V OSCILOSCOPIO HP-54600 y HP-54621

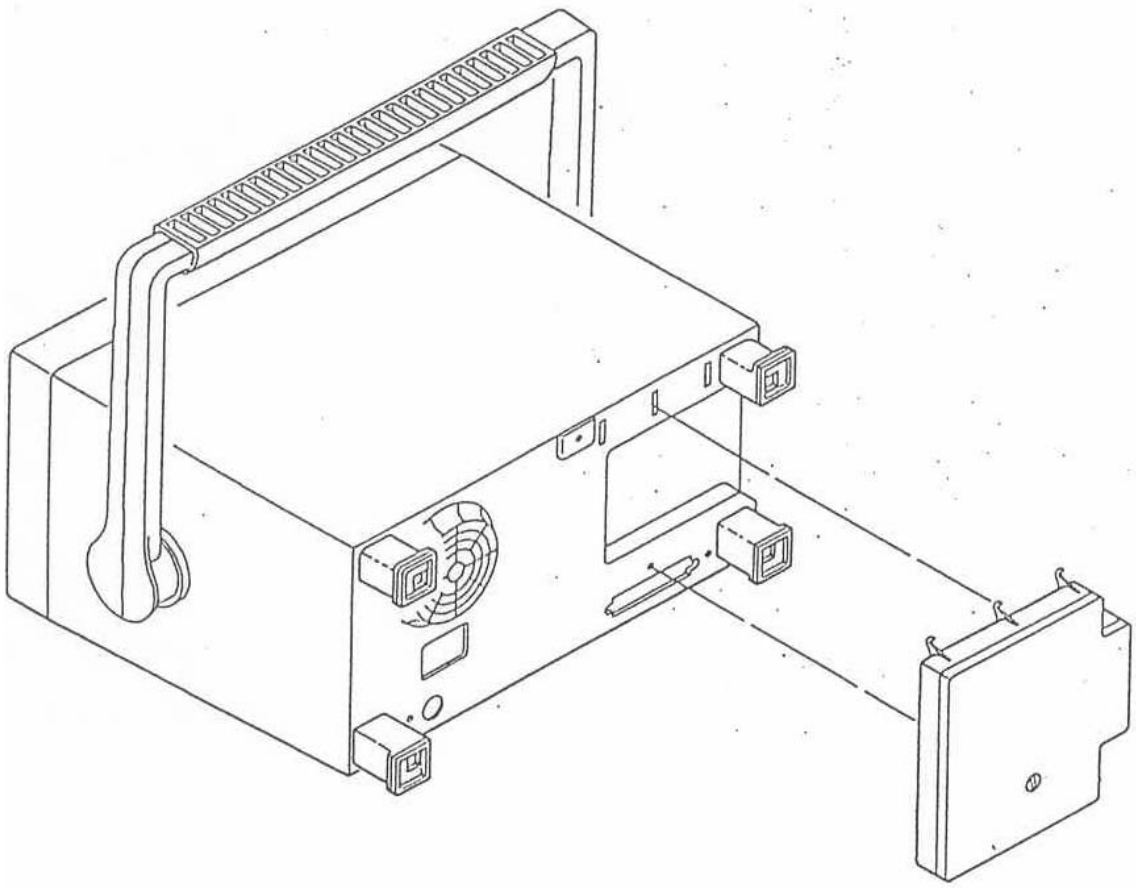
V.1 Comunicación con los osciloscopios de la familia HP-5462x.

El osciloscopio HP54600 puede se gestionado de forma remota desde un computador utilizando la interfaz para el bus GPIB. A través de la interfaz se pueden realizan operaciones básicas:

- Controlar el modo de operación del osciloscopio.
- Ejecutar operaciones de medida con el osciloscopio.
- Adquirir la información generada por el osciloscopio (formas de ondas, medidas, configuraciones, etc)
- Enviar información para ser representada en el osciloscopio (Imagen de pixels, configuración, etc).

La comunicación entre el computador que actúa como controlador, y el osciloscopio se realiza mediante intercambios de mensajes, siguiendo el estándar IEEE-488.2.

Para que el osciloscopio pueda operar de forma remota, necesita que esté dotado de la interfaz HP-IB (que es optativa).



Para que el osciloscopio responda a los comandos enviados a través del bus GPIB, se requiere que previamente haya sido programada su interfaz. Esto se lleva a cabo desde el menú correspondiente al **botón Print/Utility**. Elegimos el menú **I/O,** a través del cual se debe establecer:

1) Interfaz GPIB: Presionando la tecla Controller establecemos el tipo de interfaz que se va a emplear, en este caso GPIB.

2) Dirección GPIB: Presionando la tecla Address, y moviendo el botón Entry, establecemos la dirección GPIB, que será un entero en el rango 0-30.

Inicialización del Osciloscopio

Antes de comenzar a intercambiar mensajes con el osciloscopio, se debe inicializar tanto su interfaz, como el programa Parser. Esto se hace mediante las dos operaciones:

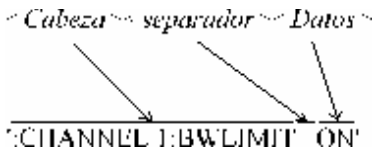
```
clrdevice(oscilo);
                % Inicializa la interfaz del instrumento, e inicializa el
                % programa de control Parser a su estado base

fprintf(oscilo, '*CLS');
                % Despeja el registro de estado y de incidencias.


fprintf(oscilo,'*RST');
                % Establece el Osciloscopio a su estado de inicialización
                % (preset)
```

Estructura de los mensajes de orden

Los mensajes que deben ser enviados para ejecutar una orden en el osciloscopio, son strings alfanuméricos, con la siguiente estructura:



Cabeza: es el identificador de la orden que se ejecuta. Está compuesta de una cadena de campos separados por delimitadores ":". Cada campo identifica un nivel dentro del árbol de ordenes, y en conjunto, identifican unívocamente una orden.

El identificador de un nivel se compone de unos caracteres obligatorios, y otros que pueden incluirse optativamente, a efecto de mayor legibilidad del programa.

CHANnel $\Rightarrow$ CHAN $\Leftrightarrow$ CHANN $\Leftrightarrow$ CHANNE $\Leftrightarrow$ CHANNEL

Los identificadores pueden expresarse indiferentemente con mayúsculas o minúsculas

CHANNEL $\Leftrightarrow$ CHANnel $\Leftrightarrow$ Channel $\Leftrightarrow$channel

Las cabezas de las ordenes de requerimiento finalizan en el carácter "?".

':MEASURE:FREQUENCY?'

Las cabezas de las ordenes básicas del protocolo IEEE-488.2 no están incluidas en el árbol y siempre empiezan por el carácter "*".

'*RST'

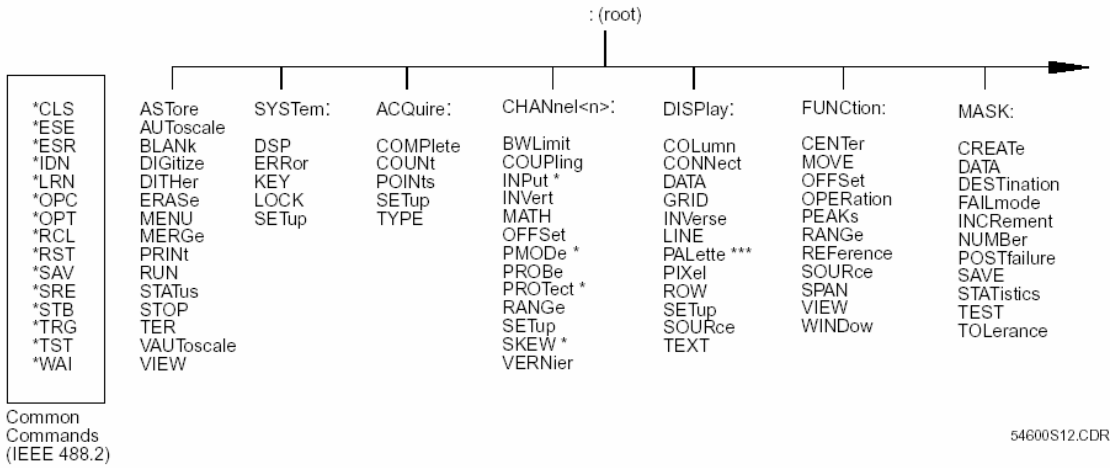
Delimitador: Es un espacio o conjunto de espacios.

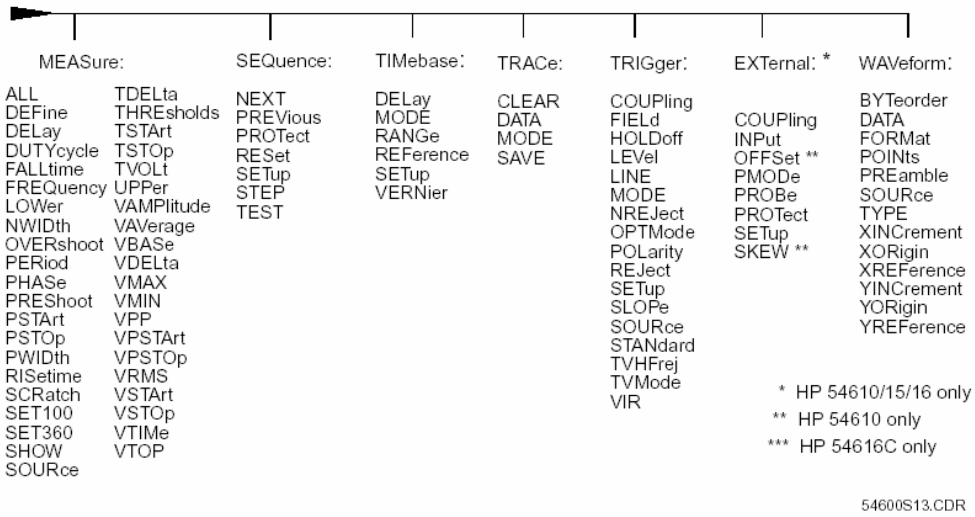
Datos de programas: Son valores numéricos o enumerados que cualifican la orden. Pueden ser uno o varios, según la orden de que se trate. En este caso los datos deben estar separados por comas o combinaciones de comas y espacios.

Los valores numéricos pueden expresarse en notación decimal o científica:

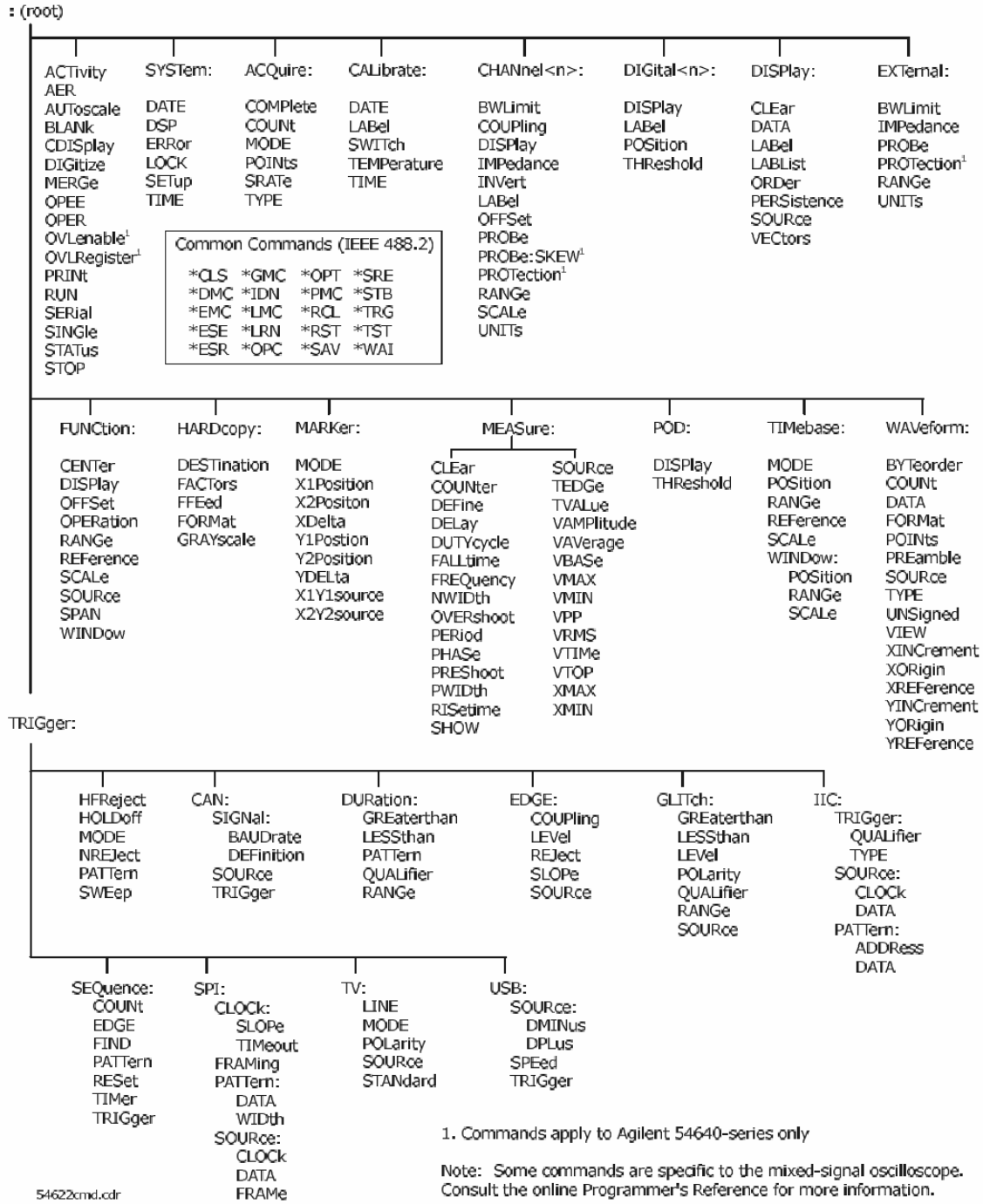
28 = 0.28E2 = 280e-1 = 28000m = 0.028K = 28e-3K

Árbol de comandos del osciloscopio HP 54600





Árbol de comandos del osciloscopio 5462x



V.2 Órdenes específicas del osciloscopio HP5462x.

Commands and Queries

The following tables facilitate easy access to each command and query for the oscilloscopes. The commands and queries are divided into separate categories with each entry alphabetized.

The arguments for each command list the minimum argument required. The part of the command or query listed in uppercase letters refers to the short form of that command or query. The long form is the combination of the uppercase and lowercase letters. The NR1 and NR3 formats refer only to the Query Return values. Input arguments are not restricted by these formats.

These commands also show specific information about how the command operates on a particular oscilloscope model. For additional information, refer to the online oscilloscopes *Programmer's Reference*.

Command	Query	Options and Query Returns
:ACQuire:COMPlete <complete>	:ACQuire:COMPlete?	<complete> ::= 100; an integer in NR1 format
:ACQuire:COUNt <count>	:ACQuire:COUNT?	<count> ::= an integer from 1 to 16384 in NR1 format
:ACQuire:MODE <mode>	:ACQuire:MODE?	<mode> ::= {RTIMe | ETIMe}
n/a	**:ACQuire:POINts?**	2,000; an integer in NR1 format.
n/a	**:ACQuire:SRATe?**	<point_argument> ::=sample rate (samples/s) in NR3 format
:ACQuire:TYPE <type>	:ACQuire:TYPE?	<type> ::= {NORMal | AVERage | PEAK}
:ACTivity	:ACTivity?	<return value> ::= <edges>, <levels> <edges> ::= presence of edges (32-bit integer in NR1 format) <levels> ::= logical highs or lows (32-bit integer in NR1 format)
n/a	**:AER?**	{0 | 1}; an integer in NR1 format
:AUToscale	n/a	n/a
:BLANk <source>	n/a	<source> ::= {CHANnel<n> | POD{1 | 2} | FUNCtion} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | POD{1 | 2} | FUNCtion} for 5462xD <n> ::= 1-2 or 1-4; an integer in NR1 format
n/a	**:CALibrate:DATE?**	<return value> ::= <day>,<month>,<year>; all in NR1 format
:CALibrate:LABel <string>	:CALibrate:LABel?	<string> ::= quoted ASCII string up to 32 characters
n/a	**:CALibrate:SWITch?**	{PROTected | UNPRotected}
n/a	**:CALibrate:TIME?**	<return value> ::= <hours>,<minutes>,<seconds>; all in NR1 format
:CDISplay	n/a	n/a
:CHANnel<n>:BWLimit {{0 | OFF} | {1 | ON}}	:CHANnel<n>:BWLimit?	{0 | 1}; <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:COUPling <coupling>	:CHANnel<n>:COUPling?	<coupling> ::= {AC | DC | GND}; <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:DISPlay {{0 | OFF} | {1 | ON}}	:CHANnel<n>:DISPlay?	{0 | 1}; <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:IMPedence <impedence>	:CHANnel<n>:IMPedence?	<impedence> ::= {ONEMeg}; <n> ::= 1-2 or 1-4; an integer in NR1 format

Command	Query	Options and Query Returns
:CHANnel<n>:INVert {{0 | OFF} | {1 | ON}}	:CHANnel<n>:INVert?	{0 | 1}; <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:LABel <string>	:CHANnel<n>:LABel?	<string>::= any series of 6 or less ASCII characters enclosed in quotation marks <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:OFFSet <offset> [suffix]	:CHANnel<n>:OFFSet?	<offset> ::= Vertical offset value in NR3 format. [suffix] ::= {V | mV} <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:PROBe <attenuation>	:CHANnel<n>:PROBe?	<attenuation> ::= Probe attenuation ratio in NR3 format ::= X1, X10, X20, X100 (obsolete form) <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:RANGe <range> [suffix]	:CHANnel<n>:RANGe?	<range> ::= Vertical full-scale range value in NR3 format. [suffix] ::= {V | mV} <n> ::= 1-2 or 1-4; an integer in NR1 format
:CHANnel<n>:SCALe <scale> [suffix]	:CHANnel<n>:SCALe?	<scale> ::= Vertical units per division value in NR3 format. [suffix] ::= {V | mV} <n> ::= 1-2 or 1-4; an integer in NR1 format
***CLS**	n/a	n/a
:DIGital<n>:DISPlay {{0 | OFF} | {1 | ON}}	:DIGital<n>:DISPlay?	{0 | 1}; <n> ::= 0-15; an integer in NR1 format
:DIGital<n>:LABel <string>	:DIGital<n>:LABel?	<string>::= any series of 6 or less ASCII characters enclosed in quotation marks <n> ::= 0-15; an integer in NR1 format
:DIGital<n>:POSition <position>	:DIGital<n>:POSition?	<n> ::= 0-15; an integer in NR1 format <position> ::= 1-8 if display size = large, 1-16 if size = medium, 1-32 if size = small
:DIGital<n>:THReshold <value>[suffix]	:DIGital<n>:THReshold?	<n> ::= 0-15; an integer in NR1 format <value> ::= {CMOS | ECL | TTL | <user defined value>} <user defined value> ::= value in NR3 format from -8.00 to +8.00 [suffix] ::= {V | mV | uV}
:DIGitize [<source>[,...,<source>]]	n/a	<source> ::= {CHANnel<n> | POD1 | POD2 | FUNCtion} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | POD1 | POD2 | FUNCtion} for 5462xD <source> can be repeated up to 5 times. <n> ::= 1-2 or 1-4; an integer in NR1 format
:DISPlay:CLEar	n/a	n/a
:DISPlay:DATA [format][,][area] <binary block data>	:DISPlay:DATA? [format][,][area]	<format> ::= {TIFF} (command only) <area> ::= {GRATicule} (command only) <format> ::= {TIFF | BMP} (query only) <area> ::= {GRATicule | SCReen} (query only) <binary block_data> ::= data in IEEE 488.2 # format
:DISPlay:LABel {{0 | OFF} | {1 | ON}}	:DISPlay:LABel?	{0 | 1}
:DISPlay:LABList #80000524 <binary block>	:DISPlay:LABList?	<binary block> ::= a time-ordered list of 75 labels. Each label can be a maximum of 6 characters followed by a comma.
:DISPlay:PERSistence <value>	:DISPlay:PERSistence?	<value> ::= {MINimum | INFinite}}
:DISPlay:SOURce <value>	:DISPlay:SOURce?	<value> ::= {PMEMory{0 | 1 | 2}}
:DISPlay:VECTors {{1 | ON} | {0 | OFF}}	:DISPlay:VECTors?	{1 | 0}

Command	Query	Options and Query Returns
***DMC <macro label>,<macro definition>**	n/a	<macro label> ::= quoted ASCII string <macro definition> ::= block data in IEEE 488.2 # format
***EMC {{0 | OFF} | {1 | ON}}**	*EMC?	{0 | 1}
***ESE <mask>**	*ESE?	<mask> ::= 0 to 255; an integer in NR1 format Bit Weight Enables 7 128 PON - Power On 6 64 URQ - User Request 5 32 CME - Command Error 4 16 EXE - Execution Error 3 8 DDE - Device Dependent Error 2 4 QYE - Query Error 1 2 TRG - Trigger Query 0 1 OPC - Operation Complete
n/a	***ESR?**	<status> ::= 0 to 255; an integer in NR1 format
:FUNCtion:CENTer <frequency>	:FUNCtion:CENTer?	<frequency> ::= the current center frequency in NR3 format. The range of legal values is from 0 Hz to 25.00 GHz.
:FUNCtion:DISPlay {{0 | OFF} | {1 | ON}}	:FUNCtion:DISPlay?	{0 | 1}
:FUNCtion:OFFSet <offset>	:FUNCtion:OFFSet?	<offset> ::= the value at center screen in NR3 format. The range of legal values is +/-10 times the current sensitivity of the selected function.
:FUNCtion:OPERation <operation>	:FUNCtion:OPERation?	<operation> ::= {SUBTract | MULTiply | INTegrate | DIFFerentiate | FFT}
:FUNCtion:RANGe <range>	:FUNCtion:RANGe?	<range> ::= the full-scale vertical axis value in NR3 format. The range for ADD, SUBT, MULT is 8E-6 to 800E+3. The range for the INTegrate function is 8E-9 to 400E+3. The range for the DIFFerentiate function is 80E-3 to 8.0E12 (depends on current sweep speed). The range for the FFT function is 8 to 800 dBV.
:FUNCtion:REFerence <level>	:FUNCtion:REFerence?	<level> ::= the current reference level in NR3 format. The range of legal values is from 400.0 dBV to +400.0 dBV (depending on current range value).
:FUNCtion:SCALe <scale value>[<suffix>]	:FUNCtion:SCALe	<scale value> ::= integer in NR1 format <suffix> ::= {V | dB}
:FUNCtion:SOURce <source>	:FUNCtion:SOURce?	<source> ::= {CHANnel<n> | ADD | SUBT | MULT}; <n> ::= 1-2 or 1-4 in NR1 format
**:FUNCtion:SPAN **	:FUNCtion:SPAN?	::= the current frequency span in NR3 format. Legal values are 1 Hz to 100 GHz
:FUNCtion:WINDow <window>	:FUNCtion:WINDow?	<window> ::= {RECTangular | HANNing | FLATtop}
n/a	***GMC? <macro label>**	<macro label> ::= quoted ASCII string, block data in IEEE 488.2 # format
:HARDcopy:DESTination <destination>	:HARDcopy:DESTination	<destination> ::= {CENTronics | FLOPpy}
:HARDcopy:FACTors {{0 | OFF} | {1 | ON}}	:HARDcopy:FACTors?	{0 | 1}
:HARDcopy:FFEed {{0 | OFF} | {1 | ON}}	:HARDcopy:FFEed?	{0 | 1}
:HARDcopy:FORMat <device>	:HARDcopy:FORMat?	<format> ::= {TIFF | BMP | CSV | LASerjet | DESKjet | EPSon | SEIKo}

Command	Query	Options and Query Returns
:HARDcopy:GRAYscale {{0 | OFF} | {1 | ON}}	:HARDcopy:GRAYscale?	{0 | 1}
n/a	***IDN?**	AGILENT TECHNOLOGIES,<model>, <serial number>,X.XX.XX <model> ::= the model number of the instrument <serial number> ::= the serial number of the instrument <X.XX.XX> ::= the software revision of the instrument
n/a	***LMC?**	<ascii string> ::= string list seperated by commas
n/a	***LRN?**	<learn_string> ::= current instrument setup as a block of data in IEEE 488.2 # format
:MARKer:MODE<mode>	:MARKer:MODE?	<mode> ::= {OFF | MEASure | MANual}
:MARKer:X1Position <position> [suffix]	:MARKer:X1Position?	<position> ::= X1 cursor position value in NR3 format [suffix] ::= {s | ms | ns | ps | Hz | kHz | MHz} <return_value> ::= X1 cursor positon value in NR3 format
:MARKer:X1Y1source <source>	:MARKer:X1Y1 <source>	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= <source>
:MARKer:X2Position <position> [suffix]	:MARKer:X2Position?	<position> ::= X2 cursor position value in NR3 format [suffix] ::= {s | ms | ns | ps | Hz | kHz | MHz} <return_value> ::= X2 cursor positon value in NR3 format
:MARKer:X2Y2source <source>	:MARKer:X2Y2 <source>	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= <source>
n/a	**:MARker:XDELta?**	<return_value> ::= X cursors delta value in NR3 format
:MARKer:Y1Position <positon>[suffix]	:MARKer:Y1Position?	<positon> ::= Y1 cursor position value in NR3 format [suffix] ::= {V | mV | dB} <return_value> ::= Y1 cursor positon value in NR3 format
:MARKER:Y2Position <positon>[suffix]	:MARKer:Y2Position?	<positon> ::= Y2 cursor position value in NR3 format [suffix] ::= {V | mV | dB} <return_value> ::= Y2 cursor positon value in NR3 format
n/a	**:MARKer:YDELta?**	<return_value> ::= Y cursors delta value in NR3 format
:MEASure:CLEar	n/a	n/a
:MEASure:DEFine <measurement>, <edge spec1, edge spec2>	:MEASure:DEFine?	<measurement> ::= DELay edge_spec1 ::= [<slope>] <occurrence> edge_spec2 ::= [<slope>] <occurrence> <slope> ::= {+ | -} <occurrence> ::= integer from 1 to 5
:MEASure:DELay [<source>]	:MEASure:DELay? [<source>]	<source> ::= {CHANnel1 | CHANnel2} <return_value> ::= floating-point number delay time in seconds in NR3 format
:MEASure:DUTYcycle [<source>]	:MEASure:DUTYcycle? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | FUNCtion | MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= ratio of positive pulse width to period in NR3 format

Command	Query	Options and Query Returns
:MEASure:FALLtime [<source>]	:MEASure:FALLtime? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= time in seconds between the lower and upper thresholds in NR3 format
:MEASure:FREQuency [<source>]	:MEASure:FREQuency? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= frequency in Hertz in NR3 format
:MEASure:NWIDth [<source>]	:MEASure:NWIDth? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= negative pulse width in seconds-NR3 format
:MEASure:OVERshoot [<source>]	:MEASure:OVERshoot? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= the percent of the overshoot of the selected waveform in NR3 format
:MEASure:PERiod [<source>]	:MEASure:PERiod? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= waveform period in seconds in NR3 format
:MEASure:PHASe [<source>]	:MEASure:PHASe? [<source>]	<source> ::= {CHANnel1 I CHANnel2} <return_value> ::= the phase angle value in degrees in NR3 format
:MEASure:PREShoot [<source>]	:MEASure:PREShoot? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= the percent of preshoot of the selected waveform in NR3 format
:MEASure:PWIDth [<source>]	:MEASure:PWIDth? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= width of positive pulse in seconds in NR3 format
:MEASure:RISEtime [<source>]	:MEASure: RISEtime? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= rise time in seconds in NR3 format
:MEASure:SHOW {1 I ON}	:MEASure:SHOW?	{1}
:MEASure:SOURce [<source>]	:MEASure:SOURce?	<source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format for HP5462xA <return_value> ::= {<source> I <none>} <none> ::= query returns "NONE" if all channels are off

Command	Query	Options and Query Returns
:MEASure:TEDGe <slope><occurrence>[,<source>]	:MEASure:TEDGe?	<slope> ::= direction of the waveform <occurrence> ::= the transition to be reported. <source> ::= {CHANnel<n> | FUNCtion | MATH} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | FUNCtion | MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format for HP5462xA <return_value> ::= time in seconds of the specified transition
:MEASure:TMAX [<source>]	:MEASure:TMAX?	<source> ::= {CHANnel<n> | FUNCtion | MATH} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | FUNCtion | MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format for HP5462xA
n/a	**:MEASure:TVALue? <value>, [<slope>]<occurrence> [,<source>]**	<value> ::= voltage level that the waveform must cross. <slope> ::= direction of the waveform when <value> is crossed. <occurrence> ::= transitions reported. <return_value> ::= time in seconds of specified voltage crossing in NR3 format <source> ::= {CHANnel<n> | FUNCtion | MATH} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | FUNCtion | MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format
:MEASure:VAMPlitude [<source>]	:MEASure:VAMPlitude? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= the amplitude of the selected waveform in volts in NR3 format
:MEASure:VAVerage [<source>]	:MEASure:VAVerage? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= calculated average voltage in NR3 format
:MEASure:VBASe [<source>]	:MEASure:VBASe? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <base_voltage> ::= voltage at the base of the selected waveform in NR3 format
:MEASure:VMAX [<source>]	:MEASure:VMAX? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= maximum voltage of the selected waveform in NR3 format
:MEASure:VMIN [<source>]	:MEASure:VMIN? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= minimum voltage of the selected waveform in NR3 format
:MEASure:VPP [<source>]	:MEASure:VPP? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= voltage peak-to-peak of the selected waveform in NR3 format
:MEASure:VRMS [<source>]	:MEASure:VRMS? [<source>]	<source> ::= {CHANnel<n> | FUNCtion | MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= calculated dc RMS voltage in NR3 format

Command	Query	Options and Query Returns
n/a	**:MEASure:VTIMe? <vtime>[,<source>]**	<vtime> ::= displayed time from trigger in seconds in NR3 format <return_value> ::= voltage at the specified time in NR3 format <source> ::= {CHANnel<n> I FUNCtion I MATH} for 5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I FUNCtion I MATH} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format
:MEASure:VTOP [<source>]	:MEASure:VTOP? [<source>]	<source> ::= {CHANnel<n> I FUNCtion I MATH} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= voltage at the top of the waveform in NR3 format
:MEASure:XMAX [<source>]	:MEASure:XMAX? [<source>]	<source> ::= {CHANnel<n> I FUNCtion} <n> ::= 1-2 or 1-4 in NR1 format <return_value> ::= horizontal value of the maximum in NR3 format
:MERGe <pixel memory>	n/a	<pixel memory> ::= {PMEMory{0 I 1 I 2}}
***OPC**	*OPC?	ASCII ""1"" is placed in the output queue when all pending device operations have completed.
:OPEE <n>	:OPEE?	<n> ::= 16-bit integer in NR1 format
n/a	**:OPER?**	<n> ::= 16-bit integer in NR1 format
n/a	***OPT?**	<return_value> ::= n, A.XX.XX n identifies the module. XX.XX identifies the module software revision. N2757A, A.XX.XX
***PMC**	n/a	n/a
:POD<n>:DISPlay {{0 I OFF} I {1 I ON}}	:POD<n>:DISPlay?	{0 I 1}; <n> ::= 1-2; an integer in NR1 format
:POD<n>:THReshold <value>[suffix]	:POD<n>:THReshold?	<n> ::= 1-2; an integer in NR1 format <value> ::= {CMOS I ECL I TTL I <user defined value>} <user defined value> ::= value in NR3 format [suffix] ::= {V I mV I uV }
:PRINt [parameter][,parameter]]	:PRIN? [parameter][,parameter] x 4]	<parameter> ::= {HIRes I LORes I TIFF I PCL I BMP I PARallel I DISK I FACTors I NOFactors} <parameter> can be repeated up to 5 times.
***RCL <value>**	n/a	<value> ::= {0 I 1 I 2}
***RST**	n/a	See reset values in the online Programmer's Reference.
:RUN	n/a	n/a
***SAV <value>**	n/a	<value> ::= {0 I 1 I 2}
	:SERial?	<return value> ::= unquoted string containing serial number
:SINGle	n/a	n/a
***SRE <mask>**	*SRE?	<mask> ::= sum of all bits that are set, 0 to 255; an integer in NR1 format. <mask> ::= following values: Bit Weight Enables 7 128 OPER - Operation Status Register 6 64 Not Used 5 32 ESB - Event Status Bit 4 16 MAV - Message Available 3 8 Not used 2 4 MSG - Message 1 2 USR - User 0 1 TRG - Trigger

Command	Query	Options and Query Returns
n/a	**:STATus? <display>**	{0 | 1} <display> ::= {CHANnel<n> | DIGital0,...,DIGital15 | FUNCtion}; <n> ::= 1-2 or 1-4 in NR1 format
n/a	***STB?**	<value> ::= 0 to 255; an integer in NR1 format, as shown in the following: Bit Weight Name Condition 7 128 OPER 0 = no enabled operations status conditions occurred 1 = an enabled operation status condition occurred 6 64 RQS/MS 0 = instrument has no reason for service 1 = instrument is requesting service 5 32 ESB 0 = no event status conditions occurred 1 = enabled event status condition occurred 4 16 MAV 0 = no output messages are ready 1 = an output message is ready 3 8 ---- 0 = not used 2 4 MSG 0 = no message has been displayed 1 = message has been displayed 1 2 USR 0 = no enabled user event conditions have occurred 1 = an enabled user event condition has occurred 0 1 TRG 0 = no trigger has occurred 1 = a trigger occurred
:STOP	n/a	n/a
:SYSTem:DATE <date>	:SYSTem:DATE?	date> ::= <year>, <month>, <day> <date> ::= <year>, <month>, <day> <year> ::= 4-digit year in NR1 format <month> ::= {1,...12 | JANuary | FEBruary | MARch | APRil | MAY | JUNe | JULy | AUGust | SEPtember | OCTober | NOVember | DECember} <day> ::= {1,..31}
:SYSTem:DSP <string>	n/a	<string> ::= up to 254 characters as a quoted ASCII string
n/a	**:SYSTem:ERRor?**	<error> ::= an integer error code <error string> ::= quoted ASCII string. See error values in the online Programmer's Reference.
:SYSTem:LOCK	:SYSTem:LOCK?	<value> ::= {ON | OFF}
:SYSTem:SETup <setup_data>	:SYSTem:SETup?	<setup_data> ::= data in IEEE 488.2 # format.
:SYSTem:TIME <time>	:SYSTem:TIME?	<time> ::= hours, minutes, seconds in NR1 format
n/a	**:TER?**	{0 | 1}
:TIMebase:MODE <value>	:TIMebase:MODE?	<value> ::= {MAIN | WINDow | XY | ROLL}
:TIMebase:POSition <pos>	:TIMebase:POSition?	<pos> ::= time from the trigger event to the display reference point in NR3 format
:TIMebase:RANGe <range_value>	:TIMebase:RANGe?	<range_value> ::= 50 ns through 500 s in NR3 format

Command	Query	Options and Query Returns
:TIMebase:REFerence {LEFT | CENTer | RIGHt}	:TIMebase:REFerence?	<return_value> ::= {LEFT | CENTer | RIGHt}
:TIMebase:SCALe <scale_value>	:TIMebase:SCALe?	<scale_value> ::= scale value in seconds in NR3 format
:TIMebase:WINDow:POSition <pos>	:TIMebase:WINDow:POSition?	<pos> ::= time from the trigger event to the delayed view reference point in NR3 format
:TIMebase:WINDow:RANGe <range_value>	:TIMebase:WINDow:RANGe?	<range value> ::= range value in seconds in NR3 format for the delayed window
:TIMebase:WINDow:SCALe <scale_value>	:TIMebase:WINDow:SCALe?	<scale_value> ::= scale value in seconds in NR3 format for the delayed window
*TRG	n/a	n/a
:TRIGger:HFReject {{0 | OFF}|{1 | ON}}	:TRIGger:HFReject?	{0 | 1}
:TRIGger:HOLDoff <holdoff_time>	:TRIGger:HOLDoff?	<holdoff_time> ::= 60 ns to 10 s in NR3 format
:TRIGger:MODE <mode>	:TRIGger:MODE?	<mode> ::= {EDGE | TV | GLITch | PATTern | SEQuence | DURation | IIC} <return_value> ::= {<mode> | <none>} <none> ::= query returns "NONE" if the :TIMebase:MODE is ROLL or XY
:TRIGger:NREJect {{0 | OFF}|{1 | ON}}	:TRIGger:NREJect?	{0 | 1}
:TRIGger:PATTern <value>, <mask>, [<source>, <edge>]	:TRIGger:PATTern?	<value> ::= integer or <string> <mask> ::= integer or <string> <string> ::= ""0xnnnnnn"" n ::= {0 ,....., 9 | A ,....., F} <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | NONE}; <n> ::= 1-2 or 1-4 in NR1 format <edge> ::= {POSitive | NEGative }
:TRIGger:SWEep <sweep>	:TRIGger:SWEep?	<sweep> ::= {AUTLevel | AUTO | NORMal}
:TRIGGER:DURation:GREaterthan <greater than time> [suffix]	:TRIGger:DURation:GREaterthan?	<greater than time> ::= floating-point number from 5 ns to 10seconds in NR3 format [suffix] ::= {s | ms | us | ns | ps}
:TRIGGER:DURation:LESSthan <less than time> [suffix]	:TRIGger:DURation:LESSthan?	<less than time> ::= floating-point number from 5 ns to 10seconds in NR3 format [suffix] ::= {s | ms | us | ns | ps}
:TRIGger:DURation:PATTern <value>, <mask>	:TRIGger:DURation:PATTern?	<value> ::= integer or <string> <mask> ::= integer or <string> <string> ::= ""0xnnnnnn"" n ::= {0 ,....., 9 | A ,....., F}
:TRIGger:DURation:QUALifier <qualifier>	:TRIGger:DURation:QUALifier?	<qualifier> ::= {GREaterthan | LESSthan | INRange | OUTRange | TIMeout}
:TRIGger:DURation:RANGe <greater than time> [suffix], <less than time> [suffix]	:TRIGger:DURation:RANGe?	<greater than time> ::= min duration from 10 ns to 9.99 seconds in NR3 format <less than time> ::= max duration from 15 ns to 10 seconds in NR3 format [suffix] ::= {s | ms | us | ns | ps}
TRIGger:[EDGE:]COUPling {AC | DC | LF}	:TRIGger:COUPling?	{AC | DC | LF}

Command	Query	Options and Query Returns
:TRIGger:[EDGE:]LEVel <level> [,<source>]	:TRIGger:[EDGE]:LEVel? [<source>]	For internal triggers, <level> ::= .75 x full-scale voltage from center screen in NR3 format. For external triggers (HP5462xA only), <level> ::= 2 volts with probe attenuation at 1:1 in NR3 format. For digital channels (HP5462xD only), <level> ::= 8 V. <source> ::= {CHANnel<n> I EXTernal I LINE} for HP5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I LINE} for HP5462xD <n> ::= 1-2 or 1-4 in NR1 format
:TRIGger:[EDGE:]REJect {OFF I LF I HF}	:TRIGger:REJect?	{OFF I LF I HF}
:TRIGger:[EDGE:]SLOPe <polarity>	:TRIGger:[EDGE]:SLOPe?	<polarity> ::= {POSitive I NEGative}
:TRIGger:[EDGE:]SOURce <source>	:TRIGger:[EDGE]:SOURce?	<source> ::= {CHANnel<n> I EXTernal I LINE} for HP5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I LINE} for HP5462xD <n> ::= 1-2 or 1-4 in NR1 format
:TRIGGER:GLITch:GREaterthan <greater than time> [suffix]	:TRIGger:GLITch:GREaterthan?	<greater than time> ::= floating-point number from 5 ns to 10 seconds in NR3 format [suffix] ::= {s I ms I us I ns I ps}
:TRIGGER:GLITch:LESSthan <less than time> [suffix]	:TRIGger:GLITch:LESSthan?	<less than time> ::= floating-point number from 5 ns to 10 seconds in NR3 format [suffix] ::= {s I ms I us I ns I ps}
:TRIGger:GLITch:LEVel <level> [<source>]	:TRIGger:GLITch:LEVel?	For internal triggers, <level> ::= .75 x full-scale voltage from center screen in NR3 format. For external triggers (HP5462xA only), <level> ::= 2 volts with probe attenuation at 1:1 in NR3 format. For digital channels (HP5462xD only), <level> ::= 6 V. <source> ::= {CHANnel<n> I EXTernal I LINE} for HP5462xA <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 I LINE} for HP5462xD <n> ::= 1-2 or 1-4 in NR1 format
:TRIGger:GLITch:POLarity <polarity>	:TRIGger:GLITch:POLarity?	<polarity> ::= {POSitive I NEGative}
:TRIGger:GLITch:QUALifier <qualifier>	:TRIGger:GLITch:QUALifier?	<qualifier> ::= {GREaterthan I LESSthan I RANGe}
:TRIGger:GLITch:RANGe <greater than time> [suffix], <less than time> [suffix]	:TRIGger:GLITch:RANGe?	<greater than time> ::= start time from 10 ns to 9.99 seconds in NR3 format <less than time> ::= stop time from 15 ns to 10 seconds in NR3 format [suffix] ::= {s I ms I us I ns I ps}
:TRIGger:GLITch:SOURce <source>	:TRIGger:GLITch:SOURce?	<source> ::= {CHANnel<n> I EXTernal} for HP 5462xA; <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 } for HP 5462xD; <n> ::= 1-2 or 1-4 in NR1 format
:TRIGger:IIC:PATTern:ADDRess <value>	:TRIGger:IIC:PATTern:ADDRess?	<value> ::= integer or <string> <string> ::= ""0xnn"" n ::= {0 ,...., 9 I A ,...., F}
:TRIGger:IIC:PATTern:DATA <value>	:TRIGger:IIC:PATTern:DATA?	<value> ::= integer or <string> <string> ::= ""0xnn"" n ::= {0 ,...., 9 I A ,...., F}
:TRIGger:IIC:[SOURce:]CLOCk <source>	:TRIGger:IIC:[SOURce:]CLOCk?	<source> ::= {CHANnel<n> I EXTernal} for HP 5462xA; <source> ::= {CHANnel<n> I DIGital0,...,DIGital15 } for HP 5462xD; <n> ::= 1-2 or 1-4 in NR1 format

Command	Query	Options and Query Returns
:TRIGger:IIC:[SOURce:]DATA <source>	:TRIGger:IIC:[SOURce:]DATA?	<source> ::= {CHANnel<n>} for HP 5462xA; <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 } for HP 5462xD; <n> ::= 1-2 or 1-4 in NR1 format
:TRIGger:IIC:TRIGger <trigger condition>	:TRIGger:IIC:TRIGger?	<trigger condition> ::= {STARt | STOP | READ | NOTRead | WRITe | NOTWrite}
:TRIGger:SEQuence:COUNt <count>	:TRIGger:SEQuence:COUNt?	<count> ::= integer in NR1 format
:TRIGger:SEQuence:EDGE{1 | 2} <source>, <polarity>	:TRIGger:SEQuence:EDGE{1 | 2}?	<source> ::= {CHANnel<n> | EXTernal} for HP 5462xA; <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 } for HP 5462xD<n> ::= 1-2 or 1-4 in NR1 format <polarity> ::= {POSitive | NEGative} <return_value> ::= query returns "NONE" if edge source is disabled
:TRIGGER:SEQuence:FIND <resource1>[,<operator>[,<resource 2>]]	:TRIGGER:SEQuence:FIND?	<resource1> ::= {PATTern1 | EDGE1} <operator> ::= {ENTered | EXITed | AND | NONE} <resource2> ::= {PATTern1 | EDGE1 | NONE}
:TRIGger:SEQuence:PATTern {1 | 2}<value>, <mask>	:TRIGger:SEQuence:PATTern{1 | 2}?	<value> ::= integer or <string> <mask> ::= integer or <string> <string> ::= ""0xnnnnnn"" n ::= {0 ,...., 9 | A ,...., F}
:TRIGGER:SEQuence:RESet <resource1>[,<operator>[,<resource 2>]]	:TRIGGER:SEQuence:RESet?	<resource1> ::= {PATTern{1 | 2} | EDGE{1 | 2} | TIMer} <operator> ::= {ENTered | EXITed | AND | NONE} <resource2> ::= {PATTern{1 | 2} | EDGE{1 | 2} | TIMer | NONE}
:TRIGGER:SEQuence:TIMer <timeout>	:TRIGGER:SEQuence:TIMer?	<timeout> ::= time from 100 ns to 10 seconds in NR3 format
:TRIGGER:SEQuence:TRIGger <resource1>[,<operator> [,<resource2>]]	:TRIGGER:SEQuence:TRIGger?	<resource1> ::= {PATTern2 | EDGE2} <operator> ::= {ENTered | EXITed | AND | COUNt | NONE} <resource2> ::= {PATTern2 | EDGE2 | NONE}
:TRIGger:TV:LINE <line number>	:TRIGger:TV:LINE?	<line number> ::= integer in NR1 format.
:TRIGger:TV:MODE <tv mode>	:TRIGger:TV:MODE?	<tv mode> ::= {Field<n> | ALLFields | LINE | ALLLines | LINEField<n> | LINEAlt | LINEVert}; <n> ::= 1-2 in NR1 format
:TRIGger:TV:POLarity <polarity>	:TRIGger:TV:POLarity?	<polarity> ::= {POSitive | NEGative}
:TRIGger:TV:SOURce <source>	:TRIGger:TV:SOURce?	<source> ::= {CHANnel<n>} <n> ::= 1-2 or 1-4 integer in NR1 format
:TRIGger:TV:STANdard <standard>	:TRIGger:TV:STANdard?	<standard> ::= {GENeric | NTSc | PAL | PALM | SECam}
:TRIGger:TV:TVMODE <tv mode>	:TRIGger:TV:TVMODE?	<tv mode> ::= {Field<n> | ALLFields | LINE | ALLLines | LINEField<n> | LINEAlt | LINEVert}; <n> ::= 1-2 in NR1 format
n/a	*TST?	<result> ::= 0 or non-zero value; an integer in NR1 format
:VIEW <source>	n/a	<source> ::= {CHANnel<n> | PMEMory{0 | 1 | 2} | FUNCtion} for 5462xA <source> ::= {CHANnel<n> | DIGital0,...,DIGital15 | PMEMory{0 | 1 | 2} | FUNCtion} for 5462xD <n> ::= 1-2 or 1-4 in NR1 format
*WAI	n/a	n/a
:WAVeform:BYTeorder <value>	:WAVeform:BYTeorder?	<value> ::= {LSBFirst | MSBFirst}
na	:WAVeform:COUNt?	<count> ::= an integer from 1 to 16384 in NR1 format

Command	Query	Options and Query Returns
n/a	**:WAVeform:DATA?**	<binary block length bytes>, <binary data> For example, to transmit 2000 bytes of data, the syntax would be: #800002000<2000 bytes of data><NL> 8 is the number of digits that follow 00002000 is the number of bytes to be transmitted <2000 bytes of data> is the actual data
:WAVeform:FORMat <value>	:WAVeform:FORMat?	<value> ::= {WORD | BYTE | ASCII}
:WAVeform:POINts <# points>	:WAVeform:POINts? [MAXimum]	<# points> ::= {100 | 250 | 500 | 1000 | 2000 | MAXimum}
n/a	**:WAVeform:PREamble?**	<preamble_block> ::= <format NR1>, <type NR1>, <points NR1>,<count NR1>, <xincrement NR3>, <xorigin NR3>, <xreference NR1>, <yincrement NR3>, <yorigin NR3>, <yreference NR1> <format> ::= an integer in NR1 format: 0 for BYTE format 1 for WORD format 2 for ASCii format <type> ::= an integer in NR1 format: 2 for AVERage type 0 for NORMal type 1 for PEAK detect type <count> ::= Average count, or 1 if PEAK detect type or NORMal; an integer in NR1 format
:WAVeform:SOURce <source>	:WAVeform:SOURce?	<source> ::= {CHANnel<n> | POD1 | POD2 | FUNCtion}; <n> ::= 1-2 integer in NR1 format
n/a	**:WAVeform:TYPE?**	<return_mode> ::= {NORM | PEAK | AVER}
:WAVeform:UNSigned {{0 | OFF} | {1 | ON}}	:WAVeform:UNSigned?	{0 | 1}
:WAVeform:VIEW <view>	:WAVeform:VIEW?	<view> ::= {NORMal}
n/a	**:WAVeform:XINCrement?**	<return_value> ::= x-increment in the current preamble in NR3 format
n/a	**:WAVeform:XORigin?**	<return_value> ::= x-origin value in the current preamble in NR3 format
n/a	**:WAVeform:XREFerence?**	<return_value> ::= 0 (x-reference value in the current preamble in NR1 format)
n/a	**:WAVeform:YINCrement?**	<return_value> ::= y-increment value in the current preamble in NR3 format
n/a	**:WAVeform:YORigin?**	<return_value> ::= y-origin in the current preamble in NR3 format
n/a	**:WAVeform:YREFerence?**	<return_value>::= y-reference value in the current preamble in NR1 format

V.3 Valores físicos de una forma de onda.

Los datos que constituyen una forma de onda consisten en un array de bytes que representan la posición vertical del píxel en la pantalla. Su valor físico (tensión en voltios), depende del "setup" establecido en el osciloscopio ("yreference", "yincrement" e "yorigin").

El índice del array representa la posición horizontal y para determinar a que tiempo físico, respecto del punto de disparo corresponde, hay que tener en cuenta también los parámetros de "setup" establecidos en el osciloscopio ("xreference", "xincrement" e "xorigin").

.

Las fórmulas de conversión son:

voltaje = [(Dato - yreference) yincrement] + yorigin*

tiempo = [(Indice de muestra - xreference) xincrement] + xorigin*

V.4 Información sobre los Osciloscopios HP-54600 y HP-54621.

La información completa sobre el osciloscopio se dispone en el documento:

"Programmer's Guide: Oscilloscopes Agilent 54621A/22A/24A/41A/42A"

Y los comandos que admite el osciloscopio se pueden tener de forma interactiva con el documento Help:

"Agilent 5462x-Series Oscilloscopes Programmer's Reference"