RIGOL

Distribution in the UK & Ireland

Lambda

Lambda Photometrics Limited
Lambda House Batford Mill
Harpenden Herts AL5 5BZ
United Kingdom
E: info@lambdaphoto.co.uk
W: www.lambdaphoto.co.uk
T: +44 (0)1582 764334
F: +44 (0)1582 712084

MHO900 Series

Digital Oscilloscope

Programming Guide

Sept. 2025

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

List of Tables

1 Document Overview

This manual is your guide to programming MHO900 series digital oscilloscope by using SCPI commands through remote interface. This series oscilloscope communicates with the PC via the USB and LAN interfaces.

TIP

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

Publication Number

PGA46100-1110

Software Version

00.01.00

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, 

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**.

4. **Connector**

   The connectors on the front or rear panel are denoted by the format of "Connector Name (Bold) + Square Brackets (Bold)". For example, **[AUX OUT]**.

Content Conventions in this Manual

MHO900 series digital oscilloscope includes the following models. Unless otherwise specified, this manual takes MHO984 as an example to illustrate the functions and operation methods of MHO900 series oscilloscope.

Model	Max. Analog Bandwidth	No. of Analog Channels
MHO984	800 MHz (single-channel[1]& half-channel[2]) 400 MHz (all-channel[3])	4
MHO954	500 MHz (single-channel[1]& half-channel[2]) 400 MHz (all-channel[3])	4
MHO934	350 MHz (for any number of channels)	4

NOTE

[1]: Single-channel mode: If any one of the channels is enabled, it is called single-channel mode.

[2]: Half-channel mode: If two of the channels are enabled, it is called half-channel.

[3]: All-channel mode: If all of the channels are enabled or any three of the channels are enabled, it is called all-channel mode.

2 Programming Overview

2.1 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 certain 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,

   :SYSTem:PSTatus <*sat*>

   :SYSTem:PSTatus?

   Wherein,

   - <*sat*> can be set to DEFault|OPEN.
   - The query returns an abbreviated form: DEF or OPEN.

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 (16 ns) to 1E +1 (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,

   :LAN:GATeway <*string*>

   Wherein, <*string*> can be set to

   ```
   192.168.1.1
   ```

Command Abbreviation

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

:DISPlay:GBRightness?

can be abbreviated as

:DISP:GBR?

2.2 Remote Control

The instrument can communicate with the PC via the USB interfaceand LAN interface to realize remote control of the instrument by using the SCPI (Standard Commands for Programmable Instruments) commands.

Web Control

When the instrument is connected to the PC via the LAN interface, you can use Web Control to send SCPI commands from the PC to the instrument. The operation procedures are as follows:

1. Obtain the instrument's IP address and input it in the browser address bar to log in to the Web Control page.
2. After you enter the Web Control interface, click the "SCPI Panel Control" button to enter the SCPI Command interface.

3. Input the specified SCPI command and then click **Send & Read** to send the command. The operation process and the returned value will be displayed in the current interface.

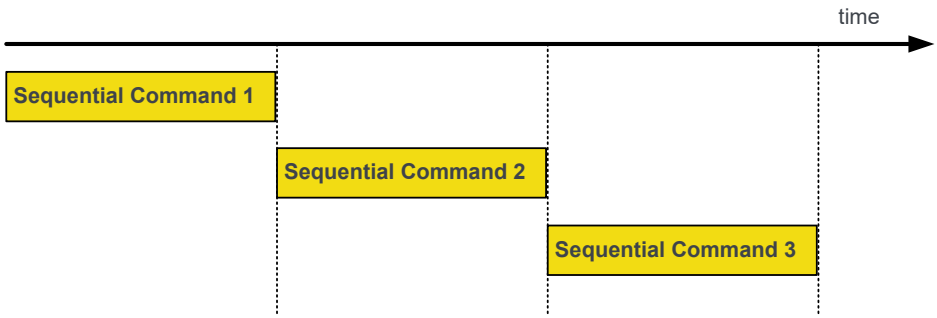
2.3 Sequential Commands and Overlapped Commands

IEEE488.2 defines the differentiation between the Sequential commands and the Overlapped commands.

- **Sequential Commands**

  Commands that shall be executed in sequence. Before completing the execution of one sequential command, another command shall not be executed. A command is a Sequential Command when Execution Control waits for the resulting device action to complete before it initiates the next device action.

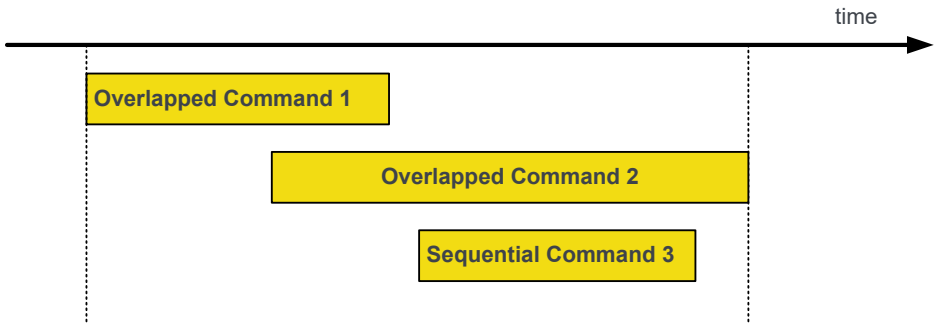
  The following figure shows the example of the sequence order for the sequential command.



- **Overlapped Commands**

  Commands that can be executed with several other commands being executed in parallel. During the execution process, several commands may be overlapped with each other. An overlapped command is a command that allows execution of subsequent commands while the device operations initiated by that overlapped command are still in progress.

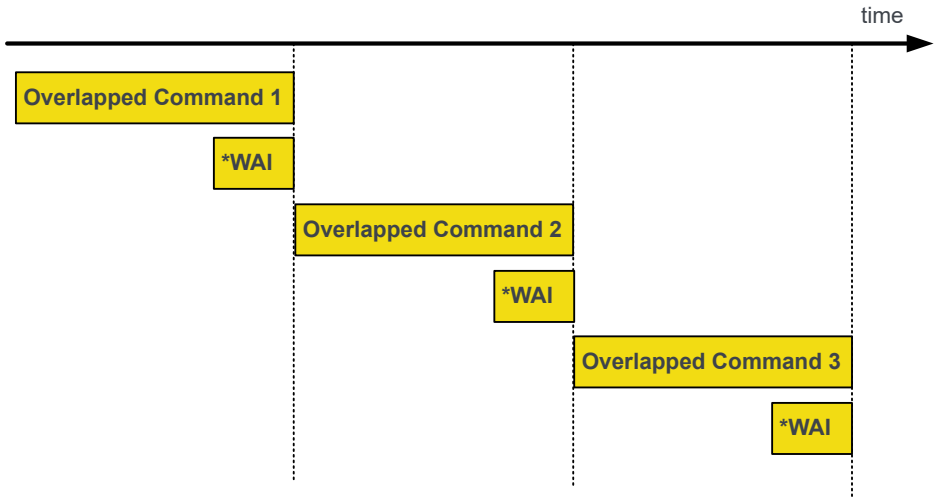
  The following figure shows the example of the sequence order for the overlapped command.



To ensure that the instrument functions can be synchronized through the commands executed in the specified order and query the command execution status, IEEE488.2 defines three general commands (**WAI*, **OPC*, and **OPC?*) to check the operation complete (OPC) function of the instrument.

- ***WAI**: The *WAI command shall prevent the instrument from executing further commands or queries until all the received commands have been completely executed. The *WAI command will not operate on the register.

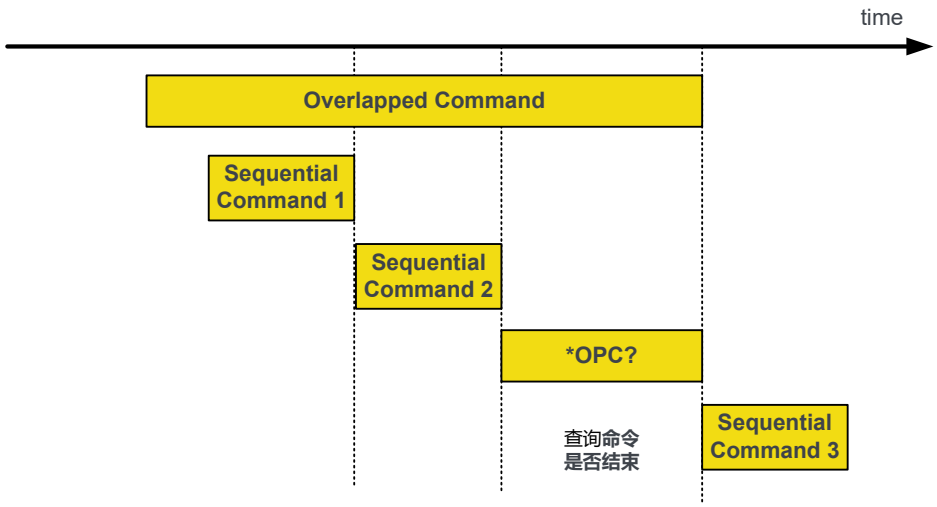
  If you want the overlapped commands to be executed in sequence just like the sequential commands, you can send the *WAI command before or after the specified overlapped command. For example,



- ***OPC**: Upon receiving the *OPC command, the instrument will wait until all the previously received commands have been completely executed. Meanwhile, it will set Bit 0 in the *Standard Event Register* to 1, and then execute the following commands.
- ***OPC?**: Upon receiving the *OPC? Command, the instrument will query whether all the previous received commands have been completely executed. If

completed, the query returns 1; if not, it will wait until all the pending operations have been completed and then returns 1. The *OPC? command shall not operate on the register.

If you want to query the execution status of a series of commands and control the execution sequence of the command, send the *OPC? Command. For example,



The following table lists the overlapped commands that support the Operation Complete (OPC) function. While using the overlapped commands, pay attention to the execution order of the command to avoid causing any abnormalities to the results.

Command	Constraint Condition
*RST	-

3 Command System

This chapter introduces the syntax, functions, parameters, and usage of each command. By default, only CH1 is enabled.

NOTE

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 Root Commands

Root level commands only have the root keywords, without the next level keywords. They control many of the basic operations of the instrument.

3.1.1 :CLEar

Syntax

```
:CLEar
```

Description

Clears all the waveforms on the screen.

This command functions the same as the front-panel key 

.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.1.2 :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; or pressing the 

key on the front panel.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.1.3 :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; or pressing the 

key on the front panel.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.1.4 :SINGle

Syntax

```
:SINGle
```

Description

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

This command functions the same as the front-panel key 

.

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 the trigger forcibly.

Return Format

N/A

Example

```
N/A
```

3.1.5 :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).

This command functions the same as the 

key in the trigger control area of the front panel.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.2 :AUToset Commands

The **:AUToset** commands are used to perform waveform auto setting operations.

3.2.1 :AUToset

Syntax

```
:AUToset
```

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 the front-panel key 

.

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
```

3.2.2 :AUToset:PEAK

Syntax

:AUToset:PEAK <*bool*>

:AUToset:PEAK?

Description

Sets or queries whether the peak-peak priority setting is enabled.

Parameter

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

Remarks

This function is intended for the shifted signal. If there is a large deviation, you can view the signal waveform in priority when you enable the function.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:PEAK OFF /*Disables the peak-peak priority setting.*/
:AUToset:PEAK? /*The query returns 0.*/
```

3.2.3 :AUToset:OPENch

Syntax

:AUToset:OPENch <*bool*>

:AUToset:OPENch?

Description

Sets or queries whether to only test the enabled channel when performing the AUTO operation.

Parameter

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

Remarks

- If you select "0|OFF", the system will test the four analog channels (CH1-CH4) in sequence when performing the AUTO operation. If no signal is found on the channel, then the channel is disabled. If a signal is found on the channel, adjust the channel to an optimal scale to show the signal.
- If you select "1|ON", the system will only test the enabled channels when performing the AUTO operation.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:OPENch ON /*Sets to test only the enabled channel(s) for
AUTO operation.*/
:AUToset:OPENch? /*The query returns 1.*/
```

3.2.4 :AUToset:OVERlap

Syntax

:AUToset:OVERlap <*bool*>

:AUToset:OVERlap?

Description

Sets or queries whether to enable the waveform display mode.

Parameter

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

Remarks

- **1|ON:** enables the waveform overlay display. Waveforms of different channels will be displayed in the same position of the screen.

- **0|OFF:** disables the waveform overlay display. Waveforms of different channels will be displayed on the screen from top to bottom in sequence.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:OVERlap OFF /*Disables the waveform overlay display.*/
:AUToset:OVERlap? /*The query returns 0.*/
```

3.2.5 :AUToset:KEEPcoup

Syntax

:AUToset:KEEPcoup <*bool*>

:AUToset:KEEPcoup?

Description

Sets or queries whether to enable keeping coupling.

Parameter

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

Remarks

- **1|ON:** Enables keeping coupling. When enabled, the system performs auto setting operation. The settings for the channel coupling remain unchanged.
- **0|OFF:** Disables keeping coupling. When disabled, the channel is, by default, DC coupled.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:KEEPcoup ON /*Sets to enable keeping coupling.*/
:AUToset:KEEPcoup? /*The query returns 1.*/
```

3.2.6 :AUToset:LOCK

Syntax

:AUToset:LOCK <*bool*>

:AUToset:LOCK?

Description

Sets or queries the on/off status of the AUTO function.

Parameter

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

Remarks

- **1|ON:** locks the AUTO key; disables the AUTO function.
- **0|OFF:** unlocks the AUTO key; enables the AUTO function.

You can also send the *:AUToset:ENAble* command to enable or disable the AUTO function.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:LOCK ON  /*Locks the AUTO key; disables the AUTO
function.*/
:AUToset:LOCK? /*The query returns 1.*/
```

3.2.7 :AUToset:ENAble

Syntax

:AUToset:ENAble <*bool*>

:AUToset:ENAble?

Description

Sets or queries whether to enable the AUTO function.

Parameter

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

Remarks

- **1|ON:** enables the AUTO function. It functions the same as enabling the front-panel key AUTO.

- **0|OFF:** disables the AUTO function. It functions the same as disabling the front-panel key 

.

You can also send the *:AUToset:LOCK* command to enable or disable the AUTO function.

Return Format

The query returns 0 or 1.

Example

```
:AUToset:ENAble OFF /*Enables the AUTO function.*/
:AUToset:ENAble? /*The query returns 0.*/
```

3.3 :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.

3.3.1 :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.
- The number of averages must be in the Nth power of 2. When the value is not in the Nth power of 2, a value that is smaller than the one you input and the closest

to the N power-of-2 increments will be input automatically. For example, if you input 9 with the numeric keypad, the average count will be input 8 automatically.

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.*/
```

3.3.2 :ACQuire:MDEPth

Syntax

:ACQuire:MDEPth <*mdep*>

:ACQuire:MDEPth?

Description

Sets or queries the memory depth of the oscilloscope (that is, 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|25M|50M|100M|125M|200M|250M|500M|1000|10000|100000|1000000|10000000|25000000|50000000|100000000|125000000|200000000|250000000|500000000|1e3|1e4|1e5|1e6|1e7|2.5e7|5e7|1e8|125e6|1.25e8|2e8|250e6|2.5e8|5e8}	10k

Remarks

When you select the "Auto" mode, the oscilloscope selects the memory depth automatically according to the current sample rate.

The max. memory depth is:

- When the 500 Mpts memory depth option is enabled: 500 Mpts in single-channel mode, 250 Mpts in half-channel mode, and 125 Mpts in full-channel mode.

- When the 500 Mpts memory depth option is disabled: 100 Mpts in single-channel mode, 50 Mpts in half-channel mode, and 25 Mpts in full-channel mode.

NOTE

When any one of the channels is enabled, it is called single-channel mode; when any two of the channels are enabled, it is called half-channel mode; when any three of the channels or all of the four channels are enabled, it is called full-channel mode.

Modifying the memory depth will affect the sample rate. To query the current sample rate, run the *:ACQuire:SRATe?* command.

Return Format

The query returns the memory depth in scientific notation.

Example

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

3.3.3 :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 noise and increase the

vertical resolution; while at the same time, it will 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 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, HRES.

Example

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

3.3.4 :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, that is, the number of waveform points sampled per second.
- The sample rate and memory depth will change accordingly in accordance with the horizontal time base.

Return Format

The query returns the sample rate in scientific notation.

Example

```
:ACQuire:SRATe?   /*The query returns 1.00000E+6.*/
```

3.3.5 :ACQuire:BITS

Syntax

:ACQuire:BITS <*bit*>

:ACQuire:BITS?

Description

Sets or queries the resolution bits supported by the oscilloscope in high-resolution mode.

Parameter

Name	Type	Range	Default
<bit>	Discrete	{14|16}	14

Remarks

This oscilloscope supports 14 bits and 16 bits in high-resolution mode; 12 bits in other modes.

Return Format

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

Example

```
:ACQuire:BITS 16    /*Sets the resolution in high-resolution mode
to 16 bits.*/
:ACQuire:BITS?         /*The query returns 16.*/
```

3.4 :BUS<n> Commands

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

3.4.1 :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|LIN|CAN|IIS|FLEXray|M1553}	PARallel

Remarks

Parallel, RS232, SPI, I2C, LIN, and CAN decodings are standard configurations. Other decodings are optional configurations. Only when the specified option has been installed, can this command be available.

Return Format

The query returns PAR, RS232, SPI, IIC, LIN, I2S, FLEX, M1553, or CAN.

Example

```
:BUS1:MODE SPI     /*Sets the decoding type of Bus 1 to SPI.*/
:BUS1:MODE?        /*The query returns SPI.*/
```

3.4.2 :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.*/
```

3.4.3 :BUS<n>:FORMat

Syntax

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

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

Description

Sets or queries the format of decoding data on 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;
- **ASCii:** indicates ASCII;
- **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.*/
```

3.4.4 :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.*/
```

3.4.5 :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.*/
```

3.4.6 :BUS<n>:DATA?

Syntax

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

Description

Reads the data from 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, and etc. The data are separated by a comma, and will automatically switch to the next line according to the data information in the decoding list. The data value is related to the numeral system that you have 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
```

3.4.7 :BUS<n>:EEXPort

Syntax

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

Description

Exports the decoding information from 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, you can use the command *:SAVE:OVERlap* to set query to overwrite the existing file.
- When the operating status of the instrument is STOP (set it by sending the *:STOP* command), you can export the time and corresponding decoding data from the current event table.
- This command is valid when the display of the event table is enabled. You can enable the display of the event table by sending the *:BUS<n>:EVENt* command.
- 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 from
the bus event table and saves it to the local Disk C, with the
filename 123.csv.*/
```

3.4.8 :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.*/
```

3.4.9 :BUS<n>:THReshold

Syntax

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

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

Description

Sets or queries the threshold of the specified decoding source on the specified decoding bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<value>	Real	(-5 x VerticalScale - OFFSet) to (5 x VerticalScale - OFFSet)	0
<type>	Discrete	{PAL|TX|RX|SCL|SDA|CS|CLK| MISO|MOSI|LIN|CAN|CANSub1| FLEX|1553|I2SClk|DATA|WS| PALCLK|CH1|CH2|CH3|CH4}	-

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

- **PAL:** indicates the bus source of Parallel decoding.
- **PALCLK:** indicates the clock source of Parallel decoding. Only when the source is enabled, can you set the threshold.
- **TX:** indicates the TX channel source of RS232 decoding.
- **RX:** indicates the RX channel source of RS232 decoding. Only when the RX source is enabled, can you set the threshold.
- **SCL:** indicates the clock source of I2C decoding.
- **SDA:** indicates the data source of I2C decoding.
- **DATA:** indicates the data source of I2S decoding.
- **WS:** indicates the audio channel of I2S decoding.
- **I2SClk:** indicates the clock source of I2S decoding.
- **CS:** indicates the source of the CS line of SPI decoding.
- **CLK:** indicates the clock source of SPI decoding.
- **MISO:** indicates the MISO data source of SPI decoding.
- **MOSI:** indicates the MOSI data source of SPI decoding.
- **LIN:** indicates the bus source of LIN decoding.
- **CAN|CANSub1:** indicates the source of CAN decoding.
- **1553:** indicates the channel source of M1553 decoding.
- **FLEX:** indicates the source of FlexRay decoding.
- **CH1|CH2|CH3|CH4:** CH1-CH4, available for all the decoding types.

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 Parallel
decoding source to 2.4 V.*/
:BUS1:THReshold? PAL       /*The query returns 2.400000E0.*/
```

3.4.10 :BUS<n>:PARallel

:BUS<n>:PARallel commands are used to set relevant parameters for parallel decoding.

Parallel bus consists of clock line and data line. As shown in the figure below, CLK is the clock line, whereas Bit0 and Bit1 are the 0 bit and 1st bit on the data line respectively. The oscilloscope will sample the channel data on the rising edge, falling edge, or the rising/falling edge of the clock and judge each data point (logic "1" or logic "0") according to the preset threshold level.

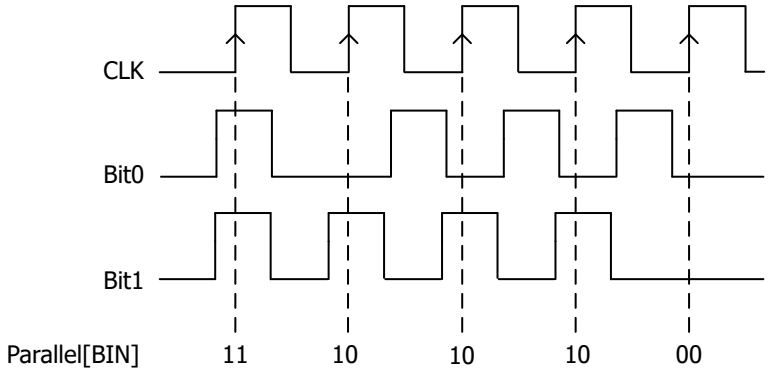


Figure 3.1 Schematic Diagram of Parallel Decoding

3.4.10.1 :BUS<n>:PARallel:BUS

Syntax

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

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

Description

Sets or queries the source of Parallel decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D7D0|D15D8|D15D0|D0D7| D8D15|D0D15|CHANnel1| CHANnel2|CHANnel3|CHANnel4| USER}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D7D0, D15D8, D15D0, D0D7, D8D15, D0D15,CHAN1, CHAN2, CHAN3, CHAN4 or USER.

Example

```
:BUS1:PARallel:BUS CHANnel1   /*Sets the source of Parallel
decoding on Bus 1 to CHANnel1.*/
:BUS1:PARallel:BUS?          /*The query returns CHAN1.*/
```

3.4.10.2 :BUS<n>:PARallel:CLK

Syntax

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

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

Description

Sets or queries the clock source of Parallel decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	OFF

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4 or OFF.

Example

```
:BUS1:PARallel:CLK CHANnel2    /*Sets the clock source of Parallel
decoding on Bus 1 to CHANnel2.*/
:BUS1:PARallel:CLK?          /*The query returns CHAN2.*/
```

3.4.10.3 :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 hopping occurs 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.*/
```

3.4.10.4 :BUS<n>:PARallel:WIDTh

Syntax

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

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

Description

Sets or queries the data width (number of bits per frame) of Parallel decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<wid>	Integer	1 to 20	1

Remarks

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

Return Format

The query returns an integer ranging from 1 to 20.

Example

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

3.4.10.5 :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

- Only when the bus source is set to User (*BUS<n>:PARallel:BUS USER*), can this command be valid.
- 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.*/
```

3.4.10.6 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	Related to the selected bit

Remarks

- Only when the bus source is set to User (*:BUS<n>:PARallel:BUS USER*), can this command be valid.
- Before sending this command, send the *:BUS<n>:PARallel:BITX* command to select the desired data bit.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.10.7 :BUS<n>:PARallel:ENDian

Syntax

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

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

Description

Sets or queries the endian of Parallel decoding on the specified decoding bus.

Parameter

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

Remarks

- **NEGative:** indicates invert.
- **POSitive:** indicates normal.

Return Format

The query returns NEG or POS.

Example

```
:BUS1:PARallel:ENDian POSitive /*Sets the endian of Parallel
decoding to POSitive*/
:BUS1:PARallel:ENDian? /*The query returns POS.*/
```

3.4.10.8 :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.*/
```

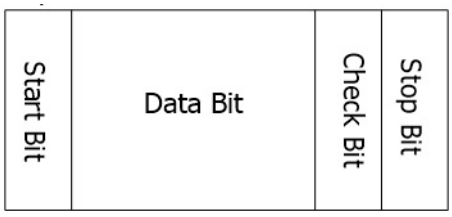
3.4.11 :BUS<n>:RS232

The :BUS<n>:RS232 commands are used to set relevant parameters for RS232 decoding.

RS232 serial bus consists of the transmitting data line (TX) and the receiving data line (RX).

Figure 3.2 Schematic Diagram of RS232 Serial Bus

In RS232, baud rate is used to represent the transmission rate (namely bits per second) of the data. In RS232, you need to set the start bit, data bits, check bit (optional), and stop bit for each frame of data.



- **Start Bit:** indicates when to output data.
- **Data Bits:** indicates the number of data bits actually contained in each frame of data.
- **Check Bit:** used to check whether the data are properly transmitted.
- **Stop Bit:** indicates when to stop outputting data.

3.4.11.1 :BUS<n>:RS232:TX

Syntax

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

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

Description

Sets or queries the Tx source of RS232 decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

The Tx and Rx sources cannot be set to OFF at the same time. The Rx source can be set by using the *:BUS<n>:RS232:RX* command.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4 or OFF.

Example

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

3.4.11.2 :BUS<n>:RS232:RX

Syntax

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

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

Description

Sets or queries the Rx source of RS232 decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	OFF

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

The Tx and Rx sources cannot be set to OFF at the same time. The Tx source can be set by using the *:BUS<n>:RS232:TX* command.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4 or OFF.

Example

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

3.4.11.3 :BUS<n>:RS232:POLarity

Syntax

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

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

Description

Sets or queries the polarity of RS232 decoding on the specified bus.

Parameter

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

Remarks

- **POSitive:** indicates positive polarity. High level is logic "1" and low level is logic "0".

- **NEGative:** indicates negative polarity. High level is logic "0" and low level is logic "1".
- 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.*/
```

3.4.11.4 :BUS<n>:RS232:PARity

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 in RS232 decoding to ODD.*/
:BUS1:RS232:PARity?              /*The query returns ODD.*/
```

3.4.11.5 :BUS<n>:RS232:ENDian

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, that is, the lowest bit of the data is transmitted first.
- **MSB:** indicates Most Significant Bit transmission sequence, that is, 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.*/
```

3.4.11.6 :BUS<n>:RS232:BAUD

Syntax

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

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

Description

Sets or queries the baud rate of RS232 decoding on the specified bus. 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

N/A

Return Format

The query returns an integer ranging from 1 to 20M.

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.*/
```

3.4.11.7 :BUS<n>:RS232:DBITs

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.*/
```

3.4.11.8 :BUS<n>:RS232:SBITs

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.*/
```

3.4.12 :BUS<n>:IIC

The :BUS<n>:IIC commands are used to set relevant parameters for I2C decoding.

I2C serial bus consists of the clock line (SCL) and the data line (SDA).

- **SCL:** samples SDA on the of rising or falling edge of the clock.
- **SDA**: indicates the data channel.

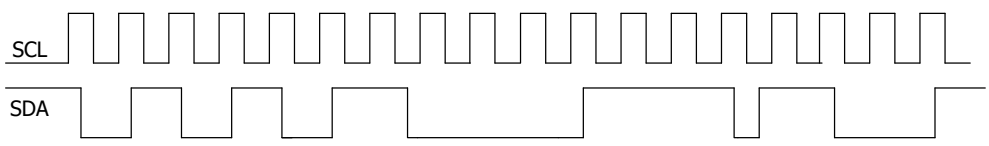


Figure 3.3 I2C Serial Bus

3.4.12.1 :BUS<n>:IIC:SCLK:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.12.2 :BUS<n>:IIC:SDA:SOURce

Syntax

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

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

Description

Sets or queries the data source of the I2C decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.12.3 :BUS<n>:IIC:EXCHange

Syntax

:BUS<*n*>**:IIC:EXCHange** <*bool*>

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

Description

Sets to exchange the clock channel source and the data channel source of I2C decoding on the specified bus; queries whether the clock channel source and the data channel source of I2C decoding on the specified bus have been exchanged.

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:IIC:EXCHange ON /*Sets to exchange the clock channel source
and the data channel source.*/
:BUS1:IIC:EXCHange? /*The query returns 1.*/
```

3.4.12.4 :BUS<n>:IIC:ADDBits

Syntax

:BUS<*n*>**:IIC:ADDBits** <*bits*>

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

Description

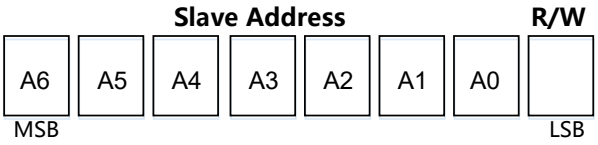
Sets or queries the address width of I2C decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<bits>	Discrete	{7|8|10}	7

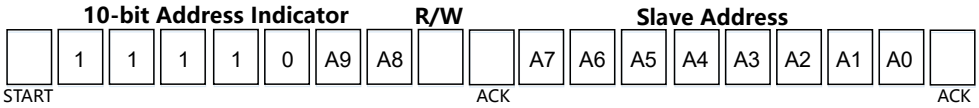
Remarks

- **7:** In 7-bit addressing, after the START condition, a slave address is sent. The address starts to transfer from the first byte, as shown in the figure below. The first seven bits of the first byte make up the slave address, and the eighth bit is the LSB (least significant bit) which determines the direction of the message, also called a data direction bit (R/W). A "zero" indicates a transmission (WRITE), a "one" indicates a request for data (READ).



- **8:** same as the 7-bit addressing. A R/W bit is included in the 8-bit addressing for the slave address.
- **10:** 10-bit addressing is compatible with, and can be combined with, 7-bit addressing. As shown in the figure below, in 10-bit addressing, the first byte is

the special reserved address 10-bit Address Indicator to indicate the current 10-bit address that is transferring.



Return Format

The query returns 7, 8, or 10.

Example

```
:BUS1:IIC:ADDBits 8     /*Sets the address width of I2C decoding on
Bus 1 to 8 bits.*/
:BUS1:IIC:ADDBits?        /*The query returns 8.*/
```

3.4.13 :BUS<n>:SPI

The :BUS<n>:SPI commands are used to set relevant parameters for SPI decoding.

SPI bus is based on the master-slave configuration and usually consists of chip select line (CS), clock line (CLK), and data line (SDA). Wherein, the data lines include the master input/slave output (MISO) data line and master output/slave input (MOSI) data line. The oscilloscope will sample the channel data on the rising edge, falling edge, or the rising/falling edge of the clock and judge each data point (logic "1" or logic "0") according to the preset threshold level).

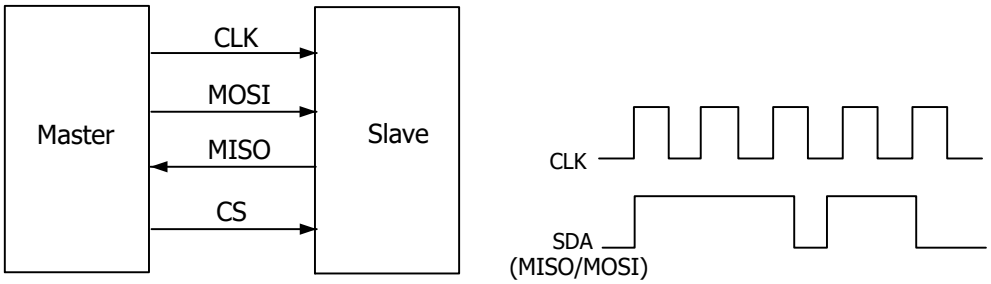


Figure 3.4 SPI Serial Bus

3.4.13.1 :BUS<n>:SPI:SCLK:SOURce

Syntax

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

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

Description

Sets or queries the clock source of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.13.2 :BUS<n>:SPI:SCLK:SLOPe

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.*/
```

3.4.13.3 :BUS<n>:SPI:MISO:SOURce

Syntax

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

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

Description

Sets or queries the MISO source of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

The source specified in this command and the *:BUS<n>:SPI:MOSI:SOURce* command cannot be set to OFF at the same time.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.13.4 :BUS<n>:SPI:MOSI:SOURce

Syntax

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

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

Description

Sets or queries the MOSI source of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	OFF

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

The source specified in this command and the *:BUS<n>:SPI:MISO:SOURce* command cannot be set to OFF at the same time.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.13.5 :BUS<n>:SPI:POLarity

Syntax

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

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

Description

Sets or queries the polarity of the SPI decoding on the specified decoding bus.

Parameter

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

Remarks

- **HIGH:** indicates positive polarity. The high level is 1, and low level is 0.
- **LOW:** indicates negative polarity. The high level is 0, and low level is 1.

Return Format

The query returns HIGH or LOW.

Example

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

3.4.13.6 :BUS<n>:SPI:MISO:POLarity

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:** positive polarity. It indicates that high level is 1, and low level is 0.
- **LOW:** negative polarity. It indicates that low level is 1, and high level is 0.

This command exists for backwards compatibility. Use the command *:BUS<n>:SPI:POLarity*.

Return Format

The query returns HIGH or LOW.

Example

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

3.4.13.7 :BUS<n>:SPI:MOSI:POLarity

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:** positive polarity. It indicates that high level is 1, and low level is 0.
- **LOW:** negative polarity. It indicates that low level is 1, and high level is 0.

This command exists for backwards compatibility. Use the command *:BUS<n>:SPI:POLarity*.

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.*/
```

3.4.13.8 :BUS<n>:SPI:DBITs

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}	-

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

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

3.4.13.9 :BUS<n>:SPI:ENDian

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, that is, the highest bit of the data is transmitted first.
- **LSB:** indicates Least Significant Bit transmission sequence, that is, 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.*/
```

3.4.13.10 :BUS<n>:SPI:MODE

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. It 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.

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.*/
```

3.4.13.11 :BUS<n>:SPI:TIMeout:TIME

Syntax

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

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

Description

Sets or queries the timeout value of SPI decoding on the specified bus. The unit is s.

Parameter

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

Name	Type	Range	Default
<time>	Real	8 ns to 10 s	1μs

Remarks

- This setting command is only valid in timeout mode. You can send *:BUS<n>:SPI:MODE* to set or query the decode mode of SPI decoding.
- The timeout must be greater than the maximum clock pulse width and less than the idle time between frames.

Return Format

The query returns the timeout value in scientific notation.

Example

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

3.4.13.12 :BUS<n>:SPI:SS:SOURce

Syntax

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

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

Description

Sets or queries the source of the CS line of SPI decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel3

Remarks

This setting command is only valid in CS mode. You can send *:BUS<n>:SPI:MODE* to set or query the decode mode of the SPI decoding.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

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

3.4.13.13 :BUS<n>:SPI:SS:POLarity

Syntax

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

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

Description

Sets or queries the polarity of CS line of SPI decoding on the specified bus.

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.

This setting command is only valid in CS mode. You can send *:BUS<n>:SPI:MODE* to set or query the decode mode of the SPI decoding.

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.*/
```

3.4.14 :BUS<n>:CAN

:BUS<n>:CANThe commands are used to set relevant parameters for CAN decoding.

The oscilloscope samples the CAN or CAN-FD signal in the specified sample position, and judges each data point to be logic "1" or logic "0" according to the set threshold level. The CAN decoding is required to specify the CANor CAN-FD signal type and the sample position.

Sample Position

Sample Position is a point within a bit's time. The oscilloscope samples the bit level at this point. The sample point position is expressed as the ratio of "time from the bit start to the sample point" to "bit time", in %.

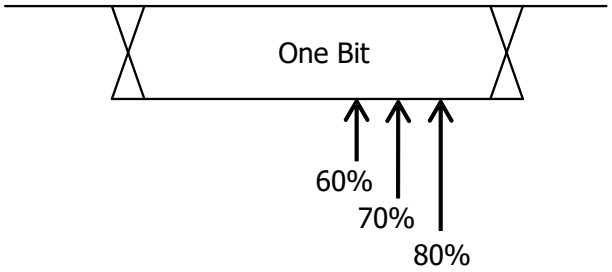


Figure 3.5 Sample Position

NOTE

MHO900 series supports the CAN decode commands (std.) and the CAN-FD decode commands (opt.). The CAN-FD decode commands are only available when the instrument has installed the MHO900-AUTOA option.

3.4.14.1 :BUS<n>:CAN:SOURce

Syntax

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

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

Description

Sets or queries the source of CAN decoding on the specific bus.

Parameter

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

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:BUS1:CAN:SOURce CHANnel2    /*Sets the source of CAN decoding on
Bus1 to CHANnel2.*/
:BUS1:CAN:SOURce?      /*The query returns CHAN2.*/
```

3.4.14.2 :BUS<n>:CAN:STYPe

Syntax

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

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

Description

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

Parameter

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

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 bus signal.
- **CANL:** indicates the actual CAN_L bus signal.

- **DIFFerential:** The CAN differential bus signals connected to an analog source 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.*/
```

3.4.14.3 :BUS<n>:CAN:BAUD

Syntax

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

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

Description

Sets or queries the baud rate of CAN decoding on the specified bus. 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

N/A

Return Format

The query returns an integer ranging from 10k to 5M.

Example

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

3.4.14.4 :BUS<n>:CAN:FDBaud (Option)

Syntax

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

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

Description

Sets or queries the CAN-FD baud rate of CAN-FD decoding on the specified bus.

Parameter

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

Remarks

Only when the MHO900 series has installed the MHO900-AUTOA option, can these commands be supported.

Return Format

The query returns an integer ranging from 1M to 10M.

Example

```
:BUS1:CAN:FDBaud 120000          /*Sets the CAN-FD baud rate to
120000bps.*/
:BUS1:CAN:FDBaud?                /*The query returns 120000.*/
```

3.4.14.5 :BUS<n>:CAN:SPOint

Syntax

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

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

Description

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

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<spoint>	Integer	10 to 90	80

Remarks

For detailed information about sample point position, refer to *Sample Position*.

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.*/
```

3.4.14.6 :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 on the specified bus (expressed in %).

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<spoint>	Integer	10 to 90	80

Remarks

For detailed information about sample point position, refer to *Sample Position*.

Only when the MHO900 series has installed the MHO900-AUTOA option, can these commands be supported.

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.*/
```

3.4.15 :BUS<n>:LIN

The :BUS<n>:LIN commands are used to set relevant parameters for LIN decoding.

The oscilloscope samples the LIN signal at the specified sample position (if the source is an analog channel, the oscilloscope will also judge each data point (logic "1" or logic "0") according to the preset threshold level). You need to specify the LIN signal protocol version.

3.4.15.1 :BUS<n>:LIN:PARity

Syntax

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

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

Description

Sets or queries whether the LIN decoding on the specified decoding bus includes the parity bit.

Parameter

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

Remarks

- **1|ON:** includes the parity bit.
- **0|OFF:** does not include the parity bit.

Return Format

The query returns 0 or 1.

Example

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

3.4.15.2 :BUS<n>:LIN:SOURce

Syntax

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

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

Description

Sets or queries the source of LIN decoding on the specific bus.

Parameter

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

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

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

3.4.15.3 :BUS<n>:LIN:STANdard

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.*/
```

3.4.15.4 :BUS<n>:LIN:BAUD

Syntax

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

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

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	19200 bps

Remarks

N/A

Return Format

The query returns an integer ranging from 2.4k to 20M.

Example

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

3.4.16 :BUS<n>:FLEXray (Option)

The :BUS<n>:FLEXray commands are used to set the relevant parameters for FLEXray decoding.

FlexRay is a type of differential serial bus configured with three consecutive segments (i.e. packet header, payload, and packet trailer). The oscilloscope samples the FlexRay signal at the specified sample position and judges each data point as logic "1" or logic "0" according to the preset threshold level. The FlexRay decoding is required to specify the signal type and baud rate.

3.4.16.1 :BUS<n>:FLEXray:BAUD

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}	-
<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.*/
```

3.4.16.2 :BUS<n>:FLEXray:SOURce

Syntax

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

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

Description

Sets or queries the source of FlexRay decoding.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.16.3 :BUS<n>:FLEXray:SPOint

Syntax

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

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

Description

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

Parameter

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

Remarks

For detailed information about sample point position, refer to *Sample Position*.

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 on Bus 1 to 70%.*/
:BUS1:FLEXray:SPOint?          /*The query returns 70.*/
```

3.4.16.4 :BUS<n>:FLEXray:STYPe

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}	-
<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.*/
```

3.4.16.5 :BUS<n>:FLEXray:CHANnel

Syntax

:BUS <*n*>**:FLEXray:CHANnel** <*ch*>

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

Description

Sets or queries the selected source channel of FlexRay decoding on the specified bus.

Parameter

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

Remarks

N/A

Return Format

The query returns A or B.

Example

```
:BUS1:FLEXray CHANnel B.          /*Sets the source channel of
FlexRay decoding to B.*/
:BUS1:FLEXray:CHANnel?                  /*The query returns B.*/
```

3.4.17 :BUS<n>:IIS (Option)

The :BUS<n>:IIS commands are used to set relevant parameters for I2S decoding.

The oscilloscope samples the I2S signal at the specified sample position (if the source is an analog channel, the oscilloscope will also judge each data point (logic "1" or logic "0") according to the preset threshold level). I2S decoding is required to specify the serial clock, channel signal, and the data's source channel. You need to set Alignment, WS Low, and other parameters.

3.4.17.1 :BUS<n>:IIS:SOURce:CLOCk

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.17.2 :BUS<n>:IIS:SOURce:DATA

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel3

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.17.3 :BUS<n>:IIS:SOURce:WSELect

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|	CHANnel2

Name	Type	Range	Default
		CHANnel1|CHANnel2|CHANnel3|CHANnel4}	

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.4.17.4 :BUS<n>:IIS:ALIGnment

Syntax

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

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

Description

Sets or queries the the alignment mode of I2S decoding on the specified bus.

Parameter

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

Remarks

- **IIS:** data transmission (MSB first) begins at the second edge of the WS transition.
- **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.*/
```

3.4.17.5 :BUS<n>:IIS:CLOCk:SLOPe

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.*/
```

3.4.17.6 :BUS<n>:IIS:RWIDth

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}	-

Name	Type	Range	Default
<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 word size of I2S
decoding to 5.*/
:BUS1:IIS:RWIDth?               /*The query returns 5.*/
```

3.4.17.7 :BUS<n>:IIS:RECewidth

Syntax

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

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

Description

Sets or queries the receive width of I2S decoding on the specified bus.

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.*/
```

3.4.17.8 :BUS<n>:IIS:WSLow

Syntax

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

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

Description

Sets or queries the audio polarity of I2S decoding on the specified bus.

Parameter

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

Remarks

- **LEFT:** indicates that the WS Low is Left.
- **RIGHt:** indicates that the WS Low is Right.

Return Format

The query returns LEFT or RIGH.

Example

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

3.4.17.9 :BUS<n>:IIS:ENDian

Syntax

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

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

Description

Sets or queries the endian of I2S decoding on the specified bus.

Parameter

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

Remarks

- **MSB|ON|1:** indicates Most Significant Bit transmission sequence, i.e. the highest bit of the data is transmitted first.

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

3.4.17.10 :BUS<n>:IIS:POLarity

Syntax

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

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

Description

Sets or queries the data polarity of I2S decoding on the specified bus.

Parameter

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

Remarks

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

Return Format

The query returns POS or NEG.

Example

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

3.4.18 :BUS<n>:M1553 (Option)

The :BUS<n>:M1553 commands are used to set relevant parameters for M1553 decoding.

The oscilloscope samples the 1553B signal at the specified sample position (if the source is an analog channel, the oscilloscope will also judge each data point (logic "1" or logic "0") according to the preset threshold level). For M1553 decoding, you need to specify the data source and threshold.

3.4.18.1 :BUS<n>:M1553:SOURce

Syntax

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

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

Description

Sets or queries the source of M1553 decoding on the specified bus.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.5 :BODeplot Commands (Option)

The **:BODeplot** commands are used to set the relevant parameters of the bode plot function.

Bode plot is a way of graphically displaying the frequency response of a system. The analysis on the system's gain and phase margins enables you to test the stability of the system.

With the built-in signal generator module, the series generates the sweep signal of a specified frequency range and outputs to the switching power supply circuit under test. Then, the oscilloscope draws a Bode plot displaying the variation of phase and gain with different frequencies.

NOTE

Only when the MHO900 series has installed the AFG option, can these :BODeplot commands be supported.

3.5.1 :BODeplot:ENABle

Syntax

:BODeplot:ENABle <*bool*>

:BODeplot:ENABle?

Description

Sets or queries the on/off status of the bode plot.

Parameter

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

Remarks

N/A

Return Format

The query returns 1 or 0.

Examples

```
:BODeplot:ENABle ON /*Enables the bode plot.*/
:BODeplot:ENABle? /*The query returns 1.*/
```

3.5.2 :BODeplot:RUNStop

Syntax

:BODeplot:RUNStop <*bool*>

:BODeplot:RUNStop?

Description

Sets or queries the run/stops status of the bode plot.

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

```
:BODeplot:RUNStop ON /*Starts the bode plot drawing.*/
:BODeplot:RUNStop? /*The query returns 1.*/
```

3.5.3 :BODeplot:SWEeptype

Syntax

:BODeplot:SWEeptype <*type*>

:BODeplot:SWEeptype?

Description

Sets or queries the sweep type of the bode plot.

Parameter

Name	Type	Range	Default
<type>	Discrete	{LOG|LINE}	LOG

Remarks

- **LOG:** logarithmic sweep, indicating that the frequency of the swept sine wave varies logarithmically with the time.
- **LINE:** linear sweep, indicating that the frequency of the swept sine wave varies linearly with the time.

Return Format

The query returns LOG or LINE.

Example

```
:BODeplot:SWEeptype LINE    /*Sets the sweep type of the bode plot
to Linear.*/
:BODeplot:SWEeptype?      /*The query returns LINE.*/
```

3.5.4 :BODeplot:REF:IN

Syntax

:BODeplot:REF:IN <*source*>

:BODeplot:REF:IN?

Description

Sets or queries the input source of the bode plot.

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

```
:BODeplot:REF:IN CHANnel1 /*Sets the input source of the bode plot
to CHANnel1.*/
:BODeplot:REF:IN? /*The query returns CHAN1.*/
```

3.5.5 :BODeplot:REF:OUT

Syntax

:BODeplot:REF:OUT <*source*>

:BODeplot:REF:OUT?

Description

Sets or queries the output source of the bode plot.

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

```
:BODeplot:REF:OUT CHANnel1 /*Sets the output source of the bode
plot to CHANnel1.*/
:BODeplot:REF:OUT? /*The query returns CHAN1.*/
```

3.5.6 :BODeplot:STARt

Syntax

:BODeplot:STARt <*freq*>

:BODeplot:STARt?

Description

Sets or queries the start frequency of the sweep signal in the bode plot. The default unit is Hz.

Parameter

Name	Type	Range	Default
<freq>	Real	10 Hz to 3 MHz	100 Hz

Remarks

The start frequency that you set shall be smaller than the stop frequency.

Start Frequency ≤ Stop Frequency/10

To set or query the stop frequency of the sweep signal, run the *:BODeplot:STOP* command.

Return Format

The query returns the start frequency value in scientific notation. The unit is Hz.

Example

```
:BODeplot:STARt 100       /*Sets the start frequency of the sweep
signal to 100 Hz.*/
:BODeplot:STARt?          /*The query returns 1.000000E+2.*/
```

3.5.7 :BODeplot:STOP

Syntax

:BODeplot:STOP <*freq*>

:BODeplot:STOP?

Description

Sets or queries the stop frequency of the sweep signal in the bode plot. The default unit is Hz.

Parameter

Name	Type	Range	Default
<freq>	Real	100 Hz to 30 MHz	1 MHz

Remarks

The stop frequency that you set shall be greater than the start frequency.

Stop Frequency ≥ 10 x Start Frequency

To set or query the start frequency of the sweep signal, run the *:BODeplot:STARt* command.

Return Format

The query returns the stop frequency in scientific notation. The unit is Hz.

Example

```
:BODeplot:STOP 500        /*Sets the stop frequency of the sweep
signal to 500 Hz.*/
:BODeplot:STOP?           /*The query returns 5.000000E+2.*/
```

3.5.8 :BODeplot:POINts

Syntax

:BODeplot:POINts <*num*>

:BODeplot:POINts?

Description

Sets or queries the number of sweep points per decade.

Parameter

Name	Type	Range	Default
<num>	Integer	10 to 100	10

Remarks

N/A

Return Format

The query returns the number of sweep points per decade in integer.

Example

```
:BODeplot:POINts 20      /*Sets the number of the sweep points per
decade to 20.*/
:BODeplot:POINts?         /*The query returns 20.*/
```

3.5.9 :BODeplot:VOLTage

Syntax

:BODeplot:VOLTage <*range*>,<*amp*>

:BODeplot:VOLTage? <*range*>

Description

Sets or queries the variable amplitude of the sweep signal in the specified frequency range in the bode plot. The default unit of voltage is V, and the default frequency unit is Hz.

Parameter

Name	Type	Range	Default
<amp>	Real	Refer to *Remarks*	200 mV
<range>	Discrete	{ALL|10|100|1K|10K|100K|1M|10M|25M|1000|10000|100000|1000000|10000000|25000000|1e1|1e2|1e3|1e4|1e5|1e6|1e7|2.5e7}	-

Remarks

The range of <amp> is 20 mV to 10 V.

- When <range> is set to ALL: sets a unified voltage amplitude for the sweep signal in all the frequency ranges, with the voltage amplitude invariable.
- When <range> is set to other values other than ALL: sets a voltage amplitude for the specified range, with the voltage amplitude variable.

Return Format

The query returns the voltage amplitude of the sweep signal for the specified output frequency range in scientific notation. The unit is V.

Example

```
:BODeplot:VOLTage 100,0.3  /*Sets the voltage amplitude of the
signal whose frequency range is greater than 100 Hz to 300 mV.*/
:BODeplot:VOLTage? 100     /*The query returns 3.000000E-1.*/
```

3.5.10 :BODeplot:GAINcurveENABle

Syntax

:BODeplot:GAINcurveENABle <*bool*>

:BODeplot:GAINcurveENABle?

Description

Sets or queries whether to enable the display of amplitude frequency curve.

Parameter

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

Remarks

Only when the operating status of the instrument is Stop, can you set whether to display the amplitude frequency curve.

Return Format

The query returns 1 or 0.

Example

```
:BODeplot:GAINcurveENABle ON /*Displays the amplitude frequency
curve.*/
:BODeplot:GAINcurveENABle? /*The query returns 1.*/
```

3.5.11 :BODeplot:PHASEcurveENABle

Syntax

:BODeplot:PHASEcurveENABle <*bool*>

:BODeplot:PHASEcurveENABle?

Description

Sets or queries whether to enable the display of phase frequency curve.

Parameter

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

Remarks

Only when the operating status of the instrument is Stop, can you set whether to display the phase frequency curve.

Return Format

The query returns 1 or 0.

Example

```
:BODeplot:PHASEcurveENABle ON /*Displays the phase frequency
curve.*/
:BODeplot:PHASEcurveENABle? /*The query returns 1.*/
```

3.6 :CHANnel<n> 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.

- Setting the bandwidth limit can reduce the noises in the displayed waveforms. For example, the signal under test is a pulse with high frequency oscillation. When the bandwidth limit is turned off, the high frequency components of the signal under test can pass the channel. When the bandwidth limit is turned on, the high frequency components found in the signal under test that are greater than the limit are attenuated.
- You can remove unwanted signals by setting the coupling mode. For example, the signal under test is a square waveform with DC offset. AC coupling mode can block the DC components.
- When you use an oscilloscope to make actual measurements, a small offset that arises from the temperature drift of the component or external environment disturbance may occur on the zero-cross voltage of the channel, which will affect the measurement results of the vertical parameters. This series oscilloscope allows you to set an offset calibration voltage for calibrating the zero point of the corresponding channel so as to improve the accuracy of the measurement results.
- When the fine adjustment is enabled, you can further adjust the vertical scale within a relatively smaller range to improve vertical resolution, making it easier to view waveform details.

3.6.1 :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

{OFF|ON|20M|250M}, with the unit Hz.

When set to OFF, the bandwidth limit is disabled.

Return Format

The query returns 20M, 250M, or OFF.

Example

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

3.6.2 :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

Remarks

- **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     /*Sets the coupling mode to AC.*/
:CHANnel1:COUPling?       /*The query returns AC.*/
```

3.6.3 :CHANnel<n>:DISPlay

Syntax

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

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

Description

Enables or disables 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}}	1|ON

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

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

3.6.4 :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.*/
```

3.6.5 :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	Refer to *Remarks*	0 V

Remarks

The range of the vertical offset of the specified channel is related to the its current vertical scaleand input impedance.

- When the input impedance is 1 MΩ:
  - ±1 V (≥1 mV/div, ≤65 mV/div)
  - ±10 V (>65 mV/div, ≤274 mV/div)
  - ±20 V (>274 mV/div, ≤2.79 V/div)
  - ±100 V (>2.79 V/div, ≤10 V/div)
- When the input impedance is 50 Ω:
  - ±1 V (≥1 mV/div, ≤136 mV/div)
  - ± 4 V (>136 mV/div)

You can send the *:CHANnel<n>:SCALe* command to set or query the vertical scale of the specified channel.

You can send the *:CHANnel<n>:IMPedance* command to set or query the input impedance of the specified analog channel.

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 1.000000E-02.*/
```

3.6.6 :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}	-

Name	Type	Range	Default
<val>	Real	-100 ns to 100 ns	0 s

Remarks

When the horizontal time base is greater than 10 μs, the parameter <val> cannot be set.

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.*/
```

3.6.7 :CHANnel<n>:IMPedance

Syntax

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

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

Description

Sets or queries the input impedance of the specified analog channel.

Parameter

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

Remarks

- **OMEG:** 1 MΩ. 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. The oscilloscope shall match a device whose output impedance is 50 Ω.

Return Format

The query returns OMEG or FIFT.

Example

```
:CHANnel1:IMPedance OMEG   /*Sets the input impedance of CH1 to 1
MΩ.*/
:CHANnel1:IMPedance?       /*The query returns OMEG.*/
```

3.6.8 :CHANnel<n>:SCALe

Syntax

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

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

Description

Sets or queries the vertical scale of the specified channel. Its default unit is V/div.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<scale>	Real	Refer to *Remarks*	50 mV

Remarks

The range of the vertical scale of the channel is related to input impedance, probe ratio.

- When the input impedance is 1 MΩ and the probe ratio is 1X, the range of <scale> is from 1 mV/div to 10 V/div.
- When the input impedance is 50 Ω and the probe ratio is 1X, the range of <scale> is from 200 μV/div to 1 V/div.

You can send the *:CHANnel<n>:PROBe* command to set or query the probe ratio of the specified analog channel.

Return Format

The query returns the vertical scale in scientific notation. The unit is V/div.

Example

```
:CHANnel1:SCALe 0.1/*Sets the vertical scale of CH1 to 0.1 V/div.*/
:CHANnel1:SCALe?   /*The query returns 1.000000E-01.*/
```

3.6.9 :CHANnel<n>:PROBe

Syntax

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

:CHANnel<*n*>:PROBe?

Description

Sets or queries the probe ratio of the specified analog 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|15|20|50|100|150|200|500|1000|1500|2000|5000|10000|15000|20000|50000}	1

Remarks

- The range of the parameter <atten> is related to the channel amplitude unit. To set or query the amplitude unit of the specified channel, run the *:CHANnel<n>:UNITs* command.

  When the amplitude unit is set to WATT, VOLTage, or UNKNown, the range of <atten> is from 0.001 to 50,000.

  When the amplitude unit is set to AMPere, the range of <atten> is from 0.001 to 10, with the unit V/A.
- Display amplitude of the signal under test = Actual amplitude of the signal under test x Probe ratio(Probe ratio does not affect the actual amplitude of the signal)
- The set probe ratio affects the value and settable range of the current vertical scale. To query the vertical scale, run the *:CHANnel<n>:SCALe* command.

Return Format

The query returns the probe ratio of the specified analog channel in scientific notation.

Example

```
:CHANnel1:PROBe 10 /*Sets the probe ratio of CH1 to 10X.*/
:CHANnel1:PROBe? /*The query returns 1.000000E+1.*/
```

3.6.10 :CHANnel<n>:LABel:SHOW

Syntax

:CHANnel<*n*>**:LABel:SHOW** <*bool*>

:CHANnel<*n*>**:LABel:SHOW?**

Description

Sets or queries whether to display the label of the specified channel.

Parameter

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

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:CHANnel1:LABel:SHOW ON          /*Displays the label of the
specified channel.*/
:CHANnel1:LABel:SHOW?            /*The query returns 1.*/
```

3.6.11 :CHANnel<n>:LABel:CONTent

Syntax

:CHANnel<*n*>**:LABel:CONTent** <*str*>

:CHANnel<*n*>**:LABel:CONTent?**

Description

Sets or queries the label of the specified channel.

Parameter

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

Example

```
:CHANnel1:LABel:CONTent ch1        /*Sets the label of Channel 1 to
ch1.*/
:CHANnel1:LABel:CONTent? /*The query returns ch1.*/
```

3.6.12 :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.*/
```

3.6.13 :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 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

N/A

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.*/
```

3.6.14 :CHANnel<n>:POSition

Syntax

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

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

Description

Sets or queries the bias voltage of the specified channel. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<offset>	Real	Refer to *Remarks*	0 V

Remarks

The range of bias voltage is related to the the vertical scale and input impedance.

- When the input impedance is 1 MΩ:
  - ±1 V (≥1 mV/div, ≤65 mV/div)

- ±10 V (>65 mV/div, ≤274 mV/div)
- ±20 V (>274 mV/div, ≤2.79 V/div)
- ±100 V (>2.79 V/div, ≤10 V/div)

• When the input impedance is 50 Ω:

- ±1 V (≥1 mV/div, ≤136 mV/div)
- ± 4 V (>136 mV/div)

You can send the *:CHANnel<n>:SCALe* command to set or query the vertical scale of the specified channel.

You can send the *:CHANnel<n>:IMPedance* command to set or query the input impedance of the specified analog channel.

Return Format

The query returns the bias voltage of the specified channel in scientific notation. The unit is V.

Example

```
:CHANnel1:POSition 10    /*Sets the bias voltage of CH1 to 10 V.*/
:CHANnel1:POSition?    /*The query returns 1.000000E+01.*/
```

3.7 :COUNter Commands

:COUNter commands are used to set or query the measurement and statistic parameters for the frequency counter.

The frequency counter analysis function provides frequency, period, or edge event counter measurements on any analog channel.

3.7.1 :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
```

3.7.2 :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.*/
```

3.7.3 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4.

Example

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

3.7.4 :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

- **FREQuency:** indicates the Frequency measurement.
- **PERiod:** indicates the Period measurement.
- **TOTalize:** indicates the Totalize measurement.

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.*/
```

3.7.5 :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 6	4

Remarks

When the measurement mode of the frequency counter is Period or Frequency, you need to set resolution. When the mode is Totalize, resolution is available to set. You can run the *:COUNter:MODE* command to set or query the measurement mode of the frequency counter.

Return Format

The query returns an integer ranging from 3 to 6.

Example

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

3.7.6 :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 only available for "Period" and "Frequency", but it is unavailable for "Totalize". You can use *:COUNter:MODE* to query or set the measurement mode.

Return Format

The query returns 1 or 0.

Examples

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

3.7.7 :COUNter:TOTalize:CLEar

Syntax

```
:COUNter:TOTalize:CLEar
```

Description

Clears the total count.

Parameter

N/A

Remarks

This command is available when the measurement type is set to "Totalize".

Return Format

N/A

Example

```
N/A
```

3.8 :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.

Before making cursor measurements, connect the signal to the oscilloscope to acquire stable display. The cursor measurement function provides the following two cursors.

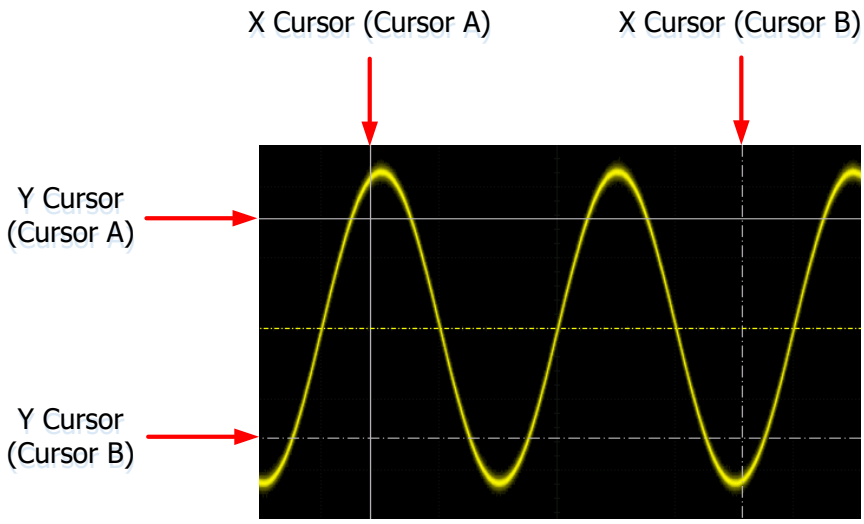


Figure 3.6 Cursors

- **X Cursor**

  X cursor is a vertical solid/dotted line that is used to make horizontal adjustments. It can be used to measure time (s) and frequency (Hz).

  - Cursor A is a vertical solid line and Cursor B is a vertical dotted line.
  - In the XY cursor mode, cursor X is used to measure the waveform amplitude of CH1.

- **Y Cursor**

  Y cursor is a horizontal solid/dotted line that is used to make vertical adjustments. It can be used to measure amplitude (the unit is the same as that of the source channel amplitude).

  - Cursor A is a horizontal solid line and Cursor B is a horizontal dotted line.
  - In XY cursor mode, cursor Y is used to measure the waveform amplitude of CH2.

Cursor Measurement Results

- AX: indicates the X value at Cursor A.
- AY: indicates the Y value at Cursor A.
- BX: indicates the X value at Cursor B.

- BY: indicates the Y value at Cursor B.
- ΔX: indicates the horizontal spacing between Cursor A and Cursor B.
- ΔY: indicates the vertical spacing between Cursor A and Cursor B.
- 1/ΔX: indicates the reciprocal of the horizontal spacing between Cursor A and Cursor B.

Cursor Mode

- **Manual Mode**

  In the manual cursor mode, you can adjust the cursor manually to measure the value of the waveforms of the specified source at the current cursor. If the settings for the parameter such as the cursor type and measurement source are different, the measurement results will be different for cursor measurement.

- **Track Mode**

  In the Track mode, you can adjust the two pairs of cursors (Cursor A and Cursor B) to measure the X and Y values on two different sources respectively. When the cursors are moved horizontally/vertically, the markers will position on the waveform automatically. When the waveform is expanded or compressed horizontally/vertically, the markers will track the points being marked at the last adjustment of the cursors.

- **XY Mode**

  By default, XY mode is unavailable. It is available only when the horizontal time base mode is "XY".

3.8.1 :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:** the manual mode of cursor measurement.
- **TRACk:** the track mode of cursor measurement.
- **XY:** the XY mode of cursor measurement. It is only valid when you select "XY" mode. You can use *:TIMebase:MODE* to query or set the mode.

For functions of different cursor measurement modes, refer to *Cursor 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.*/
```

3.8.2 :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.*/
```

3.8.3 :CURSor:MANual

3.8.3.1 :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

Remarks

- **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.*/
```

3.8.3.2 :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

When the channel source is NONE, the manual mode of cursor is disabled.

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.*/
```

3.8.3.3 :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.*/
```

3.8.3.4 :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}	SOURce

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.*/
```

3.8.3.5 :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 in 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.*/
```

3.8.3.6 :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.

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. The unit is V.

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.*/
```

3.8.3.7 :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.

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. The unit is s.

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.*/
```

3.8.3.8 :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.

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.The unit is V.

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.*/
```

3.8.3.9 :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
```

3.8.3.10 :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
```

3.8.3.11 :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
```

3.8.3.12 :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
```

3.8.3.13 :CURSor:MANual:XDELta?

Syntax

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

Description

Queries the difference (ΔX) 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
```

3.8.3.14 :CURSor:MANual:IXDelta?

Syntax

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

Description

Queries the reciprocal (1/ΔX) 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/ΔX in scientific notation.

Example

```
N/A
```

3.8.3.15 :CURSor:MANual:YDELta?

Syntax

:CURSor:MANual:YDELta?

Description

Queries the difference (ΔY) 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
```

3.8.4 :CURSor:TRACk

3.8.4.1 :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.*/
```

3.8.4.2 :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.*/
```

3.8.4.3 :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 in 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.*/
```

3.8.4.4 :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. The unit is s.

Example

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

3.8.4.5 :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. The unit is V.

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.*/
```

3.8.4.6 :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.The unit is V.

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.*/
```

3.8.4.7 :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
```

3.8.4.8 :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
```

3.8.4.9 :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
```

3.8.4.10 :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
```

3.8.4.11 :CURSor:TRACk:XDELta?

Syntax

```
:CURSor:TRACk:XDELta?
```

Description

Queries the difference (ΔX) 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
```

3.8.4.12 :CURSor:TRACk:YDELta?

Syntax

:CURSor:TRACk:YDELta?

Description

Queries the difference (ΔY) 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
```

3.8.4.13 :CURSor:TRACk:IXDelta?

Syntax

:CURSor:TRACk:IXDelta?

Description

Queries the reciprocal (1/ΔX) 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/ΔX in scientific notation.

Example

```
N/A
```

3.8.4.14 :CURSor:TRACk:MODE

Syntax

:CURSor:TRACk:MODE <*mode*>

:CURSor:TRACk:MODE?

Description

Sets or queries the axis in the track mode of cursor measurement.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{Y|X}	-

Remarks

N/A

Return Format

The query returns Y or X.

Example

```
:CURSor:TRACk:MODE X    /*Sets the axis in the track mode of
cursor measurement to X-axis.*/
:CURSor:TRACk:MODE?       /*The query returns X.*/
```

3.8.5 :CURSor:XY

The **:CURSor:XY** commands are only available when the horizontal time base mode is set to XY.

3.8.5.1 :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.

Parameter

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

Remarks

Related to the current vertical scale and vertical offset.

Return Format

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

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.*/
```

3.8.5.2 :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.

Parameter

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

Remarks

Related to the current vertical scale and vertical offset.

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.*/
```

3.8.5.3 :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.

Parameter

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

Remarks

Related to the current vertical scale and vertical offset.

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.*/
```

3.8.5.4 :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.

Parameter

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

Remarks

Related to the current vertical scale and vertical offset.

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.*/
```

3.8.5.5 :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
```

3.8.5.6 :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
```

3.8.5.7 :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
```

3.8.5.8 :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
```

3.8.5.9 :CURSor:XY:XDELta?

Syntax

```
:CURSor:XY:XDELta?
```

Description

Queries the difference (ΔX) between the X value at Cursor A and the X value at Cursor B in the XY cursor measurement.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the current difference in scientific notation.

Example

```
N/A
```

3.8.5.10 :CURSor:XY:YDELta?

Syntax

```
:CURSor:XY:YDELta?
```

Description

Queries the difference (ΔY) between the Y value at Cursor A and the Y value at Cursor B in the XY 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
```

3.9 :DISPlay Commands

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

3.9.1 :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.
- You can also send the *:CLEar* command to clear all the waveforms on the screen.
- This command functions the same as the front-panel key 

.

Return Format

N/A

Example

```
N/A
```

3.9.2 :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.*/
```

3.9.3 :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 glitches that change relatively slowly or glitches 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.*/
```

3.9.4 :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.*/
```

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

3.9.6 :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.*/
```

3.9.7 :DISPlay:CBRightness

Syntax

:DISPlay:CBRightness <*brightness*>

:DISPlay:CBRightness?

Description

Sets or queries the brightness of the cursor, expressed in percentage.

Parameter

Name	Type	Range	Default
<brightness>	Integer	0 to 100	80

Remarks

N/A

Return Format

The query returns an integer ranging from 0 to 100.

Example

```
:DISPlay:CBRightness 60     /*Sets the cursor brightness to 60%.*/
:DISPlay:CBRightness?       /*The query returns 60.*/
```

3.9.8 :DISPlay:DATA?

Syntax

:DISPlay:DATA? [<*type*>]

Description

Queries the bitmap data stream of the currently displayed image.

Parameter

Name	Type	Range	Default
<type>	Discrete	{BMP|PNG|JPG}	BMP

Remarks

The read data format is TMC header + binary data stream of the screenshot + terminator. The TMC header is in #NXXXXXX format; wherein, # is the TMC header identifier; N following # represents the number of digits (in the decimal integer) that follow; the length of the binary data stream of the screenshot is expressed in ASCII strings, and the terminator represents the ending of communication. For example, the data read for one time is #9000387356. 9 indicates the number of digits (in the decimal integer) that follow, and "000387356" indicates the length of the binary data stream, that is, the number of bytes to be transmitted.

Return Format

The query returns the binary data stream of the screenshot in a specified format.

Example

```
N/A
```

3.9.9 :DISPlay:RULers

Syntax

:DISPlay:RULers <*bool*>

:DISPlay:RULers?

Description

Enables or disables the display of the scale ruler; or queries the on/off status of the scale 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 scale ruler.*/
:DISPlay:RULers?       /*The query returns 1.*/
```

3.9.10 :DISPlay:MOVE

Syntax

:DISPlay:MOVE <*bool*>

:DISPlay:MOVE?

Description

Sets or queries whether to move the scale ruler to track the waveform coordinate.

Parameter

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

Remarks

1|ON: enables to move the scale ruler. When you move the waveform, the scale ruler moves along with its coordinate to keep track of its coordinate to be fixed.

0|OFF: disables to move the scale ruler. When you move the waveform, the scale ruler does not move along with the changing coordinate, and its coordinate changes constantly.

Return Format

The query returns 1 or 0.

Example

```
:DISPlay:MOVE 1    /*Sets to move the scale ruler to track the
waveform coordinate.*/
:DISPlay:MOVE?       /*The query returns 1.*/
```

3.9.11 :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

When it is enabled, different colors are displayed on the screen to indicate the times of data acquisition or acquisition probability.

Return Format

The query returns 1 or 0.

Examples

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

3.9.12 :DISPlay:WHOLd

Syntax

:DISPlay:WHOLd <*bool*>

:DISPlay:WHOLd?

Description

Sets to enable or disable the waveform freezing function; queries whether to enable or disable the waveform freezing 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

```
:DISPlay:WHOLd ON    /*Enables the Waveform Freeze.*/
:DISPlay:WHOLd?       /*The query returns 1.*/
```

3.10 :DVM Commands

:DVM commands are used to set or query the DVM parameters.

The built-in DVM of this oscilloscope provides 4-digit voltage measurements on any analog channel. DVM measurements are asynchronous from the oscilloscope's acquisition system and are always acquiring.

3.10.1 :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
```

3.10.2 :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.*/
```

3.10.3 :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, CHAN4.

Example

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

3.10.4 :DVM:MODE

Syntax

:DVM:MODE <*mode*>

:DVM:MODE?

Description

Sets or queries the mode of 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.*/
```

3.11 Histogram Commands

This series oscilloscope supports the histogram analysis function, which provides you a statistical view of the waveforms or measurement results, enabling you to judge the trend of waveforms, and quickly locate the potential problems of the signal.

Histogram Analysis Result

The following parameters are also included in the statistical results.

- Sum: indicates the sum of all bins (buckets) in the histogram.
- Peaks: indicates the maximum number of hits in any single bin.
- Max: indicates the maximum value of the statistical result.
- Min: indicates the minimum value of the statistical result.
- Pk_Pk: indicates the Delta (Max-Min) between the max. value and the min. value.
- Mean: indicates the average value of the histogram.
- Median: indicates the median value of the histogram.
- Mode: indicates the mode value of the histogram.
- Bin width: indicates the width of each bin (bucket) in the histogram.
- Sigma: indicates the standard deviation of the histogram.
- μ±σ: indicates the proportion of the number of frequencies or counts of the histogram hits that lie within one standard deviation of the mean to the total number of histogram hits. μ indicates the mean value in normal distribution. σ indicates the standard deviation in the normal distribution.
- μ±2σ: indicates the proportion of the number of frequencies or counts of the histogram hits that lie within two standard deviations of the mean to the total number of histogram hits. μ indicates the mean value in normal distribution. σ indicates the standard deviation in the normal distribution.
- μ±3σ: indicates the proportion of the number of frequencies or counts of the histogram hits that lie within three standard deviations of the mean to the total number of histogram hits. μ indicates the mean value in normal distribution. σ indicates the standard deviation in the normal distribution.

3.11.1 :HISTogram:ENABle

Syntax

`:HISTogram:ENABle` <*bool*>

`:HISTogram:ENABle?`

Description

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

Parameter

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

Remarks

N/A

Return Format

The query returns 1 or 0.

Examples

```
:HISTogram:ENABle ON /*Enables the histogram function.*/
:HISTogram:ENABle? /*The query returns 1.*/
```

3.11.2 :HISTogram:TYPE

Syntax

:HISTogram:TYPE <*type*>

:HISTogram:TYPE?

Description

Sets or queries the type of the histogram.

Parameter

Name	Type	Range	Default
<type>	Discrete	{HORizontal|VERTical}	HORizontal

Remarks

- **HORizontal:** horizontal histogram.
- **VERTical:** vertical histogram.

Return Format

The query returns HOR or VERT.

Example

```
:HISTogram:TYPE VERTical    /*Sets the histogram type to Vertical.*/
:HISTogram:TYPE?      /*The query returns VERT.*/
```

3.11.3 :HISTogram:SOURce

Syntax

:HISTogram:SOURce <*source*>

:HISTogram:SOURce?

Description

Sets or queries the source of the histogram.

Parameter

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

Remarks

N/A

Return Format

The query returnsCHAN1, CHAN2, CHAN3, CHAN4.

Example

```
:HISTogram:SOURce CHANnel2      /*Sets the source of the histogram
to CH2.*/
:HISTogram:SOURce?              /*The query returns CHAN2.*/
```

3.11.4 :HISTogram:HEIGht

Syntax

:HISTogram:HEIGht <*height*>

:HISTogram:HEIGht?

Description

Sets or queries the height of the histogram.

Parameter

Name	Type	Range	Default
<height>	Integer	1div to 4div	2div

Remarks

N/A

Return Format

The query returns an integer ranging from 1 to 4.

Example

```
:HISTogram:HEIGht 2    /*Sets the histogram height to 2.*/
:HISTogram:HEIGht?      /*The query returns 2.*/
```

3.11.5 :HISTogram:RANGe:LEFT

Syntax

:HISTogram:RANGe:LEFT <*number*>

:HISTogram:RANGe:LEFT?

Description

Sets or queries the left limit of the histogram.

Parameter

Name	Type	Range	Default
<number>	Real	(-5 x Horizontal Time Base + Horizontal Offset) to (5 x Horizontal Time Base + Horizontal Offset)	-

Remarks

- The left limit should be smaller than the right limit. You can use *:HISTogram:RANGe:RIGHt* to set or query the right limit of the histogram.
- You can use *:TIMebase[:MAIN]:SCALe* to set or query the horizontal time base.
- You can use *:TIMebase[:MAIN][:OFFSet]* to set or query the horizontal offset.

Return Format

The query returns the left limit in scientific notation.

Example

```
:HISTogram:RANGe:LEFT -2    /*Sets the left limit of the histogram
to -2 s.*/
:HISTogram:RANGe:LEFT?      /*The query returns -2.000000E0.*/
```

3.11.6 :HISTogram:RANGe:RIGHt

Syntax

:HISTogram:RANGe:RIGHt <*number*>

:HISTogram:RANGe:RIGHt?

Description

Sets or queries the right limit of the histogram.

Parameter

Name	Type	Range	Default
<number>	Real	(-5 x Horizontal Time Base + Horizontal Offset) to (5 x Horizontal Time Base + Horizontal Offset)	-

Remarks

- The right limit should be greater than the left limit. You can use *:HISTogram:RANGe:LEFT* to set or query the left limit of the histogram.
- You can use *:TIMebase[:MAIN]:SCALe* to set or query the horizontal time base.
- You can use *:TIMebase[:MAIN][:OFFSet]* to set or query the horizontal offset.

Return Format

The query returns the right limit in scientific notation.

Example

```
:HISTogram:RANGe:RIGHt 2    /*Sets the right limit of the histogram
to 2 s.*/
:HISTogram:RANGe:RIGHt?      /*The query returns 2.000000E0.*/
```

3.11.7 :HISTogram:RANGe:TOP

Syntax

:HISTogram:RANGe:TOP <*number*>

:HISTogram:RANGe:TOP?

Description

Sets or queries the top limit of the histogram.

Parameter

Name	Type	Range	Default
<number>	Real	(-4 x VerticalScale - OFFSet) to (4 x VerticalScale - OFFSet)	-

Remarks

- The top limit should be greater than the bottom limit. You can use *:HISTogram:RANGe:BOTTom* to set or query the bottom limit of the histogram.
- You can use *:CHANnel<n>:SCALe* to set or query the vertical scale for the specified channel.
- You can use *:CHANnel<n>:OFFSet* to set or query the vertical offset for the specified channel.

Return Format

The query returns the top limit in scientific notation.

Example

```
:HISTogram:RANGe:TOP -2     /*Sets the top limit of the histogram to
-2 V.*/
:HISTogram:RANGe:TOP?       /*The query returns -2.000000E0.*/
```

3.11.8 :HISTogram:RANGe:BOTTom

Syntax

:HISTogram:RANGe:BOTTom <*number*>

:HISTogram:RANGe:BOTTom?

Description

Sets or queries the bottom limit of the histogram.

Parameter

Name	Type	Range	Default
<number>	Real	(-4 x VerticalScale - OFFSet) to (4 x VerticalScale - OFFSet)	-

Remarks

- The bottom limit should be smaller than the top limit. You can use *:HISTogram:RANGe:TOP* to set or query the top limit of the histogram.
- You can use *:CHANnel<n>:SCALe* to set or query the vertical scale for the specified channel.

- You can use *:CHANnel<n>:OFFSet* to set or query the vertical offset for the specified channel.

Return Format

The query returns the bottom limit in scientific notation.

Example

```
:HISTogram:RANGe:BOTTom -2     /*Sets the bottom limit of the
histogram to -2 V.*/
:HISTogram:RANGe:BOTTom?       /*The query returns -2.000000E0.*/
```

3.11.9 :HISTogram:STATistics:RESult?

Syntax

`:HISTogram:STATistics:RESult?`

Description

Queries the statistics of the histogram results.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the statistics of the histogram results in strings.

```
[Sum:5.6khits,Peaks:14hits,Max:3.9us,Min:-4us,Pk_Pk:7.98us,Mean:-20n
s,Median:-20ns,Mode:-4us,Bin
width:20ns,Siqma:2.303us,meanPlusSigma:577.5,meanPlus2Sigma:1,meanPl
us3Sigma:1]
```

For details about the returned results, refer to descriptions in *Histogram Commands*.

Example

```
N/A
```

3.11.10 :HISTogram:RESet

Syntax

`:HISTogram:RESet?`

Description

Resets the statistics data of histogram.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.11.11 :HISTogram:SAVE:CSV

Syntax

:HISTogram:SAVE:CSV <*path*>

Description

Saves the histogram data file to the specified path.

Parameter

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

This saving command is only valid when valid histogram data statistics exists.

<path> includes the file storage location and the filename with a suffix.

The local path is C:/; and the path of the external storage device is D:/.

If the specified storage location already contains a file with the same filename, you can set whether to overwrite the existing file by enabling or disabling the overwriting function.

The filename can contain letters, numbers, and other non-Chinese characters. The length of the filename shall not exceed26characters.

Return Format

N/A

Example

```
:HISTogram:SAVE:CSV C:/123.csv   /*Saves the current histogram data
file to the local path Disk C, with the filename 123.csv.*/
```

3.12 IEEE488.2 Common Commands

The IEEE488.2 common commands are used to query the basic information of the instrument or executing basic operations. These commands usually start with "*", and the command keywords contain 3 characters and are related with status registers.

The standard event status register (SESR) and status byte register (SBR) record the event of a certain type happened during the use of the instrument. IEEE488.2 defines to record one specific type of event for each bit in the status register.

Table 3.153 Table of the Bit Definition of Standard Event Status Register

Bit No.	Bit Name	Decimal Value	Description
0	Operation Complete (OPC)	1	"Operation complete" indicates that all pending operations were completed following the execution of the command.
1	Not Used	2	-
2	Query Error (QYE)	4	The instrument tries to read the output buffer but it was empty. Or, a new command line was received before a previous query has been read. Or, both the input buffer and output buffer are full.
3	Device-Specific Error (DDE)	8	Indicates that an error has occurred that is neither a Command Error, a Query Error, nor an Execution Error. A Device-Specific Error is any executed device operation that did not properly complete due to some condition, such as self-check error, calibration error, or other device-specific errors.
4	Execution Error (E)	16	An execution error occurred.
5	Command Error (CME)	32	A command error (command syntax error) has occurred.
6	Not Used	64	-
7	Power On (PON)	128	Indicates that an off-to-on transition has occurred in the device's power supply since

Bit No.	Bit Name	Decimal Value	Description
			last reading or the event register was cleared.

Table 3.154 Table of the Bit Definition of Status Byte Register

Bit No.	Bit Name	Decimal Value	Description
0	Not Used	1	-
1	Not Used	2	-
2	Error Queue	4	1 or multiple errors in the error queue
3	Questionable Data Summary	8	Sets 1 or multiple bits (must be the enabled bit) in the questionable data register.
4	Message Available (MAV)	16	Indicates the available data in the output buffer.
5	Standard Event Summary	32	Sets 1 or multiple bits (must be the enabled bit) in the standard event register.
6	Master Summary Status (MSS)	64	Sets 1 or multiple bits (must be the enabled bit) in the Status Byte Register and generate the service request.
7	Operation Status Register	128	Sets 1 or multiple bits (must be the enabled bit) in the Operation Status Register.

3.12.1 *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
```

3.12.2 *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
```

3.12.3 *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
```

3.12.4 *ESE

Syntax

***ESE** <*maskargument*>

***ESE?**

Description

Sets or queries the enable register of the standard event register set.

Parameter

Name	Type	Range	Default
<maskargument>	Integer	0 to 255	0

Remarks

For the definitions of the bits in the standard event register, refer to *Table 3.153 Table of the Bit Definition of Standard Event Status Register* . The value of <maskargument> is the sum of the decimal values of all bits set in the standard event register. For example, to enable Bit 2 (4 in decimal), Bit 3 (8 in decimal), and Bit 7 (128 in decimal), set the <maskargument> to 140 (4+8+128).

Return Format

The query returns an integer. The integer equals to the decimal-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.*/
```

3.12.5 *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 (*Table 3.153 Table of the Bit Definition of 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
```

3.12.6 *OPC

Syntax

***OPC**

***OPC?**

Description

The *OPC command sets bit 0 (Operation Complete, OPC) in the standard event register to 1 after the current operation is finished.

The *OPC? command queries whether the current operation is finished.

Parameter

N/A

Remarks

For the definitions of the bits in the standard event register, refer to *Table 3.153 Table of the Bit Definition of Standard Event Status Register*.

Return Format

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

Example

```
N/A
```

3.12.7 *RCL

Syntax

***RCL**

Description

Recalls instrument settings from the specified non-volatile memory. The previous saved settings through the **SAV* command will be overwritten.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.12.8 *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.*/
```

3.12.9 *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

For the definitions of the bits in the status byte register, refer to *Table 3.154 Table of the Bit Definition of Status Byte Register*. The value of <maskargument> is the sum of the decimal values of all bits set in the status byte register. For example, to enable Bit 2 (4 in decimal), Bit 3 (8 in decimal), and Bit 7 (128 in decimal), set the <maskargument> to 140 (4+8+128).

Return Format

The query returns an integer. The integer equals to the decimal-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.*/
```

3.12.10 *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 (*Table 3.154 Table of the Bit Definition of 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 XXXXXX00 (X is 1 or 0).

Return Format

The query returns an integer. The integer equals to the decimal-weighted sum of all the bits set in the register.

Example

```
N/A
```

3.12.11 *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
```

3.12.12 *TST?

Syntax

`*TST?`

Description

Performs a self-test and returns the self-test result.

Parameter

N/A

Remarks

This command executes a self-test. If the test fails, one or more error messages will be displayed, providing more information. You can use *:SYSTem:ERRor[:NEXT]?* to read the error queue.

Return Format

The query returns 0 or 1.

- **0:** it passes.
- **1:** one or more tests fail.

Example

```
N/A
```

3.13 :LA Commands

The **:LA** commands are used to perform relevant operations on the digital channels.

The oscilloscope compares the voltages acquired in each sample with the preset logic threshold. If the voltage of the sample point is greater than the threshold, it will be stored as logic 1; otherwise, it will be stored as logic 0. The oscilloscope displays logic levels ("1" and "0") in the form of a graph for you to easily detect and analyze the errors in circuit design (hardware design and software design).

3.13.1 :LA:ENABle

Syntax

:LA:ENABle <*bool*>

:LA:ENABle?

Description

Sets to enable or disable the LA function; queries whether the LA function is enabled.

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

```
:LA:ENABle ON     /*Enables the LA function.*/
:LA:ENABle?       /*The query returns 1.*/
```

3.13.2 :LA:ACTive

Syntax

:LA:ACTive <*digital*>

:LA:ACTive?

Description

Sets or queries the current activate channel.

Parameter

Name	Type	Range	Default
<digital>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15}	D0

Remarks

- The parameter <digital> can be any of the digital channels (D0-D15). The channel label and waveform of the selected channel are displayed in red.
- Only the currently enabled digital channel can be set to the activated channel. To open the specified channel, run the *:LA:DIGital:ENABle* command. To enable the desired channel group, run the *:LA:POD<n>:DISPlay* command.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, or D15.

Example

```
:LA:ACTive D3     /*Sets the current activated channel to D3.*/
:LA:ACTive?       /*The query returns D3.*/
```

3.13.3 :LA:AUTosort

Syntax

:LA:AUTosort <*val*>

:LA:AUTosort?

Description

Sets the auto sorting mode for the waveforms of the enabled channels.

Parameter

Name	Type	Range	Default
<val>	Discrete	{D0D15|D15D0}	D15D0

Remarks

- **D0D15:** the waveforms on the display are D0-D15 in sequence from top to bottom.
- **D15D0:** the waveforms on the display are D15-D0 in sequence from top to bottom.

Return Format

The query returns D0D15 or D15D0.

Example

```
:LA:AUTosort D0D15 /*Sets the auto sorting mode to D0D15.*/
:LA:AUTosort? /*The query returns D0D15.*/
```

3.13.4 :LA:DIGital:ENABle

Syntax

:LA:DIGital:ENABle <*digital*>,<*bool*>

:LA:DIGital:ENABle? <*digital*>

Description

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

Parameter

Name	Type	Range	Default
<digital>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15}	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

The currently enabled channel can be selected as the active channel by sending the *:LA:ACTive* command.

Return Format

The query returns 1 or 0.

Example

```
:LA:DIGital:ENABle D3, ON          /*Enables D3.*/
:LA:DIGital:ENABle? D3            /*The query returns 1.*/
```

3.13.5 :LA:DIGital:LABel

Syntax

:LA:DIGital:LABel <*digital*>,<*label*>

:LA:DIGital:LABel? <*digital*>

Description

Sets or queries the label of the specified digital channel.

Parameter

Name	Type	Range	Default
<digital>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7| D8|D9|D10|D11|D12|D13|D14|D15}	-
<label>	ASCII String	It can contain English letters and numbers, and also some symbols.	-

Remarks

N/A

Return Format

The query returns the label of the specified digital channel in ASCII strings.

Example

```
:LA:DIGital:LABel D0,ACK     /*Sets the label of D0 to ACK.*/
:LA:DIGital:LABel? D0       /*The query returns ACK.*/
```

3.13.6 :LA:POD<n>:DISPlay

Syntax

:LA:POD<*n*>**:DISPlay** <*bool*>

:LA:POD<*n*>**:DISPlay?**

Description

Enables or disables the specified default channel group, or queries the on/off status of the specified default channel group.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 2	-
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

2 default channel groups: POD1 (D0 to D7) or POD2 (D8 to D15).

Return Format

The query returns 1 or 0.

Example

```
:LA:POD1:DISPlay ON          /*Enables POD1.*/
:LA:POD1:DISPlay?            /*The query returns 1.*/
```

3.13.7 :LA:POD<n>:THReshold

Syntax

:LA:POD<*n*>**:THReshold** <*thre*>

:LA:POD<*n*>**:THReshold?**

Description

Sets or queries the threshold of the specified default channel group. The default unit is V.

Parameter

Name	Type	Range	Default
<n>	Integer	1 to 2	-
<thre>	Real	-15.0 V to +15.0 V	0 V

Remarks

2 default channel groups: POD1 (D0 to D7) or POD2 (D8 to D15).

Return Format

The query returns the current threshold of the specified channel group in scientific notation.

Example

```
:LA:POD1:THReshold 3.3    /*Sets the threshold of the channel group
POD1 (D0 to D3) to 3.3 V.*/
:LA:POD1:THReshold?       /*The query returns 3.3E+00.*/
```

3.13.8 :LA:SIZE

Syntax

:LA:SIZE <*size*>

:LA:SIZE?

Description

Sets or queries the size of the waveforms of the enabled channel on the screen.

Parameter

Name	Type	Range	Default
<size>	Discrete	{SMALl|LARGe|MEDium}	MEDium

Remarks

L (large) can only be used when the number of the currently enabled channels is no more than 8.

Return Format

The query returns SMAL, LARG, or MED.

Example

```
:LA:SIZE SMALl     /*Sets the waveform display size to SMALl.*/
:LA:SIZE?          /*The query returns SMAL.*/
```

3.14 :LAN Commands

The **:LAN** commands are used to set or query the LAN-related parameters.

NOTE

After configuring all the other :LAN commands, you need to send *:LAN:APPLy* to make all the LAN configurations take effect.

3.14.1 :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.*/
```

3.14.2 :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.*/
```

3.14.3 :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 192.168.1.1.*/
```

3.14.4 :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 192.168.1.1.*/
```

3.14.5 :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
```

3.14.6 :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
```

3.14.7 :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.*/
```

3.14.8 :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 192.168.1.10.*/
```

3.14.9 :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 255.255.255.0.*/
```

3.14.10 :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 wait...
- **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
```

3.14.11 :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
```

3.14.12 :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.*/
```

3.14.13 :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
```

3.14.14 :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
```

3.14.15 :LAN:APPLy

Syntax

:LAN:APPLy

Description

Applies the network configuration.

Parameter

N/A

Remarks

After configuring all the LAN-related parameters with the :LAN commands, you need to send this command to make all the LAN configurations take effect.

Return Format

N/A

Example

```
N/A
```

3.15 :MASK Commands

:MASK commands are used to set or query the pass/fail test related parameters.

During the product design and manufacturing process, you usually need to monitor the variations of the signal or judge whether the product is up to standard. The standard pass/fail test function of this series oscilloscope can accomplish this task perfectly. You can use this function to set the test rules based on standard waveforms and define the mask. It compares the signal under test with the mask and displays the test results.

3.15.1 :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 is disabled in the following conditions:

- When the horizontal time base is in ROLL mode; (to set or query the horizontal time base mode, run *:TIMebase:MODE*.)
- When the delayed sweep mode (Zoom) is enabled; (to set or query the on/off status of the delayed sweep, run *:TIMebase:DELay:ENABle*.)
- When performing the waveform recording and playing.

Return Format

The query returns 1 or 0.

Example

```
:MASK:ENABle ON     /*Enables the pass/fail test function.*/
:MASK:ENABle?       /*The query returns 1.*/
```

3.15.2 :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, CHAN4.

Example

```
:MASK:SOURce CHANnel2    /*Sets the source of the pass/fail test to
CHANnel2.*/
:MASK:SOURce?             /*The query returns CHAN2.*/
```

3.15.3 :MASK:OPERate

Syntax

:MASK:OPERate <*oper*>

:MASK:OPERate?

Description

Starts 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    /*Starts the pass/fail test.*/
:MASK:OPERate?        /*The query returns RUN.*/
```

3.15.4 :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.*/
```

3.15.5 :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.*/
```

3.15.6 :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. You can use *:MASK:ENABle* to query or set the status of the pass/fail test function. You can use *:MASK:OPERate* to query or set the running status.
- You can use *:MASK:X* and *:MASK:Y* to query or set the horizontal and vertical adjustment parameters.

Return Format

N/A

Example

```
N/A
```

3.15.7 :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
```

3.15.8 :MASK:FAILed?

Syntax

```
:MASK:FAILed?
```

Description

Queries the total number of failed frames in the pass/fail test results.

Parameter

N/A

Remarks

N/A

Return Format

The query returns an integer.

Example

```
N/A
```

3.15.9 :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
```

3.15.10 :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
```

3.15.11 :MASK:OUTPut:ENABle

Syntax

:MASK:OUTPut:ENABle <*bool*>

:MASK:OUTPut:ENABle?

Description

Sets or queries the output on/off status of the rear-panel **[AUX OUT]** connector.

Parameter

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

Remarks

- If enabled, in the Utility menu, "AUX Out" is automatically set to "PassFail". When a successful or failed event is detected, a pulse will be output from the **[AUX OUT]** connector.

- If disabled, in the Utility menu, "AUX Out" is automatically set to "TrigOut". The output of the **[AUX OUT]** connector is irrelevant with the pass/fail test.

Return Format

The query returns 0 or 1.

Example

```
:MASK:OUTPut:ENABle ON     /*Enables the Aux output.*/
:MASK:OUTPut:ENABle?       /*The query returns 1.*/
```

3.15.12 :MASK:OUTPut:EVENt

Syntax

:MASK:OUTPut:EVENt <*item*>

:MASK:OUTPut:EVENt?

Description

Sets or queries the output event.

Parameter

Name	Type	Range	Default
<item>	Discrete	{FAIL|PASS}	FAIL

Remarks

N/A

Return Format

The query returns FAIL or PASS.

Example

```
:MASK:OUTPut:EVENt PASS /*Sets the output event to PASS.*/
:MASK:OUTPut:EVENt? /*The query returns PASS.*/
```

3.15.13 :MASK:OUTPut:TIME

Syntax

:MASK:OUTPut:TIME <*time*>

:MASK:OUTPut:TIME?

Description

Sets or queries the output pulse time.

Parameter

Name	Type	Range	Default
<time>	Real	100 ns to 10 ms	1 µs

Remarks

N/A

Return Format

The query returns the pulse time in scientific notation.

Example

```
:MASK:OUTPut:TIME 0.000003 /*Sets the pulse time to 3 µs.*/
:MASK:OUTPut:TIME? /*The query returns 3.000000E-6.*/
```

3.16 :MATH<n> Commands

:MATH<n> commands are used to set various math operation function of the waveform between channels.

This series oscilloscopes can realize multiple math operations between waveforms of different channels, including arithmetic operation, function operation, FFT operation, logic operation, and digital filter.

Operator

Arithmetic Operation

The arithmetic operations supported by this oscilloscope include A+B, A-B, A×B, and A÷B.

- **A+B** adds the waveform voltage values of signal source A and B point by point and displays the results.
- **A-B** subtracts the waveform voltage values of signal source B from that of source A point by point and displays the results.
- **A×B** multiplies the waveform voltage values of signal source A and B point by point and displays the results.
- **A÷B** divides the waveform voltage values of signal source A by that of source B point by point and displays the results. It can be used to analyze the Multiple relation of the two channels waveforms. When the voltage of signal source B is 0 V, the division result is treated as 0.

Function Operation

The available function operation types of this oscilloscope include Intg, Diff, Sqrt, Lg (Base 10 Exponential), Ln, Exp, Abs, and AX+B.

- **Intg:** calculates the integral of the selected source. For example, you can use integral to measure the area under a waveform or the pulse energy.
- **Diff:** calculates the discrete time derivative of the selected source. For example, you can use differentiate to measure the instantaneous slope of a waveform.
- **Sqrt:** calculates the square roots of the selected source point by point and displays the results.
- **Lg (Base 10 Exponential):** calculates the base 10 exponential of the selected source point by point and displays the results.
- **Ln:** calculates the natural logarithm (Ln) of the selected source point by point and displays the results.
- **Exp:** calculates the exponential of the selected source point by point and displays the results.
- **Abs:** calculates the absolute value of the selected source and displays the results.
- **AX+B:** applies a linear function to the selected source, and displays the results.

FFT Operation

FFT (Fast Fourier Transform) is used to transform time-domain signals to frequency-domain components (frequency spectrum). This oscilloscope provides FFT operation function which enables you to observe the time-domain waveform and spectrum of the signal at the same time. FFT operation can facilitate the following works:

- Measure harmonic components and distortion in the system;
- Display the characteristics of the noise in DC power;
- Analyze vibration.

Table 3.185 Window Function

Window Function	Characteristics	Waveforms Applicable to the Window Function
Rectangular	Best frequency resolution	Transient or short pulse, the signal levels before and after the multiplication are basically the same

Window Function	Characteristics	Waveforms Applicable to the Window Function
	Poorest amplitude resolution Similar to the situation when no window is applied	Sine waveforms with the same amplitudes and rather similar frequencies Wide band random noise with relatively slow change of waveform spectrum
Blackman-Harris	Best amplitude resolution Poorest frequency resolution	Single frequency signal, searching for higher order harmonics
Hanning	Better frequency resolution and poorer amplitude resolution compared with Rectangular	Sine, periodic, and narrow band random noise
Hamming	A little bit better frequency resolution than Hanning	Transient or short pulse, the signal levels before and after the multiplication are rather different
Flattop	Measure the signals accurately	Measure the signal that has no accurate reference and requires an accurate measurement
Triangle	Better frequency resolution	Measure the narrow band signal and that has strong noise interference

Spectral leakage can be considerably minimized when a window function is used. The oscilloscope provides 6 FFT window functions which have different characteristics and are applicable to measure different waveforms, as shown in the table below. You need to select the window function according to the characteristics of the waveform to be measured.

Logic Operation

The logic operations supported by this oscilloscope include A&&B, A||B, A^B, and !A. The results of logic operation of one binary bit are shown in the table below:

Table 3.186 Logic Operation Results

A	B	A&&B	A||B	A^B	!A
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

Digital Filter

The digital filters supported by this oscilloscope include: low-pass filter, high-pass filter, band-pass filter, and band-stop filter.

- **LowPass** only allows the signals whose frequencies are lower than the current upper limit frequency to pass.
- **HighPass** only allows the signals whose frequencies are higher than the current lower limit frequency to pass.
- **BandPass** only allows the signals whose frequencies are higher than the current lower limit frequency and lower than the current upper limit frequency to pass.
- **BandStop** only allows the signals whose frequencies are lower than the current lower limit frequency or higher than the current upper limit frequency to pass.

3.16.1 :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 math operation of Math1.*/
:MATH1:DISPlay?       /*The query returns 1.*/
```

3.16.2 :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|LG|LN|EXP|ABS|LPASs|HPASs|BPASs|BSTop|AXB}	ADD

Remarks

The math operation supported include A+B (ADD), A-B (SUBTract), A×B (MULTiply), A÷B (DIVision), AND, OR, XOR, NOT, FFT, INTG, DIFF, SQRT, LG (Base 10 Exponential), natural logarithm (LN), EXP, ABS, LPASs, HPASs, BPASs, BSTop, and AXB. For details, refer to *Operator*.

Return Format

The query returns ADD, SUBT, MULT, DIV, AND, OR, XOR, NOT, FFT, INTG, DIFF, SQRT, , LG, LN, EXP, ABS, LPAS, HPAS, BPAS, BST, or AXB.

Example

```
:MATH1:OPERator INTG /*Sets the math operator of Math1 to INTG.*/
:MATH1:OPERator?        /*The query returns INTG.*/
```

3.16.3 :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

- When n = 1, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10}.
- When n = 2, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1}.
- When n = 3, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1|MATH2}.
- When n = 4, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1|MATH2|MATH3}.
- 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.
- For detailed operations, refer to the descriptions in *Operator*.

Return Format

The query returns MATH1, MATH2, MATH3, CHAN1, CHAN2, CHAN3, CHAN4, REF1, REF2, REF3, REF4, REF5, REF6, REF7, REF8, REF9, or REF10.

Example

```
:MATH1:SOURce1 CHANnel3   /*Sets Source A of the arithmetic
operation to CHANnel3.*/
:MATH1:SOURce1?           /*The query returns CHAN3.*/
```

3.16.4 :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| REF8|REF9|REF10|MATH1| MATH2|MATH3}	CHANnel1

Remarks

This command is only available for arithmetic operation (containing two sources). For detailed operations, refer to the descriptions in *Operator*.

- When n = 1, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10}.
- When n = 2, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1}.
- When n = 3, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1|MATH2}.
- When n = 4, the range of the parameter is {CHANnel1|CHANnel2|CHANnel3|CHANnel4|REF1|REF2|REF3|REF4|REF5|REF6|REF7|REF8|REF9|REF10|MATH1|MATH2|MATH3}.

Return Format

The query returns MATH1, MATH2, MATH3, CHAN1, CHAN2, CHAN3, CHAN4, REF1, REF2, REF3, REF4, REF5, REF6, REF7, REF8, REF9, or REF10.

Example

```
:MATH1:SOURce2 CHANnel3    /*Sets Source B of the arithmetic
operation to CHANnel3.*/
:MATH1:SOURce2?            /*The query returns CHAN3.*/
```

3.16.5 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The logic operations include A&&B, A||B, A^B, and !A.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MATH1:LSOurce1 CHANnel3    /*Sets Source A of the logic operation
to CHANnel3.*/
:MATH1:LSOurce1?            /*The query returns CHAN3.*/
```

3.16.6 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|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 D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:MATH1:LSOurce2 CHANnel4    /*Sets Source B of the logic operation
to CHANnel4.*/
:MATH1:LSOurce2?            /*The query returns CHAN4.*/
```

3.16.7 :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 the 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.
- This command is invalid for logic operation and FFT operation.

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 200 mV.*/
:MATH1:SCALe?   /*The query returns 2.000000E-1.*/
```

3.16.8 :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

This command is invalid for logic operation and FFT operation.

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.*/
```

3.16.9 :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 and logical operation.

Return Format

The query returns 1 or 0.

Example

```
:MATH1:INVert ON     /*Enables the inverted display.*/
:MATH1:INVert?       /*The query returns 1.*/
```

3.16.10 :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
```

3.16.11 :MATH<n>:GRID

Syntax

:MATH<*n*>**:GRID** <*grid*>

:MATH<*n*>**:GRID?**

Description

Sets or queries the grid type of the math operation display.

Parameter

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

Remarks

- **FULL:** turns the background grid and coordinates on.
- **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

```
:MATH1:GRID NONE /*Turns the background grid and coordinates off.*/
:MATH1:GRID? /*The query returns NONE.*/
```

3.16.12 :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 CENT.

Example

```
:MATH1:EXPand CENTer /*Sets the waveform of Math1 to be expanded or
compressed around the screen center.*/
:MATH1:EXPand? /*The query returns CENT.*/
```

3.16.13 :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 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.*/
```

3.16.14 :MATH<n>:FFT:SOURce

Syntax

:MATH<*n*>**:FFT:SOURce** <*source*>

:MATH<*n*>**:FFT:SOURce?**

Description

Sets or queries the channel source of FFT operation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<source>	Discrete	{CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3}	CHANnel1

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, or MATH3.

Example

```
:MATH1: FFT:SOURce CHANnel3          /*Sets the channel source of FFT
operation to CHANnel3.*/
:MATH1: FFT:SOURce?                  /*The query returns CHAN3.*/
```

3.16.15 :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

- Spectral leakage can be considerably minimized when a window function is used.
- 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. For the characteristics of the window function and its applicable waveforms, refer to *Table 3.185 Window Function* .

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.*/
```

3.16.16 :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.*/
```

3.16.17 :MATH<n>:FFT:MODE

Syntax

:MATH<*n*>**:FFT:MODE** <*mode*>

:MATH<*n*>**:FFT:MODE?**

Description

Sets or queries the mode of FFT operation.

Parameter

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

Remarks

NORMal: indicates Normal mode.

AVERage: indicates Average mode.

MAXHold: indicates Max Hold mode.

Return Format

The query returns NORM, AVER, or MAXH.

Example

```
:MATH1:FFT:MODE NORMal   /*Sets the mode of FFT operation of Math1
to NORMal.*/
:MATH1:FFT:MODE?    /*The query returns NORM.*/
```

3.16.18 :MATH<n>:FFT:AVCNt

Syntax

:MATH<*n*>**:FFT:AVCNt** <*cnt*>

:MATH<*n*>**:FFT:AVCNt?**

Description

Sets or queries the average count for FFT in Average mode.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2|3|4}	-
<cnt>	Integer	2 to 1,000	10

Remarks

N/A

Return Format

The query returns an integer ranging from 2 to 1,000.

Example

```
:MATH1:FFT:AVCNt 1000  /*Sets the average count for FFT in Average
mode to 1,000.*/
:MATH1:FFT:AVCNt?   /*The query returns 1000.*/
```

3.16.19 :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}	-

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

Remarks

- When the unit is dBm/dBV, the range of the parameter <scale> is from 1.0E-9 dB to 5.00 GdB. The default value is 2.0 dB.
- When the unit is set to V_{rms}, the range of the parameter <scale> is from 1.00 nV_{rms} to 5.00 GV_{rms}. The default value is 1 mV_{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.*/
```

3.16.20 :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}	-
<offset>	Real	Refer to *Remarks*	0 dB

Remarks

- When the unit is set to dBm/dBV, the range of <offset> is from -1.00G dBV to 1.00 GdBV. The default value is 0.0 dBV.

- When the unit is set to V_{rms}, the range of <offset> is from -1.00 GV_{rms} to 1.00 GV_{rms}. The default value is 0.00 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.*/
```

3.16.21 :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 2 GHz	1 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.*/
```

3.16.22 :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 <2 GHz	1 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.*/
```

3.16.23 :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}	-

Name	Type	Range	Default
<value>	Real	0 Hz to (stop frequency -10 Hz)	1 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.*/
```

3.16.24 :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 frequency +10 Hz) to 2 GHz	10 MHz

Remarks

The range of the stop 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.*/
```

3.16.25 :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.*/
```

3.16.26 :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.*/
```

3.16.27 :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	-

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.*/
```

3.16.28 :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 x VerticalScale)	1.8 dB

Remarks

VerticalScale indicates the vertical scale of FFT.

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.*/
```

3.16.29 :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.*/
```

3.16.30 :MATH<n>:FFT:SEARch:RES?

Syntax

:MATH<*n*>**:FFT:SEARch:RES?**

Description

Queries the FFT peak search results table.

Parameter

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

Remarks

N/A

Return Format

The query returns the peak search results table in strings.

Example

```
:MATH1:FFT:SEARch:RES?    /*The query returns the peak search
results table in strings.*/
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
```

3.16.31 :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}	LPASs

Remarks

The oscilloscope provides 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.*/
```

3.16.32 :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*	Refer to *Remarks*

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 x screen sample rate) to (0.5 x screen sample rate). The screen sample rate is indicated at the top of the current math operation result window.
- When the filter type is set to BPASs (Band Pass Filter) or BSTop (Band Stop Filter), you need to set two cut-off frequencies. The cut-off frequency 1 must be smaller than the cut-off frequency 2. Run this command to set the cut-off frequency 1; 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 x screen sample rate) to (0.495 x screen sample rate).
- The default value of the parameter <freq1> is related to the filter type.
  - When the filter type is set to LPASs (Low Pass Filter), BPASs (Band Pass Filter), or BSTop (Band Stop Filter), the default value of <freq1> is (0.005 x screen sample rate).
  - When the filter type is set to HPASs (High Pass Filter), the default value of <freq1> is (0.1 x screen sample rate).
- To set or query the filter type, run the *:MATH<n>:FILTer:TYPE* command.

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 +100000000000E
+06.*/
```

3.16.33 :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. The cut-off frequency 2 must be greater than the cut-off frequency 1. 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 x screen sample rate) to (0.5 x screen sample rate). The screen sample rate is indicated at the top of the current math operation result window.

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.*/
```

3.16.34 :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.*/
```

3.16.35 :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.*/
```

3.16.36 :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}	-
<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.*/
```

3.16.37 :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 (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 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.*/
```

3.16.38 :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.*/
```

3.16.39 :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.*/
```

3.16.40 :MATH<n>:WINDow:TITLe?

Syntax

:MATH<*n*>**:WINDow:TITLe?**

Description

Queries the title of the specified math operation window.

Parameter

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

Remarks

N/A

Return Format

The query returns the title of the specified math operation window in strings.

Example

```
:MATH1:WINDow:TITLe? /*The query returns Math1 CH1*CH1 Scale:0U
Sa:2GSa/s.*/
```

3.16.41 :MATH<n>:LABel:SHOW

Syntax

:MATH<*n*>**:LABel:SHOW** <*bool*>

:MATH<*n*>**:LABel:SHOW?**

Description

Sets or queries whether to display the waveform label of the specified operation.

Parameter

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

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:MATH1:LABel:ENABle ON          /*Enables the display of the
label.*/
:MATH1:LABel:ENABle?            /*The query returns 1.*/
```

3.16.42 :MAHT<n>:DISMode

Syntax

:MATH<*n*>**DISMode** <*bool*>

:MATH<*n*>**DISMode?**

Description

Sets or queries the display area of math operation function.

Parameter

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

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

Remarks

0|OFF: The math operation waveform is displayed in the math operation window.

1|ON: The math operation waveform is displayed in the main waveform view.

Return Format

The query returns 1 or 0.

Example

```
:MATH1:DISMode 1   /*Sets the math operation waveform to be
displayed in the main waveform view.*/
:MATH1:DISMode?   /*The query returns 1.*/
```

3.17 :MEASure Commands

:MEASure commands are used to set and query the parameters related to measurements.

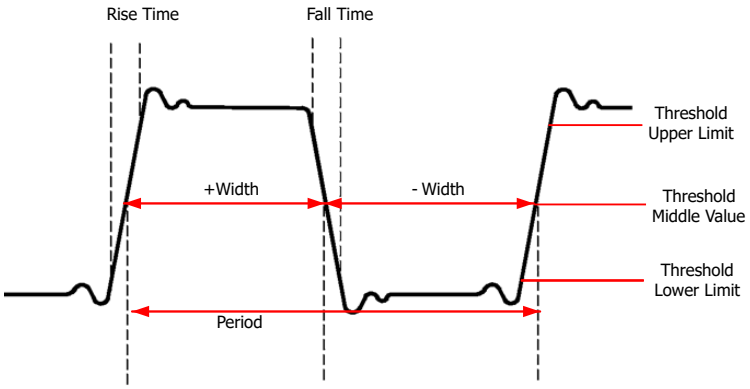
This oscilloscope allows you to set the measurement source, enable or disable the all measurement function, the statistical function, and etc.

Measurement Parameters

TIP

If there is no signal input for the current source or the measurement result is not within the valid range (too large or too small), then the measurement results are invalid.

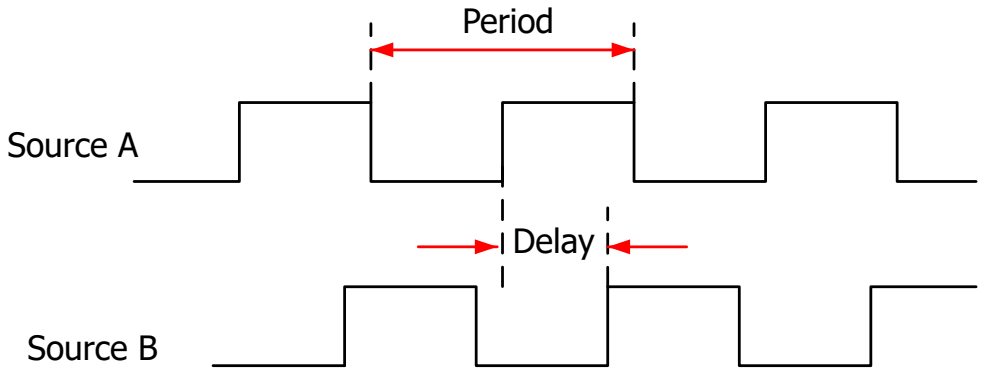
Horizontal Parameters



- **Period (PERiod):** defined as the time between the middle threshold points of two consecutive, like-polarity edges.
- **Frequency (FREQuency):** defined as the reciprocal of period.

- **Rise Time (RTIMe):** indicates the time for the signal amplitude to rise from the threshold lower limit to the threshold upper limit.
- **Fall Time (FTIMe):** indicates the time for the signal amplitude to drop from the threshold upper limit to the threshold lower limit.
- **+Width (PWIDth):** indicates the time between the threshold middle value of a rising edge to the threshold middle value of the next falling edge.
- **-Width (NWIDth):** indicates the time between the threshold middle value of a falling edge to the threshold middle value of the next rising edge.
- **+Duty (PDUTy):** indicates the ratio of the positive pulse width to the period.
- **-Duty (NDUTy):** indicates the ratio of the negative pulse width to the period.
- **Tvmax (TVMAX):** indicates the time that corresponds to the maximum value of the waveform (Vmax).
- **Tvmin (TVMIN):** indicates the time that corresponds to the minimum value of the waveform (Vmin).

Delay and Phase Parameters

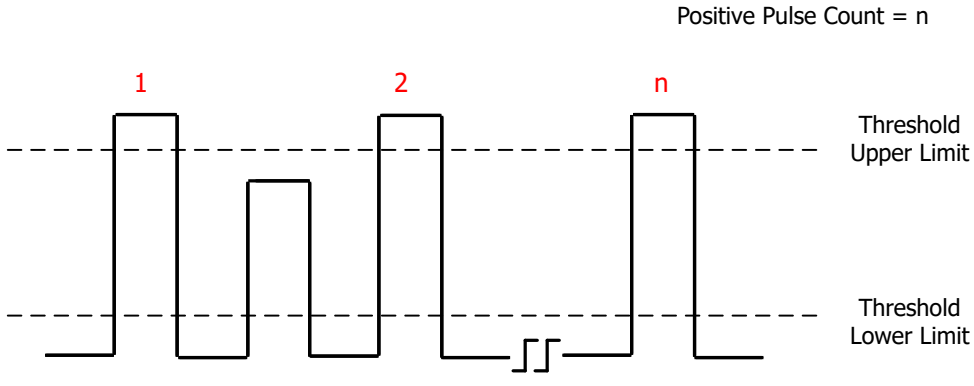


- **Delay(r-r) (RRDelay):** indicates the time difference between the threshold middle values of the rising edge of Source A and that of Source B. Negative delay indicates that the rising edge of Source A occurred after that of Source B.
- **Delay(f-f) (FFDelay):** indicates the time difference between the threshold middle values of the falling edge of Source A and that of Source B. Negative delay indicates that the falling edge of Source A occurred after that of Source B.

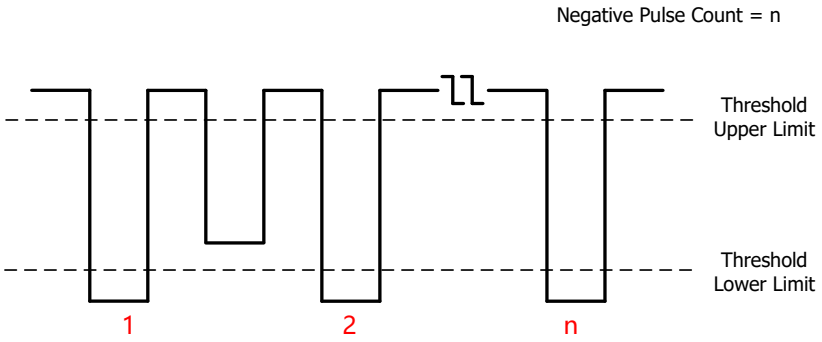
- **Delay(r-f) (RFDelay):** indicates the time difference between the threshold middle values of the rising edge of Source A and the falling edge of Source B. Negative delay indicates that the rising edge of Source A occurred after the falling edge of Source B.
- **Delay(f-r) (FRDelay):** indicates the time difference between the threshold middle values of the falling edge of Source A and the rising edge of Source B. Negative delay indicates that the falling edge of Source A occurred after the rising edge of Source B.
- **Phase(r-r) (RRPHase):** indicates the phase deviation between the threshold middle values of the rising edge of Source A and that of Source B.
- **Phase(f-f) (FFPHase):** indicates the phase deviation between the threshold middle values of the falling edge of Source A and that of Source B.
- **Phase(r-f) (RFPHase):** indicates the phase deviation between the threshold middle values of the rising edge of Source A and the falling edge of Source B.
- **Phase(f-r) (FRPHase):** indicates the phase deviation between the threshold middle values of the falling edge of Source A and the rising edge of Source B.

Count Values

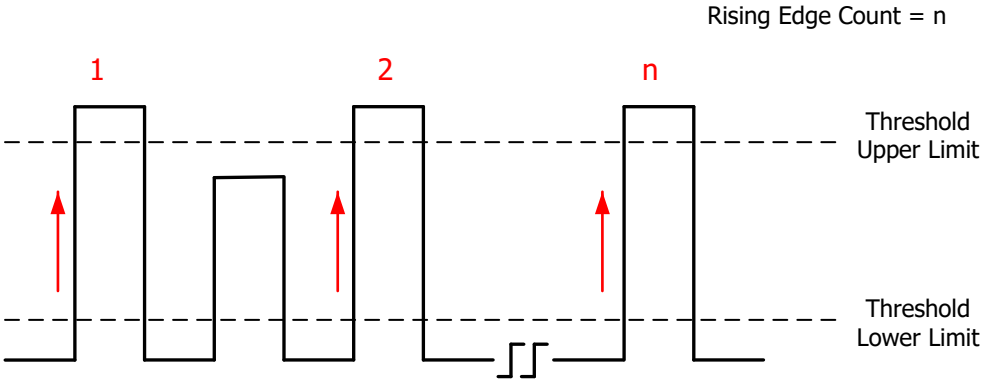
- **Positive Pulse Count (PPULses):** It is specified as the number of positive pulses that rise from under the threshold lower limit to above the threshold upper limit.



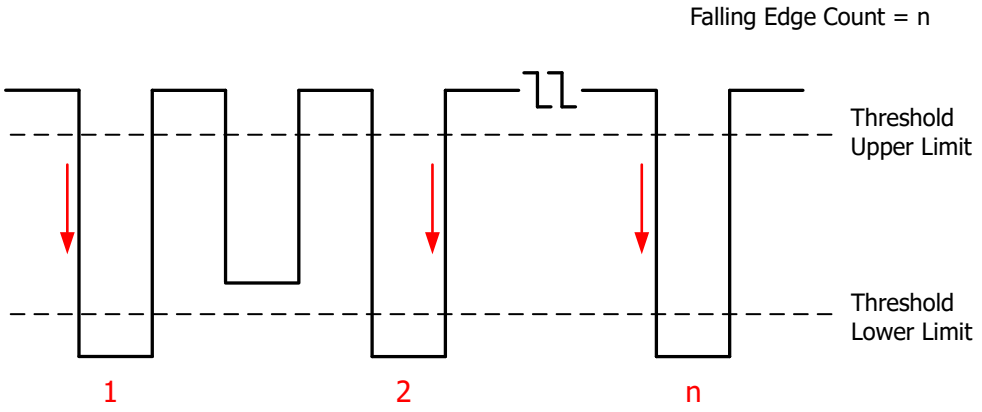
- **Negative Pulse Count (NPULses):** It is specified as the number of negative pulses that fall from above the threshold upper limit to below the threshold lower limit.



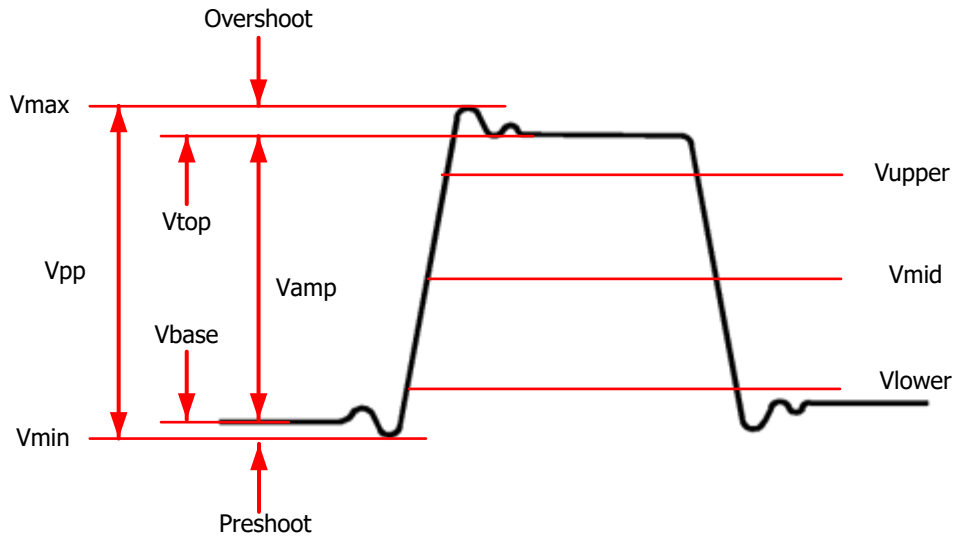
- **Rising Edge Count (PEDGes):** It is specified as the number of rising edges that rise from under the threshold lower limit to above the threshold upper limit.



- **Falling Edge Count (NEDGes):** It is specified as the number of falling edges that fall from above the threshold upper limit to below the threshold lower limit.



Voltage Parameters



- **Vmax (VMAX):** indicates the voltage value from the highest point of the waveform to the GND.
- **Vmin (VMIN):** indicates the voltage value from the lowest point of the waveform to the GND.
- **Vpp (VPP):** indicates the voltage value from the highest point to the lowest point of the waveform.
- **Vtop (VTOP):** indicates the voltage value from the flat top of the waveform to the GND.
- **Vbase (VBASe):** indicates the voltage value from the flat base of the waveform to the GND.
- **Vamp (VAMP):** indicates the voltage value from the top of the waveform to the base of the waveform.
- **Vupper (VUPPer):** indicates the actual voltage value that corresponds to the threshold maximum value.
- **Vmid (VMID)** indicates the actual voltage value that corresponds to the threshold middle value.
- **Vlower (VLOWer):** indicates the actual voltage value that corresponds to the threshold minimum value.

- **Vavg (VAVG):** indicates the arithmetic average value on the whole waveform or in the gating area.
- **VRMS (VRMS):** indicates the root mean square value on the whole waveform or in the gating area.
- **Per.VRMS (PVRMs):** indicates the root mean square value within a period.
- **Overshoot (OVERshoot):** indicates the ratio of the difference between the maximum value and the top value of the waveform to the amplitude value.
- **Preshoot (PREShoot):** indicates the ratio of the difference between the minimum value and the base value of the waveform to the amplitude value.
- **AC RMS (ACRMs):** indicates the root-mean-square value of the waveforms, with the DC component removed.

Other Parameters

- **Positive Slew Rate (PSLewrate):** On the rising edge, first calculate the difference between the high value and the low value, then use the difference to divide the corresponding time value to obtain the positive slew rate.
- **Negative Slew Rate (NSLewrate):** On the falling edge, first calculate the difference between the low value and the high value, then use the difference to divide the corresponding time value to obtain the negative slew rate.
- **Area (MARea):** indicates the area of the whole waveform within the screen. The unit is V*s. The area of the waveform above the zero reference (namely the vertical offset) is positive, and the area of the waveform below the zero reference is negative. The area measured is the algebraic sum of the area of the whole waveform within the screen.
- **Period Area (MPARea):** indicates the area of the first period of waveform on the screen. The unit is V*s. The area of the waveform above the zero reference (namely the vertical offset) is positive, and the area of the waveform below the zero reference is negative. The area measured is the algebraic sum of the whole period area.

Measurement Results

This oscilloscope can make a statistics and display the measurement results.

- **MAXimum:** the maximum value.
- **MINimum:** the minimum value.
- **CURRent:** the current value.
- **AVERages:** the average value.
- **DEViation:** the standard deviation.
- **CNT:** the count value.

3.17.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|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 D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.17.2 :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|PVRMs|PPULses|NPULses|PEDGes|NEDGes|RRDelay|RFDelay|FRDelay|FFDelay|RRPHase|RFPHase|FRPHase|FFPHase|ACRMs}	-
<src>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	-

Remarks

- For the detailed measurement item parameter <item>, refer to *Measurement Parameters*.
- The parameter [,<src>[,<src>]] is used to set the source of the parameter under measurement.
  - The available source channels are CH1-CH4 and MATH1-MATH4. After the logic probe is connected, the available sources also include digital channels (D0-D15).

- 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 current measurement value in scientific notation.

Example

```
:MEASure:ITEM OVERshoot,CHANnel2 /*Measures the overshoot value of
Channel 2.*/
:MEASure:ITEM? OVERshoot,CHANnel2 /*The query returns 8.888889E-3.*/
```

3.17.3 :MEASure:DELete

Syntax

`:MEASure:DELete`

Description

Clears all the enabled measurement items.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.17.4 :MEASure:AMSource

Syntax

:MEASure:AMSource <*chan*>

:MEASure:AMSource?

Description

Sets or queries the channel source of All Measure function.

Parameter

Name	Type	Range	Default
<chan>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|OFF}	OFF

Remarks

If <chan> is set to OFF, the All Measure function is disabled.

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, or OFF.

Example

```
:MEASure:AMSource CHANnel1          /*Sets the channel source of All
Measure function to CHANnel1.*/
:MEASure:AMSource?                  /*The query returns CHAN1.*/
```

3.17.5 :MEASure:STATistic:COUNt

Syntax

:MEASure:STATistic:COUNt <*val*>

:MEASure:STATistic:COUNt?

Description

Sets or queries the statistics count.

Parameter

Name	Type	Range	Default
<val>	Integer	2 to 100,000	1,000

Remarks

N/A

Return Format

The query returns an integer ranging from 2 to 100,000.

Example

```
:MEASure:STATistic:COUNt 1000       /*Sets the statistics count to
1,000.*/
:MEASure:STATistic:COUNt?      /*The query returns 1000.*/
```

3.17.6 :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.*/
```

3.17.7 :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
```

3.17.8 :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| PREShoot|MAREa|MPAREa| PERiod|FREQuency|RTIMe| FTIMe|PWIDth|NWIDth|PDUTy| NDUTy|TVMAX|TVMIN| PSLewrate|NSLewrate|VUPPer| VMID|VLOWer|PVRMs|PPULses| NPULses|PEDGes|NEDGes| RRDelay|RFDelay|FRDelay| FFDelay|RRPHase|RFPHase| FRPHase|FFPHase|ACRMs}	-
<src>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8| D9|D10|D11|D12|D13|D14|D15| CHANnel1|CHANnel2|	-

Name	Type	Range	Default
		CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	
<type>	Discrete	{MAXimum|MINimum|CURRent|AVERages|DEViation|CNT}	-

Remarks

- For the detailed measurement item parameter <item>, refer to *Measurement Parameters*.
- The parameter [,<src>[,<src>]] is used to set the source of the parameter under measurement.
  - The available source channels are CH1-CH4 and MATH1-MATH4. After the logic probe is connected, the available sources also include digital channels (D0-D15).
  - 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*).
- For the results of <type>, refer to *Measurement Parameters*.

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   /*Queries the maximum value.
The query returns 9.120000E-1.*/
```

3.17.9 :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

The range of the upper limit of the threshold level is related to the current threshold level middle value. You can run the *:MEASure:SETup:MID* command to set or query the middle value of the threshold level of the analog channel in auto measurement.

- When the threshold type is percentage, its range is from (threshold middle value + 1%) to 100%.
- When the threshold type is absolute, its range changes with the probe ratio. Its max. range is from -100 MV to 100 MV; and its min. range is from -20 V to 20 V.
- When the set upper limit is smaller than the current threshold middle value, a message "Set at lower limit" will be displayed, and the threshold middle value will not be modified automatically.

Return Format

The query returns the threshold in scientific notation. When the threshold type is absolute, the default unit of the return value is V.

Example

```
:MEASure:SETup:MAX 95    /*Sets upper limit of the threshold level
to 95%.*/
:MEASure:SETup:MAX?      /*The query returns 9.500000E+01.*/
```

3.17.10 :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

The set middle value must be smaller than the currently set upper limit and greater than the currently set lower limit.

You can run the *:MEASure:SETup:MAX* and *:MEASure:SETup:MIN* commands respectively to set/query the threshold level upper and lower limit of the current analog channel in auto measurement.

Return Format

The query returns the threshold in scientific notation. When the threshold type is absolute, the default unit of the return value is V.

Example

```
:MEASure:SETup:MID 89     /*Sets the middle value of the threshold
level to 89%.*/
:MEASure:SETup:MID?       /*The query returns 8.900000E+01.*/
```

3.17.11 :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

The range of the threshold level lower limit is related to the current threshold middle value. You can send the *:MEASure:SETup:MID* command to set or query the threshold middle value of the current analog channel in auto measurement.

- When the threshold type is percentage, its range is from 0% to (threshold middle value - 1%).
- When the threshold type is absolute, its range changes with the probe ratio. Its max. range is from -100 MV to 100 MV; and its min. range is from -20 V to 20 V.
- When the set lower limit is greater than the current threshold middle value, a message "Set at upper limit" will be displayed, and the threshold middle value will not be modified automatically.

Return Format

The query returns the threshold in scientific notation. When the threshold type is absolute, the default unit of the return value is V.

Example

```
:MEASure:SETup:MIN 53     /*Sets the lower limit of the threshold
level to 53%.*/
:MEASure:SETup:MIN?       /*The query returns 5.300000E+01.*/
```

3.17.12 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|	CHANnel1

Name	Type	Range	Default
		CHANnel1|CHANnel2| CHANnel3|CHANnel4|MATH1| MATH2|MATH3|MATH4}	

Remarks

This command has the same function as the *:MEASure:SOURce* and *:MEASure:SETup:DSA* commands.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.17.13 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8| D9|D10|D11|D12|D13|D14|D15| 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 D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.17.14 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

This command has the same function as the *:MEASure:SOURce* and *:MEASure:SETup:PSA* commands.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.17.15 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|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 D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.17.16 :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.*/
```

3.17.17 :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 PERC or ABS.

Example

```
:MEASure:THReshold:TYPE ABSolute    /*Sets the threshold type to
ABSolute.*/
:MEASure:THReshold:TYPE?             /*The query returns ABS.*/
```

3.17.18 :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

For the default threshold level in absolute, its upper limit and lower limit are + (vertical scale x 3) and -(vertical scale x 3), respectively.

Return Format

N/A

Example

```
N/A
```

3.17.19 :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|CURSor}	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.
- **CURSor:** when you select it, two cursors will be displayed on the screen.

Return Format

The query returns MAIN, ZOOM, or CURS.

Example

```
:MEASure:AREA ZOOM            /*Sets the type of the measurement
range to ZOOM.*/
:MEASure:AREA?                  /*The query returns ZOOM.*/
```

3.17.20 :MEASure:TYPE

Syntax

:MEASure:TYPE <*type*>

:MEASure:TYPE?

Description

Sets or queries the measurement type.

Parameter

Name	Type	Range	Default
<type>	Discrete	{THReshold|RANGe|AMPMethod}	THReshold

Remarks

- **THReshold:** sets the measurement type to Threshold.
- **RANGe:** sets the measurement type to Range.
- **AMPMethod:** sets the measurement type to Amplitude Method.

Return Format

The query returns THR, RANG, or AMPM.

Example

```
:MEASure:TYPE RANGe            /*Sets the measurement type to
RANGe.*/
:MEASure:TYPE?                  /*The query returns RANG.*/
```

3.17.21 :MEASure:CREGion:CAX

Syntax

:MEASure:CREGion:CAX <*cax*>

:MEASure:CREGion:CAX?

Description

Sets or queries the horizontal position of Cursor A when the measurement range is the "cursor region".The default unit s.

Parameter

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

Remarks

- Its range is related to the current horizontal scale and position of the instrument.
- After the setting command is sent, the value in the field of CursorA in the user interface is set.The default unit s.

- You can run the *:MEASure:AREA* command to set the measurement range to "cursor region".

Return Format

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

Example

```
:MEASure:CREGion:CAX -0.00001  /*Sets the position of Cursor A to
-10 μs.*/
:MEASure:CREGion:CAX  /*The query returns -1.000000E-5.*/
```

3.17.22 :MEASure:CREGion:CBX

Syntax

:MEASure:CREGion:CBX <*cbx*>

:MEASure:CREGion:CBX?

Description

Sets or queries the horizontal position of Cursor B when the measurement range is the "cursor region".The default unit s.

Parameter

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

Remarks

- Its range is related to the current horizontal scale and position of the instrument.
- After the setting command is sent, the value in the field of CursorB in the user interface is set.The default unit s.
- You can run the *:MEASure:AREA* command to set the measurement range to "cursor region".

Return Format

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

Example

```
:MEASure:CREGion:CBX 0.000005  /*Sets the position of Cursor B to 5
μs.*/
:MEASure:CREGion:CBX  /*The query returns 5.000000E-6.*/
```

3.17.23 :MEASure:CREGion:CABX

Syntax

:MEASure:CREGion:CABX <*bool*>

:MEASure:CREGion:CABX?

Description

Sets or queries whether Cursor A and Cursor B are linked.

Parameter

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

Remarks

1|ON: sets Cursor A and Cursor B to be linked.

0|OFF: sets Cursor A and Cursor B not to be linked.

Return Format

The query returns 1 or 0.

Example

```
:MEASure:CREGion:CABX ON  /*Sets Cursor A and Cursor B to be
linked.*/
:MEASure:CREGion:CABX  /*The query returns 1.*/
```

3.17.24 :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.*/
```

3.17.25 :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.*/
```

3.17.26 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4.

Example

```
:MEASure:COUNter:SOURce CHANnel4 /*Sets the measurement source of the frequency counter to CHANnel4.*/
:MEASure:COUNter:SOURce? /*The query returns CHAN4.*/
```

3.17.27 :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 query returns the measurement results of the frequency counter in scientific notation.

Example

```
:MEASure:COUNter:VALue?   /*The query returns 9.999996E-04.*/
```

3.17.28 :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 AUTO or MAN.

Example

```
:MEASure:AMP:TYPE MANual  /*Sets the amplitude method to MANual.*/
:MEASure:AMP:TYPE?  /*The query returns MAN.*/
```

3.17.29 :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 HIST or MAXM.

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 MAXM.*/
```

3.17.30 :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 HIST or MAXM.

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 MAXM.*/
```

3.17.31 :MEASure:HISTogram:ENABle

Syntax

:MEASure:HISTogram:ENABle <*bool*>

:MEASure:HISTogram:ENABle?

Description

Sets or queries whether to enable the measurement histogram function.

Parameter

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

Remarks

1|ON: enables the measurement histogram.

0|OFF: disables the measurement histogram.

Return Format

The query returns 1 or 0.

Example

```
:MEASure:HISTogram:ENABle 1   /*enables the measurement histogram.*/
:MEASure:HISTogram:ENABle?   /*The query returns 1.*/
```

3.17.32 :MEASure:HISTogram:STATistics:RESult?

Syntax

:MEASure:HISTogram:STATistics:RESult?

Description

Queries the statistics result of the measurement histogram.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the statistics result of the measurement histogram in strings.

Example

```
:MEASure:HISTogram:STATistics:RESult?  /*The query returns
[["92","1","0","Vpp","0","374hits","38hits","1.949V","1.907V","42.26
mV","1.924V","1.924V","1.924V","191.2uV","6.123mV","19.12mV"]].*/
```

3.17.33 :MEASure:CATegory

Syntax

:MEASure:CATegory <*val*>

:MEASure:CATegory?

Description

Sets or queries the measurement type.

Parameter

Name	Type	Range	Default
<val>	Integer	0 to 2	0

Remarks

0: vertical; 1: horizontal 2: other

Return Format

The query returns an integer ranging from 0 to 2.

Example

```
:MEASure:CATegory 1  /*Sets the measurement category to horizontal
measurement.*/
:MEASure:CATegory?  /*The query returns 1.*/
```

3.18 :QUICk Commands

The **:QUICk** commands are used to set and query the parameters related to quick operations.

3.18.1 :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|RECord| SSAVe}	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.
- **RECord:** indicates waveform recording.
- **SSAVe:** indicates saving group.

Return Format

The query returns SIM, SWAV, SSET, AME, REC, SSAV, or SRES.

Example

```
:QUICk:OPERation SWAVe        /*Sets the type of the shortcut key to
"save waveform".*/
:QUICk:OPERation?              /*The query returns SWAV.*/
```

3.19 :RECord Commands

The **:RECord** commands are used to set and query the parameters related to the waveform recording mode and frames.

Waveform recording/playing function allows you to record and play the waveforms, enabling you to analyze the waveforms better.

3.19.1 :RECord:WRECord:ENABle

Syntax

`:RECord:WRECord:ENABle` <*bool*>

`:RECord:WRECord: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

Remarks

N/A

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.*/
```

3.19.2 :RECord:ENABle

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

Remarks

This command exists for backwards compatibility. Use the command *:RECord:WRECord:ENABle*.

Return Format

The query returns 0 or 1.

Example

```
:RECord:ENABle ON    /*Enables the waveform recording function.*/
:RECord:ENABle?      /*The query returns 1.*/
```

3.19.3 :RECord:WRECord:OPERate

Syntax

:RECord:WRECord:OPERate <*operate*>

:RECord:WRECord:OPERate?

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

Remarks

N/A

Return Format

The query returns RUN or STOP.

Example

```
:RECord:WRECord:OPERate RUN    /*Sets to start recording
waveforms.*/
:RECord:WRECord:OPERate?      /*The query returns RUN.*/
```

3.19.4 :RECord:STARt

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
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command exists for backwards compatibility. Use the command *:RECord:WRECord:OPERate*.

Return Format

The query returns 1 or 0.

Example

```
:RECord:STARt ON    /*Sets to start recording the waveforms.*/
:RECord:STARt?      /*The query returns 1.*/
```

3.19.5 :RECord:WRECord:FRAMes

Syntax

:RECord:WRECord:FRAMes <*value*>

:RECord:WRECord: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	1,000

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.*/
```

3.19.6 :RECord:FRAMes

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	1,000

Remarks

This command exists for backwards compatibility. Use the command *:RECord:WRECord:FRAMes*.

Return Format

The query returns an integer ranging from 1 to the maximum number of frames that can be recorded currently.

Example

```
:RECord:FRAMes 300     /*Sets the number of recorded frames to 300.*/
:RECord:FRAMes?        /*The query returns 300.*/
```

3.19.7 :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.*/
```

3.19.8 :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
```

3.19.9 :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. The unit is s.

Example

```
:RECord:WRECord:FINTerval 1 /*Sets the time interval between frames
in waveform recording to 1 s.*/
:RECord:WRECord:FINTerval? /*The query returns 1.000000E0.*/
```

3.19.10 :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.*/
```

3.19.11 :RECord:WREPlay:FCURrent

Syntax

:RECord:WREPlay:FCURrent <*value*>

:RECord:WREPlay:FCURrent?

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.*/
```

3.19.12 :RECord:CURRent

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

This command exists for backwards compatibility. Use the command *:RECord:WREPlay:FCURrent*.

Return Format

The query returns an integer.

Example

```
:RECord:CURRent 300    /*Sets the current frame for waveform
playing to 300.*/
:RECord:CURRent?      /*The query returns 300.*/
```

3.19.13 :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
```

3.19.14 :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.*/
```

3.19.15 :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.*/
```

3.19.16 :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
```

3.19.17 :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 1 s	100 ms

Remarks

N/A

Return Format

The query returns the time interval in scientific notation. The unit is s.

Example

```
:RECord WREPlay:FINTerval 1 /*Sets the time interval between frames
in waveform playback to 1 s.*/
:RECord:WREPlay:FINTerval? /*The query returns 1.000000E0.*/
```

3.19.18 :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 REP or SING.

Example

```
:RECord:WREPlay:MODE REP    /*Sets the playback mode to REP.*/
:RECord:WREPlay:MODE?      /*The query returns REP.*/
```

3.19.19 :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 FORW or BACK.

Example

```
:RECord:WREPlay:DIRection BACK  /*Sets the direction of waveform
playback to BACK.*/
:RECord:WREPlay:DIRection?  /*The query returns BACK.*/
```

3.19.20 :RECord:WREPlay:OPERate

Syntax

:RECord:WREPlay:OPERate <*operate*>

:RECord:WREPlay:OPERate?

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

Remarks

N/A

Return Format

The query returns RUN or STOP.

Example

```
:RECord:WREPlay:OPERate RUN    /*Sets to play the waveforms.*/
:RECord:WREPlay:OPERate?      /*The query returns RUN.*/
```

3.19.21 :RECord:PLAY

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
<bool>	Bool	{{1|ON}|{0|OFF}}	0|OFF

Remarks

This command exists for backwards compatibility. Use the command *:RECord:WREPlay:OPERate*.

Return Format

The query returns 1 or 0.

Example

```
:RECord:PLAY ON    /*Sets to play the waveforms.*/
:RECord:PLAY?      /*The query returns 1.*/
```

3.19.22 :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.*/
```

3.19.23 :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.*/
```

3.19.24 :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

N/A

Example

```
:RECord:WREPlay: PLAY FEND /*Sets to play from end frame manually.*/
```

3.20 :REFerence Commands

:REFerence commands are used to set the reference waveform parameters.

This series oscilloscope provides 10 reference waveform positions (Ref1-Ref10). In the actual test process, you can compare the signal waveform with the reference waveform to locate the failure.

3.20.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4|MATH1|MATH2|MATH3|MATH4}	CHANnel1

Remarks

Digital channels (D0 to D15) are available to choose when the logic probe is connected.

Only the currently enabled channel can be selected as the source of the specified reference channel.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4, MATH1, MATH2, MATH3, or MATH4.

Example

```
:REFerence:SOURce 1,CHANnel1          /*Sets the source of Ref1 to
CHANnel1.*/
:REFerence:SOURce? 1                  /*The query returns CHAN1.*/
```

3.20.2 :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	Refer to *Remarks*	

Remarks

The range of the parameter is related to the probe ratio and input impedancesettings.

When the probe ratio is 1X and the input impedance is 1 MΩ, the value of <scale> ranges from 1 mV to 10 V. When the probe ratio is 1X and the input impedance is 50 Ω, the value of <scale> ranges from 200 μV to 10 V. When the probe ratio is 10X and the input impedance is 1 MΩ, the value of <scale> ranges from 10 mV to 100 V. When the probe ratio is 10X and the input impedance is 50 Ω, the value of <scale> ranges from 2 mV to 100 V.

This command is only available when you have saved the waveforms of the specified reference channel.

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.*/
```

3.20.3 :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}	-
<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.*/
```

3.20.4 :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
```

3.20.5 :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
```

3.20.6 :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
```

3.20.7 :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.*/
```

3.20.8 :REFerence:LABel:ENABle

Syntax

:REFerence:LABel:ENABle <*bool*>

:REFerence:LABel:ENABle?

Description

Enables or disables the label display of all the reference channels; or queries the on/off label display status of all the reference channels.

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 label display of
all the reference channels.*/
:REFerence:LABel:ENABle?            /*The query returns 1.*/
```

3.20.9 :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.*/
```

3.21 :SAVE Commands

You can save the current setups, waveforms, screen image, and parameters of the oscilloscope to the internal memory or external USB storage device (such as USB storage device) in various formats and recall the stored files when necessary.

3.21.1 :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.*/
```

3.21.2 :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.*/
```

3.21.3 :SAVE:IMAGe:FORMat

Syntax

:SAVE:IMAGe:FORMat <*format*>

:SAVE:IMAGe:FORMat?

Description

Sets or queries in what format is the image saved.

Parameter

Name	Type	Range	Default
<format>	Discrete	{PNG|BMP|JPG}	-

Remarks

N/A

Return Format

The query returns PNG, BMP, or JPG.

Example

```
:SAVE:IMAGe:FORMat PNG       /*Sets the storage format to PNG.*/
:SAVE:IMAGe:FORMat?              /*The query returns PNG.*/
```

3.21.4 :SAVE:IMAGe:HEADer

Syntax

:SAVE:IMAGe:HEADer <*bool*>

:SAVE:IMAGe:HEADer?

Description

Sets or queries whether to display the image header.

Parameter

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

Remarks

- **1|ON:** enables to display the image header. If you select "ON", the instrument model and the image creation date will be displayed in the header of the image when you save the image file.
- **0|OFF:** disables to display the image header.

Return Format

The query returns 1 or 0.

Example

```
:SAVE:IMAGe:HEADer ON /*Enables the display of the image header.*/
:SAVE:IMAGe:HEADer? /*The query returns 1.*/
```

3.21.5 :SAVE:IMAGe:DATA?

Syntax

:SAVE:IMAGe:DATA?

Description

Queries the bitmap data stream of the currently displayed image.

Parameter

N/A

Remarks

The read data format is TMC header + binary data stream of the screenshot + terminator. The TMC header is in #NXXXXXX format; wherein, # is the TMC header identifier; N following # represents the number of digits (in the decimal integer) that follow; the length of the binary data stream of the screenshot is expressed in ASCII strings, and the terminator represents the ending of communication. For example, the data read for one time is #9000387356. 9 indicates the number of digits (in the decimal integer) that follow, and "000387356" indicates the length of the binary data stream, that is, 387356 bytes.

Return Format

The query returns the binary data stream.

Example

```
:SAVE:IMAGe:DATA? /*The query returns the binary data stream.*/
```

3.21.6 :SAVE:PATHname

Syntax

:SAVE:PATHname <*name*>

:SAVE:PATHname?

Description

Sets or queries the file saving path.

Parameter

Name	Type	Range	Default
<name>	ASCII String	-	-

Remarks

N/A

Return Format

The query returns the file saving path in strings.

Example

```
:SAVE:PATHname C:/new /*Sets the file saving path to C:/new.*/
:SAVE:PATHname? /*The query returns C:/new.*/
```

3.21.7 :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 local path is C:/; and the path of the external storage device is D:/.
- The suffix of the file supported is "*.bmp", "*.png", or "*.jpg".
- Refer to *:SAVE:OVERlap* . When you send the command :SAVE:OVERlap ON:

  If the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The filename can contain letters, numbers, and other non-Chinese characters.

  The length of the filename shall not exceed 26 characters.

Return Format

N/A

Example

```
:SAVE:IMAGe D:/123.png    /*Saves the oscilloscope screenshot image
to the external memory Disk D, with the filename "*123.png".*/
```

3.21.8 :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

<path> includes the file storage location and the filename with a suffix.

- The path of the local disk is C:/ and the path of the external storage device can be D:/
- The suffix of the filename is "*.stp".

- Refer to *:SAVE:OVERlap* . When you send the command :SAVE:OVERlap ON, if the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The filename can contain letters, numbers, and other non-Chinese characters. The length of the filename shall not exceed 26 characters.

Return Format

N/A

Example

```
:SAVE:SETup D:/123.stp /*Stores the current setup parameters of the
oscilloscope into the external storage device Disk D, with the
filename 123.stp.*/
```

3.21.9 :SAVE:WAVeform

Syntax

:SAVE:WAVeform <*path*>

Description

Saves the screen waveform data to the internal or external memory as a file.

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 path of the local disk is C:/ and the path of the external storage device can be D:/.
- The suffix of the filename is "*.bin" or "*.csv".
- Refer to *:SAVE:OVERlap* . When you send the command :SAVE:OVERlap ON, if the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The filename can contain letters, numbers, and other non-Chinese characters. The length of the filename shall not exceed 26 characters.

Return Format

N/A

Example

```
:SAVE:WAVeform D:/123.csv   /*Saves the screen waveform file to the
external storage device Disk D, with the filename 123.csv.*/
```

3.21.10 :SAVE:MASK

Syntax

:SAVE:MASK <*path*>

Description

Saves the mask file to 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.

- The local path is C:/; and the external storage path is D:/.
- The suffix of the filename is "*.pf".
- Refer to *:SAVE:OVERlap* . When you have performed the :SAVE:OVERlap ON command, if the file to be saved has the same filename as the existing file in the specified storage path, then the original file will be overwritten by the newly saved file.
- The filename can contain letters, numbers, and other non-Chinese characters. The length of the filename shall not exceed 26 characters.

You can run the *:LOAD:MASK* command to load the mask file.

Return Format

N/A

Example

```
:SAVE:MASK D:/123.pf  /*Saves the mask file to the external storage
device Disk D, with the filename 123.pf.*/
```

3.21.11 :SAVE:MEMory:WAVeform

Syntax

:SAVE:MEMory:WAVeform <*path*>

Description

Saves the memory waveform data of the oscilloscope to the specified path as a file.

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 path of the local disk is C:/; and the path of the external storage device can be D:/.
- The suffix of the filename is "*.bin", "*.csv", or "*.wfm".
- Refer to *:SAVE:OVERlap* . When you send the command :SAVE:OVERlap ON, if the specified storage location already contains a file with the same filename, the original file will be overwritten.
- The filename can contain letters, numbers, and other non-Chinese characters. The length of the filename shall not exceed 26 characters.

Return Format

N/A

Example

```
:SAVE:MEMory:WAVeform D:/123.bin /*Saves the oscilloscope memory
waveform data to the external storage device Disk D, with the
filename "*123.bin".*/
```

3.21.12 :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
```

3.21.13 :SAVE:OVERlap

Syntax

:SAVE:OVERlap <*bool*>

:SAVE:OVERlap?

Description

Sets or queries whether to enable or disable the file overwriting function.

Parameter

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

Remarks

- **1|ON:** enables to overwrite the existing file. The existing file with the same name as the file to be saved will be overwritten.
- **0|OFF:** disables to overwrite the existing file.

Return Format

The query returns 1 or 0.

Example

```
:SAVE:OVERlap ON /*Enables to overwrite the existing file.*/
:SAVE:OVERlap? /*The query returns 1.*/
```

3.21.14 :SAVE:PREFix

Syntax

:SAVE:PREFix <*name*>

:SAVE:PREFix?

Description

Sets or queries the prefix of the filename when saving a file.

Parameter

Name	Type	Range	Default
<name>	ASCII String	*Remarks*	-

Remarks

Each filename consists of the given prefix, appended with an appropriate extension.

The prefix of the filename mentioned in this command shall not contain the suffix. Its prefix can contain letters, numbers, and other non-Chinese characters. The length of the prefix shall not exceed 16 characters.

Return Format

The query returns the prefix of the filename in strings.

Example

```
:SAVE:PREFix Rigol /*Sets the prefix of the filename to Rigol.*/
:SAVE:PREFix? /*The query returns Rigol.*/
```

3.21.15 :SAVe:SMB:SERVerpath

Syntax

:SAVe:SMB:SERVerpath <*path*>

:SAVe:SMB:SERVerpath?

Description

Sets or queries the server path for SMB file sharing.

Parameter

Name	Type	Range	Default
<path>	ASCII String	-	-

Remarks

The format of the server path is "\\xxx.xxx.xxx.xxx\name". Wherein, "xxx.xxx.xxx.xxx" indicates the IP address of the computer; "name" indicates the name of the SMB shared folder. Note that "name" is only allowed to be named in English letters.

Return Format

The query returns the server path of the SMB file sharing in strings.

Example

```
:SAVe:SMB:SERVerpath \\172.16.25.77\Share    /*Sets the server path
of the SMB file sharing to \\172.16.25.77\Share.*/
:SAVe:SMB:SERVerpath?  /*The query returns \\172.16.25.77\Share.*/
```

3.21.16 :SAVe:SMB:USERname

Syntax

:SAVe:SMB:USERname <*name*>

:SAVe:SMB:USERname?

Description

Sets or queries the user name for SMB file sharing.

Parameter

Name	Type	Range	Default
<name>	ASCII String	-	-

Remarks

The user name shall not contain Chinese characters.

Return Format

The query returns the user name for the SMB file sharing in strings.

Example

```
:SAVe:SMB:USERname Rigol /*Sets the user name for SMB file sharing
to Rigol.*/
:SAVe:SMB:USERname? /*The query returns Rigol.*/
```

3.21.17 :SAVe:SMB:PASSword

Syntax

:SAVe:SMB:PASSword <*password*>

:SAVe:SMB:PASSword?

Description

Sets or queries the password of the user name for SMB file sharing.

Parameter

Name	Type	Range	Default
<password>	ASCII String	-	-

Remarks

The password shall not contain Chinese characters.

Return Format

The query returns the password of the user name for SMB file sharing.

Example

```
:SAVe:SMB:PASSword Rigol /*Sets the password of the user name for
SMB file sharing to Rigol*/
:SAVe:SMB:PASSword? /*The query returns Rigol.*/
```

3.21.18 :SAVe:SMB:AUToconnect

Syntax

:SAVe:SMB:AUToconnect <*bool*>

:SAVe:SMB:AUToconnect?

Description

Sets or queries whether to enable or disable auto connection of SMB file sharing at bootup.

Parameter

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

Remarks

1: enables auto-connecting SMB file sharing at bootup. 0: disables auto-connecting SMB file sharing at bootup.

Return Format

The query returns 0 or 1.

Example

```
:SAVe:SMB:AUToconnect ON /*Enables auto-connecting SMB file sharing
at bootup.*/
:SAVe:SMB:AUToconnect? /*The query returns 1.*/
```

3.21.19 :SAVe:SMB:CONNect

Syntax

:SAVe:SMB:CONNect

Description

Configures to connect the SMB file sharing.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.21.20 :SAVe:SMB:DISConnect

Syntax

```
:SAVe:SMB:DISConnect
```

Description

Configures to disconnect the SMB file sharing.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.21.21 :SAVe:SMB:CONState?

Syntax

```
:SAVe:SMB:CONState?
```

Description

Queries the connection status of the SMB file sharing.

Parameter

N/A

Remarks

1: indicates connected; 0: indicates disconnected.

Return Format

The query returns 0 or 1.

Example

```
N/A
```

3.21.22 :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.

- The path of the local disk is C:/ and the path of the external storage device can be D:/.
- The suffix of the filename to be loaded is "*.stp".

Return Format

N/A

Example

```
:LOAD:SETup D:/123.stp      /*Loads the setup file "123.stp" from
the external storage device Disk D.*/
```

3.21.23 :LOAD:MASK

Syntax

:LOAD:MASK <*path*>

Description

Loads the mask file 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.

- The local path is C:/; and the external storage path is D:/ or E:/.
- The suffix of the filename is "*.pf".

Return Format

N/A

Example

```
:LOAD:MASK D:/123.pf  /Loads the mask file named "123.pf" from the
external storage device Disk D.*/
```

3.22 :SEARch Commands

3.22.1 :SEARch:COUNt?

Syntax

`:SEARch:COUNt?`

Description

Queries the total number of the search events.

Parameter

N/A

Remarks

N/A

Return Format

The query returns the total number of the search events in integer.

Example

```
N/A
```

3.22.2 :SEARch:STATe

Syntax

:SEARch:STATe <*bool*>

:SEARch:STATe?

Description

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

Parameter

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

Return Format

The query returns 1 or 0.

Example

```
:SEARch:STATe ON    /*Enables the search function.*/
:SEARch:STATe?      /*The query returns 1.*/
```

3.22.3 :SEARch:MODE

Syntax

:SEARch:MODE <*value*>

:SEARch:MODE?

Description

Sets the search type.

Parameter

Name	Type	Range	Default
<value>	Discrete	{EDGE|PULSe}	EDGE

Remarks

- **EDGe:** selects "Edge" as the search type.
- **PULSe:** selects "Pulse" as the search type.

Return Format

The query returns EDGE or PULS.

Example

```
:SEARch:MODE PULSe     /*Selects the search type to "PULSe".*/
:SEARch:MODE?          /*The query returns PULS.*/
```

3.22.4 :SEARch:EVENt

Syntax

:SEARch:EVENt <*value*>

:SEARch:EVENt?

Description

Sets to navigate a search event.

Parameter

Name	Type	Range	Default
<value>	Integer	0 to (the number of searched events	0

Remarks

If no search event is found in the current search event table, the query returns 0.

If the parameter <value> is set to 0, then the instrument navigates to the event with the index number 1.

Return Format

The query returns an integer.

Example

```
:SEARch:EVENt 1           /*Sets to navigate to Search Event 1.*/
:SEARch:EVENt?            /*The query returns 1.*/
```

3.22.5 :SEARch:VALue?

Syntax

:SEARch:VALue? <*x*>

Description

Queries the time value for the specified Line No. in the marktable.

Parameter

Name	Type	Range	Default
<x>	Integer	-	-

Remarks

<x> denotes the line number of the marktable.

Return Format

The query returns a time value.

Example

```
N/A
```

3.22.6 :SEARch:EDGE:SLOPe

Syntax

:SEARch:EDGE:SLOPe <*slope*>

:SEARch:EDGE:SLOPe?

Description

Sets or queries the edge for the "Edge" search type.

Parameter

Name	Type	Range	Default
<slope>	Discrete	{POSitive|NEGative|EITHer}	POSitive

Remarks

- **POSitive:** indicates the rising edge.
- **NEGative:** indicates the falling edge.
- **EITHer:** indicates the rising edge or the falling edge.

Return Format

The query returns POS, NEG, or EITH.

Example

```
:SEARch:EDGE:SLOPe NEGative     /*Sets the edge type to NEGative.*/
:SEARch:EDGE:SLOPe?             /*The query returns NEG.*/
```

3.22.7 :SEARch:EDGE:SOURce

Syntax

:SEARch:EDGE:SOURce <*source*>

:SEARch:EDGE:SOURce?

Description

Sets or queries the source for the "Edge" search type.

Parameter

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

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4.

Example

```
:SEARch:EDGE:SOURce CHANnel1     /*Sets the source for the Edge
search type to CHANnel1.*/
:SEARch:EDGE:SOURce?             /*The query returns CHAN1.*/
```

3.22.8 :SEARch:EDGE:THReshold

Syntax

:SEARch:EDGE:THReshold <*thre*>

:SEARch:EDGE:THReshold?

Description

Sets or queries the threshold for the "Edge" search type.

Parameter

Name	Type	Range	Default
<thre>	Real	(-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset)	0.000V

Remarks

N/A

Return Format

The query returns the threshold for the edge search type in scientific notation.

Example

```
:SEARch:EDGE:THReshold 0.01          /*Sets the threshold to 0.01
V.*/
:SEARch:EDGE:THReshold?            /*The query returns 1.000000E-2.*/
```

3.22.9 :SEARch:PULSe:POLarity

Syntax

:SEARch:PULSe:POLarity <*polarity*>

:SEARch:PULSe:POLarity?

Description

Sets or queries the polarity for the "Pulse" search type.

Parameter

Name	Type	Range	Default
<polarity>	Discrete	{POSitive|NEGative}	POSitive

Remarks

N/A

Return Format

The query returns POS or NEG.

Example

```
:SEARch:PULSe:POLarity POSitive    /*Sets the polarity for the
Pulse search type to POSitive.*/
:SEARch:PULSe:POLarity?           /*The query returns POS.*/
```

3.22.10 :SEARch:PULSe:QUALifier

Syntax

:SEARch:PULSe:QUALifier <*qualifier*>

:SEARch:PULSe:QUALifier?

Description

Sets or queries the search condition for the "Pulse" search type.

Parameter

Name	Type	Range	Default
<qualifier>	Discrete	{GREater|LESS|GLESs}	GREater

Remarks

- **GREater:** the positive/negative pulse width of the input signal is greater than the specified pulse width.

- **LESS:** the positive/negative pulse width of the input signal is smaller than the specified pulse width.
- **GLESs:** the positive/negative pulse width of the input signal is greater 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

```
:SEARch:PULSe:QUALifier LESS    /*Sets the search condition for the
"Pulse" search type to LESS.*/
:SEARch:PULSe:QUALifier?           /*The query returns LESS.*/
```

3.22.11 :SEARch:PULSe:SOURce

Syntax

:SEARch:PULSe:SOURce <*source*>

:SEARch:PULSe:SOURce?

Description

Sets or queries the source for the "Pulse" search type.

Parameter

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

Remarks

N/A

Return Format

The query returns CHAN1, CHAN2, CHAN3, CHAN4.

Example

```
:SEARch:PULSe:SOURce CHANnel1    /*Sets the source for the Pulse
search type to CHANnel1.*/
:SEARch:PULSe:SOURce?            /*The query returns CHAN1.*/
```

3.22.12 :SEARch:PULSe:UWIDth

Syntax

:SEARch:PULSe:UWIDth <*width*>

:SEARch:PULSe:UWIDth?

Description

Sets or queries the upper limit of the pulse width for the "Pulse" search type.

Parameter

Name	Type	Range	Default
<width>	Real	800 ps to 10 s	2 us

Remarks

N/A

Return Format

The query returns the upper limit of the pulse width in scientific notation.

Example

```
:SEARch:PULSe:UWIDth 1   /*Sets the upper limit of the pulse width
for the Pulse search type to 1 s.*/
:SEARch:PULSe:UWIDth?           /*The query returns 1.000000E0.*/
```

3.22.13 :SEARch:PULSe:LWIDth

Syntax

:SEARch:PULSe:LWIDth <*width*>

:SEARch:PULSe:LWIDth?

Description

Sets or queries the lower limit of the pulse width for the "Pulse" search type.

Parameter

Name	Type	Range	Default
<width>	Real	800 ps to 10 s	1 us

Remarks

N/A

Return Format

The query returns the lower limit of the pulse width in scientific notation.

Example

```
:SEARch:PULSe:LWIDth 0.2   /*Sets the lower limit of the pulse
width for the Pulse search type to 200 ms.*/
:SEARch:PULSe:LWIDth?           /*The query returns 2.000000E-1.*/
```

3.22.14 :SEARch:PULSe:THReshold

Syntax

:SEARch:PULSe:THReshold <*thre*>

:SEARch:PULSe:THReshold?

Description

Sets or queries the threshold for the "Pulse" search type.

Parameter

Name	Type	Range	Default
<thre>	Real	(-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset)	0.000 V

Remarks

N/A

Return Format

The query returns the threshold for the pulse search type in scientific notation.

Example

```
:SEARch:PULSe:THReshold 0.01          /*Sets the threshold to 10
mV.*/
:SEARch:PULSe:THReshold?           /*The query returns
1.000000E-2.*/
```

3.23 :NAVigate Commands

3.23.1 :NAVigate:ENABle

Syntax

:NAVigate:ENABle <*bool*>

:NAVigate:ENABle?

Description

Sets or queries the on/off status of the Navigation 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

```
:NAVigate:ENABle ON     /*Enables the Navigation function.*/
:NAVigate:ENABle?       /*The query returns 1.*/
```

3.23.2 :NAVigate:MODE

Syntax

:NAVigate:MODE <*mode*>

:NAVigate:MODE?

Description

Sets or queries the navigation mode.

Parameter

Name	Type	Range	Default
<mode>	Discrete	{TIME|SEARch}	TIME

Remarks

- **TIME:** indicates the time navigation. It is available only when in "YT" time mode.
- **SEARch:** indicates event search navigation. When you enable the navigation function and complete the event search, you can use the navigation combination keys to quickly navigate the specific event in the event mark table.

The navigation function is available only when the oscilloscope is in "STOP" state (acquisition stopped). You can send the *:STOP* command to set the oscilloscope to STOP status.

Return Format

The query returns TIME or SEAR.

Example

```
:NAVigate:MODE TIME /*Sets the navigation mode to Time navigation.*/
:NAVigate:MODE? /*The query returns TIME.*/
```

3.23.3 :NAVigate:TIME:SPEed

Syntax

:NAVigate:TIME:SPEed <*speed*>

:NAVigate:TIME:SPEed?

Description

Sets the speed of playing the waveforms in time navigation mode.

Parameter

Name	Type	Range	Default
<speed>	Discrete	{HIGH|NORMal|LOW}	NORMal

Remarks

- **HIGH:** indicates playing the waveforms at a high speed.
- **NORMal:** indicates playing the waveforms at a normal speed.
- **LOW:** indicates playing the waveforms at a low speed.

Return Format

The query returns HIGH, NORMal, or LOW.

Example

```
:NAVigate:TIME:SPEed LOW /*Sets to play the waveforms in time
navigation mode at a low speed.*/
:NAVigate:TIME:SPEed? /*The query returns LOW.*/
```

3.23.4 :NAVigate:TIME:PLAY

Syntax

:NAVigate:TIME:PLAY <*bool*>

:NAVigate:TIME:PLAY?

Description

Sets or queries whether to play the waveforms in time navigation mode.

Parameter

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

Remarks

- **1|ON:** starts playing the waveforms.
- **0|OFF:** stops playing the waveforms.

Return Format

The query returns 0 or 1.

Example

```
:NAVigate:TIME:PLAY ON /*Sets to play the waveforms in time
navigation.*/
:NAVigate:TIME:PLAY? /*The query returns 1.*/
```

3.23.5 :NAVigate:TIME:END

Syntax

`:NAVigate:TIME:END`

Description

Sets to navigate to the end waveforms (waveforms ending at the rightmost end of screen) in time navigation mode.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.6 :NAVigate:TIME:STARt

Syntax

`:NAVigate:TIME:STARt`

Description

Sets to navigate to the start waveform (waveform starting at the leftmost end of screen) in time navigation mode.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.7 :NAVigate:TIME:NEXT

Syntax

`:NAVigate:TIME:NEXT`

Description

Sets to play forward the waveform in time navigation mode.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.8 :NAVigate:TIME:BACK

Syntax

`:NAVigate:TIME:BACK`

Description

Sets to play backward the waveform in time navigation mode.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.9 :NAVigate:SEARch:END

Syntax

`:NAVigate:SEARch:END`

Description

Sets to navigate to the last event.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.10 :NAVigate:SEARch:STARt

Syntax

`:NAVigate:SEARch:STARt`

Description

Sets to navigate to the first event.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.11 :NAVigate:SEARch:NEXT

Syntax

:NAVigate:SEARch:NEXT

Description

Sets to navigate to the next event.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.23.12 :NAVigate:SEARch:BACK

Syntax

:NAVigate:SEARch:BACK

Description

Sets to navigate to the previous event.

Parameter

N/A

Remarks

N/A

Return Format

N/A

Example

```
N/A
```

3.24 :SYSTem Commands

The **:SYSTem** commands are used to set the system-related parameters.

3.24.1 :SYSTem:AOUTput

Syntax

:SYSTem:AOUTput <*auxoutput*>

:SYSTem:AOUTput?

Description

Sets or queries the type of the signal output from the rear-panel **[AUX OUT]** connector.

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 connector.
- **PFAil:** after you select this type, a pulse signal will be output from the 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.*/
```

3.24.2 :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.*/
```

3.24.3 :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.*/
```

3.24.4 :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.*/
```

3.24.5 :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.*/
```

3.24.6 :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.*/
```

3.24.7 :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
```

3.24.8 :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.

Example

```
N/A
```

3.24.9 :SYSTem:DGSTatus?

Syntax

`:SYSTem:DGSTatus?`

Description

Queries whether the DG module exists.

Parameter

N/A

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:SYSTem:DGSTatus? /*The query returns 1 if the instrument has
installed the DG module; otherwise, it returns 0.*/
```

3.24.10 :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.*/
```

3.24.11 :SYSTem:PSTatus

Syntax

:SYSTem:PSTatus <*sat*>

:SYSTem:PSTatus?

Description

Sets or queries the power status of the instrument.

Parameter

Name	Type	Range	Default
<sat>	Discrete	{DEFault|OPEN}	OPEN

Remarks

- **DEFault:** after the instrument is powered on, you need to press the Power key on the front panel to start the instrument.
- **OPEN:** after the instrument 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.*/
```

3.24.12 :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
```

3.24.13 :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 */
```

3.24.14 :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.*/
```

3.24.15 :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 available, and 0 indicates not available.

Example

```
N/A
```

3.24.16 :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 **SERVICE CENTRE (or Products & Services) > License Activation** to enter the license activation interface.
  - Input the correct key, serial number (To obtain the serial number, click or tap the function navigation icon at the lower-left corner of the screen first, then click or tap **Utility**. Click or tap **About** to acquire the serial number of the instrument.), verification code. Click **Generate** to acquire the download link for 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> format is "<series>-<opt>@<code>".
  - **<series>:** indicates the product series.
  - **<opt>:** indicates the option name. The options supported include:
    - **BND:** function and application bundle option, including AFG100, AUDIOA, AUTOA, FlexA, and AEROA.
    - **AFG100:** built-in dual-channel 100 MH function signal generator option, with Bode plot available.
    - **AFG50:** built-in dual-channel 50 MH function signal generator option, with Bode plot available.
    - **AUDIOA:** audio serial bus trigger and analysis option.
    - **AUTOA:** CAN serial bus analysis with flexible data rate.
    - **FlexA:** FlexRay serial bus trigger and analysis option.
    - **AEROA:** MIL-STD-1553 bus trigger and analysis option.
    - **RLU05:** 500 Mpts memory depth option.
    - **BWU03T05:** 350 MHz to 500 MHz bandwidth upgrade option.
    - **BWU03T08:** 350 MHz to 800 MHz bandwidth upgrade option.

- **BWU05T08:** 500 MHz to 800 MHz bandwidth upgrade option.

- **<code>:** The license is a fixed length of strings. Each instrument has a unique license.

- To query whether the option has been successfully installed, refer to *:SYSTem:OPTion:STATus?*.

Return Format

N/A

Example

```
:SYSTem:OPTion:INSTall MHO900-
AUDIOA@5422a844a7335736b308e2a3fb46f93597dd8884fa12f74f89c5774bbd44f
efe308d25141d058baac4996670f6f177e3
```

3.24.17 :SYSTem:OPTion:UNINstall

Syntax

`:SYSTem:OPTion:UNINstall`

Description

Uninstalls all the official options.

Parameter

N/A

Remarks

After the option has been uninstalled, you need to restart the instrument.

Return Format

N/A

Example

```
N/A
```

3.24.18 :SYSTem:OPTion:STATus?

Syntax

`:SYSTem:OPTion:STATus?` <*type*>

Description

Queries whether an option is activated or not.

Parameter

Name	Type	Range	Default
<type>	Discrete	{BND|AFG100|AFG50|AUDio|CAN-FD|FLEX|AERO|RLU-05|BWU03T05|BWU03T08|BWU05T08}	-

Remarks

- **BND:** function and application bundle option, including AFG100, AUDIOA, AUTOA, FlexA, and AEROA.
- **AFG100:** built-in dual-channel 100 MH Function/Arbitrary Waveform Generator option, with Bode plot supported.
- **AFG50:** built-in dual-channel 50 MH Function/Arbitrary Waveform Generator option, with Bode plot supported.
- **AUDio:** audio serial bus trigger and analysis (I2S) option.
- **CAN-FD:** CAN serial bus analysis with flexible data rate.
- **FLEX:** FlexRay serial bus trigger and analysis option.
- **AERO:** MIL-STD-1553 bus trigger and analysis option.
- **RLU-05:** 500 Mpts memory depth option.
- **BWU03T05:** 350 MHz-500 MHz bandwidth upgrade option.
- **BWU03T08:** 350 MHz-800 MHz bandwidth upgrade option.
- **BWU05T08:** 500 MHz-800 MHz bandwidth upgrade option.

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

```
N/A
```

3.24.19 :SYSTem:OPTion:VALid?

Syntax

`:SYSTem:OPTion:VALid?` <*type*>

Description

Queries whether an option is activated or not.

This command exists for backwards compatibility. Use the command *:SYSTem:OPTion:STATus?*.

Parameter

Name	Type	Range	Default
<type>	Discrete	{BND|AFG100|AFG50|AUDio|CAN-FD|FLEX|AERO|RLU-05|BWU03T05|BWU03T08|BWU05T08}	-

Remarks

- **BND:** function and application bundle option, including AFG100, AUDIOA, AUTOA, FlexA, and AEROA.
- **AFG100:** built-in dual-channel 100 MH Function/Arbitrary Waveform Generator option, with Bode plot supported.
- **AFG50:** built-in dual-channel 50 MH Function/Arbitrary Waveform Generator option, with Bode plot supported.
- **AUDio:** audio serial bus trigger and analysis (I2S) option.
- **CAN-FD:** CAN serial bus analysis with flexible data rate.
- **FLEX:** FlexRay serial bus trigger and analysis option.
- **AERO:** MIL-STD-1553 bus trigger and analysis option.
- **RLU-05:** 500 Mpts memory depth option.
- **BWU03T05:** 350 MHz-500 MHz bandwidth upgrade option.
- **BWU03T08:** 350 MHz-800 MHz bandwidth upgrade option.
- **BWU05T08:** 500 MHz-800 MHz bandwidth upgrade option.

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

```
N/A
```

3.24.20 :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>	Binary	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
```

3.24.21 :SYSTem:ERRor[:NEXT]?

Syntax

:SYSTem:ERRor[:NEXT]?

Description

Queries and clears the error queue 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
```

3.24.22 :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** key.

- After the function of the **AUTO** key is disabled, you cannot perform the Auto Scale operation.

Return Format

The query returns 1 or 0.

Example

```
:SYSTem:AUToscale ON    /*Enables the function of the front-panel
AUTO key.*/
:SYSTem:AUToscale?      /*The query returns 1.*/
```

3.24.23 :SYSTem:KEYBoard:CHECk?

Syntax

`:SYSTem:KEYBoard:CHECk?`

Description

Queries whether the front panel keys and knobs work normally.

Parameter

N/A

Remarks

Before running this command, perform the key test first to check whether some of the front panel keys or knobs work normally. Otherwise, no results will be returned after running this command.

To perform the key test, click or tap **Utility** > **Self Check** > **Key Test** to enter the key test interface. Then press the front panel keys or rotate the knobs on the front panel to test whether they work normally. When you press down the keys on the front panel of the instrument, the keys that work normally will be highlighted in the key test interface and the keys that do not make any response may be damaged.

The key test results are returned in strings, with menu key name following true or false, separated by a colon. For example, "btn_default":true. If true, it indicates that the menu key works normally. If false, it indicates that the menu key does not work or you haven't perform self-check on this key. If you are testing the knob functionality, after rotating the knob on the front panel, its result is displayed in the form of "knob number:number". Wherein, "knob number" indicates the specified knob on the front panel, the "number" following the "knob number" indicates how many times you rotate the knob clockwise or counterclockwise. If you rotate it one time clockwise, the number following the specified knob turns out to be the existing number plus one. If you rotate it one time counterclockwise, the number following the specified knob turns out to be the existing number minus one. For example, "knob5":3 indicates that knob 5 has been rotated three times clockwise.

Return Format

The query returns the results in strings.

Example

```
:SYSTem:KEYBoard:CHECk?  /*The query returns
{"btn_measure":true,"btn_cursor":true,"btn_analyse":true,"btn_single
":true,"btn_auto":true,"btn_runstop":true,"btn_default":true,"btn_cl
ear":true,"btn_touchlock":true,"btn_quick":true,"btn_trigger":true,"
btn_slope":true,"btn_force":true,"btn_ch1":true,"btn_ch2":true,"btn_
ch3":true,"btn_ch4":true,"btn_la":true,"btn_g1":false,"btn_g2":false
,"btn_math":true,"btn_ref":true,"btn_acquire":true,"btn_zoom":false,
"btn_navigate":true,"btn_search":false,"btn_last":true,"btn_pause":t
rue,"btn_next":false,"btn_knob1":false,"btn_knob2":false,"btn_knob3"
:false,"btn_knob4":false,"btn_knob5":false,"btn_knob6":false,"btn_kn
ob7":false,"knob1":2,"knob2":1,"knob3":2,"knob4":2,"knob5":3,"knob6"
:1,"knob7":1,"btn_GI":true}.*/
```

3.24.24 :SYSTem:LOWPower

Syntax

:SYSTem:LOWPower <*x*>

:SYSTem:LOWPower?

Description

Sets or queries whether the instrument is in the low power mode.

Parameter

Name	Type	Range	Default
<x>	Integer	{1|0}	0

Remarks

1: sets the instrument to be in the low power mode.

0: sets the instrument not to be in the low power mode.

Return Format

The query returns 1 or 0.

Example

```
:SYSTem:LOWPower 1 /*Sets the instrument to be in the low power
mode.*/
:SYSTem:LOWPower? /*The query returns 1.*/
```

3.25 Function/Arbitrary Waveform Generator Commands (Option)

The **:SOURce<n>** commands are used to set parameters related to the built-in function/arbitrary waveform generator. <n> can set to 1 or 2, which indicates the specified built-in function/arbitrary waveform generator channel.

NOTE

The **:SOURce<n>** commands are available to use when the instrument has installed the MHO900-AFG100 or MHO900-AFG50 option.

3.25.1 :SOURce<n>:OUTPut:STATe

Syntax

:SOURce<*n*>**:OUTPut:STATe** <*state*>

:SOURce<*n*>**:OUTPut:STATe?**

Description

Enables or disables the output of the specified function/arbitrary waveform generator channel; queries the on/off status of the output of the specified function/arbitrary waveform generator channel.

Parameter

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

Remarks

N/A

Return Format

The query returns 0 or 1.

Example

```
:SOURce1:OUTPut:STATe ON /*Enables the output of GI.*/
:SOURce1:OUTPut:STATe? /*The query returns 1.*/
```

3.25.2 :SOURce<n>:FUNCtion

Syntax

:SOURce<*n*>**:FUNCtion** <*wave*>

:SOURce<*n*>:FUNCtion?

Description

Sets or queries the type of the basic waveform output from the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<wave>	Discrete	{SINusoid|SQUare|RAMP|NOISe|DC|ARB|EXPRise|EXPFall|ECG1|GAUSsian|LORentz|HAVersine|SINC}	SINusoid

Remarks

The built-in function/arbitrary waveform generator of this series oscilloscope can output various basic waveforms, including Sine, Square, Ramp, Noise, DC, Arb, EXPRise, EXPFall, ECG1, Gaussian, Lorentz, Haversine, and Sinc.

Return Format

The query returns SIN, SQU, RAMP, NOIS, DC, ARB, EXPR, EXPF, ECG1, GAUS, LOR, HAV, or SINC.

Example

```
:SOURce1:FUNCtion SQUare /*Sets the basic waveform output from GI
to SQUare.*/
:SOURce1:FUNCtion? /*The query returns SQU.*/
```

3.25.3 :SOURce<n>:LOAD:ARBitrary

Syntax

:SOURce<*n*>:LOAD:ARBitrary <*path*>

:SOURce<*n*>:LOAD:ARBitrary?

Description

Loads the arbitrary waveform file from the specified path.

Queries the path that the arbitrary waveform file that has been loaded last time.

Parameter

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

Name	Type	Range	Default
<path>	ASCII String	Refer to *Remarks*	-

Remarks

- <path> includes the file storage location and the filename with a suffix.
- The local path is C:/; and the path of the external storage device is D:/.

Return Format

The query returns the path where to load the file last time in ASCII strings.

Example

```
:SOURce1:LOAD:ARBitrary D:/123.csv   /*Loads the arbitrary waveform
file named "123.csv" from the external storage device Disk D.*/
:SOURce1:LOAD:ARBitrary?  /*The query returns D:/123.csv.*/
```

3.25.4 :SOURce<n>:FREQuency

Syntax

:SOURce<*n*>**:FREQuency** <*freq*>

:SOURce<*n*>**:FREQuency?**

Description

Sets or queries the frequency of the basic waveform.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<freq>	Real	Refer to *Remarks*	1 kHz

Remarks

- Sine: 2 mHz to 100 MHz (with AFG100 option installed); 20 mHz to 50 MHz (with AFG50 option installed)
- Square: 2 mHz to 20 MHz (with AFG100 option installed); 20 mHz to 10 MHz (with AFG100 option installed)
- Ramp: 2 mHz to 2 MHz (with AFG100 option installed); 20 mHz to 1 MHz (with AFG100 option installed)

- Arb: 2 mHz to 20 MHz (with AFG100 installed); 20 mHz to 10 MHz (with AFG50 installed)
- Built-in waveforms: 2 mHz to 20 MHz (with AFG100 option installed); 20 mHz to 10 MHz (with AFG50 option installed)
- DC and Noise: no frequency parameter.

To set or query the type of the basic waveforms, run the *:SOURce<n>:FUNCtion* command.

Return Format

The query returns the frequency value in scientific notation. For example 2.000000E+5.

Example

```
:SOURce1:FREQuency 1000 /*Sets the frequency of the basic waveform
of GI to 1 kHz.*/
:SOURce1:FREQuency? /*The query returns 1.000000E+3.*/
```

3.25.5 :SOURce<n>:PERiod

Syntax

:SOURce<*n*>**:PERiod** <*period*>

:SOURce<*n*>**:PERiod?**

Description

Sets or queries the period of the basic waveform output from the specified function/arbitrary waveform generator channel. The unit is s.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<period>	Real	Refer to *Remarks*	-

Remarks

The period is the reciprocal of the frequency, and vice versa. To set or query the frequency of different waveform types, run the *:SOURce<n>:FREQuency* command.

Return Format

The query returns the period of of the output waveform in scientific notation. The unit is s.

Example

```
:SOURce1:PERiod 1 /*Sets the period of the waveform output from GI
to 1 s.*/
:SOURce1:PERiod? /*The query returns +1.000000000000000E+00.*/
```

3.25.6 :SOURce<n>:PHASe

Syntax

:SOURce<*n*>**:PHASe** <*phase*>

:SOURce<*n*>**:PHASe?**

Description

Sets or queries the phase of the basic waveform output from the specified function/ arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<phase>	Real	0° to 360°	0°

Remarks

N/A

Return Format

The query returns the phase of the waveform in scientific notation.

Example

```
:SOURce1:PHASe 10 /*Sets the phase of the basic waveform output
from GI to 10°.*/
:SOURce1:PHASe? /*The query returns 1.000000E+1.*/
```

3.25.7 :SOURce<n>:PHASe:SYNChronize

Syntax

:SOURce<*n*>**:PHASe:SYNChronize**

Description

Performs the "align phase" operation.

Parameter

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

Remarks

After running this command to perform the "align phase" operation, the instrument will re-configure the two Function/Arbitrary Waveform Generator channels to output according to the preset frequency and phase. If these two signals whose frequencies are identical or in multiple, performing this operation will align their phases.

Return Format

N/A

Example

```
:SOURce1:PHASe:SYNChronize /*Performs "align phase" operation on
GI.*/
```

3.25.8 :SOURce<n>:FUNCtion:RAMP:SYMMetry

Syntax

:SOURce<*n*>**:FUNCtion:RAMP:SYMMetry** <*symm*>

:SOURce<*n*>**:FUNCtion:RAMP:SYMMetry?**

Description

Sets or queries the symmetry of the Ramp waveform output from the specified function/arbitrary waveform generator channel.

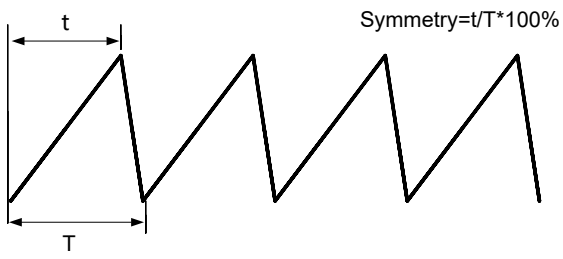
Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<symm>	Real	0% to 100%	50%

Remarks

This command is only available when the waveform type (*:SOURce<n>:FUNCtion*) is "Ramp".

Symmetry is defined as the percentage that the rising period of the ramp takes up in the whole period.



Return Format

The query returns the symmetry in scientific notation.

Example

```
:SOURce1:FUNCtion:RAMP:SYMMetry 55 /*Sets the symmetry of the Ramp
waveform output from GI to 55%.*/
:SOURce1:FUNCtion:RAMP:SYMMetry? /*The query returns 5.500000E+1.*/
```

3.25.9 :SOURce<n>:FUNCtion:SQUare:DUTY

Syntax

:SOURce<*n*>**:FUNCtion:SQUare:DUTY** <*duty*>

:SOURce<*n*>**:FUNCtion:SQUare:DUTY?**

Description

Sets or queries the duty cycle of the Square waveform output from the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<duty>	Real	1 to 99	50

Remarks

This command is only available when the waveform type is set to "Square". To set or query the waveform type of the basic waveform, run the *:SOURce<n>:FUNCtion* command.

Duty cycle is defined as the percentage that the high level takes up in the whole period of Square waveform.

Return Format

The query returns the duty cycle of the Square waveform in scientific notation.

Example

```
:SOURce1:FUNCtion:SQUare:DUTY 55 /*Sets the duty cycle of the
Square waveform output from GI to SQUare.*/
:SOURce1:FUNCtion:SQUare:DUTY? /*The query returns 5.500000E+1.*/
```

3.25.10 :SOURce<n>:VOLTage:AMPLitude

Syntax

:SOURce<*n*>**:VOLTage:AMPLitude** <*ampl*>

:SOURce<*n*>:VOLTage:AMPLitude?

Description

Sets or queries the amplitude of the basic waveform. The unit is V by default.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<ampl>	Real	Refer to *Remarks*	5 V

Remarks

The range of the amplitude is related to the frequency and the selected output impedance.

When "Load" is selected for Output Impedance, the ranges of the amplitudes in different frequency ranges are as follows:

- 1 mV_{pp} to 10 V_{pp} (@Frequency ≤ 50 MHz)
- 1 mV_{pp} to 5 V_{pp} (@Frequency ≤ 100 MHz)

When "HighZ" is selected for Output Impedance, the ranges of the amplitudes in different frequency ranges are as follows:

- 2 mV_{pp} to 20 V_{pp} (@Frequency ≤ 50 MHz)
- 2 mV_{pp} to 10 V_{pp} (@Frequency ≤ 100 MHz)

To set or query the frequency of the basic waveform, run the *:SOURce<n>:FREQuency* command.

Return Format

The query returns the amplitude of the waveform in scientific notation. The unit is V.

Example

```
:SOURce1:VOLTage:AMPLitude 1 /*Sets the amplitude of the basic
waveform of GI to 1 V.*/
:SOURce1:VOLTage:AMPLitude? /*The query returns 1.000000E0.*/
```

3.25.11 :SOURce<n>:VOLTage:OFFSet

Syntax

:SOURce<*n*>:VOLTage:OFFSet <*offset*>

:SOURce<*n*>:VOLTage:OFFSet?

Description

Sets or queries the amplitude offset of the basic waveform. The unit is V by default.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<offset>	Real	Refer to *Remarks*	0 V

Remarks

The amplitude offset of the basic waveform is related to the amplitude of the basic waveform.

Offset = ±(max. value of the current amplitude - currently set amplitude)/2

For example,

- If the current frequency of the currently selected basic waveform is 5 MHz and the input impedance is Load, the max. settable amplitude is 10 V, and set the amplitude to 6 V, then the range of the settable offset is ±(10V - 6V)/2 =±2V.
- If the current frequency of the currently selected basic waveform is 60 MHz and the input impedance is Load, the max. settable amplitude is 5 V, and set the amplitude to 3 V, then the range of the settable offset is ±(5V - 3V)/2 =±1V.

To set or query the amplitude of the basic waveform, run the *:SOURce<n>:VOLTage:AMPLitude* command.

Return Format

The query returns the amplitude offset value of the waveform in scientific notation. The unit is V.

Example

```
:SOURce1:VOLTage:OFFSet 0.2 /*Sets the amplitude offset of GI to
200 mV.*/
:SOURce1:VOLTage:OFFSet? /*The query returns 2.000000E-1.*/
```

3.25.12 :SOURce<n>:VOLTage:HIGH

Syntax

:SOURce<*n*>**:VOLTage:HIGH** <*value*>

:SOURce<*n*>**:VOLTage:HIGH?**

Description

Sets or queries the high level of the basic waveform output from the specified function/arbitrary waveform generator channel. By default, the unit is V.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<value>	Real	Refer to *Remarks*	-

Remarks

The high level is affected by the amplitude, low level, and output impedance.

The max. high level is (current max. amplitude/2); and the min. high level is (current low level + accuracy). Wherein, accuracy is 1 mV on 50Ω load and 2 mV on HighZ.

For example, under HighZ, if the current max. amplitude is 20 V and the low level is -1 mV, then the max. high level is (20V/2 = 10 V); the min. high level is (-1 mV + 2 mV = 1 mV).

To set or query the amplitude of the basic waveform, run the *:SOURce<n>:VOLTage:AMPLitude* command. To set or query the amplitude offset of the basic waveform, run the *:SOURce<n>:VOLTage:OFFSet* command. To set or query the output impedance of the specified function/arbitrary waveform generator channel, run the *:SOURce<n>:IMPedance* command.

Return Format

The query returns the high level value of the waveform in scientific notation. The unit is V.

Example

```
:SOURce1:VOLTage:HIGH 1 /*Sets the max. high level of the basic
waveform of GI to 1 V.*/
:SOURce1:VOLTage:HIGH? /*The query returns 1.000000E0.*/
```

3.25.13 :SOURce<n>:VOLTage:LOW

Syntax

:SOURce<*n*>**:VOLTage:LOW** <*value*>

:SOURce<*n*>**:VOLTage:LOW?**

Description

Sets or queries the low level of the basic waveform output from the specified function/arbitrary waveform generator channel. By default, the unit is V.

Parameter

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

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

Remarks

The high level is affected by the amplitude, low level, and output impedance.

The min. low level is (current max. amplitude/2); and the max. low level is (current high level - accuracy). Wherein, accuracy is 1 mV on 50Ω load and 2 mV on HighZ.

For example, under HighZ, if the current max. amplitude is 20 V and the high level is 3.00 mV, then the min. low level is -(20V/2) = -10 V; the max. low level is (3 mV- 2 mV = 1 mV/div).

To set or query the amplitude of the basic waveform, run the *:SOURce<n>:VOLTage:AMPLitude* command. To set or query the amplitude offset of the basic waveform, run the *:SOURce<n>:VOLTage:OFFSet* command. To set or query the output impedance of the specified function/arbitrary waveform generator channel, run the *:SOURce<n>:IMPedance* command.

Return Format

The query returns the low level value of the waveform in scientific notation. The unit is V.

Example

```
:SOURce1:VOLTage:LOW 0.001 /*Sets the low level of the basic
waveform of GI to 1 mV.*/
:SOURce1:VOLTage:LOW? /*The query returns 1.000000E-3.*/
```

3.25.14 :SOURce<n>:IMPedance

Syntax

:SOURce<*n*>**:IMPedance** <*imp*>

:SOURce<*n*>**:IMPedance?**

Description

Sets or queries the output impedance of the specified function/arbitrary waveform generator channel.

Parameter

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

Remarks

- **OMEG:** indicates HighZ.
- **FIFTy:** indicates Load.

Return Format

The query returns OMEG or FIFTy.

Example

```
:SOURce1:IMPedance FIFTy /*Sets the output impedance of GI to
Load.*/
:SOURce1:IMPedance? /*The query returns FIFTy.*/
```

3.25.15 :SOURce<n>:MOD:STATe

Syntax

:SOURce<*n*>**:MOD:STATe** <*state*>

:SOURce<*n*>**:MOD:STATe?**

Description

Enables or disables the modulation output; queries the on/off status of the modulation of the specified function/arbitrary waveform generator channel.

Parameter

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

Remarks

N/A

Return Format

The query returns 1 or 0.

Example

```
:SOURce1:MOD:STATe 1 /*Enables the modulation output of GI.*/
:SOURce1:MOD:STATe? /*The query returns 1.*/
```

3.25.16 :SOURce<n>:MOD:TYPe

Syntax

:SOURce<*n*>**:MOD:TYPe** <*type*>

:SOURce<*n*>:MOD:TYPe?

Description

Sets or queries the modulation type of the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<type>	Discrete	{AM|FM|PM}	AM

Remarks

- **AM:** amplitude modulation, namely the amplitude of the carrier waveform changes with that of the modulating waveform.
- **FM:** frequency modulation, namely the amplitude of the carrier waveform changes with that of the modulating waveform.
- **PM:** phase modulation, namely the phase of the carrier waveform changes with that of the modulating waveform.

Return Format

The query returns AM, FM, or PM.

Example

```
:SOURce1:MOD:TYPe AM /*Sets the modulation type of GI to AM.*/
:SOURce1:MOD:TYPe? /*The query returns AM.*/
```

3.25.17 :SOURce<n>:MOD:AM:DEPTh

Syntax

:SOURce<*n*>:MOD:AM:DEPTh <*depth*>

:SOURce<*n*>:MOD:AM:DEPTh?

Description

Sets or queries the modulation depth of AM.

Parameter

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

Name	Type	Range	Default
<depth>	Real	0% to 120%	100%

Remarks

Modulation depth indicates the amplitude variation. It is expressed in percentage.

- At 0% depth, the output amplitude is half of the carrier waveform amplitude.
- At 100% depth, the output amplitude is identical to the carrier's amplitude setting.
- When depth >100%, the output amplitude will not exceed the currently supported max. output amplitude.

Return Format

The query returns AM modulation depth in scientific notation.

Example

```
:SOURce1:MOD:AM:DEPTh 50 /*Sets AM modulation depth of GI to 50%.*/
:SOURce1:MOD:AM:DEPTh? /*The query returns 5.000000E+1.*/
```

3.25.18 :SOURce<n>:MOD:AM:INTernal:FREQuency

Syntax

:SOURce<*n*>**:MOD:AM:INTernal:FREQuency** <*freq*>

:SOURce<*n*>**:MOD:AM:INTernal:FREQuency?**

Description

Sets or queries the AM modulation frequency of the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<freq>	Real	2 mHz to 1 MHz	100 Hz

Remarks

N/A

Return Format

The query returns AM modulation frequency in scientific notation.

Example

```
:SOURce1:MOD:AM:INTernal:FREQ 150 /*Sets AM frequency of GI to 150
Hz.*/
:SOURce1:MOD:AM:INTernal:FREQ? /*The query returns 1.500000E+2.*/
```

3.25.19 :SOURce<n>:MOD:AM:INTernal:FUNCtion

Syntax

:SOURce<*n*>**:MOD:AM:INTernal:FUNCtion** <*func*>

:SOURce<*n*>**:MOD:AM:INTernal:FUNCtion?**

Description

Sets or queries the AM modulation waveform.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<func>	Discrete	{SINusoid|SQUare|TRIangle| UPRamp|DNRamp|NOISe}	SINusoid

Remarks

- **SINusoid:** sine waveform.
- **SQUare:** square waveform with the duty cycle of 50%.
- **TRIangle:** triangle waveform with the 50% symmetry.
- **UPRamp:** upramp waveform with the 100% symmetry.
- **UPRamp:** downramp waveform with the 0% symmetry.
- **NOISe:** noise waveform.

Return Format

The query returns SIN, SQU, TRI, UPR, DNR, or NOIS.

Example

```
:SOURce1:MOD:AM:INTernal:FUNCtion SQUare /*Sets AM modulation
waveform of GI to SQUare.*/
:SOURce1:MOD:AM:INTernal:FUNCtion? /*The query returns SQU.*/
```

3.25.20 :SOURce<n>:MOD:FM:DEViation

Syntax

:SOURce<*n*>**:MOD:FM:DEViation** <*deviation*>

:SOURce<*n*>:MOD:FM:DEViation?

Description

Sets or queries the FM frequency deviation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<deviation>	Real	2 mHz to current carrier frequency	1 kHz

Remarks

- The frequency deviation is the deviation of the frequency of the modulating waveform from that of the carrier waveform.
- The sum of frequency deviation and carrier frequency must be smaller than or equal to the upper limit of the current carrier frequency.
- For different modulation types, the ranges of the frequencies of the different carrier waveforms (refer to *:SOURce<n>:FREQuency*) are different.

Return Format

The query returns the frequency deviation in scientific notation.

Example

```
:SOURce1:MOD:FM:DEViation 100 /*Sets FM frequency deviation of GI
to 100 Hz.*/
:SOURce1:MOD:FM:DEViation? /*The query returns 1.000000E+2.*/
```

3.25.21 :SOURce<n>:MOD:FM:INTernal:FREQuency

Syntax

:SOURce<*n*>:MOD:FM:INTernal:FREQuency <*freq*>

:SOURce<*n*>:MOD:FM:INTernal:FREQuency?

Description

Sets or queries the FM modulation frequency of the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<freq>	Real	2 mHz to 1 MHz	100 Hz

Remarks

N/A

Return Format

The query returns FM modulation frequency in scientific notation.

Example

```
:SOURce1:MOD:FM:INTernal:FREQ 150 /*Sets FM frequency of GI to 150
Hz.*/
:SOURce1:MOD:FM:INTernal:FREQ? /*The query returns 1.500000E+2.*/
```

3.25.22 :SOURce<n>:MOD:FM:INTernal:FUNCtion

Syntax

:SOURce<*n*>**:MOD:FM:INTernal:FUNCtion** <*func*>

:SOURce<*n*>**:MOD:FM:INTernal:FUNCtion?**

Description

Sets or queries the FM modulation waveform.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<func>	Discrete	{SINusoid|SQUare|TRIangle| UPRamp|DNRamp|NOISe}	SINusoid

Remarks

- **SINusoid:** sine waveform.
- **SQUare:** square waveform with the duty cycle of 50%.
- **TRIangle:** triangle waveform with the 50% symmetry.
- **UPRamp:** upramp waveform with the 100% symmetry.
- **UPRamp:** downramp waveform with the 0% symmetry.

- **NOISe:** noise waveform.

Return Format

The query returns SIN, SQU, TRI, UPR, DNR, or NOIS.

Example

```
:SOURce1:MOD:FM:INTernal:FUNCtion SQUare /*Sets FM modulation
waveform of GI to SQUare.*/
:SOURce1:MOD:FM:INTernal:FUNCtion? /*The query returns SQU.*/
```

3.25.23 :SOURce<n>:MOD:PM:DEViation

Syntax

:SOURce<*n*>**:MOD:PM:DEViation** <*deviation*>

:SOURce<*n*>**:MOD:PM:DEViation?**

Description

Sets or queries the PM phase deviation.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<deviation>	Real	0° to 360°	90°

Remarks

Phase deviation indicates the deviation of the modulating waveform phase from the carrier waveform phase.

Return Format

The query returns PM phase deviation in scientific notation.

Example

```
:SOURce1:MOD:PM:DEViation 50 /*Sets PM phase deviation of GI to
50°.*/
:SOURce1:MOD:PM:DEViation? /*The query returns 5.000000E+1.*/
```

3.25.24 :SOURce<n>:MOD:PM:INTernal:FREQuency

Syntax

:SOURce<*n*>**:MOD:PM:INTernal:FREQuency** <*freq*>

:SOURce<*n*>**:MOD:PM:INTernal:FREQuency?**

Description

Sets or queries the PM modulation frequency of the specified function/arbitrary waveform generator channel.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<freq>	Real	2 mHz to 1 MHz	100 Hz

Remarks

N/A

Return Format

The query returns PM modulation frequency in scientific notation.

Example

```
:SOURce1:MOD:PM:INTernal:FREQ 150 /*Sets PM frequency of GI to 150 Hz.*/
:SOURce1:MOD:PM:INTernal:FREQ? /*The query returns 1.500000E+2.*/
```

3.25.25 :SOURce<n>:MOD:PM:INTernal:FUNCtion

Syntax

:SOURce<*n*>**:MOD:PM:INTernal:FUNCtion** <*function*>

:SOURce<*n*>**:MOD:PM:INTernal:FUNCtion?**

Description

Sets or queries the PM modulation waveform.

Parameter

Name	Type	Range	Default
<n>	Discrete	{1|2}	-
<func>	Discrete	{SINusoid|SQUare|TRIangle| UPRamp|DNRamp|NOISe}	SINusoid

Remarks

- **SINusoid:** sine waveform.
- **SQUare:** square waveform with the duty cycle of 50%.
- **TRIangle:** triangle waveform with the 50% symmetry.

- **UPRamp:** upramp waveform with the 100% symmetry.
- **UPRamp:** downramp waveform with the 0% symmetry.
- **NOISe:** noise waveform.

Return Format

The query returns SIN, SQU, TRI, UPR, DNR, or NOIS.

Example

```
:SOURce1:MOD:PM:INTernal:FUNCtion SQUare /*Sets PM modulation
waveform of GI to SQUare.*/
:SOURce1:MOD:PM:INTernal:FUNCtion? /*The query returns SQU.*/
```

3.26 :TIMebase Commands

:TIMebase commands are used to set the horizontal system. For example, you can enable the Zoom mode or set the horizontal time base mode.

Horizontal Time Base Mode

- **YT Mode:** By default, this series oscilloscope uses the YT mode for waveform display window. In YT mode, Y-axis indicates the Voltage and X-axis indicates the Time.
- **XY Mode:** In XY mode, both the X-axis and Y-axis indicate voltage. The XY mode converts the oscilloscope from a "Voltage-Time" display to a "Voltage-Voltage" display using two input channels. The XY mode can be used to measure the phase deviation occurred when the signal under test passes through a circuit network.
- **Roll Mode:** The roll mode causes the waveform to move across the screen from right to left. It allows you to view the acquired data without waiting for a complete acquisition. The Roll mode is automatically enabled when the horizontal time base is set to 50 ms/div or slower.

TIP

- If the Zoom mode is currently turned on, enabling the roll mode automatically turns off the Zoom mode.
- The following functions are not available when the roll mode is enabled: To Adjust the Horizontal Position (available when the oscilloscope run state is STOP), Zoom Mode (Delayed Sweep), Triggering the Oscilloscope, Protocol Decoding, Pass/Fail

Test, Waveform Recording and Playing, Persistence Time, XY Mode, Average, Histogram, and Search function.

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*	-

Remarks

- The maximum value of the parameter <scale> is the current scale of the main time base.
- The delayed time base scale can only be the maximum value or the values acquired by reducing the maximum value at 1-2-5 step. If the minimum value calculated according to the above formula is not one of the settable values, take the minimum settable value that is greater than the minimum value calculated.
- The default value of the delayed time base scale is determined by the scale in Main mode and its value is the next scale value in Main mode.

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

- In RUN state, the offset range of the main time base is as follows:

  MainLeftTime = -5 x MainScale

  When MainScale ≤ 10 ms, MainRightTime = 1 s

  When 10 ms < MainScale < 10 s, MainRightTime = 100 x MainScale

  When MainScale < 200 s and MainScale ≥ 10 s, MainRightTime = 1 ks

  When MainScale ≥ 200s, MainRightTime = 5 x MainScale

  MainRightTime indicates the maximum offset value of the main time base; MainLeftTime indicates the minimum offset value of the main time base; MainScale is the current scale of the main time base.

- In Stop state, the offset range of the main time base is the same as that of the memory sampling.

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 main time base scale.

Parameter

Name	Type	Range	Default
<scale>	Real	Refer to *Remarks*	2 μs/div

Remarks

The range of <scale> is related to the bandwidth of the oscilloscope and the horizontal time base mode.

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:** indicates the current time base mode. When you send the **:TIMebase:MODE MAIN**command, the time base is set to YT mode.

- **XY:** indicates the XY mode. After performing the command **:TIMebase:MODE XY** to configure the XY mode, you can send the query command **:TIMebase:MODE?** to query the time base mode. The query returns MAIN.
- **ROLL:** roll mode.

For different time base modes, refer to *Horizontal Time Base Mode*.

Return Format

The query returns MAIN, XY, 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>	Integer	-500 to 500	0

Remarks

When <pos> is set to -500, the reference position is the leftmost side of the screen. When <pos> is set to 500, the reference position is the rightmost side of the screen.

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>	Bool	{{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 mode.

Parameter

Name	Type	Range	Default
<value>	Integer	{0|1}	1

Remarks

- **0:** disables the Auto ROLL mode.
- **1:** enables the Auto ROLL mode. The system enters the ROLL mode automatically when the time base is greater than or equal to 50 ms.

Return Format

The query returns 0 or 1.

Example

```
:TIMebase:ROLL 0    /*Disables the ROLL 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, 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, CHAN4.

Example

```
:TIMebase:XY:Y CHANnel3          /*Sets the channel source of the Y
coordinate to CHANnel3.*/
:TIMebase:XY:Y?                 /*The query returns CHAN3.*/
```

3.26.15 :TIMebase:XY:GRID

Syntax

:TIMebase:XY:GRID <*grid*>

:TIMebase:XY:GRID?

Description

Sets or queries the grid type of the XY display.

Parameter

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

Remarks

- **FULL:** turns the background grid and coordinates on.
- **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

```
:TIMebase:XY: GRID NONE /*Sets the grid type of the XY display.*/
:TIMebase:XY: GRID? /*The query returns NONE.*/
```

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 generating 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|LIN| IIS|FLEXray|M1553}	EDGE

Remarks

The IIS, FlexRay, and M1553 trigger types are options for this series oscilloscope. These trigger commands are only available when the specified trigger option is installed.

Return Format

The query returns EDGE, PULS, SLOP, VID, PATT, DUR, TIM, RUNT, WIND, DEL, SET, NEDG, RS232, IIC, SPI, CAN, LIN, IIS, FLEX, 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

Selects or queries the trigger coupling type.

Parameter

Name	Type	Range	Default
<couple>	Discrete	{AC|DC|LFReject|HFReject}	DC

Remarks

This command is only available for the Edge trigger in which the analog channel is selected as the source.

- **AC:** blocks any DC components to pass the trigger path.
- **DC:** allows DC and AC components to pass the trigger path.
- **LFReject:** blocks the DC components and rejects the low frequency components to pass the trigger path.
- **HFReject:** rejects the high frequency components to pass the trigger path.

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.

Parameter

N/A

Remarks

N/A

Return Format

The query returns TD, WAIT, RUN, AUTO, or STOP.

Example

```
N/A
```

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 10s	8 ns

Remarks

- Trigger holdoff can be used to stably trigger complex waveforms (such as pulse waveform). Holdoff time indicates the time that the oscilloscope waits for re-arming the trigger module. The oscilloscope will not trigger before the holdoff time expires.
- When the trigger type is Video, Timeout, Nth Edge, RS232, I2C, SPI, CAN, LIN, FlexRay, I2S, or 1553B, holdoff time is not available.

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 you select the analog channels as the trigger source.

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

Edge trigger identifies a trigger on the trigger level of the specified edge on the input signal.

3.27.8.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, CHAN4.

Example

```
:TRIGger:EDGE:SOURce CHANnel1          /*Sets the trigger source to
CHANnel1.*/
:TRIGger:EDGE:SOURce?                 /*The query returns CHAN1.*/
```

3.27.8.2 :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.

- **RFALI:** 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.*/
```

3.27.8.3 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.

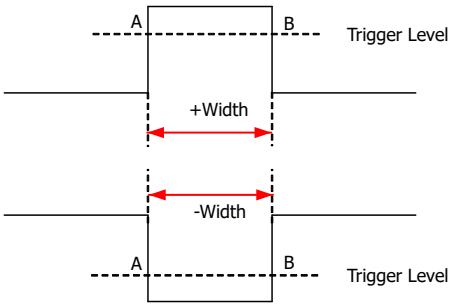
Example

```
:TRIGger:EDGE:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:EDGE:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.9 :TRIGger:PULSe

Pulse trigger sets the oscilloscope to trigger on the positive or negative pulse of a specified width. In this mode, the oscilloscope will trigger when the pulse width of the input signal satisfies the specified pulse width condition.

In this oscilloscope, positive pulse width is defined as the time difference between the two crossing points of the trigger level and positive pulse; negative pulse width is defined as the time difference between the two crossing points of the trigger level and negative pulse, as shown in the figure below.



3.27.9.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:PULSe:SOURce CHANnel1 /*Sets the trigger source to
CHANnel1.*/
:TRIGger:PULSe:SOURce? /*The query returns CHAN1.*/
```

3.27.9.2 :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.*/
```

3.27.9.3 :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.*/
```

3.27.9.4 :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 μs

Remarks

This command is only available when the trigger condition is set to LESS or GLESs. To set or query the trigger condition of the Pulse trigger, send the *:TRIGger:PULSe:WHEN* command.

When the trigger condition is set to GLESs, if the set pulse upper limit value is smaller than the lower limit, the lower limit will be automatically changed. You can send the *:TRIGger:PULSe:LWIDth* command to set or query the pulse lower limit value of the Pulse trigger.

Return Format

The query returns the pulse upper limit 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.*/
```

3.27.9.5 :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	1 ns to upper limit	1 µs

Remarks

This command is only available when the trigger condition is set to GREater or GLESs. To set or query the trigger condition of the Pulse trigger, send the *:TRIGger:PULSe:WHEN* command.

When the trigger condition is set to GLESs, if the set pulse lower limit value is greater than the upper limit, the upper limit will be automatically changed. You can send the *:TRIGger:PULSe:UWIDth* command to set or query the pulse upper limit value of the Pulse trigger.

Return Format

The query returns the pulse lower limit in scientific notation.

Example

```
:TRIGger:PULSe:LWIDth 0.000003    /*Sets the pulse lower limit of
the Pulse trigger to 3 µs.*/
:TRIGger:PULSe:LWIDth?            /*The query returns 3.000000E-6.*/
```

3.27.9.6 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.

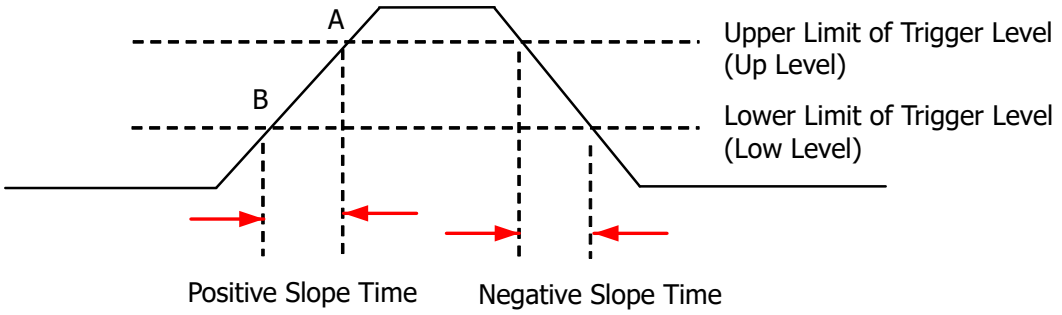
Example

```
:TRIGger:PULSe:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:PULSe:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.10 :TRIGger:SLOPe

Slope trigger sets the oscilloscope to trigger on the positive or negative slope of the specified time. This trigger mode is applicable to ramp and triangle waveforms.

In this oscilloscope, positive slope time is defined as the time difference between the two crossing points of trigger level line A and B with the rising edge; negative slope time is defined as the time difference between the two crossing points of trigger level line A and B with the falling edge. See the figure below.



3.27.10.1 :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, CHAN4.

Example

```
:TRIGger:SLOPe:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:SLOPe:SOURce?             /*The query returns CHAN2.*/
```

3.27.10.2 :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.*/
```

3.27.10.3 :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.*/
```

3.27.10.4 :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 s.

Parameter

Name	Type	Range	Default
<time>	Real	Lower limit to 10 s	1 μs

Remarks

This command is only available when the trigger condition is set to LESS or GLES. To set or query the trigger condition of the Slope trigger, send the *:TRIGger:SLOPe:WHEN* command.

When the trigger condition is set to GLESs, if the set upper time limit value is smaller than the lower limit, the lower limit will be automatically changed. You can use the *:TRIGger:SLOPe:TLOWer* command to set or query the lower time limit value of the Slope trigger.

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.*/
```

3.27.10.5 :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 s.

Parameter

Name	Type	Range	Default
<time>	Real	1 ns to upper limit	1 μs

Remarks

This command is only available when the trigger condition is set to GREater or GLESs. To set or query the trigger condition of the Slope trigger, send the *:TRIGger:SLOPe:WHEN* command.

When the trigger condition is set to GLESs, if the set lower time limit value is greater than the upper limit, the upper limit will be automatically changed. You can send the *:TRIGger:SLOPe:TUPPer* command to set or query the upper time limit value of the Slope trigger.

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.*/
```

3.27.10.6 :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.*/
```

3.27.10.7 :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	Trigger level lower limit to (4.5 x 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.*/
```

3.27.10.8 :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	(-4.5 x VerticalScale - OFFSet) to trigger level 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:SLOPe:BLEVel 0.16   /*Sets the lower limit of the trigger
level to 160 mV.*/
:TRIGger:SLOPe:BLEVel?      /*The query returns 1.600000E-1.*/
```

3.27.11 :TRIGger:VIDeo

The video signal can include image information and timing information, which adopts different standards and formats. This series can trigger on the standard video signal field or line of NTSC (National Television Standards Committee), PAL (Phase Alternating Line), or SECAM (Sequential Couleur A Memoire).

3.27.11.1 :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, CHAN4.

Example

```
:TRIGger:VIDeo:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:VIDeo:SOURce?             /*The query returns CHAN2.*/
```

3.27.11.2 :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.*/
```

3.27.11.3 :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.*/
```

3.27.11.4 :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.*/
```

3.27.11.5 :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)	Sweep Function	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    /*Sets the video standard to
NTSC.*/
:TRIGger:VIDeo:STANdard?         /*The query returns NTSC.*/
```

3.27.11.6 :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	(-4.5 x VerticalScale - OFFSet) to (4.5 x 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

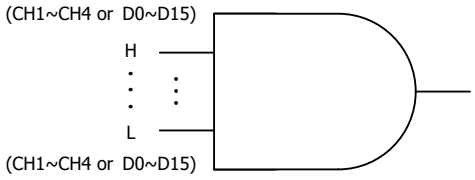
The query returns the trigger level in scientific notation.

Example

```
:TRIGger:VIDeo:LEVel 0.16     /*Sets the trigger level to 160 mV.*/
:TRIGger:VIDeo:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.12 :TRIGger:PATTern

The pattern trigger identifies a trigger condition by looking for a specified pattern. This pattern is a logical "AND" combination of channels. Each channel can be set to H (high), L (low), or X (don't care). A rising or falling edge (you can only specify a single edge) can be specified for one channel included in the pattern. When an edge is specified, the oscilloscope will trigger at the edge specified if the pattern set for the other channels are true (namely the actual pattern of the channel is the same as the preset pattern). If no edge is specified, the oscilloscope will trigger on the last edge that makes the pattern true. If all the channels in the pattern are set to "X", the oscilloscope will not trigger.



3.27.12.1 :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

Name	Type	Range	Default
<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 "CHANnel1 to CHANnel4".
- 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 "CHANnel1
to CHANnel4" to H,R,L,X.*/
:TRIGger:PATTern:PATTern? /*The query returns H,R,L,X.*/
```

3.27.12.2 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:PATTern:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:PATTern:SOURce?             /*The query returns CHAN2.*/
```

3.27.12.3 :TRIGger:PATTern:LEVel

Syntax

`:TRIGger:PATTern:LEVel` <*source*>,<*level*>

`:TRIGger:PATTern:LEVel?` <*source*>

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1
<level>	Real	Analog channel: (-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset) Digital channel: -15 V to 15 V	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

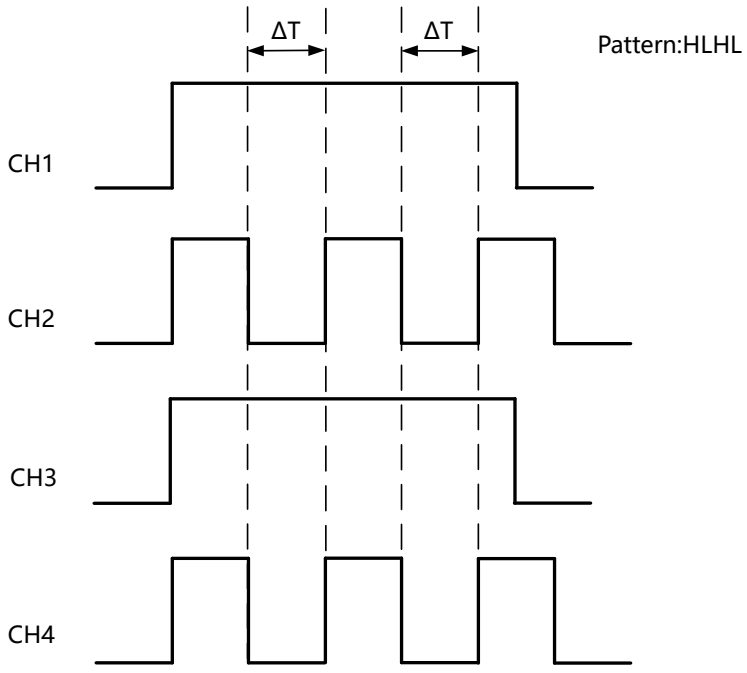
The query returns the trigger level in scientific notation.

Example

```
:TRIGger:PATTern:LEVel CHANnel2,0.16  /*Sets the trigger level of
CHANnel2 to 160 mV.*/
:TRIGger:PATTern:LEVel? CHANnel2     /*The query returns
1.600000E-1.*/
```

3.27.13 :TRIGger:DURation

In Duration trigger, the oscilloscope identifies a trigger condition by searching for the duration of a specified pattern. This pattern is a logical "AND" combination of the channels. Each channel can be set to 1 (high), 0 (low), or X (don't care). The instrument triggers when the duration (ΔT) of this pattern meets the preset time, as shown in the figure below.



3.27.13.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DURation:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:DURation:SOURce?             /*The query returns CHAN2.*/
```

3.27.13.2 :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 "CHANnel1 to CHANnel4".
- 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 "CHANnel1 to
CHANnel4" to L,X,H,L.*/
:TRIGger:DURation:TYPE? /*The query returns L,X,H,L.*/
```

3.27.13.3 :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 or 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.*/
```

3.27.13.4 :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	1.01 ns to 10 s	1 µs

Remarks

This command is only available when the trigger condition is set to LESS, GLESs, or UNGLess. To set or query the trigger condition of the Duration trigger, send the *:TRIGger:DURation:WHEN* command.

When the trigger condition is set to GLESs or UNGLess, if the set upper limit of the duration time value is smaller than the lower limit, the lower limit will be automatically changed. You can send the *:TRIGger:DURation:TLOWer* command to set or query the lower limit of the duration time value of the Duration trigger.

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.*/
```

3.27.13.5 :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	1 ns to 9.9 s	1 μs

Remarks

This command is only available when the trigger condition is set to GREater, GLESs, or UNGLess. To set or query the trigger condition of the Duration trigger, send the *:TRIGger:DURation:WHEN* command.

When the trigger condition is set to GLESs or UNGLess, if the set lower limit of the duration time value is greater than the upper limit, the upper limit will be automatically changed. You can send the *:TRIGger:DURation:TUPPer* command to set or query the upper limit of the duration time value of the Duration trigger.

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.*/
```

3.27.13.6 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1
<level>	Real	Analog channel: (-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset) Digital channel: -15 V to 15 V	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

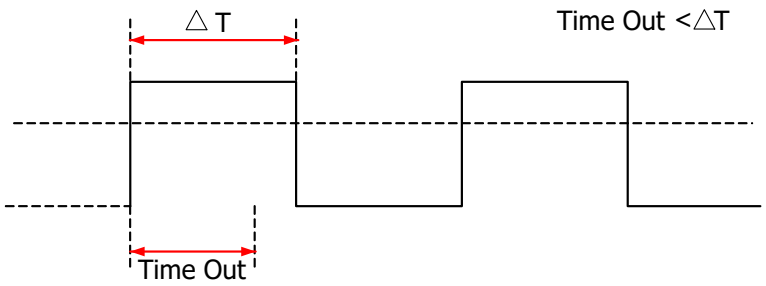
The query returns the trigger level in scientific notation.

Example

```
:TRIGger:DURation:LEVel CHANnel2,0.16  /*Sets the trigger level of CHANnel2 to 160 mV.*/
:TRIGger:DURation:LEVel? CHANnel2     /*The query returns 1.600000E-1.*/
```

3.27.14 :TRIGger:TIMeout

In Timeout trigger, the oscilloscope triggers when the time interval (ΔT) (the time from when the rising edge (or falling edge) of the input signal passes through the trigger level to the time from when the neighboring falling edge (or rising edge) passes through the trigger level) is greater than the preset timeout value, as shown in the figure below.



3.27.14.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:TIMeout:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:TIMeout:SOURce?             /*The query returns CHAN2.*/
```

3.27.14.2 :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.*/
```

3.27.14.3 :TRIGger:TIMeout:TIME

Syntax

:TRIGger:TIMeout:TIME <*time*>

:TRIGger:TIMeout:TIME?

Description

Sets or queries the timeout value of the Timeout trigger. The default unit is s, and the accuracy is 1 ns.

Parameter

Name	Type	Range	Default
<time>	Real	1 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.*/
```

3.27.14.4 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

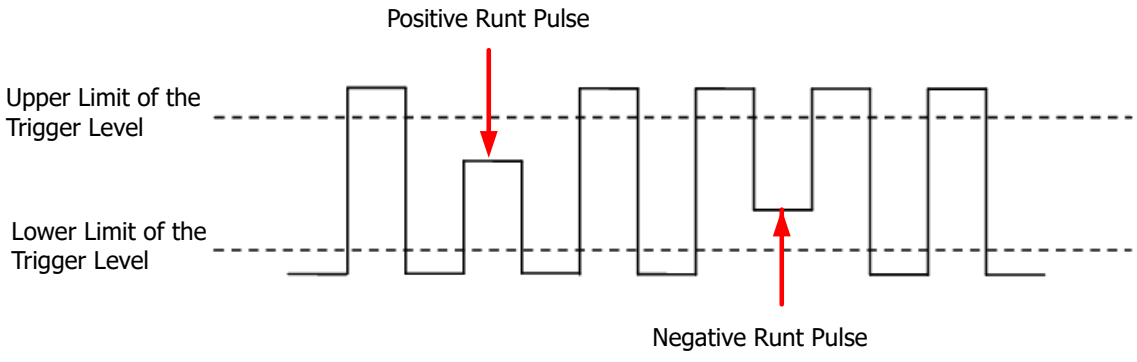
The query returns the trigger level in scientific notation.

Example

```
:TRIGger:TIMeout:LEVel 0.16     /*Sets the trigger level to 160
mV.*/
:TRIGger:TIMeout:LEVel?         /*The query returns 1.600000E-1.*/
```

3.27.15 :TRIGger:RUNT

The runt trigger sets the oscilloscope to trigger pulses that cross one trigger level but not another, as shown in the figure below.



3.27.15.1 :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, CHAN4.

Example

```
:TRIGger:RUNT:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:RUNT:SOURce?             /*The query returns CHAN2.*/
```

3.27.15.2 :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.*/
```

3.27.15.3 :TRIGger:RUNT:WHEN

Syntax

:TRIGger:RUNT:WHEN <*when*>

:TRIGger:RUNT:WHEN?

Description

Sets or queries the trigger conditions 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 of Runt
trigger to LESS.*/
:TRIGger:RUNT:WHEN?            /*The query returns LESS.*/
```

3.27.15.4 :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	1.01 ns to 10 s	2 µs

Remarks

This command is only available when the trigger condition is set to LESS or GLESs. To set or query the trigger condition of the Runt trigger, send the *:TRIGger:RUNT:WHEN* command.

When the trigger condition is set to GLESs, if the set upper limit of the pulse width is smaller than the lower limit, the lower limit will be automatically changed. You can send the *:TRIGger:RUNT:WLOWer* command to set or query the lower limit of the pulse width of Runt trigger.

Return Format

The query returns the pulse upper limit 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.*/
```

3.27.15.5 :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	1 ns to 9.9 s	1 μs

Remarks

This command is only available when the trigger condition is set to GREater or GLESs. To set or query the trigger condition of the Runt trigger, send the *:TRIGger:RUNT:WHEN* command.

When the trigger condition is set to GLESs, if the set lower limit of the pulse width is greater than the lower limit, the upper limit will be automatically changed. You can send the *:TRIGger:RUNT:WUPPer* command to set or query the upper limit of the pulse width of Runt trigger.

Return Format

The query returns the pulse lower limit 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.*/
```

3.27.15.6 :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 (4.5 x 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.*/
```

3.27.15.7 :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	(-4.5 x 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.*/
```

3.27.16 :TRIGger:WINDows

Window trigger provides a high trigger level and a low trigger level. The instrument triggers when the input signal passes through the high trigger level or the low trigger level.

3.27.16.1 :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, CHAN4.

Example

```
:TRIGger:WINDows:SOURce CHANnel2    /*Sets the trigger source to CHANnel2.*/
:TRIGger:WINDows:SOURce?             /*The query returns CHAN2.*/
```

3.27.16.2 :TRIGger:WINDows:SLOPe

Syntax

:TRIGger:WINDows:SLOPe <*type*>

:TRIGger:WINDows:SLOPe?

Description

Sets or queries the edge type of Windows trigger.

Parameter

Name	Type	Range	Default
<type>	Discrete	{POSitive|NEGative|RFALI}	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.
- **RFALI:** 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 edge type of Windows
trigger to NEGative.*/
:TRIGger:WINDows:SLOPe?            /*The query returns NEG.*/
```

3.27.16.3 :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.*/
```

3.27.16.4 :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	1 ns to 10 s	1μs

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.*/
```

3.27.16.5 :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 (4.5 x 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.*/
```

3.27.16.6 :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	(-4.5 x 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

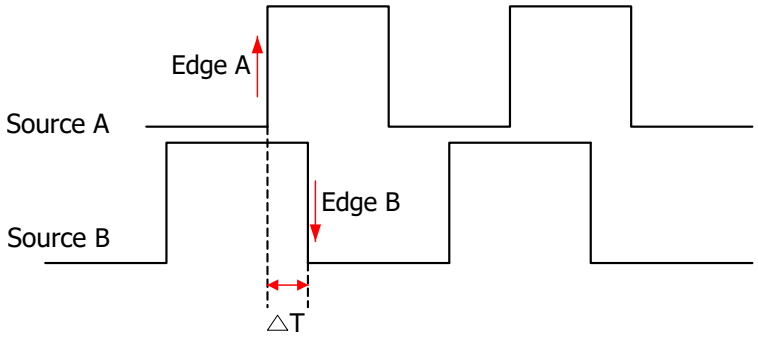
The query returns the lower limit of the trigger level in scientific notation.

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.*/
```

3.27.17 :TRIGger:DELay

In Delay trigger, you need to set Source A and Source B. The oscilloscope triggers when the time difference (ΔT) between the specified edges (Edge A and Edge B) of Source A and Source B meets the preset time limit, as shown in the figure below. Edge A and Edge B must be two neighboring edges. See the figure below.



3.27.17.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DELay:SA CHANnel2 /*Sets the trigger source A to
CHANnel2.*/
:TRIGger:DELay:SA?       /*The query returns CHAN2.*/
```

3.27.17.2 :TRIGger:DELay:ASLop

Syntax

:TRIGger:DELay:ASLop <*slope*>

:TRIGger:DELay:ASLop?

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:ASLop NEGative     /*Sets the edge type of Edge A to
NEGative.*/
:TRIGger:DELay:ASLop?             /*The query returns NEG.*/
```

3.27.17.3 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8| D9|D10|D11|D12|D13|D14|D15| CHANnel1|CHANnel2|CHANnel3| CHANnel4}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:DELay:SB CHANnel4 /*Sets the trigger source B to
CHANnel4.*/
:TRIGger:DELay:SB? /*The query returns CHAN4.*/
```

3.27.17.4 :TRIGger:DELay:BSLop

Syntax

:TRIGger:DELay:BSLop <*slope*>

:TRIGger:DELay:BSLop?

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:BSLop NEGative    /*Sets the edge type of Edge B to
NEGative.*/
:TRIGger:DELay:BSLop?            /*The query returns NEG.*/
```

3.27.17.5 :TRIGger:DELay:TYPE

Syntax

:TRIGger:DELay:TYPE <*type*>

:TRIGger:DELay:TYPE?

Description

Sets or queries the trigger condition of the 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 trigger condition to GOUT.*/
:TRIGger:DELay:TYPe?         /*The query returns GOUT.*/
```

3.27.17.6 :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	1.01 ns to 10 s	2 µs

Remarks

This command is only available when the trigger condition is set to LESS, GLESs, or GOUT. To set or query the trigger condition of the Delay trigger, send the *:TRIGger:DELay:TYPE* command.

When the trigger condition is set to GLESs or GOUT, if the set upper limit of the delay time is smaller than the lower limit, the lower limit will be automatically changed. You can send the *:TRIGger:DELay:TLOWer* command to set or query the lower limit of the delay time of the Delay trigger.

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.*/
```

3.27.17.7 :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	1 ns to 9.9 s	1μs

Remarks

This command is only available when the trigger condition is set to GREater, GLESs, or GOUT. To set or query the trigger condition of the Delay trigger, send the *:TRIGger:DELay:TYPE* command.

When the trigger condition is set to GLESs or GOUT, if the set lower limit of the delay time is greater than the upper limit, the upper limit will be automatically changed. You can send the *:TRIGger:DELay:TUPPer* command to set or query the upper limit of the delay time of the Delay trigger.

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.*/
```

3.27.17.8 :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	Analog channel: (-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset) Digital channel: -15 V to 15 V	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.*/
```

3.27.17.9 :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	Analog channel: (-4.5 x VerticalScale - Offset) to (4.5 x VerticalScale - Offset) Digital channel: -15 V to 15 V	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

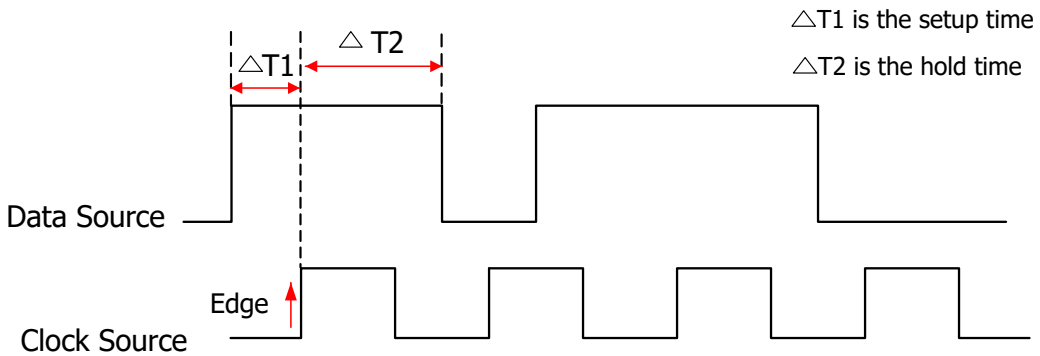
The query returns the threshold level of Source B in scientific notation.

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.*/
```

3.27.18 :TRIGger:SHOLd

In setup&hold trigger, you need to set the clock source and data source. The setup time starts when the data signal passes the trigger level and ends at the coming of the specified clock edge; the hold time starts at the coming of the specified clock edge and ends when the data signal crosses the trigger level again, as shown in the figure below. The oscilloscope triggers when the setup time or hold time is smaller than the preset time.



The data type is H

3.27.18.1 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SHOLd:DSRC CHANnel1    /*Sets the data source to
CHANnel1.*/
:TRIGger:SHOLd:DSRC?             /*The query returns CHAN1.*/
```

3.27.18.2 :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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SHOLd:CSRC CHANnel2    /*Sets the clock source to
CHANnel2.*/
:TRIGger:SHOLd:CSRC?             /*The query returns CHAN2.*/
```

3.27.18.3 :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.*/
```

3.27.18.4 :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.*/
```

3.27.18.5 :TRIGger:SHOLd:TYPE

Syntax

:TRIGger:SHOLd:TYPE <*type*>

:TRIGger:SHOLd:TYPE?

Description

Sets or queries the trigger condition 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 trigger condition to
SETHold.*/
:TRIGger:SHOLd:TYPE?           /*The query returns SETH.*/
```

3.27.18.6 :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	1 ns to 10 s	2 μs

Remarks

- Setup time indicates the time that the data remain stable and unchanged before the specified clock edge arrives.
- 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.*/
```

3.27.18.7 :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	1 ns to 10 s	1 µs

Remarks

- Hold time indicates the time that the data remain stable and unchanged after the specified clock edge arrives.
- 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.*/
```

3.27.18.8 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.*/
```

3.27.18.9 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	0 V

Remarks

For VerticalScale, refer to the *:CHANnel<n>:SCALe* command. For OFFSet, refer to the *:CHANnel<n>:OFFSet* command.

Return Format

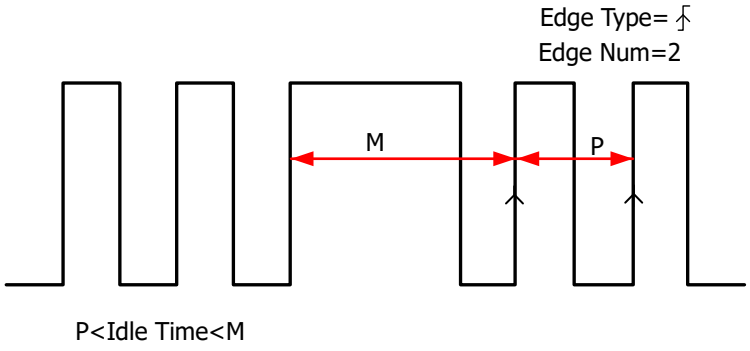
The query returns the trigger level of the clock source in scientific notation.

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.*/
```

3.27.19 :TRIGger:NEDGe

The Nth edge trigger lets you to trigger on the Nth edge that occurs after a specified idle time. For example, in the waveform as shown in the figure below, the instrument should trigger on the second rising edge after the specified idle time (the time between two neighboring rising edges), and the idle time should be within the range between P and M (P < Idle Time < M). Wherein, M is the time between the first rising edge and its previous rising edge; P is the maximum time between the rising edges that participate in counting.



3.27.19.1 :TRIGger:NEDGe:SOURce

Syntax

:TRIGger:NEDGe:SOURce <*source*>

:TRIGger:NEDGe:SOURce?

Description

Sets or queries the trigger source of Edge trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:NEDGe:SOURce CHANnel2 /*Sets the trigger source to
CHANnel2.*/
:TRIGger:NEDGe:SOURce? /*The query returns CHAN2.*/
```

3.27.19.2 :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.*/
```

3.27.19.3 :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	1μs

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.*/
```

3.27.19.4 :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.*/
```

3.27.19.5 :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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.*/
```

3.27.20 :TRIGger:RS232

RS232 bus is a serial communication mode used in data transmission between PCs or between a PC and a terminal. In RS232 serial protocol, a character is transmitted as a frame of data. The frame consists of 1 start bit, 5-8 data bits, 1 check bit, and 1-2 stop bits. Its format is as shown in the figure below. This series oscilloscope triggers when the start frame, error frame, check error, or the specified data of the RS232 signal is detected.

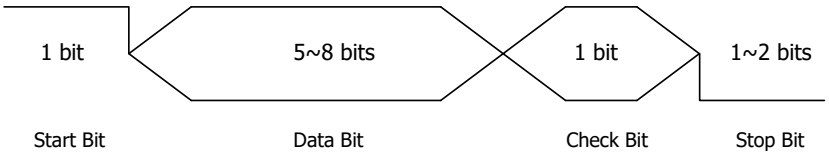


Figure 3.7 Schematic Diagram of RS232 Frame Format

3.27.20.1 :TRIGger:RS232:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:RS232:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:RS232:SOURce?             /*The query returns CHAN2.*/
```

3.27.20.2 :TRIGger:RS232:LEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 RS232 trigger 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.*/
```

3.27.20.3 :TRIGger:RS232:POLarity

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.*/
```

3.27.20.4 :TRIGger:RS232:WHEN

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.*/
```

3.27.20.5 :TRIGger:RS232:DATA

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, and 8. You can send the *:TRIGger:RS232:WIDTh* command to set or query the data width.

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.*/
```

3.27.20.6 :TRIGger:RS232:BAUD

Syntax

:TRIGger:RS232:BAUD <*baud*>

:TRIGger:RS232:BAUD?

Description

Sets or queries the baud rate of RS232 trigger. The default unit is bps.

Parameter

Name	Type	Range	Default
<baud>	Integer	1bps to 20Mbps	9600 bps

Remarks

N/A

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.*/
```

3.27.20.7 :TRIGger:RS232:WIDTh

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.*/
```

3.27.20.8 :TRIGger:RS232:STOP

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.*/
```

3.27.20.9 :TRIGger:RS232:PARity

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.*/
```

3.27.20.10 :TRIGger:RS232:BUSer

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	1bps to 20Mbps	9600 bps

Remarks

This command exists for backwards compatibility. Use the command *:TRIGger:RS232:BAUD*.

Return Format

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

Example

```
:TRIGger:RS232:BUSer 4800    /*Sets the baud rate to 4800 bps.*/
:TRIGger:RS232:BUSer?        /*The query returns 4800.*/
```

3.27.21 :TRIGger:IIC

I2C is a 2-wire serial bus used to connect the microcontroller and its peripheral device. It is a bus standard widely used in the microelectronic communication control field.

The I2C serial bus consists of SCL and SDA. Its transmission rate is determined by SCL, and its transmission data is determined by SDA, as shown in the figure below. The instrument series triggers on the start condition, restart, stop, missing

acknowledgment, specific device address, or data value. Besides, it can also trigger on the specific device address and data values at the same time.

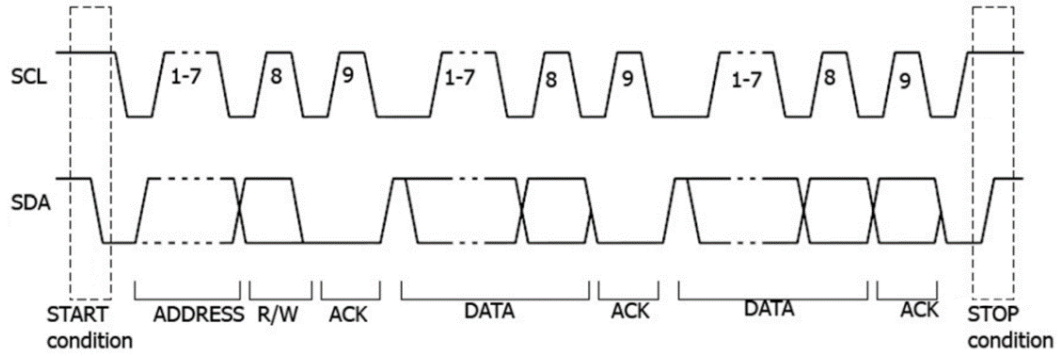


Figure 3.8 Schematic Diagram of I2C Frame Format

3.27.21.1 :TRIGger:IIC:SCL

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIC:SCL CHANnel2    /*Sets the clock source to CHANnel2.*/
:TRIGger:IIC:SCL?             /*The query returns CHAN2.*/
```

3.27.21.2 :TRIGger:IIC:CLEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 I2C trigger 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.*/
```

3.27.21.3 :TRIGger:IIC:SDA

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|	CHANnel2

Name	Type	Range	Default
		CHANnel1|CHANnel2|CHANnel3|CHANnel4}	

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIC:SDA CHANnel2    /*Sets the data source to CHANnel2.*/
:TRIGger:IIC:SDA?             /*The query returns CHAN2.*/
```

3.27.21.4 :TRIGger:IIC:DLEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 line in I2C trigger 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.*/
```

3.27.21.5 :TRIGger:IIC:WHEN

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.*/
```

3.27.21.6 :TRIGger:IIC:AWIDth

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.*/
```

3.27.21.7 :TRIGger:IIC:ADDRess

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.*/
```

3.27.21.8 :TRIGger:IIC:DIRection

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.*/
```

3.27.21.9 :TRIGger:IIC:DBYTes

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.*/
```

3.27.21.10 :TRIGger:IIC:DATA

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 data bytes. You can send the *:TRIGger:IIC:DBYTes* command to set the data bytes. The maximum byte length

can be set to 5, that is, 40-bit binary data. Therefore, the range of <data> is from 0 to 2^{40}-1.

Return Format

The query returns an integer.

Example

```
:TRIGger:IIC:DATA 64    /*Sets the data value to 64.*/
:TRIGger:IIC:DATA?      /*The query returns 64.*/
```

3.27.21.11 :TRIGger:IIC:CURRbit

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* 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. That is, the oscilloscope triggers on the 9th bit of
I2C trigger data.*/
:TRIGger:IIC:CURRbit?      /*The query returns 8.*/
```

3.27.21.12 :TRIGger:IIC:CODE

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* 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.*/
```

3.27.22 :TRIGger:SPI

In SPI trigger, after the CS or timeout condition is satisfied, the oscilloscope triggers when the specified data is found. When using SPI trigger, you need to specify the CLK clock sources and MISO data sources.

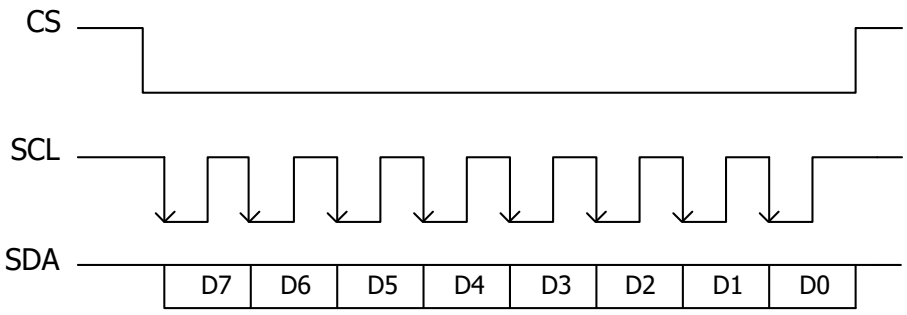


Figure 3.9 Sequential Chart of SPI Bus

3.27.22.1 :TRIGger:SPI:CLK

Syntax

:TRIGger:SPI:CLK <*source*>

:TRIGger:SPI:CLK?

Description

Sets or queries the channel source of the clock line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:CLK CHANnel3   /*Sets the channel source of the data
line of SPI trigger to CHANnel3.*/
:TRIGger:SPI:CLK?           /*The query returns CHAN3.*/
```

3.27.22.2 :TRIGger:SPI:SCL

Syntax

:TRIGger:SPI:SCL <*source*>

:TRIGger:SPI:SCL?

Description

Sets or queries the channel source of the clock line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:SCL CHANnel1    /*Sets the channel source of the clock
line to CHANnel1.*/
:TRIGger:SPI:SCL?            /*The query returns CHAN1.*/
```

3.27.22.3 :TRIGger:SPI:CLEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.*/
```

3.27.22.4 :TRIGger:SPI:SLOPe

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 data on the rising edge of the clock.
- **NEGative:** samples the 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.*/
```

3.27.22.5 :TRIGger:SPI:MISO

Syntax

:TRIGger:SPI:MISO <*source*>

:TRIGger:SPI:MISO?

Description

Sets or queries the channel source of the data line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8| D9|D10|D11|D12|D13|D14|D15| CHANnel1|CHANnel2|CHANnel3| CHANnel4}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:MISO CHANnel3 /*Sets the channel source of the data
line of SPI trigger to CHANnel3.*/
:TRIGger:SPI:MISO? /*The query returns CHAN3.*/
```

3.27.22.6 :TRIGger:SPI:SDA

Syntax

:TRIGger:SPI:SDA <*source*>

:TRIGger:SPI:SDA?

Description

Sets or queries the channel source of the data line of SPI trigger.

Parameter

Name	Type	Range	Default
<source>	Discrete	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:SPI:SDA CHANnel2 /*Sets the channel source of the data
line of SPI trigger to CHANnel2.*/
:TRIGger:SPI:SDA? /*The query returns CHAN2.*/
```

3.27.22.7 :TRIGger:SPI:DLEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 channel of SPI trigger 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.*/
```

3.27.22.8 :TRIGger:SPI:WHEN

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

- CS: if the CS signal is valid, the oscilloscope will trigger when the data (SDA) satisfying the trigger conditions is found.
- TIMeout: the oscilloscope starts to search for the data (MISO) on which to trigger after the clock signal (CLK) stays in the idle state for a specified period of time.

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.*/
```

3.27.22.9 :TRIGger:SPI:CS

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel3

Remarks

This command is valid only when the trigger condition of SPI is set to CS. You can send *:TRIGger:SPI:WHEN* to set or query the trigger condition of SPI trigger.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.27.22.10 :TRIGger:SPI:SLEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	0 V

Remarks

- This setting command is valid only when the trigger condition of SPI trigger is CS. You can run the *:TRIGger:SPI:WHEN* command to set or query the trigger condition of SPI trigger.
- 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.*/
```

3.27.22.11 :TRIGger:SPI:MODE

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

This setting command is only valid in CS mode. You can send *:TRIGger:SPI:WHEN* to set or query the trigger condition of SPI trigger.

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.*/
```

3.27.22.12 :TRIGger:SPI:TIMeout

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

This setting command is valid only when the trigger condition of SPI trigger is Timeout. You can run the *:TRIGger:SPI:WHEN* command to set or query the trigger condition of SPI trigger.

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.*/
```

3.27.22.13 :TRIGger:SPI:WIDTh

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.*/
```

3.27.22.14 :TRIGger:SPI:DATA

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. You can send the *:TRIGger:SPI:WIDTh* command to set or query the 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.*/
```

3.27.22.15 :TRIGger:SPI:CURRbit

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* 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
data to 8. That is, the oscilloscope triggers on the 9th bit of
SPI trigger data.*/
:TRIGger:SPI:CURRbit?      /*The query returns 8.*/
```

3.27.22.16 :TRIGger:SPI:CODE

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* 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.*/
```

3.27.23 :TRIGger:CAN

The :TRIGger:CAN commands are used to set relevant parameters for the CAN trigger.

This oscilloscope can trigger on the start of a frame, end of a frame, frame of the specified type (e.g. Remote, Overload, Data, etc.), or error frame of the specified type (e.g. Answer Error, Check Error, Format Error, etc.) of the CAN signal.

The data frame format of the CAN bus is as shown in the figure below.

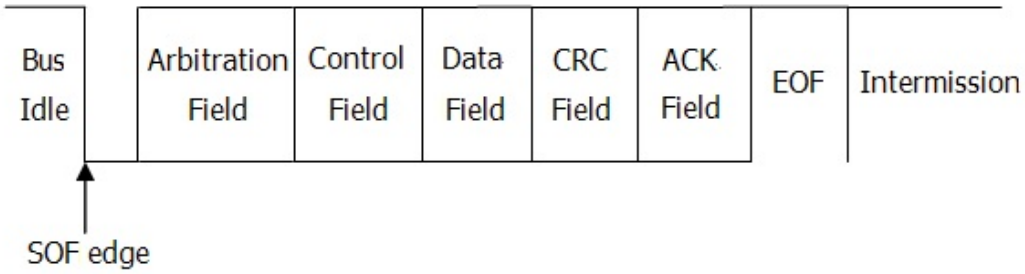


Figure 3.10 Data Frame Format of the CAN Bus

3.27.23.1 :TRIGger:CAN:BAUD

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

N/A

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.*/
```

3.27.23.2 :TRIGger:CAN:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:CAN:SOURce CHANnel2 /*Sets the trigger source to
CHANnel2.*/
:TRIGger:CAN:SOURce? /*The query returns CHAN2.*/
```

3.27.23.3 :TRIGger:CAN:STYPe

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.*/
```

3.27.23.4 :TRIGger:CAN:WHEN

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 ERB.

Example

```
:TRIGger:CAN:WHEN EOF    /*Sets the trigger condition to EOF.*/
:TRIGger:CAN:WHEN?         /*The query returns EOF.*/
```

3.27.23.5 :TRIGger:CAN:SPOint

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.*/
```

3.27.23.6 :TRIGger:CAN:EXTended

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	{0|1|OFF|ON|FALSe|TRUE}	0|OFF|FALSe

Remarks

- **0|OFF|FALSe:** disables the extended ID.
- **1|ON|TRUE:** enables the extended ID.

To set or query the trigger condition of CAN trigger, send the *:TRIGger:CAN:WHEN* command.

Return Format

The query returns 0 or 1.

Example

```
:TRIGger:CAN:EXTended ON  /*Enables the extended ID.*/
:TRIGger:CAN:EXTended    /*The query returns 1.*/
```

3.27.23.7 :TRIGger:CAN:DEFine

Syntax

:TRIGger:CAN:DEFine <*type*>

:TRIGger:CAN:DEFine?

Description

Sets to define ID or Data when the trigger condition of CAN trigger is set to Data&ID; queries to define ID or Data when the trigger condition of CAN trigger is set to Data&ID.

Parameter

Name	Type	Range	Default
<type>	Discrete	{DATA|ID|FALSe|TRUE}	DATA|FALSe

Remarks

- **DATA|FALSe:** sets Define to Data.
- **ID|TRUE:** sets Define to ID.

To set or query the trigger condition of CAN trigger, send the *:TRIGger:CAN:WHEN* command.

Return Format

The query returns DATA, ID, FALS, or TRUE.

Example

```
:TRIGger:CAN:DEFine ID /*Sets to define ID.*/
:TRIGger:CAN:DEFine? /*The query returns ID.*/
```

3.27.23.8 :TRIGger:CAN:DWIDth

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.*/
```

3.27.23.9 :TRIGger:CAN:DATA

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 data bytes. You can send the *:TRIGger:IIC:DBYTes* command to set the data bytes. 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.

This command is valid when the trigger condition is set to "Frame Data" or "Data & ID" (send *:TRIGger:CAN:WHEN* command to set or query).

- When the trigger condition is "Frame Data", the setting command is used to set the data value.
- When the trigger condition is "Data & ID", the value to be set by sending the setting command is based on the *:TRIGger:CAN:DEFine* command.
  - When Define is set to "ID", the setting command is used to set the ID value.
  - When Define is set to "Data", the setting command is used to set the data value.

Return Format

The query returns an integer.

Example

```
:TRIGger:CAN:DATA 100    /*Sets the data value of CAN trigger to
100.*/
:TRIGger:CAN:DATA?      /*The query returns 100.*/
```

3.27.23.10 :TRIGger:CAN:CURRbit

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* 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
data to 8. That is, the oscilloscope triggers on the 9th bit of
CAN trigger data.*/
:TRIGger:SPI:CURRbit?      /*The query returns 8.*/
```

3.27.23.11 :TRIGger:CAN:CODE

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* 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.*/
```

3.27.23.12 :TRIGger:CAN:LEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 CAN trigger 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.*/
```

3.27.24 :TRIGger:LIN

The :TRIGger:LIN commands are used to set relevant parameters for the LIN trigger.

The oscilloscope can trigger on the sync field of LIN signal, and can also trigger on the specified identifier, data, or frame.

The data frame format of the LIN bus is as shown in the figure below.

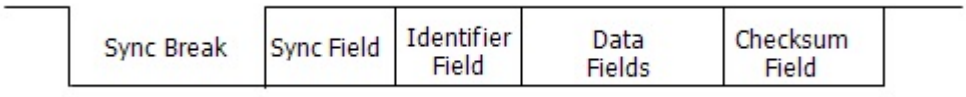


Figure 3.11 Data Frame Format of the LIN Bus

3.27.24.1 :TRIGger:LIN:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:LIN:SOURce CHANnel2    /*Sets the trigger source to
CHANnel2.*/
:TRIGger:LIN:SOURce?             /*The query returns CHAN2.*/
```

3.27.24.2 :TRIGger:LIN:LEVel

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.

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.*/
```

3.27.24.3 :TRIGger:LIN:STANdard

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.*/
```

3.27.24.4 :TRIGger:LIN:BAUD

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

N/A

Return Format

The query returns an integer ranging from 1000 to 20000000. The unit is bps.

Example

```
:TRIGger:LIN:BAUD 19200          /*Sets the baud rate of LIN trigger
to 19.2 kbps.*/
:TRIGger:LIN:BAUD?                /*The query returns 19200.*/
```

3.27.24.5 :TRIGger:LIN:SAMPlepoint

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.*/
```

3.27.24.6 :TRIGger:LIN:WHEN

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.*/
```

3.27.24.7 :TRIGger:LIN:ERRor

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:** check sum error.

Return Format

The query returns SYNC, ID, or CHEC.

Example

```
:TRIGger:LIN:ERRor ID          /*Sets the LIN trigger error type to
ID.*/
:TRIGger:LIN:ERRor?               /*The query returns ID.*/
```

3.27.24.8 :TRIGger:LIN:ID

Syntax

:TRIGger:LIN:ID <*id*>

:TRIGger:LIN:ID?

Description

Sets or queries the ID value of LIN trigger when the trigger condition is "Data & ID".

Parameter

Name	Type	Range	Default
<id>	Integer	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.*/
```

3.27.24.9 :TRIGger:LIN:DATA

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.e. 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.*/
```

3.27.24.10 :TRIGger:LIN:CURRbit

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* 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
data to 8. That is, the oscilloscope triggers on the 9th bit of
LIN trigger data.*/
:TRIGger:SPI:CURRbit?      /*The query returns 8.*/
```

3.27.24.11 :TRIGger:LIN:CODE

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* 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.*/
```

3.27.25 :TRIGger:FLEXray (Option)

The :TRIGger:FLEXray commands are used to set the relevant parameters for the FlexRay trigger.

The oscilloscope can trigger on the specified frame, symbol, error, or position of the FlexRay bus. FlexRay is a type of differential serial bus configured with three consecutive segments (i.e. packet header, payload, and packet trailer). Its data transmission rate is up to 10 Mb/s. Each frame contains a static segment and a dynamic segment, and ends with the bus idle time.

Its format is as shown in the figure below.

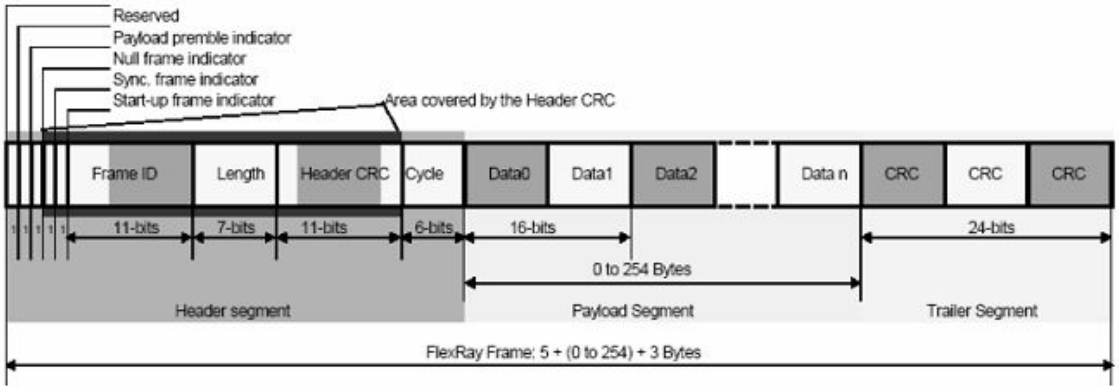


Figure 3.12 Frame Format of FlexRay Bus

NOTE

The :TRIGger:FLEXray commands are only valid when you have installed the MHO900-FlexA option.

3.27.25.1 :TRIGger:FLEXray:BAUD

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.*/
```

3.27.25.2 :TRIGger:FLEXray:POS

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.*/
```

3.27.25.3 :TRIGger:FLEXray:ERRor

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.*/
```

3.27.25.4 :TRIGger:FLEXray:SYMBol

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>	Discrete	{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.*/
```

3.27.25.5 :TRIGger:FLEXray:FRAMe

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.*/
```

3.27.25.6 :TRIGger:FLEXray:DEFine

Syntax

:TRIGger:FLEXray:DEFine <*define*>

: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
<define>	Discrete	{CYCLe|ID|TRUE|FALSe}	ID|FALSe

Remarks

- **CYCLe|TRUE:** indicates Cyc Count.
- **ID|FALSe:** indicates Frame ID.

Return Format

The query returns CYCL or ID.

Example

```
:TRIGger:FLEXray:DEFine CYCLe /*Sets to use Cyc Count to define
the trigger condition of FlexRay trigger.*/
:TRIGger:FLEXray:DEFine?      /*The query returns CYCL.*/
```

3.27.25.7 :TRIGger:FLEXray:IDCmp

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.*/
```

3.27.25.8 :TRIGger:FLEXray:CYCComp

Syntax

:TRIGger:FLEXray:CYCComp <*cycmax*>

:TRIGger:FLEXray:CYCComp?

Description

Sets or queries the CYC comparison condition of FlexRay trigger when the trigger condition is set to Frame.

Parameter

Name	Type	Range	Default
<cycmax>	Discrete	{EQUal|NOTequal|GREaterthan|LESSthan|INRange|OUTRange}	EQU

Remarks

- **EQUal:** =
- **NOTequal:** ≠
- **GREaterthan:** >
- **LESSthan:** <
- **INRange:** ><
- **OUTRange:** <>

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.*/
:TRIGger:FLEXray:CYCComp?       /*The query returns NOT.*/
```

3.27.25.9 :TRIGger:FLEXray:MAXCy

Syntax

:TRIGger:FLEXray:MAXCy <*cycmax*>

:TRIGger:FLEXray:MAXCy?

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:MINCy* 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:MAXCy 10          /*Sets the upper limit of cycle
count of FlexRay trigger to 10.*/
:TRIGger:FLEXray:MAXCy?                  /*The query returns 10.*/
```

3.27.25.10 :TRIGger:FLEXray:MINCy

Syntax

:TRIGger:FLEXray:MINCy <*cycmin*>

:TRIGger:FLEXray:MINCy?

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:MAXCy* 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:MINCy 10          /*Sets the lower limit of cycle
count of FlexRay trigger to 10.*/
:TRIGger:FLEXray:MINCy?                  /*The query returns 10.*/
```

3.27.25.11 :TRIGger:FLEXray:MAXid

Syntax

:TRIGger:FLEXray:MAXid <*idmax*>

:TRIGger:FLEXray:MAXid?

Description

Sets or queries the ID upper limit of FlexRay trigger.

Parameter

Name	Type	Range	Name
<idmax>	Integer	0 to 1,023	0

Remarks

N/A

Return Format

The returns an integer ranging from 0 to 1023.

Example

```
:TRIGger:FLEXray:MAXid 100          /*Sets the ID upper limit of
FlexRay trigger to 100.*/
:TRIGger:FLEXray:MAXid?                    /*The query returns 10.*/
```

3.27.25.12 :TRIGger:FLEXray:MINid

Syntax

:TRIGger:FLEXray:MINid <*idmin*>

:TRIGger:FLEXray:MINid?

Description

Sets or queries the ID Min value of FlexRay trigger when the trigger condition is set to Frame or Symbol.

Parameter

Name	Type	Range	Name
<idmin>	Integer	0 to 1,023	0

Remarks

N/A

Return Format

The returns an integer ranging from 0 to 1023.

Example

```
:TRIGger:FLEXray:MINid 10        /*Sets the ID Min value of
FlexRay trigger to 10 when the trigger condition is set to Frame
or Symbol.*/
:TRIGger:FLEXray:MINid?                  /*The query returns 10.*/
```

3.27.25.13 :TRIGger:FLEXray:CH

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.*/
```

3.27.25.14 :TRIGger:FLEXray:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.27.25.15 :TRIGger:FLEXray:WHEN

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, SYMBol, ERR, or TSS.

Example

```
:TRIGger:FLEXray:WHEN TSS          /*Sets the trigger condition to
TSS.*/
:TRIGger:FLEXray:WHEN?                    /*The query returns TSS.*/
```

3.27.25.16 :TRIGger:FLEXray:LEVel

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: (-4.5 x VerticalScale - OFFSet) to (4.5 x VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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.*/
```

3.27.26 :TRIGger:IIS (Option)

The :TRIGger:IIS commands are used to set the relevant parameters for the I2S trigger.

In I2S trigger, the oscilloscope searches for the specified data value and take it as the condition for identifying the trigger. You need to specify the serial clock line (SCLK, 1 pulse is found on the clock line once 1 bit of digital audio data is sent), frame clock line (WS, used for switch the audio channel data), and serial data line (SDA, used for transmit audio data represented in binary (2's complement)).

Below is the sequential chart of I2S bus.

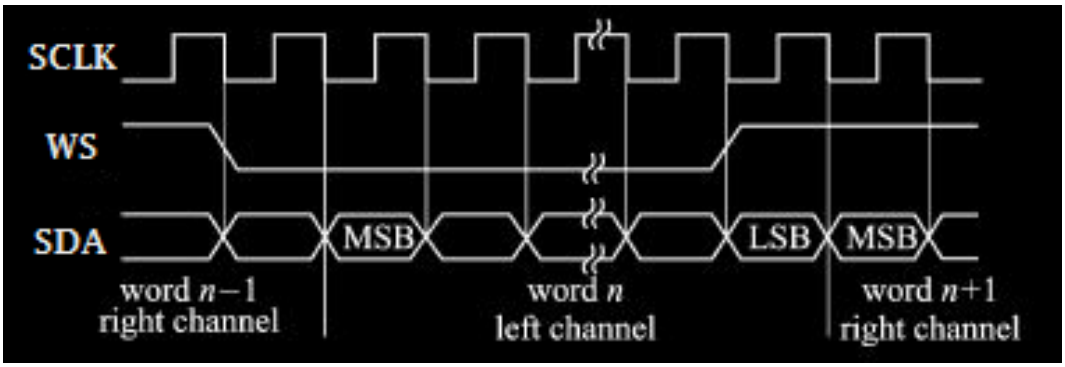


Figure 3.13 Sequential Chart of I2S Bus

NOTE

The :TRIGger:IIS commands are only valid when you have installed the MHO900-AUDIOA option.

3.27.26.1 :TRIGger:IIS:ALIGnment

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:** data transmission (MSB first) begins at the second 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.*/
```

3.27.26.2 :TRIGger:IIS:CLEVel

Syntax

:TRIGger:IIS:CLEVel <*level*>

:TRIGger:IIS:CLEVel?

Description

Sets or queries the trigger level of the clock line source in I2S trigger. The unit is V.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-4.5 × VerticalScale - OFFSet) to (4.5 × VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 of the clock line source in I2S trigger in scientific notation.

Example

```
:TRIGger:IIS:CLEVel 0.01    /*Sets the trigger level of trigger
level of the clock line source in I2S trigger to 0.01 V.*/
:TRIGger:IIS:CLEVel?        /*The query returns 1.000000E-2.*/
```

3.27.26.3 :TRIGger:IIS:SLEVel

Syntax

:TRIGger:IIS:SLEVel <*level*>

:TRIGger:IIS:SLEVel?

Description

Sets or queries the trigger level of the frame clock line source in I2S trigger. The unit is V.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-4.5 × VerticalScale - OFFSet) to (4.5 × VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 of the frame clock line source in I2S trigger in scientific notation.

Example

```
:TRIGger:IIS:SLEVel 0.01    /*Sets the trigger level of the frame
clock line source in I2S trigger to 10 mV.*/
:TRIGger:IIS:SLEVel?        /*The query returns 1.000000E-2.*/
```

3.27.26.4 :TRIGger:IIS:DLEVel

Syntax

:TRIGger:IIS:DLEVel <*level*>

:TRIGger:IIS:DLEVel?

Description

Sets or queries the trigger level of the data line source in I2S trigger. The unit is V.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-4.5 × VerticalScale - OFFSet) to (4.5 × VerticalScale - OFFSet) Digital channel: -15 V to 15 V	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 of data line source in I2S trigger in scientific notation.

Example

```
:TRIGger:IIS:DLEVel 0.16    /*Sets the trigger level of data line
source in I2S trigger to 160 mV.*/
:TRIGger:IIS:DLEVel?        /*The query returns 1.600000E-1.*/
```

3.27.26.5 :TRIGger:IIS:UWIDth

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* 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.*/
```

3.27.26.6 :TRIGger:IIS:WIDTh

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.*/
```

3.27.26.7 :TRIGger:IIS:DMIN

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* command.

After running this command, you can send the *:TRIGger:IIS:CODE* 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.*/
```

3.27.26.8 :TRIGger:IIS:DMAX

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* command.

After running this command, you can send the *:TRIGger:IIS:CODE* 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.*/
```

3.27.26.9 :TRIGger:IIS:CODE

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* or *:TRIGger:IIS:DMIN* 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.*/
```

3.27.26.10 :TRIGger:IIS:CLOCk:SLOPe

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.*/
```

3.27.26.11 :TRIGger:IIS:SOURce:CLOCk

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel1

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIS:SOURce:CLOCk CHANnel2        /*Sets clock source to
CHANnel2.*/
:TRIGger:IIS:SOURce:CLOCk?          /*The query returns CHAN2.*/
```

3.27.26.12 :TRIGger:IIS:SOURce:DATA

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel3

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:IIS:SOURce:DATA CHANnel2        /*Sets data source to
CHANnel2.*/
:TRIGger:IIS:SOURce:DATA?          /*The query returns CHAN2.*/
```

3.27.26.13 :TRIGger:IIS:SOURce:WSELect

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|CHANnel1|CHANnel2|CHANnel3|CHANnel4}	CHANnel2

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, 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.*/
```

3.27.26.14 :TRIGger:IIS:WHEN

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.*/
```

3.27.26.15 :TRIGger:IIS:AUDio

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.*/
```

3.27.26.16 :TRIGger:IIS:DATA

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, that is, 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.*/
```

3.27.27 :TRIGger:M1553 (Option)

The :TRIGger:M1553 commands are used to set the relevant parameters for the M1553 trigger.

1553B is the abbreviation for the MIL-STD-1553 bus. This oscilloscope can trigger on the sync field of 1553B bus, and can also trigger on the specified data word, command word, status word, or error type.

The command word, data word, and status word format of the 1553B bus is as shown in the figure below.

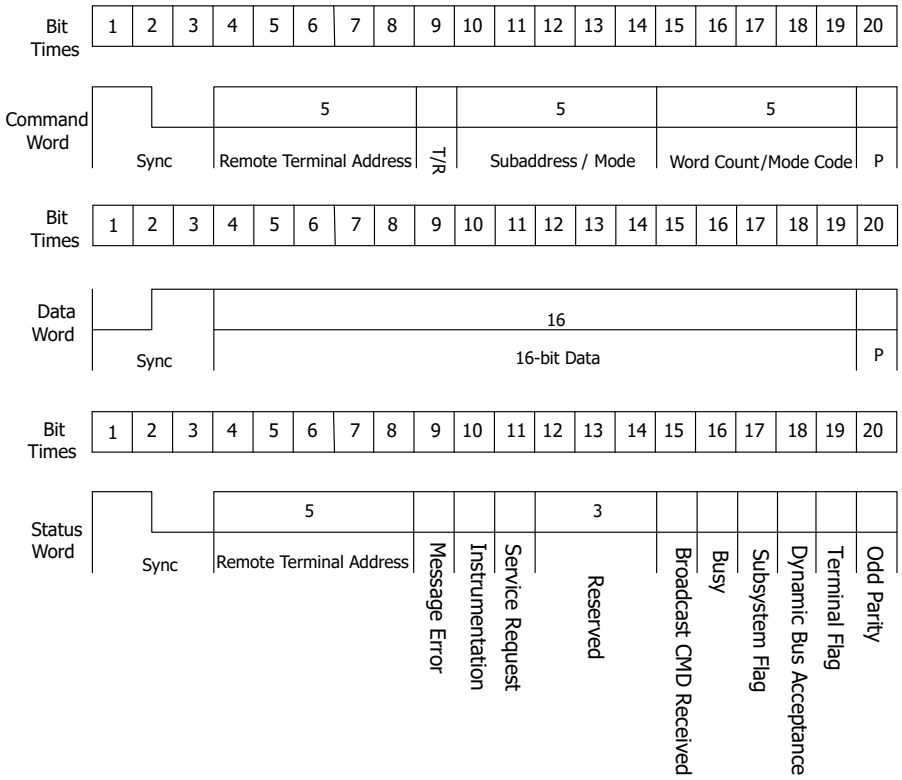


Figure 3.14 Formats of the Command Word, Data Word, and Status Word of the 1553B Bus

NOTE

The :TRIGger:M1553 commands are only valid when you have installed the MHO900-AEROA option.

3.27.27.1 :TRIGger:M1553:SOURce

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	{D0|D1|D2|D3|D4|D5|D6|D7|D8|D9|D10|D11|D12|D13|D14|D15|	CHANnel1

Name	Type	Range	Default
		CHANnel1|CHANnel2|CHANnel3|CHANnel4}	

Remarks

The digital channels are available to choose when the logic analyzer probe is connected.

Return Format

The query returns D0, D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12, D13, D14, D15, CHAN1, CHAN2, CHAN3, or CHAN4.

Example

```
:TRIGger:M1553:SOURce CHANnel2          /*Sets the trigger source to
CHANnel2.*/
:TRIGger:M1553:SOURce?                 /*The query returns CHAN2.*/
```

3.27.27.2 :TRIGger:M1553:WHEN

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.*/
```

3.27.27.3 :TRIGger:M1553:POLarity

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.*/
```

3.27.27.4 :TRIGger:M1553:WINDow

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.*/
```

3.27.27.5 :TRIGger:M1553:SYNC

Syntax

:TRIGger:M1553:SYNC <*sync*>

:TRIGger:M1553:SYNC?

Description

Sets or queries the sync type of M1553 trigger when the trigger condition is Sync.

Parameter

Name	Type	Range	Default
<sync>	Discrete	{DATA|STATus|ALL}	-

Remarks

- **DATA:** indicates data frame sync.
- **STATus:** indicates commands/status sync.
- **ALL:** indicates the all frame sync.

Return Format

The query returns DATA, STATus, or ALL.

Example

```
:TRIGger:M1553:SYNC DATA    /*Sets the sync type of M1553 trigger
to DATA.*/
:TRIGger:M1553:SYNC?           /*The query returns DATA.*/
```

3.27.27.6 :TRIGger:M1553:ERRor

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.*/
```

3.27.27.7 :TRIGger:M1553:DATComp

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.*/
```

3.27.27.8 :TRIGger:M1553:DATValue

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.*/
```

3.27.27.9 :TRIGger:M1553:DMIN

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* 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.*/
```

3.27.27.10 :TRIGger:M1553:DMAX

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* 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.*/
```

3.27.27.11 :TRIGger:M1553:DRTA

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.*/
```

3.27.27.12 :TRIGger:M1553:DBIT

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* 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.*/
```

3.27.27.13 :TRIGger:M1553:CODE

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* 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* 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* 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.*/
```

3.27.27.14 :TRIGger:M1553:ALEVel

Syntax

:TRIGger:M1553:ALEVel <*level*>

:TRIGger:M1553:ALEVel?

Description

Sets or queries Level A of M1553 trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: Lower limit to (4.5 × VerticalScale - OFFSet) Digital channel: Lower limit to 15 V	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 Level A of M1553 trigger in scientific notation.

Example

```
:TRIGger:M1553:ALEVel 0.16    /*Sets Level A of M1553 trigger to
160 mV.*/
:TRIGger:M1553:ALEVel?        /*The query returns 1.600000E-1.*/
```

3.27.27.15 :TRIGger:M1553:BLEVel

Syntax

:TRIGger:M1553:BLEVel <*level*>

:TRIGger:M1553:BLEVel?

Description

Sets or queries Level B of 1553B trigger. The unit is the same as that of the current amplitude.

Parameter

Name	Type	Range	Default
<level>	Real	Analog channel: (-4.5 × VerticalScale - OFFSet) to upper limit Digital channel: -15 V 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 Level B of 1553B trigger in scientific notation.

Example

```
:TRIGger:M1553:BLEVel 0.05   /*Sets Level B of 1553B trigger to 50
mV.*/
:TRIGger:M1553:BLEVel?      /*The query returns 5.000000E-2.*/
```

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 3.15* and *Figure 3.16* .

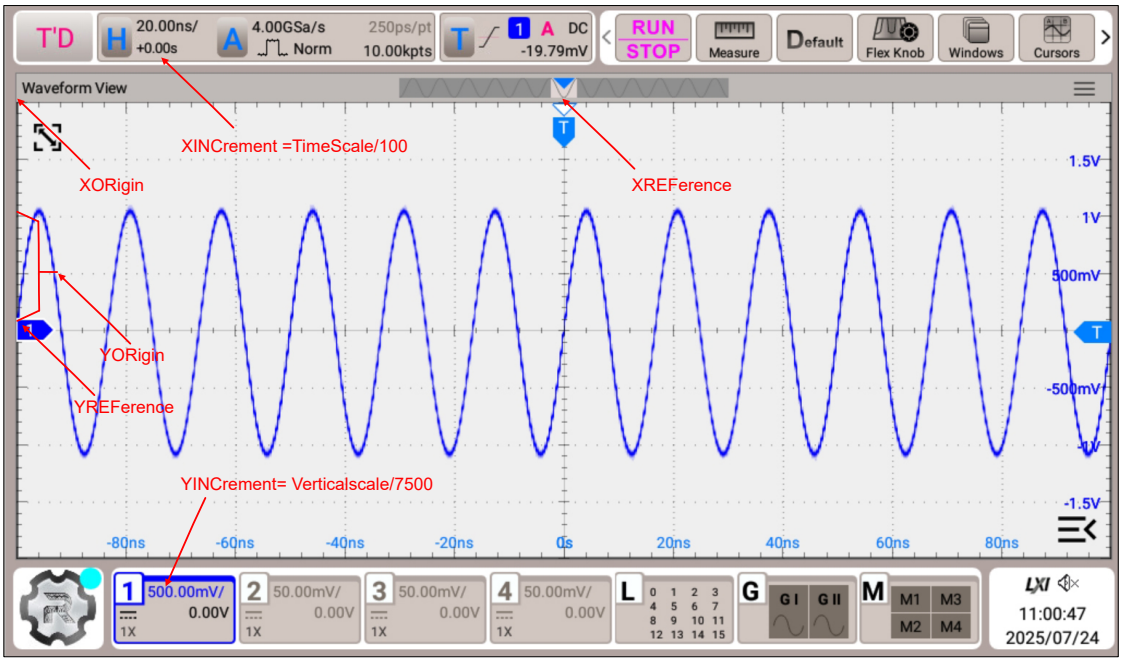


Figure 3.15 Parameter Definitions in NORMAL Mode

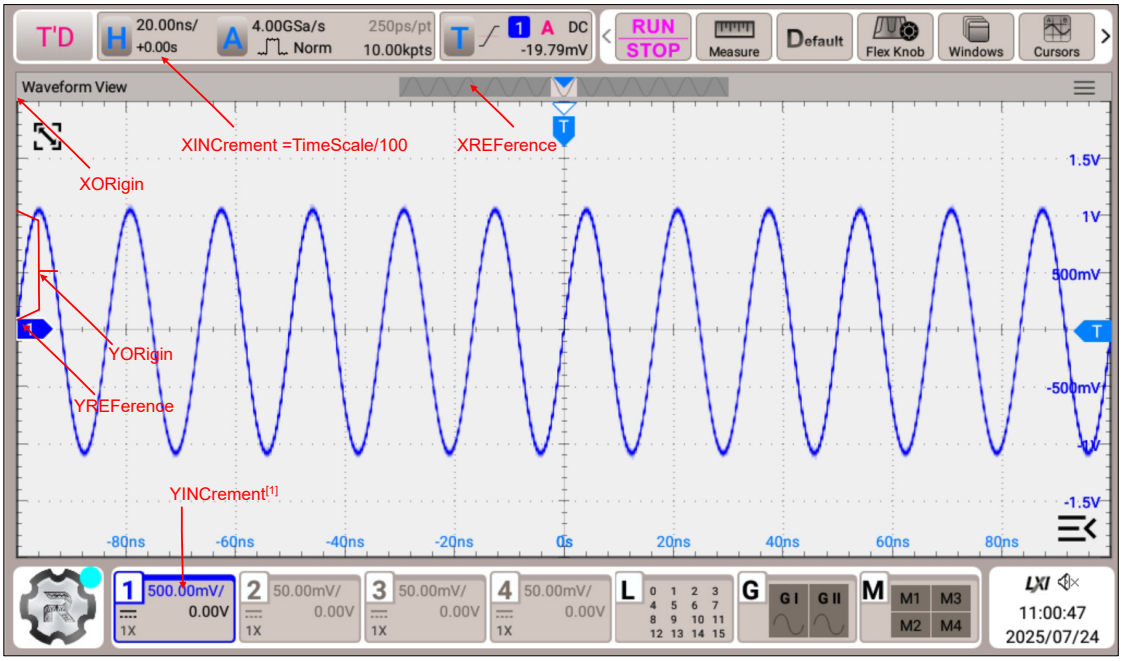


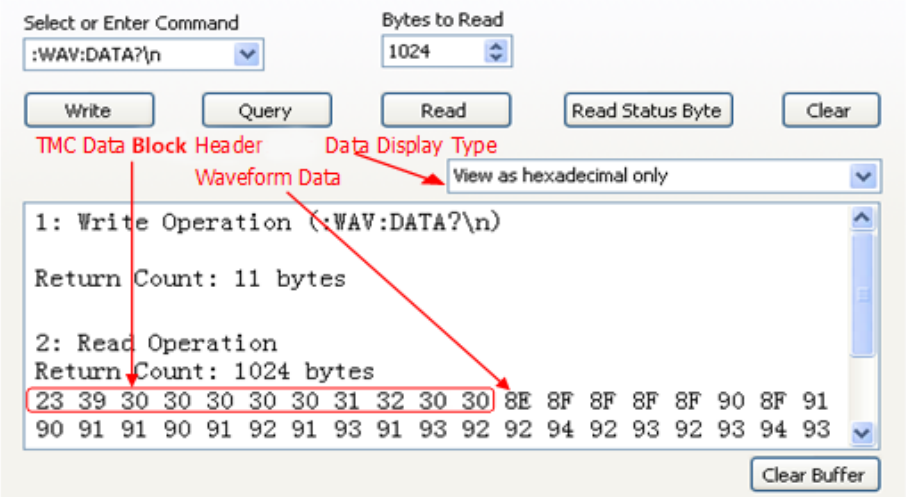
Figure 3.16 Parameter Definitions in RAW Mode

NOTE

[1]: In RAW mode, YINCrement and Verticalscale of the memory waveforms are related to the currently selected Verticalscale.

Waveform data reading

- **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, that is, 1,000 bytes.
- **ASCii format:** The read data format is waveform data points + end identifier. The waveform data point 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*.



Related Command

:WAVeform:MODE

:WAVeform:YINCrement?

:WAVeform:YORigin?

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. You can send the *:WAVeform:MODE* command to set or query the reading mode of the waveform data.

- **NORMal:** 1 to 1,000
- **RAW:** 1 to the current maximum memory depth
- **MAXimum:** In RUN state: 1 to the number of effective points on the current screen; In STOP state: 1 to the number of effective points in the current memory

Return Format

The query returns the number of waveform points in integer.

Example

```
:WAVeform:POINts 100   /*Sets the number of waveform points to be
read to 100.*/
:WAVeform:POINts?   /*The query returns 100.*/
```

3.28.5 :WAVeform:DATA?

Syntax

:WAVeform:DATA?

Description

Reads the waveform data.

Parameter

N/A

Remarks

Procedures of reading the waveform data from the screen:

```
:WAV:SOUR CHAN1    /*Sets the channel source to CHANnel1.*/
: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 from 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 CHANnel1.*/
: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 120,000th waveform point (last point).*/
```

Return Format

The return format is related to the return format of the currently selected waveform data (*:WAVeform:FORMat*). For detailed operations, refer to descriptions in *Waveform data reading*.

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/7500.
- In RAW mode, YINCrement and Verticalscale of the memory waveforms are related to the currently selected Verticalscale.
- In MAX mode, YINCrement = VerticalScale/7500 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	Refer to *Remarks*	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.

- In Normal mode, the range is from 1 to 1,000.
- In Max mode, when the oscilloscope is in RUN state, its range is from 1 to 1,000; when the oscilloscope is in STOP state, its range is from 1 to current maximum memory depth.
- In Raw mode, the range is from 1 to the current maximum memory depth.

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	Refer to *Remarks*	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.

- In Normal mode, the range is from 1 to 1,000.
- In Max mode, when the oscilloscope is in RUN state, its range is from 1 to 1,000; when the oscilloscope is in STOP state, its range is from 1 to current maximum memory depth.
- In Raw mode, the range is from 1 to the current maximum memory depth.

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 Application Examples

This chapter provides some application examples of the SCPI commands. The main functions of the instrument can be realized through a series of SCPI commands.

NOTE

- The range of certain parameters for different models may be different. Therefore, you need to adjust the parameter range for the model that you use if necessary.
- Before using the examples in this chapter, please select the desired communication interface (USB or LAN) and make correct connections. In addition, you should install the PC software that can be used to send commands.
- In each example, every command is followed by contents enclosed by two slashes ("/*" and "*/"). They are the descriptions of the command and not part of the command, which help you understand the command better.

4.1 To Configure Basic Parameters

Set the Channel Parameters

Example description: enable CH1, set the vertical scale to 0.1 V/div, and set the coupling mode to AC.

Method

```
:CHANnel1:DISPlay ON        /*Enables CHANnel1.*/
:CHANnel1:SCALe 0.1         /*Sets the vertical scale to 0.1 V/div
for CH1.*/
:CHANnel1:COUPling AC   /*Sets the coupling mode to AC for CH1.*/
```

Set the Horizontal Parameters

Example description: set the memory depth to 1M and the horizontal time base to 200 μs/div.

Method

```
:ACQuire:MDEPth 1M              /*Sets the memory depth to 1M.*/
:TIMebase:MODE MAIN             /*Sets the horizontal time base mode
to MAIN.*/
:TIMebase:MAIN:SCALe 0.0002     /*Sets the main time base to 200 μs/
div.*/
```

Set the Trigger Parameters

Example description: set the triggers on the rising edge with the specified voltage threshold (160 mV).

Method

```
:TRIGger:MODE EDGE                /*Selects the edge trigger.*/
:TRIGger:EDGE:SOURce CHANnel2     /*Sets the trigger source to
```

```
CHANnel2.*/
:TRIGger:EDGE:SLOPe POSitive    /*Sets the edge type to rising
edge.*/
:TRIGger:EDGE:LEVel 0.16        /*Sets the trigger level to 160
mV.*/
:TRIGger:STATus?                /*Queries the current trigger
status.*/
```

4.2 To Set Measurements

Read the Statistical Peak-to Peak Value

Example description: reads the statistical peak-to-peak value for CH2.

Method

```
:MEASure:THReshold:TYPE PERCent   /*Sets the threshold type to
PERCent.*/
:MEASure:SETup:MAX 95             /*Sets the upper limit to 95%.*/
:MEASure:SETup:MID 89             /*Sets the middle value to 89%.*/
:MEASure:SETup:MIN 53             /*Sets the lower limit to 53%.*/
```

```
:MEASure:ITEM VPP,CHANnel2     /*Adds the CH2 waveform peak-to-peak
value measurement item./
:MEASure:ITEM? VPP,CHANnel2    /*Queries the current measurement
results of the CH2 waveform peak-to-peak value.*/
```

```
:MEASure:STATistic:RESet              /*Clears the history
statistics data and makes statistics again.*/
:MEASure:STATistic:ITEM VPP,CHANnel2
          /*Adds the CH2 waveform peak-to-peak value measurement
item.*/
:MEASure:STATistic:ITEM? CNT,VPP,CHANnel2
          /*Queries the count value of the measurement item. */
:MEASure:STATistic:ITEM? MAXimum,VPP,CHANnel2
          /*Queries the maximum value of the CH2 peak-to-peak
value.*/
:MEASure:STATistic:ITEM? MINimum,VPP,CHANnel2
          /*Queries the minimum value of the CH2 peak-to-peak
value.*/
:MEASure:STATistic:ITEM? AVERages,VPP,CHANnel2
          /*Queries the average value of the CH2 peak-to-peak
value.*/
:MEASure:STATistic:ITEM? DEViation,VPP,CHANnel2
          /*Queries the deviation of the CH2 peak-to-peak value.*/
```

Read the Duty Cycle

Example description: read the duty cycle of the periodic pulse at a set percentage.

Method

```
:MEASure:THReshold:TYPE PERCent   /*Sets the threshold type to
PERCent.*/
:MEASure:SETup:MAX 95             /*Sets the upper limit to 95%.*/
:MEASure:SETup:MID 89             /*Sets the middle value to 89%.*/
:MEASure:SETup:MIN 53             /*Sets the lower limit to 53%.*/
```

```
:MEASure:ITEM PDUTy,CHANnel2   /*Adds the CH2 periodic pulse duty
cycle measurement item.*/
```

```
:MEASure:ITEM? PDUTy,CHANnel2  /*Queries the current duty cycle
value for CH2.*/

:MEASure:STATistic:RESet   /*Clears the history statistics data and
makes statistics again.*/
:MEASure:STATistic:ITEM PDUTy,CHANnel2
                                    /*Adds the CH2 waveform duty
cycle measurement item.*/

:MEASure:STATistic:ITEM? CNT,PDUTy,CHANnel2
          /*Queries the count value of the measurement item. */
:MEASure:STATistic:ITEM? MAXimum,PDUTy,CHANnel2
          /*Queries the maximum value of the CH2 duty cycle.*/
:MEASure:STATistic:ITEM? MINimum,PDUTy,CHANnel2
          /*Queries the minimum value of the CH2 duty cycle.*/
:MEASure:STATistic:ITEM? AVERages,PDUTy,CHANnel2
          /*Queries the average value of the CH2 duty cycle.*/
:MEASure:STATistic:ITEM? DEViation,PDUTy,CHANnel2
          /*Queries the deviation of the CH2 duty cycle value.*/
```

Read the Rise Time

Example description: read the rise time using the set voltage threshold.

Method

```
:MEASure:THReshold:TYPE ABSolute  /*Sets the threshold type to
ABSolute.*/
:MEASure:SETup:MAX 0.15           /*Sets the threshold level upper
limit to 0.15 V.*/
:MEASure:SETup:MID 0              /*Sets the threshold level middle
value to 0 V.*/
:MEASure:SETup:MIN -0.15          /*Sets the threshold level lower
limit to -0.15 V.*/

:MEASure:ITEM RTIMe,CHANnel2           /*Adds the CH2 rise time
measurement item.*/
:MEASure:STATistic:ITEM? CURRent,RTIMe /*Queries the current
measurement value of the CH2 rise time.*/

:MEASure:STATistic:RESet               /*Clears the history
statistics data and makes statistics again.*/
:MEASure:STATistic:ITEM RTIMe,CHANnel2 /*Enables the CH2 rise time
statistic function.*/
:MEASure:STATistic:ITEM? CURRent,RTIMe /*Queries the current value
of the CH2 rise time.*/

:MEASure:STATistic:ITEM? CNT,RTIMe,CHANnel2
          /*Queries the count value of the measurement item. */
:MEASure:STATistic:ITEM? MAXimum,RTIMe,CHANnel2
          /*Queries the maximum value of the CH2 rise time.*/
:MEASure:STATistic:ITEM? MINimum,RTIMe,CHANnel2
          /*Queries the minimum value of the CH2 rise time.*/
:MEASure:STATistic:ITEM? AVERages,RTIMe,CHANnel2
          /*Queries the average value of the CH2 rise time.*/
:MEASure:STATistic:ITEM? DEViation,RTIMe,CHANnel2
          /*Queries the deviation of the CH2 rise time.*/
```

4.3 To Read the Waveform

Read ASCII Data

Example description: read ASCII data and transform the data to actual waveform data.

Method

```
:ACQuire:MDEPth 100k      /*Sets the memory depth to 100k.*/
:RUN       /*Sets the oscilloscope to "RUN" state and wait for 5
s.*/
:STOP      /*Sets the oscilloscope to "STOP" state (the
oscilloscope can only read the waveform data in internal memory
when in "STOP" state).*/
```

```
:WAV:SOUR CHAN1     /*Sets the channel source to CHANnel1.*/
:WAV:MODE RAW       /*Sets the waveform reading mode to RAW.*/
:WAV:FORM ASCii     /*Sets the return format of the waveform data
to ASCii.*/
```

```
:WAV:STAR 1          /*Sets the start point to read from to the 1st
point.*/
:WAV:STOP 100000     /*Sets the stop point to the 100,000th point
(the last point).*/
:WAV:DATA?           /*Reads the waveform data.*/
:WAVeform:PREamble?  /*The query returns all waveform parameters.*/
```

Read the Waveform Data in Internal Memory

Example description: reads the waveform data from the internal memory.

Method

```
:ACQuire:MDEPth 100k   /*Sets the memory depth to 100k.*/
:RUN    /*Sets the oscilloscope to "RUN" state and wait for 5 s.*/
:STOP   /*Set the oscilloscope to "STOP" state (the oscilloscope
can only read the waveform data in internal memory when in "STOP"
state).*/
```

```
:WAV:SOUR CHAN1       /*Sets the channel source to CHANnel1.*/
: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 to read from to the
1st waveform.*/
:WAVeform:STOP 100000  /*Sets the stop point to the 100,000th point
(the last point).*/
:WAV:DATA?             /*Reads the waveform data.*/
```

Read the Waveform Data on the Screen

Example description: reads the waveform data on the display.

Method

```
:WAV:SOUR CHAN1       /*Sets the channel source to CHANnel1.*/
: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.*/
```

5 Programming Examples

This chapter illustrates how to control the instrument by programming in LabVIEW, Visual Basic, and Visual C++. These examples are programmed based on Virtual Instrument Software Architecture (VISA) library.

RIGOL official website (*http://www.rigol.com*) provides the programming examples based on LabVIEWk, Visual Basic, Visual C++, Matlab, and Python. You can go to the official website and click **SUPPORT** > **Programming Demo** to obtain the programming examples.

5.1 Programming Preparations

Before programming, you need to prepare the following tasks:

You can log in to the RIGOL official website (*http://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 instrument communicates with the PC via the USB interface. Connect the USB Device interface on the rear panel of the instrument to the PC by using the USB cable. After the instrument is properly connected to the PC, power on the instrument to start it.

The following parts will make a detailed introduction about the programming examples in LabVIEW, Visual Basic, and Visual C++.

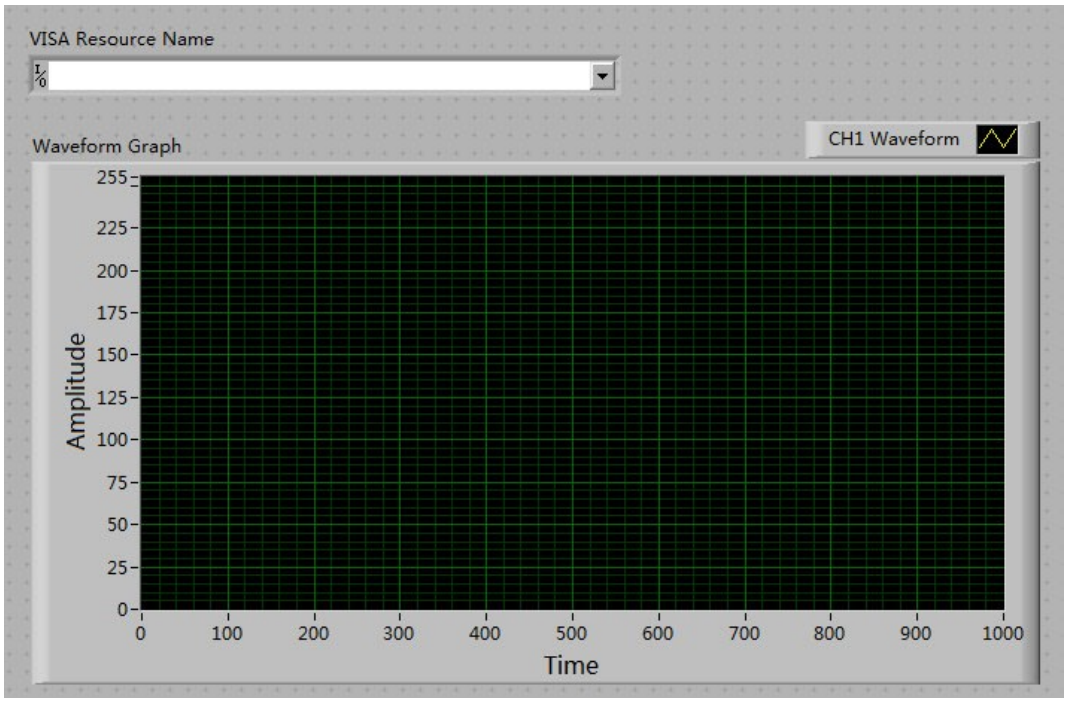
5.2 LabVIEW Programming Example

Program used in this example: LabVIEW2010

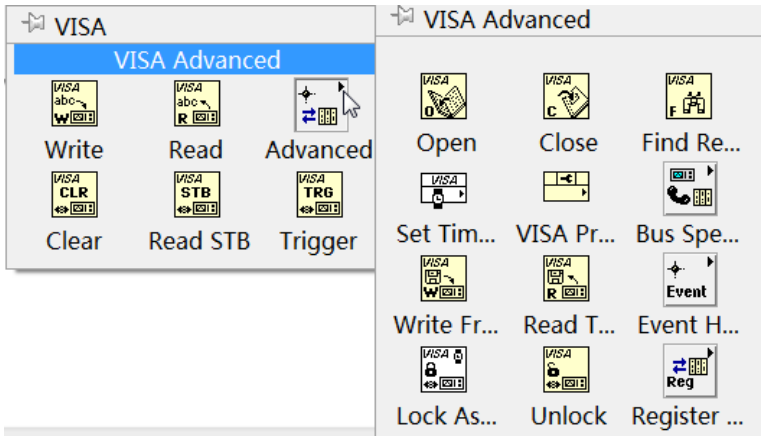
Function realized in this example: reading the waveform data of CH1 on the screen.

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.



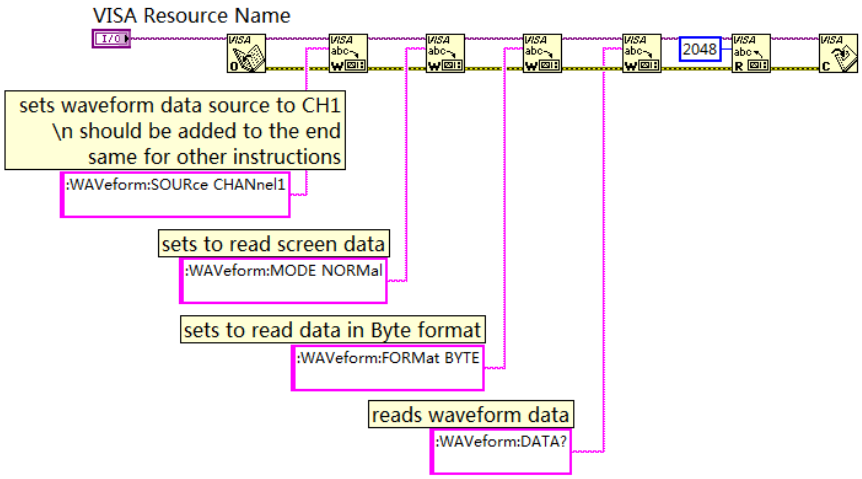
3. Open the Block Diagram panel. Click **Instrument I/O** > **VISA**. Add the following functions: VISA Open, VISA Read, VISA Write, and VISA Close.



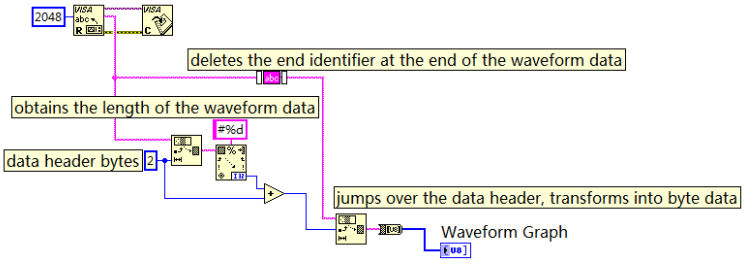
4. Connect the VISA resource name with the VISA Open. Then, connect the VISA resource name outputs of all the functions with the VISA resource name and connect the error output with the error input, as shown in the figure below.

5. 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

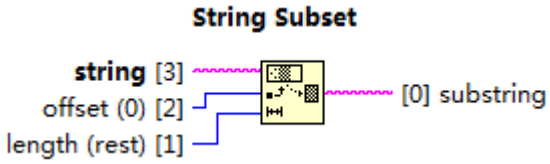
VISA Read function which requires users to input the total number of bytes to be read. In this example, the total number of bytes of waveform data to be read is less than 2048. Use the VISA Close function to close the VISA resource after the VISA operation is finished.



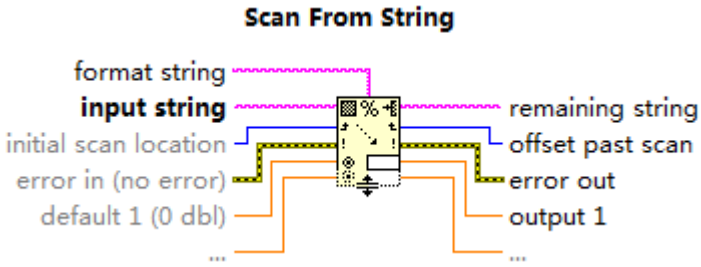
6. The data format is TMC data block header + waveform data + terminator. The TMC header is in #NXXXXXX format; wherein, # is the TMC header identifier; N following # represents the length of the waveform data; the length of the waveform data points is expressed in ASCII strings,and the terminator represents 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, that is, 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.



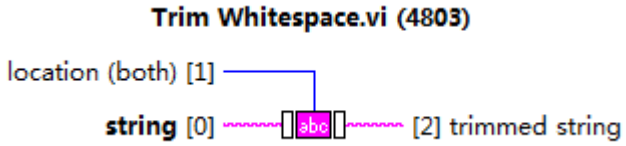
The available functions used in the above block diagram as as follows:



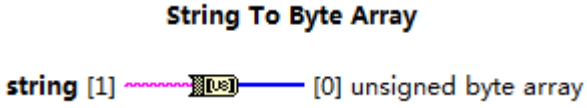
Used to obtain the TMC header"#N". After obtaining the number of bytes that the TMC header occupies, ignore the data header to obtain the waveform data strings



Used to obtain the waveform data length bytes

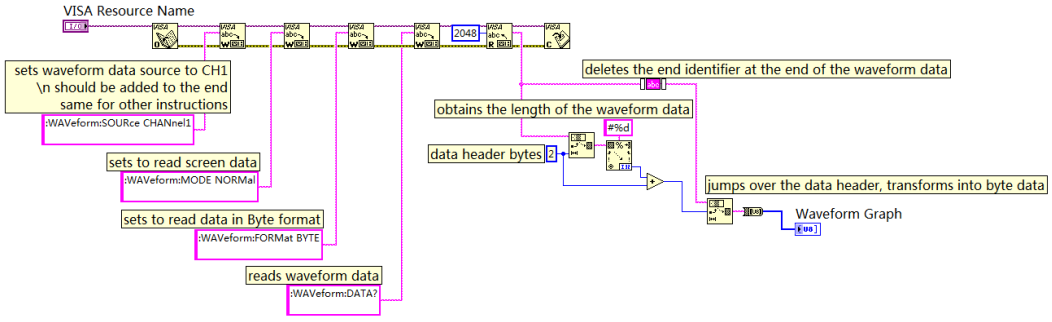


Used to delete the terminator at the end of the waveform data

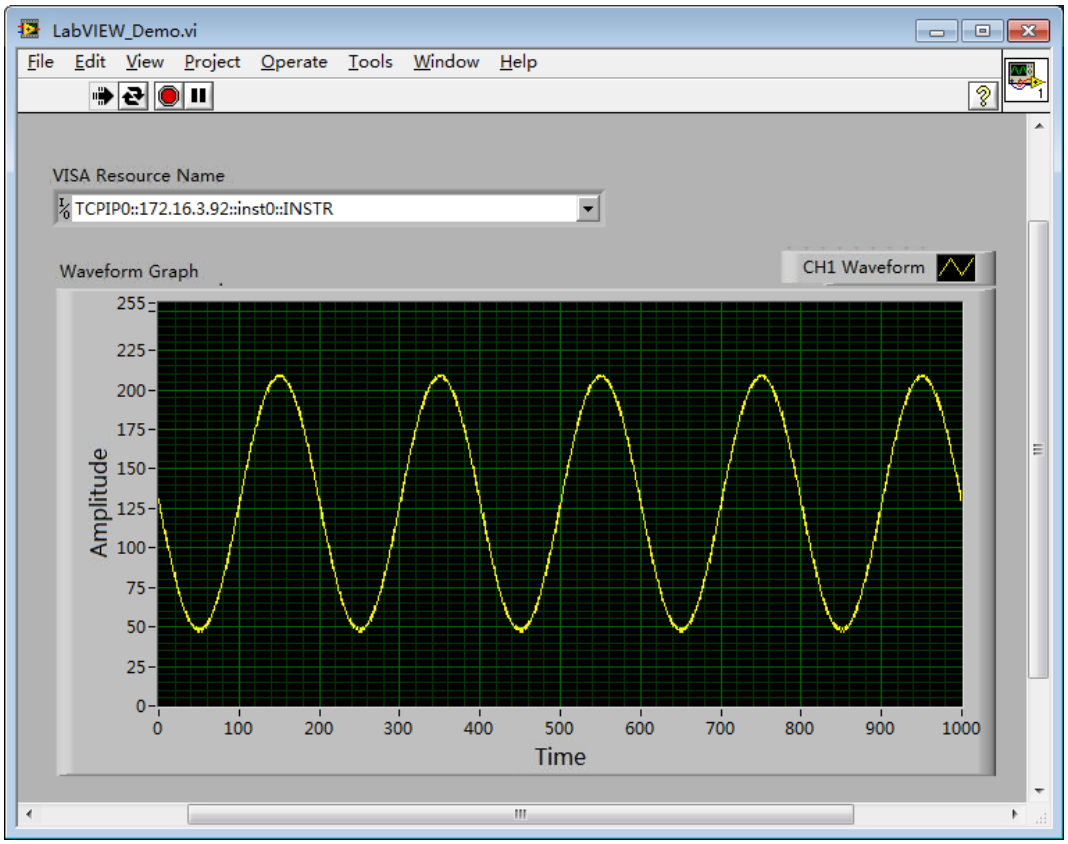


Used to transfer waveform data strings to the byte group

7. The complete program block diagram is as shown in the figure below:



8. Select the device resource from the VISA Resource Name drop-down list and run the program.



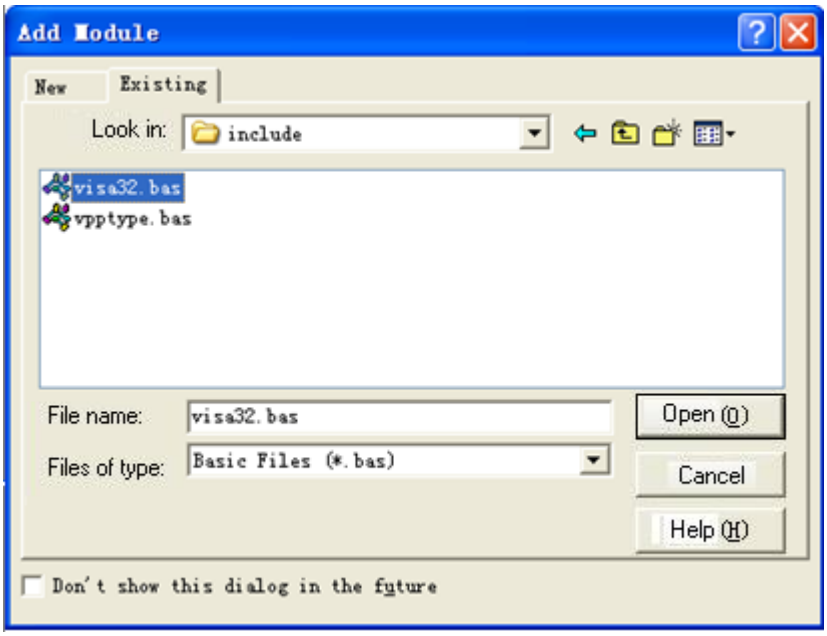
5.3 Visual Basic Programming Example

Program used in this example: Visual Basic 6.0

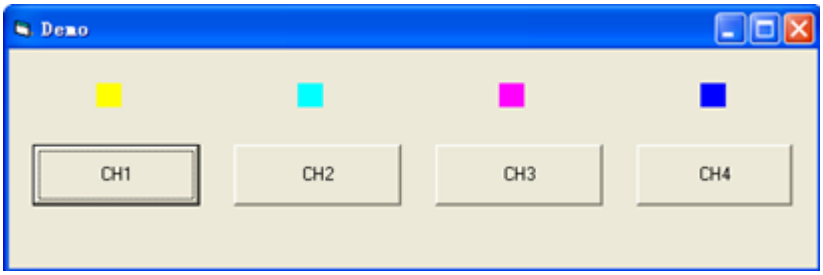
Function realized in this example: control the on/off state of any channel.

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 instrument is correctly connected to the PC, you can control the on/off status of any channel.

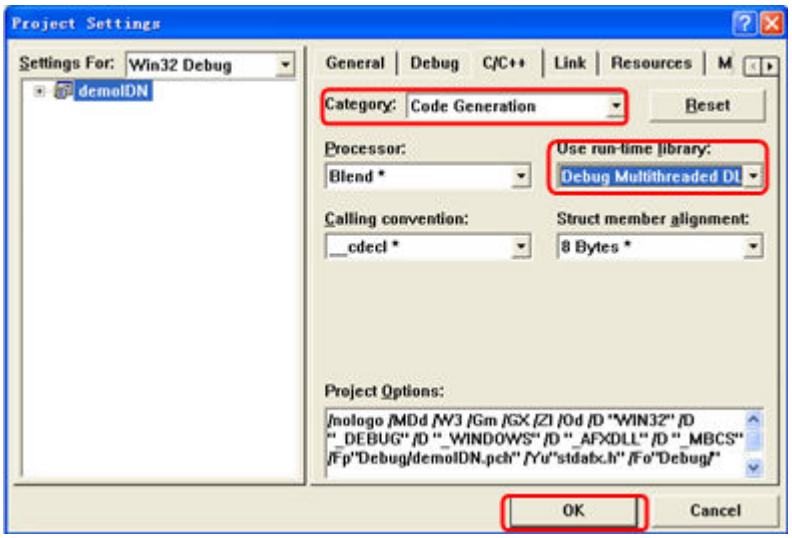
5.4 Visual C++ Programming Example

Program used in this example: Visual C++6.0

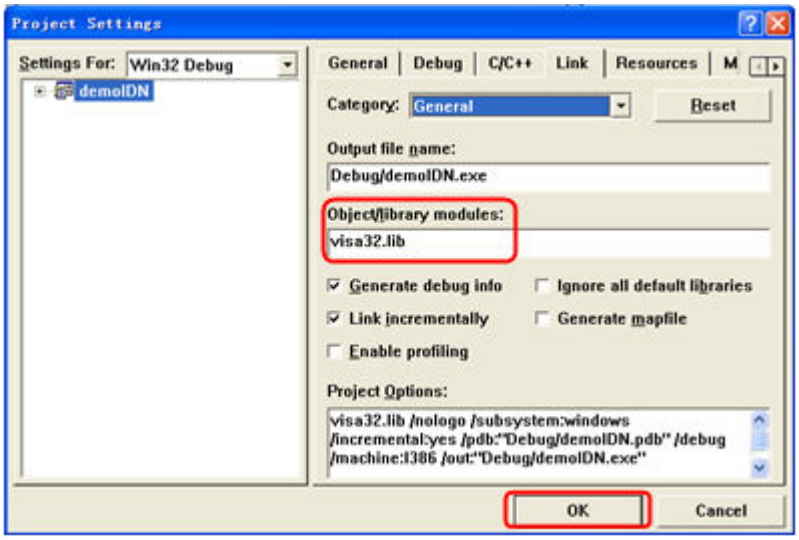
Function realized in this example:search for the instrument address, connect to the instrument, send commands, and read return values.

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.



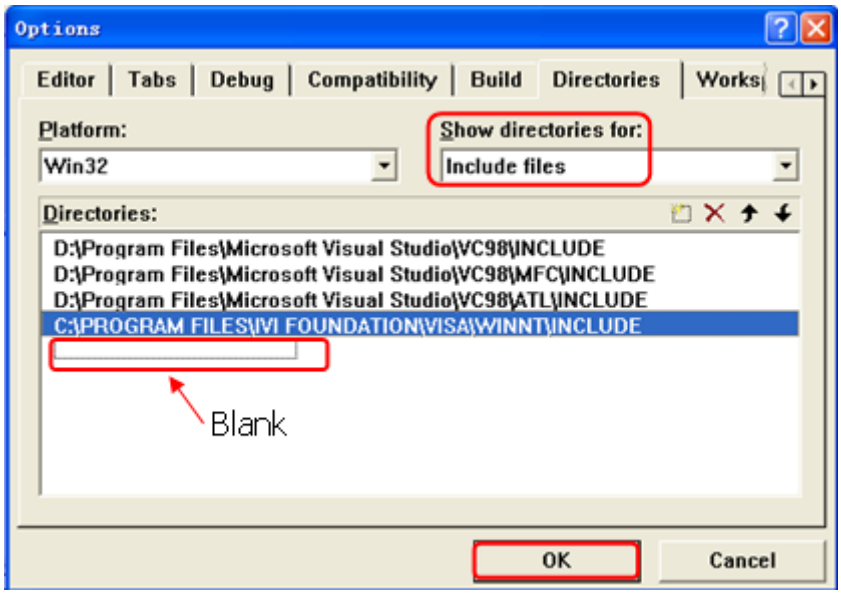
3. Click **Project** > **Settings** to open the **Project Setting** dialog box. In the dialog box, click the **Link** tab, add "visa32.lib" under **Object/library modules**, then click **OK** to close the dialog box.



4. Click **Tools** > **Options** to open the Options dialog box. Then click the **Directories** tab.

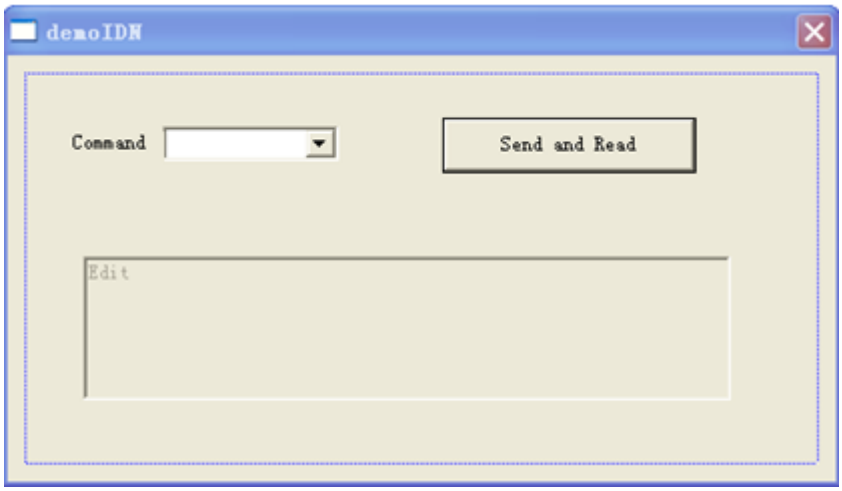
   Select **Include files** from the drop-down list under **Show directories for**. Double click the empty space under **Directories** to enter the specified path of Include files: C:\Program Files\IVI Foundation\VISA\WinNT\include. Click **OK** to close the dialog box.

   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 the **Text**, **Combo Box**, **Button**, and **Edit Box** controls. The layout interface for adding controls is as follows:



6. Modify the control attributes.

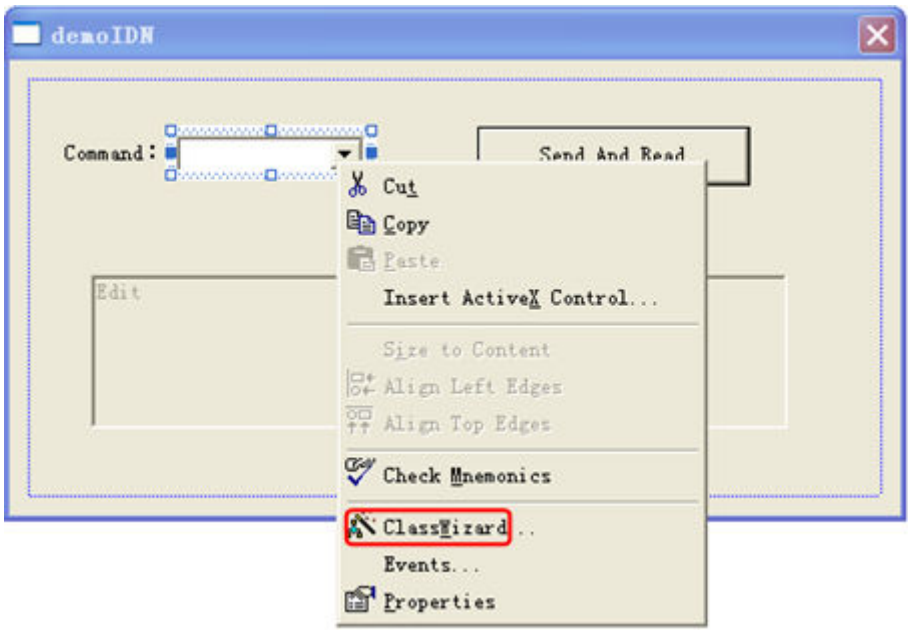
a. Name **Text** as "Command".

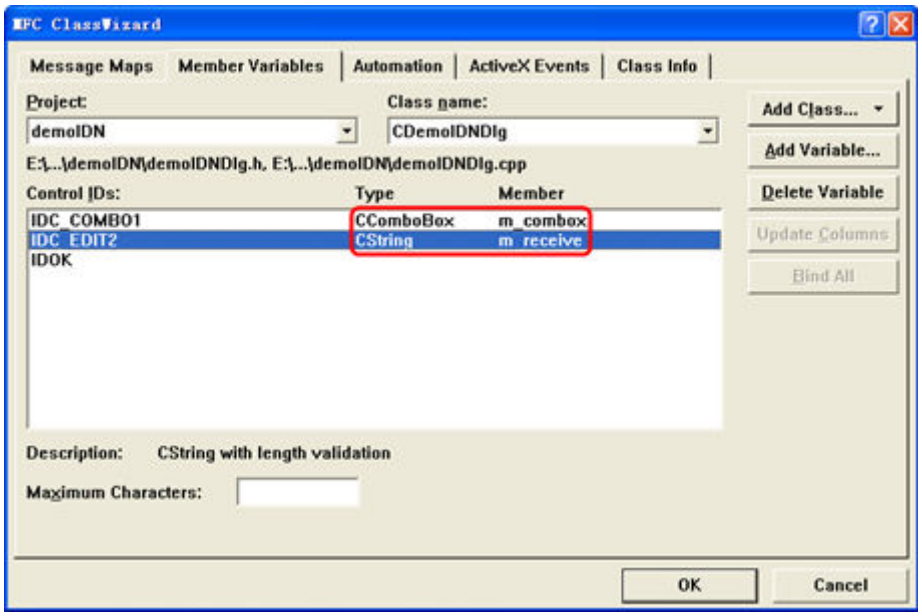
b. Open the **Data** item in the **Combo Box** attribute and input the following command *IDN? manually.

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 **Combo Box** and **Edit Box** controls respectively.





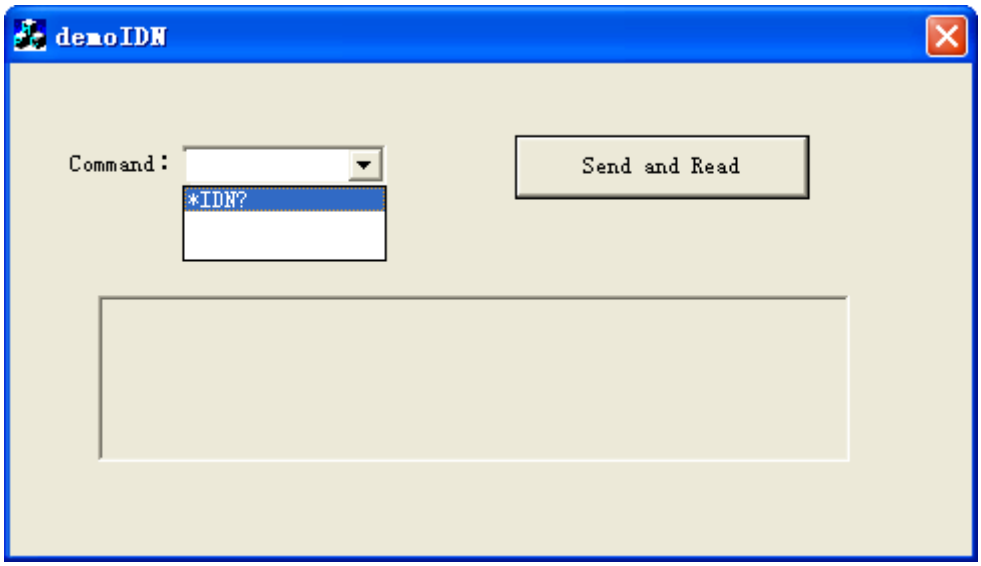
8. Add codes.

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];
```

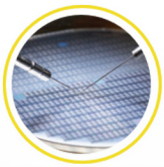
```
ViRsrc matches=buffer;
ViUInt32 nmatches;
ViFindList list;

viOpenDefaultRM (&defaultRM);
//Acquire the USB resource of VISA
viFindRsrc(defaultRM, "USB?*", &list,&nmatches, matches);
viOpen (defaultRM,matches,VI_NULL,VI_NULL,&vi);

//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 instrument 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.



Boost Smart World and Technology Innovation

Industrial Intelligent Manufacturing



Semiconductors

Education& Research

Communication

System Integration

New Energy

- Cellular-5G/WIFI
- UWB/RFID/ ZIGBEE
- Digital Bus/Ethernet
- Optical Communication

- Digital/Analog/RF Chip
- Memory and MCU Chip
- Third-Generation Semiconductor
- Solar Photovoltaic Cells

- New Energy Automobile
- PV/Inverter
- Power Test
- Automotive Electronics

Provide Testing and Measuring Products and Solutions for Industry Customers

Distribution in the UK & Ireland

Lambda

Lambda Photometrics Limited
Lambda House Batford Mill
Harpenden Herts AL5 5BZ
United Kingdom
E: info@lambdaphoto.co.uk
W: www.lambdaphoto.co.uk
T: +44 (0)1582 764334
F: +44 (0)1582 712084