



PicoScope® 9300 Series

PC Sampling Oscilloscopes

Programmer's Guide



Contents

1 PicoScope 9300 API Reference	1
1 PicoScope9300 COM Server	1
2 ExecCommand Method	1
3 COMRC Object	1
2 Commands Syntax	2
1 Command and Query Structure	2
1 Overview	2
2 Messages	2
3 Commands	3
4 Queries	3
5 Headers	3
2 Command Entry	4
1 Rules	4
2 Concatenation	4
3 Command Classification	6
1 Execution-type commands	6
2 On/off-type commands	6
3 On/off-group-type commands	6
4 Selector-type commands	8
5 Integer-type commands	8
6 Float-type commands	9
7 Data-type commands	9
4 Full list of commands	10
1 Header commands	10
2 GUI commands	10
3 System commands	11
4 Channels commands	13
5 Timebase commands	17
6 Trigger commands	20
1 Trigger Source	20
2 External Direct Trigger Commands	20
3 External Prescale Trigger Commands	20
4 Ch1/Ch2 Direct Trigger Commands	21
5 Trigger Period for Internal Clock Sources	22
6 Trigger Mode	22
7 Trigger Holdoff Commands	22
8 Attenuator Commands	23
9 Pattern Sync Trigger Commands	23
7 Acquisition commands	25
1 Sampling Mode	25

2	Channel Acquisition Commands	25
3	Termination of the Acquisition	26
4	Number of Waveforms	26
5	Action when Number of Waveforms reached	26
6	File Name	27
7	Stored Files Format	27
8	Display commands	28
9	Save/Recall commands	31
1	Work with Memo Zones (M1, M2, M3, M4)	31
2	Memory Scaling	31
3	Work with Disk	33
4	Work with Setups	35
10	Markers commands	37
11	Measure commands	38
1	Common Measures commands	38
2	Statistic Commands	39
3	Define parameter Commands	39
4	List of Measurements	42
5	Inter-signal Measurements	43
6	Measurements of Spectrum Signals	43
7	Delete all Measurements for all Sources	45
8	Getting Measurement Results	45
12	Limit Tests commands	46
1	Limit Test On/Off	46
2	Limit Test Termination Commands	46
3	Limit Test Action Commands	46
4	Parameter Definition Commands	47
13	Mathematics commands	50
1	Enable Mathematical Function	50
2	DisplayMathematical Function	50
3	Function Category	50
4	Function Operators	50
5	Function Operands	52
6	Additional Parameters for Arithmetic Functions	53
7	Additional Parameters for Algebraic Functions	54
8	Additional Parameters for Trigonometric Functions	54
9	Additional Parameters for FFT Functions	54
10	Additional Parameters for Bit Functions	55
11	Additional Parameters for Miscellaneous Functions	56
12	Function Scaling	57
14	Histogram commands	59
1	General Histogram Commands	59
2	Histogram Completion Commands	59

3 Histogram Window Commands	60
4 Histogram Calculation Commands	61
5 Histogram Scale Commands	61
6 Histogram Result Commands	63
15 Eye Diagram commands	64
1 General Eye Commands	64
2 Eye Measurements Commands	64
3 Define Parameters Commands	66
4 Eye Calculation Commands	67
5 Getting Eye Measurement Results	68
16 Mask Test commands	69
1 Common Mask Test Commands	69
2 Mask Creating	69
3 Standard Mask Test Commands	70
4 Automask Commands	72
5 Mask Test Termination	73
6 Mask Test Actions	74
7 User Mask	74
8 Getting Mask Test Results	75
17 Autocalibration commands	77
1 Single-shot Autocalibration	77
2 Periodic Autocalibration	78
3 Balancing the channels manually	79
18 Waveforms commands	80
19 Optical commands (PicoScope 9321 only)	82
1 Common optical commands	82
2 User-defined calibration points commands	83
3 Dark Level calibration commands	83
4 Calibration values saving commands	84
20 TDR/TDT Commands	85
1 Stimuli source command (PicoScope 9311 and 9312 only)	85
2 Common Internal Stimuli commands (PicoScope 9311/9312 only)	85
3 PicoScope 9311 Internal Stimuli commands	86
4 PicoScope 9312 Internal Stimuli commands	86
5 Reflectometer Setup commands	87
21 Instrument Info Commands	91
5 Revision history	94
Index	97



1 PicoScope 9300 API Reference

PicoScope 9000 provides an API for any third-party application or library to control the oscilloscope and collect signals. The API is *COM-based* and is provided by the PicoScope 9300 GUI application.

1.1 PicoScope9300 COM Server

The COM server implementing the API is called *PicoScope9300* and is implemented by the PicoScope 9300 GUI application (*PicoScope9300.exe*). It is registered in the system during the setup process, and can be explicitly unregistered and registered again by executing *PicoScope9000.exe* with the */UnregServer* or */RegServer* switches.

1.2 ExecCommand Method

The *COMRC* object contains only one method, *ExecCommand*. This method has one argument, a text string with a command or query. The method returns:

- *NULL* (*Nothing* in Visual Basic) if a command without query has been successfully executed
- The text string *ERROR* if the command was invalid
- Another text string with query results if either the command was a query or a command with a query

The syntax of the commands and queries and the full list of commands are described in the following pages.

1.3 COMRC Object

To implement the API the server exposes only one object, *COMRC*. This object supports automation, so it can be used by high-level languages like JavaScript (HTML pages) or VBA (Microsoft Word). Additionally, low-level languages like C are also supported. The string defining the system-wide name of the object and used for object creation is *PicoScope9000.COMRC*.

2 Commands Syntax

2.1 Command and Query Structure

2.1.1 Overview

The PicoScope 9300 commands consist of set commands and query commands (usually called commands and queries).

- Commands modify instrument settings or tell the instrument to perform a specific action.
- Queries cause the instrument to return data and information about its status.

Most commands have both a set form and a query form. The query form of the command differs from the set form by the addition of a question mark at the end. For example, the set command

```
ACQuire:Ch1:MODE
```

has a query form

```
ACQuire:Ch1:MODE?
```

Not all commands have both a set and a query form. Some may have set only and some have query only.

2.1.2 Messages

A command message is a command or query name followed by any information the instrument needs to execute the command or query. Command messages may contain five element types, as defined in the following table.

Symbol	Meaning
<Header>	This is the basic command name. If the header ends with a question mark, the command is a query. If the command is concatenated with other commands, the header must begin with a colon (:).
<Mnemonic>	This is the header of the sub-function. Some command headers have only one mnemonic. If a command header has multiple mnemonics, a colon (:) character always separates items from one another.
<Argument>	This is a quantity, quality, restriction or limit associated with the header. Some commands have no arguments while others have multiple arguments. A space separates arguments from the header. A comma separates arguments from one another.
<Comma>	A single comma is used between the arguments of multiple-argument commands. Optionally, there may be white space characters before and after the comma.
<Space>	A white space character is used between a command header and its argument. Optionally, a white space may consist of multiple white space characters.

Command message elements

2.1.3 Commands

Commands cause the instrument to perform a specific function or change one of its settings. Commands have this structure:

```
[ : ]<Header> [ <Space><Argument> [ <Comma><Argument> ] . . . ]
```

A command header consists of one or more mnemonics arranged in a hierarchy or tree structure. The first mnemonic is the base or root of the tree and each subsequent mnemonic is a level or branch off the previous one. Commands at a higher level in the tree may affect those at a lower level. The leading colon (:) always returns you to the base of the command tree.

2.1.4 Queries

Queries cause the instrument to return information about its status or settings. Queries have the structure:

- [:]<Header>?
- [:]<Header>? [<Space><Argument> [<Comma><Argument>] . . .]

You can specify a query command at any level within the command tree unless otherwise noted. These branch queries return information about all the mnemonics below the specified branch or level. For example

```
HISTogram:STATistics:STDdev?
```

returns the standard deviation of the histogram, whereas

```
HISTogram:STATistics?
```

returns all the histogram statistics, and

```
HISTogram?
```

returns all the histogram parameters.

2.1.5 Headers

You can control whether the instrument returns headers as part of the query response. Use the `HEADer` command to control this feature. If header is on, the query response returns command headers and formats itself as a valid set command. When the header is off, the response includes only the values. This may make it easier to parse and extract the information from the response. The table below shows the difference in responses.

Query	Header Off	Header On
Ch1:Scale?	200 mV/div	CH1:SCALE 200 mV/div
Acq:Ch1:RecLen?	512	ACQ:CH1:RECLen 512

Comparison of Header Off and Header On responses

2.2 Command Entry

2.2.1 Rules

The following rules apply when entering commands:

- A mnemonic can be followed by any letters for easier understanding of the program's text. For example, these commands are all equivalent:

```
Ch1:ATTEN:DIMENS Volt
```

```
Ch1:ATTENuator:DIMENSION Volt
```

```
Ch1:ATTENblabla:DIMENSblabla Volt
```

However, arguments must not be followed by additional characters.

- You can enter commands in upper or lower case.
- You can precede any command with white space characters. White space characters include any combination of the ASCII control characters 00 to 09, and 0B to 20 hexadecimal (0 to 9, and 11 to 32 decimal).
- The instrument will ignore commands consisting of any combination of white space characters and line feeds.

2.2.2 Concatenation

You can concatenate any combination of set commands and queries by using a semicolon (;). The instrument executes concatenated commands in the order received. The following rules apply when concatenating commands and queries:

- You can separate completely different headers with a semicolon (;), and by adding a leading colon (:) at the beginning of all commands except the first one. For example

```
TRIGger:MODE FREE
ACQuire:NUMAVg 10
```

can be concatenated into the single command

```
TRIGger:MODE FREE;:ACQuire:NUMAVg 10
```

- If concatenated commands have headers that differ by only the last mnemonic, you can abbreviate the second command and eliminate the leading colon. For example, you can concatenate the commands

```
ACQuire:Ch1:MODE ENVMINMAX
ACQuire:Ch1:NAVG 10
```

into a single command

```
ACQuire:Ch1:MODE ENVMINMAX; NAVG 10
```

The longer version also works equally well:

```
ACQuire:CH1:MODE ENVMINMAX;:ACQuire:Ch1:NAVG 10
```

- Set commands and queries may be concatenated in the same message. For example:

```
ACQuire:CH1:MODE AVGSTAB;NAVG?
```

This is a valid message that sets the acquisition mode to Stable Averaging. The message then queries the number of acquisitions for averaging. Concatenated commands and queries are executed in the order received.

- Here are some invalid concatenations:

```
DISPlay:STYLE DOTS;ACQuire:NAVG 10  
(a colon is needed before ACQuire)
```

```
DISPlay:STYLE DOTS;:FORMAT YT  
(there is an extra colon before FORMAT. Use DISPlay:STYLE DOTS;FORMAT  
YT instead.)
```

```
Acq:Ch1:Mode Sample;Ch1:RecLen 1024  
(The levels of these mnemonics are different. Either remove the second use of  
Ch1: or place :Acq: in front of Ch1:.)
```

3 Command Classification

Most commands belong to one of a few types. For example, execution-type commands tell the instrument to perform a specific action, selector-type commands modify a specific instrument setting to the one of few fixed values, and so on. All commands of a given type have similar behavior.

3.1 Execution-type commands

Execution-type commands tell the instrument to perform a specific action. For example:

```
*Run
*ClrDispl
```

There are no arguments for these commands.

All execution-type commands have a *set* form only, with no *query* form.

3.2 On/off-type commands

On/off type commands tell the instrument to turn on or turn off a specific function. For example:

```
Header Off
Ch1:Display 0
```

There are four fixed arguments possible in these commands: *On*, *Off*, *0*, *1*. Arguments *On* and *1* are equivalent and turn on the corresponding function. Arguments *Off* and *0* are also equivalent and turn off the corresponding function.

All on/off type commands have a query form, which will return one of two fixed values: *ON* or *OFF*. It is also possible to use the query form with an argument. For example:

```
Ch1:Display? 0
```

This command turns off the graphic of Channel 1 and returns *OFF*.

3.3 On/off-group-type commands

Some functions of the instrument have items that may be set on or off independently. It is also possible for the items to be either all on or all off. An example of this type of command is:

```
Meas:Ch1:XParam
```

This command has a set of parameters for automatic X-axis measurements for Ch1. It is possible to select up to 10 parameters from a list of 18:

```
Period, Freq, PosWidth, NegWidth, Rise, Fall, PosDuty,
NegDuty, PosCross, NegCross, BurstWidth, Cycles,
TimeOfMax, TimeOfMin, PosJitterPp, PosJitterRMS,
NegJitterPp, NegJitterRMS
```

There are between 2 and 64 custom items in the on/off-group-type commands. The full set of items for each command is specified in the list of commands.

The on/off-group-type commands can be used in several modes. Every such command can be used in every mode.

Single-item mode

Single-item mode is used to control one item of a command without changing its other items. In this case the item's mnemonic is added to the end of the command after a colon (:). This must be followed by a space character and then one of the following arguments: On, Off, 0, 1. For example, this command turns on a frequency measurement for Channel 1:

```
Meas:Ch1:XParam:Freq 1
```

Single-item mode has a query form similar to the On/off commands. So the query

```
Meas:Ch1:XParam:Period 1
```

or

```
Meas:Ch1:XParam:Freq?
```

returns either ON or OFF.

Group-on mode

Group-on mode is used to simultaneously turn on a custom group of items. In this case the `:Include` mnemonic is added to the end of the command. This is then followed by a space and a few items separated by commas. For example, this command turns on the rise time and fall time measurements for Channel 1:

```
Meas:Ch1:XParam:Include Rise,Fall
```

Group-off mode

Group-off mode is used to simultaneously turn off a custom group of items. In this case the `:Exclude` mnemonic is added to the end of the command. This is then followed by a space and a few items separated by commas. For example, this command turns off the frequency and period measurements for Channel 1:

```
Meas:Ch1:XParam:Exclude Freq,Period
```

All-off mode

All-off mode is used for simultaneously turning off all items. In this case the `:ClearAll` mnemonic is added to the end of the command. For example, the next command turns off all measurements for Channel 1:

```
Meas:Ch1:XParam:ClearAll
```

Group-on, Group-off and All-off modes do not have a query form.

Group-query mode

Group-query mode is used find out which items are currently turned on. This mode only has a query form. For example:

```
Meas:Ch1:XParam?
```

The answer may include one or more items separated by a comma, or ClearAll if all items are turned off. For example, the answer `Freq,Period` means there are two items turned on.

3.4 Selector-type commands

The selector-type commands modify a specific instrument setting to one of a few fixed values. For example

```
Trig:Source
```

has these possible arguments:

```
Direct, ExtHF, IntClock
```

and

```
Trig:Mode
```

has these possible arguments:

```
Free, Trig
```

Between 2 and 32 custom arguments are available for these commands. The full set of arguments for each command is specified in the list of commands.

The selector-type commands have a query form. It is possible to use the query form with an argument. For example:

```
Trig:Source? Direct
```

This command sets the Direct input as the trigger source and returns DIRECT.

3.5 Integer-type commands

The integer-type commands modify specific integer-value functions. For example, the command

```
Acq:Ch1:RecLen 1024
```

sets the length of Channel 1 signals to 1024 points. The valid range and increment of each value is different and is described in the list of commands.

The integer-type commands have a query form. It is possible to use the query form with an argument. For example

```
Acq:Ch1:RecLen? 24
```

returns 32, since 32 is the minimum valid length of a signal.

3.6 Float-type commands

The float-type commands modify specific real-value functions. For example, the command

```
Ch1: Scale 0.1
```

sets the Y-scale for Channel 1 to 100 mV/div. The valid range and increment of each value is different and is described in the list of commands.

Float-type commands have a query form. It is also possible to use the query form with an argument. For example

```
Ch1:Scale? 0.1
```

returns 100 mV/div, when V/div is a dimension of the scale, and the prefix m is milli.

The commands

```
TB:ScaleA? 0.0000001
```

```
TB:ScaleA? 100e-9
```

```
TB:ScaleA? 0.1u
```

```
TB:ScaleA? 100p
```

are equivalent and set the Scale A of the timebase to the value 100 ns/div. All of these commands return 100 ns/div.

3.7 Data-type commands

The data-type commands are used to send data to the instrument or to receive data from the instrument, such as the array of points from an acquired signal, the result of a measurement, and so on.

Some data-type commands only have a query form, while others have both a command and a query form. The structure of the data is different for each command and is specified in the list of commands.

4 Full list of commands

4.1 Header commands

Header: Header

Type: On/Off

Action: Enables/disables headers as part of the query response

4.2 GUI commands

GUI control command

Header: Gui (SW ver. before 3.20.12)

Header: Gui:Control (SW ver. 3.20.12 or newest)

Type: Selector

Arguments: RemoteLocal, RemoteOnly, Invisible

Action: Set the behavior of the GUI when it is controlled as a COM object.

GUI side menu command

Header: Gui:SideMenu:Left:Menu
Gui:SideMenu:Right:Menu

Type: Selector

Arguments: Off, Ch, Acq, Trig, Displ, Save, Mark, Meas, Math,
Hist, Eye, Mask, Aux, TDR, Module,
Util

Action: Remove or Set the specified side menu panel.

GUI side menu page command

Header: Gui:SideMenu:Left:Page
Gui:SideMenu:Right:Page

Type: Integer

Argument: 1 to N, when N is count of pages in the current side menu

Action: Select the page in the specified side menu panel.

Note: This command applies to the side menu with two or more pages.

GUI side menu signal command

Header: Gui:SideMenu:Left:Signal
Gui:SideMenu:Right:Signal

Type: Integer

Argument: 1 to N, when N is count of active signals (max. 4)

Action: Select the signal in the specified side menu panel.

Note: This command applies to the Channels, Save/Recall and Math menus.

4.3 System commands

Clear Display

Header: *ClrDispl

Type: Execution

Action: Clears the display immediately

Running Control

Header: *RunControl

Type: Selector

Arguments: Stop, Single, Run

Action: Run - Start a continuous acquisition
 Single - Start a single acquisition
 Stop - Immediately stop the acquisition

Response: Run - the instrument is in the continuous acquisition state
 Single - the instrument is in the single acquisition state
 Stop - the instrument is stopped

Start Autoscaling

Header: *Autoscale

Type: Selector

Arguments: Auto, SingleVal, NRZ, RZ

Action: Sets the type of signal and starts autoscaling the instrument

Response: Selected type of signals

Recall Default Setup

Header: *DefSetup

Type: Execution

Action: Restores the instrument to its default setup

Set Copy Mode and Copy to the Clipboard

Header: *Copy: <Mode>

when <Mode> is one of:

FullScreen	FullWindow
ClientPart	InvClientPart
ScopeScreen	InvScopeScr

Type: Executing-type command

Action: Sets the specified copy mode (All display, software window, client part of the software window, client part of the software window with colors inverted, software screen area or software screen area with color inversion) and copy specified onto the clipboard.

Copy to the Clipboard

Header: *Copy

Type: Execution

Action: Puts the image onto the clipboard, depending on the Copy Mode

Get Copy Mode query

Header: *Copy?

Argument: None

Forms: Query only

Action: Returns current Copy Mode. See [Set Copy Mode and Copy to the Clipboard](#).

4.4 Channels commands

Type of the Signals

Header: AllChs:FitAcqTo

Type: Selector

Arguments: SingleValued, MultiValued

Action: Prepares the instruments for best acquisition of single-valued or multivalued signals

Digital Feedback

Header: AllChs:DigitalFB

Type On/off

Action: Turns digital feedback on or off

Best Flatness

Header: AllChs:BestFlat

Type On/off

Action: Turns digital correction of low-frequency distortion on or off

Display a Channel

Header: Ch1:Display Ch2:Display
Ch3:Display Ch4:Display

Type: On/off

Action: Turns display of the corresponding channel's signal on or off

Acquire a Channel

Header: Ch1:AcqOnlyEn Ch2:AcqOnlyEn
Ch3:AcqOnlyEn Ch4:AcqOnlyEn

Type: On/off

Action: On - acquisition of the channel is independent of whether it is displayed or not
Off - acquisition of the channel occurs only when the channel display is On

Scale a Channel

Header: Ch1:Scale Ch2:Scale
Ch3:Scale Ch4:Scale

Type: Float

Argument: 0.002 to 0.5, or other when attenuator is used

Action: Sets the specified display scale in V/div

Scale a TDR Channel

Header: Ch1:RhoScale Ch2:RhoScale
Ch3:RhoScale Ch4:RhoScale (in Rho/div)

Header: Ch1:OhmScale Ch2:OhmScale
Ch3:OhmScale Ch4:OhmScale (in Ohm/div)

Type: Float

Argument: 0.01 to 2 (Rho scales)
1 to 100 (Ohm scales)

Action: set the specified vertical display scale for TDR non-voltage scales

Scale a TDT Channel

Header: Ch1:GainScale Ch2:GainScale
Ch3:GainScale Ch4:GainScale (in 1/div)

Header: Ch1:GainDBScale Ch2:GainDBScale
Ch3:GainDBScale Ch4:GainDBScale (in dB/div)

Type: Float

Argument: 0.01 to 100 (Gain scales)
1 to 20 (Gain dB scales)

Action: set the specified display scale for TDT non-voltage scales

Offset a Channel

Header: Ch1:Offset Ch2:Offset
Ch3:Offset Ch4:Offset

Type: Float

Argument: -1 to +1, or other when attenuator is used

Action: Sets the specified compensation voltage of the channel in V

Offset a TDR Channel

Header: Ch1:RhoOffset Ch2:RhoOffset
Ch3:RhoOffset Ch4:RhoOffset (in Rho)

Header: Ch1:OhmOffset Ch2:OhmOffset
Ch3:OhmOffset Ch4:OhmOffset (in Ohm)

Type: Float

Argument: -8 to 8 (Rho units)
-400 to 400 (Ohm units)

Action: set the specified compensation value for TDR non-voltage scales.

4.4.11 Offset a TDT Channel

Header: Ch1:GainOffset Ch2:GainOffset
Ch3:GainOffset Ch4:GainOffset (in 1)

Header: Ch1:GainDBOffset Ch2:GainDBOffset
Ch3:GainDBOffset Ch4:GainDBOffset (in dB)

Type: Float

Argument: -400 to 400 (Gain units)
-80 to 80 (Gain dB units)

Action: set the specified compensation value for TDT non-voltage scales.

Position a Channel

Header: Ch1:Position Ch2:Position
Ch3:Position Ch4:Position

Type: Float

Argument: -5 to +5

Action: Sets the specified vertical position of the channel on the screen, in divisions.

Bandwidth of Channel

Header: Ch1:Band Ch2:Band
Ch3:Band Ch4:Band

Type: Selector

Arguments: Full, Narrow

Action: Sets the bandwidth of the channel

Deskew of Channel

Header: Ch1:Deskew Ch2:Deskew
Ch3:Deskew Ch4:Deskew

Type: Float

Argument: 0 to 100e-9

Action: Sets the deskew of the channel in s

Attenuator linear/log

Header: Ch1:Atten:Unit Ch2:Atten:Unit
Ch3:Atten:Unit Ch4:Atten:Unit

Type: Selector

Arguments: Off, Ratio, DB

Action: Sets the presence and scale of the attenuator or converter used with the channel

Attenuator ratio

Header: Ch1:Atten:Ratio Ch2:Atten:Ratio
Ch3:Atten:Ratio Ch4:Atten:Ratio

Type: Float

Argument: 0.0001 to 1000000

Action: Sets the attenuation ratio. This setting is active only when the attenuator unit is *ratio*.

Attenuator dB

Header: Ch1:Atten:DB Ch2:Atten:DB
Ch3:Atten:DB Ch4:Atten:DB

Type: Float

Argument: -80 to +120

Action: Sets the attenuation in dB. This setting is only active when the attenuator unit is *decibels*.

Attenuator unit

Header: Ch1:Atten:Dimens Ch2:Atten:Dimens
Ch3:Atten:Dimens Ch4:Atten:Dimens

Type: Selector

Arguments: Volt, Watt, Ampere, Unknown

Action: Sets the units of the converter used with the channel

Sampler Delay

Header: Ch1_2:SmplDelay Ch3_4:SmplDelay

Type: Float

Argument: 0 to 2e-9

Action: Sets the sampler delay in seconds

4.5 Timebase commands

Timebase Units

Header: TB:Units

Type: Selector

Arguments: Time, Bit

Action: Sets the units of the timebase to s/div or bit/div

Bit rate value

Header: TB:BitRate

Type: Selector

Arguments: 44M736, 51M84, 125M000, 132M813, 139M264, 155M52, 265M625, 270M000, 393M22, 466M56, 480M000, 531M25, 614M4, 622M08, 786M43, 933M12, 1G0625, 1G2288, 1G24416, 1G25000, 1G485, 1G5000, 1G5729, 1G86625, 2G1231, 2G125, 2G48832, 2G5000, 2G66606, 2G8576, 2G97000, 3G000, 3G072, 3G125, 3G187, 3G25000, 3G32000, 4G25000, 5G000, 6G000, 6G25000, 6G375, 8G5000, 9G95328, 10G3125, 10G51875, 10G6642, 10G709, 11G000, 11G0957, 11G317, 12G2495, 17G000, 19G90656, 25G78125, 27G73193, 39G81312, 42G65691, 43G01841

Action: Sets one of the standard bit rate values from 44.736 Mbit/s to 43.01841 Gbit/s

Sampling Mode

Header: TB:SampleMode

Type: Selector

Arguments: Scope, TDR, Eye, Random, RealTime, Roll

Action: Sets the instrument's sampling mode

Timebase mode

Header: TB:Mode

Type: Selector

Arguments: A, AB, B

Action: Sets main, intensified, or delayed timebase

Main timebase scale, sec/div

Header: TB:ScaleA
 Type: Float
 Argument: 10e-12 to 10
 Action: Sets the scale of the main timebase when time units are used

Delayed timebase scale, sec/div

Header: TB:ScaleB
 Type: Float
 Argument: 10e-12 to 10
 Action: Sets the scale of delayed timebase when time units are used

Main timebase scale, bit/div

Header: TB:BitScaleA
 Type: Float
 Argument: 0.0005 to 100000, depending on actual bit rate
 Action: Sets the scale of the main timebase when bit units are used

Delayed timebase scale, bit/div

Header: TB:BitScaleB
 Type: Float
 Argument: 0.0005 to 100000, depending on actual bit rate
 Action: Sets the scale of the delayed timebase when bit units are used

Main Distances Scale for TDR-TDT

Header: TB:ScaleMetreA
 TB:ScaleFootA
 TB:ScaleInchA
 Type: Float
 Argument: depends on actual TDR/TDT dielectric constants or velocity
 Action: set scale of main timebase when corresponding units are used

Delayed Distances Scale for TDR-TDT

Header: TB:ScaleMetreB
 TB:ScaleFootB
 TB:ScaleInchB
 Type: Float

Argument: depends on actual TDR/TDT dielectric constants or velocity

Action: set scale of delayed timebase when corresponding units are used

Timebase delay, s

Header: TB:Delay

Type: Float

Argument: 0 to max, where max depends on main and delayed timebase

Action: Sets the delay of intensified, delayed timebase when time units are used

Timebase delay, bits

Header: TB:Delay

Type: Float

Argument: 0 to max, where max depends on main and delayed timebase

Action: Sets the delay of intensified, delayed timebase when bit units are used

Distances Delay for TDR-TDT

Header: TB:Delay

Type: Float-type command

Argument: 0 to max, where max depends of main and delayed timebase

Action: set delay of intensified, delayed timebase when distances units are used

Real Time Delay

Header: TB:RTDelay

Type: Float

Argument: 0 to 8

Action: Sets the delay when Random or RealTime sampling modes are used

Real Time Trigger Position

Header: TB:TrigPos

Type: Float

Argument: 0 to 100

Action: Sets the trigger position when Random or RealTime sampling modes are used in %

4.6 Trigger commands

4.6.1 Trigger Source

Trigger Source

Header: Trig:Source

Type: Selector

Arguments: ExtDirect, ExtPrescaler, Ch1Direct, Ch2Direct, IntClock, Auxiliary

4.6.2 External Direct Trigger Commands

External Direct Trigger Level

Header: Trig:ExtDir:Level

Type: Float

Argument: -1 to +1

Action: Sets the trigger level for external direct input, volts

External Direct Trigger Slope

Header: Trig:ExtDir:Slope

Type: Selector

Arguments: Pos, Neg

Action: Sets the Positive or Negative slope of external direct trigger

External Direct Trigger Hysteresis

Header: Trig:ExtDir:Hyst

Type: Selector

Arguments: Norm, HighSens

Action: Sets the hysteresis for external direct trigger (Norm) or set off (HighSens)

4.6.3 External Prescale Trigger Commands

Automatic external prescaler

Header: Trig:ExtPresc:AutoNDiv

Type: On/off

Action: Turn the automatic mode of the ext. prescaler on or off

External prescaler division factor

Header: Trig:ExtPresc:NDiv

Type: Selector

Arguments: 1 to 8

Forms: Must use the *set with query* or *query* forms only!

Action: Sets the external prescaler division factor. This query is necessary as the real factor may differ from the arguments.

4.6.4 Ch1/Ch2 Direct Trigger Commands

Ch1/Ch2 Direct Trigger Level

Header: Trig:Ch1:Level Trig:Ch2:Level

Type: Float

Argument: -1 to +1

Action: Sets the trigger level in volts for Ch1/Ch2 input

Ch1/Ch2 Direct Trigger Slope

Header: Trig:Ch1:Slope Trig:Ch2:Slope

Type: Selector

Arguments: Pos, Neg

Action: Sets the Positive or Negative slope of Ch1/Ch2 input

Ch1/Ch2 Direct Trigger Hysteresis

Header: Trig:Ch1:Hyst Trig:Ch2:Hyst

Type: Selector

Arguments: Norm, HighSens

Action: Sets the hysteresis for Ch1/Ch2 trigger input (Norm) or set off (HighSens)

Ch1/Ch2 Direct Trigger Coupling

Header: Trig:Ch1:Coupling Trig:Ch2:Coupling

Type: Selector

Arguments: DC, AC

Action: Sets the coupling for Ch1/Ch2 trigger input

Ch1/Ch2 Direct Trigger Rejection

Header: Trig:Ch1:Reject Trig:Ch2:Reject

Type: Selector

Arguments: Off, LF, HF

Action: Sets the rejection mode for Ch1/Ch2 trigger input

4.6.5 Trigger Period for Internal Clock Sources

Trigger Period for Internal Clock Sources

Header: Trig:IntRate

Type: Float

Argument: 8e-9 to 0.06

Action: Sets the period for the internal clock trigger source in seconds

4.6.6 Trigger Mode

Trigger Mode

Header: Trig:Mode

Type: Selector

Arguments: Free, Trig

Action: Sets Freerun or Triggered mode for the trigger

4.6.7 Trigger Holdoff Commands

Holdoff Mode

Header: Trig:HoldoffBy

Type: Selector

Arguments: Time, Events

Action: Sets the Holdoff mode by time or by events

Holdoff by Time

Header: Trig:HoldoffTime

Type: Float

Argument: 1e-6 to 17

Action: Sets the holdoff time in seconds

Holdoff by Events

Header: Trig:HoldoffEvents
Type: Integer
Argument: 50 to 50000000, depends on actual trigger frequency
Forms: Must use the *set with query* or *query* forms only
Action: Sets the holdoff by events. This query is necessary as the real number of events may differ from the arguments.

4.6.8 Attenuator Commands

Attenuator Unit for External Direct Input

Header: Trig:Atten:Unit
Type: Selector
Arguments: Off, Ratio, DB
Action: Sets the presence and unit of the attenuator or converter used with direct trigger input

External Direct Input Attenuation (ratio)

Header: Trig:Atten:Ratio
Type: Float
Argument: 0.0001 to 1000000
Action: Sets the attenuation ratio. This setting is active only when the attenuator unit is ratio.

External Direct Input Attenuation (dB)

Header: Trig:Atten:DB
Type: Float
Argument: -80 to +120
Action: Sets the attenuation in dB. This setting is active only when the attenuator unit is decibels.

4.6.9 Pattern Sync Trigger Commands

Pattern Lock Mode

Header: Trig:PatternLockMode
Type: Selector
Arguments: Off, AutoDetect, Manual
Action: Sets the pattern lock mode

Pattern Length

Header: Trig:PattLen
Type: Integer
Arguments: 7 to 8388608
Action: Sets the pattern length

Start Bit

Header: Trig:PattDelay
Type: Integer
Arguments: 0 to 126
Action: Sets the pattern delay in bits

Eye Line Mode

Header: Trig:PattEyeLine
Type: On/off
Action: Turn the eye line mode on or off

Scan Bits

Header: Trig:PattScanBits
Type: Integer-type command
Arguments: 1 to 127
Action: Sets the amount of scanning bits in the eye line mode

4.7 Acquisition commands

4.7.1 Sampling Mode

Sampling Mode

Header: Acq:Sampl

Type: Selector

Arguments: Simult, Altern

Action: Simult - sets simultaneous acquisition of all channels
 Altern - sets alternate acquisitions of all channels

4.7.2 Channel Acquisition Commands

Acquisition Mode of Channel

Header: Acq:Ch1:Mode Acq:Ch2:Mode
 Acq:Ch3:Mode Acq:Ch4:Mode

Type: Selector

Arguments: Sample, AvgStab, AvgMult, EnvMinMax, EnvMax, EnvMin

Action: Sets the acquisition mode of the specified channel

Channel Averaging

Header: Acq:Ch1:NAvg Acq:Ch2:NAvg
 Acq:Ch3:NAvg Acq:Ch4:NAvg

Type: Integer

Argument: 1, 2, 4, 8, 16, ... 4096

Action: Sets the averaging coefficient for the specified channel

Channel Envelope

Header: Acq:Ch1:NEnv Acq:Ch2:NEnv
 Acq:Ch3:NEnv Acq:Ch4:NEnv

Type: Integer

Argument: 1, 2, 4, 8, 16, ... , 4096, 8192

Action: Sets the number of signals for envelope mode for the specified channel. Argument 8192 is used for unlimited number of signals.

Channel Record Length

Header: Acq:Ch1:RecLen Acq:Ch2:RecLen
 Acq:Ch3:RecLen Acq:Ch4:RecLen

Type: Integer

Argument: 32, 64, 128, ... , 432768

Action: Sets the number of points for specified channel

4.7.3 Termination of the Acquisition

Termination of Acquisition

Header: Acq:RunUntil

Type: Selector

Arguments: StopBtn, NAcq

Action: Sets the condition for terminating acquisition when the Stop button is pressed or after the specified number of waveforms is reached.

4.7.4 Number of Waveforms

Number of Waveforms

Header: Acq:NAcq

Type: Integer

Argument: 1 to 65535

Action: Sets the number of signals for the terminating acquisition

4.7.5 Action when Number of Waveforms reached

Action when Number of Waveforms reached

Header: Acq:Action

Type: On/off-group

Items: Beep, Save

Action: If Save is turned on, every signal is stored to disk
 If Beep is turned on, the beep signal will sound after the specified number of waveforms is reached

4.7.6 File Name

File Name

Header: Acq:FileName

Type: Data

Argument: Text string contains the file path

Action: Defines the full path and base file name for storing the acquired signals onto the Disk. The name of each saved file consists of a base name, followed by an underline (_) and five-digit auto-incremented numbers. For example:

After the command:

```
Acq:FileName C:\Temp\Test1\basename
```

```
Files basename_00001.wfm, basename_00002.wfm,  
basename_00003.wfm and so on will be written to the C:\Temp  
\Test1 folder.
```

Note: The specified folder must exist

4.7.7 Stored Files Format

Stored Files Format

Header: Acq:FileFormat

Type: Selector

Arguments: Binary, Verbose, YOnly

Action: Sets the format of the file

4.8 Display commands

Mnemonic <src> in some Display Commands signifies Source
(<src> is: Ch1, Ch1B2, Ch2, Ch2B2, F1, F2, F3, F4, M1, M2, M3, M4, S1, S2)

Trace mode

Header: Displ:TraceMode

Type: Selector

Arguments: AllLocked, PerTrace

Action: In PerTrace mode, every waveform may be displayed in its own style
In AllLocked mode, the display style of all waveforms is set as the style of the active trace

Select active trace

Header: Displ:TraceSel

Type: Selector

Arguments: Ch1, Ch1B2, Ch2, Ch2B2, F1, F2, F3, F4, M1, M2, M3, M4, XY

Action: Selects the active trace for AllLocked trace mode

Display Persistence (SW v.3.20 or newer)

Header: Displ:<src>:InkStyle

Type: Selector

Arguments: Dots, VarPersist, InfinPers, VarGrayScal, InfGrayScal, VColorGrad, IColorGrad

Action: set display persistence for specified trace in PerTrace mode;
set display persistence for all traces in AllLocked mode.

Note use argument Dots for disable persistence

Display Style (SW v.3.20 or newer)

Header: Displ:<src>:ChartStyle

Type: Selector

Arguments: Dots, Vectors

Action: set display style for specified trace in PerTrace mode;
set display style for all traces in AllLocked mode

Set Display Style (SW v.3.19 or older)

Header: Displ:<src>:Style

Type: Selector

Arguments: Dots, Vectors, VarPersist, InfinPers, VarGrayScal, InfGrayScal, VColorGrad, IColorGrad

Action: Sets the display style for specified trace in PerTrace mode
Sets the display style for all traces in AllLocked mode

Persistence Time , seconds (for VarPersist Style)

Header: Displ:<src>:PersistTime

Type: Float

Argument: 0.1 to 20

Action: Sets the persistence time for specified trace in PerTrace mode
Sets the persistence time for all traces in AllLocked mode

Refresh Time, seconds (for VarGrayScal or VColorGrade Styles)

Header: Displ:<src>:RefreshTime

Type: Float

Argument: 1 to 200

Action: Sets the refresh time for specified trace in PerTrace mode
Sets the refresh time for all traces in AllLocked mode

Reset Display Style

Header: Displ:ResetAll

Type: Execution

Action: Resets Display Styles to initial state (variable persistence 2 c)

Display Format

Header: Displ:Format

Type: Selector

Arguments: Auto, YT, 2YT, 4YT, XY, CombYTXY, Comb2YTXY

Action: Selects the number and kinds of screens

Define Trace Screen (for 4YT Format)

Header: Displ:Screen4:<trace>,
when <trace> is <src> or Hist

Type: Selector

Arguments: 1, 2, 3, 4

Action: Moves the specified trace onto the specified screen in 4YT format

Define Trace Screen (for 2YT, Comb2YTXY Formats)

Header: Displ:Screen2:<trace>,
 when <trace> is <src> or Hist

Type: Selector

Arguments: 1, 2

Action: Moves the specified trace onto the specified screen in 2YT or
 Comb2YTXY formats

Source of X Axis for XY Screen

Header: Displ:XAxis

Type: Selector

Arguments: <src>, exclude XY, DB

Action: Sets the specified signal as X axis for XY screen

Source of Y Axis for XY Screen

Header: Displ:YAxis

Type: Selector

Arguments: <src>, exclude XY, DB

Action: Sets the specified signal as Y axis for XY screen

Graticule Type

Header: Displ:Gratic

Type: Selector

Arguments: Full, Frame, Axis, Off

Action: Defines the type of graticule for YT and XY screens

4.9 Save/Recall commands

4.9.1 Work with Memo Zones (M1, M2, M3, M4)

Memory Display

Header: Save:<mz>:Visible
Type: On/off-group
Items: M1, M2, M3, M4
Action: Controls the display of memory zones

Source for storing into Memory

Header: Save:Memo:Source
Type: Selector
Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4
Action: Defines the signal as source for storing into memory zone

Save into Memory

Header: Save:<mz>:Save
Type: Execution
Action: Stores the selected source into selected memory

4.9.2 Memory Scaling

Complex Format

Header: Save:<mz>:ComplexScale
Type: Selector
Arguments: Magnitude, Phase, Real, Imaginary
Action: Defines the display mode when memory contains spectrum

Vertical Scale Type

Header: Save:<mz>:VScaleType
Type: Selector
Arguments: Linear, Logarithm
Action: Defines the vertical scale type for the magnitude of the spectrum

Vertical linear Scale

Header: Save:<mz>:VertScale

Type: Float

Arguments: 1e-6 to 1e6

Action: Defines the vertical scale in volts/div for Linear vertical scale type

Vertical linear Position

Header: Save:<mz>:VoltPosit

Type: Float

Arguments: -10 to +10

Action: Defines the vertical position in div for Linear vertical scale type

Vertical logarithmic Scale

Header: Save:<mz>:VertDBScale

Type: Float

Arguments: 1 to 120

Action: Defines the vertical scale in dB/div for Logarithm vertical scale type

Vertical logarithmic Position

Header: Save:<mz>:Posit0DB

Type: Float

Arguments: -10 to +10

Action: Defines the vertical position in div for Logarithm vertical scale type

Vertical Phase Scale

Header: Save:<mz>:PhaseScale

Type: Float

Arguments: 0.125 to 8

Action: Defines the vertical scale in rad/div for Phase display mode

Vertical Phase Position

Header: Save:<mz>:PhasePosit
Type: Float
Arguments: -10 to +10
Action: Defines the vertical position in div for Phase display mode

Horizontal Scale

Header: Save:<mz>:HorScale
Type: Float
Arguments: Depends on actual timebase
Action: Defines the horizontal scale in actual X-axis unit/div

Horizontal Position

Header: Save:<mz>:HorPosition
Type: Float
Arguments: Depends on actual timebase
Action: Defines the horizontal position in actual X-axis unit

4.9.3 Work with Disk

File Type

Header: Save:Disk:FileType
Type: Selector
Arguments: Wfm, DB
Action: Defines the file type for saving

Source for saving to file

Header: Save:Disk:Source
Type: Selector
Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4
Action: Defines the signal as source for saving to file

File Name

Header: Save:Disk:FileName
Type: Data
Argument: Text string
Forms: Command, query, command with query
Action: Defines the file name for saving the specified signal to disk
Note: Specified folder must exist

File Name Mode

Header: Save:Disk:NameMode
Type: Selector
Arguments: Manual, Auto
Action: Sets the file name mode. In Auto mode the file name consists of a base name followed by an underscore (_) and a five-digit number. Each time you save a waveform, the number in the file name is automatically incremented. For example: `basename_00001.wfm`, `basename_00002.wfm`, `basename_00003.wfm`, and so on.

Format of stored files

Header: Save:Disk:FileFormat
Type: Selector
Arguments: Binary, Verbose, YOnly
Action: Sets the file format

Save to Disk

Header: Save:Disk:ExecSave
Type: Executing
Action: Saves the selected source to previously specified file

Load from Disk

Header: Save:<mz>:LoadFromDsk
Type: Executing
Action: Loads the previously specified disk file into the specified Memory Zone

4.9.4 Work with Setups

Recall Factory Setup

Header: Save: Setup: RecFact

Type: Execution

Action: Returns the instrument to manufacturer's default setting

Recall Default Setup

Header: Save: Setup: RecDefault

Type: Executing

Action: Returns the instrument to its default setting

Recall Power-Off Setup

Header: Save: Setup: RecLast

Type: Execution

Action: Returns the instrument to the last setting before the power supply was last switched off

Save Setup as Default

Header: Save: Setup: SvAsDefault

Type: Execution

Action: Stores the present front-panel setup as the default setup

Name of Custom Setup File

Header: Save: Setup: FileName

Type: Data

Argument: Text string containing file path

Action: Defines the file name for storing Custom Setup

Note: The specified folder must exist

Save Custom Setup

Header: Save: Setup: Save

Type: Execution

Action: Stores the present front-panel setup as previously specified custom setup

Recall Custom Setup

Header: Save:Setup:Recall

Type: Execution

Action: Recalls the setup previously saved to file. The name of the setup must first be defined by the command Save:Setup:FileName.

4.10 Markers commands

Marker Type

Header: Mark:Type
Type: Selector
Arguments: Off, MX, MY, XY
Action: Sets the marker type

Marker Sources

Header: Mark:M1:Source Mark:M2:Source
Type: Selector
Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4
Action: Attaches the specified marker to the specified signal

X position of Marker

Header: Mark:M1:XPos Mark:M2:XPos
Type: Float
Argument: Real value of X-axis
Action: Sets the X position of the specified marker

Y position of Marker

Header: Mark:M1:YPos, Mark:M2:YPos
Type: Float
Argument: Real value of Y-axis
Action: Sets the Y position of the specified marker

Y value of XY-Marker query

Header: Mark:M1:YVal? Mark:M2:YVal?
Type: Float
Argument: none
Forms: query only
Action: return value of the signal in the XY-marker point.
Note: return zero when marker type is Off, MX;
return Y position of marker when marker type is MY

Motion of Markers

Header: Mark:Motion

Type: Selector

Arguments: Independ, Paired

Action: When Paired motion is selected, you can move both markers with the M1 POSITION variable simultaneously, while the difference between markers can be moved with the M2 POSITION variable

4.11 Measure commands

4.11.1 Common Measures commands

The mnemonic <src> in some Measure Commands signifies the Source (<src> is: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4)

Measurement Type

Header: Meas:Display

Type: Selector

Arguments: Off, Param, Statistic

Action: Sets the measurement type

Measurement Source

Header: Meas:DisplSrc

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4

Action: Sets the source for the measurement

Measurement Mode

Header: Meas:Mode

Type: Selector

Arguments: Permanent, Single

Action: Sets the measurement mode

Execute Single Measurement

Header: Meas:SingleMeas

Type: Execution

Action: Executes a single measurement in Single mode

4.11.2 Statistic Commands

Statistic Measurement Mode

Header: Meas:StatMode

Type: Selector

Arguments: Permanent, Window, Weight

Action: Sets the Statistic Measurement mode

Windows Value

Header: Meas:Window

Type: Integer

Argument: 8 to 8192

Action: Sets the number of recently acquired waveforms for Window mode of Statistic Measurement

Weight Value

Header: Meas:Weight

Type: Integer

Argument: 8 to 8192

Action: Sets the weight variable for Weight mode of Statistic Measurement

Clear Statistics for all Measures

Header: Meas:ClrStatistics

Type: Executing-type command

Action: Clear statistics of all measurements for all signals

4.11.3 Define parameter Commands

Viewing of Define Parameters

Header: Meas:View

Type: On/off-type command

Action: Sets the visibility of *define parameters* markers for selected sources

Top/Base Definition Method

Header: Meas:<src>:Method

Type: Selector

Arguments: Hist, MinMax, Marker

Action: Sets the Top and Base vertical reference thresholds for amplitude measurements of specified signals

Top Value for Marker Method

Header: Meas:<src>:Top

Type: Integer

Argument: 257 to 1023

Action: Sets the Top vertical reference threshold for specified signals. Argument 0 corresponds to the bottom of the screen, and argument 1023 corresponds to the top of the screen independently of the real screen's height.

Base Value (for Marker Method)

Header: Meas:<src>:Base

Type: Integer

Argument: 1 to 767

Action: Sets the Base vertical reference threshold for specified signals. Argument 0 corresponds to the bottom of the screen, and argument 1023 corresponds to the top of the screen independent of the real screen's height.

Threshold Definition Method

Header: Meas:<src>:Thresh

Type: Selector

Arguments: 10-90, 20-80, Custom

Action: Sets the lower, middle, and upper thresholds for measurements of the specified signals. May be set to the fixed values 10%-50%-90%; 20%-50%-80%; or custom values.

Threshold Units

Header: Meas:<src>:Unit

Type: Selector

Arguments: Percent, Volt, Division

Action: Sets the units of thresholds for the specified signals. Used for custom threshold definition method only.

Position of Upper, Middle or Lower Threshold

Headers: Meas:<src>:UpThresh
 Meas:<src>:MidThresh
 Meas:<src>:LowThresh

Type: Float

Arguments: Absolute voltage value (for Volt threshold units only)
 -4 to +4 (for Division threshold units only)

Action: Sets the threshold position for the specified signals

Percentage of Upper, Middle or Lower Threshold

Headers: Meas:<src>:UpThPerc
 Meas:<src>:MidThPerc
 Meas:<src>:LowThPerc

Type: Integer

Arguments: -80 to +200

Action: Sets the threshold percentage for the specified signals. Used for Percent threshold units only. Argument 0 (%) corresponds to the Base of the signals, and argument 100 (%) corresponds to the Top of the signals.

Margins Definition Mode

Header: Meas:<src>:MargMode

Type: Selector

Arguments: Slope, Marker

Action: Sets the margins definition mode

Slope of Left or Right Margins

Headers: Meas:<src>:LeftSlope
 Meas:<src>:RightSlope

Type: Integer

Arguments: 0 to 127

Action: Sets the margin for the specified signals on the specified slope. Used for slope margins definition mode only. Argument 0 = the first rise, value 1 = first fall, 2 = second rise, 3 = second fall, and so on.

Thresholds of Left and Right Margin Slopes

Headers: Meas:<src>:LeftTresh
Meas:<src>:RightTresh

Type: Selector

Arguments: Upper, Middle, Lower

Action: Sets the thresholds for definitions of the left or right slope. Used for slope margins definition mode only.

Position of Left or Right Margin

Headers: Meas:<src>:LeftMarker
Meas:<src>:RightMarker

Type: Float

Arguments: Absolute time value

Action: Sets the position of margin for the specified signals. Used for marker margins definition mode only.

4.11.4 List of Measurements

List of X Measurements

Header: Meas:<src>:XParam

Type: On/off-group

Items: Period, Freq, PosWidth, NegWidth, Rise, Fall, PosDuty, NegDuty, PosCross, NegCross, BurstWidth, Cycles, TimeOfMax, TimeOfMin, PosJitterPp, PosJitterRMS, NegJitterPp, NegJitterRMS

Action: Defines the set of X-axis measurements for the specified signals

List of Y Measurements

Header: Meas:<src>:YParam

Type: On/off-group

Items: Max, Min, PP, Top, Base, Ampl, Middle, Mean, dcRMS, acRMS, Area, CycMean, CycDcRMS, CycAcRMS, CycArea, PosOver, NegOver

Action: Defines the set of Y-axis measurements for the specified signals

4.11.5 Inter-signal Measurements

Second Source for Inter-Signal Measurements

Header: Meas:Source2

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4

Action: Sets the second source for the inter-signal measurements

List of X Inter-Signal Measurements

Header: Meas:<src>:XDualPar

Type; On/off-group

Items: DellR1R, DellR1F, DellF1R, DellF1F, DellRnR,
DellRnF, DellFnR, DellFnF, PhaseDeg, PhaseRad,
PhasePerc, Gain, DBGain

4.11.6 Measurements of Spectrum Signals

Mnemonic <fft_src> in some Measurement Commands signifies Source
(<fft_src> is F1, F2, F3, F4, M1, M2, M3, M4)

4.11.6.1 Spectrum Parameter Commands

Limits Definition Method for Spectrum

Header: Meas:<src>:FFTMethod

Type: Selector

Arguments: Harmonic, Peak

Action: Sets the method of the limits definition for the specified signal. Used for spectrum signals only.

Left and Right Spectrums Margin

Headers: Meas:<src>:FFTLeft
Meas:<src>:FFTRight

Type: Float

Arguments: Absolute frequency value

Action: Sets the position of margin for the specified spectrum signals. Used for searching for peak 1 of the spectrum for the Harmonic method.

Peak Level of Spectrum

Header: Meas:<src>:PeakLevel

Type: Float

Arguments: -100 to +80 (dBV)

Action: Sets the level for the specified spectrum signals. Used for searching a peak of the spectrum for the Peak method.

Left and Right Spectrum Peaks

Headers: Meas:<src>:PeakLeft
Meas:<src>:PeakRight

Type: Integer

Arguments: 1 to 41

Action: Sets the first and second peaks for the specified spectrum signals

4.11.6.2 List of Spectrum Frequency Measurements

List of Spectrum Frequency Measurements

Header: Meas:<src>:XFFTPar

Type: On/off-group

Items: Freq, DFreq

Action: Defines the set of the frequency measurements for the specified signals

4.11.6.3 List of Spectrum Magnitude Measurements

List of Spectrum Magnitude Measurements

Header: Meas:<src>:YFFTPar

Type: On/off-group

Items: Magn, DMagn, TDH

Action: Defines the set of the magnitude measurements for the specified signals

4.11.7 Delete all Measurements for all Sources

Delete all Measurements for all Sources

Header: Meas:ClearAll

Type: Execution

Action: Clears the list of all measurements for all signals

4.11.8 Getting Measurement Results

Get List of Measured Parameters

Header: Meas:Res:List?

Type: Data

Argument: None

Forms: Query only

Action: Returns text with the list of the active measurements for all signals with ordinal index

Get Current Value of Parameter

Header: Meas:Res:<N>?

Parameter <N>: Index of the parameter in the list

Type: Data

Argument: None

Forms: Query only

Action: Returns the last result of the specified measured parameter

Get Statistic Value of Parameter

Header: Meas:Res:<N>:<Val>?

Parameter <N>: Index of the parameter in the list

Parameter <Val>: Wfm, Min, Max, Mean, StdDev

Type: Data

Arguments: None

Forms: Command with query only

Action: Returns the specified statistic parameter of the measured parameter

4.12 Limit Tests commands

4.12.1 Limit Test On/Off

Limit Test On/Off

Header: `Limit:TestOn`

Type: `On/off`

Action: Enables/disables the Limit Test. Must be set On after full definition of all other Limit Test parameters.

4.12.2 Limit Test Termination Commands

Limit Test Termination Condition

Header: `Limit:RunUntil`

Type: `Selector`

Arguments: `StopBtn, Failur, Wfm`

Action: Sets the condition of Limit Test Termination

Number of Failures

Header: `Limit:Failures`

Type: `Integer`

Argument: `1 to 10000`

Action: Sets number of failures for the `Failur` Condition of the Limit

Number of Waveforms

Header: `Limit:NWfms`

Type: `Integer`

Argument: `1 to 1000000`

Action: Sets the number of waveforms for the `Wfm` Condition of the Limit

4.12.3 Limit Test Action Commands

Action

Header: `Limit:Action`

Type: `On/off-group`

Items: `Beep, Save, Stop`

Action: `Save` - every signal with a limit condition is stored to the disk
`Beep` - the beep signal will sound for every limit condition
`Stop` - acquisition immediately stops after the first limit condition

Action If

Header: Limit:If

Type: Selector

Arguments: AnyFail, AllPass, AllFail, AnyPass

Action: Define the limit condition:

AnyFail	- one or more active measures fails
AllPass	- all active measures are good
AllFail	- all active measures fail
AnyPass	- one or more active measurements is good

Format of Stored Files

Header: Limit:FileFormat

Type: Selector

Arguments: Binary, Verbose, YOnly

Action: Sets the file format

File Name

Header: Limit:FileName

Type: Data

Argument: Text string

Forms: Command, query, command with query

Action: Defines the file name for saving the specified signals to disk

4.12.4 Parameter Definition Commands

Parameter Activity

Headers:	Limit1:Activ	Limit2:Activ
	Limit3:Activ	Limit4:Activ

Type: On/off

Action: Enables/disables the Limit Test for relevant parameter

Parameter Limit Mode

Headers:	Limit1:Mode	Limit2:Mode
	Limit3:Mode	Limit4:Mode

Type: Selector

Arguments: Center, Limit

Action: Sets the mode of limits for the relevant parameter

Upper and Lower Limits of Parameters

Headers: Limit1:UpLimit Limit1:LowLimit
 Limit2:UpLimit Limit2:LowLimit
 Limit3:UpLimit Limit3:LowLimit
 Limit4:UpLimit Limit4:LowLimit

Type: Float

Arguments: Absolute value of limit

Action: Sets the limit's value. Used only for `Limit` mode of the parameter's limit.

Parameter Center Mode

Headers: Limit1:CenterMode Limit2:CenterMode
 Limit3:CenterMode Limit4:CenterMode

Type: Selector

Arguments: `CurrMean`, `UserDef`

Action: Sets the mode of the center definition for the relevant parameter. Used only for the `Center` mode of the parameter limit.

Center Value

Headers: Limit1:CenterVal Limit2:CenterVal
 Limit3:CenterVal Limit4:CenterVal

Type: Float

Arguments: Absolute value of center

Action: Sets the absolute center value. Used for `UserDef` mode of the center of the parameter.

Parameter Delta Mode

Headers: Limit1:Delta Limit2:Delta
 Limit3:Delta Limit4:Delta

Type: Selector

Arguments: `StdDev`, `UserDef`, `UserPerc`

Action: Sets the mode of delta definition for relevant parameter. Used for `Center` mode of parameter limit only.

Parameter Delta Value for Standard Deviation mode

Headers: Limit1:StdDev Limit2:StdDev
 Limit3:StdDev Limit4:StdDev

Type: Float

Arguments: 0.1 to 100 standard deviations of the parameter

Action: Sets the delta value. Used for StdDev mode of parameter delta only.

Parameter Delta Value for User Defined Mode

Headers: Limit1:UserDef Limit2:UserDef
 Limit3:UserDef Limit4:UserDef

Type: Float

Arguments: Absolute value of delta

Action: Sets the delta value. Used for UserDef mode of delta of the parameter only.

Parameter Delta Percentage for User Defined mode

Headers: Limit1:UserPerc Limit2:UserPerc
 Limit3:UserPerc Limit4:UserPerc

Type: Float

Arguments: 0.01% to 90% standard deviations of the parameter

Action: Sets the delta value. Used for UserPerc mode of delta of the parameter only.

Failure When

Headers: Limit1:FailWhen Limit2:FailWhen
 Limit3:FailWhen Limit4:FailWhen

Type: Selector

Arguments: Outside, Inside, Always

Action: Sets the mode of the quality control for the according parameter

If Measurement Undefined

Headers: Limit1:NotFound Limit2:NotFound
 Limit3:NotFound Limit4:NotFound

Type: Selector

Arguments: Ignore, Fail, Pass

Action: Sets the limit status when measurement is undefined

4.13 Mathematics commands

4.13.1 Enable Mathematical Function

Enable Mathematical Function

Headers: F1:On F2:On
F3:On F4:On

Type: On/off

Action: Enables/disables the calculation and display of the relevant functions

4.13.2 DisplayMathematical Function

DisplayMathematical Function

Headers: F1:Display F2:Display
F3:Display F4:Display

Type: On/off

Action: Enables/disables the visibility of the relevant functions

4.13.3 Function Category

Function Category

Headers: F1:Category F2:Category
F3:Category F4:Category

Type: Selector

Arguments: Arithm, Algebra, Trigonom, FFT, BitOp, Misc, Formula

Action: Sets the category of the specified function

4.13.4 Function Operators

Arithmetic Function Operator

Headers: F1:ArithmOp F2:ArithmOp
F3:ArithmOp F4:ArithmOp

Type: Selector

Arguments: Add, Subtract, Multiply, Divide, Ceil, Floor, Fix, Round, Absolute, Invert, Common, ReScale

Action: Sets the operator of the specified function. Used for Arithm category only.

Algebraic Function Operator

Headers: F1:AlgebraOp F2:AlgebraOp
 F3:AlgebraOp F4:AlgebraOp

Type: Selector

Arguments: ExpE, LogE, Exp10, Log10, ExpA, LogA,
 Differentiate, Integrate, Square, SqRoot,
 Cube, PowerA, Inverse, SqRtOfSum

Action: Sets the operator of the specified function. Used for Algebra category only.

Trigonometric Function Operator

Headers: F1:TrigonOp F2:TrigonOp
 F3:TrigonOp F4:TrigonOp

Type: Selector

Arguments: Sine, ASine, Cosine, ACosine, Tangent,
 ATangent, Cotangent, ACotangent, HSine,
 HCosine, HTangent, HCotangent

Action: Sets the operator of the specified function. Used for Trigonom category only.

FFT Function Operator

Headers: F1:FFTOp F2:FFTOp
 F3:FFTOp F4:FFTOp

Type: Selector

Arguments: FFT, IFFT, FFTMagn, FFTPhase, FFTReal, FFTIm

Action: Sets the operator of the specified function. Used for FFT category only.

Bits Function Operator

Headers: F1:BitOp F2:BitOp
 F3:BitOp F4:BitOp

Type: Selector

Arguments: And, NAnd, Or, NOr, XOr, NXOr, Not

Action: Sets the operator of the specified function. Used for BitOp category only.

Miscellaneous Function Operator

Headers: F1:MiscOp F2:MiscOp
F3:MiscOp F4:MiscOp

Type: Selector

Arguments: LinInterp, SinXInterp, Trend, Smooth

Action: Sets the operator of the specified function. Used for Misc category only.

4.13.5 Function Operands

Operand 1

Headers: F1:Source1 F2:Source1
F3:Source1 F4:Source1

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4B2, F1, F2, F3, F4, M1, M2, M3, M4

Action: Sets the first operand of the specified function

Operand 2

Headers: F1:Source2 F2:Source2
F3:Source2 F4:Source2

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4, Constant

Action: Sets the second operand of the specified function. Used for dual- or quad-operand function.

Operands 3/4

Headers: F1:Source3 F1:Source4
F2:Source3 F2:Source4
F3:Source3 F3:Source4
F4:Source3 F4:Source4

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4, "Don'tCare"

Action: Sets the third and fourth operands for the specified function. Used for bits function.

Constant Value

Headers: F1:Const F2:Const
 F3:Const F4:Const

Type: Float

Arguments: Absolute value of constant

Action: Sets the constant for the specified function. Used when Const is the second operand.

4.13.6 Additional Parameters for Arithmetic Functions

Rounding Step

Headers: F1:RoundTo F2:RoundTo
 F3:RoundTo F4:RoundTo

Type: Float

Arguments: Value of rounding step

Action: Sets the step for rounding function. Used for Ceil, Floor, Fix, Round arithmetic functions.

Rescale Parameters

Headers: F1:ResMult F1:ResOffset
 F2:ResMult F2:ResOffset
 F3:ResMult F3:ResOffset
 F4:ResMult F4:ResOffset

Type: Float

Arguments: Value of Mult and Offset parameters

Action: Sets the Mult and Offset parameters. Used for ReScale arithmetic function.

4.13.7 Additional Parameters for Algebraic Functions

Logarithmic Base

Headers: F1:LogBase F2:LogBase
F3:LogBase F4:LogBase

Type: Float

Arguments: 1.01 to 100

Action: Sets the logarithmic base for LogA algebraic function

Number Exponent

Headers: F1:PowerExp F2:PowerExp
F3:PowerExp F4:PowerExp

Type: Float

Arguments: -100 to +100

Action: Sets the Number Exponent for ExpA algebraic function

4.13.8 Additional Parameters for Trigonometric Functions

Volt-to-Radian Coefficient

Headers: F1:YScaleRad F2:YScaleRad
F3:YScaleRad F4:YScaleRad

Type: Float

Arguments: -100 to +100

Action: Sets the volt-to-radian coefficient for att trigonometric functions

4.13.9 Additional Parameters for FFT Functions

Window

Headers: F1:Window F2:Window
F3:Window F4:Window

Type: Selector

Arguments: Rectang, Hamming, Hanning, FlatTop, BlackHarr, KaiserBess

Action: Sets the window for the specified function

Suppression

Headers: F1: Suppress F2: Suppress
F3: Suppress F4: Suppress

Type: Group-on/off

Arguments: DC, PHASE

Action: DC - on/off the suppression of the spectrum DC component;
PHASE - on/off the suppression of the spectrum phase noise.

F1:SupprLevel: Float (-120 to -10, dB)

Phase Suppression Level

Headers: F1: SupprLevel F2: SupprLevel
F3: SupprLevel F4: SupprLevel

Type: Float

Arguments: -120 dB to -10 dB

Action: Sets the phase suppression level with respect to a maximum magnitude

4.13.10 Additional Parameters for Bit Functions

Source Thresholds

Headers: F1: Thresh1 F2: Thresh1
F3: Thresh1 F4: Thresh1

F1: Thresh2 F2: Thresh2
F3: Thresh2 F4: Thresh2

F1: Thresh3 F2: Thresh3
F3: Thresh3 F4: Thresh3

F1: Thresh4 F2: Thresh4
F3: Thresh4 F4: Thresh4

Type: Float

Arguments: Value of thresholds

Action: Sets the threshold levels for each source of the bit functions

Source Inversion

Headers: F1: SorceInvert F2: SorceInvert
F3: SorceInvert F4: SorceInvert

Type: Group-on/off

Arguments: SRC1, SRC2, SRC3, SRC4

Action: Enables/disables the inversion of each source

4.13.11 Additional Parameters for Miscellaneous Functions

Smoothing Parameter

Headers: F1:SmoothLen F2:SmoothLen
F3:SmoothLen F4:SmoothLen

Type: Integer

Argument: 0 to 24

Action: Sets the length of the smoothing interval in points for the specified function. Used for Smooth operator only. Length is defined as $3 + \langle \text{Argument} \rangle * 2$.

Signal Length

Headers: F1:SignalLen F2:SignalLen
F3:SignalLen F4:SignalLen

Type: Integer-type command

Argument: 2048, 4096 or 8192

Action: Sets the length of the interpolation function signal. Used for LinInterp and SinXInterp functions.

Trend Measurement

Headers: F1:TrendMeas F2:TrendMeas
F3:TrendMeas F4:TrendMeas

Type: Selector

Arguments: Period, Freq, PosWidth, NegWidth, RiseTime, FallTime, PosDuty, NegDuty

Action: Sets the kind of trends for the specified function. Used for Trend operator only.

4.13.12 Function Scaling

Complex Format

Header: F1:ComplexScale F2:ComplexScale
F3:ComplexScale F4:ComplexScale

Type: Selector

Arguments: Magnitude, Phase, Real, Imaginary

Action: Defines the spectrum display mode for FFT function

Vertical Scale Type

Header: F1:VScaleType F2:VScaleType
F3:VScaleType F4:VScaleType

Type: Selector

Arguments: Linear, Logarithm

Action: Defines the vertical scale type for Magnitude of the FFT function

Vertical linear Scale

Header: F1:VertScale F2:VertScale
F3:VertScale F4:VertScale

Type: Float

Arguments: 1e-6 to 1e6

Action: Defines the vertical scale in volts/div for Linear vertical scale type

Vertical linear Position

Header: F1:VoltPosit F2:VoltPosit
F3:VoltPosit F4:VoltPosit

Type: Float

Arguments: -10 to +10

Action: Defines the vertical position in div for Linear vertical scale type

Vertical logarithmic Scale

Header: F1:VertDBScale F2:VertDBScale
F3:VertDBScale F4:VertDBScale

Type: Float

Arguments: 1 to 120

Action: Defines the vertical scale in dB/div for Logarithm vertical scale type

Vertical logarithmic Position

Header: F1:Posit0DB F2:Posit0DB
F3:Posit0DB F4:Posit0DB

Type: Float

Arguments: -10 to +10

Action: Defines the vertical position in div for Logarithm vertical scale type

Vertical Phase Scale

Header: F1:PhaseScale F2:PhaseScale
F3:PhaseScale F4:PhaseScale

Type: Float

Arguments: 0.125 to 8

Action: Defines the vertical scale in rad/div for Phase display mode

Vertical Phase Position

Header: F1:PhasePosit F2:PhasePosit
F3:PhasePosit F4:PhasePosit

Type: Float

Arguments: -10 to +10

Action: Defines the vertical position in div for Phase display mode

Horizontal Scale

Header: F1:HorScale F2:HorScale
F3:HorScale F4:HorScale

Type: Float

Arguments: Depends on actual timebase

Action: Defines the horizontal scale in actual X-axis unit/div

Horizontal Position

Header: F1:HorPosition F2:HorPosition
F3:HorPosition F4:HorPosition

Type: Float

Arguments: Depends on actual timebase

Action: Defines the horizontal position in actual X-axis unit

4.14 Histogram commands

4.14.1 General Histogram Commands

Histogram Axis

Header: `Hist:Axis`
Type: Selector
Arguments: `Off, Vert, Horiz`
Action: Sets the axis of the histogram

Histogram Source

Header: `Hist:Source`
Type: Selector
Arguments: `Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4`
Action: Selects the specified signal as source of the histogram

Histogram Visibility

Header: `Hist:Visible`
Type: On/off
Action: Sets the visibility of the histogram. The acquisition of the histogram proceeds independently of this commands.

4.14.2 Histogram Completion Commands

Histogram Finish Condition

Header: `Hist:RunUntil`
Type: Selector
Arguments: `StopSingle, Wfms, Samples`
Action: Sets the finish condition for acquiring the histogram

Number of Waveforms for Histogram

Header: `Hist:NWfm`
Type: Integer-type command
Argument: 1 to 1000000
Action: Sets the number of signals for the termination of histogram acquisition

Number of Samples for Histogram

Header: Hist:NSample

Type: Integer-type command

Argument: 1 to 10000000

Action: Sets the number of samples for the termination of histogram acquisition

4.14.3 Histogram Window Commands

Limit Mode for Histogram Window

Header: Hist:Limits

Type: Selector

Arguments: Paired, Independ

Action: Sets the mode of the limits of the histogram window

Limit Unit for Histograms Window

Header: Hist:Units

Type: Selector

Arguments: Absolute, Percent

Action: Sets the units of the limits of the histogram window

Left and Right Window Limits for Vertical or Horizontal Histogram

Headers:	Hist:WVert:Left	Hist:WVert:Right
	Hist:WHor:Left	Hist:WHor:Right

Type: Float

Argument: Real value of the X-axis (for Absolute units)
0% to 100% of the X-axis (for Percent units)

Action: Sets the X positions of the histogram window

Top and Bottom Window Limits for Vertical or Horizontal Histogram

Headers:	Hist:WVert:Top	Hist:WVert:Bottom
	Hist:WHor:Top	Hist:WHor:Bottom

Type: Float

Argument: Real value of the Y-axis (for Absolute units)
0% to 100% of the Y-axis (for Percent units)

Action: Sets the Y positions of the histogram window

Window Visibility

Header: `Hist:Display`
Type: On/off
Action: Sets the visibility of the window

Set Default Window

Header: `Hist:SetDefWind`
Type: Executing-type command
Action: Sets the default window depending on the axis

4.14.4 Histogram Calculation Commands

Calculation Mode

Header: `Hist:Mode`
Type: Selector
Arguments: `Normal`, `Exponent`
Action: Sets the mode of histogram calculation

Weight for Exponential Calculation

Header: `Hist:Weight`
Type: Integer-type command
Argument: `8`, `16`, `32`, ..., `8192`
Action: Sets the weight coefficient for the `Exponent` calculation mode

4.14.5 Histogram Scale Commands

Scale Type

Header: `Hist:ScaleType`
Type: Selector
Arguments: `Linear`, `Logarith`
Action: Sets the type of histogram scale

Scale Mode

Header: `Hist:ScaleMode`
Type: Selector
Arguments: `Auto`, `Manual`
Action: Sets the mode of histogram scale

Linear Scale of Vertical or Horizontal Histogram

Headers: Hist:VertScale
Hist:HorScale

Type: Float

Argument: (10 to 100) %/div

Action: Sets the scale of histograms. Used for Manual mode and Linear type of scale only.

Linear Offset of Vertical or Horizontal Histogram

Headers: Hist:VertOffset
Hist:HorOffset

Type: Float

Argument: 0% to 100%

Action: Sets the offset of the histograms. It used for Manual mode and Linear type of scale only.

Logarithmic Scale of Vertical or Horizontal Histogram

Headers: Hist:VertDBScale
Hist:HorDBScale

Type: Float

Argument: (6 to 60) dB/div

Action: Sets the scale of the histograms. Used for Manual mode and Logarith type of scale only.

Logarithmic Offset of Vertical or Horizontal Histogram

Headers: Hist:VertDBOffs
Hist:HorDBOffs

Type: Float

Argument: (-60 to 0) dB

Action: Sets the offset of the histograms. Used for Manual mode and Logarith type of scale only.

4.14.6 Histogram Result Commands

Get Histogram Data

Headers: `Hist:Data?`

Type: `Data`

Argument: `None`

Forms: `Query only`

Action: Returns a set of text strings with the pair of numbers (comma-separated). First number in the each pair is the histogram axis value, and second number is the histogram value in this point.

Get Histogram Measure

Headers: `Hist:Res:<Param>?`

Parameter<Param>:

- `InBox` – number of hints in box
- `Wfm` – number of waveforms
- `Peak` – peak value of histogram
- `PP` – difference between highest and lowest values of signal
- `Median` – centre between highest and lowest values of signal
- `Mean` – average of distribution of histogram
- `StdDev` – standard deviation of histogram
- `Mean1S` – number of hints in $\text{Mean} \pm \text{StdDev}$ region, %
- `Mean2S` – number of hints in $\text{Mean} \pm 2\text{StdDev}$ region, %
- `Mean3S` – number of hints in $\text{Mean} \pm 3\text{StdDev}$ region, %
- `Min` – min. value of signal
- `Max` – max. value of signal
- `Max-Max` – difference between two values of signal, matched two max of histogram

Type: `Data`

Argument: `None`

Forms: `Query only`

Action: Returns a text string with the value of the specified parameter

4.15 Eye Diagram commands

4.15.1 General Eye Commands

Type of Eye Measurements

Header: Eye:Measure

Type: Selector

Arguments: Off, NRZ, RZ

Action: Sets the type of eye measurements

Sources for Eye Measurements

Header: Eye:Source

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4

Action: Sets the source for eye measurements

Number of Waveforms in one Measurement

Header: Eye:WfmsInCycle

Type: Integer

Argument: 64, 128, 256, 512, 1024

Action: Sets the number of waveforms in one measurement

4.15.2 Eye Measurements Commands

List of X-Axis NRZ Measurements

Header: Eye:XNRZParam

Type: Group-on/off

Items: Area, BitRate, BitTime, CrossTime, CycleArea, DutCycDistP, DutCycDistS, EyeWidth, EyeWidthP, FallTime, Freq, JitterPP, JitterRMS, Period, RiseTime

Action: 1Defines the set of X-axis measurements for NRZ signals

List of Y-Axis NRZ Measurements

Header: Eye:YNRZParam

Type: Group-on/off

Items: AcRMS, AvgPower, AvgPWdBm, CrossPerc, CrossLevel, ExtRatioDB, ExtRatioP, ExtRatio, EyeAmpl, EyeHeight, EyeHeightDB, Max, Mean, Mid, Min, NegOver, PPNoiseOne, PPNoiseZero, RMSNoiseOne, RMSNoiseZero, OneLevel, PeakPeak, PosOver, RMS, SNRaio, SNRaioDB, ZeroLevel

Action: Defines the set of Y-axis measurements for NRZ signals

List of X-Axis RZ Measurements

Header: Eye:XRZParam

Type: Group-on/off

Items: Area, BitRate, BitTime, CycleArea, EyeWidth, EyeWidthP, FallTime, JittPpFall, JittPpRise, JittRMSFall, JittRMSRise, NegCross, PosCross, PosDutyCyc, PulseSymm, PulseWidth, RiseTime

Action: Defines the set of X-axis measurements for RZ signals

List of Y-Axis RZ Measurements

Header: Eye:YRZParam

Type: Group-on/off

Items: AcRMS, AvgPower, AvgPWdBm, Contrast, ContrastBb, ContrastP, ExtRatioDB, ExtRatioP, ExtRatio, EyeAmpl, EyeHeight, EyeHeightDB, EyeOpenFact, Max, Mean, Mid, Min, PPNoiseOne, PPNoiseZero, RmsNoiseOne, RMSNoiseZero, OneLevel, PeakPeak, RMS, SignToNoise, ZeroLevel

Action: Defines the set of Y-axis measurements for RZ signals

Measurements List Clearing

Header: Eye:ClearAllMeas

Type: Executing

Action: Clears the list of measurement parameters

4.15.3 Define Parameters Commands

Eye Frame Visibility

Header: Eye:DisplayWind

Type: On/off

Action: Sets the visibility of the eye frame

Left and Right Boundary for NRZ Top/Base Finding

Headers: Eye:LeftBound
Eye:RightBound

Type: Float

Argument: 10% to 90% of the NRZ period

Action: Sets the zone of the period of the NRZ signal for the top/base calculation

Threshold Definition Mode

Header: Eye:ThreshMode

Type: Selector

Arguments: 10-90, 20-80, Custom

Action: Sets the mode of threshold definition

Upper and Lower Threshold

Headers: Eye:UpThresh
Eye:LowThresh

Type: Float

Argument: 5% to 95% of amplitude

Action: Sets the thresholds for the slopes calculation. Used for Custom mode.

4.15.4 Eye Calculation Commands

Measurement Statistic

Header: `Eye:Statistic`

Type: `On/off`

Action: `Enables/disables measurement statistics`

Measurement Statistic Mode

Header: `Eye:Mode`

Type: `Selector`

Arguments: `Permanent, Window, Weight`

Action: `Sets the mode of statistics calculation. Used when statistic is enable.`

Windows Value

Header: `Eye:Window`

Type: `Integer`

Argument: `8, 16, 32, ..., 8192`

Action: `Sets the window value. Used for Window mode of statistics.`

Weight Value

Header: `Eye:Weight`

Type: `Integer`

Argument: `8, 16, 32, ..., 8192`

Action: `Sets the weight value. Used for Weight mode of statistics.`

4.15.5 Getting Eye Measurement Results

Get List of Measured Parameters

Header: Eye:Res:List?

Type: Data

Argument: None

Forms: Query only

Action: Returns a list of active eye measurements with ordinal index

Get Current Value of Parameter

Header: Eye:Res:<N>?

Parameter <N>: Index of parameter in the list

Type: Data

Argument: None

Forms: Query only

Action: Returns the result of the specified measured parameter

Get Statistic Value of Parameter

Header: Eye:Res:<N>:<Val>?

Parameter <N>: index of the parameter in the list

Parameter <Val>: TotalMeas, Min, Max (when Measurement Statistic off)

Parameter <Val>: Cycle, Wfm, Min, Max, Mean, StdDev (when Measurement Statistic on)

Type: Data-type command

Arguments: none

Forms: query only.

Action: return the specified statistical parameter of the measured parameter

4.16 Mask Test commands

4.16.1 Common Mask Test Commands

Mask Test On

Header: Mask:TestOn

Type: On/off

Action: Enables/disables the mask test functionality

Signal for Mask Testing

Header: Mask:CompareWith

Type: Selector

Arguments: Ch1, Ch2, CH3, CH4, F1, F2, F3, F4, DB

Action: Selects the signal for mask testing

ActuateMask Testing

Header: Mask:Testing

Type: On/off

Action: Enables/disables the comparison with current mask

Mask Erasing

Header: Mask:EraseMask

Type: Execution

Action: Clears the current mask from the display

4.16.2 Mask Creating

Mask Creating Mode

Header: Mask:CreateAs

Type: Selector

Arguments: Std, Auto, Edit

Action: Sets the mask creation method

4.16.3 Standard Mask Test Commands

Get List of Standards

Header: StdMask:StdsList?

Type: Data

Argument: None

Forms: Query only

Action: Returns a list of mask standards with ordinal index

Select Standard

Header: StdMask:StdIndex

Type: Integer

Argument: 0 to (number of standards-1)

Action: Selects the current standard by its ordinal index

Get List of Masks

Header: StdMask:MasksList?

Type: Data

Argument: None

Forms: Query only

Action: Returns a list of masks with ordinal index from the selected standard

Select Standard Mask

Header: StdMask:MaskIndex

Type: Integer

Argument: 0 to (number of masks in the current standard-1)

Action: Loads the specified mask by its ordinal index

Alignment of Signal with Standard Mask

Header: StdMask:Align

Type: On/off

Action: Enables/disables the alignment of the tested signal with the standard mask parameters

Enable Margins

Header: StdMask:MarginsOn

Type: On/off

Action: Enables/disables the margin control of eye-typed masks

Margins Value

Header: StdMask:MarginsVal

Type: Float

Arguments: -100% to +100%

Action: Sets the margin's value. Used when margins are enabled

Build Immediately

Header: StdMask:BuildImmediate

Type: On/off

Action: Enables/disables creation of the standard mask immediately after any of its parameters change

4.16.4 Automask Commands

Automask Source

Header: Automask:Source

Type: Selector

Arguments: Ch1, Ch2, CH3, CH4, F1, F2, F3, F4

Action: Selects the signal as a template for automask building

Margins Units

Header: Automask:Unit

Type: Selector-type command

Arguments: Division, Current

Action: Selects the margins units for automask building

Automask X-Margins

Header: Automask:DeltaX

Type: Float

Arguments: (0.02 to 2) div for Division margins units real X-axis value for Current margins units

Action: Sets the X-axis margins around the template signal

Automask Y-Margins

Header: Automask:DeltaY

Type: Float

Arguments: (0.03125 to 2) div for Division margins units real Y-axis value for Current margins units

Action: Sets the Y-axis margins around the template signal

Automask Build

Header: Automask:BuildAMask

Type: Execution

Action: Builds automask immediately

4.16.5 Mask Test Termination

Mask Test Finish Condition

Header: `Mask:RunUntil`

Type: Selector

Arguments: `StopBtn`, `FailedWfms`, `FailedSmpls`, `Wfms`, `Samples`

Action: Sets the condition of mask test termination

Number of Failed Waveforms

Header: `Mask:FailWfms`

Type: Integer

Argument: 1 to 1000000

Action: Sets the number of failed waveforms for the `FailedWfms` finish condition

Number of Failed Samples

Header: `Mask:FailSmpls`

Type: Integer-type command

Argument: 1 to 1000000

Action: Sets the number of failed samples for the `FailedSmpls` finish condition

Number of Waveforms

Header: `Mask:NWfms`

Type: Integer-type command

Argument: 1 to 1000000

Action: Sets the number of waveforms for the `Wfms` finish condition

Number of Samples

Header: `Mask:NSamples`

Type: Integer-type command

Argument: 1 to 1000000

Action: Sets the number of samples for the `Samples` finish condition

4.16.6 Mask Test Actions

Select Actions

Header: Mask:Action

Type: Group-on/off

Items: Beep, Save

Action: Save – every failed signal is stored to disk
Beep – the beep signal will sound for every failed signal**Format of Stored Files**

Header: Mask:FileFormat

Type: Selector

Arguments: Binary, Verbose, YOnly

Action: Sets the file format. Used when Save action is on.

Stored File Name

Header: Mask:FileName

Type: Data

Argument: Text string

Forms: Command, query, command with query

Action: Defines the name for storing failed signals on Disk. Used when Save action is on.

4.16.7 User Mask

User Masks File Name

Header: Mask:MaskFile

Type: Data

Argument: Text string

Forms: Command, query, command with query.

Action: Defines the file name for next loading or saving user mask from the disk

Load User Mask

Header: Mask:LoadUser

Type: Execution

Action: Loads the previously specified user mask

Save User Mask

Header: Mask:SaveUser

Type: Execution

Action: Saves the current mask as user with previously specified file name

4.16.8 Getting Mask Test Results

Get Integrated Results of Mask Test

Headers: Mask:Res:<Param>?

Parameter <Param>:

AllWfm	- number of waveforms
FailWfm	- number of failed waveforms
AllSmpl	- number of samples
FailSmpl	- number of failed samples

Type: Data

Argument: None

Forms: Query only

Action: Returns a text string with the value of the specified parameter

Get Number of Samples in Selected Polygons

Headers: Mask:Res:Poly<N>?

Parameter <N>: Number of the polygon, 1 to 8

Type: Data

Argument: None

Forms: Query only

Action: Returns a text string with the values of failed samples on specified polygon

Get Number of Samples in Margins of Selected Polygon

Headers: Mask:Res:Poly<N>Mar?

Parameter <N>: Number of the polygon, 1 to 4

Type: Data

Argument: None

Forms: Query only

Action: Returns a text string with the values of failed samples on the margin of specified polygon. Used when `Margins` enabled.

Get Number of Samples in Selected Polygon with Margins Together

Headers: Mask:Res:Poly<N>All?

Parameter <N>: Number of the polygon, 1 to 4

Type: Data

Argument: None

Forms: Query only

Action: Returns a text string with the total number of failed samples on the margin and on the specified polygon. Used when Margins enabled.

4.17 Autocalibration commands

4.17.1 Single-shot Autocalibration

Start autocalibration of channels 1 and 2

Header: `Flash:Sampler:Autocal:Start`

Type: Executing-type command

Action: Start self-calibration of Sampler 1

Start autocalibration of channels 3 and 4 (PicoScope 9341 only)

Header: `Flash:Smplr2:Autocal:Start`

Type: Executing-type command

Action: Start self-calibration of Sampler 2

Start autocalibration of timebase

Header: `FLASH:TBTrig:Autocal`

Type: Execution

Action: Starts self-calibration of timebase

Get the autocalibration status query

Header: `Flash:Calibr:AutocalResult?`

Type: Integer

Action: Command is ignored, and query returns an integer:

- 1 - Autocalibration in progress.
- 0 - Autocalibration finished OK.
- 1 - Signal must be disconnected from Ch1 Input. Autocalibration of the Channels is aborted.
- 2 - Signal must be disconnected from Ch2 Input. Autocalibration of the Channels is aborted.
- 3 - Signal must be disconnected from Ch1 and Ch2 Inputs. Autocalibration of the Channels is aborted.
- 5 - Autocalibration failed.

4.17.2 Periodic Autocalibration

When to Begin Autocalibration

Header: Util:CalibrWhen

Type: On/off-group

Items: PowerOn, Period, Temperat

Action: PowerOn - autocalibration begins on the next Power On
Period - autocalibration begins periodically after the specified interval
Temperat - autocalibration begins when deviation of temperature inside the instrument exceeds the specified value

Note. Periodic autocalibration must be turned off when GUI is in RemoteOnly or Invisible state. See [GUI command](#).

Autocalibration Period

Header: Util:CalPeriod

Type: Float

Argument: 0.5 to 16 hours

Action: Sets the autocalibration period in hours

Temperature Deviation

Header: Util:TempChange

Type: Float

Argument: 0.5 to 10 °C

Action: Sets the temperature deviation for autocalibration

Get the Temperature of the Instrument Query

Header: Calibr:Temperature?

Type: Float

Argument: None

Forms: Query only

Action: Returns the temperature inside the device in degrees Celsius

4.17.3 Balancing the channels manually

Balancing channels 1 and 2 manually

Header: Flash:Sampler:Ch1:FullBW:Balance
 Flash:Sampler:Ch1:NarrowBW:Balance
 Flash:Sampler:Ch2:FullBW:Balance
 Flash:Sampler:Ch2:NarrowBW:Balance

Type: Float

Arguments: -0.5 to 0.5

Action: Query or set the balance value in volts for the specified channel

Balancing channels 3 and 4 manually (PicoScope 9341 only)

Header: Flash:Smplr2:Ch3:FullBW:Balance
 Flash:Smplr2:Ch3:NarrowBW:Balance
 Flash:Smplr2:Ch4:FullBW:Balance
 Flash:Smplr2:Ch4:NarrowBW:Balance

Type: Float

Arguments: -0.5 to 0.5

Action: Query or set the balance value in volts for the specified channel of
 Sampler 2

4.18 Waveforms commands

This group of commands is designed for receiving acquired waveforms from the oscilloscope.

Waveform Source

Header: Wfm:Source

Type: Selector

Arguments: Ch1, Ch2, Ch3, Ch4, F1, F2, F3, F4, M1, M2, M3, M4

Action: Sets the signal to be received

Spectrum Format

Header: Wfm:Complex

Type: Selector

Arguments: Mod, Ph, Re, Im

Action: Selects the received component of the complex signal. Used for spectrum data.

Get Waveform Data

Header: Wfm:Data?

Type: Data

Argument: None

Forms: Query only

Action: Returns a text string with values of all points of the signal (comma-separated)

Get Number of Points in the Waveform

Header: Wfm:Preamb:Poin?

Type: Data

Argument: None

Forms: Query only

Action: Returns the number of points in the signal

Get X-Axis Step

Header: Wfm:Preamb:XInc?

Type: Data

Argument: None

Forms: Query only

Action: Returns the increment on the X-axis for one signal point

Get X-Axis Origin

Header: Wfm:Preamb:XOrg?

Type: Data

Argument: None

Forms: Query only

Action: Returns the X-axis value for the first signal point

Get X-Axis Unit

Header: Wfm:Preamb:XU?

Type: Data

Argument: None

Forms: Query only

Action: Returns the X-axis physical units

Get Y-Axis Unit

Header: Wfm:Preamb:YU?

Type: Data

Argument: None

Forms: Query only

Action: Returns the Y-axis physical units

4.19 Optical commands (PicoScope 9321 only)

4.19.1 Common optical commands

O/E converter commutation

Header: OEConv:Dest

Type: Selector

Arguments: OEConv, Ch1, Ch2

Action: set destination of the O/E converters stimuli

Current wavelength

Header: OEConv:Wavelength

Type: Selector

Arguments: WL1, WL2, WL3, WLUserDef

Action: set the current wavelength.

Note: Arguments WL1, WL2, WL3 selects one of the factory defined wavelength value. Usually WL1 is 1550 nm, WL2 is 1310 nm and WL3 is 850 nm. Argument WLUserDef allows you to select user defined values.

Select user wavelength value

Header: OEConv:UserWaveLen

Type: Selector

Arguments: WL1, WL2, WL3, . . . , WL8

Action: set the current user defined wavelength.

Note: All user-defined wavelengths are sorted by value. WL1 is the shortest user-defined wavelength.

Reading current O/E conversion gain value

Header: OEConv:ConversionGain

Type: Float

Argument: 10 V/W to 1500 V/W

Forms: query only

Action: reading the current conversion gain value.

4.19.2 User-defined calibration points commands

Erasing all user-defined calibration points

Header: Flash:OeConv:ClearAllPoints

Type: Execution

Action: Erase all user-defined wavelength points.

Erasing one user-defined calibration pointHeader: Flash:OeConv:ClearPoint1
Flash:OeConv:ClearPoint2
.
Flash:OeConv:ClearPoint8

Type: Execution

Action: Erase specified user-defined wavelength point.

Set the wavelength value for user-defined calibration pointHeader: Flash:OeConv:CalWavelen1
Flash:OeConv:CalWavelen2
.
Flash:OeConv:CalWavelen8

Type: Float

Argument: 750 nm to 1.65 μ m

Action: setting the wavelength value for user-defined point.

Set the conversion gain value for user-defined calibration pointHeader: Flash:OeConv:CalGain1
Flash:OeConv:CalGain2
.
Flash:OeConv:CalGain8

Type: Float

Argument: 10 V/W to 1500 V/W

Action: setting the conversion gain value for user-defined point.

4.19.3 Dark Level calibration commands

Dark Level autocalibration

Header: Flash:OeConv:AutoDarkLvl

Type: Execution

Action: starting the Dark Level autocalibration procedure. The duration of this process is less than 5 s.

Dark Level value

Header: Flash:OeConv:DarkLvl

Type: Float

Argument: 0 % to 100 %

Action: query about Dark Level value after autocalibration procedure;
setting Dark Level value when autocalibration is undesirable.

4.19.4 Calibration values saving commands

Store all calibration values

Header: Flash:OeConv:Calibr:Store

Type: Execution

Action: Store current Dark Level and all user defined calibration points into the nonvolatile memory of the PicoScope 9321.

Recall all calibration values

Header: Flash:OeConv:Calibr:Recall

Type: Execution

Action: Recall current Dark Level and all user defined calibration points from the nonvolatile memory of the PicoScope 9321. This command calls automatically when PicoScope 9321 power on.

4.20 TDR/TDT Commands

4.20.1 Stimuli source command (PicoScope 9311 and 9312 only)

Stimuli source command

Header: TDR:Stimul:Source
Type: Selector
Arguments: Internal, External
Action: set source of stimuli

4.20.2 Common Internal Stimuli commands (PicoScope 9311/9312 only)

Stimuli Mode

Header: TDR:Stimul:GlVsG2
Type: Selector
Arguments: Separate, Paired
Action: set mode of stimulus setups

Internal Stimuli On

Header: TDR:Stimul:Out1:On TDR:Stimul:Out2:On
Type: On/off
Action: turn on or turn off the specified stimulus

Internal Stimuli Period

Header: TDR:Stimul:Period
Type: Float
Argument: 1e-6 to 60e-3
Action: set pulse period of the stimuli in s

Internal Stimuli Width

Header: TDR:Stimul:Width
Type: Float
Argument: 2e-7 to 4e-6
Action: set pulse width of the stimuli in s

4.20.3 PicoScope 9311 Internal Stimuli commands

Internal Stimuli Amplitude

Header: TDR:Stimul:Srd:Out1:Ampl
TDR:Stimul:Srd:Out2:Ampl

Type: Float-type command

Argument: 2 to (8); max value depends on the adjustments of generators.

Action: set stimuli amplitude in V

Internal Stimuli Deskew

Header: TDR:Stimul:Deskew

Type: Float

Argument: $-500e-12$ to $500e-12$

Action: set stimuli deskew in s

Internal Stimuli pre-trigger

Header: TDR:Stimul:Pretrig

Type: Float

Argument: $20e-9$ to $25e-9$

Action: set stimuli pre-trigger in s

Internal Stimuli performance

Header: TDR:Stimul:Srd:Out1:Perform
TDR:Stimul:Srd:Out2:Perform

Type: Selector

Arguments: Flat, Fast

Action: set performance of the specified stimulus

4.20.4 PicoScope 9312 Internal Stimuli commands

Internal Stimuli Deskew

Header: TDR:Stimul:Deskew

Type: Float

Argument: 0 % to 100 %

Action: set stimuli deskew in %

Tunnel diode head calibration

Header: TDR:Stimul:Td:Out1:Autocal
TDR:Stimul:Td:Out2:Autocal

Type: Execution

Action: find the optimal tunnel diode current

Tunnel diode current mode

Header: TDR:Stimul:Td:Out1:CurrentMode
TDR:Stimul:Td:Out2:CurrentMode

Type: Selector

Arguments: Auto, Manual

Action: set mode of tunnel diode current control for the specified stimulus

Tunnel diode current

Header: TDR:Stimul:Td:Out1:Sensitive
TDR:Stimul:Td:Out2:Sensitive

Type: Float

Argument: 0 to 0.06 for positive head; -0.06 to 0 for negative head

Action: set the tunnel diode current in A

4.20.5 Reflectometer Setup commands

Reflectometer Mode

Header: TDR:Mode

Type: Selector

Arguments: Off, TDR, TDT

Action: set reflectometer mode

Device under Test

Header: TDR:DUT

Type: Selector

Arguments: Single, Differ, Common

Action: set the device under test.

Note: both the stimuli must be on for *Differ* and *Common* modes

Stimulus Destination

Header: TDR:Ch1:Destination
TDR:Ch2:Destination

Type: Selector

Arguments: G1, G2, Off

Action: set the stimulus connected to the specified channel.

Polarity of the stimulus

Header: TDR:Ch1:Polar
TDR:Ch2:Polar

Type: Selector

Arguments: Pos, Neg

Action: get the polarity of the internal stimulus of the specified channel; set polarity of the external stimulus.

Vertical scale units for TDR mode

Header: TDR:Ch1:VertScaleTDR
TDR:Ch2:VertScaleTDR

Type: Selector

Arguments: Volt, Rho, Ohm

Action: set vertical scale for TDR mode.

Vertical scale units for TDT mode

Header: TDR:Ch1:VertScaleTDT
TDR:Ch2:VertScaleTDT

Type: Selector

Arguments: Volt, Gain, DbGain

Action: set vertical scale for TDT mode.

Horizontal Scale units

Header: TDR:HorScale

Type: Selector

Arguments: Time, Meter, Foot, Inch

Action: set horizontal scale units.

Mode of transmission line parameters

Header: TDR:Unit

Type: Selector

Arguments: Velocity, DielConst

Action: set mode of transmission line parameter definitions. It used for non-time horizontal scale units

Propagation Velocity

Header: TDR:Veloc
Type: Float
Argument: 0.1 to 1
Action: set propagation velocity value for the transmission line.

Dielectric Constant

Header: TDR:DielConst
Type: Float
Argument: 1 to 100
Action: set Dielectric Constant value for the transmission line.

Reference Amplitude (TDR Mode)

Header: TDR:Ch1:RefAmplTDR
TDR:Ch2:RefAmplTDR
Type: Float
Argument: 0.001 V to 10 V
Action: set Reference Amplitude value for calculating Rho, Ohm.

Base Line (TDR Mode)

Header: TDR:Ch1:BaseLineTDR
TDR:Ch2:BaseLineTDR
Type: Float
Argument: -10 V to 10 V
Action: set Base Line value for calculating Rho, Ohm.

Reference Amplitude (TDT Mode)

Header: TDR:Ch1:RefAmplTDT
TDR:Ch2:RefAmplTDT
Type: Float
Argument: -10 V to 10 V
Action: set Reference Amplitude value for calculating Gain and Gain (dB).

Reference Zero (TDT Mode)

Header: TDR:Ch1:RefZeroTDT
TDR:Ch2:RefZeroTDT
Type: Float

Argument: -10 V to 10 V

Action: set Reference Zero value for calculating Gain and Gain (dB).

Reference Plane

Header: TDR:Ch1:RefPlane
TDR:Ch2:RefPlane

Type: Float

Argument: -1e-11 s to 1e-8 s

Action: set Reference plane position.

Corrected Time

Header: TDR:Ch1:CorrectTime
TDR:Ch2:CorrectTime

Type: Float

Argument: -1e-11 s to 1e-8 s

Action: set the Corrected Time for pulse correction.

Start Calibration Rho (TDR Mode)

Header: TDR:Ch1:DoCalibrTDR
TDR:Ch2:DoCalibrTDR

Type: Execution

Action: Rho calibration procedure. Follow the instructions on the screen.

Note before sending this command, check:
- Reflectometer Mode is TDR;
- specified channel is active;
- selected stimulus is ON.

Start Gain Calibration (TDT Mode)

Header: TDR:Ch1:DoCalibrTDT
TDR:Ch2:DoCalibrTDT

Type: Execution

Action: Gain Calibration procedure. Follow the instructions on the screen.

Note before sending this command, check:
- Reflectometer Mode is TDT;
- specified channel is active;
- selected stimulus is ON.

Correction

Header: TDR:Ch1:Correction
TDR:Ch2:Correction

Type:	On/Off
Action:	switch on Correction mode. Do this immediately after switching on. Follow the instructions on the screen.
Note	before sending this command, check: <ul style="list-style-type: none">- Reflectometer Mode is TDR or TDT;- specified channel is active;- selected stimulus is ON.

4.21 Instrument Info Commands

Get Device Model

Header: GetInfo:Model?

Type: Data

Argument: none

Forms: query only

Action: return text string with model of the instrument

Get Device Functionality

Header: GetInfo:Function?

Type: Data

Argument: none

Forms: query only

Action: return text string with functionality of the instrument

Get Device Year

Header: GetInfo:Year?

Type: Data

Argument: none

Forms: query only

Action: return text string with year of the instrument

Get Device Serial Number

Header: GetInfo:SerialNr?

Type: Data

Argument: none

Forms: query only

Action: return text string with Serial Number of the instrument

Get Device FPGA FW Version

Header: GetInfo:HwVersion?

Type: Data

Argument: none

Forms: query only

Action: return text string with FPGA firmware version of the instrument

Get Device FW Version

Header: GetInfo:FwVersion?

Type: Data

Argument: none

Forms: query only

Action: return text string with firmware version of the instrument

Get Device SW Version

Header: GetInfo:SwVersion?

Type: Data

Argument: none

Forms: query only

Action: return text string with software version of the instrument

Get Current Interface of Device

Header: GetInfo:Interface?

Type: Data

Argument: none

Forms: query only

Action: return text string with current interface of the instrument

Get Current IP Address

Header: GetInfo:IpAddress?

Type: Data

Argument: none

Forms: query only

Action: return text string with the current IP address of the instrument

Get Current Subnet Mask

Header: GetInfo:SubnetMask?

Type: Data

Argument: none

Forms: query only

Action: return text string with the current subnet mask of the instrument

Get Current Gateway

Header: GetInfo:Gateway?

Type: Data

Argument: none

Forms: query only

Action: return text string with current gateway of the instrument

Get Current MAC Address

Header: GetInfo:MacAddress?

Type: Data

Argument: none

Forms: query only

Action: return text string with the current MAC address of the instrument

5 Revision history

Date	Version	Changes
2017-02-09	4	4.2 GUI commands Chapter fully updated
		4.4 Channels commands <ol style="list-style-type: none"> Removed space from AllChs:BestFlat mnemonics Added Scale a TDR Channel command Added Scale a TDT Channel command Added Offset a TDR Channel command Added Offset a TDT Channel command
		4.5 Timebase commands <ol style="list-style-type: none"> Added Main Distances Scale for TDR-TDT command Added Delayed Distances Scale for TDR-TDT command Added Distances Delay for TDR-TDT command
		4.8 Display commands <ol style="list-style-type: none"> Added Display Persistence (SW v.3.20 or newer) command Added Display Style (SW v.3.20 or newer) command Updated arguments of Graticule Type command
		4.9.3 Work with Disk Updated header of Save to Disk command
		4.10 Markers commands Added Y value of XY-Marker query
		4.11 Measure commands Structure of Chapter updated
		4.11.2 Statistic Commands Added Clear Statistics for all Measures command
		4.13.5 Function Operands Updated arguments of Operand 2 command
		4.15.5 Getting Eye Measurement Results Updated Get Statistic Value of Parameter command
		4.16.1 Common Mask Test Commands Removed space from Mask:Testing mnemonics
		4.16.4 Automask Commands Removed space from Automask:BuildAMask mnemonics
		4.19.2 User-defined calibration points commands <ol style="list-style-type: none"> Updated header of Set the wavelength value for user-defined calibration point command Updated header of Set the conversion gain value for user-defined calibration point command
		4.20.2 Common Internal Stimuli commands Added Internal Stimuli On command
		4.20.3 PicoScope 9311 Internal Stimuli commands <ol style="list-style-type: none"> Added Internal Stimuli Amplitude command Updated header of Internal Stimuli performance command

		4.20.5 Reflectometer Setup commands <ul style="list-style-type: none">a. Added Start Calibration Rho commandb. Added Start Gain Calibration commandc. Added Correction command
		4.21 Instrument Info Commands New Chapter added
		5 Revision history New Chapter added



Index

A

- Acquire a Channel
 - Attenuator dB 23
 - Attenuator linear/log 23
 - Attenuator ratio 23
 - Attenuator unit 23
- Acquisition commands
 - Acquisition Mode of Channel 25
 - Action when Number of Waveforms reached 25
 - Channel Averaging 25
 - Channel Envelopes 25
 - Channel Record Length 25
 - Number of Waveforms 25
 - Sampling Mode 25
 - Termination of Acquisition 25
 - Type of signal 25
- All-off mode 6

C

- Case-insensitivity 4
- Channels commands
 - Acquire a Channel 13
 - Attenuator dB 13
 - Attenuator linear/log 13
 - Attenuator ratio 13
 - Attenuator unit 13
 - Bandwidth of Channel 13
 - Channels commands 13
 - Display a Channel 13
 - Offset a Channel 13
 - Scale a Channel 13
- Characteristics 6
- Command classification 6
- Command entry rules 4
- Command header 3
- Command messages 2
- Commands 3
- Common Mask Commands
 - Actuate Mask Testing 69
 - Format of Stored Files 69
 - Load User Mask 69
 - Mask Erasing 69
 - Mask Test Actions 69
 - Mask Test Finish Condition 69
 - Number of Failed Samples 69
 - Number of Failed Waveforms 69

- Number of Samples 69
- Number of Waveforms 69
- Signal for Mask Testing 69
- Stored File Name 69
- User Masks File Name 69

- COMRC Object 1, 1
- Concatenation 4

D

- Data-type commands 9
- Display commands
 - Define Trace Screen (for 2YT, Comb2YTXY Formats) 28
 - Define Trace Screen (for 4YT Format) 28
 - Display commands 28
 - Display Format 28
 - Graticule Type 28
 - Persistence Time, seconds (for VarPersist Style) 28
 - Refresh Time, seconds (for VarGrayScal or VColorGrade Styles) 28
 - Reset Display Style 28
 - Select active trace 28
 - Set Display Style 28
 - Source of X Axis for XY Screen 28
 - Source of Y Axis for XY Screen 28
 - Trace mode 28

E

- ExecCommand Method 1
- Execution-type commands 6

F

- Float-type commands 9

G

- Getting Eye Measurement Results, commands for
 - Get Current Value of Parameter 68
 - Get List of Measured Parameters 68
 - Get Statistic Value of Parameter 68
- Getting Mask Results, commands for
 - Get Integrated Results of Mask Test 75
 - Get Number of Samples in Margins of Selected Polygon 75
 - Get Number of Samples in Selected Polygon with Margins Together 75
 - Get Number of Samples in Selected Polygons 75
- Getting Measurement Results, commands for
 - Get Current Value of Parameter 45
 - Get List of Measured Parameters 45

Getting Measurement Results, commands for
 Get Statistic Value of Parameter 45
 Group-off mode 6
 Group-on mode 6
 Group-query mode 6
 GUI commands 10

H

Header commands
 Header 10
 Headers 3

I

Integer-type commands 8

L

Limit Tests commands
 Action 46
 Action If 46
 Center Value 46
 Failure When 46
 File Name 46
 Format of Stored Files 46
 If Measurement Undefined 46
 Limit Test On/Off 46
 Limit Test Termination Condition 46
 Number of Failures 46
 Number of Waveforms 46
 Parameter Activity 46
 Parameter Center Mode 46
 Parameter Delta Mode 46
 Parameter Delta Percentage for User Defined mode 46
 Parameter Delta Value for Standard Deviation mode 46
 Parameter Delta Value for User Defined Mode 46
 Parameter Limit Mode 46
 Upper and Lower Limits of Parameters 46
 Line feeds 4

M

Markers commands
 Marker Sources 37
 Marker Type 37
 Motion of Markers 37
 X position of Marker 37
 Y position of Marker 37
 Mathematics commands
 Constant Value 50

Enable Mathematical Function 50
 Function Operator 50
 Operand 1 50
 Operand 2 50
 Smoothing Parameter 50
 Trend Measurement 50

Measure commands

Base Value (for Marker Method) 38
 Delete all Measures for all Sources 38
 Execute Single Measurement 38
 List of X Inter-Signal Measurements 38
 List of X Measurements 38
 List of Y Inter-Signal Measurements 38
 List of Y Measurements 38
 Margins Definition Mode 38
 Measurement Mode 38
 Measurement Source 38
 Measurement Type 38
 Measurements of Time Domain Signals commands 38
 Percentage of Upper, Middle or Lower Threshold 38
 Position of Left or Right Margin 38
 Position of Upper, Middle or Lower Threshold 38
 Second Source for Inter-Signal Measurements 38
 Slope of Left or Right Margins 38
 Statistic Measurement Mode 38
 Threshold Definition Method 38
 Threshold Units 38
 Thresholds of Left and Right Margin Slopes 38
 Top Value for Marker Method 38
 Top/Base Definition Method 38
 Viewing of Define Parameters 38
 Weight Value 38
 Windows Value 38

Measurement of Spectrum Signals, commands for
 Left and Right Spectrum Peaks 43
 Left and Right Spectrums Margin 43
 Limits Definition Method for Spectrum 43
 List of Spectrum Frequency Measurements 43
 List of Spectrum Magnitude Measurements 43
 Peak Level of Spectrum 43

Messages 2
 Mnemonics 4

O

On/Off Group-type commands 6
 On/Off-type commands 6
 Overview 2

P

PicoScope9000 COM Server 1

Q

Queries 3

Query commands 2

S

Selector-type commands 8

Semicolon 4

Set commands 2

Single-item mode 6

Specifications 6

Standard Mask Commands

Alignment of Signal with Standard Mask 70

Enable Margins 70

Get List of Masks 70

Get List of Standards 70

Margins Value 70

Select Standard 70

Select Standard Mask 70

System commands

Clear Display 11

Recall Default Setup 11

Start Autoscaling 11

Start Cycle Acquisition 11

Start Single Acquisition / Stop Acquisition 11

System commands 11

T

Timebase commands

Acquire a Channel 17

Attenuator dB 17

Attenuator linear/log 17

Attenuator ratio 17

Attenuator unit 17

Bandwidth of Channel 17

Display a Channel 17

Offset a Channel 17

Scale a Channel 17

Timebase Commands 17

Trigger Commands

Bandwidth of Channel 20

Display a Channel 20

Offset a Channel 20

Scale a Channel 20

U

Utilities commands

Autocalibration Period 77

Start Autocalibration of Channels 77

Start Autocalibration of Timebase 77

Temperature Deviation 77

When to Begin Autocalibration 77

W

Waveforms commands

Get Number of Points in Waveform 80

Get Waveform Data 80

Get X-axis Origin 80

Get X-axis Step 80

Get X-axis Unit 80

Get Y-axis Unit 80

Spectrum Format 80

Waveform Source 80

White space 4

Work with Disk commands

File Name 33

File Name Mode 33

Format of stored files 33

Load 33

Save to Disk 33

Select Memory for loading signal from disk 33

Source for saving to file 33

Work with Memo Zones (M1, M2, M3, M4) commands

Memory Display 31

Save into Memory 31

Select Memory for Saving 31

Source for storing into Memory 31

Work with Setups commands

Name of Custom Setup File 35

Recall Custom Setup 35

Recall Factory Setup 35

Recall Power-Off Setup 35

Save Custom Setup 35

Save Setup as Default 35

UK headquarters:

Pico Technology
James House
Colmworth Business Park
St. Neots
Cambridgeshire
PE19 8YP
United Kingdom

Tel: +44 (0) 1480 396 395
Fax: +44 (0) 1480 396 296

sales@picotech.com
support@picotech.com

www.picotech.com

USA headquarters:

Pico Technology
320 N Glenwood Blvd
Tyler
Texas 75702
United States

Tel: +1 800 591 2796
Fax: +1 620 272 0981