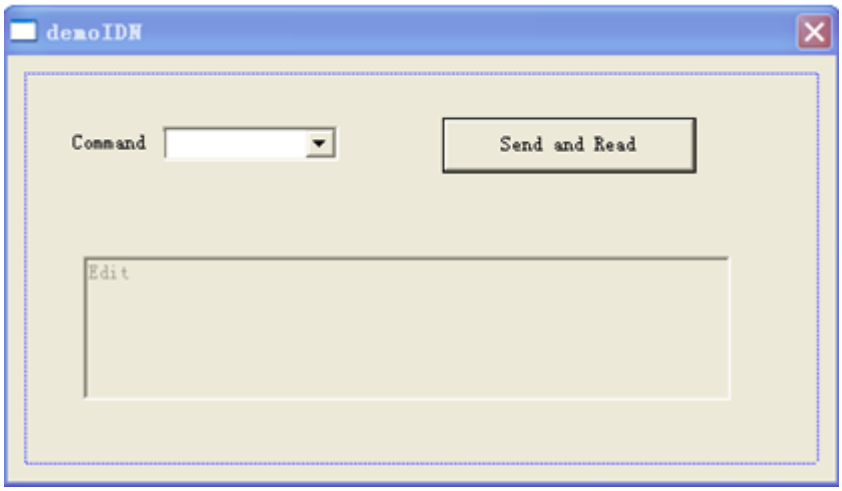
Select Include files from the drop-down list under Show directories for. Double click the empty space under Directories to enter the specified path of Include files: C:\Program Files\IVI Foundation\VISA\WinNT\include. Click OK to close the dialog box.

Select Library files from the drop-down list under Show directories for. Double click the empty space under Directories to enter the specified path of Library files: C:\Program Files\IVI Foundation\VISA\WinNT\lib\msc. Click OK to close the dialog box.



Note: By now, VISA library has been added.

5 Add Static Text, Combo Box, Button, and Edit Box controls. The layout interface for adding controls is as follows:



6 Modify the control attributes.

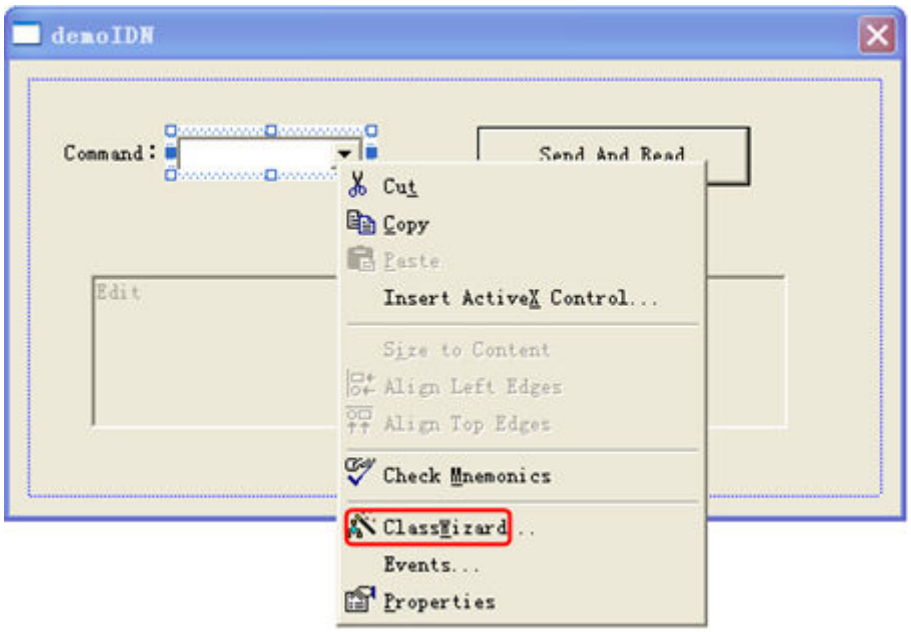
a Name Text as "Command".

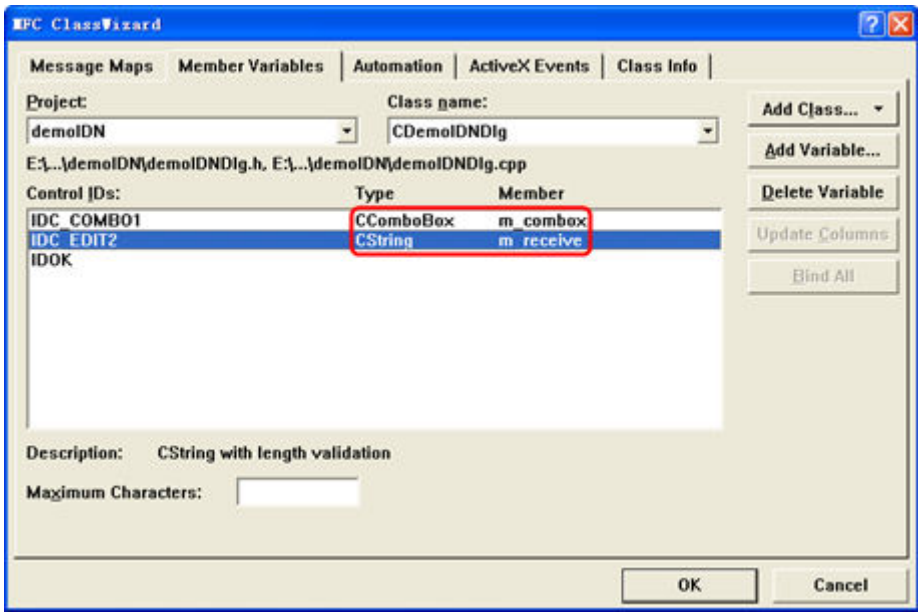
b Open the Data item in the Com box attribute and input the following command manually: *IDN?

c Open the General item in the Edit Box attribute and select Disabled.

d Name Button as Send and Read.

7 Add the variables m_combox and m_receive to the Com Box and Edit Box controls respectively.





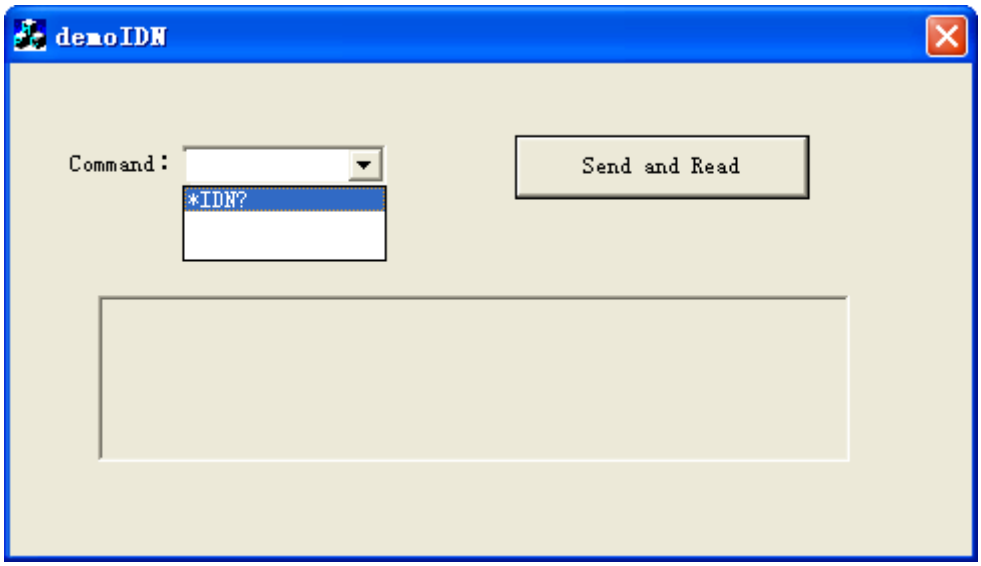
8 Add codes.

Double-click Send and Read to enter the programming environment. Declare the #include <visa.h> of the VISA library in the header file and then add the following codes:

```
ViSession defaultRM, vi;
char buf [256] = {0};
CString s,strTemp;
char* stringTemp;
ViChar buffer [VI_FIND_BUFLEN];
ViRsrc matches="buffer";
```

```
ViUInt32 nmatches;
ViFindList list;
viOpenDefaultRM (&defaultRM);

//Acquire the USB resource of VISA
viFindRsrc(defaultRM, "USB?*", &list,&nmatches, matches);
viOpen (defaultRM,matches,VI_NULL,VI_NULL,&vi);
viPrintf (vi, "*RST\n");

//Send the command received
m_combox.GetLBText(m_combox.GetCurSel(),strTemp);
strTemp = strTemp + "\n";
stringTemp = (char *)(LPCTSTR)strTemp;
viPrintf (vi,stringTemp);

//Read the results
viScanf (vi, "%t\n", &buf);

//Display the results
UpdateData (TRUE);
m_receive = buf;
UpdateData (FALSE);
viClose (vi);
viClose (defaultRM);
```

9 Save, compile, and run the project to obtain a single exe file. When the oscilloscope is correctly connected to the PC, enter a command (for example, *IDN?) and click Send and Read to execute the command. Then, the reading results will be returned.



HEADQUARTER

RIGOL TECHNOLOGIES CO., LTD.
No.8 Keling Road, New District, Suzhou,
JiangSu, P.R.China
Tel: +86-400620002
Email: info@rigol.com

EUROPE

RIGOL TECHNOLOGIES EU GmbH
Carl-Benz-Str.11
82205 Gilching
Germany
Tel: +49(0)8105-27292-0
Email: info-europe@rigol.com

NORTH AMERICA

RIGOL TECHNOLOGIES, USA INC.
8140 SW Nimbus Ave.
Beaverton, OR 97008
Tel: +1-877-4-RIGOL-1
Fax: +1-877-4-RIGOL-1
Email: info@rigol.com

JAPAN

RIGOL TECHNOLOGIES JAPAN, LLC
501, LATORRETTA, 2-37-1, Numabukuro,
Nakano-Ku, Tokyo, Japan
Tel: +81-3-6262-8932
Fax: +81-3-6262-8933
Email: info-japan@rigol.com