

**metrix**

## Virtual Digital Oscilloscopes

# mtx | 052B(W)

2-channel, 150 MHz, USB, Ethernet, (WiFi option)

# mtx | 054B(W)

4-channel, 150 MHz, USB, Ethernet, (WiFi option)

# mtx | 052C(W)

2-channel, 200 MHz, USB, Ethernet, (WiFi option)

# mtx | 054C(W)

4-channel, 200 MHz, USB, Ethernet, (WiFi option)

### Remote Programming



**metrix**

Pôle Test et Mesure de CHAUVIN-ARNOUX  
Parc des Glaisins - 6, avenue du Pré de Challes  
F - 74940 ANNECY-LE-VIEUX  
Tél. +33 (0)4.50.64.22.22 - Fax +33 (0)4.50.64.22.00

# Contents

page

<b>Introduction</b> .....	<b>4</b>
Presentation .....	4
Warning .....	4
Connection of the oscilloscope .....	4
<i>Connexion via « USB »</i> .....	4
<i>Connection via « ETHERNET »</i> .....	5
<i>Crossed cable</i> .....	5
Rear panel .....	5
Programming convention.....	6
Command syntax.....	6
<i>Common commands</i> .....	6
<i>Specific commands</i> .....	6
<i>Key words</i> .....	6
<i>Separators</i> .....	6
<i>Parameter format</i> .....	7
<i>Recall</i> .....	7
<i>Terminator</i> .....	8
<i>Response</i> .....	8
<b>“Oscilloscope” Mode</b> .....	<b>9</b>
Vertical .....	9
<i>Display</i> .....	9
<i>Sensitivity / Coupling</i> .....	9
<i>Function definition</i> .....	10
<i>Vertical scale</i> .....	11
Trigger.....	11
<i>Trigger main source</i> .....	12
<i>Trigger Auxiliary Source</i> .....	16
<i>Trigger mode / Automatic mode</i> .....	18
<i>Single mode</i> .....	18
Horizontal .....	18
<i>Acquisition min/max</i> .....	18
<i>Average</i> .....	18
<i>Zoom</i> .....	19
<i>FFT</i> .....	19
<i>Time base</i> .....	20
Display.....	20
<i>Display mode</i> .....	20
<i>Oscilloscope / XY</i> .....	21
<i>XY mode definition</i> .....	21
<i>Analog persistence</i> .....	21
Measure.....	23
<i>Reference</i> .....	23
<i>Measure query</i> .....	23
<i>Phase measure</i> .....	26
<i>Manual measurement</i> .....	26
<i>Phase manual measurement</i> .....	27
<i>Unattached cursors</i> .....	27
Memory.....	28
<i>Trace 1 → Ref 1</i> .....	28
<i>Trace 2 → Ref 2</i> .....	28
<i>Trace 3 → Ref 3</i> .....	28
<i>Trace 4 → Ref 4</i> .....	28
<i>Trace</i> .....	28

## Remote programming

---

<i>Configuration</i> .....	31
Util .....	32
<i>Files</i> .....	32
<i>I/O Port Configuration</i> .....	33
<i>Configuration</i> .....	34
<i>RUN/STOP</i> .....	35
<i>Autoset</i> .....	36
Recording time .....	37
Trigger .....	37
<b>« Harmonic Analyzer » Mode</b> .....	<b>39</b>
Measurement reference .....	39
<b>Help and error</b> .....	<b>39</b>
Help .....	39
Error .....	39
* <i>Command error: (-199 to -100)</i> .....	40
* <i>Execution errors: (-299 to -200)</i> .....	40
* <i>Specific instrument errors: (-399 to -300)</i> .....	40
* <i>Query errors: (-499 to -400)</i> .....	40
<b>IEEE 488.2 common commands</b> .....	<b>41</b>
Introduction .....	41
Events and status management .....	41
The Registers .....	41
<i>Status registers</i> .....	42
<i>Service request mask register</i> .....	42
<i>Event register</i> .....	42
<i>Event mask register</i> .....	43
<i>Event mask register</i> .....	44
<i>Event register</i> .....	44
<i>Service demand mask register</i> .....	45
<i>Status register</i> .....	46
<i>IEEE 488.2 Common commands</i> .....	47
<i>Specific commands</i> .....	47
<b>SCPI command Index</b> .....	<b>48</b>

## Introduction

### Presentation

The programming instructions comply with standard IEEE488.2, protocol SCPI.

They provide the user with the possibility of checking the instrument remotely from simple standard controls.

Communication between a controller and a generator enables users to:

- Configure the instrument
- Perform measurements campaign
- Transfer files

### Warning



- The coherence of the screens of SCOPEin@BOX is not ensured when sending the remote controls.

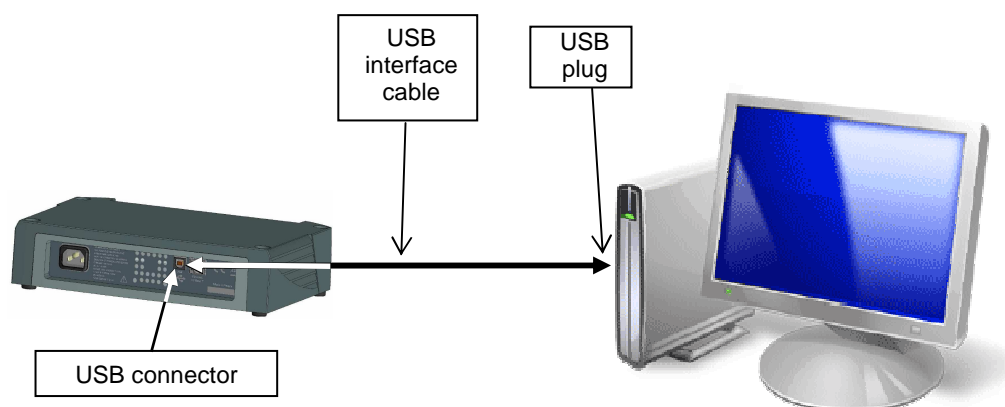
- When entering or exiting SCOPEin@BOX, the instrument is configured in saving energy mode; thus, it is necessary to select first of all **SYST: POW: OFF OFF** to make the oscilloscope operational.

### Connection of the oscilloscope

The oscilloscope can be remotely programmed using a computer (PC) with the programming kit. The dialog between the instrument and the PC can be realized through the RS232 link, or through an ETHERNET link.

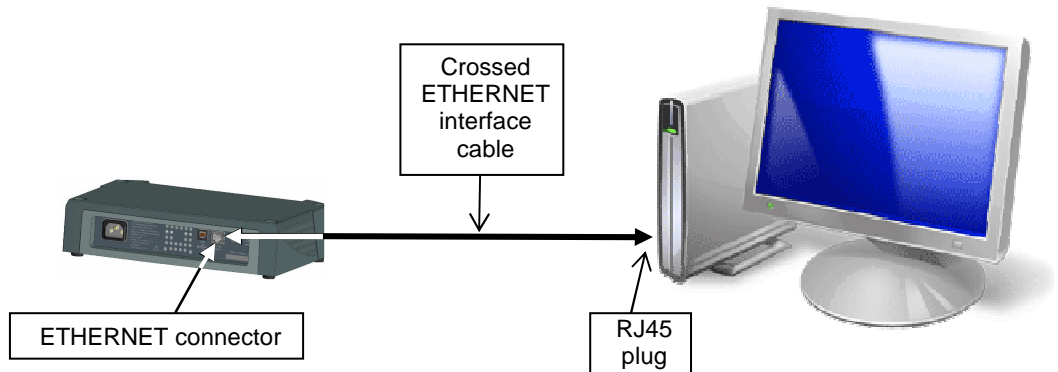
#### Connexion via « USB »

- Connect the cord to the « USB » input which is at the back of the instrument.
- Plug in the USB connector to one of the « USB » inputs of the PC. The USB peripheral is recognized by the PC ; if not, unplug the oscilloscope and install the SCOPEin@BOX application which will install the corresponding driver.
- The connection is established through the USB identifiers of the oscilloscope, which are :
  - OX2177 for the VID (Vendor ID) and
  - OX1002 for the PID (Product ID)

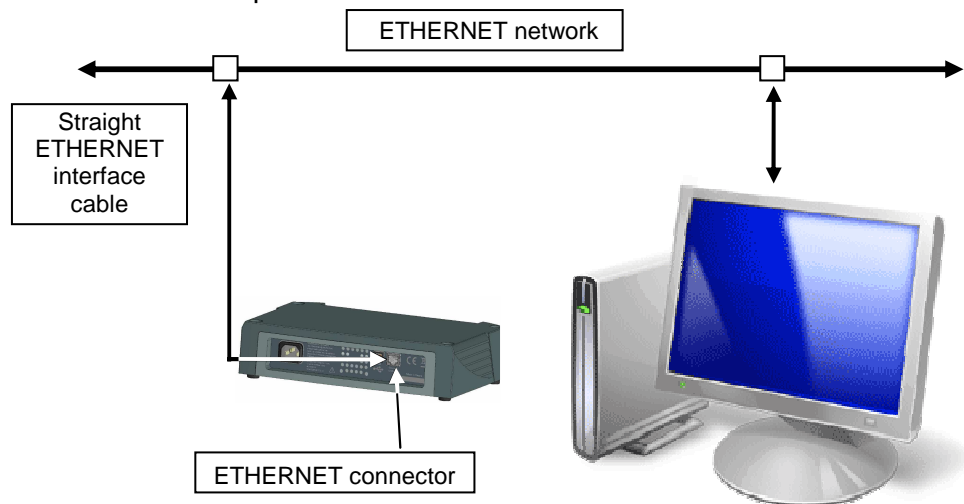


**Connection via  
« ETHERNET »**

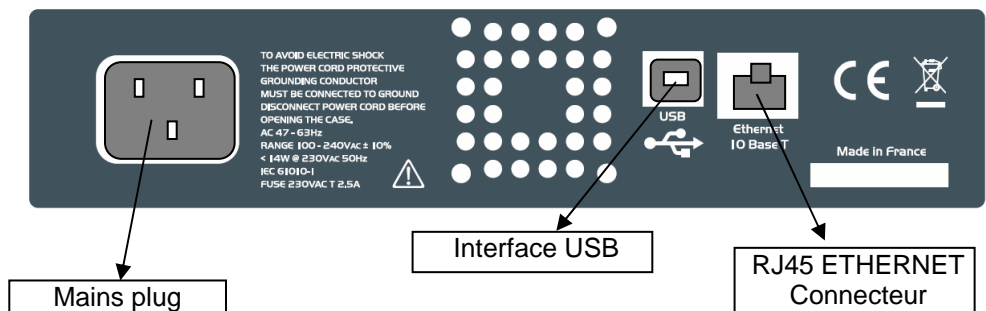
- Crossed cable**
- Connect the crossed ETHERNET interface cable directly to the PC.
  - Set the connection with a terminal (Port TELNET : 23) to the IP address which has been defined in the scope.



- Straight cable**
- Connect the oscilloscope to the PC network through a Hub with the straight ETHERNET interface cable.
  - Connect a terminal (TELNET Port : 23) to the IP address defined on the oscilloscope.



**Rear panel**



**Programming convention**

**Tree structure**

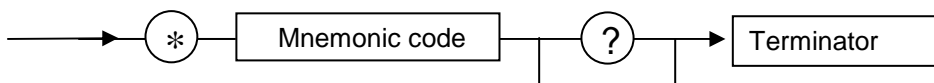
The command tree diagram includes all the commands specific to the instrument.

Common commands (standard IEEE 488.2) are listed separately, since they do not affect the position of the syntax analyzer in the tree.

When the terminator <NL> is sent to the instrument, the analyzer is positioned at root level.

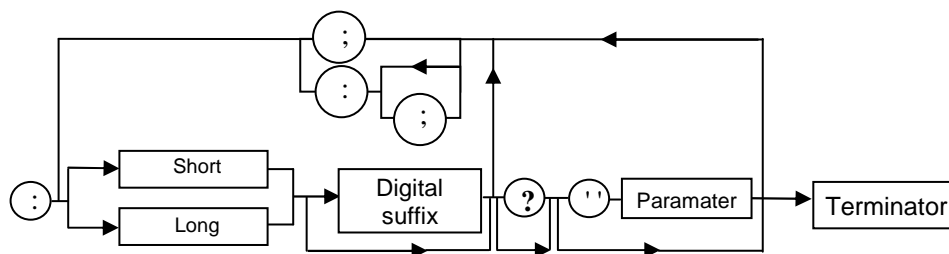
When the analyzer is in a directory, both separators ' ; : ' are necessary to return to the root.

**Command syntax**



**Common commands**

**Specific commands**



**Key words**

The brackets ([ ]) are used to frame a keyword which is optional during programming; i.e. the instrument will execute the command whether the keyword is optional or not. Uppercase and lowercase are used to differentiate the short form of the keyword (uppercase letters) and the long form (whole word).

The instrument accepts the uppercase or lowercase letters without distinction.



**DISP:TRAC:STAT 1** is equivalent to **DISPLAY:WINDOW:TRACE:STATE 1**

**Parameters**

The parameters, if any, are separated from the keyword by a space (' '). A command can accept parameters of a defined type, a literal expression or a combination of both.

< >	The defined-types are marked by the opposite characters.
( [ ] )	The brackets ([ ]) mean that the parameters are optional.
{ }	The accolades define the list of parameters allowed.
(   )	The vertical bar (   ) may be read as an "or", it is used to separate the various possible parameters.

**Separators**

' ; '	descends in the next directory or returns under the root, if preceded by a ' ; '.
' ; '	separates two commands in the same directory or marks the end of a directory command, enabling a return to the root level by adding the separator ' ; '.
' '	(space) separates the keyword from the following parameter.
' ; '	separates one parameter from the other.

**Parameter format** The parameters can be key words, numeric values, character chains or numeric expressions.

**Key words** These parameters have two forms of call, as for the instructions : the shortened form (in capital letter) and the whole form (shortened form plus complement into small letter).

Thus, for certain commands, the parameters are the following :

- **ON, OFF** corresponding to the boolean values (1,0)
- **EDGE, PULse, DELay, EVENT** or **TV** for the trigger modes

**Numeric values** They have several possible formats :

**NRf** (flexible Numeric Representation).  
In the case of physical quantity, these numbers can be followed of a sub-multiple or a multiple of its unit.

**Recall** The interpreter does not make any difference between capital and small letters.

*Example* : 1 micro second can be written either 1µs or 0.000001, 1e-6s, 1E-3ms ...

This parameter can also be replaced by the following key words :

- **MAXimum, MINimum** to get extreme values of the parameter
- **UP, DOWN** to get the value following or preceding the current status of the parameter

**Units**

<b>V</b>	Volt (Voltage)
<b>S</b>	Second (Time)
<b>PCT</b>	Percent (Percentage)
<b>Hz</b>	Hertz (Frequency)
<b>MHz</b>	Mega-Hertz (Frequency)
<b>F</b>	Farad (Capacitance)
<b>OHM</b>	Ohm (Resistance)
<b>DEG</b>	Degree Celsius

**Multiples and sub-multiples**

<b>MA</b>	Mega: 10 <sup>+6</sup>
<b>K</b>	Kilo: 10 <sup>+3</sup>
<b>M</b>	Milli: 10 <sup>-3</sup>
<b>U</b>	Micro: 10 <sup>-6</sup>
<b>N</b>	Nano: 10 <sup>-9</sup>
<b>P</b>	Pico: 10 <sup>-12</sup>

**NR1** The parameter is a signed whole number.

*Example* : 10

**NR2** The parameter is a signed real without exponent.

*Example* : 10.1

**NR3** The parameter is a signed real expressed with a mantissa and a signed exponent.

*Example* : 10.1e-3

<b>Chains of Characters</b>	They are continuations of letters and figures framed by quotation marks " " .
<b>Terminator</b>	
<b>&lt;NL&gt;</b>	<b>&lt;NL&gt;</b> is a general term for a terminator. <b>NL</b> is the character CR (code ASCII 13 or 0x0D). A line of command should not exceed 80 characters; it ends with a terminator.
<b>Response</b>	
	The response can be made up of several elements separated between them by a comma ', '. The last element is followed by the terminator < NL > .  The data are of several natures :
<b>Key words</b>	They are the same ones as those used in parameter, but here, only the shortened form is returned.
<b>Numeric Values</b>	They have three possible formats : NR1, NR2 and NR3.
<b>Chains of characters</b>	There is no difference compared to the parameters. If the chain contains a key word, it is returned in shortened form.



## Detailed description of commands

### « Oscilloscope » Mode

---

#### Vertical

##### Display

DISPlay[:WINDow] (Command/Query)  
:TRACe:STATe{[1]|2|3|4} The command **DISP:TRAC:STAT{[1]|2|3|4} <1|0|ON|OFF>** validates or devalidates the selected signal.

To the question **DISP:TRAC:STAT{[1]|2|3|4}?**, the instrument returns the validation status of the selected signal.

---

##### Sensitivity / Coupling

[SENSe]:VOLTage (Command/Query)  
{[1]|2|3|4}[:DC]:RANGe (Command/Query)  
:PTPeak The command **VOLT{[1]|2|3|4}:RANG:PTP <sensitivity|MAX|MIN|UP|DOWN>** sets the full screen vertical sensitivity of the selected channel.

**<sensitivity>** is a value in **NRf** format, it may be followed by a submultiple and by the unit (V).

By default the value is expressed in volt.

It expresses the total dynamics of the screen (8 divisions) with the selected sensitivity.

 If 10mV/div is the sensitivity displayed in the channel parameters, then the **<sensitivity>** parameter = 8 x 10 mV/div.

To the question **VOLT{[1]|2|3|4}:RANG:PTP?**, the instrument returns the full screen vertical sensitivity of the selected channel.

Response format: <sensitivity><NL>

value in format **<NR3>** expressed in volt.

INPut{[1]|2|3|4}:COUPling (Command/Query)

The command **INP{[1]|2|3|4}:COUP <AC|DC|GROund>** selects the coupling of the selected channel.

To the question **INP{[1]|2|3|4}:COUP?**, the instrument returns the coupling of the selected channel.

<p>[SENSe]:BANDwidth {[1] 2 3 4}[:RESolution]</p>	<p>(Command/Query)</p> <p>The command <b>BAND{[1] 2 3 4} &lt;band&gt;</b> selects a bandwidth limit on the selected channel.</p> <p><b>&lt;band&gt;</b> is a value at format <b>NRF</b>. It can be followed or not of a sub-multiple of the unit (Hz). By default, the value is expressed in Hz. It corresponds to the frequency of the filter cut.</p> <p>Following filters are available : (no limit) ; 5 kHz ; 1,5 MHz ; 15 MHz.</p> <p>To the question <b>BAND{[1] 2 3 4}?</b>, the instrument returns the frequency cut of the current bandwidth limit.</p>
<p>[SENSe]:BANDwidth {[1] 2 3 4}[:RESolution] :AUTO</p>	<p>(Command/Query)</p> <p>The command <b>BAND{[1] 2 3 4}:AUTO &lt;1 0 ON OFF&gt;</b> validates or devalidates the application of the 15 MHz bandwidth limit on the selected channel.</p> <p>To the question <b>BAND{[1] 2 3 4}:AUTO?</b>, the instrument returns the activation status of the bandwidth limit on the selected channel.</p> <p>If no bandwidth limit is activated, the instrument returns 1, if not 0.</p>

---

### Function definition

<p>CALCulate:MATH {[1] 2 3 4}[:EXPRession] [:DEFine]</p>	<p>(Command/Query)</p> <p>The command <b>CALC:MATH{[1] 2 3 4} &lt;(function)&gt;</b> defines and activates the mathematical function of the selected signal.</p> <p><b>&lt;(function)&gt;</b> is the definition of the mathematical function.</p> <p>(ch1 - ch2) subtracts the channel 1 from channel 2.</p> <p>To the question <b>CALC:MATH{[1] 2 3 4}?</b>, the instrument returns the mathematical function of the selected signal.</p>
<p>CALCulate:MATH {[1] 2 3 4}[:EXPRession]: DELeTe</p>	<p>(Command)</p> <p>The command <b>CALC:MATH{[1] 2 3 4}:DEL</b> deletes the mathematical function of the selected signal.</p>
<p>MMEMory:STORe:MACRo</p>	<p>(Command)</p> <p>The command <b>MMEM:STOR:MACR &lt;INT{1 2 3 4}&gt;,&lt;"file.FCT"&gt;</b> saves the mathematical function of the selected signal, in the internal memory, in a file ".FCT".</p> <p><b>&lt;"file.FCT"&gt;</b> is composed of 15 letters maximum, and is followed by a dot and a FCT extension.</p>
<p>MMEMory:LOAD:MACRo</p>	<p>(Command)</p> <p>The command <b>MMEM:LOAD:MACR &lt;INT{1 2 3 4}&gt;,&lt;"file.FCT"&gt;</b> loads a mathematical function, from a file ".FCT" of the internal memory on the selected signal.</p> <p><b>&lt;"file.FCT"&gt;</b> is composed of an 15-letter name maximum, followed by a dot and a FCT extension.</p>

**Vertical scale**

DISPlay[:WINDow] (Command/Query)  
 :TRACe:Y[:SCALe]  
 :PDIVision{[1]|2|3|4} The command **DISP:TRAC:Y:PDIV{[1]|2|3|4} <scale|MAX|MIN>** sets the value of the probe coefficient for the selected signal.

**<scale>** is a value at **NRf** format.

To the question **DISP:TRAC:Y:PDIV{[1]|2|3|4}?**, the instrument returns the value of the probe coefficient for the selected signal.

DISPlay[:WINDow] (Command/Query)  
 :TRACe:Y:LABel{[1]|2|3|4} The command **DISP:TRAC:Y:LAB{[1]|2|3|4} <"label">** determines the unit of the selected signal.

The unit is chosen among the upper-case letters of the alphabet (A to Z), and is composed of a name up to 3 letters.

To the question **DISP:TRAC:Y:LAB{[1]|2|3|4}?**, the instrument returns the unit of the selected signal.

[SENSe]:VOLTage (Command/Query)  
 {[1]|2|3|4]:DC]  
 :RANGe:OFFSet The command **VOLT{[1]|2|3|4}:RANG:OFFS <offset|MAX|MIN| UP|DOWN>** sets the vertical offset of the selected signal.

**<offset>** is a value in format **NRf**, it can be then followed or not by a submultiple and by the unit (V).

By default this value is expressed in volt.

To the question **VOLT{[1]|2|3|4}:RANG:OFFS?**, the instrument returns the vertical offset of the selected signal.

*Response format:* <offset><NL>


Value in the **<NR3>** format expressed in volt.

**Trigger**

The apparatus has several trigger modes which are:

- Sequence 1: Trigger on edge (EDGE)
- Sequence 2: Trigger on pulse width (PULse)
- Sequence 3: Delayed trigger (DELay)
- Sequence 4: Delayed trigger by event counting (EVENT)
- Sequence 5: Trigger on video signals such as television type (TV)
- Sequence 6: Trigger on line (LINE)
- Sequence 7: Standard triggering of Recorder mode.
- Sequence 8: The "Fault Capture" triggering of the Recorder mode.

Standard SCPI allows the use of these various trigger modes thanks to the key word SEQUENCE. The index of this key word makes it possible to identify these modes. To simplify the programming, only two sequences are defined :

 *Example :* in PULse mode, the trigger source of the instrument is programmed in entering the command **TRIG:SEQ2:SOUR INT1**. For the other modes, either the command **TRIG:SEQ1:SOUR INT1**, or **TRIG:SEQ:SOUR INT1** or **TRIG:SOUR INT1** can be entered.

**Trigger main source**

TRIGger[:SEQuence {1|2|3|4|5|6|7|8}]:DEFine? (Interrogation)  
 To the question TRIG:DEF? the instrument returns the description of the indicated sequence :

SEQuence1: EDGE  
 SEQuence2: PULse  
 SEQuence3: DELay  
 SEQuence4: EVENt  
 SEQuence5: TV  
 SEQuence6: LINE  
 SEQuence7: RECorder  
 SEQuence8: CAPTure


TRIGger[:SEQuence {1|2|3|4|5}]:SOURce (Command/Query)  
 The command **TRIG:SOUR <INTernal{1|2|3|4}>** determines the main trigger source of the instrument in the current sequence.

**INTernal{1|2|3|4}** corresponds to the 1, 2, 3, 4 channel of the instrument.

To the question **TRIG:SOUR?**, the instrument returns the main trigger source used in the current sequence.

TRIGger[:SEQuence {1|2|3|4}]:COUPling (Command/Query)  
 The command **TRIG:COUP <AC|DC>** determines the coupling associated to the main trigger source in the current sequence.


To the question **TRIG:COUP?**, the instrument returns the coupling associated to the main trigger source in the current sequence.

 *The coupling is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEQuence{1|2}]:FILTer:HPASs[:STATe] (Command/Query)  
 The command **TRIG:FILT:HPAS <1|0|ON|OFF>** validates or devalidates the reject of the low frequencies associated to the main trigger source in the current sequence.

- **1|ON**: activates the reject of the low frequencies (LF Reject coupling)
- **0|OFF**: deactivates the reject of the low frequencies; the DC coupling is then activated.

To the question **TRIG:FILT:HPAS?**, the instrument returns the activation status of the low frequencies reject associated to the trigger source in the current sequence.


 *The coupling is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEquence{[1]|2}] (Command/Query)  
:FILTer:LPASs[:STATE]

The command **TRIG:FILT:LPAS <1|0|ON|OFF>** validates or devalidates the reject of the high frequency associated to the trigger source in the current sequence.

- **1|ON**: activates the high frequency reject (HF Reject coupling)
- **0|OFF**: deactivates the high frequency reject; the DC coupling is then activated.

To the question **TRIG:FILT:LPAS?**, the instrument returns the activation status the reject of the high frequency associated to the trigger source in the current sequence.

 *The coupling is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEquence[5]] (Command/Query)  
:VIDeo:FIELd:FORMat  
:LPFRame

The command **TRIG:VID:FIEL:FORM:LPFR <525|625>** selects the standard of the TV frame, on which the trigger is to be realised : 525 or 625 lines.

To the question **TRIG:VID:FIEL:FORM:LPFR?**, the instrument returns the current video frame standard.

TRIGger[:SEquence[5]] (Command/Query)  
:VIDeo:LINE:Select

The command **TRIG:VID:LINE:SEL <line>** selects the line on which it is to be triggered.

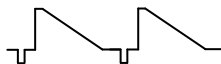
**<line>**: is a value at NR1 format between 1 and the max number of lines of the standard selected with the previous command (525 or 625 lines).

To the question **TRIG:VID:LINE:SEL?**, the instrument returns the number of current line on which it is to be triggered.

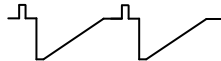
TRIGger[:SEquence[5]] (Command/Query)  
:VIDeo:SSIGnal[:POLarity]

The command **TRIG:VID:SSIG <POSitive|NEGative>** determines the polarity of the video signal.

- **POSitive**: gives following video signal



- **NEGative**: gives following video signal



At the command **TRIG:VID:SSIG?**, the instrument returns the polarity of the video signal.

 *The video signal polarity is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEQuence


{[1]|2|3|4|5|6]:SLOPe

(Command/Query)

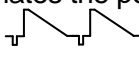
The command **TRIG:SEQ{[1]|2|3|4|5|6}:SLOP <POSitive|NEGative>** determines :

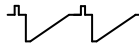
- in **SEQUence2** : the polarity of the pulse

⇒ **POSitive**: positive pulse 

⇒ **NEGative**: negative pulse 

- in **SEQUence5** : determinates the polarity of the video signal :

⇒ **POSitive**: video signal: 

⇒ **NEGative**: video signal: 

- **In other sequences**: determinates the trigger front of the main source:

To the question **TRIG:SEQ{[1]|2|3|4|5|6}:SLOP?**, the instrument returns the polarity trigger front or pulse according to the selected **SEQUence**.



*The trigger front is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEQuence

{[1]|2|3|4]:HYSTeresis

(Command/Query)

The command **TRIG:HYST <hysteresis>** sets the amplitude of the hysteresis used to reject the noise associated to the trigger main source in the current sequence.

**<hysteresis>** is a value at NR1 format taking following values :

- **0**: no noise reject, hysteresis is about 0.5 div.
- **1**: noise reject activated, hysteresis is about 1 div.
- **3**: noise reject activated, hysteresis is about 3 div.

To the question **TRIG:HYST?**, the instrument returns the amplitude of the hysteresis used to reject the noise associated to the trigger main source in the current sequence.



*The noise reject is specific to each channel INT{1|2|3|4}.*

TRIGger[:SEQuence

{[1]|2|3|4|5} :LEVel

(Command/Query)

The command **TRIG:LEV <level|MAX|MIN|UP|DOWN>** sets the trigger level of the main source in **SEQUence1**.

**<level>** is a value in format **NRf**, it may be followed by a submultiple and by the unit (V).

By default the value is expressed in volt.

To the question **TRIG:LEV?**, the instrument returns the trigger level of the main source in SEQUENCE1.

Response format: <level><NL>

value in format <NR3> expressed in volt.



*The trigger level is specific to each channel INT{1|2|3|4}.*


- TRIGger:SEQuence{2|3}  
:DELaY (Command/Query)  
The command **TRIG:SEQ{2|3}:DEL <time|MAX|MIN|UP|DOWN>**:
- In **SEQuence2**: sets the pulse time
  - In **SEQuence3**: sets the trigger delay on the main source (trig-after-delay)
- <time>** is a value in format **<NRf>**, it may be followed by a submultiple and by the unit (s).  
By default the value is expressed in second.
- To the question **TRIG:SEQ{[2]|3}:DEL?**, the instrument returns the trigger delay of the main source or the pulse time according to the selected SEQuence.
- Response format:*           <time><NL>  
value in format **<NR3>** expressed in second.
- To the question **TRIG:DEL?**, the instrument returns the type of pulse time.
- 
- TRIGger[:SEQuence[2]]  
:TYPE (Command/Query)  
The command **TRIG:TYP <EQUate|SUPERior|INFerior>** selects the type of trigger on the pulse width to be used by the trigger.
- **EQUate**: trigger for pulses, the duration of which is equal to the duration set by the command **TRIG:SEQ2:DEL**
  - **SUPERior**: trigger for pulses of superior duration
  - **INFerior**: trigger for pulses of inferior duration
- To the question **TRIG:TYP?**, the instrument returns the type of trigger on the current pulse width.
- 
- TRIGger[:SEQuence  
[1]]2|3|4|5|6]:HOLDoff (Command/Query)  
The command **TRIG:HOLD <time|MAX|MIN|UP|DOWN>** sets the inhibition time of the trigger (Holdoff).
- <time>** is a value in format **<NRf>**, it may be followed by a submultiple and by the unit (s).  
By default the value is expressed in second.
- To the question **TRIG:HOLD?**, the instrument returns the trigger Holdoff time.
- Response format:*           <time><NL>  
value in format **<NR3>** expressed in second.
- 
- TRIGger[:SEQuence[4]]  
:ECOunt (Command/Query)  
The command **TRIG:ECO <count|MAX|MIN|UP|DOWN>** sets the number of events used in the trigger mode delayed by count.
- <count>** is the number of events, it is a value in format NR1 from 3 to 16384.
- To the question **TRIG:ECO?**, the instrument returns the number of events to be counted before the trigger.


## Trigger Auxiliary Source

ARM[:SEQuence  
[3]|4]:COUPling (Command/Query)

The command **ARM:COUP <AC|DC>** determines the coupling associated to the trigger auxiliary source.

To the question **ARM:COUP?**, the instrument returns the coupling associated to the trigger auxiliary source.

 *The coupling of the trigger auxiliary source is specific to each channel INT{1|2|3|4}.*

 *If the auxiliary source is also the main source, the change of coupling modifies also that of the principal source and reciprocally.*

ARM[:SEQuence  
[3]|4]:LEVel (Command/Query)

The command **ARM:LEV <level|MAX|MIN|UP|DOWN>** sets the trigger level of the auxiliary source.


**<level>** is a value in format **<NRf>**, it may be followed by a submultiple and by the unit (V).


By default the value is expressed in volt.

To the question **ARM:LEV?**, the instrument returns the trigger level of the auxiliary source.

Response format:           <level><NL>



value in format **<NR3>** expressed in volt.

 *The level of the trigger auxiliary source is specific to each channel INT{1|2|3|4}.*


 *If the auxiliary source is also the main source, the change of level modifies also that of the principal source and reciprocally.*


ARM[:SEQuence  
[3]|4]:SLOPe (Command/Query)

The command **ARM:SLOP <POSitive|NEGative>** determines the trigger front of the auxiliary source.

- **POSitive**: rising front 
- **NEGative**: falling front 

To the question **ARM: SLOP?**, the instrument returns the polarity of the trigger front of the auxiliary source.

 *The front of the trigger auxiliary source is specific to each channel INT{1|2|3|4}.*

 *If the auxiliary source is also the main source, the change of front modifies also that of the principal source and reciprocally.*

ARM[:SEQuence  
[3]|4]:SOURce (Command/Query)

The command **ARM:SOUR <INTernal{1|2|3|4}>** determines the auxiliary trigger source of the instrument.

**INTernal{1|2|3|4}** corresponds to the channel 1, 2, 3, 4 of the instrument.

To the question **ARM:SOUR?**, the instrument returns the used trigger auxiliary source.



ARM[:SEQuence [3]|4]:HYSTerisis (Command/Query)

The command **ARM:HYST <hysteresis>** sets the amplitude of the hysteresis used to reject the noise associated to the trigger auxiliary source.

**<hysteresis>** is a value in format NR1 with following values :

- **0**: no noise rejection, hysteresis is about 0.5 div.
- **1**: noise reject activated, hysteresis is about 1 div.
- **3**: noise reject activated, hysteresis is about 3 div.


To the question **ARM:HYST?**, the instrument returns the amplitude of the hysteresis used for the noise rejection associated to the trigger auxiliary source.


ARM[:SEQuence[3]|4]:FILTer:HPASs[:STATe] (Command/Query)

The command **ARM:FILT:HPAS <1|0|ON|OFF>** validates or devalidates the reject of the low frequencies associated to the trigger auxiliary source.

- **1|ON**: activates the reject of the low frequencies (LF Reject coupling)
- **0|OFF**: deactivates the reject of the low frequencies; the coupling DC is then activated.

To the question **ARM:FILT:HPAS?**, the instrument returns the activation status of the low frequencies reject associated to the trigger auxiliary source.

 *The LF reject of the trigger auxiliary source is specific to the channel INT{1|2|3|4}.*


 *If the auxiliary source is also principal source, the change of LF reject modifies also that of the main source and reciprocally.*


ARM[:SEQuence[3]|4]:FILTer:LPASs[:STATe] (Command/Query)

The command **ARM:FILT:LPAS <1|0|ON|OFF>** validates or devalidates the high frequencies reject associated to the trigger auxiliary source.

- **1|ON**: activates the high frequencies reject (HF Reject coupling)
- **0|OFF**: deactivates the high frequencies reject ; the DC coupling is then activated.

To the question **ARM:FILT:LPAS?**, the instrument returns the activation status of the high frequencies reject associated to the trigger auxiliary source.

 *The HF reject of the trigger auxiliary source is specific to the channel INT{1|2|3|4}.*

 *If the auxiliary source is also the main source, the change of HF reject modifies also that of the main source and reciprocally.*

### Trigger mode / Automatic mode

TRIGger[:SEQuence] (Command/Query)

[1]|2|3|4|5|6]  
:ATRIGger[:STATe]

The command **TRIG:ATRIG <1|0|ON|OFF>**, validates or devalidates the automatic trigger mode.

- **ON|1** activates the automatic trigger mode.
- **OFF|0** activates the trigger mode.

To the question **TRIG:ATRIG?**, the instrument returns the activation status of the automatic trigger mode.

---

### Single mode

INITiate[:IMMediate]:NAME (Command)

The command **INIT:NAME {EDGE|PULse|DELay|EVENTt|TV|LINE|RECOorder|CAPTuRe}** runs an acquisition in single mode.

In the **CAPTuRe** (Recorder) mode, the capture of 100 faults is launched.

---

## Horizontal

### Acquisition min/max

[SENSe]:AVERage:TYPE (Command/Query)

The command **AVER:TYPE <NORMal|ENVELOpe>** validates or devalidates the mode of min/max acquisition.

- **NORMal** devalidates the mode of min/max acquisition.
- **ENVELOpe** validates the mode of min/max acquisition.

To the question **AVER:TYPE?**, the instrument returns the activation status of the mode of min/max acquisition.

---

### Average

[SENSe]:AVERage:COUNt (Command/Query)

The command

**AVER:COUN <number of acquisitions|MAX|MIN|UP|DOWN>** determines the numbers of acquisition bursts necessary to obtain a displayed trace by averaging.

**<number of acquisitions>** is a value in format **NR1**, from values **2, 4, 16 to 64**.

To the question **AVER:COUN?**, the instrument returns the numbers of acquisition bursts necessary to obtain a displayed trace by averaging.



*The trace averaging is possible only if the 'REPETITIVE SIGNAL' option is activated.*

[SENSe]:AVERage[:STATE] (Command/Query)

The command **AVER <1|0|ON|OFF>** validates or devalidates the 'REPETITIVE SIGNAL' function.

- **1|ON**: signal repetitive validated
- **0|OFF**: signal repetitive not validated

To the question **AVER?**, the instrument returns the activation status of the repetitive signal function.



*The averaging is only active when the option 'repetitive signal' is validated.*

---

### Zoom

DISPlay[:WINDow]:TRACe (Command)  
:ZOOM:Horizontal

The command **DISP:TRAC:ZOOM:H <position>, <coeff>** it makes it possible to carry out a horizontal zoom of the trace starting from the indicated position.

**<position>** is an entirety ranging between 0 and 99999; it indicates the point from which the zoom is made (the effective position of the beginning of zoom is such as one can obtain 500 points with the coefficient of provided zoom).

**<coeff>** coefficient of zoom lies between 1 and 100. It is about the coefficient of decimation of the memory :

 *In case of 50, one sample out of 50 can be taken in memory.*

---

### FFT

CALCulate:TRANSform (Command/Query)  
:FREQuency:WINDow

The command **CALC:TRAN:FREQ:WIND<RECTangular|HAMMING|HANNing|BLACKman>** selects the window used for the FFT calculation.

To the question **CALC:TRAN:FREQ:WIND?**, the instrument returns the type of window used for the FFT calculation.

CALCulate:TRANSform (Command/Query)  
:FREQuency[:STATE]

The command **CALC:TRAN:FREQ <1|0|ON|OFF>** activates or not the FFT calculation.

- **1|ON**: activates the FFT calculation
- **0|OFF**: deactivates the FFT calculation

To the question **CALC:TRAN:FREQ?**, the instrument returns the activation status of the FFT calculation.

DISPlay[:WINDow]:TRACe (Command/Query)  
:Y:SPACing

The command **DISP:TRAC:Y:SPAC <LOGarithmic|LINEar>** specifies the type of scale applied to the Y-axis in the frequencial representation.

To the question **DISP:TRAC:Y:SPAC?**, the instrument returns the type of scale applied to the Y-axis.

### Time base

DISPlay[:WINDow] (Command/Query)  
:TRACe:X[:SCALe]  
:PDIVision

The command **DISP:TRAC:X:PDIV <scale|MAX|MIN|UP|DOWN >** sets the time base value.

**<scale>** is a value in format **NRf**, it may be followed or not by a submultiple and the unit (s).

By default the value is expressed in second (s).

Example: to get a time base of 1  $\mu$ s, following values can be entered: **1E-3ms** or **1E-6** or **0.000001s** or **0.000001** or else **1us**.

To the question **DISP:TRAC:X:PDIV?**, the instrument returns the value of the time base.

Response format: <scale><NL>

value in format **<NR3>** expressed in second.

[SENSE]SWEep:OFFSet (Command/Query)  
:TIME

The command **SWE:OFFS:TIME <time|MAX|MIN|UP|DOWN>** settles the horizontal offset of the trace (run-after-delay or postrig).

**<time>** is a signed value in format **<NRf>** ; it may be followed by a submultiple and by the unit (s). By default, it is expressed in second.

To the question **SWE:OFFS:TIME?**, the instrument returns the current run-after-delay.

Response format: <time><NL>

value in format **<NR3>** expressed in second.

## Display

### Display mode

DISPlay[:WINDow]:TRACe (Command/Query)  
:MODE

The command **DISP:TRAC:MODE <NORMAL|ENVELOpe>** selects the display mode.

- **NORMAL** validates the Vecteur display mode.
- **ENVELOpe** devalidates the Envelope display mode.

To the question **DISP:TRAC:MODE?**, the instrument returns the display mode in activation.

### Oscilloscope / XY

DISPlay[:WINDow]:TRACe (Command/Query)  
:FORMat

The command **DISP:TRAC:FORM <A|XY>** selects the display mode of the instrument.

- **A** validates the Oscilloscope display mode :  $Y = f(t)$
- **XY** validates the display mode  $XY : Y = f(x)$

To the question **DISP:TRAC:FORM?**, the instrument returns the display mode in activation.

### XY mode definition

DISPlay[:WINDow]:TRACe (Command/Query)  
:XY:XDEFine

The command **DISP:TRAC:XY:XDEF <INT{1|2|3|4}>** selects the signal positioned on the X-axis.

To the question **DISP:TRAC:XY:XDEF?**, the instrument returns the signal used on the X-axis.

DISPlay[:WINDow]:TRACe (Command/Query)  
:XY:YDEFine

The command **DISP:TRAC:XY:YDEF <INT{1|2|3|4}>** selects the signal positioned on the Y-axis.

To the question **DISP:TRAC:XY:YDEF?**, the instrument returns the used signal on the Y-axis.

### Analog persistence

DISPlay[:WINDow] (Command/Query)  
:PERSistence

The command **DISP:PERS <1|0|ON|OFF>** activates or inhibits the analog persistence mode.

To the question **DISP:PERS?**, the instrument returns the status of activation of the analog persistence.

DISPlay[:WINDow]:TRACe (Query)  
:PERSistence:COLOR?

To the question **DISP:TRAC:PERS:COLOR?** the instrument returns the table of the colors which are used to display the traces in persistence. There are 252 colors. Each color has an RGB-code on 3 octets : red, green and blue.

Response format: it depends on the response format which is specified in the **FORMat[:DATA] <INTEger|ASCii|HEXadecimal|BINary>** command.

- 
- DISPlay[:WINDow]:TRACe (Command/Query)  
:PERsistence:PALEtte The **DISP:TRAC:PERS:PALE <MONOchrom|MULTicolor>** command selects the palette of colors used to display in analog persistence.  
**<MONOchrom>** visualizes the ageing of the points by means of a color gradation of the way.  
**<MULTicolor>** visualizes this ageing with rainbow colors. The warm colors (red) show the biggest events.  
To the question **DISP:TRAC:PERS:PALE?**, the instrument returns the type of current display.
- DISPlay[:WINDow]:TRACe (Command/Query)  
:PERsistence:TIME The command **DISP:TRAC:PERS:TIME <time|INF|MAX|MIN|UP|DOWN>** sets the duration of the display persistence.  
**<time>** is a value in format **<NRf>** not signed; it can be followed or not by a sub-multiple of the unit (s). By default, it is expressed in second.  
**<INF>** : selects the infinite persistence (all the points are kept without ageing).  
To the question **DISP:TRAC:PERS?**, the instrument returns the duration of the display persistence.  
Response format: <time><NL>  
value in format **<NR3>** expressed in second or INF (infinite).
- DISPlay[:WINDow]:TRACe (Command)  
:PERsistence:REFRash The command **DISP:TRAC:PERS:REFR** resets the persistence and erases the trace.

**Measure**

***In analog persistence, automatic measurements are not available. Only manual measurements with unattached cursors can be performed in this display mode.***

**Reference**

DISPlay[:WINDow]:CURSor  
:REFerence (Command/Query)

The command **DISP:CURS:REF <INT{1|2|3|4}>** selects the reference signal for the automatic and manual measurements.

To the question **DISP:CURS:REF?**, the instrument returns the signal used as a reference.

**Measure query**

MEASure:MINimum? (Query)

To the question **MEAS:MIN? <INT{1|2|3|4}>** the instrument returns the minimum value of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:MAXimum? (Query)

To the question **MEAS:MAX? <INT{1|2|3|4}>** the instrument returns the maximum value of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:PTPeak? (Query)

To the question **MEAS:PTP? <INT{1|2|3|4}>** the instrument returns the peak-to-peak value of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:LOW? (Query)

To the question **MEAS:LOW? <INT{1|2|3|4}>** the instrument returns the low level value of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:HIGH? (Query)

To the question **MEAS:HIGH? <INT{1|2|3|4}>** the instrument returns the high level value of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:AMPLitude? (Query)

To the question **MEAS:AMPLitude? <INT{1|2|3|4}>** the instrument returns the amplitude of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:AC? (Query)

To the question **MEAS:AC? <INT{1|2|3|4}>** the instrument returns the RMS Voltage of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:VOLT[:DC]? (Query)

To the question **MEAS:VOLT? <INT{1|2|3|4}>** the instrument returns the average Voltage of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:RISE:OVERshoot? (Query)

To the question **MEAS:RISE:OVER? <INT{1|2|3|4}>** the instrument returns the positive overshoot of the selected signal.

Response format: <measured value><NL>

value in format <NR2> expressed in Percent.

MEASure:FALL:OVERshoot? (Query)

To the question **MEAS:FALL:OVER? <INT{1|2|3|4}>** the instrument returns the negative overshoot of the selected signal.

Response format: <measured value><NL>

value in format <NR2> expressed in Percent.

MEASure:RISE:TIME? (Query)

:MEASure:RTIME?

To the question **MEAS:RISE:TIME? <INT{1|2|3|4}>** the instrument returns the rising time of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in second.

MEASure:FALL:TIME? (Query)

:MEASure:FTIME?

To the question **MEAS:FALL:TIME? <INT{1|2|3|4}>** the instrument returns the falling time of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in second.



MEASure:PWIDth? (Query)

To the question **MEAS:PWID? <INT{1|2|3|4}>** the instrument returns the positive pulse width of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in second.

MEASure:NWIDth? (Query)

To the question **MEAS:NWID? <INT{1|2|3|4}>** the instrument returns the negative pulse width of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in second.

MEASure:PERiod? (Query)

To the question **MEAS:PERiod? <INT{1|2|3|4}>** the instrument returns the period of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in second.

MEASure:FREQuency? (Query)

To the question **MEAS:FREQ? <INT{1|2|3|4}>** the instrument returns the frequency of the selected signal.

Response format: <measured value><NL>

value in format <NR3> expressed in Hertz.

MEASure:PDUTYcycle? (Query)

To the question **MEAS:PDUT? <INT{1|2|3|4}>** the instrument returns the duty cycle of the selected signal.

Response format: <measured value><NL>

value in format <NR2> expressed in Percent.

MEASure:PULse:COUNt? (Query)

To the question **MEAS:PUL:COUN? <INT{1|2|3|4}>** the instrument returns the pulse count on screen of the selected signal.

Response format: <measured value><NL>

value in format <NR2>.

MEASure:SUM? (Query)

To the question **MEAS:SUM? <INT{1|2|3|4}>** the instrument returns the summation of the instantaneous values of the selected signal ( $\Sigma$  elementary areas).

Response format: <measured value><NL>

value in format <NR3> expressed in volt second (Vs).

### Phase measure

MEASure:PHASe? (Query)

To the question **MEAS:PHAS? <INT{1|2|3|4}>,<INT{1|2|3|4}>** the instrument returns the phase of the first selected signal to the second.

Response format: <measured value><NL>

value in format <NR2> expressed in degrees.

---

### Manual measurement

:DISPlay[:WINDow]:CURSor  
:STATE (Command/Query)

The command **DISP:CURS:STAT <1|0|ON|OFF>** activates or inhibits the manual measurements.

- **1|ON:** activates the manual measurements
- **0|OFF:** inhibits the manual measurements

To the question **DISP:CURS:STAT?**, the instrument returns the activation status of the manual measurements.

DISPlay[:WINDow]:CURSor  
:TIME{[1]|2|3}:POSition (Command/Query)

The command **DISP:CURS:TIME{[1]|2|3}:POS <position|MAX|MIN>** sets the horizontal position of the selected manual cursor.

This command acts on the manual cursors represented on the screen by the X-symbol accompanied by an index (1, 2 or  $\phi$ ). The {[1]|2|3} index associated to the TIME key word enable the selection of those cursors.

**<position>** is a value in format **NRf**, it may be followed by a submultiple and the unit (s).

By default, the value is expressed in second.

To the question **DISP:CURS:TIME{[1]|2|3}:POS?**, the instrument returns the horizontal position of the selected manual cursor.

Response format: <position><NL>

value in format <NR3> expressed in second.

DISPlay[:WINDow]:CURSor  
:VOLT{[1]|2|3}:POSition (Query)

To the question **DISP:CURS:VOLT{[1]|2|3}:POS?**, the instrument returns the horizontal position of the selected manual cursor.

This command acts on the manual cursors represented on the screen by the X-symbol accompanied by an index (1, 2 or  $\phi$ ). The {[1]|2|3} index associated to the VOLT key word enable the selection of those cursors.

Response format: <measured value><NL>

value in format <NR3> expressed in volt.

MEASure:CURSor:DTIME? (Query)

To the question **MEAS:CURS:DTIME?**, the instrument returns the time delay between cursors 1 and 2.

Response format: <measured value><NL>  
value in format <NR3> expressed in second.

MEASure:CURSor:DVOLT? (Query)

To the question **MEAS:CURS:DVOLT?**, the instrument returns the difference between cursors 1 and 2.

Response format: <measured value><NL>  
value in format <NR3> expressed in volt.

---

### Phase manual measurement

DISPlay[:WINDow]:CURSor:PHASe:STATe (Command/Query)

The command **DISP:CURS:PHAS:STAT <1|0|ON|OFF>** activates or inhibits the phase manual measurement.

- **1|ON**: activates the manual phase measurement
- **0|OFF**: inhibits the manual phase measurement

To the question **DISP:CURS:PHAS:STAT?**, the instrument returns the activation status of the phase manual measurement.

MEASure:MANual:PHASe? (Query)

To the question **MEAS:MAN:PHAS?**, the instrument returns the phase of  $\varphi$ -cursor in relation to cursors 1 and 2. The difference between the cursor 1 and 2 represents 360°. The cursor 1 equal to 0° and the cursor 2, 360°.

Response format: <measured value><NL>  
value in format <NR2> expressed in degrees.

---

### Unattached cursors

DISPlay[:WINDow]:CURS:AUTO:STATe (Command/Query)

The command **DISP:CURS:AUTO:STAT <1|0|ON|OFF>** activates or inhibits the unattached cursors.

- **ON|1** the cursors 1 and 2 move along the reference signal.
- **OFF|0** the moving of cursors 1 and 2 is free.

To the question **DISP:CURS:AUTO:STAT?**, the instrument returns the activation status of unattached cursor mode.

## Memory

*Trace 1 → Ref 1*

*Trace 2 → Ref 2*

*Trace 3 → Ref 3*

*Trace 4 → Ref 4*

DISPlay[:WINDow]:TRACe: (Command/Query)

REFerence{[1]|2|3|4}:STATe

The command **DISP:TRAC:REF{[1]|2|3|4}:STAT <1|0|ON|OFF>** memorizes or not the selected signal in the "Ref." associated reference memory.

- **ON|1** the selected signal is memorized in the "Ref." associated reference memory.
- **OFF|0** the selected "Ref" reference memory is not active.

A reference memory cannot be offset vertically on the screen.

To the question **DISP:TRAC:REF{[1]|2|3|4}:STAT?**, the instrument returns the activation status of the selected "Refx" reference memory.

### Trace

MMEMory:STORe:TRACe (Command)

The command **MMEM:STOR:TRAC <INT{1|2|3|4} | REF{1|2|3|4}>**, **<"file.TRC"|"file.TXT"|"file.REC"|"file.PER">** saves the signal or the selected reference memory in a file of the internal memory.

- in a file ".TRC" or ".TXT" for the oscilloscope mode
- in a file ".PER" for the persistence oscilloscope mode
- in a file ".REC" for the recorder mode



*In the Recorder mode, all the traces are memorized systematically, whatever the signal selected with the first parameter is.*

MMEMory:LOAD:TRACe (Command)

The command **MMEM:LOAD:TRAC <TRACE{1|2|3|4}>**, **<"file.TRC"|"file.REC"|"file.PER">** loads :

- a trace from a file ".TRC" of the internal memory for the oscilloscope mode
- from a ".PER" file for the persistence
- all the traces of the recorder from a ".REC" file of the internal memory.

TRACe:CATalog (Query)

To the question **TRAC:CAT?**, the instrument returns the list of active signals.



**TRAC:CAT?**

responds <NL> when no signal is active.

responds INT1<NL> when only signal 1 is active

responds INT1,INT3<NL> when the signals 1 and 3 are active.

responds MATH1<NL> when signal1 is active with a math function

responds M1<NL> when signal1 is active with recall of a trace in memory, etc ...

TRACe:LIMit (Command/Query)

The command **TRAC:LIM <abscissa1>,<abscissa2>,<step>** sets the left and right limits, and the step of the data items to be transferred.

**<abscissa1>,<abscissa2>,<step>** are parameters in the format **<NR1>**.

Default value is 0, 49999 and 1.

To the question **TRAC:LIM?**, the instrument returns the left and right limits, and the step of the data items to be transferred.

To ask the N sample, enter the command :

 **TRAC :LIM N,N,0**

TRACe[:DATA] (Query)

To the question **TRAC? <INT1|2|3|4>**, the instrument transfers the selected trace to the computer.

Response format: <block><NL>

**<block>** is a data block, the format of which is given by the commands **FORMat:DINTerchange** and **FORMat[:DATA]**.

The N transferred sample quantity is defined by the command **TRAC:LIM <abscissa1>,<abscissa2>,<step>**. The block contains the value of the N samples coded on 4 bytes as follows (bit 31 = MSB) :

31	24	19	0
Validity		-	samples coded on 20 bits

The validity byte contains 3 useful bits :

31	30	29	28	27	26	25	24
I	O	E	-	-	-	-	-

- with **I** : invalid, the sample is invalid if equal to 1
- O** : old, used in slow mode, this sample is valid
- E** : extrapolated, the sample is the result of an extrapolation if equal to 1.

In analog persistence display, the returned data block is a INTx trace bitmap of 250 X 176 pixels, where each pixel is coded on a character. The total depth memory is coded in this bitmap.

TRACe:BITMAP? (Query)

To the question **TRAC:BITMAP?**, the instrument transfers to the computer the bitmap of the traces displayed on the screen.

 *Only in Oscilloscope mode with « SPO ».*

Response format : <block><NL>

**<block>** is a data block preceded by the heading **#an, n** being the data number and **a** being a figure indicating the number of figures in **n**. The data format is given by the **FORM[:DATA]** command.

In this bitmap of 250 x 176 pixels, each pixel is coded on a character.

## FORMat:DINTerchange (Command/Query)

The command **FORM:DINT <1|0|ON|OFF>** activates or inhibits the transfer of trace in DIF format.

- **ON|1** activates the transfer of trace in the DIF format.
- **OFF|0** the data items of trace transfer are raw.

To the question **FORM:DINT?**, the instrument returns the activation status of the DIF format.

Response format: The DIF format:

```
(DIF (VERsion <year.version>)
DIMension=X (TYPE IMPLicit
SCALe <interval sample>
SIZE <nb samples>
UNITs "S")

DIMension=Y (TYPE EXPLicit
SCALe <step ADC> SIZE 262144
OFFset 393216
UNITs "V")
DATA (CURVe (<data block>))<NL>
```

**<year.version>** is a number in the <NR2> format giving the year of the SCPI standard used and the software version.

☞ : 1999.1 means that the 1999 version SCPI is used. It is the first software version of the remote control management program.

**<interval sample>** is a number in the <NR3> format. It represents the offset value between two samples.

**<nb sample>** is a number in the <NR1> format. It represents the number of samples which will be transferred. It may vary from 1 to 50000.

**<Step ADC>** is a number in the <NR3> format. It represents the spacing value in volt between two resulting values of the digital/analog converter.

**<data block>** is a block containing the samples. These data items are simply the resulting value from the digital/analog converter. This block is in the format specified by the command **FORMat[:DATA]**.

**FORMat[:DATA]** (Command/Query)

The command **FORM <INTEger|ASCii|HEXadecimal|BINary>** selects the data format of trace transfer.

- **INTEger**: The transmitted data are 8-bit unsigned long integers, preceded by the **#an** header. **n** is a number, representing the number of data items to be transmitted. **a** is a digit, giving the number of digits in number **n**.

✂ Transmission for 4 data items (74, 70, 71, 76) is **#14JFGL**

- **ASCii**: Data are transferred in ASCII characters using a <NR1> of 0 to 255 numbering system. Each number is separated by a coma.

✂ Transmission for 4 data items (74, 70, 71, 76) is **74,70,71,76**

- **HEXadecimal**: data are transferred in ASCII characters using a numbering system in base 16 on 8 bits. Each number is preceded by #H and separated by a coma.

✂ Transmission for 4 data (74, 70, 71, 76) is **#H4A,#H46,#H47,#H4C**

- **BINary**: data are transferred in ASCII characters using a numbering system in base 16 on 8 bits. Each number is preceded by #H and separated by a coma.

✂ Transmission for 4 data (74, 70, 71, 76) is **#B1001010,#B1000110,#B1000111,#B1001100**

To the question **FORM?**, the instrument returns the selected format for the transfer of trace.

---

### Configuration

**MMEMory:STORe:STATe** (Command)

The command **MMEM:STOR:STAT <"file.CFG">** saves the configuration, in a file ".CFG" of the internal memory.

**<"file.CFG">** is a name containing 15 letters maximum, which is followed by a dot and a CFG extension.

**MMEMory:LOAD:STATe** (Command)

The command **MMEM:LOAD:STAT <"file.CFG">** loads the configuration from a file ".cfg" of the internal memory.

**<"file.CFG">** is a name containing 15 letters maximum, which is followed by a dot and a CFG extension.

**SYSTem:SET** (Command/Query)

The command **SYST:SET <block>** transfers the configuration from the computer to the instrument.

**<block>** is a number of data items preceded by **#an** header, **n** being the number of data items and **a** being a digit indicating the number of digits giving in **n**.

To the question **SYST:SET?**, the instrument transfers the present configuration to the computer.

Response format: <block> <NL>

## Util

## Files

MMEMory:CATalog? (Query)

To the question **MMEM:CAT?**, the instrument returns the list of present files in the internal memory.

*Response format:* < file number>, 0[,<file list>]

<file number> is in the format **NR1**.

< file list> = <"file">,<type>,<size>

<"file"> is composed of an 15-letter name, followed by a dot and a 3-letter extension.

<type> is

- STAT for configuration file.
- TRAC for a plot file.
- ASC for text file
- BIN for another file.

<size> is in the format **NR1**

MMEMory:DElete (Command)

The command **MMEM:DEL <"file">** deletes a file of the internal memory.

<"file"> is composed of an 15-letter name max., followed by a dot and a 3-letter extension.

MMEMory:DATA (Command/Query)

The command **MMEM:DATA <"file">,<block>** transfers a file from the PC to the instrument.

<"file"> is composed of an 15-letter name max., followed by dot and a 3-letter extension. If the file already exists, it will be erased by the new file.

The text files (".TXT") cannot be imported from the computer to the instrument.

<block> : this is all data contained in the file preceded by the **#an header**.

**n** is the number of data items and **a** is a number indicating the number of digits in **n**.

To the question **MMEM:DATA? <"file">**, the instrument transfers the selected file to the computer.

*Response format:* <block> <NL>



### **I/O Port Configuration**

SYSTem:COMMunicate (Command/Query)

:SOCKet:ADDRess

The command **SYST:COMM:SOCK:ADDR "<IPaddress>"** defines the IP address of the instrument.

**<IPaddress>** is a chain of characters as: ip1.ip2.ip3.ip4, each of the ipX values must be included between 0 & 255.

To the question **SYST:COMM:SOCK:ADDR?** the instrument returns the value of the current IP address.

Response format: <ip1.ip2.ip3.ip4><NL>

#### **Warning !**

- Following to the change of IP address of the apparatus, any connection calling upon the ETHERNET connection will stop. A restarting of the apparatus is necessary to take into account this new address and to establish a new connection.
- If the instrument is programmed through the ETHERNET link, the connection with the PC will stop following to the change of IP address. To go on, it is necessary to start the oscilloscope again and to define its new address on the PC.

SYSTem:COMMunicate (Command/Query)

:SOCKet:GATEway

The command **SYST:COMM:SOCK:GATE "<IPaddress>"** defines the IP address of the gateway network.

**<IPaddress>** is a chain of characters as: ip1.ip2.ip3.ip4, each of the ipX values must be included between 0 & 255.

To the question **SYST:COMM:SOCK:GATE?** the instrument returns the value of the IP address of the current gateway.

Response format: <ip1.ip2.ip3.ip4><NL>

SYSTem:COMMunicate (Command/Query)

:SOCKet:MASK

The command **SYST:COMM:SOCK:MASK "<IPmask>"** defines the subnet mask.

**<IPmask>** is a chain of characters as: ip1.ip2.ip3.ip4, each of the ipX values must be included between 0 & 255.

To the question **SYST:COMM:SOCK:MASK?** the instrument returns the value of the mask of the current subnet mask.

Response format: <ip1.ip2.ip3.ip4><NL>

SYSTem:COMMunicate (Command/Query)

:WIFI

The command **SYST:COMM:WIFI <0|OFF|1|ON>** launches the search for wireless connection WiFi set in the oscilloscope. (The connection setup occurs via the application SCOPEin @ BOX).

In case of failure of the connection, the oscilloscope returns to the ETHERNET cable configuration.

To the question **SYST:COMM:WIFI?** the instrument indicates the current ETHERNET communication mode :

- 1 for WiFi
- 0 for cabled ETHERNET

## Configuration

### SYSTem:AUTOTest (Command/Query)

To the question **SYST:AUTOT?** the instrument returns the result of the starting autotest in the form of a field of 14 bits in hexadecimal (bit at 1 if test OK (value in hexadecimal)).

bit 0	Microprocessor test	bits 4 à 7	1 to 4 channel scaling test
bit 1	RAM test	bits 8 à 11	1 to 4 channel acquisition chain test
bit 2	FPGA test	bit 12	Ethernet test
bit 3	Acquisition memory test	bit 13	Vernier test

Response format : <NR1><NL>

### DEVice:MODE (Command/Query)

The command **DEV:MOD <SCOPE|ANALYSer|RECOder>** selects the principal mode of the instrument.

To the question **DEV:MOD?**, the instrument returns the mode in which it has been configured.

### SYSTem:DATE (Command/Query)

The command **SYST:DATE <year>,<month>,<day>**, sets the date of the instrument.

The possible values are:

0 to 9999 for the **<year>** range (format **<NR1>**)  
 1 to 12 for the **<month>** range (format **<NR1>**)  
 1 to 31 for the **<day>** range (format **<NR1>**)

To the question **SYST:DATE?**, the instrument returns the date.

Response format: < YYYY,MM,DD ><NL>

with **Y** = year, **M** = month, **D** = day.

### SYSTem:TIME (Command/Query)

The command **SYST:TIME <hour>,<minute>,<second>**, sets the time of the instrument.

The possible values are:

0 to 23 for the **<hour>** range (format **<NR1>**)  
 0 to 59 for the **<minute>** range (format **<NR1>**)  
 0 to 59 for the **<second>** range (format **<NR1>**)

To the question **SYST:TIME?**, the instrument returns the hour.

Response format: < HH,MM,SS ><NL>

with **H** = hour, **M** = minute, **S** = second.

### SYSTem:LANGUage (Command/Query)

The command

**SYST:LANG <ENGLISH|FRENch|GERman|SPANish|ITALian>** selects the language of the instrument.

To the question **SYST:LANG?**, the instrument returns the selected language.

SYSTem:RESET (Command)

The command **SYST:RESET** re-initiates the instrument during 40 s about when it is not possible to communicate.

SYSTem:POWer:OFF (Command/Query)

The command **SYST:POW:OFF <0|1|ON|OFF>** sets the time from behind of automatic power off. This mode acts on the powersupply of the entry stage of the oscilloscope.

To the question **SYST:POW:OFF?**, the instrument indicates whether the POWER OFF mode is activated.

---

### **RUN/STOP**

INITiate:CONTinuous:NAME (Command)

The command

**INIT:CONT:NAME{EDGE|PULse|DELay|EVENT|TV|LINE|RECOder|CAPTure },<1|0|ON|OFF>** starts or stops the acquisition in repetitive mode in the indicated trigger mode.

In the **CAPTure** mode, the fault capture in file (Recorder) is used.

ABORt (Command)

The command **ABORt** aborts the acquisition in progress.

- If the instrument is set in the **single** mode, the acquisition is stopped. The instrument stays in the starting status.
- If the instrument is in **continuous** mode, the acquisition in progress is stopped and the following starts.
- If there is no acquisition in progress, this command has no effect.

TRIGger[:SEQuence

{[1]|2|3|4|5|6|7|8}]  
:RUN:STATe

(Command/Query)  
The command **TRIG:RUN:STAT <1|0|ON|OFF>** starts or stops the acquisition.

- **ON|1** acquisition starts.
- **OFF|0** acquisition is stopped.

To the question **TRIG:RUN:STAT?**, the instrument returns the trigger status in SEQuence1.

**Autoset**

AUTOSet:EXEcute *(Command)*

The command **AUTOS:EXE** starts a general autoset on all channels.

AUTOSet:HORIZontal *(Command)*

The command **AUTOS:HORIZ** launches an automatic setting of the channel time base, trigger source, without modifying its vertical settings.

AUTOSet:VERTical *(Command)*

The command **AUTOS:VERT <INT1|2|3|4>** launches a vertical autoset on the selected channel.

TRIGger:LEVel:AUTO *(Command)*

The command **TRIG:LEV:AUTO** searches automatically the main trigger level at 50 % of the dynamic signal.

## « Recording » Mode

---

### Recording time

[SENSe]:SWEep:TIME[?] (Command/Query)

The command **SWE:TIME**<duration|MAX|MIN|UP|DOWN> sets the recording time.

<duration> is a value in **NRf** format and may be followed or not by a sub-multiple of the unit (s).

By default, it is expressed in second (s).

To the question **SWE:TIME?** the instrument returns the recording time.

*Response format:* <duration><NL>

value in the <NR2> followed or not by a sub-multiple (m) and by the unit (s).

---

### Trigger

ARM:SEquence {7|8} (Command/Query)

:SOURce

SOURce the **ARM:Seq{7|8}:SOUR** <AINTernal|TImer> command validates the immediate (AINT) or deferred triggering (TImer).

To the question **ARM:Seq{7|8}:SOUR?**, the device returns AINT or TIM.

TRIGger[:SEquence{[7|8]} (Command/Query)

:LEVel{[1]|2|3|4}

The **TRIG:LEV[1]|2|3|4** <level|MAX|MIN|UP|DOWN> command adjusts the main triggering level on the channel indicated, in RECORDER mode.



<level> is a value in <NRf> format, it can therefore be followed or not by a sub-multiple and the unit (V). By default, the value is expressed in volt.

To the question **TRIG:LEV{[1]|2|3|4}?**, the device returns the main triggering level on the channel indicated.

*Response format:* <level><NL> value in <NR3> format, expressed in volt.



The main trigger level is specific to each channel INT{1|2|3|4}.

- TRIGger[:SEQuence{[7]|8}]  
:AUXLEVel{[1]|2|3|4} (Command/Query)  
The **TRIG:AUXLEV{[1]|2|3|4} <level|MAX|MIN|UP|DOWN>** command adjusts the auxiliary triggering level on the channel indicated, in RECORDER mode.  
**<level>** is a value in <NRf> format, it can therefore be followed or not by a multiple and the unit.  
By default, the value is expressed in volt.  
To the question **TRIG:AUXLEV{[1]|2|3|4}?**, the device returns the auxiliary triggering level on the channel indicated.  
Response format: <threshold><NL> value in <NR3> format, expressed in volt.
-  The auxiliary triggering level is specific to each channel INT{1|2|3|4}.
- ARM[:SEQuence{[7]|8}]  
:DATE (Command/Query)  
The **ARM:DATE <year>,<month>,<day>** command adjusts the date of the deferred start-up.  
**<year >,<month>,<day>** are values in <NR1> format  
To the question **ARM:DATE?**, the device returns the date of the deferred start-up.
- ARM[:SEQuence{[7]|8}]  
:TIME (Command/Query)  
The **ARM:TIME <hour>,<minute>,<second>** command is used to adjust the time of the deferred start-up.  
**<hour>,<minute>,<sec>** are values in <NR1> format  
To the question **ARM:TIME?**, the device returns the date of the deferred start-up.
- TRIGger[:SEQuence{[7]|8}]  
:SLOPe{[1]|2|3|4} (Command/Query)  
**TRIG:SLOP{[1]|2|3|4} <NO|POSitive|NEGative|EITher|WINDow>** selects the trigger type on the indicated channel.  
To the question **TRIG:SLOP{[1]|2|3|4}?**, the instrument returns the present trigger type.
-  The trigger type is specific to each channel INT{1|2|3|4}.



**\* Command error:** (-199 to -100) They indicate that a syntax error has been detected by the syntax analyzer and causes event register bit 5, called CME, CoMmand Error to be set to 1.

-101	:	Invalid character
-103	:	Invalid separator
-104	:	Data type error
-108	:	Parameter not allowed
-109	:	Missing parameter
-111	:	Header separator error
-112	:	Program mnemonic too long
-113	:	Undefined header
-114	:	Header suffix out of range
-121	:	Invalid character in number
-128	:	Numeric data not allowed
-131	:	Invalid suffix
-138	:	Suffix not allowed
-141	:	Invalid character data
-148	:	Character data not allowed
-151	:	Invalid string data
-154	:	String data too long
-171	:	Invalid expression

**\* Execution errors:** (-299 to -200) They indicate that an error has been detected at the moment of command execution and causes event register bit 4, called EXE, Execution Error, to be set to 1.

-200	:	Execution error
-213	:	Init ignored
-221	:	Settings conflict
-222	:	Data out of range
-232	:	Invalid format
-256	:	File name not found
-257	:	File name error

**\* Specific instrument errors:** (-399 to -300) They indicate that an abnormal error has been detected during execution of a task, and causes event register bit 3, called DDE, Device Dependent Error to be set to 1.

-300	:	Device-specific error
-321	:	Out of memory
-350	:	Queue overflow
-360	:	Communication error

**\* Query errors:** (-499 to -400) They indicate that an abnormal error has been detected during execution of a task, and cause event register bit 2, called QYE, QuerY Error, to be set to 1.

-400	:	Query error
------	---	-------------



## IEEE 488.2 common commands

### Introduction

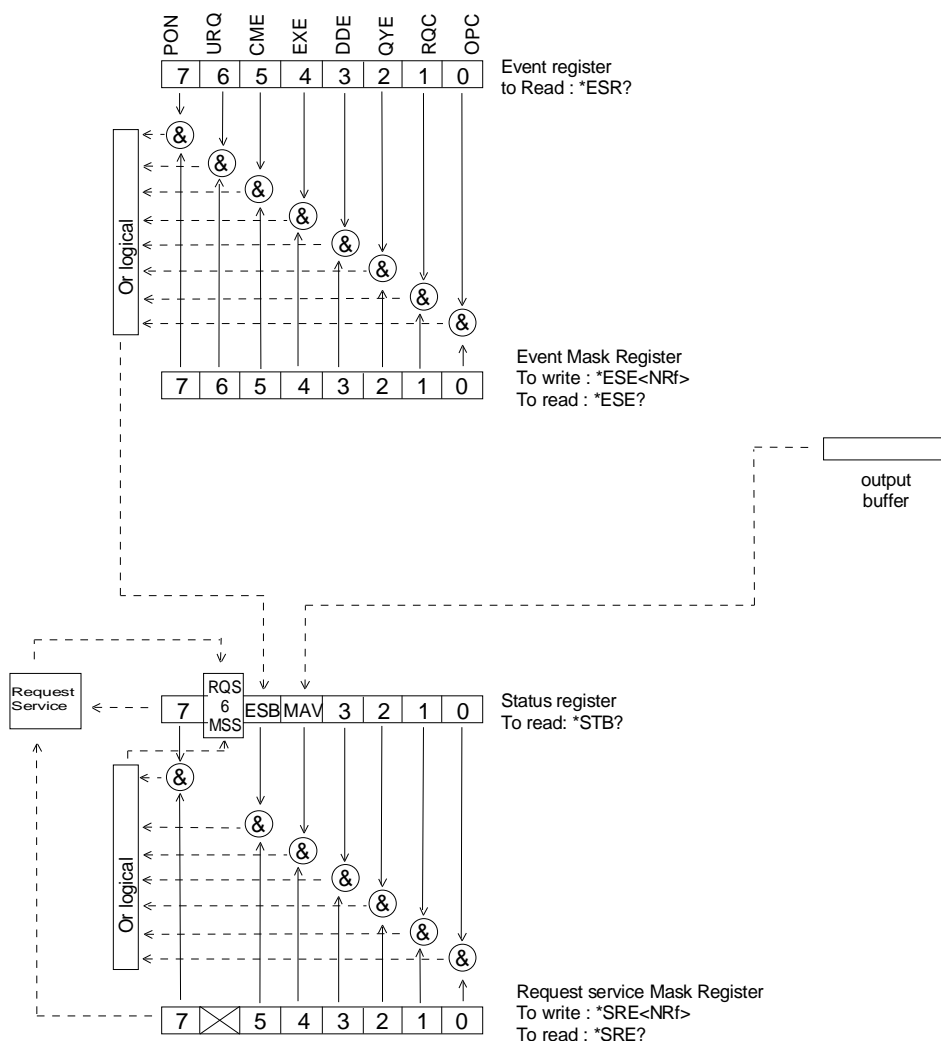
The common commands are defined by the IEEE 488.2 standard. They are operational on all instruments which are specified IEEE 488.2. They command basic functions such as:

- identification,
- reset,
- configuration reading,
- reading of event and status register,
- reset of event and status register.

If a command containing one or several directories has been received, and if a common command has been stacked up, then the instrument stays in this directory and execute normally the commands.

### Events and status management

#### The Registers

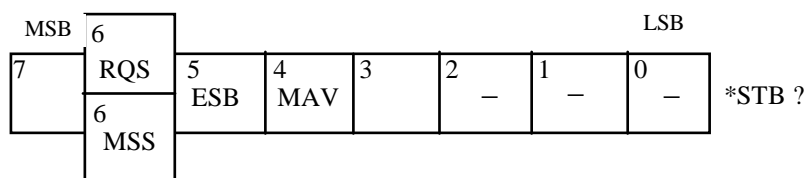


**Status registers**

Reading only → \*STB? common command.

In this case, the (MSS) 6 Bit is returned and remain in the status it was before reading [see §. \*STB (Status Byte)]

The \*CLS common command is reset to zero.

Detailed description**RQS Request Service (6 bit)**

Indicates if the instrument requests a service. The type of COMM used on the instrument does not generate a request, but the byte is accessible in reading. It is reset to 0 after reading and can switch to zero only if the event register is reset to zero (by reading or \*CLS).

**MSS Master Summary Status (6 bit)**

Indicates if the instrument has a reason to request a service. This information is accessible only in reading the status register. (\*STB? command) and stays as it is after the reading.

**ESB Event Status Bit (5 bit)**

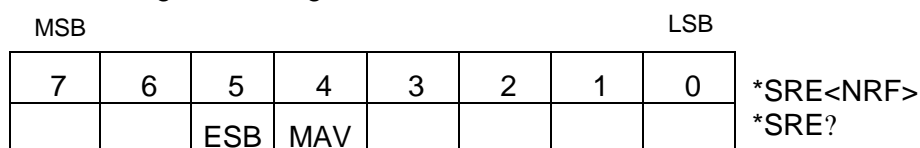
Indicates if at least one of the conditions of the event register is satisfied and not masked.

**MAV Message Available (4 bit)**

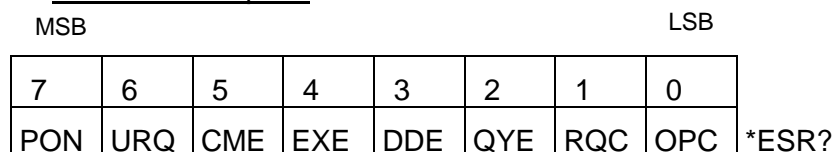
Indicates if at least one response is in the output spooler.

**Service request mask register**

Reading and writing → \*SRE command.

**Event register**

Reading → \*ESR command. Its reading resets to zero.

Detailed description

- PON**      **Power On (7 bit)**  
Not used
  
- URQ**      **User request (6 bit)**  
Not used
  
- CME**      **Command Error (5 bit)**  
A command error has been detected.
  
- EXE**      **Execution Error (4 bit)**  
An error execution has been detected.
  
- DDE**      **Device Dependant Error 3 (bit)**  
An error specific to the instrument has been detected.
  
- QYE**      **Query Error (2 bit)**  
A query error has been detected.
  
- RQC**      **Request Control (1 bit)**  
Always at zero.
  
- OPC**      **Operation Complete (0 bit)**  
All operations running are ended.

---

**Event mask register**

Reading and writing → \*ESE command.

MSB				LSB				
7	6	5	4	3	2	1	0	*ESE<NRF>
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC	*ESE?

## The Commands

**\*CLS** (Command)

(Clear Status)

The common command **\*CLS** reset the status and event register.

**\*ESE** (Command/Query)

(Event Status Enable)

The **\*ESE <mask>** common command positions the status of the event mask.

**<mask>** is a value in format **<NR1>**, from 0 to 255.

A **1** authorises the corresponding bit of the event register to generate an event, while a **0** masks it.

To the question **\*ESE?**, the instrument returns the current content of the event mask register.

Response format: <value><NL>

value in format **<NR1>** from 0 to 255.

---

### Event mask register

MSB				LSB			
7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

**\*ESR?** (Query)

(Event Status Register)

To the question **\*ESR?**, the instrument returns the content of the event register.

Once the register has been read, the content value is reset to zero.

Response format: <value><NL>

value in format **<NR1>** from 0 to 255.

---

### Event register

MSB				LSB			
7	6	5	4	3	2	1	0
PON	URQ	CME	EXE	DDE	QYE	RQC	OPC

**\*IDN?** (Query)

(Identification Number)

To the question **\*IDN?**, the instrument returns the type of instrument and the software version.

Response format:

<instrument>,<firmware version>/<hardware version><NL>

<instrument> Instrument reference

<firmware version> Software version

<hardware version> PCB version

**\*OPC** (Operation Complete) *(Command/Query)*  
 The command **\*OPC** authorises the setting to 1 of the OPC bit in the event register as soon as the current operation is completed. To the question **\*OPC?**, the instrument returns the character ASCII "1" as soon as the current operation is terminated.

**\*RST** (Reset) *(Command)*  
 The command **\*RST** reconfigures the instrument with the factory settings.

**\*SRE** (Service Request Enable) *(Command/Query)*  
 The command **\*SRE <mask>** positions the service request mask register.  
**<mask>** is a value in format **<NR1>**, from 0 to 255.  
 A value of bit at 1 enables the same-rank bit of the status register to request a service (bit of the status register contains 1). A bit value at 0 neutralizes it.

To the question **\*SRE?**, the instrument returns the value of the service demand mask register.

*Response format:*        <value><NL>  
 value in format **<NR1>** from 0 to 255.

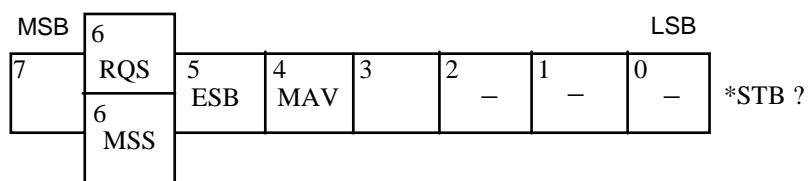
---

**Service demand mask register**

MSB						LSB	
7	6	5	4	3	2	1	0
0	0	ESB	MAV	0	0	0	0

**\*STB?** (Status Byte) *(Query)*  
 To the question **\*STB?** the instrument returns the content of its status register (Status Byte Register).

The bit 6 returned indicates the MSS value (Master Summary Status) (at 1 if the instrument has a reason for requesting a service).  
 Contrary to RQS, it is not reset to zero after reading the status register (RQS is accessible only by a series recognition, and falls to 0 at its end).

**Status register****\*TRG** (Command)

The command **\*TRG** starts an acquisition in the current mode "single" or "continuous".

**\*TST?** (Query)

(Test) To the question **\*TST?**, the instrument returns the status of the autotest procedure.

Response format: <0|1><NL>

- responds 0 when the autotest is successful.
- responds 1 when a problem has been detected.

**\*WAI** (Command)

(Wait) The command **\*WAI** prevents the instrument from performing further commands as long as the current command has not been terminated. This synchronizes the instrument with the application program in progress on the controller.

**Tree structure****IEEE 488.2  
Common commands**

Commands	Functions
*CLS	resets the status and event registers
*ESE	writes event mask
*ESE?	reads event mask
*ESR?	reads event register
*IDN?	reads identifier
*OPC	validates bit OPC
*OPC?	waits till end of execution
*RST	resets
*SRE	writes service request mask
*SRE?	reads service request mask
*STB?	reads status register
*TRG	starts an acquisition in the current mode
*TST?	returns the status of the autoseq procedure
*WAI	commands synchronization

---

**Specific commands** See next pages.

## SCPI command Index

Directory	Commands + parameters	Page	
<b>ABORT</b>		<b>35</b>	
<b>ARM</b>	[[:SEQuence{[3] 4}]:COUPling <AC DC> [:SEQuence{[3] 4}]:COUPling?	<b>16</b>	
	[[:SEQuence{[7] 8}]:DATE <yyyy>,<mm>,<dd> [:SEQuence{[7] 8}]:DATE?	<b>38</b>	
	[[:SEQuence{[3] 4}]:FILTer:HPASs[:STATe] <1 0 ON OFF> [:SEQuence{[3] 4}]:FILTer:HPASs[:STATe]?	<b>17</b>	
	[[:SEQuence{[3] 4}]:FILTer:LPASs[:STATe] <1 0 ON OFF> [:SEQuence{[3] 4}]:FILTer:LPASs[:STATe]?	<b>17</b>	
	[[:SEQuence{[3] 4}]:HYSTeresis <0 1 3> [:SEQuence{[3] 4}]:HYSTeresis?	<b>17</b>	
	[[:SEQuence{[3] 4}]:LEVel <level MAX MIN UP DOWN> [:SEQuence{[3] 4}]:LEVel?	<b>16</b>	
	[[:SEQuence{[3] 4}]:SLOPe <POSitive NEGative> [:SEQuence{[3] 4}]:SLOPe?	<b>16</b>	
	[[:SEQuence{[3] 4}]:SOURce <INTernal{1 2 3 4}> [:SEQuence{[3] 4}]:SOURce?	<b>16</b>	
	:SEQuence{7 8}:SOURce <AINTernal TImer> :SEQuence{7 8}:SOURce?	<b>37</b>	
	[[:SEQuence{[7] 8}]:TIME <hh>,<mm>,<ss> [:SEQuence{[7] 8}]:TIME?	<b>38</b>	
	<b>AUTOSet</b>	:EXEcute	<b>36</b>
		:HORIZ	<b>36</b>
		:VERTical <INT{1 2 3 4}>	<b>36</b>
	<b>CALCulate</b>	:MATH{[1] 2 3 4}[:EXPRession][:DEFine] <(fct)> :MATH{[1] 2 3 4}[:EXPRession][:DEFine]?	<b>10</b>
:MATH{[1] 2 3 4}[:EXPRession]:DELete		<b>10</b>	
:TRANSform:FREQuency[:STATe] <1 0 ON OFF> :TRANSform:FREQuency[:STATe]?		<b>19</b>	
:TRANSform:FREQuency:WINDow <RECTangular HAMMING HANning  BLACKman FLATtop> :TRANSform:FREQuency:WINDow?		<b>19</b>	
<b>DEvice</b>		:MODE <SCOPE ANALYSer RECOder> :MODE?	<b>34</b>
<b>DISPlay</b>	[[:WINDow]:CURSor:AUTO:STATe <1 0 ON OFF> [:WINDow]:CURSor:AUTO:STATe?	<b>27</b>	
	[[:WINDow]:CURSor:PHASe:STATe <1 0 ON OFF> [:WINDow]:CURSor:PHASe:STATe?	<b>27</b>	
	[[:WINDow]:CURSor:REFerence <INT{1 2 3 4}> [:WINDow]:CURSor:REFerence?	<b>23</b>	
	[[:WINDow]:CURSor:STATe <1 0 ON OFF> [:WINDow]:CURSor:STATe?	<b>26</b>	
	[[:WINDow]:CURSor:TIME{[1] 2 3}:POSition <position MAX MIN> [:WINDow]:CURSor:TIME{[1] 2 3}:POSition?	<b>26</b>	
	[[:WINDow]:CURSor:VOLT{[1] 2 3}:POSition?	<b>26</b>	
	[[:WINDow]:MEASure:HARMonic <rang> [:WINDow]:MEASure:HARMonic?	<b>39</b>	
	[[:WINDow]:PERSistence <1 0 ON OFF> [:WINDow]:PERSistence?	<b>21</b>	
	[[:WINDow]:TRACe:FORMat <A XY> [:WINDow]:TRACe:FORMat?	<b>21</b>	
	[[:WINDow]:TRACe:MODE <NORMal ENVELOpe> [:WINDow]:TRACe:MODE?	<b>20</b>	
	[[:WINDow]:TRACe:PERSistence:COLOR?	<b>21</b>	
	[[:WINDow]:TRACe:PERSistence:PALEtte <MONOchrom MULTicolor> [:WINDow]:TRACe:PERSistence:PALEtte?	<b>22</b>	



## Remote programming

Directory	Commands + parameters	Page
	[[:WINDow]:TRACe:PERsistence:TIME <temps INF MAX MIN UP DOWN> [:WINDow]:TRACe:PERsistence:TIME?	22
	[[:WINDow]:TRACe:PERsistence:REFResh	22
	[[:WINDow]:TRACe:REFerence{[1] 2 3 4}:STATe <1 0 ON OFF> [:WINDow]:TRACe:REFerence{[1] 2 3 4}:STATe?	28
	[[:WINDow]:TRACe:STATe{[1] 2 3 4} <1 0 ON OFF> [:WINDow]:TRACe:STATe{[1] 2 3 4}?	9
	[[:WINDow]:TRACe:X[:SCALe]:PDIVision <scale MAX MIN UP DOWN > [:WINDow]:TRACe:X[:SCALe]:PDIVision?	20
	[[:WINDow]:TRACe:XY:XDEFine <INT{1 2 3 4}> [:WINDow]:TRACe:XY:XDEFine?	21
	[[:WINDow]:TRACe:XY:YDEFine <INT{1 2 3 4}> [:WINDow]:TRACe:XY:YDEFine?	21
	[[:WINDow]:TRACe:Y:LABel{[1] 2 3 4} <"label"> [:WINDow]:TRACe:Y:LABel{[1] 2 3 4}?	11
	[[:WINDow]:TRACe:Y[:SCALe]:PDIVision{[1] 2 3 4} <échelle MAX MIN> [:WINDow]:TRACe:Y[:SCALe]:PDIVision{[1] 2 3 4}?	11
	[[:WINDow]:TRACe:Y:SPACing <LOGarithmic LINear> [:WINDow]:TRACe:Y:SPACing?	19
	[[:WINDow]:TRACe :ZOOM:Horizontal	19
<b>FORMat</b>	[[:DATA] <INTeger ASCii HEXadecimal BINary> [:DATA]?	31
	:DINTerchange <1 0 ON OFF> :DINTerchange?	30
<b>HELP</b>	[?] <command> [?]	39
<b>INITiate</b>	:CONTinuous:NAME <EDGE PUL DEL EVEN TV LINE REC CAPT>,<ON OFF 1 0> [:IMMediate]:NAME <EDGE PUL DEL EVEN TV LINE REC CAPT>	35
<b>INPut</b>	{[1] 2 3 4}:COUPling <AC DC GROund> {[1] 2 3 4}:COUPling?	9
<b>MEASure</b>	:AC? <INT{1 2 3 4}>	24
	:AMPLitude? <INT{1 2 3 4}>	24
	:CURSor:DTIME?	27
	:CURSor:DVOLT?	27
	:FALL:OVERshoot? <INT{1 2 3 4}>	24
	:FALL:TIME? <INT{1 2 3 4}>	24
	:FTIME? <INT{1 2 3 4}>	
	:FREQuency? <INT{1 2 3 4}>	25
	:HIGH? <INT{1 2 3 4}>	23
	:LOW? <INT{1 2 3 4}>	23
	:MANual: PHASe?	27
	:MAXimum? <INT{1 2 3 4}>	23
	:MINimum? <INT{1 2 3 4}>	23
	:NWIDth? <INT{1 2 3 4}>	25
	:PDUTYcycle? <INT{1 2 3 4}>	25
	:PERiod? <INT{1 2 3 4}>	25
	:PHASe? <INT{1 2 3 4}>,<INT{1 2 3 4}>	26
	:PTPeak? <INT{1 2 3 4}>	23
	:PULse:COUNT? <INT{1 2 3 4}>	25
	:PWIDth? <INT{1 2 3 4}>	25
	:RISE:OVERshoot? <INT{1 2 3 4}>	24
	:RISE:TIME? <INT{1 2 3 4}>	24
	:RTIME? <INT{1 2 3 4}>	
	:SUM? <INT{1 2 3 4}>	25
	:VOLT[:DC]? <INT{1 2 3 4}>	24
<b>MMEMory</b>	:CATalog?	32
	:DATA <"file">,<block>	32
	:DATA? <"file">	

Directory	Commands + parameters	Page
	:DELeTe <"file">	32
	:LOAD:MACRo <INT{1 2 3 4}>,<"file.FCT">	10
	:LOAD:STATe <"file.CFG">	31
	:LOAD:TRACe <TRACE{1 2 3 4}>,<"file.TRC" "file.REC" "file.PER">	28
	:STORE:MACRo <INT{1 2 3 4}>,<"file.FCT">	10
	:STORE:STATe <"file.CFG">	31
	:STORE:TRACe <INT{1 2 3 4} REF{1 2 3 4}>,<"file.TRC" "file.TXT" "file.REC" "file.PER">	28
<b>[SENSe]</b>	:AVERAge:COUNT <2 4 16 64 MAX MIN UP DOWN>	18
	:AVERAge:COUNT?	
	:AVERAge[:STATe] <1 0 ON OFF>	19
	:AVERAge[:STATe]?	
	:AVERAge:TYPE <NORMAl   ENVELOpe>	18
	:AVERAge:TYPE?	
	:BANDwidth{[1] 2 3 4}[:RESolution{?}]<band>	10
	:BANDwidth{[1] 2 3 4}?	
	:BANDwidth{[1] 2 3 4}[:RESolution]:AUTO <1 0 ON OFF>	10
	:BANDwidth{[1] 2 3 4}[:RESolution]:AUTO?	
	:SWEep:OFFSet:TIME <temps MAX MIN UP DOWN>	20
	:SWEep:OFFSet:TIME?	
	:SWEep:TIME <time MAX MIN UP DOWN>	37
	:SWEep:TIME?	
	:VOLTage{[1] 2 3 4}[:DC]:RANGe:OFFSet <offset MAX MIN UP DOWN>	11
	:VOLTage{[1] 2 3 4}[:DC]:RANGe:OFFSet?	
	:VOLTage{[1] 2 3 4}[:DC]:RANGe:PTPeak <sensitivity MAX MIN UP DOWN>	9
	:VOLTage{[1] 2 3 4}[:DC]:RANGe:PTPeak?	
<b>SYSTem</b>	:AUTOTest	34
	:AUTOTest?	
	:COMMunicate:SOCKeT:ADDRess <"Address IP">	33
	:COMMunicate:SOCKeT:ADDRess?	
	:COMMunicate:SOCKeT:GATEway <"Address IP">	33
	:COMMunicate:SOCKeT:GATEway?	
	:COMMunicate:SOCKeT:MASK <"mask IP">	33
	:COMMunicate:SOCKeT:MASK?	
	:COMMunicate:WIFI <0 OFF 1 ON>	33
	:COMMunicate:WIFI?	
	:DATE <yyyy>,<mm>,<dd>	34
	:DATE?	
	:ERRor[: NEXT]?	39
	:LANGUage <ENGLish FREnch GERman SPANish ITALian>	34
	:LANGUage?	
	:POWer:OFF <1 0 ON OFF>	35
	:POWer:OFF?	
	:RESET	35
	:SET <block>	31
	:SET?	
	:TIME <hh>,<mm>,<ss>	34
	:TIME?	
<b>TRACe</b>	:BITMAP?	29
	:CATalog?	28
	[[:DATA] ? <INT1 2 3 4>	29
	:LIMit <abscissa1>,<abscissa2>,<pas>	29
	:LIMit?	

Directory	Commands + parameters	Page
TRIGger	[[:SEQuence{[1] 2 3 4 5 6}]:ATRIGger[:STATe] <1 0 ON OFF> [:SEQuence{[1] 2 3 4 5 6}]:ATRIGger[:STATe]?	18
	[[:SEQuence{[7] 8}:AUXLEVel{[1] 2 3 4} <level MAX MIN UP DOWN> [:SEQuence{[7] 8}:AUXLEVel{[1] 2 3 4}?>	38
	[[:SEQuence{[1] 2 3 4}:COUPling	12
	[[:SEQuence{[1] 2 3 4 5 6 7 8}:DEFine?	12
	[[:SEQuence{[2] 3}:DELay <time MAX MIN UP DOWN> [:SEQuence{[2] 3}:DELay?	15
	[[:SEQuence[4]:ECOunt <count MAX MIN UP DOWN> [:SEQuence[4]:ECOunt?	15
	[[:SEQuence{[1] 2 3 4}:FILTer:HPASs[:STATe] <1 0 ON OFF> [:SEQuence{[1] 2 3 4}:FILTer:HPASs[:STATe]?	12
	[[:SEQuence{[1] 2 3 4}:FILTer:LPASs[:STATe] <1 0 ON OFF> [:SEQuence{[1] 2 3 4}:FILTer:LPASs[:STATe]?	13
	[[:SEQuence[1] 2 3 4 5 6]:HOLDoff <time> [:SEQuence[1] 2 3 4 5 6]:HOLDoff?	15
	[[:SEQuence{[1] 2 3 4}:HYSTeresis <0 1 3> [:SEQuence{[1] 2 3 4}:HYSTeresis?	14
	[[:SEQuence{[1] 2 3 4 5}:LEVel <level MAX MIN UP DOWN> [:SEQuence{[1] 2 3 4 5}:LEVel?	14
	[[:SEQuence{[7] 8}:LEVel{[1] 2 3 4} <level MAX MIN UP DOWN> [:SEQuence{[7] 8}:LEVel{[1] 2 3 4}?	37
	[[:SEQuence{[1] 2 3 4 5}:LEVel:AUTO	36
	[[:SEQuence{[1] 2 3 4 5 6 7 8}:RUN:STATe <1 0 ON OFF> [:SEQuence{[1] 2 3 4 5 6 7 8}:RUN:STATe?	35
	[[:SEQuence{[1] 2 3 4 5 6}:SLOPe <POSitive NEGative> [:SEQuence{[1] 2 3 4 5 6}:SLOPe?	14
	[[:SEQuence{[7] 8}:SLOPe{[1] 2 3 4} <NO POSitive NEGative EITHer WINDow> [:SEQuence{[7] 8}:SLOPe{[1] 2 3 4}?	38
	[[:SEQuence{[1] 2 3 4 5}:SOURce <INTernal{1 2 3 4}> [:SEQuence{[1] 2 3 4 5}:SOURce?	12
	[[:SEQuence[2]:TYPE <EQUate SUPerior INFerior> [:SEQuence[2]:TYPE?	15
	[[:SEQuence[5]:VIDeo:FIELd:FORMat:LPFRame <525 625> [:SEQuence[5]:VIDeo:FIELd:FORMat:LPFRame?	13
	[[:SEQuence[5]:VIDeo:LINE:SElect <line> [:SEQuence[5]:VIDeo:LINE:SElect?	13
	[[:SEQuence[5]:VIDeo:SSIGnal[:POLarity] <POSitive NEGative> [:SEQuence[5]:VIDeo:SSIGnal[:POLarity]?	13