ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Keysight / Agilent 83480A* at our website: *Click HERE*

Programmer's Quick Reference

Agilent 83480A Analyzer,

Agilent 54750A Oscilloscope

Agilent part number: 83480-90052
Printed in USA June 2000

Agilent Technologies
Lightwave Division
1400 Fountaingrove Parkway
Santa Rosa, CA 95403-1799, USA
(707) 577-1400

Notice

This manual and any information contained herein are provided "as is" and are subject to change without notice. Agilent Technologies makes no warranty of any kind with regard to this material, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Agilent Technologies shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

©Copyright Agilent Technologies 2000
All Rights Reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

HP and Hewlett-Packard are U.S. registered trademarks of Hewlett-Packard Company.

About This Quick Reference

This quick reference is intended for use by the experienced programmer. It contains a listing of all the programming commands. Refer to the *Agilent 83480A, Agilent 54750A Programmer's Guide* for detailed programming information.

The following notation conventions are used in this book:

`computer type`	All characters appearing in `computer type` are key words and must be entered exactly as shown.
CAPITAL LETTERS	Capital letters indicate the short form of a command. The actual command is not case sensitive and can be entered in upper or lowercase.
< >	Characters appearing in angular brackets indicate values that are defined in the programmer's guide.
[ ]	Square brackets indicate that whatever occurs within the brackets is optional.
{ }	Braces are used to clarify which elements are to be chosen from.
|	"Or", indicates a choice of exactly one element from a list (for example, <a>|<b> indicates <a> or <b> but not both).

Contents

Common Commands 7
Root Level Commands 9
System Commands 12
Acquire Commands 13
Calibration Commands 14
Channel Commands 17
Disk Commands 19
Display Commands 20
FFT Commands 23
Function Commands 24
Hardcopy Commands 27
Histogram Commands 29
Limit Test Commands 31
Marker Commands 35
Mask Test Commands 37
Measure Commands 43
Pixel Memory Commands 51
Service Commands 52
Timebase Commands 53
Trigger Commands 55
TriggerN Commands 56
Waveform Commands 57
Waveform Memory Commands 61

Common Commands

`*CLS`

Clears the status data structures and the Request-for-OPC flag.

`*ESE[?]` *<mask>*

Sets the Standard Event Status Enable Register bits.

`*ESR?`

Reads the Standard Event Status Register.

`*IDN?`

Returns the instrument model number, option, serial number, and software version.

`*LRN[?]` *<setup>*

Sends a response message (learn string) that contains the instrument's current setup.

`*OPC[?]`

Sets the operation complete bit when all pending device operations have finished.

`*OPT[?]`

Returns the model number and serial number of the mainframe and all installed plug-in options.

`*RCL` *<register>*

Recalls the state of the instrument from the specified save/recall register.

`*RST`

Places the instrument in a known state.

`*SAV` *<register>*

Stores the current state of the instrument in a save/recall register.

`*SRE[?]` *<mask>*

Sets the Service Request Enable Register bits.

*STB?

Reads the Status Byte.

*TRG

Has the same effect as the Group Execute Trigger message (GET) or RUN command.

*TST?

Causes the instrument to perform a self-test and places a result in the output queue.

*WAI

Prevents the instrument from executing any further commands or queries until all currently executing commands are completed.

Root Level Commands

`:AER?`

Reads the Arm Event Register.

`:AUToscale`

Causes the instrument to evaluate all input signals and to find the optimum conditions for displaying the signal.

`:BLANK` {`CHANnel`<*number*> | `FUNCtion`<*number*> | `WMEMory`<*number*> | `PMEMory`<*number*> | `FFT` | `HISTogram`}

Turns off an active channel, function, waveform memory, pixel memory, FFT, or histogram.

`:CDISplay`

Clears the display and resets all associated measurements.

`:DIGitize` [<*waveform_name*>] [,<*waveform_name*>]

Initializes the selected channels, functions, or FFT to "unacquired," and then acquires them according to the current instrument settings.

`:ERASe` [`PMEMory`<*number*>]

Erases the specified pixel memory.

`:HEEN`[`?`] <*mask*>

Sets the Histogram Event Enable register.

`:HER?`

Reads the Histogram Event Register.

`:LER?`

Reads the Local (LCL) Event Register.

`:LTEE`[`?`] <*mask*>

Sets the Limit Test Event Enable register.

`:LTER?`

Returns the Limit Test Event Register.

`:MENU[?] {APPLication | CHANnel<`*number*`> | ACQuire | TIMebase | TRIGger | DISK | DISPlay | MARKer | MEASure | MATH | WAVeform | SETup | PRINt | HELP | UTILity | FFT | LTESt | HISTogram | MTESt | MEYE}`

Selects one of the menus on the front panel.

`:MERGe [PMEMory<`*number*`>]`

Stores the active display into the pixel memory.

`:MODel? {FRAMe | PLUGin<`*number*`>}`

Returns the Agilent Technologies model number for the frame or plug-in.

`:MTEE[?] <`*mask*`>`

Sets the Mask Event Enable register.

`:MTER?`

Reads the Mask Event Register.

`:OPEE[?] <`*mask*`>`

Sets the Operation Status Enable register.

`:OPER?`

Reads the Operation Status Register.

`:PRINt`

Outputs a copy of the screen to a printer or other device destination specified in the Hardcopy subsystem.

`:RECall:SETup <`*setup_num*`>`

Recalls a setup.

`:RUN`

Places the instrument in the running state, in which waveforms are acquired according to the current settings.

`:SERial[?] {FRAMe | PLUGin<`*number*`>},<`*string*`>`

Sets the serial number for the instrument frame or plug-in.

`:SINGle`

Causes the instrument to make a single acquisition when the next trigger event occurs.

`:STOP`

Causes the instrument to stop acquiring data for the active display.

`:STORe:PMEM1 PMEM1`

Stores the current display in pixel memory 1.

`:STORe:SETup` <*setup_num*>

Saves the current instrument setup in setup memories 0 through 9.

`:STORe:WAVeform {CHANnel`<*number* > | `FUNCtion`<*number* > | `WMEMory`<*number*> | `HISTogram` | `FFT` }, {`WMEMory`<*number*>}

Copies a channel, function, stored waveform, histogram, or FFT to a waveform memory.

`:TEER[?]`

Sets the Trigger Event Enable Register.

`:TER?`

Reads the Trigger Event Register.

`:UEE[?]` <*mask*>

Sets the User Event Enable register.

`:UER?`

Reads the User Event Register.

`:VIEW {CHANnel`<*number*> | `FUNCtion`<*number*> | `WMEMory`<*number*> | `PMEMory`<*number*> | `HISTogram` | `FFT`}

Turns on a channel, function, pixel memory, waveform memory, histogram, or FFT.

System Commands

`:SYSTem:DATE[?]` *<day>,<month>,<year>*

Sets the date in the instrument.

`:SYSTem:DSP[?]` *<string>*

Writes a quoted string, excluding quotes, to the advisory line of the instrument display.

`:SYSTem:ERRor?` [{`NUMBer` | `STRing`}]

Outputs the next error number in the error queue over the HP-IB.

`:SYSTem:HEADer[?]` {`ON` | `1` | `OFF` | `0`}

Specifies whether the instrument will output a header for query responses.

`:SYSTem:KEY[?]` *<key_code>*

Simulates the pressing of a specified front-panel key.

`:SYSTem:LONGform[?]` {`ON` | `1` | `OFF` | `0`}

Specifies the format for query response headers.

`:SYSTem:SETup[?]` *<binary block data>*

Sets up the instrument, as defined by the data in the setup string from the controller.

`:SYSTem:TIME[?]` *<hour>,<minute>,<second>*

Sets the time in the instrument.

Acquire Commands

`:ACQuire:AVERage[?]` {`ON` | `1` | `OFF` | `0`}

Controls averaging.

`:ACQuire:BEST` {`THRuput` | `FLATness`}

Improves step flatness. *Agilent 54750A only.*

`:ACQuire:COUNt[?]` <*value*>

Sets the number of averages.

`:ACQuire:POINts[?]` {`AUTO` | <*points_value*>}

Specifies the requested record length for an acquisition.

Calibration Commands

`:CALibrate:FRAMe:CANCel`

Is equivalent to pressing the Cancel softkey when in the front-panel Calibrate Frame menu. This command cancels the calibration on the instrument mainframe.

`:CALibrate:FRAMe:CONTinue`

Is equivalent to pressing the front-panel CONTINUE softkey when in the Calibration Frame menu. This command continues the calibration on the instrument mainframe.

`:CALibrate:FRAMe:DATA[?]` <*cal factors*>

Sends the FRAMe data to the instrument.

`:CALibrate:FRAMe:DONE?`

Returns the pass/fail status of the last frame calibration.

`:CALibrate:FRAMe:LABel[?]` <*label*>

Accepts a string of up to 80 characters. It is displayed as part of the frame calibration status screen and is optional. It is intended for user notes, such as name/initials of the calibrator or special notes about the calibration.

`:CALibrate:FRAMe:MEMory?`

Returns the state of the frame calibration write-protect switch.

`:CALibrate:FRAMe:STARt`

Starts the annual calibration on the instrument mainframe.

`:CALibrate:FRAMe:TIME?`

Returns the date, time and temperature at which the last full frame calibration process was completed.

`:CALibrate:OUTPut[?]` <*value*>

Sets the dc level of the calibrator signal output through the front-panel CAL connector.

`:CALibrate:PLUGin:ACCuracy {SLOT1 | SLOT2 | SLOT3 | SLOT4}`

Sets the plug-in calibration accuracy level.

`:CALibrate:PLUGin:CANCel`

Is equivalent to pressing the front-panel Cancel softkey when in the Calibrate Plugin menu. This command cancels the calibration on a selected slot (plug-in).

`:CALibrate:PLUGin:CONTinue`

Is equivalent to pressing the front-panel Continue softkey when in the Calibrate Plugin menu. This command continues the calibration on a selected slot (plug-in).

`:CALibrate:PLUGin:DONE?`

Returns the pass/fail status of the last plug-in calibration.

`:CALibrate:PLUGin:MEMory? {SLOT1 | SLOT2 | SLOT3 | SLOT4}`

Returns the state of the plug-in memory write-protect switch in the specified slot.

`:CALibrate:PLUGin:OFFSet {CHAN1 | CHAN2 | CHAN3 | CHAN4}`

Sets the plug-in calibration offset.

`:CALibrate:PLUGin:OPOWer`

Sets the plug-in calibration.

`:CALibrate:PLUGin:OPTical {CHAN1 | CHAN2 | CHAN3 | CHAN4}, [WAV1 | WAV2 | USER]`

Starts the plug-in optical calibration sequence.

`:CALibrate:PLUGin:OWAVelength` <*wavelength*>

Sets the plug-in calibration wavelength.

`:CALibrate:PLUGin:TIME? {SLOT1 | SLOT2 | SLOT3 | SLOT4}`

Returns the date, time and temperature at which the plug-in, in the specified slot, was last calibrated.

`:CALibrate:PLUGin:VERTical {SLOT1 | SLOT2 | SLOT3 | SLOT4}`

Starts the plug-in electrical calibration sequence.

`:CALibrate:PROBE {CHAN1 | CHAN2 | CHAN3 | CHAN4}`

Sets the plug-in probe calibration factor.

`:CALibrate:SAMPlers {DISable | ENABle}`

Enables or disables samplers for service.

`:CALibrate:SKEW[?] {CHANnel<N>}, <skew>`

Sets the channel-to-channel skew factor for channel N.

`:CALibrate:STATus?`

Returns the calibration status of the instrument.

Channel Commands

:CHANnel<*number*>:AUToscale

Causes the instrument to evaluate the input signal and to find the optimum vertical scale for displaying the signal.

:CHANnel<*number*>:BANDwidth[?] {HIGH | LOW}

Controls an internal low-pass filter if one is present in the channel hardware.

:CHANnel<*number*>:DISPlay[?] {ON | 1 | OFF | 0}

Sets the specified channel to on or off.

:CHANnel<*number*>:FDEScription[?]

Requires firmware revision A.04.00 and above.

Returns the number of filters and a brief description of each filter for channels with more than one internal low-pass filter.

:CHANnel<*number*>:FILTer[?] {ON | 1 | OFF | 0}

Controls an internal low-pass filter if one is present in the channel hardware.

:CHANnel<*number*>:FSELect FILTer<*filter_number*>

Requires firmware revision A.04.00 and above.

Selects which filter is controlled by on/off for channels with more than one internal low-pass filter.

:CHANnel<*number*>:OFFSet[?] <*offset value*>

Sets the voltage that is represented at center screen for the selected channel.

:CHANnel<*number*>:PROBe[?] <*attenuation factor*>[,{RATio|DECibel}]

Sets the probe attenuation factor and, optionally, the units for the probe attenuation factor.

:CHANnel<*number*>:PROBe:CALibrate

Starts the probe's calibration for the selected channel.

:CHANnel<*number*>:RANGe[?] <*range value*>

Sets up acquisition and display hardware to display the waveform at a given full-scale range.

:CHANnel<*number*>:SCALe[?] <*scale value*>

Defines the vertical scale of the channel in units per division. This command is the same as the front-panel channel scale.

:CHANnel<*number*>:SKEW[?]

Sets the skew.

:CHANnel<*number*>:UNITs[?] {VOLT | AMPere | WATT | UNKNown}

Allows you to work in vertical units other than volts. The units command changes the name of the Y-axis units from VOLT to AMPere, WATT, or UNKNown. The units are implied for other pertinent channel commands (such as RANGe and OFFSet).

:CHANnel<*number*>:UNITs:ATTenuation[?] <*attenuation*>

Specifies a ratio that indicates how the unit specified in :CHANnel <*number*>:UNITs relates to one volt.

:CHANnel<*number*>:UNITs:OFFSet[?] <*offset*>

Specifies a constant value in volts that will be added to the signal to compensate for any offset.

:CHANnel<*number*>:WAVelength[?] {WAV1 | WAV2 | USER}

Sets the wavelength selection for optical channels. The Agilent 83487A plug-in module will have one factory calibration; all other optical modules will have two.

Disk Commands

`:DISK:DATA?` "<*filename*>"

Outputs the specified file on the disk via GP-IB.

`:DISK:DELete` "<*filename*>"

Deletes a file from the disk. An error is displayed on the screen if the requested file does not exist.

`:DISK:DIRectory?`

Returns the directory listing of the currently installed disk. Each entry is 63 bytes long, including a carriage return and line feed.

`:DISK:FORMat`

Formats a disk in the drive. It is assumed that the disk that is to be formatted is in the drive when the command is issued.

`:DISK:LOAD` "<*filename*>"[,<*destination*>]

Restores a setup, a waveform, a mask, a database, or a pixel memory from the disk. The type of file is determined by the filename suffix if one is present or by the destination field if one is not present. The database may only be loaded in internal format.

`:DISK:STORe` <*source*>, "<*filename*>"[,<*format*>]

Stores a setup, a waveform, a database, a mask, or pixel memory to the disk. The filename does not include a suffix. The suffix is supplied by the instrument depending on the source and file format specified.

Display Commands

`:DISPlay:ASSign[?]` {`CHANnel`<*number*> | `FUNCtion`<*number*> | `WMEMory`<*number*> | `FFT`},{`UPPer` | `LOWer`}

Assigns the specified waveform, function, or FFT to the specified portion of the waveform area on the screen. This command has no effect when the graticule format is single.

`:DISPlay:CGRade[?]` {`ON` | `1` | `OFF` | `0`}

Sets the color-graded display on or off.

`:DISPlay:CGRade:LEVels?`

Returns the range of hits represented by each color. Fourteen values are returned, representing the minimum and maximum count for each of seven colors.

`:DISPlay:COLumn[?]` <*value*>

Specifies the starting column for subsequent DISPlay:STRing and DISPlay:LINE commands.

`:DISPlay:DATA[?]` <*binary_block_data*>

Writes waveform data to the pixel memory of the instrument (PMEMory1). The DATA command is followed by a block of binary data that is transferred from the controller to the pixel memory in the instrument.

`:DISPlay:DCOLor`

Returns the screen colors to the predefined colormap.

`:DISPlay:DWAVeform[?]` {`FAST` | `CDOTs` | `HRESolution`}

Sets the waveform draw mode to FAST, Connected DOTs, or High RESolution. This command has no effect when the color-graded display is on.

`:DISPlay:FORMat[?]` {`SINGle` | `1` | `DUAL` | `2`}

Sets the number of graphs. Sending 1 or SINGle sets the number of graphs to 1. Sending 2 or DUAL sets the number of graphs to 2. Each graph defines a separate graticule area within the waveform display area.

:DISPlay:GRATicule[?] {GRID | FRAMe}

Selects the type of graticule that is displayed.

:DISPlay:INVerse[?] {ON | 1 | OFF | 0}

Determines whether or not text sent with the DISPlay:LINE or DISPlay:STRing command will be written with the INVerse attribute. If the inverse attribute is on, the text is written in inverse video.

:DISPlay:LINE "<*string*>"

Writes a quoted string parameter to the screen, starting at the location specified by the DISPlay:ROW and DISPlay:COLumn commands.

:DISPlay:MASK[?] <*mask_value*>

Inhibits the instrument from writing to selected areas of the screen. The mask parameter is an 8-bit integer in which each bit controls writing to an area of the screen. A 0 inhibits writing to the area represented by the bit, a 1 enables writing to that area.

:DISPlay:PERSistence[?] {MINimum | INFinite | <*value*>}

Sets the display persistence. It works in both real and equivalent time. The parameter for this command can be either MINimum (zero persistence), INFinite, or a real number from 0 to 40.0, representing the persistence in seconds.

:DISPlay:ROW[?] <*row_number*>

Specifies the starting row on the screen for subsequent DISPlay:STRing and DISPlay:LINE commands. The row number remains constant until another DISPlay:ROW command is received, or the row is incremented by the DISPlay:LINE command.

:DISPlay:SCOLor[?] <*color_name*>, <*hue*>, <*saturation*>, <*luminosity*>

Changes the colors used on the display of the instrument. The :DISPlay:DCOLor command restores the colors to their factory default settings.

:DISPlay:SOURce[?] PMEMory1

Specifies the destination or source for the DISPlay:DATA command and query. Pixel memory 1 (PMEMory1) is the only source available for this command.

`:DISPlay:STRing "`*<string>*`"`

Writes text to the screen of the instrument. The text is written starting at the current row and column settings. If the column limit is reached (81), the excess text is discarded. The DISPlay:STRing command does not increment the row value; however, the DISPlay:LINE command does.

`:DISPlay:TEXT BLANk`

Blanks the user text area of the screen. The user text area includes rows 0 through 27, columns 0 through 81.

FFT Commands

`:FFT:DISPlay[?] {ON | 1 | OFF | 0}`

Turns the FFT function on and off.

`:FFT:FREQuency[?]` *<frequency_value>*

Sets the center frequency value of the FFT display when MAGNify is on. Magnify is turned on and off with the FFT:MAGNify command.

`:FFT:MAGNify[?] {ON | 1 | OFF | 0}`

Turns the magnification mode of the FFT on and off. When magnify is on, changing MSPan and FREQuency results in software magnification of the display. The hardware setup is not changed.

`:FFT:MSPan[?]` *<magnify_span_value>*

Sets the span when magnify is on. Magnify is controlled with the FFT:MAGNify command. MSPan causes software magnification of the FFT display. The hardware setup is not affected.

`:FFT:OFFSet[?]` *<offset_value>*

Sets the dBm offset value represented at the center of the screen for the FFT display. This command controls software magnification of the FFT display. It does not affect the hardware settings.

`:FFT:RANGe[?]` *<range_value>*

Sets the vertical range value for the FFT. This command controls software scaling of the FFT display. The hardware settings are not affected.

`:FFT:SOURce[?] {CHANnel`*<number>* `| FUNCtion`*<number>* `| WMEMory`*<number>* `|` *<float_value>*`}`

Selects the source for the FFT.

`:FFT:WINDow[?] {RECTangular | HANNing | FLATtop}`

Sets the window type for the FFT. The FFT function assumes that the time record repeats. Unless there is an integral number of cycles in the sampled waveform in the record, a discontinuity is created at the end of the record.

Function Commands

`:FUNCtion`<*number*>`:ADD` <*operand*>,<*operand*>

Defines a function that takes the algebraic sum of two defined operands.

`:FUNCtion`<*number*>`:BWLimit` <*operand*>,<*frequency_range*>

Turns on the bandwidth limit filter function.

`:FUNCtion`<*number*>`:DIFFerentiate` <*operand*>

Defines a function that computes the discrete derivative of the defined operand's waveform.

`:FUNCtion`<*number*>`:DISPlay[?] {ON | 1 | OFF | 0}`

Turns the selected function on and off.

`:FUNCtion`<*number*>`:DIVide` <*operand*>,<*operand*>

Defines a function that divides the first operand by the second operand.

`:FUNCtion`<*number*>`:FFT:FREQuency[?]` <*center_frequency_value*>

Sets the center frequency for the FFT when magnify is on. Magnify is turned on and off with the :FUNCtion<*number*>:FFT:MAGNify command.

`:FUNCtion`<*number*>`:FFT:MAGNify[?] {ON | 1 | OFF | 0}`

Enables the magnification mode.

`:FUNCtion`<*number*>`:FFT:MSPan[?]` <*magnify_span_value*>

Sets the magnify span for the FFT function.

`:FUNCtion`<*number*>`:FFT:WINDow[?] {RECTangular | HANNing | FLATtop}`

Sets the window type for the FFT function.

`:FUNCtion`<*number*>`:FFTMagnitude[?]` <*operand*>

Computes the fast Fourier transform of the specified channel or memory. The FFT takes the digitized time record of the specified channel or memory and transforms it to the frequency domain.

`:FUNCtion`<*number*>`:HORizontal[?] { TSOurce | MANual }`

Sets the horizontal tracking to either Track SOurce or MANual.

:FUNCtion<*number*>:HORizontal:POSition[?] <*position_value*>

Sets the time value at center screen for the selected function.

:FUNCtion<*number*>:HORizontal:RANGe[?] <*range_value*>

Sets the current time range for the specified function.

:FUNCtion<*number*>:INTegrate <*operand*>

Defines a function that computes the integral of the defined operand's waveform.

:FUNCtion<*number*>:INVert <*operand*>

Defines a function that inverts the defined operand's waveform by multiplying by −1.

:FUNCtion<*number*>:MAGNify <*operand*>

Defines a function that is a copy of the operand. The magnify function is a software magnify. No hardware settings are altered as a result of using this function. It is useful for scaling channels, another function, and memories with the RANGe and OFFSet commands in this subsystem.

:FUNCtion<*number*>:MAXimum <*operand*>

Defines a function that computes the maximum value of the operand waveform in each time bucket.

:FUNCtion<*number*>:MINimum <*operand*>

Defines a function that computes the minimum of each time bucket for the defined operand's waveform.

:FUNCtion<*number*>:MULTiply <*operand*>,<*operand*>

Defines a function that multiplies the first operand by the second operand.

:FUNCtion<*number*>:OFFSet[?] <*offset_value*>

Sets the voltage represented at the center of the screen for the selected function.

:FUNCtion<*number*>:ONLY <*operand*>

Does the same thing as the MAGNIFY command.

:FUNCtion<*number*>:RANGe[?] <*full_scale_range*>

Defines the full scale vertical axis of the selected function.

:FUNCtion<*number*>:SUBTract <*operand*>,<*operand*>

Defines a function that algebraically subtracts the second operand from the first operand.

:FUNCtion<*number*>:VERSus <*operand*>,<*operand*>

Defines a function for an X versus Y display. The first operand defines the y-axis and the second defines the x-axis. The y-axis range and offset is initially equal to that of the first operand and can be adjusted with the RANGE and OFFSET commands in this subsystem.

:FUNCtion<*number*>:VERTical[?] {TSOurce | MANual | AUTO}

Sets the vertical scaling mode of the specified function to either Track SOurce, MANual, or AUTO.

:FUNCtion<*number*>:VERTical:OFFSet[?] <*offset_value*>

Sets the voltage represented at center screen for the selected function.

:FUNCtion<*number*>:VERTical:RANGe[?] <*full_scale_range*>

Defines the full-scale vertical axis of the selected function.

Hardcopy Commands

`:HARDcopy:ADDRess[?]` *<address>*

Sets the address for the printer when printing to an HP-IB printer.

`:HARDcopy:AREA[?] {GRATicule | SCReen | FACTors}`

Selects which data from the screen is to be printed.

`:HARDcopy:BACKground[?] {WHITe | NORMal}`

Controls the background color of the graticule area of an HP PaintJet printer.

`:HARDcopy:DESTination[?] {CENTronics | HPIB | DISK}`

Selects the destination for printing. The options are HPIB, CENTronics, and the internal DISK.

`:HARDcopy:DEVice[?] {THINkjet | PAINtjet | LASerjet | EPSon | TIFF | CTIFf | GIF | PCX | DJ500 | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the output format.

`:HARDcopy:FACTors[?] {ON | 1 | OFF | 0}`

Determines whether the instrument setup factors will be appended to screen or graticule images.

`:HARDcopy:FFEed[?] {ON | 1 | OFF | 0}`

Sets the form feed option. If it is set to ON, a formfeed will occur at the end of the hardcopy; otherwise, the page will scroll up by 4 lines.

`:HARDcopy:FILename` *<string>*

Specifies the filename when printing files to the disk.

`:HARDcopy:LENGth[?] {ENGLish | 11 | METRic | 12}`

Sets the length of the paper to either ENGLish (11 inches) or METRic (12 inches). The longer paper is metric size A4.

`:HARDcopy:MEDia[?] {PAPer | TRANsparency}`

Sets the speed of the printer. When TRANsparency is selected, the printer prints the data twice, which makes the contents of the paper look darker and slows down the printing process. This command applies only to HP PaintJet and color DeskJet printers.

Histogram Commands

`:HISTogram:AXIS[?] {VERTical | HORizontal}`

Selects the axis of the histogram. A horizontal or vertical histogram may be created.

`:HISTogram:MODE[?] {OFF | WAVeform}`

Selects the histogram mode. The histogram may be off or set to track the waveform database.

`:HISTogram:RRATe[?]` *<time>*

Sets the histogram refresh rate.

`:HISTogram:RUNTil[?] {FORever | {WAVeforms,` *<n waveforms>*`} | {SAMples,` *<n samples>*`}}`

Selects the histogram run until mode. The histogram may be set to run until *n* waveforms or *n* samples have been acquired or will run forever.

`:HISTogram:SCALe[?] {AUTO | MANual}`

Selects the histogram scaling mode. The scaling mode may be automatic or manual. In automatic scaling, the scale will be set to display the histogram using one half of the display.

`:HISTogram:SCALe:OFFSet[?]` *< offset >*

Sets the histogram offset. For horizontal histograms, this is the vertical offset in percentage of peak or decibels. For vertical histograms, this is the horizontal offset in percentage of peak or decibels.

`:HISTogram:SCALe:RANGe[?]` *< range >*

Selects the histogram scale across the entire display. For horizontal histograms, this is the vertical percentage of peak or decibels across the display. For vertical histograms, this is the horizontal percentage of peak or decibels across the display.

`:HISTogram:SCALe:SCALe[?]` < *scale* >

Selects the histogram scale per division. For horizontal histograms, this is the vertical percentage of peak or decibels per division. For vertical histograms, this is the horizontal percentage of peak or decibels per division.

`:HISTogram:SCALe:TYPE[?]` {`LINear` | `LOGarithmic`}

Selects the histogram scale type. The histogram may be displayed according to a linear or a logarithmic scale.

`:HISTogram:WINDow:DEFault`

Sets the histogram window markers to default values. The X1, X2, Y1, and Y2 markers are set to values corresponding to graticule positions that are one division in from the left, right, top, and bottom of the graticule respectively.

`:HISTogram:WINDow:SOURce[?]` {`CHANnel`<*N*> | `FUNCtion`<*N*> | `WMEMory`<*N*> | FFT}

Selects the source of the histogram window. The histogram window will track the source's vertical and horizontal scale.

`:HISTogram:WINDow:X1Position[?]` <*X1 position*>

Moves the X1 marker of the histogram window. The histogram window selects a portion of the database to histogram.

`:HISTogram:WINDow:X2Position[?]` <*X2 position*>

Moves the X2 marker of the histogram window. The histogram window selects a portion of the database to histogram. The histogram window markers will track the scale of the histogram window source.

`:HISTogram:WINDow:Y1Position[?]` <*Y1 position*>

Moves the Y1 marker of the histogram window. The histogram window selects a portion of the database to histogram.

`:HISTogram:WINDow:Y2Position[?]` <*Y2 position*>

Moves the Y2 marker of the histogram window. The histogram window selects a portion of the database to histogram.

Limit Test Commands

`:LTESt:FAIL[?] {INSide | OUTSide | ALWays | NEVer}`

Sets the fail condition for an individual measurement. The conditions for a test failure are set on the source selected with the last LTESt:SOURce command.

`:LTESt:LLIMit[?]` *<lower_value>*

Sets the lower test limit for the active measurement currently selected by the :LTESt:SOURce command.

`:LTESt:MNFound[?] {FAIL | PASS | IGNore}`

Sets the action to take when the measurement cannot be made. This command affects the active measurement currently selected by the last LTESt:SOURce command.

`:LTESt:RUMode[?] {FORever | FAILures,` *<total_failures>* `| WAVeforms,` *<maximum_waveforms>*`}`

Equivalent to :LTESt:RUN.

`:LTESt:RUN[?] {FORever | FAILures,` *<total_failures>* `| WAVeforms,` *<maximum_waveforms>*`}`

Determines the termination conditions for the test. The choices are FORever, FAILures, or WAVeforms.

`:LTESt:SOURce[?] {1 | 2 | 3 | 4}`

Selects the current source for the ULIMit, LLIMit, MNFound, and FAIL commands. It selects one of the active measurements as referred to by their position in the measurement window on the bottom of the screen. Source 1 is the measurement on the top line, 2 is on the second line, and so on.

`:LTESt:SSCReen[?] {OFF | PMEMory1 | PRINter | DISK [,`*<filename>*`]}`

Saves a copy of the screen in the event of a failure.

`:LTESt:SSCReen:DDISk:BACKground[?] {WHITe | NORMal}`

Controls the background color of the graticule area of an HP PaintJet printer directed to the disk. In NORMal, the selected screen color is used for that area. In WHITe, the background area is forced to white (the color of the printer paper).

`:LTESt:SSCReen:DDISk:MEDia[?] {PAPer | TRANsparency}`

Specifies either paper or transparency to be used in the printer. When TRANsparency is selected, the printer prints the data twice, which makes the contents of the media look darker and slows down the printing process. This control is used when the store screen is directed to the disk, and is valid only when printing to the HP PaintJet and color DeskJet printers.

`:LTESt:SSCReen:DDISk:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | GIF | TIFF | CTIFf | PCX | DJ500 | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the file format for a stored file or the printer format to use when storing the screen to a disk for later printing. This includes all print formats supported by the instrument.

`:LTESt:SSCReen:DPRinter:ADDRess[?]` *<address_value>*

Allows the user to select the HP-IB address for the printer. This address is used only if the port is HP-IB.

`:LTESt:SSCReen:DPRinter:BACKground[?] {WHITe | NORMal}`

Controls the background color of the graticule area of an HP PaintJet printer. It is only valid when the print format is an HP PaintJet. In NORMal, the selected screen color is used for that area.

`:LTESt:SSCReen:DPRinter:MEDia[?] {PAPer | TRANsparency}`

Specifies either paper or transparency in the printer. This control is used when the store screen is directed to a printer. When TRANsparency is selected, the printer prints the data twice, which makes the contents of the media look darker and slows down the printing process. This command is valid only for the HP PaintJet and color DeskJet printers.

`:LTESt:SSCReen:DPRinter:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | DJ500 | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the printer format to use when storing the screen to a physical printer.

`:LTESt:SSCReen:DPRinter:PORT[?] {CENTronics | HPIB}`

Selects the printer port for the screen when store screen is going to the printer.

`:LTESt:SSUMmary[?] {OFF | PRINter | DISK [,<filename>]}`

Saves the summary in the event of a failure.

`:LTESt:SSUMmary:ADDRess[?] <address_value>`

Selects the HP-IB address for the printer. This address is used only when the summary is going to the printer and the PORT command is set to HP-IB.

`:LTESt:SSUMmary:FORMat[?] {BRIef | STAtistics}`

Specifies whether the summary file is either brief or contains statistics.

`:LTESt:SSUMmary:MEDia[?] {PAPer | TRANsparency}`

Specifies whether paper or transparency film will be used in the printer. When TRANsparency is selected, the printer prints the data twice, which makes the contents of the media look darker and slows down the printing process. This control is used when the store summary is directed to the printer, and is valid only for the HP PaintJet and color DeskJet printers.

`:LTESt:SSUMmary:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the printer format to use when storing the summary to a printer. This includes the physical printers supported by the instrument, with the exception of the HP DeskJet 500C with the color cartridge installed. To set up the format for the HP DeskJet 500C, use the monochrome cartridge and the BWDeskjet parameter in the command.

`:LTESt:SSUMmary:PORT[?] {CENTronics | HPIB}`

Selects the printer port for the summary when the summary is sent to the printer.

:LTESt:SWAVeform[?] <*source*>, <*destination*>, [<*filename*>[,<*format*>]]

Saves waveforms from a channel, function, histogram, or waveform memory in the event of a failure detected by the limit test. Each waveform source can be individually specified, allowing multiple channels or functions to be saved to disk or waveform memories. Setting a particular source to OFF removes any waveform save action from that source.

:LTESt:TEST[?] {ON | 1 | OFF | 0}

Controls the execution of the limit test function. ON allows the limit test to run over all of the active measurements. When the limit test is turned on, the limit test results are displayed on screen in a window below the graticule.

:LTESt:ULIMit[?] <*upper_value*>

Sets the upper test limit for the active measurement currently selected by the last :LTESt:SOURce command.

Marker Commands

`:MARKer:CURSor? {DELTa | STARt | STOP}`

Returns the time and current measurement unit values of the specified marker or cursor (if in the waveform mode) as an ordered pair of time and measurement unit values.

`:MARKer:MEASurement:READout[?] {ON | 1 | OFF | 0}`

Controls the state of the marker readout area on the screen when markers are on in measurement mode.

`:MARKer:MODE[?] {OFF | MANual|WAVeform | MEASurement | HISTogram | TDR/TDT}`

Sets the marker mode.

`:MARKer:TDELta?`

Returns the time difference between X1 and X2 time markers or between the X cursor and + cursor, depending on the current marker mode.

`:MARKer:TSTArt[?]` *<X1 position>*

Sets the X1 position and moves the X1 marker or X cursor, depending on the current marker mode.

`:MARKer:TSTOp[?]` *<X2 position>*

Sets the X2 position and moves the X2 marker or + cursor, depending on the current marker mode.

`:MARKer:VDELta?`

Returns the current measurement unit difference between Y1 and Y2.

`:MARKer:VSTArt[?]` *<Y1 position>*

Sets the Y1 Position and moves Y1 to the specified measurement unit value on the specified source.

`:MARKer:VSTOp[?]` *<Y2 position>*

Sets the Y2 position and moves Y2 to the specified measurement unit on the specified source.

`:MARKer:X1Position[?]` *<X1 position>*

Sets the X1 position and moves the X1 marker or X cursor, depending on the current marker mode, to the specified time with respect to the trigger time.

`:MARKer:X2Position[?]` *<X2 position>*

Sets the X2 position and moves the X2 marker or + cursor, depending on the current marker mode, to the specified time with respect to the trigger time.

`:MARKer:X1Y1source[?]` {`CHANnel`*<number>* | `FUNCtion`*<number>* | `WMEMory`*<number>* | `HISTogram` | `FFT`}

Sets the source for the X1 and Y1 markers.

`:MARKer:X2Y2source[?]` {`CHANnel`*<number>* | `FUNCtion`*<number>* | `WMEMory`*<number>* | `HISTogram` | `FFT` }

Sets the source for the X2 and Y2 markers.

`:MARKer:XDELta?`

Returns the time difference between X1 and X2 time markers or between the X cursor and + cursor, depending on the current marker mode.

`:MARKer:XUNITs?`

Returns the x-axis units of the current marker mode.

`:MARKer:Y1Position[?]` *<Y1 position>*

Sets the Y1 Position and moves Y1 to the specified measurement unit on the specified source.

`:MARKer:Y2Position[?]` *<Y2 position>*

Sets the Y2 Position and moves Y2 to the specified measurement unit on the specified source.

`:MARKer:YDELta?`

Returns the current measurement unit difference between Y1 and Y2.

`:MARKer:YUNITs?`

Returns the Y-axis units of the current marker mode.

Mask Test Commands

`:MTESt:ALIGn`

Automatically aligns and scales the mask to the current waveform on the display or scales the waveform and mask to fill the display, depending on alignment mode. *Agilent 83480A only.*

`:MTESt:AMASk:CReate`

Automatically constructs a mask around the current waveform on the display, using the tolerance parameters defined by the AMASk:XDELta, AMASk:YDELta, and AMASK:UNITs commands. The mask only encompasses the portion of the waveform visible on the display, so your measurement must ensure that the waveform is acquired and displayed consistently to obtain repeatable results.

`:MTESt:AMASk:SOURce[?]` {`WMEMory`<*number*> | `FUNCtion`<*number*> | `CHANnel`<*number*> | `FFT`}

Selects the source for the interpretation of the AMASk:XDELta and AMASk:YDELta parameters when AMASk:UNITs is set to CURRent. When UNITs are CURRent, the XDELta and YDELta parameters are defined in terms of the measurement system of the selected source.

`:MTESt:AMASk:UNITs[?]` {`CURRent` | `DIVisions`}

Alters the way the mask test subsystem interprets the ΔX and ΔY tolerance parameters for automasking, defined by AMASk:XDELta and AMASk:YDELta, respectively.

`:MTESt:AMASk:XDELta[?]` <*xdelta_value*>

Sets the tolerance in the X direction around the waveform for the automasking feature. The absolute value of the tolerance will be added and subtracted to X components of the waveform to determine the boundaries of the mask.

`:MTESt:AMASk:YDELta[?]` <*ydelta_value*>

Sets the tolerance in the Y direction around the waveform for the automasking feature. The absolute value of the tolerance will be added and subtracted to Y components of the waveform to determine the boundaries of the mask.

:MTEST:AMODe[?] {MASK | FILL}

Sets the mode for automated mask alignment. *Agilent 83480A only. Requires firmware revision A.02.00 and above.*

:MTESt:COUNt:FAILures? <*POLYgon*><*number*>

Returns the number of failures that occurred within a particular polygon. By defining polygons within polygons, then counting the failures for each individual polygon, you can implement testing at different tolerance levels for a given waveform.

:MTESt:COUNt:FSAMples?

Returns the total number of failed samples in the current mask test run. This count is for all polygons and all waveforms, so if you wish to determine failures by polygon number, use the COUNt:FAILures? query.

:MTESt:COUNt:FWAVeforms?

Returns the total number of failed waveforms in the current mask test run. This count is for all polygons and all waveforms, so if you wish to determine failures by polygon number, use the COUNt:FAILures? query.

:MTESt:COUNt:HITS?

Returns the total number of failed data points, or hits, that occured when using margin mask testing.

:MTESt:COUNt:SAMPles?

Returns the total number of samples captured in the current mask test run.

:MTESt:COUNt:WAVeforms?

Returns the total number of waveforms gathered in the current mask test run.

:MTESt:FENable[?] {ON | 1 | OFF | 0}

Enables the fail action when a mask hit occurs.

`:MTESt:MASK:DEFine[?]` #6NNNNNN [<*mask_label*>]
<*polygon_id*>,<*number_of_vertices*>, <*x_coordinate*>,<*y_coordinate*>
[,<*x_coordinate*>,<*y_coordinate*>]* [,<*polygon_id*>,<*number_of_vertices*>,
<*x_coordinate*>, <*y_coordinate*> [,<*x_coordinate*>,<*y_coordinate*>]*]*

Defines a set of up to eight polygons to be used for mask testing. When you use the MASK:DEFine command, you can specify polygon definitions for up to eight polygons. You can also use the MTESt:POLYgon:DEFine command to define a single polygon at a time.

`:MTESt:MASK:DELete`

Deletes the complete currently defined mask.

`:MTESt:MMARgin:PERCent[?]` <*margin_percent*>

Sets the mask margin percentage.

`:MTESt:MMARgin:STATe[?]` {`ON` | `1` | `OFF` | `0`}

Controls whether the mask margin is active.

`:MTESt:POLYgon:DEFine[?]` #6NNNNNN <*polygon_id*>,
<*number_of_vertices*>,<*x_coordinate*>, <*y_coordinate*>
[,<*x_coordinate*>,<*y_coordinate*>]*

Defines a polygon to be used for mask testing. You can define one polygon at a time by using this command.

`:MTESt:POLYgon:DELete` <*POLYgon*> <*number*>

Requires firmware revision A.04.00 and above.

Deletes the specified polygon.

`:MTESt:POLYgon:MOVE` <*POLYgon*> <*number*>, <*xdelta_value*>,
<*ydelta_value*>

Requires firmware revision A.04.00 and above.

Moves the specified polygon by the X and Y amounts indicated.

`:MTESt:RECall` <*mask_memory*>

Recalls from the mask memory the mask associated with the mask memory number.

:MTESt:RUMode[?] {FOREver | FSAMples, <*number_of_failed_samples*> | FWAVeforms, <*number_of_failed_waveforms*> | SAMPles, <*number_of_samples*> | WAVeforms, <*number_of_waveforms*>}

Determines the termination conditions for the test.

:MTESt:SAVe <*mask_memory*>

Saves the mask to mask memory.

:MTESt:SCALe:DEFault

Sets the scaling markers to default values. The X1, XDELta, Y1, and Y2 markers are set to values corresponding to graticule positions that are two divisions in from the left and right and one division in from the top and bottom of the graticule, respectively.

:MTESt:SCALe:SOURce[?] {WMEMory<*number*> | FUNCtion<*number*> | CHANnel<*number*>}

Sets the source used by the mask subsystem for interpretation of the SCALe:Y1 and SCALe:Y2 parameters. SCALe:Y1 and SCALe:Y2 set the vertical boundaries of the coordinate system for mask testing, and are affected by the scaling of the selected source.

:MTESt:SCALe:X1[?] <*x1_value*>

Defines where X=0 in the base coordinate system used for mask testing. The other X-coordinate is defined by the SCALe:XDELta command.

:MTESt:SCALe:XDELta[?] <*xdelta_value*>

Defines the position of the X2 marker with respect to the X1 marker. In the mask test coordinate system, the X1 marker defines where X=0; thus, the X2 marker defines where X=1.

:MTESt:SCALe:Y1[?] <*y1_value*>

Defines where Y=0 in the coordinate system for mask testing.

:MTESt:SCALe:Y2[?] <*y2_value*>

Defines the Y2 marker in the coordinate system for mask testing.

:MTESt:SSCReen[?] {OFF | PMEMory1 | PRINter | DISK [,<*filename*>]}

Saves a copy of the screen in the event of a failure.

`:MTESt:SSCReen:DDISk:BACKground[?] {WHITe | NORMal}`

Controls the background color of the graticule area of an HP PaintJet printer directed to the disk. In NORMal, the selected screen color is used for that area. In WHITe, the background area is forced to white (the color of the printer paper).

`:MTESt:SSCReen:DDISk:MEDia[?] {PAPer | TRANsparency}`

Specifies whether paper or transparency is to be used in the printer. This control is used when the store screen is directed to the disk. If TRANsparency is selected, the printer makes two passes over each data row, putting more ink on the page. This darkens the page for better transparency results. The command applies only to the HP PaintJet and color DeskJet printers, and will slow print speed.

`:MTESt:SSCReen:DDISk:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | GIF | TIFF | CTIFf | PCX | DJ500 | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the printer format to use when storing the screen to a disk. This includes all print formats supported by the instrument.

`:MTESt:SSCReen:DPRinter:ADDRess[?]` *<address_value>*

Allows the user to select the HP-IB address for the printer. This address is used only if the port is HP-IB.

`:MTESt:SSCReen:DPRinter:BACKground[?] {WHITe | NORMal}`

Controls the background color of the graticule area of an HP PaintJet printer. It is only valid when the print format is an HP PaintJet. In NORMal, the selected screen color is used for that area. In WHITe, the background area is forced to white (the color of the printer paper). This control is used when the store screen is directed to the printer.

`:MTESt:SSCReen:DPRinter:MEDia[?] {PAPer | TRANsparency}`

Specifies either paper or transparency in the printer. If TRANsparency is selected, the printer makes two passes over each data row, putting more ink on the page. This darkens the page for better transparency results. The command applies only to the HP PaintJet and color DeskJet printers, and will slow print speed.

`:MTESt:SSCReen:DPRinter:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | DJ500 | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the printer format to use when storing the screen to a printer. This includes only the formats available from the setup print menu that refer to a physical printer.

`:MTESt:SSCReen:DPRinter:PORT[?] {CENTronics | HPIB}`

Selects the printer port for the screen when store screen is going to the printer.

`:MTESt:SSUMmary[?] {OFF | PRINter | DISK [,<filename>]}`

Saves the summary in the event of a failure.

`:MTESt:SSUMmary:ADDRess[?] <address_value>`

Selects the HP-IB address for the printer. This address is used only when the summary is going to the printer and the port is set to HP-IB.

`:MTESt:SSUMmary:MEDia[?] {PAPer | TRANsparency}`

Specifies either paper or transparency in the printer. When TRANsparency is selected, the printer prints the data twice, which makes the contents of the paper look darker and slows down the printing process. This control is used when the store summary is directed to the printer, and is only valid for the HP PaintJet and color DeskJet printers.

`:MTESt:SSUMmary:PFORmat[?] {THINkjet | PAINtjet | LASerjet | EPSon | DJ540 | DJ550 | DJ560 | BWPaintjet | BWDeskjet}`

Selects the printer format to use when storing the summary to a printer. This includes the physical printers supported by the instrument, with the exception of the HP DeskJet 500C with the color cartridge installed. To set up the format for the HP DeskJet 500C, use the monochrome cartridge and the BWDeskjet parameter in the command.

`:MTESt:SSUMmary:PORT[?] {CENTronics | HPIB}`

Selects the printer port for the summary when the summary is sent to the printer.

:MTESt:STANdard {DS1 | DS1C |DS2 | DS3 | FC133 | FC266 | FC531 | FC1063 | FDD1 | FX100Base | GBEThernet | OC1 | OC3 |OC12 | OC24 | OC48 | STM0 | STM1 | STM4 | STM16 | STS1E | STS1P | STS3E | STS3P0 | STS3P1 | PDH2 | PDH8 | PDH34 | PDH1390 | PDH1391 | TENGBIT}

Loads a predefined standard mask as the current mask.

OC1 and STM0, OC3 and STM1, OC12 and STM4, and OC48 and STM16 are the same tests.

The standards PDH2 . . . PDH1391 are available in firmware revision A.02.00 and above.

The standards FX100Base and GBEThernet are available in firmware revision A.06.00 and above.

TENGBIT is available in firmware revision A.07.00 and above.

:MTESt:SWAVeform[?] <*source*>, <*destination*>, [<*filename*>[,<*format*>]]

Saves waveforms from a channel, function, histogram, or waveform memory in the event of a failure detected by the Mask Test. Each waveform source can be individually specified allowing multiple channels or functions to be saved to disk or waveform memories. Setting a particular source to OFF removes any waveform save action from that source.

:MTESt:SWAVeform:RESet

Sets the save destination for all waveforms to OFF.

:MTESt:TEST[?] {ON | 1 | OFF | 0}

Controls the execution of the Mask Test function. ON allows the Mask Test to run for all of the active sources. When the Mask Test is turned on, the Mask Test results are displayed on screen in a window below the graticule in the mask test window. OFF disables mask testing.

Measure Commands

`:MEASure:APOWer[?]` [<*source*>]

Measures the average power of an optical signal. Sources are specified with the MEASure:SOURce command or with the optional parameter following the APOWer command. *Agilent 83480A only.*

`:MEASure:CGRade:COMPlete[?]` [<*comp_hits*>]

Sets the completion criterion.

`:MEASure:CGRade:CROSsing[?]`

Measures the crossing level of an eye diagram. Color grade mode must be on for this measurement.

`:MEASure:CGRade:DCDistortion[?]` {`TIME` | `PERCent`}

Measures the duty cycle distortion of an eye diagram. Color grade mode must be on for this measurement.

`:MEASure:CGRade:EHEight[?]` {`TIME` | `PERCent`}

Measures the eye height of an eye diagram. Color grade mode must be on for this measurement.

`:MEASure:CGRade:ERAtio[?]` {`RATio` | `DECibel` | `PERCent`}

Measures the extinction ratio of an eye diagram. Color grade mode must be on for this measurement. *Agilent 83480A only.*

`:MEASure:CGRade:ERCalibrate`

Measures the dark level of a signal for use in measuring its extinction ratio. *Agilent 83480A only.*

`:MEASure:CGRade:ERFactor` {`CHANnel` <*number*>} {`ON` | `OFF`}, [<*value*>]

Corrects for the frequency response of the measurement system. *Agilent 83480A only. Requires firmware revision A.03.00 and above.*

`:MEASure:CGRade:EWIDth[?]`

Measures the eye width of an eye diagram. Color grade mode must be on for this measurement.

`:MEASure:CGRade:JITTer[?] {PP | RMS}`

Measures the jitter of an eye diagram. Color grade mode must be on for this measurement.

`:MEASure:CGRade:PEAK?`

Returns the maximum number of hits in a pixel of a color grade display.

`:MEASure:CGRade:QFACtor[?]`

Measures the Q factor of an eye diagram. Color grade mode must be on for this measurement. *Agilent 83480A only.*

`:MEASure:CLEar`

Clears the measurement results from the screen.

`:MEASure:DEFine[?]` *<meas_spec>*

Sets up the definition for measurements by specifying the delta time, threshold, or top-base values. Changing these values may affect other measure commands.

`:MEASure:DEFine CGRade, <!!signal_type!!>`

Specifies the signal type for the color grade measurement.

```
:MEASure:DEFine DELTatime, <!!start edge_direction!!>,<!!start
edge_number!!>,<!!start edge_position!!>,<!!stop
edge_direction!!>,<!!stop edge_number!!>,<!!stop
edge_postion!!>
```

Specifies the deltatime definition.

`:MEASure:DEFine EWINdow, <!!ewid1PCT!!>,<!!ewid2PCT!!>`

Specifies the eye window as a percentage of the bit period unit.

```
:MEASure:DEFine THResholds,{{STANdard}| {PERCent,
<upper_pct>,<middle_pct>,<lower_pct>} | T1090 | T2080
{VOLTage,<upper_volts>,<middle_volts>,<lower_volts>|}}
```

Specifies the threshold definition.

`:MEASure:DEFine TOPBase,{{STANdard}|{<top_volts>,<base_volts>}}`

Specifies the top-base definition.

`:MEASure:DELTatime[?]` [<*source*>[,<*source*>]]

Measures the delta time between two edges. If one source is specified, the delta time from the leading edge of the specified source to the trailing edge of the specified source is measured. If two sources are specified, the delta time from the leading edge on the first source to the trailing edge on the second source is measured.

`:MEASure:DUTYcycle[?]` [<*source*>]

Measures the ratio of the positive pulse width to the period. Sources are specified with the MEASure:SOURce command or with the optional parameter following the DUTYcycle command.

`:MEASure:FALLtime[?]` [<*source*>]

Measures the time at the upper threshold of the falling edge, measures the time at the lower threshold of the falling edge, then calculates the fall time. Sources are specified with the MEASure:SOURce command or with the optional parameter following the FALLtime command.

`:MEASure:FFT:DFRequency[?]` [<*source*>]

Enables the delta frequency measurement. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:DFRequency command.

`:MEASure:FFT:DMAGnitude[?]` [<*source*>]

Enables the delta magnitude measurement. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:DMAGnitude command.

`:MEASure:FFT:FREQuency[?]` [<*source*>]

Enables the frequency measurement. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:FREQuency command.

`:MEASure:FFT:MAGNitude[?]` [<*source*>]

Measures the magnitude of the FFT. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:MAGNitude command.

:MEASure:FFT:PEAK1[?] <*1st_peak_number*> [<*source*>]

Sets the peak number of the first peak for FFT measurements. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:PEAK1 command.

:MEASure:FFT:PEAK2[?] <*2nd_peak_number*> [<*source*>]

Sets the peak number of the second peak for FFT measurements. The source is specified with the MEASure:SOURce command or with the optional parameter following the FFT:PEAK2 command.

:MEASure:FFT:THReshold[?] <*threshold_value*>

Sets the peak search threshold value.

:MEASure:FREQuency[?] [<*source*>]

Measures the frequency of the first complete cycle on the screen using the mid-threshold levels of the waveform (50% levels if standard measurements are selected). The source is specified with the MEASure:SOURce command or with the optional parameter following the FREQuency command.

:MEASURE:HISTogram:HITS[?] [<*source*>]

Measures the number of hits within the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:MEAN[?] [<*source*>]

Measures the mean of the histogram. The mean of the histogram is the average value of all the points in the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:MEDian[?] [<*source*>]

Measures the median of the histogram. The median of the histogram is the time or voltage of the point at which 50% of the histogram is to the left or right (above or below for vertical histograms).

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:M1S[?] [<*source*>]

Measures the percentage of points that are within one standard deviation of the mean of the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:M2S[?] [<*source*>]

Measures the percentage of points that are within two standard deviations of the mean of the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:M3S[?] [<*source*>]

Measures the percentage of points that are within three standard deviations of the mean of the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:OFFSet? [<*source*>]

Returns the offset of the histogram in hits or decibels. The offset is in hits for linear-scaled histograms and in decibels for logarithmically-scaled histograms.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:PEAK[?] [<*source*>]

Measures the number of hits in the greatest peak of the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:PP[?] [<*source*>]

Measures the width of the histogram. The width is measured as the time or voltage of the last histogram bucket with data in it, minus the time or voltage of the first histogram bucket with data in it.

<*source*> *only available in firmware revision A.03.00 and above.*

:MEASURE:HISTogram:SCALe? [<*source*>]

Returns the scale of the histogram in hits or decibels per division. The scale is in hits for linear-scaled histograms and in decibels per division for logarithmically-scaled histograms.

<*source*> *only available in firmware revision A.03.00 and above.*

`:MEASURE:HISTogram:STDDev[?]` [<*source*>]

Measures the standard deviation of the histogram.

<*source*> *only available in firmware revision A.03.00 and above.*

`:MEASure:NWIDth[?]` [<*source*>]

Measures the width of the first negative pulse on the screen using the mid-threshold levels of the waveform (50% levels with standard measurements selected). Sources are specified with the MEASURE:SOURce command or with the optional parameter following the NWIDth command.

`:MEASure:OVERshoot[?]` [<*source*>]

Measures the overshoot of the first edge on the screen. Sources are specified with the MEASURE:SOURce command or with the optional parameter following the OVERshoot command.

`:MEASure:PERiod[?]` [<*source*>]

Measures the period of the first complete cycle on the screen using the mid-threshold levels of the waveform (50% levels with standard measurements selected). Sources are specified with the MEASURE:SOURce command or with the optional parameter following the PERiod command.

`:MEASure:PREShoot[?]` [<*source*>]

Measures the preshoot of the first edge on the screen. Sources are specified with the MEASURE:SOURce command or with the optional parameter following the PREShoot command.

`:MEASure:PWIDth[?]` [<*source*>]

Measures the width of the first positive pulse on the screen using the mid-threshold levels of the waveform (50% levels with standard measurements selected). Sources are specified with the MEASURE:SOURce command or with the optional parameter following the PWIDth command.

`:MEASure:RESults?`

Returns the results of the continuous measurements. The measurement results always include the current results only.

`:MEASure:RISetime[?]` [<*source*>]

Measures the rise time of the first displayed edge by measuring the time at the lower threshold of the rising edge, measuring the time at the upper threshold of the rising edge, then calculating the rise time.

`:MEASure:SCRatch`

Clears the measurement results from the screen.

`:MEASure:SENDvalid[?]` {`OFF` | `0` | `ON` | `1`}

Enables the result state code to be returned with the :MEASure:RESults? query.

`:MEASure:SOURce[?]` <*source*>[,<*source*>]

Selects the source for measurements. Two sources can be specified with this command. All measurements, except MEASure:DELTatime, are made on the first specified source. The delta time measurement uses two sources, if two are specified. If only one source is specified, the delta time measurement uses that source for both of its parameters.

`:MEASure:STATistics[?]` {`OFF` | `0`| `ON` | `1`}

Turns the statistics measurements on and off. The statistics state only affects the MEASure:RESults query.

`:MEASure:TEDGe[?]` <*meas_thres_txt*> ,<*slope*><*occurrence*> [,<*source*>]

Measures the time interval between the trigger event and the specified threshold level and transition.

`:MEASure:TMAX[?]` [<*source*>]

Measures the first time at which the maximum voltage of the source waveform occurred. When FFT is the specified source, the frequency at which the first maximum value occurred is measured.

`:MEASure:TMIN[?]` [<*source*>]

Measures the time at which the first minimum voltage occurred. When FFT is the specified source, the frequency at which the first minimum value occurred is measured.

`:MEASure:TVOLt[?]` <*voltage*>,<*slope*><*occurrence*> [,<*source*>]

Measures the time interval between the trigger event and the defined voltage level and transition.

:MEASure:VAMPlitude[?] [<*source*>]

Calculates the difference between the top and base voltage of the specified source.

:MEASure:VAVerage[?] {CYCLe | DISPlay} [, <*source*>]

Calculates the average voltage over the displayed waveform.

:MEASure:VBASe[?] [<*source*>]

Measures the statistical base of the waveform. Sources are specified with the MEASure:SOURce command or with the optional parameter following the VBASe command.

:MEASure:VLOWer[?] [<*source*>]

Measures the voltage value at the lower threshold of the waveform.

:MEASure:VMAX[?] [<*source*>]

Measures the absolute maximum voltage present on the selected source waveform. When FFT is the specified source, the maximum value in the spectrum is measured.

:MEASure:VMIDdle[?] [<*source*>]

Measures the voltage level at the middle threshold of the waveform. Sources are specified with the MEASure:SOURce command or with the optional parameter following the VMIDdle command.

:MEASure:VMIN[?] [<*source*>]

Measures the absolute minimum voltage present on the selected source waveform. When FFT is the specified source, the minimum value in the spectrum is measured.

:MEASure:VPP[?] [<*source*>]

Measures the maximum and minimum voltages on the selected source, then calculates the peak-to-peak voltage as the difference between the two voltages. When FFT is the specified source, the range of values in the spectrum are measured.

:MEASure:VRMS[?] {CYCLe | DISPlay}, {AC | DC} [,<*source*>]

Measures the RMS voltage of the selected waveform by subtracting the average value of the waveform from each data point on the display.

:MEASure:VTIMe[?] <*time*> [,<*source*>]

Measures the voltage at the specified time. The time is referenced to the trigger event and must be on screen. When FFT is the specified source, the value at the specified frequency is measured.

:MEASure:VTOP[?] [<*source*>]

Measures the statistical top of the selected source waveform.

:MEASure:VUPper[?] [<*source*>]

Measures the voltage value at the upper threshold of the waveform.

Pixel Memory Commands

`:PMEMory1:ADD`

Merges the current waveform(s) to the pixel memory. This is identical to the :PMEMory1:MERGe command.

`:PMEMory1:CLEar`

Clears the display memory. This is identical to the :PMEMory1:ERASe command.

`:PMEMory1:DISPlay[?] {ON | 1 | OFF | 0}`

Determines whether or not the pixel memory is viewable.

`:PMEMory1:ERASe`

Clears the pixel memory.

`:PMEMory1:MERGe`

Merges the current waveform(s) to the pixel memory.

Service Commands

`:SERVice:COMMents[?]PLUGin` <*number*><*comments*>

Returns the comments field associated with the plug-in.

`:SERVice:DECLassify`

Declassifies memory.

Timebase Commands

`:TIMebase:BRATe[?]` *<bit_rate>*

Sets the bit rate for use when the units are BITS. *Agilent 83480A only.*

`:TIMebase:DELay[?]` *<delay_value>*

Sets the time interval between the trigger event and the on-screen delay reference point. The delay reference point is set with the TIMebase:REFerence command.

`:TIMebase:POSition[?]` *<delay_value>*

Sets the time interval between the trigger event and the on-screen delay reference point. The delay reference point is set with the TIMebase:REFerence command.

`:TIMebase:RANGe[?]` *<full_scale_range>*

Sets the full-scale horizontal time in seconds. The range value is ten times the time per division value and covers the range of 1 ns (100 ps/div) to 10 seconds (1 s/div).

`:TIMebase:REFerence[?] {LEFT | CENTer | RIGHt}`

Sets the delay reference to the left, center, or right side for the main trace.

`:TIMebase:SCALe[?]` *<time>*

Sets the time base scale.

`:TIMebase:UNITs[?] {TIME | BITS}`

Selects the units for setting timebase parameters. *Agilent 83480A only.*

`:TIMebase:VIEW[?] {MAIN | WINDow | INTensified}`

Turns the windowed view on and off. If the windowed view is on, the main waveforms are not shown.

`:TIMebase:WINDow:DELay[?]` *<delay_position>*

Sets the position of the time-base window on the main sweep. The range for this command is determined by the main sweep range (or size) and the main sweep delay value. The value for this command must keep the time-base window within the main sweep range.

`:TIMebase:WINDow:POSition[?]` *<delay position>*

Sets the position of the time-base window on the main sweep. The range for this command is determined by the main sweep range (or size) and the main delay value. The value for this command must keep the time-base window on the main sweep range.

`:TIMebase:WINDow:RANGe[?]` *<full_scale_range>*

Sets the full-scale range of the time-base window. The range value is ten times the time per division of the window. The maximum range of the window is the current main range. The minimum window range is 10 ps.

`:TIMebase:WINDow:SCALe[?]` *<scale_value>*

Sets the per-division scale of the time-base window.

`:TIMebase:WINDow:SOURce[?] ALL`

Selects the source for the time-base window. The only parameter available for this command is ALL, which selects all sources displayed on the main screen.

Trigger Commands

`:TRIGger:HYSTeresis[?] {NORMal | HSENsitivity}`

Specifies the trigger hysteresis. NORMal is the typical hysteresis selection, High SENsitivity gives minimum hysteresis and highest bandwidth.

`:TRIGger:LEVel[?]` TRIGgerN,<*level*>

Specifies the trigger level on the specified trigger source.

`:TRIGger:SLOPe[?] {POSitive | NEGative}`

Specifies the slope of the edge on which to trigger. It sets the slope of the channel previously selected with the TRIGger:SOURce command.

`:TRIGger:SOURce[?] TRIGger`<*N*>

Selects the channel or trigger that produces the trigger.

`:TRIGger:SWEep[?] {TRIGgered | FREerun}`

Selects the sweep mode.

TriggerN Commands

`:TRIGger<`*N*`>:BWLimit[?] { ON | 1 | OFF | 0 | DIVided | GATE}`

Controls an internal lowpass filter in the plug-ins. When ON, the bandwidth of the specified trigger channel is limited to approximately 100 MHz.

Waveform Commands

`:WAVeform:BANDpass?`

Returns an estimation of the maximum and minimum bandwidth limits of the source signal. Bandwidth limits are computed as a function of the coupling and filter mode selected. Cutoff frequencies are derived from the acquisition path and software filtering.

`:WAVeform:BYTeorder[?] {MSBFirst | LSBFirst}`

Selects the order in which bytes are transferred to and from the instrument, over the HP-IB, for the WORD and LONG formats. If MSBFirst is selected, the most significant byte is transferred first. Otherwise, the least significant byte is transferred first. The default is MSBFirst.

`:WAVeform:COMPlete?`

Returns the percent of time buckets that are complete for the currently selected waveform.

`:WAVeform:COUNt?`

Returns the fewest number of hits in all of the time buckets for the currently selected waveform. For the AVERage waveform type, the count value is the fewest number of hits for all time buckets. This value may be less than or equal to the value specified with the ACQuire:COUNt command.

`:WAVeform:COUPling?`

Returns the input coupling of the currently selected waveform.

`:WAVeform:DATA[?]` <*block_data*>[,<*block_data*>]

Transfers waveform data to the instrument over HP-IB and stores the data in a previously specified waveform memory. The waveform memory is specified with the WAVeform:SOURce command. Only waveform memories may have waveform data sent to them. The format of the data being sent must match the format previously specified by the waveform preamble for the destination memory.

:WAVeform:FORMat[?] {ASCii | BYTE | LONG | WORD}

Sets the data transmission mode for waveform data output. This command controls how the data is formatted when the data is sent from the instrument and pertains to all waveforms.

:WAVeform:POINts?

Returns the points value in the current waveform preamble. The points value is the number of time buckets contained in the waveform selected with the WAVeform:SOURce command.

:WAVeform:PREamble[?] <*preamble_block*>

Sends a waveform preamble to the previously selected waveform memory in the instrument. The preamble contains the scaling and other values used to describe the data. The waveform memory is specified with the WAVeform:SOURce command. Only waveform memories may have waveform data sent to them.

:WAVeform:SOURce[?] {WMEMory<*number*> | FUNCtion<*number*> | CHANnel<*number*> | DATAbase | HISTogram | FFT}

Selects a channel, function, waveform memory, database, histogram, or FFT as the waveform source.

:WAVeform:TYPE? {NORMal | RAW | AVERage | DATAbase | HHIStogram | VHIStogram | VERSus}

Returns the current acquisition data type for the currently selected source. The type returned describes how the waveform was acquired.

:WAVeform:VIEW[?] {ALL | MAIN | WINDow}

Selects which view of the waveform is selected for data and preamble queries. The view has different meanings depending upon the waveform source selected.

:WAVeform:XDISplay?

Returns the X-axis value at the left edge of the display. For time domain signals, it is the time at the start of the display. For VERSus type waveforms, it is the value at the center of the X-axis of the display. This value is treated as a double precision 64-bit floating point number.

:WAVeform:XINCrement?

Returns the duration between data points on the x-axis. For time domain signals, this is the time difference between consecutive data points for the currently specified waveform source. For VERSus type waveforms, this is the duration between levels on the X-axis. For voltage waveforms, this is the voltage corresponding to one level.

:WAVeform:XORigin?

Returns the X-axis value of the first data point in the data record. For time domain signals, it is the time of the first point. For VERSus type waveforms, it is the X-axis value at level zero. For voltage waveforms, it is the voltage at level zero. The value returned by this query is treated as a double precision 64-bit floating point number.

:WAVeform:XRANge?

Returns the X-axis duration of the displayed waveform. For time domain signals, it is the duration of the time across the display. For VERSus type waveforms, it is the duration of the waveform that is displayed on the X-axis.

:WAVeform:XREFerence?

Returns the data point or level associated with the Xorigin data value. It is at this data point or level that the X origin is defined. In this instrument, the value is always zero.

:WAVeform:XUNits?

Returns the X-axis units of the currently selected waveform source. The currently selected source may be a channel, function, database, histogram, or waveform memory.

:WAVeform:YDISplay?

Returns the Y-axis value at the center of the display. For voltage signals, it is the voltage at the center of the display.

:WAVeform:YINCrement?

Returns the duration between the Y-axis levels.

:WAVeform:YORigin?

Returns the Y-axis value at level zero.

:WAVeform:YRANge?

Returns the Y-axis duration of the displayed waveform. For voltage signals, it is the amount of voltage across the display.

:WAVeform:YREFerence?

Returns the level associated with the Y origin. It is at this level that the Y origin is defined. In this instrument, the value is always zero.

:WAVeform:YUNits?

Returns the Y-axis units of the currently selected waveform source. The currently selected source may be a channel, function, database, histogram, or waveform memory.

Waveform Memory Commands

`:WMEMory<N>:DISPlay[?] {ON | 1 | OFF | 0}`

Enables or disables the viewing of the selected memory.

`:WMEMory<N>:SAVE { CHANnel<N> | WMEMory<N> | FUNCtion<N> | HISTogram | FFT}`

Stores the specific channel, waveform, memory, function, histogram, or FFT waveform to the waveform memory.

`:WMEMory<N>:XOFFset[?] <offset>`

Sets the X-axis (horizontal) offset for the selected waveform memory's display scale. Offset is referenced to center screen.

`:WMEMory<N>:XRANge[?] <range>`

Sets the X-axis (horizontal) range for the selected waveform memory's display scale.

`:WMEMory<N>:YOFFset[?] <offset>`

Sets the X-axis (horizontal) offset for the selected waveform memory's display scale. Offset is referenced to center screen.

`:WMEMory<N>:YRANge[?] <range>`

Sets the X-axis (horizontal) range for the selected waveform memory's display scale.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP