Table of Contents

Pages

Due to continuous improvements in the GDS-2000 series Digital Storage Oscilloscope, information contained in this manual is subject to change without notice. Visit GOOD WILL's website, for revisions and corrections.

GOOD WILL Instrument Co., LTD.

No.7-1, Jhongsing Rd., Tucheng City,

Taipei County 236, Taiwan

Telephone – 886-2-22680389 Fax – 886-2-22680639

E-mail - marketing@goodwill.com.tw

http://www.gwinstek.com.tw

1. Introduction

Users can drive this digital storage oscilloscope by using the GPIB (*General Purpose Interface Bus*) system with a computer, or from a computer across the RS-232 or USB serial connection. Commands sent over either interface can read or set any oscilloscope's instructions. This chapter explains how to carry out the following tasks.

Notes for GPIB installation

If you are setting up the oscilloscope with a GPIB system, please check the following regulations:

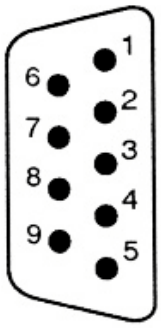
- *Only a maximum of 15 devices can be connected to a single GPIB bus.*
- *Do not use more than 20 m of cable to connect devices to a bus.*
- *Connect one device for every 2 m of cable used.*
- *Each device on the bus needs a unique device address. No two devices can share the same device address.*
- *Turn on at least two-thirds of the devices on the GPIB system while you use the system.*
- *Do not use loop or parallel structure for the topology of GPIB system.*

Notes for RS-232 Configuration

This oscilloscope contains a DB 9-pin, male RS-232 connector for serial communication with a computer or terminal. The RS-232 interface of this oscilloscope is configured as an RS-232 "***Data Terminal Equipmen***t", so that data is sent from pin 3 and received on pin 2. For remote controls, the RS-232 interface has to be connected with a computer or terminal.

Pin Assignments

The pin assignments for RS-232 interface of the oscilloscope are listed below.



1. No connection
2. Receive Data(RxD) (input)
3. Transmit Data(TxD) (output)
4. No connection
5. Signal Ground (GND)
6. No connection
7. No connection
8. No connection
9. No connection

Figure 1-1: Pin assignments for the RS232 connector

DB9 to DB9 Wiring

The wiring configuration is used for computer with DB9 connectors that configured as Data Terminal Equipment.

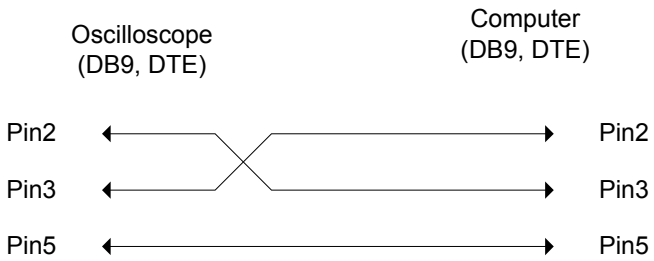


Figure 1-2: DB9 to DB9 wiring

When the oscilloscope is set up with a RS-232 interface, please check the following points:

- ***Do not connect the output line of one DTE device to the output line of the other.***
- ***Many devices require a constant high signal on one or more input pins.***
- ***Ensure that the signal ground of the equipment is connected to the signal ground of the external device.***
- ***Ensure that the chassis ground of the equipment is connected to the chassis ground of the external device.***
- ***Do not use more than 15m of cable to connect devices to a PC.***
- ***Ensure the same configurations are used on the device as the one used on PC terminal.***
- ***Ensure the connector for the both side of cable and the internal connected line are met the demand of the instrument.***

Computer's Connection for RS-232

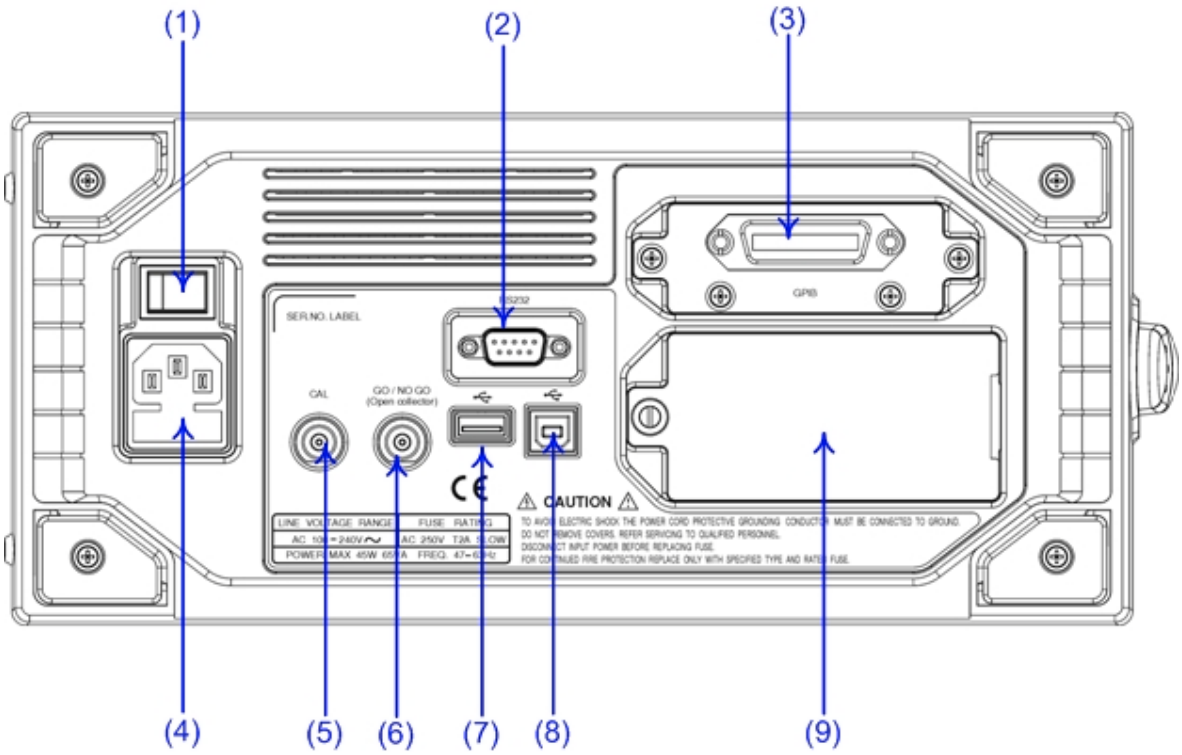
A personal computer with a COM port is the essential facilities in order to operate the digitizing oscilloscope via RS-232 interface.

The connections between oscilloscope and computer are as follows:

I. Connect one end of a RS-232 cable to the computer.
II. Connect the other end of the cable to the RS-232 port on the oscilloscope.
III. Turn on the oscilloscope.
IV. Turn on the computer.

Figure 1-3, shows the GPIB port, RS-232 and USB locations on the rear panel of the oscilloscope.

Figure 1-3. Rear panel of the oscilloscope



(1): Main power switch

(2): RS-232 port

(3): GPIB port (option)

(4): AC socket and fuse drawer

(5): “SELF CAL” BNC output

(6): “GO/NO GO” BNC output. A 10μs pulse will be generated (the minimum time interval between each pulse is around 2ms) 5V Maximum/10mA TTL open collect output.

(7): USB host connector

(8): USB device connector

(9): Battery socket (option)

Computer's Connection for USB

User may connect the USB device connector of oscilloscope with personal computer's USB in order to remote control the oscilloscope via USB interface. This USB device connector is USB 2.0 full speed compatible.

The connections between oscilloscope and computer are as follows:

I. Connect the USB device connector and computer's USB port with a USB cable. The USB cable should be type AB, USB2.0 compliant. User may order a USB cable from GWInstek, the part number for USB cable is 4242-1041820 (GTL-242 USB cable).

II. Connect the USB connector of type B with oscilloscope's USB device connector.

III. Connect the USB connector of type A with computer's USB port.

2. Computer's Connection (GPIB, RS-232, USB)

A personal computer with a GPIB card, RS-232 or USB interface are three of the essential stuffs in order to operate the oscilloscope via GPIB interface, RS-232 or USB interface.

The connections between oscilloscope and computer via GPIB are following:

I. Connect one end of a GPIB cable to the computer.

II. Connect the other end of the GPIB cable to the GPIB port on the Oscilloscope.

III. Turn on the oscilloscope.

IV. Turn on the computer.

The GPIB interface capabilities:

The GPIB interface of the oscilloscope corresponds to the standard of IEEE488.1-1987, IEEE488.2-1992 and SCPI-1994. The GPIB interface functions are listed as follows:

SH1 (Source Handshake):	The oscilloscope can transmit multilane messages across the GPIB.
AH1 (Acceptor Handshake):	The oscilloscope can receive multilane messages across the GPIB.
T6 (Talker):	Talker interface function includes basic talker, serial poll, and un-address if MLA capabilities, without talk only mode function.
L4 (Listener):	The oscilloscope becomes a listener when the controller sends its listen address with the ATN (attention) line asserted. The oscilloscope does not have listen only capability.

SR0 (Service Request):	The oscilloscope has no SRQ (Service request) function.
RL0 (Remote/Local):	The oscilloscope will ignore the LLO (local lockout) command.
PP0 (Parallel Poll):	The oscilloscope has no Parallel Poll interface function.
DC1 (Device Clear):	The oscilloscope has Device clear capability to return the device to power on status.
DT0 (Device Trigger):	The oscilloscope has no Device Trigger interface function.

C0 (Controller) : This oscilloscope can not control other devices.

The GPIB address setting

To change the GPIB address, please use the following steps:

- Press the **UTILITY** button on the front panel. The utility menu provides **Interface Menu** sub menu by pressing **F2** softkey which GPIB sub menu is included. Press **F1** softkey to select GPIB setting menu.

  For GPIB sub menu

  - **Type GPIB**: Select GPIB port.
  - **Addr 1~30**: select the appropriate address for GPIB.
  - **Previous Menu**: back to previous menu.

The GPIB connection testing

If you want to test the GPIB connection is whether working or not, use the National Instrument's "Interactive Control utility" for instance, you communicate with the GPIB devices through calls you interactively type in at the keyboard.

The Interactive Control can help you to learn about the instrument and to troubleshoot problems by displaying the following information on your screen after you enter a command:

- Results of the status word (`ibsta`) in hexadecimal notation
- Mnemonic constant of each bit set in `ibsta`
- Mnemonic value of the error variable (`iberr`) if an error exists (the ERR bit is set in `ibsta`)
- Count value for each read, write, or command function
- Data received from your instrument

You can access online help in Interactive Control by entering help at the prompt, or you can get help on a specific function by entering help <function> at the prompt, where <function> is the name of the function for which you want help.

To start Interactive Control within National Instrument's "Measurement & Automation Explorer", complete the following steps:

1. Select Tools→I-488.2 Utilities→Interactive Control.

2. Open either a board handle or device handle to use for further NI-488.2 calls. Use `ibdev` to open a device handle, `ibfind` to open a board handle, or the set 488.2 command to switch to a 488.2 prompt.

The following example uses `ibdev` to open a device, assigns it to access board gpib0, chooses a primary address of 7 with no secondary address, sets a timeout of 10 seconds, enables the `END` message, and disables the `EOS` mode.

```
:ibdev
    enter board index: 0
    enter primary address: 7
    enter secondary address: 0
    enter timeout: 13
    enter 'EOI on last byte' flag: 1
    enter end-of-string mode/byte: 0
ud0:
```

Note: If you type a command and no parameters, Interactive Control prompts you for the necessary arguments. If you already know the required arguments, you can type them at the command prompt, as follows:

```
:ibdev 0 7 0 13 1 0
ud0:
```

Note: If you do not know the primary and secondary address of your GPIB instrument, right-click on your GPIB interface in Measurement & Automation Explorer and select Scan for Instruments. After Explorer scans your interface, it displays your instrument address in the right window panel.

3. After you successfully complete `ibdev`, you have a `ud` prompt. The new prompt, `ud0`, represents a device-level handle that you can use for further NI-488.2 calls. To clear the device, use `ibclr`, as follows:

```
ud0: ibclr
[0100] (cmpl)
```

4. To write data to the device, use `ibwrt`.

```
ud0: ibwrt "*IDN?"
[0100] (cmpl)
count: 5
```

5. To read data from your device, use `ibrd`. The data that is read from the instrument is displayed. For example, to read 29 bytes, enter the following:

```
ud0: ibrd 29
[0100] (cmpl)
count: 29

47 57 2C 20 47 44 53 2D     GW, GDS-
32 32 30 32 2C 20 50 39     2202, P9
32 30 31 33 30 2C 20 56     20130, V
2E 31 2E 30 30              .1.00
```

6. When you finish communicating with the device, make sure you put it offline using the `ibonl` command, as follows:

```
ud0: ibonl 0
[0100] (cmpl)
:
```

The `ibonl` command properly closes the device handle and the `ud0` prompt is no longer available.

7. To exit Interactive control, type `q`.

For the details, please refer to National Instrument's manual.

If you do not receive a proper response from the oscilloscope, please check the power is on, the GPIB address is correct, and all cable connections are active,

The RS232 connection testing

If you want to test whether the RS-232 connection is working or not, you can send a command from computer. For instance, using a hyper-terminal program send the query command

```
*idn?
```

should return the Manufacturer, model number, serial number and firmware version in the following format:

```
GW,GDS-2202,EG150000,V1.00
```

If you do not receive a proper response from the oscilloscope, please check if the power is on, the RS-232 configurations are the same on both sides, and all cable connections are active.

The USB connection testing

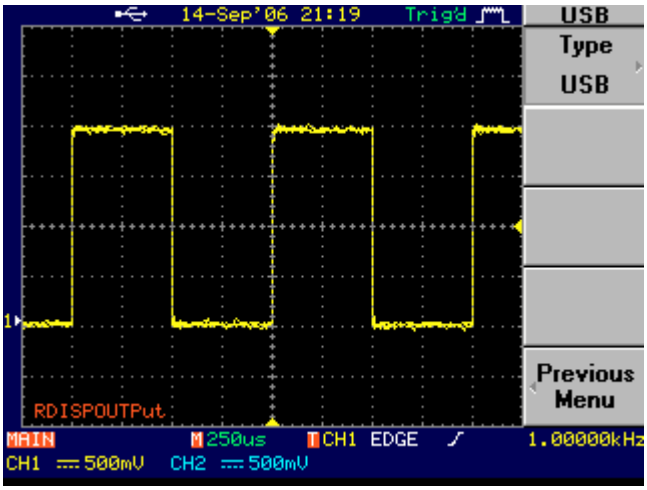
User may test the USB connectivity of oscilloscope and computer by using a hyper-terminal program. For instance, a free PC hyper-terminal program of "MTTTY.exe" was an example here.

This "MTTTY.exe" file may download from following link:
http://www.jps.com/downloads/support/mitty.html

Furthermore, a required driver for Windows2000/XP operation system which named "dso_cdc.inf" should be downloaded for company website.

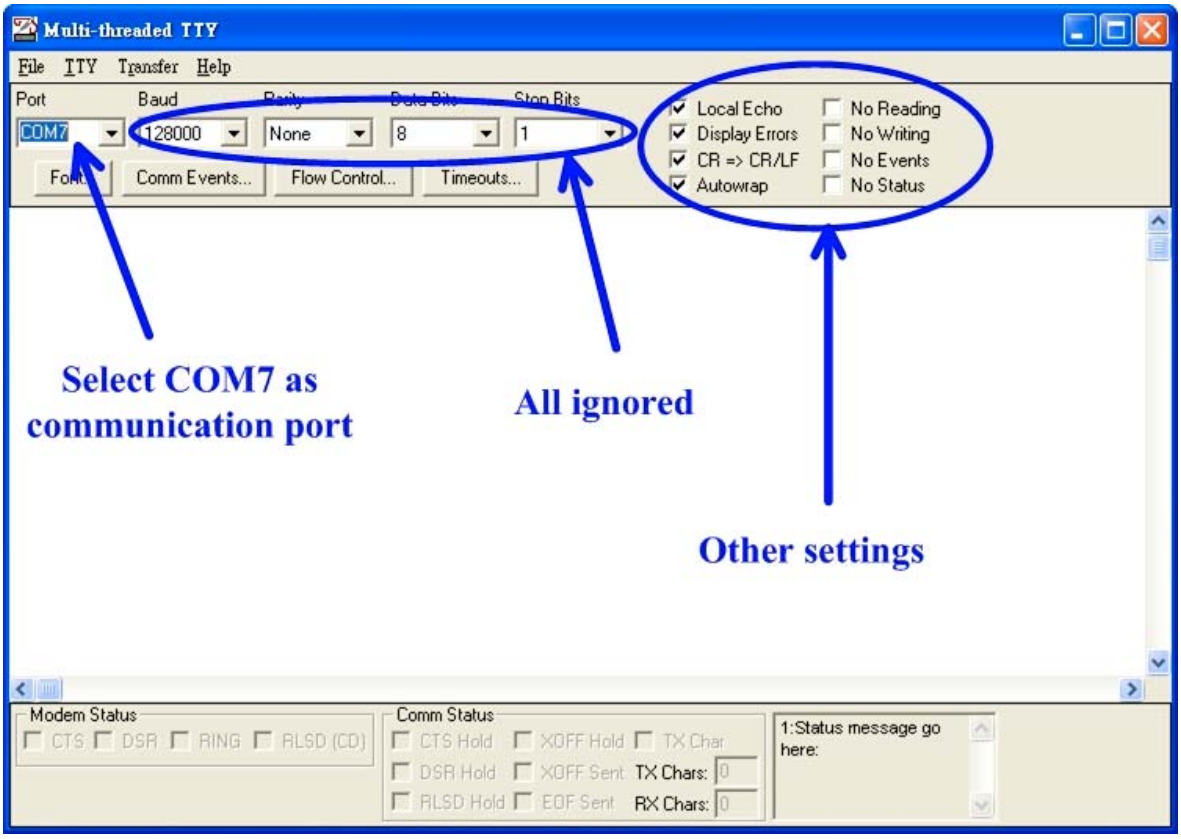
Testing steps:

1. Turn on oscilloscope and press **UTILITY** → **F2** → **F1** to change the interface as USB interface.

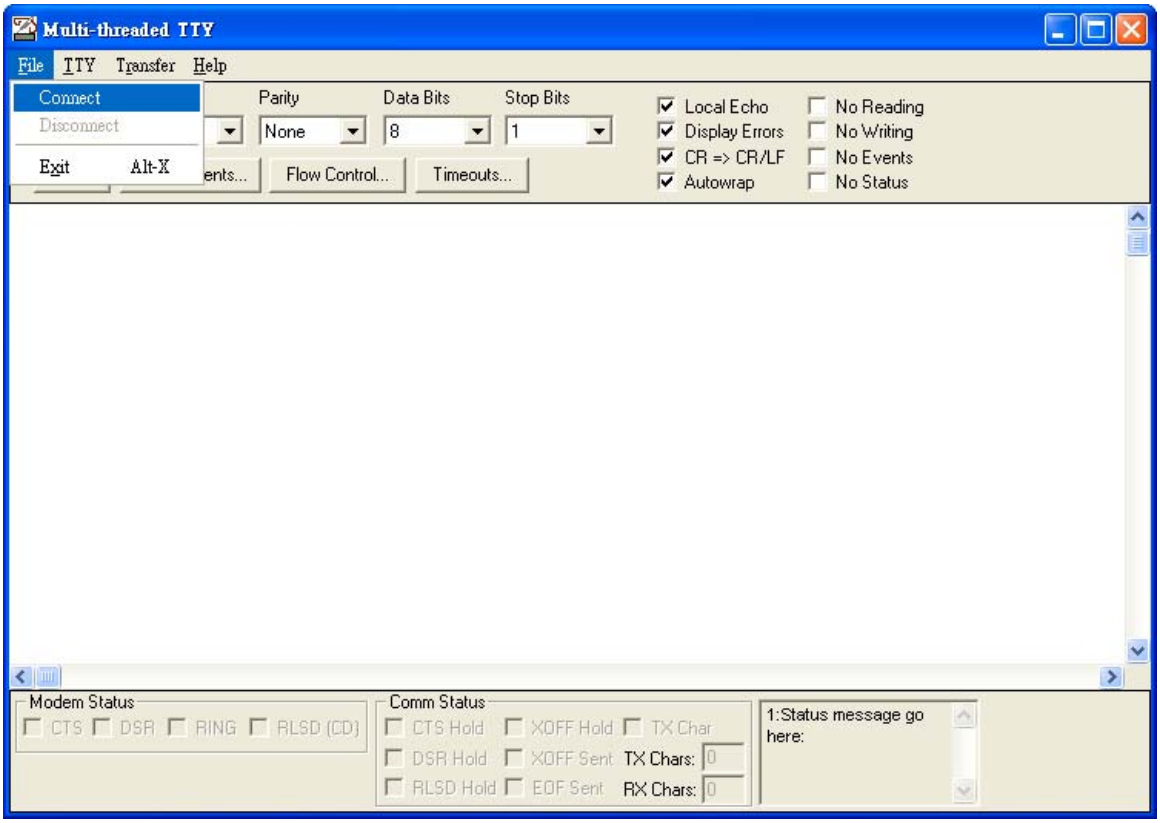


2. Connected both oscilloscope and computer with a USB cable type AB.

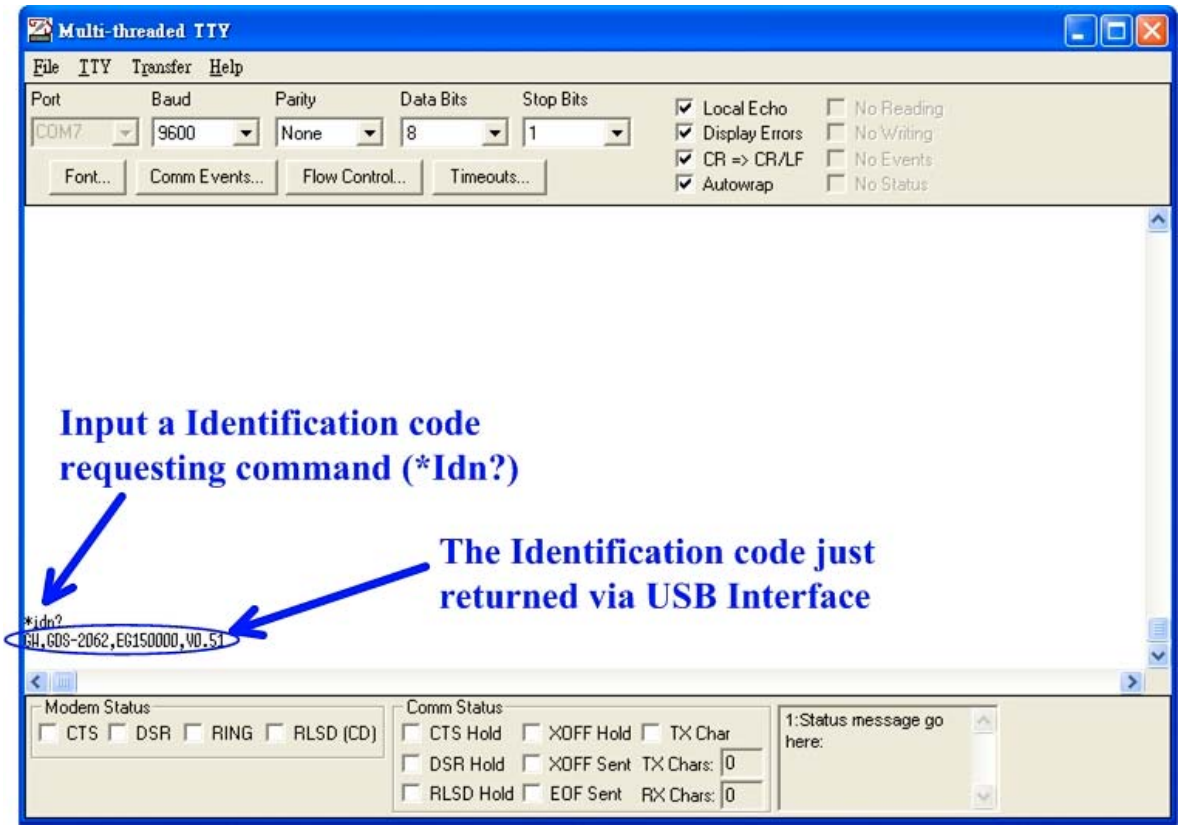
3. Download the “inf” oscilloscope USB driver file from company website. Right click the “inf” oscilloscope USB driver file and install it into the target computer.
4. If driver file is installed successfully and USB interface of oscilloscope working normally, user may check the COM port number by opening the “Device Manager”, one of the specific COM port should be assigned as the communication port for both oscilloscope and computer.
5. Open the “MTTTY.exe” and select the properly COM port number. Only the COM port is necessary for setting, please ignore all other communication setting items.



6. Click the file and select “Connect”.



7. For example, a remote control command of “*idn?” is inputted, the oscilloscope should return the information of manufacturer, model number, serial number and firmware version if all the conditions are working normally. The figure is following.



3. Remote Control's Commands

This oscilloscope can be operated from computer via the GPIB port, RS-232, or USB port. The remote commands of this oscilloscope are compatible with IEEE-488.2 and SCPI standards partially.

SCPI

SCPI (Standard Commands for Programmable Instruments) is a standard that created by an international consortium of the major test and measurement equipment manufacturers. The IEEE-488.2 syntax has been adopted by SCPI to furnish common commands for the identical functions of different programmable instruments.

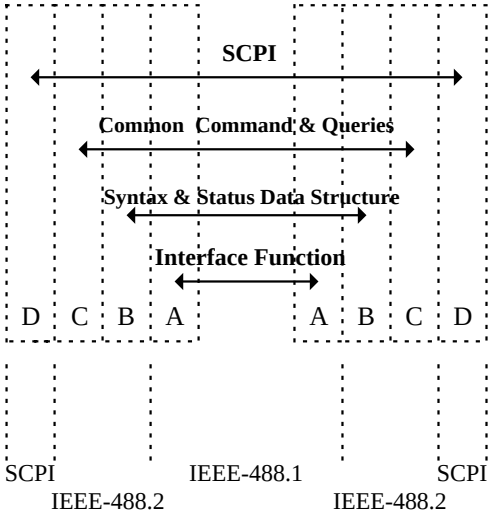


Figure 3-1: the relationship between IEEE-488.1, IEEE-488.2, and SCPI.

As shown in the figure 3-1, the IEEE-488.1 standard locates at layer A, the layer A is belonged to the protocol of interface function on the GPIB bus. The source handshake (SH), acceptor handshake (AH) and talker are included in this layer (10 interface functions totally).

At layer B, the syntax and data structure could be the essence of entire IEEE-488.2 standard. The syntax defines the function of message communication, which contain the <PROGRAM MESSAGE> (or simply “commands”) and <RESPONSE MESSAGE>. The two kinds of messages are represented the syntax formation of device command and return value. The data structure is the constitution of status reporting, which IEEE-488.2 standard have been defined.

The common commands and queries are included with layer C. Commands and queries can be divided into two parts: mandatory and optional. Commands modify control settings or tell the instrument to perform a specific action. Queries cause the instrument to send data or status information back to the computer. A question mark at the end of a command identifies it as a query.

Layer D is interrelated with device information. Different devices have different functions. SCPI command sets are belonged to this layer.

Command Syntax

If you want to transfer any of the instructions to an instrument, and comply with SCPI, there are three basic elements must be included.

- Command header
- Parameter (if required)
- Message terminator or separator

Command Header

The command header has a hierarchical structure that can be represented by a command tree (Figure 3-2).

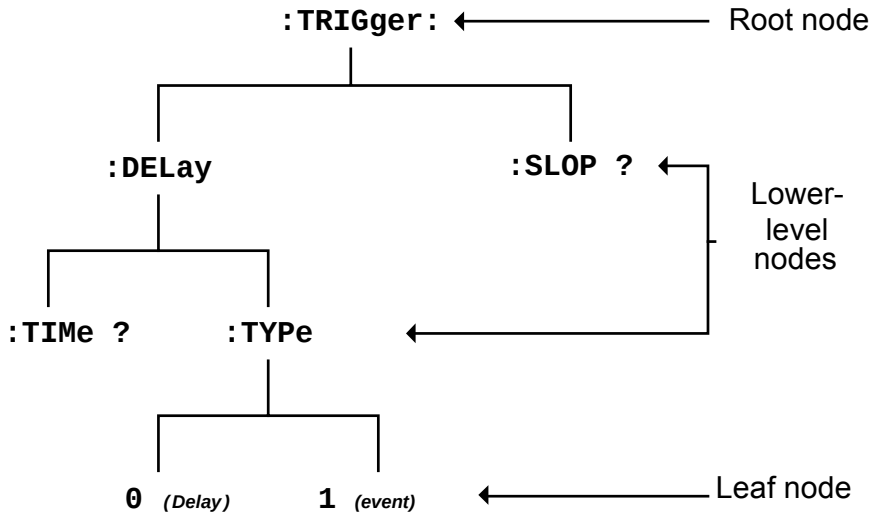


Figure 3-2: Tree hierarchy

The top level of the tree is the root level. A root node is located at the root level. A root node and one or more lower-level nodes form a header path to the last node called the leaf node.

The command header is configured by header path and leaf node. Figure 3-3 shows the command header for the leaf node.

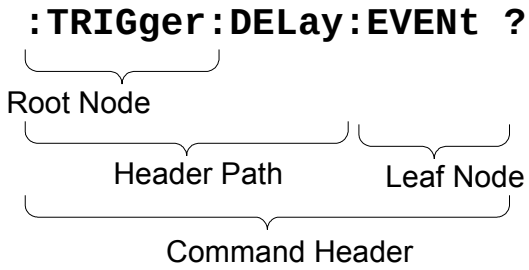


Figure 3-3: Command header

Parameter

If the command has parameters, the values have to be included. In this manual, when we expressed the syntax of the command, the < > symbols are used for enclosing the parameter type. For instance, the syntax of the command in Figure 8-5 includes the Boolean parameter type

NOTE: Do not include the <, >, or | symbols when entering the actual value for a parameter.

Figure 3-4: Command Header with Parameter

Table 3-1 defines the Boolean and other parameter types for the oscilloscope.

Parameter Type	Description	Example
Boolean	Boolean numbers or values	1, 0
NR1	Integers	0, 1, 18
NR2	Decimal numbers	1.5, 3.141, 8.4
NR3	Floating point numbers	4.5E-1, 8.25E+1
NRf	NR1, NR2, or NR3	1, 1.5, 4.5E-1

Table 3-1: Parameter Types for Syntax Descriptions

For the actual value of the parameter type <Boolean>, you have to enter 0 instead of “OFF” or enter 1 instead of “ON”.

The following example includes both the header and a value for the parameter type:

```
:TRIGger:NREJ 0
```

Parameter values that appeared in this manual are often separated by a vertical line. This vertical line means the same thing as the word "or," For example, values for the parameter <Boolean> are

```
0|1
```

This is the same thing as saying "0 (off) or 1 (on)" Any single value is a valid parameter.

Message Terminator and Message Separator

In accordance with IEEE 488.2 standard, this oscilloscope accepts any of the following message terminators:

- LF^END Line feed code (hexadecimal 0A) with END message
- LF Line feed code
- <dab>^END Last data byte with END message

These terminators are compatible with most application programs.

A semicolon separates one command from another when the commands appear on the same line.

Entering Commands

The standards that govern the command set for this oscilloscope allow for a certain amount of flexibility when you enter commands. For instance, you can abbreviate many commands or combine commands into one message that you send to the oscilloscope. This flexibility, called friendly listening, saves programming time and makes the command set easier to remember and use.

Command Characters

This oscilloscope is not sensitive to the case of command characters. You can enter commands in either uppercase or lowercase.

You can precede any command with white space characters. You must, however, use at least one space between the parameter and the command header.

Abbreviating Commands

Most commands have a long form and a short form. The listing for each command in this section shows the abbreviations in upper case. For instance, you can enter the query

```
:TIMebase:SCALe ?
```

simply as

```
:TIM:SCAL ?
```

Combining Commands

You can use a semicolon (;) to combine commands and queries. This oscilloscope executes coherent commands in the order it receives them. When you coherent queries, the oscilloscope will combine the responses into a single response message with LineFeed (0x0A) for each command. For example, if the frequency and amplitude of the signal are equal to 1kHz and 1V, the command

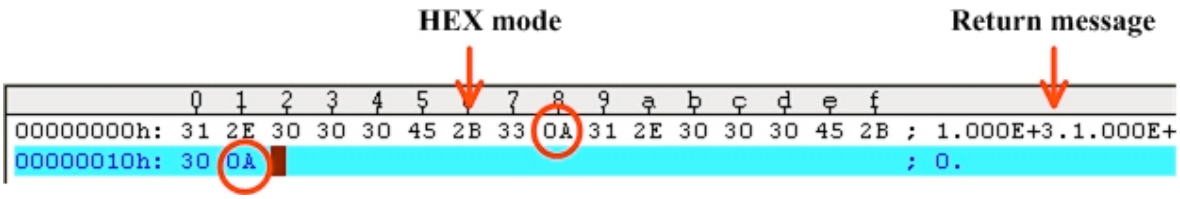
```
:MEASure:FREQuency?;:MEASure:VAMPlitude?
```

returns the message

1.000E+3 1.000E+0

**: Please note that a LineFeed (0x0A) message terminator will be added behind each responded command.*

Therefore the return message of "1.000E+3 1.000E+0" should be exact looked like as following example in HEX mode:



4. Details of Command Reference

Each command in this chapter will give a brief description. The examples of each command will be provided and what query form might return.

*IDN? (query only)

Returns the unique identification code of the oscilloscope.

Syntax

```
*IDN?
```

Examples

```
*IDN?
```

Returns `Company Name,Model Name,Series number,Firmware version`

*LRN? (query only)

Returns the string that the oscilloscope settings will be listed.

Syntax

```
*LRN?
```

Returns

```
                                    :SYSTem:TIMe 17 59 57;DATe 18 10 2006;:TRIG
ger:TYPe 0;COUPle 1;LEVel 940mV ;MODe 1;NREJ 0;REJect 0;SLOP 0;SOURce 0;ADVance:
DELay 20.0ns;EVENt 3;LEVel 1.48V ;MODe 0;TYPe 0;PULSe:TIMe 0.000E-9 ;MODe: 0;TV:
FIELd 1;LINe 1;POLarity 1;TYPe 1;:ACQuire:AVERage 1;LENGth 500;MODe 0;POINt;:DIS
Play:WAVeform 0;DISPCONTrast -10;GRATicule 0;:CURSor:SOURce 1;X1Position
  ;X2Position          ;Y1Position          ;Y2Position          ;XDELTa
  ;YDELTa          ;XDISPlay 2;YDISPlay 2;:CHANnel1:BWLimit 0;COUPling 0;DISPlay
 1;INVert 0;MATH 0;OFFSet 4.700e-01;PROBe 0;SCALe 5.000e-01;:CHANnel2:BWLimit 0;
COUPling 0;DISPlay 0;INVert 0;MATH 0;OFFSet 0.000e+00;PROBe 0;SCALe 5.000e-01;:M
EASure:SOURce 1;FALL ?;FREQuency ?;NWIDth ?;PDUTy ?;PERiod ?;PWIDth ?;RISe 7.918
us;VAMPlitude ?;VAVerage 941mV  ;VHI ?;VLO ?;VMAX 1.94V  ;VMIN -60.0mV  ;VPP ?;V
RMS 1.35V  ;ROVShoot ?;FOVShoot ?;RPReshoot ?;FPReshoot ?;:MEASure:SOURce 2;FALL
 ?;FREQuency ?;NWIDth ?;PDUTy ?;PERiod ?;PWIDth ?;RISe chan off;VAMPlitude ?;VAV
erage chan off;VHI ?;VLO ?;VMAX chan off;VMIN chan off;VPP ?;VRMS chan off;ROVSh
oot ?;FOVShoot ?;RPReshoot ?;FPReshoot ?;DELAY1 1;DELAY2 2;FRRDelay ?;FRFDelay ?
;FFRDelay ?;FFFDelay ?;LRRDelay ?;LRFDelay ?;LFRDelay ?;LFFDelay ?;:TIMebase:DEL
ay 0.000e+00;SCALe 2.500e-04;SWEep 0;:AUToset;:PRINt;:REFResh;:RUN;:STOP
```

*RCL

Recall the setting data from memory which previous saved. The settings of oscilloscope can be stored in memory of S1~S20. However, if users recall the stored memory which the settings of RS-232 or GPIB are different with present settings, the RS-232 (or GPIB) settings will keep with the present situation. The RS-232 (or GPIB) settings will not to be influenced by the recall setting of RS-232 (or GPIB) for this moment.

Syntax

```
*RCL <NR1>
```

Arguments: 1~20

Examples

```
*RCL 1
```

Recalls the setting data which located at first position of memory address.

*RST (no query form)

Sets all control settings of oscilloscope to their default values but does not purge stored setting.

Syntax

```
*RST
```

*SAV

Saves the setting data to memory.

Syntax

```
*SAV <NR1>
```

Arguments

1~20

Examples

```
*SAV 2
```

Saves the setting data to the second position of memory queue.

:ACQuire:AVERage

Select the average number of waveform acquisition. The range for averaging is from 2 to 256 in powers of 2.

Note: Before implement this instrument, please apply ":ACQuire:MODe 2" in advance!

Syntax

```
:ACQuire:AVERage {1|2|3|4|5|6|7|8}

:ACQuire:AVERage?
```

Arguments

1→Average number is 2

2→Average number is 4

3→Average number is 8

4→Average number is 16

5→Average number is 32

6→Average number is 64

7→Average number is 128

8→Average number is 256

Returns

<NR1>

:ACQuire:LENGth

Select the number of record length. This oscilloscope provides record length of 500, or long record length (5000/12500/25000). Depend on the input channels On/Off; the long record length is variety.

Record length / Channels	500 pts	5000 pts	12500 pts	25000 pts
Any one channel is turned **ON**	Available	NA	NA	Available
Any two channels are turned **ON**	Available	NA	Available	NA
Any three of channels are turned **ON** (for four channels models).	Available	Available	NA	NA
All four channels are turned **ON** (for four channels models).	Available	Available	NA	NA

Syntax

```
:ACQuire:LENGth {0|1}

:ACQuire:LENGth?
```

Arguments

0→Record length is 500

1→Record length is the record length at present (depend on the channels ON/Off)

Returns

<NR1>

:ACQuire:MODe

Select the waveform acquisition mode. There are four different acquisition mode: sample, peak detection, average and accumulate.

Syntax

```
:ACQuire:MODe {0|1|2}
:ACQuire:MODe?
```

Arguments

0→Select the normal mode　　1→Select the peak detection mode

2→Select the average mode

Returns

<NR1>

Note: Please select the specific acquire mode before implement any acquisition.

:ACQuire<X>:MEMory? (query only)

Transfer the total waveform data from the acquisition memory. (The memory capacity can be selected as 500, 5000, 12500, or 25000 points. See Page 28 for details.)

Syntax

:ACQuire<X>:MEMory?

Arguments

<X>→Specify the channel number (1|2|3|4)

Returns

The string of data is following.

#	Data size digit	Data size	The time duration between two adjacent sampling points	Channel indicator	Reserved data	Waveform data

#: Begin a transmission of data string.

Data size digit: Indicate the digits of following data string amount (1 digit).

Data size: the amount of current data string (4, or 5 digits). Next table lists the relations between data size digit, data size and record length.

Data String / Record length	Begin a transmission of data string	Data size digit	Data size	
Record length=500	#	4	1008	The rest of data
Record length=5000	#	5	10008	The rest of data
Record length=12500	#	5	25008	The rest of data
Record length=25000	#	5	50008	The rest of data

The time duration between two adjacent sampling points: Indicates the corresponding time interval between two adjacent sampling points of received waveform data (4 bytes). The time interval is indicated by floating point format which compatible with IEEE 754 standards.

Note: This block uses little-endian byte ordering. See the demo code of Appendix A for more conversion information

Channel indicator: Show the channel which sent the waveform data (1 byte).

1→Channel one　　2→Channel two

3→Channel three　　4→Channel four

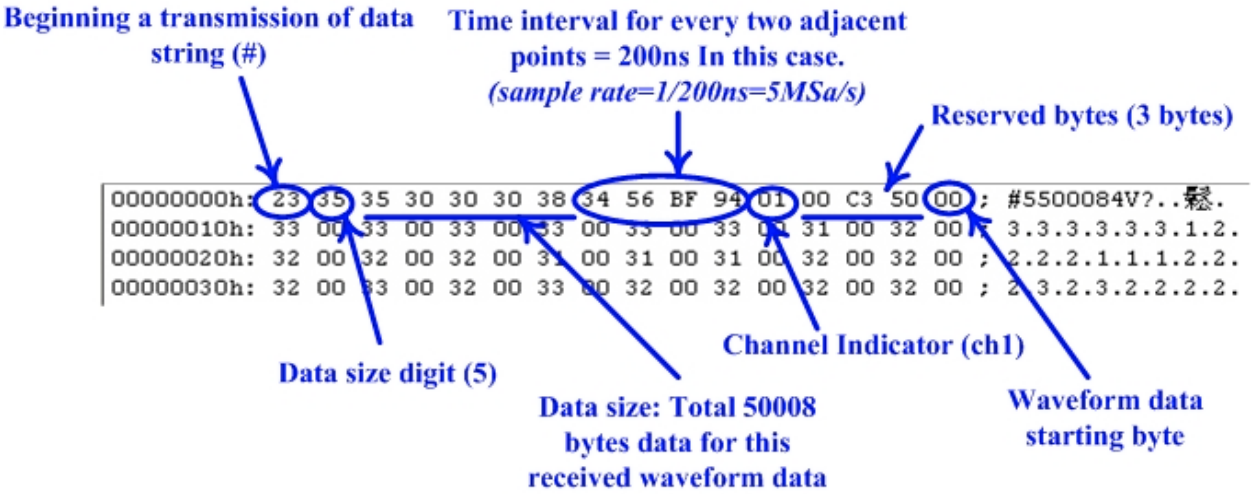
Reserved Data: Not used (3 bytes).

Waveform data: Depends on the specify record length, the effective waveform data which covers 500 points (1000 bytes), 5000 points (10000 bytes), 12500 points (25000 bytes), or 25000 points (50000 bytes).
Each point is composed by two bytes (the integer value of 16 bits). The high byte (MSD) will be prior transferred.

Example 1

Transfer the waveform data (sample rate = 5MSa/s) of channel 1 at 25000 points record length:

`:ACQuire1:MEMory?` The oscilloscope will return the following data string:



Note: Please select the specific memory length before implement any acquisition. See explanation for "`:ACQuire:LENGth`".

:AUToset

Perform an automatic setup in order to optimize the acquisition parameters.

Syntax

```
:AUToset
```

:CHANnel<X>:BWLimit

Enable or disable the bandwidth limit function.

Syntax

```
:CHANnel<X>:BWLimit {0|1}
```

```
:CHANnel<X>:BWLimit?
```

Arguments

<X>→Specify the channel number (1|2|3|4)

0→Disable bandwidth limit 1→Enable bandwidth limit

Returns <NR1>

:CHANnel<X>:COUPling

Select the different coupling states for the oscilloscope.

Syntax

```
:CHANnel<X>:COUPling {0|1|2}
```

```
:CHANnel<X>:COUPling?
```

Arguments

<X>→Specify the channel number (1|2|3|4)

0→Place scope in AC coupling state 1→Place scope in DC coupling state

2→Place scope in grounding state

Returns

<NR1>

:CHANnel<X>:DISPlay

Enable or disable the channel's display.

Syntax

```
:CHANnel<X>:DISPlay {0|1}
```

```
:CHANnel<X>:DISPlay?
```

Arguments

<X>→Specify the channel number (1|2|3|4)

0→Disable channel <X> display 1→Enable channel <X> display

Returns

<NR1>

:CHANnel<X>:INVert

Enable or disable the waveform invert function.

Syntax

```
:CHANnel<X>:INVert {0|1}
```

```
:CHANnel<X>:INVert?
```

Arguments

<X>→Specify the channel number (1|2|3|4)

0→Disable invert function 1→Enable invert function

Returns

<NR1>

:CHANnel<X>:MATH

Set the math expression.

Syntax

```
:CHANnel<X>:MATH {0|1|2|3}
```

Arguments

<X>→Specify the channel number (1|2|3|4)

The mathematical operations are specified as following:

For two channels modes: CH1+CH2, CH1-CH2, FFT.

For four channels modes: CH1+CH2, CH1-CH2, CH3+CH4, CH3-CH4, FFT

0→Select the add operator

1→Select the subtract operator

2→Select the FFT operation

3→Turn off math function

Returns

<NR1>

Examples

```
:CHANnel2:MATH 1
```

Display the remainder which the channel 2 waveform is subtracted from the channel 1 waveform.

```
:CHANnel3:MATH 1
```

Display the remainder which the channel 4 waveform is subtracted from the channel 3 waveform.

```
:CHANnel4:MATH 1
```

Display the remainder which the channel 4 waveform is subtracted from the channel 3 waveform.

:CHANnel<X>:OFFSet

Sets or query the offset voltage.

Syntax

:CHANnel<X>:OFFSet <NR3>

:CHANnel<X>:OFFSet?

Arguments

<X>→Specify the channel number (1|2|3|4)

<NR3> is the desired offset value in volts. The range is dependent on the scale and the probe attenuation factor. The offset ranges are following:

Offset Range:	
2mV/div ~ 50mV/div	±0.5V
100mV/div ~ 500mV/div	±5V
1V/div ~ 5V/div	±50V

Next table shows the relationship between the <NR3> value and matching offset voltage.

2.00E-3→2mV	1.00E-2→10mV	1.00E-1→100mV	1.00E+0→1V
5.00E-3→5mV	2.00E-2→20mV	2.00E-1→200mV	2.00E+0→2V
	5.00E-2→50mV	5.00E-1→500mV	5.00E+0→5V

Returns

<NR3>

:CHANnel<X>:PROBe

Select the different probe attenuation factor.

Syntax

```
:CHANnel<X>:PROBe {0|1|2}

:CHANnel<X>:PROBe?
```

Arguments

<X>→Specify the channel number (1|2|3|4)

0→1X　　1→10X　　2→100X

Returns

<NR1>

:CHANnel<X>:SCALe

Sets or query the vertical scale of the specified channel.

Syntax

:CHANnel<X>:SCALe <NR3>

:CHANnel<X>:SCALe?

Arguments

<X>→Specify the channel number (1|2|3|4)

<NR3> is the desired gain value in volts per division. The range is 2mV/div to 5V/div (with 1X probe).

Next table shows the relationship between the <NR3> value and matching scale.

2.00E-3→2mV	1.00E-2→10mV	1.00E-1→100mV	1.00E+0→1V
5.00E-3→5mV	2.00E-2→20mV	2.00E-1→200mV	2.00E+0→2V
	5.00E-2→50mV	5.00E-1→500mV	5.00E+0→5V

Returns

<NR3>

Examples

:CHANnel4:SCALe 1.00E+0, setup the channel four at 1V per division.

:COPY

This command is equivalent to pressing the **HARDCOPY** button which located on the front panel of the oscilloscope. This command sends a copy of the screen display, oscilloscope settings, or raw data to the flash disk or printer which specified by `:HARDcopy:MODe`.

Both format and layout of the output are specified with `:HARDcopy:INKSaver`, `:HARDcopy:LAYout`, and `:HARDcopy:RATio` commands.

Syntax

```
:COPY
```

:CURSor:X<X>Position

Adjust the cursors position of X axis.

Syntax

```
:CURSor:X<X>Position <NR1>
:CURSor:X<X>Position?
```

Arguments

`<X>`→Specify the cursor (1|2)

`<NR1>` is the desired position. For x-axis operation, the range is 1 to 249 when MENU ON/OFF button is turned on; the range is 1 to 299 when MENU ON/OFF button is turned off.

Returns

<NR3>→Returns position value of T1 or T2.

CURSor:Y<X>Position

Adjust the cursors position of Y axis.

Syntax

```
:CURSor:Y<X>Position <NR1>
```

```
:CURSor:Y<X>Position?
```

Arguments

<X>→Specify the cursor (1|2)

<NR1> is the desired position. The range is 1 to 199.

Returns

<NR3>→Returns position value of V1 or V2.

:CURSor:<X>DELta? (query only)

Return the time or voltage diversity between the two vertical or horizontal cursors.

Syntax

```
:CURSor:XDELta?
```

```
:CURSor:YDELta?
```

Arguments

<X>→Specify the time or voltage diversity (X|Y)

Returns

<NR3>

:CURSor:XDISplay

Enable or disable the cursors display for X axis.

Syntax

```
:CURSor:XDISplay {0|1}
```

Arguments

0→Disable cursors display　　1→Enable cursors display

Returns

<NR1>

:CURSor:YDISplay

Enable or disable the cursors display for Y axis.

Syntax

```
:CURSor:YDISplay {0|1}
```

Arguments

0→Disable cursors display　　1→Enable cursors display

Returns

<NR1>

:CURSor:SOURce

Select which channel cursors is active for front panel control.

Syntax

```
:CURSor:SOURce {1|2|3|4|5}
```

```
:CURSor:SOURce?
```

Arguments

1→Select channel 1 for cursors measurement

2→Select channel 2 for cursors measurement

3→Select channel 3 for cursors measurement

4→Select channel 4 for cursors measurement

5→Select math function for cursors measurement

Returns

<NR1>

:DISPlay:ACCumulate

Select the accumulate display mode.

Syntax

```
:DISPlay:ACCumulate {0|1}
```

```
:DISPlay:ACCumulate?
```

Arguments

0→Disable accumulate display mode 1→Enable accumulate display mode

Returns

<NR1>

:DISPlay:CONTrast

Select contrast level of LCD screen. This command is equivalent to setting the Contrast option in the **DISPLAY** menu.

Syntax

```
:DISPlay:CONTrast {-1|1}
:DISPlay:CONTrast?
```

Arguments

-1→ lower the screen contrast for one scale.

1→ greater the screen contrast for one scale.

Returns

<NR1> The range is from -10~10.

:DISPlay:GRATicule

Select graticule display type for LCD screen.

Syntax

```
:DISPlay:GRATicule {0|1|2}
:DISPlay:GRATicule?
```

Arguments

0→Select full grids

1→Select cross type

2→Only the outer frame is displayed

Returns

<NR1>

:DISPlay:WAVeform

Select the dots (or vectors) display for data.points.

Syntax

```
:DISPlay:WAVeform <0|1>
```

```
:DISPlay:WAVeform?
```

Arguments

0→Enable vectors display 1→Enable dots display

Returns

<NR1>

:GONogo:CLEar

Clear the total and failure counter value on the screen. This command is equal to the function key "**F5**" of GO/NO-GO template edit menu.

Syntax

```
:GONogo:CLEar
```

Examples

```
:GON:CLE clear the GO/NO-GO counter displayed.
```

Note: Before running any Go/NoGo commands, the oscilloscope has to be initialized, please add the command of "`:GONogo:FUNCtion 1`" in order to initial the oscilloscope.

:GONogo:EXECute

Start or stop the execution of GO/NO-GO comparison function. This command is equal to the function key "**F4**" of GO/NO-GO menu.

Syntax

```
:GONogo:EXECute {0|1}
:GONogo:EXECute?
```

Arguments

0→Stop GO/NO-GO comparison 1→Start GO/NO-GO comparison

Returns

<NR1>

Note: Before running any Go/NoGo commands, the oscilloscope has to be initialized, please add the command of ":GONogo:FUNCtion 1" in order to initial the oscilloscope.

:GONogo:FUNCtion

This command is used to initialize or relieve the GO/NO-GO function. Before the GO/NO-GO comparison is started. This command must be executed to enable the comparison unit.

Syntax

```
:GONogo:FUNCtion {0|1}
:GONogo:FUNCtion?
```

Arguments

0→Relieve GO/NO-GO function

1→Initialize GO/NO-GO function

Returns

<NR1>

:GONogo:NGCount? (query only)

Return the value of fail count and total count.

Syntax

```
:GONogo:NGCount?
```

Returns

<NR1>

:GONogo:NGDefine

Sets and query the user-defined NO-GO conditions. This command is equal to the function key "**F4**" pressing for UTILITY menu of second page.

Syntax

```
:GONogo:NGDefine {0|1}
```

```
:GONogo:NGDefine?
```

Arguments

0→Define waveform NOT exceed the template means NO-GO

1→Define waveform exceed the template means NO-GO

Returns

<NR1>

Note: Before running any Go/NoGo commands, the oscilloscope has to be initialized, please add the command of "`:GONogo:FUNCtion 1`" in order to initial the oscilloscope.

:GONogo:SOURce

Sets and query the user-defined GO/NO-GO comparison source channel.

This command is equal to the function key "**F2**" pressing of GO/NO-GO menu.

Syntax

```
:GONogo:SOURce {1|2|3|4}
```

```
:GONogo:SOURce?
```

Arguments

1→Select CH1 to be the comparison source.

2→Select CH2 to be the comparison source.

3→Select CH3 to be the comparison source.

4→Select CH4 to be the comparison source.

Returns

<NR1>

Note: Before running any Go/NoGo commands, the oscilloscope has to be initialized, please add the command of ":GONogo:FUNCtion 1" in order to initial the oscilloscope.

:GONogo:VIOLation

Sets and query the user-defined GO/NO-GO violation conditions. This command is equal to the function key "**F3**" pressing of GO/NO-GO menu.

Syntax

```
:GONogo:VIOLation {0|1|2|3}
:GONogo:VIOLation?
```

Arguments

0→Stop　　1→Stop + beep

2→Continue　　3→Continue + beep

Returns

<NR1>

Note: Before running any Go/NoGo commands, the oscilloscope has to be initialized, please add the command of "`:GONogo:FUNCtion 1`" in order to initial the oscilloscope.

:HARDcopy:INKSaver

Sets the output format of hardcopy to “ink-save” mode.

Syntax

```
:HARDcopy:INKSaver {0|1}
```

```
:HARDcopy:INKSaver?
```

Arguments

0→ OFF　　1→ ON

Returns

<NR1>

:HARDcopy:LAYout

Sets the print-out of hardcopy in black & white or in color.

Syntax

```
:HARDcopy:LAYout {1|2}
```

```
:HARDcopy:LAYout?
```

Arguments

1→ Color Portrait　　2→ Gray Portrait

Returns

<NR1>

:HARDcopy:MODe

Select the hardcopy output style. The entire LCD screen, waveform raw data, oscilloscope settings can be saved on flash disk or print out the entire LCD screen to mostly USB printers.

Syntax

`:HARDcopy:MODe {0|1|2}`

`:HARDcopy:MODe?`

Arguments

`0`→ Save image `1`→ Save all

`2`→ USB Printer

Returns

<NR1>

:HARDcopy:RATio

Select the print-out image size for hardcopy function. The adjustable range is from 10% to 100%. The one hundred percent will fill the entire image with selected printer's paper size; ten percent will reduce 90% of image size with original selected printer's paper size. The default setting is 50%.

Syntax

`:HARDcopy:MODe <NR1>`

`:HARDcopy:MODe?`

Arguments

<NR1> between 10~100

Returns

<NR1>%

:MEASure:DELAY1

Select the first source for eight different delay time measurements (FFFDelay, FRFDelay, FFRDelay, FRRDelay, LRRDelay, LRFDelay, LFRDelay, LFFDelay).

Syntax

```
:MEASure:DELAY1 {1|2|3|4}
```

```
:MEASure:DELAY1?
```

Arguments

1→Channel 1　　2→Channel 2

3→Channel 3　　4→Channel 4

Returns

<NR1>

:MEASure:DELAY2

Select the second source for eight different delay time measurements (FFFDelay, FRFDelay, FFRDelay, FRRDelay, LRRDelay, LRFDelay, LFRDelay, LFFDelay).

Syntax

```
:MEASure:DELAY2 {1|2|3|4}
```

```
:MEASure:DELAY2?
```

Arguments

1→Channel 1　　2→Channel 2

3→Channel 3　　4→Channel 4

Returns

<NR1>

:MEASure:FALL? (query only)

Return the value of timing measurement that taken for falling edge of the first pulse in the waveform.

Syntax

```
:MEASure:FALL?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:FFFDelay? (query only)

Return the time value that difference time between the first falling edge of source1 signal and the first falling edge of source2 signal.

Syntax

```
:MEASure:FFFDelay?
```

Returns

<NR3>.

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for ":MEASure:DELAY1 and :MEASure:DELAY2"

:MEASure:FFRDelay? (query only)

Return the time value that difference time between the first falling edge of source1 signal and the first rising edge of source2 signal.

Syntax

```
:MEASure:FFRDelay?
```

Returns

<NR3>.

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for "`:MEASure:DELAY1 and :MEASure:DELAY2`"

:MEASure:FOVShoot? (query only)

Return the ratio of waveform amplitude that expressed the difference between the low level and the negative peak level of the measured signal as it transitions from its high state to its low state.

Syntax

```
:MEASure:FOVShoot?
```

Returns

<NR2> with a symbol of percentage "%".

Note: Please select the specific channel before implement any measurement. See explanation for "`:MEASure:SOURce`"

:MEASure:FPReshoot? (query only)

Return the ratio of waveform amplitude that expressed the difference between the high level and the maximum level of the measured signal as it transitions from its high state to its low state.

Syntax

```
:MEASure:FPReshoot?
```

Returns

<NR2> with a symbol of percentage “%”.

Note: Please select the specific channel before implement any measurement. See explanation for “:MEASure:SOURce”

:MEASure:FREQuency? (query only)

Return the value of Frequency measurement.

Syntax

```
:MEASure:FREQuency?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for “:MEASure:SOURce”

:MEASure:FRFDelay? (query only)

Return the time value that difference time between the first rising edge of source1 signal and the first falling edge of source2 signal.

Syntax

```
:MEASure:FRFDelay?
```

Returns

<NR3>.

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for ":MEASure:DELAY1 and :MEASure:DELAY2"

:MEASure:FRRDelay? (query only)

Return the time value that difference time between the first rising edge of source1 signal and the first rising edge of source2 signal.

Syntax

```
:MEASure:FRRDelay?
```

Returns

<NR3>.

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for ":MEASure:DELAY1 and :MEASure:DELAY2"

:MEASure:LFFDelay? (query only)

Return the time value that difference time between the first falling edge of source1 signal and the last falling edge of source2 signal.

Syntax

```
:MEASure:LFFDelay?
```

Returns

<NR3>.

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for ":MEASure:DELAY1 and :MEASure:DELAY2"

:MEASure:LFRDelay? (query only)

Return the time value that difference time between the first falling edge of source1 signal and the last rising edge of source2 signal.

Syntax

```
:MEASure:LFRDelay?
```

Returns

<NR3>

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for ":MEASure:DELAY1 and :MEASure:DELAY2"

:MEASure:LRFDelay? (query only)

Return the time value that difference time between the first rising edge of source1 signal and the last falling edge of source2 signal.

Syntax

```
:MEASure:LRFDelay?
```

Returns

<NR3>

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for "`:MEASure:DELAY1 and :MEASure:DELAY2`"

:MEASure:LRRDelay? (query only)

Return the time value that difference time between the first rising edge of source1 signal and the last rising edge of source2 signal.

Syntax

```
:MEASure:LRRDelay?
```

Returns

<NR3>

Note: Please select these two specific measured sources before implement the delay time measurement. See explanation for "`:MEASure:DELAY1 and :MEASure:DELAY2`"

:MEASure:NWIDth? (query only)

Return the value of timing measurement of the first negative pulse in the waveform.

Syntax

```
:MEASure:NWIDth?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for "`:MEASure:SOURce`"

:MEASure:PDUTy? (query only)

Return the ratio of the positive pulse width to the signal period.

Syntax

```
:MEASure:PDUTy?
```

Returns

<NR2>. is the percentage of ratio. The range is from 1 to 99.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:PERiod? (query only)

Return the timing value of period measurement.

Syntax

```
:MEASure:PERiod?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:PWIDth? (query only)

Return the value of timing measurement of the first positive pulse in the waveform.

Syntax

```
:MEASure:PWIDth?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:RISe? (query only)

Return the value of timing measurement that taken for rising edge of the first pulse in the waveform.

Syntax

```
:MEASure:RISe?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:ROVShoot? (query only)

Return the ratio of waveform amplitude that expressed the difference between the high level and the positive peak level of the measured signal as it transitions from its low state to its high state.

Syntax

```
:MEASure:ROVShoot?
```

Returns

<NR2> with a symbol of percentage “%”.

Note: Please select the specific channel before implement any measurement. See explanation for “:MEASure:SOURce”

:MEASure:RPReshoot? (query only)

Return the ratio of waveform amplitude that expressed the difference between the low level and the negative peak level of the measured signal as it transitions from its low state to its high state.

Syntax

```
:MEASure:RPReshoot?
```

Returns

<NR2> with a symbol of percentage “%”.

Note: Please select the specific channel before implement any measurement. See explanation for “:MEASure:SOURce”

:MEASure:SOURce

Select the measured channel (channel 1, 2, 3, or 4). The default setting of measured channel is channel one.

Note: Please select the specific channel before implement any measurement.

Syntax

```
:MEASure:SOURce {1|2|3|4}
```

Arguments

1→Enable the measurement functions for channel 1

2→Enable the measurement functions for channel 2

3→Enable the measurement functions for channel 3

4→Enable the measurement functions for channel 4

Returns

<NR1>.

:MEASure:VAMPlitude? (query only)

Return the voltages of high value minus the low value.

Syntax

```
:MEASure:VAMPlitude?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VAVerage? (query only)

Return the average voltages.

Syntax

```
:MEASure:VAVerage?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VHI? (query only)

Return the value of global high voltage.

Syntax

```
:MEASure:VHI?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VLO? (query only)

Return the value of global low voltage.

Syntax

```
:MEASure:VLO?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VMAX? (query only)

Return the value of maximum amplitude.

Syntax

```
:MEASure:VMAX?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VMIN? (query only)

Return the value of minimum amplitude.

Syntax

```
:MEASure:VMIN?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VPP? (query only)

Return the value of V_{max} minus V_{min}.

Syntax

```
:MEASure:VPP?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for ":MEASure:SOURce"

:MEASure:VRMS? (query only)

Return the value of true Root Mean Square voltage.

Syntax

```
:MEASure:VRMS?
```

Returns

<NR3>.

Note: Please select the specific channel before implement any measurement. See explanation for "`:MEASure:SOURce`"

:MEMory<X>:RECall:SETup

Recalls setup form internal non-volatile memory S1~S20.

Syntax

:MEMory<X>:RECall:SETup

Arguments

<X>→Specify the location of waveform memory (1~20)

:MEMory<X>:RECall:WAVeform

Recalls waveform form internal non-volatile memory W1 ~ W20 and save the waveform to one of the reference waveform Ref A~D.

Syntax

:MEMory<X>:RECall:WAVeform <NR1>

Arguments

<X>→Specify the location of waveform memory (1~20)

<NR1>

1→ Ref A　　2→ Ref B

3→ Ref C　　4→ Ref D

:MEMory<X>:SAVe:SETup

Save the setup to one of the internal non-volatile memory S1~S20.

Syntax

:MEMory<X>:SAVe:SETup

Arguments

<X>→Specify the location of setup memory (1~20)

:MEMory<X>:SAVe:WAVeform

Save the specific reference waveform (Ref A~D) to internal non-volatile memory W1~W20.

Syntax

:MEMory<X>:SAVe:WAVeform <NR1>

Arguments

<X>→Specify the location of setup memory (1~20)

<NR1>

1→ Ref A　　2→ Ref B

3→ Ref C　　4→ Ref D

:REFResh

Refresh the waveform data of LCD screen and re-display the waveform data.

Syntax

```
:REFResh
```

:REF<X>:DISPlay

Select the reference waveform and display the waveform.

Syntax

```
:REF<X>:DISPlay {0|1}
```

```
:REF<X>:DISPlay?
```

Arguments

<X>→Specify the location of reference waveform (1|2|3|4)

1→ Ref A　　2→ Ref B

3→ Ref C　　4→ Ref D

0→ OFF　　1→ ON

Returns

<NR1>

:REF<X>:LOCate

Adjust the position of reference waveform Ref A ~ D.

Syntax

:REF<X>:LOCate <NR1>

:REF<X>:LOCate?

Arguments

<X>→ Specify the location of reference waveform (1|2|3|4)

1→ Ref A　　2→ Ref B

3→ Ref C　　4→ Ref D

<NR1> between 100 ~ -100

Returns

<NR1>

:REF<X>:SAVe

Save input waveform as reference waveform Ref A ~D

Syntax

:REF<X>:SAVe {1|2|3|4|5}

Arguments

<X>→ Specify the location of reference waveform (1|2|3|4)

1→ Ref A　　2→ Ref B

3→ Ref C　　4→ Ref D

SAVe {1|2|3|4|5}→ Specify the input waveform source

1|2|3|4→ channel 1~4.　　5→ Math

Example

:REF1:SAVe 2　Save channel two's waveform as reference waveform A.

:RUN

Controls the RUN state of trigger system. The acquisition cycle will follow each qualified trigger in the RUN state.

Syntax

`:RUN`

:STOP

Controls the STOP state of trigger system. The acquisition cycle only triggered when the `:RUN` command is received.

Syntax

`:STOP`

:TEMPlate:MAX

Sets and query the number of template used for maximum boundary.

Before running this command, the oscilloscope has to be initialized as “Normal Template” mode, please add the command of “`:TEMPlate:MODe 0`” in order to change the settings.

Syntax

```
:TEMPlate:MAX {0|20}
```

```
:TEMPlate:MAX?
```

Arguments

`0`→ Ref A

`1~20`→ Internal memory W1~W20

Returns

<NR1>

Note: A template can be defined as either MAX or MIN template only.

Note B: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of “`:GONogo:FUNCtion 1`” in order to initial the oscilloscope.

:TEMPlate:MIN

Sets and query the number of template used for minimum boundary.

Before running this command, the oscilloscope has to be initialized as “Normal Template” mode, please add the command of “`:TEMPlate:MODe 0`” in order to change the settings.

Syntax

```
:TEMPlate:MIN {0|20}
```

```
:TEMPlate:MIN?
```

Arguments

`0`→ Ref B

`1~20`→ Internal memory W1~W20

Returns

<NR1>

Note: A template can be defined as either MAX or MIN template only.

Note B: Before running this command, the oscilloscope has to be initialized, please add the command of “`:GONogo:FUNCtion 1`” in order to initial the oscilloscope.

:TEMPlate:MODe

Sets and query the template mode. When the Auto mode selected, the auto template source is coming from CH1~4. When the normal mode selected, the template can be selected from the internal memory of W1~W20, RefA or RefB.

Syntax

```
:TEMPlate:MODe {0|1}
```

```
:TEMPlate:MODe?
```

Arguments

0→ Select normal template. 1→ Select auto template.

Returns

<NR1>

:TEMPlate:POSition:MAX

Sets and query the position of maximum template. This command will not change the position value which stored in internal memory (W1~W20), unless you save the template by the command of `:TEMPlate:SAVe:MAXimum`. That means that if the oscilloscope is re-opened again without present template storage, the oscilloscope will recall the previous stored position value from internal memory automatically.

Before running this command, the oscilloscope has to be initialized as "Normal Template" mode, please add the command of "`:TEMPlate:MODe 0`" in order to change the settings.

Syntax

```
:TEMPlate:POSition:MAX <NR2>
```

```
:TEMPlate:POSition:MAX?
```

Arguments

<NR2> is the desired template position. The range is from -12.0 ~ +12.0 div.

Returns

<NR2>

Note A: 1 division = 25 dots

Note B: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of "`:GONogo:FUNCtion 1`" in order to initial the oscilloscope.

:TEMPlate:POSition:MIN

Sets and query the position of minimum template. This command will not change the position value that stored in internal memory, unless you save the template by the command of :TEMPlate:SAVe:MINimum. That means that if the oscilloscope is re-opened again without present template storage, the oscilloscope will recall the previous stored position value from internal memory automatically.

Before running this command, the oscilloscope has to be initialized as "Normal Template" mode, please add the command of ":TEMPlate:MODe 0" in order to change the settings.

Syntax

:TEMPlate:POSition:MIN <NR2>

:TEMPlate:POSition:MIN?

Arguments

<NR2> is the desired template position. The range is from -12.0 ~ +12.0 div.

Returns

<NR2> Div

Note: 1 division = 25 dots

Note: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of ":GONogo:FUNCtion 1" in order to initial the oscilloscope.

:TEMPlate:SAVe:AUTo

Save the Auto template. This command is equivalent to pressing the “Save & Create” (**F4**) function key of the auto template editing menu.

Before running this command, the oscilloscope has to be initialized as “Auto Template” mode, please add the command of “`:TEMPlate:MODe 1`” in order to change the settings.

Syntax

```
:TEMPlate:SAVe:AUTo
```

Note: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of “:GONogo:FUNCtion 1” in order to initial the oscilloscope.

:TEMPlate:SAVe:MAXimum

Save the Maximum template. This command is equivalent to pressing the “Save & Create” (**F4**) function key of the maximum template editing menu.

Before running this command, the oscilloscope has to be initialized as “Auto Template” mode, please add the command of “`:TEMPlate:MODe 0`” in order to change the settings.

Syntax

`:TEMPlate:SAVe:MAXimum`

Note: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of “`:GONogo:FUNCtion 1`” in order to initial the oscilloscope.

:TEMPlate:SAVe:MINimum

Save the Minimum template. This command is equivalent to pressing the “Save & Create” (**F4**) function key of the minimum template editing menu.

Before running this command, the oscilloscope has to be initialized as “Auto Template” mode, please add the command of “`:TEMPlate:MODe 0`” in order to change the settings.

Syntax

`:TEMPlate:SAVe:MAXimum`

Note: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of “`:GONogo:FUNCtion 1`” in order to initial the oscilloscope.

:TEMPlate:TOLerance

Sets and query the tolerance of auto template.

Before running this command, the oscilloscope has to be initialized as “Auto Template” mode, please add the command of “:TEMPlate:MODe 1” in order to change the settings.

Syntax

:TEMPlate:TOLerance <NR2>

:TEMPlate:TOLerance?

Arguments

<NR2> is the desired tolerance percentage. The range is from 0.4~40 (or 0.4% to 40%).

Returns

<NR2> %

Note: Before input any template editing commands, the oscilloscope has to be initialized, please add the command of “:GONogo:FUNCtion 1” in order to initial the oscilloscope.

TIMebase:DELay

Sets the horizontal position (delay timebase) parameter.

Syntax

```
:TIMebase:DELay <NR3>
```

```
:TIMebase:DELay?
```

Arguments

<NR3> is the desired delay time.

Returns

<NR3>

:TIMebase:SCALe

Sets the horizontal timebase scale per division (SEC/DIV).

Syntax

`:TIMebase:SCALe <NR3>`

`:TIMebase:SCALe?`

Arguments

Sec/div	**NR3**	Sec/div	**NR3**	Sec/div	**NR3**
1ns	**1e^{-9}**	5 μs	**5e^{-6}**	25ms	**25e^{-3}**
2.5ns	**2.5e^{-9}**	10 μs	**10e^{-6}**	50ms	**50e^{-3}**
5ns	**5e^{-9}**	25 μs	**25e^{-6}**	100ms	**100e^{-3}**
10ns	**10e^{-9}**	50 μs	**50e^{-6}**	250ms	**250e^{-3}**
25ns	**25e^{-9}**	100 μs	**100e^{-6}**	500ms	**500e^{-3}**
50ns	**50e^{-9}**	250 μs	**250e^{-6}**	1s	**1**
100ns	**100e^{-9}**	500 μs	**500e^{-6}**	2.5s	**2.5**
250ns	**250e^{-9}**	1ms	**1e^{-3}**	5s	**5**
500ns	**500e^{-9}**	2.5ms	**2.5e^{-3}**	10s	**10**
1 μs	**1e^{-6}**	5ms	**5e^{-3}**		
2.5 μs	**2.5e^{-6}**	10ms	**10e^{-3}**		

`<NR3>` is the desired timebase scale per division.

Returns

<NR3>

Examples

`:TIMebase:SCALe 5e-3` sets timebase scale to 5ms per division.

:TIMebase:SWEep

Selects the horizontal timebase sweep mode. This command is equivalent to setting the horizontal menu.

Syntax

`:TIMebase:SWEep <0|1|2|3|4>`

`:TIMebase:SWEep?`

Arguments

`0`→Main timebase

`1`→Window

`2`→Window Zoom

`3`→Roll mode

`4`→XY mode

Returns

<NR1>

:TIMebase:WINDow:DELay

Setting and query the zoomed area (the gray color area) for window zoomed display.

Syntax

`:TIMebase:WINDow:DELay <NR3>`

`:TIMebase:WINDow:DELay?`

Arguments

`<NR3>` is the desired position (delay time).

Returns

<NR3>

:TIMebase:WINDow:SCALe

Sets and query the scale (length) of the windows zoomed timebase.

Syntax

`:TIMebase:WINDow:SCALe <NR3>`

`:TIMebase:WINDow:SCALe?`

Arguments

`<NR3>` is the desired scale (length) of the windows zoomed timebase.

Returns

<NR3>

:TRIGger:COUPle

Select and query the type of trigger coupling.

Syntax

`:TRIGger:COUPle <0|1>`

`:TRIGger:COUPle?`

Arguments

`0`→AC `1`→DC

Returns

<NR1>

:TRIGger:DELay:TIMe

Sets and query the user-defined delay trigger time.

Syntax

```
:TRIGger:DELay:TIMe <NR3>
```

```
:TRIGger:DELay:TIMe?
```

Arguments

<NR3> is the desired user-defined delay time. The range is from 100ns~1.3ms.

Returns

<NR3>

Note: Please select the specific delay type before implement any measurement. See explanation for ":TRIGger:DELay:TYPe"

:TRIGger:DELay:EVENt

Sets and query the user-defined delay trigger events.

Syntax

```
:TRIGger:DELay:EVENt <NR1>
```

```
:TRIGger:DELay:EVENt?
```

Arguments

<NR1> is the desired user-defined delay trigger events. The range is from 2~65000.

Returns

<NR1>

Note: Please select the specific delay type before implement any measurement. See explanation for ":TRIGger:DELay:TYPe"

:TRIGger:DELay:LEVel

Sets and query the user-defined start trigger signal level.

Syntax

:TRIGger:DELay:LEVel <NR3>

:TRIGger:DELay:LEVel?

Arguments

<NR3> is the desired user-defined start trigger signal level. The range is ±12.

Returns

<NR3>

:TRIGger:DELay:MODe

Select and query the different start trigger (i.e. external trigger) signal level.

Syntax

:TRIGger:DELay:MODe <0|1|2>

:TRIGger:DELay:MODe?

Arguments

0→TTL

1→ECL

2→USR

Returns

<NR1>

:TRIGger:DELay:TYPe

Select and query the different delay trigger settings.

Syntax

```
:TRIGger:DELay:TYPe <0|1>
```

```
:TRIGger:DELay:TYPe?
```

Arguments

0→Time setting

1→Event setting

Returns

<NR1>

:TRIGger:FREQuency? (query only)

Return the readout value of trigger frequency counter.

Syntax

```
:TRIGger:FREQuency?
```

Returns

<NR3>.

:TRIGger:LEVel

Select and query the trigger level.

Syntax

`:TRIGger:LEVel <NR3>`

`:TRIGger:LEVel?`

Arguments

`<NR3>` is the desired trigger level voltage.

Returns

<NR3>

:TRIGger:MODe

Select and query the trigger mode.

Syntax

`:TRIGger:MODe <0|1|2|3>`

`:TRIGger:MODe?`

Arguments

`0`→Auto Level

`1`→Auto

`2`→Normal

`3`→Single

Returns

<NR1>

:TRIGger:NREJ

Switch and query the noise rejection mode.

Syntax

```
:TRIGger:NREJ <0|1>
```

```
:TRIGger:NREJ?
```

Arguments

0→OFF 1→ON

Returns

<NR1>

:TRIGger:PULSe:MODe

Switch and query different pulse trigger type.

Syntax

```
:TRIGger:PULSe:MODe <0|1|2|3>
```

```
:TRIGger:PULSe:MODe?
```

Arguments

0→**<**

1→**>**

2→**=**

3→≠

Returns

<NR1>

:TRIGger:PULSe:TIMe

Select the time value for pulse width. The setting range is from 20ns to 10s.

Syntax

`:TRIGger:PULSe:TIMe <NR3>`

`:TRIGger:PULSe:TIMe?`

Arguments

`<NR3>` is the desired time value of pulse width, the unit is in second.

Returns

<NR3>

:TRIGger:REJect

Select and query the frequency rejection mode.

Syntax

`:TRIGger:REJect <0|1|2>`

`:TRIGger:REJect?`

Arguments

`0`→OFF

`1`→Low frequency reject mode

`2`→High frequency reject mode

Returns

<NR1>

:TRIGger:SLOP

Switch and query the rising or falling trigger slope.

Syntax

```
:TRIGger:SLOP <0|1>

:TRIGger:SLOP?
```

Arguments

0→Rising slope　　1→Falling slope

Returns

<NR1>

:TRIGger:SOURce

Select and query the trigger source.

Syntax

```
:TRIGger:SOURce <0|1|2|3|4|5>

:TRIGger:SOURce?
```

Arguments

0→Channel one　　1→Channel two

2→Channel three　　3→Channel four

4→External trigger　　5→AC line voltage

Returns

<NR1>

:TRIGger:TYPe

Select and query the trigger type.

Syntax

```
:TRIGger:TYPe <0|1|2|3>
```

```
:TRIGger:TYPe?
```

Arguments

0→Edge

1→Video

2→Pulse

3→Delay

Returns

<NR1>

:TRIGger:VIDeo:FIELd

Select and query the field on which the video trigger mode will be triggered.

Syntax

`:TRIGger:VIDeo:FIELd <0|1|2>`

`:TRIGger:VIDeo:FIELd?`

Arguments

`0`→Line `1`→Odd frame (Field 1)

`2`→Even frame (Field 2)

Returns

<NR1>

:TRIGger:VIDeo:LINe

Select and query the specified line for video signal.

Syntax

`:TRIGger:VIDeo:LINe <NR1>`

`:TRIGger:VIDeo:LINe?`

Arguments

`<NR1>` is the desired line.

For NTSC system, the range of line is from 1~263 for Odd frame, 1-262 for even frame.

For PAL system, the range of line is from 1~313 for Odd frame, 1-312 for even frame.

Returns

<NR1>

:TRIGger:VIDeo:POLarity

Select and query the input video polarity.

Syntax

`:TRIGger:VIDeo:POLarity <0|1>`

`:TRIGger:VIDeo:POLarity?`

Arguments

`0`→Positive-going sync pulses

`1`→Negative-going sync pulses

Returns

<NR1>

:TRIGger:VIDeo:TYPe

Select and query the TV broadcast system.

Syntax

`:TRIGger:VIDeo:TYPe <0|1|2>`

`:TRIGger:VIDeo:TYPe?`

Arguments

`0`→PAL

`1`→NTSC

`2`→SECAM

Returns

<NR1>

:USB:RECall:SETup

Recall the setup from USB flash disk.

Syntax

```
:USB:RECall:SETup
```

Note: This command always picks up the latest setting file as target (recalled) file.

:USB:RECall:WAVeform

Recall the saved waveform form USB flash disk and save to Ref A ~ D

Syntax

```
:USB:RECall:WAVeform <NR1>
```

Arguments

```
<NR1>

1→ Ref A                    2→ Ref B

3→ Ref C                    4→ Ref D
```

Note: This command always picks up the latest waveform file as target (recalled) file.

:USB:SAVe:ALL

Save all oscilloscope settings, waveform raw data, and image to USB flash disk.

Syntax

```
:USB:SAVe:ALL
```

:USB:SAVe:IMAGe

Save present waveform image to USB flash disk.

Syntax

```
:USB:SAVe:IMAGe
```

:USB:SAVe:SETup

Save all settings of oscilloscope to USB flash disk.

Syntax

```
:USB:SAVe:SETup
```

:USB:SAVe:WAVeform

Save present waveform raw data to USB flash disk

Syntax

```
:USB:SAVe:WAVeform <NR1>
```

Arguments

```
<NR1>

1→ Ref A                    2→ Ref B

3→ Ref C                    4→ Ref D
```

Appendix A: How can we convert the hexadecimal format to a floating point format

Question: *As the previous example listed on page 32, how can the hexadecimal value of "0×34 0×56 0×BF 0×94" transfer to 200ns?*

Answer:

just use the attached C language program:

```
#include <stdio.h>
int main()
{
   union data
   {
      char a[4];
      float f;
   } myData;
   myData.a[0]=0x94;   /* little-endian byte ordering here,         */
   myData.a[1]=0xBF;   /* so, the last of 0x94 should be placed     */
   myData.a[2]=0x56;   /* in the first order.                        */
   myData.a[3]=0x34;
   printf("Here is the Data:\n%0x\n%0x\n%0x\n%0x\n%.3e\n",\
            myData.a[0]&0xff,
            myData.a[1]&0xff,
            myData.a[2]&0xff,
            myData.a[3]&0xff,\
            myData.f );
   return 0;
}
```

and the output result is following: 2.000e -007=200ns