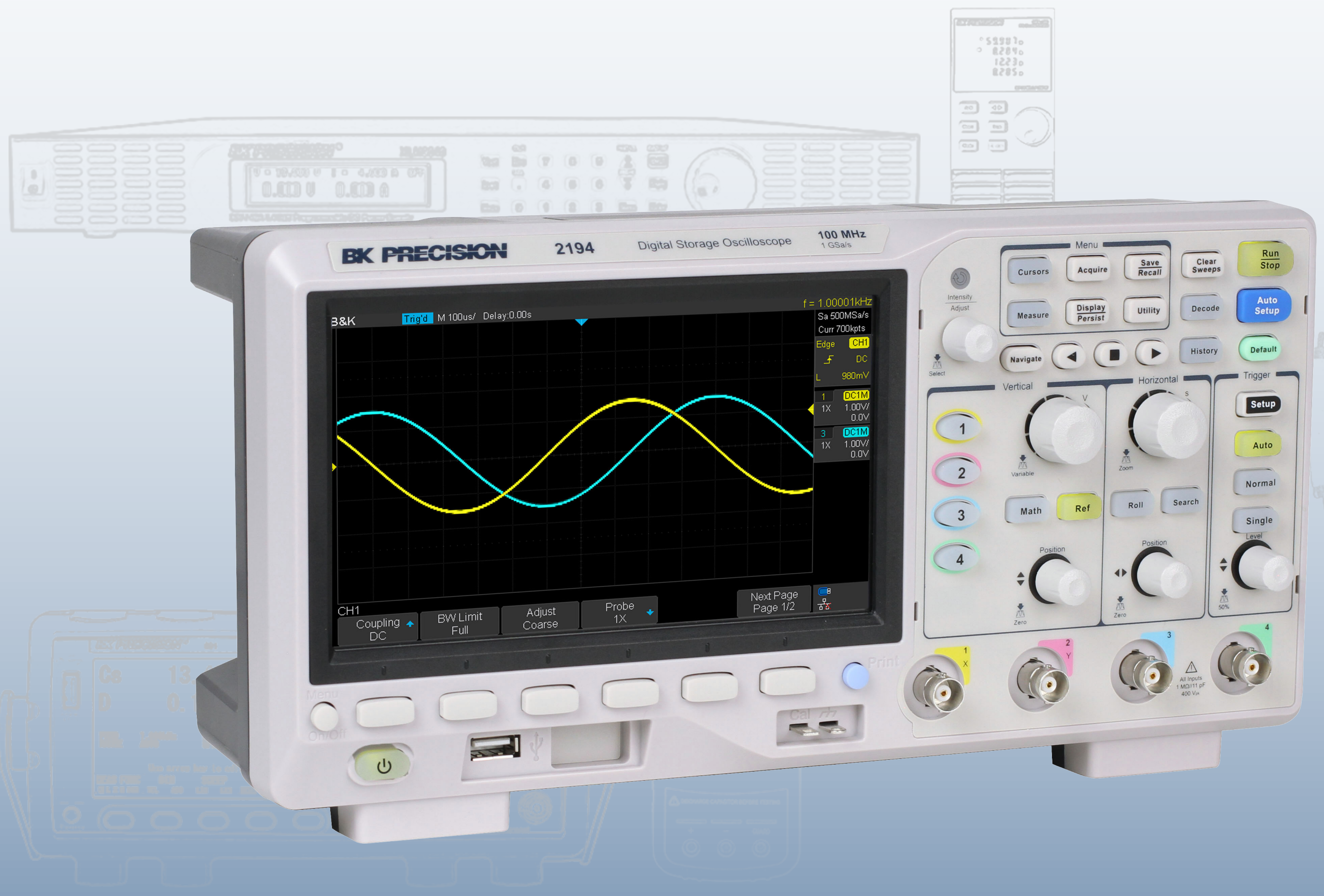
Programming Manual

Model 2194

Digital Storage Oscilloscope



bkprecision.com

B&K PRECISION

Contents

About Commands & Queries

This section lists and describes the remote control commands and queries recognized by the instrument. All commands and queries can be executed in either local or remote state.

The description, command syntax, query syntax, example and respond can be found in a section. The commands are given in both long and short form. All examples are shown in short form. Queries perform actions such as obtaining information are recognized by the question mark (?) following the header.

1.1 How They are Listed

The commands are listed by subsystem and alphabetical order according to their short form.

1.2 How They are Described

In the descriptions themselves, a brief explanation of the function performed is given. This is followed by a presentation of the formal syntax, with the header given in Upper-and-Lower-Case characters and the short form derived from it in ALL UPPER-CASE characters. Where applicable, the syntax of the query is given with the format of its response.

1.3 When can they be used?

The commands and queries listed here can be used for 2194 Digital Storage Oscilloscope.

1.4 Command Notation

The following notation is used in the commands:

< > Angular brackets enclose words that are used as placeholders, of which there are two types: the header path and the data parameter of a command.

:= A colon followed by an equals sign separates a placeholder from the description of the type and range of values that may be used in a command instead of the placeholder.

{ } Braces enclose a list of choices, one of which one must be made.

[ ] Square brackets enclose optional items.

... An ellipsis indicates that the items both to its left and right may be repeated a number of times.

Common Command Introduction

The IEEE 488.2 standard defines the common commands used for querying the basic inSyntaxion of the instrument or executing basic operations. These commands usually start with "*" and the length of the keywords of the command is usually 3 characters.

Short	Long Form	Subsystem	What Command/Query does
*IDN?	*IDN?	SYSTEM	Returns a string that uniquely identifies the instrument.
*OPC	*OPC	SYSTEM	Generates the OPC message in the standard event status register when all pending overlapped operations have been completed.
*OPC?	*OPC?	SYSTEM	Returns an ASCII "+1" when all pending overlapped operations have been completed.
*RST	*RST	SYSTEM	Initiates a device reset.

Table 2.1

2.1 *IDN?

Description The *IDN? query causes the instrument to identify itself. The response comprises manufacturer, model, serial number, software version and firmware version.

Query Syntax *IDN?

Response Syntax *IDN, <device id>,<model>,<serial number>,<Uboot-OS version><software version>, <hardware version>.

<device id>:="BK" is used to identify instrument.

<model>:= A model identifier less than 14 characters will contain the model number.

<serial number>:= Each product has its own number, the serial number can labeled product uniqueness.

<Uboot-OS version>:= The Uboot-OS revision of the instument.

<software version>:= A serial numbers about software version.

Example *IDN?

Returns: BK Precision,2194,XXXXXXXXXXXXX,2.1.1.1.5R7

2.2 *OPC

Description	The operation complete command causes the device to generate the operation complete message in the Standard Event Status Register, on completion of the selected device operation. The operation complete query places an ASCII character 1 in the output queue on completion of the selected device operation.
Command Syntax	*OPC
Query Syntax	*OPC?
Ecample	OUTP:STAT 1;*OPC
Response Syntax	1

2.3 *RST

Description	The *RST command initiates a device reset. The *RST recalls the default setup equivalent to the **Default** key on the front panel..
Command Syntax	*RST
Example	*RST
REIated Commands	:RECall:FDEFault :RECall:SETup :SAVE:DEFault :SAVE:SETup

Communication

CHDR

3.1 COMM_HEADER,CHDR

Description

The COMM_HEADER command controls the way the oscilloscope formats responses to queries. There are three response formats: LONG, in which responses start with the long form of the header word; SHORT, where responses start with the short form of the header word; and OFF, for which headers are omitted from the response and units in numbers are suppressed.

Note:

Unless you request otherwise, the SHORT response format is used.

This command does not affect the interpretation of messages sent to the oscilloscope. Headers can be sent in their long or short form regardless of the COMM_HEADER setting.

Querying the vertical sensitivity of Channel 1 may result in one of the following responses:

CHDR	Response
LONG	C1:VOLT_DIV 1.00E+01V
SHORT	C1:VDIV 1.00E+01V
OFF	1.00E+01

Table 3.1 Response Format

Command Syntax

Comm_HeaDeR <mode>
<mode> : = SHORT, LONG, OFF

Query Syntax

Comm_HeaDeR?

Response Format

Comm_HeaDeR <mode>

Example

The following code sets the response header format to SHORT:

Command message: CHDR SHORT

Acquire Commands

The ACQUIRE subsystem controls the way in which waveforms are acquired. These commands set the parameters for acquiring and storing data.

ARM	STOP	ACQW
AVGA	MSIZ	SAST?
SARA?	SANU?	SXSA

4.1 ARM_ACQUISITION

Description	The ARM_ACQUISITION command enables the signal acquisition process by changing the acquisition state (trigger mode) from “stopped” to “single”.
Commnad Syntax	ARM acquisition

Example The following command enables signal acquisition:

1. Set the trigger mode to single, and input a signal which can be triggered. Once triggered, you can see the state of acquisition changes to stop. Send the query.
   - **Query message**: INR?
   - **Response message**: INR 8193 (trigger ready)
2. Send the query again to clear the register.
   - **Query message**: INR?
   - **Response message**: INR 0
3. Now, send the command to start a new signal acquisition.
   - **Command message**: ARM
4. Send the query to see the effect of ARM.
   - **Query message**: INR?
   - **Response message**: INR 8193

Related Commands **STOP**
INR?
TRIG_MODE

4.2 STOP

Description	The STOP command immediately stops the acquisition of a signal. If the trigger mode is AUTO or NORM.
Commnad Syntax	STOP
Query Syntax	
Example	The following command stops the acquisition process. **Command message**: STOP
Related Commands	**ARM**, **TRIG_MODE**

4.3 ACQUIRE_WAY

Description
The ACQUIRE_WAY command specifies the acquisition mode.

Commnad Syntax
ACQUIRE_WAY <mode>[,<time>]
<mode>:={SAMPLING,PEAK_DETECT,AVERAGE,HIGH_RES}
<time>:={4,16,32,64,128,256,512,...}

- SAMPLING: Sets the oscilloscope in the normal mode.
- PEAK_DETECT: Sets the oscilloscope in the peak detect mode.
- AVERAGE: Sets the oscilloscope in the averaging mode.
- HIGH_RES: Sets the oscilloscope in the enhanced resolution mode (also known as smoothing). This is essentially a digital boxcar filter and is used to reduce noise at slower sweep speeds.

Query Syntax
ACQUIRE_WAY?

Response Format
ACQUIRE_WAY <mode>[,<time>]

Example
The following command sets the acquisition mode to average mode and also sets the average time to 16.
Command message: ACQW AVERAGE,16

Related Commands
AVGA

4.4 AVERAGE_ACQUIRE

Description
The AVERAGE_ACQUIRE command selects the average times of average acquisition.

The AVERAGE_ACQUIRE? query returns the currently selected count value for average mode.

Commnad Syntax
AVERAGE_ACQUIRE <time>
<time>:= {4,16,32,64,128,256,...}

Query Syntax
AVERAGE_ACQUIRE?

Response Format
AVERAGE_ACQUIRE <time>

Example
The following command turns the average times of average acquisition to 16.
Command message: AVGA 16

Related Commands
ACQUIRE_WAY

4.5 MEMORY_SIZE

Description	The MEMORY_SIZE command sets the maximum depth of memory. The MEMORY_SIZE? query returns the maximum depth of memory.
Commnad Syntax	MEMORY_SIZE <size> <size>:={7K,70K,700K,7M}
Query Syntax	MEMORY_SIZE?
Response Format	MEMORY_SIZE <size>
Example	The following command sets the maximum depth of memory to 14M in interleave mode. **Command message**: MSIZ 7M

4.6 SAMPLE_STATUS

Description	Returns the acquisition status of the scope.
Query Syntax	SAST?
Response Format	SAST <status>
Example	The following query returns the acquisition status of the scope. **Query message**: SAST? **Response message**: SAST Trig'd

4.7 SAMPLE_RATE?

Description	Returns the sample rate of the scope.
Query Syntax	SARA?
Response Format	SARA <value> DI:SARA <value> Numerical value in E-notation with SI unit, such as 5.00E+08Sa/s
Example	The following query returns the sample rate of the analog channel. **Query message**: SARA? **Response message**: SARA 5.00E+05Sa/s

4.8 SAMPLE_NUM?

Description	Returns the number of data points that the hardware will acquire based on the horizontal scale and memory/acquisition depth selections.
Query Syntax	SANU? <channel> <channel>:={C1,C2,C3,C4}
Response Format	SANU <value> Numerical value in E-notation with SI unit, such as 7.00E+05pts.
Example	The following query returns the number of sampled points available from last acquisition from Channel 2. **Query message**: SANU? C2 **Response message**: SANU 7.00E+05pts

4.9 SINXX_SAMPLE

Description	Sets the interpolation mode.
Commnad Syntax	SINXX_SAMPLE <state> <state>:={ON,OFF} – **ON**: sine interpolation. – **OFF**: linear interpolation.
Query Syntax	SINXX_SAMPLE?
Response Format	SINXX_SAMPLE <state>
Example	The following command sets the way of the interpolation to sine interpolation. **Command message**: SXSA ON

4.10 XY_DISPLAY

Description	The XY_DISPLAY command enables or disables to display the XY format.
Commnad Syntax	XY_DISPLAY <state> <state>= {ON, OFF}
Query Syntax	XY_DISPLAY?
Response Format	XY_DISPLAY <state>
Example	The following command enables to display the XY format: **Command message**: XYDS

Autoset Commands

The AUTOSET subsystem commands control the function of automatic waveform setting. The oscilloscope will automatically adjust the vertical position, the horizontal time base and the trigger mode according to the input signal to make the waveform display to the best state.

ASET

5.1 AUTO_SETUP

Description	The AUTO_SETUP command attempts to identify the waveform type and automatically adjusts controls to produce a usable display of the input signal.
Commnad Syntax	AUTO_SETUP
Example	The following command instructs the oscilloscope to perform an auto-setup. **Command message**: ASET

Channel Commands

The CHANNEL subsystem commands control the analog channels. Channels are independently programmable for offset, probe, coupling, bandwidth limit, inversion, and more functions. The channel index (1, 2, 3, or 4) specified in the command selects the analog channel that is affected by the command.

ATTN	BWL	CPL
OFST	SKEW	TRA
UNIT	VDIV	INVS

6.1 ATTENUATION

Description
Specifies the probe attenuation factor for the selected channel.This command does not change the actual input sensitivity of the oscilloscope. It changes the reference constants for scaling the display factors, for making automatic measurements, and for setting trigger levels.

Commnad Syntax
<channel>:ATTENUATION <attenuation>
<channel>:={C1,C2,C3,C4}
<attenuation>:={0.1,0.2,0.5,1,2,5,10,20,50,100,200,500,10 00,2000,5000,10000}

Query Syntax
<channel>:ATTENUATION?

Response Format
<channel>:ATTENUATION <attenuation>

Example
The following command sets the attenuation factor of Channel 1 to 100:1. To ensure the data matches the true signal voltage values, the physical probe attenuation must match the scope attenuation values for that input channel.

Command message: C1:ATTN 100

Related Commands
VDIV, **OFST**

6.2 BANDWIDTH_LIMIT

Description
Enables/disables the bandwidth-limiting low-pass filter. If the bandwidth filters are on, it will limit the bancdwidth to reduce display noise. When you turn bandwidth limit on, the bandwidth limit value is set to 20 MHz. It also filters the signal to reduce noise and other unwanted high frequency components.

Commnad Syntax
BANDWIDTH_LIMIT<channel>,<mode>[,<channel>,<mode>[,<channel>,<mode>
[, <channel>,<mode>]]]
<channel>:={C1,C2,C3,C4}
<mode>:={ON,OFF}

Query Syntax
BANDWIDTH_LIMIT?

Response Format
BANDWIDTH_LIMIT <channel>,<mode>[,<channel>,<mode>[,<channel>,<mod e>
[,<channel>,<mode>]]]

Example
The following command turns on the bandwidth filter for all channels.

Command message: BWL C1,ON,C2,ON,C3,ON,C4,ON

The following command turns the bandwidth filter on for Channel 1 only.

Command message: BWL C1,ON

6.3 COUPLING

Description Selects the coupling mode of the specified input channel.

Commnad Syntax <channel>:COUPLING <coupling>
<channel>:={C1,C2,C3,C4}
<coupling>:={A1M,D1M,GND}

- **A**: alternating current.
- **D**: direct current.
- **1M**: 1MΩ input impedance.

Query Syntax <channel>:COUPLING?

Response Format <channel>:COUPLING <coupling>

Example The following command sets the coupling of Channel 2 to 1 MΩ, DC.

Command message: C2:CPL D1M

6.4 OFFSET

Description The OFFSET command allows adjustment of the vertical offset of the specified input channel. The maximum ranges depend on the fixed sensitivity setting.

Commnad Syntax <channel>: OFfSeT <offset>
<channel> : = {C1, C2, C3,C4}
<offset> : = See specifications.

Query Syntax <channel>: OFfSeT?

Response Format <channel>: OFfSeT <offset>
<offset>:= Numerical value in E-notation with SI unit.

Example The following command sets the offset of Channel 2 to -3 V:

Command message: C2: OFST -3V

Related Commands **VDIV**, **ATTN**

6.5 SKEW

Description Sets the skew value of the specified trace.

Commnad Syntax <trace>:SKEW <skew>
<trace> : = {C1,C2,C3,C4 }
<skew>: = {-100 ns to +100 ns}

Query Syntax <trace>:SKEW <skew>

Response Format Numerical value in E-notation with SI unit, such as 9.99E-08S.

Example The following command sets skew value of Channel 1 to 3ns.

Command message: C1:SKEW 3NS

6.6 TRACE

Description	Enable/disable the trace of the specified channel.

Commnad Syntax	<trace>:TRACE <mode>
	<trace>:={C1,C2,C3,C4}
	<mode>:={ON,OFF}

Query Syntax	<trace>:TRACE?
Response Format	<trace>:TRACE <mode>
Example	The following command displays Channel 1. **Command message**: C1:TRA ON

6.7 UNIT

Description	Sets the unit of the specified trace. Measurement results, channel sensitivity, and trigger level will reflect the measurement units you select.

Commnad Syntax	<channel>:UNIT <type>
	<channel>:={C1,C2,C3,C4}
	<type>:={V,A}

Query Syntax	<channel>:UNIT?
Response Format	<channel>:UNIT <type>
Example	The following command sets the unit of Channel 1 to V. **Command message**: C1:UNIT V

6.8 VOLT_DIV

Description	Sets the vertical sensitivity in Volts/div.If the probe attenuation is changed, the scale value is multiplied by the probe's attenuation factor.

Commnad Syntax	<channel>:VOLT_DIV <v_gain>
	<channel>:={C1,C2,C3,C4}
	<v_gain>:= {500uV to 10V}
	
	If there is no unit (V/mV/uV) added, it defaults to volts (V).

Query Syntax	<channel>:VOLT_DIV?
Response Format	<channel>:VOLT_DIV <v_gain> <v_gain>:= Numerical value in E-notation with SI unit.
Example	The following command sets the vertical sensitivity of Channel 1 to 50 mV/div. **Command message**: C1:VDIV 50mV
Related Commands	**ATTN**

6.9 INVERTSET

Description	Mathematically inverst hte specified traces or math waveform.

Commnad Syntax	<trace>:INVERTSET <state>
	<trace>:={C1,C2,C3,C4,MATH}
	<state>:= {ON,OFF}

Query Syntax	<trace>:INVERTSET?
Response Format	<trace>:INVERTSET <state>
Example	The following command inverts the trace of Channel 1. **Command message**: C1:INVS ON

Cursor Commands

The CURSOR subsystem commands set and query the settings of X-axis markers(X1 and X2 cursors) and the Y-axis markers (Y1 and Y2 cursors). You can set and query the marker mode and source, the position of X and Y cursors, and query delta X and delta Y cursor values.

CRMS

CRST

CRTY

CRVA?

7.1 CURSOR_MEASURE

Description

Specifies s the type of cursor or parameter measurement to be displayed.

Commnad Syntax

CURSOR_MEASURE <mode> <mode> := {OFF,MANUAL,TRACK}
OFF: close the cursors.
MANUAL: manual mode.
TRACK: track mode.

Query Syntax

CURSOR_MEASURE?

Response Format

CURSOR_MEASURE <mode>

Example

The following command sets cursor function to manual.
Command message:CRMS MANUAL

Related Commands

CRVA?
CRST

7.2 CURSOR_SET

Description

Configure the position of any one of the four independent cursors at a given screen location. The positions of the cursors can be modified or queried even if the required cursor is not currently displayed on the screen. When setting a cursor position, a trace must be specified, relative to which the cursor will be positioned.

Commnad Syntax

<trace>:CURSOR_SET <cursor>,<position>[,<cursor>,<position>[,<cursor>,<positi on> [,<cursor>,<position>]]]

<trace>:={C1,C2,C3,C4}
<cursor>:={VREF,VDIF,TREF,TDIF,HRDF,HDIF}

- **VREF**: The voltage-value of Y1 (curA) under manual mode.
- **VDIF**: The voltage-value of Y2 (curB) under manual mode.
- **TREF**: The time value of X1 (curA) under manual mode.
- **TDIF**: The time value of X2 (curB) under manual mode.
- **HREF**: The time value of X1 (curA) under track mode.
- **HDIF**: The time value of X2 (curB) under track mode.

<position> := -(grid/2)*DIV to (grid/2)*DIV
when <cursor> := {TREF,TDIF, HRDF, HDIF} (horizontal) grid: The grid numbers in horizontal direction.
<position> := -4*DIV to 4*DIV
when <cursor> := {VREF,VDIF} (vertical)

Note:

The horizontal position range is related to the size of screen. The unit to the position value must be added.

Query Syntax
<trace>:CURSOR_SET?
<cursor>[,<cursor>[,<cursor>[,<cursor>]]]
<cursor> := {VREF,VDIF,TREF,TDIF,HREF,HDIF}

Response Format
<trace>:CURSOR_SET
<cursor>,<position>[,<cursor>,<position>[,<cursor>,<positi on>[,<cursor>,<position>]]]

Example
When the current time base is 1 us, vdiv is 500 mV, the cursor mode is manual, the following command sets the X1 positions to -3 DIV, Y2 position to −1 DIV, using Channel 1 as a reference.
Command message: C1:CRST TREF,-3us,VDIF,-500mV

Related Commands
CRMS
CRVA?

7.3 CURSOR_TYPE

Description
Specififes the type of cursor to be displayed when the cursor mode is manual.

Commnad Syntax
CURSOR_TYPE <type>
<mode> := {X,Y,X-Y}

Query Syntax
CURSOR_TYPE?

Response Format
CURSOR_TYPE <type>

Example
The following command sets cursor type to Y.
Command message: CRTY Y

Related Commands
CRMS

7.4 CURSOR_VALUE?

Description
Retruns the values measured by the specified cursors for the given trace.

Query Syntax
trace>:CURSOR_VALUE? <mode>
<trace> := {C1,C2,C3,C4}
<mode> := {HREL,VREL}

- **HREL**: Returns the delta time value, reciprocal of delta time value, X1 (curA) time value and X2 (curB) time value.
- **VREL**: Returns the delta volt value, Y1 (curA) volt value and Y2 (curB) volt value under manual mode.

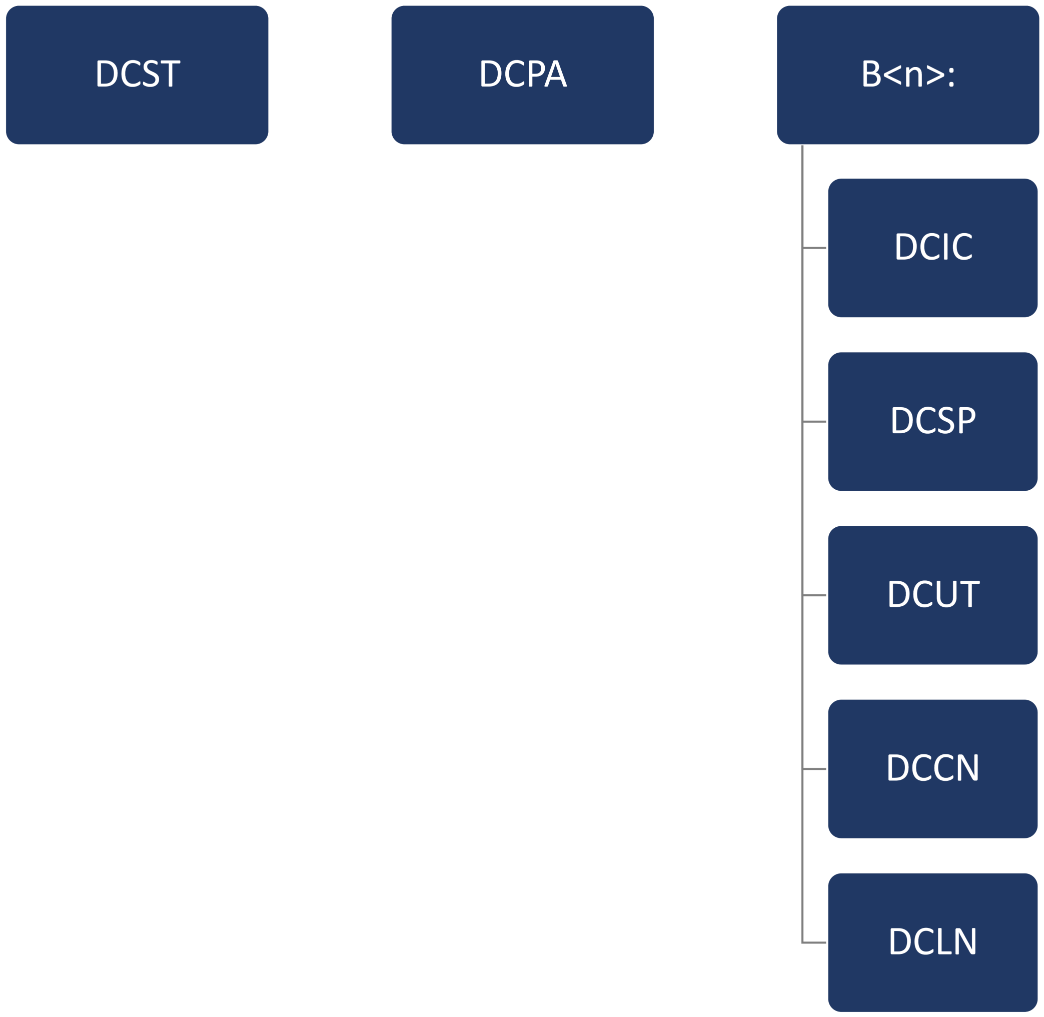
Response Format
<trace>:CURSOR_VALUE HREL,<delta>,<1/delta>,<value1>,<value2>
<trace>:CURSOR_VALUE VREL,<delta>,<value1>,<value2>

Example
When the cursor mode is manual, and the cursor type is Y, the following query returns the vertical value on channel 1.
Query message: C1:CRVA? VREL

Related Commands
CRMS

Decode Commands

The DECODE subsystem commands control the serial protocols and parameters for each serial bus decode. The commands control the serial decode bus viewing, and other options.



8.1 DCST

Description Sets the state of decode.

Commnad Syntax DCST <state>
<state> := {OFF,ON}

Query Syntax DCST?

Response Format DCST <state>

Example The following command sets Decode function on. **Command message**: DCST ON

8.2 DCPA

Description Sets the common parameters of serial decode bus.

Commnad Syntax DCPA <param>,<value>[,<param>,<value>[,..]]

<param>	<value>	Description
BUS	{B1,B2}	Decode bus, set B1 as BUS1 and B2 as BUS2.
LIST	{OFF,D1,D2}	Decode list, set OFF to turn off the list, set D1 to select the list of bus1 and set D2 to select the list of bus2.
FOMT	{BIN,DEC,HEX}	Format of the decode data.
LINK	{TR_TO_DC, DC_TO_TR}	Copy setting, set TR_TO_DC to copy from trigger, and set DC_TO_TR to copy to trigger.
LSSC	1 to lines of list	List scroll.
LSNM	1 to 7	list lines

Table 8.1 DCPA Parameters

Example The following command sets the current decode bus to bus1 separately.
Command message: DCPA BUS,B1

The following command sets the current decode bus to bus2, set format of bus2 data to hex, select the list of bus2 and set the list lines to 5.
Command message: DCPA BUS,B2,LIST,D2,FOMT,HEX,LSNM,5

8.3 B<n>:DCIC

Description Sets the parameters of the IIC decode bus.

Commnad Syntax B<n>:DCIC <param>,<value>[,<param>,<value>[,..]]
<n> := {1,2}

<param>	<value>	Description
DIS	{OFF,ON}	Display the current bus.
SCL	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D 7,D8,D9,D10,D11,D12,D13,D14,D15}	Set the SCL source for the IIC bus.
SCLT	value with unit	Set the threshold of the SCL.
SDA	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D 7,D8,D9,D10,D11,D12,D13,D14,D15}	Set the SDA source for the IIC bus.
SDAT	value with unit	Set the threshold of the SDA.
RW	{OFF,ON}	Set whether the read/write bit is included in the address. Set on to include and set off to not include.

Table 8.2 DCIC Parameters

Example The following command sets the threshold of SCL source for IIC bus2 to 200mv separately.
Command message: B2:DCIC SCLT,0.2V

The following command sets IIC bus1 to display, sets the SCL source to D0, sets the SDA source to D1, and includes the R/W bit in the address.
Command message: B1:DCIC DIS,ON,SCL,D0,SDA,D1,RW,ON

8.4 B<n>:DCSP

Description Sets the parameters of the SPI decode bus.

Commnad Syntax B<n>:DCSP <param>,<value>[,<param>,<value>[,..]]
<n> := {1,2}

<param>	<value>	Description
DIS	OFF,ON	Display the current bus.
CLK	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Set the CLK soruce for the SPI bus.
CLKT	value with unit	Set the threshold of the CLK.
EDGE	RISING,FALLING	Set the edge of the clock that data latched on.
	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Set the MISO source for the SPI bus.
MISOT	value with unit	Set the threshold of the MISO.
MOSI	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Sets the MISO source for the SPI bus.
MOSIT	value with unit	Set the threshold of the MISO.
CSTP	CS,NCS,TIMEOUT	Set the chip selection type for the SPI bus.
CS	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Set the CS source for the SPI bus.
CST	value with unit	Set the threshold of the CS.
NCS	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Set the CS source for the SPI bus.
NCST	value with unit	Set the threshold of the CS
TIM	value with unit	Set the timeout value when the CS type is CLK Timeout.
BIT	MSB,LSB	Set the bit order for the SPI bus.
DLEN	4 to 32	Set the data length for the SPI bus.

Table 8.3 SPI Parameters

Example The following command sets the threshold of CLK source for SPI bus2 to 200mV separately.
Command message: B2:DCSP CLKT,0.2V

The following command sets SPI bus1 to display, sets the CLK source to D0, sets the MOSI source to D1, sets the CS type to TIMEOUT and the timeout value to 2us, sets the bit order to MSB, and set the data length to 32.

Command message: B1:DCSP DIS,ON,CLK,D0,MOSI,D1,CSTP,TIMEOUT,TIM,2uS,BIT, MSB,DLEN,32

8.5 B<n>:DCUT

Description Sets the parameters of UART decode bus.

Commnad Syntax B<n>:DCUT <param>,<value>[,<param>,<value>[,..]]
<n> := {1,2}

<param>	<value>	Description
DIS	{OFF,ON}	Display the current bus.
RX	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6 ,D 7,D8,D9,D10,D11,D12,D13,D14,D15}	Set the RX source for the UART bus.
RXT	value with unit	Set the threshold of the RX.
TX	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D 7,D8,D9,D10,D11,D12,D13,D14,D15}	Set the TX source for the UART bus.
TXT	value with unit	Set the threshold of the TX.
BAUD	value without unit, 300 to 50000000	Set the baud rate for the UART bus.
DLEN	5 to 8	Set the data length for the UART bus.
PAR	{NONE,EVEN,ODD}	Set the parity check for the UART bus.
STOP	{1,1.5,2}	Set the length of stop bit for the UART bus.
POL	{LOW,HIGH}	Set the idle level for the UART bus.
BIT	{MSB,LSB}	Sets the bit order for the UART bus.

Table 8.4 UART Parameters

Example The following command sets the threshold of RX source for UART bus2 to 200mV separately.
Command message: B2:DCUT RX,0.2V

The following command sets UART bus1 to display, sets the RX source to D0, sets the baud rate to 9600 bit/s, sets the parity check to ODD, sets the stop bit length to 2, sets the idle level to HIGH and the bit order to MSB.
Command message: B1:DCUT DIS,ON,RX,D0,BAUD,9600,PAR,ODD,STOP,2POL,HIG H,BIT,MSB

8.6 B<n>:DCCN

Description Sets the parameters of CAN decode bus.

Commnad Syntax B<n>:DCCN <param>,<value>[,<param>,<value>[..]]
<n> := {1,2}

<param>	<value>	Description
DIS	{OFF,ON}	Display the current bus.
CANH	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7, D 8,D9,D10,D11,D12,D13,D14,D15}	Set the CANH source for the CAN bus.
CANHT	value with unit	Set the threshold of the CANH.
CANL	{C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7, D 8,D9,D10,D11,D12,D13,D14,D15}	Set the CANL source for the CAN bus.
CANLT	value with unit	Set the threshold of the CANL.
SRC	{CAN_H,CAN_L,SUB_L}	Set the decode source for the CAN bus.
BAUD	5000 to 1000000	Set the baud rate for the CAN bus.

Table 8.5 CAN Parameters

Example The following command sets the threshold of CANH source for CAN bus2 to 200mV separately.

Command message: B2:DCCN CANH,0.2V

The following command sets CAN bus1 to display, sets the CANH source to D0, sets the decode source to CANH and the baud rate to 9600 bit/s.
Command message: B1:DCCN DIS,ON,CANH,D0,SRC,CANH,BAUD,9600

8.7 B<n>:DCLN

Description Sets the parameters of LIN decode bus.

Commnad Syntax B<n>:DCLN <param>,<value>[,<param>,<value>[,..]]
<n> := {1,2}

<param>	<value>	Description
DIS	OFF,ON	Display the current bus.
SRC	C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6, D7, D8,D9,D10,D11,D12,D13,D14,D15	Set the source for the LIN bus.
SRCT	value with unit	Set the threshold of the Source.
BAUD	300 to 2000	Set the baud rate for the LIN bus.

Table 8.6 LIN Parameters

Example The following command sets the threshold of source for LIN bus2 to 200mV separately.
Command message: B2:DCLN SRCT,0.2V

The following command sets LIN bus1 to display, sets the decode source to D0 and the baud rate to 9600 bit/s.
Command message: B1:DCCN DIS,ON,SRC,D0,BAUD,9600

Display Commands

The **DISPLAY** subsystem is used to control how waveforms, and the graticules are displayed on the screen.

DTJN

GRDS

INTS

MENU

PESU

9.1 DOT_JOIN

Description	Sets the interpolation lines between data points.
Commnad Syntax	DOT_JOIN <state> <state> := {ON,OFF} **ON**: Dots. This mode displays data more quickly than vector mode but does not draw lines between sample points. **OFF**: Vectors. This is the default mode and draws lines between points.
Query Syntax	DOT_JOIN?
Response Format	DOT_JOIN <state>
Example	The following command turns off the interpolation lines. **Command message**: DTJN ON

9.2 GRID_DISPLAY

Description	Sets the type of the diplay grid.
Commnad Syntax	GRID_DISPLAY <type> < type > := {FULL,HALF,OFF}
Query Syntax	GRID_DISPLAY?
Response Format	GRID_DISPLAY <type>
Example	The following command changes the type of grid to full grid. **Command message**:GRDS FULL

9.3 INTENSITY

Description	Sets the intensity level of the grid or the trace.
Commnad Syntax	INTENSITY GRID,<value>,TRACE,<value> <value> := 0 (or 30) to 100
Query Syntax	INTENSITY?
Response Format	INTENSITY TRACE,<value>,GRID,<value>
Example	The following command changes the grid intensity level to 75%. **Command message**: INTS GRID,75

9.4 MENU

Description	Enable/disable to diplay the menu.
Commnad Syntax	MENU <state>
	<state>:={ON,OFF}
Query Syntax	MENU?
Response Format	MENU <state>
Example	The following command enables the display of the menu.
	Command message: MENU ON

9.5 PERSIST_SETUP

Description	Sets the persistence duration of the display,in seconds, in persistence mode.
Commnad Syntax	PERSIST_SETUP <time>
Query Syntax	PERSIST_SETUP?
Response Format	PERSIST_SETUP <time>
Example	The following command sets the variable persistence at 5 seconds.
	Command message: PESU 5

History Commands

The **HISTORY** subsystem commands control the waveform recording function and the history waveform play function.

FRAM	FTIM?
HSMD	HSLST

10.1 FRAME_SET

Description	Sets the current frame number in the history function.
Commnad Syntax	FRAM <frame_num> <frame_num> := 0 to the max frame number.

Note:

Sending the query FRAM? will return the max
frame number when the history function is initiated.

Query Syntax	FRAM?
Response Format	FRAM <frame_num>
Example	When the history function is on, the following command sets current frame number to 50. Then you can see the response on the screen as shown below. **Command message**: FRAM 50

10.2 FRAME_TIME?

Description	Returns the acquire timestamp of the current frame.
Query Syntax	FTIM?
Response Format	FTIM hour: minute: second. micro-second
Example	Query returns the acquire time of the current frame. **Query message**: FTIM? **Response message**: FTIM 00: 05: 12. 650814
Related Commands	

10.3 HISTORY_MODE

Description	Sets the state of history mode.
Commnad Syntax	HSMD <state> <state> := {ON,OFF}
Query Syntax	HSMD?
Response Format	HSMD <state>
Example	The following command sets the state of history mode to ON. **Command message**: HSMD ON

10.4 HISTORY_LIST

Description Sets the state of history list.

Commnad Syntax HSLST <state>
<state> := {ON,OFF}

Note:

This command can only be used when History function is turned on.

Query Syntax HSLST?

Response Format HSLST <state>

Example When History function is on, the following command sets the state of history list to ON.
Command message: HSLST ON

Math Commands

The **MATH** subsystem controls the math functions in the oscilloscope. As selected by the **DEFINE** command, these math functions are available:

Operators: Add, Subtract, Multiply, Divide. Operators perform their function on two analog channel sources.

Transforms: DIFF, Integrate, FFT, SQRT.

DEF	INVS	MTVD
MTVP	FFTC	FFTF
FFTP	FFTS	FFTT?
FFTU	FFTW	

11.1 DEFINE

Description Sets the desired math operation.

Commnad Syntax DEFINE EQN,'<equation>'

Note:

<equation> is the mathematical expression, enclosed by single or double quotation marks.

Function Equations	
<source1> + <source2>	Addition
<source1> - <source2>	Subtraction
<source1> * <source2>	Multiplication
<source1> / <source2>	Ratio
FFT<source>	FFT
INTG<source>	Integral
DIFF<source>	Differentiator
SQRT<source>	Square Root

Table 11.1 Function Equations

Query Syntax DEFINE?

Response Format DEFINE EQN,'<equation>'

Example When the Math function is on, and both Channel 1 and Channel 2 are on, the following command sets the math operation to Multiplication, source1 to C1, source2 to C2
Command message: DEFINE EQN,'C1*C2'
When the Math function is on, and Channel 1 is on, the following command sets the math operation to Differentiator, source to C1.
Command message: DEFINE EQN,'DIFFC1

11.2 INVERTSET

Description Inverts the selected math waveform.

Note:

This command is only valid in add, subtract, multiply and divide operation.

Commnad Syntax <trace>:INVERTSET <state>
<trace> := {MATH}
<state> := {ON,OFF}

Query Syntax <trace>:INVERTSET?

Response Format <trace>:INVERTSET <state>

Example When the Math function is on, and the operation is Add, the following command inverts the math waveform.
Command message: MATH:INVS ON

11.3 MATH_VERT_DIV

Description Sets the vertical scale of the selected math operation.

Note:

This command is only valid in add, subtract, multiply and divide operation.

Commnad Syntax MATH_VERT_DIV <scale>
<scale> := {500uV,1mV,2mV,5mV,10mV,
20mV, 50mV,100mV,200mV,500mV,1V,2V,5V,10V,20V,50V,100 V}

Note:

Legal values for the scale depend on the selected operation.
For details, please refer to the math menu of the oscilloscope

---

Query Syntax MATH_VERT_DIV?

Response Format MATH_VERT_DIV <scale>

Example When the Math function is on, and the operator is Add, the following command changes the vertical scale of the math waveform to 1 V.
Command message: MTVD 1V

11.4 MATH_VERT_POS

Description Sets the vertical position of the math waveform with specified source.

Commnad Syntax MATH_VERT_POS <point>
<point> := { -255 to 255}

Note:

The point represents the screen pixels and is related to the screen center.
For example, if the point is 50. The math waveform will be displayed 1
grid above the vertical center of the screen. Namely one grid is 50.

Query Syntax MATH_VERT_POS?

Response Format MATH_VERT_POS <point>

Example When the Math function is on, the following command sets the vertical position of the math waveform to 1 grid above the screen vertical center.
Command message: MTVP 50

Related Commands **FFT_POSITION**

11.5 FFT_CENTER

Description Sets the center frequency when FFT is selected.

Commnad Syntax FFT_CENTER <center>
<center> := frequency value with unit (MHz/ kHz/ Hz).

Note:

Set the center to a value outside of the legal range, the center value is automatically set to the nearest legal value. Legal values are affected by the Hz/div setting.

The range for center is related to the horizontal scale of FFT and varied by models. See the math menu of oscilloscope as shown below for details.

Query Syntax FFT_CENTER?

Response Format FFT_CENTER <center>

Example When the Math function is on, the operator is FFT, and the horizontal scale is 100 MHz, the following command sets the center frequency of FFT to 58 MHz.
Command message: FFTC 58MHz

Related Commands **FFT_TDIV?**

11.6 FFT_FULLSCREEN

Description Sets the display mode of FFT waveform.

Commnad Syntax FFT_FULLSCREEN <state>
<state> := {OFF,ON,EXCLU}

- **OFF**: Split Screen.
- **ON**: Full Screen.
- **EXCLU**: Exclusive.

Query Syntax FFT_FULLSCREEN?

Response Format FFT_FULLSCREEN <state>

Example When the Math function is on, and the operator is FFT, the following command sets the display mode of FFT waveform to Full Screen.
Command message: FFTF ON

11.7 FFT_POSITION

Description Sets the vertical offset of the FFT waveform. The unit is related to the vertical scale type of the currrent FFT and the unit of the channel.

Note:

The command is only valid when the scale type is Vrms.

Commnad Syntax FFT_POSITION <offset>
<offset> := {-24.4*DIV to 15.6*DIV}

Note:

If there is no unit (V/mV/uV) added, it defaults to volts (V).

Setting t the offset to a value outside of the legal range will set the center to the nearest legal value. Legal values are affected by the Scale setting.

Query Syntax FFT_POSITION?

Response Format FFT_POSITION <offset>
<offset> := Numerical value in E-notation with SI unit.

Example When the Math function is on, the operator is FFT, and the scale is 10 mV, the following steps set the offset of FFT waveform to 28 mV.

- Send command to set the scale unit to Vrms.
  Command message: FFTU VRMS

- Send command to set the offset to 28 mV.
  Command message: FFTP 28mV

Related Commands **FFT_SCALE**
FFT_UNIT

11.8 FFT_SCALE

Description Sets the vertical scale of the FFT waveform. The unit is related to the vertical scale type of the current FFT and the unti of the channel.

Commnad Syntax FFT_SCALE <scale>
<scale> := {0.1,0.2,0.5,1,2,5,10,20} when scale type is dBVrms or dBm.
<scale> := {0.001,0.002,0.005,0.01,0.02,0.05,0.1,0. 2,0.5,1, 2,5,10,20} when scale type is Vrms.

Query Syntax FFT_SCALE?

Response Format FFT_SCALE <scale>
<scale> := Numerical value in E-notation with SI unit.

Example When the Math function is on, and the operator is FFT, the following steps set the vertical scale of FFT to 5 dBVrms.

- Send command to set the scale unit to dBVrms.
  Command message: FFTU DBVRMS

- Send command to set the scale to 5.
  Command message: FFTS 5

Related Commands **UNIT**
FFT_UNIT
FFT_POSITION

11.9 FFT_TDIV?

Description Returns the current horizontal scale of the FFT waveform.

Query Syntax FFT_TDIV?

Response Format FFT_TDIV <value>
<value> := Numerical value with measurement unit and physical unit.

Example The following query returns the horizontal scale unit of FFT.
Query message: FFTT?
Response message: FFTT 100.00MHz

11.10 FFT_UNIT

Description Sets the vertical scale type of the FFT waveform.

Commnad Syntax FFT_UNIT <unit>
<unit> := {VRMS,DBM,DBVRMS}

Query Syntax FFT_UNIT?

Response Format FFT_ UNIT <unit>

Example When the Math function is on, and the operator is FFT, the following command sets the vertical scale unit of FFT to dBVrms.
Command message: FFTU DBVRMS

Related Commands **FFT_SCALE**
FFT_POSITION

11.11 FFT_WINDOW

Description Sets the window tranform or the operations for the FFT function.

Commnad Syntax FFT_WINDOW <window>
<window> := {RECT,BLAC,HANN,HAMM,FLATTOP}

- **RECT**: Rectangle is useful for transient signals, and signals where there are an integral number of cycles in the time record.
- **BLAC**: Blackman reduces time resolution compared to the rectangular window, but it improves the capacity to detect smaller impulses due to lower secondary lobes (provides minimal spectral leakage).
- **HANN**: Hanning is useful for frequency resolution and general purpose use. It is good for resolving two frequencies that are close together, or for making frequency measurements.
- **HAMM**: Hamming.
- **FLATTOP**: Flattop is the best for making accurate amplitude measurements of frequency peaks.

Query Syntax FFT_WINDOW?

Response Format FFT_WINDOW <window>

Example When the Math function is on, and the operator is FFT, the following command sets the FFT window to Hamming.
Command message:FFTW HAMM

Measure Commands

The commands in the **MEASURE** subsystem are used to make parametric measurements on displayed waveforms.

To make a measurement, the portion of the waveform required for that measurement must be displayed on the oscilloscope screen.

CYMT?	MEAD	PACU
PAVA?	PASTAT	MEACL
MEGS	MEGA	MEGB

12.1 CYMOMETER?

Description

Measures and returns the frequency counter of the specified source. The counter measurement counts the trigger level crossings at the selected trigger slope and displays the results in MHz/kHz/Hz.

Query Syntax

CYMOMETER?

Response Format

CYMOMETER <freq>
<freq> := Numerical value in E-notation with SI unit, such as 1.00E+03Hz.

Note:

When the signal frequency is less than 10 Hz, it returns “10 Hz” or “<10Hz”.

Example

When the frequency of input signal is l Hz, the following returns the value of cymometer which displaying on the screen of the instrument.
Response message: CYMT 10Hz

When the frequency of input signal is 25.000137 MHz, the following returns the value of cymometer which displaying on the screen of the instrument.
Response message: CYMT 2.50E+07Hz

12.2 MEASURE_DELAY

Description Places the instrument in he continuous measurement mode and starts a type of delay measurement.

Commnad Syntax MEASURE_DELAY <type>,<sourceA-sourceB>
<sourceA-sourceB> := {C1-C2,C1-C3,C1-C4,C2-C3,C2- C4,C3-C4}
<type> := {PHA,FRR,FRF,FFR,FFF,LRR,LRF,LFR,LFF,SK EW}

Type	Description
PHA	The phase difference between two channels.(rising edge - rising edge)
FRR	Delay between two channels.(first rising edge - first rising edge)
FRF	Delay between two channels.(first rising edge - first falling edge)
FFR	Delay between two channels.(first falling edge - first rising edge)
FFF	Delay between two channels.(first falling edge - first falling edge)
LRR	Delay between two channels.(first rising edge - last rising edge)
LRF	Delay between two channels.(first rising edge - last falling edge)
LFR	Delay between two channels.(first falling edge - last rising edge)
LFF	Delay between two channels(first falling edge - last falling edge)
SKEW	Delay between two channels.(edge – edge of the same type)

Table 12.1 Delay Measurements

Query Syntax <sourceA-sourceB>:MEASURE_DELY? <type>

Response Format <sourceA-sourceB>:MEAD <type>,<value>
<sourceA-sourceB> := Numerical value in E-notation with SI unit, such as 1.24E-04S. Except for PHA, it returns as “44.65degree”.

Example The following steps show how to get the measured value of phase between C2 and C4.

- Send the message to set the measurement to Phase between C2 and C4, and then there displays a phase measurement on the screen.
  Command message: MEAD PHA,C2-C4
- Send the message to get the measured value of phase. **Command message**: C2-C4:MEAD? PHA
  Response message: C2-C4:MEAD PHA,-89.46degree

12.3 PARAMETER_CUSTOM

Description Installs a measurement and starts the specified measurement of the specified source.

See the command **PARAMETER_VALUE?** to get the measured value of specified measurement.

See the command **MEASURE_DELAY** to install the measurement of delay class.

Commnad Syntax PARAMETER_CUSTOM <parameter>,<source>
<source> := {C1,C2,C3,C4}
<parameter> :={PKPK,MAX,MIN,AMPL,TOP,BASE,CMEA N,MEAN,STDEV,VSTD,RMS, CRMS,OVSN,FPRE,OVSP, RPRE,LEVELX,DELAY,TIMEL,PER,FREQ,PWID,NWID,R ISE, FALL,WID,DUTY,NDUTY,ALL}

Parameter	Description
PKPK	vertical peak-to-peak
MAX	maximum vertical value
MIN	minimum vertical value
AMPL	vertical amplitude
TOP	waveform top value
BASE	waveform base value
CMEAN	average value in the first cycle
MEAN	average value
STDEV	standard deviation of the data
VSTD	standard deviation of the data in the first cycle
RMS	RMS value
CRMS	RMS value in the first cycle
OVSN	overshoot of a falling edge
FPRE	preshoot of a falling edge
OVSP	overshoot of a rising edge
RPRE	preshoot of a rising edge
LEVELX	Level measured at trigger position
PER	period
FREQ	frequency
PWID	positive pulse width
NWID	negative pulse width
RISE	rise-time
FALL	fall-time
WID	Burst width
DUTY	positive duty cycle
NDUTY	negative duty cycle
DELAY	time from the trigger to the first transition at the 50% crossing
TIMEL	time from the trigger to each rising edge at the 50% crossing
ALL	All measurements snapshot, equal to turn on the switch of all measure

Table 12.2 Measurements

Query Syntax

Response Format

Example The following command sets the type of measure to PKPK of Channel 1.
Command message: PACU PKPK,C1

Related Commands **PARAMETER_VALUE?**
MEASURE_DELAY
MEASURE_CLEAR

12.4 PARAMETER_VALUE?

Description

Measures and returns the specified measurement value present on the selected waveform.

There are three uses for this command:

1. Specify the source and the measurement.
   - See the command "MEAD?" to get the measured value of delay measurement.
2. Use "PAVA? CUST<x>" to get customized.
3. Use "PAVA? STAT<x>" to get statistics.

Query Syntax

Usage 1
<source>:PARAMETER_VALUE? <parameter>
<source> := {C1,C2,C3,C4}
<parameter> := {PKPK,MAX,MIN,AMPL,TOP,BASE,CMEA N,MEAN,STDEV,VSTD,RMS ,CRMS,OVSN,FPRE,OVSP, RPRE,LEVELX,PER,FREQ,PWID,NWID,RISE,FALL,WID, DUTY, NDUTY,DELAY,TIMEL,ALL} See the table Description of Parameter for details.

Response Format

<source>:PARAMETER_VALUE <parameter>,<value>
<value>:= Numerical value in E-notation with SI unit.

Query Syntax

Usage 2
PARAMETER_VALUE? CUST<x>
<x> := {1 to 5, and ALL}

Custom Parameters	Description
CUST1	The first measure parameter specified by "PACU"
CUST2	The second measure parameter specified by "PACU"
CUST3	The third measure parameter specified by "PACU"
CUST4	The fourth measure parameter specified by "PACU"
CUST5	The fifth measure parameter specified by "PACU"
CUSTALL	All measure parameters specified by "PACU"

Table 12.3 Custom Parameters

Note:

Install the measurement as CUST<x> by using command "PACU", before using usage 2.

When the number of installed measurements is less than 5 and you send the command "PAVA? CUSTALL", it will return OFF as value for remaining custom parameters.

Response Format

PARAMETER_VALUE
CUST<x>:<source>,<parameter>,<value>
<value> := Numerical value in E-notation with SI unit.

Query Syntax

Usage 3
PARAMETER_VALUE? STAT<x>
<x> := {1 to 5}

Custom Parameters	Description
STAT1	Statistics of the first measure parameter specified by "PACU"
STAT2	Statistics of the second measure parameter specified by "PACU"
STAT3	Statistics of the third measure parameter specified by "PACU"
STAT4	Statistics of the fourth measure parameter specified by "PACU"
STAT5	Statistics of the fifth measure parameter specified by "PACU"

Table 12.4 Custom Parameters

Note:

Installing the statistics of the measurement as STAT<x> by using command "PACU" and turn on the statistics by using the command "PASTAT" before using usage 3.

Response Format PARAMETER_VALUE STAT<x> <source> <parameter>:cur,<value1>,mean,<value2>, min,<value3>, max,<value4>,std-dev,<value5>,count,<value6>

Parameter	Description
cur	Current value of measurement
mean	Mean value of measurement
min	Minimum value of measurement
max	Maximum value of measurement
std-dev	Standard deviation of measurement
count	Measurement count

Table 12.5 Parameters

<value>:= Numerical value in E-notation with SI unit.

Example The following query returns the rise time of Channel 2.
Query message: C2:PAVA? RISE
Response message: C2:PAVA RISE,3.6E-9S

The following query returns all measurement of Channel 1.
Query message: C1:PAVA? ALL

The following steps show how the user customize the measurement parameters and get the measured value.

- Send the command to set the measurement parameter.
  Command message: PACU PKPK,C1
- Send the query to get the measured value.
  Query message:PAVA? CUST1
  Response message: PAVA CUST1:C1,PKPK,4.08E+00V
- You can also send the query to get the measured value.
  Command message: PAVA? CUSTALL
  Response message: PAVA
  CUST1:C1,PKPK,4.08E+00V;CUST2:OFF;CUST3:OFF;C UST4:OFF;CUST5:OFF

Related Commands **PARAMETER_CUSTOM**
MEASURE_DELAY
PASTAT

12.5 PASTAT

Description Sonctrols the operation and display of the measurement statistics.

Commnad Syntax PASTAT <state>
<state> :={OFF, ON, RESET}

- **OFF**: turn off the measurement statistics.
- **ON**: turn on the measurement statistics.
- **RESET**: reset the measurement statistics.

Example When the measurement item is turned on, the following command turn on the measurement statistics.
Command message: PASTAT ON
To clear all of the statistics accumulated for all periodic measurements, the following command shows.

Command message: PASTAT RESET

12.6 MEASURE_CLEAR

Description Removes all user specified measurements.

Commnad Syntax MEACL

Example When measurement items are turned on, the following command remove all displayed measurement items.
Command message: MEACL

Related Commands **PARAMETER_CUSTOM**

12.7 MEASURE_GATE_SWITCH

Description Controls the switch of the gate measurement. When the gate is turned on, only the waveform within the threshold will be measured.

Commnad Syntax MEGS <state>
<state> :={OFF, ON}

- **OFF**: turn off the gate measurement.
- **ON**: turn on the gate measurement.

Example When the measure is turned on, the following command turn on the gate measurement.
Command message: MEGS ON

12.8 MEASURE_GATEA

Description Specifies the position of the measurement gate A.

Commnad Syntax MEGA <value>
<value> := value with unit.

Note:

The time unit (s/ms/us/ns) must be added to the position. If there is no unit added, it defaults to be S.

The range of the value is related to the timebase and horizontal position.

The value of gateA must not be greater than the value of gateB, otherwise it will be automatically set to the value of gateB.

Example When the gate switch is on, the following command set the position of gateA to 20us.
Command message: MEGA 20us

Related Commands **MEASURE_GATE_SWITCH**
MEASURE_GATEB

12.9 MEASURE_GATEB

Description Specifies the position of the measurement gate B.

Commnad Syntax MEGB <value>
<value>:= value with unit.

Note:

The time unit (s/ms/us/ns) must be added to the position. If there is no unit added, it defaults to be S.

The range of the value is related to the timebase and horizontal position.

The value of gate B must not be greater than the value of gate A, otherwise it will be automatically set to the value of gate A.

Example When the gate switch is on, the following command set the position of gateB to 1.68 ms.
Command message: MEGB 1.68ms

Related Commands **MEASURE_GATE_SWITCH**
MEASURE_GATEA

Pass/Fail Commands

The **PASS/FAIL** subsystem commands and queries control the mask test features.

PACL	PFBF	PFCM
PFDD?	PFDS	PFEN
PFFS	PFOP	PFSC
	PFST	

13.1 PARAMETER_CLR

Description	Resets the P/F test statistics.
Commnad Syntax	PARAMETER_CLR
Example	PACL
Related Commands	**PF_DATADIS**

13.2 PF_BUFFER

Description Sets the output mode when the test fails. This is the same as pressing the "Output" button on the menu of PASS/FAIL on the front panel.

Commnad Syntax PF_BUFFER <state>
<state> := {ON,OFF}

- **ON**: The statistical result is displayed when the failed waveform is detected, and the buzzer alarm. (not related to the state of the sound switch)
- **OFF**: The statistical result is displayed when the failed waveform is detected, but the buzzer does not alarm.

Query Syntax PF_BUFFER?

Response Format PF_BUFFER <state>

Example When the PASS/FAIL function is on, the following command sets “output” to “ON”.
Command message: PFBF ON

13.3 PF_DATADIS

Description Returns the nmber of the failed frames, passed frames and total frames.

Query Syntax PF_ DATADIS?

Response Format PF_DATADIS FAIL,<num>,PASS,<num>,TOTAL,<num>

Example The following query returns the number of the message display of the pass/fail.
Query message: PFDD?
Response message: PFDD FAIL,0,PASS,0,TOTAL,0

13.4 PF_CREATEM

Description

Creates a Pass/Fail test rule around the current selected channel, using the horizontal adjustment parameters and the vertical adjustment paraters. The parameters are defined by the **PF_SET** command.

Note:

This command is valid only if the Pass/Fail test function has been opened (**PF_OPERATION**) and is not in operation (**PF_ENABLE**).

Commnad Syntax

PF_ CREATEM

Example

The following steps create the mask of the Pass/Fail.

1. Send command to set the Pass/Fail test enable.
   Command message: PFEN ON
2. Send command to stop the operation.
   Command message: PFOP OFF
3. Send command to create the rule.
   Command message: PFCM

Related Commands

PF_OPERATION
PF_ENABLE
PF_SOURCE
PF_SET

13.5 PF_DISPLAY

Description

Displays information in Pass/Fail test features.

Commnad Syntax

PF_DISPLAY <state>
<state> := {ON,OFF}

Query Syntax

PF_DISPLAY?

Response Format

PF_DISPLAY <state>

Example

The following steps display the message of Pass/Fail.

- Send command to set the Pass/Fail test enable.
  Command message: PFEN ON
- Send command to display the message of Pass/Fail.
  Command message: PFDS ON

Related Commands

PF_ENABLE

13.6 PF_ENABLE

Description Enables/disables the Pass/Fail test feature.

Commnad Syntax PF_ENABLE <state>
<state> := {ON,OFF}

- **ON**: Enable the mask test features.
- **OFF**: Disable the mask test features.

Query Syntax PF_ENABLE?

Response Format PF_ENABLE <state>

Example The following command enables mask test features.
Command message: PFEN ON

13.7 PF_FAIL_STOP

Description Sets the switch of the "stop on fail" function.

Commnad Syntax PF_FAIL_STOP <state>
<state> := {ON,OFF}

- **ON**: To monitor the failure waveform, the oscilloscope stops testing and enters the "STOP" state. At this point, the screen displays the last statistical result.(if the display is already open)
- **OFF**: To monitor the failure waveform, the oscilloscope will continue to test and update the statistics on the screen immediately.

Query Syntax PF_FAIL_STOP?

Response Format PF_FAIL_STOP <state>

Example The following command sets “stop on fail” to “off”.
Command message: PFFS OFF

13.8 PF_OPERATION

Description Controls the run or stop state of the Pass/Fail funtion.

Commnad Syntax PF_OPERATION <state>
<state> := {ON,OFF}

Query Syntax PF_OPERATION?

Response Format PF_OPERATION <state>

Example The following command controls to run Pass/Fail test.
Command message: PFOP ON

Related Commands **PF_ENABLE**

13.9 PF_SOURCE

Description	Sets the measurement sources for the Pass/Fail test.
Commnad Syntax	PF_SOURCE <trace> <trace> := {C1,C2,C3,C4}
Query Syntax	PF_SOURCE?
Response Format	PF_SOURCE <trace>
Example	The following command sets the measurement source to Channel 1 when Channel 1 is on. **Command message**: PFSC C1

13.10 PF_SET

Description	Sets the tolerance in the X/Y direction around the selected waveform defined by **PF_SOURCE** for the Pass/Fail feature. The value of the tolerance will be added and subtracted to horizontal/Vertical values of the waveform to determine the boundaries of the mask.
Commnad Syntax	PF_ SET XMASK,<div>,YMASK,<div> <div> := {0.04 to 4.0} **Note:** The step value is 0.04.
Query Syntax	PF_ SET?
Response Format	PF_ SET XMASK,<div>,YMASK,<div>
Example	The following command sets the X mask to 0.4 and the Y mask to 0.52. **Command message**: PFST XMASK,0.4,YMASK,0.52
Related Commands	**PF_SOURCE**

Print Commands

SCDP

14.1 SCREEN_DUMP

Description Captures the screen and returns the data of the bmp file.

Query Syntax SCDP

Response Format <bmp header><bmp screen data>

Example The following step shows how to transfers the screen information as a file named screen.bmp in a Python shell.

1. Send the query to get the bmp data. Query message: SCDP
2. Create a new bmp file named “screen.bmp”.
3. Write the data to the file.
4. Close the file.

```
import visa
def main():
_rm = visa.ResourceManager()
inst = _rm.open_resource("USB0::0xF4EC::0xEE38::0123456789::INSTR")
inst.chunk_size = 20*1024*1024 #default value is 20*1024(20k bytes)
inst.timeout = 30000 #default value is 2000(2s)
file_name = "F:
SCDP.bmp"
inst.write("SCDP")
result_str = sds.read_raw()
f = open(file_name,'wb')
f.write(result_str)
f.flush()
f.close()
if __name__=='__main__':
main()
```

Recall Commands

Recall previously saved oscilloscope setups and reference waveforms.

*RCL   RCPN

15.1 *RCL

Description

Recalls the complete front-panel setup of the instrument from internal memory, using one of the twenty non-volatile panel setups.

Commnad Syntax

*RCL <num>
<num> := {0 to 20}

Note:

When num is 0, the default setup will be recalled.

Example

To save the current setup in to slot 3.
Command message: *SAV 3

To recall the setup, send the following command. **Command message**: *RCL 3

Related Commands

RECALL_PANEL
***SAV**

15.2 RECALL_PANEL

Description

Recalls a front-panel setup from the specified-DOS path directory in an external memory device.

Note:

The filename string is up to eight characters, with the extension ".xml".

Commnad Syntax

RECALL_PANEL DISK,<device>,FILE,'<filename>'
<device> := {UDSK}
<filename> := A waveform file under a legal DOS path.

Example

The following command recalls the frontpanel setup from a file called "TEST.xml" in root directory of the USB memory device.
Command message: RCPN DISK,UDSK,FILE,'TEST.xm

Related Commands

STORE_PANEL
***RCL**

Reference Commands

The **REFERENCE** system controls the reference waveforms.

REFCL

REFDS

REFLA

REFPO

REFSA

REFSC

REFSR

16.1 REF_CLOSE

Description	Closes the reference function.
Commnad Syntax	REF_CLOSE
Example	The following command closes the Reference function. **Command message**: REFCL

16.2 REF_DISPLAY

Description Enables or disables the current reference channel shown on the screen.

Note:

Only used when the current reference channel has been stored, and the Reference function is enable.

Commnad Syntax	REF_ DISPLAY <state> <state> := {ON,OFF}
Query Syntax	REF_ DISPLAY?
Response Format	REF_ DISPLAY <state>
Example	The following command displays the waveform of the current reference channel. **Command message**: REFDS ON
Related Commands	**REF_CLOSE**

16.3 REF_LOCATION

Description	Selects the current reference channel.
Commnad Syntax	REF_LOCATION <location> <location> := {REFA,REFB,REFC,REFD}
Query Syntax	REF_LOCATION?
Response Format	REF_LOCATION <location>
Example	The following command selects REFA as the current reference channel. **Command message**: REFLA REFA

16.4 REF_POSITION

Description

Sets the vertical offset of the current reference channel. This command is only used when the current reference channel has been saved, and the dispaly state is on.

Commnad Syntax

REF_ POSITION <offset>
<offset> := vertical offset value with unit.

Note:

The range of legal offset varies with the value set by the **REF_SCALE** command. If you set the offset to a value outside of the legal range, the offset value is automatically set to the nearest legal value.

Query Syntax

REF_ POSITION?

Response Format

REF_ POSITION <offset>
<offset> := Numerical value in E-notation with SI unit.

Example

When the Reference function is on, REFB has been saved and the scale is 2 V, the following command sets the current reference channel vertical offset to 0.2 V.
Command message: REFPO 0.2V

Related Commands

REF_SCALE

16.5 REF_SAVE

Description

Saves the waveform (screen range) of the specified source as the reference waveform of the current reference channel to the memory and displays it on the screen.

Commnad Syntax

REF_SAVE

Example

When the Reference function is on, the REF source is Channel 2, and the REF location is REFA, the following command saves Channel 2 as REFA and displays REFA on screen.
Command message: REFSA

Related Commands

16.6 REF_SCALE

Description

Sets the vertical scale of the current reference channel. This commands is only used when the current reference channel has been strored and the display state is on.

Commnad Syntax

REF_ SCALE <scale>
<scale> := {500uV to 10V}

Query Syntax

REF_ SCALE?

Response Format

REF_ SCALE <scale>
<scale> := Numerical value in E-notation with SI unit.

Example

When the Reference function is on, and REFA has been saved, the following command sets the vertical scale of REFA to 100 mV.
Command message: REFSC 100mV

16.7 REF_SOURCE

Description	Sets the refernce waveform source.
Commnad Syntax	REF_SOURCE <source> <source> := {C1,C2,C3,C4,MATH}
Query Syntax	REF_SOURCE?
Response Format	REF_SOURCE <source>
Example	When Channel 1 is on, the following command selects Channel 1 as the source of current reference channel. **Command message**: REFSR C1

Save Commands

Save oscilloscope setups and waveform data.

*SAV

PNSU

STPN

17.1 *SAV

Description

Stores the complete front-panel setup of the instrument in internal memory.

This instruction does not support storing to external temporarily. See the command **STORE_PANEL** for external storage.

Note:

If there is already a file in the specified location, it will overwrite the original file.

Commnad Syntax

*SAV <setup_num>
<setup_num> := {1 to 20}

Example

To save the current setup to the internal slot No.3.
Command message: *SAV 3

To recall this setup, send the following command. **Command message**: *RCL 3

Related Commands

STORE_PANEL
***RCL**

17.2 STORE_PANEL

Description

Stores the complete frontpanel setup of the instrument into a file on the specifiedDOS path directory in a USB memory device.

Commnad Syntax

STORE_PANEL DISK,<device>,FILE,‘<filename>’
<device> := {UDSK}
<filename> := A waveform file under a legal DOS path.

Note:

The filename string is up to eight characters, with the extension “.xml”.

Example

The following command saves the current setup to root directory of the USB memory device in a file called “TEST.xml”.
Command message: STPN DISK,UDSK,FILE,’TEST.xml

Related Commands

***SAV**
RECALL_PANEL

17.3 PANEL_SETUP

Description

Sets the panel setup based on the acquired data from the query **PNSU?**.

The returned data is in XML form.

Note:

The query will take time and return a long data, so it is necessary to set the timeout value and the buffer size before sending the query :

- Set the I/O buffer size The data length is related to the current panel setup. It is recommended to set the read buffer size to 500k bytes or more.

- Set the timeout value The timeout value is related to the network speed or USB transmission speed. The initial value is generally 2s. It is recommended to set the value to 10s or more.

Commnad Syntax

PANEL_SETUP <header><setup data>
<header> := Characters in the format of "#9<9-Digits>" is used to describe the length of <setup data>, which is returned by the query PNSU?
<setup data> := A setup previously returned by the query PNSU?

Query Syntax

PANEL_SETUP?

Response Format

PANEL_SETUP <header><setup data>

Example

The following steps show how to use the query and command to set the panel setup.

1. Send the command to set the response format.
   Command message: CHDR OFF

2. Send the query to get the binary data of setup.
   Command message: PNSU?
   Response message: <header><setup data>

3. Change the panel setup, and then send the command to restore setup get from step2.
   Command message: PNSU <header><setup data>

4. The data in step 2 can be saved to a file to make it easier to recall later. Store the file of format according to the returned data.

Status Commands

IEEE 488.2 defines data structures, commands, and common bit definitions for status reporting. There are also instrument-defined structures and bits.

INR?

18.1 INR?

Description Reads and clears the content of the INternal state change Register (INR). The INR register records the completion of various internal operations and state transitions.

Query Syntax INR?

Response Format INR <value>
<value> := {0 to 65535}

Example The following steps show the change of INR.

1. When the trigger mode is single, and there is no signal input, send the query.
   Response message: INR 0
2. Now, input a signal to trigger. The acquisition mode is Stop. Then, send the query.
   Response message: INR 1
3. Now, change the trigger mode to Auto. Then, send the query.
   Response message: INR 8193
4. Now, change the trigger mode to Single. The acquisition mode changes to be Stop. And then, send the query.
   Response message: INR 8193
5. After sending the query in step 4, send the query again.
   Response message: INR 0
6. After step 2, not to input the signal, change the trigger mode to single. And then, send the query.

   Response message: INR 8192

Bit	Bit value	Description
15	—	Not used (always 0)
14	—	Not used (always 0)
13	8192	Trigger is ready
12	4096	Pass/Fail test detected desired outcome
11	2048	Waveform processing has terminated in Trace D
10	1024	Waveform processing has terminated in Trace C
9	512	Waveform processing has terminated in Trace B
8	256	Waveform processing has terminated in Trace A
7	128	A memory card, floppy or hard disk exchange has been detected
6	64	Memory card, floppy or hard disk has become full in “AutoStore Fill” mode
5	—	Not use(always 0)
4	16	A segment of a sequence waveform has been acquired
3	8	A time-out has occurred in a data block transfer
2	4	A return to the local state is detected
1	2	A screen dump has terminated
0	1	A new signal has been acquired

Table 18.1 State Register

System Commands

The **SYSTEM** subsystem commands control basic system functions of the oscilloscope.

*CAL?

BUZZ

CONET

SCSV

EMOD

19.1 *CAL?

Description Starts the user calibration procedure and returns 0 when the calibration is sucessful.

The user calibration can quickly make the oscilloscope achieve the best working state, in order to obtain the most accurate measurement value.

All function keys are disabled during the self calibration process.

Before starting the user calibration procedure, you must disconnect anything from inputs.

Query Syntax *CAL?

Response Format *CAL 0
0: Calibration successful.

Example The following query starts a self-calibration.
Query message: *CAL?
Response message: *CAL 0

19.2 BUZZER

Description Enables or disables the buzzer.

Commnad Syntax BUZZER <state>
<state> := {ON,OFF}

Query Syntax BUZZER?

Response Format BUZZER? <state>

Example The following command enables the oscilloscope buzzer.
Command message: BUZZ ON

19.3 COMM_NET

Description Sets the IP address of the oscilloscope's internal network interface.

When using this command, DHCP should be off. The COMM_NET? query returns the IP address of the oscilloscope's internal network interface.

Commnad Syntax COMM_NET <ip_add0>,<ip_add1>,<ip_add2>,<ip_add3>
< ip_add0 > := 1 to 223(except 127).
< ip_add1 > := 0 to 255.
< ip_add2 > := 0 to 255.
< ip_add3 > := 0 to 255.

Query Syntax COMM_NET?

Response Format COMM_NET <ip_add0>,<ip_add1>,<ip_add2>,<ip_add3>

Example The following command sets the IP address to 10.0.0.230.
Command message: CONET 10,0,0,230

19.4 SCREEN_SAVE

Description

Controls the automatic screensaver, which automatically shuts down the internal color monitor after a preset time.

Note:

When the screensaver is enabled, the oscilloscope is still fully functional.

Commnad Syntax

SCREEN_SAVE <time>
<time> := {OFF,1MIN,5MIN,10MIN,30MIN,60MIN}

- **OFF**: Do not use screensaver.
- **Others**: When the oscilloscope enters the idle state and holds for the specified time, screensaver will be enabled.

Query Syntax

SCREEN_SAVE?

Response Format

SCREEN_SAVE?<time>

Example

The following comand sets the automatic screensaver to 10 mins.
Command message:SCSV 10MIN

19.5 EduMode

Description

Places the instrument in education mode. In education mode autosetup, measure and cursors of the oscilloscope are locked.

Commnad Syntax

EduMode <func>,<lock>
<func> := {AutoSetup, Measure, Cursors}
<lock> := {ON, OFF}

- **ON**: Enable the function
- **OFF**: Disable the function.

Query Syntax

EduMode? <func>
<func> := {AutoSetup, Measure, Cursors}

Note:

The query without parameters will return the lock status of all functions.

Response Format

Query with the parameter "AutoSetup":
EduMode AutoSetup,ON;

Query with no parameters:
EduMode AutoSetup,<lock>;Measure,<lock>;Cursors,<lock>;

Example

The following command disables the Autosetup function.
Command message: EMOD AutoSetup,OFF

Query message: EMOD?

Response message: AutoSetup,OFF;Measure,ON;Cursors,ON;

Timebase Commands

The TIMEBASE subsystem commands control the horizontal (X-axis) functions. The time per division, delay, and reference can be controlled for the main and window(zoomed) time bases.

TDIV

TRDL

HMAG

HPOS

20.1 TIME_DIV

Description Sets the horizontal scale per division for the main window.

Commnad Syntax TIME_DIV <value>
<value> := {2NS,5NS,10NS,20NS,50NS,100NS,200N S,500NS,1US,2US,5US,10US,20US, 50US,100US,200US, 500US,1MS,2MS,5MS,10MS,20MS,50MS,100MS,200MS, 500MS,1S ,2S,5S,10S,20S,50S,100S}

- **NS**: for nanoseconds.
- **US**: for microseconds.
- **MS**: for milliseconds.
- **S**: for seconds.

Query Syntax TIME_DIV?

Response Format TIME_DIV <value>
<value> := Numerical value in E-notation with SI unit.

Example The command sets the horizontal scale to 500 s.
Command message: TDIV 500US

Related Commands **TRIG_DELAY**
HOR_MAGNIFY
HOR_POSITION

20.2 TRIG_DELAY

Description Sets the time interval between the trigger event and the horizontal center point on the screen. The maximum position value depends on the time/division settings.

- **Pre-trigger acquisition**: Data acquired before the trigger occurs. Negative trigger delays must be given in seconds.
- **Post-trigger acquisition**: Data acquired after the trigger has occurred.

Commnad Syntax TRIG_DELAY <delay>
<delay> := **time value with unit**

Note:

The range of delay is related to the time base.

If the delay is outside of the legal range, the delay value is automatically set to the nearest legal value.

Query Syntax TRIG_DELAY?

Response Format RIG_DELAY <value> Numerical value in E-notation with SI unit, such as 1.00E-04 s.

Example When the time base is 1us/div, the following command sets the trigger delay to -4.8 us (pre trigger).
Command message: TRDL -4.8US

Related Commands **TIME_DIV**

20.3 HOR_MAGNIFY

Description Sets the zoomed (delayed) window horizontal scale (s/div). The main sweep scale determines the range for this command. The maximum value is the **TIME_DIV** value.

Commnad Syntax HOR_MAGNIFY <value>
<value > := {2NS,2NS,5NS,10NS,20NS,50NS,100NS,200N S,500NS,1US,2US,5US,10US, 20US,50US,100US,200US, 500US,1MS,2MS,5MS,10MS,20MS}

Note:

The range of value is related to the current time base. It is from 2NS to the current time base.

Query Syntax HOR_MAGNIFY?

Response Format HOR_MAGNIFY? <value>

Example the following command sets the zoomed (delayed) window horizontal scale to 1 us.
Command message: HMAG 1US

Related Commands **TIME_DIV**

20.4 HOR_POSITION

Description Sets the horizontal position in the zoomed (delayed) view of the main sweep. The main sweep range and the main sweep horizontal position determine the range for this command. The must

keep the zoomed view window within the main sweep range.

Commnad Syntax HOR_POSITION <position>
<position> := time value with unit.

Note:

The range of position is related to the main sweep range and the main sweep horizontal position. The range after magnifying which beyond the screen could display, and it will be adjusted to the proper value.

Query Syntax HOR_POSITION?

Response Format HOR_POSITION? <position>

Example The following command sets the zoom position to 100 ns.
Command message: HPOS 100ns

Related Commands **HOR_MAGNIFY**
TIME_DIV
TRIG_DELAY

Trigger Commands

The **TRIGGER** subsystem controls the trigger modes and parameters for each trigger type.

SET50	TRCP	TRLV
TRLV2	TRMD	TRPA
TRSE	TRSL	TRWI

21.1 SET50

Description Autotmatically sets the trigger levels to center of the trigger source waveform.

Note:

When High and Low (dual) trigger levels are used (as Runt triggers, for example), this command has no effect.

Commnad Syntax SET50

Example When the trigger type is edge and the trigger source is Channel 1, the following command sets the trigger level to the center of Channel 1.
Command message: SET50

Related Commands **TRIG_LEVEL**

21.2 TRIG_COUPLING

Description Sets the input coupling for the selected trigger sources.

Commnad Syntax <trig_source>:TRIG_COUPLING <trig_coupling>
<trig_source> := {C1,C2,C3,C4,EX,EX5}
<trig_coupling> :={AC,DC,HFREJ,LFREJ}

- **AC**: AC coupling block DC component in the trigger path, removing dc offset voltage from the trigger waveform. Use AC coupling to get a stable edge trigger when your waveform has a large dc offset.
- **DC**: DC coupling allows dc and ac signals into the trigger path.
- **HFREJ**: HFREJ coupling places a low-pass filter in the trigger path.
- **LFREJ**: LFREJ coupling places a high-pass filter in the trigger path.

Query Syntax <trig_source>:TRIG_COUPLING?

Response Format <trig_source>:TRIG_COUPLING <trig_coupling>

Example The following command sets the coupling mode of the trigger source Channel 2 to AC.
Command message: C2:TRCP AC

Related Commands **TRIG_SELECT**

21.3 TRIG_LEVEL

Description

Sets the trigger level voltage for the active trigger source.

When there are two trigger levels to set, this command is used to set the higher trigger level voltage for the specified source. **TRIG_LEVEL2** is used to set the lower trigger level voltage.

Commnad Syntax

<trig_source>:TRIG_LEVEL <trig_level>
<trig_source> := {C1,C2,C3,C4,EX,EX5}
<trig_level> := -4.5*DIV to 4.5*DIV for internal triggers.
<trig_level> := -3*DIV to 3*DIV for external triggers.

Note:

An out-of-range value will be adjusted to the closest legal value.

Query Syntax

<trig_source>:TRIG_LEVEL?

Response Format

<trig_source>:TRIG_LEVEL <trig_level>
<trig_level> := Numerical value in E-notation with SI unit.

Example

When the vertical scale of Channel 3 is 200 mV, and the trigger source is Channel 3, the following command sets the trigger level of Channel 3 to 52.00 mV.
Command message: C3:TRLV 52mV

Related Commands

TRIG_SELECT
TRIG_LEVEL2

21.4 TRIG_LEVEL2

Description

Sets the lower rigger level voltage for the specified source.

Higher and lower trigger levels are used with runt /slope triggers.

Commnad Syntax

<trig_source>:TRIG_LEVEL2 <trig_level>
<trig_source> := {C1,C2,C3,C4,EX,EX5}
<trig_level> := -4.5*DIV to 4.5*DIV for internal triggers.

Note:

An out-of-range value will be adjusted to the closest legal value.

Query Syntax

<trig_source>:TRIG_LEVEL2?

Response Format

<trig_source>:TRIG_LEVEL2 <trig_level>
<trig_level> := Numerical value in E-notation with SI unit.

Example

When the trigger type is slope, the following steps set the high trigger level of Channel 2 to 3.5 V, and the low trigger level of Channel 2 to 800 mV.

- Send the command to set high trigger level.
  Command message: C2:TRLV 3.5V

- Send the command to set low trigger level.
  Command message: C2:TRLV2 800mV

Related Commands TRIG_SELECT
TRIG_LEVEL

21.5 TRIG_MODE

Description Sets the trigger sweep mode.

Commnad Syntax TRIG_MODE <mode>
<mode> := {AUTO,NORM,SINGLE,STOP}

- **AUTO**: When AUTO sweep mode is selected, the oscilloscope begins to search for the trigger signal that meets the conditions. If the trigger signal is satisfied, the running state on the top left corner of the user interface shows Trig'd, and the interface shows stable waveform. Otherwise, the running state always shows Auto, and the interface shows unstable waveform.
- **NORM**: When NORMAL sweep mode is selected, the oscilloscope enters the wait trigger state and begins to search for trigger signals that meet the conditions. If the trigger signal is satisfied, the running state shows Trig'd, and the interface shows stable waveform. Otherwise, the running state shows Ready, and the interface displays the last triggered waveform (previous trigger) or does not display the waveform (no previous trigger).
- **SINGLE**: When SINGLE sweep mode is selected, the backlight of SINGLE key lights up, the oscilloscope enters the waiting trigger state and begins to search for the trigger signal that meets the conditions. If the trigger signal is satisfied, the running state shows Trig'd, and the interface shows stable waveform. Then, the oscilloscope stops scanning, the RUN/STOP key is red light, and the running status shows Stop. Otherwise, the running state shows Ready, and the interface does not display the waveform.
- **STOP**: STOP is a part of the option of this command, but not a trigger mode of the oscilloscope.

Query Syntax TRIG_MODE?

Response Format TRIG_MODE <mode>

Example The following command sets the trigger mode to Normal.
Command message: TRMD NORM

Related Commands STOP
ARM_ACQUISITION

21.6 TRIG_PATTERN

Description Specifies the channel values to be used in the pattern trigger and sets the condition of the pattern trigger.

Commnad Syntax TRIG_PATTERN<source>,<status>
[,<source>,<status>[,<source>,<status> [,<source>,<status>]]],STATE,<condition>
< source > := {C1,C2,C3,C4}
<status> := {X,L,H}

< condition > := {AND,OR,NAND,OR}

- **X**: Ignore this channel. When all channels are set to X, the oscilloscope will not trigger.
- **L**: Low level.(lower than the threshold level of the channel)
- **H**: High level.(higher than the threshold level of the channel)

Note:

The status of source can only be set when the source is on.

Query Syntax TRIG_PATTERN?

Response Format TRIG_PATTERN <source>,<status>,<source>,<status>,<source>,<status>, <source>,<status>,STATE,<condition>

Example When the trigger type is Pattern, and Channel 2 & Channel 3 are on, the following command sets the Channel 2 and Channel 3 to low and the condition to AND.
Command message: TRPA C2,L,C3,L,STATE,AND

21.7 TRIG_SELECT

Description Selects the condition that will trigger the acquisition of the waveform.

Depending on the trigger type, additional parameters must be specified. These additional parameters are grouped in pairs. The first in the pair names the variable to be modified, while the second gives the new value to be assigned. Pairs may be given in any order and restricted to those variables to be changed.

Commnad Syntax TRIG_SELECT <trig_type>,SR,<source>,HT,<hold_type>,HV,<hold_value 1>[,HV2,<hold_value2>]
<trig_type> := {EDGE,SLEW,GLIT,INTV,RUNT,DROP}
<source> := {C1,C2,C3,C4,LINE,EX,EX5}

Note:

LINE/EX/EX5 can only be selected when the trigger type is Edge.

<hold_type> := {TI,OFF} for EDGE trigger.
<hold_type> := {TI} for DROP trigger.
<hold_type> := {PS,PL,P2,P1}for GLIT/RUNT trigger.
<hold_type> := {IS,IL,I2,I1} for SLEW/INTV trigger.
<hold_value1> := a time value with unit.
<hold_value2> := a time value with unit.

Note:

The range of hold_values varies from trigger types.
[80nS, 1.5S] for Edge trigger, and [2nS, 4.2S] for others.

TV Command Syntax
TRIG_SELECT <trig_type>,SR,<source>,STAN,<standard>,SYNC,<sync_ type>[,LINE,<line>

[,FLD,<field>]]

Parameter	Description
STAN	Standard
FLD	field
CUST	Custom

Table 21.1 Standard

<trig_type> := {TV}
<source> :={C1,C2,C3,C4}
<standard> :={NTSC,PAL,720P/50,720P/60,1080P/50,108 0P/60,1080I/50,1080I/60, CUST}
<line> := allow triggering on a specific line of video. The line number limits vary with the standard and mode, as shown in the following table.

Standard	Mode		
	Line	Field1	Field2
NTSC		1 263	1 to 262
PAL		1 to 313	1 to 312
720P/50	1 to 750		
720P/60	1 to 750		
1080P/50	1- 1125		
1080P/60	1- 1125		
1080I/50		1 to 563	1 to 562
1080I/60		1 to 563	1 to 562
CUST	1 to number of lines		

Table 21.2 TV Trigger Line Number Limits

<field> := [1,2] for NTSC/PAL/1080I/50/1080I/60
<field> :=1 to <field_count>for CUST.
<field_count> :=1 to 8 depending on the interlace.

Note:

Field can only be selected when the standard is NTSC/PAL/1080I/50/1080I/60/CUST.

Query Syntax TRIG_SELECT?

Response Format TRIG_SELECT <trig_type>,SR,<source>,HT,<hold_type>,HV,<hold_value 1>[,HV2,<hold_value2>]

TV RESPONSE FORMAT
<trig_type>,SR,<source>,STAN,<standard>,SYNC,<sync_ type>[,LINE,<line>[,FLD,<field>]]

Example To set trigger type to Edge, trigger source to Channel 1, hold type to TIME, and the time value to 1.43uS, the following comes true.
Command message: TRSE EDGE,SR,C1,HT,TI,HV,1.43uS

TV Example
To set trigger type to Video, trigger source to Channel 1, standard to NTSC, and SYNC to ANY, the following comes true.
Command message: TRSE TV,SR,C1,STAN,NTSC,SYNC,ANY

21.8 TRIG_SLOPE

Description Sets the trigger slope of the specified triggers source.

Commnad Syntax <trig_source>:TRIG_SLOPE <trig_slope>
<trig_source> := {C1,C2,C3,C4,EX,EX5}
<trig_slope> := {NEG,POS,WINDOW} for edge trigger.
<trig_slope> := {NEG,POS} for other trigger.

- **NEG**: falling edg.
- **POS**: rising edge.
- **WINDOW**: altering edge.

Query Syntax <trig_source>:TRIG_SLOPE?

Response Format <trig_source>:TRIG_SLOPE <trig_slope>

Example The following command sets the trigger slope of Channel 2 to negative.
Command message: C2:TRSL NEG

Related Commands **TRIG_SELECT**

21.9 TRIG_WINDOW

Description Sets the relative hieght of the two trigger line of the trigger window type.

Note:

This command is only valid when the window type is relative.

The TRIG_WINDOW? query returns relative height of the two trigger line of the trigger window type.

Commnad Syntax TRIG_WINDOW <value>
<value> := 0 to 9*DIV when the center level is 0.

Query Syntax TRIG_WINDOW?

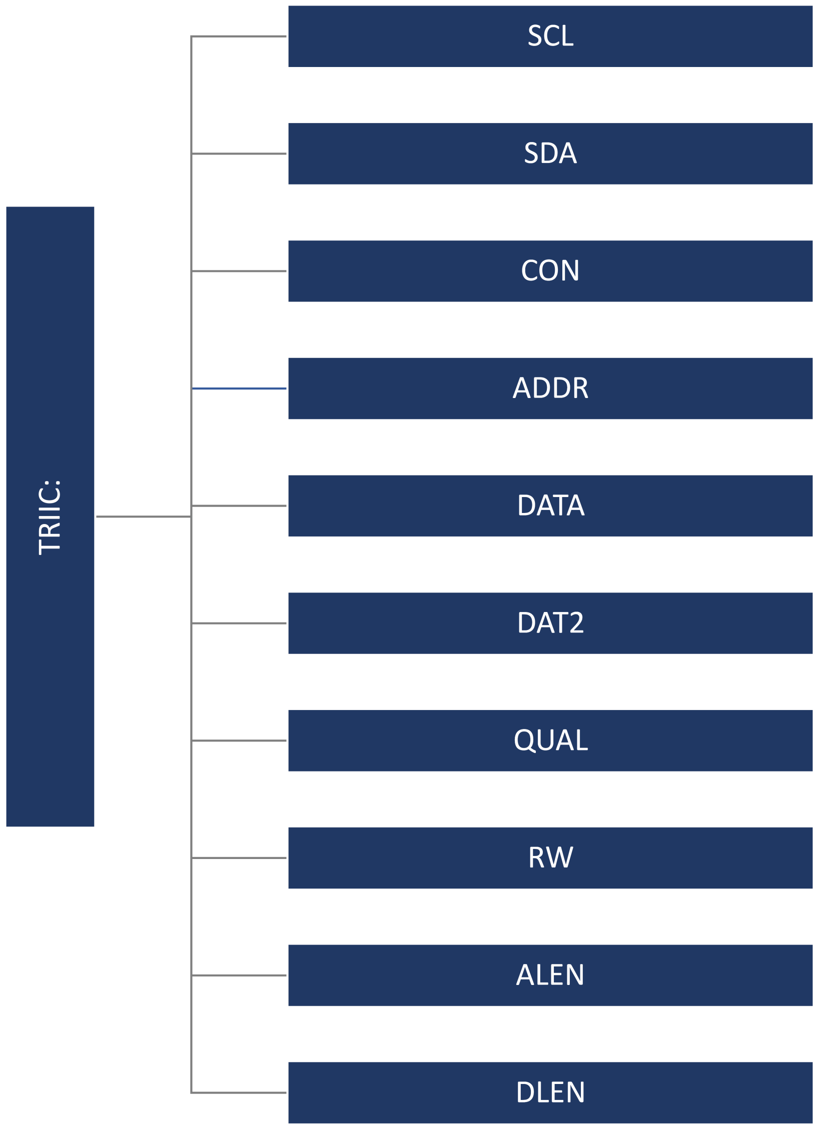
Response Format TRIG_WINDOW <value>
<value> := Numerical value in E-notation with SI unit.

Example When the window type is relative, and the center level is 1 V, the following command sets the relative height of the two trigger line to 2 V.
Command message: TRWI 2V

Related Commands **TRIG_LEVEL2**
TRIG_LEVEL

IIC Signal Commands

To set up a IIC serial trigger, set the trigger type to Serial using the command TRSE SERIAL.



22.1 TRIIC:SCL

Description	Sets the soruce and threshold for the serial clock (SCL) of IIC trigger.
Commnad Syntax	TRIG_IIC:SCL <source>[,<threshold>] <source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15} <threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax	TRIIC:SCL?
Response Format	TRIIC:SCL <source>[,<threshold>] <threshold> := numerical value in E-notation with SI unit.
Example	When the serial protocol is IIC, the following command sets the source of SCL to channel 3 and the threshold to 200 mV. **Command message**: TRIIC:SCL C3,0.2
Related Commands	**TRIIC:SDA**

22.2 TRIIC:SDA

Description	Sets the soruce and threshold for the serial data input channel (SDA) of IIC trigger.
Commnad Syntax	TRIG_IIC:SDA <source>[,<threshold>] <source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15} <threshold>:= value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax	TRIIC:SDA?
Response Format	TRIIC:SDA <source>[,<threshold>] <threshold>:= numerical value in E-notation with SI unit.
Example	When the serial protocol is IIC, the following command sets the source of SDA to channel 3 and the threshold to 200 mV. **Command message**: TRIIC:SDA C3,0.2V
Related Commands	**TRIIC:SCL**

22.3 TRIIC:CON

Description	Sets the trigger condition of IIC trigger.
Commnad Syntax	TRIG_IIC:CON <condition> <condition> := {START,STOP,RESTART,NOACK,EEPRO M,7ADDA,10ADDA,DALENTH}

- **START**: Start condition.
- **STOP**: Stop condition.
- **RESTART**: Another start condition occurs before a stop condition.
- **NOACK**: Missing acknowledge.
- **EEPROM**: EEPROM frame containing (Start:Control byte:R:Ack:Data).
- **7ADDA**: 7-bit address frame containing (Start:Address7:R/W:Ack:Data:Data2).
- **10ADDA**: 10-bit address frame containing (Start:Address10:R/W:Ack:Data:Data2).
- **DALENTH**: specifie a search based on address length and data length

Query Syntax	TRIIC:CON?
Response Format	TRIIC:CON <condition>
Example	When the serial protocol is IIC, the following command sets the trigger condition to 7 ADD&Data. **Command message**: TRIIC:CON 7ADDA
Related Commands	**TRIIC:ADDR** **TRIIC:DATA** **TRIIC:DAT2**

22.4 TRIIC:ADDR

Description Sets the address value used for the IIC trigger when the trigger condition is set to 7ADDA or 10ADDA.

Commnad Syntax TRIIC:ADDR <value>
<value> := 0 to 2^n when n is the address length(7 or 10)

Note:

Use the don't care data (128) to ignore the address value when trigger condition is 7ADDA.

Use the don't care data (1024) to ignore the
address value when trigger condition is 10ADDA.

Query Syntax TRIIC:ADDR?

Response Format TRIIC:ADDR <value>

Example When the serial protocol is IIC and the trigger condition is 10ADD&Data, the following command sets the address value to 0x122.
Command message: TRIIC:ADDR 290

Related Commands **TRIIC:CON**

22.5 TRIIC:DATA

Description Sets data1 value used for IIC trigger when the rigger condition is set to 7ADDA, 10ADDA or EEPROM.

Commnad Syntax TRIIC:DATA <value>
<value> := {0 to 256}

Note:

Use the don't care data (256) to ignore the data value.

Query Syntax TRIIC:DATA?

Response Format TRIIC:DATA <value>

Example When the serial protocol is IIC and the trigger condition is 10ADD&Data, the following command sets the data1 value to 0x29.
Command message: TRIIC:DATA 41

Related Commands **TRIIC:CON**
TRIIC:DAT2

22.6 TRIIC:DAT2

Description Sets the data2 value used for IIC trigger when the trigger condition is set to 7ADDA or 10ADDA.

Commnad Syntax TRIIC:DAT2 <value>
<value> := {0 to 256}

Note:

Use the don't care data (256) to ignore the data value.

Query Syntax TRIIC:DAT2?

Response Format TRIIC:DAT2 <value>

Example When the serial protocol is IIC and the trigger condition is 10ADD&Data, the following command sets the data2 value to 0x29.
Command message: TRIIC:DAT2 41

Related Commands **TRIIC:CON**
TRIIC:DATA
TRIIC:ADDR

22.7 TRIIC:QUAL

Description Sets the IIC data qualifier when the trigger condition is set to EEPROM.

Commnad Syntax TRIIC:QUAL <value>
<value> := {EQUAL,MORE,LESS}

- **EQUAL**: sets the IIC data qualifier to equal.
- **MORE**: sets the IIC data qualifier to greater than.
- **LESS**: sets the IIC data qualifier to less than.

Query Syntax TRIIC:QUAL?

Response Format TRIIC:QUAL <value>

Example When the serial protocol is IIC and the trigger condition is EEPROM, the following command sets the data qualifier to equal.
Command message: TRIIC:QUAL EQUAL

Related Commands **TRIIC:CON**
TRIIC:DATA
TRIIC:ADDR

22.8 TRIIC:RW

Description

sets the IIC trigger type to be valid on a Read, Write, or Either condition. Read or write is indicated by the R/W bit in the IIC protocol.

Commnad Syntax

TRIIC:RW <value>
<value> := {READ,WRITE,DONT_CARE}

- **READ**: sets read as the data direction.
- **WRITE**: sets write as the data direction.
- **DONT_CARE**: sets either as the data direction.

Query Syntax

TRIIC:RW?

Response Format

TRIIC:RW <value>

Example

When the serial protocol is IIC and the trigger condition is 10ADD&Data, the following command sets the data direction to write.
Command message: TRIIC:RW WRITE

Related Commands

TRIIC:CON
TRIIC:ADDR

22.9 TRIIC:ALEN

Description

Sets the IIC address type when the trigger condition is set to **Data Length**.

Commnad Syntax

TRIIC:ALEN <value>
<value> := {7BIT,10BIT}

Query Syntax

TRIIC:ALEN?

Response Format

TRIIC:ALEN <value>

Example

When the serial protocol is IIC and the trigger condition is Data Length, the following command sets the address type to 7 bit.
Command message: TRIIC:ALEN 7BIT

Related Commands

TRIIC:CON
TRIIC:ADDR

22.10 TRIIC:DLEN

Description

Sets the length of the data in bytes to be used for IIC trigger if the trigger condition is **Data Length**.

Commnad Syntax

TRIIC:DLEN <value>
<value> := {1 to 12}

Query Syntax

TRIIC:DLEN?

Response Format

TRIIC:DLEN <value>

Example

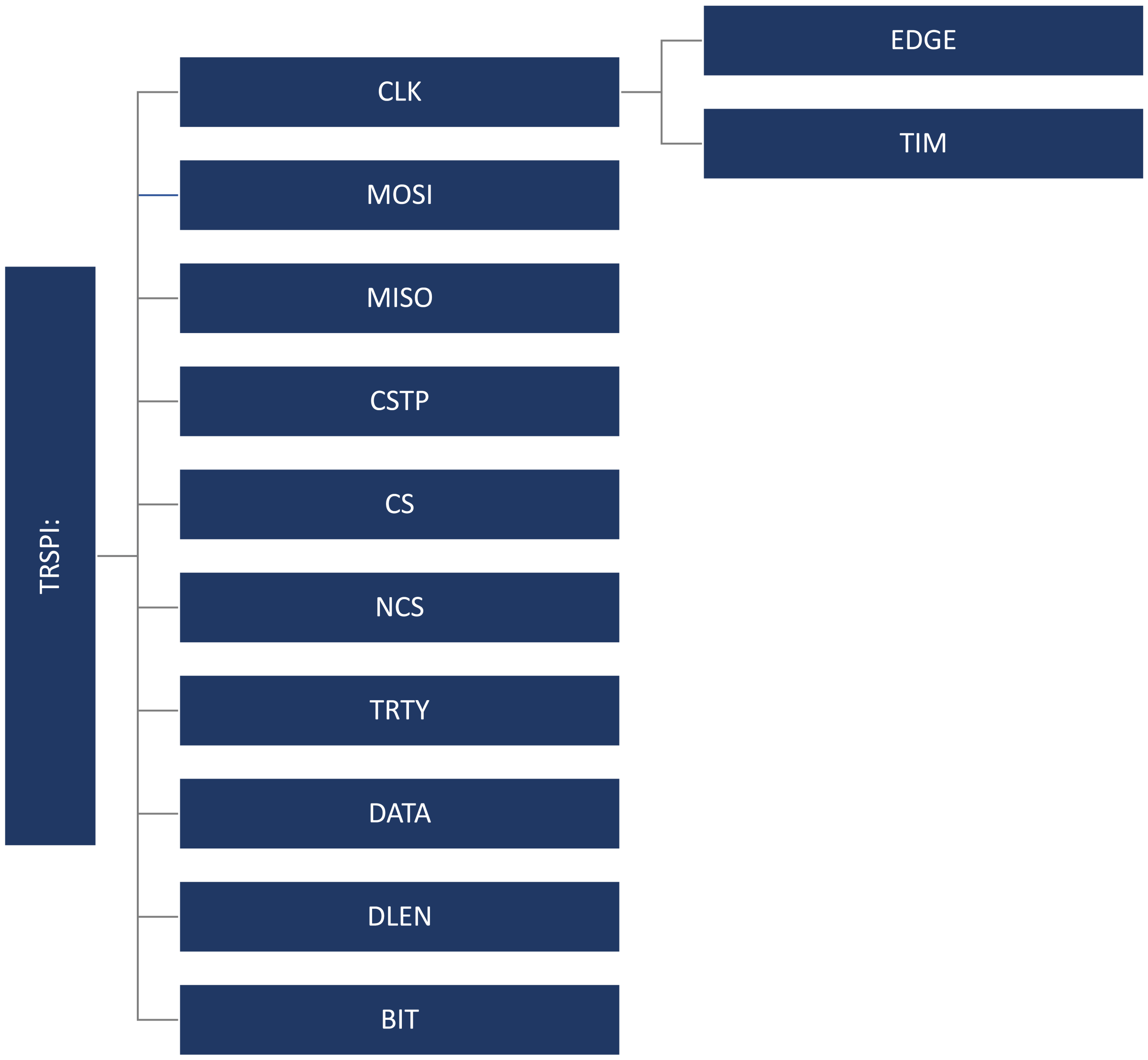
When the serial protocol is IIC and the trigger condition is Data Length, the following command sets the length of the data bytes to 8.
Command message: TRIIC:DLEN

Related Commands

TRIIC:CON
TRIIC:ADDR

SPI Trigger Commands

To set up a SPI serial trigger, set the trigger type to **Serial** using the command TRSE SERIAL.



23.1 TRSPI:CLK

Description Sets the source and threshold for the serial clock of SPI trigger.

Commnad Syntax TRSPI:CLK <source>[,<threshold>]
<source> :={C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRSPI:CLK?

Response Format TRSPI:CLK <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is SPI, the following command sets the source of CLK to channel 3 and the threshold to 200 mV.
Command message: TRSPI:CLK C3,0.2V

Related Commands **TRSPI:MISO**
TRSPI:MOSI

23.2 TRSPI:CLK:EDGE

Description Sets the edge of the clock that data latches on.

Commnad Syntax TRSPI:CLK:EDGE <edge>
<edge> := {RISING,FALLING}

Query Syntax TRSPI:CLK:EDGE?

Response Format TRSPI:CLK:EDGE <edge>

Example When the serial protocol is SPI, the following command sets the edge of the clock to rising.
Command message: TRSPI:CLK:EDGE RISING

Related Commands **TRSPI:CLK**

23.3 TRSPI:CLK:TIM

Description	Sets the timeout value for the clock of SPI trigger when the CS type is set to CLK Timeout.
Commnad Syntax	TRSPI:CLK:TIM <value> <value> := {100ns to 5ms}, value with unit
Query Syntax	TRSPI:CLK:TIM?
Response Format	TRSPI:CLK:TIM <value> <threshold>:= numerical value in E-notation with SI unit.
Example	When the serial protocol is SPI and the CS type is CLK Timeout, the following command sets the timeout value for the clock to 2us. **Command message**: TRSPI:CLK:TIM 2us
Related Commands	**TRSPI:CSTP**

23.4 TRSPI:MOSI

Description Sets the source and threshold for MOSI of SPI trigger.

Commnad Syntax TRSPI:MOSI <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRSPI:MOSI?

Response Format TRSPI:MOSI <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is SPI, the following command sets the source of MOSI to channel 3 and the threshold to 200 mV.
Command message: TRSPI:MOSI C3,0.2V

Related Commands **TRSPI:MISO**

23.5 TRSPI:MISO

Description	Sets the source and threshold for MISO of SPI trigger.
Commnad Syntax	TRSPI:MISO <source>[,<threshold>] <source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15} <threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax	TRSPI:MISO?
Response Format	TRSPI:MISO <source>[,<threshold>] <threshold> := numerical value in E-notation with SI unit.
Example	When the serial protocol is SPI, the following command sets the source of MISO to channel 3 and the threshold to 200 mV. **Command message**: TRSPI:MISO C3,0.2V
Related Commands	**TRSPI:MOSI**

23.6 TRSPI:CSTP

Description	Sets the serial chip selection type of SPI trigger.
Commnad Syntax	TRSPI:CSTP <type> <type> := {CS,NCS,TIMEOUT}
Query Syntax	TRSPI:CSTP?
Response Format	TRSPI:CSTP <type>
Example	When the serial protocol is SPI, the following command sets the CS type to CS. **Command message**: TRSPI:CSTP CS
Related Commands	**TRSPI:CS** **TRSPI:NCS** **TRSPI:CLK:TIM**

23.7 TRSPI:CS

Description Sets the source and threshold for CS signal of SPI trigger when the CS type is CS.

Commnad Syntax TRSPI:CS <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRSPI:CS?

Response Format TRSPI:CS <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is SPI and the CS type is CS, the following command sets the source of CS to channel3 and the threshold to 200 mV.
Command message: TRSPI:CS C3,0.2V

Related Commands **TRSPI:CSTP**

23.8 TRSPI:NCS

Description Sets the source and threshold for CS signal of SPI trigger when the ~CS type is ~CS

Commnad Syntax TRSPI:NCS <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRSPI:NCS?

Response Format TRSPI:NCS <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is SPI and the ~CS type is ~CS, the following command sets the source of ~CS to channel 3 and the threshold to 200 mV.
Command message: TRSPI:NCS C3,0.2V

Related Commands **TRSPI:CSTP**

23.9 TRSPI:TRTY

Description Sets the trigger source for SPI trigger.

Commnad Syntax TRSPI:TRTY <source>
<source> := {MOSI,MISO}

Query Syntax	TRSPI:TRTY?
Response Format	TRSPI:TRTY <source>
Example	When the serial protocol is SPI, the following command sets the trigger source to MOSI. **Command message**: TRSPI:TRTY MOS

23.10 TRSPI:DATA

Description	Sets the data value of every bit used for SPI trigger.
Commnad Syntax	TRSPI:DATA <value1>[,<value2>[,..[,<value>]]] <value> := {0,1,X}
Example	When the serial protocol is SPI and the data length is 4, the following command sets data value to 1011. **Command message**: TRSPI:DATA 1,0,1,1
Related Commands	**TRSPI:DLEN**

23.11 TRSPI:DLEN

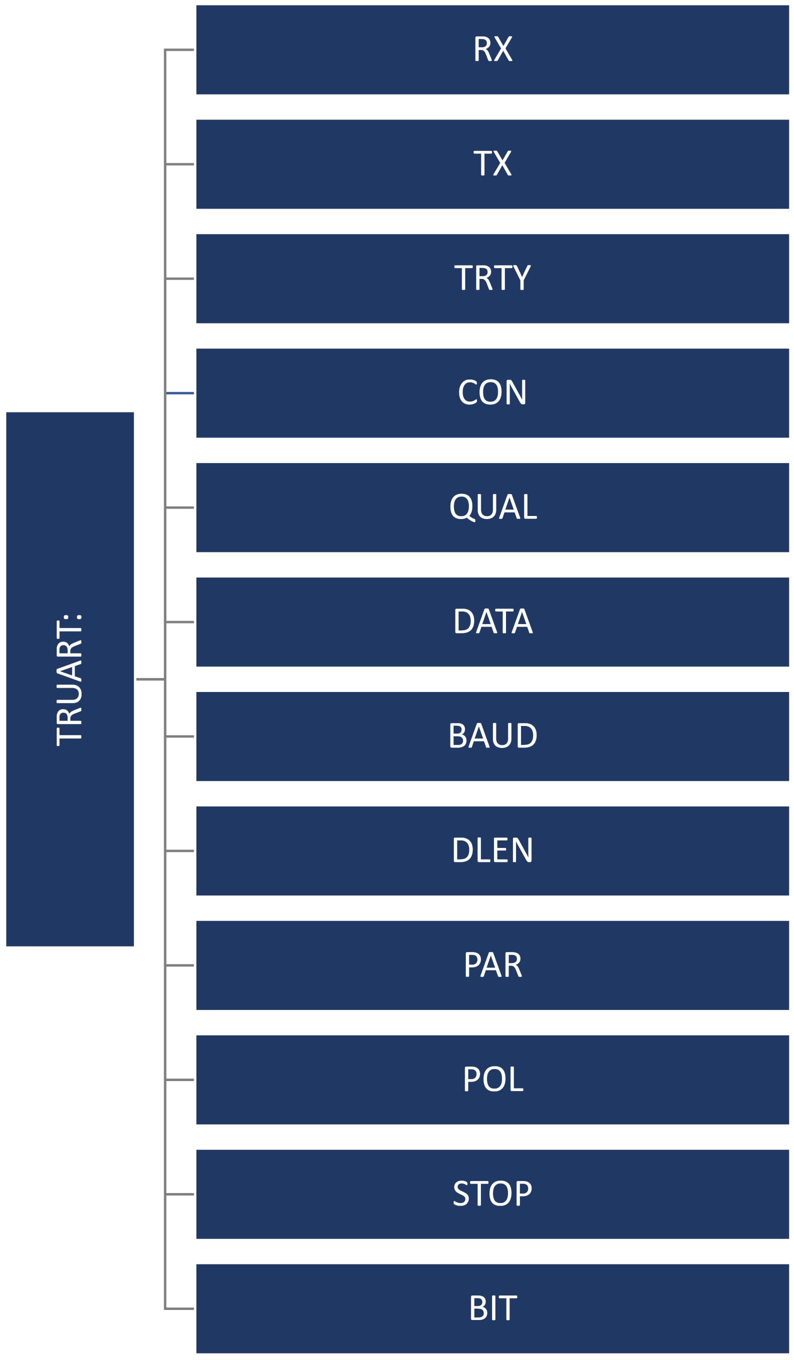
Description	Sets the length of data for SPI trigger.
Commnad Syntax	TRSPI:DLEN <value> <value> := {4 to 96}
Query Syntax	TRSPI:DLEN?
Response Format	TRSPI:DLEN <value>
Example	When the serial protocol is SPI, the following command sets the trigger data length to 8. **Command message**: TRSPI:DLEN 8
Related Commands	**TRSPI:DATA**

23.12 TRSPI:BIT

Description	Sets the bit order for SPI trigger.
Commnad Syntax	TRSPI:BIT <order> <order> := {MSB,LSB}
Query Syntax	TRSPI:BIT?
Response Format	TRSPI:BIT <order>
Example	When the serial protocol is SPI, the following command sets the bit order to MSB. **Command message**: TRSPI:BIT MSB

UART Trigger Commands

To setup a UART serial trigger, set the trigger type to **Serial** using the command TRSE SERIAL.



24.1 TRUART:RX

Description Sets the source and threshold for RX of UART trigger.

Commnad Syntax TRUART:RX <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRUART:RX?

Response Format TRUART:RX <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit

Example When the serial protocol is UART, the following command sets the source of RX to channel 3 and the threshold to 200 mV.
Command message: TRUART:RX C3,0.2V

Related Commands **TRUART:TX**

24.2 TRUART:TX

Description Sets the source and threshold for TX.

Commnad Syntax TRIG_UART:TX <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRUART:TX?

Response Format TRUART:TX <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is UART, the following command sets the source of TX to channel 3 and the threshold to 200 mV. Command message: TRUART:TX C3,0.2V

Related Commands **TRUART:RX**

24.3 TRUART:TRTY

Description Sets the trigger source for UART trigger.

Commnad Syntax TRUART:TRTY <source>
<source> := {RX,TX}

Query Syntax TRUART:TRTY?

Response Format TRUART:TRTY <source>

Example When the serial protocol is UART, the following command sets the trigger source to TX.
Command message: TRUART:TRTY TX

Related Commands TRUART:RX
TRUART:TX
TRSPI:DLEN

24.4 TRUART:CON

Description Sets the trigger condition of UART trigger.

Commnad Syntax TRUART:CON <condition>
<condition> := {START,STOP,DATA,ERROR}

- **START**: Start condition.
- **STOP**: Stop condition.
- **DATA**: Specify a search based on data.
- **ERROR**: Error condition.

Query Syntax TRUART:CON

Response Format TRUART:CON <condition>

Example When the serial protocol is UART, the following command sets the trigger condition to START.

Command message: TRUART:CON START

24.5 TRUART:QUAL

Description Sets the UART data qualifier when the trigger condition is set to DATA.

Commnad Syntax TRUART:QUAL <condition>
<condition>:= {EQUAL,MORE,LESS }

- **EQUAL**: sets the UART data qualifier to equal.
- **MORE**: sets the UART data qualifier to greater than.
- **LESS**: sets the UART data qualifier to less than.

Query Syntax TRUART:QUAL?

Response Format TRUART:QUAL <condition>

Example When the serial protocol is UART and the trigger condition is DATA, the following command sets the data qualifier to EQUAL.
Command message: TRUART:QUAL EQUAL

Related Commands **TRUART:CON**

24.6 TRUART:DATA

Description Sets the data value used for UART trigger when the trigger condition is set to DATA.

Commnad Syntax TRUART:DATA <value>
<value>:= 0 to 256

Note:

Use the don't care data (256) to ignore the data value.

Query Syntax TRUART:DATA?

Response Format TRUART:DATA <value>

Example When the serial protocol is UART and the trigger condition is DATA, the following command sets the data value to 0x29.
Command message: TRUART:DATA 41

Related Commands **TRUART:CON**

24.7 TRUART:BAUD

Description Sets the baud rate value used for UART trigger.

Commnad Syntax TRUART:BAUD <value1>[,<value2>]
<value1> := {600,1200,2400,4800,9600,19200,38400,5760 0,115200,CUSTOM}
<value2> := 300 to 5000000 When the value1 is CUSTOM.

Query Syntax TRUART:BAUD?

Response Format TRUART:BAUD <value>[,<value2>

Example When the serial protocol is UART the following command sets the baud rate value to 9600 bit/s.

Command message: TRUART:BAUD 9600

24.8 TRUART:DLEN

Description Sets the data length value used for UART trigger.

Commnad Syntax TRUART:DLEN <value>
<value> :={ 5 to 8}

Query Syntax TRUART:DLEN?

Response Format	TRUART:DLEN <value>
Example	When the serial protocol is UART, the following command sets data length value to 6. **Command message**: TRUART:DLEN 6

24.9 TRUART:PAR

Description	Sets the parity check used for UART trigger.
Commnad Syntax	TRUART:PAR <value> <value> := {NONE,ODD,EVEN}
Query Syntax	TRUART:PAR?
Response Format	TRUART:PAR <value>
Example	When the serial protocol is UART, the following command sets parity check to odd. **Command message**: TRUART:PAR ODD

24.10 TRUART:POL

Description	Sets the idle level used for UART trigger.
Commnad Syntax	TRUART:POL <value> <value> := {LOW,HIGH}
Query Syntax	TRUART:POL?
Response Format	TRUART:POL <value>
Example	When the serial protocol is UART, the following command sets idle level to low. **Command message**: TRUART:POL LOW

24.11 TRUART:STOP

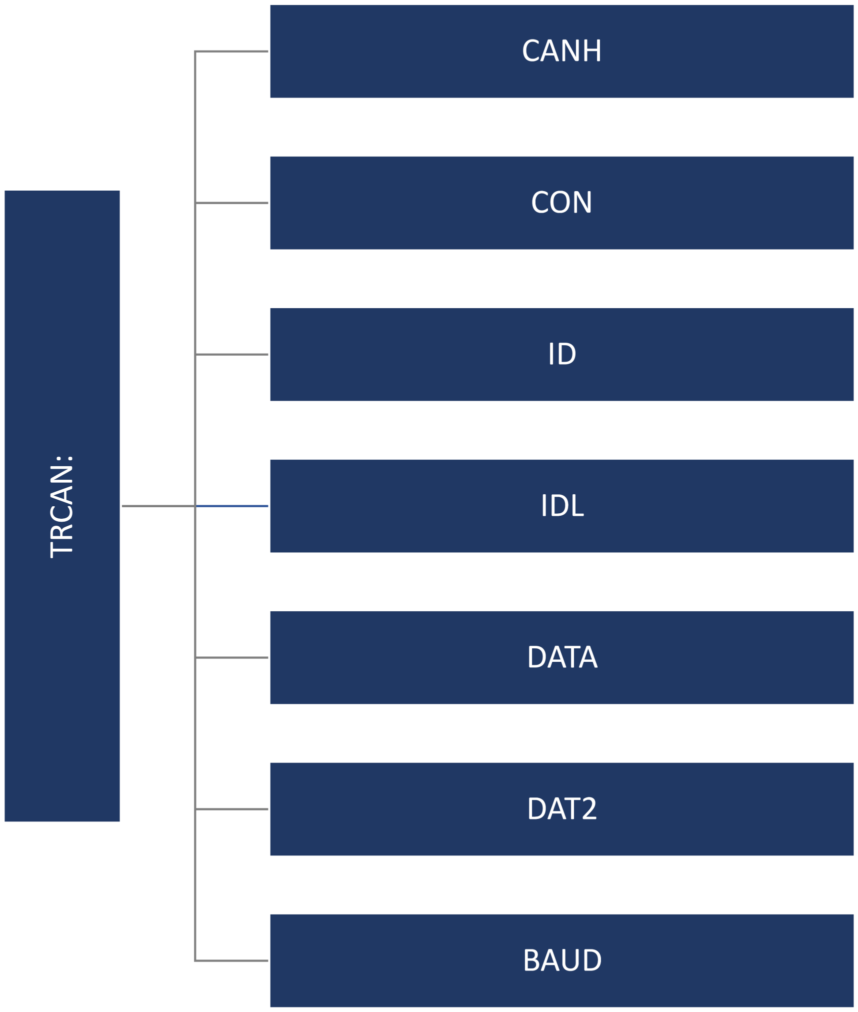
Description	Sets the length of stop bit for UART trigger.
Commnad Syntax	TRUART:STOP <value> <value> := {1,1.5,2}
Query Syntax	TRUART:STOP?
Response Format	TRUART:STOP <value>
Example	When the serial protocol is UART, the following command sets the length of stop bit to 1. **Command message**: TRUART:STOP 1

24.12 TRUART:BIT

Description	Sets the bit order for UART trigger.
Commnad Syntax	TRIG_UART:BIT <value> <value> := {LSB,MSB}
Query Syntax	TRUART:BIT?
Response Format	TRUART:BIT <value>
Example	When the serial protocol is UART, the following command sets the bit order to MSB. **Command message**: TRUART:BIT MSB

CAN Trigger Commands

To setup a CAN serial trigger, set the trigger type to **Serial** using the command TRSE SERIAL.



25.1 TRCAN:CANH

Description Sets the source and threshold for source of CAN trigger.

Commnad Syntax TRCAN:CANH <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> := value with unit. It is necessary to set when the source is analog channel.

Note:

The range of threshold is related to the vertical scale of the source.

Query Syntax TRCAN:CANH?

Response Format TRCAN:CANH <source>[,<threshold>]

Example When the serial protocol is CAN, the following command sets the source to channel 3 and the threshold to 200 mV.

bf Command message: TRCAN:CANH C3,0.2V

25.2 TRCAN:CON

Description Sets the trigger condition of CAN trigger.

Commnad Syntax TRCAN:CON <condition>
<condition> := {START,REMOTE,ID,ID_AND_DATA,ERROR}

- **START**: Start condition.
- **REMOTE**: Remote frame
- **ID**: Specifies a search based on ID bits and ID.
- **ID_AND_DATA**: Specify a search based on ID bits, ID and data.
- **ERROR**: Error frame.

Query Syntax TRCAN:CON?

Response Format TRCAN:CON <condition>

Example When the serial protocol is CAN, the following command sets the trigger condition to START.
Command message: TRCAN:CON START

Related Commands **TRCAN:ID**
TRCAN:DATA
TRCAN:DAT2

25.3 TRCAN:ID

Description Sets he ID value for CAN trigger when the trigger condition is set to ID or ID_AND_DATA.

Commnad Syntax TRCAN:ID <value>
<value> := $\{0$ to $2^n\}$ when n is the ID bits (11 or 29)

Note:

Use the don't care data (2048) to ignore the address value when trigger condition is ID or ID_AND_DATA, and the ID length is 11.

Use the don't care data (536870912) to ignore the address value when trigger condition is ID or ID_AND_DATA, and the ID length is 29.

Query Syntax TRCAN:ID?

Response Format TRCAN:ID <value>

Example When the serial protocol is CAN and the trigger condition is ID, the following command sets the ID value to 0x29.
Command message: TRCAN:ID 41

Related Commands **TRCAN:CON**
TRCAN:IDL

25.4 TRCAN:IDL

Description Sets the ID length for CAN trigger when the trigger condition is set to ID or ID_AND_DATA.

Commnad Syntax TRCAN:IDL <value>
<value> := {11BITS,29BITS}

Query Syntax TRCAN:IDL <value>

Response Format TRCAN:IDL <value>

Example When the serial protocol is CAN and the trigger condition is ID, the following command sets the ID length value to 11bits.
Command message: TRCAN:IDL 11BITS

Related Commands **TRCAN:CON**

25.5 TRCAN:DATA

Description Sets he data1 value used for CAN trigger when the trigger condition is set to ID_AND_DATA.

Commnad Syntax TRCAN:DATA <value>
<value> := {0 to 256}

Note:

Use the don't care data (256) to ignore the data value.

Query Syntax TRCAN:DATA?

Response Format TRCAN:DATA <value>

Example When the serial protocol is CAN and the trigger condition is ID_AND_DATA, the following command sets the data1 value to 0x29.
Command message: TRCAN:DATA 41

Related Commands **TRCAN:CON**
TRCAN:DAT2

25.6 TRCAN:DAT2

Description Sets the data2 value used for CAN trigger when the trigger condition is set to ID_AND_DATA.

Commnad Syntax TRCAN:DAT2 <value>
<value> := {0 to 256}

Note:

Use the don't care data (256) to ignore the data value.

Query Syntax TRCAN:DAT2?

Response Format TRCAN:DAT2 <value>

Example When the serial protocol is CAN and the trigger condition is ID_AND_DATA, the following command sets the data1 value to 0x29.
Command message: TRCAN:DAT2 41

Related Commands **TRCAN:CON**
TRCAN:DATA

25.7 TRCAN:BAUD

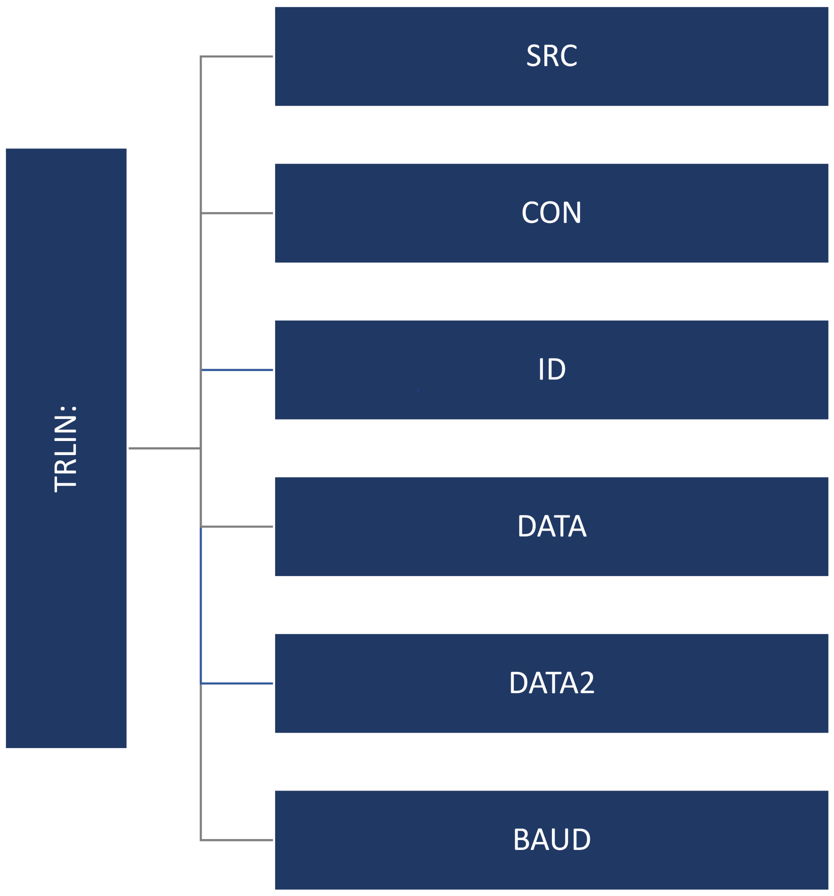
Description	Sets the baud rate for CAN trigger.

Commnad Syntax	TRCAN:BAUD <value1>[,<value2>]
	<value1> := {5k,10k,20k,59k,100k,125k,250,500k,800k,1M, CUSTOM}
	<value2> := {5000 to 1000000 When the value1 is CUSTOM.}

Query Syntax	TRCAN:BAUD?
Response Format	TRCAN:BAUD <value1>[,<value2>]
Example	When the serial protocol is CAN the following command sets the baud rate to 5 kbit/s. **Command message**: TRCAN:BAUD 5k

LIN Trigger Commands

To setup a LIN serial trigger, set the trigger type to **Serial** using the command TRSE SERIAL.



26.1 TRLIN:SRC

Description Sets the source and threshold for the source of LIN trigger.

Commnad Syntax TRLIN:SRC <source>[,<threshold>]
<source> := {C1,C2,C3,C4,D0,D1,D2,D3,D4,D5,D6,D7,D8, D9,D10,D11,D12,D13,D14,D15}
<threshold> `:= value with unit. It is necessary to set when the source is analog channel.

Note:

The range of value is related to the vertical scale of the source.

Query Syntax TRLIN:SRC?

Response Format TRLIN:SRC <source>[,<threshold>]
<threshold> := numerical value in E-notation with SI unit.

Example When the serial protocol is LIN, the following command sets the source to channel 3 and the threshold to 200 mV.
Command message: TRLIN:SRC C3,0.2V

26.2 TRLIN:CON

Description Sets teh triger condition of LIN trigger.

Commnad Syntax TRIG_LIN:CON <condition>
<condition> := {BREAK,ID,ID_AND_DATA,DATA_ERROR}

- **BREAK**: Break condition.
- **ID**: Specify a search based on ID.
- **ID_AND_DATA**: Specify a search based on ID and data.
- **DATA_ERROR**: Error frame.

Query Syntax TRLIN:CON?

Response Format TRLIN:CON <condition>

Example When the serial protocol is LIN, the following command sets the trigger condition to break.
Command message: TRLIN:CON BREAK

Related Commands **TRLIN:ID**
TRLIN:DATA
TRLIN:DAT2

26.3 TRLIN:ID

Description Sets the ID value for LIN trigger when the trigger condition is set to ID or ID_AND _DATA.

Commnad Syntax TRLIN:ID <value>
<value> := {0 to 64}

Note:

Use the don't care data (64) to ignore the ID value.

Query Syntax TRLIN:ID?

Response Format TRLIN:ID <value>

Example When the serial protocol is LIN and the trigger condition is ID, the following command sets the data1 value to 0x29.
Command message: TRLIN:ID 41

Related Commands **TRLIN:CON**

26.4 TRLIN:DATA

Description Sets the data1 value used for LIN trigger when the trigger condition is set to ID_AND_DATA.

Commnad Syntax TRIG_LIN:DATA <value>
<value> := {0 to 256}

Query Syntax TRLIN:DATA?

Response Format TRLIN:DATA <value>

Example When the serial protocol is LIN and the trigger condition is ID_AND_DATA, the following command sets the data1 value to 0x29.
Command message: TRLIN:DATA 41

Related Commands **TRLIN:CON**
TRLIN:DAT2

26.5 TRLIN:DAT2

Description
Sets the data2 value used for LIN trigger when the trigger condition is set to ID_AND_DATA.

Commnad Syntax
TRLIN:DAT2 <value>
<value> := {0 to 256}

Note:

Use the don't care data (256) to ignore the ID value.

Query Syntax
TRLIN:DAT2?

Response Format
TRLIN:DAT2 <value>

Example
When the serial protocol is LIN and the trigger condition is ID_AND_DATA, the following command sets the data2 value to 0x29.
Command message: TRLIN:DAT2 41

Related Commands
TRLIN:CON
TRLIN:DATA

26.6 TRLIN:BAUD

Description
Sets the baud rate for LIN trigger.

Commnad Syntax
TRLIN:BAUD <value1>[,<value2>]
<value1> := {600,1200,2400,4800,9600,19200,CUSTOM}
<value2> := {300 to 20000 When the value1 is CUSTOM}

Query Syntax
TRCAN:BAUD?

Response Format
TRLIN:BAUD <value1>[,<value2>]

Example
When the serial protocol is LIN, the following command sets the baud rate value to 9600 bit/s.
Command message: TRLIN:BAUD 9600

Waveform Commands

The WAVEFORM subsystem is used to transfer data to a controller from the oscilloscope waveform memory.

The waveform record is contained in two portions: the preamble and waveform data. The waveform record must be read from the oscilloscope by the controller using two separate commands. The waveform data is the actual data acquired for each point in the specified source. The preamble contains the information for interpreting the waveform data.

WF?

WFSU

27.1 WAVEFORM?

Description Transfers a waveform from the oscilloscope to the controller.

Note:

The format of the waveform data depends on the current settings specified by the last WFSU command.

When using the visa library:

Set the I/O buffer size The read buffer size depends on the number of waveform points. When it needs to read in segments, the size of each segment is vary from the models.

Set the timeout value The timeout value is related to the network speed or USB transmission speed. Please evaluate by yourself. The initial value is generally 2s.

Query Syntax <trace>:WAVEFORM? <section>
<trace> :={C1,C2,C3,C4,MATH}

- **C[X]** : Analog channel.
- **MATH** : Not valid for FFT.

<section> := {DAT2}

- **DAT2**: Return the main data include the head, the wave data and the ending flag. The length of data is current memory depth

Response Format <trace>:WAVEFORM <data block>

Example

- Send the query to get the data of waveform. **Query message**: C1:WF? DAT2

  Response message: The head of message: C1:WF DAT2. These are followed by the string #9000000070, the beginning of a binary block in which nine ASCII integers are used to give the length of the block (70 bytes). After the length of block, is beginning of the wave data. "0A 0A" means the end of data.

  C1:WF DAT2,#9002800000 DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DF DF DF DF DF DF DF DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DF DF DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DF DF DF DF DE DE DE DE DE DF DF DF DE DE DE DE DE DF DF DF DE DE DE
  DE DE DE DE DE DE DE DF DF DF DF DF DE DF DF DF DF DF DF DE DE DE DE DE
  DE DF DF DF DF DF DE DE DE DE DF DF DF DF DE DE DE DE DE DE DE DE DE DE
  DE DD DD DD DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
  DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DF DE DE

```
DE DE DE DE DE DE DF DF DF DE DE DE DE DE DF DF DF DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DD DD DD DE
DE DE DE DE DE DD DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DF DF DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DD DD DD DD DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DD DD DD DD DD DE DE DE DE DE DE DE DE DE DE DE DD DD
DD DE DE DE DF DF DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DD DD DD DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DF DF DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DF
DF DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DD DD DD DE DE DE DE DE DE DE DE DE DD DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DF DF DF DE DE DE DE DE
DE DE DF DE DE DE DE DE DE DE DE DE DE DE DE DE DD DD DD DD DD DD DD DD
DD DD DE DE DE DE DD DD DD DD DD DD DE DE DE DE DE DE DE DE DE DE DE
DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DE DD DD DD DD DD DD DE DE
DE DE DD DD DD DD DD DE DE DE DE DD DD DD DD DD DD DD DD DD DD DD
DE DE DD DD DD DD DD DE DE DE DE DE DE DE DE DD DE DE DE DE DE DE DE DE
DD DD DE DE DE DE DE DE DE DE DE DE DD DE DE DE DE DE DD DD DD DD DD
DD DD DE DE DE DE DE DE DE DE DE DE DE DD DD DD DD DD DD DD DE DE DE
DE DE DE DD DD DD DD DD DD DD DD DD DD DE DE DE DD DD DD DD DD DE DE
DE DE DE DD DD DD DD DE DE DE DE DE DE DD DD DD DD DE DE DE DE DE DE
DE DE DE DE DE DE DD
```

- Calculate the voltage value corresponding to the data point. Using the formula:

$$voltage\ value(V) = code\ value * (vdiv/25) - voffset$$

code value: The decimal of wave data.

Note:

If the decimal is greater than "127", it should minus 256.Then the value is code value. Such as the wave data is "FC" convert to decimal is "252". So the code value is 252-256 = -4.)
vdiv: The Volts/div value.
voffset: The voltage position value.

- To acquire the remaining parameters:
  - **Send command** C1:VDIV?
  - **Send command** C1:OFST?
- Calculate the time value of the data point. Using the formula:

$$time\ value\ (S) = -(timebase * grid/2)$$

timebase: The timebase value. grid: The grid numbers in horizontal direction.

- To acquire the remaining parameters:
  - **Send command** TDIV?
  - **Send command** SARA?

Use python to reconstruct the waveform:

Note:

If you want the command return the “numerical” data type only (i.e. return as “1.00E+09” when send the command “SARA?”), send the command “CHDR OFF” at the first. See CHDR for details.

```
import pyvisa
import pylab as pl
def main():
    rm = pyvisa.ResourceManager()
    inst = rm.open_resource("USB0::0x3121::0x2100::SDSAHBAD4R0328::INSTR")
    inst.write("chdr off")
    vdiv = inst.query("c1:vdiv?")
    ofst = inst.query("c1:ofst?")
    tdiv = inst.query("tdiv?")
    sara = inst.query("sara?")
    sara_unit = {'G':1E9,'M':1E6,'k':1E3}
    for unit in sara_unit.keys():
        if sara.find(unit)!=-1:
            sara = sara.split(unit)
            sara = float(sara[0])*sara_unit[unit]
            break
    sara = float(sara)
    inst.timeout = 30000 #default value is 2000(2s)
    inst.chunk_size = 20*1024*1024 #default value is 20*1024(20k bytes)
    inst.write("c1:wf? dat2")
    recv = list(inst.read_raw())[15:]
    print(len(recv))
    recv.pop()
    recv.pop()
    volt_value = []
    for data in recv:
        if data > 127:
            data = data - 256
        else:
            pass
        volt_value.append(data)
    time_value = []
    for idx in range(0,len(volt_value)):
        volt_value[idx] = volt_value[idx]/25*float(vdiv)-float(ofst)
        time_data = -(float(tdiv)*14/2)+idx*(1/sara)
        time_value.append(time_data)
    print("Data covert finish,start to draw")
    pl.figure(figsize=(7,5))
    pl.plot(time_value,volt_value,markersize=2,label=u"Y-T")
    pl.legend()
    pl.grid()
    pl.show()

if __name__=='__main__':
    main()
```

Related Commands **WAVEFORM_SETUP**

27.2 WAVEFORM_SETUP

Description Specifies the amount of data in a waveform to be transmitted to the controller.

Commnad Syntax WAVEFORM_SETUP SP,<sparsing>,NP,<number>,FP,<point>

- **SP**: Sparse point. It defines the interval between data points.
  For example:
  - **SP** = 0 sends all data points.
  - **SP** = 1 sends all data points.
  - **SP** = 4 sends every 4th data point
- **NP**: Number of points. It indicates how many points should be transmitted.
  For example:
  - **NP** = 0 sends all data points.
  - **NP** = 50 sends a maximum of 50 data points.
- **FP**: First point. It specifies the address of the first data point to be sent.
  For example:
  - **FP** = 0 corresponds to the first data point.
  - **FP** = 1 corresponds to the second data point.

Note:

Set the sparse point or number of points or the first point using key-value pairs alone. See the example for details.

After power on, SP is set to 0, NP is set to 0, and FP is set to 0.

Query Syntax WAVEFORM_SETUP?

Response Format WAVEFORM_SETUP SP,<sparsing>,NP,<number>,FP,<point>

Example The following command specifies that every 3th data point (SP=3) starting at the 200th point should be transferred.
Command message: WFSU SP,3,FP,200

Related Commands **WAVEFORM?**