Programmer's Guide

HP 71450B/1B/2B
Optical Spectrum
Analyzers

Remote operation with HP 71450B/1B/2B

This introduction provides you with a quick overview of some basic concepts you will need to successfully program the HP 71450B/1B/2B optical spectrum analyzer (OSA). The OSA offers extensive remote programming capability which includes the following features:

- Setting measurement ranges
- Querying values to the computer
- Creating new functions
- Creating user-defined variables and trace arrays
- Processing information with math and move commands
- Activating new functions with front-panel keys
- Allowing front-panel control during remote operation
- Placing graphics and text on the screen
- Monitoring instrument operation by returning a status byte to the computer

Start here

1. Finish reading this introduction and learn what areas of basic programming control are covered in this manual.

2. Read Chapter 1, "Learning the Basics" to connect the computer to the OSA.

   The OSA module should arrive installed in the display mainframe as described in the *HP 71450B/1B/2B Optical Spectrum Analyzer Installation and Verification Manual.*

3. Read the remaining chapters to learn specific details about remote programming.

4. Chapter 7, "Language Reference" documents each OSA command in alphabetical order.

5. Chapter 8, "Tables and Charts" contains additional information related to programming the OSA. Functional groups of programming commands with softkey cross-references are also provided.

Computer Model in Examples

The programming examples provided in this book used an HP 9000 Series 300 technical computer and the HP-BASIC language. However, you can use other computers and languages with OSA commands and query data over HP-IB.

As an example, you could use an HP Vectra PC compatible computer. Simply install an HP 82335A HP-IB Interface and Command Library card in the PC and use a compatible language such as Microsoft QuickC.

Local and Remote Control Indicators

Whenever the analyzer is addressed via HP-IB, the (MENU) and (USER) keys are disabled. The front-panel LEDs indicate the following:

- RMT indicates remote HP-IB control.

- LSN indicates information received via HP-IB.

- TLK indicates information sent via HP-IB to computer.

- SRQ indicates the OSA is requesting computer attention.

Designing Programs

OSA programs are composed of commands that correspond to front-panel keys. Except for timing considerations, remote operation is very similar to manual operation. To design a program, use the following procedure:

1. Note the order of keystrokes used to perform an OSA measurement.

2. Identify which commands correspond with the keystrokes used. Refer to the functional group tables of commands in Chapter 8, "Tables and Charts."

3. Incorporate the commands into output statements.

Default HP-IB Addresses

Typically, you will program the OSA using the factory default HP-IB addresses provided in the following table. If you want to change these addresses, refer to Chapter 1, "Learning the Basics."

Default HP-IB Addresses

MMS unit	Address
HP 70950B/1B/2B OSA module	23
HP 70004A Display	4

Safety Notes

The following safety notes are used throughout this manual. Familiarize yourself with each of the notes and its meaning before operating this instrument.

CAUTION   The *caution* note denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in damage to or destruction of the instrument. Do not proceed beyond a *caution* note until the indicated conditions are fully understood and met.

WARNING   The *warning* note denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in injury or loss of life. Do not proceed beyond a *warning* note until the indicated conditions are fully understood and met.

Instruction Manual

⚠

The **instruction manual** symbol. The product is marked with this symbol when it is necessary for the user to refer to the instructions in the manual.

General Safety Considerations

Certification

Hewlett-Packard Company certifies that this product met its published specifications at the time of shipment from the factory. Hewlett-Packard further certifies that its calibration measurements are traceable to the United States National Institute of Standards and Technology, to the extent allowed by the Institute's calibration facility, and to the calibration facilities of other International Standards Organization members.

Regulatory Information The specifications and characteristics chapter contains regulatory information.

Warranty

This Hewlett-Packard instrument product is warranted against defects in material and workmanship for a period of one year from date of shipment. During the warranty period, Hewlett-Packard Company will, at its option, either repair or replace products which prove to be defective.

For warranty service or repair, this product must be returned to a service facility designated by Hewlett-Packard. Buyer shall prepay shipping charges to Hewlett-Packard and Hewlett-Packard shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to Hewlett-Packard from another country.

Hewlett-Packard warrants that its software and firmware designated by Hewlett-Packard for use with an instrument will execute its programming instructions when properly installed on that instrument. Hewlett-Packard does not warrant that the operation of the instrument, or software, or firmware will be uninterrupted or error-free.

Limitation of Warranty

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by Buyer, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED. HEWLETT-PACKARD SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Exclusive Remedies

THE REMEDIES PROVIDED HEREIN ARE BUYER'S SOLE AND EXCLUSIVE REMEDIES. HEWLETT-PACKARD SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

Assistance

Product maintenance agreements and other customer assistance agreements are available for Hewlett-Packard products.

For any assistance, contact your nearest Hewlett-Packard Sales and Service Office.

Typesetting Conventions

(Front-Panel Key)	This represents a key physically located on the instrument.
`Softkey`	This indicates a "softkey," a key whose label is determined by the instrument's firmware.
`Screen Text`	This indicates text displayed on the instrument's screen.

Contents

Contents-8

8. Tables and Charts

Index

Contents

1

Learning the Basics

Learning the Basics

This chapter contains basic, remote program information, for the operation of an optical spectrum analyzer. In addition to learning how to send commands and receive data, you can learn about a debugging feature for locating programming errors.

Contents

Preparing the OSA

When preparing the OSA for programming, you must connect the HP-IB cables, confirm the HP-IB address, and reset all instruments connected to the bus.

Typically, you will use the default address setting provided in the introduction of this book. However, the OSA's HP-IB address can be changed using front-panel keys or manually by changing SWITCHES\MEMORY on the HP 70950B/1B/2B module. The switches are located on the module cover.

In the illustration below, the rectangles are half-black (off position) and half-white (on position).

ON

HP-IB

OFF

SWI | MEM

MASTER | SLAVE

NORMAL | TEST

HP-IB

COLUMN 1	(1)
COLUMN 2	(2)
COLUMN 3	(4)
COLUMN 4	(8)
COLUMN 5	(16)
ROW 1	(1)
ROW 2	(2)
ROW 3	(4)

1   0
(ON)(OFF)

HP 70950A/70951A
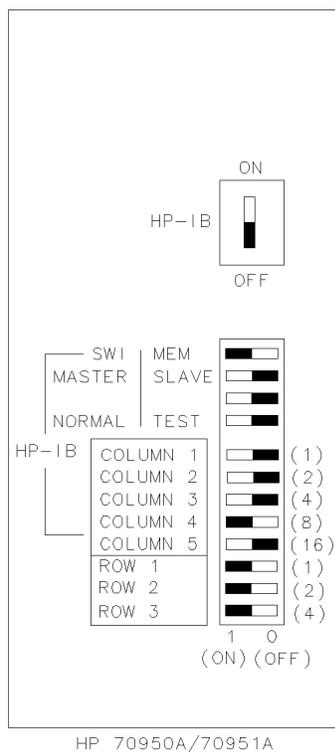
SWITCH SETTINGS SHOWN:
COLUMN ADDRESS 23 = 1 + 2 + 4 + 0 + 16
ROW ADDRESS 0 = 0 + 0 + 0
HP-IB=ON   ckb1

OSA Address Switches

In this figure, switches ROW 1, ROW 2, and ROW 3 correspond to row-address numbers which position the module vertically in the address map. The row switches for the optical spectrum module address must be set to 0 to allow HP-IB control.

The COLUMN switches select the HP-IB address and correspond to the column-address numbers on the address map. Next to each switch are numbers in parentheses which indicate the binary value of the individual switches. The column or row address is the sum of the values in the parentheses beside each switch that is set to 1. For example, a column

address of 23 is selected by setting column switches 1, 2, 3, and 5 to 1, and setting column switch 4 to 0. (23 = 1 + 2 + 4 + 0 + 16.)

Illegal Addresses

HP-IB address 31 and HP-MSIB address 0, 31 are illegal and cannot be used.

Set the following OSA module switches to the settings shown in the following list:

HP-IB .... ON
SWITCHES/MEMORY .... MEMORY
MASTER/SLAVE .... MASTER
NORMAL/TEST .... NORMAL

On the display mainframe or module, three switches should be set as follows:

SYSTEM CONTROLLER .... 0 (OFF)
TALK ONLY .... 0 (OFF)
TEST MODE .... 0 (OFF)

For a complete discussion of switch settings, refer to the *HP 71450B/1B/2B Optical Spectrum Analyzer Installation and Verification Manual.*

To prepare the OSA

1. Connect an HP-IB cable between the computer's HP-IB connector and the HP-IB connector on the HP 70000 modular measurement system's rear panel.

   HP 70950B/1B/2B OSA modules do not have an HP-IB connector. Use the connector that is located on the mainframe in which the module is installed. Be careful not to confuse the display or mainframe's HP-IB connector with any connectors that may be present on the rear panel of any modules in the HP 70000 modular measurement system.

2. On the OSA, press (DISPLAY), and then press `Address Map`.

3. Turn the front-panel knob until the OSA module appears in the address map.

   The HP-IB address is identical to the address map column where the module is listed. The OSA must be shown in row 0 of the map.

4. Ensure that the **SYSTEM CONTROLLER** switch on the HP 70004A display (or HP 70205 or HP 70206 display modules) is set to 0 (OFF).

5. Reset all instruments connected to the bus by turning the line power off and then on again.

   Most peripherals like plotters or printers are reset by pressing a reset key. If you are not sure how to reset a device, cycle its power off, then on, to reset it.

To change the HP-IB address using switches

The HP-IB address is forced to the switch address setting when the OSA module SWITCHES/MEMORY selector is set to SWITCHES.

N O T E

The SWITCHES setting must be selected. The MEMORY setting is the factory default setting.

---

1. Turn off any line-power, and remove the HP 70950B/1B/2B module from the mainframe.

2. Locate the switches on the top of the module.

3. Set the SWITCHES/MEMORY selector to the SWITCHES position.

4. Be sure the three switches labeled ROW 1, ROW 2, and ROW 3 are set to 0 (OFF).

5. Set the five COLUMN switches so that their added values (as indicated by the numbers in parenthesis) add up to the HP-IB value that you want.

   For example, to set the switches for address 23, set switches COLUMN 1, 2, 3, and COLUMN 5 to 1 (ON).

To change the HP-IB address from the front panel

The HP-IB address can be changed at the front panel when the OSA module's SWITCHES/MEMORY switch is set to MEM. (This is the default factory setting.)

1. Press $\boxed{\text{DISPLAY}}$, `Address Map`.

2. With the front-panel knob, position the box around the OSA module, then press `HP-IB ADDRSET`.

3. Enter the HP-IB address using the front-panel numeric keypad, then press `ENTER`.

The current HP-IB address is shown inside the OSA module box on the address map. Changing the HP-IB address from the front-panel does not affect the HP-MSIB address of the module.

Beginning Your Program

Programs should begin with system-level and OSA commands that initialize the system. Three such commands are discussed here: CLEAR, IP, and CLRDSP.

The CLEAR command is an HP BASIC command that resets a device on HP-IB to a known state. As an example, CLEAR 723 clears the device (the OSA in this case) at address 23 on interface 7. CLEAR erases the OSA's input buffer, interrupts most incomplete operations, and prepares the OSA to receive instructions.

The instrument-preset command, IP, sets all of the analog parameters of the OSA to a known state and provides a good starting point for all measurement processes. Refer to IP in Chapter 7, "Language Reference" for complete information on the IP command.

The CLRDSP command clears any graphics from the display screen. Because CLRDSP also erases stored graphics from memory, such as an item or a graphics window, execute CLRDSP discriminately. Chapter 5, "Creating Graphics," gives more information about initializing the analyzer display for graphic presentations.

To initialize the bus and the OSA

- Place the following commands at the start of your program.

```
10   CLEAR 723              Initialize the bus.
20   OUTPUT 723;"IP;"       Initialize the OSA.
30   OUTPUT 723;"CLRDSP;"   Clears any user graphics.
```

Sending Commands

The commands listed in Chapter 7 are sent to the OSA as ASCII strings. The method used depends on the programming language and environment. The examples in this book use the HP-BASIC language with an HP 9000 Series 300 technical computer. Using HP-BASIC, the following example shows an output statement to send a command to the OSA:

    10 OUTPUT 723;"CENTERWL 1300NM;"   *Set the center wavelength.*

Be sure to place a semicolon between commands and at the end of any command string.

Using an HP Vectra computer with the HP-IB Interface and Command Library (and programming in *C*), the same command could be sent as follows:

    IOOUTPUTS(723L,"CENTERWL 1300NM;",15);

Most analyzer commands first activate a function, then enter a number followed by a units code and a command terminator:

```
                                          OUTPUT723;"CENTERWL  800NM;"


        Activate  center  wavelength ────────────────────┘

        Space  recommended  by  syntax ────────────────────┘

        Number  (data  entry) ────────────────────┘

        Units  code ────────────────────┘

        Command  terminator ────────────────────┘

                                                  XXXX
                                                  ckb2
```
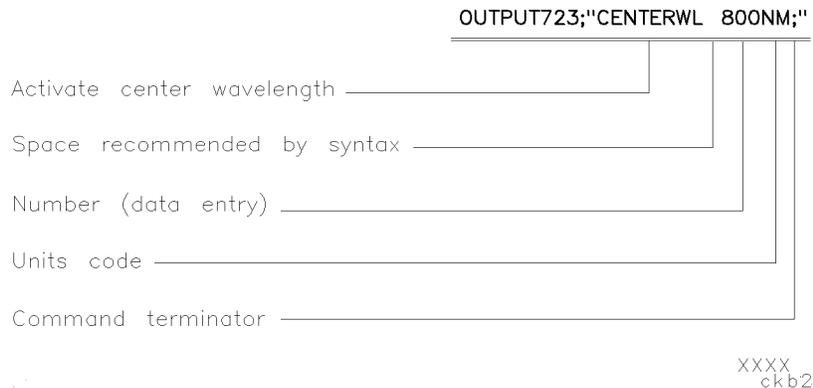
The number within the quote field must be a string of decimal numbers plus an optional decimal point and may be preceded by a minus or plus sign. Either fixed- or floating-point notation may be used. For example, "12.3E3", "1.23E4", and "12300" each enter the same number. A units code follows a data entry.

Units Codes

	Units	Code
Wavelength	Meters	M
	Millimeters	MM
	Micrometer	UM
	Nanometer	NM
	Angstrom	ANG
	Picometer	PM
Time	Second	S
	Millisecond	MS
	Microsecond	US
Current	Ampere	A
	Milliampere	MA
	Microampere	UA
Ratio Measurements	Decibel	DB
Power	Watt	W
	Milliwatt	MW
	Microwatt	UW
	Nanowatt	NW
	Picowatt	PW
Decibels relative to units.	1 Milliwatt (dBm)	DBM

To send commands to the OSA

1. Locate the appropriate commands in Chapter 7.

2. Construct a ASCII string of the commands, placing a semicolon between each command in the string.

3. Use your programming language's output statement to send the command string to the address of the OSA.

Returning Data to the Computer

Data is requested from the OSA using a query. Queries usually take the form of a command with the question-mark character. For example, `CENTERWL?` requests the OSA's center wavelength setting. The syntax diagrams in Chapter 7 document available command queries. Most command queries and variable queries are normally sent in the form of ASCII strings. However, binary query results are returned for some commands. It is your responsibility to allocate memory for returned values in your application.

The OSA returns information as decimal values in the units shown in the following table. Units are determined by the settings of the reference level (RL) for some functions, such as the marker, the scale (LN, LG), or the amplitude units (MKREAD, AUNITS).

Parameter Units

Measured Data	Fundamental Unit
Current	Ampere
Frequency	Hertz
Power	dBm
Ratio	dB
Time	Second
Wavelength	Meters

Trace information that has been manipulated with math and move commands, or has been formatted with the MDS (measurement data size) or TDF (trace data format) command may be returned in other units. Refer to Chapter 2.

Trace information is returned two different ways, either indirectly with the use of markers, or directly as trace data with the TRA, TRB, or TRC commands. The next example marks trace A with a marker and returns the wavelength and amplitude values at the marker. Note that for every query executed, a value must be stored in the computer.

```
10   CLEAR 723                                          Initialize analyzer.
20   OUTPUT 723;"IP;SNGLS;TS;"
30   OUTPUT 723;"CENTERWL 1300NM;SPANWL 10NM;RL -5DBM;"  Set measurement range.
40   OUTPUT 723;"TS;"                                   Sweep trace A.
50   OUTPUT 723;"MKPK HI;"                              Mark highest signal level.
60   OUTPUT 723;"MKWL?;"                                Return wavelength at marker.
70   ENTER 723;W                                        Assign wavelength to computer variable.
80   OUTPUT 723;"MKA?;"                                 Return amplitude at marker.
90   ENTER 723;A                                        Assign amplitude to computer variable.
100  W=W/1.E-9                                          Convert wavelength from meters to nanometers.
110  PRINT W,A                                          Print values.
```

To return the amplitude of the 400th element of trace A, you would send the following:

```
40   OUTPUT 723;"SNGLS;TS;;"
50   OUTPUT 723;"TRA[400]?;"
60   ENTER 723;N                                        Assign amplitude to computer variable.
70   PRINT N                                            Print value.
```

To return the amplitude of all elements in trace A send the following commands:

```
30   DIM Array (1:800)                                  Declare computer array for storage of 800 values.
40   OUTPUT 723;"SNGLS;TS;;"
60   OUTPUT 723;"TRA?;                                  Return amplitude of points in trace A.
70   ENTER 723;Array(*)                                 Store trace amplitude values in a computer array.
```

The FETCH command returns the trace data for the currently active trace. Data is returned in binary while the sweep progresses. Use FETCH to speed data transfer and processing. Refer to Chapter 2 for more information on returning trace data.

To query a value

1. Allocate a variable to hold the data returned by the query.

2. Use an output statement to send the query to the OSA.

3. Enter the returned data to the variable.

For example, to return the sensitivity setting, send `SENS?`.

```
OUTPUT 723;"SENS?";"     Send the sensitivity query string.
ENTER 723;S              Assign value to computer variable.
```

Using Commands as Arguments

Some OSA commands accept other commands as arguments. The following example shows the MKA command used as an argument for the MOV command.

```
10 OUTPUT 723;"MOV DL,MKA;"
```

The value of the command MKA is returned as input to the calling command, MOV. Commands that accept command arguments have syntax diagrams that show *predefined function* as an argument. Refer to Chapter 7. When using a command as an argument, the value returned is similar to the query value. Be sure the value returned is appropriate for the calling command. For example, the `TRA` command returns all the data in a trace, but `TRA[50]` returns the fiftieth element.

Do not follow the argument command with a `?` or termination character. In the example shown above, the MOV moves the marker amplitude value into the predefined variable, DL. Notice that MKA is not terminated with a `?` or termination character. The ending semicolon terminates the MOV command.

To use a command argument

- The calling command must list *predefined function* in its syntax diagram for the argument.

- Do not follow the argument command with a `?` or termination character.

Using the MOV Command

It is recommended that the MOV command be used to execute all
programming commands that load variables. Using the MOV command is
faster and avoids displaying text on the analyzer screen. For example, send
the following to set the reference level to −10 dBm:

```
OUTPUT 723;"MOV RL,−10DBM;"
```

Controlling Program Timing

Most remote controlled measurements require control of the sweep. The TS (take sweep) command initiates a sweep when the trigger conditions are met. When TS is executed as part of a command sequence, the analyzer starts and completes one full sweep before the next analyzer command is executed. Use the SNGLS (single sweep) command to maintain absolute control over the sweep and to reduce execution time. Once SNGLS activates the single-sweep mode, TS can be used to initiate a sweep only when necessary.

Since many remote commands process trace information, updating trace information with TS becomes important whenever the input signal or measurement range changes. Use TS to update the trace *after* the analyzer settings or input signals change, but *before* the trace information is returned to the computer or processed by other commands, like trace math or marker commands.

When developing measurement algorithms with the front-panel controls, use the `SINGLE SWEEP` key to simulate the effect of the TS command updating the trace. The following examples demonstrates changing the measurement range. The trace data is not updated with the settings in line 30 until the TS command in line 40 is executed.

```
20   OUTPUT 723;"IP;SNGLS;TS;"              Activate single sweep.
30   OUTPUT 723;"CENTERWL                   Change measurement range.
              1300NM;SPANWL 10NM;"
40   OUTPUT 723;"TS;"                       Measure with trace A.
```

Note that SNGLS cannot be substituted successfully for TS in all analyzer programs. In the following program, TS is executed *after* analyzer settings are changed, but *before* trace information is processed. If SNGLS were substituted for TS, SNGLS would initiate a sweep, but `MKPK HI` would be executed before the completion of the sweep.

```
30   OUTPUT 723;"CENTERWL                   Change measurement range.
              1300NM;SPANWL 10NM;"
40   OUTPUT 723;"TS;"                       Update trace.
50   OUTPUT 723;"MKPK HI;"                  Place marker on highest signal
                                            level.
```

There are two commands that change the measurement range indirectly, MKCWL and MKRL. They set the center wavelength and reference level equal to the marker wavelength and amplitude, respectively. If valid

trace information is needed, a TS command must be executed after these commands are executed. In all cases, executing TS invokes at least one entire sweep. However, TS invokes more than one sweep when certain commands are active, such as video averaging (VAVG) or marker tracking (MKTRACK).

Program timing can also be controlled with the RQS, DONE, and WAIT commands. The RQS command interrupts the computer operation when the analyzer has attained a predefined state. The DONE command allows the program to hold off the execution of subsequent commands until the OSA returns a "1" in response to the query. The WAIT command suspends program operation for a specified time. TS, RQS, and DONE are used often to synchronize data acquisition or the analyzer state with other events. The WAIT command, on the other hand, is used to improve human interface with the program. The program examples in this manual use WAIT to control the duration of displayed messages, or use WAIT to delay program execution to give you time to observe events on the display.

Entering Data from the Front Panel

Data may be entered from the front panel when the OSA is under remote control by using the EP (enable) argument with a command. EP halts execution of analyzer commands so that values may be entered with the front-panel numeric keypad, step keys, knob, and unit-code keys. Data may be entered from the front-panel until the `ACCEPT VALUE` key is pressed. The syntax diagrams in Chapter 7 provide the commands that can be followed by EP.

To allow front-panel data entry

- Use the EP argument with a compatible command.

40	OUTPUT 723;"LG 20DB;"	*Set the log scale to 20 dB/division.*
50	OUTPUT 723;"LG EP;"	*Change the log scale with the front-panel keys. Enter a number, then press* `ACCEPT VALUE`

Controlling Advanced Measurement Programs

The `FP`, `DFB`, and `LED` measurement keys available under the (USER) menu activate advanced measurement routines. You can also make these measurements over HP-IB. Refer to Chapter 7, "Language Reference" for more information about the following advanced measurement program commands:

- "FP_" for characterizing a Fabry-Perot laser source.

- "DFB_" for characterizing a distributed feedback laser source.

- "LED_" for characterizing a light-emitting diode laser source.

To learn how to measure the characteristics of these laser sources, manually, refer to Chapter 3, "Measuring Signals" in the OSA users' guide.

Making advanced measurements over HP-IB

To make advanced measurements over HP-IB, connect HP-IB as needed. Connect the laser source to the OSA.

Follow the instructions below, or refer to the commands in Chapter 7, "Language Reference", of this manual:

1. To measure a Fabry-Perot source and obtain measurement data, enter the following command lines:

   ```
   30   OUTPUT 723;"AUTOMEAS;"     Locate and display the output of the laser source.
   40   OUTPUT 723;"FP_;"          Measure the output of the Fabry-Perot source.
   50   OUTPUT 723;"FP_ ?;"        Query the instrument and return the FP charac-
                                   terization data.
   ```

2. To measure a distributed feedback laser source and obtain measurement data, enter the following command lines:

   ```
   30   OUTPUT 723;"AUTOMEAS;"     Locate and display the output of the laser source.
   40   OUTPUT 723;"DFB_;"         Measure the output of the laser source.
   50   OUTPUT 723;"DFB_ ?;"       Return the measurement data.
   ```

3. To measure an LED laser source and obtain measurement data, enter the following command lines:

    ```
    30   OUTPUT 723;"AUTOMEAS;"        Locate and display the output of the laser source.
    40   OUTPUT 723;"LED_;"            Measure the output of the LED source.
    50   OUTPUT 723;"LED_ ?;"          Return the measurement data.
    ```

4. To set the right and left power integration points on a signal, enter the following command lines:

    ```
    10   OUTPUT 723;"CENTERWL 900NM;"  Set the center wavelength to 900 nm.
    20   OUTPUT 723;"TS;"              Update the current trace.
    30   OUTPUT 723;"AMPMKR −10,dB;"   Set the right and left power integration
                                       points −10 dB from the peak.
    ```

2

Using Variables and Traces

Using Variables and Traces

This chapter can help you learn to create variables and process both predefined and user-defined trace arrays.

- Predefined traces, for example trace A, exist permanently in reserved memory.
- User-defined variables and trace arrays are created by the user, given preset values, and stored in internal memory. User-defined variables and trace arrays remain in internal memory, even when power is turned off.

Reserved Words

When creating user-defined variables and traces, do not use the permanently stored OSA reserved words (as listed in Chapter 8, "Tables and Charts") for names. Reserved word usage interrupts the defining process and can disrupt OSA operation.

In addition, any names created with TRDEF, FUNCDEF, ACTDEF or other VARDEF commands become reserved words until they are erased from internal memory.

Defining Variables

The OSA retains in reserved memory a set of predefined variables. Each predefined variable stores a current OSA setting or state. Note the two characteristics of these variables:

1. Values depend on current instrument settings.

2. Names are identical with associated programming commands.

Programming commands which have the same name as a predefined variable load that variable. For example, the command CENTERWL sets the OSA's center wavelength. CENTERWL also loads the predefined variable CENTERWL. Predefined variables reside in reserved memory and are always available to your programs. Both predefined and user-defined variables can be queried to return their values to the computer. Also, variables can be used as arguments to many of the OSA commands listed in chapter 8.

Partial List of Predefined Variables

CENTERWL	CNVLOSS	DL	INZ	LG
MKA	MKACT	MKAL	MKAR	MKN
MKP	MKPAUSE	MKPX	MKT	MKTV
NSTATE	RB	RBR	RL	RLPOS
ROFFSET	SP	SS	ST	STARTWL
STOPWL	TH	VB	VTL WLOFFSET	

Use the **VARDEF** command to create and store user-defined variables in internal memory. User-defined variable names are from 1 to 12 characters long and are assigned values immediately after **VARDEF** is executed. Use **PURGE** or **DISPOSE** to erase user-defined variables from internal memory.

To create a user-defined variable

- Use the `VARDEF` command.

    `10 OUTPUT 723;"VARDEF MY_VAR,0;"`   *Create variable called MY_VAR and assign it a value of 0.*

To erase a user-defined variable

- Use the `DISPOSE` command.

    `10 OUTPUT 723;"DISPOSE MY_VAR;"`

Defining Traces

The OSA stores data for traces A, B, and C in reserve-memory trace arrays. These arrays comprise a series of data points (elements) that contain amplitude information. Position units describe the position of an element (point) along the horizontal axis of a trace. The point at the left end of the trace has a position-unit value of 1. The point at the right end of an 800-point trace has a position-unit value of 800. Whenever trace A, B, or C is swept, the analyzer updates the trace's array with new data. You can create your own user-defined trace arrays in **internal memory** using the `TRDEF` command. Unlike traces A, B, and C, user-defined traces cannot be swept.

The `POSU` command converts the wavelength of a point on a trace to position units. The `WLU` command converts a point on a trace indicated in position units to meters, or in zero-span mode, to units of time.

Predefined and user-defined traces can be 3 to 2048 elements. The default trace length is 800 elements. Use the `TRDEF` command to change the length of any trace. Use the `IP` or `TRPST` (trace preset) command to reset the lengths to their preset values. Use `PURGE` or `DISPOSE` to erase a user-defined trace from internal memory.

User-defined trace names can be from 1 to 12 characters long. In the following example, measurement results are transferred to a user-defined trace array called TRACE_Z.

10	`CLEAR 723`	*Initialize analyzer.*
20	`OUTPUT 723;"IP;"`	
30	`OUTPUT 723;"SNGLS;TS;"`	*Activate single-sweep mode and sweep once.*
40	`OUTPUT 723;"TRDEF TRACE_Z,800;"`	*Create user-defined trace array.*
50	`OUTPUT 723;"MOV TRACE_Z,TRA;"`	*Move copy of trace A contents into TRACE_Z.*

To create a user-defined trace

- Use the `TRDEF` command.

  ```
  10 OUTPUT 723;"TRDEF TRACE_Z,1000;"
  ```

To erase a user-defined trace

- Use the `DISPOSE` command.

  ```
  10  OUTPUT 723;"DISPOSE TRACE_Z;"
  ```

Using Amplitude Data

Measurement results are expressed in either parameter units or measurement units. A parameter unit is a standard scientific unit such as Watts, decibels, seconds, hertz, amperes, and ohms. Measurement units are an internal representation of the data. Measurement units apply only to traces and trace elements. In its preset state, the analyzer returns all measurement results in parameter units.

The amplitude parameter units used are shown in the following table. Use the AUNITS command to change the active amplitude units shown in display annotation and returned by queries to the computer.

Amplitude Parameter Units

	Units
Decibels	dB, dBm
Watts	W, mW, μW

For linear trace information, measurement units are defined as 0 (zero) for the bottom of the display, and 10,000 for the top (reference level). For logarithmic trace information, a measurement unit is one hundredth of a dBm:

$$measurement\ unit\ =\ (value\ in\ dBm)(100)$$

Thus, a -10 dBm value is equal to -1000 measurement units:

-1000 measurement units $= -10$ dBm (100)

The measurement-unit range is restricted to integers between $+32{,}767$ and $-32{,}768$. Since measurement units are restricted to integers, -10.115 dBm equals -1012 measurement units, *not* -1011.5 measurement units because decimal values are rounded to the nearest integer.

-1012 measurement units $= -10.115$ dBm (100)

Use the AMPU command to convert measurement-unit values to parameter-unit values. The MEASU command converts parameter-unit values to measurement-unit values.

To select the amplitude units

- Use the `AUNITS` command.

To convert measurement and parameter units

- Use the `AMPU` command to convert measurement units to parameter units.
- Use the `MEASU` command to convert parameter units to measurement units.

To return wavelength and position values

- Use the `POSU` command to determines the trace point that corresponds to a wavelength value.
- Use the `WLU` command determines the wavelength value of a given trace point.

Formatting Trace-Data Output

The trace-data format (TDF) command controls the format of trace-amplitude data output from the OSA. All the available format modes return amplitude information in measurement units except for TDF P, which returns parameter units. The output formats are summarized in the following table. (Parenthetical numbers in the table are decimal values representing binary, 8-bit numbers.)

Trace Data Output Formats for Amplitude Information

Format Command	OSA Output	Trace Element Output Amplitude of +10 dBm
TDF P	Returns ASCII decimal value. Values are in parameter units. Data output is followed by line feed (ASCII code 10) and end-or-identify (EOI)	+10.00
TDF M	Returns ASCII integer value. Values are in measurement units. Line feed (ASCII code 10) and EOI follow data output.	+1000
TDF B	Returns binary value as one or two 8-bit bytes (determined by the MDS command), with most significant byte first. Values are in measurement units. Sends EOI with last byte.	(3) (232)
TDF A	Sends series of one or two 8-bit bytes (A-block data field and the number of bytes is determined by the MDS command). Data is preceded by "#" and "A" and two bytes representing length of data field. Values are in measurement units.	(#)(A) (0)(2) (3)(232)
TDF I	Sends a series of one or two 8-bit bytes (I-block data field and the number of bytes is determined by the MDS command). Data is preceded by "#" and "I". Values are in measurement units. Sends EOI with last byte.	(#)(I) (3) (232)

TDF P formats data as ASCII, decimal values in parameter units. Instrument preset (IP) activates the TDF P format, and the AUNITS (amplitude units) command specifies the kind of parameter units. The default units are volts for linear trace information and dBm for logarithmic trace information. A line feed (ASCII code 10) follows data output. The end-or-identify state (EOI) is sent with line feed. The TDF P format is convenient because it returns parameter-unit values, but it is the slowest of the data-format modes.

TDF M formats data as ASCII, integer values in measurement units. (Thus, the AUNITS command has no affect on returned data.) A line feed (ASCII code 10) follows data output. The end-or-identify state (EOI) is sent with line feed. The **TDF M** format is faster than the **TDF P** format, but requires conversion to parameter units.

TDF B formats data in measurement units as binary numbers. Numbers are transmitted usually as words. (A word is two 8-bit bytes.) The results are transmitted in the following order:

- The most significant byte is sent first.

- The least significant byte is sent next.

- EOI is sent with the last byte.

For speed at the cost of accuracy, the measurement-data-size command (MDS) may be used to consolidate the information contained in a word into one byte (MDS B). However, since trace math operations often produce numbers too large to be contained in one byte, the use of bytes instead of words as the data size is not recommended, especially during trace-math operations. Instrument preset (IP) selects words as the default data size.

TDF A and **TDF I** format data into absolute and indefinite block-data fields. Both formats return the same kind of data, binary words (**MDS W**) or bytes (**MDS B**) as a block of data. However, the formats differ in the way they indicate the number of bytes or words returned. The **TDF A** format precedes the data with # A and two bytes that specify the number of data bytes returned. On the other hand, the **TDF I** format precedes the data with "# I" and sends EOI (end or identify) with the last byte of data. The block-data formats are available for compliance with IEEE standards. They have the same limitations as the binary format, **TDF B**, when bytes are chosen as the measurement data size.

The following table shows an HP-IB transmission sequence for each format. Each one transmits the +10 dBm amplitude level of a trace element. Remember that even though the analyzer transmits binary or ASCII-decimal information on HP-IB, a decimal number is always returned to the HP 9000 series 200 or 300 controller. The parenthetical numbers in the table are decimal values representing binary, 8-bit numbers.

Transmission Sequence for Trace-Data Formats

	TDF P	TDF M	TDF B	TDF A	TDF I
Byte 1	1	1	|3|	#	#
Byte 2	0	0	|232-EOI|	A	I
Byte 3	.	0		|0|	|3|
Byte 4	0	0		|2|	|232-EOI|
Byte 5	0	10 |EOI|		|3|	
Byte 6	10|EOI|			|232|	
				|10-EOI|	

The format of the computer must be compatible with the output format of the analyzer. The following table and programming examples illustrate this concept.

Computer Formats

Analyzer Format	Computer-Format Requirements	Example in HP BASIC
TDF M	Free field	10 ENTER 723;Amplitude
TDF P	Free field	10 ENTER 723;Amplitude
TDF B	Binary. Read twice for each value if data size is words. |See MDS.|	10 ASSIGN @Sa TO 723;FORMAT OFF 20 ENTER @Sa USING "#,W";Z|*|
TDF A	Initially read four values |for #, A, and length| in any format. Use integer format for data that follows. |Refer to MDS|	10 ASSIGN @Sa TO 723;FORMAT OFF 20 ENTER @Sa;Pounda,Num_bytes 30 ALLOCATE INTEGER Z|1:Num_bytes/2| 40 ENTER @Sa;Z|*|
TDF I	Initially read two values |for #, I| in any format. Use integer format for data that follows. |Refer to TDF|	10 ASSIGN @Sa TO 723;FORMAT OFF 20 ENTER @Sa;Poundi 30 ENTER @Sa USING "#,W";Z|*|

The following examples illustrate the four ways trace data can be formatted for return to the computer. The last example also shows how to send the trace data from the computer back to the analyzer. To return trace data as decimal numbers in parameter units (TDF P):

```
10 OUTPUT 723;"TDF P;"       Return amplitude of trace element in parame-
20 OUTPUT 723;"TRA[10]?;"    ter units.
30 ENTER 723;N               Assign value of (−10.33) to N.
```

To return trace data as integers in measurement units (TDF M):

```
10 OUTPUT 723;"TDF M;"       Activate measurement-unit format.
20 OUTPUT 723;"TRA[10]?;"    Return amplitude of trace element in measure-
                             ment units.
30 ENTER 723;N               Assign value (−1033) to N.
```

To return trace data as binary numbers with data size of one word (TDF B).

```
10 INTEGER Z(1:800)          Declare integer array.
```

```
20 ASSIGN @Sa TO 723
30 ASSIGN @Sa_fast TO 723;FORMAT OFF    Assign fast I/O path to 723.
40 OUTPUT @Sa;"MDS W;"                  Define data size as one word.
50 OUTPUT @Sa;"TDF B;"                  Activate binary data format.
60 OUTPUT @Sa;"TRA?;"                   Return amplitude of all trace elements
                                        in measurement units.
70 ENTER @Sa_fast USING "#,W";Z(*)      Load binary values into array Z. "#"
                                        specifies that line execution is termi-
                                        nated when last ENTER item is termi-
                                        nated. "W" specifies that items loaded
                                        are words (two bytes).
```

To return trace data in block-data format with data size of one word (TDF A), then send data back to analyzer.

```
1   OUTPUT 723;"SNGLS;SP 1MHZ;CF 300MHZ;"    Activate single-sweep mode.  Set
                                             measurement range.
5   OUTPUT 723;"TS;"                         Sweep trace A.
10  INTEGER Pounda,Num_bytes                 Declare integer array.
20  ASSIGN @Sa TO 723
30  ASSIGN @Sa_fast TO 723;FORMAT OFF        Assign fast I/O path to 723.
40  OUTPUT @Sa;"MDS W;"                      Define data size as one word.
50  OUTPUT @Sa;"TDF A;"                      Activate absolute block-data format.
60  OUTPUT @Sa;"TRA?;"                       Return amplitude of all trace elements.
70  ENTER @Sa_fast;Pounda,Num_bytes          Assign #A to Pounda. Assign length,
                                             that is, the number of bytes re-
                                             turned to computer, to Num_bytes.
                                             (Temporary variables.)
80  ALLOCATE INTEGER A(1:Num_bytes/2)        Declare array with length of Num_bytes/2.
90  ENTER @Sa_fast; A(*)                     Store contents of trace A in an
                                             integer array.
100 OUTPUT @Sa;"TRDSP TRA,OFF;"              Stop display of trace A.
110 OUTPUT @Sa;"TRC";                        This line and the next line store
                                             the contents of the integer array in
                                             trace C. The semicolon is required
                                             to suppress carriage-return/line-feed
                                             so that the TRC-command statement
                                             is not terminated prematurely.
120 OUTPUT @Sa_fast;Pounda;Num_bytes;A(*)    Send #A and length to OSA, fol-
                                             lowed by contents of integer array.
                                             As required by the TRC-command
                                             syntax, the @Sa_fast format removes
                                             commas that separate the numbers
                                             inside the integer array.
130 OUTPUT 723;"VIEW TRC;"                   View trace C.
```

Using Math and Move Commands

This section covers math and move commands which modify data and store the results in memory. In addition, four rules are provided for using math with trace arrays and elements.

Each math and move command requires both source and destination arguments. The value of the source is used to obtain the data stored in the destination. Only variables or traces can be a destination. Sources can be variables, traces, real numbers, or any command listed in the following table. (The commands in this table can only be used with commands whose syntax diagram shows `predefined function` as a source. Syntax diagrams for each command are illustrated in Chapter 7.)

Commands That Can Be a Source

Command	Definition
AMPU	Amplitude-unit conversion
FUNCDEF	Create user-defined function
MEAN	Mean value of data points
MEASU	Measurement-unit conversion
MEM	Available internal memory
MKBW	Marker bandwidth
MODADD	Return slave module's HP-MSIB address
MODID	Identify module or instrument HP-MSIB address
PEAKS	Trace peaks found
POSU	Position-unit conversion
PWRBW	Power bandwidth
RMS	Root mean square
STB	Status byte
STDEV	Standard deviation of trace values
SUM	Sum
SUMSQR	Sum of squares
VARIANCE	Variance
WLUNITS	Wavelength-unit conversion

Below, the `MOV` command transfers a value from the source (predefined variable) to the destination (user-defined variable).

```
10  OUTPUT 723;"MOV WAVELENGTH,CENTERWL;"
```
Copy center wavelength value into WAVELENGTH variable.

The next example uses the multiple command, `MPY`, to increase the center wavelength by 10%.

```
10  OUTPUT 723;"MPY CENTERWL,CENTERWL,1.1;"
```

Trace Math Rule 1

- Avoid truncating data from using a destination length that is shorter than the source length.

Data is lost when a trace is operated on and stored in a trace of lesser length (that is, of fewer points) or in a variable. When using an active source trace, the portion of the trace not stored is lost when it is updated with new sweep information. The example below shows how trace information may be lost when moving a trace array into a shorter trace array. Since the user-defined trace array, `TRACE`, is shorter than trace A, amplitude information contained in the last 400 data points is lost when trace A is updated.

```
30  OUTPUT 723;"SNGLS;TS;"          Single sweep.
40  OUTPUT 723;"TRDEF TRACE,400;"   Define user-defined trace with 400 elements.
50  OUTPUT 723;"MOV TRACE,TRA;"      Copy amplitude information of trace A
                                     into TRACE.
60  OUTPUT 723;"TS;"                 Update trace A with new sweep information.
```

When a short trace or variable is operated on and stored in a trace of longer length, the last trace element is extended for operations with the longer length.

Trace Math Rule 2

- Trace elements greater than $+32,767$ and less than $-32,768$ are truncated.

Math operations are restricted to the legal range of measurement units when trace arrays or elements form the destination or source.

```
10 OUTPUT 723;"MPY TRA[100],1000,1000;"
```
Multiplies 1000 times 1000, then stores truncated value (32,767) in trace element.

Trace Math Rule 3

- When the source is a real number or variable and the destination is a trace array or element, use the **MEASU** command to preserve decimal values and yield parameter units

The source and destination express numbers in different units, making a conversion necessary. All trace amplitude information is stored internally and manipulated in measurement units. If the conversion to measurement units is omitted, returned data is inaccurate by a factor of 100 and significant digits may be lost.

In the following example, the marker is assumed to be marking a signal at -10.33 dBm.

```
10  OUTPUT 723;"MOV TRB[100],MKA;"
```
Copy integer value of marker (−10) into trace B element. TRB[100] now contains −10 measurement units.

```
20  OUTPUT 723;"TRB[100]?;"
```
Return −0.1 dBm (false value).

In following example uses the **MEASU** command to preserves decimal values and returns trace amplitude values in parameter units.

```
10  OUTPUT 723;"MOV TRB[100],MEASU MKA;"
```
Convert marker amplitude (−10.33) to measurement units, then move value (−1033) into trace element. TRB[100] now contains −1033 measurement units.

```
20  OUTPUT 723;"TRB[100]?;"
```
Return −10.33 (correct value).

The following table lists math and move commands that may require the use
of the **MEASU** command.

Math and Move Commands Requiring Conversion with MEASU

Command	Definition
ABS	Compute absolute value.
ADD	Add sources, point by point.
AVG	Average source with destination, point by point.
BIT	Store or return value of bit.
CONCAT	Concatenate traces.
DIV	Divide sources, point by point, then discard remainder.
EXP	Divide source by scaling factor and raise to power of 10.
INT	Compute integer value.
LOG	Compute logarithm.
MEAN	Compute mean value of points.
MIN	Retain minimum value.
MOD	Divide sources, point by point, then retain remainder.
MOV	Duplicate value|s| contained in source and place in destination.
MPY	Multiply sources, point by point.
MXM	Retain maximum value.
RMS	Compute root mean square of source, point by point.
SQR	Compute square root of source, point by point.
STDEV	Compute standard deviation.
SUB	Subtract sources, point by point.
SUM	Add sources, point by point.
SUMSQR	Add square of sources, point by point.
VARIANCE	Compute variance.
XCH	Exchange sources.

Trace Math Rule 4

- For greatest , use single point destinations, such as a trace element or a variable.

If the destination consists of multiple points, the math functions round the sources first, then perform the requested operation. If the destination is a single point, the math functions compute the results in floating point, and then rounds the results if necessary. The following example shows how the product of 2.45 and 3.45 is 6 or 8 depending on the destination.

```
50   OUTPUT 723;"MPY TRC,2.45,3.45;"
60   OUTPUT 723;"TRC[1]?;"
70   OUTPUT 723;"MPY TRC[1],2.45,3.45;"

90   OUTPUT 723;"TRC[1]?;"
```

Fill trace C array with 2 X 3, or 6.
Return 6 to computer.
Loads trace element with integer value of 2.45 X 3.45, or 8.
Return 8 to computer.

3

Controlling Memory

Controlling Memory

Memory is an area where information can be stored for future recall. Memory that is available for the user is called **user memory**. This chapter discusses the control of the three types of user memory:

- Internal user memory, located within the optical spectrum analyzer

- External user memory located on HP-IB, such as a disc drive

- External user memory located on HP-MSIB, such as the memory card in the HP 70004A display

Reserved memory is for operations that are not under the user's control. Reserved memory is affected by power loss.

Contents

Saving and Recalling Items

You can save the following items in internal memory. The commands for saving are shown in parentheses.

- User-defined traces (TRDEF, LOAD)
- User-defined variables (VARDEF, LOAD)
- User-defined functions (FUNCDEF, LOAD)
- Active user-defined functions (ACTDEF, LOAD)
- On-end-of-sweep algorithms (ONEOS, LOAD)
- Trace-amplitude correction offsets (AMPCOR)
- State register files (instrument settings - STOR, NSTATE, SAVES)
- Trace files (STOR, SAVET)
- (USER) keys files (STOR, SAVEU)
- Limit-line files (STOR, LIMISAV)

Internal Battery

Internal memory is nonvolatile and is not altered by loss of power. Internal memory remains nonvolatile as long as the 3.6V lithium battery supplies power. The battery should last 8 years at 25° C, or one year at 55° C. Refer to the *Installation and Verification Manual* for installation instructions.

Unlike internal memory, external memory can only store items that are saved using the **SAVE** and **STOR** commands. External memory is nonvolatile and is not altered by loss of power. The following items can be stored in external memory.

- State register file (Instrument settings - **STOR**, **SAVES**)
- Trace file (**STOR**, **SAVET**)
- (USER) keys file (**STOR**, **SAVEU**)
- Limit-line file (**STOR**, **LIMISAV**)
- Program file (**STOR**, **SAVED**)

Use the mass-storage-interface command, **MSI**, to select the desired user memory. When power is applied to the analyzer, internal memory is selected

until the `MSI` command or the `catalog & MSI` softkey selects another storage device.

The catalog of user memory gives a description of articles stored in the currently selected user memory. To view the catalog, execute `DSPMODE CAT`. To return the catalog to the computer, use the `DSPTEXT` command. Stored articles are listed by name, followed by the article type, and the size of the article in bytes. The names of files stored by the save commands consist of a number preceded by a prefix that indicates the type of data stored in the file. The prefix can be changed with the PREFX command.

The articles that can be stored in user memory are listed in the following table.

Catalog of User Memory

Stored Article	Storing and Recalling Commands	File Type	Default File Prefix
User-defined trace*	TRDEF, LOAD, RCLD	TRACE	none
User-defined variable*	VARDEF, LOAD, RCLD	VAR	none
User-defined function *	FUNCDEF, LOAD, RCLD	FUNC	none
Active user-defined function *	ACTDEF, LOAD, RCLD	AFUNC	none
On-end-of-sweep algorithm *	ONEOS, LOAD, RCLD	ONEOS	none
Trace-amplitude correction offsets*	AMPCOR	AMPCOR	none
Limit-line trace arrays, LIMIT_LO and LIMIT_HI*	LIMILINE, LOAD, RCLD	TRACE	none
State-register file (instrument settings and title-line)	STOR, LOAD, SAVES, NSTATE, RCLS	ASCII	s_
Program file †	STOR, SAVED, RCLD	ASCII	d_
Trace file	STOR, LOAD, SAVET, RCLT	ASCII	t_
(USER) keys file	STOR, LOAD, SAVEU, RCLU	ASCII	u_
Limit-line file	STOR, LOAD, LIMISAV, LIMIRCL	ASCII	l_

*Indicates article can be stored in internal memory only

†Indicates file can be stored in external memory only

To select user memory

- Send the `MSI` command.

To catalog memory

- Send the `DSPMODE CAT` command to display catalog on OSA.

  or

- Send the `DSPTEXT` command to return the catalog to the computer.

To save a file

- Use the `SAVE` and `STOR` commands when saving to external memory.

  or

- Use the `ACTDEF`, `AMPCOR`, `FUNCDEF`, `LIMISAV`, `LOAD`, `NSTATE`, `ONEOS`, `SAVES`, `SAVET`, `SAVEU`, `STOR`, `TRDEF`, and `VARDEF` commands when saving to internal memory.

To recall a file

- Use the `LIMIRCL`, `LOAD`, `RCLD`, `RCLS`, `RCLT`, and `RCLU` commands to recall files from memory.

Erasing Memory

Use the `DISPOSE` command to erase items from internal memory. `DISPOSE ALL` erases *all* contents of internal memory. Use the PURGE command to erase items from either internal or external memory, depending on the storage device selected.

The `ERASE` command erases *all* memory in the OSA for security purposes, including internal memory, instrument settings, user-entered serial-number information, trace data, user-entered correction factors, and offsets. `ERASE` does not erase data in external memory, such as memory on a memory card or a disk drive. The partial erase command, PERASE, erases all internal memory and instrument settings. PERASE does not affect the correction factors or serial-number information.

CAUTION Use the `ERASE` command for security purposes only. `ERASE` destroys *all* contents of internal memory.

In the event of power loss, reserve memory is lost. For example, all active trace information is lost. The current values of all predefined variables and instrument settings are lost unless they are stored in a state-register file before power is removed. Instrument settings and predefined variables not saved in state registers may be retrieved when power is returned if the `POWERON` command is set to LAST before power is removed. Otherwise, the instrument settings and predefined variables return to their preset state when power is returned.

The FORMAT command erases internal or external memory. If a memory card or a disk drive (floppy) is selected with the MSI command, FORMAT initializes it.

Commands that Erase User Memory

Memory Components Storing Commands Shown in Parentheses	Commands that Erase or Reset Memory		
	PURGE FORMAT	DISPOSE* PERASE* ERASE*	NSTATE
user-defined variable (VARDEF, LOAD, RCLD)	X	X	
user-defined trace (TRDEF, LOAD, RCLD)	X	X	
user-defined function (FUNCDEF, LOAD, RCLD)	X	X	
active user-defined function (ACTDEF, LOAD, RCLD)	X	X	
on-end-of-sweep-algorithm (ONEOS, LOAD, RCLD)	X	X	
trace-amplitude correction factors (AMPCOR)	X	X	
state register file (instrument settings and title line, SAVES, STOR)	X	X	X
program file (SAVED, STOR)	X	X	
trace file(SAVET, STOR)	X	X	
(USER) keys file (SAVEU, STOR)	X	X	
Limit line file (LIMISAV, STOR)	X	X	
*DISPOSE, PERASE, and ERASE erase internal memory only.			

To erase files

- Use the `DISPOSE`, `ERASE`, `FORMAT`, `PERASE`, or `PURGE` command.

To erase all memory except correction factors and serial-number

- Execute the `PERASE` command.

To erase all memory for security

C A U T I O N

Use the **ERASE** command for security purposes only. **ERASE** destroys *all* contents of internal memory.

1. Execute the **ERASE** command.

2. Execute system-level command, `CLEAR 723` (in HP BASIC), to reset the OSA's input buffer.

3. Recycle the OSA's power.

4. Execute the error-correction routines to obtain calibrated measurement results. (Refer to the `CAL` command.)

Protecting Memory

The **PROTECT** command protects internal memory against erasure by the PERASE, FORMAT and DISPOSE commands. The PSTATE command protects state-register files on both internal or external memory.

On memory cards, the write-protect switch, when set to SAFE, protects the card's contents from erasure. For other HP-MSIB or HP-IB devices, refer to protection instructions for your particular mass storage device.

Internal Memory Erasure

The **PROTECT** and **PSTATE** commands do not prevent internal memory erasure with the **ERASE** command.

To protect files from erasure

- Use **PROTECT** to protect internal memory files.
- Use **PSTATE** to protect state register files.
- Use the memory card's write-protect switch to protect files on a card.

Commands that Erase or Protect User Memory

Memory Components Storing Commands Shown in Parentheses	Commands that Protect Memory	
	PROTECT†	PSTATE
user-defined variable (VARDEF, LOAD, RCLD)	X	
user-defined trace (TRDEF, LOAD, RCLD)	X	
user-defined function (FUNCDEF, LOAD, RCLD)	X	
active user-defined function (ACTDEF, LOAD, RCLD)	X	
on-end-of-sweep-algorithm (ONEOS, LOAD, RCLD)	X	
trace-amplitude correction factors (AMPCOR)	X	
state register file (instrument settings and title line, SAVES, STOR)	X	X
program file (SAVED, STOR)	X	
trace file (SAVET, STOR)	X	
(USER) keys file (SAVEU, STOR)	X	
Limit line file (LIMISAV, STOR)	X	
†PROTECT protects internal memory only.		

Commands that Erase Reserved Memory

Memory Components	Commands that Erase or Reset Memory		
	PERASE	ERASE	IP
correction factors (CAL)		X	
serial numbers		X	
active trace data (trace A, B, or C)	X	X	X
instrument settings not stored in state registers	X	X	X

Freeing Memory

When internal memory is full, an overflow message appears on the screen. You can increase available internal memory by erasing user-defined traces, variables, and functions. Use the `PURGE` and `DISPOSE` commands to erase these items.

Reduce trace size using the `TRDEF` command. An 800 element trace occupies 1646 bytes of memory, that is, 2 bytes for each element and 46 bytes for trace overhead. Reduce the size of user-defined functions by inhibiting the carriage-return/line-feed (CR/LF) generated by the computer at the end of each program line. The following HP BASIC example shows the HIGH_PEAK function with CR/LFs suppressed. (In HP BASIC, a semicolon placed at the end of a source line suppresses the CR/LF characters. With CR/LFs removed, HIGH_PEAK occupies 11% less memory space, that is, 47 bytes instead of 53.

```
30   OUTPUT 723;"FUNCDEF HIGH_PEAK,@";
40   OUTPUT 723;"STARTWL 700NM;STOPWL 900NM;";
50   OUTPUT 723;"TS;MKPK HI;";
60   OUTPUT 723;"@;"
```

To free the greatest amount of internal memory

```
10  OUTPUT 723;"MSI INT;"              Select internal memory.
20  OUTPUT 723;"PROTECT ALL,OFF;"      Disable protection.
30  OUTPUT 723;"DISPOSE ALL;"          Dispose all unprotected vari-
                                       ables, traces, functions, keys,
                                       and on-end-of-sweep algorithms.
```

To reduce the size of user-defined functions

- Suppress any carriage-return/line-feeds located at the end of source-code lines.

4

Monitoring System
Operation

Monitoring System Operation

This chapter describes how service requests are used to monitor system operation. Service requests can be set to occur automatically. You can also monitor system operation without using service requests, as explained in this chapter.

Contents

Monitoring System Operation

The programming techniques discussed so far describe communication
between the analyzer and the computer, where the sequence of all data
transfer is controlled by a computer program. This section tells how the
analyzer can interrupt computer operation when the analyzer has attained a
particular state.

The interrupting process is called a service request. Service requests have
many applications. They facilitate economical use of computer processing
time when the analyzer is part of a large-scale measurement system. For
example, after the computer initiates an analyzer measurement, the computer
can make calculations or control other devices via HP-IB while the analyzer
is measuring. When the analyzer is through, it signals the computer with a
service request. The service-request subprogram then determines what the
computer will do next. Service requests can also report analyzer errors while
interrupting computer processing.

Interrupt Process

The interrupt process begins when the analyzer "requests" attention by setting the HP-IB service-request (SRQ) line true. The computer responds by branching to a special subroutine that examines the status of each instrument on HP-IB. Computer operation then branches to another subroutine that is determined by the status of the instrument that requested service (the analyzer in this case). In this way, the computer "services" the analyzer's "request." These steps are summarized below:

1. Computer monitors HP-IB service-request (SRQ) line.

2. Analyzer requests service by setting SRQ line true.

3. Computer branches to subroutine that examines analyzer state.

4. Computer branches to subroutine determined by analyzer state.

Several system-level commands are required to make the computer responsive to service requests. The HP BASIC command, ENABLE INTR (enable interrupt), tells the computer to monitor the service-request line. The on-interrupt command, ON INTR, specifies where the computer program will branch when a service request occurs. If more than one service request is possible, the serial-poll command, SPOLL, is needed to examine the nature of the service request. These HP BASIC commands, ENABLE INTR, ON INTR, and SPOLL, are used with the HP 9000 Series 200 and 300 computers.

Analyzer Status-Byte Register

The analyzer **status-byte register** is the communicating link between the analyzer and the computer. The status-byte register indicates what state the analyzer is in.

The status-byte register consists of eight bits. Bits 0, 1, 2, 4, and 5 represent particular aspects of analyzer operation, such as end-of-sweep, or error conditions. These are listed in Table 1-13. The state of these bits, set (true) or cleared (false), is determined by the analyzer state.

If the service-request mode is enabled, all of the status-byte bits reflect the current state of the analyzer until the conditions of a service request are met.

In this case, the bit or bits that satisfy the service-request condition remain in the true state until the status byte is interrogated by the system-level, serial-poll command SPOLL, or until the system-level CLEAR command is executed. These commands clear the status byte so that all the bits in the byte once again reflect the current state of the analyzer until another service request occurs. In this way, the process begins again.

Service-Request Mask   The service-request mode is enabled and controlled by the request-service-condition command, RQS. It defines a **service-request mask** which specifies which of the status-byte bits may generate a service request. Below, RQS specifies the ERROR-PRESENT and COMMAND-COMPLETE states (bits 5 and 4) for service requests.

EXAMPLE Enable service requests for an ERROR-PRESENT condition.

    OUTPUT 723;"RQS 48;"   *Selects bit 5 and 32 to enable service-request mode for ERROR-PRESENT.*

Once RQS is executed, the analyzer requests service by setting the SRQ line true when any of the enabled conditions become true.

Optical Spectrum-Analyzer Status-Byte Register

Bit Number	Decimal Equivalent	Analyzer State	Description
7	128		Unused
6	64	RQS	Set when any selected service condition is met.
5	32	ERROR-PRESENT	Set when error register contains an error.
4	16	COMMAND-COMPLETE	Set at completion of command execution.
3	8		Unused
2	4	END-OF-SWEEP	Set at completion of sweep.
1	2	MESSAGE	Set when messages appear, such as the uncalibrated or corrected prompts, "UNCAL" or "USERCAL."
0	1	TRIGGER-ARMED	Set when analyzer is ready to receive a sweep trigger.

Service-Request Examples

The following program uses service requests to monitor measurement errors
in the analyzer. If an error occurs, computer operation is interrupted and a
description of the error is printed.

EXAMPLE Enable ERROR-PRESENT service request and return error
description.

```
10   ASSIGN @Sa TO 723                    Initialize analyzer.
20   COM\INSTR\@Sa
30   CLEAR @Sa
40   OUTPUT @Sa;"IP;SNGLS;TS;"
50   ON INTR 7 CALL Error                 Define computer interrupt branching.
60   ENABLE INTR 7;2                      Enable computer interrupts.
70   OUTPUT @Sa;"RQS 32;"                 Enable ERROR-PRESENT service request.
80   !
90   Idle:GOTO Idle                       Monitor service request line while measur-
                                          ing.
                                          (This is where your measurement routine
                                          belongs.)

100  !
110  END
120
130  SUB Error                           Interrupt-handling subroutine.
140  COM\INST\@Sa
150  DIM Mod_num$[6],Err_text$[30]
160  OUTPUT @Sa;"XERR?;"                  Return error description.
170  ENTER @Sa USING"5(K)";Error_num,
Mod_num$,Row_num,Col_num,Err_text$
180  PRINT Error_num;Mod_num$;
Row_num;Col_num;Err_text$
190  SUBEND
```

The next program uses service requests to allow the computer to process data or control other instruments while the analyzer is taking a long sweep. When the sweep is completed, the analyzer generates a service request which interrupts the computer, instructing it to accept the measurement data from the analyzer.

EXAMPLE Enable END-OF-SWEEP interrupt, then return trace data when the sweep has ended.

10	OPTION BASE 1	*Define first element of array as "1."*
20	ASSIGN @Sa TO 723	*Initialize analyzer.*
30	COM\INSTR\@Sa	
40	COM Trace_a(800)	
50	CLEAR @Sa	
60	OUTPUT @Sa;"IP;SNGLS;TS;"	
70	ON INTR 7 CALL Trace_out	*Define computer-interrupt branching.*
80	ENABLE INTR 7;2	*Enable interrupts.*
90	OUTPUT @Sa;"RQS 4;"	*Enable END-OF-SWEEP service request.*
100	OUTPUT @Sa;"ST 100S;"	*Set sweep time to 100 seconds.*
110	OUTPUT @Sa;"TS;"	*Sweep analyzer.*
120	!	
130	Idle: GOTO Idle	*Monitor service-request line while controlling other devices via HP-IB.*
140	!	
150	END	
160	!	
170	SUB Trace_out	*Trace-output subprogram.*
180	OPTION BASE 1	
190	COM\INST\@Sa	
200	COM Trace_a(800)	
210	OUTPUT @Sa;"TRA?;"	*Return trace data.*
220	ENTER @Sa;Trace_a(*)	
230	SUBEND	

The following program handles two types of service requests. One of them requests service when the analyzer has completed a sweep. The other requests service when an error occurs in the analyzer. When either request is generated, the computer examines the status-byte register using the SPOLL command to determine what caused the service request (end of sweep).

EXAMPLE Enable multiple service requests. Use HP BASIC command, SPOLL, to examine status byte.

```
10   OPTION BASE 1
20   ASSIGN @Sa TO 723
30   COM\INSTR\@Sa
40   COM Trace_a(800)
50   CLEAR @Sa
60   OUTPUT @Sa;"IP;SNGLS;TS;"
70   ON INTR 7 CALL Interrupt
80   ENABLE INTR 7;2
90   OUTPUT @Sa;"RQS 36;"
100 OUTPUT @Sa;"ST 10S;"
110 OUTPUT @Sa;"TS;"
120 !
130 Idle: GOTO Idle
140 !
150 END
160 !
```

```
170 SUB Interrupt
180 S=SPOLL (723)
190 IF BIT(S,2)THEN CALL Trace_out
200 IF BIT(S,5)THEN CALL Error
210 SUBEND
220 !
230 !
240 SUB Trace_out
250 OPTION BASE 1
260 COM\INSTR\@Sa
270 COM trace_a(800)
280 OUTPUT @Sa;"TRA?;"
290 ENTER @Sa;Trace_a(*)
300 SUBEND
310 !
320 SUB Error
330 DIM Mod_num$[6],Err_text$[30]
340 ASSIGN @Sa TO 723
350 OUTPUT @Sa;"XERR?;"
360 ENTER @Sa USING
    "S(K)";Error_num,Mod_num$,Row_num,Col_num,Err_text$
370 PRINT Error_num;Mod_num$;Row_num;Col_num;Err_text$
380 SUBEND
```

Monitoring System Operation Without Using Service Requests

The easiest method to synchronize the execution of commands with other events is with the DONE command. This has the disadvantage that the computer is not available for other processing while waiting for the analyzer to complete its commands.

The STB command may be used to monitor the value of the status byte at any time. Executing STB? returns the decimal equivalent of the status-byte register to the computer. Executing STB? is similar but not identical to conducting a serial poll of the status byte. Both methods will return the current value of hte status byte. When STB? is executed, the status-byte register is cleared, and then bit 4, COMMAND-COMPLETE, is set. When a serial poll is done, the status byte is cleared only if the service request mask is nonzero, as set by the RQS command.

To use the status byte to monitor the completion of any command without using the computer interrupt system, the following sequence should be followed:

1. Send all commands required before the command that is to be monitored.

2. Send DONE? to the analyzer followed by a query. The purpose is to ensure that all above commands have been completed.

3. Send RQS 16, COMMAND-COMPLETE, to the analyzer. This instructs the analyzer to generate a service interrupt when the command is complete. It is not necessary to enable the interrupt system of the computer.

4. Do a serial poll of the analyzer status byte (by using SPOLL in HP BASIC). This clears the status byte including bit 4, COMMAND-COMPLETE.

5. Send the command that is to be monitored to the analyzer.

6. Do a serial poll of the analyzer status byte anytime that it is desired in order to determine if the command sent in step 5 above has been completed. Bit 4 of the status byte will be clear until the command completes. Once the command is completed, bit 4 will be set the next time that the status byte is queried.

Debugging Programs

The OSA's debug feature help to locate faulty program code. When the mode is on, each command sent to the OSA is tested and then displayed on the OSA's data line (located below the graticule). When a faulty command is encountered, subsequent commands are not executed. The faulty command is the last command shown to the right end of the displayed data line. Use the `DEBUG` command to turn debugging on and off.

The debug mode has fast and slow settings. Use fast debugging for trapping errors. Slow debugging executes the program slowly so you can monitor the execution of each command, much like stepping through a program on a computer. Both slow and fast debugging modes slow the execution of analyzer commands and should be turned off when not needed, either with the front-panel controls or remotely via HP-IB.

The PAUSE command is another debugging function. When executed, PAUSE suspends program operation and activates the debugging front-panel keys, `STEP` and `CONT`. `STEP` executes the analyzer program, one command at a time. `CONT` continues program operation. Once program operation is halted, either by command or by error during debug mode, no further remote input is possible until `CONT` is pressed or `DEBUG On Off` is set to off, or a device-clear is sent (HP BASIC command CLEAR 723).

To debug a program

- Place the `DEBUG` command before a section of code that needs testing.

```
30   OUTPUT 723;"DEBUG SLOW;"          Activate slow debug mode.
40   OUTPUT 723;"SENS UP;ILLEGAL;"     Operation stops at illegal command, called
                                       ILLEGAL.
50   OUTPUT 723;"TS;"
60   OUTPUT 723;"MKPK HI;"
```

5

Creating Graphics

Creating Graphics

This chapter explains how to draw objects and print text on the display. A display, such as the HP 70004A, HP 70205A, or HP 70206A, is required for graphics capability.

Graphics allow you to customize the display for your applications. Some of the tasks you can perform include the following:

- Creating line drawings
- Displaying text messages
- Displaying measurement results
- Creating custom sized windows
- Changing display sizing and scaling

Contents

Clearing the Display

Use the following commands to clear the display for graphics:

- IP blanks the display line, trace B, and trace C by executing DL OFF, BLANK TRB, and BLANK TRC automatically.

- CLRDSP erases *all* user-created graphics from the display and memory, except the contents of the title line. If you have stored graphics you wish to recall for future display, use the DELETE command to erase items individually.

- ANNOT OFF blanks all annotation except the display status box and the data line.

- GRAT OFF blanks the graticule. You may create a different graticule with the GRID command explained at the end of this section.

- TRDSP blanks traces. Unlike BLANK, traces blanked with TRDSP can be updated with each sweep or marked with markers.

To clear the display

- Send the following command strings to the OSA.

  ```
  40 OUTPUT 723;"CLRDSP;"              Clear graphics from the display and memory.
  50 OUTPUT 723;"ANNOT OFF;GRAT OFF;"  Blank annotation and graticule.
  60 OUTPUT 723;"TRDSP TRA,OFF;"       Blank trace A.
  70 OUTPUT 723;"TRDSP TRB,OFF;"       Blank trace B.
  80 OUTPUT 723;"TRDSP TRC,OFF;"       Blank trace C.
  90 OUTPUT 723;"DL OFF;"              Blank display line.
  ```

Creating Graphics

Creating graphics is just like using a pen and paper. Graphics are drawn on the display using an imaginary pen on an electronic tablet. The PD (pen down) command draws on the display as the pen is moved. The PU (pen up) command lifts the pen. Use PU to reposition the pen without drawing. When the pen is lifted, previously drawn graphics are not erased (just as with a pen and paper). Use the LINET command to select the type of line drawn.

Once drawn, objects *cannot* be moved, erased, or redrawn. However, if you use IT to assign graphics objects to an item (refer to "Drawing Graphics Items" in this chapter), you can move, erase, or redraw them. The pen must be placed down to draw all objects except those defined with the item command, IT, or characters written with the TITLE or TEXT command. You must place the pen down in order to draw lines (PA and PR), grids (GRID), graphics markers (MK), or traces (GRAPH).

The pen is positioned on an X-Y coordinate system different ways. The PA command (plot absolute) moves the pen to an absolute position on the display. The PR command (plot relative) moves the pen to a position on the display relative to the current pen position. With each command, you specify a series of X-Y coordinates that moves the pen. The number of X-Y coordinates that can follow pen-positioning commands, like PA, is limited only by the memory in the display. Some graphics are drawn more effectively with the PR command, especially when the same object appears in more than one position on the display. Use PA to position the object on the display, then use PR to draw the object.

The following example shows how to draw an identical box using first the PA command and then the PR command.

```
100 OUTPUT 723;"PU;PA 0,0;PD;"              Position pen.
120 OUTPUT 723;"PA 300,0,300,200,0,200;";   Draw a rectangle using PA. Semicolon suppresses carriage
                                            return and line feed.
130 OUTPUT 723;"PU;PA 0,0;PD;"
140 OUTPUT 723;"PR 300,0,0,200,-300,0,0,-200;"   Draw a rectangle using PR.
```
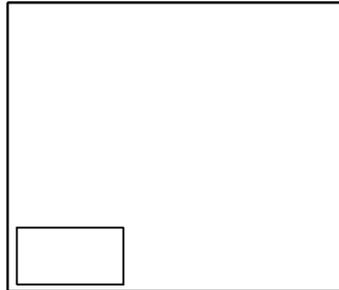
Draw a Rectangle with PA or PD

Carriage-return/line-feed

PA and PR commands are followed by a series of X-Y coordinate pairs. If several program lines are used to specify coordinate pairs, carriage return and line feeds must be suppressed at the end of each program line. Including these characters between X-Y coordinate pairs terminates the PA or PR commands before the entire list of coordinate pairs is processed.

The next example shows one way to move the rectangle to different positions on the display.

```
100 OUTPUT 723;"PU;PA 0,0;PD;"                  Position pen.
110 OUTPUT 723;"PR 300,0,0,200,-300,0,0,-200;"  Draw rectangle.
120 OUTPUT 723;"PU;PA 300,300;PD;"              Position pen.
130 OUTPUT 723;"PR 300,0,0,200,-300,0,0,-200;"  Draw rectangle.
140 OUTPUT 723;"PU;PA 600,600;PD;"              Position pen.
150 OUTPUT 723;"PR 300,0,0,200,-300,0,0,-200;"  Draw rectangle.
```
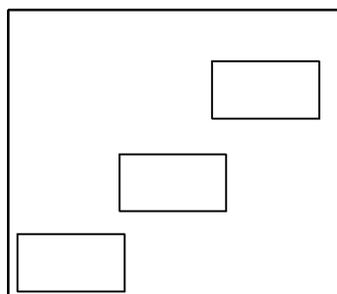

Reposition the Rectangle

The OSA draws different types of lines, which are specified by the line-type command, LINET. Once the line type is specified, all subsequent lines are drawn with that type. The following example draws rectangles with different line types.

```
100 FOR N=0 TO 750 STEP 150                     Repeat lines 110 through 140 for six values of N: 0,150,300,450,600,
                                                and 750.
110 L=L+1                                        Increase L by one.
120 OUTPUT 723;"LINET ";L;";"                   Select line type.
130 OUTPUT 723;"PU;PA ",N,N;";PD;"              Position pen.
140 OUTPUT 723;"PR 300,0,0,200,-300,0,0,-200;"  Draw rectangle.
150 NEXT N
```

The OR command offsets the position of graphics on the display relative to P_1, in the current scale units. The programs below draw a box. In the first example, the box is drawn without any offset. In the second, the box is offset 500 units in the X and Y directions.

```
120 OUTPUT 723;"PU;PA 0,0;PD;"                  Place pen down at 0,0.
```
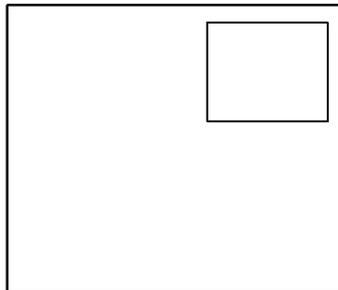
130 OUTPUT 723;"PA 0,400,400,400,400,0,0,0;"   *Draw to coordinates X1, Y1, X2, Y2, X3, Y3, X4, Y4.*
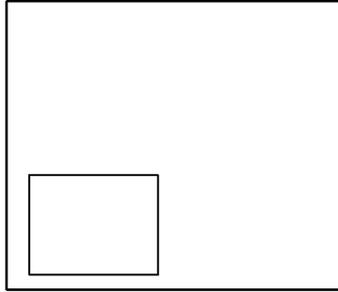
Draw Box Without Offset

120 OUTPUT 723;"OR 500,500;"   *Offset origin.*
130 OUTPUT 723;"PU;PA 0,0;PD;"   *Draw box.*
140 OUTPUT 723;"PA 0,400,400,400,400,0,0,0;"

Draw Box With Offset

To create a drawing

- Use the PU and PD commands to turn on or off the pen.
- Use PA to position the pen.
- Use PA and PR to draw the object.
- Use LINET to select the type of line (solid, dotted, or dashed.)

Drawing Grids

The GRID command draws a grid anywhere on the display. GRID
specifies the size and number of squares forming the grid. Use the PA
command to position the lower-left corner of the grid. LINET and PEN
may be used to modify the grid lines. The following example places a
five-division-by-four-division grid in the upper right quadrant of the display.
(GRID does not work after DWINDOW ON is executed.)

```
100 OUTPUT 723;"PU;PA 500,500;PD;"    Position pen at center of display.
110 OUTPUT 723;"GRID 100,125,5,4;"    Draw 5 by 4 division grid.
```
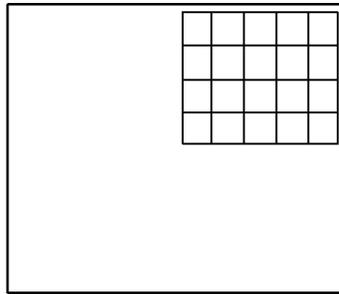
Illustration of the Drawn Grid

Drawing Graphics Items

The item command, IT, activates the special graphics mode where groups of graphics commands are assigned to an *item* number. These items can then be redrawn or modified without reexecuting the individual graphics commands. This is useful when a graphics item needs to be blanked and redrawn, changed, recolored, or repositioned at a later time. Executing IT assigns a series of graphics commands to an item number ranging from 1 to 50. The following program lines create items 1 and 2.

```
120 OUTPUT 723;"IT 1;"                            Assign box item number 1.
130 OUTPUT 723;"PA 0,0,0,400,400,400,400,0,0,0;"  Draw box.
140 OUTPUT 723;"IT 2;"                            Assign triangle item number 2.
150 OUTPUT 723;"PA 600,0,700,200,800,0,600,0;"    Draw triangle.
```
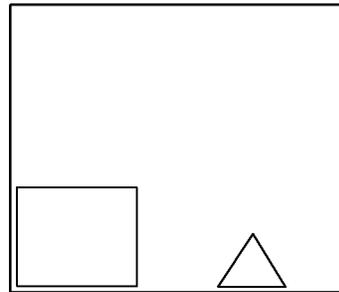
Graphics Items 1 and 2

Turning off IT mode

To turn off the special graphics mode entered with the IT command, you must execute one of the following commands: `IT O`, `IP`, `DELETE`, or `CLRDSP`.

The item command, IT, can be used to display line drawings, text, grids, or markers. These tasks are shown in the following table.

Item Tasks

Task	Command
Draw lines	PA, PR
Display traces	GRAPH
Draw grids	GRID
Display text	TEXT, DSPLY
Display graphics marker	MK

A single item can be assigned to do only one task. For example, a single item cannot draw a box and display text. In this case, two items are required: one for the box, another for the text. Mixing item types generates a display error, indicated by "E" in the display status box.

In previous chapter examples, graphics were positioned with the PA and PR commands and were drawn by placing the pen down with the PD command. The IT mode differs in that each item has its own pen, which is automatically in the down position. Thus, a pen-down command (PD) is not required. Also, in IT mode, graphics are positioned with the OR command without the use of the PU or PD. Their shape, color, and appearance (or disappearance) is further modified by a variety of commands. For example, consider the following lines that draw a grid using normal graphics techniques:

```
100 OUTPUT 723;"PU;PA 500,500;PD;"
110 OUTPUT 723;"GRID 100,125,5,4;"
```

The same grid can be drawn using IT (OR positions the grid; PD is not needed.)

```
110 OUTPUT 723;"IT 1;OR 500,500;"
110 OUTPUT 723;"GRID 100,125,5,4;"
```

The pen starts at the item origin when the item is first drawn. The origin of the item is 0,0 unless specified otherwise with the OR command. If the origin is changed, the entire item moves. Notice that when the reference-graphics mode is off (IT 0), graphics are always drawn at the current pen position and can never be moved.

An item can be redrawn or modified whenever it is active. Only one item can be active at a time. The active item is determined by the last executed IT command. The following table shows commands that modify items:

Commands that Modify Items

Description	Command
Blank item.	VW OFF |view item off|
Redraw item.	VW ON |view item on|
Offset item position.	OR |origin|
Change item shape.	TP |tablet pointer|
Stop display of item and delete from memory.	DELETE |delete item|

In the following example, item 1 is activated, blanked, and then redrawn. Notice that it is not necessary to execute the PD command or reexecute the individual graphics commands assigned to the item.

```
160 OUTPUT 723;"IT 1;"        Activate item.
170 OUTPUT 723;"VW OFF;"      Blank item.
180 OUTPUT 723;"WAIT 5;"      Pause program to view display.
190 OUTPUT 723;"VW ON;"       Redraw item.
```

Use the OR command to move an item on the display.

```
160 OUTPUT 723;"IT 1;"          Activate item 1.
170 OUTPUT 723;"OR 500,500;"    Offset item 1.
```
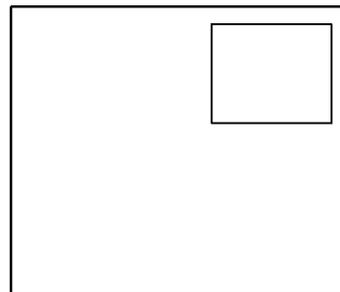
Moving Graphics Items

The TP (tablet pointer) command modifies item shapes. The TP command
may be thought of as a tablet pointer that selects one of the X-Y coordinates
of a graphics item for modification. For example, consider item 2, the
triangle, which is defined by four coordinates, numbered 0 through 3:

<div align="center">

Point 0   Point 1   Point 2 Point 3

|600,0| |700,200| |800,0| |600,0|

</div>

Executing **TP 1** selects (700,200) for modification. (TP begins counting with
zero.) The new coordinate is specified with the plot absolute command, PA,
that follows the TP command. In the program below, point 1 of the triangle is
modified.

```
160 OUTPUT 723;"IT 2;"          Activate item.
170 OUTPUT 723;"TP 1;"          Select point.
180 OUTPUT 723;"PA 300,600;"    Specify new point value.
```
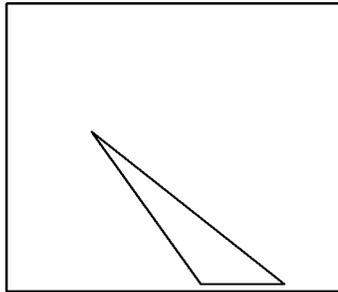

Modify Shape of Item

Any modification made with OR, VW, or TP become permanent characteristics
of the item unless the item is remodified or deleted with the DELETE or
CLRDSP command.

To use a graphics item

1. Create graphics items using the IT command, specifying item numbers from 1 to 50.

2. Activate selected items with the IT command.

3. Modify selected items with the VW, OR, TP, or DELETE commands.

To delete a graphics item

- DELETE blanks an *active* item and clears it from memory.

```
160 OUTPUT 723;"IT 1;"      Activate item.
170 OUTPUT 723;"DELETE;      Delete item.
```

- CLRDSP blanks and deletes *all* items, regardless of their state.

```
400 OUTPUT 723;"CLRDSP;"     Delete all items.
```

Displaying Text

Use TITLE and TEXT to display messages.

Messages may be written with the TITLE and TEXT commands. The TITLE command is the easiest way to write text. TITLE automatically places the pen down and positions text in the title line, located at the top of the display. The following example writes a message. Delimiters ($) mark the beginning and end of the message and must be different from any characters in the message. For a complete list of valid delimiters, refer to the "Syntax Diagram Elements" table in the introduction to Chapter 7. To erase the title, execute `IP` or `TITLE$$`. Executing `CLRDSP` does not erase the title line.

The TEXT command places messages (up to 35 characters) anywhere on the display. Use PA to position the pen, then use PD to place the pen down. (TEXT does not work after `DWINDOW ON` is executed.)

Use escape characters to enhance text.

You can enhance any text you write on the screen by using the following control and escape characters. Place these codes and characters within the label functions, or with any function that defines a string of text between delimiters, such as TITLE, TEXT, KEYDEF, or ACTDEF programming commands. Use the following codes to position the pen, clear or move text, or sound a beep.

Control Codes

ASCII Code	Function	Description
7	beeper $|B_L|$	Sound beep.
8	back space $|B_S|$	Move pen back by one character width.
12	form feed $|F_F|$	Clear line of text, then move pen to left side of line.
13	carriage return $|C_R|$	Move pen to left side of display, to beginning of line.
27	escape $|E_C|$	Use with functions listed in following tables.

In the following table, E_CY, turns on the display-function mode. When the display-function mode is on, the control codes and characters are displayed in the label; their associated functions (inverse video, underlining, and so on) are not executed, with the exception of carriage return and line feed, which are executed. When the mode is off, the code and character functions are executed instead of displayed. The mode is off when a label is created. When

a label is assigned a non-zero number with the IT command, the mode is on, and E_CY and E_CZ have no effect.

The table also shows character sequences used to enhance text. The characters must be preceded by the ASCII code for escape (27) and "&". All characters and codes must be entered between the delimiters of the label function, which must be assigned a non-zero item number. (Refer to the IT programming command.)

Escape Characters that Control the Pen, Clear Text, and Scroll Text

Character[1]	Description
E_CC	Move pen right by one character.
E_CG	Move pen to the left side of the display |carriage return|.
E_CK	Clear to the end of the line.
E_CY	Turn on the display-function mode.
E_CZ	Turn off the display-function mode.
E_C&dA	Turn on blinking.
E_C&Db	Turn on inverse video.
E_C&dC	Turn on inverse video and blinking.
E_C&dD	Turn on underline.
E_C&dE	Turn on underline and blinking.
E_C&dF	Turn on underline and inverse video.
E_C&dG	Turn on underline, inverse video, and blinking.
E_C&d@	Turn off enhancements.
E_C&v#S[2]	Selects color of text |# is between 1 and 15.|
E_C&v-#S[2]	Selects color of text background |# is between 1 and 15.|
E_C&v0S[2]	Resets to default color of text.

1 E_C indicates ASCII code for escape |27|.

2 Valid for color displays only.

Choose the color you want.

On HP 70004A displays, color text is available using the PEN programming command. To select the color, issue the PEN command before using the TITLE or TEXT command. The selected color applies to all subsequent text. To change the color of specific text within the text string, use the escape codes that are listed in the previous table. The escape-character sequence must be entered between the delimiters of the label function, which must be assigned a non-zero item number. (See IT programming command.) Colors can be selected with the escape sequence up to three times for each

label. Pen colors are not available on the HP 70205A or HP 70206A displays.
Sending pen-color escape sequences to the HP 70205A or HP 70206A displays
causes errors. The default PEN colors for the HP 70004A display are shown in
the following table:

Default Color Assignments for HP 70004A Display

Pen Number	Default Color[1]	Display Assignment
1	gray	GRATICULE
2	yellow	TRACE A
3	light blue	TRACE B
4	light red	TRACE C
5	green	LINES
6	bright white	TITLES
7	gray	ANNOTATION
8	orange	ADVISE text
9	red	ERRORS
10	bright white	MARKERS
11	white	softKEY LABELS
12	white	
13	red	
14	green	
15	blue	
16	black	BACKGROUND

1 Color can be adjusted using (DISPLAY) menus.

To display text

- Use the TITLE command to write text in the title area.

  ```
  20 OUTPUT 723;"TITLE $Your Message Here$;"
  ```

- Use the TEXT command to write text at any display location.

  ```
  30 OUTPUT 723;"PU;PA 400,700;"          Position pen.
  40 OUTPUT 723;"PD;"                     Place pen down.
  50 OUTPUT 723;"TEXT$Your Message Here$;"   Display text.
  ```

To erase text

- Execute CLRDSP to text written with the TEXT command.
- Execute **TITLE$$** to blank text written with the TITLE command.
- Assign the text to an *item* and blank the item.

  Refer to "Drawing Graphics Items" in this chapter.

Displaying Variables

Variables are displayed by using the display-variable command, DSPLY. DSPLY specifies the field width (number of characters) and decimal places of the displayed variable. When specifying the field width, be sure to account for the greatest number of characters that will be displayed, including the sign and decimal point. DSPLY places the variable at the current pen position.

To display a variable

- Use DSPLY to display values of variables.

```
110 OUTPUT 723;"PU;PA 500,700;"
120 OUTPUT 723;"DSPLY CENTERWL,12,0;"
```

To erase displayed variables

- Execute CLRDSP to blank variables written with the TEXT command.

Changing the Scale

Graphics are drawn in the OSA window on the display, usually 800 dots wide by 384 dots high. Most graphics are drawn between the lower-left position, P_1, and the upper right position, P_2, when the OSA has been assigned the full screen as shown in the following figure. P_1 is located at the left edge of the graticule, just above the data line. P_2 is located at the right edge of the graticule at the top of the display.
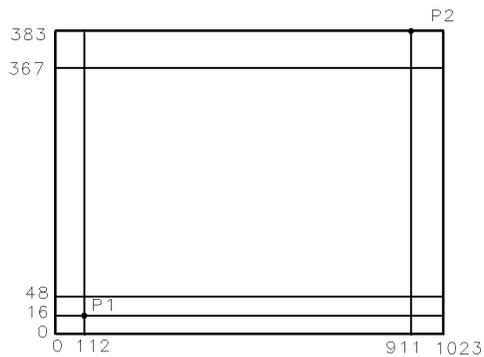
Default Positions for P_1 and P_2

The scale of the coordinates that P_1 and P_2 define has a preset value of 0,0 and 1000,1000. Most graphics commands position the pen according to the scale between P_1 and P_2. Graphics can be drawn outside the window, but this is not recommended.
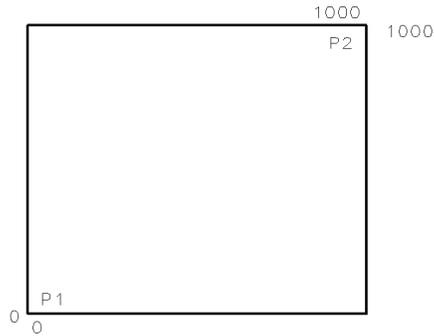
Default Scale for P₁ and P₂

The scale for P_1 and P_2 may be changed from its preset value to any values between $-32{,}768$ and $+32{,}767$. Below, SCALE changes the X and Y coordinates to new terms in decibel and degree units to illustrate an antenna pattern adjustment.
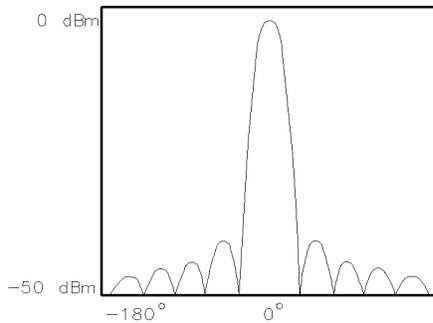
Set Scale in Terms of Degrees and Decibels

The SCALE command defines X-Y coordinates in integer values only. When redefining the scale, use numbers large enough to give adequate resolution. Units greater than 50 are recommended. The `build window` softkeys also change the scale of P_1 and P_2.

The following table list the commands that are affected by the SCALE command.

Commands Affected by the SCALE Command

Command	Description
DSPLY	Display variable
GRAPH	Graph trace
GRID	Display grid
MK	Display marker
OR	Set origin
PA	Plot absolute value
PR	Plot relative value
TEXT	Display text

Using Graphics Windows

Two window commands are provided for drawing on a limited and controlled area of the display: IWINDOW and DWINDOW. Use IWINDOW (instrument window) to increases or decreases the area on the screen dedicated to displaying OSA measurement results. Use DWINDOW to create a graphics window for drawing. DWINDOW cannot be used when displaying variables, grids, or text with the DSPLY, GRID, or TEXT commands.

Reducing the size of the graticule using IWINDOW leaves more room for messages and other graphics at the sides of the display screen.

Create a reduced display window using the following program.

```
20  CLEAR 723;
30  OUTPUT 723;"IP;SNGLS;TS;ANNOT OFF;"
40  OUTPUT 723;"SP 10NM;"                      Set span.
50  !
60  OUTPUT 723;"IWINDOW 1000,800;"             Reduce display area for mea-
                                               surement results.
70  OUTPUT 723;"PU;PA 0,955;PD;"               Position pen.
80  OUTPUT 723;"TEXT$SIGNALS FOUND:$;"         Write text.
90  !
100 FOR N=1 TO 3
110 OUTPUT 723;"CENTERWL ";(N*300);"NM;TS;"    Set center wavelength. Sweep
                                               trace A.
120 OUTPUT 723;"MKPK HI;"                       Mark highest signal level.
130 OUTPUT 723;"PU;PA 20,";(955-N*45);";";PD;" Position pen.
140 OUTPUT 723;"DSPLY ";N;",3,0;"              Label signal.
150 OUTPUT 723;"TEXT $AMPLITUDE = $;"
160 OUTPUT 723;"DSPLY MKA,6,1;"                Print amplitude.
170 OUTPUT 723;"TEXT$ dBm$;"
180 NEXT N
```

The DWINDOW command creates a graphics window on the display. The window has a coordinate system that you specify. Use the window for viewing re-sized measurement results, graphics, or traces. DWINDOW first specifies the physical size of the window in the current scale units. (As before, current scale units are either default values of 0 to 1000 for X and Y coordinates, or are specified by the SCALE command.) DWINDOW then defines a scale for the new coordinate system *within* the window. For example, the following program lines draw two boxes. One box is on the default coordinate system. The other box (using the same coordinates) is

placed on a window created by the DWINDOW command. Since the sides of
the box are still 1000 units long, it outlines the limits of the window.

```
20 OUTPUT 723;"PU;PA 0,0;PD;"                          Draw box.
30 OUTPUT 723;"PA 0,1000,1000,1000,1000,0,0,0;"        Define window size and scale.
40 OUTPUT 723;"DWINDOW 400,400,0,1000,0,1000;"
50 OUTPUT 723;"PU;PA 0,0;PD;"                          Redraw box.
60 OUTPUT 723;"PA 0,1000,1000,1000,1000,0,0,0;"
```
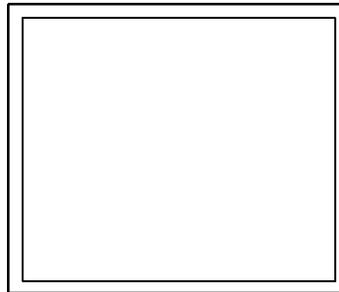
Use OR to position the window anywhere on the display. The OR command
offsets coordinates of the window in the current scale units, as defined by the
SCALE command.

```
100 OUTPUT 723;"PU;PA 0,0;PD;"                         Draw box.
110 OUTPUT 723;"PA 0,1000,1000,1000,1000,0,0,0;"       Define window size.
120 OUTPUT 723;"DWINDOW 400,400,0,2000,0,2000;"
130 OUTPUT 723;"OR 500,500;"                           Offset window position.
140 OUTPUT 723;"PU;PA 0,0;PD;"                          Redraw box.
150 OUTPUT 723;"PA 0,1000,1000,1000,1000,0,0,0;"
```
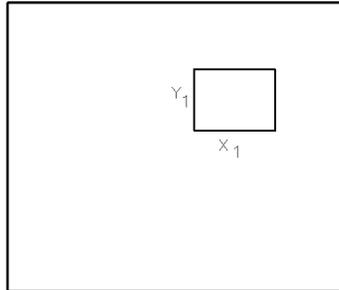
DWINDOW is especially useful for displaying traces. Trace A is reproduced below in a window.

```
140 OUTPUT 723;"DWINDOW 400,400,0,1000,0,1000;"
150 OUTPUT 723;"PU;PA 0,0;PD;"                    Draw box to outline trace.
160 OUTPUT 723;"PA 0,1000,1000,1000,1000,0,0,0;"
170 OUTPUT 723; "DWINDOW 400,400,TRA;"            Define window size and contents.
180 OUTPUT 723; "GRAPH TRA;"                      Display trace A.
```
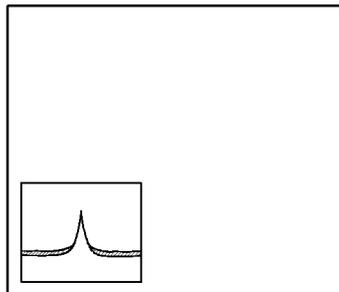

Display Trace Data within a Graphics Window

```
10 OUTPUT 723;"DWINDOW OFF;"
```

To reduce the display area

- Use the IWINDOW command.

```
100 OUTPUT 723;"IWINDOW 1000,800;"
```

The example above defines reduces the window's height from 1000 units to 800 units. The window's default height and width is 1000 units.

To create a graphics window

- Use the DWINDOW command to define the size of the window and its scale.

The following line defines a window at the current pen location. The window is 400 units wide and 300 units high. Its scale is 1000 units per side.

```
10 OUTPUT 723;"DWINDOW 400,300,0,1000,0,1000;"
```

To place a trace in a window

- Use the DWINDOW command. GRIG

The following program line creates a widow with a reduced height. Trace A is assigned to the window.

```
10 OUTPUT 723;"DWINDOW 1000,400,TRA;"
```

6

Creating Downloadable
Programs

Creating Downloadable Programs

This chapter tells how to store a series of commands in **internal memory** to form new, custom-made commands or user-defined functions called downloadable programs (DLPs). The looping and conditional constructs, REPEAT/UNTIL and IF/THEN/ELSE, and the interrupt functions, ABORT and RETURN, are also described.

DLPs have many applications. They are ideal for repetitive measurements, and make it possible to conduct complicated measurements with the press of a single front-panel key.

When a DLP is operating, HP-IB operation is not suspended. This leaves the computer free to control other devices on the bus while the OSA measures, processes, and stores data.

In addition, the operation of DLPs does not require a computer; the functions can be activated with the OSA keys. This is useful for smaller test systems, where custom keys help create a friendly user interface to simplify test procedures.

Contents

Creating a Downloadable Program

The FUNCDEF (define function) command consolidates a series of OSA commands into one operation and gives it a name. Executing the function name is the same as executing the series of commands contained in the function.

Reserved Words

Do not use the reserved words (as listed in Chapter 8 of this manual) as DLP names. Using reserved words interrupts the defining process and executes the invoked function. For example, if a DLP is named "ALL," the reserved word would disrupt the operation of any function modified by ALL, such as `DISPOSE ALL`.

Any names created by the VARDEF, ACTDEF, TRDEF, or FUNCDEF commands become reserved words until they are erased from internal memory.

A function must be *stored* in internal memory before it can be *executed*. In the example below, the FUNCDEF command stores a DLP named HIGH_PEAK in internal memory. Notice the matching delimiter characters (@) mark the beginning and end of a command series assigned to HIGH_PEAK.

```
10   CLEAR 723
20   OUTPUT 723;"IP;";
30   OUTPUT 723;"FUNCDEF HIGH_PEAK,@";
40   OUTPUT 723;"STARTWL 700NM;STOPWL 1200NM;";
50   OUTPUT 723;"TS;MKPK HI;";
60   OUTPUT 723;"@;"
```

The simplest way to store new functions in internal memory is with a computer program. However, if you have the auxiliary HP-HIL keyboard, you can easily store DLPs without the use of a computer.

The following figure illustrates a program structure that simplifies the process. The model simplifies debugging and documentation, and conserves memory.
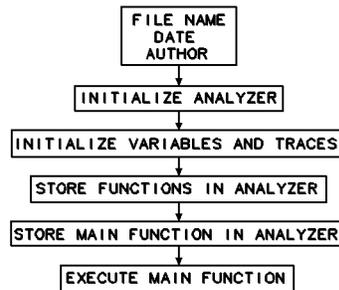
Program Structure for Storing DLPs to Internal Memory

In the model, the first part tells what the program does and gives other appropriate information such as the author's name, the date, and a file name.

The second part initializes the OSA in order to form a good starting point for measurements. `CLEAR 723` and `IP` set the OSA to a known state. `DISPOSE ALL` clears internal memory to maximize storage space for functions, variables, and traces. (Omit `DISPOSE ALL` if you do *not* wish to erase all of internal memory.)

The third part defines all user-defined variables and traces that are utilized by any of the user-defined functions, specifying a size for each trace and a value for each variable. To avoid memory-overflow problems, define variables and traces apart from any user-defined functions. For instance, if the user-defined function allocates memory for several trace arrays, memory is not allocated to the traces until the function has been executed. Thus, if the amount of available internal memory is checked with the `MEM?` command or the `catalog & MSI` softkey before the function has been run, the value returned or displayed accounts only for the space occupied by the function, not for the space occupied by the function *plus* the traces.

The fourth part is reserved for user-defined functions that are incorporated into and executed by a main user-defined function, which is the last part of the program. When a series of commands are consolidated in a function, they can be executed many times, even though they are stored in the OSA only once. This conserves memory and simplifies programming.

The following example illustrates this program structure. Note that the sub-function called VIEW_SIGNAL is executed twice in the main function, SCAN.

```
10  !File Name: Peaks  Date: 4/1/91  Author: Name
20  !View the largest signal in two ranges using narrow span.
30  !Store the amplitude of the largest signal in each range.
40   CLEAR 723                                          Initialize OSA.
50    OUTPUT 723;"IP;SNGLS;TS;";
60    OUTPUT 723;"DISPOSE ALL;";                        Erase internal memory.
70    OUTPUT 723;"VARDEF AMP_A,0;";
80    OUTPUT 723;"VARDEF AMP_B,0;";
90    ! Store sub-function in internal memory.
100   OUTPUT 723;"FUNCDEF VIEW_SIGNAL,@";               Begin definition of VIEW_SIGNAL function.
110   OUTPUT 723;"TS;MKPK HI;MKTRACK ON;TS;SP 100NM;";  Place marker on largest signal and view
                                                        it with narrow span width.
120   OUTPUT 723;"@;";                                  End definition.
130   ! Store main function in internal memory.
140   OUTPUT 723;"FUNCDEF SCAN,@";                      Begin definition of SCAN function.
150   OUTPUT 723;"IP;";                                 Set measurement range.
160   OUTPUT 723;"STARTWL 600NM;STOPWL 1000NM;";
170   OUTPUT 723;"VIEW_SIGNAL;";                        Execute function that marks the highest
                                                        signal and narrows span.
180   OUTPUT 723;"MOV AMP_A,MKA;";                      Store marker amplitude in a variable.
190   OUTPUT 723;"WAIT 2;";
200   OUTPUT 723;"STARTWL 1000NM;STOPWL 1700NM;";       Select new wavelength range.
210   OUTPUT 723;"VIEW_SIGNAL;";                        Execute function that marks the highest
                                                        signal and narrows the span.
220   OUTPUT 723;"MOV AMP_B,MKA;";                      Store marker amplitude in a variable.
230   OUTPUT 723;"@;";                                  End definition.
240   ! Execute main function.
250   OUTPUT 723;"SCAN;"
260   END
```

To store a function

- Write the FUNCDEFs using short program lines.
- Indent program lines to mark sub-functions and points of looping and branching.
- Use descriptive labels for variables, functions, and traces (memory space permitting).
- Define all variables and traces at the beginning of the program, not within a FUNCDEF definition.
- Use recommended syntax shown in the Language Reference chapter.
- Use output statements to send the FUNCDEFs and variables to the OSA.

To confirm a function is stored

1. Press (MENU), Misc, and then MORE 1 of 3.

2. Press catalog & MSI.

3. Press INTERNL MEMORY.

   The catalog of internal memory should now be displayed.

4. If the user-defined function was stored, it will be listed in the catalog.

For each stored article, the catalog gives the name of the article, followed by AFUNC or FUNC (for active or ordinary user-defined functions, respectively) and the size in bytes.

Creating an Active User-Defined Function

The ACTDEF command creates a user-defined function that operates like an active function. Active functions have the following characteristics:

- Their current value is displayed in the active-function readout.

- Their value can be changed with front-panel keys.

- Their value is expressed as a variable for operations with other OSA functions. For example, if ACTDEF creates a function called XXXX, the variable XXXX can be used wherever *predefined variable* appears in the syntax charts.

Like the FUNCDEF command, ACTDEF consolidates a series of OSA commands into one function and gives it a name. Executing the function name is the same as executing the series of commands contained in the function.

In the following example, the function is assigned to a (USER) key. Pressing the key executes the function. Active user-defined functions can also be executed with other functions, or used in computer programs as you use any other OSA command.

60	OUTPUT 723;"ACTDEF NEW_SPAN,10,NM";	*Begin definition of NEW_SPAN, that has an initial value of 10 nm.*
70	OUTPUT 723;"@MPY SP,2,NEW_SPAN@"	*Multiply NEW_SPAN by 2, then set span equal to the product. End definition and store the function.*
80	OUTPUT 723;"KEYDEF 3,NEW_SPAN;"	*Assign function to a (USER) key.*
90	OUTPUT 723;"VARDEF YY,0;"	*Create a user-defined variable, YY.*
100	OUTPUT 723;"MOV YY,NEW_SPAN;"	*Set variable equal to NEW_SPAN.*
110	OUTPUT 723;"NEW_SPAN?"	*Return value of NEW_SPAN to computer.*
120	ENTER 723;N	*Assign value to computer variable.*

In addition to creating a user-defined function, the ACTDEF command creates a temporary variable that has the same name as the user-defined function. The variable can be used for operations as you would use any **predefined variable**. The variable exists as long as the user-defined function is stored in internal memory. Do not use existing user-defined function names or reserved words as labels for user-defined functions.

When using a string- or block-data field to transfer a command definition, be sure to use compatible terminators. Also, any block-field commands within the list must be the absolute type (A# block).

Branching, Looping, and Interrupting

Use the IF/THEN/ELSE, REPEAT/UNTIL, ABORT, and RETURN commands to add decision-making, looping, or interrupt capability to user-defined functions.

The ABORT command stops the execution of all user-defined functions and interrupts the operation of repeat loops. If ABORT is encountered in a function that has been executed by pressing a key, the function is interrupted and front-panel control is returned.

If the function is activated by a computer program, ABORT resumes program operation at the first OSA command following the function. When, in addition, the function is nested within one or more user-defined functions, ABORT resumes program operation at the first OSA command following *all* of the user-defined functions.

The RETURN command stops the operation of the currently executing user-defined function, and returns to the point where the function was originally encountered.

The following example contains a user-defined function, PROGRAM_1, nested within another function, PROGRAM_2, which itself is nested within another function PROGRAM_3. Since none of the functions contain ABORT, executing PROGRAM_3 executes all three sequentially.

```
5    CLEAR 723                                    Initialize OSA.
10   OUTPUT 723;"IP;"
20   OUTPUT 723;"TRDSP TRA,OFF;"
30   OUTPUT 723;"ANNOT OFF;GRAT OFF;"
40    OUTPUT 723;"VARDEF YY,0;"
60    OUTPUT 723;"FUNCDEF PROGRAM_1,^";         Define PROGRAM_1.
70    OUTPUT 723;"REPEAT;";
80    OUTPUT 723;"ADD YY,YY,100;";
90    OUTPUT 723;"PU;PA 100,YY;PD;";
100   OUTPUT 723;"DSPLY YY,4,0;";
110   OUTPUT 723;"IF YY,EQ,300;THEN;ENDIF;";
120   OUTPUT 723;"UNTIL YY,EQ,400;";
130   OUTPUT 723;"^;"
140   !
150   !
160   OUTPUT 723;"FUNCDEF PROGRAM_2,";           Define PROGRAM_2.
```

```
170  OUTPUT 723;"^PROGRAM_1;TEXT@INSIDE PROGRAM_2@;";
180  OUTPUT 723;"^;";
185  !
190  OUTPUT 723;"FUNCDEF PROGRAM_3,";
200  OUTPUT 723;"^PROGRAM_2;TEXT@INSIDE PROGRAM_3@;";
210  OUTPUT 723;"^;";
220  OUTPUT 723;"PROGRAM_3;";
225  !
230  OUTPUT 723;"PU;PA 300,200;";
240  OUTPUT 723;"TEXT@OUTSIDE USER-DEFINED FUNCTIONS@;"
```

Execute PROGRAM_1 and write "INSIDE PROGRAM_2."

Define PROGRAM_3.

Execute PROGRAM_3. When all three user-defined functions have been executed, go to line 230.

What if in the program above, line 110 were modified to include the ABORT command in the PROGRAM_1 function?

```
110 OUTPUT 723;"IF YY,EQ,300;THEN; ABORT ;ENDIF;";
```

When ABORT is encountered, the PROGRAM_1 function is interrupted, the PROGRAM_2 and PROGRAM_3 functions are ignored, and the OSA commands following PROGRAM_3 are executed. Program operation resumes at line 230.

To use a conditional branch

● Use the IF/THEN/ELSE command

This example shows multiple conditional branching. If MKA is greater than −10 dBm or less than −30 dBm, the value of MKA is displayed. Otherwise, the message "SIGNAL WITHIN TOLERANCE" is displayed.

```
50  OUTPUT 723;"IP;CENTERWL 900NM;SP 400NM;ANNOT OFF;SNGLS;TS;MKPK
HI;"
60  OUTPUT 723;"IF MKA,GT,-10DBM;THEN;DSPLY MKA,6,2;"
70  OUTPUT 723;"ELSIF MKA,LT,-30DBM;THEN;DSPLY MKA,6,2;"
80  OUTPUT 723;"ELSE TEXT$SIGNAL WITHIN TOLERANCE$;"
90  OUTPUT 723;"ENDIF;"
```

To use a looping construct

● Use the REPEAT/UNTIL command

In this example, repetition is controlled with a user-defined variable, N, which incrementally increases each time the loop is repeated. The example stores the amplitudes of the three largest signals in the first three elements of a trace array. In addition, N doubles as a pointer that specifies where data is stored in the trace array.

10	CLEAR 723	*Initialize OSA.*
20	OUTPUT 723;"IP;"	
30	OUTPUT 723;"STARTWL 750NM;STOPWL 1450NM;"	*Activate single-sweep mode and*
40	OUTPUT 723;"SNGLS;TS;"	*sweep.*
50	OUTPUT 723;"DISPOSE ALL;"	*Clear internal memory. (Optional—see "Memory" section in this chapter.)*
60	OUTPUT 723;"TRDEF TR_AMP,10;"	*Define 10-element trace, TR_AMP.*
70	OUTPUT 723;"VARDEF N,0;"	*Define variable, N, equal to 0.*
80	OUTPUT 723;"MKPK HI;"	*Place marker on highest peak.*
90	OUTPUT 723;"REPEAT;"	*Begin repeat loop.*
100	OUTPUT 723;"ADD N,N,1;"	*Add 1 to N.*
110	OUTPUT 723;"MKPK NH;"	*Move marker to next highest peak.*
120	OUTPUT 723;"MOV TR_AMP[N],MEASU MKA;"	*Store amplitude in nth element of trace.*
130	OUTPUT 723;"UNTIL N,EQ,3;"	*Mark end of repeat loop. Looping ends when N equals 3.*

To abort loops and user-defined functions

● Use the ABORT command to continue at the next commands after the function.

● Use the RETURN command to continue from the point at which the current function was called.

Executing User-Defined Functions

After storing a user-defined function in internal memory with the FUNCDEF or ACTDEF command, execute the function from a computer or with the front-panel controls. To recall a function from external memory to internal memory, use the LOAD or RCLD command.

If you want your downloadable program to automatically appear in the (USER) menu, the name of the main FUNCDEF must end in a single underscore character.

Use `execute DLP` with caution.

The `execute DLP` softkey allows you to select and start a FUNCDEF. However, use caution because the function lists all FUNCDEFs—even those that are subroutines and do not run properly as stand-alone programs. To prevent a FUNCDEF from being listed by `execute DLP`, end the FUNCDEF's name with two underscore characters.

To assign a function to the USER menu

- End the name of the main FUNCDEF with a single underscore character.

To execute a function via computer

- Use an output statement to send the function name as you would any OSA command.

Be sure to terminate the function name with a semicolon. For example, to execute a user-defined function named HIGH_PEAK:

```
10  OUTPUT 723;"HIGH_PEAK;"
```

To execute a function via front-panel keys

1. Press `MENU` and then `Misc`.

2. Press `MORE 1 of 3`.

3. Press `execute DLP`.

    The `execute DLP` softkey lists all user-defined functions stored in internal memory (except those whose name ends in two underscore characters).

4. Press a key to execute its function. The LOAD or RCLU command recalls stored `USER` key files.

Building USER Menus

The KEYDEF and READMENU commands allow you to build complex interactive menus for your user-defined functions. Keys that you create are displayed by pressing the (USER) key. Up to 14 keys can be displayed at one time in the (USER) menu. The keys are numbered 1 through 14, keys 1 and 8 occupying the upper right and left corners, respectively. The following example, the HIGH_PEAK function is assigned to softkey 7 and labeled "FIND PEAK." To execute the HIGH_PEAK function, press (LOCAL), (USER), FIND PEAK.

```
10  OUTPUT 723;"KEYDEF 7,HIGH_PEAK,$FIND PEAK$;"
```

Use the READMENU command to create interactive (USER) key menus. In the following example, READMENU works with the IF/THEN/ELSE commands to build a (USER) key menu that activates three different wavelength ranges, called bands A, B, and C. Whenever a key is pressed, the READMENU command transfers the key number to a user-defined variable. The IF/THEN/ELSE commands then test the key number and activate the associated function. The REPEAT/UNTIL commands repeat the key-testing process until the exit key is pressed.

```
10   CLEAR 723
20   OUTPUT 723;"IP;DISPOSE ALL;"
30   !
40   OUTPUT 723;"FUNCDEF BANDA,@STARTWL 600NM;STOPWL 900NM;@;"    Band A.
50   OUTPUT 723;"FUNCDEF BANDB,@STARTWL 900NM;STOPWL              Band B
1200NM;@;"
60   OUTPUT 723;"FUNCDEF BANDC,@STARTWL 1200NM;STOPWL             Band C
1700NM;@;"
70   !
80   OUTPUT 723;"VARDEF KEY,0;"                                   Variable for storing key number.
90   OUTPUT 723;"VARDEF END_LOOP,0;"                              Declare variable for monitoring REPEAT
                                                                 loop.
100 !
110 OUTPUT 723;"REPEAT;"                                         Begin repeat loop.
160 OUTPUT 723;"READMENU KEY,1,$BAND                             Assign labels to (USER) keys 1, 2, 3,
A$,2,$BANDB$,3,$BANDC$,14,$EXIT$;";                              and 14.  Store pressed-key number in
                                                                 variable, KEY.
180 OUTPUT 723; "IF KEY,EQ,1;THEN;BANDA;";                       Evaluate KEY to execute function.
190 OUTPUT 723;" ELSIF KEY,EQ,2;THEN;BANDB;";
200 OUTPUT 723;" ELSIF KEY,EQ,3;THEN;BANDC;";
210 OUTPUT 723;" ELSIF KEY,EQ,14;THEN;MOV END_LOOP,1;";          When  key  14  is  pressed,  END_LOOP
                                                                 contains a "1" and ends repeat loop.
220 OUTPUT 723;" ENDIF;";
230 OUTPUT 723;"UNTIL END_LOOP,EQ,1;"
240 END
```

With further use of the READMENU, KEYDEF, and FUNCDEF commands, you can assign a (USER) key menu to a (USER) key that can be activated by yet another (USER) key, thus creating multilevel menus. The above example has been altered slightly in the next example to demonstrate one way of assigning a menu to another (USER) key. The entire program algorithm is incorporated into a user-defined function, which in turn is assigned to a single (USER) key called SELECT_SPAN .

```
10 CLEAR 723
20 OUTPUT 723;"IP;DISPOSE ALL;"
30 !
40  OUTPUT 723;"FUNCDEF BANDA,@STARTWL 600NM;STOPWL 900NM;@;"
50  OUTPUT 723;"FUNCDEF BANDB,@STARTWL 900NM;STOPWL 1200NM;@;"
60  OUTPUT 723;"FUNCDEF BANDC,@STARTWL 1200NM;STOPWL 1700NM;@;"
70 !
80 OUTPUT 723;"VARDEF KEY,0;"
90 OUTPUT 723;"VARDEF END_LOOP,0;"
100 !
105 OUTPUT 723;"FUNCDEF SELECT_SPAN,@";                          Create SELECT_SPAN.
110 OUTPUT 723;"REPEAT;";
160 OUTPUT 723;"READMENU KEY,1,$BAND A$,2,$BAND B$,3,$BAND
C$,14,$EXIT$;";
170 !
180 OUTPUT 723;"IF KEY,EQ,1;THEN;BANDA;";
190 OUTPUT 723;"ELSIF KEY,EQ,2;THEN;BANDB;";
200 OUTPUT 723;"ELSIF KEY,EQ,3;THEN;BANDC;";
210 OUTPUT 723;"ELSIF KEY,EQ,14;THEN;MOV END_LOOP,1;";
220 OUTPUT 723;"ENDIF;";
230 OUTPUT 723;"UNTIL END_LOOP,EQ,1;";
232 OUTPUT 723;"MOV END_LOOP,0;"
235 OUTPUT 723;"@;"
237 OUTPUT 723;"KEYDEF 7,SELECT_SPAN,$SELECT SPAN$;"            Assign SELECT_SPAN to softkey 7.
240 END
```

To assign a function to the ⌈USER⌉ menu

- Use the KEYDEF command.

To build interactive menus

1. Create a variable for recording the softkey pressed.

2. Use the READMENU command to load variable with number of softkey pressed.

3. Use IF/THEN/ELSE statements to make decisions based on variable value.

Avoiding Illegal Recursion

Executing a user-defined function within the FUNCDEF command statement that defines it is an illegal recursion.

```
10 OUTPUT 723;"FUNCDEF  SELECT_SPAN, @";
20 OUTPUT 723;"STARTWL 800NM;STOPWL 900WL;";
30 OUTPUT 723;"SELECT_SPAN;";
40 OUTPUT 723;"@;"
```

On the other hand, recursion is advantageous when creating *active* user-defined functions with the ACTDEF command.

`60  OUTPUT 723;"ACTDEF NEW_SPAN,500,nm,";`	*Begin definition of active, user-defined function, called NEW_SPAN, that has an initial value of 500 nm.*
`70  OUTPUT 723;"@MPY SP,2,NEW_SPAN;@"`	*Multiply NEW_SPAN by 2, then set span equal to the product. End definition and store the function.*
`80  OUTPUT 723;"KEYDEF 3,NEW_SPAN;"`	*Assign function to a* (USER) *key.*
`90  OUTPUT 723;"VARDEF YY,0;"`	*Create a user-defined variable, YY.*
`100 OUTPUT 723;"MOV YY,NEW_SPAN;"`	*Set variable equal to NEW_SPAN.*

Controlling Instruments Over HP-IB

This section tells how to use the OSA as a controller on HP-IB. As a controller, the OSA transfers data to or accepts data from other HP-IB devices, thus expanding its measurement capability. For example, the OSA can control an HP-IB switch driver to select the correct optical input for measurement. Or, the OSA can control a power meter and signal source.

Use the OUTPUT command to send data to an HP-IB device located at a particular HP-IB address. Below, the OSA sends the message "SWITCH ON" to a device at HP-IB address 19.

```
10   OUTPUT 723;"OUTPUT 19,KL,$SWITCH ON;$;"
```

Data is sent as free-field ASCII numbers in three different output formats, called, K, KC, and KL. The K format sends data without a terminator. The KC format terminates the data with a carriage- return/line-feed. The KL format terminates data with a line-feed and EOI (end-or-identify). Use the format that is compatible with the HP-IB device receiving the data. Data may also be sent as non-ASCII numbers using the F format, which specifies the number of characters (field width) and decimal places.

Use the ENTER command to receive data from an HP-IB device. Below, ENTER receives a value sent by the device at HP-IB address 19 and stores the value in a user-defined variable in the OSA.

```
10   OUTPUT 723;"OUTPUT 19,KL,$CENTERWL?;$;"      Request center wavelength value.
20   OUTPUT 723;"ENTER 19,K,NN;"                  Receive center wavelength value.
```

Three input formats for data are available: K for free-field ASCII numbers, B for bytes (8 bits), and W for words (2 bytes). Be sure to use an input format that is compatible with the output format of the HP-IB device. In the example below, the OSA sets the power of a source to the value of a variable. The message must be sent with several output statements, since one part of the message is a string of characters and another part is a OSA variable. Notice that a character string must be delimited.

```
10   OUTPUT 723;"OUTPUT 19,K,$PW $;"        Send letters "PW " to the source.
20   OUTPUT 723;"OUTPUT 19,K,POWER_W;"       Send current value of variable.
30   OUTPUT 723;"OUTPUT 19,KL,$WATTS;$;"     Send letters "WATTS" to the source.
```

Before the ENTER and OUTPUT functions are executed, the computer must be disconnected from HP-IB. The OSA can control the HP-IB only if a

computer is not in control of the HP-IB. (Only one controller can control HP-IB at any time.) For this reason, ENTER and OUTPUT commands are incorporated into user-defined functions that are executed by the OSA with user-defined keys. This is accomplished by either physically disconnecting the computer from the bus, or by executing the HP-IB system-level commands that release the computer from HP-IB. The HP-BASIC command sequence that does this is shown below.

```
10  LOCAL 7
20  ABORT 7
```

The release HP-IB command (RELHPIB) must be included at the end of any user-defined functions containing ENTER or OUTPUT commands. This ensures that the OSA surrenders control of HP-IB after all ENTER and OUTPUT commands have been executed.

To control instruments over HP-IB

- Disconnect any computer from the bus.

- Use the OUTPUT command to send data to an instrument.

- Use the ENTER command to receive data from an instrument.

7

Language Reference

Language Reference

This chapter is organized alphabetically and documents each of the remote programming commands for the optical spectrum analyzer. Documented with each command is a syntax diagram, a program example, and a complete description.

Command syntax is represented pictorially as shown in the following figure.

- Characters enclosed by an oval are commands and their arguments and must be entered *exactly as shown*.

- Characters enclosed by a circle are separators or terminators and must be entered as shown.
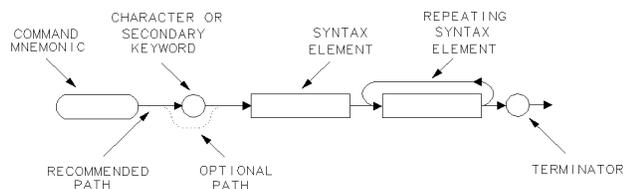
Syntax-diagram elements are connected by solid and dotted lines. Any combination of elements generated by following the lines in the proper direction is correct syntactically. *Solid-line paths are recommended.*

The following two tables define the syntax diagram elements used in this manual.

Syntax Diagram Elements

Syntax Component	Definition/Range
analyzer command	Any spectrum-analyzer command in this chapter, with required parameters and terminators.
character	S_P " - < = > @ ` { | } ~ ! " # $ % & ' | | + , . / 0 1 2 3 4 5 6 7 8 9 : ; A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\ ] ? '| a b c d e f g h i j k l m n o p q r s t u v w x y z _
compatible function	Any spectrum-analyzer command in this chapter that operates on other functions.
data byte	8-bit byte containing numeric or character data.
data byte & EOI	8-bit byte containing numeric or character data sent with end-or-identify |EOI| condition, where the EOI control line on HP-IB is asserted to indicate the end of the transmission.
delimiter	~ | \ @ = < > / ` $ % ! ' ' : " & Matching characters that mark the beginning and end of a character string, or a list of user-defined functions or analyzer commands. To avoid misinterpretation, choose delimiting characters that are not also within the string they delimit.
digit	0 1 2 3 4 5 6 7 8 9
lsb length	Represents the least significant byte of a two-byte word that describes the number of bytes returned or transmitted. See *msb length*.
msb length	Represents the most significant byte of a two-byte word that describes the number of bytes returned or transmitted. See *lsb length*.
number	Expressed as integer, decimal, or in exponential |E| form. Real Number Range: $\pm$ 1.797,693,134,862,315 $\times$ 10^{308}, including 0. Up to 15 significant figures allowed. Numbers may be as small as $\pm$ 2.225,073,858,507,202 $\times$ 10^{-308} Integer Number Range: $-32,768$ through $+32,767$
output termination	Line feed |L_F| with end-or-identify |EOI| condition. ASCII code 10 |line feed| is sent via HP-IB, with the end-or-identify control line on HP-IB set to indicate the end of the transmission.

Syntax Diagram Elements (continued)

Syntax Component	Definition/Range
predefined function	AMPU, WLUNITS, MEASU, MEM, MKBW, PEAKS, POSU, PWRBW, RMS, STB, STDEV, SUM, SUMSQR, VARIANCE. A *predefined function* is an *analyzer command* that returns a number that can be operated on by other commands called *compatible functions*. Insert a *predefined function* into a command statement where *predefined function* appears in the command syntax chart. (See *compatible function* in this table.)
predefined variable	The values of the following variables change depending on the current instrument settings. Each variable represents the value of the command function that has the same name as the variable. AT CALPWR CENTERWL CNVLOSS DL LG MKA MKACT MKAL MKAR MKWL MKN MKP MKPAUSE MKPX MKT MKTV ML NSTATE RB RBR RL RLPOS ROFFSET SIGDEL SP SS ST STARTWL STOPWL TH VB VBR VTL WLOFFSET
trace element	Value contained in one trace point. Notated as TRA[N] where N specifies the point position in the trace array. Values for N are 1 to 2048, or 1 to the length specified by TRDEF, whichever is less. Same for trace B (TRB[N]), trace C (TRC[N]), and user-defined traces (LABEL[N]).
trace range	Values contained in trace segment. Multi-point segments are notated as TRA[N,M], where N and M are end points of a segment and specify point positions in trace array. Values for N or M are 1 to 2048, or 1 to the length specified by TRDEF, whichever is less. Same for trace B (TRB[N,M]), trace C (TRC[N,M]), and user-defined traces (LABEL[N,M]). Single-point segments are notated same as *trace element* above.
units	Represent standard scientific units. Amplitude Units: DB, DBM, MW, NW, PW, UW, W Current Units: A, MA, UA Frequency Units: HZ, KHZ, MHZ, GHZ, KZ, MZ, GZ Time Units: S, MS, US, SC Wavelength Units: ANG, KM, NM, UM, PM

Syntax Diagram Elements (continued)

Syntax Component	Definition/Range
user-defined function	A label 1 to 12 characters long that is defined by the FUNCDEF or ACTDEF command. Choice of characters is A – Z, 0 – 9, and the underscore |_|. First character must be a letter. Numbers must be preceded by the underscore.
user-defined trace	A label 1 to 12 characters long that is defined by the TRDEF command. Choice of characters is A – Z, 0 – 9, and the underscore|_|. First character must be a letter. Numbers must be preceded by the underscore.
user-defined variable	A label 1 to 12 characters long that is defined by the VARDEF or ACTDEF command. Choice of characters is A – Z, 0 – 9, and the underscore|_|. First character must be a letter. Numbers must be preceded by the underscore.
value	See expanded diagram in syntax diagram.

Terminators, Separators, and Commonly Used Arguments

Item	Description
,	Comma. ASCII-decimal number 44.
;	Semicolon. ASCII-decimal number 59.
S$_P$	Space. ASCII-decimal number 32.
C$_R$	Carriage return. ASCII-decimal number 13.
L$_F$	Line feed. ASCII-decimal number 10.
DN	Down. Change function value by step increment. Simulates press of (▼) key.
UP	Up. Change function value by step increment. Simulates press of (▲) key.
EP	Enter parameter. Suspend program operation until number is entered with front-panel controls |step keys, knob, or numeric keypad.| Pressing **ACCEPT VALUE** enters the number and resumes program operation.
?	Query. Returns current value of function to computer.
TRA	Select trace A for operation.
TRB	Select trace B for operation.
TRC	Select trace C for operation.

ABORT

Stop the execution of a user-defined function.
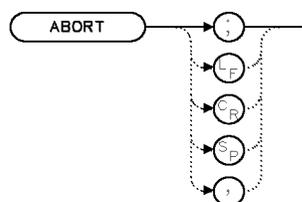
XABORT

Example Use ABORT to interrupt operation of a repeat loop.

120	`CLEAR 723`	*Initialize analyzer.*
130	`OUTPUT 723;"IP;"`	
140	`OUTPUT 723;"VARDEF KEY,0;"`	*Define a variable for storing the number of the key that was last pressed.*
145	`OUTPUT 723;"FUNCDEF EXAMPLE,%";`	*Create user-defined function EXAMPLE.*
150	`OUTPUT 723;"REPEAT;";`	*Begin repeat loop.*
160	`OUTPUT 723;"READMENU KEY,";`	*Store the number of the pressed key in the user-defined variable, KEY. The keys are defined in the next program lines.*
170	`OUTPUT 723;"1,%SET SPAN  %,";`	SET SPAN *is key number 1.*
180	`OUTPUT 723;"2,%EXPAND RANGE%,";`	EXPAND RANGE *is key number 2.*
190	`OUTPUT 723;"14,%EXIT          %;";`	EXIT *is key number 14.*
200	`OUTPUT 723;"IF KEY,EQ,1;";`	*If KEY equals 1, set measurement range to 1300 nm.*
210	`OUTPUT 723;"THEN;CENTERWL 1300NM;SP 10NM;";`	
220	`OUTPUT 723;"ELSIF KEY,EQ,2;";`	*If KEY equals 2, change the measurement range to 1550 nm.*
230	`OUTPUT 723;"THEN;CENTERWL 550NM;SP 10NM;";`	
240	`OUTPUT 723;"ELSIF KEY,EQ,14;";`	*If KEY equals 14, then stop repeating lines 150 – 280, and go to line 290 (ABORT).*
250	`OUTPUT 723;"THEN;ABORT;";`	
260	`OUTPUT 723;"ENDIF;";`	
270	`OUTPUT 723;"PA 500,500;";`	
280	`OUTPUT 723;"UNTIL 1,EQ,0;";`	*End repeat-loop definition.*
290	`OUTPUT 723;"TRDSP TRA,OFF;"`	*Turn off trace A.*
300	`OUTPUT 723;"TEXT @EXIT KEY ENDS REPEAT LOOP.@;%;";`	

Description The ABORT command stops the execution of a user-defined function. If ABORT is encountered in a function that has been executed by pressing a key, the function is interrupted and front-panel control is returned. If the function is activated by a computer program, ABORT resumes program operation at the first analyzer command following the function. When, in addition, the function is nested within one or more user-defined functions, ABORT resumes program operation at the first analyzer command following *all* of the user-defined functions.
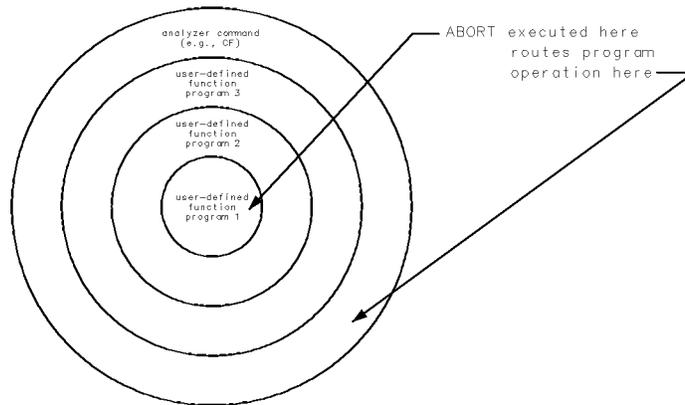
The RETURN command also interrupts operation of a user-defined function, but instead returns operation to the point where the user-defined function was called.

See Also ACTDEF, FUNCDEF, REPEAT/UNTIL, IF/THEN

ABS   Store the absolute values of the source, point by point, in the destination.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example

```
140   OUTPUT 723;"VARDEF NN,0;"
150   OUTPUT 723;"ABS NN,TRA[2];"
```

Define variable NN equal to 0.
Move absolute value of second element of trace A into NN.

Description The ABS command performs a point-by-point absolute value of the source variable or trace, the stores the results in the destination variable or trace. When the source length is longer than the destination length, excess data is ignored or truncated. When the source length is shorter than the destination length, the last point is duplicated to fill the destination.

When using TRA, TRB, TRC, or trace range for source, use SNGLS and TS to be sure data reflects one complete sweep. TRDEF required when using user-defined trace. VARDEF required when using user-defined variable.

ACTDEF   Activate a user-defined active function.

Item	Description
Function-Name Range	1 – 12 characters: A – Z, 0 – 9, and underscore |_|. First character must be a letter. Numbers must be preceded by underscore.
Readout Range	As allowed by selected units.
Initial-Value Range	As allowed by selected units.

Example Create a user-defined function whose value can be changed with front-panel controls or with math operations.

60	OUTPUT 723;"ACTDEF NEW_SPAN,500,NM,";	*Begin definition of active, user-defined function, called NEW_SPAN, that has an initial value of 500 nm.*
70	OUTPUT 723;"@MPY SP,2,NEW_SPAN;@"	*Multiply NEW_SPAN by 2, then set span equal to the product. End definition and store the function.*
80	OUTPUT 723;"KEYDEF 3,NEW_SPAN;"	*Assign function to a* (USER) *key.*
90	OUTPUT 723;"VARDEF YY,0;"	*Create a user-defined variable, YY.*
100	OUTPUT 723;"MOV YY,NEW_SPAN;"	*Set variable equal to NEW_SPAN.*

Description The ACTDEF command stores in **internal memory** a user-defined function that operates like an active function. ACTDEF first consolidates a series of analyzer commands into one function, and then gives it a name. Executing the function name is the same as executing the series of commands contained in the function. (ACTDEF does *not* execute the function.)

In addition to creating a user-defined function, the ACTDEF command creates a temporary variable that has the same name as the user-defined function. The variable can be used for operations as you would use any **predefined variable**. The variable exists as long as the user-defined function is stored in internal memory.

The ACTDEF command is unique in that it makes the user-defined function act like an active function. Active functions have the following characteristics:

- Their current value is displayed in the active-function readout.

- Their value can be changed with front-panel controls, provided the function is activated by a front-panel key.

- Their value is expressed as a variable for operations with other spectrum-analyzer functions.

For example, consider the active function for center wavelength. When the center-wavelength key is pressed, its value is displayed in the active-function readout. Turning the front-panel knob changes the active-function readout *and* the value of the predefined variable for center wavelength, CENTERWL. Changing the value of CENTERWL, in turn, changes the value of other active functions related to wavelength, such as start and stop wavelength:

$$STARTWL \quad = \quad CENTERWL - SPAN/2$$

$$STOPWL \quad = \quad CENTERWL + SPAN/2$$

The user-defined functions created by ACTDEF have the following characteristics:

- Text is displayed in the active-function readout, as indicated in the syntax diagram. If omitted, the *user-defined function name* is displayed in the active-function readout.

- A temporary variable is created and assigned the *initial value* indicated in the syntax diagram. The value is displayed in the active function readout.

- The value of the temporary variable can be changed. Each time the value is changed, the user-defined function (having the same name as the variable) is executed. If the user-defined function has been executed with a front-panel key, the value is changed with the front-panel step keys, knob, or numeric keypad. Remotely, the value is changed by math operations or by executing the user-defined function name followed by a number (see syntax diagram after description).

- The value of the temporary variable can be returned to the computer, or used for operations with any spectrum-analyzer commands that have *predefined variable* in their syntax diagrams.

```
10  OUTPUT 723;"ACTDEF COUNT, $Enter number.$,5,INT,$$;"
```
Create user-defined function.
```
20  OUTPUT 723;"COUNT;"
```
Execute user-defined function.

produces these results:

- The user-defined function called COUNT is stored in internal memory.

- "Enter number" is displayed in the active function readout.

- A temporary variable called COUNT is created, and assigned the initial value. The variable is available for math operations.

- The current value of the function (and of the temporary variable) is displayed in the active function readout. Number 5 is displayed initially.

- The step keys, knob, and numeric keypad can change the function value.

In the preceding example, the user-defined function routine is a null function. (There are no commands between delimiters $$ in line 10.) In comparison, the following program characterizes the value of COUNT in line 70. In addition, COUNT is executed by front-panel key instead of a programming command.

The following example creates an active user-defined function.

60	OUTPUT 723;"ACTDEF COUNT,500,NM,";	*Begin definition of active, user-defined function, called COUNT, that has an initial value of 500 nm.*
70	OUTPUT 723;"$MPY SP,2,COUNT;$"	*Multiply the current value of COUNT by 2, then set the span equal to the product. The "$" ends the definition.*
80	OUTPUT 723;"KEYDEF 3,COUNT;"	*Assign the function to a key.*

To execute the COUNT function defined above, press (LOCAL), (USER), COUNT .

Do not use existing user-defined function names or reserved words as names for user-defined functions. When using a string- or block-data field to transfer a command definition, be sure to use compatible terminators. Also, any block-field commands within the list must be the absolute type (A#-block).

Units argument   In the syntax diagram, the *units* arguments determine

- what unit keys are activated during front-panel number entry.

- whether the temporary variable can be used in place of **predefined variable** during math operations.

- how the step functions, UP, (▲), DN, and (▼), change the value of the temporary variable.

Behavior of Unit Arguments

Argument	Step-Increment Value	Unit-Keys	Math Operation Compatibility
STEP	1	none	yes
INT |integer|	1	none	no
H	Resolution bandwidth during zero-span mode; otherwise, 10% of span.	frequency	yes
ABSH |absolute hertz|	1, 3, 10 sequence	frequency	yes
M	Resolution bandwidth during zero-span mode; otherwise, 10% of span.	wavelength	yes
ABSM |absolute meters|	1, 3, 10 sequence	wavelength	yes
SEC	1, 3, 10 sequence	time	yes
DBM	10 dB	absolute amplitude	yes
W	1, 3, 10 sequence	watts	yes
A	1, 3, 10 sequence	amperes	yes

Step types argument   In the syntax diagram, the *step types* argument is an optional parameter that selects unit step progressions for the various supported ACTDEF unit types. A value of 0, as well as the omission of this argument, selects the default. The following tables lists the valid values for this argument.

Behavior of Step Types Arguments

Argument	Progression	Definition
0	linear	Default step.
1	linear	Step by 1.
2	linear	Step by the current vertical division size.
3	linear	Step by the current CF step size.
4	linear	Step using the same pattern as for wavelength |LAMBDA| stepping.
5	linear	Step using the same pattern as for responsivity stepping.
6	linear	Step by one wavelength division in non-zero span, otherwise by the resolution bandwidth.
7	linear	Step by the current horizontal division size.
8	linear	Step by the current vertical division size, but switch to a 1,2,5 log step if the current value is less than one vertical division.
9	linear	Step by the current span step size using the current source power step size.
10	linear	Step by 10.
11	log	Do a 1,2,5 log step progression.
12	log	Do a 1,3 log step progression.

See Also   DISPOSE, FORMAT, LOAD, PROTECT, RCLD, PURGE

ACTPARM   Turn the active parameter area on or off.
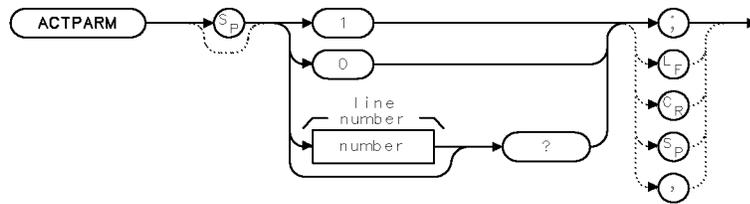
xactpa

Item	Description
Preset State	OFF
Number Range	0, 1, 2, 3, 4 |If the "number" is omitted, the command default is 0, the first line.|

Example

```
10   OUTPUT 723;"ACTPARM 1;"        Turn on the active parameters.
20   OUTPUT 723;"ACTPARM?;"         Query the top line value.
30   OUTPUT 723;"ACTPARM 0?;"       Query the top line value.
```

Description The ACTPARM command turns the display of the active parameter area on or off. The active parameter area has the capability of containing four lines of data. Therefore, when querying the active parameter area, the following data lines are placed on the bus for the computer:

- 0 queries the first line.
- 1 queries the second line.
- 2 queries the third line.
- 3 queries the fourth line.
- 4 queries all four lines.

Query Response
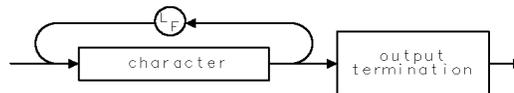
qgenchar

ADAPBTL

Performs trace math A * DL / (A + B) ▶ A with each sweep.
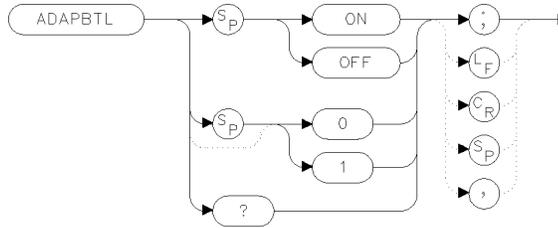
xadatl

Item	Description
Preset State	ADAPBTL OFF

Example

120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"LN;"	*Select linear amplitude scale.*
140	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and store it.*
150	OUTPUT 723;"ADAPBTL ON;"	

Description

The ADAPBTL command requires a linear display mode. It multiplies trace A by the display line and then divides the result by the sum of traces A and B. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep.

This command allows ratio measurements to be made as a sweep is taken. If trace B has points near the bottom of screen, overflow or poor precision can occur. If a division by zero is attempted, a division by the smallest representable value will be performed instead.

To ensure that correct data is displayed, be sure that a sweep is taken before using this command. Only one trace math operation may be active at any given time.

Query Response
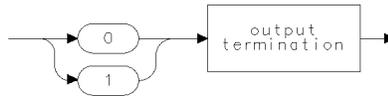
qgen0

ADAPBPCTL

Performs trace math A * DL / (A + B + C) ▶ A with each sweep.
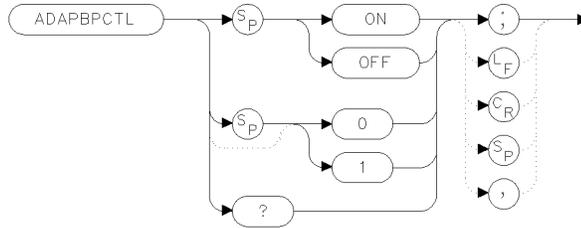
xadapc

Item	Description
Preset State	ADAPBPCTL OFF

Example

```
120   OUTPUT 723;"SNGLS;"          Activate single-sweep mode.
130   OUTPUT 723;"LN;"             Select linear amplitude scale.
140   OUTPUT 723;"TS;VIEW TRA;"    Sweep trace A and store it.
150   OUTPUT 723;"ADAPBPCTL ON;"
```

Description

The ADAPBPCTL command requires a linear display mode. It multiplies trace A by the display line and divides the result by the sum of traces A, B, and C. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep.

This command allows ratio measurements to be made as a sweep is taken. If trace B has points near the bottom of screen, overflow or poor precision can occur. If a division by zero is attempted, a division by the smallest representable value will be performed instead.

To ensure that correct data is displayed, be sure that a sweep is taken before using this command. Only one trace math operation may be active at any given time.

Query Response
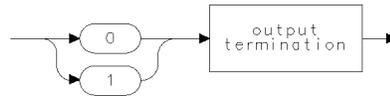
qgen0

See Also ADBTL

ADBTL

Performs trace math A * DL / B ▶ A with each sweep.
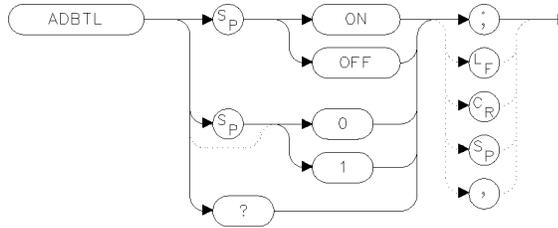
xadbtl

Item	Description
Preset State	ADBTL OFF

Example

120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"LN;"	*Select linear amplitude scale.*
140	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and store it.*
150	OUTPUT 723;"ADBTL ON;"	

Description The ADBTL command requires a linear display mode. It multiplies trace A by the display line value then divides by trace B. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep.

This command allows ratio measurements to be made as a sweep is taken. If trace B has points near the bottom of screen, overflow or poor precision can occur. If a division by zero is attempted, a division by the smallest representable value will be performed instead.

To ensure that correct data is displayed, be sure that a sweep is taken before using this command. Only one trace math operation may be active at any given time.

Query Response
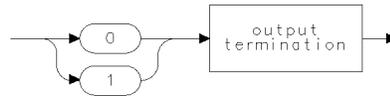
qgen0

ADCTL

Performs trace math A * DL / C ▶ A with each sweep.
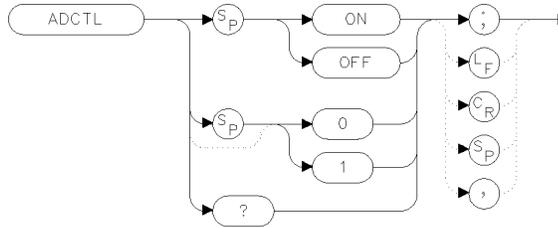
xadctl

Item	Description
Preset State	`ADCTL OFF`

Example

```
120   OUTPUT 723;"SNGLS;"              Activate single-sweep mode.
130   OUTPUT 723;"LN;"                 Select linear amplitude scale.
140   OUTPUT 723;"TS;VIEW TRA;"        Sweep trace A and store it.
150   OUTPUT 723;"ADCTL ON;"
```

Description The ADCTL command requires a linear display mode. It multiplies trace A by the display line value, then divides by trace C. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep.

This command allows ratio measurements to be made as a sweep is taken. If trace B has points near the bottom of screen, overflow or poor precision can occur. If a division by zero is attempted, a division by the smallest representable value will be performed instead.

To ensure that correct data is displayed, be sure that a sweep is taken before using this command. Only one trace math operation may be active at any given time.

Query Response
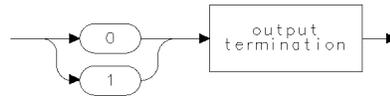
qgen0

ADCTRG

Selects the source of the ADC trigger.

xadct
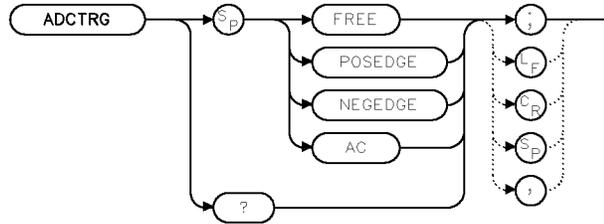
Item	Description
Preset State	ADCTRG FREE

Example   OUTPUT 723;"ADCTRG POSEDGE;"

Description The ADCTRG command selects the type of external triggering used by the optical spectrum analyzer's analog-to-digital converter. The TTL-compatible trigger signal from the light source is connected to the rear-panel **EXT TRIG IN** connector. The following types of triggering can be selected:

Softkey	Type of Triggering
ADCTRIG FREE	External triggering is ignored. The ADC is triggered during sweeps by the optical spectrum analyzer's internal clock.
ADCTRIG POSEDGE	ADC is triggered by the rising edge of the external trigger signal.
ADCTRIG NEGEDGE	ADC is triggered by the falling edge of the external trigger signal.
ADCTRIG AC	ADC is alternately triggered by rising and filling edges of the external trigger signal. The amplitudes measured on opposite edges are subtracted and the absolute value of the result becomes the trace amplitude value.

For external ADC triggering, connect the signal to the rear-panel **EXT TRIG IN** connector. To delay external ADC triggering, refer to "ADCTRGDLY".

External trigger frequency can affect measurement accuracy

When using external ADC triggering in non-zero spans, the frequency of the external trigger signal can affect measurement accuracy. The trigger signal's frequency must be greater than

$$frequency > \frac{1.5\,(trace\ length)}{sweep\ time}.$$

In some cases, a factor as high as 2.0 may be required instead of 1.5. If the ADC trigger rate is too low for the current combination of trace length and sweep time, the warning message **16005 Sweep too fast** is displayed. This warning is reported once for every sweep where the external trigger doesn't occur often enough.

Avoid oversweeping the video bandwidth

When using external ADC triggering, displayed video bandwidths are not accurate and care must be taken not to over sweep the actual video bandwidth. Use the following equation to estimate the actual video bandwidth:

$$video\ bandwidth \approx \frac{(factor)(displayed\ bw)(external\ trigger\ frequency)}{27,000}$$

where:

factor is a function of the external trigger frequency. *factor* is always less than 1.0 and may be as low as 0.5.

To obtain the proper sweep time, estimate the actual video bandwidth (as described above), and then set the optical spectrum analyzer's bandwidth to that setting. Then, use a computer to query the coupled sweep time. Refer to the **ST** command in this manual. The value returned to the computer is the proper sweep time setting.

Query Response
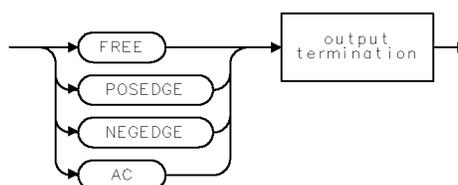
qadct

See Also   ADCTRGDLY, ADCTRGSYN

ADCTRGDLY Delays ADC triggering.
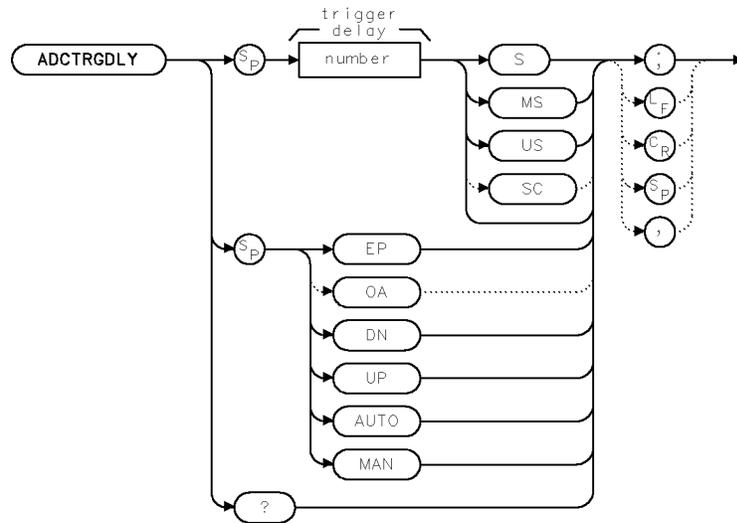
xadctd

Item	Description
Preset State	Zero seconds of delay.

Example   OUTPUT 723;"ADCTRGDLY;"

Description The ADCTRGDLY command delays external triggering of the analog-to-digital converter after the selected edge of the trigger signal.

Query Response
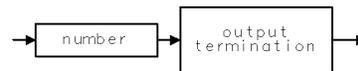
qadctd

See Also ADCTRG

ADCTRGSYN   Sets status of `ADC SYNC OUT` during active (non-free) ADC trigger modes.
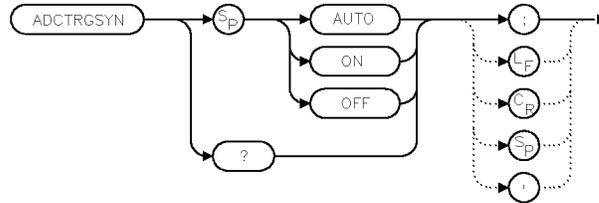
XADCTRG

Item	Description
Preset State	Auto.

Example   `OUTPUT 723;"ADCTRGSYN ON;"`

Description   When `AUTO` is specified, the rear-panel `ADC SYNC OUT` signal is driven high when ADC trigger occurs and is driven low when the ADC trigger delay has expired. When `ON` is specified, the rear-panel `ADC SYNC OUT` signal is driven to a constant high state. When `OFF` is specified, the rear-panel `ADC SYNC OUT` signal is driven to a constant low state.

Query Response
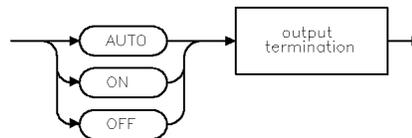
QADCTRG

See Also   ADCTRG

ADD

Add sources 1 and 2, point by point, then send their sum to the destination.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	± 1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Destination-Length Range	1 – 2048 position units.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.

Example   Add 25 nm to the center wavelength, then store their sum in a user-defined variable.

```
140  OUTPUT 723;"SP 200NM;CENTERWL 750NM;TS;"
```
Set measurement range and sweep it.
```
150  OUTPUT 723;"VARDEF I,0;"
```
Define user-defined variable I equal to 0.
```
160  OUTPUT 723;"ADD I,CENTERWL,25 NM;"
```
Add 25 nm to the center wavelength, then place the sum in I.

Description   The ADD command performs a point-by-point addition of the source variable or trace, then stores the results in the destination variable or trace. When the source length is longer than the destination length, excess data is ignored or truncated. When the source length is shorter than the destination length, the last point is duplicated to fill the destination.

When using TRA, TRB, TRC, or trace range for source, use SNGLS and TS to be sure data reflects one complete sweep. TRDEF required when using user-defined trace. VARDEF required when using user-defined variable.

ALIGN

Adjusts the fiber positioner's tracking table.
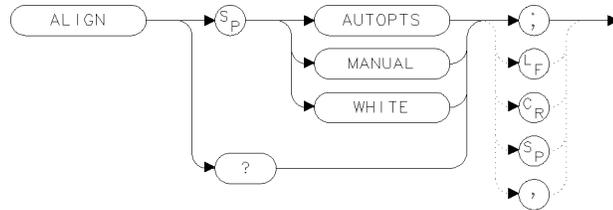
xalign .

Example   `100  OUTPUT 723;"ALIGN WHITE;"`

Description The ALIGN command adjusts the fiber positioner's tracking table at multiple wavelengths. (The fiber positioner aligns the monochromator's output beam with an internal optical-fiber cable.) The alignment uses either the Option 002 built-in white light source or a user-supplied source.

The command replaces pre-existing tracking data with new data and zeroes the (AUTO ALIGN) data. The resulting tracking data may be viewed via the extended state display: press State , MORE 1 of 3 , MORE 2 of 3 , show states , and then EXTEND STATE .

If an AUTO ALIGN is executed after this alignment, all of the alignment tracking data will be shifted. In this sense, AUTO ALIGN is done on top of this alignment.

The following command arguments are available:

Argument	Definition
WHITE	Specifies the Option 002 built-in white light source. The alignment determines tracking corrections at 17 equally spaced wavelengths between 900 nm and 1700 nm. Current tracking data is replaced with new data, and the AUTO ALIGN data is zeroed. The alignment usually requires 2 to 3 minutes to execute. Be sure to connect the white light source output to the HP 71450A's OPTICAL INPUT (or HP

71451A's MONOCHROMATOR INPUT) before issuing the ALIGN command.

AUTOPT   Specifies that an external broad-band light source is used. You must set the stop and start wavelengths prior to the alignment. These wavelengths are included in the alignment. The alignment takes about 2 or 3 minutes to complete. To run this alignment, you must observe the following points:

- Connect your own broadband light source to the front-panel input connector.
- Set the start and stop wavelengths within the input range of 600 nm to 1700 nm.
- Ensure that the span of the alignment is greater than 10 nm.
- The signal amplitude must be greater than −78 dBm in a 10 nm resolution bandwidth at all X's to be aligned.

If the above conditions are not met, a "Bad Wavelength Range" error message is displayed. The alignment occurs at equally spaced points between the start and stop wavelengths. No points are separated by more than 50 nm.

MANUAL   Runs an (AUTO ALIGN) at a wavelength apecified by the marker and transfers the results to the fiber positioner's tracking table. (A maximum of 32 alignments can be performed.) Before initiating the alignment, connect your own light source to the front-panel input connector. After the alignment, the AUTO ALIGN data is zeroed. The data being added to the tracking table is tested to see if it results in a tracking curve which is too steep. If the curve is too steep, an error is reported, and the data is not added to the table (neither is the AUTO ALIGN data zeroed). If

the current AUTO ALIGN motor position is close enough to a motor position already listed in the correction table, the AUTO ALIGN data will replace that entry instead of generating a new one. This "replacement" limit is wavelength and order dependent, but amounts to about 10 nm at 1300 nm in first order.

In the rare case where the `AUTO ALIGN` yields zero X and Y tracking corrections, the command will fail, indicating that no `AUTO ALIGN` data exists.

The following errors can be reported as a result of issuing the ALIGN command:

- 7055 Cal Failed (0B) Trajectory Table Overflow: This error indicates an attempt to add a point to an already-full trajectory tracking table. Neither the table or the current autoalign data are altered. Trajectory tracking table data may be viewed via the extended state display (`EXTEND STATE` softkey).

- 7055 Cal Failed (2B) Invalid Trajectory Corrections: The data being added to the correction table would cause the correction curve to become too steep. Neither the table or the current `AUTO ALIGN` data are altered. This can happen when executing the function after a long period of time has elapsed since the last ALIGN, of after the instrument has received a substantial mechanical shock. It can usually be remedied by executing the ALIGN AUTOPTS function.

- Bad xxx ROM: The data stored in one or both calibration ROMs is invalid. This situation must be remedied before ALIGN will function.

Query Response
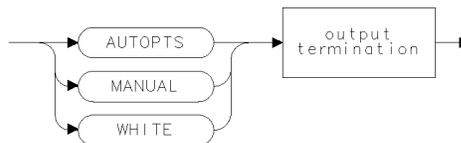
qalign

See Also ALIGNPRST

ALIGNPRST

Reset the output-fiber alignment of the monochromator.
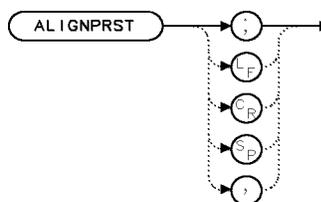
XALIG

Example   `10  OUTPUT 723;"ALIGNPRST;"`

Description The ALIGNPRST command resets the optical spectrum analyzer monochromator output-fiber alignment. The monochromator's Trajectory tracking table is set to factory calibration values.

See Also AUTOALIGN

AMB

Performs trace math A − B ▶ A with each sweep.
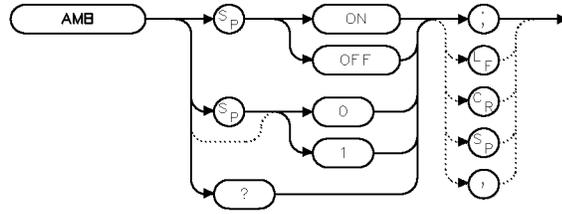
XAMB

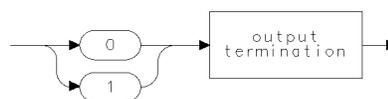| Item	Description
Preset State	AMB OFF

Example

```
30  OUTPUT 723;"SNGLS;"          Activate single-sweep mode.
40  OUTPUT 723;"LG 5DB;"         Select logarithmic scale.
50  OUTPUT 723;"TS;VIEW TRA;"    Measure with trace A and store results.
60  OUTPUT 723;"CLRW TRB;TS;"    Measure with trace B.
70  OUTPUT 723;"AMB ON;"         Subtract trace B from trace A.
```

Description The AMB command subtracts trace B from trace A, point by point, then places the results in trace A. If trace A is in clear-write mode and **AMB ON** is active, the AMB function is continuous and occurs every sweep. Executing **AMB ON** places trace B in view mode. The AMBPL command overrides AMB. Be sure each trace contains a complete sweep of measurement information.

Query Response
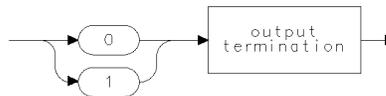
qgen0

See Also TS, TRPST, TRSTAT, VIEW, TM, VTL, VTH, CONTS, SNGLS, CLRW, STORREF

AMBMC

Performs trace math (A - B) - C ▶ A with each sweep.
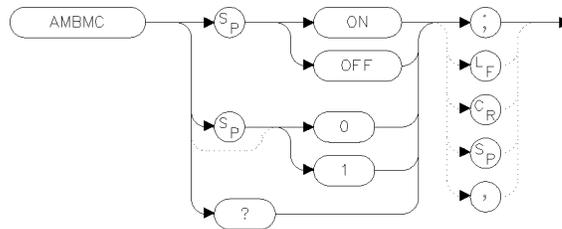
x ambmc

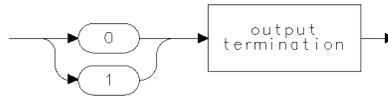| Item	Description
Preset State	AMBMC OFF

Example

```
120   OUTPUT 723;"SNGLS;"              Activate single-sweep mode.
130   OUTPUT 723;"LG 5DB;"             Select logarithmic scale.
140   OUTPUT 723;"TS;VIEW TRA;"        Sweep trace A and store it.
150   OUTPUT 723;"AMBMC ON;"
```

Description The AMBMC command subtracts both trace B and trace C from trace A. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep. Only one trace math operation may be active at any given time.

Query Response
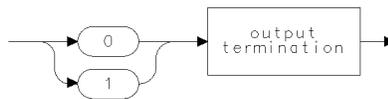
qgen0

AMBMCPL

Performs trace math (A - B) - C+DL ▶ A with each sweep.
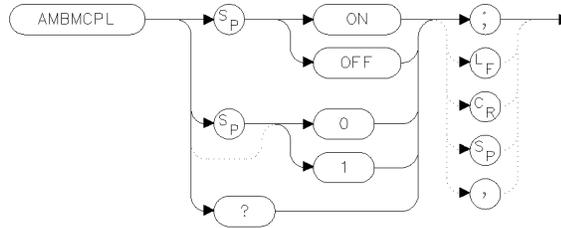
xabmpl

Item	Description
Preset State	AMBMCPL OFF

Example

```
120   OUTPUT 723;"SNGLS;"          Activate single-sweep mode.
130   OUTPUT 723;"LG 5DB;"         Select logarithmic scale.
140   OUTPUT 723;"TS;VIEW TRA;"    Sweep trace A and store it.
150   OUTPUT 723;"AMBMCPL ON;"
```

Description The AMBMCPL command subtracts both trace B and trace C from trace A, and then adds the display line. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep. Only one trace math operation may be active at any given time.

Query Response
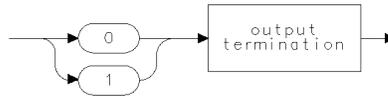
qgen0

AMBPL

Performs trace math A − B + DL ▶ A with each sweep.
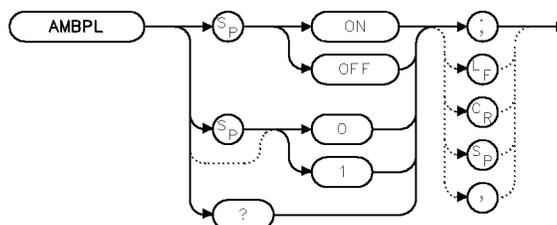
x ambp l

Item	Description
Preset State	AMBPL OFF

Example

```
40  OUTPUT 723;"SNGLS;"           Activate single-sweep mode.
50  OUTPUT 723;"LG 5DB;"          Select logarithmic scale.
60  OUTPUT 723;"TS;VIEW TRA;"     Measure with trace A and store results.
70  OUTPUT 723;"RL -10DBM;"       Change reference level to 10 dBm.
80  OUTPUT 723;"CLRW TRB;TS;"     Measure with trace B
90  OUTPUT 723;"DL -10DBM;"       Set display line to −10 dBm.
100 OUTPUT 723;"AMBPL ON;"        Subtract trace B from trace A, then add
                                  display line.
```

Description The AMBPL command subtracts trace B from trace A, point by point, adds the display line to the difference, then stores the result in trace A. When AMBPL is executed, trace B is placed in view mode and cannot be swept. If trace A is in clear-write mode when AMBPL is executed, the AMBPL function is continuous and occurs every sweep. The AMBPL command overrides the AMB command. Be sure each trace contains a complete sweep of measurement information.

This function is effective when wanting to place the results of trace subtraction at a specific location. Placing the optical spectrum analyzer in single sweep mode allows you to view the results of the trace math. Trace math is mainly used to normalize the display during stimulus-response measurements.

Query Response
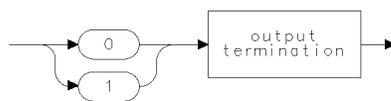
qgen0

See Also   AMB, TS, TRPST, TRSTAT, VIEW, TM, VTL, VTH, CONTS, SNGLS, CLRW, STORREF

AMC

Performs trace math A − C ▶ A with each sweep.
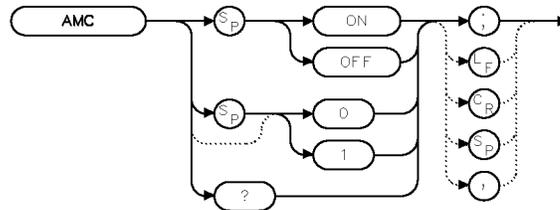
XAMC

Item	Description
Preset State	AMC OFF

Example

```
30  OUTPUT 723;"SNGLS;"          Activate single-sweep mode.
40  OUTPUT 723;"LG 5DB;"         Select logarithmic scale.
50  OUTPUT 723;"TS;VIEW TRA;"    Measure with trace A and store results.
60  OUTPUT 723;"CLRW TRC;TS;"    Measure with trace C.
70  OUTPUT 723;"AMC ON;"         Subtract trace C from trace A.
```

Description The AMC command subtracts trace C from trace A, point by point, then places the results in trace A. If trace A is in clear-write mode and `AMC ON` is active, the AMC function is continuous and occurs every sweep. Executing `AMC ON` places trace C in view mode. Be sure each trace contains a complete sweep of measurement information.

Query Response

qgen0

See Also TS, TRPST, TRSTAT, VIEW, TM, VTL, VTH, CONTS, SNGLS, CLRW, STORREF

AMCPL

Performs trace math (A - C)+DL ▶ A with each sweep.
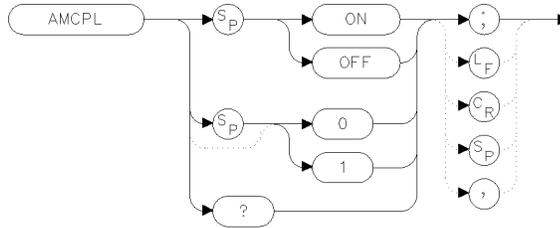
xamcpl

Item	Description
Preset State	AMCPL OFF

Example

120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"LG 5DB;"	*Select logarithmic scale.*
140	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and store it.*
150	OUTPUT 723;"AMCPL ON;"	

Description

The AMCPL command subtracts trace C from trace A, and then adds the display line. The results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep. Only one trace math operation may be active at any given time.

Query Response

qgen0

AMETER

Enable readout of currently updated trace point.
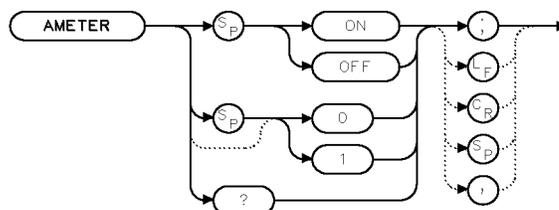
XAMTR

Item	Description
Preset State	AMETER OFF

Example

```
70  OUTPUT 723;"AMETER ON;"
```

Description The AMETER command enables a readout of the amplitude of the blank-ahead marker in the display's message area. The blank-ahead marker is the point on the trace which is currently being updated by new measurement data.

Query Response
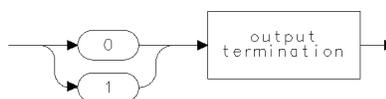
qgen0

AMPCOR

Apply amplitude corrections at specified wavelengths.
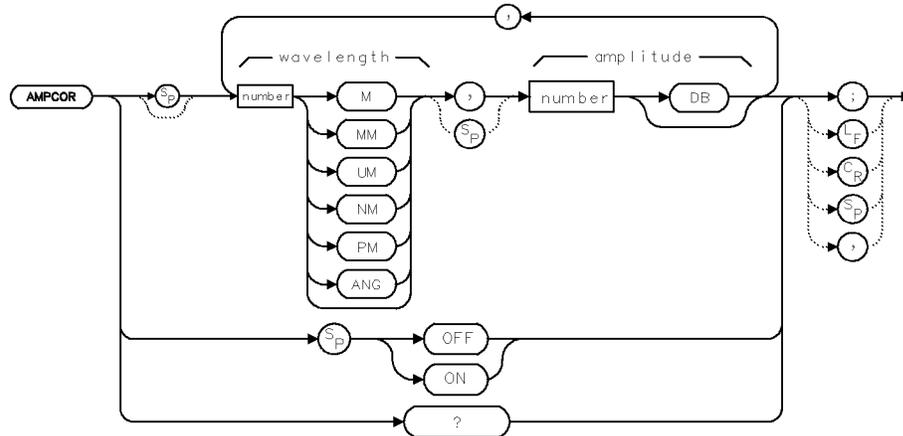
ckb3

Item	Description
Wavelength Range	600 – 1700 nm
Amplitude Range	full measurement range
Number of wavelength/amplitude pairs	Depends on available internal memory.
Preset State	AMPCOR OFF

Example Compensate for wavelength-dependent amplitude inaccuracies at the input.

```
150  OUTPUT 723;"CENTERWL 950NM;SP 200NM;"        Set center wavelength and span.

160  OUTPUT 723;"AMPCOR 750NM,-5DB,950NM,-10DB,1150NM,-15DB;"
```
Store wavelength-amplitude pairs in analyzer. Notice that wavelengths are in ascending order.

```
170 !
180 OUTPUT 723;"AMPCOR?;"                 Query correction values to computer.
190 FOR I=1 TO 3                          Store first three pairs in computer array
                                          A$[I].
200 ENTER 723;A$[I]
210 PRINT A$[I]
220 NEXT I
```

```
230 DISP "PRESS CONTINUE"
240 PAUSE                             Press CONTINUE on computer keyboard to
                                      continue program.
250 DISP ""
260  OUTPUT 723;"AMPCOR OFF;"         Turn off AMPCOR function.
270  OUTPUT 723;"AMPCOR?;"            Query correction values to computer. Val-
                                      ues are zeros, because AMPCOR is turned
                                      off.
280 ENTER 723;A$[1]
290 PRINT A$[1]
```

Description Use AMPCOR to compensate for wavelength-dependent amplitude variations at the input port. Pairs of wavelength-amplitude correction points can be entered, depending on the availability of adequate internal memory. The wavelength values entered must be equal or be in increasing order, or an error condition results. Whenever AMPCOR is on, the correction values are added to all measurement results.

The values of the correction points are applied across the active measurement range. Between points, the correction values are interpolated. When measuring at wavelengths outside the first and last correction points, the first or last value (as appropriate) is used as the correction value.

When AMPCOR is executed, the wavelength-amplitude pairs are stored in internal memory. **DISPOSE ALL**, FORMAT, and **PERASE** erase the amplitude-correction pairs from memory.

Query Response Queried values are 0, 0 when AMPCOR is set to OFF.
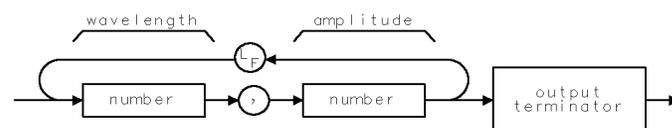
See Also PERASE, ERASE, DISPOSE, PROTECT, LIMTOCOR

AMPMKR   Position the left and right power integration points to the value specified on either side of the peak.
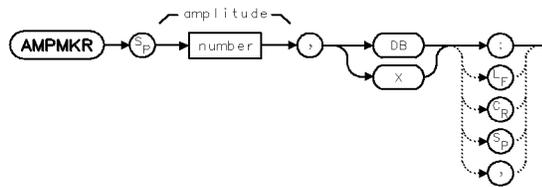
ckb15

Example   110   OUTPUT 723;"AMPMKR −10,DB;"   *Set the left and right integration points down 10 dB from the peak.*

120   OUTPUT 723;"AMPMKR .1,X;"   *Set the left and right integration points down 10% (or −10 dB) from the peak.*

Description   The AMPKMKR command positions the power integration points down from the peak according to the entered amplitude. The left point is specified as marker number 4 and the right point is specified as marker number 5.

See Also   WLMKRR, WLMKRL

AMPU

Convert measurement-unit values to amplitude-unit values.
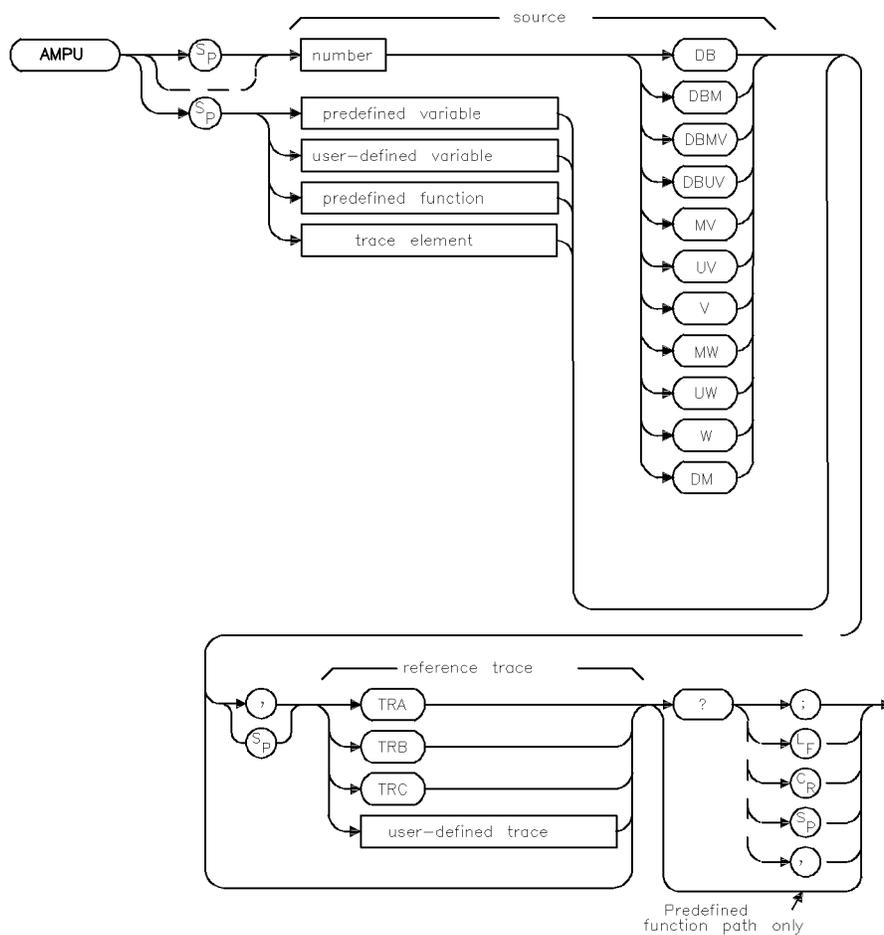
ckb4

Item	Description
Source Amplitude Range	For *trace element*, 0 – 10,000 measurement units for linear trace data, and −32,768 – +32,767 measurement units for logarithmic trace data.
	$\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048
Reference-Trace Amplitude Range	0 – 10,000 measurement units for linear trace data.
	−32,768 – +32,767 measurement units for logarithmic trace data.
Reference-Trace Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example Display trace-amplitude information in both measurement and amplitude units on the computer display.

160	OUTPUT 723;"VARDEF NN,0;"	*Define user-defined variable, NN equal to 0.*
170	OUTPUT 723;"MOV NN,TRA[15];"	*Place the value of the fifteenth element of trace A in NN.*
180	OUTPUT 723;"NN?;"	*Query value of NN in measurement units.*
190	ENTER 723;N	*Store value in computer variable, N.*
200	OUTPUT 723;"AMPU NN,TRA?;"	*Query value of NN in amplitude units.*
210	ENTER 723;M	*Store value in computer variable, M.*
220	OUTPUT 723;"AUNITS?;"	*Query description of active units.*
230	ENTER 723;Units$	*Store response in computer variable, Units $.*
240	PRINT N;"MEASUREMENT UNITS"	*Print value in measurement units.*
250	PRINT M;Units$	*Print value in amplitude units.*

Description Measurement units describe trace-amplitude information as follows. For logarithmic trace information, a measurement unit is one hundredth of a dBm.

$$value\ in\ measurement\ units\ =\ 100\ (value\ in\ dBm)$$

For linear trace information, a measurement unit is defined as 0 (zero) for the bottom of the graticule, and as 10,000 for the top (reference level). Amplitude units describe trace-amplitude information in Decibels (dBm) or Watts (W, mW, μW). If the *reference trace* is specified, AMPU converts to the amplitude values according to the amplitude scale of the reference trace. Otherwise, AMPU converts to units determined by the reference level, AUNITS command, or MEASURE command.

AMPU must be used as either a query or as a source in another analyzer command. Form a query by ending the AMPU statement with a question

mark (?). When used as a query, AMPU sends the amplitude-unit value of the source on the bus to the computer.

Use AMPU as a source by incorporating the AMPU statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When AMPU is used as a source, the amplitude-unit value of the source is used for operation by the command containing AMPU.

Query Syntax

```
                                        ┌──────── source ────────┐
  ( AMPU )──→(Sp)──→ number ────────────────────────→( DB )
           │                                          ( DBM )
           └──(Sp)──→ predefined variable             ( DBMV )
                   ──→ user-defined variable          ( DBUV )
                   ──→ predefined function            ( MV )
                   ──→ trace element                  ( UV )
                                                      ( V )
                                                      ( MW )
                                                      ( UW )
                                                      ( W )
                                                      ( DM )

              ┌──────── reference trace ────────┐
  (,)───→( TRA )                          ( ? )──→(;)
  (Sp)──→( TRB )                                  (LF)
     ──→( TRC )                                   (CR)
     ──→ user-defined trace                       (Sp)
                                                  (,)
```

Prede-
fined-Function
Syntax

```
                    ┌─────────────┐
                    │ compatible  │
                    │  function   │
                    └─────────────┘

                                          ──── source ────
        ┌──────┐    ┌─┐    ┌──────────┐              ┌────┐
        │ AMPU │────│S│────│  number  │──────────────│ DB │
        └──────┘    │P│    └──────────┘              └────┘
                    └─┘                              ┌─────┐
                    ┌─┐    ┌────────────────────┐    │ DBM │
                    │S│────│ predefined variable │    └─────┘
                    │P│    └────────────────────┘    ┌──────┐
                    └─┘    ┌──────────────────────┐  │ DBMV │
                          │ user-defined variable │   └──────┘
                          └──────────────────────┘   ┌──────┐
                          ┌────────────────────┐     │ DBUV │
                          │ predefined function │     └──────┘
                          └────────────────────┘     ┌────┐
                          ┌────────────────┐          │ MV │
                          │ trace element  │          └────┘
                          └────────────────┘          ┌────┐
                                                       │ UV │
                                                       └────┘
                                                       ┌───┐
                                                       │ V │
                                                       └───┘
                                                       ┌────┐
                                                       │ MW │
                                                       └────┘
                                                       ┌────┐
                                                       │ UW │
                                                       └────┘
                                                       ┌───┐
                                                       │ W │
                                                       └───┘
                                                       ┌────┐
                                                       │ DM │
                                                       └────┘

                              ──── reference trace ────
                 ┌─┐    ┌─────┐
                 │,│────│ TRA │
                 └─┘    └─────┘
                 ┌─┐    ┌─────┐
                 │S│────│ TRB │
                 │P│    └─────┘
                 └─┘    ┌─────┐
                        │ TRC │
                        └─────┘
                        ┌───────────────────┐
                        │ user-defined trace │
                        └───────────────────┘
```

Query Response

```
        ┌──────────┐    ┌─────────────┐
        │  number  │────│   output    │
        └──────────┘    │ termination │
                        └─────────────┘
```

qgennum

See Also AUNITS, RL, MEASURE, ROFFSET, INZ, LN, LG

ANNOFF

Turn off all display annotation.
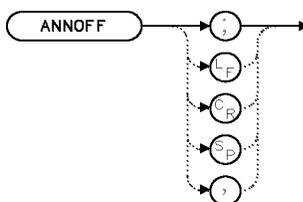
xannof

Description The ANNOFF command turns off all annotation on the display screen similar to the ANNOT OFF command. The difference is that, in order to reenable the display annotation, either execute the IP command or press the INSTRUMENT PRESET key on the front panel. This command provides measurement security by blanking all display annotation for the current instrument setup and then requiring an instrument preset to turn the annotation back on.

See Also ANNOT OFF, IP

ANNOT

Turn annotation on or off.
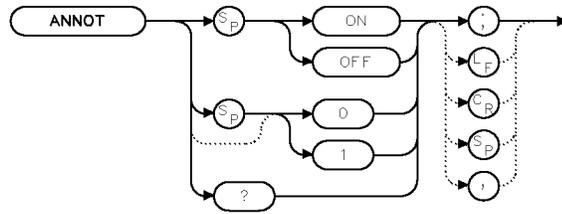
XANNOT

Item	Description
Preset State	ANNOT ON

Example   110 OUTPUT 723;"ANNOT OFF;"   *Turn off the annotation.*

Description The ANNOT command turns off the annotation on the analyzer display. When the annotation is turned off, the time between sweeps decreases. The annotation command also blanks text displayed with the TITLE command.

Query Response
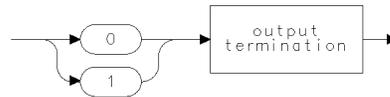
qgen0

See Also TITLE

APB   Add trace B and trace A, then enter sum in trace A. (A + B ▶ A)
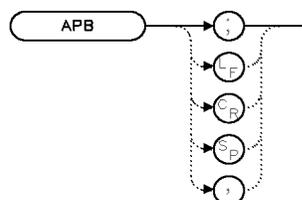
XAPB

Example

30	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
40	OUTPUT 723;"RL 10DBM;SP 200NM;CENTERWL 950NM;"	*Set measurement range.*
50	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and view results.*
60	OUTPUT 723;"CLRW TRB;TS;"	*Clear write trace B. Sweep trace B.*
70	OUTPUT 723;"AMB ON;"	*Subtract trace B from trace A.*

Description The APB command adds traces A and B, point by point, then sends the result
to trace A. Thus, APB can restore the original trace after A-minus-B function
(AMB) is executed. However, unlike AMB, the math operation is done only
once, making `APB` equivalent to executing `ADD TRA,TRA,TRB`. Be sure data
reflects one complete sweep for trace A and trace B by using SNGLS and TS.

APBDCTL

Performs trace math DL * (A+B) / C ▶ A with each sweep.
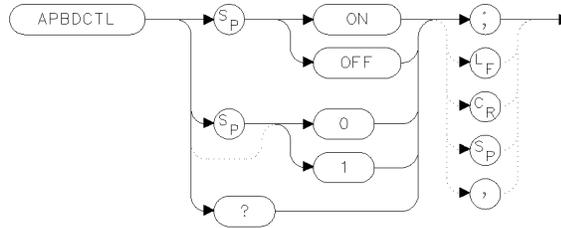
Item	Description
Preset State	APBDCTL OFF

Example

```
120  OUTPUT 723;"SNGLS;"          Activate single-sweep mode.
130  OUTPUT 723;"LN;"             Select linear amplitude scale.
140  OUTPUT 723;"TS;VIEW TRA;"    Sweep trace A and store it.
150  OUTPUT 723;"APBDCTL;"
```

Description The APBDCTL command requires a linear display mode. It multiplies the sum of traces A and B by the display line. Then, this result is divided by trace C. The final results are stored in trace A. If trace A is in clear-write mode, the function is continuous and occurs every sweep.

This command allows ratio measurements to be made as a sweep is taken. If trace B has points near the bottom of screen, overflow or poor precision can occur. If a division by zero is attempted, a division by the smallest representable value will be performed instead.

To ensure that correct data is displayed, be sure that a sweep is taken before using this command. Only one trace math operation may be active at any given time.

Query Response
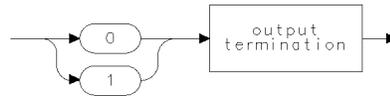
qgen0

AUNITS

Select amplitude units or annotation and query data.
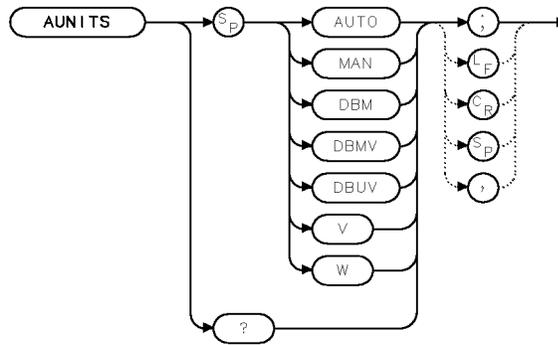
Item	Description
Preset State	AUNITS DBM

Example Change the units of the amplitude readouts.

    160   OUTPUT 723;"AUNITS AUTO;"   *Indicate amplitude units in dBm.*

Description The AUNITS command changes the units of the amplitude readouts displayed on the analyzer display or queried to the computer. The AUNITS command also determines the amplitude units for analyzer commands where the default-units path is specified. AUNITS does not affect trace amplitude information queried to the computer with the TRA, TRB, or TRC commands.

Query Response
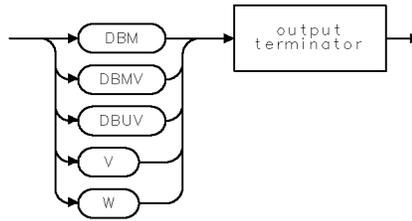
qaunit

AUTOALIGN

Automatically realign the monochromator output fiber.
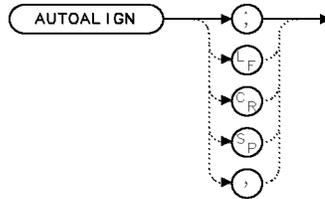
XAUTOA

Example   `10  OUTPUT 723;"AUTOALIGN;"`

Description The AUTOALIGN command realigns the optical spectrum analyzer monochromator output fiber with the output beam. If there is a marker on screen, the alignment is done at the marker's wavelength. If there is no marker, the alignment is done at the wavelength of the largest signal detected in full span.

The auto-alignment routine is performed in a 10 nm resolution bandwidth setting. At the end of the auto-alignment routine, the instrument returns to its previous state.

See Also ALIGN, ALIGNPRST

AUTOMDB

Set vertical scale after the automatic measurement procedure.
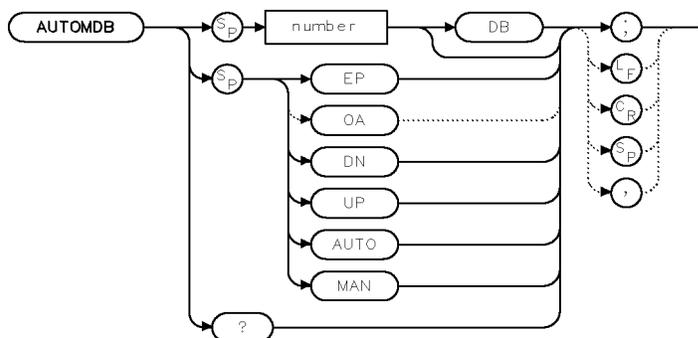
XAUTDB

Item	Description
Vertical Scale Limits	0.01 dB to 20 dB

Example

```
10  OUTPUT 723;"AUTOMDB 10DB;"
```

Description The AUTOMDB command sets the vertical scale (decibels-per-division) after performing the auto-measurement procedure. If AUTO is specified, the measurement procedure uses the vertical scale prior to initiating the measurement procedure. Specifying MAN causes the measurement procedure to use the vertical scale set by the AUTOMDB MAN command.

Query Response
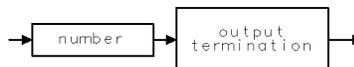
qgennum

See Also AUTOMEAS

7-61

AUTOMEAS

Automatically zoom in on largest signal at input.
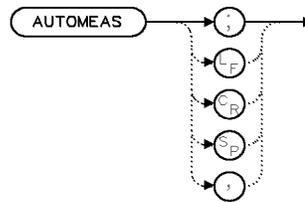
XAUTOM

Example   `10   OUTPUT 723;"AUTOMEAS;"`

Description The AUTOMEAS command causes the optical spectrum analyzer to automatically locate the largest input signal. When the command is executed, the optical spectrum analyzer resets to the full wavelength span. Then, the largest signal is located, and the span is reduced to observe this signal. If the AUTOMMKR command is set to ON, the automatic measurement routine finds the signal that is closest to the current marker.

Use the AUTOMSP command to set the final reduced span used by the automatic measurement routine. Use the AUTOMDB command to set the final vertical scale set by the automatic measurement routine. Use the AUTOMOPT command to cause the AUTOMEAS function to optimize the sensitivity.

An error is generated if a signal is not detected, and the instrument returns to its previous state.

See Also AUTOMDB, AUTOMMKR, AUTOMSP, AUTOMOPT

AUTOMMKR Locate the closest signal to the current marker.

XAUTMK

Example   10 OUTPUT 723;"AUTOMMKR ON;"

Description The AUTOMMKR command causes the automatic measurement routine
to locate the signal that is closest to the current marker. This alters the
routine's normal function of locating the largest input signal across the optical
spectrum analyzer's full wavelength range. When AUTOMMKR in ON, the
routine starts from the current wavelength span rather than the full span.

Query Response

qgen0

See Also AUTOMDB, AUTOMEAS, AUTOMSP

AUTOMOPT

Control sensitivity optimization during the automeasure routine.
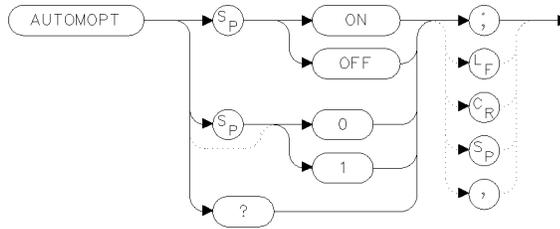
xamopt

Item	Description
Preset State	AUTOMOPT OFF

Example

```
100   OUTPUT 723;"AUTOMOPT ON;"
```

Description The AUTOMOPT command controls sensitivity optimization during the automatic measurement routine. When AUTOMOPT is on, the AUTOMEAS routine optimizes the sensitivity after locating the signal. This feature is useful for signals that have a large dynamic range such as DFB lasers. With AUTOMOPT off, the DFB laser's side modes would be hidden in the displayed noise.

Query Response
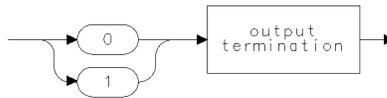
qgen0

See Also AUTOMEAS

AUTOMSP

Determine the final wavelength span set by the automatic measurement routine.
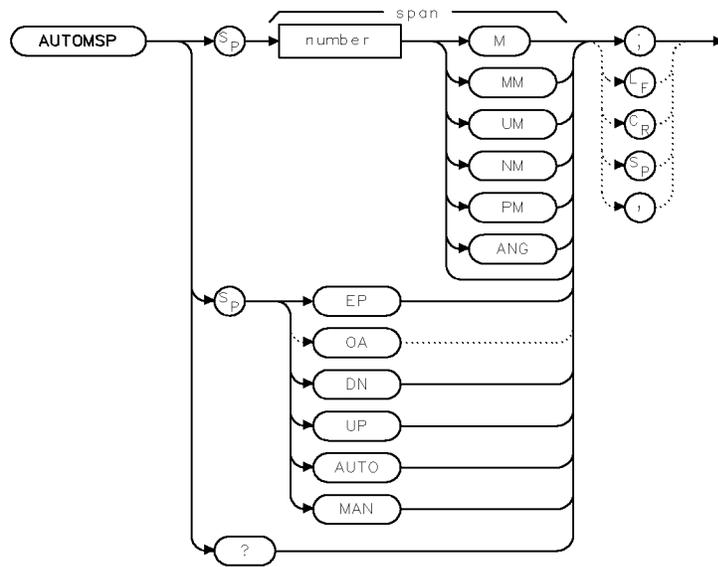
XAUTSP

Example

```
10  OUTPUT 723;"AUTOMSP 100NM;"
```

Description The AUTOMSP command sets the wavelength span for viewing the signal located by the AUTOMEAS function.

Query Response

qgennum

See Also AUTOMDB, AUTOMEAS, AUTOMMKR

AUTORNG

Determine whether automatic ranging is enabled or disabled.

XAUTOR

Example   10 OUTPUT 723;"AUTORNG ON;"   *Enable the automatic-ranging feature. When the signal is less powerful, select the most sensitive measuring range. When the signal is more powerful, select the less sensitive measuring range.*

Description   The AUTORNG command is used to turn on or off the automatic ranging feature. When the incoming signal strength is sufficient, a less sensitive measurement mode is needed. When the incoming signal strength is weak, a more sensitive measurement mode is needed.

AUTORNG ON (1) allows the analyzer to select the appropriate measurement sensitivity.

AUTORNG OFF (0) selects the least sensitive measurement mode as determined by the reference level and turns off the automatic ranging feature.

Query Response

qgen0

See Also   AUTOMDB, AUTOMEAS, AUTOMMKR, SENS, VB

AVG

Average the source and the destination, then store the result in the destination.

XAVG

Item	Description
Average-Ratio Range	−32,768 – +32,767, except 0
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range,* or *user-defined trace.*
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range,* or *user-defined trace.*
	± 1.8 × 10^{308} when using *number, predefined variable,* or *user-defined variable.*
Destination-Length Range	1 – 2048 position units.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC.* Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.

Example Measure with traces A and B. Average the measurement results in the traces, then view the averaged results in trace A.

130	OUTPUT 723;"IP;"	*Instrument preset.*
140	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
150	OUTPUT 723;"TS;"	*Take a sweep. Measure with trace A.*
160	OUTPUT 723;"VIEW TRA;"	*Store measurement results and view trace A.*
170	OUTPUT 723;"VB 100HZ;"	*Activate the video bandwidth filter.*
180	OUTPUT 723;"CLRW TRB;TS;"	*Measure with trace B*
190	OUTPUT 723;"VIEW TRB;"	*Store measurement results and view trace B.*
200	OUTPUT 723;"AVG TRA,TRB,2;"	*Average trace B with 2 and store results in trace A.*
210	OUTPUT 723;"BLANK TRB;"	*Blank trace B leaving only averaged results.*

Description The AVG command averages the source and the destination, then stores the result in the destination according to the following algorithm:

$$destination_{new} = \frac{((average\ ratio - 1)(destination_{old}) + source)}{average_{ratio}}$$

When the source is a trace, be sure it contains a complete sweep of measurement information by using SNGLS and TS commands.

AXB

Exchange contents of trace A and trace B.

XAXB

Example

120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"CENTERWL 950NM;SP 500NM;"	*Set measurement range.*
140	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and store it.*
150	OUTPUT 723;"FS;"	*Measure with full span.*
160	OUTPUT 723;"CLRW TRB;TS;"	*Clear-write trace B.*
170	OUTPUT 723;"AXB;"	*Exchange traces A and B.*

Description The AXB command exchanges the contents of traces A and B. If trace A and trace B are in clear-write mode, AXB puts them in view mode. If the traces are of different lengths, data is lost. Use TRDEF or TRPST to set the lengths equal. Be sure the traces contain a complete sweep of measurement information by using SNGLS and TS commands.

See Also TS, VIEW, CLRW, SNGLS

AXC

Exchange contents of trace A and trace C.

xaxc

Example

120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"CENTERWL 950NM;SP 500NM;"	*Set measurement range.*
140	OUTPUT 723;"TS;VIEW TRA;"	*Sweep trace A and store it.*
150	OUTPUT 723;"AXC;"	*Exchange traces A and C.*

Description The AXC command exchanges the contents of traces A and C. If trace A and trace C are in clear-write mode, AXC puts them in view mode. If the traces are of different lengths, data is lost. Use TRDEF or TRPST to set the lengths equal. Be sure the traces contain a complete sweep of measurement information by using SNGLS and TS commands.

See Also TS, VIEW, CLRW, SNGLS

BIT

Store the state of a specified bit in the destination.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	± 1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Destination-Length Range	1 – 2048 position units.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using TRA, TRB, TRC. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.

Example

160	OUTPUT 723;"VARDEF ERROR,0;"	*Define user-defined variable ERROR equal to 0.*
170	OUTPUT 723;"BIT ERROR,STB,5;"	*Store value of bit 5 in ERROR.*
180	OUTPUT 723;"ERROR?;"	*Query value of ERROR.*
190	ENTER 723;Error	*Assign value to computer variable, Error.*
200	IF NOT Error THEN DISP NO	*If Error equals zero, display "NO" on computer display.*
210	DISP "ERROR PRESENT"	*Display "ERROR PRESENT" on computer display.*

Description The BIT command stores the state (0 or 1) of a specified bit in the destination. When using TRA, TRB, TRC, or trace range for source, use SNGLS and TS to be sure data reflects one complete sweep. TRDEF required when using user-defined trace. VARDEF required when using user-defined variable.

BLANK

Blank and store display of trace.
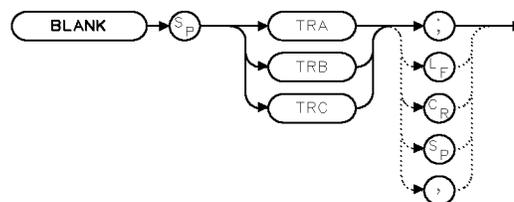
XBLANK

Example   110 OUTPUT 723;"BLANK TRA;"   *Blank trace A.*

Description The BLANK command stops sweeping (updating) the specified trace and deletes it from the display. Any data contained in the trace is stored in **reserved memory**.

See Also TRPST,TRSTAT

BML

Subtract display line value from trace B. (B − DL ▶ B)

XBML

Example

110	OUTPUT 723;"BLANK TRA;"	*Blank trace A.*
120	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
130	OUTPUT 723;"CENTERWL 750NM;SP 100NM;"	*Set measurement range.*
140	OUTPUT 723;"DL -10DBM;RL 0DBM;"	*Set display line and reference level.*
150	OUTPUT 723;"CLRW TRB;TS;"	*Sweep trace B.*
160	PAUSE	*Press CONTINUE on computer keyboard.*
170	OUTPUT 723;"BML;"	*Subtract display line from trace B.*

Description The BML command subtracts the display line from trace B, then stores the difference in trace B and places trace B in view mode. Use SNGLS and TS to be sure trace B contains a complete sweep of measurement information.

See Also TRSTAT, TRPST, VIEW TRDSP, CLRW

BP

Send the "beep" command to the display.
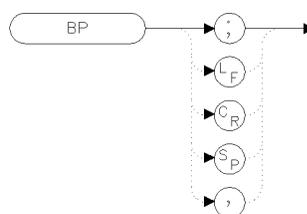
xbp

Example   `100  OUTPUT 723;"BP;"`

Description The BP command sends the `BP;` command to the display causing a single audible beep. The length and pitch of the tone can not be controlled. If there is no display link, neither a beep or error message occurs.

BTC

Store a copy of trace B in trace C, then store and display trace C contents.
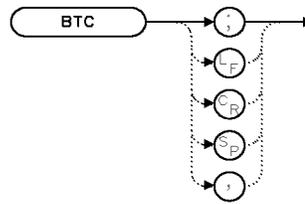
```
   ┌─────────┐      ╭───╮
───┤   BTC   ├──────┤ ; ├──────────►
   └─────────┘   ╎  ╰───╯  ╎
                 ╎  ╭───╮  ╎
                 ╎  │ L │  ╎
                 ╎  │ F │  ╎
                 ╎  ╰───╯  ╎
                 ╎  ╭───╮  ╎
                 ╎  │ C │  ╎
                 ╎  │ R │  ╎
                 ╎  ╰───╯  ╎
                 ╎  ╭───╮  ╎
                 ╎  │ S │  ╎
                 ╎  │ P │  ╎
                 ╎  ╰───╯  ╎
                 ╎  ╭───╮  ╎
                 ╰──┤ , ├──╯
                    ╰───╯
                                          XBTC
```

Example

| 120 | OUTPUT 723;"SNGLS;CLRW TRB;TS;" | *Sweep trace B.* |
| 130 | OUTPUT 723;"BTC;" | *Transfer trace B to trace C.* |

Description The BTC command transfers the contents of trace B to trace C, then stops updating trace C by placing it in the view mode. Be sure trace B contains a complete sweep of measurement information. To retain all data, the length of traces A and B must be equal. Use TRDEF to change trace length, if necessary.

See Also TRSTAT, TRPST, VIEW, TRDSP, CLRW

BXC   Exchange contents of trace B and trace C.
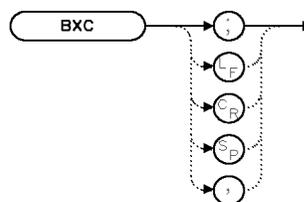
XBXC

Example   190 OUTPUT 723;"BXC;"   *Exchange traces B and C.*

Description The BXC command exchanges the contents of traces B and C, then stops
             updating them by placing them in the view mode. Be sure each trace
             contains a complete sweep of measurement information before executing
             BXC. To retain all data, the size of the traces must be equal. Use TRDEF to
             change trace length if necessary.

See Also VIEW, TRSTAT, TRPST, CLRW, TRDSP

CAL

Execute a calibration routine.
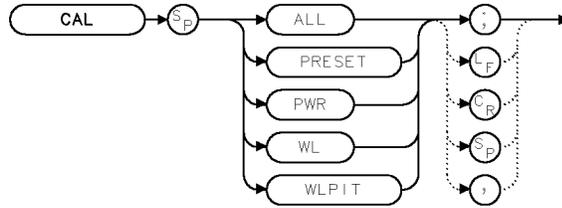
Item	Description
Preset State	CAL ALL

Example   `10  OUTPUT 723;"CAL WL;"`

Description The CAL command performs error-correction routines which calculate and apply correction factors for wavelength or power. An external calibration signal must be connected to the optical spectrum analyzer before executing CAL. If a signal is not connected, an error condition results. Use the CALCOR command to deactivate the correction factors.

The command arguments have the following definitions:

Argument	Definition
CAL ALL	Calibrates for both wavelength and power.
CAL PRESET	Erases all user calibration data.
CAL PWR	Calibrates at the power level specified by the CALPWR command at the entered calibration wavelength. The maximum adjustment range is limited to ±5 dB. The CAL PWR command also verifies that the calibration signal is within ±1 to 2 nm of the entered calibration wavelength.
CAL WL	Calibrates at the wavelength specified by the CALWL command. The maximum adjustment range is limited to about 1 to 2 nm. The CAL WL command also verifies

that the calibration signal is within ±5 dB of the entered calibration power.

`CAL WLPIT`   Calibrates at the wavelength of the designated signal pit.

See Also CALCOR, CALPWR, CALWL

CALCOR

Individually remove applied calibration correction factors.

XCALCR
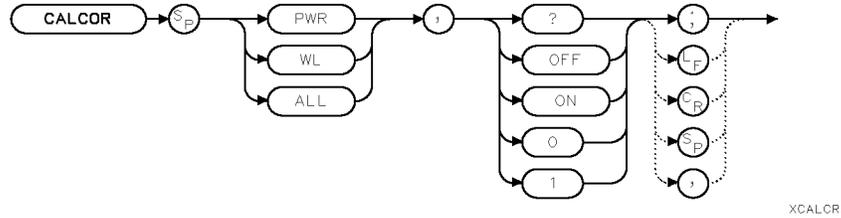
Item	Description
Preset State	CALCOR ALL,ON

Example

170   OUTPUT 723;"CALCOR PWR,?;"   *Query status of power correction factor.*
180   ENTER 723;A   *Assign response (1 or 0) to computer variable.*

Description   The CALCOR command disables the correction factors computed by the CAL command. The factors can be disabled or re-enabled individually or all at once. During normal operation, all correction factors are on. Disable the factors during special applications, or when you wish to supply your own calibration procedure.

CALCOR PWR,OFF disables power correction factors. These factors improve the power accuracy of displayed measurement results by compensating for irregularities in the system.

CALCOR WL,OFF disables wavelength correction factors. These factors improve the wavelength accuracy of displayed measurement results by compensating for irregularities in the monochromator.

Query Response
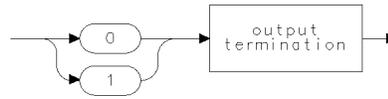
qgen0

See Also CAL, CALWL, CALPWR

CALDATA

Return factory-supplied calibration data.
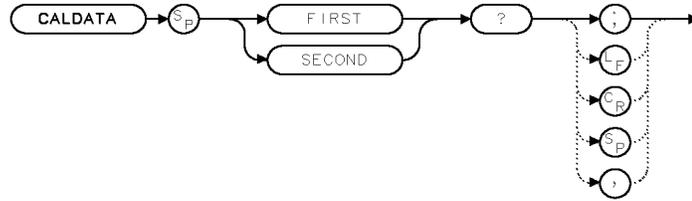
XCALD

Example

```
10  INTEGER NUM_PTS
20  REAL CAL(1:300,1:2)
30  MAT CAL=(-2E9)
40  OUTPUT 723;"CALDATA FIRST?;"
50  ENTER 723 USING "%,K";CAL(*)
60  NUM_PTS=1
70  WHILE CAL(NUM_PTS,1)>-1E9
80  NUM_PTS=NUM_PTS+1
90  END_WHILE
```

Description

The CALDATA command queries the optical spectrum analyzer for factory supplied calibration data. The calibration data returned is determined by the INSTMODE setting. The data may be DIODE only correction data or normal correction data for both the diode and the monochromator.

When INSTMODE is set to PWRMTR, the data returned is only the diode correction data. When INSTMODE is set to OSA or SR, the data returned is the combination of the grating order and diode correction data. Grating-only data is returned when INSTMODE is set to PRESEL or PD.

The CALDATA command queries factory supplied calibration data for the first or second grating order. The optical spectrum analyzer's internal monochromator uses a diffraction grating operating at either the first or second order modes. Second order correction data is used when the stop wavelength is less than 900 nm and the default setting, GRATORDER AUTO , is selected.

Query Response
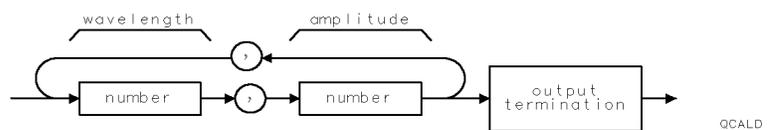
See Also INSTMODE, GRATORDER

CALPWR

Specify power level of reference signal used by calibration routine.
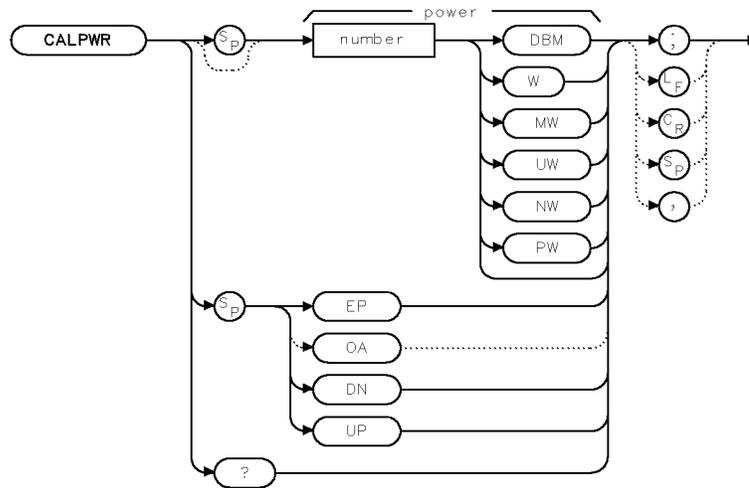
Item	Description
Power Range	Minimum level is —300 dBm.
Step Increment	1 major vertical-scale division

Example

140	OUTPUT 723;"CALPWR -15DBM;"	*Specify —15 dBm as the power level of the calibrator reference.*
150	OUTPUT 723;"CALPWR?;"	*Query value to computer.*
160	ENTER 723;N	*Assign value to computer variable.*
170	PRINT "Calibrate to ";N;"dBm"	*Print value on computer display.*

Description The CALPWR command specifies the power of the calibration reference used by the CAL command. Use CALPWR to specify a different calibration power. To prevent the generation of an error, the power value entered must be within ±5 dB of the calibration signal.

Query Response
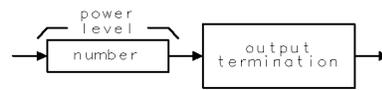
See Also CAL, CALCOR, CALWL

CALWL

Specify wavelength used by calibration routine.
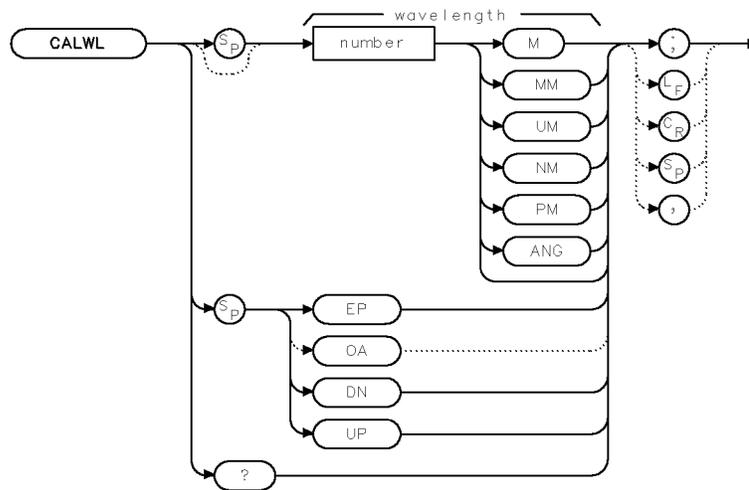
xcalwl

Item	Description
Wavelength Range	600nm - 1700nm
Step Increment	10% of span
Preset State	1300 nm

Example   150 OUTPUT 723;"CALWL 600NM;"   *Calibrate to 600 nm.*

Description The CALWL command specifies the wavelength of the calibration reference used with the CAL command. (The CAL command performs a calibration at the wavelength specified by CALWL.) Use CALWL to specify a different calibration wavelength. To prevent the generation of an error, the wavelength value entered must be within 1 to 2 nm of the calibration signal.

Query Response


```
         cal
      wavelength
      ┌─────────┐        ┌───────────────┐
─────►│ number  ├───────►│    output     ├─────►
      └─────────┘        │  termination  │
                         └───────────────┘
```
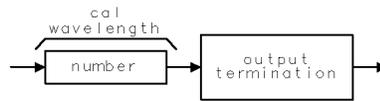
qcalwl

See Also CAL, CALCOR, CALPWR

CATALOG   Query the catalog of internal memory in a condensed format.
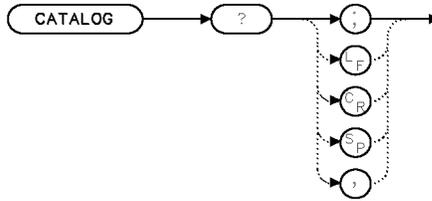
```
CATALOG  →  ?  →  ;
                  LF
                  CR
                  SP
                  ,
```
<div align="right">xcatal</div>

Example   `150  OUTPUT 723;"CATALOG?;"`

Description The CATALOG command queries the catalog of internal memory in a condensed format as follows:

```
0917504,0914480          total mem, avail mem
CONFIG,CONFIG,0005992    filename,filetype,filesize
LARGEST,LARGEST,0911438
FOO,FUNC      00011
FRED,VAR      00008
```

Name and filetype are restricted to a maximum number of 15 characters; otherwise, both fields are truncated. If the file size exceeds 99,999 bytes, the size is returned in kilobytes, indicated with a "K" suffix. Filetypes are categorized as listed below:

CONFIG   internal configuration data space
LARGEST   largest contiguous data space available
FUNC   FUNCDEF storage area
VAR   VARDEF storage area
TRACE   user-defined trace storage area
ASCII   data storage area for instrument states, stored traces A, B, or C, stored user keys, and data stored to a memory card.
AFUNC   user-defined active function (see the ACTDEF command)
PROG   HP generated personality or program
APROG   HP generated active function

Cataloging MSI

To catalog the contents of the memory selected by the `MSI` command, refer to the `DSPTEXT CAT?` command.

When system DEBUG ON is active, the format is modified to give more information on actual file status. With DEBUG ON the output is as follows:

```
0917504,0914480
CONFIG,CONFIG, 0005992
LARGEST,LARGEST,0911438
,AG      00042
FOO,G       000040
FRED,G      000056
```

The output format of all non-system files is as follows:

<filename>,<file status>   <total memory used>

Where:

The <filename> is the name of one of the files listed above.
The <file status> is one or more of the letters listed in the
List of <file status> Letter Descriptions.

List of <file status> Letter Descriptions

A   indicates the memory block that is available.

D   indicates the memory block that is disposed. If the block is in use, it is marked and disposed of as soon as it is no longer in use.

G   indicates a valid memory block (one that is a real block with valid data).

P   indicates the memory block is protected. The user has protected the block to prevent its being disposed. If memory is formatted or erased, these blocks are removed.

*   indicates the memory block is in use. Either an HP generated function is using the block, or a downloadable program (DLP) is currently being executed.

Note the following items when using the CATALOG command with DEBUG ON.

- The listing reflects the ordering and structure of what is physically in memory as opposed to what is indicated in the sorted table. Therefore, some understanding of how the memory manager manipulates, resizes, and moves around these memory blocks is sometimes needed to understand what is shown.

- Rather than stepping down through the table, the listing uses pointers stored within each block to locate the next block. A successful catalog listing verifies that the pointers are all correct.

- The size represents the entire size of the block (as calculated from the internal pointers), not just the user data area. A block includes the above-mentioned pointers, the name, the data type, and so on.

- Instead of showing the data type, the block's status as used by the memory manager is displayed. Refer to the List of <file status> Letter Descriptions above.

- Normally the garbage collection routine is run before a catalog listing is displayed. This is not done when DEBUG is turned ON.

Query Response

With DEBUG OFF, the output is as follows:

```
0917504,0914480          total mem, avail mem
CONFIG,CONFIG,0005992    filename,filetype,filesize
LARGEST,LARGEST,0911438
FOO,FUNC    00011
FRED,VAR    00008
```

With DEBUG ON, the output is as follows:

```
0917504,0914480
CONFIG,CONFIG, 0005992
LARGEST,LARGEST,0911438
,AG      00042
FOO,G    000040
FRED,G   000056
```

See Also DSPTEXT

CENTERWL   Set the center wavelength of measurement range.

Item	Description
Wavelength Range	Actual range is hardware dependent. Firmware range is 600 nm to 1700 nm. With wavelength limit OFF, the range is 350 nm to 2000 nm.
Step Increment	Determined by SS command.
Preset State	|highest possible stop wavelength — lowest possible start wavelength|÷2

Example

```
100   OUTPUT 723;"SNGLS;"
110   OUTPUT 723;"CENTERWL 1200NM;"      Set center wavelength to 1200 nanometers.
120   OUTPUT 723;"TS;"                    Update displayed trace.
```

Description   The CENTERWL command sets the center wavelength. The span remains
constant. The center-wavelength and the start- and stop-wavelength settings
are related as follows:

$$center\ wavelength = \frac{stop\ wavelength + start\ wavelength}{2}$$

CENTERWL UP and **CENTERWL DN** increase or decrease the center wavelength
by increments determined by the step function, SS.

Query Response
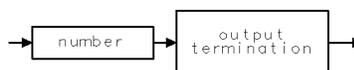
qgennum

See Also   SS, STARTWL, STOPWL, MKSS, FS, SP, WLOFFSET, WLLIMIT ON, WLLINIT
OFF

CHEIGHT

Query the character height.

XCH
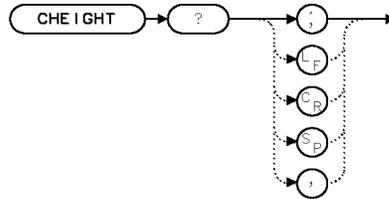
Item	Description
Preset State	full window, 4.34

The CHEIGHT? query computes the character height as a percentage of the window height. This command is necessary for programs that want to adjust their graphics and TEXT displays depending on the window size. The characters remain the same size regardless of what size the window is. The units of the number returned is percent. For Example, executing the CHEIGHT? query on a full-screen window display returns 4.34. That is, the character height is 4.34% of the window height where the window is 23 (100 ÷ 4.34) characters high.

Query Response
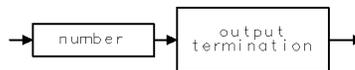
qgennum

See Also TEXT, CWIDTH, PA, PR

CHOP

Turn on or off the input and light for chop mode.
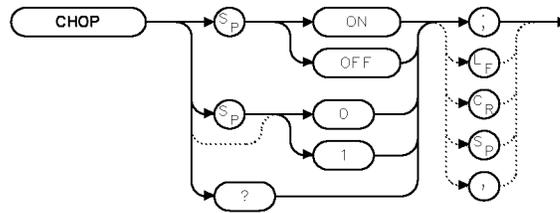
XCHOP

Item	Description
Preset State	ON

Example

```
150   OUTPUT 723;"CHOP ON;"
```

Description The CHOP command turns the input and light for chop mode on or off.

Provided the sweep times are long, the CHOP command increases dynamic range. CHOP mode subtracts the ambient light from the signal. With CHOP mode ON, the instrument automatically enables chop mode for sweep times 40s, or longer. CHOP mode is disabled when it is set to OFF.

Query Response
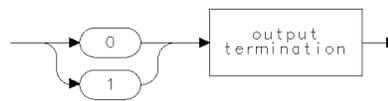
qgen0

See Also ST

CLRDSP

Clear the display.
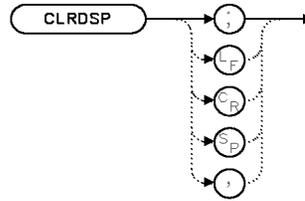
XCLRDS

Example Clear the analyzer of user-created graphics.

110	OUTPUT 723;"PU;PA 300,900;"	*Place pen at (300,900).*
120	OUTPUT 723;"TEXT$WARNING$;"	*Write text.*
130	OUTPUT 723;"WAIT 4S;"	*Wait four seconds.*
140	OUTPUT 723;"CLRDSP;"	*Clear graphics.*

Description The CLRDSP command removes all spectrum-analyzer graphics from the display and its memory. CLRDSP executes the following:

- Deletes all items from memory (IT).
- Turns off the reference-graphics mode (IT 0).
- Removes all graphic offsets (OR).
- Sets the scale of the X-Y coordinates to 0 to 1000 for both X and Y (SCALE).
- Sets the origin to 0,0 (OR).
- Deletes all graphics windows (DWINDOW).
- Deletes text displayed by the TEXT command (*not* by the TITLE command).
- Deletes numbers displayed with the DSPLY command.

See Also IT, OR, SCALE, DWINDOW, TEXT, DSPLY, PA, PR, PD, LINET, PU

CLRW   Continuously measure and store detected signals.
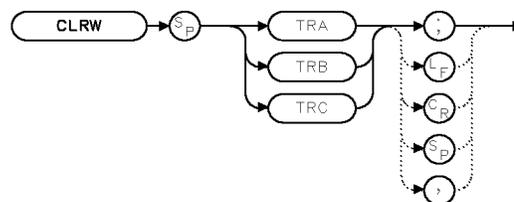
XCLRW

Item	Description
Preset State	`CLRW TRA`

Example   120   OUTPUT 723;"CLRW TRA;"   *Clear-write trace A.*

Description The CLRW command enables the clear-write mode, which continuously
measures and stores any signals present at the spectrum-analyzer input port
when the trigger conditions for the sweep are met. If the spectrum analyzer
has a display, measurement results are continuously displayed during the
clear-write mode. CLRW operates on trace A, B, or C. CLRW initially clears
the selected trace, setting the amplitude of all trace elements equal to the
bottom graticule-line value. The trigger then signals the start of the sweep
and the trace is updated continuously as the sweep progresses.

See Also TRPST, BML, BTC, BXC, TRSTAT, AMB, AMBPL, AMC, AXB, TRDSP

CLS   Set all bits in the status-byte register to 0.
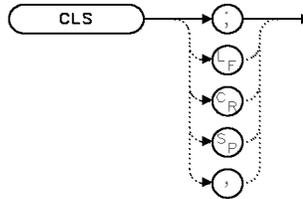
XCLS

Example   `10  OUTPUT 723;"CLS;"`

Description The CLS (clear status) command clears all the bits in the status register. Upon completion of this command, the **Command Complete** bit of the status register is set. Refer to Chapter 4, "Monitoring System Operation" for additional information about status byte use.

See Also RQS, SRQ, STB

COMPRESS

Reduce trace length while retaining relative wavelength and amplitude characteristics of trace data.
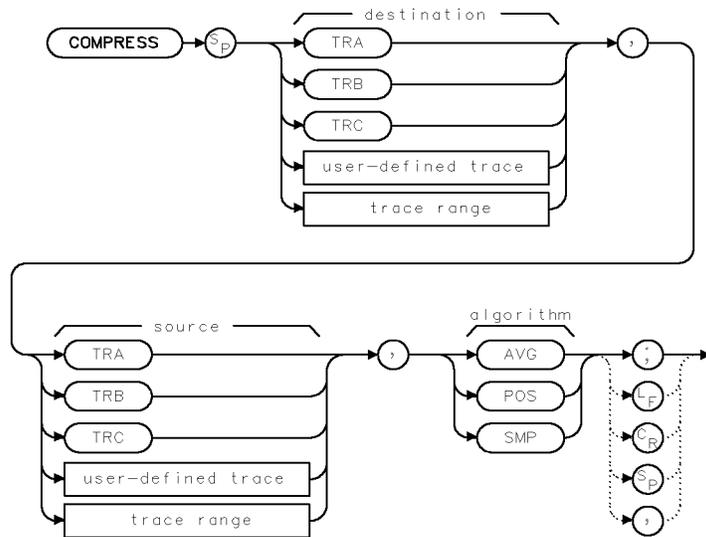
XCOMPR

Item	Description
Source or Destination Length Range	3 – 2048 position units.
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data. −32,768 – +32,767 measurement units for logarithmic trace data.

Example

110	OUTPUT 723;"CENTERWL 1000NM;SP 300NM;"	*Set measurement range.*
120	OUTPUT 723;"TRDEF TRC,100;"	*Change length of trace C to 100 elements.*
130	OUTPUT 723;"BLANK TRA;SNGLS;"	*Blank trace A. Activate single-sweep mode.*
140	OUTPUT 723;"CLRW TRB;TS;"	*Measure with trace B. Sweep measurement range.*
150	OUTPUT 723;"COMPRESS TRC,TRB,POS;"	*Compress trace C to fit trace B.*
160	OUTPUT 723;"BLANK TRB;"	*Blank trace B.*
170	OUTPUT 723;"VIEW TRC;"	*View trace C.*

Description The COMPRESS command stores a compressed copy of the source trace in a smaller destination trace. The source trace is divided into the same number of intervals as points in the destination trace. The data within each source interval is compressed into a single value for the corresponding destination trace point. For example, if the source trace is 1000 points long, and the destination trace is 100 points long, COMPRESS divides the source trace into 100 ten-point intervals and computes a compressed value for each interval according to the compression algorithm selected. The compressed value for the first interval becomes the amplitude of the first point in the destination trace. The 99 remaining compressed values determine the amplitude of the last 99 points of the destination trace.

The compression algorithms are as follows:

COMPRESS AVG selects the average value of the points in each interval.

COMPRESS POS selects the highest value in each interval.

COMPRESS SMP (sample) selects the last point in the interval.

CONCAT

Concatenate two traces to make one trace.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*. −32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	± 1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if sum of sources is longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example
    200 OUTPUT 723;"CONCAT TRC,TRA[1,400],TRB[401,800];"
Concatenate last half of trace B to end of first half of trace A. Store new array in trace C.

Description The CONCAT command concatenates source 2 to the end of source 1, then stores the new array in the destination. The size of the destination and sources can vary from 3 to 2048 points. If necessary, reduce trace lengths with COMPRESS, TRA, TRB, TRC, or TRDEF. Otherwise, the concatenated array may not fit in the destination and trace information is lost. When using TRA, TRB, TRC, or trace range for source, use SNGLS and TS to be sure data reflects one complete sweep. TRDEF required when using user-defined trace. VARDEF required when using user-defined variable.

See Also TS, COMPRESS, TRDEF, TRA, TRB, TRC

CONFIG

Query the instrument configuration for model numbers and MSIB addresses of all slaves.
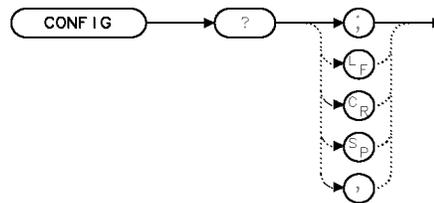

```
100   OUTPUT 723;"CONFIG;"
```

Description The CONFIG command queries a series of ASCII strings, separated by carriage returns, indicating the model number and MSIB address of this module and any slave instruments. The format is as follows:

<model number>,<MSIB row number>,<MSIB column number>

An example of the CONFIG query response is:

70950A,0,23

Query Response
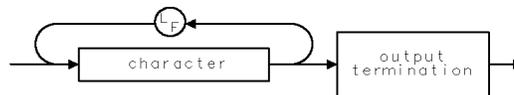
See Also DSPTEXT CONFIG

CONTS

Sweep continuously each time trigger conditions are met.
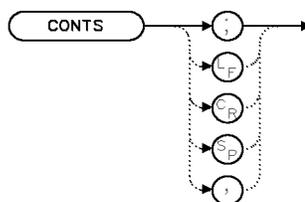
XCONTS

Example

```
170   OUTPUT 723;"CONTS;"
```

Description The CONTS command activates the continuous-sweep mode where the analyzer sweeps continuously as long as the trigger and data-entry conditions are met. Executing CONTS aborts the current sweep to start a new one.

See Also TM, VTL, VTH, SNGLS, TRPST

CORSEL

Select the flatness correction table.
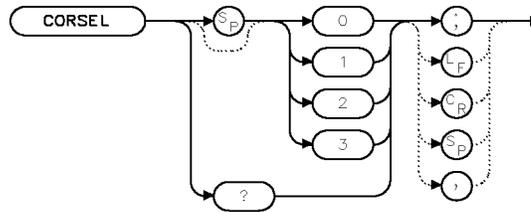
XCORS

Example   10 OUTPUT 723;"CORSEL 0;"

Description The CORSEL command selects which flatness correction table will be applied to the measurement data.

> **NORMAL** : Applies both diode and grating flatness correction tables. Returned as 0 by the query.

> **DIODE** : Applies only the diode flatness correction table. Returned as 1 by the query.

> **GRATING** : Applies only the grating flatness correction table. Returned as 2 by the query.

> **NONE** : Turns off diode and grating flatness correction tables. Returned as 3 by the query.

Query Response
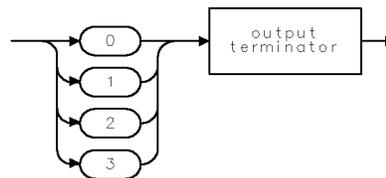
QCORS

See Also OPTSW, INSTMODE, CALDATA

CORTOLIM

Transfer AMPCOR data to a limit line.
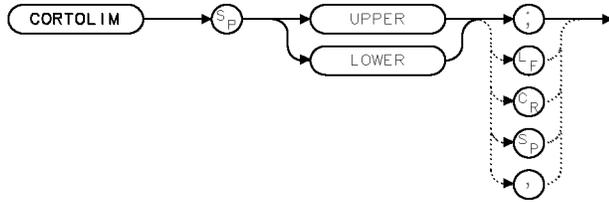
xcorto

Example   10 OUTPUT 723;"CORTOLIM UPPER;"   *Transfer amplitude-correction data into the upper limit line.*

Description The CORTOLIM command transfers data to the specified limit line. This allows the AMPCOR data to be viewed or edited. Use LIMTOCOR to bring the edited data back to the amplitude-correction storage.

See Also AMPCOR, LIMTOCOR

CWIDTH

Query the character width as a percentage of the window width.
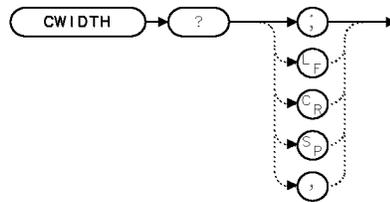
XCWID

Item	Description
Preset State	full window, 1.87

Description The CWIDTH query computes the character width as a percentage of the window width. This command is necessary for programs that want to adjust their graphics and TEXT displays depending on the window size. The characters remain the same size regardless of what size the window is. The units of the number returned is percent. For Example, executing the CWIDTH query on a full-screen window display returns 1.87. That is, the character width is 1.87% of the window width where the window is 53 (100 ÷ 1.87) characters wide.

Query Response
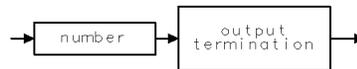
qgennum

See Also CHEIGHT, TEXT, PA, PR

DEBUG

Start the program debugging mode.
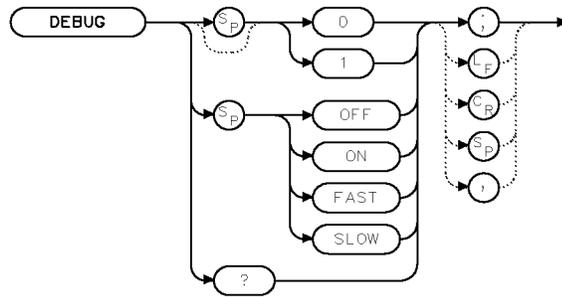
XDEBUG

Item	Description
Preset State	DEBUG OFF

Example

210	OUTPUT 723;"DEBUG SLOW;"	*Activate the slow debug mode.*
220	OUTPUT 723;"DEBUG OFF;"	*Turn off debug mode.*

Description The DEBUG command is a troubleshooting aid for locating faulty syntax in spectrum-analyzer programs. **DEBUG ON, DEBUG SLOW,** and **DEBUG FAST** activate the debug mode, where spectrum analyzer commands are displayed on the data line below the graticule as the commands are executed. When a faulty spectrum-analyzer command is encountered, subsequent analyzer commands are not executed. The faulty command is the last command shown at the right end of the data line. DEBUG also activates keys that step, pause, or continue program operation.

DEBUG FAST processes commands quickly.

DEBUG SLOW processes command slowly.

Query Response
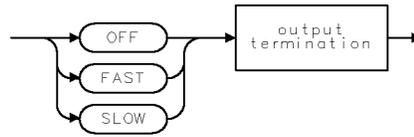
QDEBUG

DELETE

Delete graphics item from memory.

XDELET

Example

120	CLEAR 723	*Initialize analyzer.*
130	OUTPUT 723;"IP;"	
140	OUTPUT 723;"CLRDSP;"	*Clear graphics.*
150	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
160	OUTPUT 723;"IT 1;VW OFF;"	
170	OUTPUT 723;"TEXT$An Item$;"	*Define item 1 as text located at (150,600).*
180	OUTPUT 723;"OR 150,600;"	
190	OUTPUT 723;"VW ON;"	*Display item 1.*
200	OUTPUT 723;"IT 2;VW OFF;"	*Define item 2 as a triangle at (500,500).*
210	OUTPUT 723;"PA 0,0,50,50,";	
220	OUTPUT 723;"50,0,0,0;"	
230	OUTPUT 723;"OR 500,500;"	
240	OUTPUT 723;"VW ON;"	*Display item 2.*
250	OUTPUT 723;"WAIT 2;"	*Wait two seconds.*
260	OUTPUT 723;"IT 1;DELETE;"	*Delete item 1 from display and memory.*

Description The DELETE command blanks the active graphics item from the display and from its memory. This command does not apply to item 0 (zero). Items are stored in memory located in the display (such as HP 70004, HP 70205, or HP 70206) with the IT command.

See Also IT

DFB_

Start the DFB laser characterization measurement DLP.
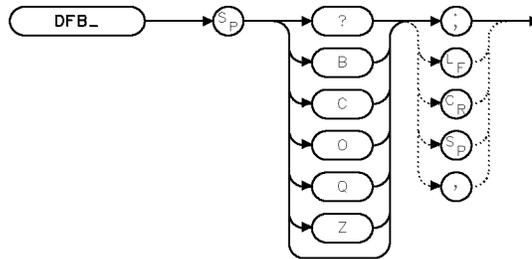
xdfb

Parameters require one underscore and one space.

Item	Description
DFB_ ?	Queries the DLP and returns: peak wavelength wavelength offset stop band center offset SMSR (side mode suppression ratio) peak amplitude bandwidth bandwidth amplitude
DFB_ B	Turns on or off the view of the stop-band peaks.
DFB_ C	Turn on or off the view of the SMSR peaks.
DFB_ O	Turn on or off the calculations.
DFB_ Q	Exit the measurement DLP.
DFB_ Z	Performs the measurement system optimization function, increasing instrument sensitivity to ensure the side-modes are properly located and displayed.

Example

10 OUTPUT 723;"IP;";	*Preset the instrument.*
20 OUTPUT 723;"DFB_;";	*Start the DFB characterization measurement.*
30 OUTPUT 723;"DFB_ C;";	*Turn on or off, the view of SMSR peaks.*
40 OUTPUT 723;"DFB_ Q;";	*Exit the DFB characterization measurement DLP.*

Description The DFB_ command automatically measures the characteristics of a distributed-feedback laser source. Refer to the *HP 71450B/1B/2B User's Guide* for additional information.

Query Response
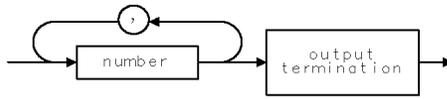
qgnumr

DISPOSE

Erase all or part of internal memory.

Example
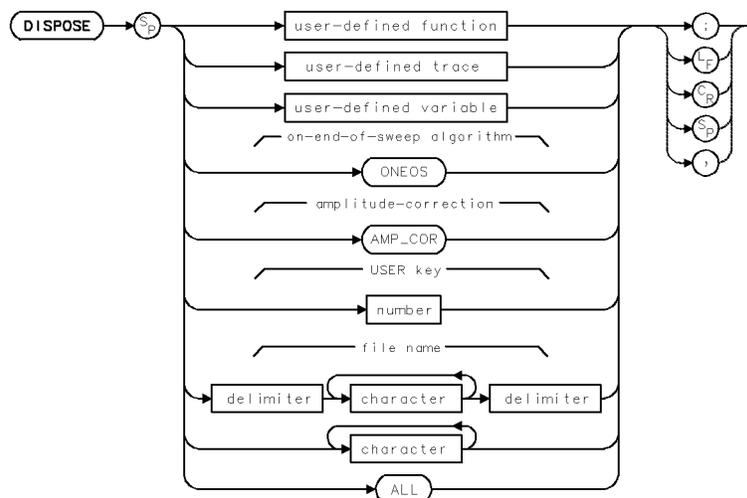
120	CLEAR 723	*Initialize analyzer.*
130	OUTPUT 723;"IP;"	
140	OUTPUT 723;"VARDEF VA,0;"	*Define user-defined variable, VA, equal to 0.*
150	OUTPUT 723;"FUNCDEF NNN,@ @;"	*Define null, user-defined function, called NNN. (There are no commands between delimiters @ and @.)*
160	OUTPUT 723;"KEYDEF,7,NNN,C;"	*Define key 7, called C, which has the null function.*
170	OUTPUT 723;"PROTECT VA,OFF;PROTECT NNN,OFF;"	*Turn off internal-memory protection.*
180	OUTPUT 723;"DISPOSE 7;"	*Erase key from internal memory.*
190	OUTPUT 723;"DISPOSE VA;"	*Erase user-defined variable from internal memory.*
200	OUTPUT 723;"DISPOSE NNN;"	*Erase user-defined function from internal memory.*

Description The DISPOSE command clears internal memory, which is reserved for operator use. Delimiters required before and after *file name* when it contains a number or lower-case letters. When *file name* is upper-case alpha characters only, delimiters are optional.

Executing **DISPOSE ALL** clears the following from internal memory.

Internal Memory Contents

Article	Storing Command
User-defined functions	ACTDEF, FUNCDEF, RCLD, LOAD
User-defined variables	VARDEF, RCLD, LOAD
On-end-of-sweep algorithm	ONEOS, RCLD, LOAD
User-defined key	KEYDEF
User-defined trace arrays	TRDEF, RCLD, LOAD
Amplitude correction factors	AMPCOR
Limit-line trace arrays	LIMILINE, FUNCDEF, RCLD, LOAD
(USER) keys files	STOR, SAVEU
Limit line files	STOR, LIMISAV
Trace files	STOR, SAVET
State register files	STOR, SAVES
Program Files	SAVED

An article can be cleared from internal memory individually by specifying its name, except for user-defined keys which are specified by number. **DISPOSE ALL** does not clear any articles from reserved memory, such as the predefined traces A, B, and C. **DISPOSE ALL** does not clear items (IT) or graphics windows (DWINDOW). The PROTECT command protects articles in internal memory against erasure with the DISPOSE command. If any articles are protected, **DISPOSE ALL** erases only unprotected articles. When DISPOSE attempts to erase a single protected article, an error message is generated. The following table summarizes commands that protect, erase, or allocate the contents of **internal memory**. For more information, see Chapter 3, "Controlling Memory."

Commands that Erase or Protect Internal Memory

Memory Components	Commands that Erase or Reset Memory		Commands that Protect Memory	
	PURGE/DISPOSE/PERASE/ERASE	NSTATE	PROTECT	PSTATE
user-defined variable	X		X	
user-defined trace	X		X	
user-defined function	X		X	
active user-defined function	X		X	
on-end-of-sweep-algorithm	X		X	
trace-amplitude correction factors	X		X	
limit-line trace arrays	X		X	
trace-amplitude correction factors	X		X	
state register file |settings and title line,|	X	X	X	X
trace file	X		X	
(USER) keys file	X		X	
Limit line file	X		X	

See Also PROTECT. Any command that stores or loads articles into internal memory: ACTDEF, AMPCOR, FUNCDEF, KEYDEF, LIMILINE, LIMISAV, LOAD, NSTATE, ONEOS, SAVES, SAVET, SAVEU, STOR, USERERR, USERKEY, USERMSG, VARDEF

DISPU

Dispose of a user menu.

```
( DISPU )──▶(Sp)──▶[ number ]──┬──▶(;)──┐──▶
                               ├──▶(Lғ)──┤
                               ├──▶(Cᴿ)──┤
                               ├──▶(Sᴘ)──┤
                               └──▶(,)──┘
```
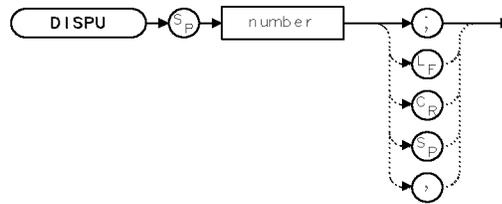
XDISPU

Item	Description
Range	integer, 0 - 9999, depending on memory

Example   10 OUTPUT 723;"DISPU 10;"   *Dispose the specified user menu.*

Description The DISPU command is used to dispose USER menus that have been saved with the SAVE USERKEY softkey or the equivalent SAVEU command. The current PREFX is used.

See Also SAVEU, PREFX

DIV

Divide source 1 by source 2, then send the result to the destination:

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range,* or *user-defined trace.*
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range,* or *user-defined trace.*
	$\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable,* or *user-defined variable.*
Source-Length Range	3 – 2048. Default is 800 when using *TRA, TRB, TRC.* When source lengths differ, the last element of the shorter source is repeated for the division process. When the sources are longer than the destination trace, they are truncated to fit.
Destination-Length Range	3 – 2048 position units.

Example

140	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
150	OUTPUT 723;"TS;"	*Sweep trace A.*
160	OUTPUT 723;"DIV TRB,TRA,2;"	*Divide trace A by 2. Store results in trace B.*
170	OUTPUT 723;"VIEW TRB;"	*View trace B.*

Description The DIV command divides source 1 by source 2, then sends the result to the destination. If traces are used, be sure each contains a complete sweep of measurement information before executing DIV. The sources and destination may be of different length. The length of traces can vary from 3 to 2048. Variables have a length of one. When sources differ in length, the last element of the shorter source is repeated for the division process. When the sources are longer than the destination, they are truncated to fit. When using TRA, TRB, TRC, or trace range for source, use SNGLS and TS to be sure data reflects one complete sweep. TRDEF required when using user-defined trace. VARDEF required when using user-defined variable.

DL

Position a line on the display.
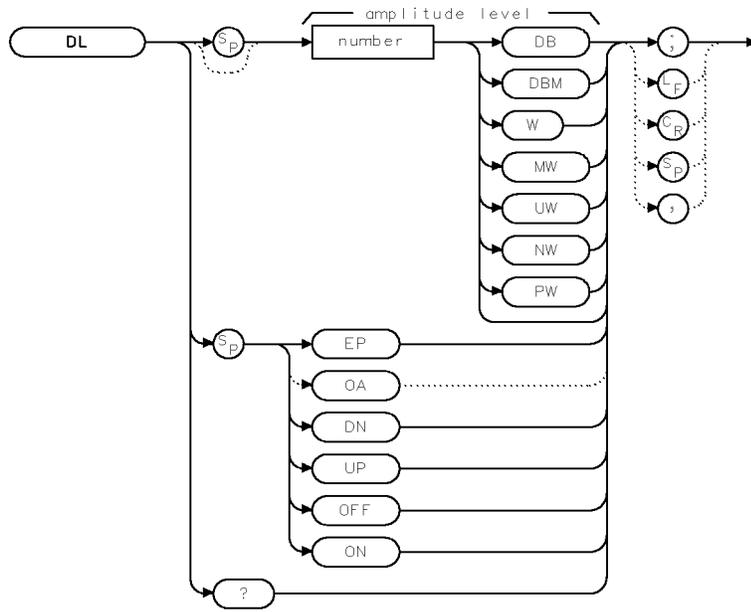
Item	Description
Step Increment	1 major vertical division
Preset State	DL OFF; DL 0DBM

Example Place display line at highest amplitude value of trace A.

150	OUTPUT 723;"MKPK HI;"	*Mark highest amplitude value.*
160	OUTPUT 723;"DL ON;"	*Turn on the display line.*
170	OUTPUT 723;"MOV DL,MKA;"	*Move the marker value into DL (predefined variable).*

Description The DL command positions a line on the display for visual and computational purposes.

Query Response
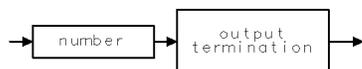
qgennum .

See Also TRPST and any amplitude-related commands, such as ROFFSET, MEASURE, RL, LN, LG, INZ and math commands.

DONE

Return a "1" to the controller after the execution of all commands preceding the DONE command.
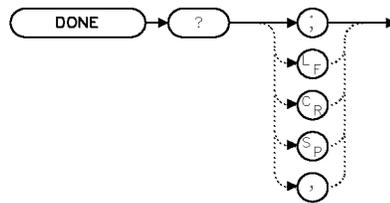
XDONE

Example Use DONE to delay the execution of additional program commands until the entire measurement range is swept.

160	OUTPUT 723;"TS;DONE?;"	*Sweep the measurement range, then send a "1" when the sweep is completed.*
170	ENTER 723;A	*Assign "1" to computer variable.*

Description The DONE command returns a "1" to the controller when all of the commands encountered before the DONE command have executed.

For example, when a **DONE?** query is followed by an "ENTER" (or computer read) command at some point in the command sequence, the program waits for the spectrum analyzer to return the "1" before executing additional program lines. This function allows the controller to synchronize with the spectrum analyzer.

Query Response
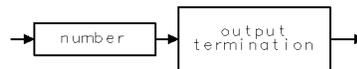
qgennum

DSPLY

Display a variable anywhere on the analyzer display.
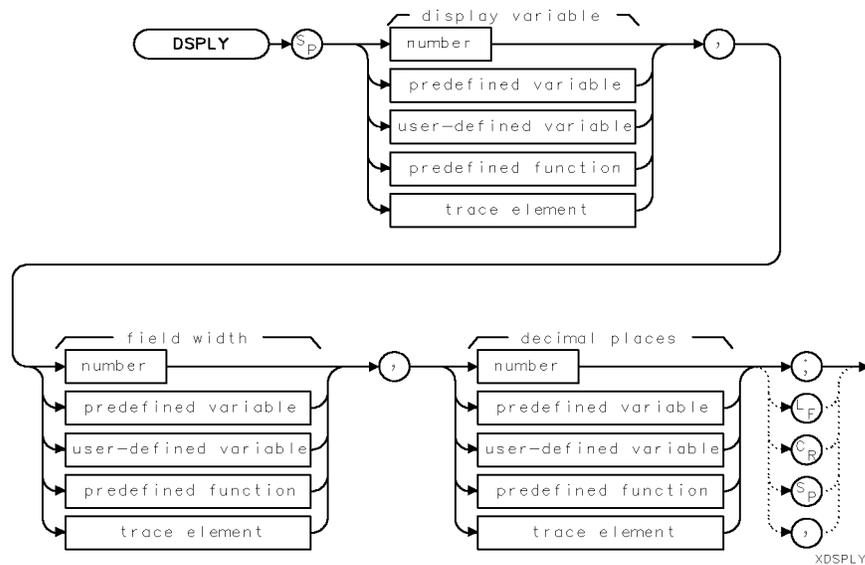
Item	Description
Field width	Determines the number of characters displayed, including digits, sign, and decimal point.
Field-Width Range	1 – 18
Decimal Places	Determines the number of digits to the right of the decimal point.
Decimal-Place Range	For field widths greater than 3, range is from 0 to field width less three. For field widths equal to or less than 3, range is 0 (zero).

Example

130	OUTPUT 723;"CLRDSP;"	*Clear graphics. Reset origin to (0,0), and scale to 0 to 1000 for X and Y.*
140	OUTPUT 723;"PU;PA 500,700;"	*Position pen at (500,700).*
150	OUTPUT 723;"DSPLY CENTERWL,12,0;"	*Display the center wavelength on the analyzer display without decimal values, using a field width of 12. Notice that 11 numbers are displayed. The twelfth character in the data field is sign information.*

Description   The DSPLY command displays the value of a variable at the current position
of the pen, determined by the plot-absolute (PA) or plot-relative (PR)
commands or determined by the OR command during the reference-graphics
mode. If the value does not fit in the specified parameters, the variable is
displayed in exponential notation. DSPLY cannot be executed after `DWINDOW`
`ON`.

See Also   CLRDSP, PA, PR

DSPMODE

Display catalog of user memory contents; HP-MSIB addresses of modules; module-level parameters; or system-level parameters, serial numbers, and revision numbers.
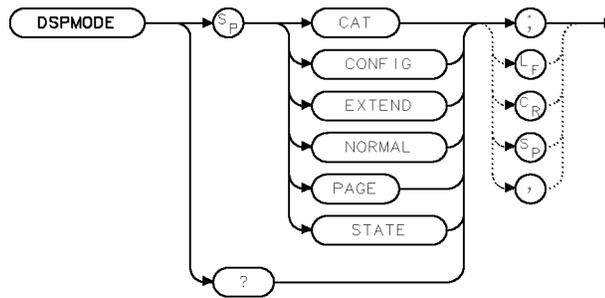
XDSPMD

Item	Description
Preset State	DSPMODE NORMAL

Example

140 OUTPUT 723;"DSPMODE CAT;"   *Display contents of currently selected user memory.*

Description The DSPMODE command lists descriptions of system or module configurations on the analyzer display:

DSPMODE CAT displays the contents of the currently selected user memory. Listed are any user-defined functions, variables, traces, files, and the name and byte size of each entry. The size of traces A, B, and C are also provided if **internal memory** is selected.

DSPMODE CONFIG describes the placement of the optical spectrum analyzer module in the system by its HP-MSIB address (row and column number).

DSPMODE EXTEND displays module-level parameters for each module in the system.

DSPMODE NORMAL activates the normal analyzer display where the graticule and swept responses are shown.

DSPMODE PAGE displays the next "page" of information, if any. Execute it when previous executions of DSPMODE provide more than one page of information.

DSPMODE STATE lists the current state of the system-level parameters under operator control, such as wavelength ranges and bandwidth settings. It also lists the identification and serial numbers, and firmware revision of the spectrum analyzer.

Use the DSPTEXT command to send the listings shown on the analyzer display to the computer. To view the catalog of the desired mass storage device, (HP-IB device, HP-MSIB device, or internal memory), execute MSI before executing **DSPMODE CAT**. Internal memory is the storage device when power is applied to the spectrum analyzer. Execute **DSPMODE PAGE** after **DSPMODE CAT, DSPMODE CONFIG, DSPMODE EXTEND**, or **DSPMODE STATE** has been executed.

Query Response
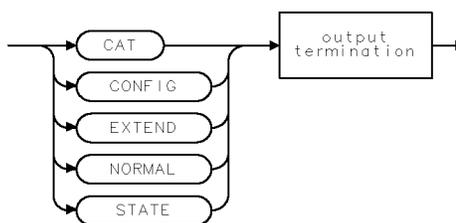
qdspmd

See Also DSPTEXT, MSI

DSPTEXT   Query catalog of user memory contents; HP-MSIB addresses of modules; module-level parameters; or system-level parameters, serial numbers, or revision numbers.

XDSPTX
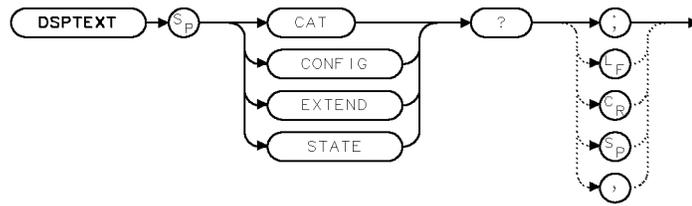
Example   10 OUTPUT 723;"DSPTEXT CAT?;"   *Query listing of contents of currently selected user memory.*

Description The DSPTEXT command sends system-level or module-level information to the computer. The information sent duplicates the information displayed on the analyzer display with the DSPMODE command or the `catalog & MSI` key. Each line of data is sent as one record separated by a line feed.

DSPTEXT CAT? sends the contents of the currently selected user memory. Listed are any user-defined functions, variables, traces, or files. Also listed are the name and byte size of the trace registers for traces A, B, and C. To send the catalog of the desired mass storage device, (HP-IB device, HP-MSIB device, or internal memory), execute MSI before executing **DSPTEXT CAT**. Internal memory is the storage device when power is applied to the spectrum analyzer.

DSPTEXT CONFIG? sends the placement of the optical spectrum analyzer module in the system by its HP-MSIB address (row and column number).

DSPTEXT EXTEND? sends module-level parameters for each module in the system.

DSPTEXT STATE? sends the current state of the system-level parameters under operator control, such as wavelength ranges and bandwidth settings. It also sends the identification and serial numbers, and firmware revision of the spectrum analyzer.

Query Response
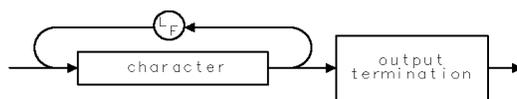
qgenchar .

See Also MSI, CATALOG, DSPMODE

DWINDOW

Define alternate scale for graphics.
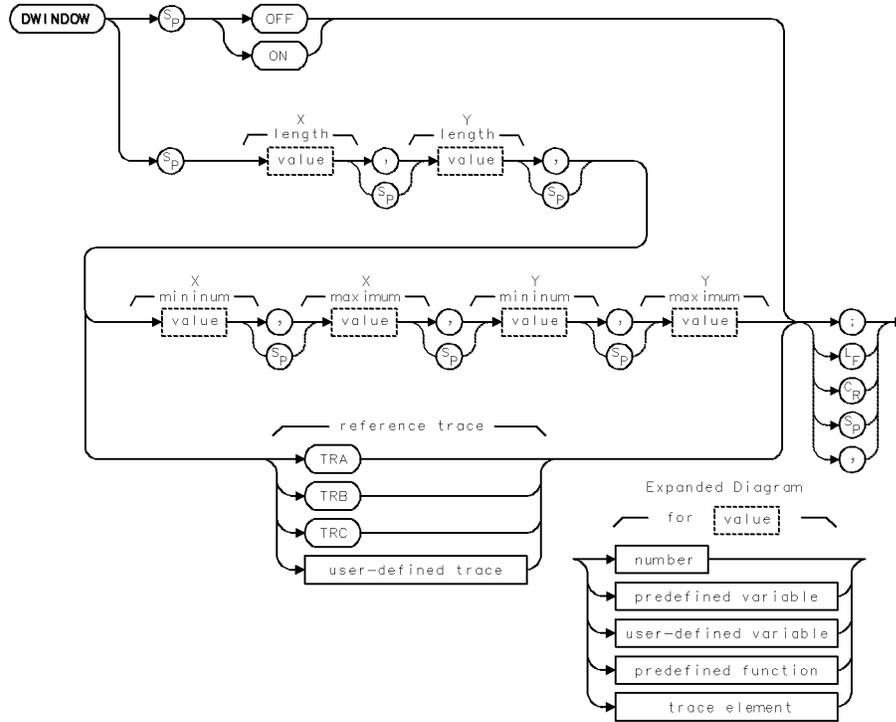
Item	Description
X-Length Range	− 32,768 – 32,767, in units set by SCALE command.
Y-Length Range	− 32,768 – 32,767, in units set by SCALE command.
X-Minimum Range	− 32,768 to + 32,767
X-Maximum Range	− 32,768 to + 32,767
Y-Minimum Range	− 32,768 to + 32,767
Y-Maximum Range	− 32,768 to + 32,767
Preset State	DWINDOW OFF

Example Draw Y = X² on the analyzer display.

180	CLEAR 723	*Initialize analyzer.*
190	OUTPUT 723;"IP;"	
200	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
210	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
220	OUTPUT 723;"OR 600,550;"	*Set origin to (600,550) in current scale units. (CLRDSP sets units of X and Y axis from 0 to 1000.)*

230   OUTPUT 723;"DWINDOW 400,400,-10,10,0,100;"

Define a graphics window that is square and has sides 400 units long, in units determined by CLRDSP above. The window itself contains its own scale. The minimum and maximum value for its X axis are −10 and 10. The minimum and maximum values for its Y axis are 0 and 100.

240	OUTPUT 723;"PU;PA −10,0;"	*Outline the window.*

250   OUTPUT 723;"PD;PA −10,100,10,100,10,0,−10,0;"

260	OUTPUT 723;"VARDEF XXIS,−10;"	*Assign −10 to a user-defined variable.*
270	OUTPUT 723;"VARDEF YAXIS,0;"	*Assign 0 to a user-defined variable.*
280	OUTPUT 723;"PU;PA −10,100;PD;";	*Place the pen at (−10,100) in window units.*
290	OUTPUT 723;"REPEAT;";	*Begin repeat loop.*
300	OUTPUT 723;"ADD XXIS,XXIS,1;";	*Add 1 to XXIS.*
310	OUTPUT 723;"MPY YAXIS,XXIS,XXIS;";	*Set YAXIS equal to (XXIS).*
320	OUTPUT 723;"PA XXIS,YAXIS;";	*Draw square function point-by-point.*
330	OUTPUT 723;"UNTIL XXIS,EQ,10;"	*End repeat loop when XXIS equals 10.*
340	OUTPUT 723;"PU;PA −2,80;"	*Place pen at (−2,80) in window units.*
350	OUTPUT 723;"DWINDOW OFF;"	*Turn off the window. Window parameters are retained in memory. The following TEXT command does not operate when the window is on.*
360	OUTPUT 723;"TEXT$Y=X"	*Write text at current pen position.*
370	OUTPUT 723;CHR$(130);"$;"	

Description The DWINDOW command creates a new coordinate system that repositions, magnifies or reduces graphics on the analyzer display. This new coordinate system is called a graphics window. DSPLY, GRID or TEXT cannot be executed after DWINDOW ON . DWINDOW defines the *size* and *scale* of the graphics window. The size of the window is determined by *X Length* and *Y Length*, which are specified in the current scale units. (Scale units are 0 to 1000 for both X and Y unless specified otherwise with the SCALE command.) The scale of the coordinate system within the window is determined by *X Minimum, X Maximum, Y Minimum,* and *Y Maximum*.

The position of the window on the analyzer display is specified with the OR command. Use DWINDOW to display custom graphics created with the plotting commands (PA, PR, PD, PU, MK) or to display trace information with the GRAPH command. If custom graphics are displayed, *X Minimum*, *X Maximum*, *Y Minimum*, and *Y Maximum* scale the coordinate system of the window in increments. DWINDOW cannot be used when displaying variables, grids, or text with the DSPLY, GRID, or TEXT commands.

See Also SCALE, OR, TRCOND, SCALE, GRAPH, IT, PA, PR, PD, PU, MK

ENTER

Receive data over HP-IB as a controller.

XENTER

Item	Description
HP-IB Address Field	specifies the HP-IB address of the device.
HP-IB Address Range	0 – 30
Format field	Formats the queried data.
K	Expects real, free-field numbers |ASCII decimal|.
B |byte|	Expects a single 8-bit byte.
W |word|	Expects two 8-bit bytes.

Example Use the analyzer to control an HP 3478 multimeter via HP-IB. Run the program, then press the user-defined key, VDM, to communicate with the multimeter.

```
150   REMOTE 7                        Place all devices in remote status.
160   CLEAR 723                       Initialize analyzer.
170   OUTPUT 723;"IP;"
180   OUTPUT 723;"VARDEF NN,0;";      Assign 0 to user-defined variable, NN.
```

190	OUTPUT 723;"FUNCDEF VDM,^";	*Begin definition of user-defined function, called VDM.*
200	OUTPUT 723;"OUTPUT 25,K,$H5$;";	*Send message, "H5" to multimeter.*
210	OUTPUT 723;"ENTER 25,K,NN;";	*Receive value in amperes from multimeter. Store value in NN.*
220	OUTPUT 723;"CLRDSP;";	*Clear graphics from memory.*
230	OUTPUT 723;"PA 400,500;";	*Set pen at (400,500).*
240	OUTPUT 723;"MPY NN,NN,1000;";	*Convert queried value to milliamperes.*
250	OUTPUT 723;"DSPLY NN,7,4;";	*Display value on analyzer display.*
260	OUTPUT 723;"RELHPIB;";	*Release control of HP-IB*
270	OUTPUT 723;"^;"	*End definition of user-defined function.*
280	OUTPUT 723;"KEYDEF 7,VDM,VDM;"	*Assign function to user-defined key, called* VDM.
290	LOCAL 723	*Place analyzer in local control. Press* [USER]*, then press the user-defined key,* VDM*, to communicate with the multimeter.*
300	SEND 7;UNT UNL DATA	*Releases control of the HP-IB and clears HP-IB of all talkers and listeners.*

Description The ENTER command establishes the analyzer as a controller on HP-IB. Use ENTER to request data from an HP-IB device. Since HP-IB allows only one controller on HP-IB, the execution of ENTER must be either synchronized with computer operation or incorporated into user-defined functions that are executed with keys. Execute RELHPIB to discontinue analyzer control of HP-IB.

See Also RELHPIB, VARDEF

ERASE

Erase *all* memory, including internal memory and reserved memory, serial numbers, correction factors, and all analyzer settings.
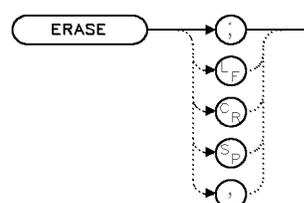
XERASE

Example

```
10   OUTPUT 723;"ERASE;"
```

Erase all memory, except factory calibration data stored in ROM.

Description The ERASE command erases *all* memory, except factory calibration data stored in ROM. ERASE destroys any stored conditions that may explain faulty operation and aid in troubleshooting. The information that is erased is listed below:

- Memory data protected by the PSTATE or PROTECT command.

- The contents of internal memory. Internal memory contains user-defined variables, traces, functions, keys, on-end-of-sweep algorithms, and files).

- The contents of reserved memory. Reserved memory contains serial-number information, correction factors, offsets, trace data, the limit-line table, and all analyzer settings.

- The contents of the input buffer holding unexecuted commands (including any commands immediately following ERASE).

ERASE also sets the predefined variable, NSTATE, equal to 2.

Use ERASE for security applications to destroy all previous instrument history.

After executing ERASE, (1) cycle power, or (2) press the (INSTR) key; or (3) press (DISPLAY), NEXT INSTR .

The ERASE command does not affect memory external to the spectrum analyzer, such as memory in the display, memory card, or HP-IB device.

ERR

Send numeric codes that describe the nature of errors.
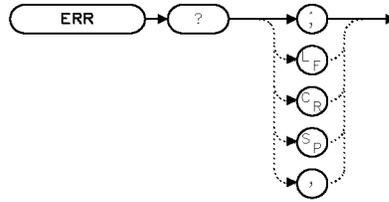
XERR

Example

20	CLEAR 723	*Initialize analyzer.*
30	OUTPUT 723;"IP;";	
40	ALLOCATE Error_numbers$[50]	*Create 50-element array in computer for the storage of any queried analyzer errors.*
50	OUTPUT 723;"SAVES 1;VIEW;"	*Send two faulty commands to analyzer. Execute SAVES using out-of-limit parameter. Use the VIEW command with incorrect syntax.*
60	OUTPUT 723;"ERR?;"	*Send analyzer codes to the computer.*
70	ENTER 723;Error_numbers$	*Store error codes in computer array.*

Description The ERR command sends numeric codes (to the computer) that describe the nature of errors, if any have occurred.

Error codes are stored in the error register as they occur. ERR sends codes as long as they are in the register. If none are present, ERR sends "0" (zero). Instrument preset (IP) clears the error register.

Query Response
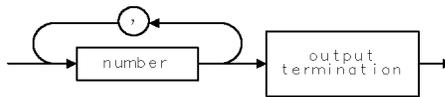
qgnumr

See Also IP

EXP

Divide by N, retain quotient, then store 10^{quotient}.

XEXP

Item	Description
Source or Destination Amplitude Range	0 to 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 to +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*. $\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example   60 OUTPUT 723;"EXP P_MW,MKA,10;"

Description The EXP command first divides the source by the scaling factor, then uses the quotient as an exponent of 10:

$$destination \ = \ 10^{(source/overscaling\ factor)}$$

Be sure data reflects one complete sweep when using TRA, TRB, TRC, or trace range for source by using SNGLS and TS.

FETCH   Query trace data before end-of-sweep.
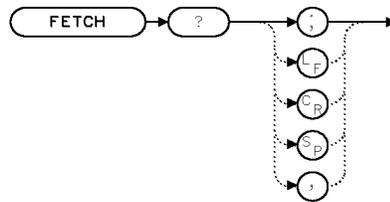
XFETCH

Example   `20  OUTPUT 723;"FETCH?;"`

Description   The FETCH command initiates a sweep and sends the contents of trace data in binary format, as it sweeps, to the computer.

The TRA, TRB, and TRC commands differ because they usually follow the TS command. As a result, the trace data is returned to the computer at the completion of a sweep.

Query Response

QFETCH
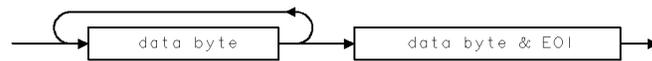
FFT

Execute fast Fourier transform.
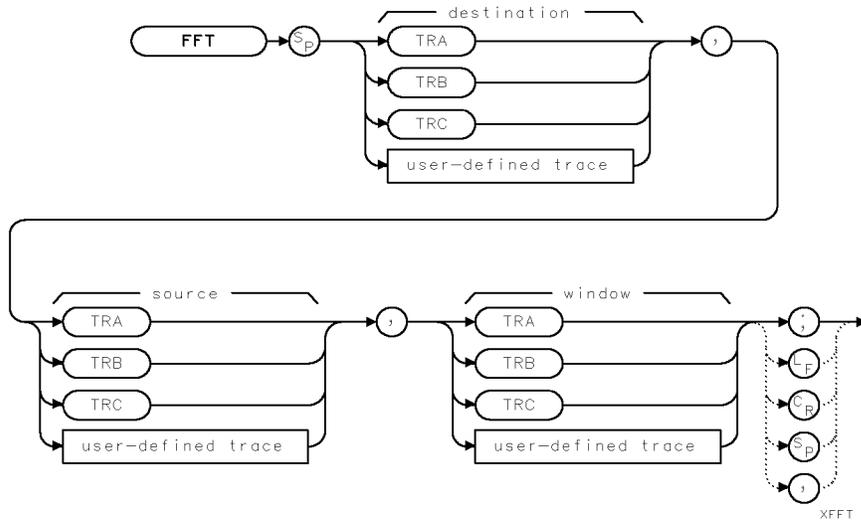
Item	Description
Window-Length Range	Equals source length.
Source-Length Range	1 - 2048. Default for TRA, TRB, TRC is 800.
Destination-Length Range	Equals source length.

Example Connect a 750 nm signal to analyzer input.

120	CLEAR 723	*Initialize analyzer.*
130	OUTPUT 723;"IP;";	
140	OUTPUT 723;"CLRDSP;";	
150	OUTPUT 723;"CENTERWL 750NM;"	*Set center wavelength.*
160	OUTPUT 723;"SP 0;ST 50MS;"	*Set span to zero. Set sweep time.*
170	OUTPUT 723;"TWNDOW TRC,FLATTOP;"	*Place the window algorithm, FLATTOP, into trace C.*
180	OUTPUT 723;"CLRW TRB;"	*Clear-write trace B.*
190	OUTPUT 723;"BLANK TRA;"	*Blank trace A.*
200	OUTPUT 723;"SNGLS;TS;"	*Activate single-sweep mode and sweep trace B.*

210 OUTPUT 723;"FFT TRA,TRB,TRC;"	*Weight trace B with trace C data, then perform Fourier transform. Store results in trace A.*
220 OUTPUT 723;"BLANK TRB;"	*Blank trace B.*
230 OUTPUT 723;"VIEW TRA;"	*Store and view trace A.*
240 OUTPUT 723;"MKN;"	*Place a marker at the center of trace A.*
250 OUTPUT 723;"HD;"	*Blank active function readout.*

Description The FFT command performs a discrete Fourier-series transform on the source-trace array, then stores the logs of the magnitudes of the results in the destination array as follows:

1. The source trace is lengthened with zero-amplitude elements if it is shorter than the destination trace.

2. The amplitude of the source trace is weighted according to the window algorithm.

3. The Fourier transform is performed.

Only magnitude components are stored. No phase or sign information is preserved in the results. Because of aliasing, the FFT command computes only the values of the even points of the destination trace. The odd values are obtained by interpolation. The source trace is sent unchanged. Four predefined window algorithms are available with the TWNDOW command. In addition, you can create your own with the TRA, TRB, or TRC command. For maximum precision, the peak values of user-created windows should approach 32,767. The window trace may contain negative values (to −32,768) but they do not increase weighting resolution. The values in the window trace are treated as fractional numbers. The average window value is computed and used to correct (normalize) the results in absolute units.

What to Remember When You Execute FFT

- Be sure trace data reflects one complete sweep measured with a 0 nm span. SNGLS and TS suggested.

- TWNDOW (Hanning, Hamming, uniform, flat-top) required when using TRA, TRB, or TRC. Use TWNDOW to select a weighting function (trace window) from one of the four available algorithms, or create your own with the TRA, TRB, and TRC commands.

- The lengths of the source and window traces must be equal and greater than four position units.

- Transform traces that have been measured during zero span mode.

Trace Windows   The FFT algorithm assumes the source trace array is one period of an
infinitely long string of concatenated, duplicate arrays. Thus, the beginning
and end elements of the source trace array must gradually diminish to
zero amplitude. If the end points of the original trace array are of different
magnitude, the resulting array series could contain discontinuities that
introduce high wavelength components into the Fourier transform. The
TWNDOW command offers four weighting algorithms that gradually reduce
the amplitude at the ends of the source-trace array to zero.

The TWNDOW command formats trace arrays with one of the four algorithms:
HANNING, HAMMING, UNIFORM, and FLATTOP. Each simulates a series
of equally spaced filters. The detected, spectral line traces the top of the
passband while moving from N Δ f to (N+1) Δ f.
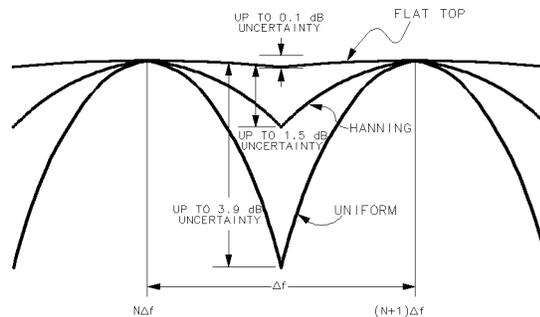
Hanning, Flat Top, and Uniform Trace Windows (Hamming not shown)

The amplitude and wavelength uncertainty of the Fourier-transformed
display depends on the choice of trace windows, and the analyzer sweep
time. Amplitude uncertainty is maximum when the spectral component
falls midway between the filter shapes. (See preceding figure.) Passbands
that are flatter in shape, like the FLATTOP filter, contribute less amplitude
uncertainty, but wavelength resolution and sensitivity are compromised.

The UNIFORM window algorithm has the least wavelength uncertainty and
greatest amplitude uncertainty. Its worst case accuracy uncertainty is -3.9
dB and its 3-dB resolution bandwidth is 60% of the HANNING bandwidth.
The UNIFORM window does not contain time-domain weighting and leaves
the data alone. Use it for transforming noise signals or transients that decay
within one sweep time period. The UNIFORM window yields the best

wavelength resolution, but also produces the highest side lobes for periodic signals.

The FLATTOP window has the greatest wavelength uncertainty of the windows, but it has outstanding side lobe suppression and amplitude flatness. Use FLATTOP to transform periodic signals.

The HANNING window is a traditional passband window found in most real-time analyzers. The HANNING window offers a compromise between the FLATTOP and UNIFORM windows. The HANNING window amplitude uncertainty is -1.5 dB and its 3-dB bandwidth is 40% of the FLATTOP bandwidth. Use the HANNING window when transforming periodic or random data.

The HAMMING window has a bandwidth between the HANNING and FLATTOP windows, with more uniform side-lobe suppression than the HANNING window, but less suppression than either HANNING or FLATTOP for signals farther out of the passband.

Interpreting Displayed Results The FFT results are displayed on the spectrum analyzer in a logarithmic scale. The frequency at the left graticule line is 0 Hz. The maximum frequency (F_{max}) at the right graticule line equals the inverse of the sample rate, divided by two. Thus, a 20-millisecond sweep time yields a display spanning 20 kHz:

$$F_{max} = 1/|\text{sample rate}| \div 2$$

$$F_{max} = 1/[|\text{sweep time}| \div N] \div 2$$

$$F_{max} = 1/[20 \text{ ms} \div 800] \div 2$$

$$F_{max} = 20 \text{ kHz}$$

Where

N equals number of elements in trace array containing amplitude information. 800 is assumed.

Fourier transforms of the predefined weighting algorithms are shown on the following graphs. Use these graphs to estimate resolution and amplitude uncertainty of a Fourier-transform display. Each horizontal division of the graphs equals 1/(sweep time) or $F_{max} \div 400$, which can be calculated from the previous equations.
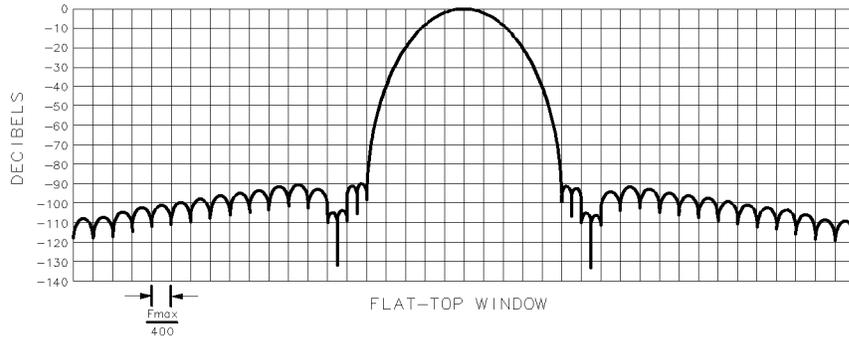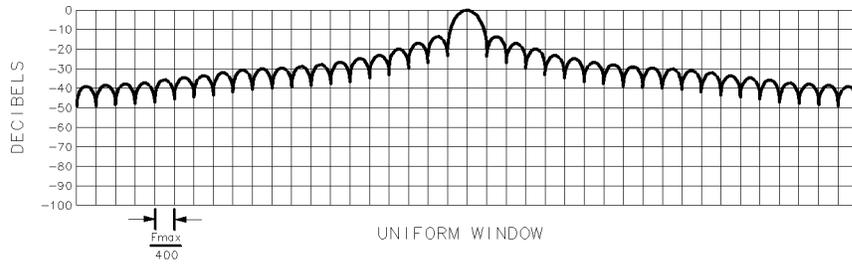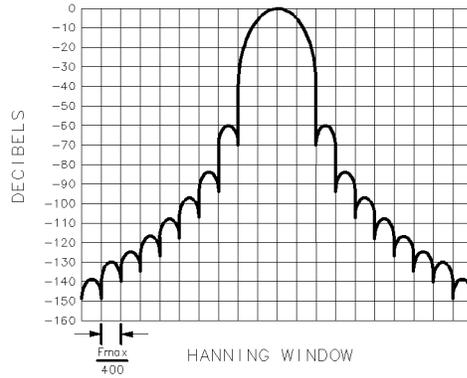
Fourier Transforms of Hanning, Flat Top, and Uniform Trace Windows

FFTKNL   Perform a 16-bit discrete Fourier transform.
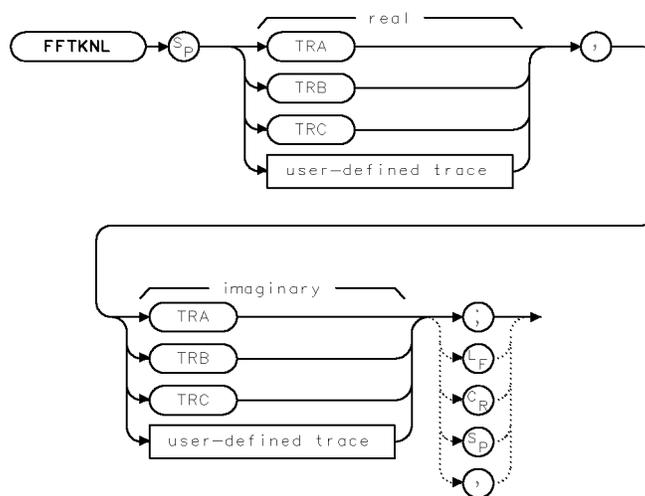
Example   `20  OUTPUT 723;"FFTKNL TRA,TRB;"`

Description The FFTKNL performs a discrete Fourier transform on the specified traces, overlaying them with the results. Both traces must be the same length and the length must be a power of two. The two traces represent the real and imaginary components of one complex-value trace. FFTKNL does no other normalizing, scaling, clipping, or magnitude determination.

FORMAT   Erase and format the currently selected memory device.
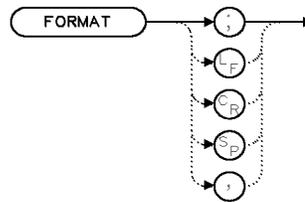
XFORMA

Example   20   OUTPUT 723;"FORMAT;"

Description The FORMAT command formats and erases the currently selected memory device. Internal memory is selected automatically when power is applied to the spectrum analyzer. To select another memory device, execute MSI. When internal memory is selected, FORMAT erases all unprotected files from internal memory. (Refer to the PROTECT and PSTATE commands for additional information.) When a memory card is selected, FORMAT erases all entries on the memory card. Use format to initialize new memory cards. Memory cards are formatted with the logical interchange format (LIF). When a disk drive is selected, FORMAT erases all entries on a floppy disk and initializes it. All new memory cards need to be formatted.

See Also MSI, PROTECT, PSTATE

FP_

Start the FP laser characterization measurement DLP.
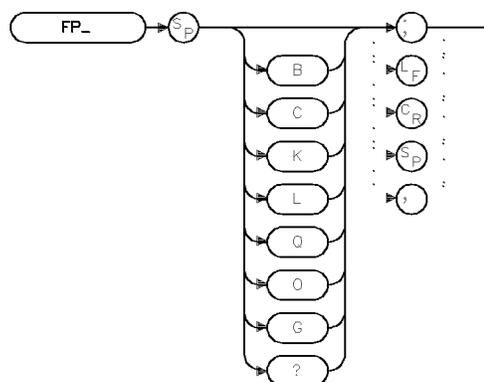
xckb11

Argument	Definition
?	Queries the DLP and returns the numerical equivilants of the parameters listed below:
	mean wavelength
	peak wavelength
	mode spacing (M)
	mode spacing (Hz)
	FWHM (full-width, half-maximum) or envelope bandwidth,
	if envelope distribution is selected
	peak amplitude
	total power
	sigma
B	Turns on or off the view of the peaks used in the calculations
C	Turn on or off the view of the distribution trace
K	Select Gausian distribution
L	Select Lorenzian distribution
O	Turn on or off the calculations

Q	Exit the measurement DLP
G	Select envelope distribution

Example

```
10  OUTPUT 723;"IP;";        Preset the instrument.
20  OUTPUT 723;"FP_;";       Start the FP characterization measurement.
30  OUTPUT 723;"FP_ C;";     Turn on, the view of the distribution trace.
40  OUTPUT 723;"FP_ K;";     Display the results in the Gausian distribution.
50  OUTPUT 723;"FP_ Q;";     Exit the FP characterization measurement DLP.
```

Description The FP_ command automatically measures the characteristics of a Fabry-Perot laser source. Refer to the *HP 71450B/1B/2B User's Guide* for additional information.

Query Response
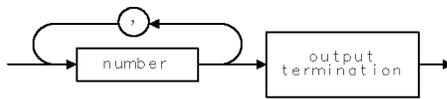
qgnumr

See Also FP_TH

FP_MKBW

Set FP laser envelope bandwidth amplitude.

x f p m k b w
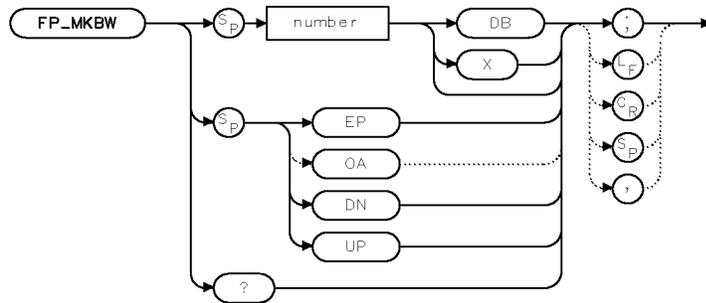
Item	Description
Preset State	−3.0 dB from peak

Example

10	OUTPUT 723;"IP;";	*Preset the instrument.*
20	OUTPUT 723;"FP_;";	*Start the FP measurement.*
30	OUTPUT 723;"FP_ C;";	*Display the distribution.*
40	OUTPUT 723;"FP_ G;";	*Select envelope distribution.*
50	OUTPUT 723;"FP_MKBW −6 DB;";	*Set bandwidth amplitude to −6 dB.*

Description The FP_MKBW command allows the bandwidth to be measured at selected amplitude levels down from the peak. If FP_MKBW is set to −6 dB, the envelope bandwidth is measured at 6 dB down from the peak. This command is only available when envelope distribution (FP_ G) has been selected.

Query Response

qgennum
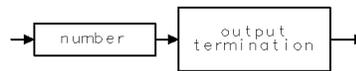
See Also FP_

FP_TH

Set or query the Fabry-Perot laser threshold amplitude.

ckb14
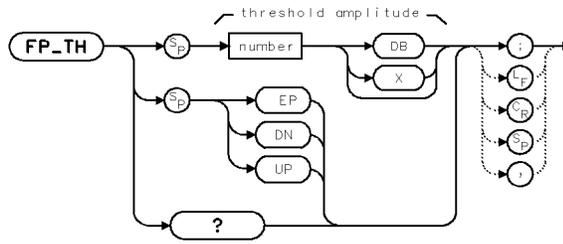
Item	Description
FP_TH?	Queries the DLP and returns the preset threshold value
Preset State	−90 dB from peak

Example

10	OUTPUT 723;"IP;";	*Preset the instrument.*
20	OUTPUT 723;"FP_;";	*Start the FP_ measurement DLP.*
30	OUTPUT 723;"FP_TH?;";	*Query the instrument for the current threshold amplitude setting of the Fabry-Perot laser.*
40	OUTPUT 723;"FP_TH −20 DB;";	*Set the threshold amplitude for measured peaks to −20 dB.*

Description

The FP_TH command allows measurements using only the peaks above the threshold. (All signals above the threshold are used in the calculations.) The threshold is relative to the peak mode. If FP_TH is set to −20 dB, only those peaks greater than 20 dBc are used in the power calculations. Refer to the *HP 71450B/1B/2B User's Guide* for additional information.

Query Response
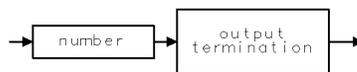
qgennum

See Also

FP_

FS   Activate full measurement range (wavelength).
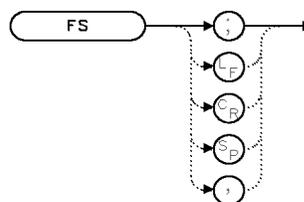
XFS

Example   `10  OUTPUT 723;"FS;"`

Description   The FS command activates the widest measurement range allowed by the optical spectrum analyzer, setting the start and stop wavelengths to their minimum and maximum values.

See Also   SS, STARTWL, STOPWL, CENTERWLSS, SP, CENTERWL, WLOFFSET

FUNCDEF

Define a list of analyzer commands that are executed as a user-defined function.
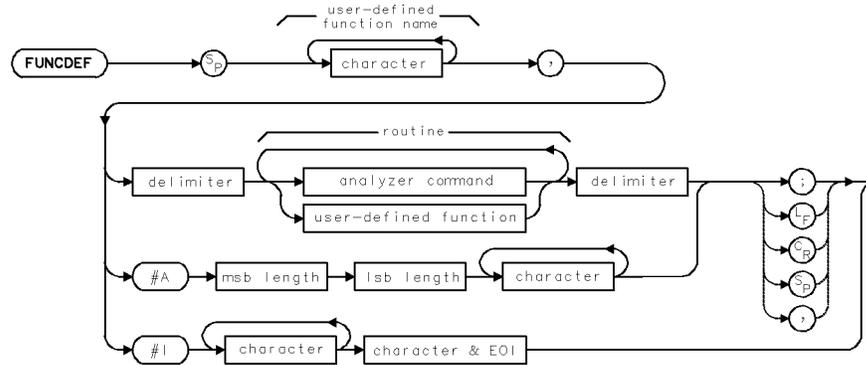
Item	Description
Function-Name Range	1 – 12 characters: A – Z, 0 – 9, and underscore |_|. First character must be a letter. Numbers must be preceded by underscore.

Example Create a user-defined function, then assign it to a user-defined key.

5	CLEAR 723	*Initialize analyzer.*
7	OUTPUT 723;"IP;"	
10	OUTPUT 723;"VARDEF H_SPAN,0;"	*Define user-defined variable equal to 0.*
20	OUTPUT 723;"FUNCDEF S_HIFT,^";	*Create user-defined function, called S_HIFT. Shift divides the span by two and adds the results to the center wavelength. This increases the measurement range. The "^" delimit the definition.*
30	OUTPUT 723;"DIV H_SPAN,SP,2;";	
40	OUTPUT 723;"ADD CENTERWL,CENTERWL,H_SPAN;";	
50	OUTPUT 723;"^;"	
60	OUTPUT 723;"KEYDEF 7,S_HIFT,%SHIFT_UP%;"	

Assign the user-defined function to a user-defined key, called SHIFT_UP.

```
70 OUTPUT 723;"DSPMODE CAT;"
```
Display catalog of currently selected user memory. Notice the listing includes the S_HIFT function and `SHIFT_UP` *key.*

```
80  OUTPUT 723;"CENTERWL 1300NM;SP 10NM;"
```
```
90  LOCAL 723
```
Press `USER` *to access* `SHIFT_UP` *key.*

```
100  OUTPUT 723;"S_HIFT;"
```
Execute function from the computer.

Description   The FUNCDEF command defines a routine consisting of spectrum-analyzer commands, assigns it a label, then stores the routine and its label in internal memory. FUNCDEF does *not* execute the routine. FUNCDEF *stores* the routine. *After* FUNCDEF is executed, the routine is executed whenever the *function name* is encountered. The function name and the list of commands it executes form a user-defined function. Once user-defined functions are stored, they can be executed with the front-panel keys, with other user-defined functions, or with computer programs (as you would other spectrum-analyzer commands).

Delete user-defined functions from internal memory with the DISPOSE command. Use the PROTECT command to prevent inadvertent erasure with DISPOSE. PURGE and FORMAT also erase user-defined functions. PROTECT does not prevent erasure or overwriting with the PURGE, LOAD, RCLD, or ERASE command.

Use ABORT to stop operation of user-defined functions. Use IF/THEN, REPEAT/UNTIL, or RETURN to alter their operation.

Do not use existing user-defined function names or reserved words as labels for user-defined functions. Do not execute a user-defined function within its FUNCDEF statement.

When using string- and block-data fields to transfer command definitions, be sure to use compatible terminators. Also, any block-field commands within the list must be the absolute type (#A-block), because the EOI terminator of the #I-block type terminates the entire FUNCDEF command prematurely.

Syntax to
Execute a
User-Defined
Function
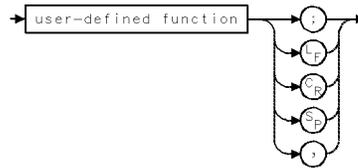
Query Syntax Requests the definition of the function.
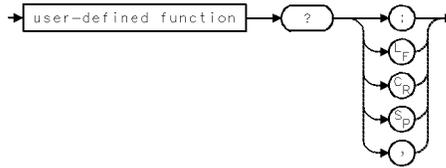
Query Response Returns the definition of the function.

See Also DISPOSE, FORMAT, LOAD, RCLD, PROTECT, PURGE

GATESWP

Enable time-gated measurements.
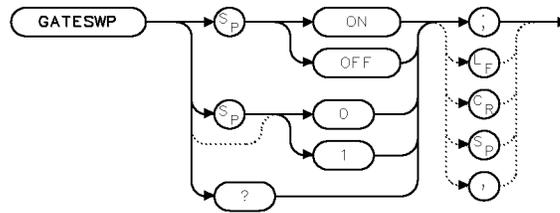
XGATE

Item	Description
Preset State	OFF

Example

```
10  OUTPUT 723;"GATESWP ON;"
```

Description The GATESWP command enables measurements to be gated by a signal that is input to the module's rear-panel **EXT TRIG IN** connector. When the external trigger input is at a high logic level (TTL), measurement data is used for the display. When the external trigger is low, the displayed sweep continues and trace points are displayed at the bottom of the screen. (The trace data value is loaded with a minimum value.) To properly display the data, use the MXMH command to place the trace in maximum hold.

During zero-span sweeps, gated sweeps can be configured so that the trace points are displayed only when the external trigger signal is high. That is, the sweep stops whenever the trigger signal is low. To use this feature, set the video bandwidth to manual mode (VB MAN). Be aware that this can result in two adjacent points being separated by an unknown amount of time.

Query Response

qgen0

See Also MXMH

GRAPH

Draw trace information inside a window created with the DWINDOW command.

XGRAPH

Item	Description
Trace-length Range	3 –2048 position units. Default for *TRA, TRB, TRC* is 800 position units.
Trace-Amplitude Range	For linear trace data, 0 – 10,000 measurement units.
	For logarithmic data, – 32,768 – + 32,767 measurement units.

Example

10 OUTPUT 723;"CLRDSP;"	*Clear graphics from memory. Reset origin to 0,0 and scale to 0 to 1000 for X and Y.*
20 OUTPUT 723;"IWINDOW 1000,600;"	*Reduce size of analyzer trace and graticule.*
30 OUTPUT 723;"MXMH TRA;"	*Retain maximum values of trace A.*
40 OUTPUT 723;"OR 0,600;"	*Set origin to 0,600.*
50 OUTPUT 723;"LINET 7;"	*Select line size for drawing.*
60 OUTPUT 723;"PD;GRID 100,30,10,10;"	*Draw grid.*
70 OUTPUT 723;"DWINDOW 1000,300,TRA;"	*Create a window scaled to display trace A.*
80 OUTPUT 723;"TS;"	*Sweep trace A.*
90 OUTPUT 723;"LINET;"	*Select solid line for drawing.*
100 OUTPUT 723;"PU;PA 1,TRA[1];"	*Place pen at starting point of trace A.*
110 OUTPUT 723;"PD;GRAPH TRA;"	*Place pen down and draw trace A in window (at top of display).*
120 OUTPUT 723;"CLRW TRB;"	*Measure with trace B (in active area at bottom of display).*
130 OUTPUT 723;"BLANK TRA;"	*Blank trace A.*

Description The GRAPH command draws previously stored measurement results, sized
according to the current scale units determined by the SCALE command,
or the DWINDOW command if GRAPH is used with DWINDOW. If no units
are specified, the default scale units are 0 to 1000 for both X and Y (set by
CLRDSP). Use PD to place pen down, except during reference-graphics mode.

The graph can be drawn at the current pen position, or can be positioned
with the OR command during the reference-graphics mode. For example,
in the preceding example, lines 100 and 110 can be replaced with `IT`
`1;GRAPH TRA;`. Notice that the PD command is required except during the
reference-graphics mode.

See Also TRCOND, CLRDSP, DWINDOW, SCALE, IT

GRAT

Turn on and off the display of the graticule.
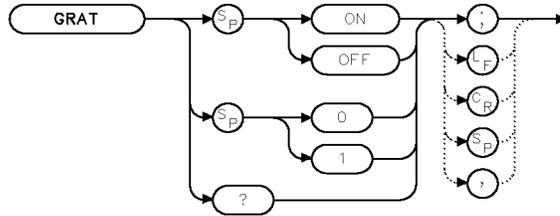
xgrat

Item	Description
Preset State	GRAT ON

Example

```
10   OUTPUT 723;"GRAT OFF;"
```
Turn off the graticule.

Query Response
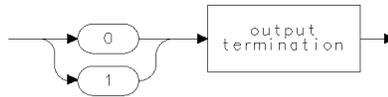
qgen0

GRATORDER Select reflection order of analyzer's internal diffraction grating.

XGTODR
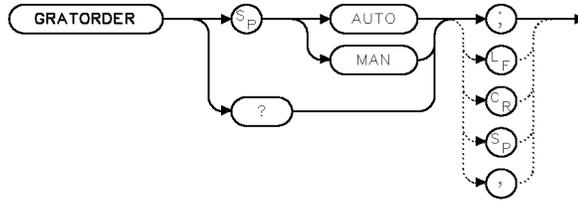
Item	Description
Preset State	AUTO
Wavelength Crossover	900 nm

Example   10 OUTPUT 723;"GRATORDER MAN;"   *Select only the first order.*

Description Use the GRATORDER command to manually select the first or second reflection order of the optical spectrum analyzer's diffraction grating. The diffraction grating is located in the optical spectrum analyzer's internal monochromator. Normally, the optical spectrum analyzer automatically selects the proper reflection order. The following list describes the conditions you can expect:

- If GRATORDER is set to MAN mode, the analyzer is locked into first-order diffraction grating.

- If GRATORDER is set to AUTO mode, and the stop wavelength is less than 900 nm (RES BW = 10 nm), the analyzer selects first-order diffraction grating.

- If GRATORDER is set to AUTO mode, and the stop wavelength is less than 900 nm (RES BW < 10 nm), the analyzer selects second-order diffraction grating.

Whether 1st or 2nd order diffraction-grating mode is selected depends on the following instrument conditions:

Full λ span .... 1st order
Stop $\lambda > 900$ nm (RES BW $\neq$ 10 nm) .... 1st
Stop $\lambda < 900$ nm (RES BW $=$ 10 nm) .... 1st order
Stop $\lambda < 900$ nm (RES BW $\neq$ 10 nm) .... 2nd

The optical spectrum analyzer never switches the order during a sweep. In full span (1st order), the monochromator loss near 600 nm will be greater than if 2nd order is used.

Query Response
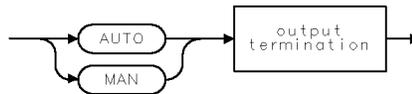
qgam

GRATSCRL

Controls the positioning of the displayed horizontal graticule lines.
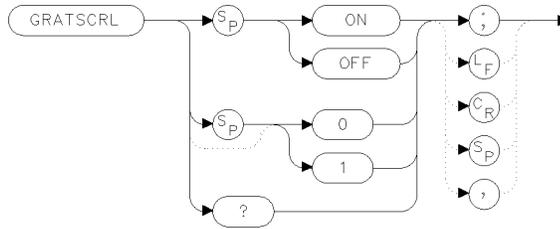
xgrscl

Item	Description
Preset State	GRATSCRL OFF

Example 100 OUTPUT 723;"GRATSCRL ON;"

Description When a log scale is displayed, the GRATSCRL command positions the horizontal graticule lines on multiples of the current dB-per-div scale. When GRATSCRL is set to off, the horizontal graticule lines are evenly spaced between the bottom and top screen values.

When a linear display scale is used, the horizontal graticule lines are evenly spaced between the bottom and top screen values regardless of the value of GRATSCRL.

Query Response

qgen0

GRID

Draw a grid on the analyzer display.

XGRID

Item	Description
X-Dimension Range	Affected by SCALE command. Default range is 0 – 1000.
Y-Dimension Range	Affected by SCALE command. Default range is 0 – 1000.
Number-of-X-Boxes-Range	Depends on scale resolution.
Number-of-Y-Boxes-Range	Depends on scale resolution.
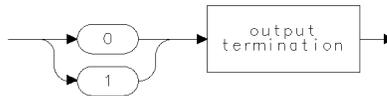
Example

40	OUTPUT 723;"GRAT OFF;"	*Turn off the graticule.*
50	OUTPUT 723;"SCALE 112,911,17,338;"	*Define scale.*
60	OUTPUT 723;"OR 112,48;"	*Specify origin.*
70	OUTPUT 723;"LINET 7;"	*Select dotted line for drawing.*
80	OUTPUT 723;"PD;GRID 160,64,5,4;"	*Place pen down. Draw large grid.*

Description   The GRID command draws a grid on the analyzer display where the X and Y dimensions specify the horizontal and vertical length of a single box in the grid. Number of X boxes and Number of Y boxes specify the number of boxes on the X and Y axis of the grid. The grid is positioned at the current pen position or at the item origin during the reference-graphics mode. Use PD to place pen down before using GRID. GRID cannot be executed after `DWINDOW ON`

See Also   SCALE, PEN, IT, OR, PA, PR, LINET

HD

Blank the active function readout and disable front-panel numeric data entry.
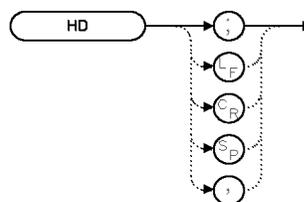
XHD

Example

120	CLEAR 723	*Initialize analyzer*
130	OUTPUT 723;"IP;"	
140	OUTPUT 723;"CENTERWL 850NM;"	*Set center wavelength. Notice that the active function area (left side of graticule) displays the center wavelength value.*
150	OUTPUT 723;"WAIT 4S;"	*Wait 4 seconds*
160	OUTPUT 723;"HD;"	*Blank the active function readout.*

Description The HD command blanks the active function readout and disables the front-panel numeric keypad, step keys, and knob, until another active function is executed.

ID

Query the model number (identification) of the system-controlling module.
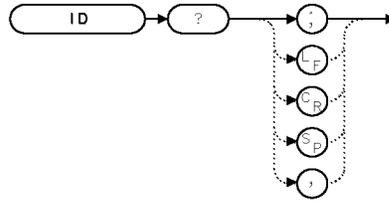
XID

Example

130	OUTPUT 723;"ID?;"	*Query value*
140	ENTER 723;N$	*Assign value to computer string variable.*

Query Response
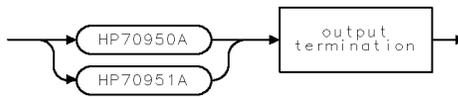
QID

IF/THEN

Execute a list of commands if conditions of your choosing are met.
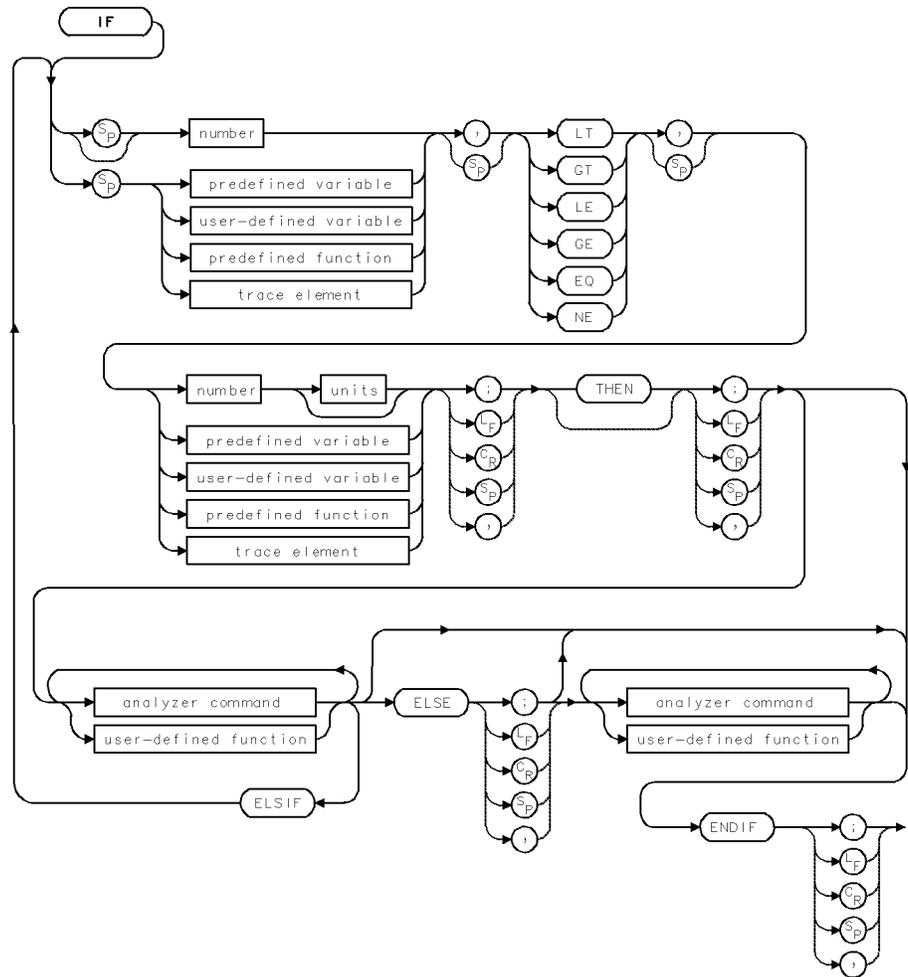
Example Print a message depending on the level of a measured signal.

```
20  CLEAR 723                                        Initialize analyzer.
30  OUTPUT 723;"IP;"
40  OUTPUT 723;"SNGLS;"                              Activate single-sweep mode.
50  OUTPUT 723;"PA 250,500;"                         Set pen position
60  OUTPUT 723;"SP 50NM;CENTERWL 750NM;"             Set measurement range.
70  OUTPUT 723;"TS;MKPK HI;"                         Measure, then mark highest signal.
80  OUTPUT 723;"IF MKA,LT,-10.5DBM;THEN;TEXT @FAIL@
90  OUTPUT 723;"ELSIF MKA,GT,-9.5DBM;THEN;TEXT @FAIL@;"
100 OUTPUT 723;"ELSE TEXT @LEVEL OK.@;ENDIF;"        If signal level is between −10.5 and −9.5 dBm,
                                                     print "LEVEL OK." If otherwise, print "FAIL"
```

Description The IF/THEN/ELSIF/ELSE/ENDIF commands compare the values of two operands according to the following conditions:

- Less than (LT)
- Greater than (GT)
- Less than or equal to (LE)
- Greater than or equal to (GE)
- Equal to (EQ)
- Not equal to (NE)

The operations are as follows:

If the operand values satisfy the specified condition, the analyzer commands that follow "THEN" are executed. If the specified condition is not satisfied, the analyzer commands that follow "ELSE" or "ELSIF" are executed. The command "THEN" is a no-operation function provided for syntax clarification.

The "IF" and "ELSE" commands delimit two operands and their comparison condition. "ELSIF" delimits an alternate comparison test. "ELSE" and "ENDIF" delimit analyzer commands that are executed if the condition(s) are not satisfied.

"ENDIF" ends the comparison construct.

IGEN

Turn on the current source.

N O T E

This command is valid only if Option 001, current source, is installed.
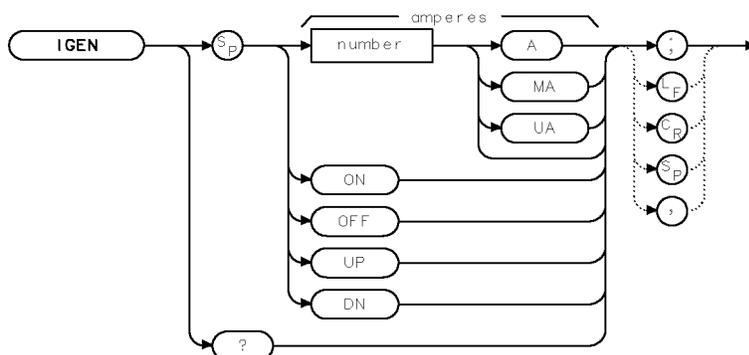
XIGEN

Item	Description
Range	±200 mA (but limited by IGENLIMIT)
Preset State	OFF
Duty Cycle	0.1% to 100%
Duty Cycle Preset State	100%

Example

10	OUTPUT 723;"IGEN 0.100;"	*Set the current source to +100 mA.*
20	OUTPUT 723;"IGEN ON;"	*Turn the current source on.*

Description The IGEN command turns on the optical spectrum analyzer's current source. The output of the source is available at the rear-panel **CURRENT SOURCE** connector. When using IGEN to set the current level of the source, enter current values in amperes. The current level applies for both steady state and pulsed operation. (Use IGENDTYCY or IGENPW to enter pulsed mode.)

The maximum IGEN current available at the output connector is limited by the IGENLIMIT setting. If a value is sent for IGEN that is greater than the IGENLIMIT setting and the analyzer is queried, the value returned is the IGENLIMIT value.

Query Response
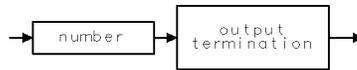
qgennum

See Also IGENDTYCY, IGENLIMIT, IGENPW

IGENDTYCY

Set the duty cycle of the current generator pulse mode.

N O T E

This command is valid only if Option 001, current source, is installed.
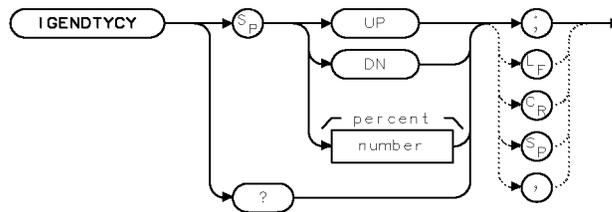
XIGDTY

Item	Description
Range	0—100% (but limited by IGENLIMIT)
Preset State	100%

Example   `10  OUTPUT 723;"IGENDTYCY 50;"`   *Set the duty cycle to 50%.*

Description The IGENDTYCY command selects the duty cycle of the optical spectrum analyzer's current source. The maximum current available at the output connector is limited by IGENLIMIT. Use IGEN to turn on the current source. Normally, the current source is set for a steady dc state (100% duty cycle.) You can set the duty cycle from 0 to 100 percent.

Query Response

```
  ┌──────────┐   ┌──────────────┐
→─┤  number  ├──→┤    output    ├──→
  └──────────┘   │ termination  │
                 └──────────────┘
```
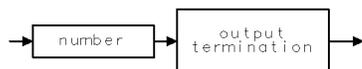
qgennum

See Also IGEN, IGENLIMIT, IGENPW

IGENLIMIT   Set safety limit on current source's output.

N O T E

This command is valid only if Option 001, current source, is installed.
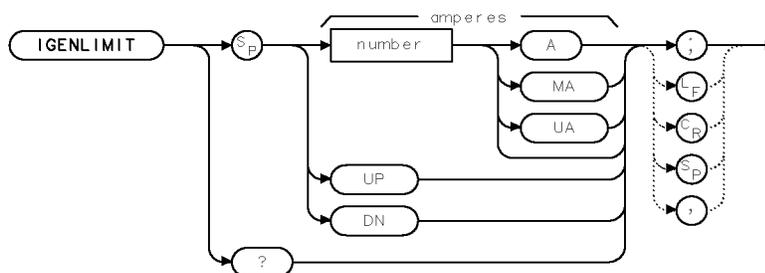
XIGLMT

Item	Description
Range	±200 mA
Preset State	±50 mA

Example   `10  OUTPUT 723;"IGENLIMIT 0.10;"`   *Set the limit to ±100 mA.*

Description   The IGENLIMIT command sets an absolute limit for the current source. This limit protects a test device from damage if the current setting is too high. If the current setting for the IGEN command is too high, the IGENLIMIT current value is used. When using IGENLIMIT, enter current limit value in amperes (0 to 0.20).

Query Response

```
number → output termination →
```
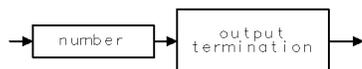
qgennum

See Also   IGEN, IGENDTYCY, IGENPW

IGENPW

Set the pulse width of the current generator's pulse mode.

N O T E

This command is valid only if Option 001, current source, is installed.
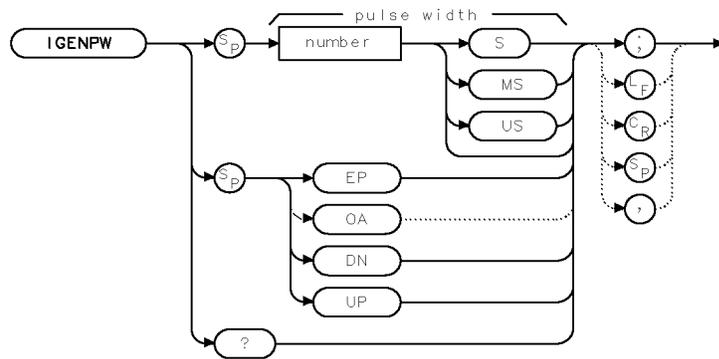
Item	Description
Range	1.0 μs to 6.5 ms
Preset State	20 μs

Example

```
10  OUTPUT 723;"IGENPW 1.0US;"
```

Description The IGENPW command selects the pulse width of the optical spectrum
analyzer's current source. Use IGEN to turn on the current generator.
Normally, the current source is set for a steady dc state. You can set the pulse
width from 1.0μs to 6.5 ms. Use IGENDTYCY to set the pulse's duty cycle.

If VB is modified after IGENPW and IGENDTYCY command values are set,
use the MSG command to confirm that the UNCAL message is not present,
indicating that the amplitude measurement is valid.

Video Bandwidth Setting vs Minimum Pulse Width

VBW	Minimum Pulse Width
2 MHz	20.0 μs
800 kHz	20 μs
80 kHz	20 μs
9 kHz	71 μs
800 Hz	800 μs

Query Response
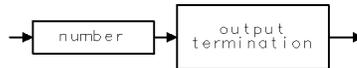
qgennum

See Also   IGEN, IGENDTYCY, IGENLIMIT, MSG

INSTMODE

Select optical spectrum analyzer measurement mode.
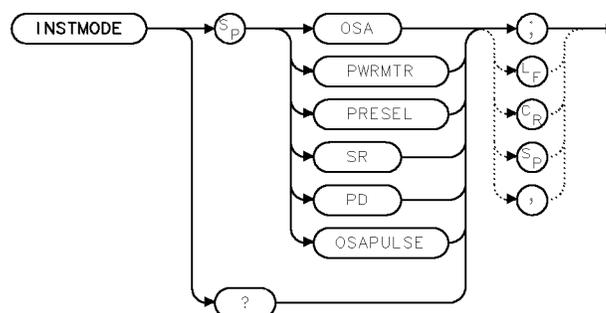
ximode

Item	Description
Requirement	HP 70951A module
Preset State	OSA

Example   `10  OUTPUT 723;"INSTMODE PRESEL;"`

Description The INSTMODE command selects the instrument's operating mode. For HP 71451B instruments all six of the following selections are available. On HP 71450B/2B instruments, only the OSA, SR, and PULSE selections are available:

- OSA (Optical Spectrum Analyzer). This is the preset state and it uses a combination of monochromator and photodiode flatness correction data.

- PWRMTR (Power Meter). The front-panel's PHOTODETECTOR INPUT connector is enabled. Only photodiode flatness correction data is used. Signals connected to this input are detected without passing through the optical spectrum analyzer's monochromator. The AMETER function is turned on. The trace shows data as it is taken.

- OSAPULSE. This option places the optical spectrum analyzer in an operation mode that is optimized for making accurate fast pulse measurements. The range of the internal transimpedance amplifier is

limited in order to select wide bandwidth and fast settling characteristics.
One example of using this mode is measuring the signal-to-noise ratio (SNR)
of EDFA systems by measuring the amplified spontaneous emission (ASE)
level immediately (10 μs) after the modulation is turned off.

WARNING

In PRESEL, SR, and PD modes, the input signal passes through the
monochromator and is emitted from the front-panel's MONOCHROMATOR
OUTPUT connector. The power at the MONOCHROMATOR OUTPUT
may be attenuated very little from the power being applied at the
MONOCHROMATOR INPUT.

- PRESEL (Preselector). The front-panel's MONOCHROMATOR INPUT and
  MONOCHROMATOR OUTPUT connectors are enabled. Monochromator
  and diode flatness correction data only are used. Since in this mode a
  complete sweep is first taken in OSA mode to place a trace on the screen
  that represents the expected output power, both monochromator and
  photodiode correction data is used. In this mode, the optical spectrum
  analyzer's monochromator can be used to preselect the input optical signal.

- SR (Stimulus-Response). The front-panel's PHOTODETECTOR INPUT
  and MONOCHROMATOR OUTPUT connectors are enabled. Only
  monochromator and photodiode flatness correction data are used.

- PD (Photo-diode Test). The front-panel's MONOCHROMATOR INPUT,
  MONOCHROMATOR OUTPUT, and PHOTODETECTOR INPUT connectors
  are enabled. In PD mode, a normalization sweep is first taken using
  correction data for the internal photodiode only. Measurements are
  then made without correction data being applied. Refer to the following
  information when making a PD test:

  1. Connect a jumper between the MONOCHROMATOR OUTPUT and the
     PHOTODETECTOR INPUT.

  2. Send the STORREF THRU command to store the light source flatness
     value in trace B.

  3. Connect the MONOCHROMATOR OUTPUT to the device-under-test and
     the diode output to the rear-panel TRANS-Z INPUT.

  4. Send the PDMEAS ON command to subtract the light source flatness
     (trace B) from the detected signal in trace A.

  5. The normalized trace results are the diode responsivity where
     0 dBm = 1 A/watt.

N O T E

In both SR and PD mode, the use of a broadband spontaneous emissions source is required. Examples of broadband sources include an LED or white-light source.

Query Response
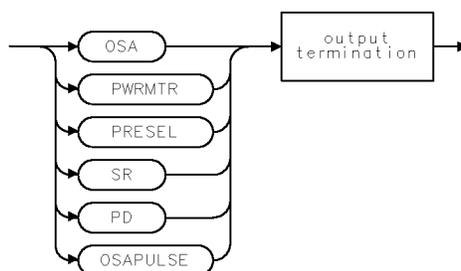
qimode

See Also   ADCTRG, ADCTRGDLY, CALDATA, CORSEL, OPTSW, MKTUNE, PDMEAS, STORREF, TRNSZLOCK

INT

Compute integer value(s) in source.

XINT

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	—32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Obtain the integer value of a number.

```
130   OUTPUT 723;"ST 2.5S;TS;"        Set sweep time. Sweep.
140   OUTPUT 723;"INT ST,ST;"         Reset the sweep time using the integer value
                                      of the previous sweep time.
150   OUTPUT 723;"TS;"                Sweep with the new sweep time.
```

Description The INT command calculates the largest integer which is less than or equal to the value of the source, then stores the integer in the destination. The MOV and INT commands are the same for trace data, since trace data is in integer form (measurement units) already. Use SNGLS and TS to ensure trace data reflects one complete sweep when using TRA, TRB, TRC, or trace range for source.

IP

Perform an instrument preset which sets all functions to their preset state.
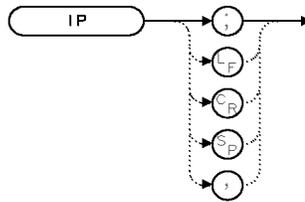
XIP

Example   `10  OUTPUT 723;"IP;"`

Description The IP command presets the analyzer settings placing the spectrum analyzer in a known state.

Programming Commands Affected by Instrument Preset

Command	Preset State/Response	Command	Preset State/Response
ACTPARM	OFF	GATESWP	OFF
ADCTRG	OFF	GRAPH	Blank graphics trace.
ADCTRGDLY	$2\mu s$	GRAT	GRAT ON
ADCTRGSYN	AUTO	GRATEDSW	FALSE
AMB	AMB OFF	GRATORDER	AUTO
AMBPL	AMBPL OFF	GRID	Blank grid.
AMC	AMC OFF	HD	HD
AMPCOR	AMPCOR OFF	IGEN	OFF
ANNOFF	Annotation is reenabled after IP.	IGENDTYCY	100%
ANNOT	ANNOT ON	IGENLIMIT	50 mA
APB	APB OFF	IGENPW	$20\mu s$
AUNITS	AUNITS AUTO	INSTMODE	OSA
AUTOMDB	10 dB/div AUTO	IT	IT 0
AUTOMSP	100 nm AUTO	IWINDOW	IWINDOW 1000,1000
AUTORANG	TRUE	LG	LG 10DB
BLANK	BLANK TRB;BLANK TRC;	LIMIBEEP	LIMIBEEP OFF
CALCOR	CALCOR ALL	LIMIEDIT	OFF
CENTERWL	Set to center of measurement range.	LIMIHALF	LIMIHALF UPPER
CHOP	ON	LIMIREL	LIMIREL OFF
CLRDSP	CLRDSP	LIMITEST	LIMITEST OFF
CLRW	CLRW TRA	LINES	LINES ON
CONTS	CONTS	LINET	LINET 0
DEBUG	DEBUG OFF	LN	Off
DL	DL 0 DBM; DL OFF	MDS	MDS W
DSPLY	Blank displayed variable.	MEASURE	MEASURE SA
DSPMODE	DSPMODE NORMAL	MK	Blank graphics marker.
DWINDOW	DWINDOW OFF	MKA	Change value.
ERR	Clear error register.	MKACT	MKACT 0
FETCH	Interrupted	MKAL	MKAL 0
FS	Full Span		

Programming Commands Affected by Instrument Preset (continued)

Command	Preset State/Response	Command	Preset State/Response
MKAR	`MKAR 0`	ROFFSET	`ROFFSET 0`
MKBW	`MKBW OFF`	SCALE	`SCALE 0,1000,0,1000`
MKBWA	`-3 dB`	SENS	`AUTO`
MKN	Change value.	SNGLS	OFF
MKNOISE	`MKNOISE OFF`	SP	Set to maximum measurement range.
MKOFF	Blank markers.	SS	Set to SPAN/10
MKP	`MKP 401`, or center of trace	ST	`ST AUTO`
MKPAUSE	`MKPAUSE 0`	STARTWL	Set to beginning of measurement range.
MKPX	`MXPX 3DB`	STOPWL	Set to end of measurement range.
MKREAD	`MKREAD AUTO`	TDF	`TDF P`
MKT	Change value.	TEXT	Blank text.
MKTRACK	`MKTRACK OFF`	TH	`TH -90DBM; TH OFF.`
MKTV	`MKTV 5DB`	TITLE	Blank title.
MKTYPE	`MKTYPE NORMAL`	TM	`TM FREE`
MKWL	Change value.	TP	`TP 0`
MSG	Blank message.	TRCOND	Change Value
ONEOS	`ONEOS OFF`	TRDEF	Traces A, B, C set to 800 points
ONMENU	OFF	TRDSP	`TRDSP TRA,ON`
ONMKR	OFF	TRSTAT	Write, on
ONUSER	OFF	VAVG	`VAVG 100; VAVG OFF`
ONWINDOW	OFF	VB	`VB AUTO`
OR	`OR 0,0`	VIEW	Off
PA	`PA 0,0`	VTH	`VTH 3DB`
PAUSE	Interrupted	VTL	`VTL -100 DBM`
PEN	`PEN 0`	VW	Blank item.
PERSIST	`PERSIST OFF`	WAIT	Interrupted.
PU	`PU`	WLLIMIT	`WLLIMIT ON`
RB	`RB AUTO`	WLOFFSET	`WLOFFSET 0M`
RBR	`RBR .01`	ZERO	AUTO
RL	`RL -10DBM`		
RLPOS	`RLPOS 9`		

IT   Assign a number to a series of graphics commands for future recall or modification.
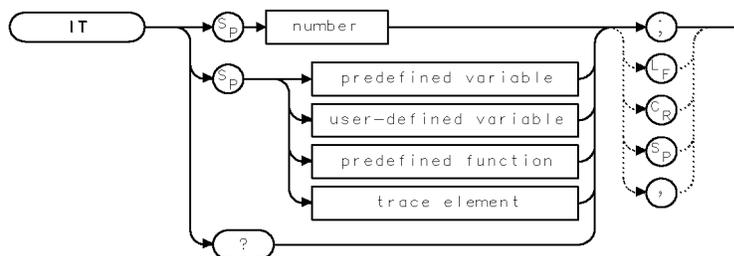
Item	Description
Item-Number Range	0 – 50, depending on model of display.
Preset State	IT 0

Example

150	CLEAR 723	*Initialize analyzer.*
160	OUTPUT 723;"IP;"	
170	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
180	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
190	OUTPUT 723;"IT 1;VW OFF;"	*Address item 1 and blank it. This way, item 1 is not displayed while it is being created.*
200	OUTPUT 723;"TEXT$This is ";	*Define item 1 as text.*
210	OUTPUT 723;"the contents of ";	
220	OUTPUT 723;"item 1.$;"	
230	OUTPUT 723;"IT 2;VW OFF;"	*Address item 2 and blank it.*
240	OUTPUT 723;"GRID 10,20,10,5;"	*Define item 2 as a grid.*
250	OUTPUT 723;"IT 1;OR 300,900;"	*Address item 1 and move it to (300,900) in current scale units.*
260	OUTPUT 723;"VW ON;"	*Display item 1.*
270	OUTPUT 723;"IT 2;OR 500,500;"	*Address item 2 and move it to (500,500).*
280	OUTPUT 723;"VW ON;"	*Display item 2.*
290	OUTPUT 723;"WAIT 2;"	*Wait 2 seconds.*
300	OUTPUT 723;"IT 1;DELETE;"	*Delete item 1 from memory.*

Description   The IT command activates the reference-graphics mode by assigning a
nonzero number to a series of graphics commands that form a shape. These
shapes are called items. Use IT to blank, redraw, reshape, or reposition items.
Graphics drawn when the reference-graphics mode is off cannot be modified
at a later time. Execute IT 0 to turn the mode off. Manipulate an item by
first activating it with IT, then modify it with the commands listed in the
following table.

Commands that Modify Graphics Items

Command	Function
VW OFF |view item off|	Blank item
VW ON |view item on|	Redraw item
OR |origin|	Offset item position
TP |pointer|	Change item shape
DELETE	Delete item from memory

When assigned a non zero number, items are stored in memory for future
display and manipulation. Execute CLRDSP to clear items from memory.
(DISPOSE does not clear items from memory.) In addition to point-by-point
drawing, items can be used to display text, grids, markers, or trace
information. The commands associated with these tasks are divided into
groups listed in the following table.

Commands Assignable to Items with the IT command.

Task	Description
Plot absolute, point-by-point drawing	PA, PD, PU
Plot-relative, point-by-point drawing	PR, PD, PU
Display trace data	GRAPH
Draw grid	GRID
Display text	TEXT,DSPLY
Display graphics marker	MK

A single item can be assigned to do only one task. For example, a single item
cannot draw a box and display text. In this case, two items are required, one
for the box and another for the text. Mixing item types generates a display
error indicated by "E" in the display status box. For example, IT 1; PA
0,0,100,100;MK 100,100; generates an error in the display.

Query Response

```
  ┌──────────┐    ┌──────────────┐
──▶│  number  │──▶│    output    │──▶
  └──────────┘    │ termination  │
                  └──────────────┘
```
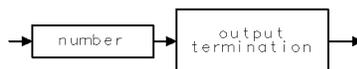
qgennum .

See Also CLRDSP, OR, SCALE, VW, TP, DELETE, PA, PR, PD, PU, GRAPH, GRID, TEXT, DSPLY, MK

IWINDOW

Change size (but not scale) of displayed measurement results.
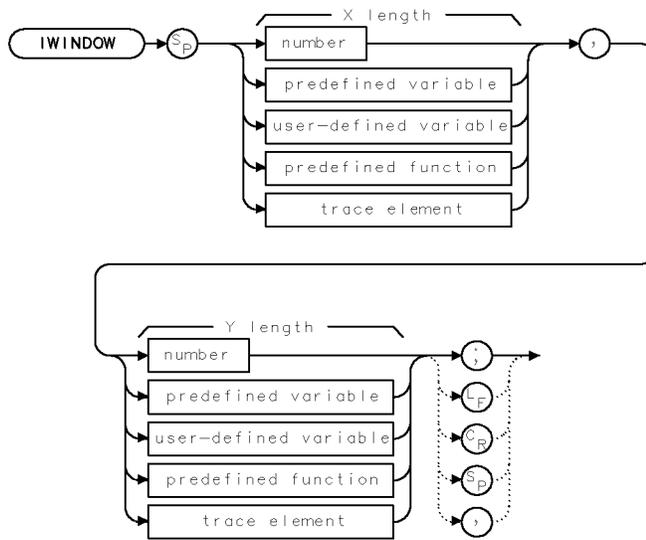
Item	Description
X-Length Range	−32,768 – 32,767, in units set by SCALE command.
Y-Length Range	−32,768 – 32,767, in units set by SCALE command.
Preset State	`IWINDOW 1000,1000`

Example

`120 OUTPUT 723;"IWINDOW 700,800;"`   *Define instrument window to be a 700 X 800 dimension in current scale units. Notice annotation disappears when there is not enough room to display it.*

Description   The IWINDOW command varies the size of the graticule and measurement results on the analyzer display. Size is specified in the current scale units (SCALE). The new graticule is called an instrument window.

See Also   SCALE, CLRDSP, OP

KEYCLR   Clear keys currently accessed with (USER) key.
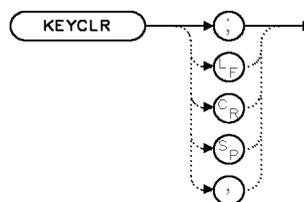
```
                              ┌─────────┐       ╭───╮
                              │ KEYCLR  │───────│ ; │──────▶
                              └─────────┘       ╰───╯
                                              ╭───╮
                                              │ L │
                                              │ F │
                                              ╰───╯
                                              ╭───╮
                                              │ C │
                                              │ R │
                                              ╰───╯
                                              ╭───╮
                                              │ S │
                                              │ P │
                                              ╰───╯
                                              ╭───╮
                                              │ , │
                                              ╰───╯
```
<div align="right">XKEYCL</div>

Example   10 OUTPUT 723;"KEYCLR;"

Description The KEYCLR command blanks the (USER) menu.

See Also KEYDEF, RCLU, READMENU, KEYPST

KEYDEF

Assign a label and user-defined function to a (USER) key.
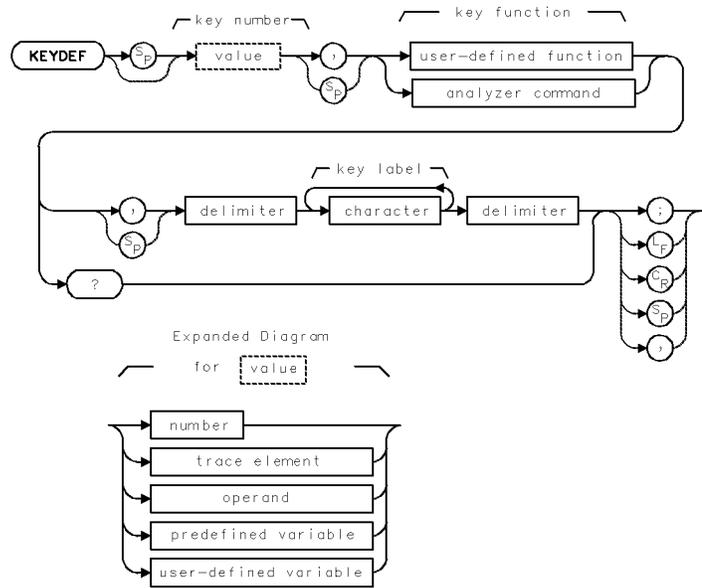
Item	Description
Key-Number Range |softkeys|	1 – 14
Key-Number Range |keypad|	15 – 29
Key-Label Range	2 lines of up to 7 characters each.

Example

```
10  OUTPUT 723;"FUNCDEF T_10,^";          Create user-defined function called T_10
20  OUTPUT 723;"CENTERWL 1300NM;";
30  OUTPUT 723;"SP 10NM^;"
40  OUTPUT 723;"KEYDEF 7,T_10,%TEST%;"    Define key, labeled TEST .
```

Description   The KEYDEF command creates a key that executes a user-defined function.
The key can be a softkey that is located in the (USER) menu, or it can be
a key that is located on an *undefined* custom keypad. The keys on the
normal recognized keypad cannot be redefined. Once a key is defined, the
user-defined function is executed whenever the key is pressed.

Error 2013

Error 2013 is displayed if a user-defined function is called in a KEYDEF before the function is defined
and stored.

Up to 14 user-defined softkeys can be displayed at one time. They are all
accessed with the (USER) key. The softkeys are numbered from 1 to 14. The
following figure shows the correspondence between softkey number and
softkey location. The softkey label can be up to 14 characters long, but if
longer than 7, the characters are divided into 2 lines of up to 7 characters
each. Additional characters can be incorporated into the label for control
codes. For example, a line feed may be used to specify which character
begins the second line of the label. Use the READMENU command as
another way to build menus of keys. Use the PROTECT command to prevent
inadvertent erasure or redefinition. KEYPST protects key #8 automatically.
Use PROTECT to turn off the protection.
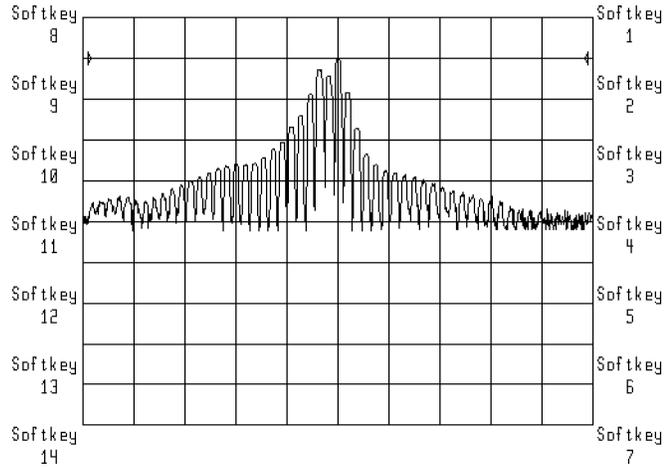
Key Numbers for Softkeys

A key on the custom keypad can only be assigned to a user-defined function if the keypad is unrecognized by the display. If you attempt to redefine the keys of a recognized keypanel, the built-in definitions for the keypanel will still be used. There are 15 possible keypad keys labeled 15 through 29. The following figure shows the numbering of the keypad keys. There is no label annotation for USER keys 15 through 29. Labels may be defined, but they will never be displayed.

Assigned Key Numbers for Custom Keypad
(As viewed from the front of keypad.)

15	18	21	24	27
16	19	22	25	28
17	20	23	26	29

Define all valid key presses

If you're building key definitions for a custom keypad, be sure to define all of the keypad positions which could be pressed by a key. For example, if a single key will be over both positions 15 and 18, both of these keypad positions should be defined to be the same function.

Query Response Sends the name of the user-defined function assigned to the key to the computer.

```
 ┌─────────────────────────┐   ┌──────────────┐
→┤  user—defined function  ├──→┤   output     ├──→
 └─────────────────────────┘   │  termination │
                               └──────────────┘
```

qkeyde

See Also KEYPST, PROTECT, READMENU, KEYCLR, SAVEU, RCLU

KEYPST   Preset (USER) keys to a default state.
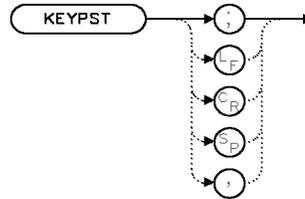
KEYPST

XKEYPS

Example   10 OUTPUT 723;"KEYPST;";   *Reset user-defined keys to their preset values. Push* (LOCAL) *to view the* (USER) *keys.*

Description The KEYPST command resets the menu of keys that is accessed with the (USER) key to contain these three keys only:

- DEFINE USR KEY - Define user-defined key.

- SAVE USER - Save menu of (USER) keys.

- RECALL USER - Recall menu of (USER) keys.

Executing **KEYPST** protects key #8, DEFINE USR KEY, automatically. Use PROTECT to turn off the protection.

See Also KEYDEF, KEYCLR, PROTECT

LED_

Start the LED_ (light-emitting diode) laser characterization measurement DLP.

ckb16

Argu-ment	Definition
B	Turns on or off the view of the integration window trace.
C	Turn on or off the view of the distribution trace.
K	Select Gausian distribution.
L	Select Lorenzian distribution.
Q	Exit the measurement DLP.
O	Turn on or off calculations.
?	Queries the DLP and returns numerical equivilants of the parameters listed below:

> mean wavelength
> peak wavelength from −3 dB points
> peak wavelength
> sigma
> full width, half maximum
> width at −3dB points
> total power
> peak spectral density

Example

10	OUTPUT 723;"IP;";	*Preset the instrument.*
20	OUTPUT 723;"LED_;";	*Start the LED_ characterization measurement.*
30	OUTPUT 723;"LED_ K;";	*Select Gausian distribution for the measurement results.*
40	OUTPUT 723;"LED_ Q;";	*Exit the LED_ characterization measurement DLP.*

Description The LED_ command automatically measures characteristics of a light-emitting diode laser source. Refer to "Measuring Signals," in the *Optical Spectrum Analyzer User's Guide* for additional information.

Query Response
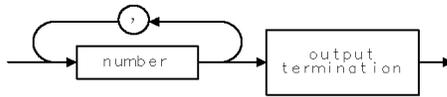
qgnumr

See Also AMPMKR, WLMKRR, WLMKRL

LG

Set the vertical logarithmic amplitude scale.
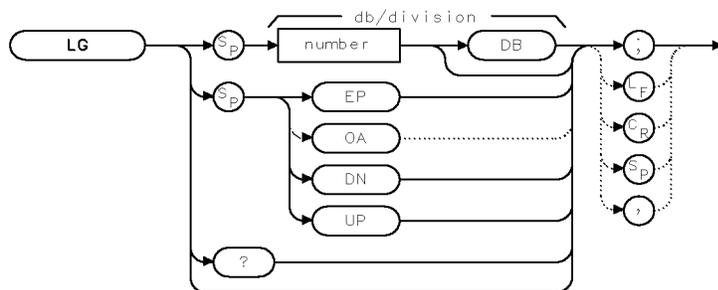
Item	Description
Scale Range	0.01 dB – 20 dB
Step Increment	1,2,5,10 sequence
Preset State	LG 10DB

Example

40	OUTPUT 723;"LG 20DB;"	*Set the log scale to 20 dB/division.*
50	OUTPUT 723;"LG EP;"	*Change the log scale with the front-panel keys. Enter a number, then press* ACCEPT VALUE

Description The LG command specifies the vertical graticule divisions as logarithmic units without changing the reference level. This is called the logarithmic display mode. If no value is specified, the logarithmic scale is 10 dB per division.

Query Response
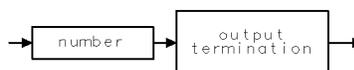
See Also LN

LIGHT

Turn internal white light source on or off.

NOTE

This command is valid only if Option 002, white light source, is installed.

---
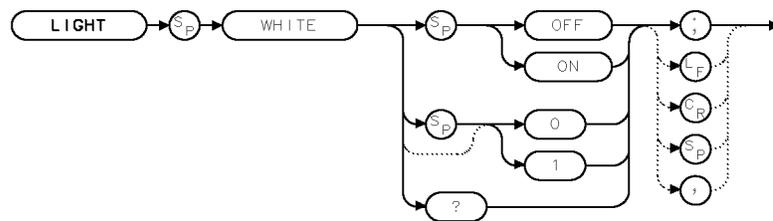
xlwhite

Example   10   OUTPUT 723;"LIGHT WHITE ON;"   *Turns on white light source.*

Description   The LIGHT command turns on or off the internal white light source. The output is available on the front panel.

The query returns the current state, on or off.

Query Response
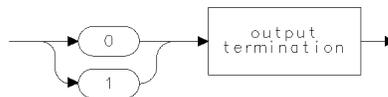
qgen0

LIMIAMP

Set the limit-line segment amplitude.
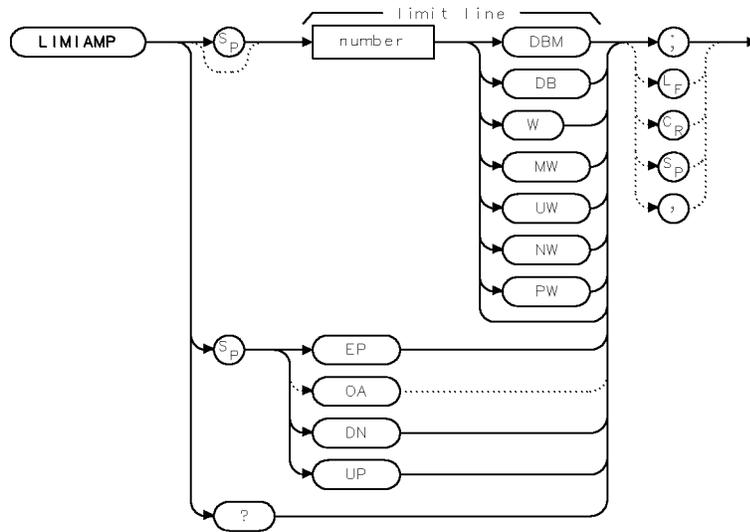
xlimia

Example   10 OUTPUT 723;"LIMIAMP -10DBM;"

Description The LIMIAMP command sets the amplitude of the currently active limit-line segment. The query sends the amplitude value of the current segment.

See Also LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

Query Response

qgennum

LIMIBEEP   Sound a beep each time the currently measured trace data exceeds test limits.

XLIMIB

Item	Description
Preset State	LIMIBEEP OFF

Description The LIMIBEEP command sounds a beep when trace data exceeds limit
line parameters during limit-line testing. See LIMILINE and LIMITEST for
a description of limit lines and limit-line testing, and see LIMIFAIL for an
alternative failure indicator. LIMIBEEP ON sounds a beep when a signal
response fails the test limit as the sweep progresses. The beep is sounded
once per sweep at the first failure point. LIMIBEEP OFF turns off the
function.

Query Response

qgen0

See Also LIMIAMP, LIMIDEL, LIMIWL, LIMILINE, LIMIRCL, LIMIREL, LIMISDEL,
LIMISEG, LIMITEST, LIMITYPE, and any commands that change trace data
values.

LIMIBOT

Activate the last segment of the limit-line table.
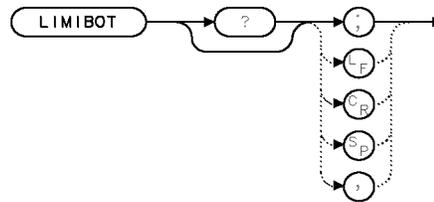
xlibot

Example

```
10  OUTPUT 723;"LIMIBOT;"
```

Description The LIMIBOT command specifies that the last segment of the limit line become the active segment. The query sends the number of the last segment. If there are 12 segments, the number returned is 12.

Query Response
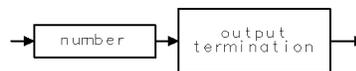
qgennum

See Also LIMIDEL, LIMIDONE, LIMIEDIT, LIMILINE, LIMIRCL, LIMISDEL, LIMISEG

LIMIDEL

Erase the contents of the limit-line table.
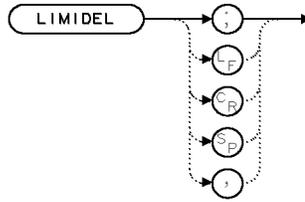
XLIDEL

Description When testing with limit-line parameters that reside in traces, execution of LIMIDEL is required before storing these limit-line parameters in the LIMIT_LO and LIMIT_HI trace arrays.

See Also LIMILINE, LIMITEST

LIMIDONE

Turn off the limit-line editor.
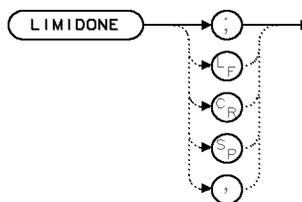
XLIDON

Example   10   OUTPUT 723;"LIMIDONE;"

Description The LIMIDONE command turns off the limit-line editor which removes the
limit-line table and editing softkeys from the display.

LIMIEDIT

Turn on the limit-line editor.
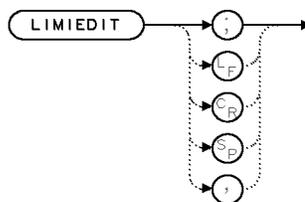
XLIEDT

Example   10   OUTPUT 723;"LIMIEDIT;"

Description The LIMIEDIT command activates the limit-line editor.

LIMIFAIL

Query limit-line test results.
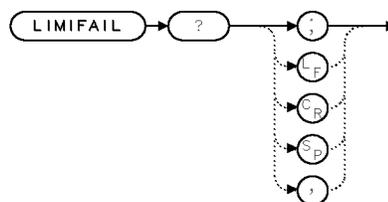
XLIFAL

Example   10 OUTPUT 723;"LIMIFAIL?;"

Description Execute **LIMITEST ON** before executing LIMIFAIL. The LIMIFAIL command returns a number based on the condition of the sweep at the moment your query is sent. You may send the query at anytime during the sweep, there is no need to wait for the trace to complete. The following numbers indicate the condition for the returned value:

- "0" indicates data has been compared to limits and does not exceed the limit-line parameters.

- "1" indicates that the data fell below the lower limit line, prior to your execution of the query.

- "2" indicates that the data exceeded the upper limit line, prior to your execution of the query.

- "3" indicates that the data exceeded both the upper and lower limit lines, prior to your execution of the query.

Query Response
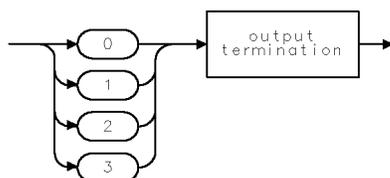
QLIFAL

See Also   LIMIAMP, LIMIBEEP, LIMIDEL, LIMIFREQ, LIMILINE, LIMIRCL, LIMIREL,
LIMISDEL, LIMISEG, LIMITEST, LIMITYPE, and any commands affecting trace
data.

LIMIHALF

Select upper or lower limit line for modification or creation.
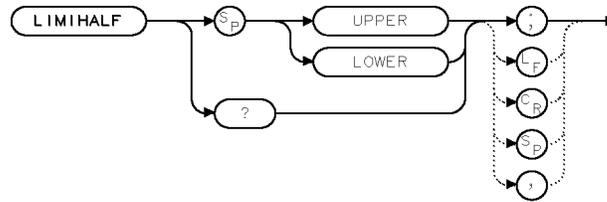
Item	Description
Preset State	LIMIHALF UPPER

Example

```
10  OUTPUT 723;"LIMIHALF ;"
```

Description The LIMIHALF command selects for modification either the upper or lower limit line in the limit-line table. The LIMIHALF command works in conjunction with the following commands to define a limit line stored in the limit-line table:

1. LIMILINE clears the limit-line table, prepares it for data entry, and specifies the number of segments in a limit line.
2. LIMISEG defines the amplitude and wavelength values of each of the limit-line segments.
3. LIMIREL determines whether the values of the limit line are absolute or are relative to the center-wavelength and reference-level settings.

Query Response
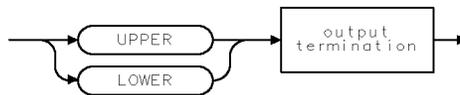
See Also LIMILINE

LIMILINE

Query the current limit-line parameters for future recall or build a new limit line.
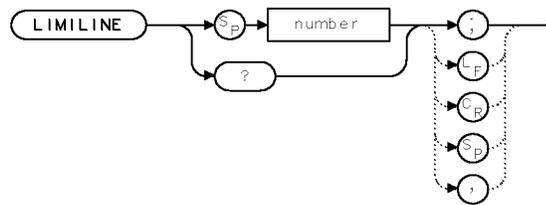
XLILIN

Item	Description
Range	1 to 400 segments, depending on available memory.
	If only 1 segment is entered, the segment type "POINT" is used.

Example

```
10 OUTPUT 723;"LIMILINE 20;"
```

Description The LIMILINE command initiates the process that loads new **limit lines** into the **limit-line table**. When creating a limit line, following LIMILINE with a number clears the limit-line table, prepares it for data entry, and specifies the number of segments in the limit line. Thus, the expression LIMILINE 20 clears the limit-line table and allocates memory space for a 20-segment limit line.

A limit line is a test limit consisting of a series of line segments positioned according to wavelength and amplitude in the spectrum-analyzer measurement range. Two limit lines can be defined, one for the upper limit, another for the lower limit. Once defined, trace data can be compared to the limit lines as the spectrum analyzer sweeps. When the trace data exceeds the upper or lower limits, the spectrum analyzer beeps or sends test results to the computer. Limit lines can be stored in a table or in a trace. Use these commands in the order given to build a limit-line for storing in the limit-line table.

1. LIMILINE clears the limit-line table, prepares it for data entry, and specifies the number of segments in a limit line.

2. LIMIHALF specifies whether the limit line is an upper or lower limit-line.

3. LIMISEG defines the amplitude and wavelength characteristics of each of the limit-line segments.

4. LIMIREL determines whether the values of the limit line are absolute values or positioned relative to the reference-level and center-wavelength settings.

The contents of the limit-line table can be stored in a file with the LIMISAV or STOR command.

When the LIMILINE command is executed, the user-defined trace arrays, called LIMIT_LO and LIMIT_HI, are stored in internal memory. These arrays each contain a trace that reflects the contents of the limit-line table.

Execute `LIMILINE?` to send the contents of the limit-line table to a computer.

See Also LIMIBEEP, LIMIFAIL, LIMITEST

Query Response
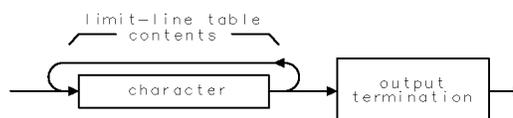
The query response is a character string consisting of LIMILINE, LIMIREL, LIMIHALF, and LIMISEG commands, which restore the limit-line table.

LIMINEXT

Activate the next limit-line segment.
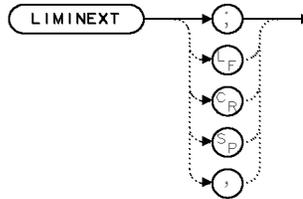
XLINXT

Example

```
10   OUTPUT 723;"LIMINEXT;"
```

Description The LIMINEXT command moves the active position from the current limit line segment to the next limit line segment. If the segment prior to executing this command was the last segment, the first segment will become active.

See Also LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

LIMIRCL

Recall stored limit line into limit-line table.
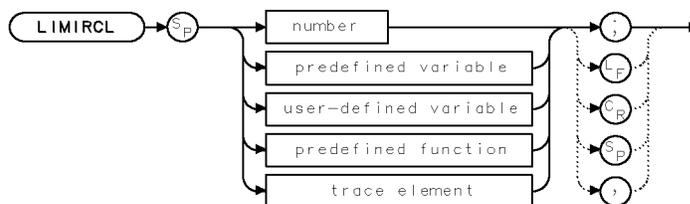
XLIRCL

Item	Description
Range	0 – 9999
Requirements	To select storage location |HP-IB device, HP-MSIB device, or internal memory|, execute MSI before executing LIMIRCL. Internal memory is the storage device when power is applied to the spectrum analyzer.|

Example

```
10  OUTPUT 723;"LIMIRCL 2;"
```

Description The LIMIRCL command recalls limit lines that have been stored previously by the LIMISAV command. LIMISAV specifies a limit-line file number. For example, **LIMISAV 2** stores a limit line in file 2. Execute **LIMIRCL 2** to recall the limit line.

When files are stored in user memory, limit-line entries appear in the user memory catalog as "l_N" where N is the number of the register and "l" indicates the file contains limit lines. For example, the entry for **LIMISAV 2** is listed in the catalog as "l_2" unless the PREFX command has been used to alter the "l" prefix. Press State , recall , catalog to view the catalog of user memory.

See Also LIMILINE, LIMITEST

LIMIREL

Determine whether limit line values are absolute or relative.
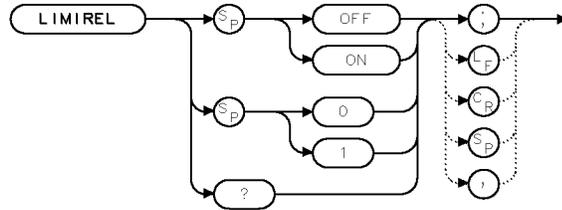
XLIREL

Item	Description
Preset State	LIMIREL OFF

Example

```
10  OUTPUT 723;"LIMIREL;"
```

Description The LIMIREL command determines whether the wavelength and amplitude parameters in the limit-line table represent absolute values or relative values referenced to the reference-level and center wavelength settings. Executing **LIMIREL OFF** specifies limit-line table parameters as absolute values, and positions the limit line accordingly. Executing **LIMIREL ON** specifies limit-line table parameters as relative values, positioning the limit line relative to the reference-level and center-wavelength settings. The LIMIREL command does not affect limit-line testing with limit lines stored in trace arrays.

Query Response
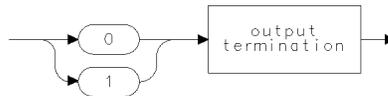
qgen0

See Also LIMILINE, LIMITEST

LIMISAV

Save contents of limit-line table in limit-line file for future recall.
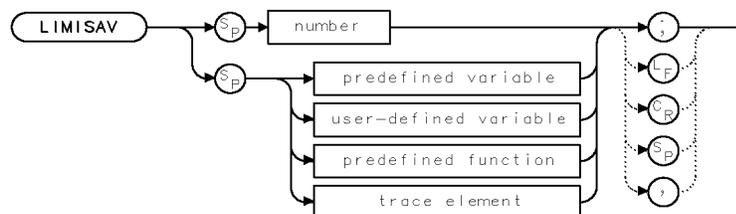
Item	Description
Range	0 – 9999

Example   `10  OUTPUT 723;"LIMISAV 2;"`

Description The LIMISAV command saves in user memory the contents of the limit-line table only. LIMISAV does not save limit lines that are present in the LIMIT_HI or LIMIT_LO trace array. (LIMIT_HI and LIMIT_LO are described under the LIMITEST command.)

LIMISAV specifies a register number. For example, `LIMISAV 2` stores a limit line in register 2. Execute `LIMIRCL 2` to recall the limit line. Files are saved in user memory unless specified otherwise by the MSI command. When stored in user memory, limit-lines entries appear in the user-memory catalog as "l_N", where N is the number of the register and "l" indicates the file contains limit lines. For example, the entry for `LIMISAV 2` is listed in the catalog as "l_2". Press (State), `recall`, `catalog & MSI` to view the catalog of user memory. Use the PREFX command to change the letters of the "l" prefix.

LIMISCRL

Scroll the active limit-line segment from the current active segment.

xlisrl
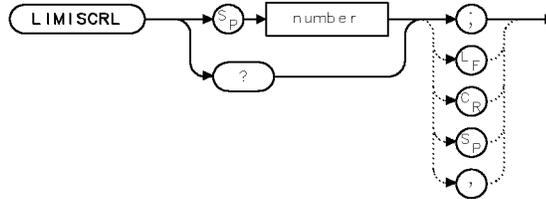
Example   `10   OUTPUT 723;"LIMISCRL 19;"`

Description The LIMISCRL command scrolls the active limit-line segment from the current active segment. If there are a total of 15 segments and you enter the number 9 for segments to scroll, only 9 are scrolled.

If the number entered is larger than the current number of segments, the count will wrap around to the first segment and continue counting from there. If the number is larger than the maximum number of segments, an error will be generated. For example, if there are 20 segments in your limit-line table, only the first 10 segments can be displayed on-screen. However, using the LIMISCRL command and entering in number 19 will allow you to view segments 10 through 20. By using LIMISCRL in this manner, it tends to have a "scrolling" affect on the table in which different portions of the limit-line table can be viewed.

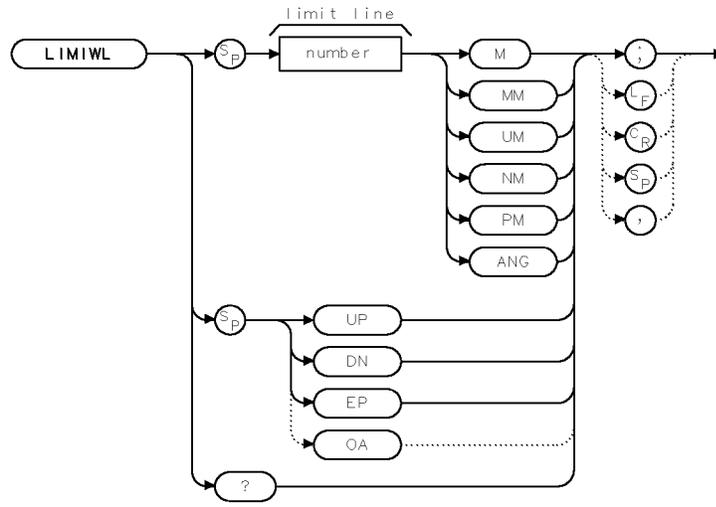**Query Response** The query response returns the number of the segment that is currently active.
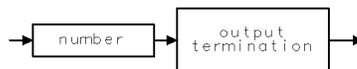
qgennum

See Also LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

LIMISDEL   Delete the currently active limit-line segment.
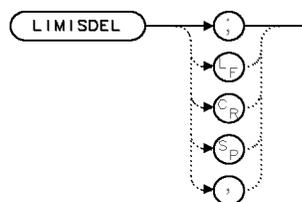
XLISDL

Example   `10  OUTPUT 723;"LIMISDEL;"`

Description The LIMISDEL command deletes the currently active limit-line segment.

See Also   LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

LIMISEG

Define slope and offset of limit-line segments.
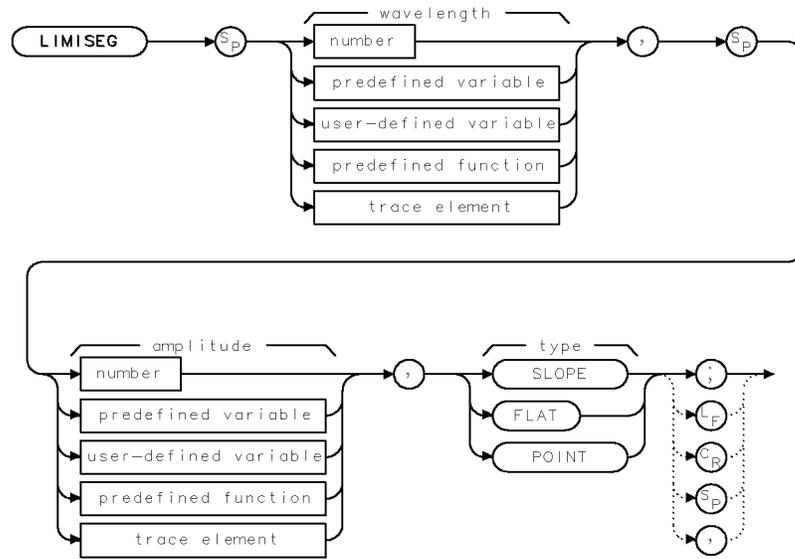
Item	Description
Range	1 to 400 segments, depending on available memory.
	If only 1 segment is entered, the segment type "POINT" is used.

Example   10 OUTPUT 723;"LIMISEG;"

Description The LIMISEG command specifies the characteristics of a limit line that is stored in the limit-line table. When creating new limit lines, execute LIMILINE, and LIMIHALF if necessary, before executing LIMISEG.

A limit line is divided into segments. LIMISEG defines the characteristics of each segment in terms of wavelength, amplitude, and, in most cases, slope. The amplitude and wavelength terms define the beginning point of the current segment; the type parameter defines the segment characteristics to the next point.

The segment type of the last segment of a limit line is not returned since the type parameter defines how the segments are drawn (connected) to the next segment. The last segment does not have a "next" segment. If there are 10 segments, the segment type for the first 9 is returned.

The three segment types are:

- FLAT draws a zero-slope line between the beginning points of two segments, producing limit-line values equal in amplitude for all wavelengths between the two points.

- SLOPE produces limit-line values of potentially varying amplitude for all wavelengths between the two points using linear interpolation.

- POINT specifies a limit value for the beginning point only.

If the segment type is omitted, the type of the previous segment is used. Limit-line segments can be added during limit-line testing. The LIMISEG command works in conjunction with the following commands to define a limit line in the limit-line table:

1. LIMILINE clears the limit-line table, prepares it for data entry, and specifies the number of segments in the limit line.

2. LIMIHALF specifies whether the limit-line is an upper or lower limit-line.

3. LIMIREL determines whether the values of the limit line are absolute or are relative to the center-wavelength and reference level settings.

See Also LIMIDEL, LIMIHALF, LIMILINE, LIMIRCL, LIMISDEL

LIMITEST

Compare active trace data to limit-line parameters.
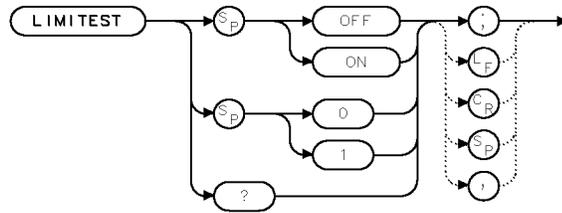
XLITST

Item	Description
Preset State	LIMITEST OFF

Example 10 OUTPUT 723;"LIMITEST ON;"

Description The LIMITEST command compares data in the currently sweeping trace to a limit line. A limit line must be stored in a table or trace before testing begins. (For a description of limit lines, refer to the LIMILINE command.) When in table form, the wavelength-amplitude characteristics reside in the **limit-line table**. To view the limit-line table, press `Misc`, `MORE 1 of 3`, `limit lines`. During measurements, the limit line characteristics are displayed graphically as a **limit line** when active-trace data is displayed. When in trace form, the wavelength-amplitude characteristics reside in either of two trace arrays, called LIMIT_HI and LIMIT_LO, which are stored in user memory.

Testing with Limit Line Limit lines stored in traces can be used for limit line testing.

First, load a limit line into the limit-line table. Use LIMIRCL to load a previously stored limit line or execute the following commands, in the order given, to create a limit line:

1. LIMILINE clears the limit-line table, prepares it for data entry, and specifies the number of segments in a limit line.

2. LIMIHALF specifies whether the limit line is an upper or lower limit-line.

3. LIMISEG defines the amplitude and wavelength characteristics of each of the limit-line segments.

4. LIMIREL determines whether the values of the limit line are absolute or are positioned relative to the reference-level and center wavelength settings.

Next, execute **LIMITEST ON** and sweep the measurement range. Executing **LIMITEST ON** automatically copies the contents of the limit-line table into the LIMIT_HI and LIMIT_LO trace arrays and compares active trace data to the line parameters.

When LIMIREL has been executed to make limit-line values relative, remember that changing the amplitude or wavelength range during limit-line testing will change the limit-line parameters. During single-sweep mode, **LIMITEST ON** compares the limit line to trace A data only. Executing **LIMITEST OFF** erases the LIMIT_HI and LIMIT_LO trace arrays.

Testing with Limit Lines Stored in Traces

First, use the LIMISAV command to store the contents of the limit-line table if needed for future use, then erase the contents of the limit-line table:

```
40   OUTPUT 723;"LIMIDEL;"
```

Next, use the TRDEF command to define a trace array called LIMIT_HI or LIMIT_LO. Store upper limits in LIMIT_HI and lower limits in LIMIT_LO. Below, LIMIT_HI is defined to contain 800 wavelength amplitude pairs:

```
90   OUTPUT 723;"TRDEF LIMIT_HI,800;"
```

Next, store limit-line parameters in the trace array. Use the TRA, TRB, or TRC command to enter limit-line data point-by-point. If preferred, use swept trace data as a limit line.

In the example below, limit-line parameters for trace B are stored in LIMIT_HI. Note that the MOV command performs the transfer.

```
200   OUTPUT 723;"SNGLS;CLRW TRB;TS;MOV LI MIT_HI,TRB;"
```
Store trace B data as a limit line called LIMIT_HI.
```
210   OUTPUT 723;"BLANK TRB;"
```
Turn trace B off.

Next, execute **LIMITEST ON** and sweep the measurement range.

```
220   OUTPUT 723;"LIMITEST ON;"
```
Execute the limit test on trace A.

Executing **LIMITEST ON** compares trace data to the line parameters stored
in the LIMIT_HI and LIMIT_LO arrays. As before, during single-sweep
mode, **LIMITEST ON** compares the limit line to trace A data only. Executing
LIMITEST OFF erases the LIMIT_HI and LIMIT_LO trace arrays. Use the STOR
command to save trace array contents in a file. Below, the LIMIT_HI trace
array is stored as a trace file called "LIMIT."

```
500  OUTPUT 723;"STOR T,$LIMIT$,LIMIT_HI;"
```

Analyzer Test
Results
The limit-line test fails when trace-data amplitude is greater than the upper
test limit or less than the lower test limit. Use LIMIFAIL or LIMIBEEP to
signal test-limit failures. Use the marker and math commands to analyze trace
data. For example, executing **SUB TRB,TRB,LIMIT_HI** stores the difference
between the upper test limit and the trace B data. Execute **MKPK HI;MKWL?**
of the signal that failed the test limit by the largest amount.

Query Response
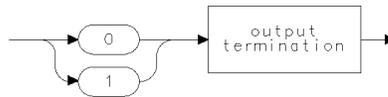
qgen0

LIMITYPE

Specify the type of limit-line segment.
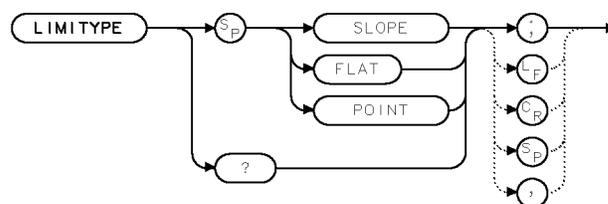
xlityp

Item	Description
Preset State	SLOPE

Example

```
10  OUTPUT 723;"LIMITYPE SLOPE;"
```

Description The LIMITYPE command specifies the type of the current limit-line segment to be either POINT, SLOPE, or FLAT. The query returns the current type or "NA" if not available.

- FLAT draws a zero-slope line between the beginning points of two segments, producing limit-line values equal in amplitude for all wavelengths between the two points.

- SLOPE produces limit-line values of potentially varying amplitude for all wavelengths between the two points using linear interpolation.

- POINT specifies a limit value for the beginning point only.

Query Response
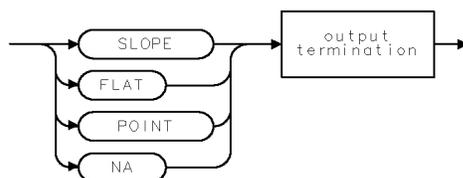
QLITYP

See Also LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

LIMIWL

Set the wavelength of the currently active limit-line segment.

Item	Description
Range	dependent on hardware

Example   10 OUTPUT 723;"LIMIWL 600NM;"

Description The LIMIWL command sets the wavelength of the currently active limit-line segment.

The query sends the wavelength of the current segment.
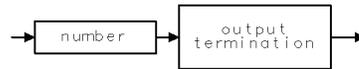
Query Response

qgennum .

See Also   LIMIBOT, LIMIDEL, LIMIDONE, LIMIEDIT, LIMIHALF, LIMILINE, LIMINEXT, LIMIRCL, LIMISCRL, LIMISDEL, LIMISEG, LIMISEL

LIMTOCOR

Transfer limit-line data to amplitude-correction data.
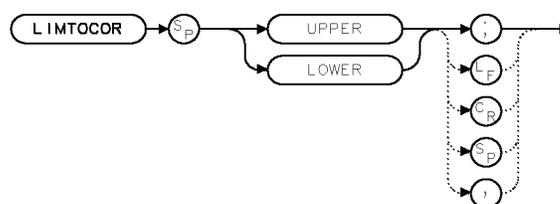
XLITOC

Item	Description
Preset State	UPPER

Example

```
10  OUTPUT 723;"LIMTOCOR UPPER;"
```

Description The LIMTOCOR command transfers limit-line data to amplitude-correction data. Points stored as limit-line segments, which can also be edited via the front panel, are converted to amplitude-correction points and used as if entered with the AMPCOR command. When used with CORTOLIM, the amplitude-correction points can be edited.

See Also AMPCOR, CORTOLIM

LINES

Connect lines between displayed trace points.
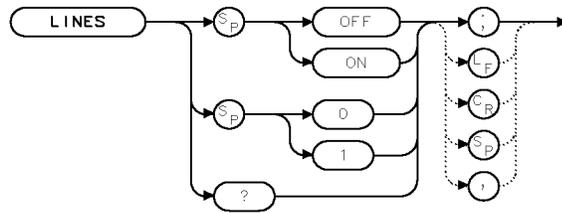
XLINES

Item	Description
Preset State	ON

Example 10 OUTPUT 723;"LINES OFF;"

Description The LINES command connects or disconnects lines between the displayed trace points. When OFF is specified, displayed traces are shown as unconnected dots. The query sends the current state.

Query Response
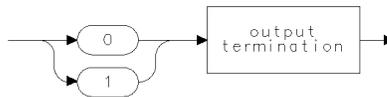
qgen0

LINET

Draw dots or dashed lines.
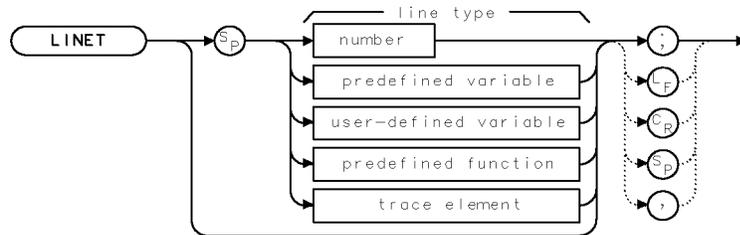
Item	Description
Line-Type Range	null, 0 – 8

Example

```
10   OUTPUT 723;"CLRDSP;BLANK TRA;"      Clear graphics from memory.
20   OUTPUT 723;"PA 0,600;"             Place graphics pen at (0,600).
30   OUTPUT 723;"LINET 6;"              Activate dotted-line drawing.
40   OUTPUT 723;"PD;PA 500,600;"        Place pen down and draw line to (500,600).
```

Description The LINET command selects the kind of line "drawn" on some displays (for example, HP 70205 or HP 70206) with the plot commands, such as PA and PR. LINET is not applicable to HP 70004 Displays. The line types are not used in plotter or printer outputs. Ten types of lines are available:

LINET plots a continuous line.
LINET 0 plots the end points only.
LINET 1 plots . . . . .
LINET 2 plots _____ _____ _____
LINET 3 plots _____ _____ _____
LINET 4 plots _____ . _____ . _____
LINET 5 plots _____ _ _____ _ _____ _
LINET 6 plots ___ ___ ___ ___ ___
LINET 7 plots __ __ __ __ __ __ __
LINET 8 plots _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

LN

Set vertical linear amplitude scale.
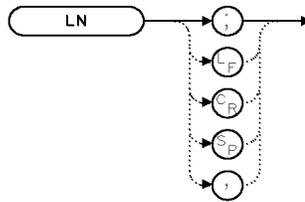
Example   `10 OUTPUT 723;"LN;"`

Description The LN command activates the linear display mode. It scales the vertical divisions of the graticule proportional to the input power. The bottom graticule line represents a signal level of 0 watts and the top is determined by the reference level. **MEASURE SR** makes LN inactive.

See Also MEASURE, LG

LOAD

Load article or file into internal memory.
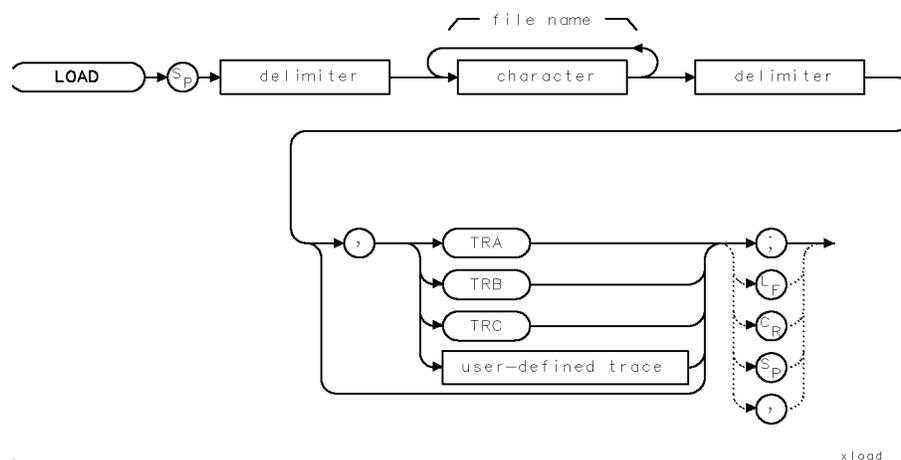
Item	Description
Range	1 – 10 characters
Delimiters	Required before and after *file name* when it contains a number or lower-case letters. When *file name* is upper-case alpha characters only, delimiters are optional.

Example

```
110   OUTPUT 723;"CLRW TRB;TS;          Sweep trace B.
120   OUTPUT 723;"STOR T,$ALPHA$,TRB;"  Store trace B in trace file called ALPHA.
130   OUTPUT 723;"LOAD $ALPHA$,TRC;"    Load trace into trace C.
```

Description The LOAD command recalls to internal memory any ASCII file that consists of spectrum-analyzer commands. These include files stored by the following commands: SAVES (state register files), SAVET (trace files), LIMISAV (limit-line files), SAVEU (user-key files), SAVED (files containing traces, user-defined functions or variables, or on-end-of-sweep algorithms), STOR (any file).

The operation of the LOAD command varies with the contents of the file recalled.

- Files containing trace data are loaded into the specified trace, or trace A if the destination is not specified. All instrument settings are changed to reproduce the state the spectrum analyzer was in when the trace was stored. In addition, the trace is placed in view mode.

- Files containing user-defined variables, functions, traces, or keys are loaded into user memory. Any duplicate user-defined articles over-write the existing versions.

- Files containing an on-end-of-sweep algorithm are loaded into user memory. The existing on-end-of-sweep algorithm, if present, is erased and replaced with the loaded algorithm.

- Files containing a limit line are loaded into the limit-line table.

- Loading state-register files reactivates the stored instrument state.

- Loading user-key files reactivates the stored (USER)-key menu.

Always follow LOAD with the file name. The file name varies depending on the storing command used.

- When loading files stored by the STOR command, the file name is the same as that specified by STOR.

- When loading files stored by the SAVED, SAVES, SAVET, LIMISAV, or SAVEU command, the file name is a combination of the file prefix plus the file number.

  For example, executing **SAVEU 10** stores a file in user memory that is listed in the catalog as "u_10". The "u_" is the prefix. The "10" is the file number. Together they comprise the file name. The following table shows the default file prefixes.

Standard File Prefixes for Files Names

Article Stored	Storing Command	File Prefix
file containing user-key menu	SAVEU	u_
file containing trace	SAVET	t_
file containing user-defined variable, function, trace, or key	SAVED	d_
file containing on-end-of-sweep algorithm	SAVED	d_
file containing limit line	LIMISAV	l_
file containing state register (instrument settings and title-line)	SAVES	s_

The PREFX command changes the file prefix for all files stored with SAVED, SAVES, SAVET, LIMISAV, and SAVEU. Thus, executing `PREFX $MARCH$;SAVEU 10` stores a file called "uMARCH_10". Notice that the PREFX command inserts letters between the "u" and "_".

To verify file names, press `State`, `recall`, `catalog & MSI` to view the catalog of user memory, which is the currently selected storage device.

See Also   MSI

LOG

Take the logarithm, then multiply it by N.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*. $\pm$ 1.8 $\times$ 10^{308} when using *number*, *predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA*, *TRB*, *TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Calculate the decibel value of a power ratio using the LOG command.

20	CLEAR 723	*Initialize analyzer.*
30	OUTPUT 723;"IP;"	
40	OUTPUT 723;"VARDEF POWER,0;"	*Define user-defined variable, called POWER, equal to 0*
50	OUTPUT 723;"LOG POWER,5,10;"	*POWER = 10 × LOG 5*
60	OUTPUT 723;"POWER?;"	*Query value of power to computer.*
70	ENTER 723;N	*Assign value to computer variable.*
80	DISP USING "D.DD,K";N;"dB"	*Display value on computer.*

Description The LOG command computes the logarithm$_{(base\ 10)}$ of the source, multiplies it by the scaling factor, then stores the result in the destination:

$$destination = (scaling\ factor)(LOG\ source)$$

When a trace is used, be sure it contains a complete sweep of measurement information.

MDS

Specify binary data as one byte (B) or two bytes (W). Also specify number of significant figures.
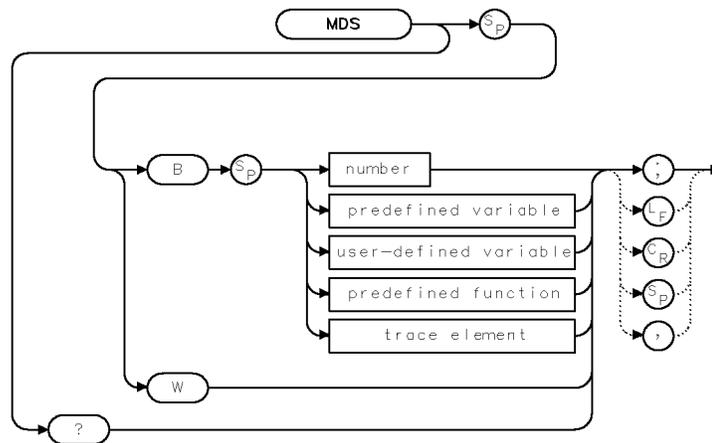
XMDS

Item	Description
Scaling-Factor Range	0 – 16
Preset State	MDS W

Example Send trace data to computer as binary words (two 8-bit bytes).

5	CLEAR 723	*Initialize analyzer.*
7	OUTPUT 723;"IP;"	
10	INTEGER Z(1:800)	*Declare integer array.*
20	ASSIGN @Sa TO 723	*Assign fast I/O path to 723.*
30	ASSIGN @Sa_fast TO 723;FORMAT OFF	
40	OUTPUT @Sa;"MDS W;"	*Define word as data size.*
50	OUTPUT @Sa;"TDF B;"	*Activate binary data format.*
60	OUTPUT @Sa;"TRA?;"	*Query amplitude of all trace elements in measurement units.*
70	ENTER @Sa_fast USING "#,W";Z(*)	*Terminate queried data. "W" specifies that items stored in Z(*) are words (two bytes).*
80	PRINT Z(*)	

Description   The MDS command determines if binary data is transmitted as one byte
or one word (two 8-bit bytes). In addition, MDS specifies the number of
significant digits contained in the binary data. Use MDS to improve speed and
economize on storage space in memory. Better throughput (speed) is achieved
with MDS B, at the sacrifice of resolution and thus accuracy. During normal
operation, trace data is provided in decimal form. Use MDS when the TDF
command specifies trace data as binary data. MDS also determines the format
of data the spectrum analyzer receives from the computer. MDS B multiplies
each byte of data *received* by 2 raised to the power of the scaling factor.
MDS B divides each byte of data *sent* by 2 raised to the power of the scaling
factor. In the example below, the number 16 is processed with a scaling
factor of 3. (Number 16 represents the amplitude of a trace data point.)

Input Data   Output Data
$0001\ 0000 \times 2^3 = 1000\ 0000$ $0001\ 0000 \div 2^3 = 0000\ 0010$

A scaling factor of 8 provides adequate resolution for most measurement
applications. When a scaling factor of 8 is used, trace data from 0 to 255 is
sent as 0. Trace data from 256 to 511 is sent as 1 (0000 0001). Trace data
from 512 to 767 is sent as 2 (0000 0010), and so on.

If the trace is in linear units, the range is 0 to 10,000. For linear mode, a
scaling factor of 6 is more appropriate (returning values from 0 to 156 for
trace data).

In standard 10 dB log mode, values range from −32,768 to +32,767. A
scaling factor of 8 offers coverage of the full range (measurement units are
hundredths of a dB). However, if it is known that the signal will not exceed
±156 dB, a scaling factor of 7 is more appropriate. The scaling factor of 7
preserves more resolution and accuracy. For a scaling factor of 7 in log mode,
the minimum resolution is 1.28 dB. A scaling factor of 8 provides 2.56 dB
resolution.

Query Response

```
number → output
         termination →
```
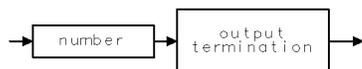
qgennum

See Also TDF, FETCH

MEAN

Calculate the mean value of the amplitudes in the source.

xmean

Item	Description
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC.*
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC,* or *user-defined trace.*
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC,* or *user-defined trace.*

Example

```
20  CLEAR 723                          Initialize analyzer.
30  OUTPUT 723;"IP;"
40  OUTPUT 723;"SNGLS;"                Activate single-sweep mode.
50  OUTPUT 723;"CENTERWL 1300NM;SP 10NM;"   Set measurement range.
60  OUTPUT 723;"TS;"                   Sweep trace A.
70  OUTPUT 723;"MEAN TRA?;"            Query mean value of trace A.
80  ENTER 723;N                        Assign value to computer variable.
90  DISP "MEAN OF TRACE IS ";N         Display value on computer display.
```

Description   The MEAN command calculates the mean value of the source. If a trace is used as the source, be sure it contains a complete sweep of measurement information. MEAN must be used as either a query or as a source in another analyzer-command function. Form a query by ending the MEAN statement with a question mark (?). When used as a query, MEAN sends the mean value of the source. Use MEAN as a source by incorporating the MEAN statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When MEAN is used as a source, the mean value of the source is used for operation by the command that contains MEAN.

Predefined-
Function
Syntax
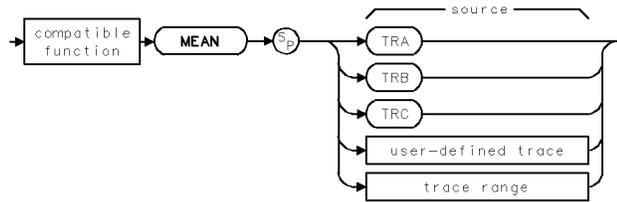
Query Syntax
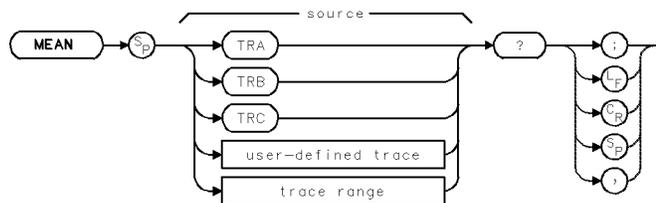
Query Response
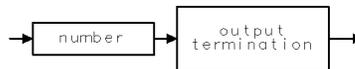
qgennum

MEASU

Convert parameter units of amplitude to measurement units.

ckb5

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*.
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*.
Reference-Trace Length Range	3 – 2048 position units. Default is 800 position units when using *TRA*, *TRB*, *TRC*.
Reference-Trace Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*.
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA*, *TRB*, *TRC*, or *user-defined trace*.

Example | Use the MEASU and POSU commands to position graphics on the analyzer display in measurement units and in position units.

```
10   CLEAR 723
20   OUTPUT 723;"IP;"
30   OUTPUT 723;"CLRDSP;"
40   OUTPUT 723;"SP 10NM;CENTERWL 1300NM;TS;"
50   OUTPUT 723;"SCALE 0,800,-11003,502;"
60   OUTPUT 723;"PU;PA 0,MEASU -37DBM,TRA;"
70   OUTPUT 723;"PD;PA POSU 1300NM,TRA,MEASU -22DBM,TRA;"
80   OUTPUT 723;"PU;PA 0,MEASU -25DBM,TRA;"
90   OUTPUT 723;"PD;PA 800,MEASU -30DBM,TRA;"
```

Initialize analyzer.

Clear graphics from memory.
Set measurement range.
Set graphics scale for the display of traces.
Place the pen on the left edge of the graticule at −37 dBm.
Draw line to the point at 1300 nm and −22 dBm.
Move pen to the left edge of the graticule, at −25 dBm.
Draw a line to the right edge of the graticule, ending at −30 dBm.

Description | The MEASU command converts parameter-unit values to measurement-unit values. Measurement units describe trace amplitude information.

- For linear trace information, measurement units are defined as 0 (zero) for the bottom of the display, and 10,000 for the top of the display. A measurement unit equals the reference level/10,000:

$$measurement \ units = \frac{(value \ in \ reference \ level \ values)}{((reference \ level)(10,000))}$$

- For logarithmic trace information, the range of trace data is −32,768 measurement units to +32,767 measurement units. A measurement unit is one hundredth of a dBm:

$$measurement \ units = (value \ in \ dBm)(100)$$

Parameter units are standard scientific units:

Parameter Units for Amplitude

Decibels	Watts
dBm	W
	mW
	nW
	pW

Use the MEASU command to preserve decimal values. All trace amplitude information is stored internally and manipulated in measurement units. This means a −10 dBm reference-level value is converted to −1000 measurement units before it is transferred to a trace element. (Recall that a measurement unit is one hundredth of a decibel.) This process is invisible to the user except in math and move functions when the source consists of a variable or real number, and the destination is a trace array or element. In this case, the source and destination express numbers in different units, making a conversion necessary. If the conversion is omitted, resulting data will be inaccurate by a factor of 100 and significant digits can be lost. If the reference trace is not specified, the last active trace is the reference trace. If the units are not specified, the AUNITS command determines the units.

MEASU must be used as either a query or as a source in another analyzer-command function. Form a query by ending the MEASU statement with a question mark (?). When used as a query, MEASU sends the measurement-unit value of the source to the computer.

Use MEASU as a source by incorporating the MEASU statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When MEASU is used as a source, the measurement-unit value of the source is used for operation by the command that contains MEASU.

The following table lists commands that require the use of the MEASU command.

Commands that May Require Conversion using MEASU

Command	Definition
ABS	Compute absolute value.
ADD	Add sources, point by point.
AVG	Average source with destination, point by point.
BIT	Store value of bit.
CONCAT	Concatenate trace.
DIV	Divide sources, point by point, discard remainder.
EXP	Divide source by scaling factor and raise to power of 10.
INT	Compute integer value.
LOG	Compute logarithm.
MEAN	Compute mean value of data points.
MIN	Retain minimum value.
MOD	Divide sources, point by point, retain remainder.
MOV	Duplicate value|s| contained in source and place in destination.
MPY	Multiply sources, point by point.
MXM	Retain maximum value.
RMS	Compute root mean square of source, point by point.
SQR	Compute square root of source, point by point.
STDEV	Compute standard deviation.
SUB	Subtract sources, point by point.
SUM	Add sources, point by point.
SUMSQR	Add square of sources, point by point.
VARIANCE	Compute variance.
XCH	Exchange sources.

**Predefined-
Function
Syntax**
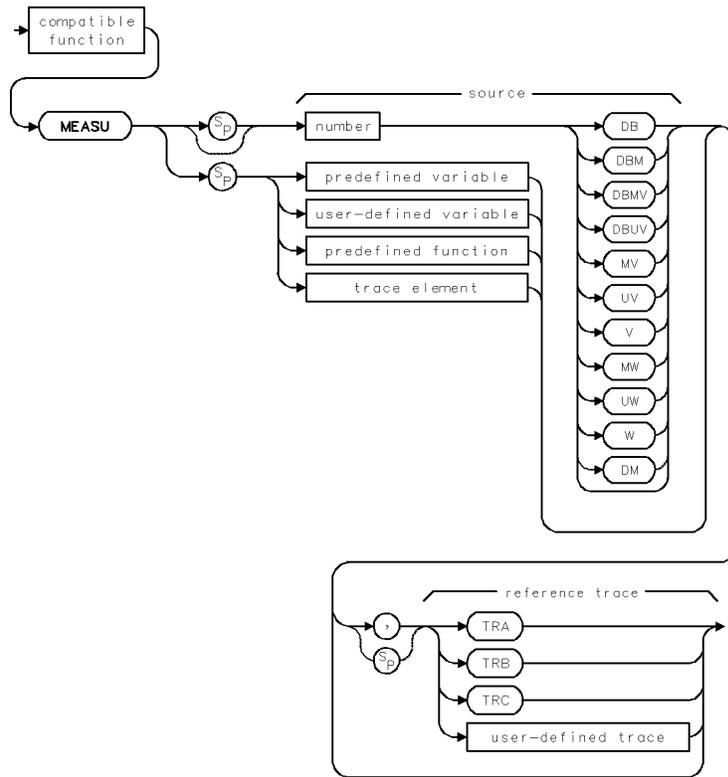
Query Syntax
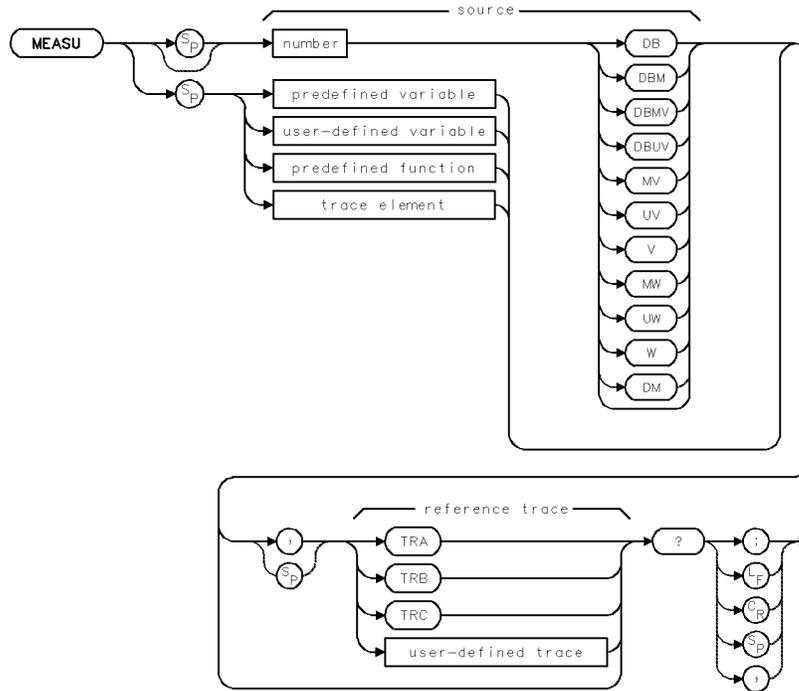
Query Response
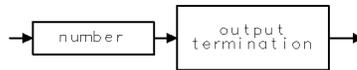
qgennum .

See Also ROFFSET, MEASURE, INZ, LN, LG, RL, AUNITS

MEASURE

Select spectrum-analysis or stimulus-response mode.
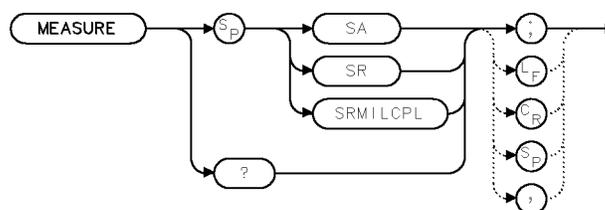
xmeasr

Item	Description
Preset State	MEASURE SA

Example

120 OUTPUT 723;"MEASURE SA;"

Description The MEASURE command determines what kind of measurements the modular measurement system makes.

Argument	Definition

SA Activates spectrum analysis, where all amplitude units are specified in absolute values determined by:

- Amplitude units (AUNITS)
- Reference level (RL)
- Reference-level offset (ROFFSET)
- Logarithm scale (LG)
- Linear scale (LN)
- Amplitude correction (AMPCOR)

SR Activates stimulus-response measurements where amplitude units are specified relative to a reference level of 0 dB when the A-minus-B or A-minus-C commands, AMB or AMC, are executed. During relative-amplitude mode, the following parameters are in dB:

- Trace data (TRA, TRB, TRC)
- Trace data conversions (AMPU and MEASU)

- Display line (DL)
- Threshold (TH)
- Marker amplitude (MKA)
- Reference level

SRMILCPL   This argument is identical to the SR argument except that system gains are allowed to change. For example, gain can change due to changes in the reference level.

Query Response
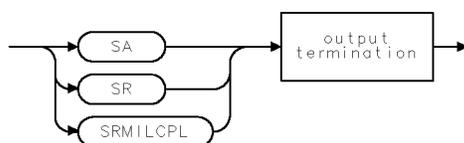
qmeasr

When the INSTMODE command is set to PD or SR, MEASURE SR is automatically activated.

See Also LN, AUNITS, RL, ROFFSET, AMPU, MEASU, DL, TH, MKA, MKREAD, MKRL

MEM

Return amount of available **internal memory.**
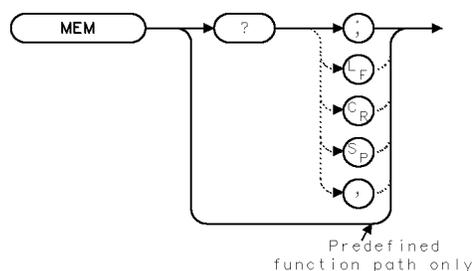
XMEM

Example Determine how many trace arrays can fit into available internal memory.

20	`CLEAR 723`	*Initialize analyzer.*
30	`OUTPUT 723;"IP;"`	*Define user-defined variable equal to 0.*
40	`OUTPUT 723;"VARDEF TEMP,0;"`	
50	`OUTPUT 723;"DIV TEMP,MEM,846;"`	*Divide the number of available user-memory bytes by 846 and store the results in TEMP. (A 400-point trace occupies 846 bytes.)*
60	`OUTPUT 723;"INT TEMP,TEMP;"`	*Store the integer value of TEMP in TEMP.*
70	`OUTPUT 723;"MEM?;"`	*Query the amount of available internal memory to the computer.*
80	`ENTER 723;N`	*Assign value to N.*
90	`OUTPUT 723;"TEMP?;"`	*Send the value of TEMP to computer.*
100	`ENTER 723;M`	*Assign value to M.*

Description The MEM command computes the amount of unused internal memory. The amount of total internal memory available is dependent on the analyzer configuration and the local-oscillator firmware (ROM) version. MEM must be used as either a query or as a source in another analyzer command. Form a query by ending the MEM statement with a question mark (?). When used as a query, MEM sends the amount of unused memory to the computer. Use MEM as a source by incorporating the MEM statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When MEM is used as a source, the value of unused memory is used for operation by the command that contains MEM.

**Predefined-
Function
Syntax**
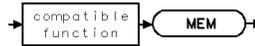
Query Syntax
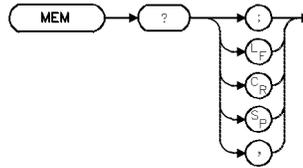
Query Response
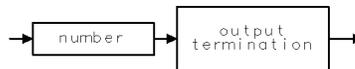
qgennum

See Also ACTDEF, AMPCOR, FUNCDEF, LIMILINE, LIMISAV, LOAD, NSTATE, ONEOS, PDA, PDF, RCLD, READMENU, REPEAT/UNTIL, SAVES, SAVET, SAVEU, STOR, TRDEF, USERERR, USERKEY, USERMSG, USTATE, VARDEF

MIN

Compare two values or traces, then store lesser value or trace.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm$ 1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Compare the marker amplitude to −20 dBm, then display the lower value on the analyzer display.

```
10   CLEAR 723                              Initialize analyzer.
20   OUTPUT 723;"IP;"
30   OUTPUT 723;"CLRDSP;"                   Clear graphics from memory.
40   OUTPUT 723;"SNGLS;"                     Activate single-sweep mode.
50   OUTPUT 723;"VARDEF MINIMUM,0;"          Define user-defined variable equal
                                             to 0.
60   OUTPUT 723;"TS;MKPK HI;"                Sweep trace A. Place active marker
                                             at the highest signal level.
70   OUTPUT 723;"MIN MINIMUM,MKA,-20DBM;"    Compare the marker amplitude to
                                             −20 dBm. Store the smaller value
                                             in MINIMUM.
80   OUTPUT 723;"PU;PA 300,700;"             Move graphics pen to (300,700).
90   OUTPUT 723;"DSPLY MINIMUM,6,2;"         Display value of minimum.
100 OUTPUT 723;"TEXT $dBm$;"                 Write "dBm" next to the variable.
```

Description The MIN command compares sources, point by point, sending the lesser values of each comparison to the destination. If one of the sources is a single value, it acts as a threshold, and all values equal to or less than the threshold pass to the destination. If traces are used as sources, be sure each contains a complete sweep of measurement information.

MINH

Compare amplitude of current trace data to previous sweep, then retain lesser values.
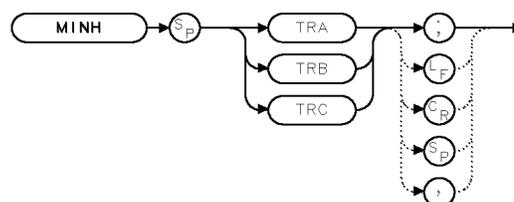
XMINH

Example   30 OUTPUT 723;"MINH TRA;"

Description The MINH command (like the minimum hold function) and compares the current amplitude value of each point on a trace as it sweeps to each point value detected during the previous sweep, then retains the minimum values. The trace is continuously swept, updated, and displayed.

The optical spectrum analyzer system controls the detection mode. When MINH is activated, the current instrument detection mode (whether sample or peak) is displayed with the results. The minimum trace value is displayed with an S when sample detection mode is active, or a P when peak detection mode is active. This line of text appears in the bottom line at the center of the display.

See Also TRPST, TM, VTL, VTH

MK

Position a marker anywhere on the display.
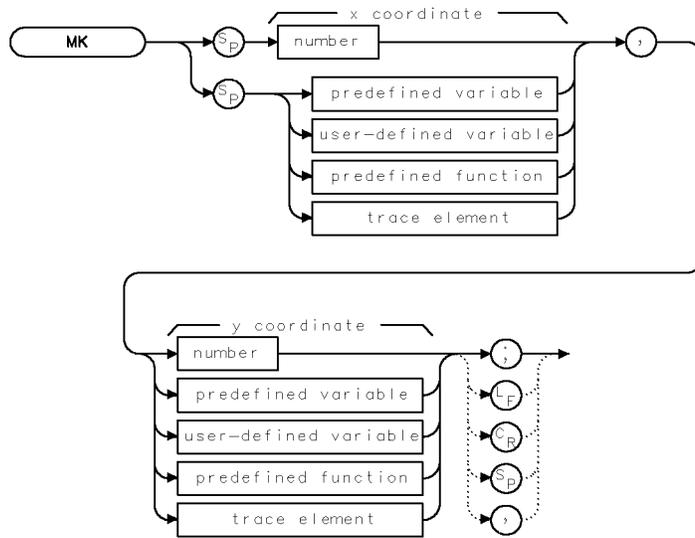
Item	Description
X-Coordinate Range	Depends on SCALE or DWINDOW command.
Y-Coordinate Range	Depends on SCALE or DWINDOW command.

Example

30	OUTPUT 723;"PD;MK 600,800;PU;"	*Place pen down. Position marker at (600,800). Lift pen.*
40	OUTPUT 723;"IT 1;MK 700,700;"	*Activate reference-graphics mode and assign MK to item 1. Position marker at (700,700).*

Description   The MK command positions a marker anywhere on the analyzer display. Use the PD command to first place the pen down (except during reference-graphics mode.) Coordinates are specified in the current scale units, which are 0 to 1000 for the X and Y coordinates unless specified differently with the SCALE command or with the DWINDOW command if a window is active. Use the MK command in conjunction with the PU and PD commands when the reference-graphics mode is off. During the reference-graphics mode, the marker is drawn at the item origin, as in the preceding example.

See Also   IT, SCALE, DWINDOW, CLRDSP

MKA

Query the amplitude of the current marker.
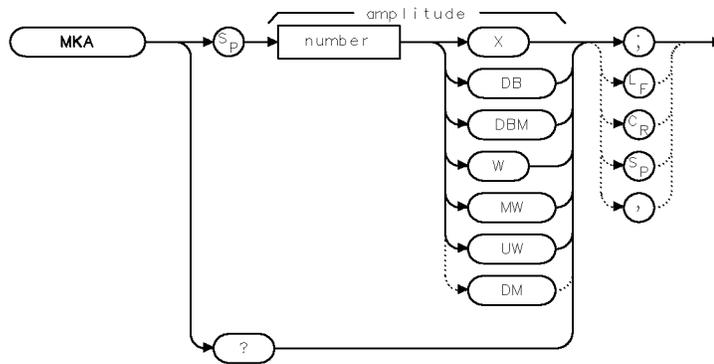
Item	Description
Amplitude Range	± 300 dB

Example   Query amplitude of signal at marker.

40	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
50	OUTPUT 723;"TS;"	*Sweep trace A.*
60	OUTPUT 723;"MKN;"	*Display a marker at the center of the trace.*
70	OUTPUT 723;"MKA?;"	*Send marker amplitude to computer.*

Description   In marker normal mode, the MKA? query places the marker at the center wavelength (CENTERWL) specified. If an active marker is not displayed when you send this command, the marker is placed at the specified amplitude. If more than one trace is displayed, MKA marks the currently active trace. If more than one trace is active, MKA marks the traces in the sequence order of A, B, then C, unless the MKTRACE command specifies otherwise.

When the vertical scale is in logarithmic mode, the amplitude units are in dBm; in linear mode, the amplitude units are in watts (W, MW, or μW).

When relative-marker mode is activated with the MKAL or MKAR command, the MKA command specifies the amplitude difference from the reference marker.

When the vertical scale is in logarithmic mode, the amplitude units are in dBm; in linear mode, the amplitude units are in ratio, as indicated with the X in the syntax diagram.

Query Response The MKA query gives the amplitude of the active marker. As a predefined variable (such as in a math operation), MKA represents the active-marker amplitude.
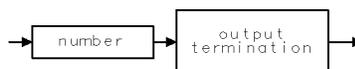
qgennum

See Also MKTRACE, MKOFF, MKREAD and commands that affect amplitude, such as AMPCOR, MEASURE, RL, ROFFSET, INZ

MKACT

Select active marker.
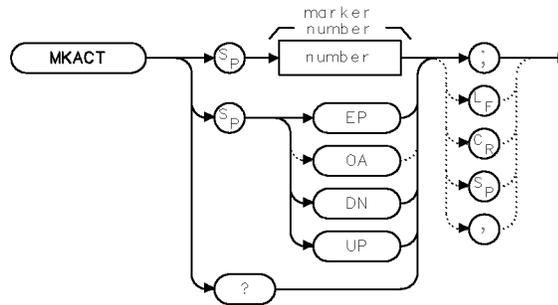
xmkact

Item	Description
Marker-Number Range	Integers, 1 – 5
Step Increment	1
Preset State	MKACT 0

Example   10 OUTPUT 723;"MKACT 5;"   *Define marker #5 as the active marker.*

Description The MKACT command selects one of five markers as the active marker. All five markers can be displayed simultaneously, but only one can be active at a time. The active marker is highlighted and its number and position are notated at the upper right corner of the graticule. If no markers are displayed, MKACT places the active marker at the center of the trace. Instrument preset (IP) turns off the markers. Marker #1 is the active marker unless specified differently by the MKACT command.

Query Response If markers are off, MKACT sends a 0 to the computer.

```
  ──▶┌──────────┐   ┌──────────────┐
     │  number  │──▶│    output    │──▶
     └──────────┘   │ termination  │
                    └──────────────┘
```
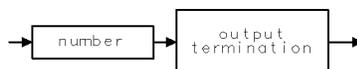
qgennum

See Also MKAL, MKN, MKAR, MKD, MKOFF

MKAL

MKAL activates marker #2 and positions it on a trace *relative* to the amplitude of marker #1.
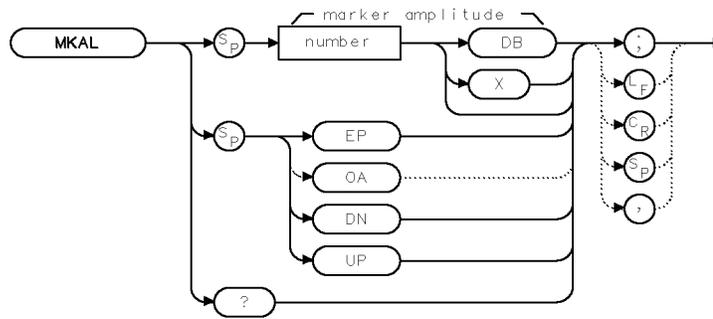
XMKAL

Item	Description
Marker-Amplitude Range	± 300 dBm
Step Increment	1 major scale division
Preset State	MKAL 0

Example

40	OUTPUT 723;"CENTERWL 1000NM;SP 500NM;TS;"	*Set measurement range. Sweep the analyzer.*
50	OUTPUT 723;"MKPK HI;"	*Set marker at highest signal peak.*
60	OUTPUT 723;"MKAL -60DB;"	*Set amplitude-left marker 60 dB below reference marker.*

Description The MKAR command makes marker #1 a reference marker and marker #3 an active amplitude-left marker. The MKAR command also activates the relative-marker mode, which displays two markers and calculates the wavelength and amplitude difference between them. When this mode is active, the MKA and MKAR commands specify the amplitude difference between the amplitude-right and reference markers. If the vertical scale of the display is logarithmic, the difference between the markers is in decibel units. If the vertical scale is linear, the difference between the markers is a power ratio. In the syntax diagrams, X represents the ratio of the marker amplitudes.

Use MKAR with the MKAL and MKBW or MKBWA commands to simplify bandwidth measurements.

If marker #1 is not displayed, MKAR positions it at the center of the trace. If more than one trace is displayed, MKAR positions marker #1 on the active trace or on trace A, B, or C, in that order; or marks the trace specified by MKTRACE. Update trace data after executing MKAR.

Query Response

```
──▶┌──────────┐──▶┌──────────────┐──▶
   │  number  │   │   output     │
   └──────────┘   │ termination  │
                  └──────────────┘
```

qgennum

See Also MKAR, MKBW, MKBWA, MKREAD, MKOFF, MKPX and commands affecting amplitude such as MEASURE, AMPCOR, ROFFSET

MKAR

MKAR activates marker #3 and positions it on a trace *relative* to the amplitude of marker #1.

XMKAR

Item	Description
Marker-Amplitude Range	± 300 dBm
Step Increment	1 major scale division
Preset State	MKAR 0

Example

40	OUTPUT 723;"CENTERWL 1000NM;SP 500NM;TS;"	*Set measurement range. Sweep the analyzer.*
50	OUTPUT 723;"MKPK HI;"	*Set marker at highest signal peak.*
60	OUTPUT 723;"MKAR -3DB;"	*Set amplitude-right marker 3 dB below reference marker.*

Description

The MKAR command makes marker #1 a reference marker and marker #3 an active amplitude-right marker. The MKAR command also activates the relative-marker mode, which displays two markers and calculates the wavelength and amplitude difference between them. When this mode is active, the MKA and MKAR commands specify the amplitude difference between the amplitude-right and reference markers. If the vertical scale of the display is logarithmic, the difference between the markers is in decibel units. If the vertical scale is linear, the difference between the markers is a power ratio. In the syntax diagrams, X represents the ratio of the marker amplitudes.

Use MKAR with the MKAL and MKBW or MKBWA commands to simplify bandwidth measurements.

If marker #1 is not displayed, MKAR positions it at the center of the trace. If more than one trace is displayed, MKAR positions marker #1 on the active trace or on trace A, B, or C, in that order; or marks the trace specified by MKTRACE. Update trace data after executing MKAR.

Query Response

```
→ number → output
         termination →
```
qgennum

See Also MKAL, MKBW, MKBWA, MKREAD, MKOFF, MKPX and commands affecting amplitude such as MEASURE, AMPCOR, ROFFSET

MKBW

Display wavelength difference between #2 and #3 markers.
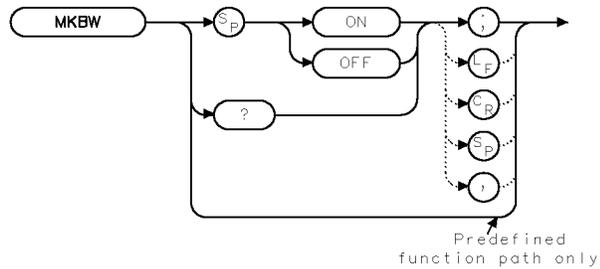
Item	Description
Preset State	MKBW OFF

Example

50 OUTPUT 723;"MKPK HI;"	*Set marker at highest signal peak.*
60 OUTPUT 723;"MKAL —60DB;"	*Set amplitude-left marker 60 dB below reference marker.*
70 OUTPUT 723;"MKAR —60DB;"	*Set amplitude-right marker 60 dB below reference marker.*
80 OUTPUT 723;"MKBW ON;"	*Display wavelength difference between amplitude-left and amplitude-right markers.*
90 OUTPUT 723;"MKBW?;"	*Query wavelength difference.*

Description The MKBW command displays the wavelength difference between markers #2 and #3, even if the markers are not displayed. `MKBW?` sends the bandwidth value to the computer. `MKBW ON` displays the bandwidth in the message area. `MKBW OFF` blanks the message.

MKBW can be used as a source of another command. Use MKBW as a source by incorporating the MKBW statement into the source of any spectrum analyzer command having *predefined function* in its syntax diagram. When used as a source, MKBW uses the wavelength difference for operation by the command that contains MKBW.

Display
Bandwidth,
On/Off Syntax
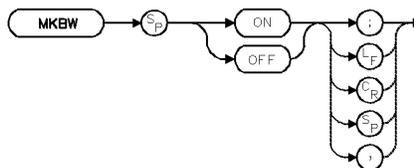
Predefined-
Function
Syntax

Query Syntax
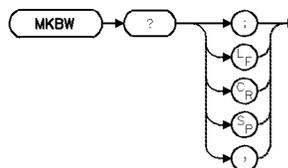
Query Response A zero is returned after `MKBW OFF` is executed.
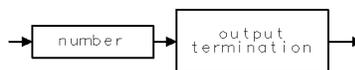
qgennum

See Also MKAL, MKAR, MKBWA

MKBWA

Display wavelength difference X dB below marker peak.
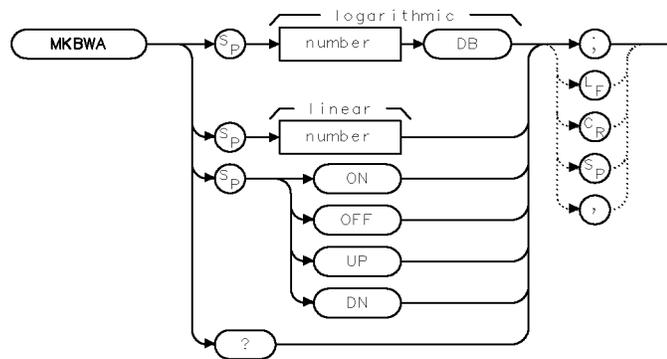
XMKBWA

Item	Description
Preset State	MKBWA 3DB
Linear Mode	Use bypass path; do not send number as dB units
LOG Mode	Use dB path; include dB with the number.

Example

10 OUTPUT 723;"MKBWA −6DB;"	*Set amplitude markers 6 dB below marker peak.*
20 OUTPUT 723;"MKBWA ON;"	*Display wavelength difference between markers.*
30 OUTPUT 723;"MKBW?;"	*Query wavelength difference.*

Description   The MKBWA command displays a signal's bandwidth at a user-specified amplitude below the signal's peak. Two markers are automatically placed at the amplitude offset you specify, and then the wavelength difference between the markers is calculated and displayed. First, use the command to specify the amplitude to perform the bandwidth measurement. Then, turn MKBWA on to view the bandwidth.

Query Response

```
──► │ number │ ──► │ output      │ ──►
                   │ termination │
```
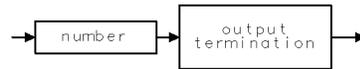
qgennum

See Also MKBW

MKCONT

Enable the sweep to continue past the marker.
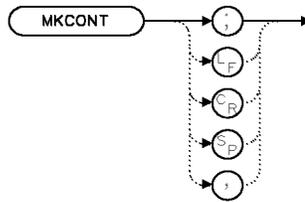
XMKCON

Example   10   OUTPUT 723;"MKCONT;"

Description The MKCONT command enables the sweep to continue past the marker by turning off the MKSTOP command. Use the MKCONT command to reestablish normal sweeps after the MKSTOP command is used. MKSTOP stops the sweep from continuing past any displayed marker.

See Also MKSTOP

MKCWL Set center wavelength equal to wavelength at marker.
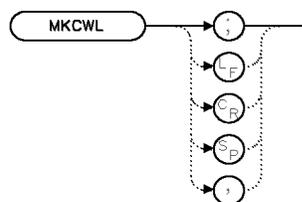
XMKCWL

Example	40 OUTPUT 723;"TS;"	*Sweep measurement range.*
	50 OUTPUT 723;"MKN EP;"	*Move the marker using the keypad, step keys, or knob. Press* ACCEPT VALUE.
	60 OUTPUT 723;"MKCWL;"	*Move marker to center wavelength.*
	70 OUTPUT 723;"TS;"	*Sweep.*

Description The MKCWL command centers the active marker and the trace point it marks by setting the center wavelength equal to the marker wavelength. If the active marker is a relative type that has been Activated by the MKD, MKAL, or MKAR commands, an error condition occurs. If no marker is displayed, MKCWL positions a marker at the center of the trace. If more than one trace is displayed, MKCWL positions it on the active trace or on trace A, B, or C, in that order; or marks the trace specified by MKTRACE. Update the trace data after executing MKCWL.

See Also MKTRACE, MKOFF

MKD

Position marker relative to reference marker, according to wavelength.
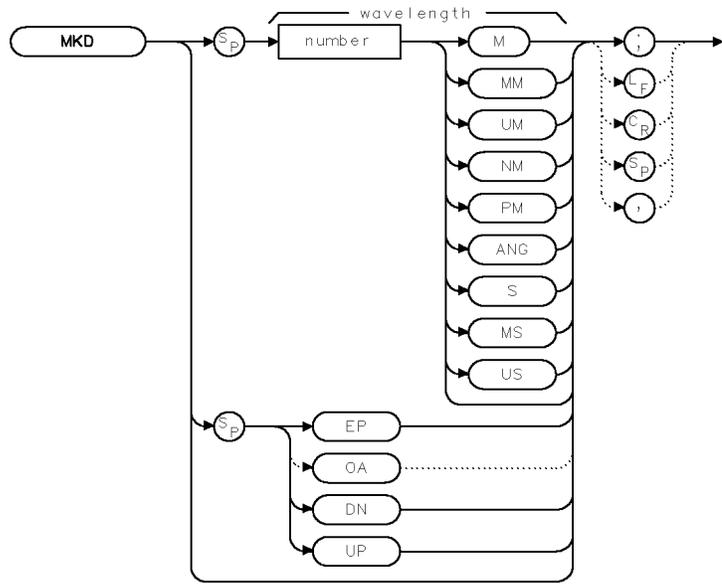
Item	Description
Wavelength Range	± 600 nm to 1700 nm
Step Increment	1 major horizontal scale division

Example Use markers to find the two highest signal levels displayed.

40	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
50	OUTPUT 723;"STARTWL 600NM;STOPWL 850NM;"	*Set measurement range.*
60	OUTPUT 723;"TS;"	*Sweep trace A.*
70	OUTPUT 723;"MKPK HI;"	*Place marker on highest signal peak.*
80	OUTPUT 723;"MKD;"	*Activate the delta marker mode. Since no value was specified with MKD, it places the delta marker at the position of the first marker. The delta marker is now active and under your control.*
90	OUTPUT 723;"MKPK NH;"	*Move the delta marker to the next highest signal level.*

Description MKD positions a second marker on a trace *relative* to the position of the active marker, called the reference marker. *Once positioned*, the new marker becomes the active marker and is called a delta marker. The delta marker is positioned in terms of wavelength, or time if the analyzer is in zero span mode. Thus, MKD 0 places a delta marker at the same wavelength as on the reference marker. MKD 10xxx places a delta marker at a wavelength 10 xx higher than the reference marker. If no marker is active, MKD places two markers at the center of the active trace. If more than one trace is active, MKD places the marker on trace A, B, or C, in that order; or on the trace specified by the MKTRACE command.

The MKD command also activates the relative-marker mode, which displays two markers and calculates the wavelength and amplitude difference between them. When this mode is active, both the MKWL and MKD commands specify the wavelength difference between the delta and reference markers.

See Also SP, MKTRACE

MKDACT?

Report which marker is the currently active delta marker.
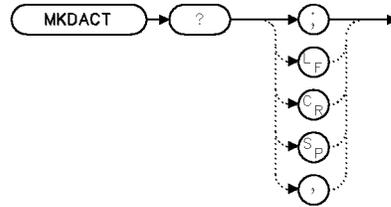
xmkact

Example   100   OUTPUT 723;"MKDACT?;"

Description The MKDACT? command reports the marker number of the currently active delta marker. The marker numbering is the same as for MKACT. If no delta marker is currently active, a value of 0 is returned.

Query Response
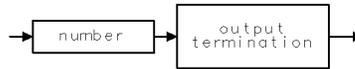
qgennum

See Also MKACT

MKDREFA

Sets a reference marker for a delta marker at a specific amplitude. Marker position is not restricted to a trace point.
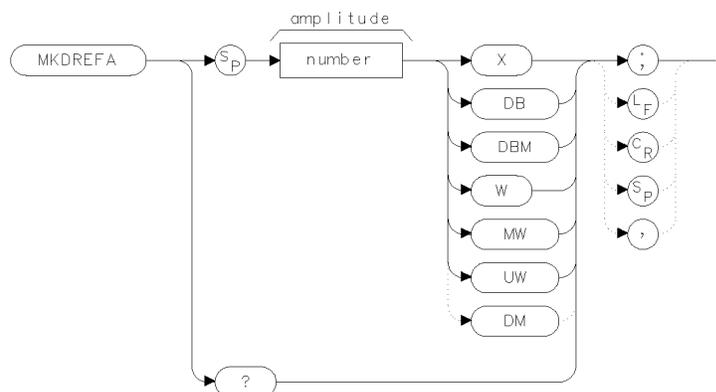
xmkdra

Item	Description
Range	±300 dB

Example

```
100   OUTPUT 723;"MKDREFA -23.5DB;"     Set the marker's vertical position.
110   OUTPUT 723;"MKDREFF 1300NM;"      Set the marker's horizontal position.
```

Description The MKDREFA command, in conjuction with the MKDREFF command, positions the general reference marker anywhere on the display. (Normal active markers must reside upon a trace point, you can set only the marker's wavelength or amplitude, but not both. Refer to MKN, MKAL or MKAR.)

When the vertical scale is in logarithmic mode, the amplitude units are in dBm; in linear mode, the amplitude units are in watts (W, MW, or μW).

When relative-marker mode is activated with the MKAL or MKAR command, the MKDREFA command specifies the amplitude difference from the reference marker. When the vertical scale is in logarithmic mode, the amplitude units are in dBm; in linear mode, the amplitude units are in ratio, as indicated with the X in the syntax diagram.

Query Response
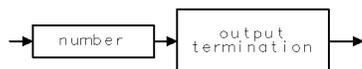
qgennum

See Also MKDREFF, MKA, MKWL

MKDREFF

Set the reference marker for a delta marker to a specific wavelength (or time in zero span). Marker position is not restricted to a trace point.
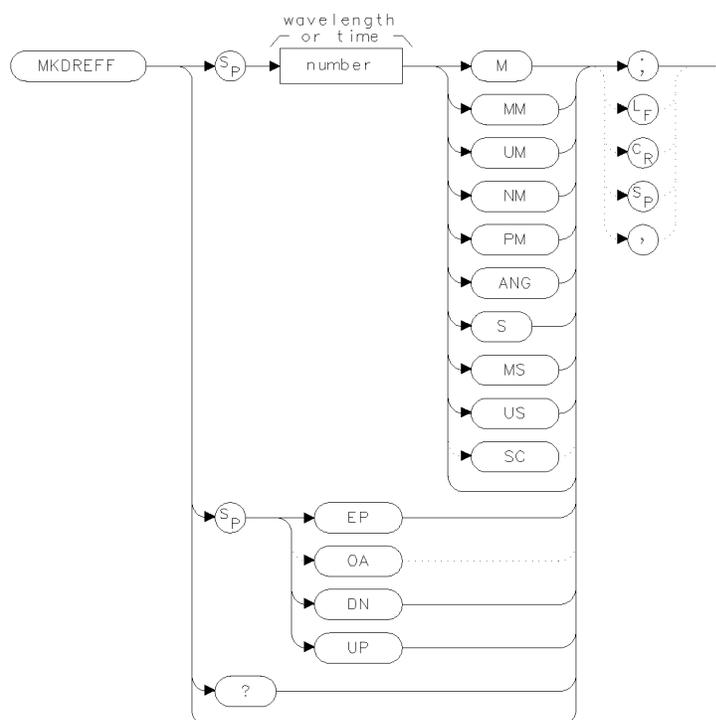
xmkrff

Item	Description
Range	600 nm to 1700 nm

Example

```
100  OUTPUT 723;"MKDREFA -23.5DB;"     Set the marker's vertical position.
110  OUTPUT 723;"MKDREFF 1300NM;"      Set the marker's horizontal position.
```

Description The MKDREFF command, in conjuction with the MKDREFA command, positions the general reference marker anywhere on the display. This is in contrast to normal active markers which must reside upon a trace point. The MKDREFA command must be issued before the MKDREFF command. If the active marker is not displayed, MKDREFF places it at the specified wavelength. If the position of the marker is not specified, MKDREFF places it at center screen.

For wavelength spans greater than zero, the MKDREFF command returns the wavelength of the active marker, or the wavelength difference between the reference and delta marker when the relative-marker mode has been activated by the MKD command. For wavelength spans equal to zero, MKDREFF returns the marker position in terms of time. As a predefined variable, MKDREFF represents either the wavelength or time position of the active marker and can be very useful during math operations and data analysis. The marker readout and returned information varies depending on span and marker-readout setting selected by the MKREAD command.

Query Response

```
→■→ number →■→ output
            termination →■→
```
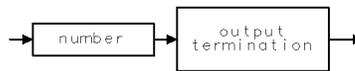
qgennum .

See Also MKDREFA, MKWL, MKA

MKMIN   Move displayed marker to lowest signal level in trace.
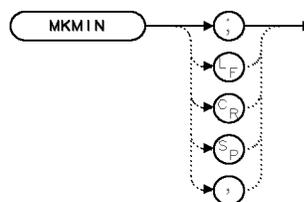
XMKMIN

Example   `60  OUTPUT 723;"MKMIN;"`   *Find lowest signal level.*

Description The MKMIN command moves the active marker to the minimum value
detected. MKMIN operates on the displayed, active trace, or if more than one
is displayed, it operates on trace A, B, or C, in that order; or on the trace
specified by MKTRACE.

See Also MKPX, SMOOTH, VAVG, VB, RB, MKTRACE

MKN

Position active marker on trace according to wavelength.

XMKN
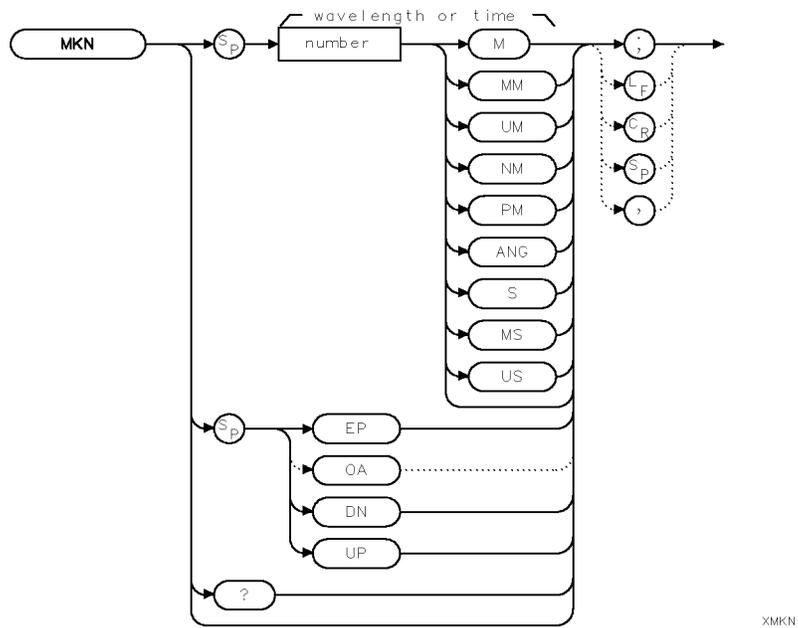
Item	Description
Wavelength Range	600 nm—1700 nm

Example   50   OUTPUT 723;"MKN 800NM;"   *Set marker at 800 nanometers.*

Description The MKN command places the active marker on a trace at the specified wavelength or, during zero span mode, at the specified time. If the position of the marker is not specified, MKN places it at the center of the trace.

MKN also activates the normal marker mode, where the wavelength and amplitude of the active, displayed marker is annotated in the upper-right corner of the graticule. MKN turns off all other marker functions that display marker information in other forms, including the following functions: MKBW, MKAL, MKAR, MKD, MKNOISE, and MKPAUSE.

If more than one trace is displayed, MKN marks the active trace, or marks trace A, B, or C, in that order; or marks the trace specified by MKTRACE. When MKN is turned OFF, the marker is turned off.

Query Response

```
  ┌──────────┐   ┌──────────────┐
→─┤  number  ├─→─┤    output    ├─→
  └──────────┘   │ termination  │
                 └──────────────┘
```
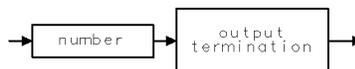
qgennum

See Also SP, MKREAD, MKACT, MKTRACE

MKNOISE

Display or query noise bandwidth at marker.

XMKNOS
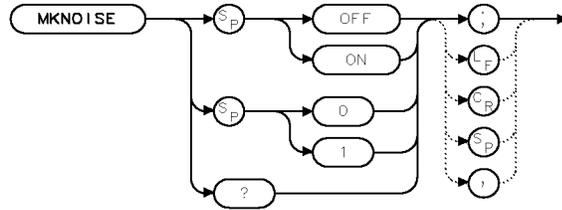
Item	Description
Preset State	MKNOISE OFF

Example Calculate the spontaneous emission power spectral density at the marker position.

10	CLEAR 723	*Initialize analyzer.*
20	OUTPUT 723;"IP;"	
30	OUTPUT 723;"MKN EP;"	*Display the active marker and position it in the noise using the front-panel controls. Press* ACCEPT VALUE *.*
40	OUTPUT 723;"MKNOISE ON;"	*Display the rms noise level at the marker.*
50	OUTPUT 723;"MKA?;"	*Query the noise level to the computer.*
60	ENTER 723;N	*Assign value to computer variable.*

Description The MKNOISE command computes the spontaneous emission power spectral density at the displayed, active marker.

The command averages the trace buckets with 16 of the active markers and normalizes the measured power to 1 nm resolution bandwidth. If no marker is displayed, MKNOISE places the active marker at the center of the active trace. If more than one trace is active, MKNOISE operates on trace A, B, or C, in that order; or on the trace specified by MKTRACE.

Query Response
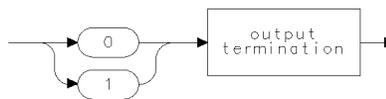
qgen0

See Also   RB, VB, SMOOTH, VAVG, MKTRACE and commands that affect amplitude, such as AMPCOR, RL, MEASURE

MKOFF

Turn off markers.
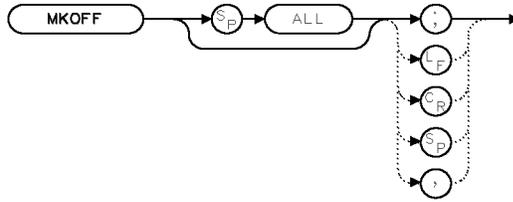
XMKOFF

Example   50   OUTPUT 723;"MKOFF;"   *Turn off all markers.*

Description Use the MKOFF command to turn off the active marker, or all markers. `MKOFF` erases the active marker. `MKOFF ALL` erases all markers.

MKP

Position the active marker on a horizontal trace in position units.

XMKP
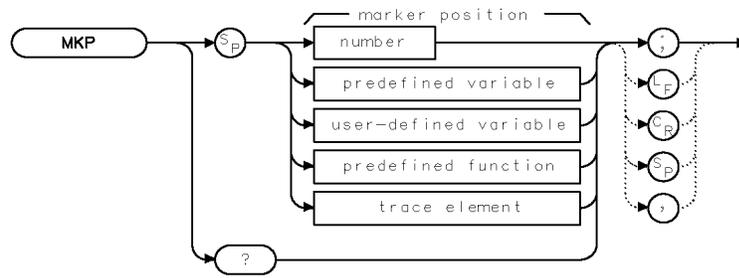
Item	Description
Range	1 – 2048

Example Query the position of a marker in position units.

    20 OUTPUT 723;"MKP 1;"   *Position marker at first trace point.*

Description The MKP command positions the active marker on a trace horizontally in position units.

Position units describe the position of a point along the horizontal axis of a trace. The point at the left end of the trace has a position-unit value of 1. The point at the right end of a 800-point trace has a position-unit value of 800. If more than one trace is displayed, MKP chooses the active trace, or chooses trace A, B, or C, in that order; or chooses the trace specified by MKTRACE.

Query Response Returns the delta value when delta marker mode is on.

qgennum
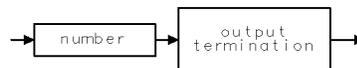
See Also   MKTRACE

MKPABS

Sets the current active marker to the bucket requested even in delta marker mode.
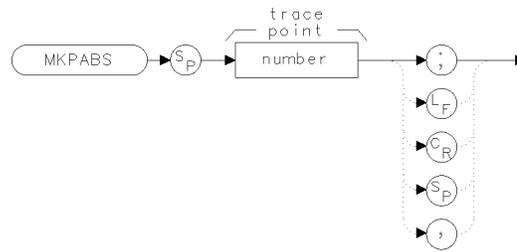
xmkpbs

Item	Description
Range	0 to 2047

Example

```
100  OUTPUT 723;"MKPABS 431;"
```

Description The MKPABS command sets a marker to an absolute bucket. If the value is larger than the current trace length, the marker will be placed on the last trace point. For example, if the trace length was 800, and the value sent was 2000, no error would occur, but the marker would appear on bucket number 799, not number 2000.

If the value is set to a bucket number that is beyond what the optical spectrum analyzer is capable of displaying (not what it is currently displaying) a "Parameter out of range" error occurs.

The query returns the absolute wavelength. Use the MKD command to set a marker relative to the reference marker.

Query Response
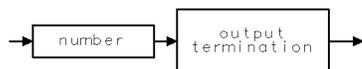
qgennum .

See Also MKFABS

MKPAUSE

Pause the sweep at the displayed active marker for the specified length of time.
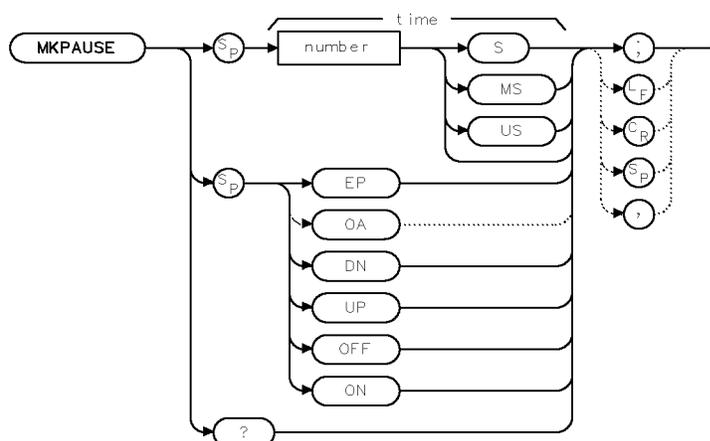
XMKPAS

Item	Description
Time Range	0 – 1000 seconds
Step Increment	1, 3, 10 sequence
Preset State	**MKPAUSE 0** |off|

Example

```
30   OUTPUT 723;"TS;"              Sweep trace A.
40   OUTPUT 723;"MKN;"             Display active marker.
50   OUTPUT 723;"MKPAUSE 1.3S;"    Pause sweep for 1.3 seconds at marker.
```

Description   The MKPAUSE command pauses the sweep at the displayed active marker for the specified length of time.

The delay can be removed by executing **MKPAUSE 0** or by turning off the active marker with the MKOFF command. If no marker is displayed, MKPAUSE first marks the highest level of the active trace, or marks trace A, B, or C, in that order; or marks the trace specified by MKTRACE.

Query Response

```
number → output
           termination →
```
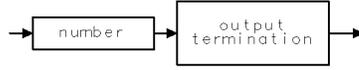
qgennum

See Also MKOFF

MKPITX   Control the excursion value for marker pit operations.
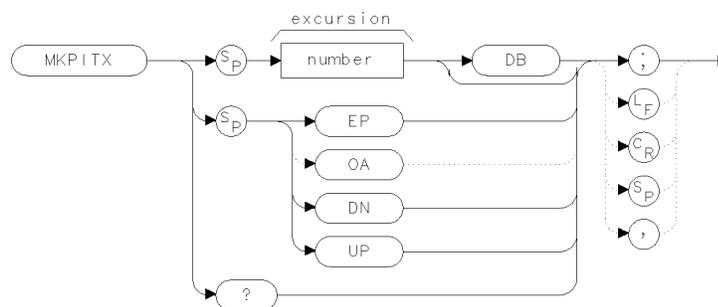
xmkpit

Item	Description
Pit-Excursion Range	0.01 – 300 dB
Step Increment	1 scale division during log mode. 1 scale division at top graticule line |.915 dB| during linear mode.
Preset State	`MKPITX 3DB`

Example   `100  OUTPUT 726;"MKPITX 6DB;"`   *Set the pit excursion value to 6dB.*

Description   Just as MKPX sets the peak excursion value for marker peak operations, MKPITX sets the pit excursion value for marker pit operations. The MKPITX command specifies the minimum signal excursion for the marker-pit functions (MKPK). Signal movement less than the pit-excursion value is not considered a pit. To be a pit, the skirt of the signal response must dip by at least the pit-excursion value, and then rise by at least the pit-excursion value.

The default pit-excursion value is 3 dB for pits displayed on a logarithmic scale. This means that any pit with an excursion of less than 3 dB on either side of its bandwidth shape is not identified as a pit. If the marker-pit command, `MKPK MIPIT`, is executed on such a pit, the optical spectrum analyzer does not place a marker at the pit. For signals displayed on a linear scale, the default pit excursion equals

$$excursion = 1 - 10^{\frac{-MKPITX}{200}}$$

Note that if the pit excursion is too small, noise pits are identified as pits. Use VAVG or SMOOTH to minimize noise pits so that signal peaks can be distinguished from noise pits.

Query Response

```
number → output termination →
```
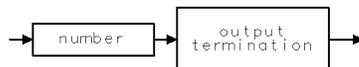
qgennum

See Also MKPK, MKPX

MKPK

Position active marker at a peak (or pit) on the trace.
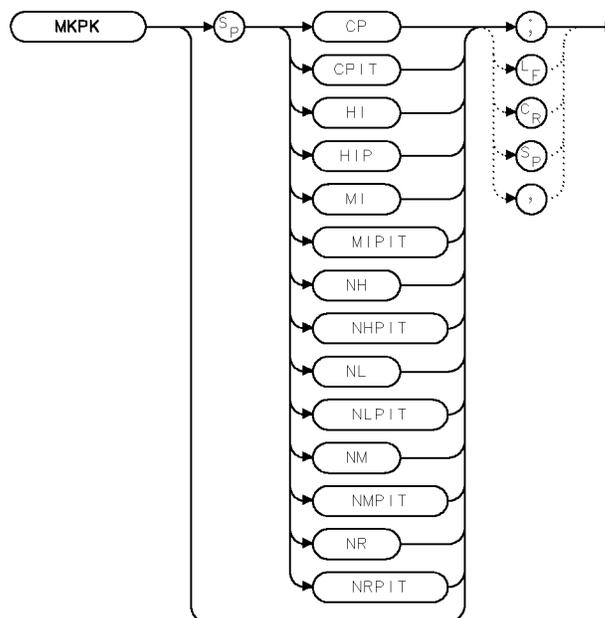
xmkpk

Example

30	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
40	OUTPUT 723;"STARTWL 600NM;STOPWL 1700NM;"	*Set measurement range.*
50	OUTPUT 723;"TS;"	*Sweep the measurement range.*
60	OUTPUT 723;"MKPK HI;"	*Move the active marker to the highest signal level (amplitude) on the trace.*
70	OUTPUT 723;"MKD;MKPK NH;"	*Move the delta marker to the next highest signal level on the trace.*
80	OUTPUT 723;"MKPK NR;"	*Move the marker to the next signal peak of higher wavelength.*

Description The MKPK command parameters reposition the *active* marker on the trace as follows:

Argument	Definition
CP	Moves the displayed, active marker to the closest peak.
CPIT	Select the closest pit.
HI	Moves the active marker to the highest point on the trace.
HIP	Move the displayed, active marker to the highest peak (as opposed to the highest point).
MI	Select the minimum peak (this is NOT the same as the minimum point, which is selected by MKMIN).
MIPIT	Select the lowest pit. A pit is an inverse peak: it consists of a local minimum point with a positive excursion on each side.
NH	Moves the displayed, active marker to the next highest signal level detected.
NHPIT	Select the next highest pit. This moves through the pits in the opposite direction of NLPIT.
NL	Moves the displayed, active marker to the next signal peak of lower wavelength (to the left).
NLPIT	Select the next left pit.
NM	Select next minimum peak. Just as NH (next highest) goes to the next highest peak, this goes to the next lowest peak.
NMPIT	Select the next lowest pit. This would be used to move up after using MKPIT.
NR	Moves the displayed, active marker to the next signal peak of higher wavelength (to the right).
NRPIT	Select the next right pit.

The peak-excursion command, MKPX, determines what constitutes a peak in a trace. The pit-excursion command, MKPITX, determines what constitutes a pit in a trace.

See Also MKPX, VAVG, SMOOTH, VB, MKTRACE

MKPX

Select minimum signal excursion for peak signal identification.
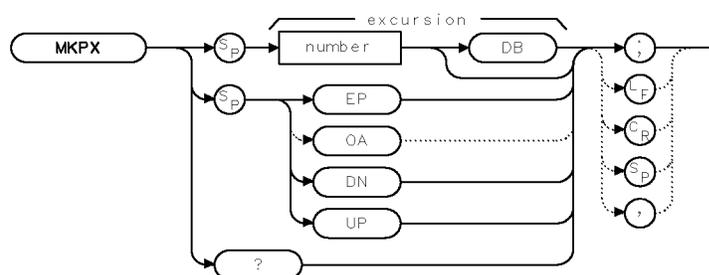
Item	Description
Peak-Excursion Range	0.01 – 300 dB
Step Increment	1 scale division during log mode. 1 scale division at top graticule line |.915 dB| during linear mode.
Preset State	`MKPX 3DB`

Example Set the marker-peak excursion to 20 dB, so that only very large signal excursions are recognized by the marker-peak and signal-identification functions.

```
70  OUTPUT 723;"MKPX 20DB;"        Set peak excursion to 20 dB.
```

Description The MKPX command specifies the minimum signal excursion for signal identification and the marker-peak functions (MKPK). Signal movement less than the peak-excursion value is not considered a peak. To be a peak, the skirt of the signal response must rise by at least the peak-excursion value, and then fall by at least the peak-excursion value.

The default peak-excursion value is 3 dB for signals displayed on a logarithmic scale. This means that any signal with an excursion of less than 3 dB on either side of its bandwidth shape is not identified as a signal. If the marker-peak command, `MKPK HI`, is executed on such a signal, the analyzer does not place a marker at the signal peak. For signals displayed on a linear scale, the default peak excursion equals

$$excursion = 1 - 10^{\frac{-MKPX}{200}}$$

Note that if the peak excursion is too small, noise spikes are identified as signals. Use VAVG or SMOOTH to minimize noise spikes so that signal peaks can be distinguished from noise spikes.

Query Response

```
  ┌──────────┐   ┌──────────────┐
─→│  number  │─→│   output     │─→
  └──────────┘   │ termination  │
               └──────────────┘
```
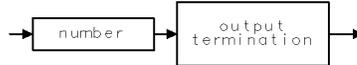
qgennum

See Also MKPK, PEAKS and marker functions.

MKREAD

Select the active trace information type displayed by the marker readout.
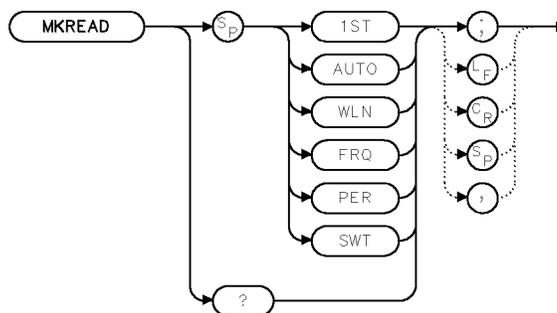
xmkred

Item	Description
Preset State	`MKREAD AUTO`

Example

```
10  OUTPUT 723;"TS;MKN;"        Sweep. Activate marker.
20  OUTPUT 723;"MKREAD WLN;"    Display wavelength at marker position.
```

Description The MKREAD command selects the type of active trace information displayed by the marker readout and queried with the `MKWL?`, `MKT?`, and `MKA?` commands.

The information listed by the marker varies depending on whether the analyzer is in zero-span mode or sweeping a wavelength range. Use MKREAD to set the readout to one of the settings listed in the following table.

Marker Readout Information

Command Argument	Contents of Marker Readout	Wavelength Query (MKWL?)	Time Query (MKT?)	Amplitude Query (MKA?)
WLN	Wavelength at marker position	Gives wavelength at marker position	Gives marker position in units of sweep time |marker sweep time|	Gives marker amplitude
FRQ	Frequency at marker position	Gives frequency marker position	Gives marker position in units of sweep time |marker sweep time|	Gives marker amplitude
PER	Marker period |equals inverse of marker wavelength|	Marker period |equals inverse of marker wavelength|	Gives marker position in units of sweep time	Gives marker amplitude
SWT	Marker position in units of sweep time	Gives marker wavelength	Gives marker position in units of sweep time	Gives marker amplitude
IST	Inverse of sweep time	Gives marker wavelength	Gives 1/sweep-time	Returns marker amplitude
AUTO	Wavelength at marker position |during zero-span mode, marker position in units of sweep time|	Gives marker wavelength	Gives marker position in units of sweep time	Gives marker amplitude

Query Response
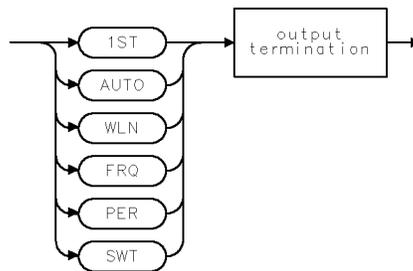
qmkred

See Also   MKN, MKD, MKAL, MKAR, MKWL, MKT, MKA, SP, MEASURE

MKRL

Move the active marker to the reference level.
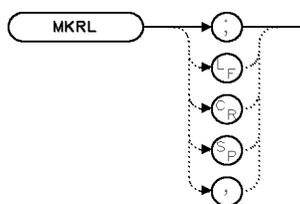
XMKRL

Example

```
50  OUTPUT 723;"SNGLS;TS;"    Sweep trace A.
60  OUTPUT 723;"MKPK HI;"     Place marker at highest signal.
70  OUTPUT 723;"MKRL;"        Set reference level equal to marker amplitude.
80  OUTPUT 723;"TS;"          Sweep new measurement range.
```

Description The MKRL command sets the reference level equal to the amplitude of the displayed active marker. Update trace data after executing MKRL. An error condition occurs if the active marker is a relative type that has been activated by the MKD, MKAL, or MKAR commands.

MKSP   Set span equal to wavelength difference of markers.

XMKSP

Example

60	OUTPUT 723;"SNGLS;TS;"	*Activate single-sweep mode.*
70	OUTPUT 723;"MKN;MKPK HI;"	*Mark the highest point on the trace.*
80	OUTPUT 723;"MKD;"	*Activate the delta marker.*
90	OUTPUT 723;"MKPK NH;"	*Move the delta marker to the second highest peak.*
100	OUTPUT 723;"MKSP;TS;"	*Set the span equal to the wavelength difference between the markers.*

Description The MKSP command operates only during the relative-marker mode (delta). MKSP is inoperative after MKAL or MKAR is executed. If MKD has been executed, MKSP sets the start and stop wavelength equal to the wavelength of the delta marker and reference marker. The start and stop wavelength correspond to the markers on the left and right, respectively.

See Also MKD, MKAL, MKAR

MKSS

Set center-wavelength step size equal to the wavelength difference between delta markers.

XMKSS

Example

30	OUTPUT 723;"STARTWL 600NM;STOPWL 800NM;"	*Set measurement range.*
40	OUTPUT 723;"MKN EP;"	*Move marker to fundamental, using front-panel knob, then press* ACCEPT VALUE *key.*
50	OUTPUT 723;"MKCWL;"	*Move the marked signal to the center of the display.*
60	OUTPUT 723;"MKD;"	*Turn on delta marker.*
65	OUTPUT 723;"MKPK NH;"	*Move the marker to the next highest peak.*
70	OUTPUT 723;"MKSS;"	*Set the center-wavelength step size equal to the delta marker wavelength separation.*

Description The MKSS command uses delta marker information to set the center-wavelength step size. Delta markers must be displayed before using this command. MKSS is inoperative when executed after MKAL or MKAR. MKSS sets the center-wavelength step size equal to the wavelength difference of the delta and reference markers.

See Also SS, MKD

MKSTOP   End the sweep at the current active marker.
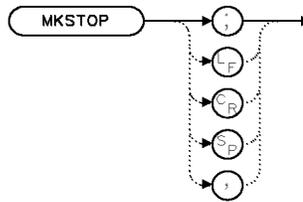
XMKSTP

Example   `10  OUTPUT 723;"MKSTOP;"`

Description The MKSTOP command ends the sweep at the current active marker instead of at the end of the trace when in continuous sweep mode. Sweep time can be reduced by not sweeping past the marker. The next sweep starts the trace over.

See Also MKCONT, MKN

MKT   Position marker in units of time.
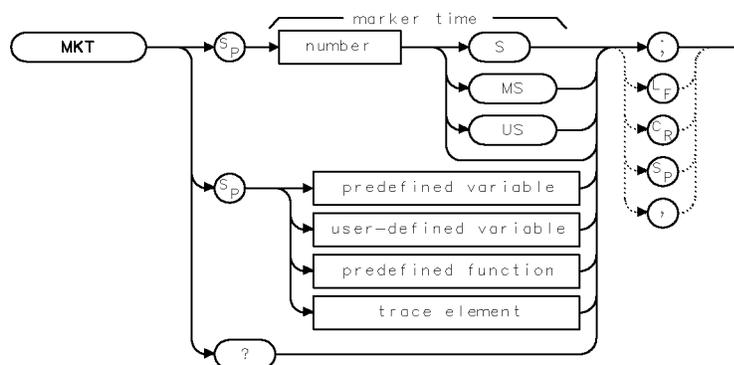
xmkt

Item	Description
Marker-Time Range	± 1000 seconds
Step Increment	1 major horizontal scale division.

Example

30	OUTPUT 723;"SP 0;SNGLS;TS;"	*Set measurement range to zero span.*
40	OUTPUT 723;"MKN;"	*Display active marker.*
50	OUTPUT 723;"MKT?;"	*Query value of the marker position, in units of time.*

Description Use the MKT command during zero span mode. **MKT?** queries the marker position in units of time, unless **MKREAD IST** has been executed, when **MKT?** gives the marker position as an inverse of the sweep time at the marker. As a predefined variable, MKT represents either the wavelength or the time position of the active marker and can be very useful during math operations and data analysis. MKT can also position the displayed, active marker on a trace in units of time, relative to the left end of the trace. If the marker is a delta marker, the time is figured relative to the reference marker.

Query Response
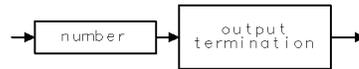
qgennum

See Also MKREAD, SP

MKTRACE

Position the active marker on the specified trace.
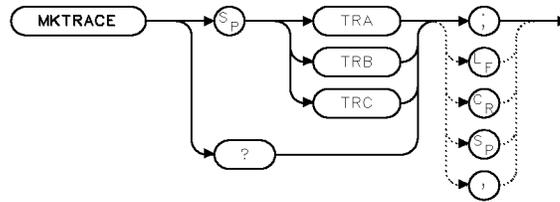
XMKTRC

Item	Description
Preset State	MKTRACE TRA

Example   10   OUTPUT 723;"MKTRACE TRB;"   *Move the marker to trace B.*

Description The MKTRACE command moves the displayed, active marker to the specified trace.

Query Response
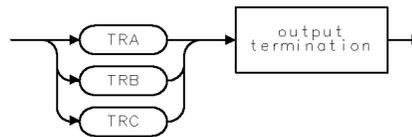
QMKTRC

See Also MKA, MKAL, MKAR, MKD, MKWL, MKMIN, MKN, MKNOISE, MKP, MKPAUSE, MKPK

MKTRACK

Maintain the marked signal at center frequency.
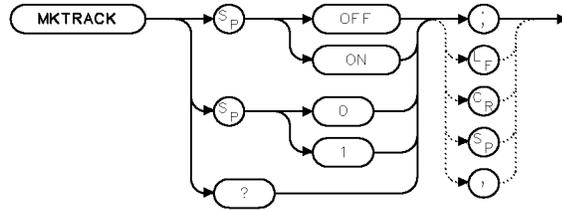
XMKTRK

Item	Description
Preset State	`MKTRACK OFF`

Example

```
50   OUTPUT 723;"MKPK HI;MKTRACK ON;"     Turn on marker tracking.
60   OUTPUT 723;"SP 100NM;"               Reduce the span.
70   OUTPUT 723;"MKTRACK OFF;"            Turn off marker tracking.
```

Description

The MKTRACK command keeps the active marker and the point it marks at the center of the display during changes in wavelength span.

When MKTRACK ON is executed, the spectrum analyzer initiates several sweeps as it reduces the span in stages. If span is reduced significantly when marker tracking is off, the signal is likely to be outside of the measurement range.

Query Response
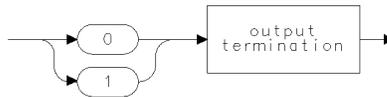
qgen0

See Also

MKTV, SP, ZOOMRB

MKTUNE   Fix tune the optical spectrum analyzer to a specified wavelength.
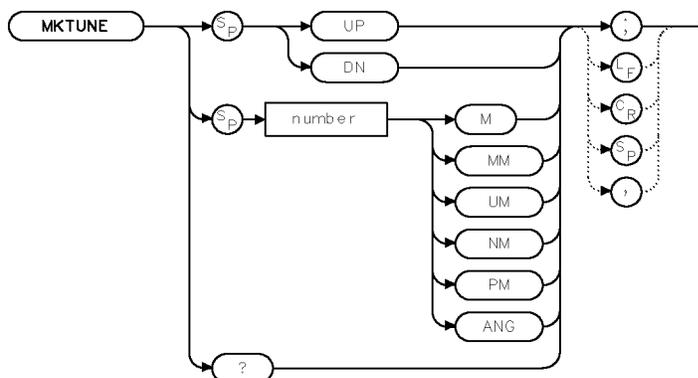
Example   10 OUTPUT 723;"MKTUNE 850NM;"

Description The MKTUNE command places the optical spectrum analyzer in single sweep and tunes to the specified wavelength. A marker is also activated at the wavelength.

N O T E

If INSTMODE is not set to PRESEL, MKTUNE generates an error message.

Query Response

```
  →│ number │→│ output        │→
                │ termination   │
```
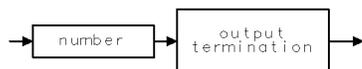
See Also INSTMODE

MKTV

Specify amplitude variance monitored by the marker-tracking function.
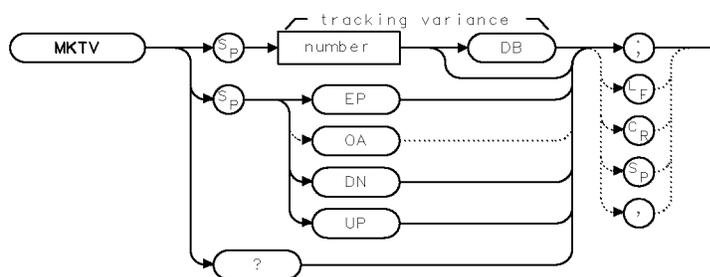
Item	Description
Tracking-Variance Range	0 – 300 dB
Step Increment	1 major vertical scale division during log mode. 1 major vertical scale division at top graticule line |.915 dBm| during linear mode.
Preset State	`MKTV 5DB`

Example

`30 OUTPUT 723;"STARTWL 600NM;STOPWL 1700NM;"`	*Set measurement range.*
`40 OUTPUT 723;"TS;MKPK HI;"`	*Sweep trace A. Place the active marker on the highest signal peak.*
`50 OUTPUT 723;"MKTV 1DB;"`	*Change marker-tracking variance to 1 dB.*
`60 OUTPUT 723;"MKTRACK ON;"`	*Turn on marker tracking.*
`70 OUTPUT 723;"SP 200NM;"`	*Change the span.*

Description The MKTV command specifies the amplitude variance monitored by the marker-tracking function (MKTRACK) to determine the proper signal to track.

With each sweep, the marker-tracking function verifies that the marker is kept on the same signal by comparing the amplitude of the currently marked signal to the amplitude of the signal marked during the previous sweep. If the two amplitudes are not within the tracking-variance limit, the marker is moved to an adjacent signal and amplitudes are compared again. Reduce the marker-tracking variance when tracking a signal that has an amplitude close to the level of the other signals in the measurement range.

Query Response
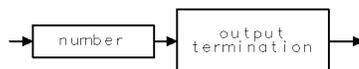
qgennum

See Also   MKTRACK

MKTYPE   Return the type of active marker.
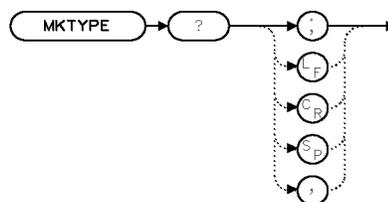
XMKTYP

Example
```
30  OUTPUT 723;"MKN;"        Activate normal marker.
40  OUTPUT 723;"MKTYPE?;"    Send marker type to computer.
50  ENTER 723;A$             Assign response to computer variable.
```

Description   The MKTYPE query returns one of the four types of active markers:

- Position marker (normal)

- Delta marker

- Amplitude-left marker

- Amplitude-right marker

The position marker (*PSN*) is activated by MKN and is positioned at any point on the trace meeting the wavelength or time criteria. MKPK, MKPAUSE, MKTRACK, and MKNOISE also activate the position marker. The delta marker (*DELTA*) is activated by MKD and is positioned at a point, either side of the reference marker, that meets the wavelength or time criteria. The amplitude-right marker (*AMPR*) is activated by MKAR and is positioned to the right of marker #1 at the first point that meets the amplitude criteria. The amplitude-left marker (*AMPL*) is activated by MKAL and is positioned to the left of marker #1 at the first point that meets the amplitude criteria.

Query Response
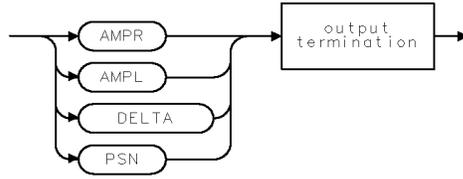
QMKTYP

MKWL

Position active marker according to wavelength, or return the marker wavelength.
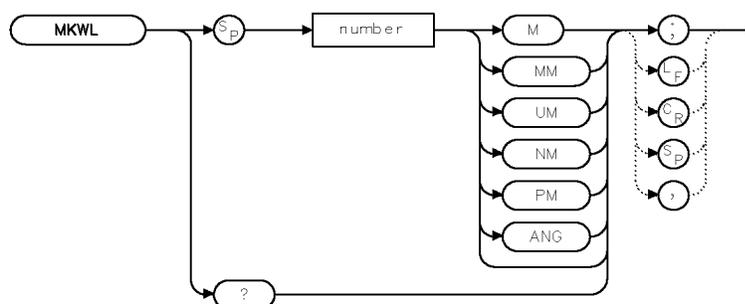
xmkwl

Item	Description
Range	600 nm to 1700 nm

Example   10 OUTPUT 723;"MKWL 750NM;"

Description For wavelength spans greater than zero, the MKWL command returns the wavelength of the active marker. MKWL can also return the wavelength difference between the reference and delta marker when the relative-marker mode has been activated by the MKD command.

For wavelength spans equal to zero, MKWL sends the marker position in terms of time. As a predefined variable, MKWL represents either the wavelength or the time position of the active marker and can be very useful during math operations and data analysis. The marker readout and queried information varies depending on span and the marker-readout setting selected by the MKREAD command.

MKWL can also be used to position the active marker on the trace in terms of time or wavelength. If the active marker is not displayed, MKWL places it on the trace at the specified wavelength. If the position of the marker is not specified, MKWL places it at the center of the trace. If more than one trace is

displayed, MKWL marks the active trace, or marks trace A, B, or C, in that order; or marks the trace specified by MKTRACE.

Query Response

```
         ┌──────────┐   ┌──────────────┐
─────→───┤  number  ├─→─┤    output    ├─→─────
         └──────────┘   │ termination  │
                        └──────────────┘
```
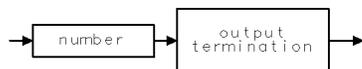
qgennum

See Also SP, MKTRACE, MKREAD, or commands affecting wavelength such as WLOFFSET, CENTERWL, SP, STARTWL, STOPWL

MOD

Retain remainder after division.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	±1.8 × 10^{308} when using *number, predefined variable* or *user-defined variable*.
	Zero not allowed for source 2.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Set the measurement range to the remainder of the center wavelength divided by 100 nm.

30	OUTPUT 723;"MOD SP,CENTERWL,100NM;"	*Set span equal to remainder of center wavelength divided by 100 nm.*
40	OUTPUT 723;"TS;"	*Sweep new measurement range.*

Description The MOD command stores in the destination the remainder left by the division of source 1 by source 2.

Source 2 cannot be equal to 0, or an error results. When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep with the SNGLS and TS commands.

MODADD

Return the HP-MSIB address of the specified slave module.
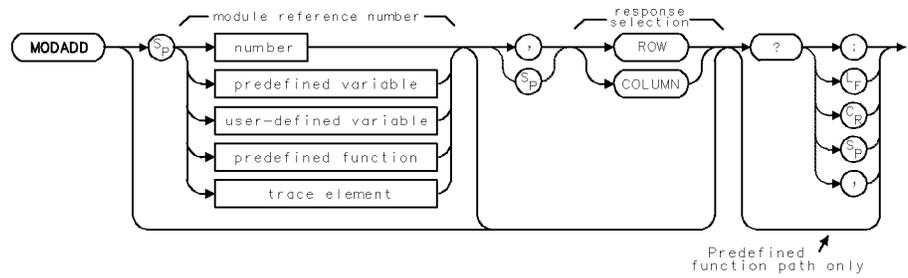
Item	Description
Reference-Number Range	1 to hardware limit

Example

```
30   OUTPUT 723;"MODADD?;"
40   ENTER 723;N,M,O,P,Q,R,S,T
```

Query HP-MSIB address of all modules.
Assign row and column addresses to computer variables.

Description The MODADD command returns the HP-MSIB address of the specified slave module. The HP-MSIB address corresponds to a module position in the address map. When either the *row* parameter or *column* parameter is included, MODADD only returns the HP-MSIB row or column address.

Predfined-Function Syntax
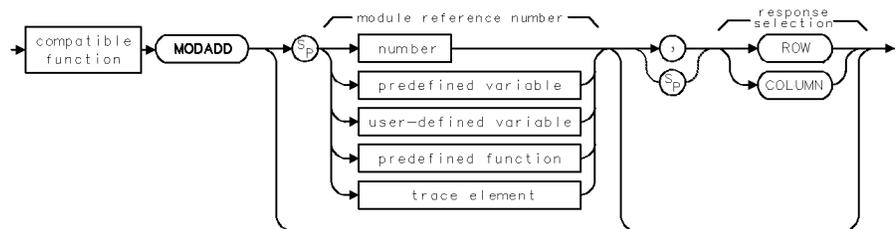
Query Syntax
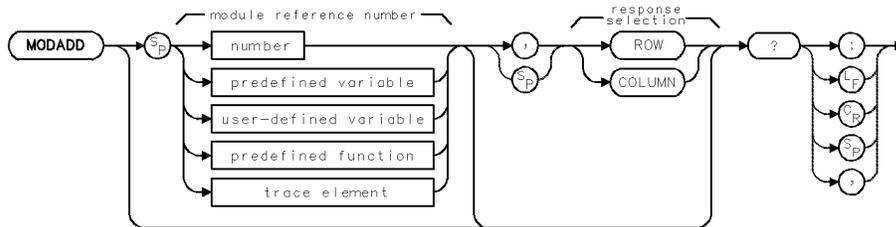
Query Response
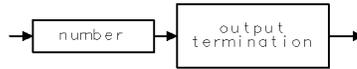
Row or Column is Returned
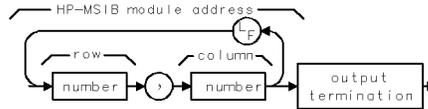
Complete HP-MSIB Addresses are Returned

MODID

Identify module or instrument at HP-MSIB address.
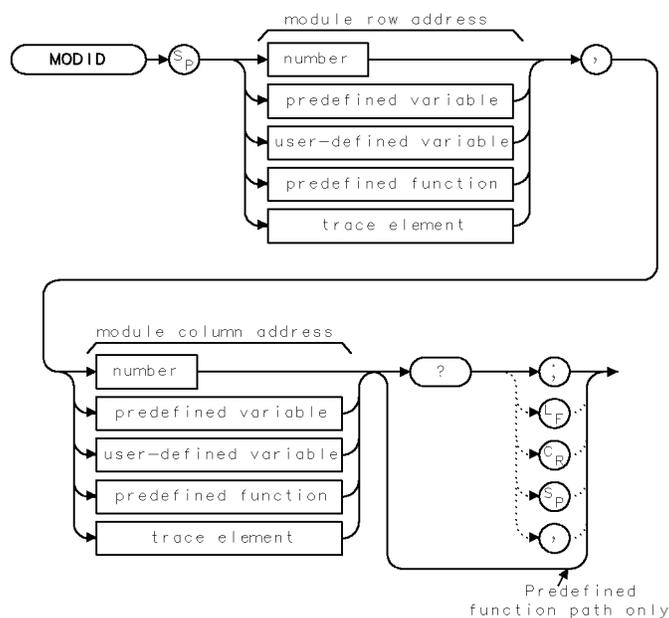
XMODID

Item	Description
Row Range	Integers, 0 – 7
Column Range	Integers, 0 – 31
Restrictions	`MODID 0,31` is illegal

Example

50	`OUTPUT 723;"MODID M_OD,23?;"`	*Query model number of module at HP-MSIB address M_OD,23 (row = M_OD, column = 23).*
60	`ENTER 723;N`	*Assign model number to N.*
70	`OUTPUT 723;"M_OD?;"`	*Query value of M_OD, which is the row number.*
80	`ENTER 723;M`	*Assign row number to computer variable.*
90	`PRINT "THE";N;" MODULE IS IN ROW";M;",COLUMN 23"`	

Description The MODID command identifies the modular spectrum analyzer module or instrument located at the specified HP-MSIB address.

MODID must be used as either a query or as a source in another analyzer-command function. Form a query by ending the MODID statement with a question mark (?). When used as a query, MODID sends the model number of the module at the specified HP-MSIB address to the computer.

Use MODID as a source by incorporating the MODID statement into the source of any spectrum-analyzer command that contains *predefined function* in its syntax diagram. When MODID is used as a source, the HP-MSIB address of the specified module is used for operation by the command that contains MODID.

Predefined-Function Syntax
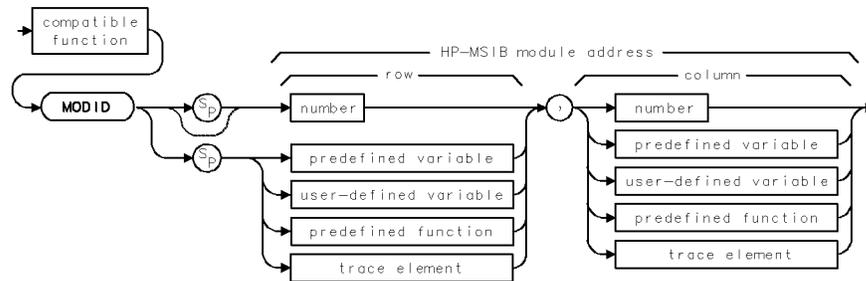
Query Syntax
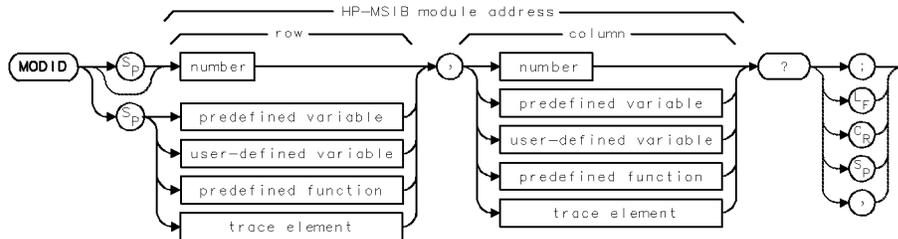
Query Response The model number less any alpha-character prefix or suffix is sent. If no module is present or the address is not within the optical spectrum analyzer's control area (slave area), a 0 is given. Alpha characters are converted as follows:

A becomes .0	F becomes .5
B becomes .1	T becomes .6
C becomes .2	H becomes .7
D becomes .3	I becomes .8
E becomes .4	J becomes .9

```
  →  ┌─→┌──────────┐──→┌──────────────┐──→
     │  │  number  │   │   output     │
     └──┤          ├──→│  termination │
        └──────────┘   └──────────────┘
```
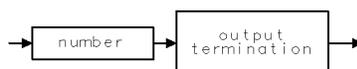
qgennum .

MOV

Transfer value of trace, variable, or number to a trace or variable.

Item	Description
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Position a signal peak at the display line.

20 OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
30 OUTPUT 723;"STARTWL 600NM;STOPWL 1700NM;"	*Set measurement range.*
40 OUTPUT 723;"TS;MKPK HI;"	*Sweep trace A. Place a marker at the highest signal level detected.*
50 OUTPUT 723;"DL ON;"	*Turn on the display line.*
60 OUTPUT 723;"MOV DL,MKA;"	*Set the display-line value equal to the marker amplitude by storing the value of the marker-amplitude variable, MKA, in the display-line variable, DL.*

Destination The MOV command stores the contents of the source in the destination. When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep using the SNGLS and TS commands. The default is 800 position units when using *TRA, TRB, TRC* and is truncated if the trace is longer than destination length. If the trace is shorter than the destination length, the last element is replicated to complete the trace.

MPY

Multiply sources and store in the destination.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm$ 1.8 $\times$ 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	3 – 2048 position units.

Example

```
10  OUTPUT 723;"MPY CENTERWL,CENTERWL,2;"    Double the center wavelength.
```

Description The MPY command multiplies the sources, point by point, and stores the contents in the destination.

In case of overflow, an error is reported and the result is limited to the amplitude range. When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep using the SNGLS and TS commands.

The default is 800 position units when using *TRA, TRB, TRC* and is truncated if the trace is longer than destination length. If the trace is shorter than the destination length, the last element is replicated to complete the trace.

MSG

Return the characters displayed in the message area.
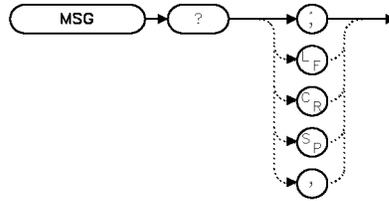
XMSG

Example

40	OUTPUT 723;"MSG?;"	*Query the message to the computer.*
50	ALLOCATE Message$[40]	*Create computer array (string) for storage of character message.*
60	ENTER 723 USING "-K";Cal,Cor,Msg$	*Assign the messages to two Boolean variables and one string. (The HP BASIC code for image, −K, enters line feeds into the string.)*
70	PRINT	
80	IF Cal THEN PRINT "uncalibrated"	
90	IF Cor THEN PRINT "uncorrected"	
100	PRINT Msg$[3]	*Print the string, beginning with the third character.*

Description The MSG query returns the characters displayed in the message area on the screen.

The query returns 2 Boolean responses that indicate the presence of UNCAL and UNCOR, respectively. Whenever a message is in the message area, or **UNCAL** or **UNCOR** is displayed, the MESSAGE bit of the status byte is set true.

Query Response
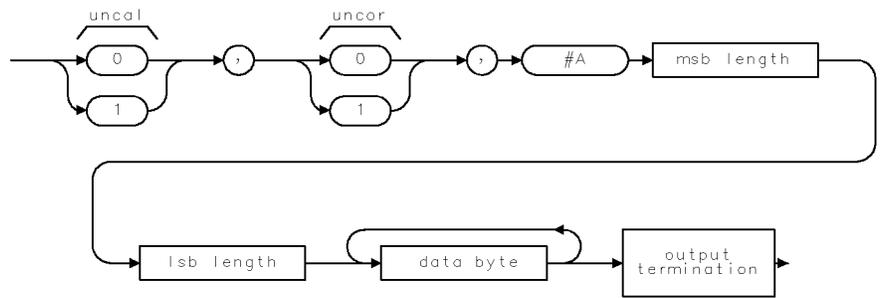
QMSG

See Also USERMSG

MSI

Select location for storing files: HP-IB device, HP-MSIB device, or internal memory.
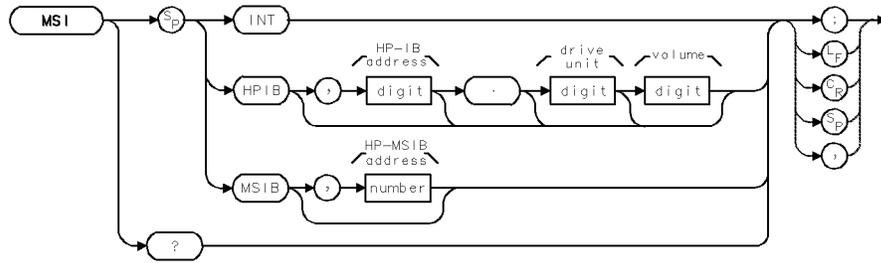
Item	Description
HP-IB Address Range	0 - 7
Drive-Unit Range	0 - 9
Volume Range	0 - 9
HP-MSIB Address Range	0 - 255

Description When power is applied to the spectrum analyzer, internal memory is selected. The IP command does not change the MSI selection.

INT selects internal memory.

HPIB selects external memory that is located on HP-IB, such as a disk drive. Connect the device to the spectrum-analyzer HP-IB port before executing MSI. The MSI command indicates the HP-IB address of the device, and for disk drives, indicates the unit and volume number. The default selection is HP-IB address 0, unit 0, volume 0 (MSI HPIB,0.00).

MSIB selects external memory that is located on the HP-MSIB, such as a memory card. The MSI command indicates the HP-MSIB address of the device. The default address is the HP-MSIB address of the display (for example, the HP 70004) connected to the spectrum analyzer. New addresses must be specified as a single number. Since most storing devices are at row-address 0, the number that MSI specifies is the column address. If you wish to store on a device that is not at row zero, calculate the HP-MSIB address as follows:

$$address = column\ address + [(row\ address)(32)]$$

Query Response
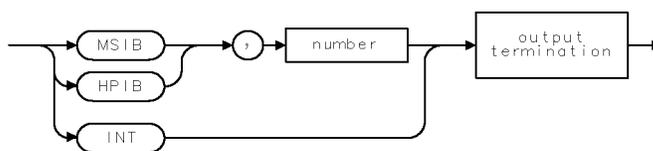
QMSI

See Also FORMAT, LIMIRCL, LIMISAV, LOAD, MODADD, PREFX, PURGE, RCLD, RCLS, RCLT, RCLU, STOR, SAVED, SAVES, SAVET, SAVEU.

MXM

Compare traces and variables, then store greater values.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*. −32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*. ±1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example   Set the display-line level equal to the larger amplitude of two trace elements.

```
40   OUTPUT 723;"SNGLS;"                              Activate single-sweep mode.
50   OUTPUT 723;"SP 500NM;CENTERWL 1000NM;"           Set measurement range.
60   OUTPUT 723;"TS;"                                 Sweep measurement range.
70   OUTPUT 723;"DL ON;"                              Turn on the display line.
80   OUTPUT 723;"MXM DL,AMPU TRA[1],AMPU TRA[400];"   Compare the amplitude of the first and four-hundredth
                                                      elements of trace A, then set the display line equal to
                                                      the larger one.
```

Description   The MXM command compares the sources, point by point, sending the greater value of each comparison to the destination. If one of the sources is a single value, it acts as a threshold and all values greater than the threshold are sent to the destination.

The default is 800 position units when using *TRA, TRB, TRC* and is truncated if the trace is longer than destination length. If the trace is shorter than the destination length, the last element is replicated to complete the trace.

MXMH

Activate maximum hold.
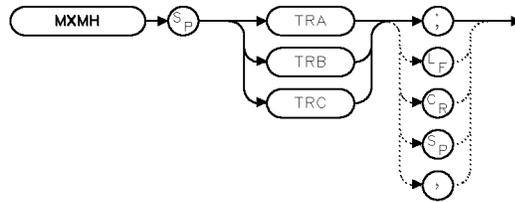
Item	Description
Preset State	Inactive

Example   40   OUTPUT 723;"MXMH TRA;"   *Write data into trace A. The value of the incoming data is compared with the current data, then the maximum of the two is placed in trace A.*

Description The MXMH command compares the current amplitude value of each point on a trace to each corresponding value detected during the previous sweep, then retains the maximum values.

See Also TRPST

NORM

Controls trace normalization for stimulus-response measurements.
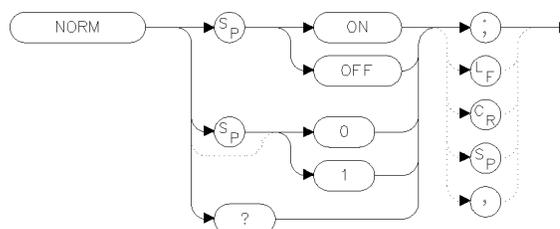
xnorm

Example   `100   OUTPUT 723;"NORM ON;"`

Description The NORM command turns on trace normalization. When trace normalization is turned on, its operation depends on whether a logarithmic or linear display is selected. With logarithmic displays, AMB ON is executed, and the reference level is set to 0 dB. With linear displays, the reference level offset and display line are set to place the current reference trace at the top of the screen and set it equal to a value of 1.000X. Next, the ADBTL trace math function is turned on. This normalizes the current trace to 1.000X.

Turning off trace normalization simply disables the trace math enabled when NORM ON was executed. Note that turning off normalization in linear mode does not reset ROFFSET or DL to their previous values.

Query Response The query response value depends on the state of the NORM and MEASURE commands as shown in the following table:

NORM Query Response

MEASURE State	Query Response NORM OFF	NORM ON
SA	0	0
SR	0	1
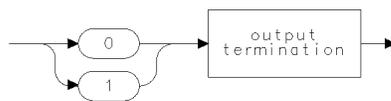
qgen0

See Also MEASURE, SRINPUT

NSTATE

Set number of state registers.
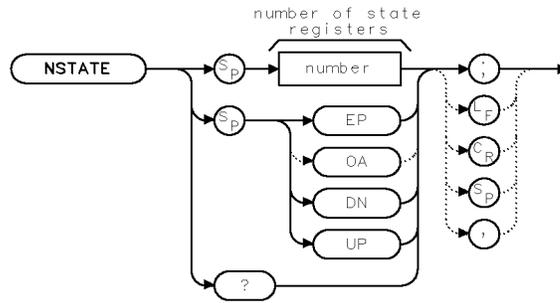
XNSTAT

Item	Description
Number Range	0 to value limited by available memory, or maximum of 20.

Example

```
10   OUTPUT 723;"PSTATE OFF;"
20   OUTPUT 723;"NSTATE 0;"
```
Deactivate state-register protection.
Specify 0 registers, erasing contents of all registers.

Description The NSTATE command allocates user memory for the storage of state-register files.

When the number of state-register files is decreased, the higher numbered files are deleted and their contents are lost. Since state-register files occupy large amounts of user memory, conserve memory space by limiting their number. Use the PSTATE command to protect state registers against accidental disposal with the NSTATE, STOR, or SAVES command. PSTATE does not prevent erasure with the ERASE or PURGE command.

The number of state-register files is not affected by instrument preset (IP) or power loss. However, the ERASE command sets the number of files to two after erasing the contents of internal memory. The last-state register, controlled by the POWERON command, does not form a state-register file, and thus is not affected by the NSTATE command. (The last-state register saves the state of the spectrum analyzer before power is removed, depending on the setting of the POWERON function.)

To select storage location (HP-IB device, HP-MSIB device, or internal memory), execute MSI before executing NSTATE. Internal memory is the storage device when power is applied to the spectrum analyzer.)

Query Response
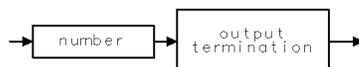
qgennum

See Also   POWERON, SAVES, PSTATE

ONEOS

Define list of functions executed at the end of the sweep.
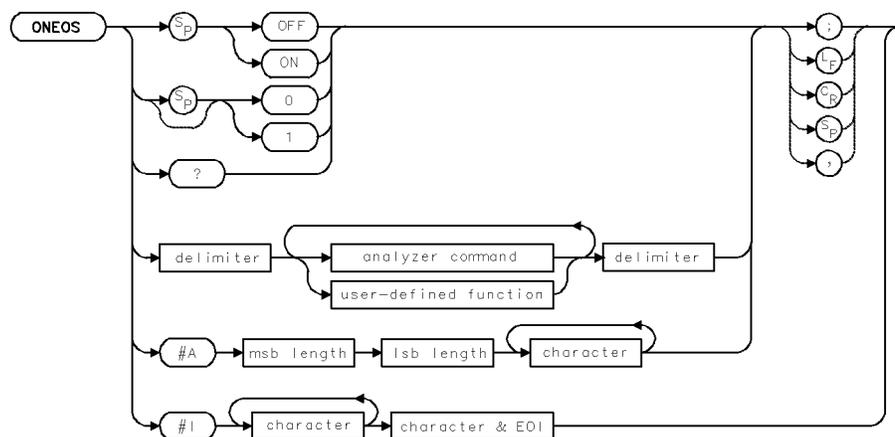
Item	Description
Delimiters	Mark the beginning and end of the algorithm.
Preset State	`ONEOS OFF`

Example Automatically measure a new wavelength range with each sweep.

```
140   OUTPUT 723;"VARDEF I,0;"
150   OUTPUT 723;"VARDEF W,700NM;"
160   OUTPUT 723;"ONEOS^";


170   OUTPUT 723;"MOV SP,200NM;";
180   OUTPUT 723;"IF I,LT,500;THEN;";

190   OUTPUT 723;"ADD I,I,100;ELSE;MOV I,100;ENDIF;";
210   OUTPUT 723;"ADD CENTERWL,W,I;";
220   OUTPUT 723;"^;"
```

Define user-defined variable equal to 0.
Define user-defined variable equal to 700 nm.
Begin definition of on-end-of-sweep algorithm. The following algorithm measures a higher harmonic of the calibration wavelength with each sweep.
Set span to 200 nm.
If I is less than 500, set I equal to I + 100. Otherwise, set I equal to 100.

Set the center wavelength equal to F × I.
End on-end-of-sweep algorithm.

Description The ONEOS command executes the algorithm in the data field when the sweep, trace processing, and other internal end-of-sweep functions are completed. The ONEOS algorithm is user-defined and consists of other analyzer commands, much like the user-defined functions created with the FUNCDEF command. The algorithm is stored in internal memory. Each time the sweep is completed, the algorithm is executed. The end-of-sweep status bit in the status byte is not set until the on-end-of-sweep algorithm is completed. **DISPOSE ONEOS** and **FORMAT** deletes the on-end-of-sweep algorithm from internal memory. The command parameters are as follows:

- **ONEOS ON** or **ONEOS 1** activates the algorithm.

- **ONEOS OFF**, **ONEOS 0**, or **IP** deactivates the algorithm.
  (Executing CLEAR 723 also deactivates the algorithm.)

- **ONEOS?** queries the algorithm definition to the computer.

Query Response
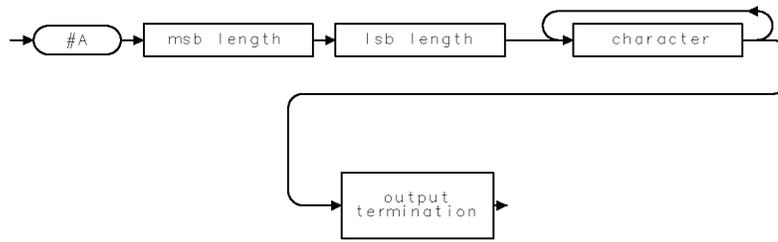
qgablock.

See Also PAUSE, DEBUG, ABORT, RETURN, DISPOSE, FORMAT, PROTECT and sweep-controlling commands, such as TS, SNGLS, CONTS, TM, VTL, VTH

ONMENU

Define list of functions executed when (MENU) is pressed.
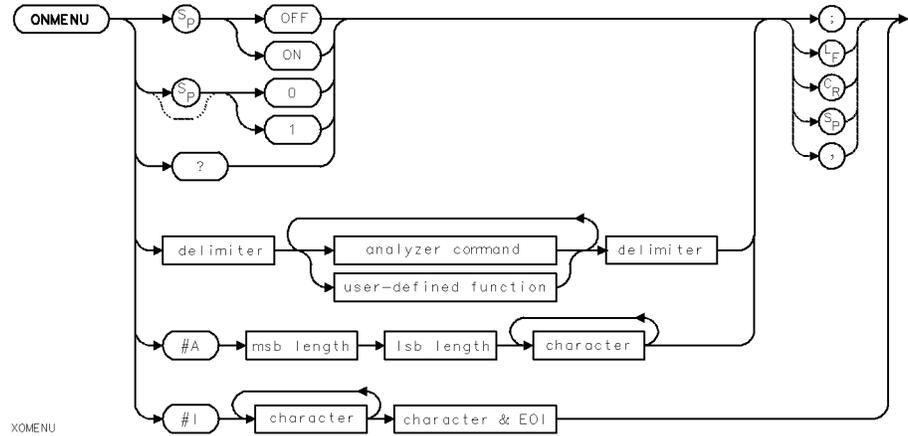
Item	Description
Delimiters	Mark the beginning and end of the algorithm.
Preset State	OFF

Example

```
10   OUTPUT 723;"ONMENU |;"                          Begin definition of on-menu-keypress algorithm.
20   OUTPUT 723;"TITLE &MENU key pressed.&;"          Set title.
30   OUTPUT 723;"|;"                                  End on-menu-keypress algorithm.
```

Description The ONMENU command executes the algorithm in the data field when
internal (MENU) key functions are completed. The ONMENU algorithm
is user-defined and consists of other analyzer commands, much like the
user-defined functions created with the FUNCDEF command. The algorithm
is stored in internal memory. `DISPOSE ONMENU` and `FORMAT` delete the
on-menu-keypress algorithm from internal memory. Command parameters
are as follows:

- `ONMENU ON` or `ONMENU 1` activates the algorithm.

- `ONMENU OFF`, `ONMENU O`, or `IP` deactivates the algorithm.
  (Executing CLEAR 723 also deactivates the algorithm.)

- `ONMENU?` queries the algorithm definition to the computer.

Query Response
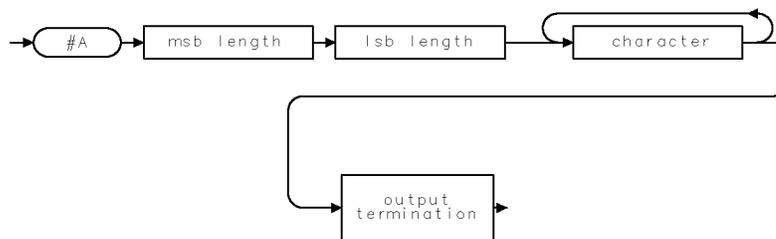
qgablock.

See Also ABORT, DEBUG, DISPOSE, FORMAT, PAUSE, PROTECT, RETURN

ONMKR

Define list of functions executed when sweep reaches pause marker.
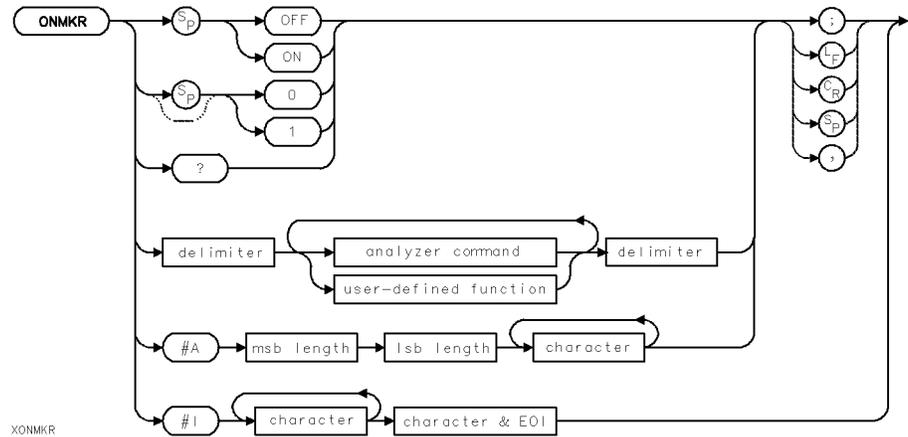
Item	Description
Delimiters	Mark the beginning and end of the algorithm.
Preset State	OFF

Example

```
10  OUTPUT 723;"ONMKR |;"
```
Begin definition of on-marker-pause algorithm.

```
20  OUTPUT 723;"TITLE &Reached marker.&;"
```
Set title.

```
30  OUTPUT 723;"|;"
```
End on-marker-pause algorithm.

Description The ONMKR command executes the algorithm in the data field when internal marker functions are completed. MKPAUSE must be set to a non-zero time. The ONMKR algorithm is user-defined and consists of other analyzer commands, much like the user-defined functions created with the FUNCDEF command. The algorithm is stored in internal memory. Each time the sweep reaches the marker, the algorithm is executed. **DISPOSE ONMKR** and **FORMAT** delete the on-marker-pause algorithm from internal memory. Command parameters are as follows:

- **ONMKR ON** or **ONMKR 1** activates the algorithm.

- **ONMKR OFF**, **ONMKR O**, or **IP** deactivates the algorithm. (Executing CLEAR 723 also deactivates the algorithm.)

- **ONMKR?** queries the algorithm definition to the computer.

Query Response
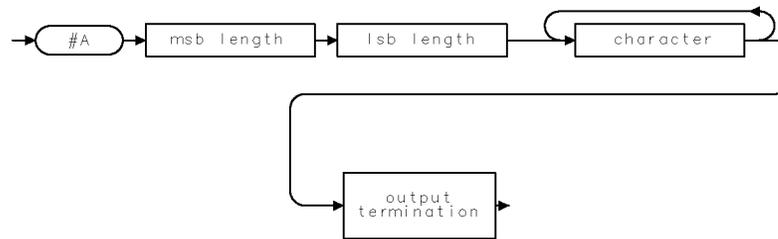
qgablock.

See Also ABORT, DEBUG, DISPOSE, FORMAT, MKPAUSE, PAUSE, PROTECT, RETURN

ONUSER

Define list of functions executed when (USER) is pressed.
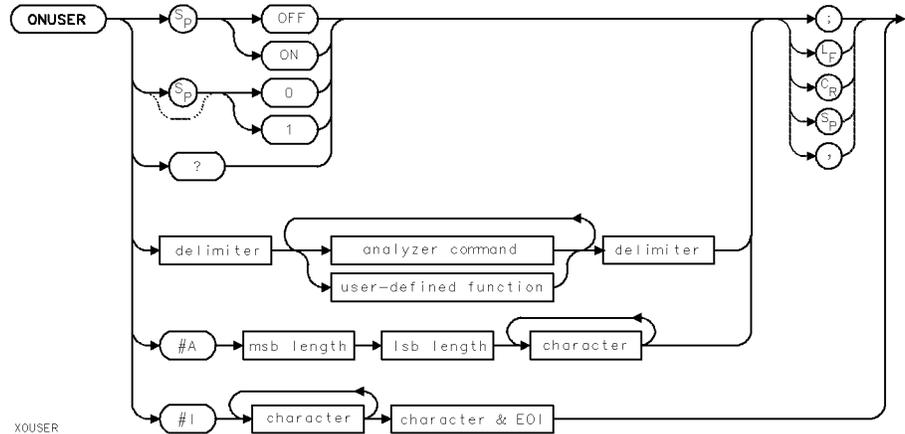
Item	Description
Delimiters	Mark the beginning and end of the algorithm.
Preset State	OFF

Example

10	OUTPUT 723;"ONUSER |;"	*Begin definition of on-user-keypress algorithm.*
20	OUTPUT 723;"TITLE &USER key pressed.&;"	*Set title.*
30	OUTPUT 723;"|;"	*End on-user-keypress algorithm.*

Description The ONUSER command executes the algorithm in the data field when internal
[USER] key functions are completed. The ONUSER algorithm is user-defined
and consists of other analyzer commands, much like the user-defined
functions created with the FUNCDEF command. The algorithm is stored
in internal memory. Each time the [USER] key is pressed, the algorithm
is executed. `DISPOSE ONUSER` and `FORMAT` delete the on-user-keypress
algorithm from internal memory. Command parameters are as follows:

- `ONUSER ON` or `ONUSER 1` activates the algorithm.

- `ONUSER OFF`, `ONUSER 0`, or `IP` deactivates the algorithm.
  (Executing CLEAR 723 also deactivates the algorithm.)

- `ONUSER?` queries the algorithm definition to the computer.

Query Response
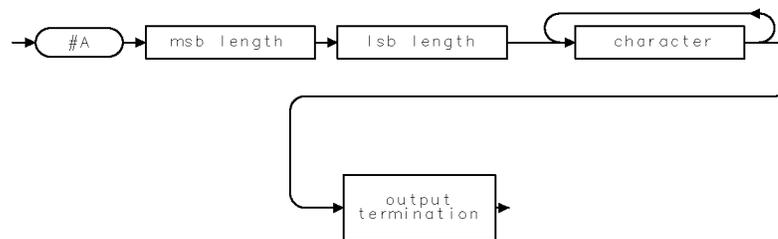
qgablock.

See Also ABORT, DEBUG, DISPOSE, FORMAT, PAUSE, PROTECT, RETURN

ONWINDOW

Define list of functions executed when display window is recreated.
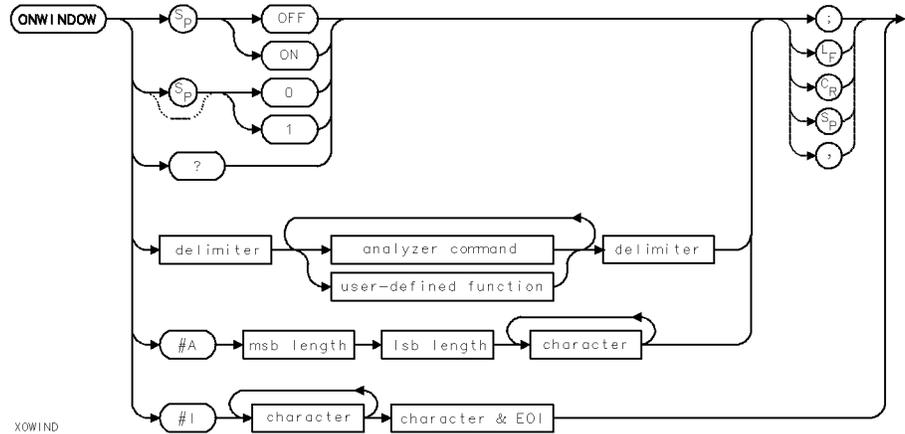
Item	Description
Delimiters	Mark the beginning and end of the algorithm.
Preset State	OFF

Example

```
10   OUTPUT 723;"ONWINDOW |;"            Begin definition of on-window-recreation algorithm.
20   OUTPUT 723;"TITLE &Redrawing window.&;"   Set title.
30   OUTPUT 723;"|;"                       End on-window-recreation algorithm.
```

Description The ONWINDOW command executes the algorithm in the data field when
the analyzer display needs to be redrawn. The ONWINDOW algorithm
is user-defined and consists of other analyzer commands, much like
the user-defined functions created with the FUNCDEF command. The
algorithm is stored in internal memory. Each time the display is redrawn,
the algorithm is executed. **DISPOSE ONWINDOW** and **FORMAT** delete the
on-window-recreation algorithm from internal memory. Command parameters
are as follows:

- **ONWINDOW ON** or **ONWINDOW 1** activates the algorithm.

- **ONWINDOW OFF**, **ONWINDOW O**, or **IP** deactivates the algorithm.
  (Executing CLEAR 723 also deactivates the algorithm.)

- **ONWINDOW?** queries the algorithm definition to the computer.

Query Response
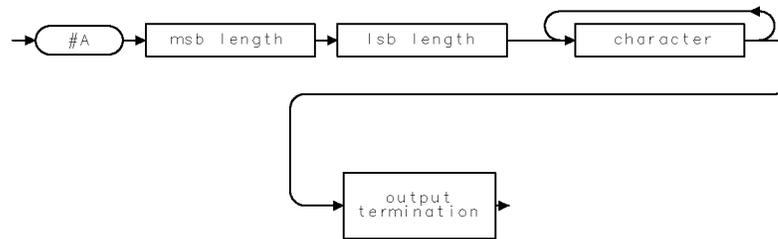
qgablock.

See Also ABORT, DEBUG, DISPOSE, FORMAT, PAUSE, PROTECT, RETURN

OP Return coordinates of instrument window.
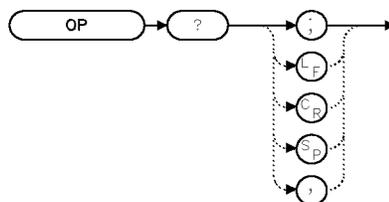
Example 110 OUTPUT 723;"OP?;" *Query coordinates of P1 and P2.*
120 ENTER 723;P1x,P1y,P2x,P2y *Assign values to computer variables.*

Description The OP command queries the values of P_1 and P_2 for the current instrument window scale.

For the default instrument window, the X and Y coordinates for P_1 and P_2 are (112,16) and (911,383). P_1 corresponds to the lower-left vertex. P_2 corresponds to the upper-right vertex. If more than one instrument window is displayed, OP gives the values P_1 and P_2 for all windows.

Query Response
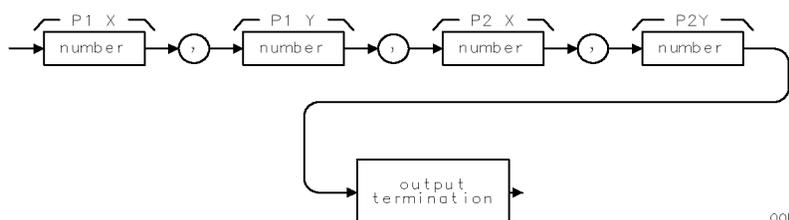
See Also IWINDOW, CLRDSP

OPTSW

Activate the front-panel MONOCHROMATOR OUTPUT and PHOTODETECTOR INPUT ports.
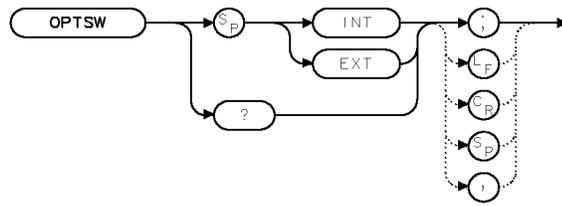
XOPTSW

Example   10   OUTPUT 723;"OPTSW INT;"

Description The OPTSW command selects the signal path on HP 71451B instruments (HP 70951B modules). The INT argument selects the normal signal path through the optical spectrum analyzer's monochromator to the photo diode for detection. The EXT argument makes the monochromator's output available at the front panel's MONOCHROMATOR OUTPUT connector. Also, the front panel's PHOTODETECTOR INPUT connector is activated for input.

The correction data is not changed when the switch is changed.

Query Response
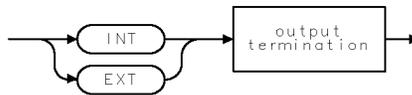
QOPTSW

See Also CORSEL, INSTMODE

OR   Set position of origin.
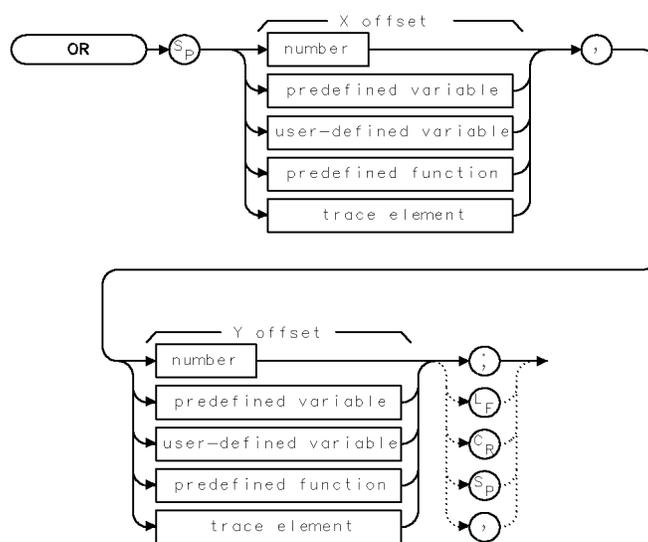
Item	Description
X-Offset Range	Determined by **SCALE** or **DWINDOW**.
Y-Offset Range	Determined by **SCALE** or **DWINDOW**.
Preset State	OR 0,0

Example Position a grid.

10	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
20	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory. Locate origin at 0,0 (lower-left corner). Set the units for the X and Y scale to their preset values, 0 to 1000.*
30	OUTPUT 723;"OR 100,200;"	*Offset the origin 100 units horizontally and 200 units vertically.*
40	OUTPUT 723;"PD;GRID 20,40,5,10;"	*Draw a grid. Its position is offset according to OR (line 180).*

Position a graphics window.

```
10   OUTPUT 723;"CLRDSP;"                 Clear graphics from memory.
20   OUTPUT 723;"OR 100,500;"            Offset the origin 100 units horizontally
                                          and 500 units vertically.
30   OUTPUT 723;"DWINDOW 100,500,TRA;"   Define a graphics window, scaled to trace-
                                          A parameters.
40   OUTPUT 723;"PD;GRAPH TRA;"           Draw trace A inside the window.
50   PAUSE                                Observe display, then press CONTINUE
                                          on computer keyboard.
60   OUTPUT 723;"DWINDOW OFF;"            Turn off the window.
```

Position a graphics item.

```
10   OUTPUT 723;"BLANK TRA;"      Blank trace A.
20   OUTPUT 723;"CLRDSP;"         Clear graphics from memory.
30   OUTPUT 723;"IT 1;VW OFF;"    Address item 1 and blank it.
40   OUTPUT 723;"TEXT$Item 1$;"   Define item 1 as text.
50   OUTPUT 723;"OR 700,200;"     Offset origin by 700 units horizontally and
                                  200 units vertically.
60   OUTPUT 723;"VW ON;"          Display item.
```

Description The OR command offsets the position of graphics on the display relative to P_1, in the current scale units set by the SCALE or DWINDOW command.

Use OR to position items created with the IT command or to position new coordinate systems (graphics window) created with the window command, DWINDOW.

See Also SCALE, DWINDOW, IWINDOW, IT, PA, PR

OUTPUT   Control HP-IB with the analyzer as the controller.
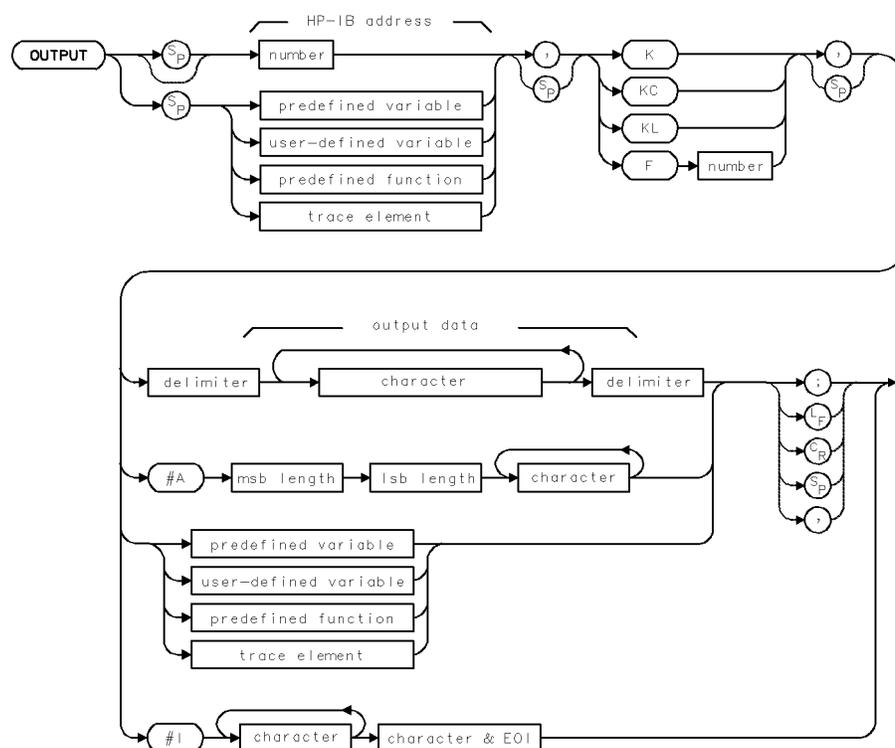
Item	Description
Address Range	Integers, 0 – 30

Example Use the analyzer as a controller on HP-IB. In the example, the controlled HP-IB device is an HP 3478 multimeter at address 25. Press the user-defined key, VDM, to communicate with the multimeter.

150	REMOTE 7	*Place all devices in remote status.*
160	CLEAR 723	*Clear the analyzer HP-IB interface buffer.*
170	OUTPUT 723;"IP;"	*Preset analyzer.*
180	OUTPUT 723;"VARDEF NN,0;"	*Assign 0 to user-defined variable, NN.*

190	OUTPUT 723;"FUNCDEF VDM,^";	*Begin definition of user-defined function, called VDM.*
200	OUTPUT 723;"OUTPUT 25,K,$H5$;";	*Send message, "H5" to multimeter.*
210	OUTPUT 723;"ENTER 25,K,NN;";	*Receive value in amperes from multimeter. Store value in NN.*
220	OUTPUT 723;"CLRDSP;";	*Clear graphics from memory.*
230	OUTPUT 723;"PA 400,500;";	*Set pen at (400,500).*
240	OUTPUT 723;"MPY NN,NN,1000;";	*Convert value to milliamperes.*
250	OUTPUT 723;"DSPLY NN,7,4;";	*Display value on analyzer screen.*
260	OUTPUT 723;"RELHPIB;";	*Release control of HP-IB.*
270	OUTPUT 723;"^;"	*End definition of user-defined function.*
280	OUTPUT 723;"KEYDEF 7,VDM,VDM;"	*Assign function to user-defined key, called* VDM.
290	SEND 7;UNL UNT DATA	*Clear HP-IB of all talkers and listeners.*
300	LOCAL 723	*Place analyzer in local control. Press* ⌐USER⌐, *then press the user-defined key,* VDM, *to communicate with the multimeter.*

Description The OUTPUT command establishes the analyzer as a controller on HP-IB. Use OUTPUT to send data or instructions to an HP-IB device.

Since HP-IB allows one controller only on HP-IB, OUTPUT must be synchronized with computer operation or else must be incorporated into user-defined functions that are executed with keys when a computer is not connected to HP-IB. (Press ⌐LCL⌐ for manual operation.) If another controller is detected on HP-IB, the OUTPUT function is aborted. (See "Controlling Instruments Over HP-IB" in Chapter 6.)

Execute `RELHPIB` to discontinue analyzer control of HP-IB. *HP-IB Address* specifies the address of the device. *Format* formats the data as follows:

K sends characters (ASCII) with no terminator.

KC sends characters (ASCII), followed by carriage-return/line-feed (ASCII codes 13 and 10).

KL sends characters (ASCII), followed by line-feed (ASCII code 10) and END (end-or-identify).

F sends data as ASCII-decimal numbers. In the syntax diagrams, the integer part of the number specifies the field width. The decimal part specifies the number of decimal places.

See Also ENTER, RELHPIB, ACTDEF, FUNCDEF

OVRW   Enable clear-write mode, but skip the initial clear.

```
        ┌─────────┐    ┌──┐        ┌────────┐    ┌───┐
        │  OVRW   ├─►(S)├────────►│  TRA   ├──┬─►│ ; │──────────►
        └─────────┘   (P)    │    └────────┘  │  └───┘
                             │    ┌────────┐  │   ┌─┐
                             ├──►│  TRB   ├──┤  ►(L)
                             │    └────────┘  │   (F)
                             │    ┌────────┐  │   ┌─┐
                             └──►│  TRC   ├──┘  ►(C)
                                  └────────┘       (R)
                                                  ┌─┐
                                               ►(S)
                                                  (P)
                                                  ┌─┐
                                               ►( , )
```
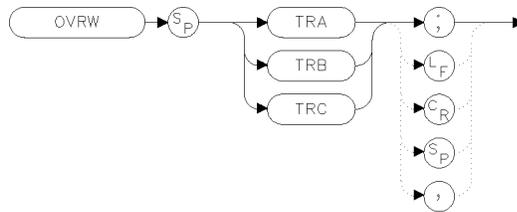
                                                                    x o v r w

Example   100   OUTPUT 723;"OVRW TRA;"   *Set this mode for trace B.*

Description The action of OVRW is identical to CLRW except that OVRW does not initially clear the internal representation of the trace (the display is still cleared).

PA

Move pen to absolute coordinate.
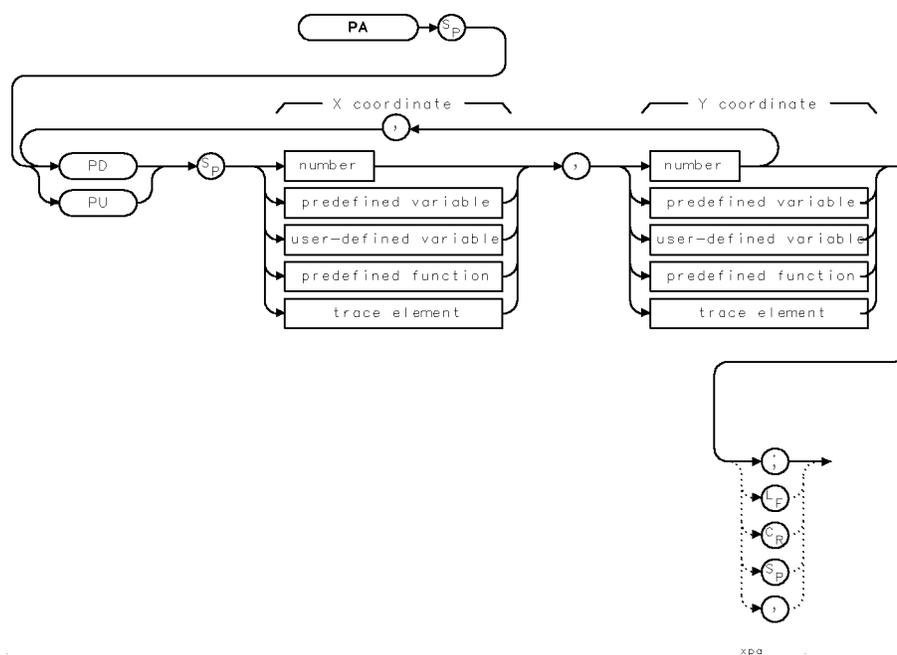
Item	Description
X-Coordinate Range	Determined by **SCALE** or **DWINDOW**
Y-Coordinate Range	Determined by **SCALE** or **DWINDOW**

Example

140	OUTPUT 723;"PA PU 0,0;"	*Move the pen in the up position to (0,0).*
150	OUTPUT 723;"PA PD 1000,1000;"	*Move the pen in the down position to (1000,1000), drawing a diagonal line.*
160	OUTPUT 723;"PU;PA 0,1000;"	*Place pen up and move it to (0,1000). (Notice that pen-up can be specified using PU as a separate command or as part of the PA syntax.)*
170	OUTPUT 723;"PD;PA 1000,0;"	*Place pen down and move it to (1000,0), drawing a line.*

Description The PA command moves the graphics pen from its current position to the position specified. A line is drawn on the analyzer display if the pen is down. Use the LINET command to vary the type of line drawn. Positions are specified as coordinates in the current units, which are set by the SCALE command, or by the DWINDOW command when a graphics window is on.

See Also CLRDSP, LINET, OR, PEN, SCALE, DWINDOW, IT, PU, PD, GRID, GRAPH, DSPLY, MK

PAUSE   Stop program operation and activate debug mode.
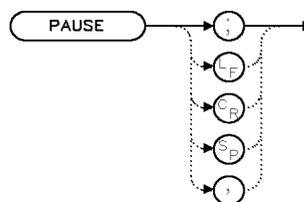
XPAUSE

Example   150   OUTPUT 723;"CENTERWL 750NM;"   *Set center wavelength.*
          160   OUTPUT 723;"PAUSE;"   *Press* CONT *key on spectrum analyzer*
                                                 to resume program operation.
          170   OUTPUT 723;"SP 100NM;"   *Set span.*

Description   The PAUSE command stops program operation until one of the following
              occurs:

              ● The CONT softkey is pressed.

              ● The INSTR PRESET or I-P key is pressed.

              ● DEBUG OFF is executed.

              ● The optical spectrum-analyzer input buffer is cleared via HP-IB. (For
                example, HP BASIC command CLEAR 723 clears the buffer when the
                optical spectrum analyzer is at HP-IB address 23, interface 7.)

PD

Place the graphics pen down.
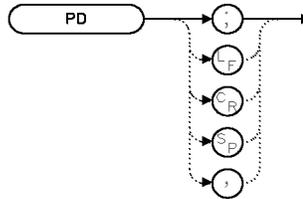
Item	Description
Preset State	Deactivated, pen up

Example

```
160  OUTPUT 723;"CLRDSP;"          Clear graphics from memory.
170  OUTPUT 723;"BLANK TRA;"       Blank trace A.
180  OUTPUT 723;"ANNOT OFF;"       Blank annotation.
190  OUTPUT 723;"PU;PA 400,400;"   Pen up. Move pen to (400,400).
200  OUTPUT 723;"PD;PA 1000,0;"    Place pen down. Draw line to (1000,0).
```

Description The PD command places the graphics pen in the down position enabling it to draw on the analyzer display whenever its position is changed. The pen must be placed down to draw all objects except for those referenced with item command, IT, or for characters written with the TITLE command. Place the pen down to draw the following objects. (Associated commands are in parenthesis.)

- Lines (PA,PR)
- Grids (GRID)
- Graphics marker (MK)
- Graphs (GRAPH)
- Variables (DSPLY)

See Also CLRDSP, MK, GRAPH, GRID, PA, PR, DSPLY

PDA

Analyze distribution of detected signals according to amplitude.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC,* or *user-defined trace.*
	–32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC,* or *user-defined trace.*
Resolution Range	Limited by source amplitude.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC.* Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	3 – 2048 position units.

Example   Perform amplitude distribution function on trace A.

90	OUTPUT 723;"SNGLS;TS;"	*Activate single-sweep mode. Clear-write trace A.*
100	OPTION BASE 1	*Specify lower boundary of arrays to be 1.*
110	OUTPUT 723;"TRDEF NN,20;"	*Define user-defined trace of 20 elements, NN.*
120	OUTPUT 723;"MOV NN,0;"	*Store 0 in all elements of trace array.*
130	OUTPUT 723;"PDA NN,TRA,5;"	*Perform amplitude distribution function on trace A.*
140	DIM A(20)	*Dimension 20-element computer array.*
150	OUTPUT 723;"NN?;"	*Query values stored in NN.*
160	ENTER 723;A(*)	*Store values in computer array.*
170	FOR I=1 TO 20	*Print values on computer display.*
180	PRINT A(I)	
190	NEXT I	

Description   This command replaces the destination trace with the amplitude distribution function of the source trace. The data in the source trace is processed point by point. The value of the bottom of the display is subtracted from each point, then the difference is divided by the resolution value which has been rounded to the nearest integer. If the result falls within the range of the elements of the destination trace, the content of the corresponding destination trace element is increased by one. An error results if there is not enough memory to execute PDA.

For example, assume that the source trace is displayed on a logarithmic scale of 10 dB per division, which produces an amplitude range of 100 dB. Choosing a resolution parameter of 5 divides the displayed amplitude range of trace A (100 dB) into twenty 5-dB increments numbered 1 through 20, with increment 1 beginning at the bottom graticule line and increment 20 ending at the reference level.

A resolution parameter of 5 yields a function that evaluates the amplitude of each point of trace A as follows. A trace-A point that is within 5 dB of the reference level adds 1 to the 20th point of the destination trace. A trace-A point that is between 5 to 10 dB below the reference level adds 1 to the 19th point in the destination trace, and so forth. Thus, if the amplitude of the first point is 12 dB below the reference level, the amplitude of the eighteenth point of the destination trace increases by 1. Likewise, if the amplitude of the second point of the source trace is 17 dB below the reference level, the seventeenth point of the destination trace increases by 1. Note that the destination trace must be long enough, in this case, 20 points long.

If the source trace is displayed on a linear scale, a resolution parameter of 5 divides the amplitude range into 20 increments of 500 measurement units each (5% of 10,000 is 500).

The PDA function assumes the amplitude range of the source trace represents displayed trace data. If the source trace is displayed on a logarithmic scale, the amplitude range is 100 dB. If the source trace is displayed on a linear scale, the amplitude range is 10,000 measurement units. When executing the probability function on source traces other than trace A, B, or C, assume the amplitude range of the source trace is in measurement units when selecting the resolution parameter. For example, if the amplitude range is 0 to 200 and you require twenty increments (1-10, 11-20, 21-30,. . . .), divide 200 by 20, then express the result (10, in this case) as a percentage of 10,000:

$$resolution\ parameter = \frac{100\,(amplitude\ range)}{10,000\,(number\ of\ increments)}$$

$$0.1 = \frac{100\,(200)}{20\,(10,000)}$$

The above equation yields a resolution parameter of 0.1.

PDL_

Start the PDL downloadable program and loads the related commands.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---

                                                                xpdl

Example

```
100   OUTPUT 723;"PDL_;"                    Run PDL program.
110   !
120   OUTPUT 723;"PDL_SRC?;"                Return light source status.
130   ENTER N;
140   IF N=0 THEN
150   OUTPUT 723;"PDLSRC;"                  Turn light source on.
160   END IF
170   !
180   OUTPUT 723;"PDLINIT;"                 Initialize PDL measurement.
190   PRINT "Press RETURN to Continue";
200   PAUSE;
210   !
220   OUTPUT 723;"PDLCALC;"                 Perform PDL calculation and dis-
                                            play results.
230   !
240   OUTPUT 723;"PDLSCALE;"                Perform auto-scale.
250   !
260   PLOT;
270   !
280   OUTPUT 723;"PDLEXIT;"                 Terminate PDL program.
290   END;
```

Description   The PDL_ command runs the swept Polarization Dependent Loss
measurement procedure which requires the following steps:

1. Set the start and stop wavelengths.

2. Set the resolution bandwidth (RB) and perform an auto-scale, as needed.

3. Set the video bandwidth (VB) as needed.

See Also   PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?,
PDLSCALE, PDLSRC, PDL_SRC?

PDLCALC

Perform calculation and display polarization dependent loss.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---
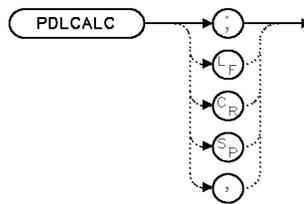
xpdlcalc.

Example   `220  OUTPUT 723;"PDLCALC;"`   *Perform PDL calculation and display results.*

Description The PDLCALC command calculates the peak-to-peak polarization dependent loss and displays the result. After the calculation is performed, the measurement is stopped and the result is displayed as trace A. Trace A is the result of trace B (maximum) minus trace C (minimum).

See Also PDL_, PDLINIT, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDLDEV

Toggle between the internal or external detector.

> **N O T E**
>
> Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.
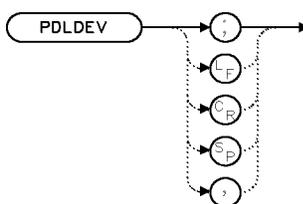
xpdldev

Item	Description
Preset State	external

Example   10 OUTPUT 723;"PDLDEV;"   *Toggles the receive device.*

Description The PDLDEV command toggles the receive device between either the internal or an external detector thus selecting either optical-to-optical testing or optical-to-electrical testing.

See Also PDL_, PDLINIT, PDLCALC, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDL_DEV?

Query the selected detector.

> **N O T E**
>
> Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.
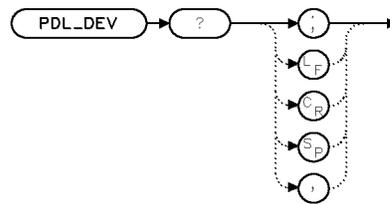
xpdldevq.

Example   10 OUTPUT 723;"PDL_DEV?;"   *Query the receive device.*

Description The PDL_DEV? query returns the current status value of the receive device to be either internal (0) or external detector (1).

Query Response
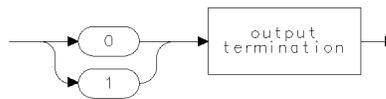
qgen0

See Also PDL_, PDLINIT, PDLCALC, PDLDEV, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDLEXIT   Terminate PDL measurement procedure.

> **N O T E**
>
> Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.
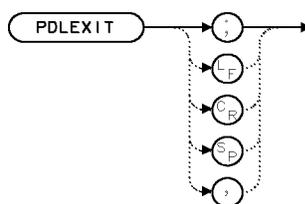
xpdlex

Example   280   OUTPUT 723;"PDLEXIT;"   *Terminate PDL program.*

Description   The PDLEXIT command terminates the polarization dependent loss (PDL) measurement program.

See Also   PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDLINIT   Initialize PDL measurement.

NOTE

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---
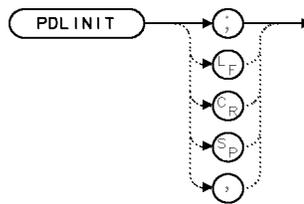
xpdlin

Example
```
180   OUTPUT 723;"PDLINIT;"          Initialize PDL measurement.
190   PRINT "Press RETURN to Continue";
200   PAUSE;
```

Description   The PDLINIT command initializes the polarization dependent loss (PDL) measurement procedure. The traces are initialized to measure PDL, continuous sweep mode is activated, and the results are displayed as follows:

- Trace A contains the detected signal.
- Trace B contains the maximum signal.
- Trace C contains the minimum signal.

See Also   PDL_, PDLCALC, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDLREV　　Display PDL revision number.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---
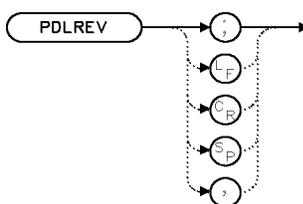
xpdlrev .

Example　　10　OUTPUT 723;"PDLREV;"　　*Display the PDL revision number.*

Description　The PDLREV command displays the polarization dependent loss (PDL) measurement program revision number.

See Also　PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?,PDLEXIT, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDL_REV?

Return PDL revision number.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---
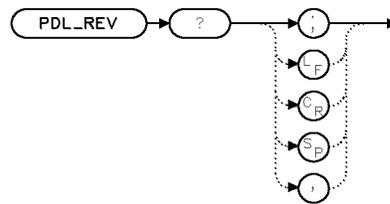
xpdlrevq.

Example   10 OUTPUT 723;"PDL_REV?;"   *Return the PDL revision number.*

Description The PDL_REV? query returns the polarization dependent loss (PDL) measurement program revision number.

Query Response
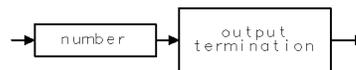
qgennum .

See Also PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLREV, PDL_REV?, PDLSCALE, PDLSRC, PDL_SRC?

PDLSCALE   Auto-scale displayed signal during PDL measurement.

> **N O T E**
>
> Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.
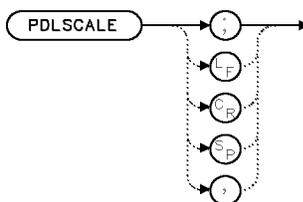
xpdlscal.

Example   240   OUTPUT 723;"PDLSCALE;"   *Perform auto-scale.*

Description The PDLSCALE command automatically scales the displayed signal during a polarization dependent loss (PDL) measurement.

Prior to initializing the PDL measurement, the PDLSCALE command positions the peak of trace A to the reference level and selects the scaling (that is, dB/division) based upon the minimum signal level of trace A.

After the PDLINIT command is executed, the PDLSCALE command positions the peak of trace B (that is, the maximum-hold signal) to the reference level and selects the scaling based upon the minimum point in trace C (that is, the minimum-hold signal).

After the PDLCALC command is used, the PDLSCALE command positions the peak of trace A as a result of trace B minus trace C by selecting the maximum dB/division which keeps the maximum signal below the top graticule.

See Also PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSRC, PDL_SRC?

PDLSRC

Toggle the light source on and off.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit. This also requires Option 002 built-in white light source.

---
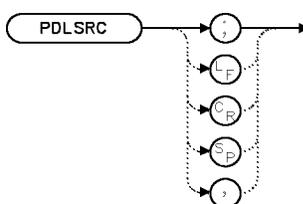
xpdlsrc

Item	Description
Preset State	on

Example

```
120   OUTPUT 723;"PDL_SRC?;"        Return light source status.
130   ENTER N;
140   IF N=0 THEN
150   OUTPUT 723;"PDLSRC;"          Turn light source on, if light source is off.
160   END IF
```

Description The PDLSRC command toggles the internal white light source on and off.
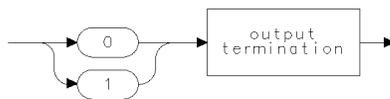
Query Response

qgen0

See Also PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDL_SRC?

PDL_SRC?   Return light source status.

N O T E

Only for use with HP 71451B Option 003, Swept Polarization Dependent Loss (PDL) Kit.

---
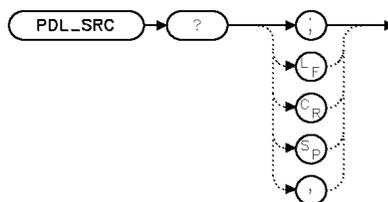
xpdlsrcq.

Example
```
120   OUTPUT 723;"PDL_SRC?;"      Return light source status.
130   ENTER N;
140   IF N=0 THEN
150   OUTPUT 723;"PDLSRC;"        Turn light source on, if light source is off.
160   END IF
```

Description The PDL_SRC? query returns the light source status, where 0 indicates off and 1 indicates on.

Query Response
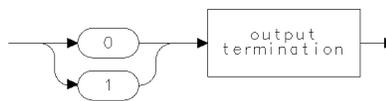
qgen0

See Also PDL_, PDLINIT, PDLCALC, PDLDEV, PDL_DEV?, PDLEXIT, PDLREV, PDL_REV?, PDLSCALE, PDLSRC

PDMEAS

Measure the flatness of a photodiode under test.

N O T E

This command is only available for systems using the HP 70951A module.

Before the command is executed, perform the following steps:

1. Connect the front-panel `MONOCHROMATOR OUTPUT` to the `PHOTODETECTOR INPUT` with a fiber-optic cable.

2. Press `STORE THRU -> B`.

3. Connect the fiber-optic cable to the photodiode under test which is connected to TRANS–Z IN on the rear panel.

XPDMS
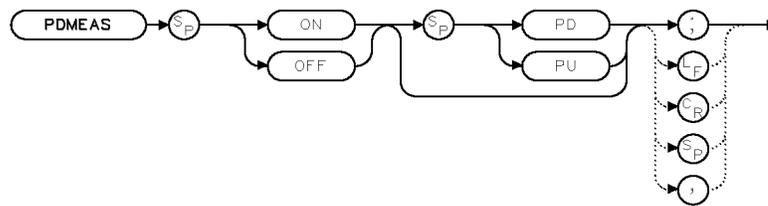
Item	Description
Requirement	HP 70951A module
Preset State	OFF

Example
```
100   CLEAR 723                          Initialize analyzer.
110   OUTPUT 723;"IP;"
120   PRINT "STORE REFERENCE TRACE"       Initiate photodiode flatness measurement.
130   OUTPUT 723;"STORREF THRU;"
140   PRINT "HIT RETURN TO CONTINUE"
150   PAUSE
160   OUTPUT 723;"PDMEAS ON;"
```

Description The PDMEAS command is used to measure the flatness of an external photodiode. The diode-flatness correction factors are not included in the flatness correction.

When the PDMEAS ON parameter is used, the following conditions are satisfied:

- Trace A − B is turned on.

- Internal, optical switch is set to external.

- Diode flatness correction factors are turned off.

- External diode is selected.

- IN and OUT indicator LEDs are turned on.

- DET IN indicator LED is turned off.

When the PDMEAS OFF parameter is USED, the following conditions are satisfied:

- Trace A − B is turned off.

- Internal, optical switch is set to external.

- Diode flatness correction factors are turned on.

- Internal diode is selected.

- IN and OUT indicator LEDs are turned on.

- DET IN indicator LED is turned on.

See Also INSTMODE, STORREF

PDWL

Analyze distribution of detected signals according to wavelength.

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*. −32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
Source-Wavelength Range	Hardware dependent
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	3 – 2048 position units.

Example	100	CLEAR 723		*Initialize analyzer.*
	110	OUTPUT 723;"IP;"		
	120	OPTION BASE 1		*Specify lower boundary of computer arrays to be 1.*
	130	DIM A(20)		*Dimension 20-element computer array.*
	140	ASSIGN @Sa TO 723		*Assign address 723 to @Sa.*
	150	OUTPUT @Sa;"TRDEF B_IN,20;MOV B_IN,0;"		*Define 20-element user-defined trace, B_IN.*
	160	OUTPUT @Sa;"TRDEF P_WL,20;MOV P_WL,0;"		*Define 20-element user-defined trace, P_WL. Store zero in all points of P_WL.*
	170	OUTPUT @Sa;"CENTERWL 750NM;SP 100NM;"		*Set measurement range.*
	180	OUTPUT @Sa;"SNGLS;TH -60DBM;TS;"		*Activate single-sweep mode. Set threshold to −60 dBm. Sweep trace A once.*
	190	OUTPUT @Sa;"COMPRESS B_IN,TRA,POS;"		*Compress contents of trace A and store results in B_IN.*
	200	OUTPUT @Sa;"PDWL,P_WL,B_IN;"		*If the amplitude of a point in B_IN exceeds the threshold, store a "1" in the corresponding point in P_WL.*
	210	OUTPUT @Sa;"TDF M;"		*Format trace-output data in measurement units.*
	220	OUTPUT @Sa;"P_WL?;"		*Query values stored in P_WL to computer.*
	230	ENTER @Sa;A(*)		*Store values in computer array.*
	240	FOR I=1 TO 20		*Print values (0 or 1).*
	250	PRINT A(I)		
	260	NEXT I		

Description The PDWL command augments the destination trace with the wavelength-distribution function of the source trace. Elements in the source trace that are above the threshold value cause corresponding elements in the destination trace to be increased in amplitude by one measurement unit. Use the threshold command, TH, to change the threshold value. Otherwise, its default value is nine major divisions below the reference level.

See Also TH

PEAKS

Sort detected signals from lowest wavelength to highest wavelength, or from lowest amplitude to highest amplitude.

XPEAKS

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	3 – 2048 position units.

Example

60	OUTPUT 723;"VARDEF WL1,0;"	*Define user-defined variable, WL1, equal to 0.*
70	OUTPUT 723;"CENTERWL 1000NM;SP 300NM;TS;"	*Set the measurement range. Sweep trace A.*
80	OUTPUT 723;"TH -65DBM;MKPX 10DB;TS;"	*Set the threshold to −65 dBm. Set marker-peak excursion to 10 dB. Sweep trace A.*
90	OUTPUT 723;"PEAKS TRB,TRA,WLN?;"	*Sort signals by wavelength and place results in trace B. Trace B now contains position information. Query number of peaks to computer.*
100	ENTER 723;A	*Assign number of peaks to computer variable, A.*
110	PRINT "SIGNALS ABOVE -65 DBM:";A	*Print number of peaks.*
120	OUTPUT 723;"MOV WL1,WLUNITS TRB[2],TRA;"	*Move wavelength of second element in trace B to WL1.*
130	OUTPUT 723;"WL1?;"	*Query value of WL1.*
140	ENTER 723;W	*Assign value to computer variable.*
150	PRINT "THE WAVELENGTH OF POINT 2 IS ";W	

Description   The PEAKS command sorts signal peaks by wavelength or amplitude, stores in the destination trace the horizontal position of each peak in position units, then computes the number of peaks found.

When sorting by wavelength, PEAKS first computes the horizontal position of all peaks. These positions are consecutively loaded into the destination trace, the lowest value occupying the first element. Thus, the destination trace from left to right corresponds to signal wavelengths from low to high.

When sorting by amplitude, PEAKS first computes the amplitudes of all peaks in the source trace in measurement units, and sorts these values from high to low. The horizontal positions of the peaks are then loaded into the destination trace, with the horizontal position of the highest value occupying the first element. Thus, the destination trace amplitude, from left to right, is proportional to signal amplitudes from high to low.

The destination trace values range from 1 to the length of the source trace. If necessary, the last sorted value is repeated to fill the remaining elements of the destination trace.

PEAKS uses the current value set by the marker-peak-excursion command, MKPX, as the criterion for determining peaks.

PEAKS must be used as either a query or as a source in another
analyzer-command function. Form a query by ending the PEAKS statement
with a question mark (?). When used as a query, PEAKS queries the number
of peaks found.

Use PEAKS as a source by incorporating the PEAKS statement into any
spectrum-analyzer command having *predefined function* in its syntax
diagram. When PEAKS is used as a source, the number of peaks found is
used for operation by the command that contains PEAKS.

Predefined-
Function
Syntax
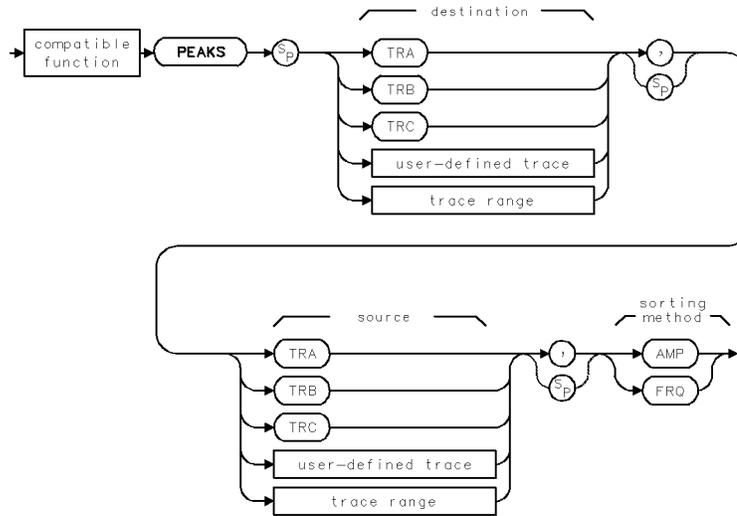
Query Syntax
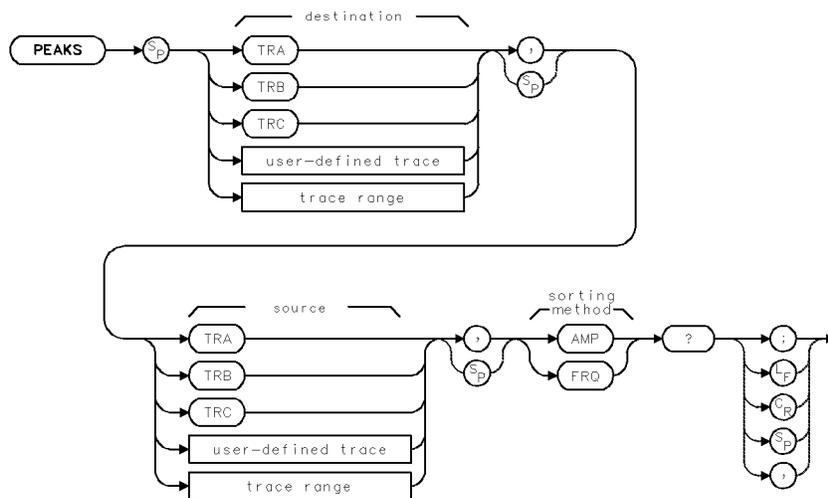
Query Response
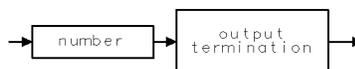
qgennum .

See Also MKPX

PEN

Choose pen for plotting on plotter.

XPEN
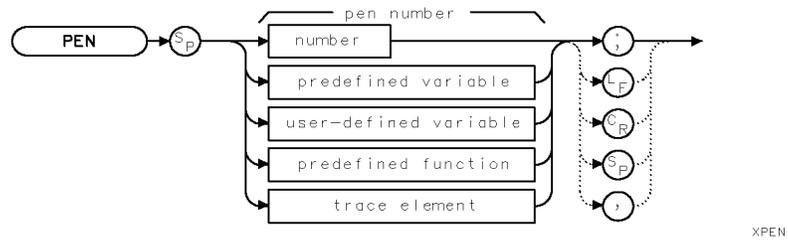
Item	Description
Pen-Number Range	0 – 255, depending on display model
Preset State	PEN 0

Example

150	OUTPUT 723;"CLRDSP;"	*Clear user graphics.*
160	OUTPUT 723;"TRDSP TRA,OFF;"	*Remove trace A from display.*
170	OUTPUT 723;"GRAT OFF;"	*Turn off graticule.*
180	OUTPUT 723;"ANNOT OFF;"	*Turn off annotation.*
190	OUTPUT 723;"IT 1;VW OFF;"	*Define item 1 while it is off (blanked).*
200	OUTPUT 723;"TEXT$Item 1$;"	*Assign text to item 1.*
210	OUTPUT 723;"IT 1;OR 300,700;"	*Move item 1 to (300,700).*
220	OUTPUT 723;"PEN 1;"	*Assign pen color to item 1.*
230	OUTPUT 723;"VW ON;"	*Display item 1.*

Description The PEN command specifies the color of objects plotted or displayed on devices having color capability.

Default Color Assignments for HP 70004A Display

Pen Number	Default Color[1]	Display Assignment
1	gray	GRATICULE
2	yellow	TRACE A
3	light blue	TRACE B
4	light red	TRACE C
5	green	LINES
6	bright white	TITLES
7	gray	ANNOTATION
8	orange	ADVISE text
9	red	ERRORS
10	bright white	MARKERS
11	white	softKEY LABELS
12	white	
13	red	
14	green	
15	blue	
16	black	BACKGROUND

[1] Color can be adjusted using (DISPLAY) menus.

See Also CLRDSP, LINET

PERASE

Partially erase internal memory and analyzer settings.
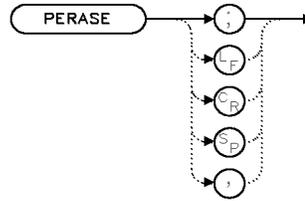
XPERAS

Description The PERASE command erases all unprotected memory that resides in the spectrum analyzer. PERASE erases **internal memory**, which contains user-defined variables, traces, functions, on-end-of-sweep algorithms, amplitude-correction factors (AMPCOR), and files. (Refer to the following tables for a complete listing.) PERASE also sets all instrument settings and their associated predefined variables to their preset values, which are all stored in reserved memory.

PERASE does not erase information stored in **external memory** located on HP-IB or HP-MSIB devices, such as a disk drive or a memory card. PERASE does not erase the serial number, the correction-factor values, (USER) keys, or limit-line table parameters from reserved memory.

Various portions of different memories can be erased or protected against erasure with the commands summarized in the following two tables.

Commands that Erase Reserved Memory

Memory Components	Commands that Erase or Reset Memory		
	PERASE	ERASE	IP
(USER) keys (KEYDEF)		X	
limit-line table (LIMILINE)		X	
serial numbers		X	
active trace data (trace A, B, or C)	X	X	X
instrument settings not stored in state registers	X	X	X

Commands that Erase or Protect User Memory

Memory Components	Commands that Erase or Reset Memory			Commands that Protect Memory	
	PURGE FORMAT	DISPOSE* PERASE* ERASE*	NSTATE	PROTECT†	PSTATE
user-defined variable	X	X		X	
user-defined trace	X	X		X	
user-defined function	X	X		X	
active user-defined function	X	X		X	
on-end-of-sweep-algorithm	X	X		X	
trace-amplitude correction factors	X	X		X	
state register file (instrument settings and title line)	X	X	X	X	X
program file	X	X		X	
trace file	X	X		X	
(USER) keys file	X	X		X	
Limit line file	X	X		X	
* Indicates command erases internal memory only.					
† Indicates command that protects internal memory only.					

PERSIST

Set the number of previous traces to be simultaneously displayed onscreen.
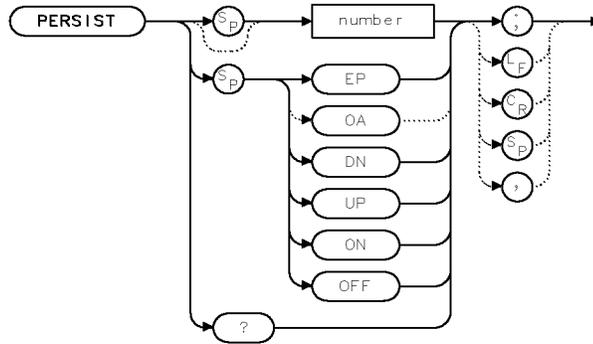
xpersis .

Item	Description
Preset State	OFF, 16 traces
Range	1 – 50

Description The PERSIST command sets the number of previous traces to be simultaneously displayed onscreen. Range may be limited by available display memory. To increase the number of traces that can be displayed, reduce the number of trace points in each trace.

Overlapping points are brightened. The color map of the display is changed to a monochrome green scale.

Query Response
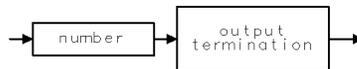
qgennum .

PLOT

Plot the display screen.
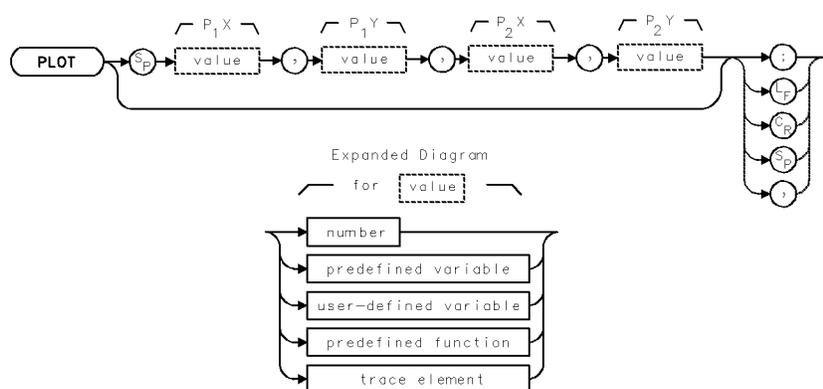
Item	Description
$P_1 X$, $P_1 Y$, $P_2 X$, and $P_2 Y$ Range	Depends on plotting device.

Example of plotting DLP contents.

```
plot 0,0,700,500; wait 180s;
```

The first line of the above example initiates a printer-generated copy of the screen currently displayed. The second line places a 3-minute wait on the DLP while the plot completes.

Example

The following example initiates a plotter-generated copy of the screen currently displayed. The plot occurs via external (remote) control.

```
10    Hpib=7                                    Dump DEVICE = HP-IB PLOTTER.
20    Dsplay=4
30    Dsp=Hpib*100+Dsplay
40    Sa=Hpib*100+23
50    Plt=5
60    !
70    OUTPUT Dsp;"PI 1,0,"&VAL$(Plt)&",0;"
80    !                                         HP-IB address is 5.
90    Dummy=SPOLL(Dsp)                          Clear status register.
100   !                                         Allow all errors, SRQ when done.
110   OUTPUT Dsp;"IM 255,16,0;"
```

```
120    ON INTR Hpib GOTO Done              When finished, GO TO ...
130    ENABLE INTR Hpib;2                  Interrupt when finished.
140    !                                   Dump plot.
150    OUTPUT Dsp;"CY 1;";
160    !                                   No CR/LF.
170    SEND Hpib;UNL LISTEN Plt TALK Dsplay DATA   Get controller off HP-IB.
180    !                                   Do not use "LOCAL 7".
190 Loop:GOTO Loop                         Wait for completion.
200    !
210 Done:  !
220    OFF INTR
230    Status_byte=SPOLL(Dsp)              Read display status.
240    IF BIT(Status_byte,4) THEN          Bit 4 is command done.
250      DISP "DONE!"
260    ELSE
270      DISP "UNEXPECTED INTERRUPT"
280    END IF
290    !
300    END
```

Description The PLOT command generates a copy of the spectrum-analyzer display on paper with a plotter or printer.

The plotter or printer must be properly configured prior to issuing PLOT. P_1X and P_1Y , and P_2X and P_2Y specify the lower-left and upper-right corners of the plot, respectively. If the P_1X, P_1Y, P_2X , and P_2Y parameters are omitted, the parameters stored in the display module are used.

The instrument addressed with the PLOT command also must be assigned to the display keyboard. The PLOT command instructs the display to do a graphics plot dump. Execution of the plot command is not complete until the plotter has finished plotting.

POSU

Convert wavelength or time units to position units.

xposuw

Item	Description
Source Amplitude Range	0 - 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 - +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm 1.8 \times 10^{308}$ when using *number, predefined variable*, or *user-defined variable*.
Source Length Range	3 - 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Reference-Trace-Amplitude Range	0 - 10,000 measurement units for linear trace data in *TRA, TRB, TRC*, or *user-defined trace*. −32,768 - +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC*, or *user-defined trace*.
Reference-Trace-Length Range	3 - 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example

160	OUTPUT 723;"CLRDSP;"	*Clear user graphics from memory.*
170	OUTPUT 723;"SCALE 0,800,-11003,502;"	*Define the units for scale. Set the minimum and maximum values for the X axis to 0 and 800, and the Y axis to −11003 and 502.*
180	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
190	OUTPUT 723;"PU;PA 0,MEASU -37DBM,TRA;"	*Lift up pen. Move pen to edge of the graticule at −37 dBm.*
200	OUTPUT 723;"PD;PA POSU 800NM,TRA,MEASU -62DBM,TRA;"	*Place pen down. Draw to the point at 800 nm and −62 dBm.*
210	OUTPUT 723;"PU;PA 0,MEASU -25DBM,TRA;"	*Place pen up. Move it to the left edge of the graticule.*
220	OUTPUT 723;"PD;PA 800,MEASU -30DBM,TRA;"	*Draw a line to the right edge of the graticule, ending at −30 dBm.*

Description   The POSU command converts wavelength or time values to position-unit values. Position units describe the position of a point along the horizontal axis of a trace. The point at the left end of the trace has a position-unit value of 1. The point at the right end of an 800-point trace has a position-unit value of 800. POSU must be used as either a query or as a source in another analyzer-command function. Form a query by ending the POSU statement with a question mark (?). When used as a query, POSU gives the position-unit value of the source. Use POSU as a source by incorporating the POSU statement into any spectrum-analyzer command containing

predefined function in its syntax diagram. When POSU is used as a source, the position-unit value of the source is used for operation by the command that contains POSU. POSU converts the wavelength of the reference-trace element to position units when the trace is displayed in wavelength spans greater than zero. For spans equal to zero, POSU converts the sweep-time position of the trace element to position units. If the reference trace is not specified, the last active trace is assumed to be the reference trace.

Predefined-
Function
Syntax
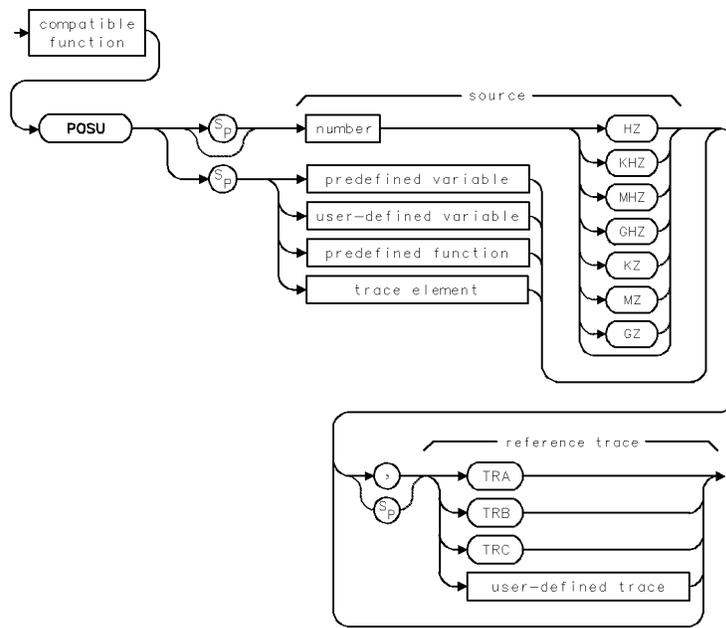
Query Syntax
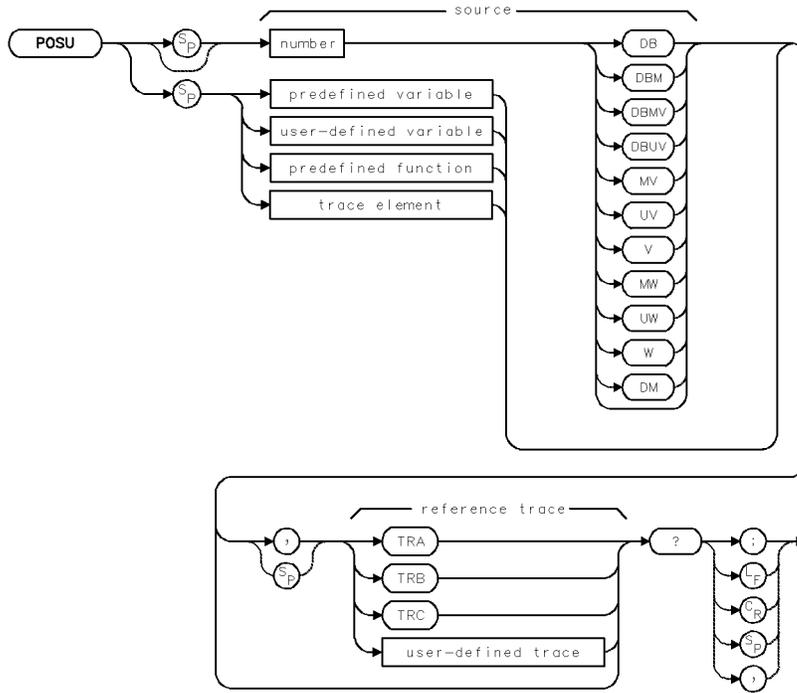
Query Response
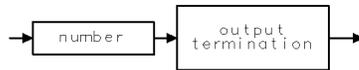
qgennum

POWERON

Specify the analyzer's state at power-up.
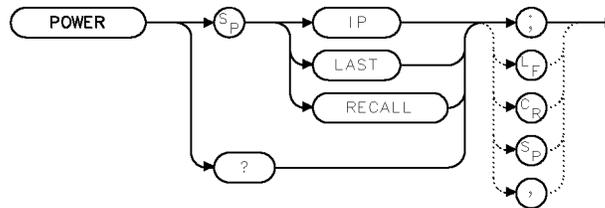
xpower

Example 120 OUTPUT 723;"POWERON LAST;"   *Recall at power-up the state the analyzer was in when power was removed.*

Description The POWERON command specifies the state the analyzer will be in when power is applied.

Argument	Definition
IP	Places the analyzer in its preset state when power is applied.
LAST	Recalls at power-up the state the analyzer was in when power was removed (the exception is that the mass-storage device is set to internal memory).
RECALL	Recalls state 0 at power-up. If state 0 has not been saved, and so does not exist, warning message 16006 is displayed when **POWERON RECALL** is executed.

Query Response
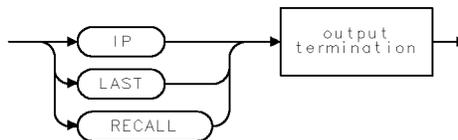
qpower

PR

Move pen relative to current position.
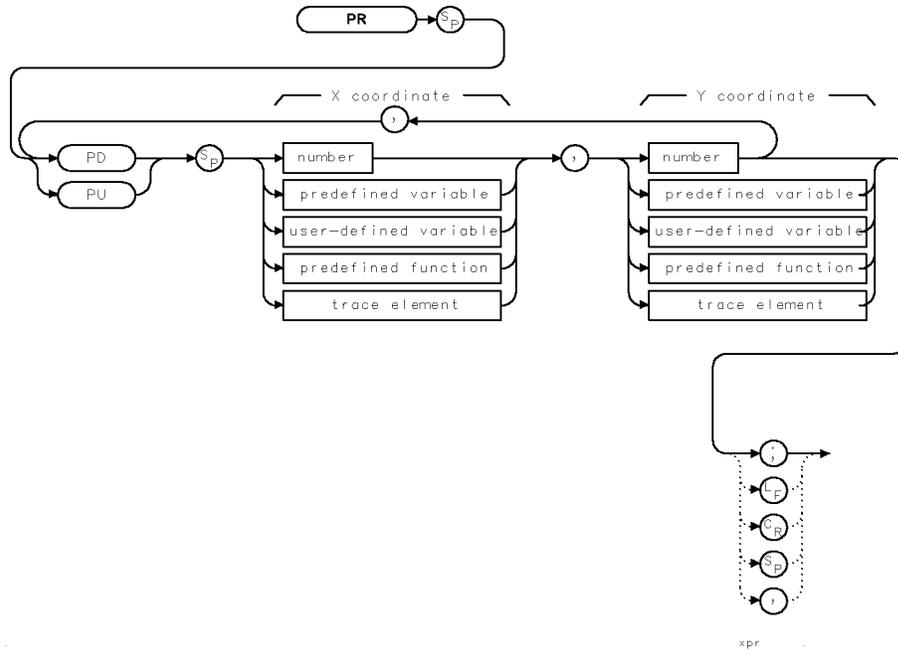
Item	Description
X-Coordinate Range	Determined by **SCALE** or **DWINDOW**
Y-Coordinate Range	Determined by **SCALE** or **DWINDOW**

Example Draw a line on the analyzer display.

110	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
120	OUTPUT 723;"PD;PR 800,600,-500,100;"	*Place pen down. Move pen 800 units to the right and 600 units up from its current position, then move pen 500 units to the left, and 100 units up.*

Description The PR command moves the graphics pen relative to its current position.

A line is drawn on the analyzer display if the pen is down. Positions are specified as coordinates in scale units, which are set by the SCALE command or by the DWINDOW command if a graphics window is on.

See Also LINET, OR, PEN, SCALE, DWINDOW, IT, PU, PD, GRID, MK, GRAPH

PREFX

Change the file prefix that identifies user-memory entries.

XPREFX
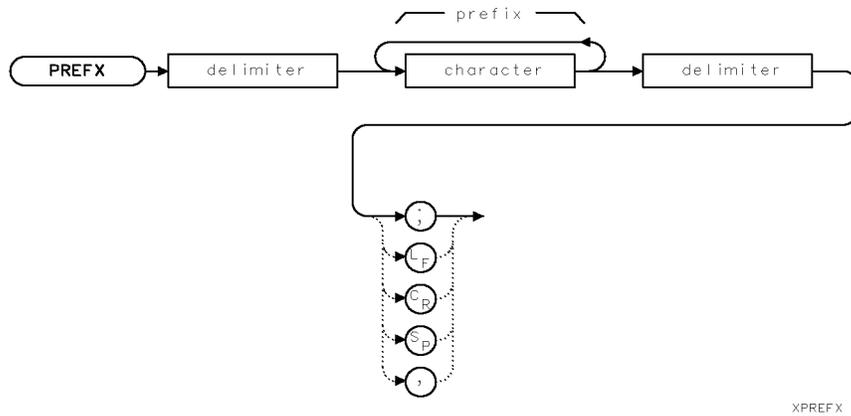
Item	Description
Prefix Range	0 – 8 characters
Requirements	If executed, it must precede store and save commands, such as:
	LIMISAVE, SAVED,
	SAVES, SAVET,
	SAVEU, STOR

Example

```
110  OUTPUT 723;"PREFX;"
```

Description The PREFX command changes the prefix letter appended to file numbers when the SAVED, SAVES, SAVET, and SAVEU commands (or corresponding keys) are used to store data in user memory. The following table shows the default prefixes.

User Memory Contents

Article Stored	Storing Command	File Prefix for Entry in User-Memory Catalog
file containing user-key menu	SAVEU	u_
file containing trace	SAVET	t_
file containing user-defined variable*	SAVED	d_
file containing user-defined function*	SAVED	d_
file containing on-end-of-sweep algorithm*	SAVED	d_
file containing limit line	LIMISAV	l_
file containing state register (instrument settings and title-line)	SAVES	s_
* Cannot be stored in internal memory.		

For example, executing the SAVEU command as shown below:

    SAVEU 10

stores a file in user memory with "u_" as the prefix and "10" is the file number. The file is listed in the user-memory catalog as:

    u_10

Executing the PREFX command as shown below:

    PREFX $MARCH$;SAVEU 10;

changes the prefix and inserts the prefix between the "u" and the "_" as shown below:

    uMARCH_10

The recall and load commands (RCLD, RCLT, RCLS, LIMIRCL, RCLU, and LOAD) require the complete file name, which includes both the prefix and the file number. To verify file names, press State , recall , catalog & MSI to view the catalog of user memory, which is the currently selected storage device.

See Also RCLD, RCLS, RCLT, RCLU, LOAD, DISPOSE, FORMAT, ERASE, PERASE

PROTECT

Prevent disposal or redefinition of internal memory.
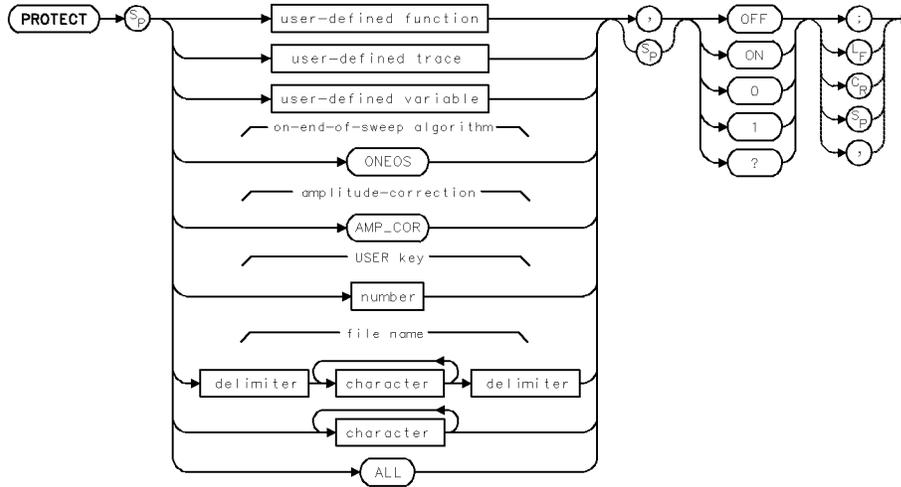
Item	Description
Range	For *user-defined trace*, *user-defined function*, or *user-defined variable*, range is 1 – 12 characters: A – Z, 0 – 9, and underscore |_|. First character must be a letter. Numbers must be preceded by underscore.
	For *number*, range is integers, 1 – 14.
Delimiters	Required before and after *file name* when it contains a number or lower-case letters. When *file name* is upper-case alpha characters only, delimiters are optional.

Example   10 OUTPUT 723;"PROTECT DLP,ON;"   *Protect a stored user-defined function called DLP.*

Description The PROTECT command prevents the disposal or redefinition of allocated internal memory.

PROTECT ALL protects the entire contents of internal memory against erasure or overwriting by the following commands.

- ACTDEF (active user-defined function)
- AMPCOR (amplitude correction factors)

- DISPOSE (Erases articles in this list.)
- FORMAT
- FUNCDEF (user-defined functions)
- KEYDEF (user-defined key)
- LIMILINE (user-defined traces, LIMIT_HI and LIMIT_LO)
- LIMISAV (limit-line files)
- LOAD (Load files and articles into internal memory.)
- ONEOS (on-end-of-sweep algorithm)
- PERASE (partial erase)
- SAVES (state-register file)
- SAVEU ($\boxed{\text{USER}}$ keys file)
- SAVET (trace file)
- STOR (Store any file.)
- TRDEF (user-defined trace)

N O T E

PROTECT does not prevent erasure by the ERASE or PURGE command

---

To selectively protect an article, specify the article name or number, as indicated in the syntax diagram. To discontinue protection, execute **PROTECT OFF**. The following table summarizes commands that protect or alter the different memories available, including internal memory.

**Commands that Erase or
Protect User Memory**

Memory Components	Commands that Erase or Reset Memory			Commands that Protect Memory	
	PURGE FORMAT	DISPOSE* PERASE* ERASE*	NSTATE	PROTECT†	PSTATE
user-defined variable |VARDEF, LOAD|	X	X		X	
user-defined trace |TRDEF, LIMILINE|	X	X		X	
user-defined function |FUNCDEF, LOAD|	X	X		X	
active user-defined function |ACTDEF, LOAD|	X	X		X	
on-end-of-sweep-algorithm |ONEOS, LOAD|	X	X		X	
trace-amplitude correction factors |AMPCOR|	X	X		X	
state register file |instrument settings and title line, SAVES, STOR|	X	X	X	X	X
program file |SAVED, STOR|	X	X		X	
trace file |SAVET, STOR|	X	X		X	
(USER) **keys file** |SAVEU, STOR|	X	X		X	
Limit line file |LIMISAV, STOR|	X	X		X	

* indicates command erases internal memory only.
†indicates command that protects internal memory only.

Query Response   `PROTECT ALL?` gives a 1 if any part of internal memory is protected. If all of internal memory is unprotected, `PROTECT ALL?` gives a 0.
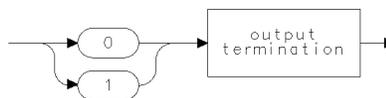
qgen0

See Also   ACTDEF, AMPCOR, DISPOSE, FUNCDEF, LIMISAV, LOAD, ONEOS, PERASE, SAVES, SAVET, SAVEU, STOR, VARDEF

PSTATE

Protect state-register files stored in user memory.
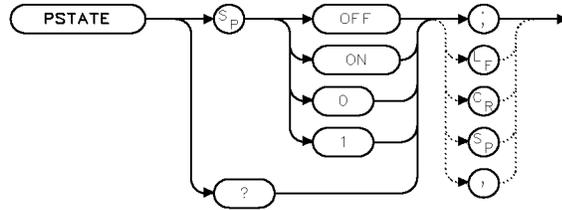
XPSTAT

Example

20	OUTPUT 723;"MSI MSIB;"	*Select memory card on HP 70004A Display.*
30	OUTPUT 723;"PSTATE OFF;"	*Turn of state-register protection.*
40	OUTPUT 723;"NSTATE 3;"	*Make 3 state registers available for storing instrument states.*
50	OUTPUT 723;"PSTATE ON;"	*Turn on state-register protection.*

Description The PSTATE command protects the contents of the state-register files from accidental erasure or over-writing by NSTATE, LOAD, STOR, or SAVES command. PSTATE does not prevent erasure of register contents with the ERASE, FORMAT, PERASE, or PURGE command.

Query Response
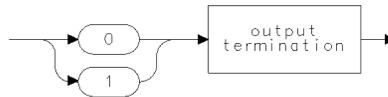
qgen0

See Also LOAD, NSTATE, PROTECT, SAVES, STOR

PU

Lift pen up.
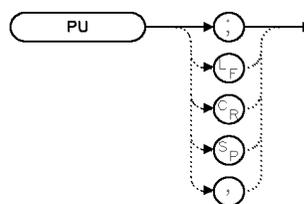
XPU

Item	Description
Preset State	Active. Pen is up.

Example

```
130   OUTPUT 723;"CLRDSP;"
140   OUTPUT 723;"PU;PA 500,500;"
150   OUTPUT 723;"PD;PA 1000,1000;"

160   OUTPUT 723;"PA PU 0,1000;"

170   OUTPUT 723;"PD;PA 1000,0;"
```

Clear graphics from memory.
Lift pen. Move pen to (500,500).
Place pen down. Draw line to upper-right corner.
Move pen to upper-left corner with pen in the up position. (See syntax diagram for PA.)
Place pen down and draw line to lower-right corner.

Description   The PU command "lifts up" the graphics pen from the analyzer display. Use PU to move the pen position with the PA and PR commands without drawing on the analyzer display.

See Also   PA, PR, GRID, MK, GRAPH, DSPLY, LINET

PURGE

Erase file from internal or external memory.
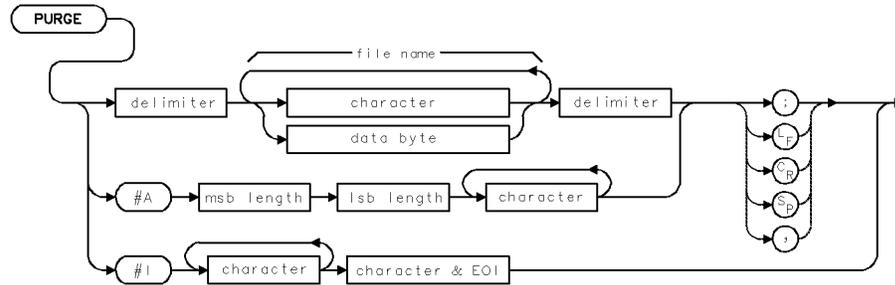
Item	Description
Number of characters	0 - 36

Example

```
10   OUTPUT 723;"MSI MSIB;"        Select memory card located at default HP-MSIB
                                   device.
20   OUTPUT 723;"PURGE $s_2$;"     Erase state-register file. Delimiters required when
                                   file name contains a number.
```

Description The PURGE command erases files from internal or external memory, regardless of protection asserted by the PSTATE or PROTECT command.

To purge a file, enter its complete name as it is listed in the catalog of user memory. To verify file names, press State , recall , catalog & MSI to view the catalog of user memory, which is the currently selected storage device.

PWRBW

Compute the bandwidth that contains a percentage of the total measured power.

XPWRBW

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$-32{,}768 - +32{,}767$ measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm 1.8 \times 10^{308}$ when using *predefined variable* or *user-defined variable*.
Percentage Range	0 – 100
Source Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.

Example

130	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
140	OUTPUT 723;"CENTERWL 750NM;SP 100NM;	*Activate maximum-hold mode. Sweep*
150	OUTPUT 723;"MXMH TRA;TS;TS;TS;TS;"	*trace A four times.*
160	OUTPUT 723;"PWRBW TRA,99.0?;"	*Query bandwidth containing 99%*
		of the power contained in full span.
170	ENTER 723;P	*Assign value to computer variable.*

Description The PWRBW command first computes the combined power of all signal responses contained in a trace array, then calculates the bandwidth which contains a percentage of the total power. For example, if 100% is specified, the power bandwidth equals the wavelength range of the trace data, which equals the span setting. If 50% is specified, trace elements are eliminated from each end of the array, until the combined power of the remaining signal responses equals 50% of the total power. The wavelength span of these remaining trace elements is the power bandwidth queried, centered around the center wavelength.

As another example, if PWRBW of 60% is requested, trace points are reduced from both ends until 20% of the power in one end and 20% of the power in the other end is eliminated. The PWRBW command does not alter the contents of the source trace.

PWRBW must be used as either a query or as a source in another optical spectrum-analyzer command. Form a query by ending the PWRBW statement with a question mark (?). When used as a query, PWRBW gives the bandwidth equal to a percentage of the total power. Use PWRBW as a source by incorporating the PWRBW statement into any optical spectrum-analyzer command containing *predefined function* in its syntax diagram. When PWRBW is used as a source, the bandwidth value is used for operation by the command that contains PWRBW.

Predefined-
Function
Syntax

Query Syntax

Query Response

```
      ┌──────────┐   ┌──────────────┐
─────▶│  number  │──▶│    output    │────▶
      └──────────┘   │ termination  │
                     └──────────────┘
```
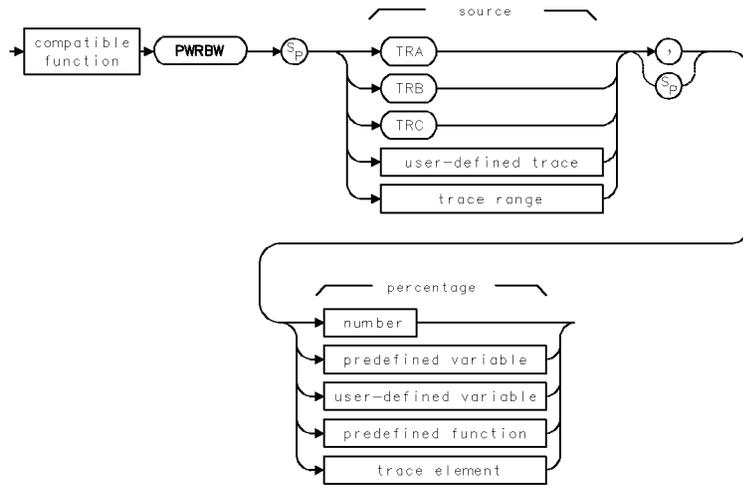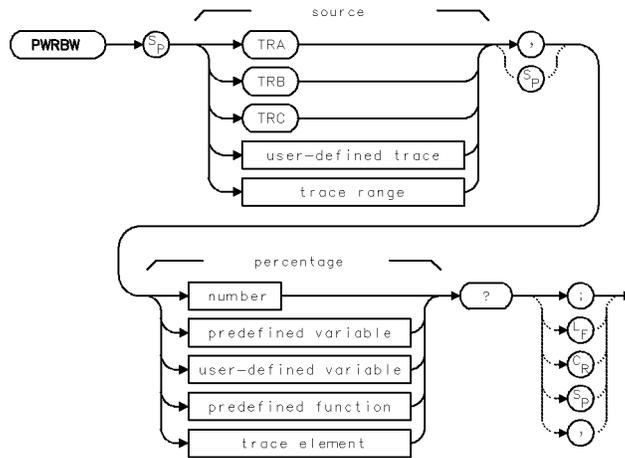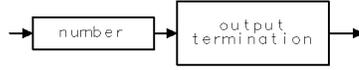
qgennum

RB

Set the resolution bandwidth.
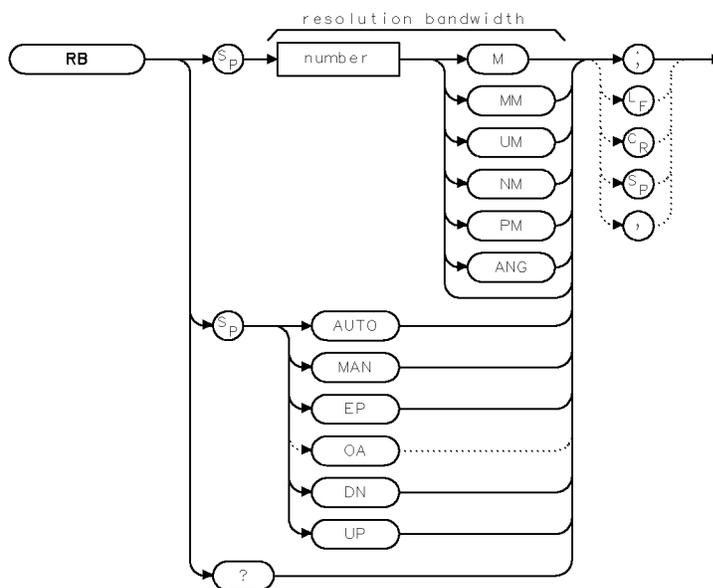
Item	Description
Resolution-Bandwidth Range	0.08 nm to 10 nm
Step Increment	0.08 and 1, 2, 5 sequence
Coupling	Refer to the optical spectrum analyzer users' guide.
Preset State	`RB AUTO`

Example   30   `OUTPUT 723;"RB 100NM;"`   *Set resolution bandwidth filter to 100 nm.*

Description   The RB command sets the resolution bandwidth. During normal operation, the resolution bandwidth, sweep time, and span settings are coupled to yield specified amplitude and wavelength accuracy with maximum measurement speed. Whenever any of these settings are changed, this optimum coupling is overridden, but the optical spectrum analyzer continues to maintain specified accuracy whenever possible. The uncalibrated warning, "UNCAL" is displayed whenever the new selection affects amplitude or wavelength accuracy.

Operation is as follow:

- If resolution bandwidth is set to automatic, the resolution bandwidth changes as the span changes according to the ratio set by RBR. If RB MAN is executed, or if the resolution bandwidth is selected with RB, the resolution bandwidth does not change with span.

- If the sweep time is set to automatic, the sweep time changes as the resolution bandwidth changes.

Query Response

```
──▶┌─────────┐──▶┌──────────────┐──▶
   │ number  │   │   output     │
   └─────────┘   │ termination  │
                 └──────────────┘
```
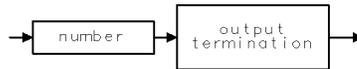
qgennum .

See Also   RBR, SP and wavelength functions when they change span significantly, such as CENTERWL, STARTWL, STOPWL, FS, SP

RBR

Set ratio of resolution bandwidth to span.
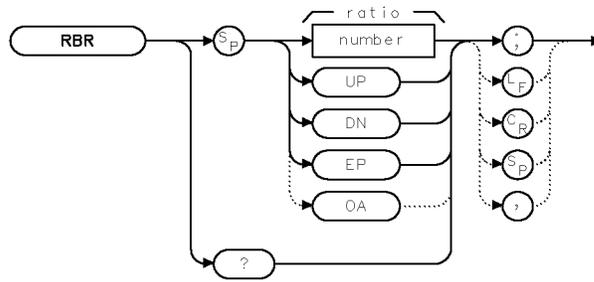
xrbr

Item	Description
Ratio Range	Actual range is hardware dependent. Firmware range is 10^{-100} – 10^{100}
Step Increment	1, 3, 10 sequence
Preset State	RBR .01

Example   160 OUTPUT 723;"RBR 0.03;"   *Set resolution-bandwidth/span ratio to 0.03.*

Description The RBR command determines the ratio between the resolution bandwidth and span settings when they are set to automatic.

$$RBR = \frac{resolution\ bandwidth}{span}$$

Query Response
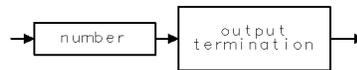
qgennum

See Also RB, SP

RCLD

Recall stored limit lines and user-defined functions, variables, keys, and traces.

```
                              ┌──── program file ────┐
  ( RCLD )─(S_P)──┬──→[ number ]──────────────────┬──→( ; )──→
                  │     ┌────────────────────────┐ │   ┊
                  └──→[ predefined variable ]──────┘  (L_F)
                  └──→[ user—defined variable ]──────→(C_F)
                  └──→[ predefined function ]────────→(S_P)
                  └──→[ trace element ]──────────────→( , )
```
XRCLD
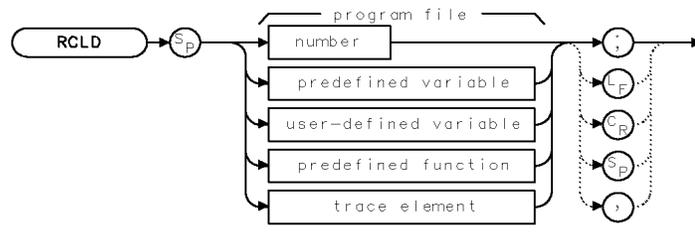
Item	Description
Range	0 – 9999, depending on memory

Example

```
160   OUTPUT 723;"RCLD 2;"
```

Description

The RCLD command recalls the contents of a program file from the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. Since program files cannot reside in internal memory, use MSI to select external memory.

RCLD specifies the number of the file for recall. In the catalog of user memory, the number is shown after the underscore. Refer to the SAVED and the LOAD commands for more information.

The contents of the recalled file consist of spectrum-analyzer programming commands. When the file is recalled, the contents are executed, storing articles in spectrum analyzer memory. The operation of the RCLD command varies with the contents of the file recalled.

- Files containing trace data are loaded into the specified trace, or trace A if the destination is not specified. All instrument settings are changed to reproduce the state the spectrum analyzer was in when the trace was stored. In addition, the trace is placed in view mode.

- Files containing user-defined variables, functions, traces, or keys are loaded into internal memory. Any duplicate user-defined articles over-write the existing versions.

- Files containing an on-end-of-sweep algorithm are loaded into internal memory. The existing on-end-of-sweep algorithm, if present, is erased and replaced with the loaded algorithm.

- Files containing a limit line are loaded into the limit-line table.

- Loading state-register files reactivates the stored instrument state.

- Loading user-key files reactivates the stored USER-key menu.

RCLS

Recall instrument settings and title line from a state-register file.
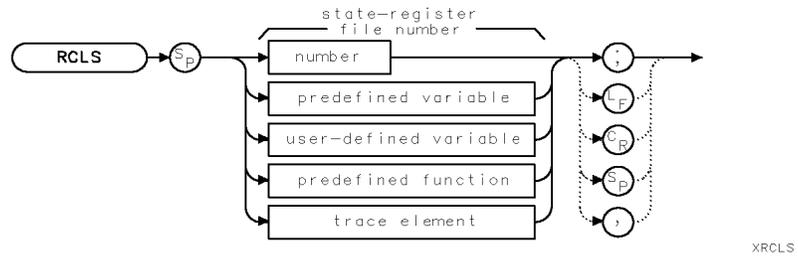
Item	Description
Range	0 – 9999, depending on memory

Example
```
160  OUTPUT 723;"RCLS 2;"
```

Description The RCLS command recalls the contents of a state-register file from the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. RCLS specifies the number of the file for recall. In the catalog of user memory, the number is shown after the underscore. The contents of the file are loaded into reserved memory, changing the instrument settings to those of the recalled state.

See Also LOAD, SAVES

RCLT

Recall contents of a trace file.
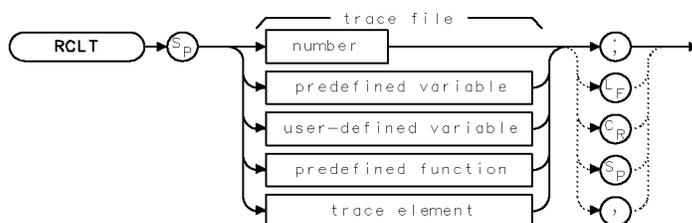
Item	Description
Range	0 – 9999, depending on memory

Example   160   OUTPUT 723;"RCLT 2;"

Description The RCLT command replaces trace A with the contents of a trace file from the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. All instrument settings are changed to reproduce the state the spectrum analyzer was in when the trace was stored. View mode is activated. RCLT specifies the number of the file for recall. In the catalog of user memory, the number is shown after the underscore.

See Also LOAD, SAVET

RCLU

Recall stored menu of (USER) keys.
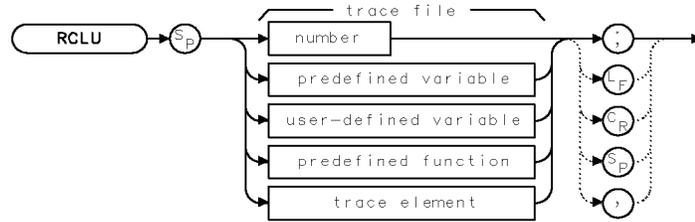
Item	Description
Range	0 – 9999, depending on memory

Example   160 OUTPUT 723;"RCLU 2;"

Description The RCLU command recalls the contents of a (USER) key file from the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. RCLU specifies the number of the file for recall. In the catalog of user memory, the number is shown after the underscore.

The contents of the file are loaded into the menu for the (USER) key, erasing the previous menu. Any user-defined custom keypad keys are also loaded. Refer to the KEYDEF command for information on defining custom keypad keys.

See Also KEYDEF, LOAD, SAVEU

READMENU   Build an interactive set of key functions.
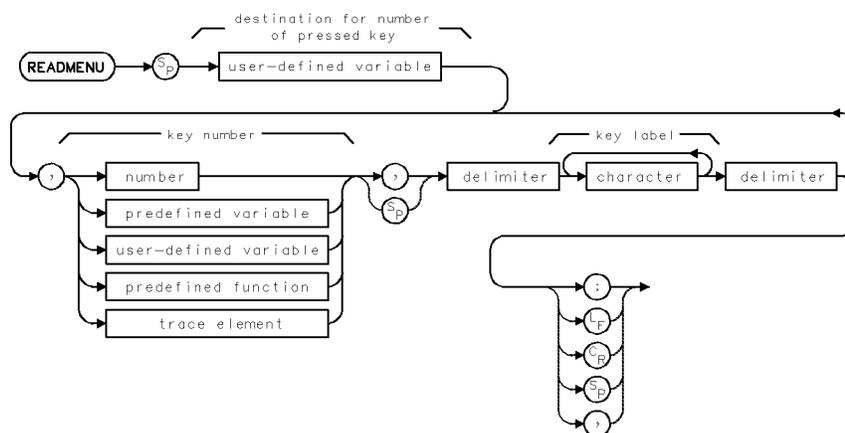
Item	Description
Key-Number Range	Integers, 1 – 14

Example

```
120   CLEAR 723                                        Initialize analyzer.
130   OUTPUT 723;"IP;"
140   OUTPUT 723;"VARDEF KEY,0;"                       Define a variable for storing the number of the
                                                       key that was last pressed.
150   OUTPUT 723;"REPEAT;";                            Begin repeat loop.
155   OUTPUT 723;"PA 400,700;";                        Position pen.
156   OUTPUT 723;"TEXT @EXIT KEY ENDS REPEAT LOOP.@;";
160   OUTPUT 723;"READMENU KEY,";                      Select a user-defined variable, KEY, where the
                                                       number of the key can be stored when it is
                                                       pressed. The keys are defined in the next pro-
                                                       gram lines.
170   OUTPUT 723;"1,%SET SPAN%,";                      SET SPAN is key number 1.
180   OUTPUT 723;"2,%EXPAND RANGE%,";                  EXPAND RANGE is key number 2.
190   OUTPUT 723;"14,%EXIT          %;";               EXIT is key number 14.
200   OUTPUT 723;"IF KEY,EQ,1;";                       If KEY equals 1, set measurement range to mea-
                                                       sure the calibrator signal.
210   OUTPUT 723;"THEN;CENTERWL 750NM;SP 100NM;";
```

```
220   OUTPUT 723;"ELSIF KEY,EQ,2;";                    If KEY equals 2, increase the measurement range.
230   OUTPUT 723;"THEN;CENTERWL 1000NM;SP 100NM;";
240   OUTPUT 723;"ELSIF KEY,EQ,14;";                   If KEY equals 14, then stop repeating lines 150 –
                                                       270.
250   OUTPUT 723;"THEN;ABORT;";
260   OUTPUT 723;"ENDIF;";
270   OUTPUT 723;"UNTIL KEY,EQ,14;"                    End repeat-loop definition.
```

Description   The READMENU command assigns a series of key labels to form a menu of
keys that can be accessed with the (USER) key. READMENU can be used
in conjunction with the IF/THEN/ELSE commands. No further commands
are processed until the user presses one of the menu keys given by the
READMENU command. After the execution is completed, the menus return
to their previous state, and the *destination for number of pressed key* (see
syntax diagram) contains the number of the key that was pressed. Keys are
numbered 1 through 14. Keys 1 through 7 and 8 through 14 line the right
and left sides of the display, respectively.

RELHPIB   Discontinue control of HP-IB.
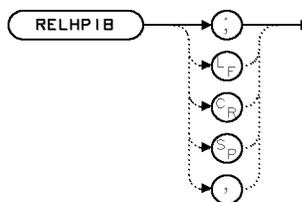
XRELHP

Example   140   OUTPUT 723;"OUTPUT 19,K,'STARTWL';OUTPUT 19,K,'F_A';OUTPUT
19,K,'HZ';"
The analyzer controls a device on HP-IB. The analyzer sends data via HP-IB.
150   OUTPUT 723;"RELHPIB;"
The analyzer releases control of HP-IB so that another device can control the bus.

Description The RELHPIB command releases analyzer control of HP-IB.

See Also   ENTER, OUTPUT

REPEAT/ UNTIL

Repeat a list of analyzer commands until a condition is met.

```
REPEAT ──┬─ ; ──┬──┬─ analyzer command ──┬── UNTIL ── Sp ──
         ├─ Lf ─┤  └─ user-defined function ─┘
         ├─ Cr ─┤
         ├─ Sp ─┤
         └─ , ──┘

  ┌─ number ─ units ─┬── , ──┬─ LT ─┬── , ──
  ├─ predefined variable ─┤  Sp    ├─ GT ─┤  Sp
  ├─ user-defined variable ┤       ├─ LE ─┤
  └─ trace element ────────┘       ├─ GE ─┤
                                   ├─ EQ ─┤
                                   └─ NE ─┘

  ┌─ number ─ units ─┬── ; ──
  ├─ predefined variable ─┤  Lf
  ├─ user-defined variable ┤  Cr
  └─ trace element ────────┘  Sp
                              ,
```

Example

250	OUTPUT 723;"STARTWL 600NM;SP 100NM"	*Set measurement range.*
270	OUTPUT 723;"REPEAT;";	*Begin repeat loop.*
280	OUTPUT 723;"MOV STARTWL,STOPWL;TS;";	*Set the start wavelength equal to the stop wavelength and take a sweep.*
290	OUTPUT 723;"UNTIL STARTWL,EQ,1000NM;"	*End loop when the start wavelength equals 1000 nm.*

Description The REPEAT and UNTIL commands work together to form a looping construct. All commands following the REPEAT command are executed repeatedly until the condition specified after the UNTIL command is satisfied.

The conditions are

- Less than (LT).

- Greater than (GT)

- Less than or equal to (LE).

- Greater than or equal to (GE).

- Equal to (EQ).

- Not equal to (NE).

If the condition is already satisfied at the time REPEAT/UNTIL is executed the first time, the commands following REPEAT are executed once.The function of REPEAT/UNTIL is limited to the amount of internal memory available at the time they are executed. An error results if there is not enough memory. Executing HP-IB device clear (CLEAR 723) stops the repeating process.

See Also IF/THEN, ABORT, RETURN

RETURN

Interrupt operation of current user-defined function, then return to the point where the function was originally encountered.

```
  ( RETURN )──────( ; )──────────▶
              ┊···( L )···┊
              ┊    ( F )  ┊
              ┊···( C )···┊
              ┊    ( R )  ┊
              ┊···( S )···┊
              ┊    ( P )  ┊
              └···( , )···┘
```
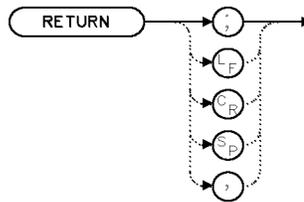
xretrn

Example The following example contains a user-defined function, called PROGRAM_1, nested inside another function, called PROGRAM_2, which itself is nested inside another function called PROGRAM_3. Thus, PROGRAM_3 executes PROGRAM_2, and PROGRAM_2 executes PROGRAM_1. PROGRAM_1 displays numbers (100, 200, 300, and 400), and also contains the RETURN command. When RETURN is encountered, operation returns to PROGRAM_2.

5	OUTPUT 723;"IP;"	*Instrument preset.*
10	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
20	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
30	OUTPUT 723;"ANNOT OFF;GRAT OFF;"	*Blank annotation and graticule.*
40	OUTPUT 723;"VARDEF YY,0;"	*Define user-defined variable equal to 0.*
50	!	
60	OUTPUT 723;"FUNCDEF PROGRAM_1,^";	*Define first user-defined function, PROGRAM_1, that prints YY + 100 until YY equals 300.*
70	OUTPUT 723;"REPEAT;";	*Begin repeat loop.*
80	OUTPUT 723;"ADD YY,YY,100;";	*Add 100 to YY.*
90	OUTPUT 723;"PU;PA 100,YY;PD;";	*Position pen at (100,YY).*
100	OUTPUT 723;"DSPLY YY,4,0;";	*Write value for YY.*
110	OUTPUT 723;"IF YY,EQ,300;THEN;RETURN;ENDIF;";	*If YY equals 300, execute RETURN, then skip to line 170 and execute the next analyzer command following PROGRAM_1.*
120	OUTPUT 723;"UNTIL YY,EQ,400;";	*End loop when YY equals 400.*
130	OUTPUT 723;"^;"	
140	!	
150	!	
160	OUTPUT 723;"FUNCDEF PROGRAM_2,";	*Define second user-defined function, PROGRAM_2.*

```
170 OUTPUT 723;"^PROGRAM_1; TEXT@INSIDE PROGRAM_2..@;^ ;";
180 !
190 OUTPUT 723;"FUNCDEF PROGRAM_3,";
200 OUTPUT 723;"^PROGRAM_2;TEXT@INSIDE PROGRAM_3@;^;";
210 !
220 OUTPUT 723;"PROGRAM_3;";
230 OUTPUT 723;"PU;PA 300,200;"
240 OUTPUT 723;"TEXT@OUTSIDE USER-DEFINED FUNCTIONS@;"
```

Execute PROGRAM_1 and write "INSIDE PROGRAM_2.."

Define third user-defined function, PROGRAM_3.
Execute PROGRAM_2 and write "inside PROGRAM_3."

Execute PROGRAM_3.
Position pen.
Write text.

Description The RETURN command interrupts operation of the currently operating user-defined function, as illustrated. The ABORT command also interrupts user-defined function operation, but interrupts operation of all user-defined functions instead.
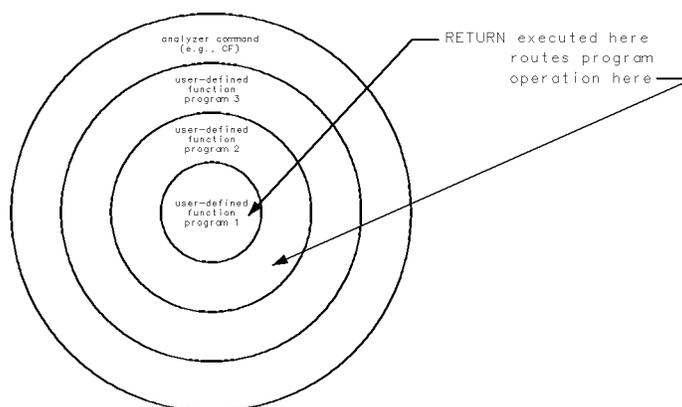
See Also IF/THEN, ACTDEF, FUNCDEF

REV

Return firmware date code.
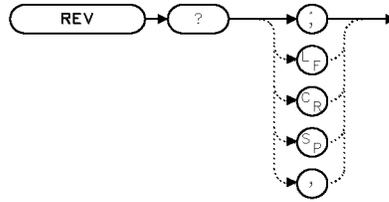
XREV

Example   10   OUTPUT 723;"REV?;"   *Query firmware version to computer.*

Description   The REV query returns the revision number of the firmware (ROM). The revision number is coded by date as shown below:

year − month − day

Query Response
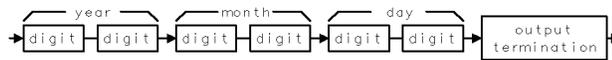
RL

Set the reference level.
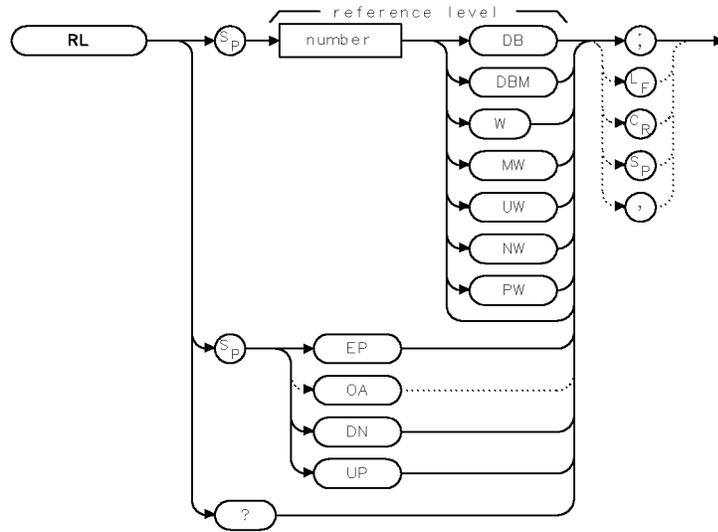
Item	Description
Reference-Level Range	Actual range is hardware dependent. Firmware range is ±300 dBm.
Step Increment	One vertical scale division.
Requirements	Update trace data after executing RL.
Preset State	`RL 0DBM`

Example

```
10  OUTPUT 723;"RL UP;"
```

Description The RL command sets the absolute amplitude value of the reference level. The default units for the reference level are watts for linear mode and dBm for log mode. Use the AUNITS command to specify different reference level units. The RLPOS command can be used to change the position of the reference level.

C A U T I O N	The reference level can be set to signal levels that damage the optical spectrum analyzer. Be sure to observe the maximum input power limits.

Query Response
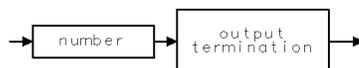
qgennum

See Also RLPOS

RLPOS

Select position of reference level.
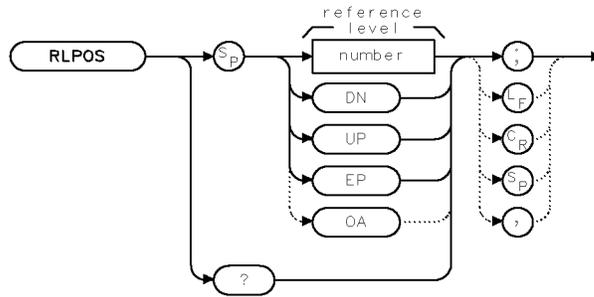
xrlpos

Item	Description
Range	0 - 10
Step Increment	1
Preset State	RLPOS 9

Example

    10 OUTPUT 723;"RLPOS 7;"   *Position the reference level at the seventh major graticule division.*

Description The RLPOS command changes the position of the reference level during log display mode. The top and bottom graticule lines correspond to 10 and 0, respectively. Arrows indicate the reference level line when it is not at the top graticule line.

Query Response
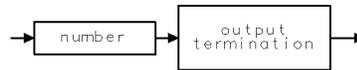
qgennum

See Also RL

RMS

Compute the root mean square of the specified trace.

XRMS

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA*, *TRB*, *TRC*, *trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA*, *TRB*, *TRC*, *trace range*, or *user-defined trace*.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA*, *TRB*, *TRC*.

Example Query the root mean square of all elements in trace A.

```
140   OUTPUT 723;"RMS TRA?;"          Query root-mean square.
150   ENTER 723;N                     Assign value to computer variable.
```

Description The RMS command computes the root-mean-square of the trace, in measurement units. RMS must be used as either a query or as a source in another analyzer-command function. Form a query by ending the RMS statement with a question mark (?). When used as a query, RMS gives the root-mean-square of the trace, in measurement units. Use RMS as a source by incorporating the RMS statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When RMS is used as a source, the root-mean-square value is used for operation by the command that contains RMS.

Predefined-
Function
Syntax
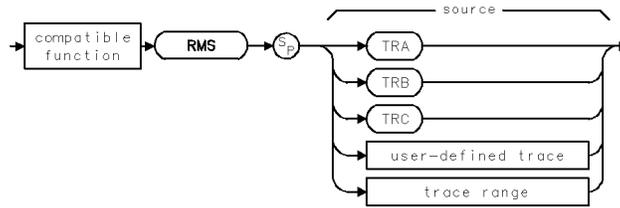
Query Syntax
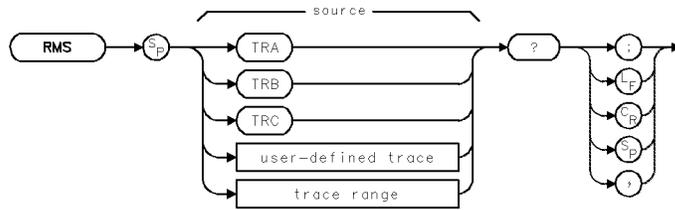
Query Response
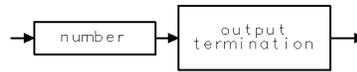
qgennum .

ROFFSET

Offset absolute value of amplitude measurements.

XROFF
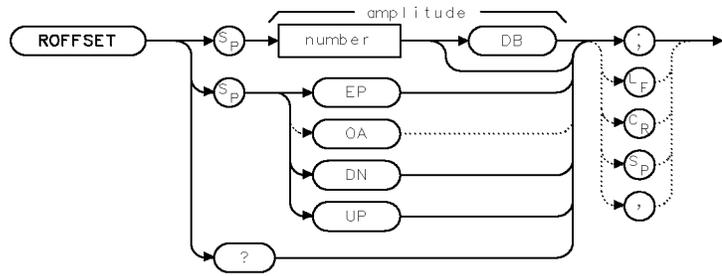
Item	Description
Range	Actual range is hardware dependent. Firmware range is ±300 dB.
Step Increment	1 major vertical scale division for log mode. 1 major vertical scale division at top of graticule (.915 dB) for linear mode.
Preset State	ROFFSET 0DB

Example

```
140   OUTPUT 723;"ROFFSET 60DB;"
```

Description The ROFFSET command offsets all amplitude readouts on the display without affecting the trace data. "ROFFSET" appears on the display when the offset is on. Values queried to the computer on HP-IB are also offset.

Query Response
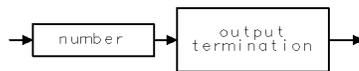
qgennum

See Also RL

RQS

Define conditions that interrupt computer operation.
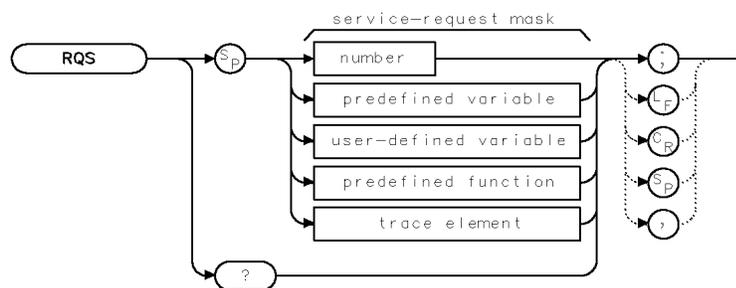
Item	Description
Range	Integers from 0 – 255

Example Set the service-request mask so that service is requested when the sweep is completed.

```
100   ASSIGN @Sa TO 723
110   CLEAR @Sa                                      Initialize analyzer.
120   OUTPUT @Sa;"IP;"
130   OUTPUT @Sa;"RQS 4;"                            Specify bit 2, the END-OF-SWEEP bit.
140   OUTPUT @Sa;"CENTERWL 750NM;SP 100NM;SNGLS;TS;"  Set measurement range, then sweep it.
150   ON INTR 7 GOTO Srq                             Go to the subroutine Srq at the end of the sweep.
160   S=SPOLL(@Sa)                                   Clear previous service requests.
170   ENABLE INTR 7;2                                Prepare to receive interrupt.
180   OUTPUT @Sa;"VAVG 10;"                          Begin video averaging, which repeatedly sweeps
                                                     the analyzer.

190 Wait:  GOTO Wait
200 Srq:   Sbyte=SPOLL(@Sa)                          Store value of status byte in Sbyte.
210        PRINT Sbyte                               Print status byte value.
220        OUTPUT @Sa;"RQS 0;"                       Clear service-request mask.
```

Description The RQS command enables the service-request mode. RQS defines the service-request mask that selects which bits of the status-byte register can generate a service request. Once RQS is executed, computer service is requested whenever any of the selected service conditions occur. For example, executing RQS 2 means that service is requested whenever a message is present in the message area. The analyzer requests service by setting bit 2 in the status-byte register true. Bit 6 of the status-byte register is set whenever the service-request condition is met. Bit 6 in the service-request mask does not need to be set in order for a service-request to occur. The service-request mask is shown in the following table.

Spectrum-Analyzer Service-Request Mask

Bit Number	Decimal Equivalent	Spectrum-Analyzer State	Description
7	128	— —	Unused.
6	64	RQS	Does not need to be set in order for a service-request to occur.
5	32	ERROR PRESENT	Set to request service when error register contains an error.
4	16	COMMAND COMPLETE	Set to request service at completion of command execution.
3	8	— —	Unused.
2	4	END OF SWEEP	Set to request service at completion of sweep.
1	2	MESSAGE	Set to request service when messages appear, such as UNCAL |uncalibrated|, UNCOR |uncorrected|, or power bar zeroing.
0	1	TRIGGER ARMED	Set to request service when analyzer is ready to receive a sweep trigger.

RQS is set to zero when power is applied to the spectrum analyzer or when it is cleared via HP-IB. (For example, the HP BASIC command, CLEAR 723, clears the optical spectrum analyzer when it is at HP-IB address 23, interface 7.)

Query Response
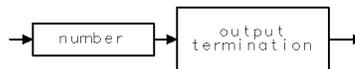
qgennum

See Also SRQ, STB, CLS

SAVED

Store limit lines and user-defined functions, variables, keys, and traces in a file.
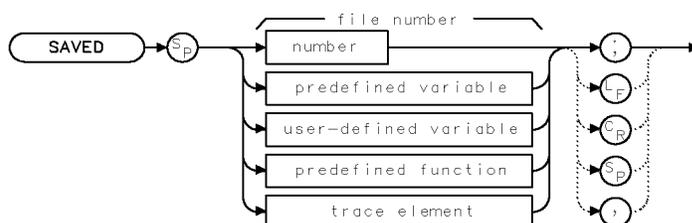
Item	Description
Range	0 – 9999, depending on memory

Description The SAVED command stores all of the following articles in a file if the articles are present in memory. The file type or identifier, as shown in the catalog of internal memory, is shown in parentheses.

- user-defined variables (VAR)
- user-defined functions (FUNC or AFUNC)
- on-end-of-sweep algorithms (ONEOS)
- user-defined traces (TRACE)
- currently active menu of (USER) keys (not listed in catalog)
- (USER) keys file
- limit lines

SAVED stores files in external memory only, which is specified by the MSI command. Since internal memory is the storage device when power is applied, the MSI command is needed to select the desired external memory. SAVED specifies the number of the file where the article(s) will be stored. The file name is a combination of the file number and the file prefix. When SAVED stores files, entries appear in the catalog as "d_N", where N is the number of the file, and "d_" indicates the file contains a user-defined article, such as a user-defined variable. For example, the entry for **SAVED 2** is listed in the catalog as "d_2". Use the PREFX command to change the "d_" prefix from its default setting to another combination of letters.

Use SAVED to transfer user-defined articles to memory outside of the optical spectrum analyzer. Once transferred to an external device, erase the entries from internal memory with the DISPOSE, PURGE, or PERASE command if additional internal-memory space is desired. Recall stored files with the RCLD or LOAD command.

SAVES   Save instrument settings and title line in a state-register file for future recall.
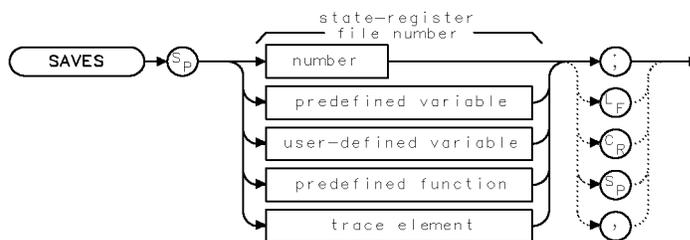
Item	Description
Range	1 – 9999

Example

```
30   OUTPUT 723;"PSTATE OFF;"      Turn off state-register file protection.
40   OUTPUT 723;"SAVES 3;"         Save instrument in file number 3.
```

Description The SAVES command saves the current spectrum-analyzer state in a state-register file, specified by a number. Information saved includes all front-panel settings and any text in the title line. SAVES stores files in user memory, which is the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. Files are listed in the catalog by default as "s_N", where N is the number of the file and "s_" indicates the file contains an instrument state. For example, the entry for **SAVES 2** is listed in the catalog as "s_2". Use the PREFX command to change the "s_" prefix from its default setting to another combination of letters. (Press State, recall, catalog & MSI to view the catalog of user memory.)

Use the RCLS or LOAD command to recall file contents. Use PURGE or DISPOSE to erase files. Use PROTECT or PSTATE to prevent inadvertent file erasure from internal memory only.

N O T E

The PSTATE and PROTECT commands do not protect state-register files against erasure or overwriting with the PURGE, ERASE, or LOAD command.

See Also DISPOSE, POWERON, PSTATE, NSTATE, RCLS

SAVET

Store trace A in a file.
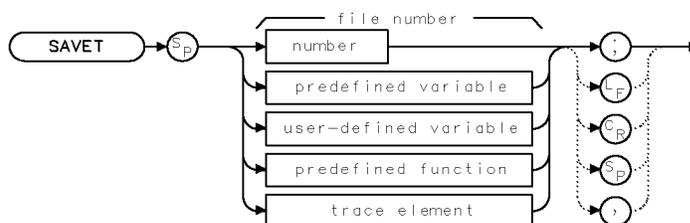
Item	Description
Range	0 – 9999, depending on memory

Example   30 OUTPUT 723;"SAVET;"

Description SAVET stores the contents of trace A in a file in user memory, which is the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. Files are listed in the catalog by default as "t_N", where N is the number of the file, and "t_" indicates the file contains a trace array. For example, the entry for **SAVET** 2 is listed in the catalog as "t_2". Use the PREFX command to change the "t_" prefix from its default setting to another combination of letters. (Press State , recall , catalog to view the catalog of user memory.) Use the RCLT or LOAD command to recall file contents. Use PURGE or DISPOSE to erase files. Use PROTECT to prevent inadvertent file erasure from internal memory only. The PROTECT command does not protect trace files against erasure or overwriting with the PURGE, ERASE, or LOAD command.

See Also DISPOSE, PROTECT, TRA, FETCH

SAVEU

Save (USER) keys in a file for future recall.
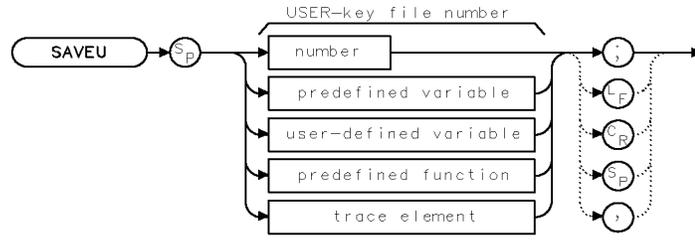
Item	Description
Range	Integer, 0 – 9999, depending on memory.

Example   30   OUTPUT 723;"SAVEU;"

Description The SAVEU command stores the user-defined keys that are currently accessed with the (USER) key. It also saves any user-defined custom keypad keys. Refer to the KEYDEF command for information on defining custom keypad keys.

SAVEU stores files in user memory, which is the currently selected mass storage device. Unless specified otherwise by the MSI command, internal memory is the storage device. Files are listed in the user-memory catalog by default as "u_*N*", where *N* is the number of the file, and "u_" indicates the file contains (USER) keys. For example, the entry for **SAVEU 2** is listed in the catalog as "u_2". Use the PREFX command to change the "u_" prefix from its default setting to another combination of letters. (Press State , recall , catalog & MSI to view the catalog of user memory.) Use the RCLU or LOAD command to recall file contents. Use PURGE or DISPOSE to erase files. Use PROTECT to prevent inadvertent file erasure from internal memory only. The PROTECT command does not protect trace files against erasure or overwriting with the PURGE, PERASE, ERASE or LOAD command.

See Also DISPOSE, KEYDEF, PROTECT

SCALE

Select X and Y scale for graphics.
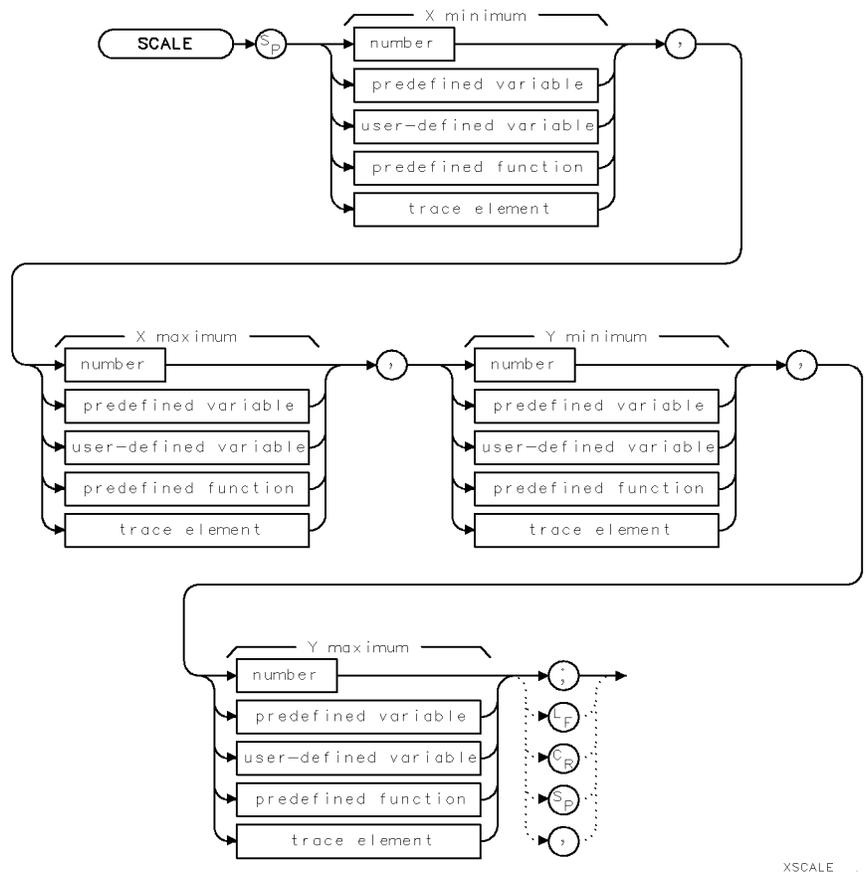
Item	Description
Range for X-Minimum, Y-Minimum, X-Maximum, Y-Maximum	Integers, $\pm 1.8 \times 10^{308}$
Preset State	`SCALE 0,1000,0,1000`

Example Use the SCALE command to draw $Y = X^2$ on the spectrum-analyzer display.

130	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
140	OUTPUT 723;"TRDSP TRA,OFF;"	*Turn off trace A.*
150	OUTPUT 723;"ANNOT OFF;"	*Turn off annotation.*
160	OUTPUT 723;"GRAT OFF;"	*Turn off graticule.*
170	OUTPUT 723;"SCALE -10,10,0,100;"	*Change minimum and maximum values for the X axis to −10 and 10 and the Y axis to 0 and 100.*
180	OUTPUT 723;"PU;PA -10,0;"	*Outline the perimeter of the coordinate system.*
190	OUTPUT 723;"PD;PA -10,100,";	*Semicolon suppresses carriage-return/linefeed.*
200	OUTPUT 723;"10,100,10,0,-10,0;"	
210	OUTPUT 723;"VARDEF XAXIS,-10;"	*Define user-defined variable, XAXIS, equal to −10.*
220	OUTPUT 723;"VARDEF YAXIS,0;"	*Define user-defined variable, YAXIS, equal to 0.*
230	OUTPUT 723;"PU;PA -10,100;PD;"	*Move pen.*
240	OUTPUT 723;"REPEAT;";	*Begin repeat loop.*
250	OUTPUT 723;"ADD XAXIS,XAXIS,1;";	*Increase XAXIS by 1.*
260	OUTPUT 723;"MPY YAXIS,XAXIS,XAXIS;";	*Square XAXIS.*
270	OUTPUT 723;"PA XAXIS,YAXIS;";	*Plot square function point by point until reaching the end of the X axis scale.*
280	OUTPUT 723;"UNTIL XAXIS,EQ,10;"	*End repeat loop when XAXIS = 10.*
290	OUTPUT 723;"PA PU -1,80;"	*Write label.*
300	OUTPUT 723;"TEXT$Y=X"	
310	OUTPUT 723;CHR$(130);"$;"	

Description The SCALE command scales the spectrum-analyzer display for drawing graphics. The default scale is 0 to 1000 for both the X and Y axis, which extend across the width and height of the spectrum-analyzer display. The following commands draw on the spectrum-analyzer display according to the units set by SCALE.

- Display variable (DSPLY)

- Graph trace (GRAPH)

- Display grid (GRID)

- Display marker (MK)

- Set origin (OR)

- Plot absolute value (PA)

- Plot relative value (PR)

- Display text (TEXT)

See Also DWINDOW, OR, MK, IWINDOW, DSPLY, GRAPH, GRID, PA, PR, TEXT, CLRDSP

SENS   Selects the input sensitivity level.
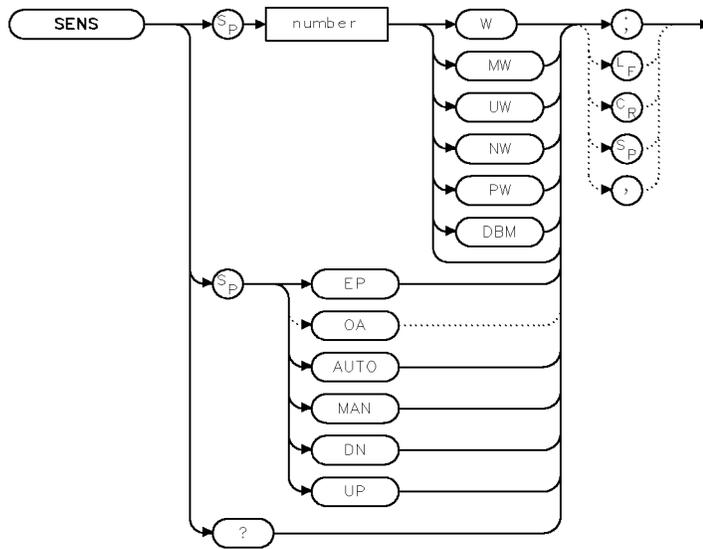
XSENS

Item	Description
Range	Actual range is hardware dependent. Firmware range is ± 300 dBm.
Step Increment	One vertical scale division.
Preset State	SENS AUTO

Example   10 OUTPUT 723;"SENS DN;"

Description The SENS command sets the instrument's sensitivity. When the AUTO argument is used, the optical spectrum analyzer automatically selects the sensitivity based on the current reference level. If video bandwidth is set manually, its value is also used to determine the sensitivity. Autoranging is turned off with AUTO. When the SENS command is set to MAN, autoranging is allowed and the video bandwidth is reduced, if necessary, to provide the requested sensitivity. The default units for the sensitivity include Watts for linear mode and dBm for log mode.

Notice that video bandwidth and internal gains are set as needed to achieve the requested sensitivity based on nominal values. Actual sensitivity realized may not equal the setting chosen.

Query Response   Results are always returned in dBm regardless of the value set by AUNITS.
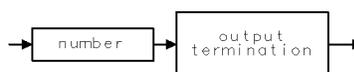
qgennum .

See Also   VBW

SER

Enter user serial number of instrument.
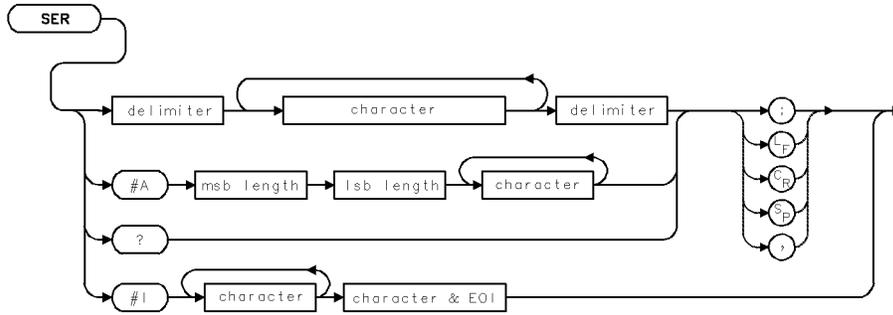
Item	Description
Range	Maximum of 21 characters

Example

150   OUTPUT 723;"SER?;"   *Query optical spectrum-analyzer serial number to computer.*

Query Response
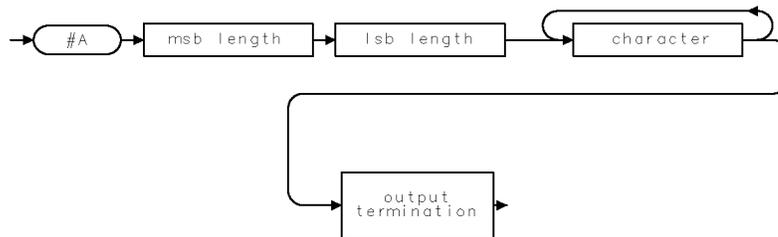
qgablock.

SMOOTH

Reduce amplitude variations on a trace.
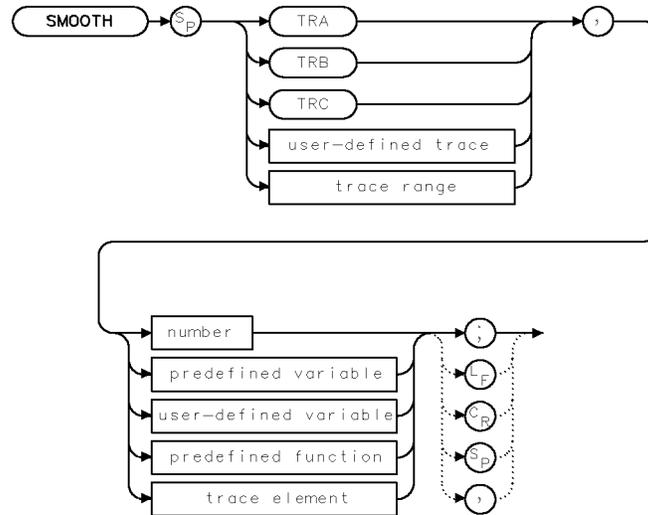
XSMOOT

Item	Description
Number-of-Points Range	3 to length of source
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	—32,768 – +32,767 measurement units for logarithmic trace data.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example

```
5   CLEAR 723
7   OUTPUT 723;"IP;"
10  OUTPUT 723;"VARDEF NN,0"
20  OUTPUT 723;"CENTERWL 750NM;SP 100NM;"
30  OUTPUT 723;"REPEAT;";
40  OUTPUT 723;"ADD NN,NN,10;";
50  OUTPUT 723;"SNGLS;CLRW TRA;TS;";
60  OUTPUT 723;"VIEW TRA;";
```

Initialize optical spectrum analyzer.

Define user-defined variable equal to 0.
Set measurement range.
Begin repeat loop.
Add 10 to NN.
Activate single-sweep mode. Sweep trace A.
Store and view trace A.

```
70 OUTPUT 723;"SMOOTH TRA,NN;";        Average each point of trace A with points NN/2 either
                                        side of it.
80 OUTPUT 723;"CLRDSP;";               Clear graphics from memory.
90 OUTPUT 723;"PA 100,700;TEXT$Smoothing over $;";   Write text.
100 OUTPUT 723;"DSPLY NN,4,1;";        Write the value of NN on the spectrum-analyzer
                                        display.
110 OUTPUT 723;"PA 740,700;TEXT$points$;";
120 OUTPUT 723;"WAIT 2S;";             Wait 2 seconds.
130 OUTPUT 723;"UNTIL NN,EQ,100;";     End repeat loop when each trace point has been
                                        averaged with 50 points on its left, and 50 points
                                        on its right side.
140 OUTPUT 723;"WAIT 1S;"              Wait 1 second.
150 OUTPUT 723;"IP;"                   Instrument preset.
```

Description   The SMOOTH command smooths the amplitude level of a trace. The amplitude values of the trace are modified according to the number of points specified for the running average. Each point value is replaced with the average of the values (in measurement units) of the given number of points centered around the point. The SMOOTH function performs a spatial, video averaging, in comparison to the temporal averaging achieved by the video-averaging command, VAVG. By replacing the value of each point in a trace with the average value of neighboring points, high-wavelength noise or signals are attenuated without a corresponding effect on low wavelength signals. Thus, the function of SMOOTH resembles narrowing of the video-bandwidth filter without the corresponding increase in sweep time. SMOOTH does reduce wavelength resolution.

Use SMOOTH as you would use video filtering to view low level signals near the noise floor. If the *number of points* parameter is an even number, then *number of points* is increased by one. If *number of points* is greater than the length of the source, then the size of the source is used for the smoothing operation (unless the size of the source is even, in which case the size of the source less 1 is used). The SMOOTH command requires sufficient internal memory to operate. Once the command is completed, any memory required by the SMOOTH function is released for other operations.

See Also   VAVG

SNGLS

Sweep on demand only.
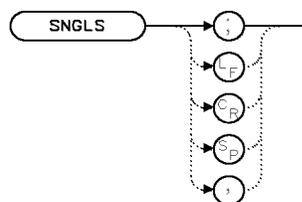
XSNGLS

Example			
	10	OUTPUT 723;"SNGLS;"	*Turn on single-sweep mode and sweep once.*
	20	OUTPUT 723;"TS;"	*Sweep once.*

Description The SNGLS command activates the single-sweep mode. During this mode, the spectrum analyzer sweeps once from the start to the stop wavelength whenever the TS or SNGLS command is executed and the trigger conditions are met. The SNGLS command cannot be substituted for the TS command. Though SNGLS can initiate a sweep, subsequent commands are executed before the sweep is completed. If a sweep is initiated with TS, subsequent commands are not executed until the sweep is completed.

See Also CONTS, DONE, TS, VAVG, ONEOS

SP

Set wavelength span of measurement.
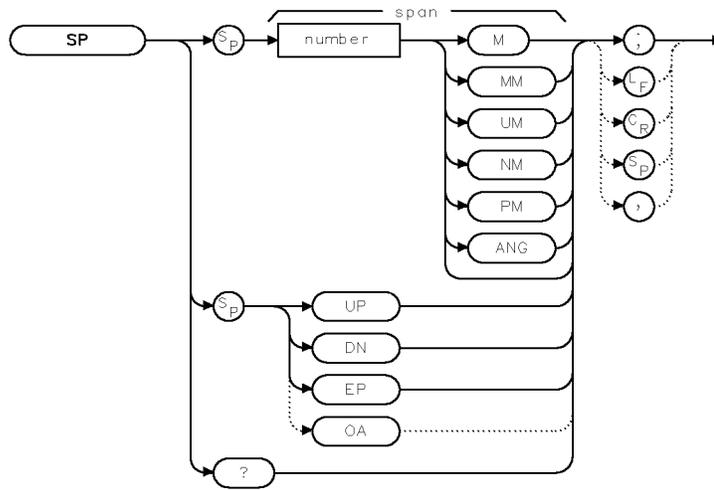
Item	Description
Range	Actual range is hardware dependent.
Step Increment	1,2,5,10 sequence
Coupling	$WL_{span} = WL_{stop} - WL_{start}$
Preset State	nm

Example

```
10  OUTPUT 723;"SP 500NM;"
```

Description The SP command sets the wavelength range symmetrically around the center wavelength. The zero-span mode is activated whenever the span is 0 Hz. Zero-span mode configures the spectrum analyzer as a fixed-tuned receiver, where measured signals are displayed in the time domain instead of the wavelength domain. Sweep time (ST) and resolution bandwidth (RB) are coupled to span.

Query Response

```
number → output
             termination →
```
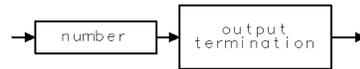
qgennum .

See Also RB, ST

SQR

Compute square root of source, point-by-point, and store results in the destination.

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm$ 1.8 $\times$ 10^{308} when using *number*, *predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example Set span to square root of another number.

```
110   OUTPUT 723;"SQR SP,1E9;"        Set span to square root of 10 to the ninth power.
```

SRINPUT

Select optical input for stimulus-response measurements.
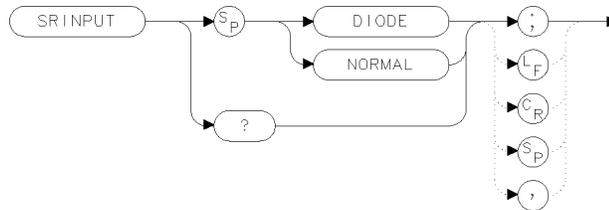
xsrput

Example   100 OUTPUT 723;"SRINPUT NORMAL;"   *Select the MONOCHROMATOR INPUT.*

Description On HP 70951B optical spectrum analyzer modules, the SRINPUT command selects the input connector for stimulus-response measurements. Choices are the MONOCHROMATOR INPUT (SRINPUT NORMAL) or the PHOTODETECTOR INPUT (SRINPUT DIODE) connectors.

An error is reported if the instrument mode is not set to stimulus-response when this command is executed or if the photodetector input is selected on an HP 70950B/2B module.

Query Response
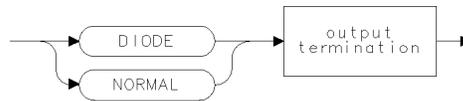
qsrput

See Also NORM

SRQ

Set bit(s) in status-byte register to simulate service request.

XSRQ
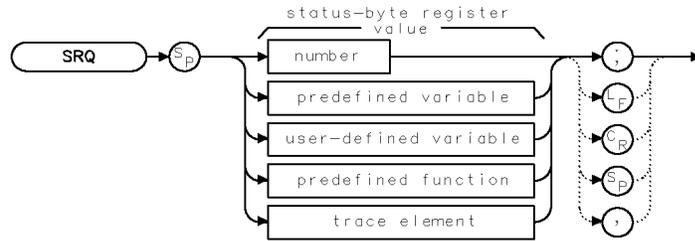
Item	Description
Range	0 – 255

Example   30 OUTPUT 723;"SRQ 4;"

Description The SRQ command simulates service requests by setting bits in the status-byte register. Use the SRQ command to test the operation of programs containing service requests. For example, suppose your program branched to a subroutine whenever a spectrum-analyzer error was detected. Instead of testing the program by initiating spectrum-analyzer errors, execute SRQ 32 to simulate an error condition.

Notice that bit 4, command complete, sets when this command finishes executing, regardless of whether the "command complete bit" was included in the value supplied.

Spectrum-Analyzer Status-Byte Register

Bit Number	Decimal Equivalent	Spectrum-Analyzer State	Description
7	128		Unused.
6	64	RQS	Set when status-byte register bit and corresponding bit in service-request mask are true. Indicates service-request has occurred.
5	32	ERROR PRESENT	Set when error register contains an error.
4	16	COMMAND COMPLETE	Set at completion of command execution.
3	8		Unused
2	4	END OF SWEEP	Set at completion of sweep.
1	2	MESSAGE	Set when messages appear, such as UNCAL |uncalibrated|, UNCOR |uncorrected|, or power bar zeroing.
0	1	TRIGGER ARMED	Set when analyzer is ready to receive a sweep trigger.

SS

Set the step increment of CENTERWL.
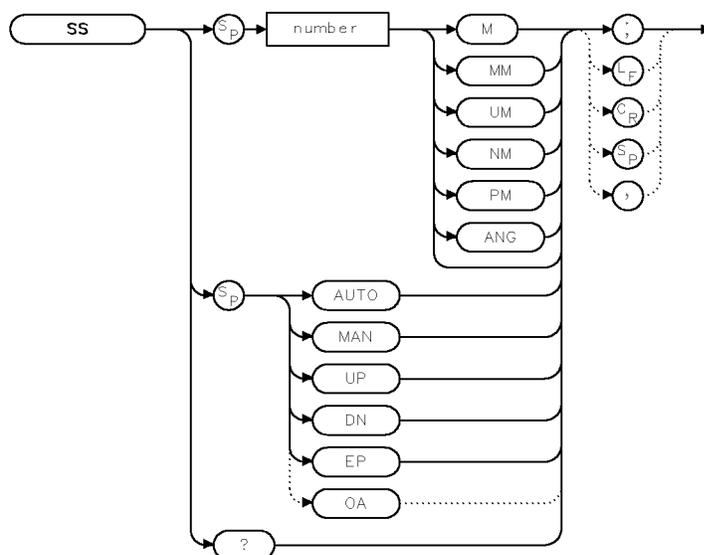
xsswl

Item	Description
Step-Size Range	Actual range is hardware dependent. Firmware range is $\pm$ 500 nm.
Step Increment	1, 2, 5, 10 sequence
Preset State	SS AUTO

Example

```
10 OUTPUT 723;"CENTERWL 750NM;SP 200NM;"
20 OUTPUT 723;"SS 25NM;"
30 OUTPUT 723;"CENTERWL UP;"
```

Set measurement range.
Set step size to 25 nm.
Increase center wavelength by one step size (25 nm).

Description The SS command sets the step increment of the center wavelength changes whenever **CENTERWL UP** or **CENTERWL DN** is executed.

SS AUTO activates the default step size. For spans greater than zero, the step size equals one graticule ($\frac{span}{10}$). For spans equal to zero, the step size equals the resolution bandwidth.

SS MAN indicates that the system uses a value entered an a previous, unknown occasion. In actuality, the SS MAN command de-couples STEP SIZE and sets the step size to the value current in memory.

Query Response
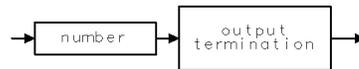
qgennum

See Also CENTERWL, MKSS, RB

ST   Set the time it takes to sweep from the start wavelength to the stop wavelength.
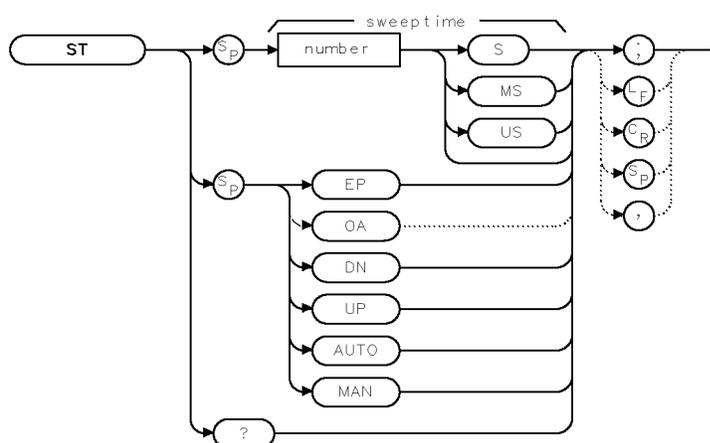
Item	Description
Sweep-Time Range	0 – 1000 seconds, or hardware limit.
Step Increment	1, 2, 5, 10 sequence
Coupling	Coupled to sensitivity |SENS|, video-bandwidth |VB|, and span |SP|.
Preset State	`ST AUTO`

Example   `120 OUTPUT 723;"ST 1S;"`   *Set sweep time to 1 second.*

Description   Sweep time, sensitivity, and video bandwidth are coupled to yield the maximum, signal-amplitude accuracy. Selecting a sweep time with ST, or executing `ST MAN` breaks the coupling. Execute `ST AUTO` to reestablish coupling. Faster sweep times or measurements are available by changing spectrum analyzer settings as follows:

- Reduce trace length (TRDEF). The spectrum analyzer requires 10 ms to store measurement results in an 800-point trace. Reducing trace length reduces storage time proportionately.

● Reduce the measurement range (SP).

The time between each sweep (retrace time) is reduced when the active trace and annotation are not displayed (`ANNOT OFF`; `TRDSP OFF`).

Query Response
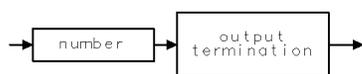
qgennum

See Also ANNOT, FETCH, SENS, SP, TRDEF, TRDSP, VB

STARTUP   Controls the start up configuration operation of the optical spectrum analyzer.
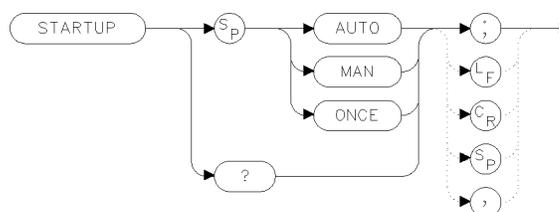
xstrup

Item	Description
Default (no argument)	ONCE

CAUTION

The STARTUP MAN and STARTUP ONCE commands both result in the loss of any internal user calibration data when the system restarts. This is because the data is part of the configuration. In addition, STARTUP MAN and STARTUP ONCE both force the POWERON command into IP if it was in LAST.

Example   100   OUTPUT 723;"STARTUP AUTO;"

Description   The first time the the optical spectrum analyzer is turned on, it establishes and checks the communications links with the display. Thereafter, these configuration steps are skipped at turn on, unless the MMS configuration has changed.

Argument	Definition

AUTO   The default operation which performs an automatic MMS configuration the first time the instrument is turned on.

MAN   Forces the system to reconfigure at each power up.

ONCE   Forces the system to reconfigure. After the reconfiguration, the value of reverts to AUTO.

For both STARTUP MAN and STARTUP AUTO, no effect is seen until the next power cycle. When the STARTUP ONCE command is executed, the

instrument break its links (if any) to a display. After the link manipulations and communications are complete, the optical spectrum analyzer performs a warm start reconfiguration with a power-up IP. After the reconfiguration the value of STARTUP reverts to AUTO.

Query Response
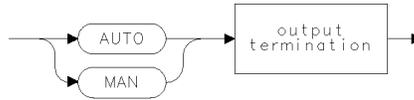
qstrup

STARTWL   Set start wavelength of measurement range.
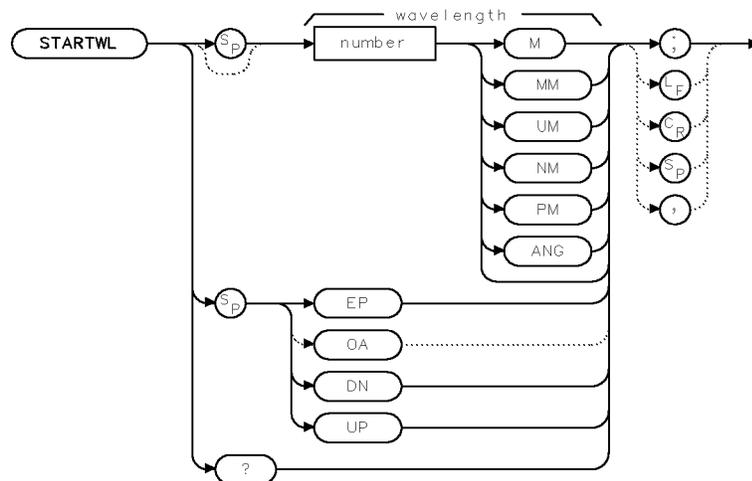
XSTAWL

Item	Description
Wavelength Range	Actual range is hardware dependent. Firmware range is 600 nm to 1700 nm. With wavelength limit OFF, the range is 350 nm to 2000 nm.
Step Increment	Span/10
Preset State	Minimum value allowed by system configuration.

Example

```
110  OUTPUT 723;"STARTWL 750NM;"     Set start wavelength to 750 nm.
120  OUTPUT 723;"STARTWL UP;"        Increase start wavelength by 10% of the span.
```

Description The STARTWL command specifies the start wavelength of the measurement range. The start- and stop-wavelength settings are independent, except when the stop wavelength is set lower than the start. In this case, the start wavelength is set equal to the new stop wavelength.

Query Response
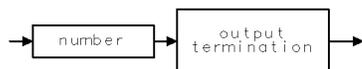
qgennum

See Also   Commands that affect wavelength, such as CENTERWL, STOPWL, FS, SP,
WLOFFSET,

STATE

Return or receive instrument state information.
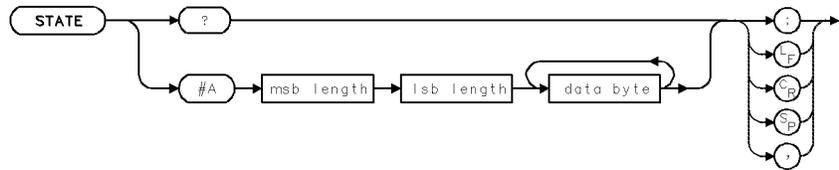
Example	140	ALLOCATE State$[1000]	*Allocate a string variable array, State, for storing the spectrum-analyzer state in the computer.*
	160	OUTPUT 723;"STATE?;"	*Query spectrum-analyzer state to computer.*
	170	ENTER 723 USING "-K";State$	*Save response in State in the form of ASCII tokens, requiring a special image ("−K") to receive the data.*
	190	OUTPUT 723;"IP;"	*Preset spectrum-analyzer state.*
	210	OUTPUT 723;State$[5];	*Restore spectrum-analyzer state by transmitting the relevant contents of State from the computer back to the spectrum analyzer, starting with 5 element in array.*

Description The STATE command transmits all instrument-state information (including front-panel settings) from or to a computer.

Query Response Initial data in *data byte* is "STATE #A".
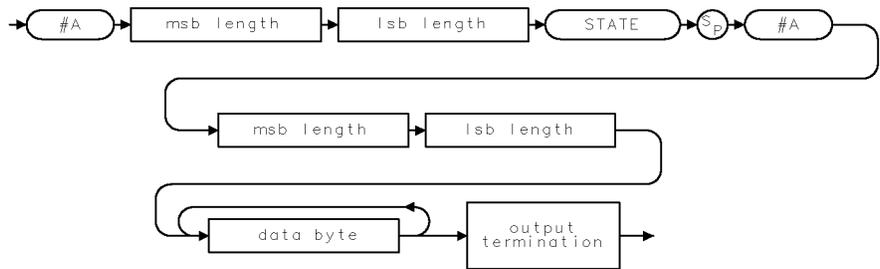
qstate

STB

Return value of status-byte register.
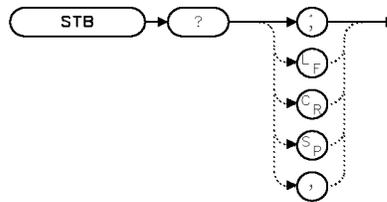
```
         ┌─────────┐   ┌───────┐      ┌───┐
────────▶│   STB   │──▶│   ?   │──┬──▶│ ; │──────────────────▶
         └─────────┘   └───────┘  ┊   └───┘        ▲
                                  ┊   ┌───┐        ┊
                                  ┊┄▶│ L_F │┄┄┄┄┄┄┄┤
                                  ┊   └───┘        ┊
                                  ┊   ┌───┐        ┊
                                  ┊┄▶│ C_R │┄┄┄┄┄┄┄┤
                                  ┊   └───┘        ┊
                                  ┊   ┌───┐        ┊
                                  ┊┄▶│ S_P │┄┄┄┄┄┄┄┤
                                  ┊   └───┘        ┊
                                  ┊   ┌───┐        ┊
                                  └┄▶│ , │┄┄┄┄┄┄┄┘
                                      └───┘
                                                        XSTB
```

Example   Monitor spectrum-analyzer operation without using service-request methods.

10	`CLEAR 723`	*Preset spectrum analyzer.*
20	`OUTPUT 723;"IP;CLRDSP;"`	
30	`OUTPUT 723;"SNGLS;"`	*Activate single-sweep mode.*
40	`OUTPUT 723;"CLS;"`	*Clear status-byte register.*
50	`OUTPUT 723;"TS;"`	*Sweep spectrum analyzer.*
60	`OUTPUT 723;"STB?;"`	*Query status of status byte, then clear it. The value queried is the sum of 16 (command complete), 4 (end of sweep), and 1 (trigger armed), which equals 21.*
70	`ENTER 723;N`	*Assign value to computer.*
80	`OUTPUT 723;"STB?;"`	*Query status of status byte. Value queried is 16 (command complete), since this is the only task the spectrum analyzer has performed since the status byte was cleared in line 60.*
90	`ENTER 723;M`	
100	`DISP N,M`	

Description   Use the STB command to monitor spectrum analyzer operation. Certain multiple-tasked computer systems, such as UNIX-based computers, monitor spectrum-analyzer operation more efficiently with the STB command than with the service-request command, RQS.

Use STB to monitor spectrum-analyzer operation randomly at any time. Executing STB is similar to conducting a serial poll of the status-byte register. When STB is executed, the status-byte register is cleared (all bits set to zero). The STB command gives the decimal equivalent of the status-byte register. For example, the COMMAND-COMPLETE and ERROR-PRESENT bits are set when 48 (16 + 32) is returned.

Executing **STB?** always sets the command-complete bit. For this reason, STB cannot be used to monitor the completion of commands. Refer to the DONE command.

STB must be used as either a query or as a source in another analyzer-command function. Form a query by ending the STB statement with ?. When used as a query, STB gives the decimal equivalent of the status-byte register to the computer. Use STB as a source by incorporating the STB statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When STB is used as a source, the decimal equivalent is used for operation by the command that contains STB.

Spectrum-Analyzer Status-Byte Register

Bit Number	Decimal Equivalent	Spectrum-Analyzer State	Description
7	128		Unused.
6	64	RQS	Set when status-byte register bit and corresponding bit in service-request mask are true. Indicates service-request has occurred.
5	32	ERROR PRESENT	Set when error register contains an error.
4	16	COMMAND COMPLETE	Set at completion of command execution.
3	8		Unused.
2	4	END OF SWEEP	Set at completion of sweep.
1	2	MESSAGE	Set when messages appear, such as UNCAL |uncalibrated|, UNCOR |uncorrected|, or power bar zeroing.
0	1	TRIGGER ARMED	Set when analyzer is ready to receive a sweep trigger.

Query Response
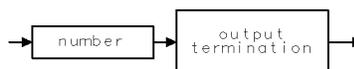
qgennum .

See Also CLS, SRQ, RQS

STDEV

Compute standard deviation of trace amplitude.

XSTDEV

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	– 32,768 – + 32,767 measurement units for logarithmic trace data.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example

130	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
140	OUTPUT 723;"CENTERWL 750NM;SP 20NM;RB 100NM;"	*Set measurement range.*
150	FOR N=1 TO 10	*Begin repeat loop using system level commands in HP BASIC.*
160	WAIT 1	*Pause.*
170	OUTPUT 723;"SP DN;TS;"	*Reduce the span. Sweep trace A.*
180	OUTPUT 723;"STDEV TRA?;"	*Query standard deviation of trace A.*
190	ENTER 723;S	*Assign value to computer variable.*
200	PRINT "Standard deviation is";S/100;"DB"	
210	NEXT N	*End repeat loop.*

Description   The STDEV command calculates the standard deviation of the trace amplitude, in measurement units.

STDEV must be used as either a query or as a source in another analyzer-command function. Form a query by ending the STDEV statement with a question mark (?). When used as a query, STDEV queries the standard deviation to the computer. Use STDEV as a source by incorporating the STDEV statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When STDEV is used as a source, the standard deviation is used for operation by the command that contains STDEV.

Predefined-
Function
Syntax
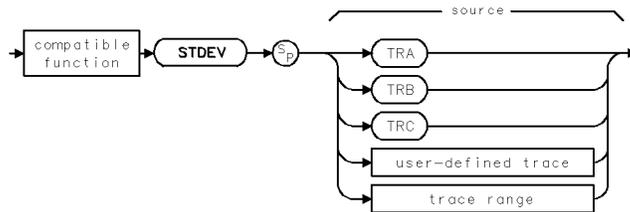
Query Syntax
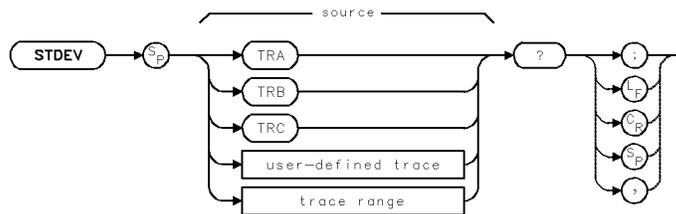
Query Response
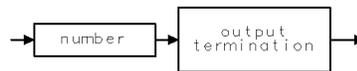
qgennum

STOPWL

Set stop wavelength of measurement range.
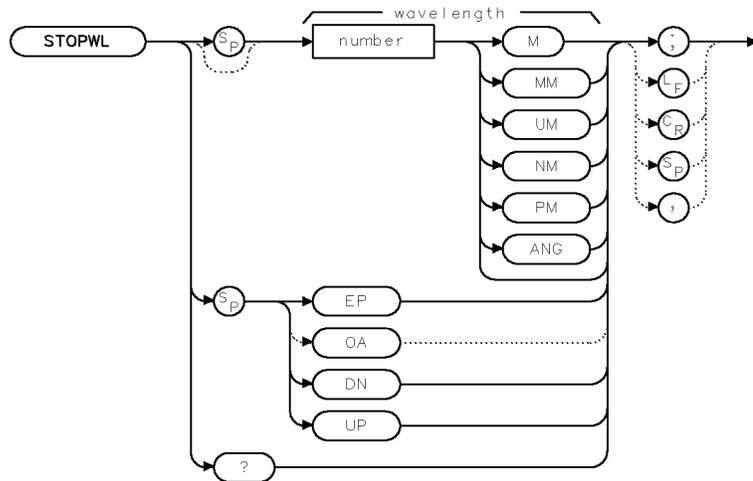
XSTOWL

Item	Description
Wavelength Range	Actual range is hardware dependent. Firmware range is 600 nm to 1700 nm. With wavelength limit OFF, the range is 350 nm to 2000 nm.
Step Increment	Span/10
Preset State	Maximum value allowed by system configuration.

Example

```
10   OUTPUT 723;"STOPWL 1200NM;"
```

Description The STOPWL command specifies the stop wavelength of the measurement range. The start- and stop-wavelength settings are independent, except when the start wavelength is set higher than the stop wavelength. In this case, the stop wavelength is set equal to the new start wavelength.

Query Response
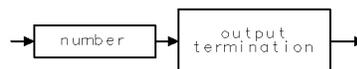
qgennum

See Also   Commands affecting wavelength, such as CENTERWL, STARTWL, FS, SPAN, WLOFFSET

STOR

Store user-defined articles.
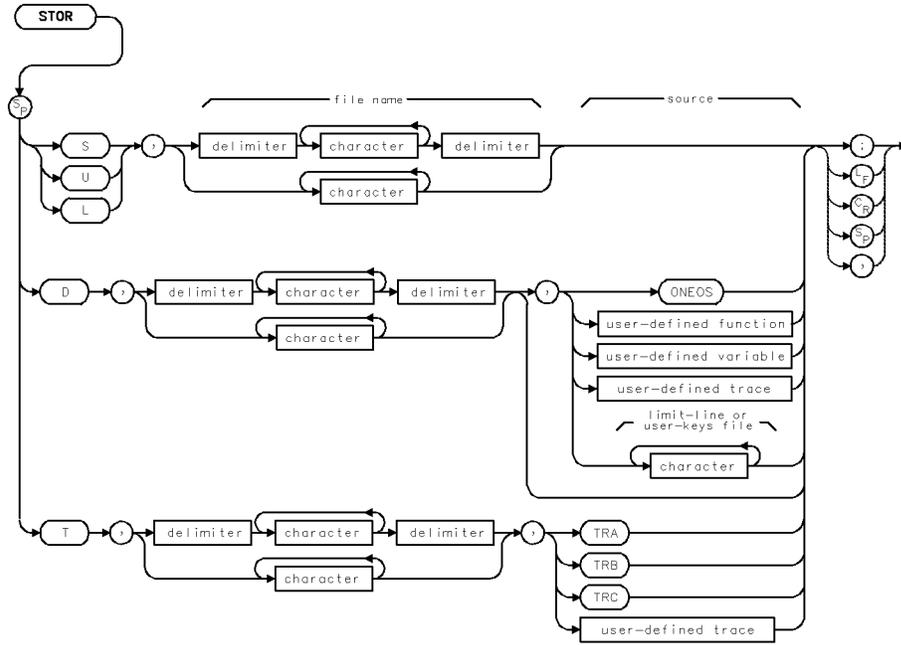
Item	Description
File-Name Range	0 – 10 characters
Delimiters	Required before and after *file name* when it contains a number or lower-case letters. When *file name* is upper-case alpha characters only, delimiters are optional.

Example

```
10  OUTPUT 723;"STOR S,$file_1$;"
```

Store the file "FILE_1" of instrument settings in internal memory.

Description   STOR stores files in the currently selected user memory. Unless specified otherwise by the MSI command, internal memory is the storage device. The information stored depends on the argument selected.

S stores the current instrument settings and the contents of the title line.

D stores all of the following user-defined articles in **external memory** only.

- user-defined variable
- user-defined function
- on-end-of-sweep algorithm
- user-defined trace
- user-defined keys
- limit lines

T stores trace A, B, or C, or a user-defined trace. If the item for storage is not specified, trace A is stored.

U stores the current configuration of keys accessed with the (USER) key.

L stores the current contents of the limit-line table.

When storing a user-defined article, enter its complete name. To verify names, view the catalog of user memory. (Press `State`, `recall`, `catalog & MSI` to view the catalog.) Recall stored files with the LOAD command. Erase stored files with the PURGE or DISPOSE command. Use the PROTECT command to prevent inadvertent file erasure from internal memory only. (The PROTECT command does not prevent erasure with the PURGE, PERASE, or ERASE command.)

Description of User-Memory Catalog

Article Stored	Storing Command	File Type	Standard File Prefix
user-defined function†	FUNCDEF, RCLD, LOAD	FUNC	none
active user-defined function†	ACTDEF, RCLD, LOAD	AFUNC	none
user-defined variable†	VARDEF, RCLD, LOAD	VAR	none
user-defined trace†	TRDEF, LIMILINE, RCLD, LOAD	TRACE	none
on-end-of-sweep algorithm†	ONEOS, RCLD, LOAD	ONEOS	none
file containing user-key menu	SAVEU, STOR	ASCII	u_
file containing trace	SAVET, STOR	ASCII	t_
file containing user-defined variable*	SAVED, STOR	ASCII	d_
file containing user-defined function*	SAVED, STOR	ASCII	d_
file containing on-end-of-sweep algorithm*	SAVED, STOR	ASCII	d_
file containing limit line	LIMISAV, STOR	ASCII	l_
file containing state register [instrument settings and title-line]	SAVES, STOR	ASCII	s_
*Stored in external memory only. † Stored in internal memory only.			

See Also PROTECT, PSTATE

STORREF   Store through references for stimulus-response measurements.
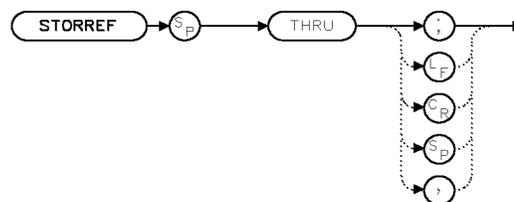
XSTORF

Example   `10  OUTPUT 723;"STORREF THRU;"`

Description **STORREF THRU** discontinues measurements with trace B, then transfers through-reference amplitude information measured with trace A to trace B.

STORREF displays "Not stored; A − X ▶ A on" and does not operate when the following functions are on:

- Trace A − Trace B (**AMB ON**)
- Trace A − Trace B + Display Line (**AMBPL ON**)
- Trace A − Trace C (**AMC ON**)

If **AMB** is executed after STORREF, measurements made with trace A are normalized (weighted) according to the contents of trace B. If **AMC** is executed after STORREF, trace A is normalized according to the contents of trace C.

See Also CLRW, PDMEAS, AMB, AMC, AMBPL

SUB

Subtract.

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	$\pm\ 1.8 \times 10^{308}$ when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*. Truncated if longer than destination length. If shorter than destination length, last element is extended for operation.
Destination-Length Range	1 – 2048 position units.

Example   10   OUTPUT 723;"SUB CENTERWL,CENTERWL,100NM;"   *Reduce center wavelength by 100 nm.*

Description   The SUB command subtracts source 2 from source 1, point by point, then stores the value in the destination.

When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep using the SNGLS and TS commands.

SUM

Compute the sum of amplitude levels detected in each point of a trace.
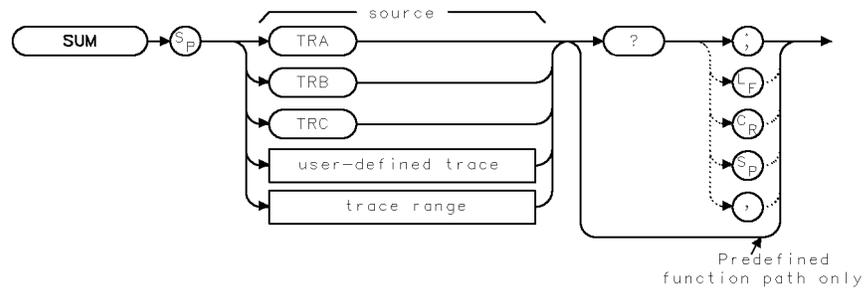
XSUM

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	–32,768 – +32,767 measurement units for logarithmic trace data.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC.*

Example Add the amplitudes of all points in trace A.

100	OUTPUT 723;"SNGLS;TS;"	*Activate single-sweep mode. Sweep trace A.*
110	OUTPUT 723;"SUM TRA?;"	*Query sum to computer.*
120	ENTER 723;N	*Assign value to computer variable.*
130	DISP N;"MEASUREMENT UNITS"	*Display value.*

Description The SUM command gives the sum of the trace amplitudes in measurement units. SUM must be used as either a query or as a source in another analyzer-command function. Form a query by ending the SUM statement with a question mark (?). When used as a query, SUM queries the sum of the trace amplitudes to the computer. Use SUM as a source by incorporating the SUM statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When SUM is used as a source, the sum of the trace amplitudes is used for operation by the command that contains SUM. When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep using the SNGLS and TS commands.

Predefined-Function Syntax
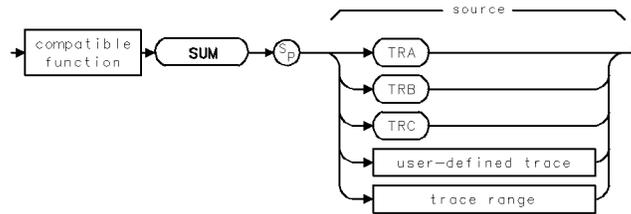
Query Syntax
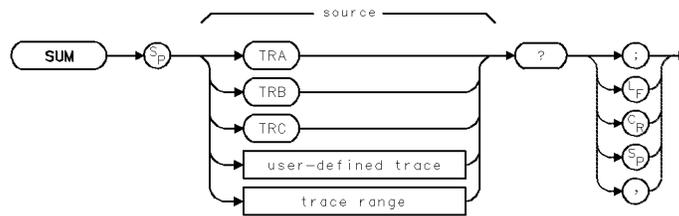
Query Response
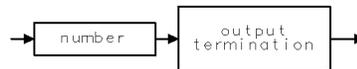
qgennum

SUMSQR

Compute the sum of the square of the amplitude levels detected in each point of a trace.

XSUMSQ

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	−32,768 – +32,767 measurement units for logarithmic trace data.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example

```
100   OUTPUT 723;"SNGLS;TS;"
110   OUTPUT 723;"SUMSQR TRA?;"

120   ENTER 723;N
130   DISP N;"MEASUREMENT UNITS"
```

Activate single-sweep mode. Sweep trace A.
Square the amplitude of all points in trace A, then query their sum.
Assign value to computer variable.
Display value.

Description The SUMSQR command gives the sum of the square of the trace-point amplitudes in measurement units. SUMSQR must be used as either a query or as a source in another analyzer-command function. Form a query by ending the SUMSQR statement with a question mark (?). When used as a query, SUMSQR queries the sum of the square of the trace amplitudes to the computer. Use SUMSQR as a source by incorporating the SUMSQR statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When SUMSQR is used as a source, the sum of the squares is used for operation by the command that contains SUMSQR. When the source is TRA, TRB, TRC, or trace range, ensure the data reflects one complete sweep using the SNGLS and TS commands.

Predefined-Function Syntax
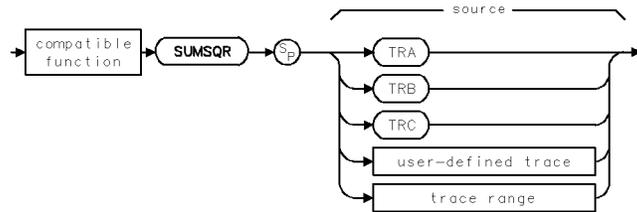
Query Syntax
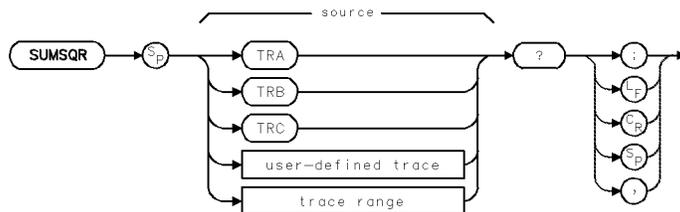
Query Response
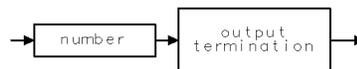
qgennum

SWEEP

Start or stop analyzer sweep.
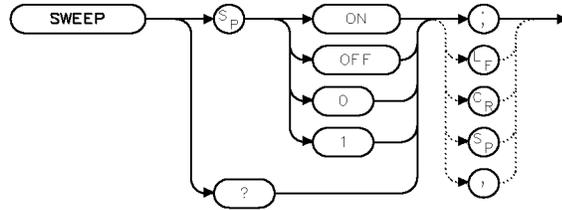
XSWEEP

Example   30 OUTPUT 723;"SWEEP OFF;"

Description The SWEEP command is used to stop the optical spectrum analyzer from sweeping, immediately. Sending SWEEP ON, IP, TS, CONTS, or SNGLS starts the sweep.

Query Response
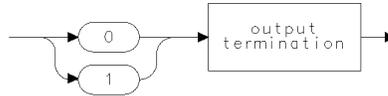
qgen0

SWPMODE?

Return the current sweep mode.
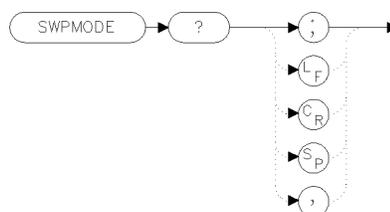
xswpde

Example   `100  OUTPUT 723;"SWPMODE?;"`

Description The SWPMODE command returns the current sweep mode of the instrument. The returned string will be either SNGLS or CONTS, depending upon whether the instrument is in single sweep or continuous sweep respectively.

Query Response
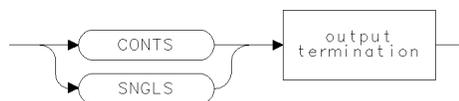
qswpde

See Also CONTS, SNGLS

TDF

Format trace data: binary, block, ASCII-decimal, measurement units, or parameter units.
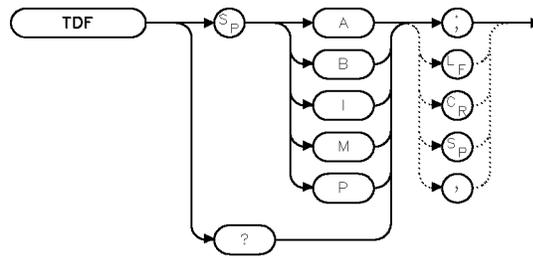
Item	Description
Default Mode	P

Example

```
110   CLEAR 723;                                         Preset spectrum analyzer.
120   OUTPUT 723;"IP;"
130   OUTPUT 723;"VIEW TRA;TRA[50]?;"                     Query value of 50th element of trace A.
140   ENTER 723;M                                         Assign value to computer variable.
150   DISP M                                              Print value.
160   OUTPUT 723;"TDF A;"                                 When requested, send data as an absolute,
                                                          block-data field. (Trace data is preceded by
                                                          #A.)
170   OUTPUT 723;"TRA?;"                                  Query trace-A amplitude information.
180   ENTER 723 USING "#,AA,W";Pound_a$,Number_of_bytes   Format computer so that it receives data until
                                                          the variable list is satisfied. "AA" expects two
                                                          characters (#A) and W reads data as a word
                                                          (16-bit words).
190   ALLOCATE Data$[Number_of_bytes]                     Define a computer array for storing the infor-
                                                          mation. The length of the array is as long as
                                                          the trace.
200   ENTER 723 USING "-K";Data$                          Store data in the computer array.
210   OUTPUT 723;"MOV TRA,0;"                             Store zeroes in trace A.
220   OUTPUT 723;"TRA #I";Data$                           Send data from computer back to spectrum
                                                          analyzer as an indefinite, block-data field. The
                                                          "I#" indicates that ASCII code 10 (line feed)
                                                          marks the end of the transmission of data.
```

230 OUTPUT 723;"TDF P;TRA[50]?;"	*Query trace data as decimal values in absolute units (dBm). Query only the 50th element.*
240 ENTER 723;N	*Assign data to variable.*
250 DISP N	*Print value.*

Description The TDF command formats trace information for return to the controller.

TDF P (parameter) gives data as ASCII-decimal values in absolute units (dBm). Trace data values are separated by a "," character.

TDF M (measurement) gives data as ASCII-decimal values in measurement units. Trace data values are separated by a "," character.

TDF A gives data as an absolute block data field. The MDS command determines whether data comprises one 8-bit byte or a 16-bit word.

TDF I gives data as an indefinite block data field. The MDS command determines whether data comprises one 8-bit byte or a 16-bit word.

TDF B gives data as binary values in measurement units. The MDS command determines whether data comprises one 8-bit byte or a 16-bit word.

The following tables summarize how the TDF command affects trace data transmission.

HP-IB Transmission Sequence for Different Trace-Data Formats (TDF)

	TDF P	TDF M	TDF B	TDF A	TDF I
Byte 1	1	1	|3|	#	#
Byte 2	0	0	|232-EOI|	|A|	|I|
Byte 3	.	0		|0|	|3|
Byte 4	0	0		|2|	|232-EOI|
Byte 5	0	10 |EOI|		|3|	
Byte 6	10|EOI|			|232|	
				|10-EOI|	

Trace Data Output Formats for Amplitude Information

Format Command	Spectrum Analyzer Output	Trace Element Output Amplitude of +10 dBm
TDF P	Gives ASCII decimal value. Values are in parameter units. Data output is followed by line feed |ASCII code 10| and end-or-identify |EOI|. Trace points are separated by a comma character.	+10.00
TDF M	Gives ASCII integer value. Values are in measurement units. Line feed	ASCII code 10| and EOI follow data output.	+1000
TDF B	Gives binary value as a 16-bit word, with most significant byte first. Values are in measurement units. Sends EOI with last byte.	|3| |232|
TDF A	Sends series of 8-bit bytes |A-block data field|. Data is preceded by "#" and "A" and 2 bytes that give the length of the data field. Data followed by line feed |ASCII code 10| and EOI. Values are in measurement units.	|#||A| |0||2| |3||232|
TDF I	Sends a series of 8-bit bytes |I-block data field|. Data is preceded by "#" and "I." Values are in measurement units. Sends EOI with last byte.	|#||I| |3| |232|

Query Response
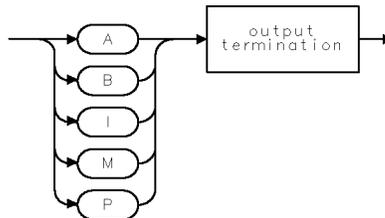
QTDF

See Also TRA, TRB, TRC, MDS

TEST   Perform analyzer self-test.
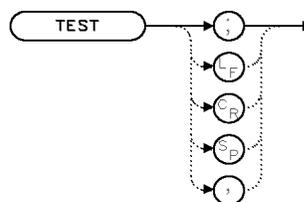
XTEST

Example   10 OUTPUT 723;"TEST;"   *Conduct the self test.*

Description The TEST command tests spectrum-analyzer operation, then reports errors to the error register.

The optical spectrum analyzer returns to its previous state when the test is complete. Execute **ERR?** to query error codes to the controller. (**IP** and **ERR?** clear the error register where errors are recorded.) The system-level command, device-clear, stops the test. The device-clear command in HP BASIC is **CLEAR 723** for optical spectrum analyzers at HP-IB address 23 on interface 7.

TEXT

Write message anywhere on spectrum-analyzer display.

Example
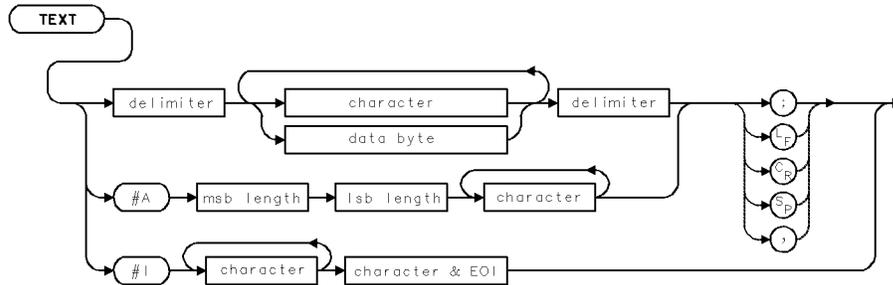
130	CLEAR 723	*Initialize spectrum analyzer.*
140	OUTPUT 723;"IP;"	
150	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
160	OUTPUT 723;"PU;PA 200,800;"	*In the up position, move the pen to (200,800).*
170	OUTPUT 723;"TEXT$";	*Write text. The "$" is a delimiter which marks the beginning and end of the text.*
180	OUTPUT 723;"Text can be ";	
190	OUTPUT 723;"positioned on the ";	
200	OUTPUT 723;CHR$(10);CHR$(13);	*The ASCII codes for carriage return and line feed move the pen to the next available line.*
210	OUTPUT 723;"display with the ";	
220	OUTPUT 723;"PA and PR ";	
230	OUTPUT 723;CHR$(10);CHR$(13);	
240	OUTPUT 723;"commands.$;"	
250	OUTPUT 723;"IT 1;VW OFF;"	*Address item 1 and blank it.*
260	OUTPUT 723;"TEXT$Text can be put into items$;"	*Assign text to item 1.*
270	OUTPUT 723;"OR 400,600;VW ON;"	*Position the item at (400,600) and display it.*

Description

The TEXT command writes text on the spectrum-analyzer display. Place identical delimiters at the beginning and end of the text you wish to display. Text can be positioned two ways. First, PA or PR can position the text. Since TEXT automatically lowers the pen, PD (pen down) is not needed. After TEXT is executed, the pen position remains at the end of the last letter drawn, in the up position. The PA and PR commands should be used to position large paragraphs, such as help screens, that do not require modification at a later time.

Second, text can be positioned with the OR command during the reference-graphics mode. During this mode, text is assigned a nonzero item number with the IT command. Each item can contain up to 35 characters and up to 50 item numbers are available. Once assigned an item number, text can be modified at a later time or enhanced (blinking, inverse video, etc.). TEXT does not function when a **graphics window** is on (`DWINDOW ON`).

See Also DWINDOW, IT

TH

Blank display of trace data below an amplitude level.
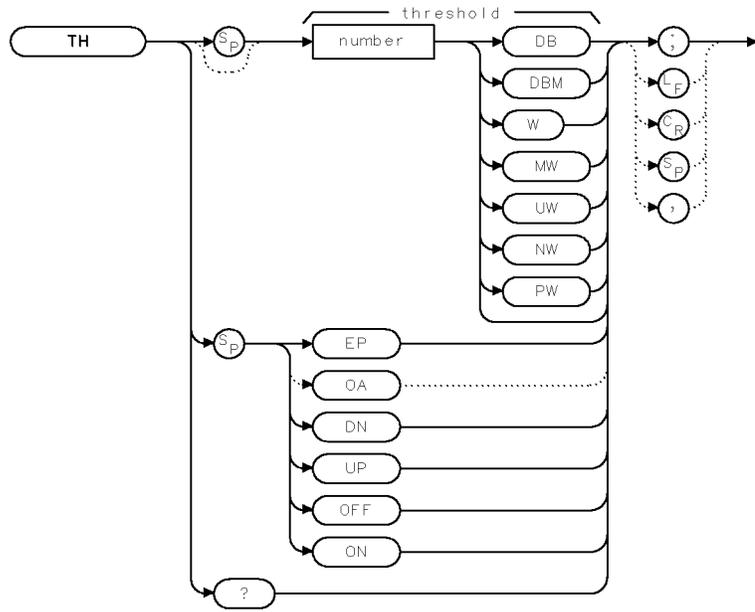
Item	Description
Threshold Range	± 300 dBm
Step Increment	1 vertical scale division
Preset State	TH -90DBM (or one major division above the bottom graticule line)

Example   140   OUTPUT 723;"TH -70DBM;"

Description The TH command blanks signal responses below the specified threshold level (similar to "base-line clipper" operation). The responses below the threshold value are blanked when a value is entered with TH, or when `TH ON`, `TH UP`, or `TH DN` is executed. The TH command does not affect the amplitude information contained in the trace array.

Query Response
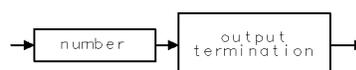
qgennum

See Also PDA, TRPST, AUNITS, RL, MEASURE, CNVLOSS, AMPCOR

THREED

Turn on or off the three-dimensional trace display capability.

XTHRD
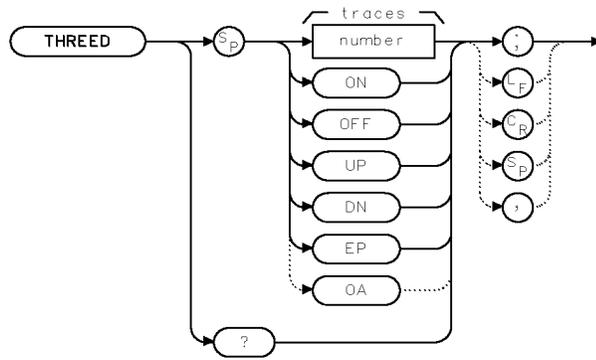
Item	Description
Preset State	OFF
ON	Default is 16 Traces
Range	1 to 50 traces

Example

```
10  OUTPUT 723;"THREED ON;"      Turn on three-dimensional capability.
20  OUTPUT 723;"THREED UP;"      Increase traces displayed by one increment.
30  OUTPUT 723;"THREED?;"        How many traces are creating the three-dimensional
                                 display.
40  ENTER 723;N
```

Description

The THREED command sets the number of previous traces to be displayed simultaneously as a three-dimensional display function. The function is activated when you enter a value between 1 and 50, or when you turn the function on. This value is retained if you turn the function off. It is not changed until you enter a different value.

Query Response

```
→  | number | →  | output
                   termination | →
```
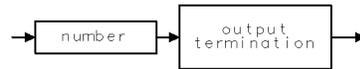
qgennum

See Also THREEDV, THREEDH

THREEDH

Select the horizontal offset of the three-dimensional trace display.
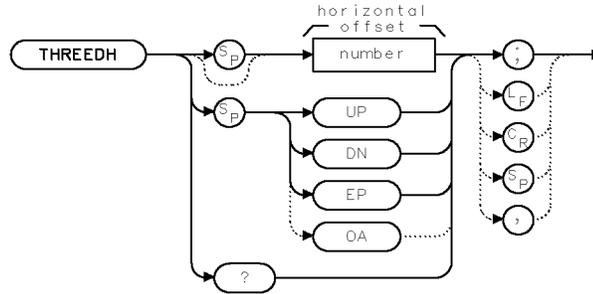
Item	Description
Preset State	4 positions, moving toward the right-hand side of the display
Range	0 to ±200 positions
Step Increment	1 position

Example

10	OUTPUT 723;"THREED ON;"	*Turn on three-dimensional capability.*
20	OUTPUT 723;"THREEDH −10;"	*Rotate the three-dimensional display 10 positions left, from its current position.*
30	OUTPUT 723;"THREEDH?;"	*What is the current horizontal position of the three-dimensional traces displayed.*
40	ENTER 723;N	

Description The THREEDH command allows you to change the horizontal position of the traces, left or right, of a three-dimensional trace display. The function is activated with the THREED command, plus a value between 0 and ±200. The horizontal position of the traces is retained if you turn the function off with the THREED OFF command. It is not changed until you enter a new value.

Query Response
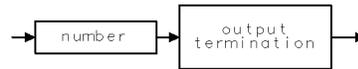
qgennum .

See Also THREED, THREEDV

THREEDV

Select the number of traces to create the three-dimensional trace display.
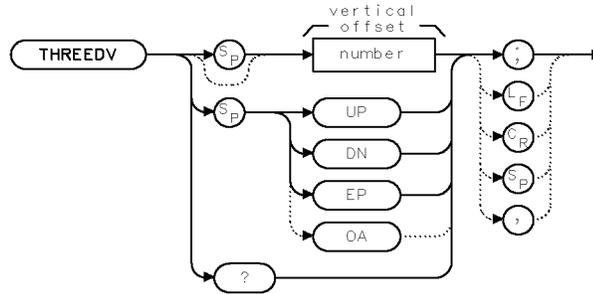
xthreedv.

Item	Description
Preset State	10 positions moving toward the top of the display
Range	0 to ±200 positions
Step Increment	1 position

Example

```
10  OUTPUT 723;"THREED ON;"      Turn on three-dimensional capability.
20  OUTPUT 723;"THREEDV 20;"      Rotate the three-dimensional display 20 positions
                                  up from its current position.
30  OUTPUT 723;"THREEDV?;"        What is the current vertical position of the three-
                                  dimensional display.
40  ENTER 723;N
```

Description The THREEDV command allows you to change the vertical position of the traces, up or down, from the current position of a three-dimensional trace display. The function is activated with the THREED command, plus a value between 0 and ±200 positions. The vertical position of the traces is retained if you turn the function off with the THREED OFF command. It is not changed until you enter a new value.

Query Response

```
   ┌──────────┐   ┌──────────────┐
──→│  number  │──→│    output    │──→
   └──────────┘   │ termination  │
                  └──────────────┘
```
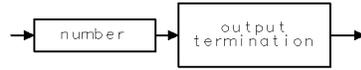
qgennum

See Also   THREED, THREEDH

TIME

Monitor time.
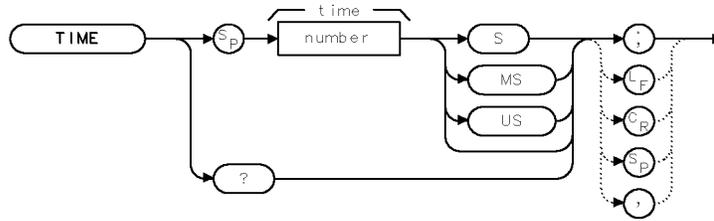
xtime

Item	Description
Time Range	$\pm 10^{12}$
Clock Accuracy	1 μs

Example

10	OUTPUT 723;"TIME 0S;"	*Set clock to zero.*
20	OUTPUT 723;"WAIT 3;"	*Wait three seconds.*
30	OUTPUT 723;"TIME?;"	*Query elapsed time to controller.*
40	ENTER 723;N	*Assign value to controller variable.*

Description The TIME command sets the clock that measures operating time. Clock precision is 1 μs. Execute **TIME?** to query the time lapsed since the clock was last set. The clocked time is annotated beneath "TIME STAMP" in the active-function readout until **IP** or another spectrum-analyzer command changes the active function readout. IP does not change the value of TIME. Time can be used to record the time of an occurrence, especially if the display is monitored with a camera.

Query Response
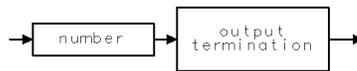
qgennum

TITLE

Write a title in the title line.
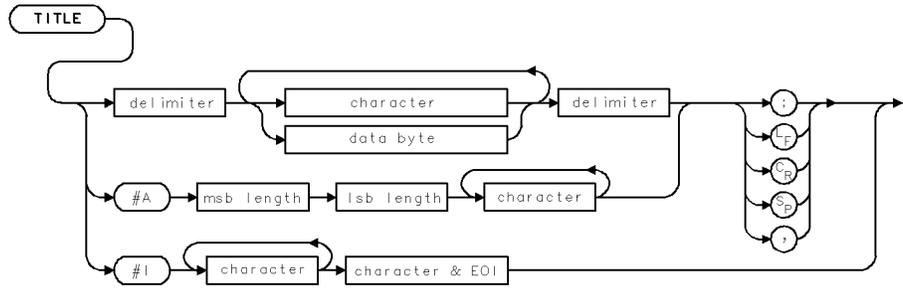
Item	Description
Number of characters	0 – 36
Preset State	Blank title

Example

130	OUTPUT 723;"CLRDSP;"	*Clear graphics from memory.*
140	OUTPUT 723;"TITLE$Title your data$;"	*Write text in the title line. The "$" delimits text.*
150	OUTPUT 723;"WAIT 2S;"	*Wait 2 seconds.*
160	OUTPUT 723;"TITLE$";	*Write text. Semicolon required to suppress line feed.*
170	OUTPUT 723;CHR$(27);"&dB";	*Turn on inverse-video display of text.*
180	OUTPUT 723;"^Your title^";	
190	OUTPUT 723;CHR$(27);"&d@";	*Turn off inverse video.*
200	OUTPUT 723;"$;"	*End of second title.*

Description   The TITLE command displays text in the title line located in the upper
portion of the graticule. The TITLE command automatically places the
pen down before text is written and raises it after text is written. Place
identical delimiters at the beginning and end of the text you wish to display.
The maximum number of text characters allowed is 36. Standard escape
sequences may be used to produce enhancements such as blinking, inverse
video, and underlining. The example shows the command to turn the inverse
video on and off. CLRDSP does not affect the contents of the title line. The
SAVES command stores the contents of the title line in a state register.

See Also   TEXT, SAVES

TM

Determine what triggers the sweep (the sweep-trigger mode).

x tm

Item	Description
Preset State	TM FREE

Example

```
10 OUTPUT 723;"TM EXT;"
```

Description The TM command selects the trigger mode that determines how the sweep is triggered.

TM FREE (free run) uses the internal circuitry of the spectrum analyzer so that the trigger conditions are satisfied.

TM VID (video) triggers the sweep with the amplitude level of the detected, video signal.

TM LINE (line) triggers the sweep with the line voltage.

TM EXT (external) triggers the sweep with an external trigger connected to the external-trigger input port.

Query Response

QTM

See Also VTH, VTL, DONE, CONTS, SNGLS, VAVG

TP Modify shape of stored, graphics item.
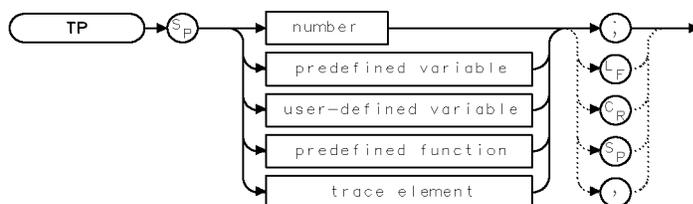
Item	Description
Point Range	0 – 65,535
Preset State	TP 0

Example

160	`OUTPUT 723;"CLRDSP;"`	*Clear graphics from memory.*
170	`OUTPUT 723;"DWINDOW 200,200,0,27,0,30;"`	*Define new coordinate system that fits inside a square with sides 200 units long, in the current units for SCALE (0 to 1000 for both X and Y). Define new coordinate system: X1 axis is from 0 to 27; Y1 axis from 0 to 30.*
180	`OUTPUT 723;"IT 1;VW OFF;"`	*Address item 1 and blank it.*
190	`OUTPUT 723;"PA 10,0,17,8,";`	*Define item 1 with eleven*
200	`OUTPUT 723;"25,4,21,13,27,20,";`	*coordinate pairs, numbered*
210	`OUTPUT 723;"18,19,12,30,";`	*0 through 10. The coordinates*
220	`OUTPUT 723;"11,18,0,16,";`	*are specified in the new*
230	`OUTPUT 723;"11,11,10,0;"`	*coordinate system (X1 and Y1).*
240	`!`	
250	`OUTPUT 723;"OR 500,600;VW ON;"`	*Position item at (500,600) and turn it on.*
260	`OUTPUT 723;"WAIT 2S;"`	*Wait 2 seconds.*
270	`OUTPUT 723;"TP 4;"`	*Address the fourth point (27,20).*
280	`OUTPUT 723;"PA 40,30;"`	*Change it from (27,20) to (40,30).*

Description The TP command works in conjunction with the item and plot commands,
IT and PA, to modify the shape of referenced graphics called items. The TP
command acts as a pointer that selects a point on an item for modification.
For example, the triangle below is a graphics item defined by points,
numbered 0 through 3.

#0	#1	#2	#3
|600,0|	|700,200|	|800,0|	|600,0|

Executing **TP 1** selects point (700,200) for modification. Follow **TP 1** with a
PA command to specify the new point coordinates.

See Also IT

TRA/TRB/TRC Return or store trace data.
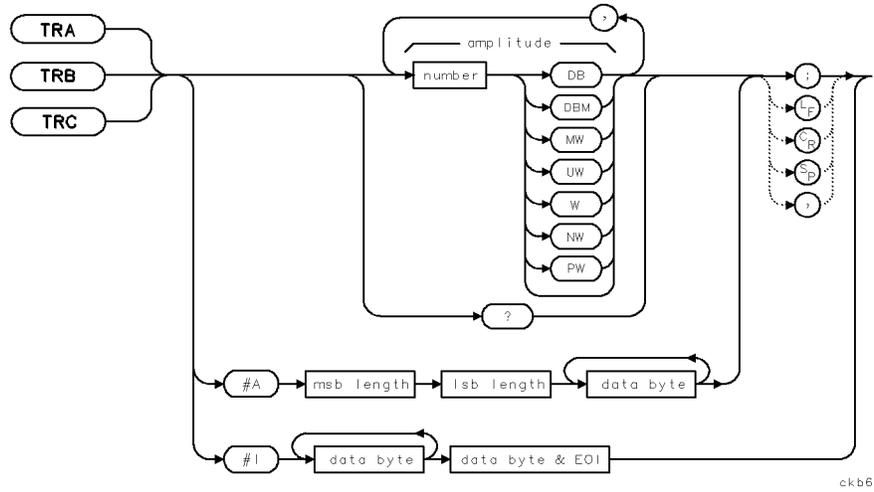
Syntax to Store or Query Array
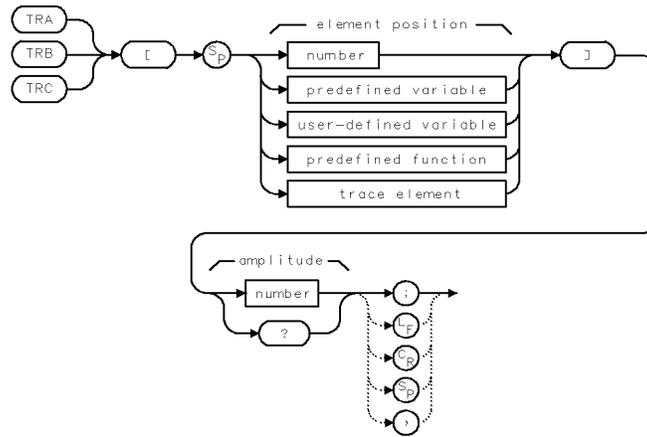
Syntax to Store or Query One Element

Item	Description
Amplitude Range	0 - 10,000 measurement units for linear trace data.
	-32,768 - +32,767 measurement units for logarithmic trace data.
Length Range	3 - 2048 position units. Default is 800 position units when using TRA, TRB, TRC.

Example Use TRC to store data in a trace.

```
10   OUTPUT 723;"BLANK TRA;"
20   OUTPUT 723;"TRDEF TRC,3;"

30  OUTPUT 723;"TRC -10DBM,-20DBM,-15DBM;"

40  OUTPUT 723;"VIEW TRC;"
```

Blank trace A.
Define trace C as a three-element trace.
Store amplitude values in trace C.
View trace C.

Transmit trace data to the computer.

```
10   OUTPUT 723;"TDF A;"


20   OUTPUT 723;"TS;"
30   OUTPUT 723;"VIEW TRA;"

40   OUTPUT 723;"TRA?;"
```

When requested, send data as an absolute, block-data field. (Trace data is preceded by #A.)
Sweep trace A.
Stop sweeping trace A and store measurement results.
Query trace-A amplitude information.

```
50   ENTER 723 USING "#,AA,W";Pound_a$,Number_of_bytes
```
Format computer to receive data until the variable list is satisfied. "AA" expects two characters (#A) and W reads data as a 16-bit word (two 8-bit bytes).
```
60   ALLOCATE Data$[Number_of_bytes]
```
Define a computer array for storing the trace information. The length of the computer array is as long as the trace.

```
70   ENTER 723 USING "-K";Data$
80   OUTPUT 723;"MOV TRA,0;"
90   OUTPUT 723;"TRA#I";Data$
```

Store data in the computer array.
Store zeroes in trace A.
Send data from computer back to spectrum analyzer as an indefinite block data field. The "#I" indicates that ASCII code 10 (line feed) marks the end of the transmission of data.

```
100   OUTPUT 723;"TDF P;TRA[50]?;"



110   ENTER 723;N
120   DISP N
```

Query trace data as decimal values in absolute units (dBm). Query only the fiftieth element.
Assign data to variable.
Print value.

Description   The TRA, TRB, and TRC commands send or receive trace-amplitude
information for trace A, B, or C. Trace A is described below. Traces B and C
operate the same way. To store an amplitude value in one or more points of
a trace, follow TRA by amplitude values for each point. Values are stored
starting with the first point.

When *number* (refer to the syntax diagram) is used to store trace data (for
example, `TRA -20 DBM,-21DBM`), the units for stored data can be specified
as shown in the syntax diagram (such as, dB, dBm), provided that the
trace data format is in its preset mode, `TDF P`. If `TDF B, TDF A, TDF I,` or
`TDF M` has been executed, numbers are stored in **measurement units.** If
relative-amplitude mode is active, stored data is in dB.

When #A or #I (block data fields) store trace data, the units for the stored
data are **measurement units**, and are interpreted as bytes or words, as
specified by the MDS command. When TRA is used to store block data in a
trace, TRA overrides any format selection made with the TDF command.

To query the value of a single trace element, follow TRA by the trace element
position, in position units, and a question mark. The value queried is in
parameter units, unless the trace-data-format command, TDF, specifies
measurement units. The amplitude-units command, AUNITS, can also specify
the type of parameter units queried.
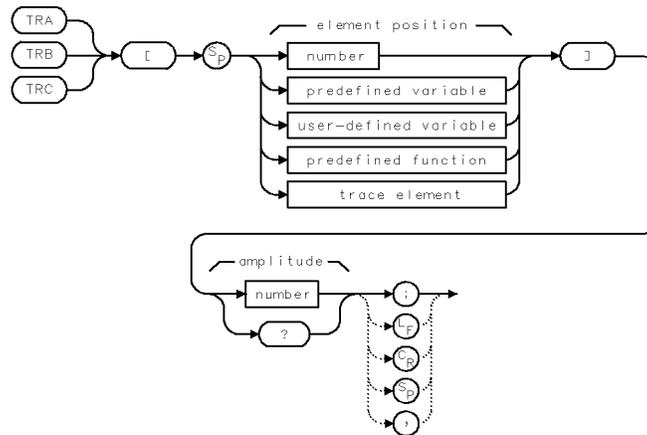
ckb20

Data in ASCII Format (TDF P or TDF M)

Query Response
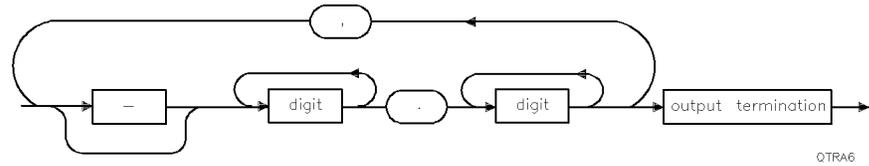
Data in ASCII Format (TDF P or TDF M)
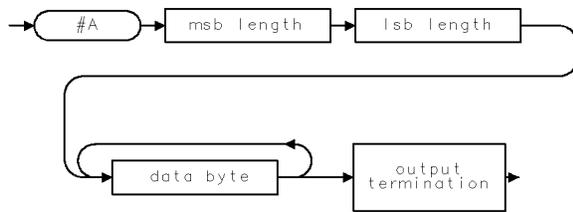
Data in Absolute, Block-Data Format (TDF A)
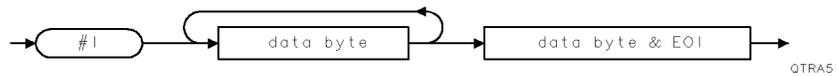
Data in Indefinite, Block-Data Format (TDF I)
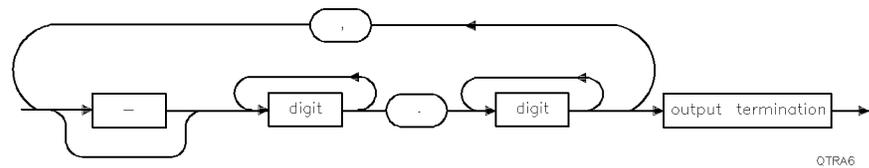
Data in Binary Format (TDF B)

See Also MDS, TDF, and commands that affect wavelength or amplitude range.

TRCOND

Scale trace data displayed in a graphics window.

xtrcond
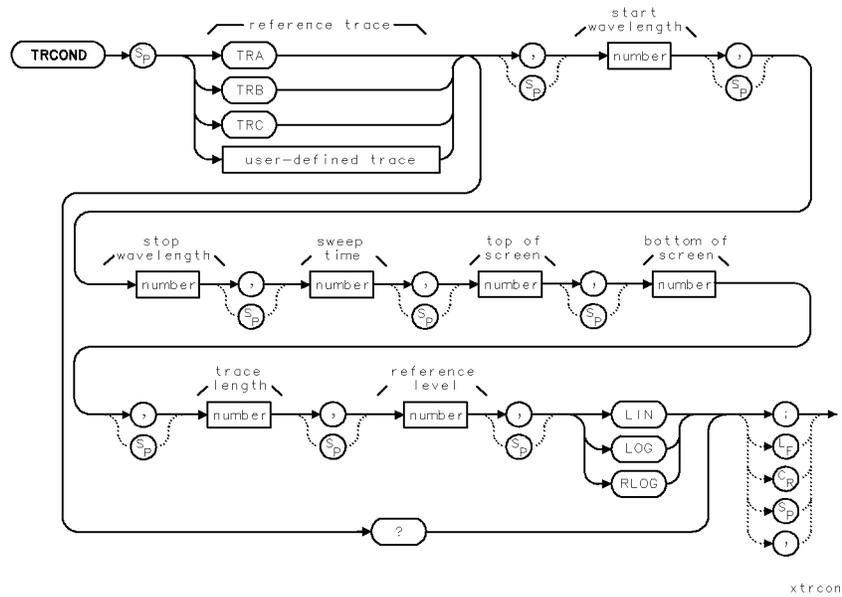
Item	Description
Start-Wavelength Range	$\pm$ 1000 m or hardware limit.
Stop-Wavelength Range	$\pm$ 1000 m or hardware limit.
Sweep-Time Range	0 – 1000 seconds or hardware limit.
Top-of-Screen Range	$-32,768 - +32,767$
Bottom-of-Screen Range	$-32,768 - +32,767$
Trace-Length Range	3 – 2048
Reference-Level Range	$\pm$ 300 dBm or hardware limit.
Reference-Trace-Length Range	3 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.
Reference-Trace Amplitude Range	0 – 10,000 measurement units for linear trace data.
	$-32,768 - +32,767$ measurement units for logarithmic trace data.

Example

140   ALLOCATE Disp_mode$[3]
Allocate string variable in computer for storing conditions.
150   OUTPUT 723;"TRCOND TRA?;"
Query trace conditions to computer.
160   ENTER 723 USING "#,K";Start,Stop,Sweep
Return data.
170   ENTER 723 USING "#,K";Top,Bottom,Data

180   ENTER 723 USING "#,K";Reference,Disp_mode$

190   DISP Bottom;"MEASUREMENT UNITS"
Display minimum trace value.

Description   The TRCOND command formats the spectrum-analyzer settings so that measurement results stored in trace arrays are properly displayed in a graphics window defined by the DWINDOW command.

Set the *top-of-screen* parameter, in measurement units, equal to the maximum amplitude you want to display.

Set the *bottom-of-screen* parameter, in measurement units, equal to the lowest value you want to display.

Set the *trace-length* parameter equal to the number of elements in the trace.

Set the *reference-level* parameter, in dBm, equal to the reference level active when trace data was stored.

Set the *start-* and *stop-wavelength* parameters, in meters, equal to the start and stop frequencies that were active when the trace was stored (swept).

Set the *sweep-time* parameter, in seconds, equal to the sweep time that was active when the trace was stored (swept).

Command arguments LINPWR, LOG, and RLOG determine how data is interpreted. LINPWR interprets data in an absolute, linear power scale. LOG interprets data in an absolute, logarithmic scale. RLOG interprets data in a relative, logarithmic scale.

Query Response
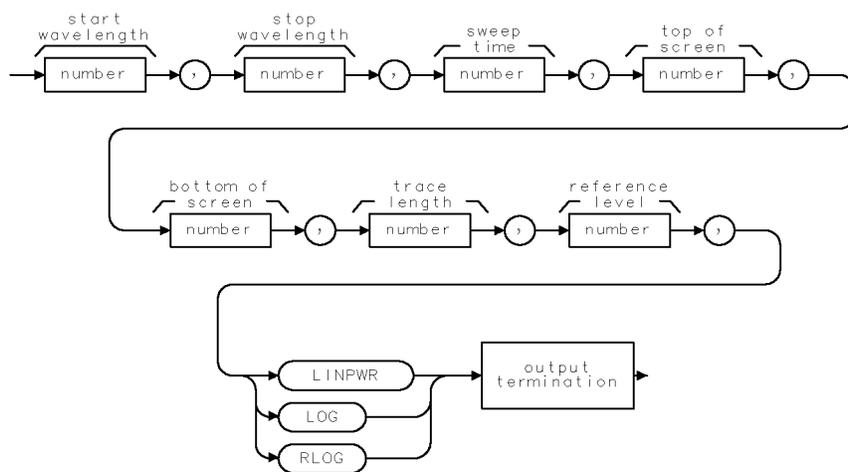
QTRCON

See Also DWINDOW, GRAPH

TRDEF

Change trace length or create user-defined traces.
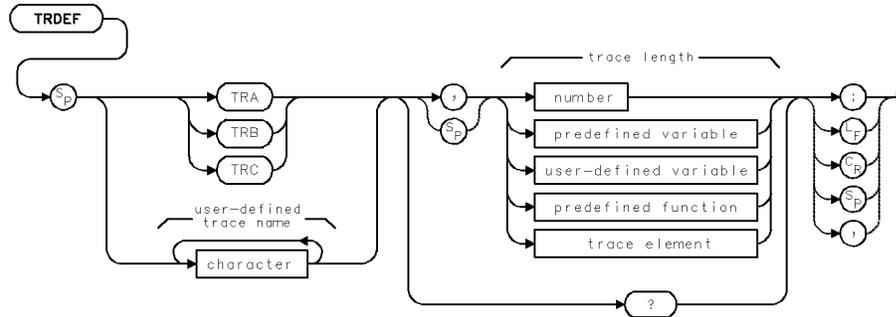
Item	Description
Trace-Name Range	1 – 12 characters: A – Z, 0 – 9, and underscore |_|. First character must be a letter. Numbers must be preceded by underscore.
Length Range	3 to 2048
Preset State	`TRDEF TRA,800; TRDEF TRB,800; TRDEF TRC,800`

Example

10	`OUTPUT 723;"TRDEF TRA,400;"`	*Change the length of trace A to 400 elements.*
20	`OUTPUT 723;"TRDEF MLB,3;"`	*Define user-defined trace that is 3 points long.*
30	`OUTPUT 723;"TRDEF MLB,?;"`	*Query length of user-defined trace to computer.*
40	`ENTER 723;N`	*Assign value (3) to computer variable.*
50	`OUTPUT 723;"TS;MOV MLB,TRA;"`	*Store contents of TRA in MLB.*
60	`OUTPUT 723;"MLB?;"`	*Query contents of MLB to computer.*
70	`ENTER 723;A,B,C`	*Assign each trace element to a computer variable.*

Description   The TRDEF command changes the length of traces A, B, or C, or creates new trace arrays, called user-defined traces, for data storage or manipulation. After a user-defined trace is created, it can be used wherever *user-defined trace* appears in the syntax diagrams.

Memory is reserved to contain up to 2048 elements of trace information (from 1 to 2048) for each of the predefined traces A, B, and C. This memory is always protected and is separate from internal memory where user-defined traces, variables, functions, etc., are stored. TRDEF stores user-defined traces in internal memory provided there is sufficient room. Use DISPOSE or PURGE to delete user-defined traces from internal memory. If traces A, B, and C have different lengths, the active trace with the shortest length is used for the specified span. The shorter trace accepts data until filled.

Query Syntax
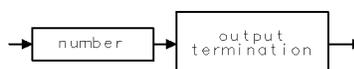
qgennum .

See Also   DISPOSE, ERASE, FORMAT, LOAD, PROTECT, RCLD, PURGE, TRA, TDF

TRDSP

Blank display of active trace(s) while measuring.
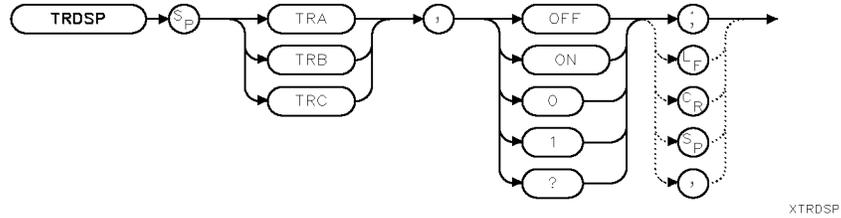
XTRDSP

Item	Description
Preset State	TRDSP TRA,ON; TRDSP TRB,OFF; TRDSP TRC,OFF

Example

30	OUTPUT 723;"CENTERWL 750NM;SP 100NM;"	*Set measurement range.*
40	OUTPUT 723;"TS;"	*Sweep trace A.*
50	OUTPUT 723;"TRDSP TRA,0;"	*Turn off display of trace A.*
60	OUTPUT 723;"TRDSP TRA,ON;"	*Display trace A.*
70	OUTPUT 723;"TRDSP TRA,?;"	*Query display-status of trace A.*

Description The TRDSP command turns off trace A, B, or C without clearing the trace data or interrupting measurements. Thus, data continues to be stored in the active trace. The **OFF** argument reduces the time duration between sweeps.

Query Response

qgen0

See Also TRPST, TRSTAT

7-508
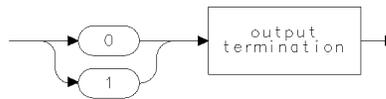
TRNSZLOCK

Locks transimpedance amplifier for fast pulse measurements.
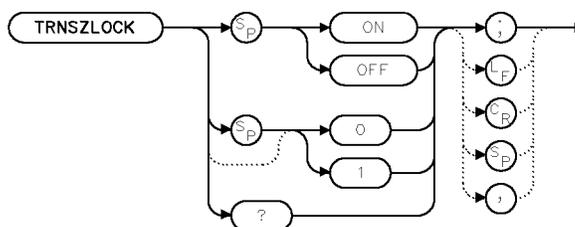
xtnszl

Item	Description
Preset State	off

Example   `OUTPUT 723;"TRNSZLOCK;"`

Description The TRNSZLOCK command locks the optical spectrum analyzer's internal transimpedance amplifier.

Locking the transimpedance amplifier limits its range to ensure wide bandwidth and fast settling characteristics. This allows the optical spectrum analyzer to make accurate fast pulse measurements. Using the `INSTMODE PULSE` command automatically locks the transimpedance amplifier.

Normally, the transimpedance amplifier is unlocked, and the full transimpedance range of 1 kΩ to 100 MΩ is available. However, when locking is selected, the amplifier is limited to the 1 kΩ and 10 kΩ values.

Query Response
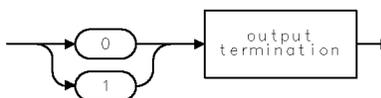
qtnszl

TRPST   Preset active trace conditions.
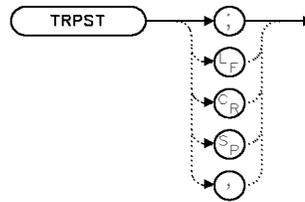
XTRPST

Example   110   OUTPUT 723;"TRPST;"   *Reset the traces to a defined state.*

Description Use the TRPST command to reset the traces to a defined state without affecting the measurement parameters. The TRPST command executes the following commands to preset trace functions:

- Clear-write trace A. (**CLRW TRA**)
- Blank trace B. (**BLANK TRB**)
- Blank trace C. (**BLANK TRC**)
- Turn off trace-A-minus-trace-B. (**AMB OFF**)
- Turn off trace-A-minus-trace-B-plus-display-line. (**AMBPL OFF**)
- Turn off video averaging. (**VAVG OFF**)
- Set trace A length to 800 elements. (**TRDEF TRA,800**)
- Set trace B length to 800 elements. (**TRDEF TRB,800**)
- Set trace C length to 800 elements. (**TRDEF TRC, 800**)
- Display trace A. (**TRDSP TRA,ON;TRDSP TRB,OFF;TRDSP TRC,OFF**)

See Also CLRW, BLANK, AMB, VIEW, AMBPL, VAVG, TRDEF, TRDSP, TRSTAT

TRSTAT   Return the status of the active trace.
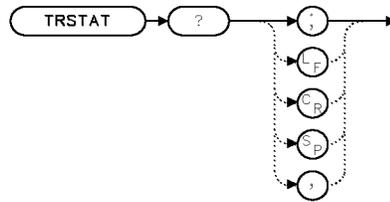
XTRSTA

Example   10 ALLOCATE Status$[40]   *Define computer array for storing characters.*
             20 OUTPUT 723;"TRSTAT?;"   *Query trace states.*
             30 ENTER 723 USING "-K";States$ *Store response in States$, including line*
                                                 feeds.

Description The TRSTAT query returns the status of traces A, B, and C. The status can be clear-write, blank, max-hold, min-hold, on (view mode), or off.

Query Response In a query response, trace A is described first, followed by trace B and trace C. *Writing status* describes how the trace data is updated with each sweep. *Display status* tells whether the trace is displayed on the display.
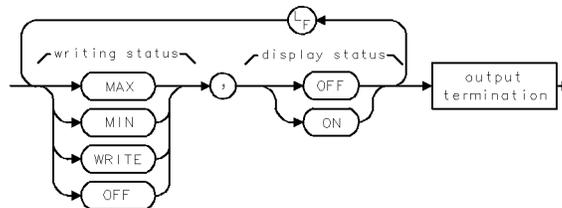
- MAX is queried when trace data is updated using positive-peak detection.
- WRITE is queried when the trace is updated with normal detection.
- ON is queried when the trace is displayed.
- OFF is queried for both writing and display status when the trace is not updated with each sweep.

See Also CLRW, VIEW, BLANK, TRDSP, TRPST

TS Sweep the current measurement range.
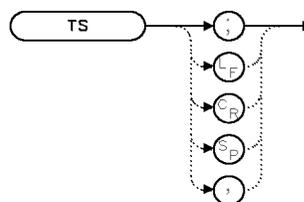
XTS

Example `10  OUTPUT 723;"MKN;TS;MKMIN;"`

Description The TS command sweeps the current measurement range and stores new amplitude information in the active trace(s).

Use TS for the following reasons:

- Insure trace reflects current measurement range settings.
- Capture traces before processing them with math commands.
- Capture traces before positioning markers.

Each sweep begins when the trigger conditions are met. The take-sweep command is used to synchronize a measurement with the sequence of commands sent to the instrument. When a command sequence is preceded by TS, the user is assured a complete sweep of measurement information has been taken and the end-of-sweep bit in the status-byte register has been set before further commands are processed.

The TS command does not set the command complete bit or the end-of-sweep bit in the status register until all of the sweeps specified by VAVG (video averaging) are completed.

Since there is an input buffer for commands sent via HP-IB, multiple commands may be stored in the buffer, but they are not executed until the sweep is completed. To synchronize computer operation with the completion of a measurement, follow TS with a `DONE?`.

See Also TM, VTL, VAVG, VTH, SNGLS, CONTS, DONE, FETCH

TWNDOW

Format trace array for use with fast Fourier transform.
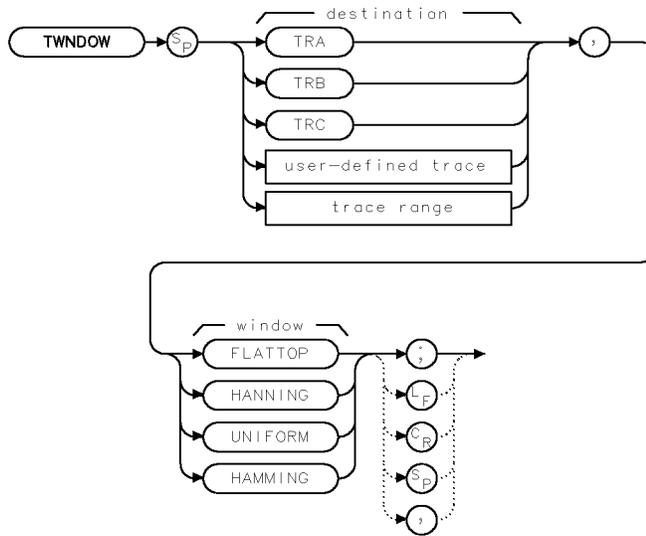
Item	Description
Destination-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	—32,768 – +32,767 measurement units for logarithmic trace data.
Destination-Length Range	3 – 2048 position units.

Example

170	OUTPUT 723;"CENTERWL 750NM;"	*Set measurement range.*
180	OUTPUT 723;"SP 0;ST 50MS;"	
190	OUTPUT 723;"TWNDOW TRC,FLATTOP;"	*Store flat-top-window algorithm in trace C.*
200	OUTPUT 723;"CLRW TRB;BLANK TRA;"	*Measure with trace B. Blank trace A.*
210	OUTPUT 723;"SNGLS;TS;"	*Sweep once.*
220	OUTPUT 723;"FFT TRA,TRB,TRC;"	*Perform fast-Fourier transform on trace B and store the results in trace A.*
230	OUTPUT 723;"BLANK TRB;VIEW TRA;"	*Blank trace B. Store and view trace A.*

Description The TWNDOW function formats a trace array with real values for use with the fast Fourier transform function (FFT). Execute TWNDOW on a trace that has the same length as the trace you wish to transform with FFT.

The window function weights the contents of a trace array according to four algorithms, called trace windows: uniform, Hanning, Hamming, or flat-top. Three of the filters are shown below as graphs in the time domain. The TWNDOW command multiplies a trace array with one of these windows. If the predefined windows do not meet your measurement needs, use the TRA, TRB, TRC commands to build your own window function.
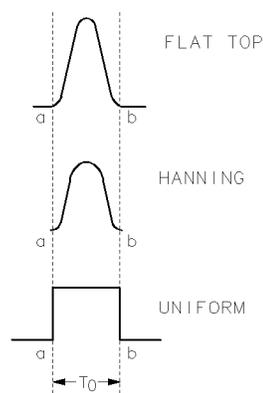
Hanning, Flat top, and Uniform Trace Window Shapes

The predefined windows simulate passband shapes that represent a
give-and-take between amplitude uncertainty, sensitivity, and wavelength
resolution. They also minimize side effects caused by the non-continuous,
finite nature of the discrete Fourier transform. One side effect is the
appearance of aliasing. This is handled by rapid sampling and signal
filtering. Another side effect is spurious responses caused by sampling the
measurement range for a finite period. This is minimized by weighting the
sampled data, but this reduces the resolution of real signals. The compromise
of how much to reduce side lobes at the expense of resolution is the purpose
of the window choices.

The uniform-window algorithm has the least wavelength uncertainty and
greatest amplitude uncertainty. Worst-case accuracy uncertainty is -3.9 dB
and its 3 dB resolution bandwidth is 60% of the Hanning bandwidth.
The uniform window does not contain time-domain weighting. Thus, the
amplitude data is unchanged. Use the uniform window when transforming
noise signals or transients that decay within one sweep-time period. The
uniform window also yields the best wavelength resolution, but also produces
the highest side lobes for periodic signals.

The Hanning window is a traditional passband window found in most
real-time spectrum analyzers. The Hanning window offers a compromise
between the flat-top and uniform windows. Its amplitude uncertainty is
-1.5 dB and its 3-dB bandwidth is 40% of the flat-top bandwidth. Use the
Hanning window when transforming periodic or random data.

The flat-top window has the greatest wavelength uncertainty of the windows,
but it has outstanding side-lobe suppression and amplitude flatness. Use the
flat-top window to transform periodic signals.

The Hamming window has a bandwidth somewhere between the Hanning and
flat-top windows, with more uniform side-lobe suppression than the Hanning
window, but less suppression than either Hanning or flat-top for signals
farther out of the passbands.

See Also FFT, TWNDOW

USERERR   Create customized error messages and error conditions.

cgz_u01x
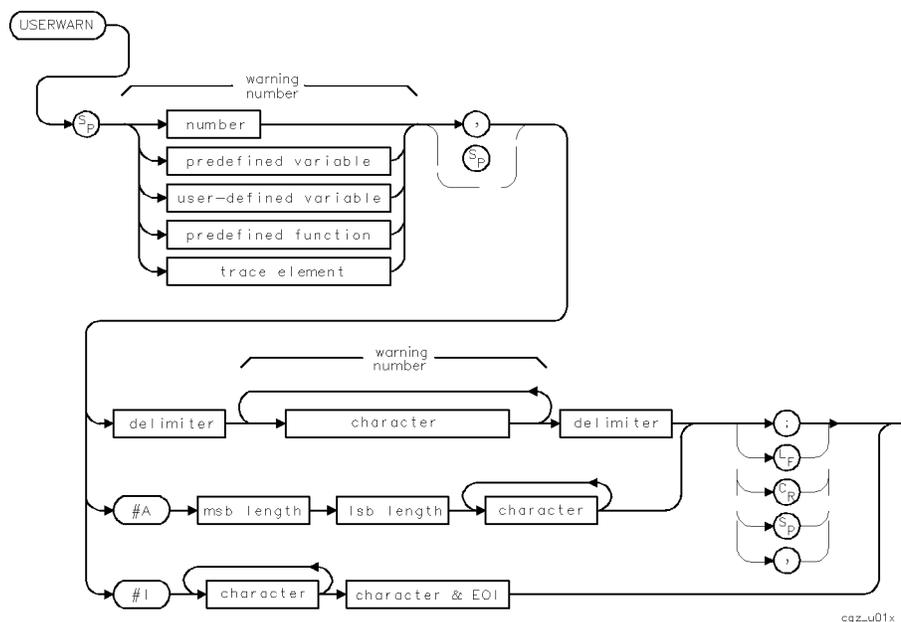
Item	Description
Error-Number Range	1 – 9999
Error-Text Range	0 – 20 characters

Example

150	OUTPUT 723;"USERERR,1234,@MAG@;"	*Report error 1234.*
160	OUTPUT 723;"WAIT 2S;"	*Wait 2 seconds.*
170	OUTPUT 723;"XERR?;"	*Query error report to computer.*

Description   The USERERR command displays an error message of your own creation and assigns the error a number. The error is reported through the error reporting mechanism.

See Also   MSD

USERKEY

Return or receive data that configures (USER) keys.
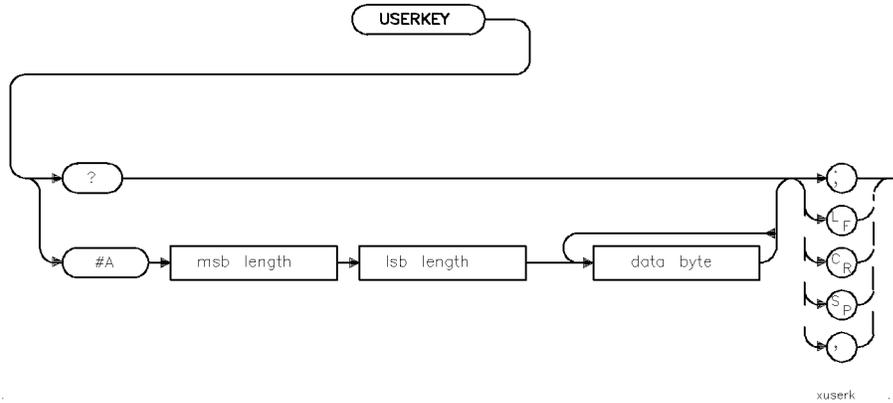
xuserk

Example Transmit the (USER) key menu to and from the computer.

200 OUTPUT 723;"USERKEY?;"	*Return key menu to computer.*
210 ALLOCATE Userkeys$[350]	*Define string variable for storing keys. Up to 350 characters can be stored.*
220 ENTER 723 USING "-K";Userkeys$	*Assign key names to string variable, called Userkeys$. "-K" accepts ASCII characters which might otherwise be interpreted as terminators.*
230 OUTPUT 723;"KEYPST;"	*Set the (USER) menu to its preset state.*
240 P=POS(Userkeys$,"USERKEY")	*Set variable P to the position of the "USERKEY" in the string variable, Userkeys$.*
250 OUTPUT 723;Userkeys$[P];	*Send contents of Userkeys$ to optical spectrum analyzer, starting at character P.*
260 DEALLOCATE Userkeys$	

Description The USERKEY command transmits the key labels accessed with the front-panel (USER) key. Any labels that are associated with custom keypad keys will also be transmitted. Refer to the KEYDEF command for an explanation of defining custom keypad keys.

USERKEY? returns the key labels to the controller.

The block-data field syntax (#A) reloads the key labels. The label names are sent as an absolute, block-data field.

Query Response
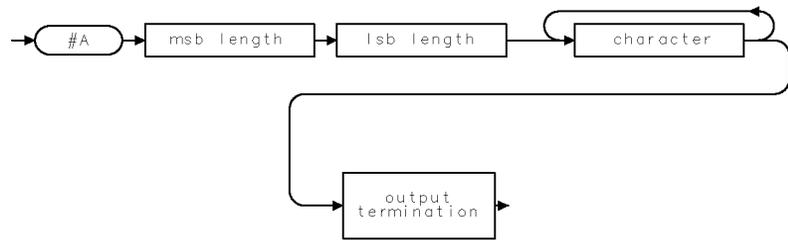
qgablock.

USERLOCK

Limit front-panel operation to the (USER) keys.

```
  ( USERLOCK )──────( Sp )──────( OFF )──────────( ; )──────────────▶
                                ( ON )──────────( LF )
                                ( 0 )───────────( CR )
                                ( 1 )───────────( Sp )
                         ( ? )──────────────────( , )
```
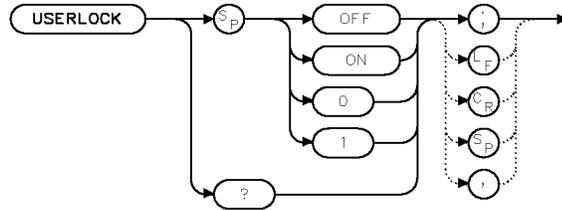
XUSERL

Example

10 OUTPUT 723;"USERLOCK ON;"   *Observe that* (MENU) *keys cannot be accessed*
                                 when (MENU) *is pressed.*
20 OUTPUT 723;"USERLOCK OFF;"   *Make* (MENU) *keys operational.*

Description The USERLOCK command limits front-panel operation to keys accessed with the (USER) key, making the (INSTR PRESET) or (I-P) key, and keys accessed with the (MENU) key, inoperative. Use USERLOCK ON to guarantee that a system operator can not stop or override the functions provided in the (USER) menu. USERLOCK OFF restores operation of the (MENU) keys and the (INSTR PRESET) or (I-P) key.

Query Response

```
  ──────( 0 )──────┌─────────────────┐────▶
        ( 1 )      │     output      │
                   │  termination    │
                   └─────────────────┘
```
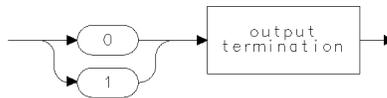
qgen0

USERMSG   Write a message on the spectrum-analyzer display that is cleared at your discretion.
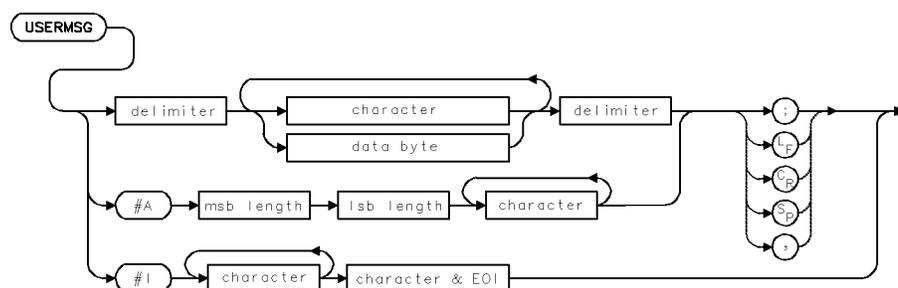
Example   10 OUTPUT 723;"USERMSG @Set center wavelength to 300MHz.@;"

Description The USERMSG command writes a message of 39 characters or less in the message area on the display. The message is cleared by pressing any key, or executing IP or MSG?.

See Also MSG

USERWARN

Display a user defined warning message.

cgz_u01x
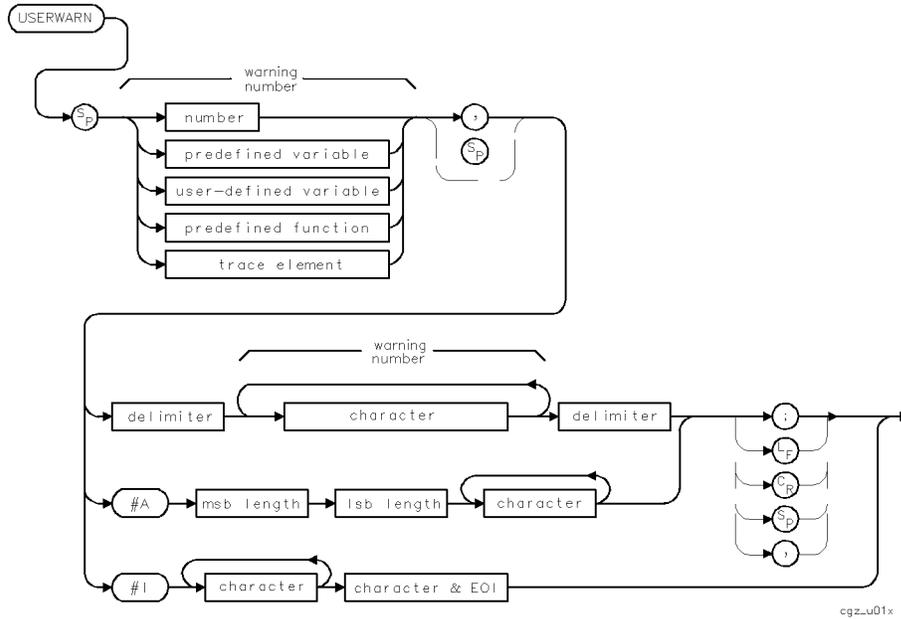
Item	Description
Warning-Number Range	10001—19999
Warning-Text Range	0—20 characters

Example

```
100   OUTPUT 723;";"
100   OUTPUT 723;"USERWARN 17000, ^Connect the input signal!^;"
```

Description The USERWARN command allows you to write your own warning message on the display. Use warnings to indicate that the user might not be performing a task properly. Warnings are displayed in a 2 line by 22 character area that is centered at the bottom of the graticule area. The actual warning text is displayed in inverse video using the ADVISE color; this is the same color that is used for the UNCAL indicator.

When creating a user message, do not use the numbers 10000, or 16001 through 16012; these are reserved warning message numbers. Refer to Chapter 4 of the *HP 71450B/1B/2B Optical Spectrum Analyzer Reference* for definitions of these reserved warnings.

The warning message removed from the display after one of the following events occurs:

- 6 seconds elapse from the time the message is displayed.
- Another warning message is displayed.
- A display key is pressed.
- An WARN or XWARN command is used to read the warning.
- An ERR or XERR command is used to read an error message.

Bit 1 of the status byte is used to report the presence of a warning. Refer to STB and related commands for other information concerning the status byte.

The WARNCTRL command can be used to prevent the display of warnings. An instrument preset (IP command) enables warnings. If display of the active function area is disabled, the warning area will be suppressed; however, the warning can still be accessed via WARN? and XWARN.

See Also STB, USERERR, WARN?, WARNCTRL, XWARN

USTATE

Query or receive data that configures internal memory.
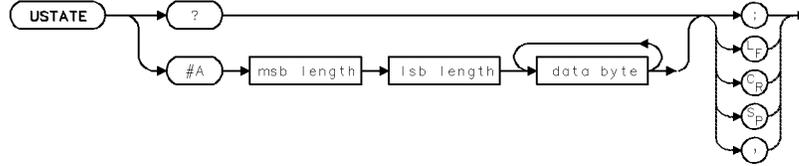
Example   10 OUTPUT 723;"USTATE?;"   *Query user state to computer.*

Description The USTATE command transmits all information that is stored in **internal memory** by the user. This information is called the user state and includes settings determined by the user and unaffected by instrument preset, such as the HP-IB addresses, power-up setting (POWERON), and the protect-state-register setting (PSTATE). Also transmitted are the contents of internal memory, which are listed in the following table.

Catalog of Internal Memory

Stored Article	Storing Commands	File Type	Default File Prefix
User-defined trace	TRDEF, LIMILINE, LOAD, RCLD	TRACE	none
User-defined variable	VARDEF, LOAD, RCLD	VAR	none
User-defined function	FUNCDEF, LOAD, RCLD	FUNC	none
Active user-defined function	ACTDEF, LOAD, RCLD	AFUNC	none
On-end-of-sweep algorithm	ONEOS, LOAD, RCLD	ONEOS	none
Trace-amplitude correction offsets	AMPCOR	AMPCOR	none
State-register file (Instrument settings and title-line)	STOR, LOAD, SAVES, NSTATE	ASCII	s_
Trace file	STOR, SAVET	ASCII	t_
(USER) keys file	STOR, SAVEU	ASCII	u_
Limit-line file	STOR, LIMISAV	ASCII	l_

Query Response USTATE? queries the user state to the computer. Also queried are instructions required by the spectrum analyzer when the user state information is transmitted to the spectrum analyzer. User-state data is in absolute, block-data format and is preceded by spectrum-analyzer instructions.

```
→( #A )→[ msb length ]→[ lsb length ]→( USTATE )→(Sp)→( #A )→
              →[ msb length ]→[ lsb length ]→
         →[ data byte ]→[ output
                          termination ]→
```
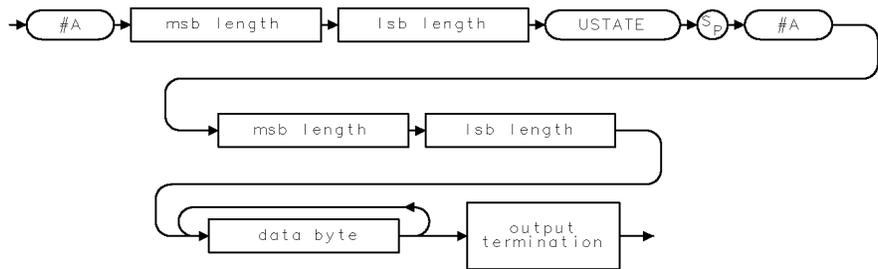
qustat

See Also DISPOSE, FUNCDEF, VARDEF, ONEOS, KEYDEF, READMENU, TRDEF, SAVES, NSTATE, SAVEU, POWER, PSTATE, PROTECT, USERERR, USERMSG

VARDEF

Store user-defined variables in internal memory.
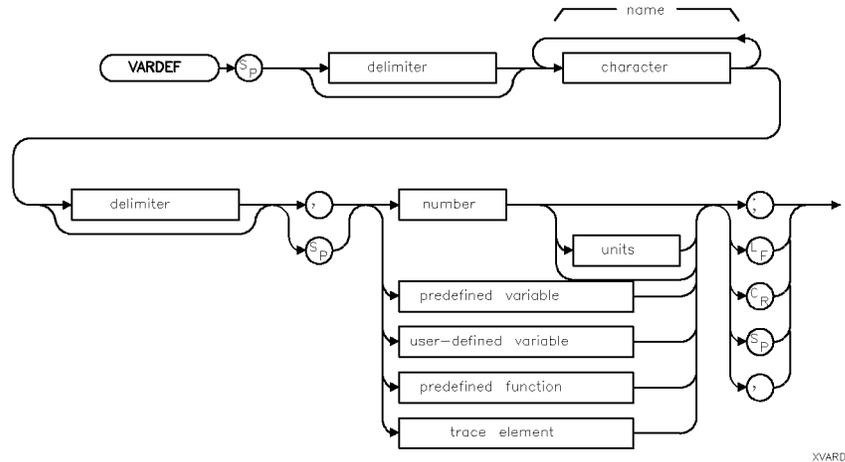
Item	Description
Variable-Name Range	1 – 12 characters: A – Z, 0 – 9, and the underscore |_|
Variable-Name Restrictions	A delimiter is required before and after a file name when it contains a number or lowercase letters. When the file name is uppercase alpha characters only, delimiters are optional.

Example

160	OUTPUT 723;"VARDEF VAR,0;"	*Define user-defined variable, called VAR, equal to zero.*
170	OUTPUT 723;"VAR EP;"	*Change the value of VARDEF using the front-panel controls. Enter a value, then press* ACCEPT VALUE.
180	OUTPUT 723;"VAR?;"	*Query value of VAR to computer.*

Description   The VARDEF command creates a user-defined variable and assigns it a value. User-defined variables can be used in many of the spectrum-analyzer remote-control processes. Use user-defined variables wherever *user-defined variable* appears in the syntax diagrams. An error results if a variable name is the same as any reserved word. User-defined variables occupy internal memory. Use the PURGE, FORMAT, or DISPOSE command to clear user-defined variables from memory.

Syntax to
Change Value
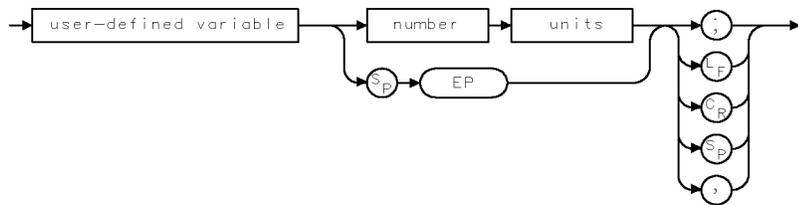
qvardecv.

Query Syntax
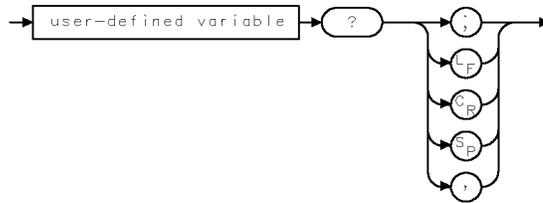
qvardeqs.

Query Response
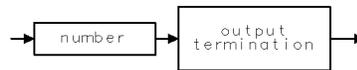
qgennum .

See Also   DISPOSE, LOAD, PROTECT, FORMAT, RCLD, PURGE

VARIANCE

Computes the amplitude variance of the specified trace, in measurement units.

XVARIA

Item	Description
Source-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	−32,768 – +32,767 measurement units for logarithmic trace data.
Source-Length Range	3 – 2048 position units. Default is 800 position units when using TRA, TRB, TRC.

Example

```
140   OUTPUT 723;"SNGLS;TS;"
```
Activate single-sweep mode and sweep measurement range.

```
150   OUTPUT 723;"VARIANCE TRA?;"
```
Query variance of trace A to computer.

Description The VARIANCE command must be used as either a query or as a source in another analyzer-command function. Form a query by ending the VARIANCE statement with a question mark (?). When used as a query, VARIANCE gives the variance to the computer. Use VARIANCE as a source by incorporating the VARIANCE statement into the source of any command having *predefined function* in its syntax diagram. When VARIANCE is used as a source, the variance is used for operation by the command that contains VARIANCE.

Predefined-
Function
Syntax
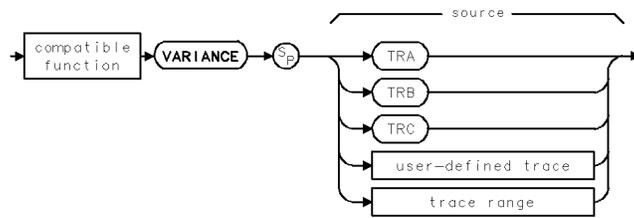
Query Syntax
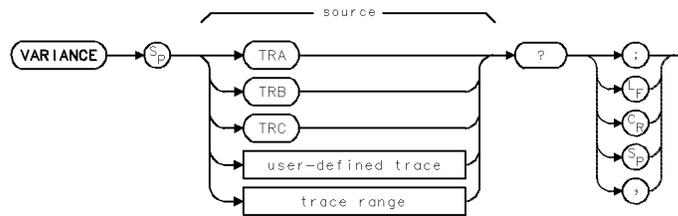
Query Response
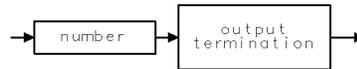
qgennum .

VAVG

Reduce amplitude variation of trace using digital averaging.
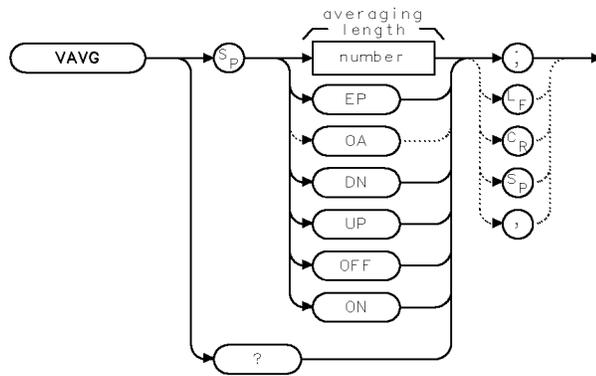
XVAVG

Item	Description
Averaging-Length Range	0 – 10,000
Step Increment	1, 2, 5, 10 sequence
Preset State	VAVG OFF; VAVG 100

Example

60	OUTPUT 723;"SNGLS;"	*Activate single-sweep mode.*
70	OUTPUT 723;"TS;"	*Sweep measurement range.*
80	OUTPUT 723;"VAVG EP;"	*Specify number of sweeps averaged. Enter value with front-panel controls, then press* ACCEPT VALUE.
90	OUTPUT 723;"TS;"	*Sweep.*

Description The VAVG command is a digital averaging routine that reduces the noise floor level, but does not affect the sweep time, bandwidth, or any other analog characteristic of the spectrum analyzer. VAVG operates on either trace A, B, or C, one-at-a-time, during clear-write mode.

The averaging function sets the averaging counter to 1, then begins averaging the trace as it sweeps. The active readout indicates the number of sweeps averaged; the default is 100 unless otherwise specified. Increasing the number of sweeps averaged increases the reduction of displayed noise.

When video averaging is on, the take-sweep command (TS) resets the counter to 1, then initiates the number of sweeps specified for averaging. The counter is also reset to 1 whenever any command changes the measurement data. VAVG activates sample detection. Commands that follow VAVG are executed while the averaging routine is executing. However, if TS follows VAVG, subsequent commands after TS are not executed until the end of the averaging routine. Use VAVG to view low-level signals without slowing the sweep time. If a large averaging number is specified, video averaging can lower the noise floor more than 1-Hz video-bandwidth filtering can. Video averaging may also be used to monitor instrument state changes (changing bandwidths, center frequencies, and so forth) while maintaining a low noise floor.

Query Response
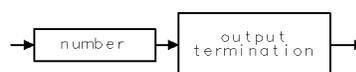
qgennum .

See Also TRPST, TM, VTL, VTH, VB, SMOOTH

VB

Select post-detection, video-bandwidth filter to reduce noise floor of trace data.
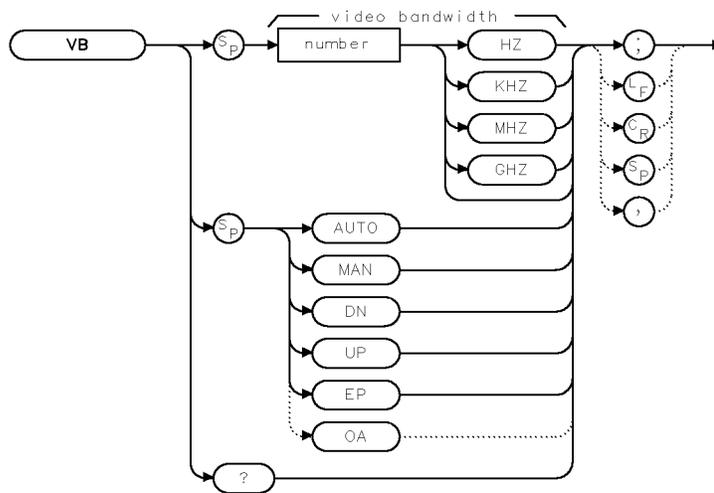
Item	Description
Bandwidth Range	1 – 3 kHz, or hardware limit.
Coupling	Sweep time |ST| is coupled to video bandwidth. Video bandwidth is coupled to resolution bandwidth |RB|.
Step Increment	1, 3, 10 sequence
Preset State	VB AUTO

Example   140   OUTPUT 723;"VB 3KHZ;"   *Set video bandwidth to 3 kHz.*

Description The VB command sets the post-detection, video-bandwidth filter. The video-bandwidth and sensitivity settings are coupled to yield the maximum amplitude response. Specifying a bandwidth with VB, or executing `VB MAN` breaks the coupling. To reestablish coupling, execute `VB AUTO`.

There are two types of detectors used in conjunction with post-detection processing. The optical spectrum analyzer firmware automatically determines which detection mode to enable. The following information explains the uses of the different detector modes.

- Sample detection is used when digital filtering is enabled, and the analyzer is in gated sweep and pulsed current modes of operation. When the sample detector is active, an `S` appears on the display beside the video bandwidth information.

- Peak detection is enabled at all other times. (The `S` is not displayed.)

There are two modes of operation for video bandwidth. They are automatic and manual mode. The conditions of these modes is listed below:

- In automatic, video-bandwidth mode, the optical spectrum analyzer firmware determines whether to use digital filtering and determines the appropriate bandwidth for the current measurement conditions.

  The video bandwidth information displayed is the overall post-detection bandwidth, and it may represent the bandwidth of the transimpedance amplifier hardware in cases where digital filtering is disabled.

- In manual, video-bandwidth mode, digital filtering is enabled, unless gated sweeps and pulsed current modes are selected. The filtering is fixed at the bandwidth entered by the operator.

  The minimum bandwidth available is 0.1 Hz. The maximum bandwidth available is the lesser of 3 kHz and the bandwidth of the currently selected transimpedance amplifier.

Query Response

```
┌──────────┐   ┌──────────────┐
│  number  │──▶│   output     │──▶
└──────────┘   │ termination  │
               └──────────────┘
```
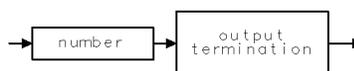
qgennum

See Also RB, SP, ST, SENS

VIEW

Stop sweeping the trace and store trace data.
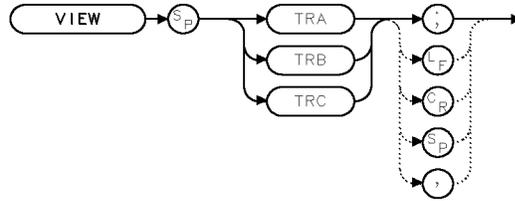
XVIEW

Example	120	OUTPUT 723;"CLRW TRB;BLANK TRA;"	*Measure with trace B. Blank trace A.*
	130	OUTPUT 723;"VIEW TRB;"	*View trace B.*

Description The VIEW command stops sweeping the specified trace and displays it.

See Also TRSTAT

VTDL

Control the video trigger display lines.
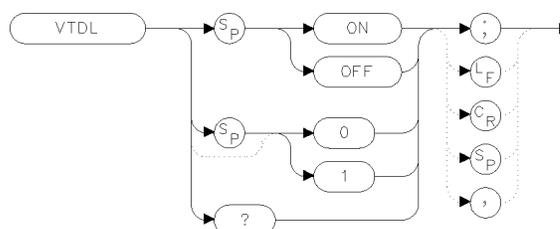
Item	Description
Preset State	VTDL OFF

Example

```
100   OUTPUT 723;"VTDL ON;"
```

Description The VTDL command displays two horizontal video trigger display lines: trigger level and hysteresis offset. The trigger level line is a line that is placed at a level which corresponds to the current video trigger level. On HP 70004A displays, the line is the same color as defined for MARKERS. This is determined by the display's "Adjust Color" menu. On monochrome displays, a dot-dash line is used.

The hysteresis offset line is an line that is offset from the trigger level line so as to indicate the size and polarity of video hysteresis necessary to cause a video trigger. On HP 70004A displays, the line is the same color as defined for ADVISE (UNCAL) indicators. On monochrome displays, a sparce dot line is used.

The video trigger display lines are automatically activated when video triggering is selected and when either the video trigger level command (VTL) or the video trigger hysteresis command (VTH) is active. (The video trigger level display line cannot be turned off when either VTL or VTH is active.)

The display of these lines is automatically deactivated when another trigger mode is selected (as long as neither VTL nor VTH is active). The VTDL

command can be used to override the automatic operation. The displays work in logarithmic, linear volts, and linear watts display modes.

Query Response
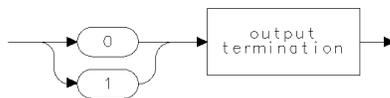
qgen0

See Also VTDL, VTH, VTL

VTH

Select slope of detected signal shape that triggers sweep.
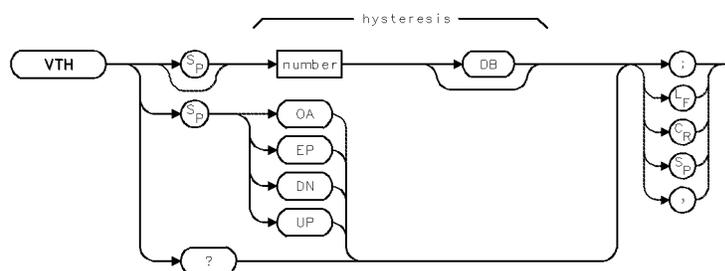
Item	Description
Range	± 300 dB
Step Increment	1 major vertical division during log mode. 1 major vertical division at top graticule line |.915 dB| during linear mode.
Preset State	**VTH 3DB**

Example

```
30  OUTPUT 723;"VTH;"
```

Description The VTH command determines the **video trigger hysteresis**. The hysteresis is the direction and magnitude of change in the detected video-signal level that triggers a sweep during the video trigger mode. Triggering occurs when the following conditions are met:

- The video-signal level changes by the magnitude specified by the VTH command.

- The video-signal level passes the video-trigger level specified by the VTL command.

When VTH specifies a positive value, the sweep triggers on the positive-going edge of the detected signal envelope. A negative value triggers a sweep on the negative-going edge.

Query Response

```
         ┌──────────┐    ┌──────────────┐
  ──────►│  number  │───►│    output    │────►
         └──────────┘    │ termination  │
                         └──────────────┘
```
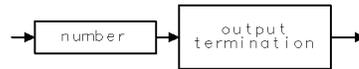
qgennum

See Also TS, VTL, TM

VTL

Select signal level that triggers sweep.

xvtl
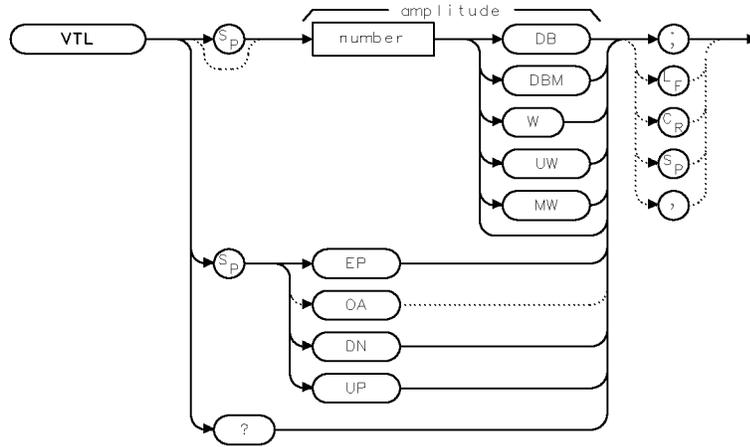
Item	Description
Amplitude Range	± 300 dBm or hardware limit.
Step Increment	1 major vertical division
Preset State	**VTL –100DBM** |or value at bottom of graticule|.

Example

```
10  OUTPUT 723;"VTL -50DBM;"
```

Description During the video trigger mode, the sweep begins when the detected signal level has reached a specific amplitude level. The VTL command specifies the amplitude level.

Query Response
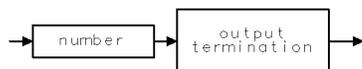
qgennum

See Also   TS, TM, VTH

VW

Blank or display graphics item.

XVW

Example

50 OUTPUT 723;"IT 1;VW OFF;"	*Address item 1. Blank item so that it is not seen while it is being defined.*
60 OUTPUT 723;"TEXT$This is item #1.$;"	*Define item 1 as text.*
70 OUTPUT 723;"IT 1;OR 300,600;"	*Position item.*
80 OUTPUT 723;"VW ON;"	*Display item 1.*

Description The VW command blanks or displays graphics items that have been created with the IT command.

See Also CLRDSP, IT

WAIT

Suspend program execution.
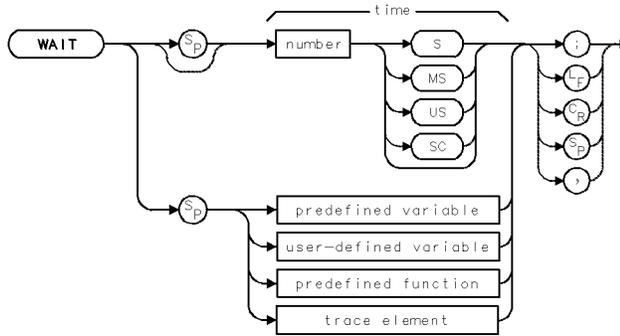
Item	Description
Range	0 - 1000 seconds

Example

```
130   OUTPUT 723;"TRDSP TRA,OFF;"      Blank trace A.
140   OUTPUT 723;"WAIT 2S;"           Suspend program operation for 2 seconds.
150   OUTPUT 723;"TRDSP TRA,ON;"      Display trace A.
```

Description The WAIT command suspends program execution for the specified time.

WARN?

Return a listing of all reported warning codes.
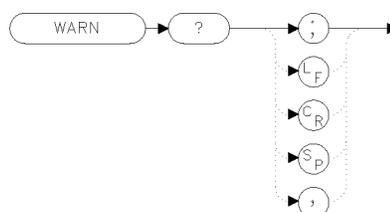
WARN ► ? ► ;
► L F
► C R
► S P
► ,

xwarn

Example

```
100   OUTPUT 723;";"
100   OUTPUT 723;"WARN?;"
```

Description If no warnings have been displayed, the default value of 10000 is returned. Bit 1 of the status byte is used to report the presence of a warning. Refer to STB and related commands for other information concerning the status byte. The following list shows the possible error messages. Of course, user-defined error messages are not included. For more information on the error messages, refer to Chapter 4 of the *HP 71450B/1B/2B Optical Spectrum Analyzer Reference*.

Error 10000: *Default - No warning displayed.*
Error 16001: Simulated hardware
Error 16002: Trace length changed (due to ADC switch)
Error 16003: Usable RBW limited
Error 16004: Usable VBW limited
Error 16005: Sweep too fast
Error 16006: No current state 0
Error 16007: Possible compression
Error 16008: No autoalign data (OSA)
Error 16009: RBW too wide for PP (currently unused)
Error 16010: RBW too wide for PC
Error 16011: Span too large (OSA)
Error 16012: PSDAC adjust past RL

Query Response

qgnumr

See Also   STB, USERWARN, WARN?, WARNCTRL, XWARN

Language Reference

WARNCTRL

WARNCTRL Enable or disable warning messages.
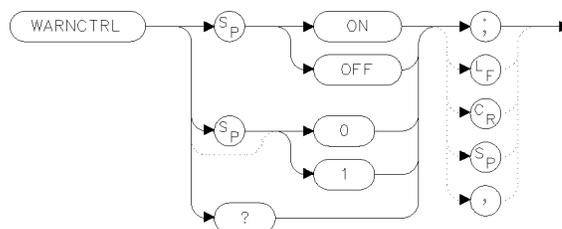
xwrncl

```
100   OUTPUT 723;";"
100   OUTPUT 723;"WARNCTRL ON;"
```

Description The WARNCTRL command prevents the display and reporting of warnings.
An instrument preset (IP command) enables warnings. If display of the active
function area is disabled, the warning area will also be suppressed. Warnings
can still be accessed via WARN and XWARN.

Query Response
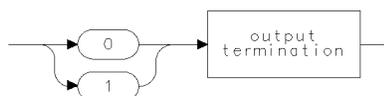
qgen0

See Also STB, USERWARN, WARN?, WARNCTRL, XWARN

7-545

WLLIMIT

Extend wavelength measurement range.
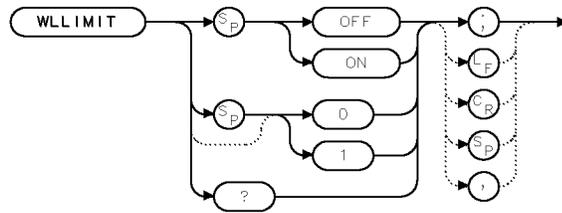
XWLLIM

Item	Description
Preset State	WLLIMIT ON

Example   `10   OUTPUT 723;"WLLIMIT OFF;"`

Description The WLLIMIT command extends the wavelength measurement range. The normal range (`WLLIMIT ON`) is from 600 nm to 1700 nm. The extended range (`WLLIMIT OFF`) is from 350 nm to 2000 nm (uncorrected).

Query Response
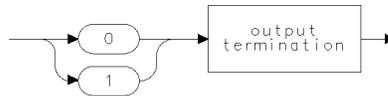
qgen0

See Also Commands affecting wavelength, such as CENTERWL, STARTWL, STOPWL, FS, and SP.

WLMKRL

Position the left power integration point to a location based upon the wavelength value entered.


ckb18

Example

100	OUTPUT 723;"CENTERWL 900NM;"	*Set the center wavelength to 900 nm.*
105	OUTPUT 723;"TS;"	*Update the current trace.*
110	OUTPUT 723;"WMLKRR 1000E-9;"	*Position the right power integration point to 1000 nm.*
12	OUTPUT 723;"WLMKRL 800 NM;"	*Position the left power integration point to 800 nm.*

Description The WLMKRL command positions the left power integration point to a location based upon the wavelength value entered. The left point is specified as marker number 4 and the right point is specified as marker number 5.

See Also AMPMKR, WLMKRR

WLMKRR

Position the right power integration point to a location based upon wavelength value entered.

ckb17

Example

100	OUTPUT 723;"CENTERWL 900NM;"	*Set the center wavelength to 900 nm.*
105	OUTPUT 723;"TS;"	*Update the current trace.*
110	OUTPUT 723;"WMLKRR 1000E-9;"	*Position the right power integration point to 1000 nm.*
12	OUTPUT 723;"WLMKRL 800 NM;"	*Position the left power integration point to 800 nm.*

Description The WLMKRR command positions the right power integration point based on upon the wavelength value entered. The left point is specified as marker number 4 and the right point is specified as marker number 5.

See Also AMPMKR, WLMKRL

WLOFFSET

Offset wavelength scale for all absolute wavelength readouts, inputs, and outputs.
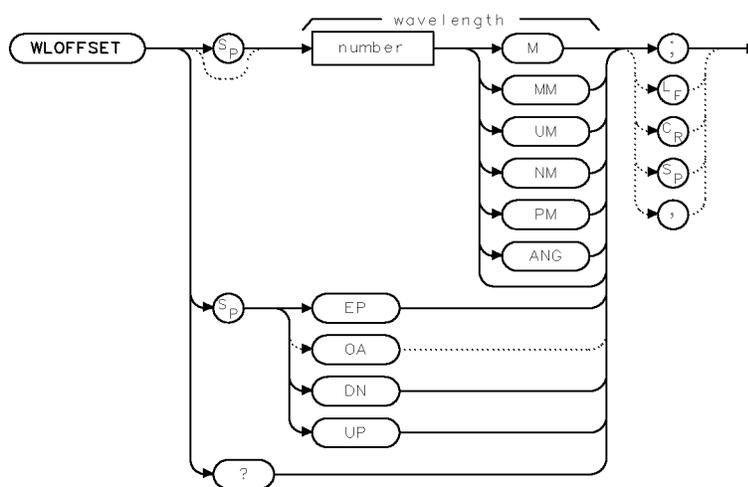
XWLOFF

Item	Description
Offset range	Actual range limited by hardware.
Step Increment	For spans greater than 0, the step increment is span/10. For zero span, the step increment is the resolution bandwidth.
Preset State	WLOFFSET 0

Example

```
10  OUTPUT 723;"WLOFFSET 10NM;"
```

Description The WLOFFSET command offsets the wavelength scale for all absolute wavelength readouts, inputs, and outputs.

Relative values, like span and delta marker, are not offset. Whenever the offset is active, "WLOFFSET" appears below the graticule. Execute WLOFFSET 0 or IP to turn off the offset.

Query Response   WLOFFSET value is returned as measurement units.
Measurement units = $100 \times$ value in dBm.
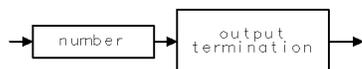
qgennum

See Also   Commands affecting wavelength, such as CENTERWL, STARTWL, STOPWL,
FS, and SP.

WLUNITS

Convert trace position units to wavelength.
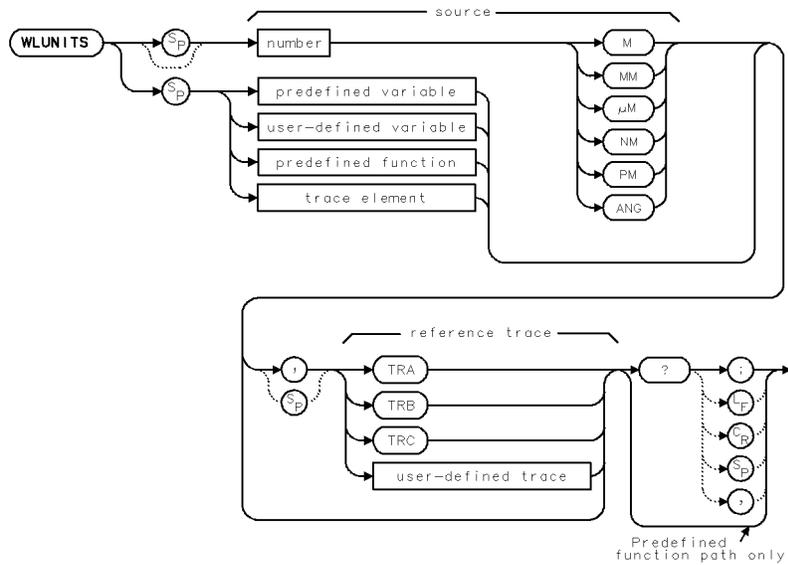
ckb7

Item	Description
Reference-Trace Length Range	3 – 2048 position units. Default is 800 when using *TRA, TRB, TRC*.
Source Range	1 to 2048 position units.
Reference-Trace-Amplitude Range	0 – 10,000 measurement units for linear trace data.
	—32,768 – +32,767 measurement units for logarithmic trace data.

Example Determine the wavelength that corresponds to a point's relative position on a trace.

10	`CLEAR 723`	*Initialize analyzer.*
20	`OUTPUT 723;"IP;";`	
30	`OUTPUT 723;"CLRDSP;"`	*Clear graphics from memory.*
40	`OUTPUT 723;"PU;PA 200,700;"`	*Place pen up, then move it to (200,700).*
50	`OUTPUT 723;"TEXT$8th element is $;"`	*Write text.*
60	`OUTPUT 723;"DSPLY WLUNITS 8,TRA,7,2;"`	*Display wavelength value of 8th element in trace A.*
70	`OUTPUT 723;"TEXT$M$;"`	*Print "M" after variable.*

Description The WLUNITS command converts a trace point indicated in position units to wavelength. In zero-span mode, position units are converted to units of time. A position unit describes the position of a point along the horizontal axis of a trace. The point at the left end of the trace has a position-unit value of 1. The point at the right end of an 800-point trace has a position-unit value of 800. If a reference trace is not indicated, the number of points in the currently active or last active trace is assumed for WLUNITS operations and the conversion is based on the wavelength values of that trace.

WLUNITS must be used either as a query or as an argument to another function. Form a query by ending the WLUNITS statement with a question mark (?). When used as a query, WLUNITS gives the wavelength value of the specified trace position. Use WLUNITS as an argument by incorporating the WLUNITS statement into the source of any spectrum-analyzer command having *predefined function* in its syntax diagram. When used as a source, WLUNITS uses the wavelength value of the specified point for operation by the command that contains WLUNITS.
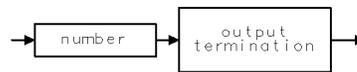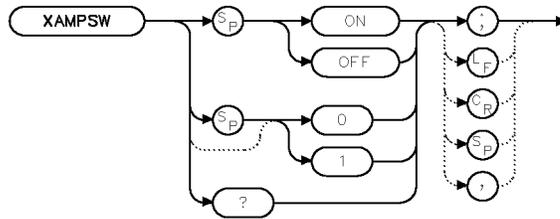
Prede-
fined-Function
Syntax
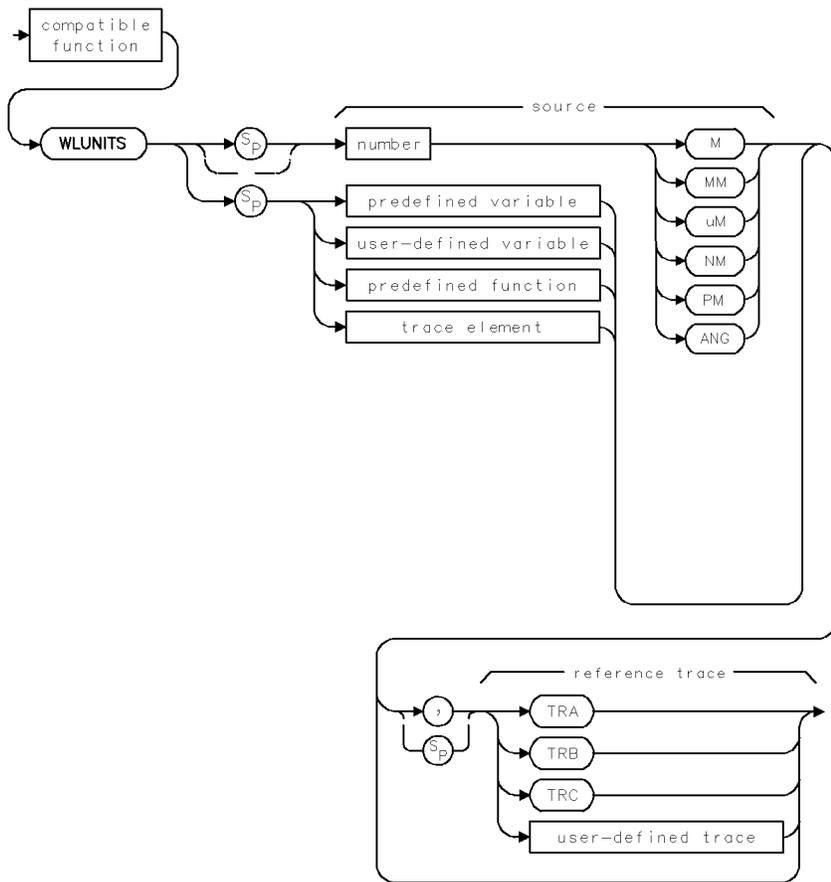
cgz_f07y

Query Syntax
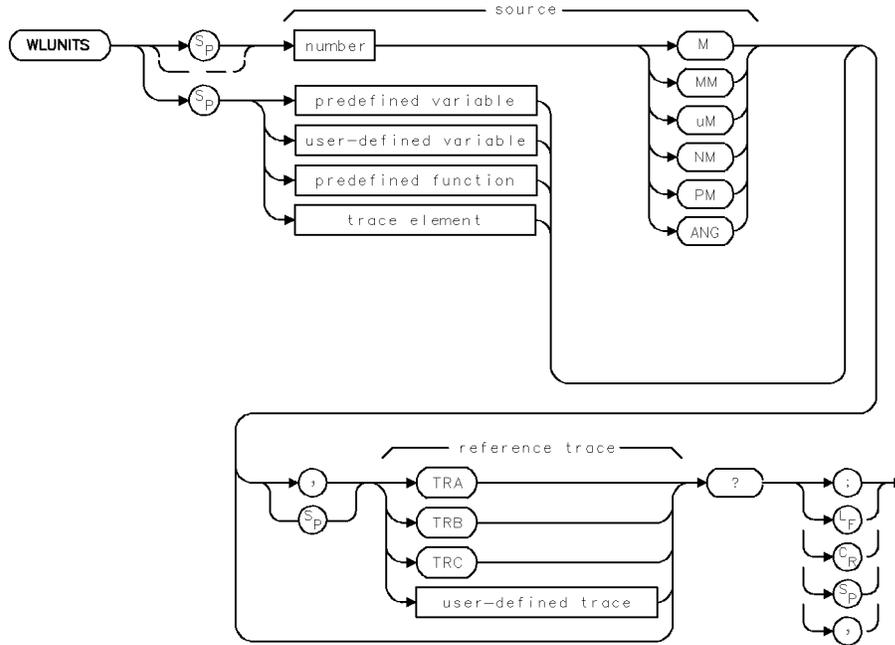
Query Response

qgennum

XAMPSW

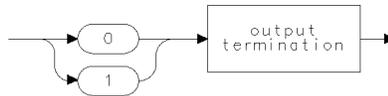Activate the rear-panel TRANS-Z INPUT connector.

x x amp

Example   `10  OUTPUT 723;"XAMPSW ON;"`

Description The XAMPSW command allows input via the rear-panel TRANS-Z INPUT connector. Normally, the output of the optical spectrum analyzer's internal photodetector connects to a transimpedance amplifier. Using the ON argument switches the amplifier's input from the photodetector to the rear-panel TRANS-Z INPUT connector.

CAUTION The maximum current allowed at the rear-panel TRANS-Z INPUT connector is ± 10 mA and the maximum input voltage is ± 10V.
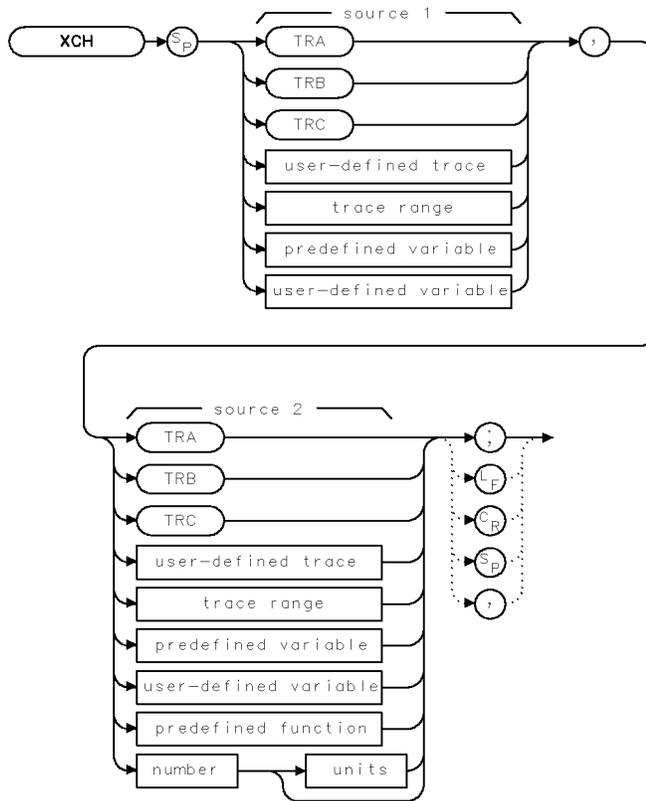
Query Response

qgen0

XCH

Exchange source contents.

XXCH

Item	Description
Source or Destination Amplitude Range	0 – 10,000 measurement units for linear trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	−32,768 – +32,767 measurement units for logarithmic trace data in *TRA, TRB, TRC, trace range*, or *user-defined trace*.
	± 1.8 × 10^{308} when using *number, predefined variable*, or *user-defined variable*.
Source-Length Range	1 – 2048 position units. Default is 800 position units when using *TRA, TRB, TRC*.

Example

|		
100	OUTPUT 723;"SNGLS;TS;"	*Activate single-sweep mode. Sweep trace A.*
110	OUTPUT 723;"XCH TRA[1,400],TRA[401,800];"	*Exchange the amplitudes of the first 400 elements of trace A with the last 400 elements.*
120	OUTPUT 723;"VIEW TRA;"	*View trace A.*

Description The XCH command exchanges the contents of sources 1 and 2. If either of the sources is trace A, B, or C, the trace conditions are updated accordingly.

XERR

Return error information.
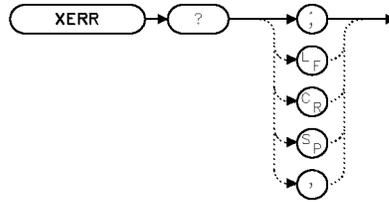
XXERR

Example

140 ALLOCATE Error$[200]	
150 OUTPUT 723;"JKLD;"	*Send the spectrum analyzer a meaningless command.*
160 OUTPUT 723;"VB -2NM;"	*Attempt to set the spectrum analyzer to an unavailable video bandwidth.*
170 OUTPUT 723;"CLRW;"	*Send the spectrum analyzer illegal syntax.*
180 OUTPUT 723;"WAIT 2S;"	*Wait 2 seconds.*
190 OUTPUT 723;"XERR?;"	*Query reported errors to computer.*
200 ENTER 723 USING "-K";Error$	*Store error messages in string variable, accepting line feeds.*
210 PRINT Error$	*Print error report on computer.*

Description

The XERR command gives error information when an instrument in a system encounters an error.
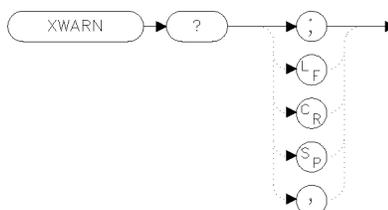
Information queried includes the name and HP-MSIB address of the module reporting the error, the error code, and a description. IP clears errors from register.

Query Response
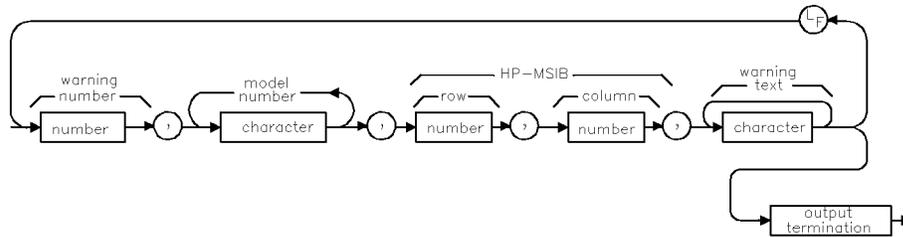
cgz_x02q

XWARN

Return an extended listing of all reported warnings.
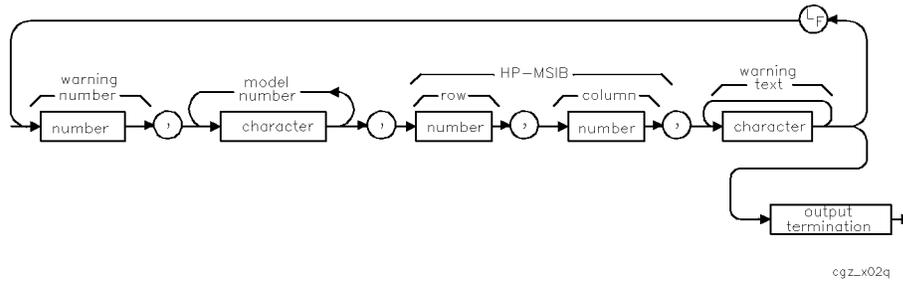


```
xwarnx
```

```
100   OUTPUT 723;";"
100   OUTPUT 723;"XWARN?;"
```

Description If no warnings have been displayed, the default value of 10000 is returned. Bit 5 of the status byte is used to report the presence of a warning Refer to STB and related commands for other information concerning the status byte. The following list shows the possible error messages. Of course, user-defined error messages are not included. For more information on the error messages, refer to Chapter 4 of the *HP 71450B/1B/2B Optical Spectrum Analyzer Reference*.

Error 10000: *Default - No warning displayed.*
Error 16001: Simulated hardware
Error 16002: Trace length changed (due to ADC switch)
Error 16003: Usable RBW limited
Error 16004: Usable VBW limited
Error 16005: Sweep too fast
Error 16006: No current state 0
Error 16007: Possible compression
Error 16008: No autoalign data (OSA)
Error 16009: RBW too wide for PP (currently unused)
Error 16010: RBW too wide for PC
Error 16011: Span too large (OSA)
Error 16012: PSDAC adjust past RL

Query Response

cgz_x02q

See Also STB, USERWARN, WARN?, WARNCTRL, XWARN

ZERO Enable automatic zeroing of photo-diode's dark current.