

# Programming Manual

## HP 8590 Series Spectrum Analyzer

### SERIAL NUMBERS

This manual applies to HP 8590B, HP 8591A, HP 8592B and HP 8593A Spectrum Analyzers with serial numbers prefixed 3003A and below, with firmware version 03.01.90 or later. If your analyzer has an earlier version of firmware, see page vi, "Analyzers with Earlier Versions of Firmware."



HP Part No. 5958-7094    Microfiche Part No. 5958-7095  
Printed in USA    January 1990

Faint, illegible text at the top of the page, possibly a header or title area.

Faint, illegible text in the middle section of the page.

Faint, illegible text in the lower middle section of the page.

---

## 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.

---

## 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 HP. Buyer shall prepay shipping charges to HP and HP shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to HP from another country.

HP warrants that its software and firmware designated by HP for use with an instrument will execute its programming instructions when properly installed on that instrument. HP 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. HP 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. HP 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.*

---

## HP 8590 Series Spectrum Analyzer Documentation Description

### Manuals Shipped with your HP 8591A or HP 8593A

#### HP 8591A/8593A Installation, Verification, and Operation Manual

- Tells you how to install the HP 8591A or HP 8593A Spectrum Analyzer.
- Tells you how to make measurements with your HP 8591A or HP 8593A Spectrum Analyzer.
- Describes analyzer features.
- Tells you what to do in case of a failure.

#### HP 8591A/8593A Quick Reference Guide

- Describes how to make a simple measurement with the HP 8591A or HP 8593A.
- Briefly describes the analyzer functions.
- Lists all the programming commands.

### Manuals Shipped with your HP 8590B or HP 8592B

#### HP 8590B/8592B Installation, Verification, and Operation Manual

- Tells you how to install the HP 8590B or HP 8592B Spectrum Analyzer.
- Tells you how to make measurements with your HP 8590B or HP 8592B Spectrum Analyzer.
- Describes analyzer features.
- Tells you what to do in case of a failure.

#### HP 8590B/8592B Quick Reference Guide

- Describes how to make a simple measurement with the HP 8590B or HP 8592B.
- Briefly describes the analyzer functions.
- Lists all the programming commands.

### How to Order Manuals

Each of the manuals listed above can be ordered individually. To order, contact your local HP Sales and Service Office.

---

## How to Use This Manual

### Where to Start

- If you have not configured your spectrum analyzer in your computer system, first read Chapter 1, "Preparation for Use." This chapter tells you how to set up your computer/spectrum-analyzer system.
- If you are familiar with spectrum analyzer programming and wish to find the description of a programming command, turn to Chapter 4, "Programming Commands."
- If you are not familiar with spectrum analyzer programming,

Turn to Chapter 2, "Programming Fundamentals." This chapter introduces spectrum analyzer programming by leading you through a simple spectrum analyzer measurement.

After you've successfully made your first measurement (or if you are experienced in remote operation of the spectrum analyzer), you may turn to Chapter 3, "Advanced Programming," which demonstrates advanced programming techniques. Or, if you begin writing your own programs, turn to Chapter 4, "Programming Commands," for command descriptions.

### Manual Terms and Conventions

The six keys along the right side of the display are called softkeys. Their labels are displayed on the screen. The softkeys appear in shaded boxes in this manual, for example, **REF LVL**. Pressing the labeled keys on the front panel changes the softkey labels or initiates functions. The front-panel keys appear in unshaded boxes, for example, **FREQUENCY**.

---

#### Caution



The CAUTION symbol 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 symbol until the indicated conditions are fully understood and met.

---

## Analyzers with Earlier Versions of Firmware

This manual documents functions which may not be available with earlier versions of firmware. The following table lists commands that have been added with a firmware revision.

**Analyzer Commands Added with Firmware Revision**

Command Mnemonic	Option Required	First Available with Firmware Revision
AMPCOR		03.01.90
LIMIDEL		03.01.90
LIMIFAIL		03.01.90
LIMILINE		03.01.90
LIMIMIRROR		03.01.90
LIMIMODE		03.01.90
LIMIREL		03.01.90
LIMISEG		03.01.90
LIMITEST		03.01.90
MEASURE	010 or 011	03.01.90
NRL		03.01.90
POWERON		03.01.90
RESETRL		03.01.90
RLPOS		03.01.90
SEGDEL		03.01.90
SENER		03.01.90
SRCALC	010 or 011	03.01.90
SRCAT	010 or 011	03.01.90
SRCNORM	010 or 011	03.01.90
SRCPOFS	010 or 011	03.01.90
SRCPSTP	010 or 011	03.01.90
SRCPSWP	010 or 011	03.01.90
SRCPWR	010 or 011	03.01.90
SRCTK	010 or 011	03.01.90
SRCTKPK	010 or 011	03.01.90
SWPCPL	010 or 011	03.01.90
TVSTND	101 and 102	03.01.90

## Contents

---

<b>1. Preparation for Use</b>	
What You'll Learn in This Chapter . . . . .	1-1
Connecting Your Analyzer to a Computer . . . . .	1-1
Configuring Your Computer System . . . . .	1-1
Computer/Analyzer Interconnections . . . . .	1-1
For the HP-IB Interface . . . . .	1-1
For the RS-232 Interface . . . . .	1-2
The Test Program . . . . .	1-2
HP 9000 Series 200 Technical Computers . . . . .	1-3
Equipment . . . . .	1-3
Interconnection Instructions . . . . .	1-3
Test Program . . . . .	1-4
HP 9000 Series 300 Technical Computers . . . . .	1-5
Equipment . . . . .	1-5
Interconnection Instructions . . . . .	1-5
Test Program . . . . .	1-6
HP Vectra Personal Computer . . . . .	1-7
With the HP-IB Interface . . . . .	1-7
Equipment . . . . .	1-7
Interconnection Instructions . . . . .	1-7
Test Program . . . . .	1-8
HP Vectra Personal Computer . . . . .	1-8
With the RS-232 Interface . . . . .	1-8
Equipment . . . . .	1-8
Interconnection Instructions . . . . .	1-9
Test Program . . . . .	1-9
IBM PC/AT and Compatible Computers . . . . .	1-10
With the RS-232 Interface . . . . .	1-10
Equipment . . . . .	1-10
Interconnection Instructions . . . . .	1-10
Test Program . . . . .	1-11
Connecting Your Analyzer to a Printer or Plotter . . . . .	1-12
To Print with an HP-IB Interface . . . . .	1-12
Equipment . . . . .	1-12
Interconnection Instructions . . . . .	1-12
To Plot with an HP-IB Interface . . . . .	1-13
Equipment . . . . .	1-13
Interconnection Instructions . . . . .	1-13
To Print with an RS-232 Interface . . . . .	1-14
Equipment . . . . .	1-14
Interconnection Instructions . . . . .	1-14
To Plot with an RS-232 Interface . . . . .	1-15

Equipment . . . . .	1-15
Interconnection Instructions . . . . .	1-15
Printing after Plotting or Plotting after Printing . . . . .	1-16
If There is a Problem . . . . .	1-17
<b>2. Programming Fundamentals</b>	
What You'll Learn in This Chapter . . . . .	2-1
Writing Your First Program . . . . .	2-1
Composing the Program . . . . .	2-1
Program Example for the HP-IB Interface . . . . .	2-2
Program Example for the RS-232 Interface . . . . .	2-2
Modifying the Program . . . . .	2-3
Program Example for the HP-IB Interface . . . . .	2-3
Program Example for the RS-232 Interface . . . . .	2-4
Enhancing the Program with Variables . . . . .	2-5
Program Example for the HP-IB Interface . . . . .	2-5
Program Example for the RS-232 Interface . . . . .	2-6
Getting Information from the Analyzer . . . . .	2-7
Program Example for the HP-IB Interface . . . . .	2-7
Program Example for the RS-232 Interface . . . . .	2-8
Programming Guidelines . . . . .	2-9
<b>3. Advanced Programming</b>	
What You'll Learn in This Chapter . . . . .	3-1
Controlling Trace Data with a Computer . . . . .	3-1
Reading Trace Data . . . . .	3-1
Program Example for the HP-IB Interface . . . . .	3-1
Program Example for the RS-232 Interface . . . . .	3-2
Saving Trace Data . . . . .	3-3
Program Example for the RS-232 Interface . . . . .	3-4
Reading Trace Data from a Computer Disk . . . . .	3-5
Program Example for the HP-IB Interface . . . . .	3-5
Program Example for the RS-232 Interface . . . . .	3-5
Saving and Recalling Instrument States . . . . .	3-6
Saving the Analyzer's State . . . . .	3-6
Program Example for the HP-IB Interface . . . . .	3-6
Program Example for the RS-232 Interface . . . . .	3-7
Returning the Analyzer to its Former State . . . . .	3-9
Program Example for the HP-IB Interface . . . . .	3-9
Program Example for the RS-232 Interface . . . . .	3-10
Measuring Harmonic Distortion . . . . .	3-11
Program Example for the HP-IB Interface . . . . .	3-12
Program Example for the RS-232 Interface . . . . .	3-14
Trace Data Transfers . . . . .	3-16
P Format . . . . .	3-17
Example of Using the P Format . . . . .	3-17
B Format . . . . .	3-17
Example of Using the B Format . . . . .	3-17
A-Block Format . . . . .	3-18
Example of Using the A-Block Format . . . . .	3-18
I-Block Format . . . . .	3-18

Example of Using the I-Block Format . . . . .	3-19
M Format . . . . .	3-19
Example of Using the M Format . . . . .	3-20
<b>4. Programming Commands</b>	
What You'll Learn in This Chapter . . . . .	4-1
Syntax Conventions . . . . .	4-1
<b>A. Analyzer Error Messages</b>	
<b>B. HP-IB Option 021</b>	
<b>C. RS-232 Option 023</b>	
What You'll Learn in This Appendix . . . . .	C-1
Introducing the RS-232 Interface . . . . .	C-1
The RS-232 Data Lines . . . . .	C-1
The RS-232 Handshaking Lines . . . . .	C-1
Baud Rate . . . . .	C-2
Protocol . . . . .	C-2
Connecting a ThinkJet Printer . . . . .	C-3
ThinkJet Printer Mode Switches: . . . . .	C-3
Connecting a Modem . . . . .	C-4
System Settings . . . . .	C-4
Connecting an HP-GL Plotter . . . . .	C-4
Switch Settings . . . . .	C-4
Setting the Analyzer Baud Rate . . . . .	C-5

## Index

## Figures

---

1-1. Connecting the HP 9000 Series 200 Computer to the Analyzer . . . . .	1-3
1-2. Connecting the HP 9000 Series 300 Computer to the Analyzer . . . . .	1-5
1-3. Connecting the HP Vectra Personal Computer to the Analyzer . . . . .	1-7
1-4. Connecting the HP Vectra Personal Computer to the Analyzer . . . . .	1-9
1-5. Connecting an IBM PC/AT Compatible Computer to the Analyzer . . . . .	1-11
3-1. Trace Amplitude . . . . .	3-20
4-1. Command Syntax Figure . . . . .	4-1
4-2. Hanning Filter Window . . . . .	4-127
4-3. Uniform Filter Window . . . . .	4-127
4-4. Flat Top Filter Window . . . . .	4-127
B-1. HP-IB Connector . . . . .	B-1
C-1. RS-232 Connector . . . . .	C-1

## Tables

---

4-1. Syntax Elements . . . . .	4-3
4-2. Characters and Secondary Keywords (Reserved Words) . . . . .	4-6
4-3. Summary of Compatible Commands . . . . .	4-10
4-4. Functional Index . . . . .	4-12
4-5. Command Cross Reference for the HP 8590 Series . . . . .	4-25
4-6. Character Set . . . . .	4-172
4-7. Label Functions . . . . .	4-173
4-8. Analyzer Status Byte . . . . .	4-375
C-1. Full Handshaking Connection . . . . .	C-3
C-2. 3-Wire Connection . . . . .	C-3
C-3. ThinkJet Printer Connection . . . . .	C-3
C-4. Modem Connection . . . . .	C-4
C-5. HP-GL Plotter Connection . . . . .	C-4

## Preparation for Use

---

### What You'll Learn in This Chapter

This chapter tells you how to connect a computer to your spectrum analyzer via the Hewlett-Packard Interface Bus (HP-IB) or the RS-232 Interface Bus, and how to connect a printer or a plotter. The remainder of the chapter covers procedures to follow if a problem is encountered.

### Connecting Your Analyzer to a Computer

The analyzer works with many popular computers. However, the steps required to connect your spectrum analyzer to a specific computer depends on the computer you are using. Before turning to the interconnection instructions for your computer, please read the following general information.

### Configuring Your Computer System

Every computer system has a specific configuration. Your system configuration might include a printer, external disk drive, or plotter. Whenever you add another piece of equipment (for example, your spectrum analyzer), you may need to reconfigure your computer system so that the computer knows where and how to send information to the newly added device.

Some computers do not require configuring when a spectrum analyzer is connected; others require a simple modification. The most common modification is changing the configuration information stored on the computer's operating system disk. A few computers require the insertion of an add-on board, or "card." Refer to your computer documentation if your system needs these modifications.

All of the test programs for HP-IB and RS-232 interfaces are written using the BASIC language of the computer under consideration. If you have never entered or run a BASIC program, refer to your computer documentation.

### Computer/Analyzer Interconnections

#### For the HP-IB Interface

Refer to Appendix B for a detailed description of the HP-IB interface.

Interconnection instructions for the HP-IB interface are provided for use on an HP 9000 Series 200 and Series 300 computer, and an HP Vectra PC equipped with an HP 82300B BASIC Language Processor. If your computer is not listed and it supports an HP-IB interface, there is a good possibility that it can be connected to the spectrum analyzer. Consult your computer documentation to determine how to connect external devices on the bus.

### For the RS-232 Interface

Refer to Appendix C for a detailed description of the RS-232 interface.

Interconnection instructions for an HP Vectra PC and IBM PC/AT or compatible computers are provided for the RS-232 option interface. If your computer is not listed and it supports a standard RS-232 interface, there is a good possibility that the spectrum analyzer may be connected to the computer. Consult your computer documentation to determine how to connect external devices to your computer's RS-232 connector.

There are two types of RS-232 devices: Data Terminal Equipment (DTE) and Data Communication Equipment (DCE). Types of DTE devices include CRT terminals. DCE equipment includes modems and, generally, other computer RS-232 devices. The spectrum analyzer RS-232 port is the DTE-type. Connections from the computer (DCE) to the analyzer (DTE) are shown in Appendix C.

### The Test Program

To test the system configuration, a simple test program is provided for each computer listed. After you have connected your computer and analyzer, you should enter and run the test program on your computer to make sure it is sending instructions to the analyzer through the interface cable. If the interface is working and the program is entered correctly, a statement is displayed on the computer screen.

---

**Note**

The listed computer and analyzer equipment includes the minimum components necessary to establish communication between your analyzer and computer. If you are using application software, check with your software supplier for specific computer hardware and memory requirements.

---

**Note**

Using an interface cable other than the one listed with your computer's interconnection instructions may prevent proper communication between the analyzer and computer.

---

Pressing **CONFIG** (LOCAL) removes the analyzer from remote mode and enables front-panel control.

## HP 9000 Series 200 Technical Computers

### Equipment

- HP 9816, 9826, or 9836 Series 200 Technical Computer.
- HP 8590 Series Spectrum Analyzer, Option 021.
- HP 10833 (or equivalent) HP-IB cable.

### Interconnection Instructions

Connect the spectrum analyzer to the computer using the HP-IB cable. Figure 1-1 shows an HP Model 9836 connected to the analyzer.

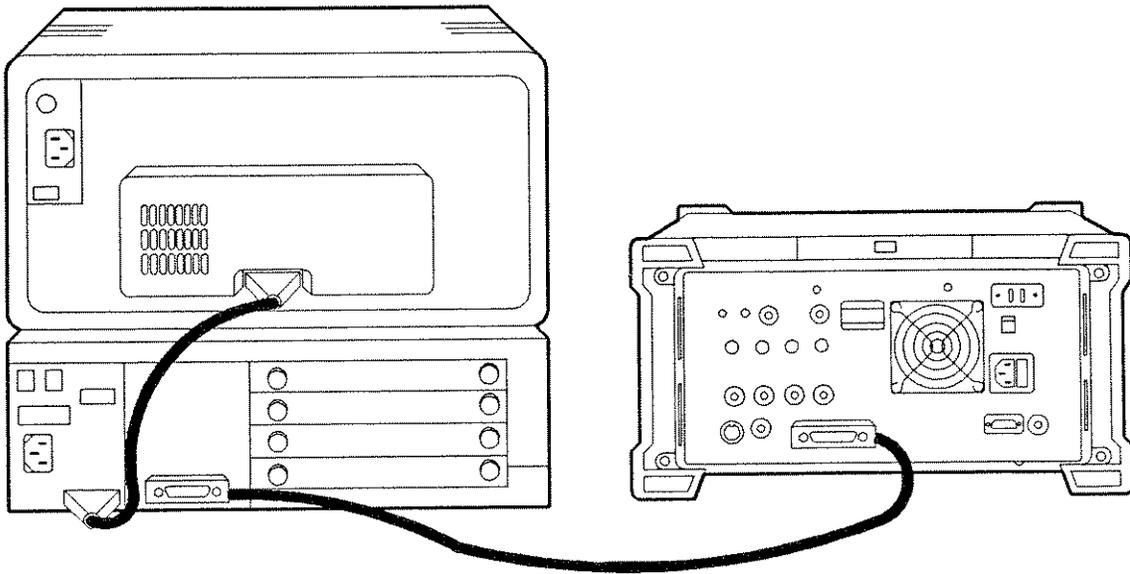


Figure 1-1. Connecting the HP 9000 Series 200 Computer to the Analyzer

## Test Program

To test the computer/analyzer interconnection, turn on your analyzer and follow the instructions below.

Your HP 9000 Series 200 computer may have either a soft-loaded or built-in language system. If your language system is built-in, remove any disks from the drives and turn on the computer.

If your language is soft-loaded, install the BASIC language disk into the proper drive. Turn the computer power on. After a few seconds, the BASIC READY message appears; the computer is now ready for use.

For further information on loading BASIC on your system, consult your BASIC manual.

Check the HP-IB address of the spectrum analyzer. Press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**. If necessary, reset the address of the analyzer: press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**, 18 **Hz** (or enter the appropriate address).

Enter the following program and press **RUN** on the computer. If you need help entering and running the program, refer to your computer and software documentation.

```
10 PRINTER IS 1
20 Analyzer=718
30 CLEAR Analyzer
40 OUTPUT Analyzer;"IP;SNGLS;"
50 OUTPUT Analyzer;"CF 300MZ;TS;"
60 OUTPUT Analyzer;"CF?;"
70 ENTER Analyzer;A
80 PRINT "CENTER FREQUENCY = ";A;"Hz";
90 END
```

The program tells the analyzer to perform an instrument preset and enter single-sweep mode. Next, the program sets the center frequency to 300 MHz and takes a sweep. The program then queries the center frequency value and the computer displays **CENTER FREQUENCY = 3.0E+8 Hz**. The program shows that the computer is able to send instructions to, and read information from, the analyzer.

If the computer does not display the center frequency, refer to "If There is a Problem" at the end of this chapter.

## HP 9000 Series 300 Technical Computers

### Equipment

- HP 98580A, 98581A, 98582A, or 98583A Series 300 Technical Computer.
- HP 8590 Series Spectrum Analyzer, Option 021.
- HP 10833 (or equivalent) HP-IB cable.

### Interconnection Instructions

Connect the spectrum analyzer to the computer using the HP-IB cable as shown in Figure 1-2.

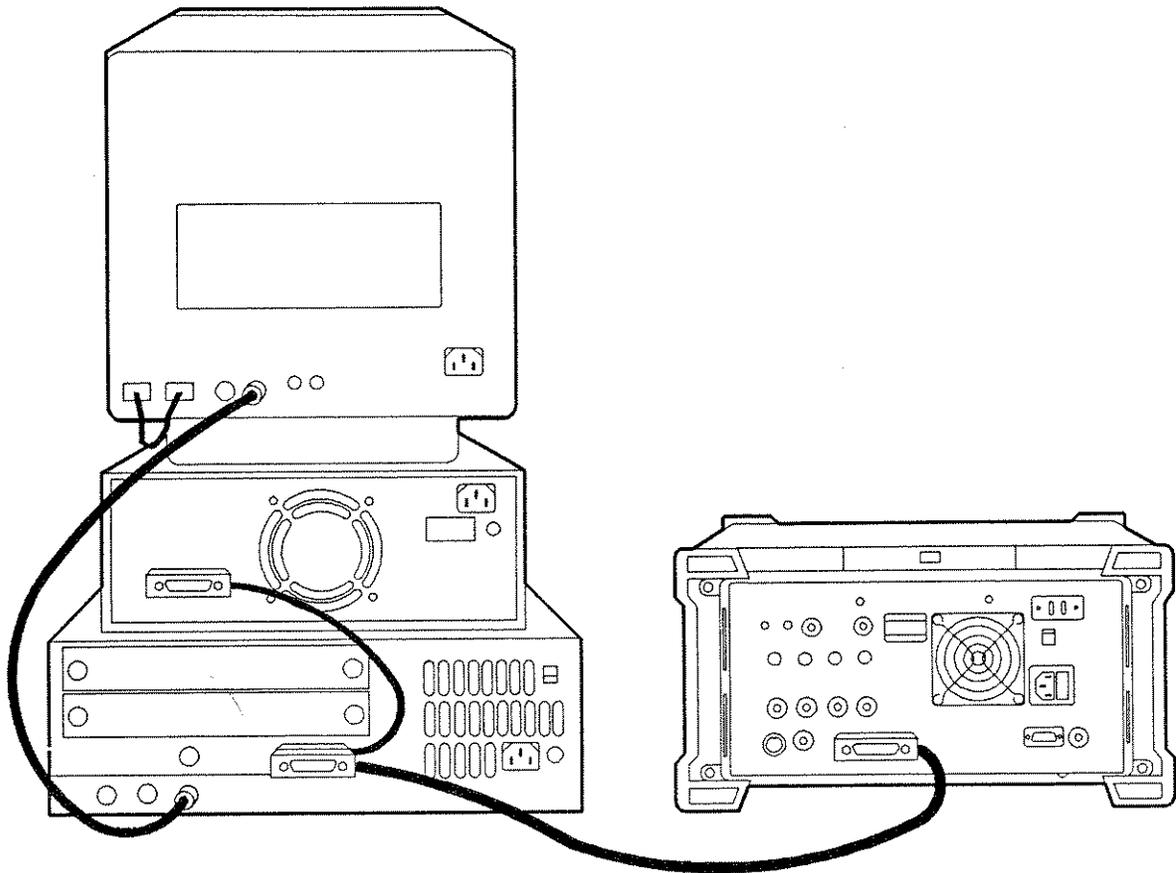


Figure 1-2. Connecting the HP 9000 Series 300 Computer to the Analyzer

## Test Program

To test the computer/analyzer interconnection, turn on your analyzer and follow the instructions below.

Your HP 9000 Series 300 computer may have either a soft-loaded or built-in language system. If your language system is built-in, remove any disks from the drives and turn on the computer.

If your language is soft-loaded, install the BASIC language disk into the proper drive. Turn the computer power on. After a few seconds, the BASIC READY message appears; the computer is now ready for use.

For further information on loading BASIC on your system, consult your BASIC manual.

Check the HP-IB address of the spectrum analyzer. Press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**. If necessary, reset the address of the analyzer: press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**, 18 **Hz** (or enter the appropriate address).

Enter the following program and press **RUN** on the computer. If you need help entering and running the program, refer to your computer and software documentation.

```

10 PRINTER IS 1
20 Analyzer=718
30 CLEAR Analyzer
40 OUTPUT Analyzer;"IP;SNGLS;"
50 OUTPUT Analyzer;"CF 300MZ;TS;"
60 OUTPUT Analyzer;"CF?;"
70 ENTER Analyzer;A
80 PRINT "CENTER FREQUENCY = ";A;"Hz";
90 END

```

The program tells the analyzer to perform an instrument preset and enter single-sweep mode. Next, the program sets the center frequency to 300 MHz and takes a sweep. The program then queries the center frequency value and the computer displays CENTER FREQUENCY = 3.0E+8 Hz. The program shows that the computer is able to send instructions to, and read information from, the analyzer.

If the computer does not display the center frequency, refer to "If There is a Problem" at the end of this chapter.

## HP Vectra Personal Computer

### With the HP-IB Interface

#### Equipment

- HP Vectra Personal Computer, Required Option: HP 82300B, HP BASIC Language Processor.
- HP 8590 Series Spectrum Analyzer, Option 021.
- HP 10833 (or equivalent) HP-IB cable.

#### Interconnection Instructions

Connect the spectrum analyzer to the computer using the HP-IB cable as shown in Figure 1-3.

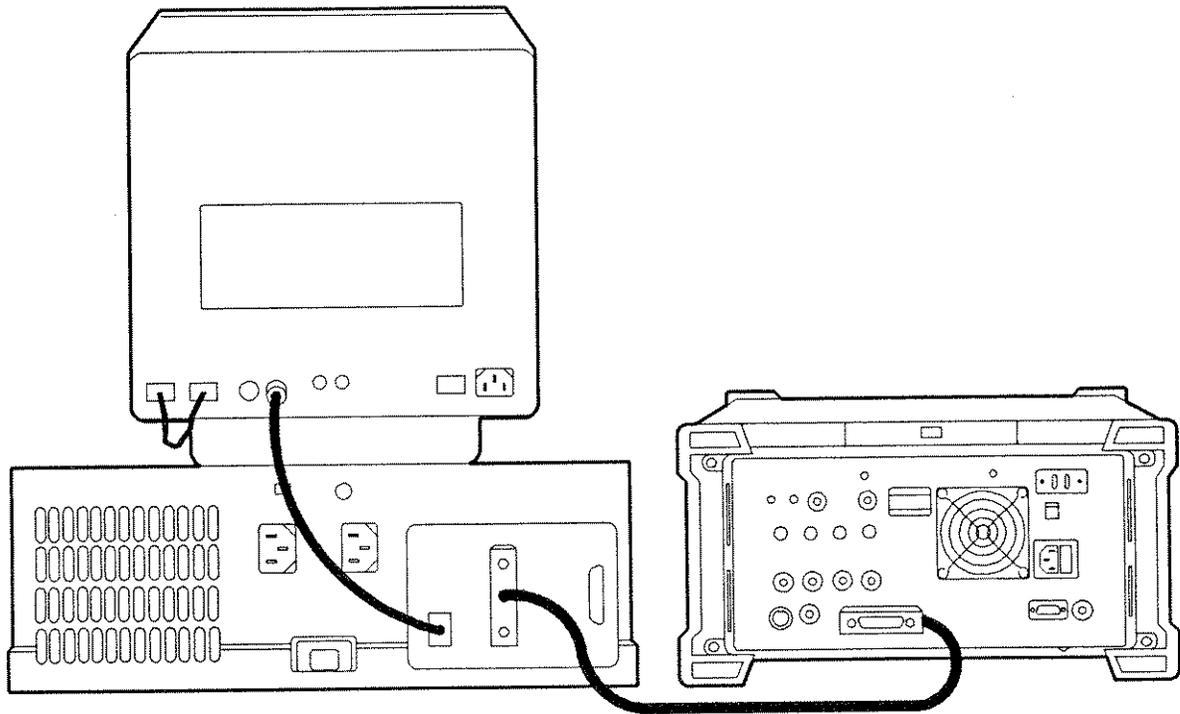


Figure 1-3. Connecting the HP Vectra Personal Computer to the Analyzer

## Test Program

To test the computer/analyzer interconnection, turn on your analyzer and follow the instructions below.

Refer to the HP 82321 Language Processor documentation to install the language processor board in your Vectra and load the BASIC programming language into your Vectra.

Check the HP-IB address of the spectrum analyzer. Press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**. If necessary, reset the address of the analyzer: press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**, 18 **Hz** (or enter the appropriate address).

Enter the following program and press **F10** on the computer. If you need help entering and running the program, refer to your computer and software documentation.

```

10 PRINTER IS 1
20 Analyzer=718
30 CLEAR Analyzer
40 OUTPUT Analyzer;"IP;SNGLS;"
50 OUTPUT Analyzer;"CF 300MZ;TS;"
60 OUTPUT Analyzer;"CF?;"
70 ENTER Analyzer;A
80 PRINT "CENTER FREQUENCY = ";A;"Hz";
90 END

```

The program tells the analyzer to perform an instrument preset and enter single-sweep mode. Next, the program sets the center frequency to 300 MHz and takes a sweep. The program then queries the center frequency value and the computer displays **CENTER FREQUENCY = 3.0E+8 Hz**. The program shows that the computer is able to send instructions to, and read information from, the analyzer.

If the computer does not display the center frequency, refer to "If There is a Problem" at the end of this chapter.

---

## HP Vectra Personal Computer

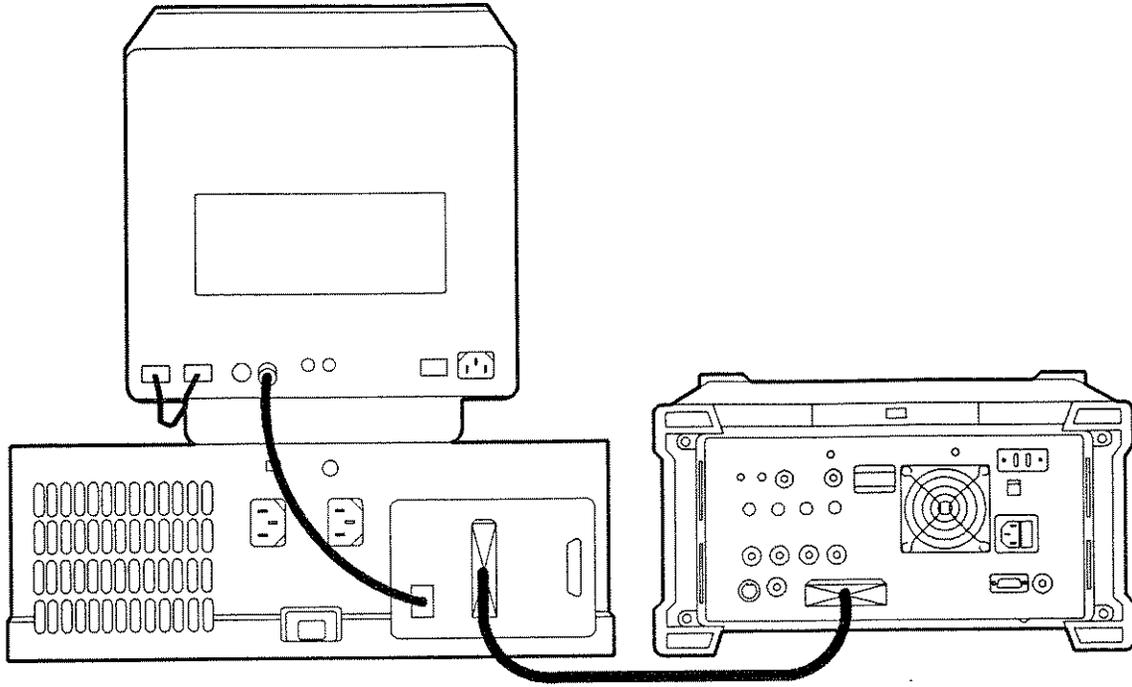
### With the RS-232 Interface

#### Equipment

- HP Vectra Personal Computer with RS-232 interface (with a 9-pin female port).
- HP 8590 Series Spectrum Analyzer, Option 023.
- HP 24542G RS-232 cable.

## Interconnection Instructions

1. Connect the spectrum analyzer to the computer using the RS-232 cable as shown in Figure 1-4.



**Figure 1-4. Connecting the HP Vectra Personal Computer to the Analyzer**

2. Turn on the analyzer and the computer.

## Test Program

The program shown below is written to work with any of the following computers and BASIC language versions installed:

- HP Vectra PC using Vectra BASIC (HP 45952A). The MS BASIC Interpreter (HP 35190A) is compatible with Vectra BASIC.
- IBM PC/AT and compatible computers using BASICA (version 2.0 or later) or GW BASIC.

To test the interconnection, first load the BASIC language for your computer and specify a communications buffer of 4096 bytes. Use the following command:

```
BASICA/C:4096
```

Set the analyzer baud rate to 1200, to match the baud rate set up for the computer port in the test program. (In line 20, the "1200" indicates 1200 baud for the computer port.) Refer to Appendix C for details on setting the analyzer baud rate.

Enter the following test program.

```
10 'File = TESTPGM
20 OPEN "COM1:1200,N,8,1" AS #1
30 PRINT #1,"IP;"
40 PRINT #1,"SNGLS;"
50 PRINT #1,"CF 300MZ;TS;"
60 PRINT #1,"CF?;"
70 INPUT #1,CENTER
80 PRINT,"CENTER FREQ = ";CENTER;"Hz"
90 END
```

When you have entered the program, type:

```
SAVE "TESTPGM"
```

When you are ready to run the program, turn on the analyzer and run your program.

The program tells the analyzer to perform an instrument preset and enter single sweep mode. Next, the program sets the center frequency to 300 MHz and takes a sweep. The program then queries the center frequency value and the computer displays `CENTER FREQUENCY = 3.0E+8 Hz`. The program shows that the computer is able to send instructions to, and read information from, the analyzer.

If the computer does not display the center frequency, refer to "If There is a Problem" at the end of this chapter.

---

## IBM PC/AT and Compatible Computers

### With the RS-232 Interface

#### Equipment

- IBM PC/AT or compatible with RS-232 interface.
- HP 8590 Series Spectrum Analyzer, Option 023.
- HP 13242G, RS-232 cable (DCE-DCE), 7 pins used (refer to Appendix C for wiring of this cable).

---

#### Interconnection Instructions

1. Connect the analyzer to the computer with the RS-232 cable. (See Figure 1-5.) The analyzer uses a female RS-232 connector; the IBM PC/AT computer usually uses a male RS-232 connector. Some compatibles use a female RS-232 connector.

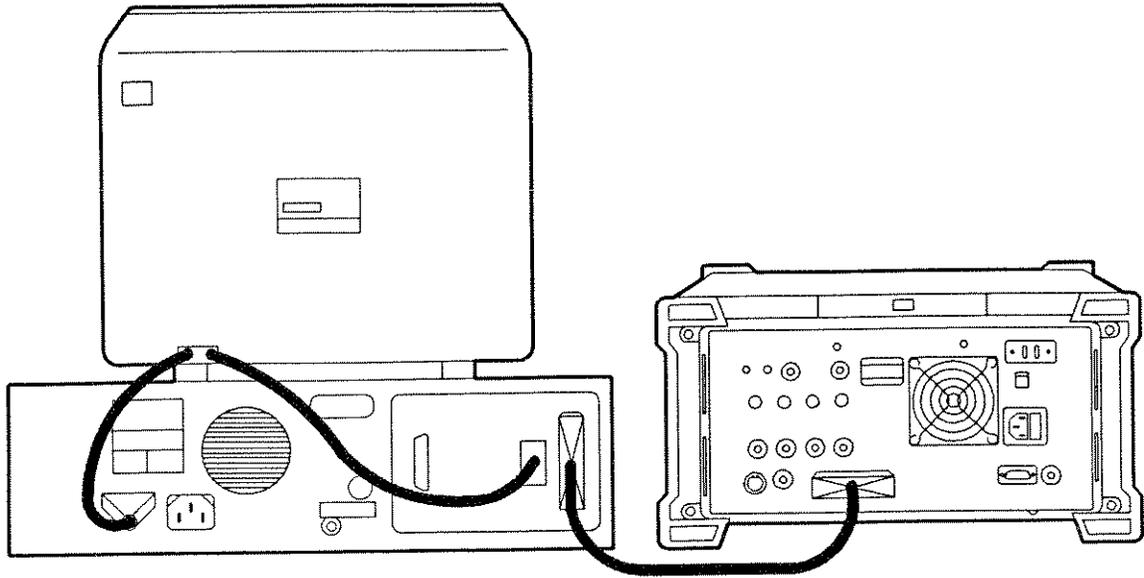


Figure 1-5. Connecting an IBM PC/AT Compatible Computer to the Analyzer

2. Turn on the analyzer and the computer.

### Test Program

The program shown below is written to work with BASICA (version 2.0 or later) or GW BASIC.

To test the interconnection, first load the BASIC language for your computer and specify a communications buffer of 4096 bytes. Use the following command:

```
BASICA/C:4096
```

Set the analyzer baud rate to 1200, to match the baud rate set up for the computer port in the test program. (In line 20, the "1200" indicates 1200 baud for the computer port.)

To set the baud rate to 1200:

1. Press **CONFIG**, **MORE 1 of 2**.
2. Press the **BAUD RATE** softkey.
3. Press these keys: 1200 **Hz**.

Enter the following test program.

```
10 'File = TESTPGM
20 OPEN "COM1:1200,N,8,1" AS #1
30 PRINT #1,"IP;"
40 PRINT #1,"SNGLS;"
50 PRINT #1,"CF 300MZ;TS;"
60 PRINT #1,"CF?;"
70 INPUT #1,CENTER
80 PRINT,"CENTER FREQ = ";CENTER;"Hz"
```

90 END

When you have entered the program, type:

SAVE "TESTPGM"

When you are ready to run the program, turn on the analyzer and run your program.

The program tells the analyzer to perform an instrument preset and enter single-sweep mode. Next, the program sets the center frequency to 300 MHz and takes a sweep. The program then queries the center frequency value and the computer displays CENTER FREQUENCY = 3.0E+8 Hz. The program shows that the computer is able to send instructions to, and read information from, the analyzer.

If the computer does not display the center frequency, refer to "If There is a Problem" at the end of this chapter.

---

## Connecting Your Analyzer to a Printer or Plotter

You may wish to obtain a permanent record of data displayed on the analyzer screen. This can be done using the **COPY** key of the analyzer, and a printer or plotter.

### To Print with an HP-IB Interface

#### Equipment

- HP 8590 Series Spectrum Analyzer, Option 021.
- HP 2225 ThinkJet Printer or HP 3630A PaintJet Color Printer.
- HP 10833 (or equivalent) HP-IB cable.

#### Interconnection Instructions

1. Turn off the printer to be connected, and the analyzer.
2. Connect the printer to the analyzer using the HP-IB cable.

---

#### Note



The HP-IB cables allow a "party line" to exist between equipment, since the HP-IB cables can be daisy-chained together. A printer and a plotter can be connected to the analyzer using two HP-IB cables. Do not connect more than one device at the same address.

---

#### Note



Since the analyzer cannot print or plot with two controllers (the computer and the analyzer) connected, the computer must be disconnected from the HP-IB.

3. Turn on the analyzer and printer.
4. On the analyzer, press **CONFIG**, **PRINT CONFIG**.

5. The printer usually resides at the first device address. To enter address 1 for the printer, press **PRINTER ADDRESS**, 1 **(Hz)**.
6. If the analyzer is connected to a PaintJet printer and you want a color printout, press **PAINTJET PRINTER**. If the analyzer is connected to a PaintJet printer and you want a black and white printout, press **B & W Printer**.
7. If you want the softkey labels to be printed with the analyzer display printout, press **PRT MENU ON OFF** so that ON is underlined.
8. Press **PREV MENU**, **COPY PRNT PLT** (PRNT should be underlined), and **(COPY)**.

## To Plot with an HP-IB Interface

### Equipment

- HP 8590 Series Spectrum Analyzer, Option 021.
- HP 7440A ColorPro Plotter.
- HP 10833 (or equivalent) HP-IB cable.

### Interconnection Instructions

1. Turn off the plotter to be connected, and the analyzer.
2. Connect the plotter to the analyzer using the HP-IB cable.

#### Note



The HP-IB cables allow a “party line” to exist between equipment, since the HP-IB cables can be daisy-chained together. A printer and a plotter can be connected to the analyzer using two HP-IB cables. Do not connect more than one device at the same address.

#### Note



Since the analyzer cannot print or plot with two controllers (the computer and the analyzer) connected, the computer must be disconnected from the HP-IB.

3. Turn on the analyzer and the plotter.
4. On the analyzer, press **(CONFIG)**, **PLOT CONFIG**.
5. The plotter usually resides at the fifth device address. To set the plotter address, press **PLOTTER ADDRESS**, 5 **(Hz)**, to enter the address 5 for the plotter.
6. With **PLTS/PG 1 2 4**, you can choose a full-page, half-page, or quarter-page plot. Press **PLTS/PG 1 2 4** to underline the number of plots per page desired.
7. If two or four plots per page are chosen, a function is displayed that allows you to select the location of the plotter output on the paper. If two plots per page are selected, **PLT [ ]LOC \_ \_** function is displayed. If four plots per page are selected **PLT [ ] \_LOC \_ \_** is displayed. Press the softkey until the rectangular marker is in

the desired section of softkey label. The upper and lower sections of the softkey label graphically represent where the plotter output will be located.

**Note**

For a multi-pen plotter, the pens of the plotter draw the different components of the screen as follows:

Pen Number	Description
1	Draws the annotation and graticule.
2	Draws trace A.
3	Draws trace B.
4	Draws trace C and the display line.
5	Draws user-generated graphics.

8. Press **PREV MENU**, **COPY DEV PRNT PLT** (PLT should be underlined), and **COPY**.

**Note**

Once the address of the printer and plotter have been entered, the analyzer remembers these addresses even though the power is turned off. There is no need to reenter them when the analyzer is turned off and on.

## To Print with an RS-232 Interface

### Equipment

- HP 8590 Series Spectrum Analyzer, Option 023.
- HP 2225 ThinkJet Printer with an RS-232 interface, or HP 3630A PaintJet Color Printer with an RS-232 interface.

**Note**

Refer to Appendix C of this manual for the appropriate RS-232 cable connectors.

## Interconnection Instructions

1. Turn off the analyzer and the printer to be connected to the analyzer.

**Note**

The RS-232 interface allows only one device (the printer or the plotter) to be connected to the analyzer.

2. Connect the device using an RS-232 cable.
3. Turn on the analyzer and printer.

4. Turn on the power to the analyzer.
5. Press **CONFIG**, **MORE 1 of 2**.
6. To set the baud rate to 9600 baud, press **BAUD RATE**, 9600 **(Hz)**. To set the baud rate to 1200 baud, press: **BAUD RATE**, 1200 **(Hz)**.

**Note**

Some of the programs in this manual utilize 1200 baud. If your system uses the RS-232 handshake lines, you can use 9600 baud for all of the programs.

7. Press **MORE 2 of 2**, **PRINT CONFIG**.
8. If the analyzer is connected to a PaintJet printer and you want a color printout, press **PAINTJET PRINTER**. If the analyzer is connected to a PaintJet printer and you want a black and white printout, press **B & W Printer**.
9. If you want the softkey labels to be printed with the analyzer display print out, press **PRT MENU ON OFF** so that ON is underlined.
10. Press **PREV MENU**, **COPY DEV PRNT PLT** (PRNT should be underlined), and **(COPY)**.

**To Plot with an RS-232 Interface****Equipment**

- HP 8590 Series Spectrum Analyzer, Option 023.
- HP 7440A ColorPro Plotter with an RS-232 interface.

**Note**

Refer to Appendix C of this manual for the appropriate RS-232 cable connectors.

**Interconnection Instructions**

1. Turn off the analyzer and the plotter to be connected to the analyzer.

**Note**

The RS-232 interface allows only one device (the printer or the plotter) to be connected to the analyzer.

2. Connect the device using an RS-232 cable.
3. Turn on the analyzer and the plotter.
4. Press **CONFIG**, **MORE 1 of 2**.
5. To set the baud rate to 9600 baud, press **BAUD RATE**, 9600 **(Hz)**. To set the baud rate to 1200 baud, press: **BAUD RATE**, 1200 **(Hz)**.

**Note**

Some of the programs in this manual utilize 1200 baud. If your system uses the RS-232 handshake lines, you can use 9600 baud for all of the programs.

6. Press **MORE 2 of 2**, **PLOT CONFIG**. You can choose a full-page, half-page, or quarter-page plot with the **PLTS/PG 1 2 4** softkey. Press **PLTS/PG 1 2 4** to underline the number of plots per page desired.
7. If two or four plots per page are chosen, a function is displayed that allows you to select the location of the plotter output (on the paper). If two plots per page are selected, **PLT [ ]LOC \_ \_** function is displayed. If four plots per page are selected **PLT [ ]\_LOC \_ \_** is displayed. Press the softkey until the rectangular marker is in the desired section of softkey label. The upper and lower sections of the softkey label graphically represent where the plotter output will be located.

**Note**

For a multi-pen plotter, the pens of the plotter draw the different components of the screen as follows:

Pen Number	Description
1	Draws the annotation and graticule.
2	Draws trace A.
3	Draws trace B.
4	Draws trace C and the display line.
5	Draws user-generated graphics.

8. Press **PREV MENU**, **COPY DEV PRNT PLT** (so that PLT is underlined), and **COPY**.

**Printing after Plotting or Plotting after Printing**

Pressing **COPY** without pressing **CONFIG**, **COPY DEV PRNT PLT**, produces the function last entered (print or plot). To print after doing a plot, press **CONFIG**, **COPY DEV PRNT PLT** (so that PRNT is underlined), and **COPY**. To plot after printing, press **CONFIG**, **COPY DEV PRNT PLT** (so that PLT is underlined), and **COPY**.

---

## If There is a Problem

This section offers suggestions to help get your computer and spectrum analyzer working as a system. The test programs provided in this chapter let you know if the computer/analyzer interconnection is working properly. If the test program does not run, try the following suggestions:

1. You may need to modify the program syntax to work with your computer. Refer to your BASIC manual for correct syntax.
2. The program must be executed correctly. Refer to your computer manual for information about program execution.
3. Check your program for errors.

If the test program runs on the computer, but the analyzer does not respond, try the following suggestions:

1. Make sure the analyzer is turned on. If the analyzer has power, the green indicator light above the line switch is on.
2. If the display is garbled, or you want to reset the analyzer configuration to the state it was in when it was originally shipped from the factory, use the **DEFAULT CONFIG** function. To access **DEFAULT CONFIG** softkey, press **CONFIG**, **MORE 1 of 2** (the bottom softkey), **DEFAULT CONFIG** (the fourth softkey from the top of the analyzer).
3. Make sure the interface cable is connected securely. Check the interface cable for defects. Make sure the correct cable is used.
4. If you are using an HP-IB interface, the analyzer must be set to the correct address setting. Press **CONFIG**, **MORE 1 of 2**, **ANALYZER ADDRESS**.
5. If you are using the RS-232 interface, check the analyzer baud rate. Refer to Appendix C for setting the baud rate on the analyzer.

If you suspect your computer is causing the problems, check it by running a program that you know works. If your system still has problems, contact your HP salesperson. Your salesperson will be able to help solve the problem or refer you to someone who can.



## Programming Fundamentals

---

### What You'll Learn in This Chapter

This chapter introduces spectrum analyzer programming. The first section of this chapter, "Writing Your First Program," helps you write your first spectrum analyzer program and introduces programming fundamentals. The second section, "Getting Information from the Analyzer," shows how to get data out of the spectrum analyzer. A summary at the end of this chapter reviews the programming guidelines introduced.

If the computer is not connected to the analyzer, follow the instructions in Chapter 1, "Preparation for Use."

A general knowledge of the BASIC programming language and the spectrum analyzer is recommended before reading this chapter. Refer to your software documentation manuals for more information about BASIC. For reference, Chapter 4 of this manual provides spectrum analyzer commands in alphabetical order.

---

**Note**

All programming examples in this chapter for the HP-IB interface are written in HP BASIC 4.0, using an HP 9000 Series 200 Computer. For the RS-232 interface, examples are written in GW BASIC, using an HP Vectra Personal Computer or compatible controller.

---

---

### Writing Your First Program

When the analyzer has been connected to a computer via HP-IB or RS-232 interface, the computer can be used to send instructions to the analyzer. These instructions tell the analyzer such things as frequency span, resolution bandwidth, and sweep mode. If a properly selected sequence of instructions is sent to the analyzer, a measurement is made. Sequences of coded instructions are called programs.

### Composing the Program

Most spectrum analyzer programs contain several common statements, or "commands," that address the spectrum analyzer, preset it, and select its sweep mode. As an example, we will write a short program that executes only these common commands.

The following programs are for the HP-IB and the RS-232 interfaces. Note the quotation marks that contain analyzer commands in each line. Also note the semicolons at the end of each line, inserted at the end of each set of analyzer commands within the quotation marks. Using semicolons makes programs easier to read, prevents command misinterpretation, and is recommended by IEEE Standard 728.

**Note**

In commands where quotation marks occur, the computer recognizes data as character data and not BASIC programming language commands.

**Program Example for the HP-IB Interface**

```

05 !File: "IBPROG1"
10 Analyzer=718
20 CLEAR Analyzer
30 OUTPUT Analyzer;"IP;"
40 OUTPUT Analyzer;"SNGLS;TS;"
50 LOCAL 7
60 END

```

Line 10 of our program assigns a variable called "Analyzer" to our spectrum analyzer at address 718. This instruction is followed by the HP BASIC CLEAR command, which resets the analyzer on the HP-IB. With these two program lines, we have set up a clear communication path between the computer and the analyzer.

Line 30 introduces the instrument preset (IP) command, which corresponds to the **PRESET** key on the spectrum analyzer. The IP command sets all of the analog parameters of the spectrum analyzer to known values and provides a good starting point for every measurement.

**Note**

All manual functions on the spectrum analyzer have corresponding programming commands. As you continue programming, you will learn the command names that correspond to the front-panel keys and softkeys.

Line 40 activates the single-sweep mode. Most remotely controlled measurements require control of the sweep. Once SNGLS has activated the single-sweep mode, take sweep (TS) starts and completes one full sweep. TS maintains absolute control over the sweep, which is necessary for accurate computer data transfer and reduced program execution time.

Before we end the program, we return the spectrum analyzer to front-panel control with line 50, LOCAL 7. The LOCAL command corresponds to the **LOCAL** key on the front panel of the analyzer. (LOCAL 7 commands everything on the bus to go to local mode.)

Finally, in line 60, we end the program with the END command. (If you forget to include the END command, the computer will give an error message.)

Enter the program lines, press **RUN** on the computer, and watch the spectrum analyzer display as it completes each instruction.

**Program Example for the RS-232 Interface**

```

10 'File = 232PROG1
20 OPEN "COM1: 9600,N,8,1" AS #1
30 PRINT #1,"IP;"
40 PRINT #1,"SNGLS;TS;"
50 END

```

Line 20 of the program opens the RS-232 COM1:line, identifies it as #1, and sets the RS-232 parameters as follows:

```
9600 baud
no parity
8 bits/character
1 stop bit
```

Line 30 of the program introduces the instrument preset (IP) command, which corresponds to the **PRESET** key on the analyzer. The IP command sets all of the analog parameters of the analyzer to known values and provides a good starting point for every measurement. All manual functions on the analyzer have corresponding programming commands. As you continue programming, you will learn the command names that correspond to the front-panel keys and softkeys.

Line 40 activates the single-sweep mode. Most remotely controlled measurements require control of the sweep. Once SNGLS has activated the single-sweep mode, take sweep (TS) starts and completes one full sweep. The TS command maintains absolute control over the sweep, which is necessary for accurate computer data transfer and reduced program execution time. Finally, in line 50, end the program with the END command.

Make sure that the analyzer baud rate is 9600 via the **BAUD RATE** softkey. Enter the program lines, then press **RUN** on the computer. Watch the spectrum analyzer display as it completes each instruction.

---

#### Note



When using an HP 9000 Series 200 or 300 computer, END commands are necessary. Refer to your BASIC manual to determine END statement requirements for your specific computer.

---

## Modifying the Program

Remote operation of the spectrum analyzer is similar to manual operation. Remote measurements are executed by commands that correspond to front-panel keys and softkeys.

The first chapter in the spectrum analyzer operating manual shows you how to make a simple measurement using the calibration signal. We can add instructions to our program so that it will make the same measurement. (Since the manual process closely resembles that of the program, you may want to review "Making a Measurement" in the *HP 8590B/8592B Spectrum Analyzer Installation, Verification, and Operation Manual* or *HP 8591A/8593A Spectrum Analyzer Installation, Verification, and Operation Manual*.)

By inserting a few lines into the initial program, we can set functions such as the center frequency and span, and we can activate a marker to find a signal's frequency and amplitude.

## Program Example for the HP-IB Interface

First, we set the center frequency to 300 MHz. The CF command corresponds to the center frequency function and **FREQUENCY** front-panel key on the analyzer. (All spectrum analyzer commands, such as CF, are described in Chapter 4.)

Insert the following program line between lines 40 and 50:

```
41 OUTPUT Analyzer;"CF 300MZ;"
```

Next, we set the span to 200 MHz with the SP command. Add the following program line:

```
42 OUTPUT Analyzer;"SP 200MZ;"
```

Since we are controlling the sweep, we must update the spectrum analyzer display screen with the following program line:

```
43 OUTPUT Analyzer;"TS;"
```

When the program is executed, the analyzer takes one full sweep before executing the next line, line 41. Line 41 changes the center frequency to 300 MHz, and line 42 changes the span to 200 MHz.

Enter the following program line to place a marker at the highest peak on the trace with a MKPK HI command:

```
44 OUTPUT Analyzer:"MKPK HI;"
```

The completed program is shown below:

```
05 !File: "IBPROG2"
10 Analyzer=718
20 CLEAR Analyzer
30 OUTPUT Analyzer;"IP;"
40 OUTPUT Analyzer;"SNGLS;TS;"
41 OUTPUT Analyzer;"CF 300MZ;"
42 OUTPUT Analyzer;"SP 200MZ;"
43 OUTPUT Analyzer;"TS;"
44 OUTPUT Analyzer:"MKPK HI;"
50 LOCAL 7
60 END
```

Run the program to make the measurement. Watch the spectrum analyzer display as it completes each instruction. Notice that the program executes the instructions faster than is possible from the front panel.

When a certain measurement is repeated often, a computer program can save time. In addition, the computer is less likely to make an error than an operator manually entering the same instructions from the front panel.

### Program Example for the RS-232 Interface

First, we set the center frequency to 300 MHz. The CF command corresponds to the center frequency function and the **FREQUENCY** front-panel key on the analyzer. (All analyzer commands, such as CF, are described in Chapter 4.)

Insert the following program lines between lines 40 and 50 of the previous program.

```
41 PRINT #1,"CF 300MZ;"
```

Next, set the span to 200 MHz with the SP command. Add the following program line:

```
42 PRINT #1,"SP 200MZ;"
```

Since we are controlling the sweep, we must update the analyzer display with the following program line:

```
43 PRINT #1,"TS;"
```

When the program is executed, the analyzer takes one full sweep before executing the next lines. This changes the center frequency to 300 MHz and the span to 200 MHz.

Enter the following program line to place a marker at the highest peak on the trace:

```
44 PRINT #1,"MKPK HI;"
```

The completed program is shown below:

```
10 'File = 232PROG2
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"IP;"
40 PRINT #1,"SNGLS;TS;"
41 PRINT #1,"CF 300MZ;"
42 PRINT #1,"SP 200MZ;"
43 PRINT #1,"TS;"
44 PRINT #1,"MKPK HI;"
50 END
```

Run the program to make the measurement. Watch the analyzer display as it completes each instruction. When a certain measurement is repeated often, a computer program can save time. Also, the computer is much less likely to make an error than an operator manually entering the same instructions from the front panel.

### Enhancing the Program with Variables

In the last program, specific center frequency and span values were set. By modifying the program, we can cause different values to be set each time the program is run.

### Program Example for the HP-IB Interface

In the following program, the exclamation point (!) allows the words that follow to be ignored by the computer. Thus, they serve as comments in the program.

```
10 !FILE: "VAR10"
20 REAL Center_freq,Span !define the variables
30 Analyzer=718
40 CLEAR Analyzer
50 OUTPUT Analyzer;"IP;SNGLS;TS;"
60 !ask for the desired center frequency:
70 INPUT "CENTER FREQUENCY(MHz)?",Center_freq
80 !ask for the desired span:
90 INPUT "SPAN(MHz)?",Span
100 !send the center frequency and span to the
110 !analyzer and take a sweep to update the
120 !analyzer screen:
130 OUTPUT Analyzer;"CF ";Center_freq;"MZ;"
140 OUTPUT Analyzer;"SP ";Span;"MZ;"
150 OUTPUT Analyzer;"TS;"
160 !find the signal peak with peak search:
170 OUTPUT Analyzer;"MKPK HI;"
180 LOCAL 7
190 END
```

Three modifications are made to the previous program so it includes center frequency and span variables. First, using the HP BASIC REAL command, we define two variables, Center.freq and Span. The frequency and span parameters are stored in these variables. (Refer to line 20.)

Second, using the HP BASIC INPUT command, we prompt the user to enter the desired center frequency and span. The center frequency and span values are entered on the computer; since the measurement units will be entered by the program, the user does not enter them. (See lines 70 to 140.)

Third, we modify the output parameter statements so that the values stored in Center.freq and Span are sent to the analyzer. (See lines 130 to 140.)

A sweep is taken after the parameters are sent to the analyzer, to ensure that the analyzer screen is updated before the marker is placed on the highest signal peak.

### Program Example for the RS-232 Interface

In the following program, the apostrophe (') allows the words that follow to be ignored by the computer. Thus, they serve as comments in the program.

```

10 'File = 232PR0G3
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"IP;"
40 PRINT #1,"SNGLS;TS;"
50 'Ask for the center frequency and span
60 PRINT "INPUT THE CENTER FREQUENCY (MZ) ";
70 INPUT CENTER
80 PRINT "INPUT THE SPAN (MZ) ";
90 INPUT SPAN
100 'Send center freq and span to analyzer
110 'take a sweep to update screen
120 PRINT #1,"CF ";CENTER;"MZ;"
130 PRINT #1,"SP ";SPAN;"MZ;"
140 PRINT #1,"TS;"
150 'find the signal peak with peak search
160 PRINT #1,"MKPK HI;"
170 END

```

Three modifications are made to the previous program in order to include center frequency and span variables. First, we use two variables, CENTER and SPAN, to store the frequency and span parameters.

Second, using the BASIC INPUT command, we ask the computer operator to enter the desired center frequency and span. (See lines 70 and 90 in previous program example). Next, we modify the output parameter statements so that the values stored in the CENTER and SPAN variables are sent to the analyzer. (See lines 120 and 130.) A sweep is taken after the parameters are sent to the analyzer, to ensure that the analyzer screen is updated before the marker is placed on the highest signal peak.

## Getting Information from the Analyzer

The first part of this chapter demonstrated techniques for setting spectrum analyzer parameters. This section demonstrates a technique for getting information out of the analyzer.

In the second program of this chapter, we placed a marker at the highest peak of a trace. The value of the marker could be read in the upper right-hand corner of the analyzer display.

If we add some commands to that program, we can read the values of the marker from the computer.

### Program Example for the HP-IB Interface

```

10 !FILE: "MKR"
20 REAL Amp_marker,Freq_marker !define variables
30 Analyzer=718
40 OUTPUT Analyzer;"IP;"
50 !set the output format of the analyzer for
60 !real numbers:
70 OUTPUT Analyzer;"TDF P;"
80 !set the analyzer parameters:
90 OUTPUT Analyzer;"SNGLS;"
100 OUTPUT Analyzer;"CF 300MZ;"
110 OUTPUT Analyzer;"SP 200MZ;"
120 OUTPUT Analyzer;"TS;"
130 OUTPUT Analyzer;"MKPK HI;"
140 !ask the analyzer for the marker's
150 !amplitude value:
160 OUTPUT Analyzer;"MKA?;"
170 !send the amplitude value to the computer:
180 ENTER Analyzer;Amp_marker
190 !ask the analyzer for the marker's
200 !frequency value:
210 OUTPUT Analyzer;"MKF?;"
220 !send the frequency value to the computer:
230 ENTER Analyzer;Freq_marker
240 !print the amplitude and frequency:
250 PRINT "THE SIGNAL PEAK IS ";Amp_marker;
260 PRINT " dBm AT ";Freq_marker/1.E+6;" MHz"
270 !set the analyzer to continuous sweep mode:
280 OUTPUT Analyzer;"CONTS;"
290 LOCAL 7
300 END

```

First, using the HP BASIC REAL command, we define two variables, Amp\_marker and Freq\_marker. The amplitude and frequency values of the marker are stored in these variables. (See line 20.)

Second, we set the output format of the spectrum analyzer for real numbers with the analyzer's trace data format (TDF) command. (See line 70.) As in our original program, we set the center frequency and span values. A sweep is taken and the marker is placed on the trace.

Next, we ask the analyzer for the amplitude value of the marker. We have the analyzer send the marker amplitude value to the computer. Note that there can be only one analyzer query per programming line. We also ask the analyzer for the frequency value of the marker, and we have the analyzer send the marker frequency value to the computer. (See lines 100 through 230.)

Finally, we print the values on the computer screen:

```
"THE SIGNAL PEAK IS ... dBm AT ... MHz"
```

Before we end the program, we return the spectrum analyzer to continuous-sweep mode and local control.

### Program Example for the RS-232 Interface

```
10 'File = 232PROG4
20 OPEN "COM1:9600,N,8,1" AS #1
30 PRINT #1,"IP;"
40 'Set the output format of the analyzer for
50 'real numbers
60 PRINT #1,"TDF P;"
70 'set the analyzer's parameters
80 PRINT #1,"SNGLS;"
90 PRINT #1,"CF 300MZ;"
100 PRINT #1,"SP 200MZ;"
110 PRINT #1,"TS;"
120 PRINT #1,"MKPK HI;"
130 'ask the analyzer for the marker's
140 'amplitude value
150 PRINT #1,"MKA?;"
160 'get the amplitude from the analyzer
170 INPUT #1,AMPMARKER
180 'ask the analyzer for the marker's frequency value
190 PRINT #1,"MKF?;"
200 'get the frequency value from the analyzer
210 INPUT #1,FREQMARKER
220 'print the amplitude and frequency
230 PRINT "THE SIGNAL PEAK IS ";AMPMARKER;
240 PRINT " dbm AT ";FREQMARKER/1000000!;" MZ"
250 'set the analyzer to continuous sweep mode
260 PRINT #1,"CONTS;"
270 END
```

First, set the output format of the analyzer to real numbers with the analyzer's trace data format (TDF) command (line 60).

As in the original program, the center frequency and span values are set and a sweep is taken. Next, the marker is placed on the trace.

Two additional variables are used for AMPMARKER and FREQMARKER. The amplitude and frequency values of the marker are stored in these variables (lines 170 and 210). The program requests the amplitude and frequency values of the marker (lines 150 and 190). Note that there can be only one analyzer query per programming line.

Finally, the amplitude and frequency values are displayed on the computer screen:

```
"THE SIGNAL PEAK IS ... dBm AT ... MZ"
```

---

## Programming Guidelines

1. Perform the measurement manually, keeping track of the sequence of functions used.
2. In the written program, execute an instrument preset and set single-sweep mode before setting other spectrum analyzer functions.
3. Use variables for function values. List variables at the beginning of the program.
4. Activate spectrum analyzer functions in logical order. Place quotation marks around spectrum analyzer commands. Separate commands with semicolons.
5. After setting analyzer functions, execute a take sweep command before reading data or activating markers.
6. The analyzer can return only one value per programming line. Do not have more than one query per programming line.
7. Use the exclamation point (!) to include comment lines when using HP BASIC.  
Use the apostrophe (') or REM to create comment lines when using GW BASIC.  
(The use of the exclamation point and the apostrophe to create comment lines are controller/language-dependent, not interface-dependent, and may be different for your system.)



## Advanced Programming

---

### What You'll Learn in This Chapter

This chapter demonstrates advanced programming techniques. In the first section, "Controlling Trace Data with a Computer," traces are read from the spectrum analyzer and saved with the computer. In the second section, "Saving Trace Data," spectrum analyzer states are saved with the computer, then the states are returned to the spectrum analyzer. "Measuring Harmonic Distortion" uses a harmonic-distortion program to illustrate programming techniques used to develop spectrum analyzer measurements. The last section, "Trace Data Transfers," summarizes formatting trace data.

Many of the programming suggestions discussed in Chapter 2, "Programming Fundamentals," have been incorporated into the programs in this chapter.

---

**Note**

All programming examples for the HP-IB interface in this chapter are written in HP BASIC 4.0. For the RS-232 interface, examples are written in GW BASIC.

---

A general knowledge of the BASIC programming language is recommended before reading this chapter. (Refer to your software documentation manuals.) Chapter 4 of this manual, "Programming Commands," defines analyzer commands alphabetically.

---

### Controlling Trace Data with a Computer

Using sample programs, this section shows you how to read trace data and store the data with your computer.

#### Reading Trace Data

The following program, which has been annotated with comments, reads a trace from the analyzer and stores the trace data in a variable.

#### Program Example for the HP-IB Interface

```
10 !FILE: "IBPROG5"  
20 !create a 401 point trace array:  
30 REAL Trace_a(1:401)  
40 Analyzer=718  
50 OUTPUT Analyzer;"IP;"  
60 !set the output format of the analyzer for  
70 !real numbers:  
80 OUTPUT Analyzer;"TDF P;"
```

```

90 !set the analyzer parameters:
100 OUTPUT Analyzer;"SNGLS;"
110 OUTPUT Analyzer;"CF 300MZ;"
120 OUTPUT Analyzer;"SP 200MZ;"
130 OUTPUT Analyzer;"TS;"
140 OUTPUT Analyzer;"MKPK HI;"
150 !move peak to center of analyzer screen:
160 OUTPUT Analyzer;"MKCF;"
170 OUTPUT Analyzer;"TS;"
180 !ask the analyzer for trace data:
190 OUTPUT Analyzer;"TRA?;"
200 !send the trace data to the computer:
210 ENTER Analyzer;Trace_a(*)
220 OUTPUT Analyzer;"CONTS;"
230 LOCAL 7
240 END

```

Trace data can be read with the computer by making three changes to the program created in Chapter 2. First, we modify the program to create a 401-point trace array, called Trace\_a, in which the trace data will be stored. Second, the program uses the TRA command to request Trace A data. (The MKA and MKF commands from the previous program have been deleted.) Third, the analyzer sends trace A data to the variable, Trace\_a.

### Program Example for the RS-232 Interface

```

10 'File = 232PROG5
20 OPEN "COM1:9600,N,8,1" AS #1
30 'create a 401-point trace array
40 DIM TRCA(401)
50 PRINT #1,"IP;"
60 'set output format of analyzer for real numbers
70 PRINT #1,"TDF P;"
80 'set analyzer parameters
90 PRINT #1,"SNGLS;"
100 PRINT #1,"CF 300MZ;"
110 PRINT #1,"SP 200MZ;"
120 PRINT #1,"TS;"
130 PRINT #1,"MKPK;"
140 'move peak to center of analyzer screen
150 PRINT #1,"MKCF;"
160 PRINT #1,"TS;"
170 'ask analyzer for trace data
180 PRINT #1,"TRA?;"
190 'retrieve trace data from analyzer
200 FOR I=1 TO 401
210 INPUT #1,TRCA(I)
220 NEXT I
230 'set continuous-sweep mode
240 PRINT #1,"CONTS;"
250 END

```

Trace data can be read with the computer by making three changes to the program created in Chapter 2. First, we modify the program to create a 401-point trace array, called TRCA in which trace data will be stored (line 40). Second, the program uses the TRA command to request trace A data (line 180). (The MKA and MKF commands in the original program have been deleted.) Third, we will have the analyzer send the trace A data into the TRCA variable (line 210).

---

## Saving Trace Data

The trace data in the previous program can be stored on a computer disk by making three program modifications.

```

10 !FILE: "IBPROG6"
20 !create a 401 point trace array:
30 REAL Trace_a(1:401)
40 Analyzer=718
50 OUTPUT Analyzer;"IP;"
60 !set the output format of the analyzer for
70 !real numbers:
80 OUTPUT Analyzer;"TDF P;"
90 !set the analyzer parameters:
100 OUTPUT Analyzer;"SNGLS;"
110 OUTPUT Analyzer;"CF 300MZ;"
120 OUTPUT Analyzer;"SP 200MZ;"
130 OUTPUT Analyzer;"TS;"
140 OUTPUT Analyzer;"MKPK HI;"
150 !move peak to center of analyzer screen:
160 OUTPUT Analyzer;"MKCF;"
170 OUTPUT Analyzer;"TS;"
180 !ask the analyzer for trace data:
190 OUTPUT Analyzer;"TRA?;"
200 !send the trace data to the computer:
210 ENTER Analyzer;Trace_a(*)
220 !create file to store trace
230 !file is 13 records long:
240 CREATE BDAT "DATA_A",13
250 !assign path for the file:
260 ASSIGN @File TO "DATA_A"
270 !send trace data to the file:
280 OUTPUT @File;Trace_a(*)
290 OUTPUT Analyzer;"CONTS;"
300 LOCAL 7
310 !close file:
320 ASSIGN @File TO *
330 END

```

First, using the CREATE command, we create an empty file on the disk for storing the trace. The file is 13 records long. (To determine the number of records, the 401-point trace is multiplied by 8 bytes per point, the storage required for real numbers, then divided by 256 bytes per record. The result is rounded to the next largest integer.)

Next, we assign an input/output path to the file DATA\_A. Then, we send the trace data to the file. (See lines 260 through 280.) Finally, in line 320, we close the file.

---

**Note**


If a program containing the CREATE command is run twice, the computer will report an error the second time because the file already exists. To prevent this error, place an exclamation mark before the CREATE command to "comment out" the line after the first run. (See line 240.)

---

### Program Example for the RS-232 Interface

```

10 'File = 232PROG6
20 OPEN "COM1:9600,N,8,1" AS #1
30 'create a 401-point trace array
40 DIM TRCA(401)
50 PRINT #1,"IP;"
60 'set output format of analyzer for real numbers
70 PRINT #1,"TDF P;"
80 'set analyzer parameters
90 PRINT #1,"SNGLS;"
100 PRINT #1,"CF 300MZ;"
110 PRINT #1,"SP 200MZ;"
120 PRINT #1,"TS;"
130 PRINT #1,"MKPK;"
140 'move peak to center of analyzer screen
150 PRINT #1,"MKCF;"
160 PRINT #1,"TS;"
170 'ask analyzer for trace data
180 PRINT #1,"TRA?;"
190 'input the trace data to the BASIC program
200 FOR I=1 TO 401
210 INPUT #1,TRCA(I) 'data input in dBm
220 NEXT I
230 'create file to store trace on disk
240 OPEN "TRACEA" FOR OUTPUT AS #2
250 'print the trace data to the disk
260 FOR I=1 TO 401
270 PRINT #2,TRCA(I)
280 NEXT I
290 'put analyzer into continuous-sweep mode
300 PRINT #1,"CONTS;"
310 END

```

Using the OPEN command, we create an empty file on the disk for storing the trace and assign an input/output path to the file TRACEA. Then we send the trace data to the file. (See lines 260 through 280.)

Lines 20 through 220 of 232PROG6 are identical to the previous program, 232PROG5.

---

## Reading Trace Data from a Computer Disk

If we want to return trace data to the spectrum analyzer for later viewing, we must work the "saving" process in reverse. The following program reads a trace previously stored on a computer disk and stores the trace in an array variable.

### Program Example for the HP-IB Interface

```
10 !FILE: "IBPROG7"
20 !create a 401-point trace array:
30 REAL Trace_a(1:401)
40 !assign path to the file with the
50 !trace in it:
60 ASSIGN @File TO "DATA_A"
70 !enter trace into variable Trace_a:
80 ENTER @File;Trace_a(*)
90 !close file:
100 ASSIGN @File TO *
110 END
```

First, in line 30, the program creates a 401-point trace array. Then, in line 60, the program assigns a path to the trace file. Finally, in line 80, the program sends the trace data to the variable Trace\_a(\*).

### Program Example for the RS-232 Interface

```
10 'File = 232PROG7
20 OPEN "COM1:9600,N,8,1" AS #1
30 'create a 401-point trace array
40 DIM TRCA(401)
50 'assign number to file with trace data in it
60 OPEN "TRACEA" FOR INPUT AS #2
70 'enter the trace into the array
80 FOR I=1 TO 401
90 INPUT #2,TRCA(I)
100 NEXT I
110 CLOSE
120 END
```

First, in line 40, the program creates a 401-point trace array. Then, in lines 60 through 100, the program reads the disk file TRACEA and stores data in the array variable TRCA.

## Saving and Recalling Instrument States

The spectrum analyzer's control settings (or its "state") can be saved with a computer and retrieved later to streamline test sequences or repeat manual measurements. Control settings can be stored in one of nine state registers in the analyzer, in computer memory, or on a computer disk.

The first program in this section demonstrates techniques for saving an instrument state, along with its current trace A data. The second program demonstrates how the state information and the trace data is read from the computer and returned to the spectrum analyzer.

If you wish to save states in the analyzer, see the descriptions of the save state (SAVES) and recall state (RCLS) commands in Chapter 4.

### Saving the Analyzer's State

The following two programs read and store a trace from the analyzer.

#### Program Example for the HP-IB Interface

```

10 !FILE: "IBPROG8"
20 !define 202 character string:
30 DIM Learn_string$[202]
40 !create 401-point array to store trace:
50 INTEGER Trace_a(1:401)
60 Analyzer=718
70 !set output format for two byte integers:
80 OUTPUT Analyzer;"TDF B;"
90 !ask analyzer for trace data:
100 OUTPUT Analyzer;"TRA?;"
110 !send trace to the computer:
120 ENTER Analyzer USING "#,W";Trace_a(*)
130 !get learnstring from analyzer:
140 OUTPUT Analyzer;"OL;"
150 ENTER Analyzer USING "#,202A";Learn_string$
160 !create file to store trace:
170 CREATE BDAT "STATE",4
180 !assign path to the file:
190 ASSIGN @File TO "STATE"
200 !send trace to the file:
210 OUTPUT @File;Learn_string$,Trace_a(*)
220 !return output format to default mode:
230 OUTPUT Analyzer;"TDF P;"
240 !close file:
250 ASSIGN @File TO *
260 END

```

The HP-IB version stores the trace in the variable called Trace\_a(\*). The state of the analyzer is stored in the variable Learn\_string\$. These two variables are then saved in a file called STATE. Finally, the file is stored on a disk.

Using the data stored in STATE, the spectrum analyzer settings can be reset according to the saved state. Then, using the stored trace data, trace data can be viewed on the spectrum analyzer display.

Line 30 gives the dimensions of the learn string using the HP BASIC DIM command. Learn strings for the analyzer require 202 bytes of storage space. Also see the output learn string (OL) command.

Line 70 uses TDF B to format the output in binary. Binary provides the fastest data transfer and requires the least amount of memory to store data. Each data point is transferred in binary as two 8-bit bytes. The data points are in the internal representation of measurement data. (See "Trace Data Transfers" at the end of this chapter.)

When the trace and state data are sent from the analyzer to the computer, they must be formatted. Lines 120 and 150 format trace data with the HP BASIC USING command. In the formatting statement, "#" indicates that the statement is terminated when the last ENTER item is terminated. EOI (end-or-identify) and LF (line feed) are item terminators, and early termination will result in an inaccurate learn string. "W" specifies word format. "202A" indicates the size of the learn string.

Line 170 creates a file called STATE that is 4 records long. (To determine the number of records for the computer in our example, the 401-point trace is multiplied by 2 bytes per point and the 202-byte learn string is added to give 1004 bytes total. This total is divided by 256 bytes per record, resulting in 4 records.)

---

#### Note



If the program containing the CREATE command is run twice, the computer will report an error the second time because the file already exists. To prevent this, place an exclamation mark before the CREATE command to "comment out" line 170 after the program has been executed.

---

### Program Example for the RS-232 Interface

```

10 'File = 232PROG8
20 OPEN "COM1:9600,N,8,1" AS #1
30 'Define 202-character string
40 DIM LEARN$(202)
50 'Create 802-character string to store trace data
60 DIM TR1$(200),TR2$(200),TR3$(200)
70 DIM TR4$(200),TR5$(2),TR6$(200)
80 'ask analyzer for trace data in binary format
90 PRINT #1,"TDF B; TRA?;"
100 'enter trace data from analyzer
110 TR1$=INPUT(200,#1) 'first 200 characters
120 TR2$=INPUT(200,#1) 'second 200 characters
130 TR3$=INPUT(200,#1) 'third 200 characters
140 TR4$=INPUT(200,#1) 'fourth 200 characters
150 TR5$=INPUT(2,#1) 'last two characters
160 'ask for learn string from analyzer
170 PRINT #1,"OL;"
180 'get learn string from analyzer
190 LEARN$=INPUT$(202,#1)
200 'create file to store trace on disk

```

```
210 OPEN "TRACEA" FOR OUTPUT AS #2
220 'change ASCII data to integers for disk storage
230 'since ASCII 26 will put EOF on disk
240 DEFINT X,Y,I 'integer variables
250 DIM X1(202),Y1(802) 'arrays for the data
260 'first format the learn string
270 FOR I= 1 TO 202
280 'get ASCII character from string
290 L2$=MID$(LEARN$,I,1)
300 'make integer of ASCII value 0-255
310 X1(I)=ASC(L2$)
320 NEXT I
330 'format the data strings
340 T5=1 'set counter
350 TR6$=TR1$ 'set string to be converted
360 GOSUB 620 'do the conversion
370 T5=201 'set counter
380 TR6$=TR2$ 'set string to be converted
390 GOSUB 620 'do conversion
400 T5=401
410 TR6$=TR3$
420 GOSUB 620
430 T5=601
440 TR6$=TR4$
450 GOSUB 620
460 'convert last two characters
470 L2$=MID$(TR5$,1,1)
480 Y1(801)=ASC(L2$)
490 L2$=MID$(TR5$,2,1)
500 Y(802)=ASC(L2$)
510 'data is now formatted, write to disk
520 FOR I=1 TO 202
530 PRINT #2,X1(I)
540 NEXT I
550 FOR I=1 TO 802
560 PRINT #2,Y1(I)
570 NEXT I
580 'close the data file
590 CLOSE
600 GOTO 680
610 'subroutine for converting data:
620 FOR I=1 TO 200
630 L2$=MID$(TR6$,I,1) 'get ASCII character
640 Y1(T5)=ASC(L2$) 'set value in array
650 T5=T5+1
660 NEXT I
670 RETURN 'done with conversion
680 END
```

The previous program reads a trace from the analyzer, then stores it in the variable called TRCA. The state of the analyzer is stored in the variable LEARN\$. These two variables are then saved in a file called TRACEA. Finally, the file is stored on a disk.

Using the data stored in TRCA, the analyzer settings can be reset according to the saved state. Then, using the stored trace data, trace data can be viewed on the analyzer display.

Line 40 gives the dimensions of the learn string using the GW BASIC DIM command. Learn strings for the analyzer require 202 bytes of storage space. Refer to the output learn string (OL) command description in Chapter 4 for more information.

Line 90 uses TDF B to format the output in binary. Binary provides the fastest data transfer and requires the least amount of memory to store data. Each data point is transferred in binary as two 8-bit bytes. The data points are in the internal representation of measurement data. (See "Trace Data Transfers" at the end of this chapter.)

When the trace and state data is sent from the analyzer to the computer, it must be formatted. Lines 270 through 320 format the trace data.

---

## Returning the Analyzer to its Former State

The following programs read a trace stored in a file and load it into a variable.

### Program Example for the HP-IB Interface

```

10 !FILE: "IBPROG9"
20 !define 202 character string:
30 DIM Learn_string$[202]
40 !create 401 point array to store trace:
50 INTEGER Trace_a(1:401)
60 Analyzer=718
70 !assign path to the file:
80 ASSIGN @File TO "STATE"
90 !get values for Learn_string$
100 !and Trace_a(*) from disk:
110 ENTER @File;Learn_string$,Trace_a(*)
120 !send learnstring to analyzer:
130 OUTPUT Analyzer;"IP DONE;"
140 ENTER Analyzer
150 OUTPUT Analyzer;Learn_string$
160 !set single sweep mode:
170 OUTPUT Analyzer;"SNGLS;"
180 !prepare analyzer for a trace from
190 !the computer:
200 OUTPUT Analyzer;"TRA #A";
210 !send trace to the analyzer
220 OUTPUT Analyzer USING "#,W";802,Trace_a(*)
230 !view trace to see it was sent:
240 OUTPUT Analyzer;"VIEW TRA;"
250 !close file:
260 ASSIGN @File TO *
```

270 END

The HP-IB program reads a trace stored in the file STATE, then loads it into the variable Trace.a(\*).

First, the settings of the analyzer that were stored in the variable LEARN\$ are recalled. The analyzer state is changed to the same state as when the trace was stored. Then previously stored trace data is returned to the analyzer and the trace is viewed on the analyzer screen. Finally, line 220 uses the HP BASIC USING command to format the trace data.

### Program Example for the RS-232 Interface

```

10 'File = 232PROG9
20 OPEN "COM1:9600,N,8,1" AS #1
30 DEFINT X,Y,I 'integer variable
40 'define 202-character string
50 DIM LEARN$(202),X1(202)
60 'create an 802-character string to store disk data
70 DIM TR1$(200),TR2$(200),TR3$(200),TR4$(200)
80 DIM TR5$(2),TR6$(200),Y1(802)
90 'open disk file "TRACEA"
100 OPEN "TRACEA" FOR INPUT AS #2
110 'enter learn array from disk
120 FOR I=1 TO 202
130 INPUT #2,X1(I) 'get integer variable from disk
140 NEXT I
150 'enter trace data from disk
160 FOR I=1 TO 802
170 INPUT #2,Y1(I)
180 NEXT I
190 'close the disk file
200 CLOSE #2
210 'format the integer data into strings
220 'for the analyzer. See 232PROG8 for explanation
230 LEARN$="" 'null out the learn string
240 FOR I=1 TO 202 'format learn string first
250 LEARN$=LEARN$+CHR$(X1(I))
260 NEXT I
270 'format the trace data
280 I2=1 'set the counter
290 GOSUB 500 'do the conversion
300 TR1$=TR6$ 'set the string
310 I2=201
320 GOSUB 500 'do the conversion
330 TR2$=TR6$
340 I2=401
350 GOSUB 500
360 TR3$=TR6$
370 I2=601
380 GOSUB 500
390 TR4$=TR6$
400 'format last two characters

```

```

410 TR5$=""
420 TR5$=TR5$+CHR$(Y1(801))+CHR$(Y1(802))
430 'write to analyzer
440 PRINT #1,LEARN$
450 'output trace data
460 PRINT #1,"IB";TR1$;TR2$;TR3$;TR4$;TR5$;
470 PRINT #1,"VIEW TRB;"
480 GOTO 560 'end program
490 'subroutine for converting integer data to ASCII
500 TR6$="" 'set the string to a null value
510 FOR I=1 TO 200
520 TR6$=TR6$+CHR$(Y1(I2))
530 I2=I2+1
540 NEXT I
550 RETURN 'done with conversion
560 END

```

The RS-232 program, 232PROG9, reads a trace stored in the file TRACEA and loads it into the variable TRCA. This program assumes that trace data is stored on the disk from the previous program example, 232PROG8.

First, the settings of the analyzer that were stored in the variable LEARN\$ are recalled. The analyzer state is changed to the same state as when the trace was stored. Then previously stored trace data is returned to the analyzer and the trace is viewed on the analyzer screen.

---

## Measuring Harmonic Distortion

The harmonic distortion program presented here illustrates how the spectrum analyzer can be directed by a computer to make a complete measurement. Measuring the percent of total harmonic distortion is tedious when performed manually: it involves tuning to the fundamental and to each harmonic of interest, recording the amplitude of each signal, converting these amplitudes to linear units (volts), and calculating the result using a formula. The following program measures percent of total harmonic distortion automatically, quickly, and accurately.

The program operates as if we were making the measurement manually.

---

### Note



This program is designed to measure harmonics of a signal that is greater than 20 MHz.

---

**Program Example for the HP-IB Interface**

```

10 !FILE: "THD_TEST"
20 ASSIGN @Sa TO 718      ! assign IO path to analyzer
30 Variables:           ! define variables:
40 REAL Fundamental,Fund_amptd_v,Fund_amptd_dbm
50 REAL Prcnt_distort,Sum_sqr
60 INTEGER Max_harmonic,I,Number
70 !allow user to change the number of harmonics:
80 Max_harmonic=4
90 ALLOCATE REAL Harmonic_v(2:Max_harmonic)
100 ALLOCATE REAL Harmonic_dbc(2:Max_harmonic)
110 GOSUB Clearscreen   ! clear the alpha screen
120 !ask for the frequency of the fundamental:
130 OUTPUT CRT USING "4/,10X,K,3/";"***HARMONIC
DISTORTION***"
140 OUTPUT CRT USING "10X,K";"CONNECT SOURCE TO INPUT"
150 OUTPUT CRT USING "10X,K";"ENTER FUNDAMENTAL FREQUENCY IN
MHz"
160 OUTPUT CRT USING "10X,K";"WHEN READY, PRESS ENTER "
170 INPUT Fundamental
180 GOSUB Clearscreen ! clear the alpha screen
190 Fundamental:      ! write message on screen:
200 DISP "MEASURING FUNDAMENTAL"
210 !preset the analyzer, set single sweep mode, and
220 !take sweep:
230 OUTPUT @Sa;"IP; SNGLS; TS;"
240 !tune the analyzer to the fundamental freq and set
250 !20 MHz span:
260 OUTPUT @Sa;"CF ";Fundamental;"MZ;"
270 OUTPUT @Sa;"SP 20MZ; TS;"
280 !put a marker on signal peak, move marker to
290 !reference level:
300 OUTPUT @Sa;"MKPK HI; MKRL; TS;"
310 !find signal peak, activate signal track, and
320 !narrow span:
330 OUTPUT @Sa;"MKPK HI; TS;"
340 OUTPUT @Sa;"MKTRACK ON; SP 100KZ; TS;"
350 !turn off signal track:
360 OUTPUT @Sa;"MKTRACK OFF;"
370 !find the peak of the signal; move peak to center
380 !of screen:
390 OUTPUT @Sa;"AUNITS V;!" MAKE READOUT UNITS VOLTS
400 !find peak of signal; send amplitude value to
410 !computer
420 !enter the amplitude of the fundamental:
430 OUTPUT @Sa;"MKPK HI; MKA?;"
440 ENTER @Sa;Fund_amptd_v
450 !send marker frequency to the computer, enter
460 !frequency value:
470 OUTPUT @Sa;"MKF?;"

```

```

480 ENTER @Sa;Fundamental
490 !make the fundamental frequency the center freq
500 !step size:
510 OUTPUT @Sa;"MKSS;"
520 !set the fundamental frequency units to MHz:
530 Fundamental=Fundamental/1.E+6
540 Harmonics: !measure the amplitudes of the harmonics:
550 FOR Number=2 TO Max_harmonic
560 DISP "MEASURING HARMONIC #";Number
570 OUTPUT @Sa;"SP 20MZ;" !set span to 20 MHz
580 OUTPUT @Sa;"CF UP; TS;" !tune to next harmonic
590 !take second sweep to allow analyzer to move to the
600 !center frequency; find the signal peak; activate
610 !signal track:
620 OUTPUT @Sa;"TS;"
630 OUTPUT @Sa;"MKPK HI; MKTRACK ON; SP 100KZ; TS;"
640 !turn off signal track:
650 OUTPUT @Sa;"MKTRACK OFF;"
660 !find signal peak; send amplitude value to computer
670 !enter the amplitude of the harmonic:
680 OUTPUT @Sa;"MKPK HI; MKA?;"
690 ENTER @Sa;Harmonic_v(Number)
700 NEXT Number
710 !set amplitude units to dBm:
720 OUTPUT @Sa;"AUNITS DBM;"
730 !calculate the fundamental amplitude in dBm since
740 !it was measured in volts:
750 Fund_amptd_dbm=10*LGT(Fund_amptd_v^2/.05)
760 !calculate the sum of the squares of the amplitudes
770 !of the harmonics; calculate amplitudes of
780 !harmonics (dBm):
790 Sum_sqr=0
800 FOR I=2 TO Max_harmonic
810 Sum_sqr=Sum_sqr+Harmonic_v(I)^2
820 Harmonic_dbc(I)=20*LGT(Fund_amptd_v/Harmonic_v(I))
830 NEXT I
840 !calculate the percent distortion:
850 Prcnt_distort=SQR(Sum_sqr)/Fund_amptd_v*100
860 GOSUB Clearscreen !clear the alpha screen:
870 !Output_data:
880 !send data to the screen of the computer:
890 OUTPUT CRT USING "7/,1X,K";"HARMONIC DISTORTION RESULTS"
900 OUTPUT CRT USING "11X,K,DDDD.D,K";"FREQ = "
;Fundamental;" MHz"
910 OUTPUT CRT USING "11X,K,DDDD.D,K";"AMP = ";Fund_amptd_dbm;"
dBm"
920 OUTPUT CRT USING "11X,K,DDD.D,K";"2nd HARMONIC =
-";Harmonic_dbc(2);" dBc"
930 OUTPUT CRT USING "11X,K,DDD.D,K";"3rd HARMONIC =
-";Harmonic_dbc(3);" dBc"
940 FOR I=4 TO Max_harmonic

```

```

950 OUTPUT CRT USING "10X,DD,K,DDD.D,K";I;"th HARMONIC =
-";Harmonic_dbc(I);"
dBc"
960 NEXT I
970 OUTPUT CRT USING "11X,K,DDD.D,K";"TOTAL DISTORTION =
";Prct_distort;" %"
980 !
990 LOCAL 7
1000 STOP
1010 !
1020 Clearscreen: !alpha clear subroutine
1030 !the statement below presses the "CLR SCR" key on
1040 !the keyboard:
1050 OUTPUT KBD USING "#,B";255,75
1060 RETURN
1070 END

```

The program prompts the user to connect a source to the spectrum analyzer INPUT and enter the source frequency. It sets the spectrum analyzer center frequency to the value of the source, or fundamental, frequency. It measures and records the frequency and amplitude of the fundamental, then measures and records the amplitude of the second, third, and fourth harmonics. These values are used to compute percent of harmonic distortion. The result of the harmonic distortion percentage computation, plus harmonic amplitudes in dBc (decibels relative to the carrier), are displayed on the computer display. Extensive annotation has been added (after the exclamation points) to help clarify the program.

If necessary, change the number of harmonics in line 80.

### Program Example for the RS-232 Interface

```

10 'File = THDTEST
20 OPEN "COM1:9600,N,8,1" AS #1
30 'allow user to change the number of harmonics
40 MAXHARMONIC=4
50 DIM HARMONICV(10),HARMONICDBC(10)
60 'clear the screen
70 CLS
80 'ask for the frequency of the fundamental
90 PRINT "***** HARMONIC DISTORTION *****"
100 PRINT
110 PRINT "CONNECT SOURCE TO ANALYZER INPUT, THEN"
120 PRINT "ENTER FREQUENCY OF THE FUNDAMENTAL IN MHZ"
130 PRINT
140 INPUT FUNDAMENTAL
150 CLS
160 'print measuring fundamental on screen
170 PRINT "MEASURING FUNDAMENTAL"
180 'preset the analyzer, set single-sweep and
185 'take sweep
190 PRINT #1,"IP;SNGLS;TS;"
200 PRINT #1,"DONE;"
210 INPUT #1,DONE

```

```
220 'tune the analyzer to the fundamental freq and set
225 '20 MHz span
230 PRINT #1,"CF ";FUNDAMENTAL;"MHZ"
240 PRINT #1,"SP 20MZ;TS;"
250 PRINT #1,"DONE;"
260 INPUT #1,DONE
270 'put a marker on signal peak, move marker to
275 'reference level
280 PRINT #1,"MKPK HI;MKRL;TS;"
290 'find signal peak, activate signal track, and
295 'narrow span
300 PRINT #1,"MKPK HI;TS;"
310 PRINT #1,"MKTRACK ON;SP 100KZ;TS;"
320 PRINT #1,"DONE;"
330 INPUT #1,DONE
340 'turn off signal track
350 PRINT #1,"MKTRACK OFF;"
360 'find peak of signal, move peak to center of screen
370 'make units in volts
380 PRINT #1,"AUNITS V;"
390 'find peak of signal, send amplitude value to
395 'computer
400 PRINT #1,"MKPK HI;MKA?;"
410 INPUT #1,FUNDAMPTDV
420 'send marker frequency to computer, enter frequency
425 'value
430 PRINT #1,"MKF?;"
440 INPUT #1,FUNDAMENTAL
450 'make the fundamental frequency the center freq
455 'step size
460 PRINT #1,"MKSS;"
470 'set the fundamental frequency units to MHZ
480 FUNDAMENTAL=FUNDAMENTAL/1000000!
490 FOR NUMBER = 2 TO MAXHARMONIC
500 PRINT "MEASURING HARMONIC # ";NUMBER
510 'set span and tune to next harmonic
520 PRINT #1,"SP 20MZ;"
530 PRINT #1,"CF UP;TS;"
540 PRINT #1,"DONE;"
550 INPUT #1,DONE
560 'take a second sweep to allow analyzer to move to
570 'the center frequency, find the signal peak,
575 'activate the signal track
580 PRINT #1,"TS;"
590 PRINT #1,"MKPK HI;MKTRACK ON;SP 100KZ;TS;"
600 PRINT #1,"MKTRACK OFF;"
610 'find signal peak, send amplitude value to computer
620 'enter amplitude of harmonic
630 PRINT #1,"MKPK HI;MKA?;"
640 INPUT #1,HARMONICV(NUMBER)
650 NEXT NUMBER
```

```

660 'set amplitude units to dBm
670 PRINT #1,"AUNITS DBM;"
680 'calculate the fundamental amplitude in dBm since
690 'it was measured in volts
700 FUNDAMPTDDBM=10*(LOG(FUNDAMPTDV^2/.05)/2.3026)
710 'calculate the sum of the squares of the amplitudes
720 'of the harmonics, calculate amplitudes of
725 'harmonics (dBm)
730 SUMSQR=0
740 FOR I=2 TO MAXHARMONIC
750 SUMSQR=SUMSQR+HARMONICV(I)^2
760 HARMONICDBC(I)=20*(LOG(FUNDAMPTDV/HARMONICV(I))/2.3026)
770 NEXT I
780 'calculate the percent distortion
790 PRCNTDISTORT=SQR(SUMSQR)/FUNDAMPTDV*100
800 CLS
810 'output the data
820 PRINT "***** HARMONIC DISTORTION RESULTS *****"
830 PRINT "FREQUENCY = ";FUNDAMENTAL;"MHZ"
840 PRINT "AMPLITUDE = ";FUNDAMPTDDBM;" dbc"
850 FOR I=2 TO MAXHARMONIC
860 PRINT "HARMONIC # ";I;" = -";HARMONICDBC(I);" dbm"
870 NEXT I
880 PRINT "TOTAL DISTORTION = ";PRCNTDISTORT;" %"
890 END

```

The RS-232 program prompts the operator to connect a source to the analyzer INPUT and enter the source frequency. It sets the analyzer center frequency to the value of the source, or fundamental, frequency. It measures and records the frequency and amplitude of the fundamental, then measures and records the amplitude of the second, third, and fourth harmonics. These values are used to compute percent of harmonic distortion. The results of the harmonic distortion percentage computation, plus harmonic amplitude in dBc (decibels relative to the carrier), are displayed on the computer display.

If necessary, change the number of harmonics in line 40.

---

## Trace Data Transfers

Two different ways to format trace data using the TDF command were introduced earlier in this chapter (TDF P and TDF B). This section describes all the available trace data formats.

The HP 8590 Series Spectrum Analyzer provides five formats for trace data transfers: real number (P) format, binary (B) format, A-block format, I-block format, and measurement units (M) format.

## P Format

The P format allows you to receive or send trace data in a real-number format. This is the default format when the instrument is powered up. Numbers are in dBm, dBmV, dB $\mu$ V, volts, or watts. The AUNITS command can be used to specify the amplitude units. Real-number data may be an advantage if you wish to use the data later in a program. However, data transfers using P format tend to be slow and take up a lot of memory (compared to binary format, in some cases P format can take up to four times the amount of memory). Data is transferred as ASCII type.

### Example of Using the P Format

This example sends trace data to the computer using P format.

<pre> 1 REAL A(1:401) 10 OUTPUT 718;"IP;CF 300MHZ;SP 20MHZ;SNGLS;TS;" 20 OUTPUT 718;"TDF P;TRA?;"  30 ENTER 718;A(*)  40 END </pre>	<p><i>Declare an array for trace data.</i></p> <p><i>Take a measurement sweep.</i></p> <p><i>Activate the P format, output trace A data.</i></p> <p><i>The computer receives trace A data from the analyzer.</i></p>
---	--

The result is in parameter units. A parameter unit is a standard scientific unit. For the TDF P format, the parameter unit would depend on the current amplitude units (dBm, dBmV, dB $\mu$ V, V, W). Use the AUNITS command to change the units.

## B Format

The B format allows you to receive or send trace data in a binary format. The B format provides the fastest data transfer and requires the least amount of memory to store data. Each data point is transferred in binary as two 8-bit bytes. The data points are in the internal representation of measurement data. Binary data can also be easily converted into measurement data.

### Example of Using the B Format

This example sends trace data from the analyzer to the computer in B format.

<pre> 10 INTEGER Tra_binary(1:401) 20 ASSIGN @Sa_bin TO 718;FORMAT OFF 30 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;SNGLS;TS;" 40 OUTPUT 718;"TDF B;TRA?;" 50 ENTER @Sa_bin;Tra_binary(*) 60 END </pre>	<p><i>Declare an integer array.</i></p> <p><i>Assign analyzer to @Sa_bin.</i></p> <p><i>Take a measurement sweep.</i></p> <p><i>Output trace A in binary format.</i></p> <p><i>Enter trace data into array Tra_binary.</i></p>
---	--

The result is transmitted as binary information. The MDS command can be used to change the data size from one 8-bit byte to two 8-bit bytes.

---

#### Note



It is not possible to return data to the analyzer using binary format. You must use either A-block or I-block format.

---

## A-Block Format

The A-block format is similar to binary format in that each data point is sent as two 8-bit bytes (this, too, is in the internal representation of measurement data). A-block format also transfers a four-byte header before the 401 points of trace data. These bytes are the ASCII character "#", "A", and two-byte number representing the length of the trace data, followed by the data bytes.

### Example of Using the A-Block Format

This example sends trace data from the analyzer to the computer and back to the analyzer in A-block format.

```
10 INTEGER Tra_binary(1:401)
```

*Declare an array for trace data.*

```
20 DIM Header$(4)
```

*Declare a string for the #, A, MSB length, and LSB length header.*

```
30 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;SNGLS;TS;"
```

*Take a measurement sweep.*

```
40 OUTPUT 718;"MDS W;TDF A;TRA?;"
```

*Send trace A to the computer in A-block format.*

```
50 ENTER 718 USING "#,4A,401(W)";Header$,Tra_binary(*)
```

*The computer receives the header and the trace data.*

```
60 PRINT "PRESS CONTINUE TO RETURN DATA TO THE ANALYZER"
```

```
70 PAUSE
```

```
80 OUTPUT 718;"IP;TS;VIEW TRA;"
```

*View trace A.*

```
90 OUTPUT 718;"TDF A;"
```

```
100 OUTPUT 718 USING
```

```
"#,K,W,401(W)";"TRA#A",802,Tra_binary(*),";"
```

*The analyzer receives the trace data from the computer.*

```
110 END
```

The transferred trace data consists of #A, a two-byte number representing the most significant byte (MSB) length and the least significant byte (LSB) length, and the data bytes. Depending on the terminal you are using, the data bytes may appear as symbols instead of numbers. Consult your computer documentation to determine the numeric value of the data bytes.

### Note

The I-block format is not recommended for use with the RS-232 interface (Option 023).



## I-Block Format

The I-block format transfers data points as two 8-bit bytes in the internal representation of measurement data. In addition to transferring trace data, I-block format also transfers the characters "#" and "I". These characters indicate that the trace data is in I-block format. The I-block format allows the spectrum analyzer to accept up to 401 points of trace data when using I-block format. Fewer than 401 points of trace data can be specified, and the spectrum analyzer will accept data until an EOI signal is sent to it. Therefore, returning the trace data to the analyzer requires an important instruction, END. (See following example.)

## Example of Using the I-Block Format

This example sends trace data from the analyzer to the computer and back to the analyzer in I-block format.

<pre> 10 INTEGER Tra_binary(1:401) 20 DIM Header\$(2) 30 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;SNGLS;TS;" 40 OUTPUT 718;"TDF I;TRA?;" 50 ENTER 718 USING "#,2A,401(W)";Header\$,Tra_binary(*) 60 PRINT "PRESS CONTINUE TO RETURN DATA TO THE ANALYZER" 70 PAUSE 80 OUTPUT 718;"IP;TS;VIEW TRA;" 90 OUTPUT 718;"TDF I;" 100 OUTPUT 718 USING "#,K,W,401(W)";"TRA#I", Tra_binary(*)END 110 END </pre>	<p><i>Declare an array for trace data.</i></p> <p><i>Declare an array for #, I header.</i></p> <p><i>Take a measurement sweep. Send trace A data in I-block format.</i></p> <p><i>The computer receives the header and trace A data.</i></p> <p><i>View trace A.</i></p> <p><i>The trace data is returned to the analyzer.</i></p>
---	--

The END statement in line 100 sends the spectrum analyzer the last data byte stored in the array and sets the HP-IB EOI line "true," as required by the I-block format.

The transferred trace data consists of #I, followed by data bytes until the EOI line is set true.

## M Format

The M format is for sending trace data only. It formats the trace data in the internal format used by the spectrum analyzer, also known as measurement units.

Refer to Figure 3-1. The displayed amplitude of each element falls on one of 8000 vertical points with 8000 equal to the reference level. For log scale data, each point is equal to 0.01 dB. The peak of the signal in Figure 3-1 is equal to -10 dBm, or one division below the reference level. In measurement units, it is equal to 7000 ( $8000 - 1000 = 7000$ ). In linear mode, each point has a resolution of [reference level in volts/8000]. The range of internal data is -32,768 to +32,767. In practice however, the range limits can be reached during trace math operations only.

### Note



All trace math functions are done using measurement units. See Table 4-4 for a list of all trace math functions. See "AMB" in Chapter 4 for an example of trace math subtraction in measurement units.

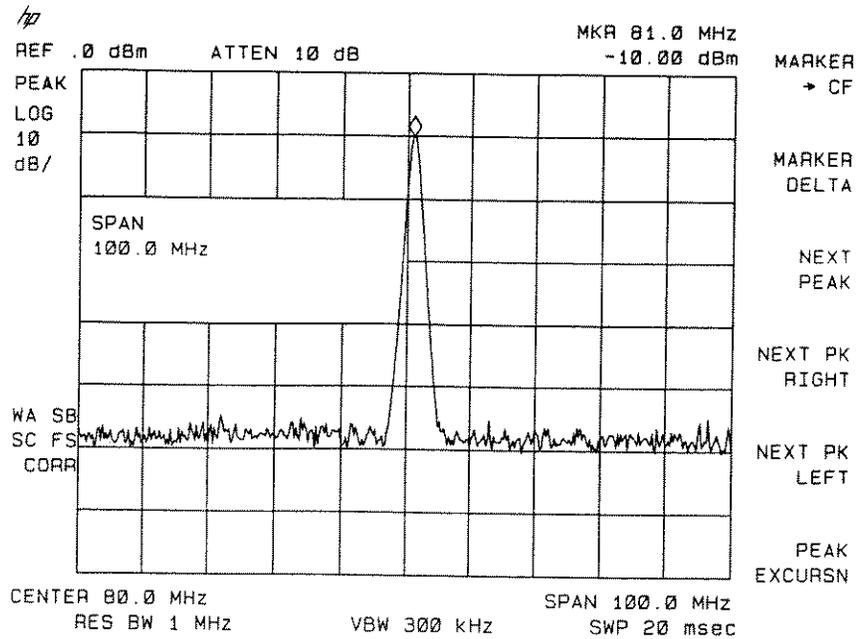


Figure 3-1. Trace Amplitude

### Example of Using the M Format

This example sends trace data from the analyzer to the computer in M format.

```

10 INTEGER A(1:401)
30 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;SNGLS;TS;"
40 OUTPUT 718;"TDF M;TRA?;"
50 ENTER 718;A(*)
60 PRINT A(*)
70 END

```

*Dimension array A.  
 Take a measurement sweep.  
 Send trace A data in M format.  
 The computer receives the trace data.  
 Print trace data.*

The result is in measurement units (-32768 to +32767).

## Programming Commands

---

### What You'll Learn in This Chapter

This chapter is a reference for the HP 8590 Series Spectrum Analyzer command language. It is a command dictionary; commands are listed alphabetically.

To find a programming command that performs a particular function, first refer to the Functional Index (Table 4-4) where commands are categorized by function. Once the desired command is found in the functional index, refer to the command in the chapter.

This chapter includes the reference tables listed below:

- Table 4-1, Syntax Elements.
- Table 4-2, Characters and Secondary Keywords (Reserved Words).
- Table 4-3, Summary of Compatible Commands.
- Table 4-4, Functional Index.
- Table 4-5, Command Cross Reference.

---

### Syntax Conventions

Command syntax is represented pictorially.

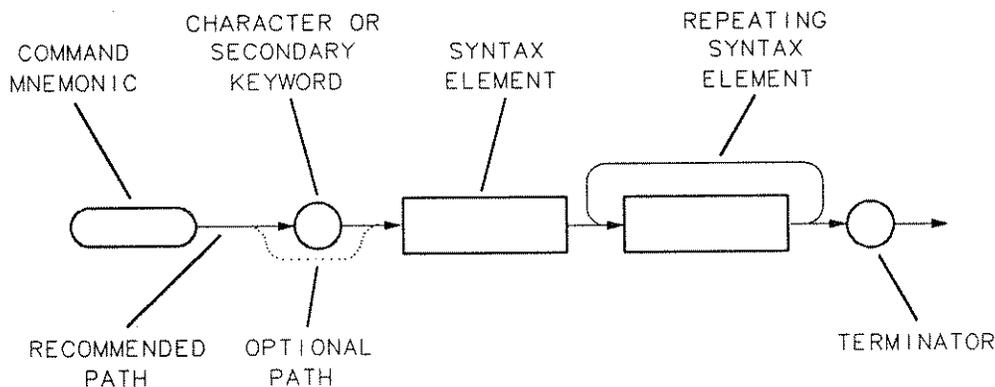


Figure 4-1. Command Syntax Figure

- Narrow ovals enclose command mnemonics. The command mnemonic must be entered exactly as shown.

- Circles and wide ovals surround secondary keywords, or special numbers and characters. The characters in circles and ovals are considered reserved words and must be entered exactly as shown. See Table 4-2.
- Rectangles contain the description of a syntax element defined in Table 4-1.
- A loop above a syntax element indicates that the syntax element can be repeated.
- Solid lines represent the recommended path.
- Dotted lines indicate a optional path for bypassing secondary keywords or using alternate units.
- Curved intersections indicate command path direction.
- Semicolons are the recommended command terminators. Using semicolons makes programs easier to read, prevents command misinterpretation, and is recommended by IEEE Standard 728.

---

**Note** Uppercase is recommended for entering all commands unless otherwise noted.



Syntax Elements are shown in the syntax diagrams as elements within rectangles.

**Table 4-1. Syntax Elements**

Syntax Component	Definition/Range
analyzer command	Any spectrum-analyzer command in this chapter, with required parameters and terminators.
character	Sp ! " # \$ % & ' ( ) + , / 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 (Characters are a subset of data byte.)
character & EOI	8-bit byte containing character data only, followed by end-or-identify (EOI) condition, where the EOI control line on HP-IB is asserted to indicate the end of the transmission. END signifies the EOI condition.
compatible function	Any spectrum-analyzer command that operates on other functions, that is, contains predefined function in its syntax diagram. (See predefined function in this table.)
data byte	8-bit byte containing numeric or character data.
data byte & EOI	8-bit byte containing numeric or character data followed by end-or-identify (EOI) condition, where the EOI control line on HP-IB is asserted to indicate the end of the transmission. END signifies the EOI condition.
delimiter	~   \ @ = < > / ^ \$ % ; ! ' ' : " Matching characters that mark the beginning and end of a character string, or a list of user-defined functions or analyzer commands.  Choose delimiting characters that are not used 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: $\pm 1.797693134862315 \times 10^{308}$ , including 0.  Up to 15 significant figures allowed. Numbers may be as small as $\pm 2.225073858507202 \times 10^{-308}$  Integer Number Range: -32,768 through +32,767
output termination	Carriage return (C <sub>R</sub> ) and line feed (L <sub>F</sub> ) with end-or-identify (EOI) condition. ASCII codes 13 (carriage return) and 10 (line feed) is sent via HP-IB, then the end-or-identify control line on HP-IB sets to indicate the end of the transmission.

Table 4-1. Syntax Elements

Syntax Component	Definition/Range
predefined function	<p>CNTLI, DONE, HAVE, HN, MEAN, MEM, MINPOS, MKBW, MSI, PEAKS, PKPOS, PWRBW, REV, RMS, SER, STB, STDEV, SUM, SUMSQR, TRCMEM, VARIANCE.</p> <p>A predefined function is an <b>analyzer command</b> that returns a number that can be operated on by other commands called <b>compatible functions</b>. Insert a <b>predefined function</b> into a command statement where predefined function appears in the command syntax chart. (See compatible function in this table.)</p> <p>If a predefined function takes a parameter (for example, PKPOS TRA), it can only be used as the last parameter of an analyzer command that has two or more predefined functions as parameters. For example, MPY V_AR,PKPOS TRB,HAVE CARD; is illegal but MPY V_AR,DONE,HAVE CARD; is not.</p>
predefined variable	<p>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.</p> <p>AMB, AMBPL, ANNOT, AT, CF, CNTLA, CNTLB, CNTLC, CNTLD, CRTHPOS, CRTVPOS, DATEMODE, DET, DL, FA, FB, FMGAIN, FOFFSET, GR, GRAT, INZ, LG, LIMIMODE, LIMIREL, LIMITEST, MEASURE, MENU, MF, MKA, MKA CT, MKF, MKFCR, MKN, MKNOISE, MKP, MKPAUSE, MKPX, ML, MKTRACK, MODE, MSI, POWERON, PSTATE, RB, RL, RLPOS, ROFFSET, SAVRCLN, SETDATE, SETTIME, SP, SQLCH, SRCALC, SRCAT, SRCNORM, SRCPOFS, SRCPSTP, SRCPSWP, SRCPWR, SRCTK, SS, ST, SWPCPL, TH, TIMEDSP, TVSTND, VB, VBR</p>
trace element	<p>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 401 for traces A, B, C, or 1 to 2047, for traces specified by TRDEF. Same for trace B (TRB[N]), trace C (TRC[N]), and user-defined traces (LABEL[N]).</p>
trace range	<p>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 401 for traces A, B, C, or 1 to the length of a trace specified by TRDEF. Same for trace B (TRB[N,M]), trace C (TRC[N,M]), and user-defined traces (LABEL[N,M]). Single-point segments are notated the same as the trace element above.</p>
units	<p>Represent standard scientific units.</p> <p>Frequency Units: GHZ or GZ, MHZ or MZ, KHZ or KZ, HZ</p> <p>Amplitude Units: DB, DM, DBMV, DBUV, V, MV, UV, W, MW, UW</p> <p>Time Units: SC, MS, US</p> <p>Current Units: A, MA, UA</p> <p>Impedance Units: OHM</p>

**Table 4-1. Syntax Elements  
Continued**

Syntax Component	Definition/Range
user-defined function	A label 2 to 11 characters long that is defined by the FUNCDEF command. Choice of characters is A through Z and the underscore (.). The underscore should be used as the second character of the label. Omitting the underscore, or using the underscore in other than the second character in a label, is not recommended.
user-defined trace	A label 2 to 11 characters long that is defined by the TRDEF command. Choice of characters is A through Z and the underscore(.). The underscore should be used as the second character of the label. Omitting the underscore, or using the underscore in other than the second character in a label, is not recommended.
user-defined variable	A label 2 to 11 characters long that is defined by the VARDEF or ACTDEF command. Choice of characters is A through Z and the underscore(.). The underscore should be used as the second character of the label. Omitting the underscore, or using the underscore in other than the second character in a label, is not recommended.

Characters and Secondary Keywords are shown within circles or ovals and must be entered exactly as shown.

**Table 4-2. Characters and Secondary Keywords (Reserved Words)**

Element	Description
a	Amplitude correction factors.
A	Amp (unit) or A-block data field.
ABSHZ	Absolute Hz (unit).
ALL	All.
AM	Amplitude modulation (Option 102 only).
AMP	Amplitude.
AMPCOR	Amplitude correction.
AUTO	Auto couple.
AVG	Average.
B	8-bit byte or binary format.
BOTH	Both odd and even frames trigger (Options 101 and 102 only).
BW	Black and white.
CARD	Memory card.
COLOR	Color.
CNT	Counter-lock (HP 8591A or HP 8593A only).
d	Downloadable programs.
DB	Decibel (unit).
DBM	Absolute decibel milliwatt (unit).
DBMV	Decibel millivolt (unit).
DBUV	Decibel microvolt (unit).
DELTA	Delta.
DISP	Display.
DLP	Downloadable program.
DM	Absolute decibel milliwatt (unit).
DMY	Day, month, year format.
DN	Decreases parameter one step size.
DUMP	Dump.
EP	Pauses program for data entry from front panel.
EQ	Equal to.
EVEN	Even video frame (Options 101 and 102 only).
EXT	External trigger.
FADC	Fast ADC (Option 101 only).
FETCH	Fetch.
FIXED	Fixed.
FFT	Fast Fourier transform.
FLAT	Flat.
FLATTOP	Flat top filter window.
FMD	FM demodulator (Option 102 only).
FM	Frequency modulation (Option 102 only).
FMV	Frequency modulation detection (Option 102 only).
FREE	Free run.
FREQ	Frequency.

Table 4-2.  
**Characters and Secondary Keywords (Reserved Words) (continued)**

Element	Description
FRQ	Frequency.
GE	Greater than or equal to.
GHZ	Gigahertz (unit).
GT	Greater than.
GZ	Gigahertz (unit).
HANNING	Hanning filter window.
HI	Highest.
HPIB	HP-IB (Option 021 only).
HZ	Hertz (unit).
I	I-block data field.
INIT	Initialize.
INT	Internal.
INTEGER	Integer.
IP	Instrument preset.
IST	Inverse sweep time.
K	Free field ASCII format with no terminator.
KC	Free field ASCII format with "CR" an "LF" terminator.
KHZ	Kilohertz (unit).
KL	Free field ASCII format with "CR" an "END" terminator.
KZ	Kilohertz (unit).
l	Limit-line.
LAST	Last state.
LE	Less than or equal to.
LIMILINE	Limit-line.
LINE	Line trigger.
LLCMD	Limit-line command.
LOWER	Lower limit-line.
LT	Less than.
M	Measurement units.
MA	Milliamp (unit).
MDY	Month, day, year format.
MHZ	Megahertz (unit).
MS	Millisecond (unit).
MTR	Meter.
MV	Millivolts (unit).
MW	Milliwatt (unit).
MZ	Megahertz (unit).
NE	Not equal to.
NEG	Negative.
NH	Next highest peak.
NL	Next peak left.
NONE	No units.
NR	Next peak right.
NRM	Normal.
NTSC	NTSC video format.

Table 4-2.  
 Characters and Secondary Keywords (Reserved Words) (continued)

Element	Description
OA	Output amplitude.
ODD	Odd video frame trigger (Options 101 and 102 only).
OFF	Turn function off.
ON	Turn function on.
P	Parameter units.
PAL	PAL video format.
PALM	PAL-M video format.
PER	Period.
PKAVG	Peak average.
PKPIT	Peak pit.
POINT	Point.
POS	Positive.
PSN	Position.
RECALL	Recall operation.
RS232	RS-232 interface (Option 023).
s	State.
SA	Signal analysis.
SAVE	Save operation.
SC	Seconds (unit).
SECAML	SECAM-L video format.
SLOPE	Slope.
SMP	Sample detection mode.
SP	Space.
SR	Stimulus response.
STATE	State register.
STEP	Step key ability.
STORE	Store.
SWT	Sweep time.
t	Trace.
TG	Tracking generator.
TRA	Trace A.
TRB	Trace B.
TRC	Trace C.
TV	TV trigger (Options 101 and 102 only).
UA	Microamp (unit).
UNIFORM	Uniform filter window.
UP	Increases the parameter one step size.
UPLOW	Upper and lower limit-lines.
UPPER	Upper limit-line.
US	Microseconds (unit).
UV	Microvolts (unit).
UW	Microwatt (unit).
V	Volts (unit).
VERTICAL	Vertical triggering (Options 101 and 102 only).
VID	Video trigger.

Table 4-2.  
Characters and Secondary Keywords (Reserved Words) (continued)

Element	Description
W	Watts or Word (for MDS command).
YTF	YIG-tuned filter.
XTAL	Crystal.
*	Asterisk (wildcard).
;	Semicolon (ASCII code 59).
,	Comma (ASCII code 44).
0	Off. Command argument.
1	On. Command argument.
50	50Ω.
75	75Ω.
?	Returns a query response containing the value or state of the associated parameter. The query response is followed by a carriage-return/line-feed.

The alternate commands listed in the left-hand column are provided if backward compatibility with commands in HP 8566A/B, HP 8568A/B, and HP 70000 Series instruments is required. The equivalent commands for the HP 8590 Series Spectrum Analyzer are listed in the right-hand column.

**Table 4-3. Summary of Compatible Commands**

Alternate Commands	Description	HP 8590 Series Command
A1	Clear write trace A	CLRW TRA
A2	Max hold trace A	MXMH TRA
A3	Store and view trace A	VIEW TRA
A4	Store and blank trace A	BLANK TRA
B1	Clear write trace B	CLRW TRB
B2	Max hold trace B	MXMH TRB
B3	Store and view trace B	VIEW TRB
B4	Store and blank trace B	BLANK TRB
BL	B - DL -> B	BML
C1	A - B off	AMB OFF
C2	A - B on	AMB ON
CA	Coupled input attenuation	AT AUTO
CR	Coupled resolution bandwidth	RB AUTO
CS	Coupled step size	SS AUTO
CT	Coupled sweep time	ST AUTO
CV	Coupled video bandwidth	VB AUTO
E1	Peak search	MKPK HI
E2	Enter marker into center frequency	MKCF
E3	Enter marker delta into center frequency step size	MKSS
E4	Enter marker amplitude into reference level	MKRL
EM	Erase graphics memory	CLR DSP
EX	Exchange trace A and B	AXB
KSA	dBm amplitude units	AUNITS DBM
KSB	dBmV amplitude units	AUNITS DBMV
KSC	dB $\mu$ V amplitude units	AUNITS DBUV
KSD	Volt amplitude units	AUNITS V
KSE	Screen title	TITLE
KSG	Video average on	VAVG ON
KSH	Video average off	VAVG OFF
KSM	Marker noise	MKNOISE
KSO	Marker value to span	MKSP
KSZ	Reference level offset	ROFFSET
KSc	A + B -> A	APB
KSi	Exchange trace B and C	BXC
KSl	Trace B -> trace C	BTC
KSm	Graticule off	GRAT OFF
KSn	Graticule on	GRAT ON
KSo	Annotation off	ANNOT OFF
KSp	Annotation on	ANNOT ON

Table 4-3. Summary of Compatible Commands (continued)

Alternate Commands	Description	HP 8590 Series Command
L0	Display line off	DL OFF
M1	Marker off	MKOFF
M2	Marker normal	MKN
M3	Marker delta	MKD
MA	Marker amplitude	MKA?
MC	Marker count	MKFC
MT0	Marker track off	MKTRACK OFF
MT1	Marker track on	MKTRACK ON
O1	Output format—real	TDF P
O2	Output format—word	TDF B;MDS W
O3	Output format—measurement unit	TDF M
O4	Output format—byte	TDF B;MDS B
R1	Activates illegal command service request only	RQS 32
R2	Activates end-of-sweep, illegal command	RQS 36
R3	Activates broken hardware, illegal command	RQS 40
R4	Activates units-key pressed, illegal command	RQS 34
RC	Recall state	RCLS
S1	Sweep continuous	CONTS
S2	Sweep single	SNGLS
SV	Save state	SAVES
T0	Threshold off	TH OFF
T1	Trigger mode free run	TM FREE
T2	Trigger mode line	TM LINE
T3	Trigger mode external	TM EXT
T4	Trigger mode video	TM VID

This functional index categorizes the commands by function. The index organizes the programming commands according to the functions that can be accessed with a front-panel key or the type of function the command performs. Use the "Corresponding Key Function" column to identify the command that is similar to front-panel or softkey function. The command and a brief definition are shown. Once the desired command is found, refer to the alphabetical listing of commands later in this chapter for further definition.

Table 4-4. Functional Index

Function Category	Command	Corresponding Key Function	Description
AMPLITUDE	AMPCOR		Applies amplitude corrections at specified frequencies.
	AT	ATTEN AUTO MAN	Specifies input attenuation.
	AUNITS	AMPTD UNITS	Specifies amplitude units for input, output, and display.
	INZ	INPUT Z 50 75	Specifies input impedance.
	LG	SCALE LOG LIN (LOG)	Selects log scale.
	LN	SCALE LDG LIN (LIN)	Selects linear scale.
	ML	MAX MXR LEVEL	Specifies mixer level.
	NRL		Normalizes trace data with respect to the reference level.
	RESETRL		Resets the reference level to instrument preset value.
	PP*	PRESEL PEAK *	Performs a preselector peak.
RL	REF LVL	Specifies reference level.	
ROFFSET	REF LVL OFFSET	Specifies reference level offset.	
AUTO COUPLING	AUTO	AUTO ALL	Recouples active function or recouples all functions.
AUXILIARY CONTROL (AUX CTRL)	CNTLA	CNTL A 0 1	Turns control line A on or off.
	CNTLB	CNTL B 0 1	Turns control line B on or off.
	CNTLC	CNTL C 0 1	Turns control line C on or off.
	CNTLD	CNTL D 0 1	Turns control line D on or off.
	CNTLI	CNTLI	Returns the status of control line I.
	COMB*	COMB GEN ON OFF *	Turns comb generator on or off.
* For HP 8592B or HP 8593A only.			

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
AUXILIARY CONTROL (AUX CTRL)	DEMOM*	DEMOM ON OFF*, DEMOM AM FM*	Turns the demodulator on or off, selects between AM and FM demodulation.
	FMGAIN*	FM GAIN*	Specifies frequency for FM gain.
	MEASURE†		Determines the type of measurement: signal analysis, stimulus response, or signal normalization.
	NRL		Sets the normalized reference level.
	RLPOS		Selects the position of reference level.
	SPEAKER*	SPEAKER ON OFF*	Turns the internal speaker on or off.
	SQLCH*	SQUELCH*	Sets the squelch level.
	SRCALC†	ALC MTR INT XTAL †	Selects internal or external leveling for the tracking generator.
	SRCAT†	SRC ATH MAN AUTO †	Attenuates the source output level.
	SRCNORM		Subtracts trace B from trace A, adds the display line, and sends the result to trace A.
	SRCPOFS†	SRC PWR OFFSET †	Offsets the source power level.
	SRCPSTP†	SRC PWR STP SIZE †	Selects the source-power step size.
	SRCPSWP†	PWR SWP ON OFF †	Selects sweep range of source output.
	SRCPWR†	SRC PWR ON OFF †	Selects the source power level.
	SRCTK†	MAN TRK ADJUST †	Adjusts tracking of source output with spectrum-analyzer sweep.
	SRCTKPK†	TRACKING PEAK †	Adjusts tracking of source output with analyzer sweep.
SWPCPL†	SWP CPLG SR SA †	Selects a stimulus-response (SR) or spectrum-analyzer (SA) auto-coupled sweep time.	
* For Option 102 only.			
† For Option 010 or 011 only.			

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
BANDWIDTH (BW)	RB	RES BW AUTO MAN, 9 kHz EMI BW, 120 kHz EMI BW	Specifies resolution bandwidth.
	VAVG	VID AVG ON OFF	Turns video averaging on or off.
	VB	VID BW AUTO MAN	Specifies video bandwidth.
	VBR	VBW/RBW RATIO	Specifies coupling ratio of video bandwidth to resolution bandwidth.
CALIBRATION (CAL)	CAL	<u>CAL</u> calibration functions	Initiates calibration routines.
	CNF	CONF TEST	Performs the confidence test. Returns a "1" if the correction factors are on.
	CRTHPOS	CRT HORZ POSITION	Specifies the CRT horizontal position.
	CRTVPOS	CRT VERT POSITION	Specifies the CRT vertical position.
COMMAND TRIGGER	ONCYCLE		Performs command list periodically.
	ONDELAY		Performs command list once after a time period.
	ONEOS		Performs command list on end of every sweep.
	ONMKR		Performs command list at the marker.
	ONSRQ		Performs command list on every service request.
	ONSWP		Performs command list at beginning of every sweep.
	ONTIME		Performs command list at a specific time.
CONFIGURATION (CONFIG)	CAT	CATALOG CARD *	Displays directory information from the specified or current mass storage device.
	DATEMODE	DATEMODE MDY DMY	Sets the format of the display of the date.
	DISPOSE	DISPOSE USER MEM	Deletes user-defined functions.
	FORMAT*	FORMAT CARD *	Formats the memory card.
	PREFX	CHANGE PREFIX	Specifies the prefix.
	SETDATE	SET DATE	Sets the date of the real-time clock.
* Option 003 required for HP 8590B or HP 8592B.			

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
CONFIGURATION (CONFIG)	SETTIME	SET TIME	Sets the time of the real-time clock.
	TIMEDATE		Sets the time and date of the real-time clock.
	TIMEDSP	TIMEDATE ON OFF	Turns the display of the real-time clock on or off.
DISPLAY	ANNOT	ANNOTATE ON OFF	Turns annotation on or off.
	DL	DSP LINE ON OFF	Specifies display line level.
	DSPLY		Writes the value of a variable on the analyzer screen.
	GRAT	GRAT ON OFF	Turns graticule on or off.
	HD	HOLD	Holds or disables data entry and blanks active function.
	MENU		Displays specified menu on the analyzer screen.
	PREFX TH	CHANGE PREFIX THRESHLD ON OFF	Specifies the prefix. Specifies displayed threshold level.
TITLE	CHANGE TITLE	Writes text string to the top line of the analyzer screen.	
FREQUENCY	CF	CENTER FREQ	Specifies center frequency.
	FA	START FREQ	Specifies start frequency.
	FB	START FREQ	Specifies stop frequency.
	FOFFSET	FREQ OFFSET	Specifies frequency offset.
	SS	CF STEP AUTO MAN	Specifies center-frequency step size.
GRAPHICS	CLRDRSP		Erases user-generated graphics.
	DT		Defines label terminator.
	GR		Graphs specified <i>y</i> values on the analyzer screen.
	LB		Writes label to display.
	PA		Moves pen to current position.
	PD		Places pen down.
	PR		Draws vector from last position (plot relative).
	PRINT	COPY to a printer	Prints screen data.
	PU		Lifts pen up.
TEXT		Writes text string to screen at current pen position.	
TRGRPH		Graphs compressed trace.	

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
INFORMATION	ACTVF		Returns a "0" if the function is not active.
	BIT		Returns the state of a bit.
	CLS		Clears status byte.
	HAVE		Returns a "0" if a device or option is not installed.
	ID		Returns the HP model number of the analyzer.
	MDU		Returns the analyzer's baseline and reference level.
	OP		Returns the lower-left and upper-right coordinates of the analyzer display.
	REV		Returns the analyzer's firmware date.
	RQS		Provides service request mask bits which are enabled for service requests.
	SER	SHOW OPTIONS	Returns the serial number of the analyzer.
	SRQ		Sets service request.
	STB		Queries status byte.
INPUT/OUTPUT	EE		Enables front-panel number entry.
	EK		Enables front-panel knob control.
	ENTER		Controls the HP-IB in order to receive data.
	EP		Enables parameter entry from front panel.
	OA		Returns active function.
	OL		Returns learn string.
	OUTPUT		Controls the HP-IB in order to send data.
	RELHPIB		Releases HP-IB control.
	TA		Controls trace A output.
	TB		Controls trace B output.
	TDF		Selects trace data output format.
	TRA/TRB/TRC		Controls trace data input or output.
LIMIT LINES	LIMIDEL	PURGE LIMITS NEW LIMIT	Deletes all segments in the current limit-line table.
	LIMIFAIL		Returns a "0" if the measurement sweep passes.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
LIMIT LINES	LIMILINE	SAVE LIMIT	Outputs the current limit-line table definitions.
	LIMIMIRROR		Reflects the limit-line about the amplitude axis at the largest frequency.
	LIMIMODE	EDIT UPPER, EDIT LOWER, EDIT UP/LOW, EDIT MID/DELTA	Selects type of limit-line table format—upper, lower, upper and lower, or mid/delta.
	LIMIREL	LIMITS FIX REL	Specifies the current limit-lines as fixed or relative.
	LIMISEG	EDIT UPPER, EDIT LOWER	Adds new segments to the current limit-line in the upper limit-line or the lower limit-line.
	LIMITEST	LIMITS ON OFF	Compares the active trace data with the current limit-line.
	SEGDEL	DELETE SEGMENT	Deletes the specified segment from the limit-line table(s).
	SENER	EDIT UP/LOW, EDIT MID/DELTA	Enters the limit-line data in the upper and lower limit-line tables or the mid/delta table.
MARKER	MDS		Specifies measurement data size as byte or word.
	MF		Returns marker frequency.
	MKA		Specifies amplitude of the active marker.
	MKACT		Specifies active marker: 1, 2, 3, or 4.
	MKBW		Specifies marker bandwidth.
	MKCF	MARKER -> CF	Moves marker frequency into center frequency.
	MKCONT		Continues sweep after MKSTOP.
	MKD	MARKER DELTA	Moves delta marker to specified position.
	MKF		Specifies frequency of active marker.
	MKFC*	MKR CNT ON OFF *	Turns marker frequency count on or off.
MKFCR*	CNT RES AUTO MAN *	Sets the marker counter resolution.	

\* For HP 8591A and HP 8593A only.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
MARKER	MKMIN	MINIMUM -> MARKER	Moves active marker to minimum signal detected.
	MKN	MARKER NORMAL	Moves active marker to specified frequency as frequency type marker.
	MKNOISE	MKNOISE ON OFF	Returns average value at marker, normalized to 1 Hz bandwidth.
	MKOFF	MARKERS OFF	Turns off all markers.
	MKP		Places the marker at the given x-axis position.
	MKPAUSE	MKPAUSE ON OFF	Pauses sweep at marker.
	MKPK	PEAK SEARCH, NEXT PEAK, NEXT PK RIGHT, NEXT PK LEFT	Moves active marker to maximum signal detected.
	MKPX	PEAK EXCURSN	Specifies minimum excursion for peak identification.
	MKREAD		Selects type of marker readout to be displayed.
	MKRL	MARKER -> REF LVL	Sets reference level to marker amplitude.
	MKSP	MKR Δ -> SPAN	Sets span to marker frequency value.
	MKSS	MARKER -> CF STEP	Sets to center-frequency step-size.
	MKSTOP		Stops the sweep at the active marker.
	MKTRACE MKTRACK MKTYPE M4	SIGNAL TRACK MARKER AMPTD	Assigns marker to trace. Turns signal track on or off. Specifies the marker type. Turns on marker zoom.
	MATH (see also Trace Math)	ABS	
ADD			Calculates the sum of the operands.
AVG			Averages two trace operands.
BIT			Returns the state of a bit.
CTA			Converts to absolute units.
CTM			Converts to measurement units.
DIV			Returns the result of the division of two operands.
EXP			Calculates the exponential of an operand.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
MATH	INT		Calculates integer value of an operand.
	LOG		Calculates log of operand.
	MEAN		Returns the mean value of a trace.
	MIN		Finds the minimum of two operands.
	MINPOS		Finds the <i>x</i> -axis position of the minimum trace value.
	MOD		Finds the remainder from division.
	MPY		Multiplies two operands.
	MXM		Finds the maximum of two operands.
	PDA		Finds the probability distribution of the amplitude.
	PDF		Finds the probability distribution of frequency.
	RMS		Finds the root mean square.
	SQR		Finds the square root.
STDEV		Finds the standard deviation.	
SUB		Subtracts one operand from another.	
VARIANCE		Finds amplitude variance of operand.	
MEASURE/USER (MEAS/USER)	LIMIDEL	PURGE LIMITS , NEW LIMIT	Deletes all segments in limit-line table and presets limit-line settings.
	LIMIFAIL		Returns a "0" if the measurement sweep passes.
	LIMILINE	SELECT SEGMENT	Outputs the current limit-line table definitions.
	LIMIMIRROR		Reflects the limit-line about the amplitude axis at the largest frequency.
	LIMIMODE	EDIT UPPER , EDIT LOWER , EDIT UP/LOW , EDIT MID/DELT	Selects type of limit-line table format—upper, lower, upper and lower, or mid/delta.
	LIMIREL	LIMITS FIX REL	Specifies the current limit-lines as fixed or relative.
	LIMISEG	EDIT UPPER , EDIT LOWER	Adds new segments to the current limit-line in the upper limit-line or the lower limit-line.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
MEASURE/USER (MEAS/USER)	LIMITEST	LIMITS ON OFF	Compares the active trace data with the current limit-line.
	PWRBW	99% PWR BW	Returns power bandwidth of signal.
	SEGDEL	DELETE SEGMENT	Deletes the specified segment from the limit-line table(s).
	SENER	EDIT UP/LOW, EDIT MID/DELTA	Enters the limit-line data in the upper and lower limit-line tables or the mid/delta table.
MODE	MODE	SPECTRUM ANALYZER	Returns the operating mode of the analyzer.
OPERATOR ENTRY	EE		Enables front-panel data number entry.
	EK		Enables front-panel knob control.
	EP		Enter parameter from front panel.
	HD	HOLD	Holds or disables entry and blanks active function readout.
PLOTTER	PLOT	COPY to a plotter	Plots screen data to previously addressed plotter.
PRESET	IP	PRESET	Performs an instrument preset.
	LF		Performs an instrument preset to the base band (band 0).
	POWERON		Selects the state the analyzer when turned on: IP (instrument preset) or LAST state.
	RESETRL		Resets the reference level to instrument preset value.
PRINTER	PRINT	COPY to a printer	Prints screen data to previously addressed printer.
PROGRAM FLOW	ABORT		Aborts all user-defined functions.
	IF		IF/THEN/ELSE/ENDIF forms a conditional construct.
	REPEAT		REPEAT/UNTIL forms a looping construct.
	RETURN		Returns from user-defined function.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
RECALL or SAVE	CAT	CATALOG INTRNL, CATALOG CRD*	Displays directory information from the specified or current mass storage device.
	LOAD*	LOAD FILE*	Loads data from the memory card.
	MSI	INTRNL CRD*	Defines the mass storage device.
	PREFX	CHANGE PREFIX	Specifies the prefix.
	PSTATE	SAV LOCK ON OFF	Protects internal state registers.
	PURGE	DELETE FILE	Deletes the file from the current mass storage device.
	RCLS	INTRNL -> STATE	Recalls state from internal state register.
	RCLT	INTRNL -> TRACE	Recalls state and trace, limit lines, or amplitude factors from the internal trace register.
	SAVES	STATE -> INTRNL	Saves state into internal state register.
	SAVET	INTRNL -> STATE	Saves state and trace, limit lines, or amplitude factors in the internal trace register.
	SAVRCLF	SAVE or RECALL	Indicates that a save or recall operation is in progress.
	SAVRCLN		Appends number to prefix for save and recall operations.
SAVRCLW		Used to specify what is to be saved or recalled.	
STOR*	STATE -> CARD*, TRACE -> CARD*, ALL DLP -> CARD*	Stores item from instrument to memory card.	
SPAN	FS	FULL SPAN	Specifies full frequency span.
	HN†		Returns the harmonic number (band).
	HNLOCK†	Band selection accessed by BAND LOCK† or BND LOCK ON OFF †	Locks the tuning band.
	HNUNLK†	BND LOCK ON OFF (OFF)†	Unlocks the tuning band.
* Option 003 required for HP 8590B or HP 8592B.			
† For HP 8592B or HP 8593A only.			

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
SPAN	SP	SPAN	Specifies frequency span. Places marker on highest on-screen signal peak, and turns on the signal track function.
	SPZOOM	SPAN ZOOM	
SWEEP	CONTS	SWEEP CONT SGL (CONT)	Selects continuous-sweep mode.
	ST	SWP TIME AUTO MAN	Specifies sweep time.
SYNCHRONIZATION	DONE		Returns a "1" after preceding commands are begun.
	TS		Begins a new sweep.
TRACE	AMB	A - B -> A ON OFF	Subtracts trace B from trace A and places the result in trace A. Subtracts trace B from trace A, adds the display line, and places the result in trace A. Exchanges trace A and trace B.
	AMBPL	NORMLIZE ON OFF	
	AXB	A <--> B	
TRACE	BLANK	BLANK A, BLANK B, BLANK C	Blanks trace.
	BML	B - DL -> B	Subtracts display line from trace B, and places the result in trace B.
	BTC	B -> C	Transfers trace B into trace C.
	BXC	B <--> C	Exchanges trace B and trace C.
	CLRW	CLEAR WRITE A, CLEAR WRITE B, CLEAR WRITE C	Clear-writes trace.
	DET IB	DETECTOR SAMPL PK	Specifies detection mode. Inputs trace B in binary units
	MINH	MIN HOLD C	Updates trace C elements with minimum level detected.
	MOV		Moves trace from source to destination.
	MXMH	MAX HOLD A, MAX HOLD B	Updates trace elements with maximum level detected.
	PKPOS TA		Returns maximum value of trace. Returns trace A data.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
TRACE	TB TRA/TRB/TRC		Returns trace B data. Controls trace A/B/C data input/output.
	TRCMEM		Returns the save trace memory capability.
	TRDEF		Declares a user-defined trace.
	TRDSP		Turns trace display on or off.
	TRGRPH		Graphs a compressed trace.
	TRPRST		Returns traces to preset state.
	TRSTAT TWINDOW		Returns status of traces. Specifies trace window for FFT.
VAVG	VID AVG ON OFF	Turns video averaging on or off.	
VIEW	VIEW A, VIEW B, VIEW C	Views trace.	
TRACE MATH (see also Math)	APB		Adds trace A and trace B and places the result in trace A.
	CLRAVG		Restarts video averaging.
	COMPRESS		Compresses a trace to the desired length.
	CONCAT		Concatenates two traces.
	FFT	FFT MEAS	Calculates fast Fourier transform.
	MIRROR		Displays the mirror image of a trace.
	PEAKS		Specifies trace peaks.
	SMOOTH		Smooths a trace.
SUM		Returns the sum of the amplitudes of the trace elements.	
SUMSQR		Returns the sum of the squares of the amplitude of each trace element.	
TRMATH		Performs trace math.	
XCH		Exchanges traces.	
TRIGGER	ONEOS		Performs the command list on end of sweep.
	ONSWP		Performs the command list at beginning of sweep.
	SNGLS	SGL SWP, SWEEP CONT SGL (SGL)	Selects single-sweep mode.

Table 4-4. Functional Index (continued)

Function Category	Command	Corresponding Key Function	Description
TRIGGER	TM	FREE RUN, LINE, VIDEO, EXTERNAL, TV TRIG*	Specifies trigger mode.
	TS		Begins a new sweep.
	TVLINE*	TV LINE #*	Specifies horizontal line of video to trigger on.
	TVSFRM*	TV TRIG ODD FLD*, TV TRIG EVEN FLD*, TV TRIG VERT INT*	Specifies type of video frame to trigger on.
	TVSTND*	TVSTND*	Selects the triggering for NTSC, PAL, PAL-M, and SECAM-L formats.
	TVSYNC*	TV SYNC NEG POS*	Selects between negative and positive triggering for video frame formats.
USER-DEFINED	ABORT		Aborts all user-defined functions.
	ACTDEF		Defines an active function.
	DISPOSE	DISPOSE USER MEM	Deletes user-defined functions.
	ERASE		Performs a dispose-all.
	FUNCDEF		Defines a function.
	KEYCLR		Clears softkeys 1 through 6.
	KEYCMD		Defines the function and label of a softkey based on a condition and updates label whenever a key is pressed.
	KEYDEF		Defines a softkey.
	KEYENH		Activates inverse video and underlining of a softkey.
	KEYEXC		Executes a softkey.
	KEYLBL		Relabels a softkey.
	MEM		Returns the amount of memory available.
	MENU		Displays the softkey menu.
	RETURN		Returns from user-defined function.
SAVEMENU		Saves softkeys 1-6 in the menu specified.	
TRDEF		Declares a user-defined trace.	
USTATE		Returns/sends user state.	
VARDEF		Declares a user-defined variable.	

\* Options 101 and 102 required.

This command cross-reference table indicates which commands are specific to the operation of an option or model number. An "X" indicates that the remote command is available for a standard model. "N/A" indicates the remote command is not available for that model. A number represents the option required to use the command with that model.

**Table 4-5. Command Cross Reference for the HP 8590 Series**

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
ABORT	X	X	X	X
ABS	X	X	X	X
ACTDEF	X	X	X	X
ACTVF	X	X	X	X
ADD	X	X	X	X
AMB	X	X	X	X
AMBPL	X	X	X	X
AMPCOR	X	X	X	X
ANNOT	X	X	X	X
APB	X	X	X	X
AT	X	X	X	X
AUNITS	X	X	X	X
AUTO	X	X	X	X
AVG	X	X	X	X
AXB	X	X	X	X
BIT	X	X	X	X
BLANK	X	X	X	X
BML	X	X	X	X
BTC	X	X	X	X
BXC	X	X	X	X
CAL	X	X	X	X
CAT	X	X	X	X
CF	X	X	X	X
CLRAVG	X	X	X	X
CLRDSP	X	X	X	X
CLRW	X	X	X	X
CLS	X	X	X	X
CNF	X	X	X	X
CNTLA	X	X	X	X
CNTLB	X	X	X	X
CNTLC	X	X	X	X
CNTLD	X	X	X	X
CNTLI	X	X	X	X
COMB	N/A	N/A	X	X
COMPRESS	X	X	X	X
CONCAT	X	X	X	X
CONTS	X	X	X	X
CORREK	X	X	X	X
CRTHPOS	X	X	X	X

Table 4-5. Command Cross Reference for the HP 8590 Series (continued)

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
CRTVPOS	X	X	X	X
CTA	X	X	X	X
CTM	X	X	X	X
DATEMODE	X	X	X	X
DEMOD	N/A	102	N/A	102
DET	X	X	X	X
DISPOSE	X	X	X	X
DIV	X	X	X	X
DL	X	X	X	X
DN	X	X	X	X
DONE	X	X	X	X
DSPLY	X	X	X	X
DT	X	X	X	X
EE	X	X	X	X
EK	X	X	X	X
ENTER	X	X	X	X
EP	X	X	X	X
ERASE	X	X	X	X
EXP	X	X	X	X
FA	X	X	X	X
FB	X	X	X	X
FFT	X	X	X	X
FMGAIN	N/A	102	N/A	102
FOFFSET	X	X	X	X
FORMAT	003	X	003	X
FS	X	X	X	X
FUNCDEF	X	X	X	X
GR	X	X	X	X
GRAT	X	X	X	X
HAVE	X	X	X	X
HD	X	X	X	X
HN	N/A	N/A	X	X
HNLOCK	N/A	N/A	X	X
HNUNLK	N/A	N/A	X	X
IB	X	X	X	X
ID	X	X	X	X
IF (THEN ELSE ENDIF)	X	X	X	X
INT	X	X	X	X
INZ	X	X	X	X
IP	X	X	X	X
KEYCLR	X	X	X	X
KEYCMD	X	X	X	X
KEYDEF	X	X	X	X
KEYENH	X	X	X	X

Table 4-5. Command Cross Reference for the HP 8590 Series (continued)

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
KEYEXC	X	X	X	X
KEYLBL	X	X	X	X
LB	X	X	X	X
LF	X	X	X	X
LG	X	X	X	X
LIMIDEL	X	X	X	X
LIMIFAIL	X	X	X	X
LIMILINE	X	X	X	X
LIMIMIRROR	X	X	X	X
LIMIMODE	X	X	X	X
LIMIREL	X	X	X	X
LIMISEG	X	X	X	X
LIMITEST	X	X	X	X
LN	X	X	X	X
LOAD	003	X	003	X
LOG	X	X	X	X
MDS	X	X	X	X
MDU	X	X	X	X
MEAN	X	X	X	X
MEASURE	010 or 011	010 or 011	N/A	N/A
MEM	X	X	X	X
MENU	X	X	X	X
MF	X	X	X	X
MIN	X	X	X	X
MINH	X	X	X	X
MINPOS	X	X	X	X
MIRROR	X	X	X	X
MKA	X	X	X	X
MKACT	X	X	X	X
MKBW	X	X	X	X
MKCF	X	X	X	X
MKCONT	X	X	X	X
MKD	X	X	X	X
MKF	X	X	X	X
MKFC	N/A	X	N/A	X
MKFCR	N/A	X	N/A	X
MKMIN	X	X	X	X
MKN	X	X	X	X
MKNOISE	X	X	X	X
MKOFF	X	X	X	X
MKP	X	X	X	X
MKPAUSE	X	X	X	X
MKPK	X	X	X	X

Table 4-5. Command Cross Reference for the HP 8590 Series (continued)

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
MKPX	X	X	X	X
MKREAD	X	X	X	X
MKRL	X	X	X	X
MKSP	X	X	X	X
MKSS	X	X	X	X
MKSTOP	X	X	X	X
MKTRACE	X	X	X	X
MKTRACK	X	X	X	X
MKTYPE	X	X	X	X
ML	X	X	X	X
MOD	X	X	X	X
MODE	X	X	X	X
MOV	X	X	X	X
MPY	X	X	X	X
MSI	003	X	003	X
MXM	X	X	X	X
MXMH	X	X	X	X
M4	X	X	X	X
NRL	X	X	X	X
OA	X	X	X	X
OL	X	X	X	X
ONCYCLE	X	X	X	X
ONDELAY	X	X	X	X
ONEOS	X	X	X	X
ONMKR	X	X	X	X
ONSRQ	X	X	X	X
ONSWP	X	X	X	X
ONTIME	X	X	X	X
OP	X	X	X	X
OUTPUT	X	X	X	X
PA	X	X	X	X
PD	X	X	X	X
PDA	X	X	X	X
PDF	X	X	X	X
PEAKS	X	X	X	X
PKPOS	X	X	X	X
PLOT	X	X	X	X
POWERON	X	X	X	X
PP	N/A	N/A	X	X
PR	X	X	X	X
PREFX	X	X	X	X
PRINT	X	X	X	X
PSTATE	X	X	X	X
PU	X	X	X	X

Table 4-5. Command Cross Reference for the HP 8590 Series (continued)

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
PURGE	X	X	X	X
PWRBW	X	X	X	X
RB	X	X	X	X
RCLS	X	X	X	X
RCLT	X	X	X	X
RELHPIB	021	021	021	021
REPEAT (UNTIL)	X	X	X	X
RESETRL	X	X	X	X
RETURN	X	X	X	X
REV	X	X	X	X
RL	X	X	X	X
RLPOS	X	X	X	X
RMS	X	X	X	X
ROFFSET	X	X	X	X
RQS	X	X	X	X
SAVEMENU	X	X	X	X
SAVES	X	X	X	X
SAVET	X	X	X	X
SAVRCLF	X	X	X	X
SAVRCLN	X	X	X	X
SAVRCLW	X	X	X	X
SEGDEL	X	X	X	X
SENER	X	X	X	X
SER	X	X	X	X
SETDATE	X	X	X	X
SETTIME	X	X	X	X
SMOOTH	X	X	X	X
SNGLS	X	X	X	X
SP	X	X	X	X
SPEAKER	N/A	102	N/A	102
SPZOOM	X	X	X	X
SQLCH	N/A	102	N/A	102
SQR	X	X	X	X
SRCALC	010 or 011	010 or 011	N/A	N/A
SRCAT	N/A	010 or 011	N/A	N/A
SRCNORM	010 or 011	010 or 011	N/A	N/A
SRCPOFS	010 or 011	010 or 011	N/A	N/A
SRCPSTP	010 or 011	010 or 011	N/A	N/A
SRCPSWP	010 or 011	010 or 011	N/A	N/A
SRCPWR	010 or 011	010 or 011	N/A	N/A
SRCTK	010 or 011	010 or 011	N/A	N/A
SRCTKPK	010 or 011	010 or 011	N/A	N/A
SRQ	X	X	X	X
SS	X	X	X	X

Table 4-5. Command Cross Reference for the HP 8590 Series (continued)

COMMAND	HP 8590B	HP 8591A	HP 8592B	HP 8593A
ST	X	X	X	X
STB	X	X	X	X
STDEV	X	X	X	X
STOR	003	X	003	X
SUB	X	X	X	X
SUM	X	X	X	X
SUMSQR	X	X	X	X
SWPCPL	010 or 011	010 or 011	N/A	N/A
TA	X	X	X	X
TB	X	X	X	X
TDF	X	X	X	X
TEXT	X	X	X	X
TH	X	X	X	X
TIMEDATE	X	X	X	X
TIMEDSP	X	X	X	X
TITLE	X	X	X	X
TM	X	X	X	X
TRA/TRB/TRC	X	X	X	X
TRCMEM	X	X	X	X
TRDEF	X	X	X	X
TRDSP	X	X	X	X
TRGRPH	X	X	X	X
TRMATH	X	X	X	X
TRPRST	X	X	X	X
TRSTAT	X	X	X	X
TS	X	X	X	X
TVLINE	N/A	101 and 102	N/A	101 and 102
TVSFRM	N/A	101 and 102	N/A	101 and 102
TVSTND	N/A	101 and 102	N/A	101 and 102
TVSYNC	N/A	101 and 102	N/A	101 and 102
TWNDOW	X	X	X	X
UP	X	X	X	X
USTATE	X	X	X	X
VARDEF	X	X	X	X
VARIANCE	X	X	X	X
VAVG	X	X	X	X
VB	X	X	X	X
VBR	X	X	X	X
VIEW	X	X	X	X
XCH	X	X	X	X

## ABORT

### Abort

The ABORT command stops the execution all user-defined functions and readies the instrument for the next command received.

### Syntax



Related Commands: ACTDEF, FUNCDEF, REPEAT/UNTIL, RETURN.

### Example

In the example below, ABORT is in the function called DLP.

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"CLRDSF;"	<i>Clears graphics from memory.</i>
30 OUTPUT 718;"TRDSP TRA,OFF;"	<i>Turns off trace A.</i>
40 OUTPUT 718;"ANNOT OFF;GRAT OFF;"	<i>Blanks annotation and graticule.</i>
50 OUTPUT 718;"VARDEF C_OUNT,0;"	<i>Declares a user-defined variable called C_OUNT.</i>
60 !	
70 OUTPUT 718;"FUNCDEF DLP,^;"	<i>Declares a user-defined function called DLP.</i>
80 OUTPUT 718;"REPEAT;"	<i>Begins a repeat loop.</i>
90 OUTPUT 718;"ADD C_OUNT,C_OUNT,100;"	<i>Adds 100 to C_OUNT.</i>
100 OUTPUT 718;"PU,PA 100,100;PD;"	
110 OUTPUT 718;"DSPLY C_OUNT,4.0;"	<i>Displays the value of C_OUNT.</i>
120 OUTPUT 718;"IF C_OUNT,EQ,300;THEN;ABORT;ENDIF;"	<i>Aborts the function when C_OUNT is equal to 300.</i>
130 OUTPUT 718;"UNTIL C_OUNT,EQ,400;"	<i>This is not executed because of the ABORT command in line 120.</i>
140 OUTPUT 718;"^;"	<i>Marks end of DLP.</i>
150 OUTPUT 718;"FUNCDEF S_HELL,^;"	<i>Defines second user-defined function called S_HELL.</i>
160 OUTPUT 718;"DLP;TEXT@INSIDE S_HELL@;^;"	<i>Executes DLP and displays message on screen.</i>
170 !	
180 OUTPUT 718;"S_HELL;"	<i>This calls the S_HELL function which in turn calls the DLP function.</i>
190 END	

**ABORT Abort****Description**

If ABORT is encountered in a function that has been executed by pressing a softkey, the function is interrupted and front-panel control is returned.

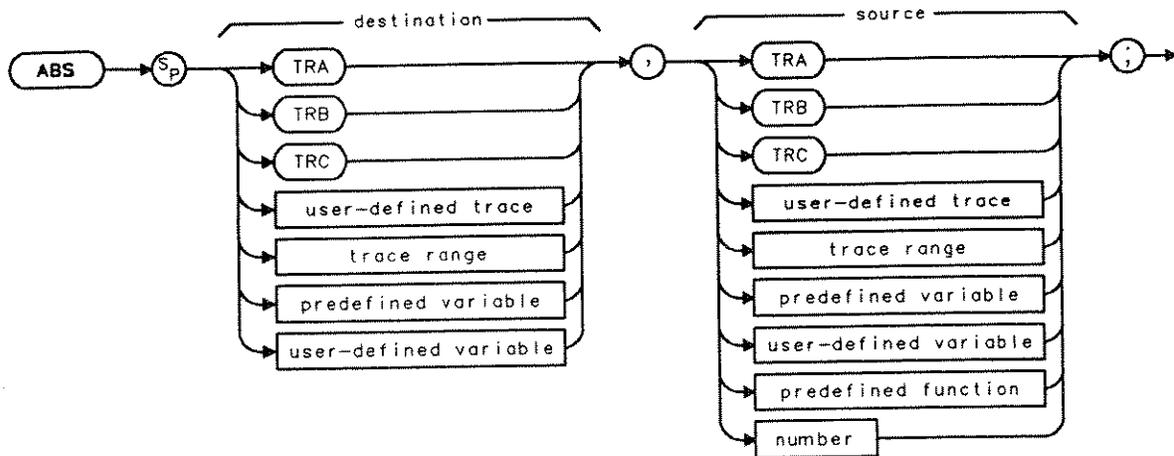
If the ABORT function is nested within one of more user-defined functions, ABORT stops the execution of all user-defined functions and readies the analyzer to act on the next command received.

In comparison, the RETURN command also interrupts operation of a user-defined function, but it returns operation to the point at which the user-defined function was called. (See "RETURN.")

## ABS Absolute

The ABS command places the absolute value of the source value(s) in the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using user-defined trace. ACTDEF or VARDEF when using user-defined variable. TS when using trace data.

**ABS Absolute****Example**

10 OUTPUT 718;"IP;SNGLS;"	<i>Initializes analyzer, stops sweeping.</i>
20 OUTPUT 718;"VARDEF P_OINT,0;"	<i>Defines a variable, called P_OINT, and initializes it to 0.</i>
30 OUTPUT 718;"ABS P_OINT,-2"	<i>Places the absolute value of -2 into POINT.</i>
40 OUTPUT 718;"P_OINT?;"	<i>Returns value of POINT to computer.</i>
50 ENTER 718;Second	<i>Assigns value to computer variable, Second.</i>
60 DISP Second	<i>Displays absolute value (2).</i>
70 END	

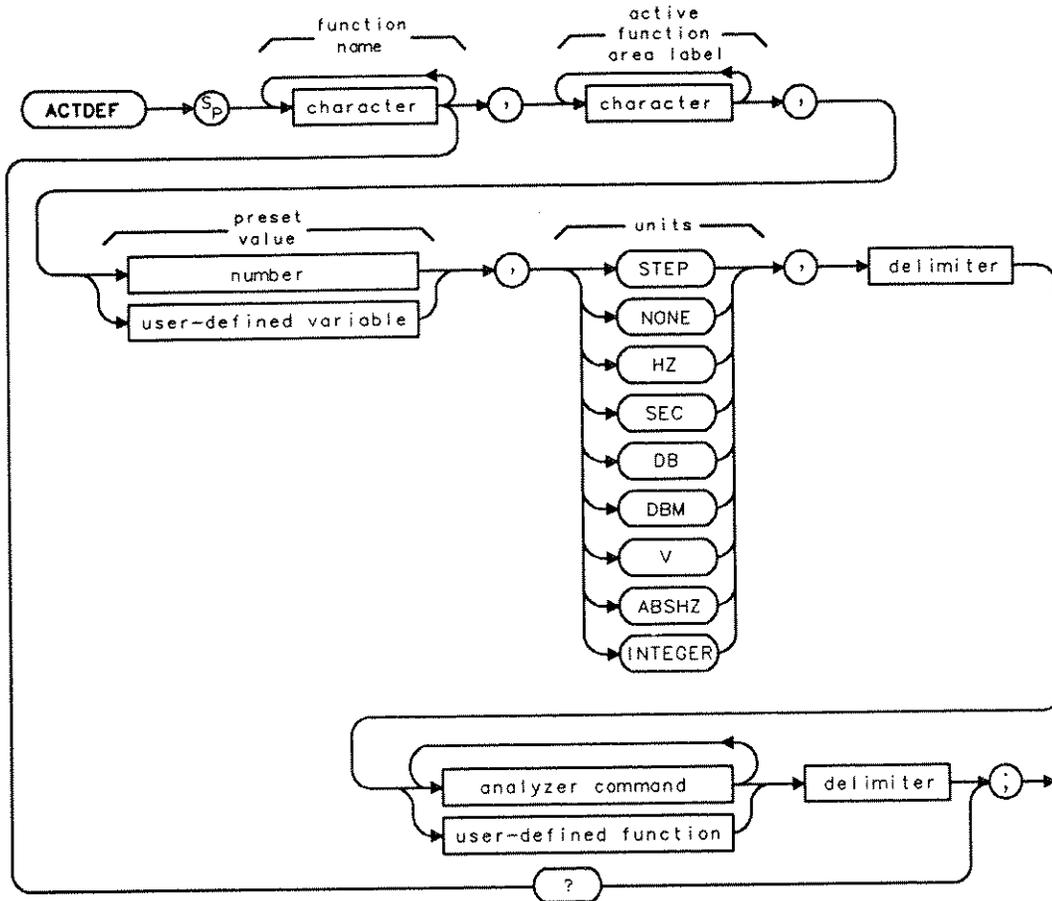
**Description**

The source and destination may be different lengths. The length of predefined traces is 401. User-defined traces have a length of up to 2047, and variables have a length of one. When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

# ACTDEF Active Function Definition

The ACTDEF command creates a user-defined active function.

## Syntax



## ACTDEF Active Function Definition

Item	Description/Default	Range
Character (function name)	Any valid character. Use the function name as the remote command name.	2 to 11 characters long, A through Z and the underscore (the underscore should be used as the second character of the label).
Character (active function label)	Any valid character. The active function label is displayed in the active function block (when the function is active).	
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Number	Any real or integer number. Default value is 0, default unit is none.	Real number range.
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Analyzer command	Any valid analyzer command.	
User-defined function	A function defined by the FUNCDEF command.	Any valid function name.

Prerequisite Command: FUNCDEF when using a user-defined function.

**Example**

```
10 DIM A$[150]
```

```
20 OUTPUT 718;"ACTDEF M_BW,%MY BANDWIDTH%,5MHZ,STEP,!MOV  
RB,M_BW!;"
```

```
30 OUTPUT 718;"M_BW;"
```

```
40 OUTPUT 718;"ACTDEF M_BW?;"
```

```
50 ENTER 718;A$
```

```
60 DISP A$
```

```
70 END
```

*Dimensions array to hold query response.*

*Defines a function called M\_BW. It allows manipulation of the initial value of RES BW (5 MHz) by the step keys and the knob. The resolution bandwidth will be rounded to the nearest allowable bandwidth, however. Activates the M\_BW function. Queries the definition of the M\_BW function.*

*Displays the definition of the M\_BW function.*

## Description

With the ACTDEF command, the user can create an active function similar to the predefined active functions (CF, DL, AT, MKA, MKFCR, MKD, MKF, MKN, RB, SS, ST, TH, VAVG, VB, VBR).

The function value can be manipulated by the knob and step keys. The unit parameters specify the units of the function value (Hz, seconds, dB, dBm, V). The STEP, INTEGER and NONE parameters are unitless values.

Use the following guidelines when defining ACTDEF:

- The function name used in the ACTDEF declaration must be unique (do not use an existing remote command mnemonic or other reserved words; see Table 4-2).
- Use of a user-defined function instead of a list of analyzer commands within the ACTDEF declaration is recommended when the list does not fit on one or two program lines.
- Use the function name in a programming line or in a softkey definition.

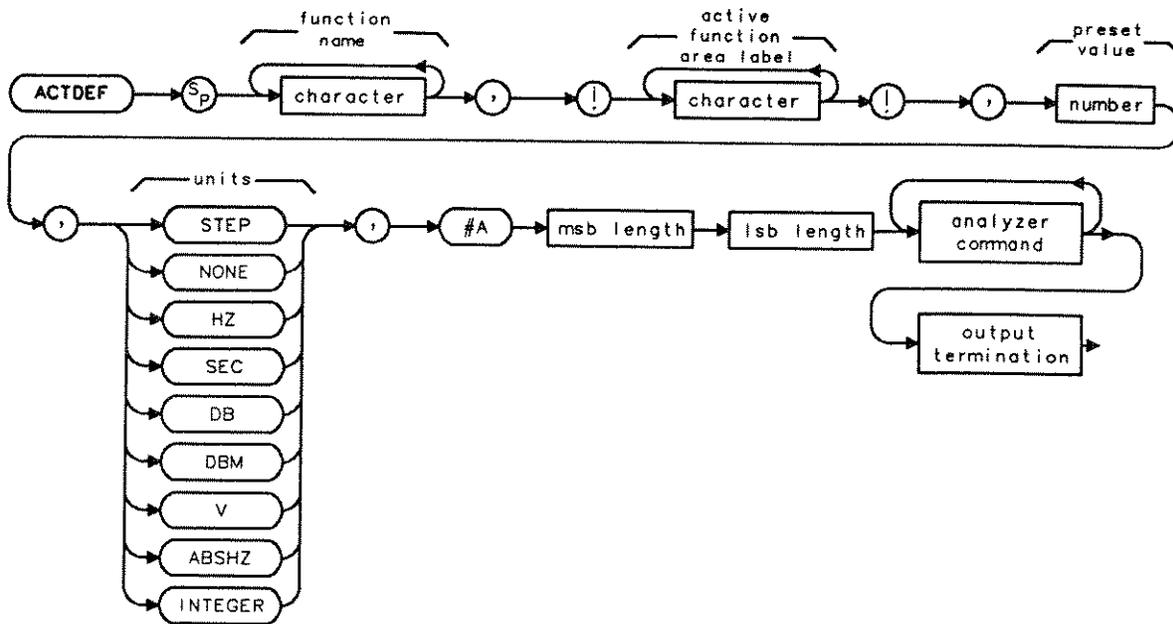
## Query Response

There are two ways to query the ACTDEF command. Following the function name with a question mark returns the value of the function. For example, if the function name is NAME, executing `OUTPUT 718;"NAME?;"` returns the value of function NAME in the following format:



**ACTDEF Active Function Definition**

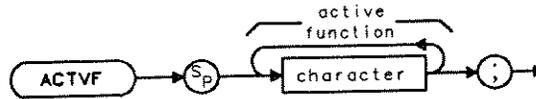
Querying ACTDEF with the function name returns the definition of the function. For example, execute OUTPUT 718;“ACTDEF NAME?,” (NAME is the function name).



## ACTVF Active Function

The ACTVF command returns a "0" if the given function is not active, a "1" if it is active.

### Syntax



Item	Description/Default	Range
Character	Any valid character.	Any active function name.

Related Commands: Any active function (see command description).

### Example

```

OUTPUT 718;"RB 100KHZ;"  Makes resolution bandwidth the active function.
OUTPUT 718;"ACTVF RB;"  Determines if resolution bandwidth is the active function.
ENTER 718;A              Gets the response from the analyzer.
DISP A                   Displays response from the analyzer.
    
```

### Description

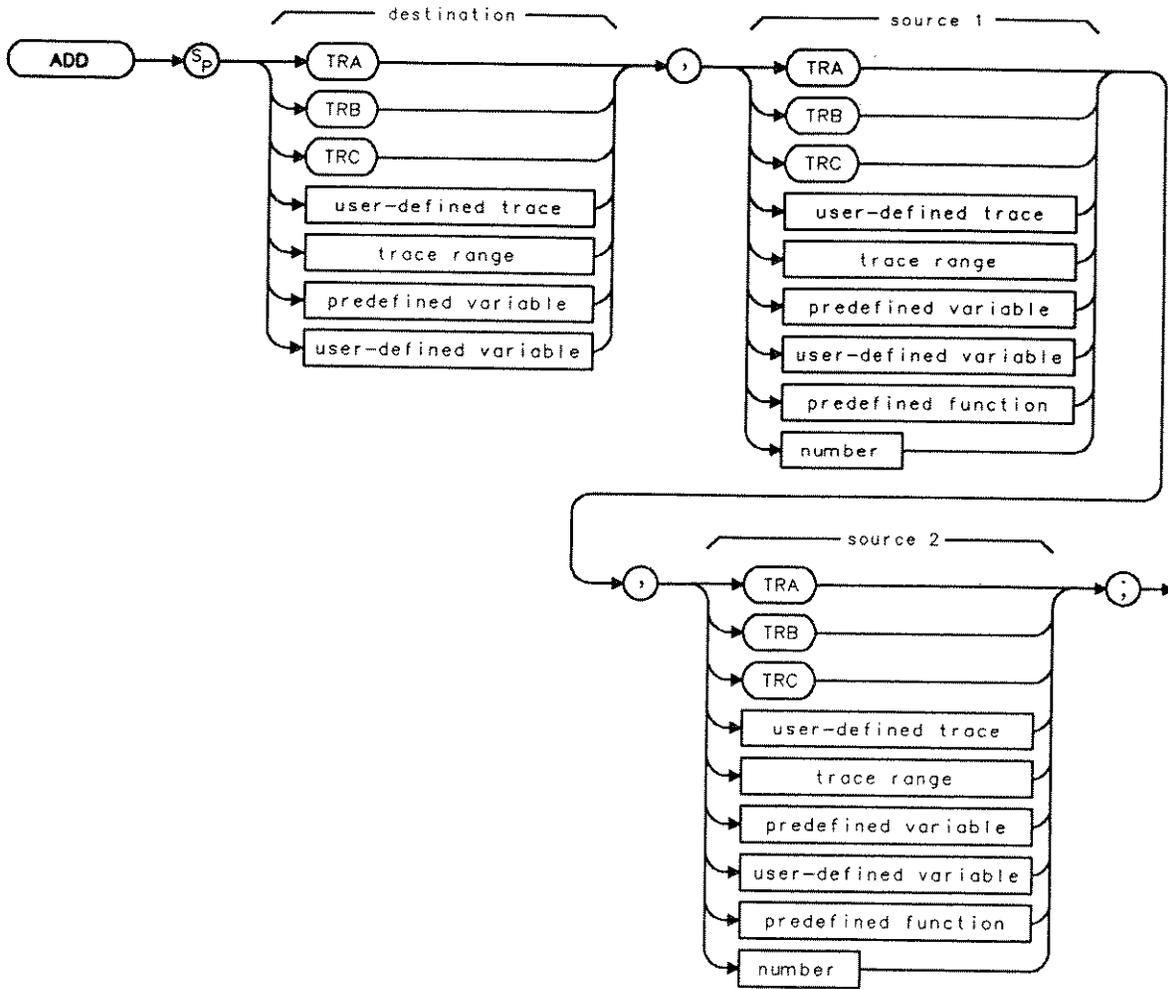
The active functions are AT, CF, DL, FA, FB, FMGAIN, FOFFSET, INZ, LG, MKA, MKD, MF, MKFCR, MKN, MKPAUSE, MKPX, ML, MODE, NRL, RB, ROFFSET, RL, RLPOS, SAVRCLN, SETDATE, SETTIME, SP, SQLCH, SRCAT, SRCPOFS, SRCPSTP, SRCPSWP, SRCPWR, SRCTK, SS, ST, TH, TIMEDATE, TVLINE, VB, VBR, and user-defined active function specified by the ACTDEF command.

# ADD

## Add

The ADD command adds the sources and sends the sum to the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using user-defined trace. ACTDEF or VARDEF when using user-defined variable. TS when using trace data.

### Example

Add 38 MHz to the center frequency, then store the sum in a user-defined variable.

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"CF 300MHZ;"	<i>Changes the center frequency.</i>
30 OUTPUT 718;"VARDEF N_EW,0;"	<i>Defines a variable, N_EW, and initializes it to 0.</i>
40 OUTPUT 718;"ADD N_EW,CF,38000000;"	<i>Adds 38 MHz to the center frequency, then places the sum in N_EW.</i>
50 OUTPUT 718;"N_EW?;"	<i>Returns value of N_EW to the computer.</i>
60 ENTER 718;Freq	<i>Assigns value to variable, Freq.</i>
70 DISP Freq	<i>Displays Freq on computer screen.</i>
80 END	

### Description

The ADD command adds values of source 1 and source 2, point by point, and sends the sum to the destination.

Traces, user-defined traces, and trace ranges are added as 16-bit integers. Negative numbers are represented in two's complement format. Single variables and numbers are treated as floating point numbers and must be within the real number range as defined in Table 4-1.

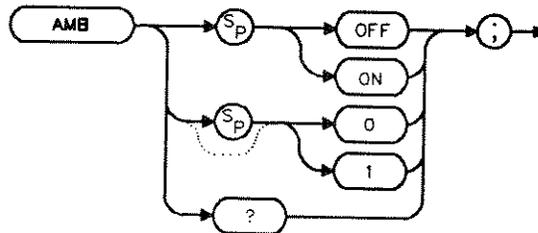
The sources and destination may be different lengths. The length of predefined traces is 401. User-defined traces have a length of up to 2047, and variables have a length of one. When sources differ in length, the last element of the shorter source is repeated for the addition process. After the addition, the last element is repeated if the destination is longer than the sum trace. When the sources are longer than the destination, they are truncated to fit.

## AMB

### Trace A Minus Trace B

The AMB command subtracts trace B from trace A and sends the result to trace A during every sweep of the analyzer.

#### Syntax



Preset State: OFF.

Related Commands: CLRW, CONTS, MXMH, SNGLS, TS, VAVG, VIEW.

#### Example 1

```

10 OUTPUT 718;"IP;"
20 OUTPUT 718;"CLRW TRB;TS;VIEW TRB;AMB ON;"

30 OUTPUT 718;"AMB?;"
40 ENTER 718;Reply$

50 DISP Reply$

60 END

```

*Initializes analyzer.*

*Views trace B and turns on the AMB function. Since trace A and trace B contain the same trace data, the result is trace data at 0 measurement units, at the bottom of the screen.*

*Queries the state of the AMB function. The query response is placed in a string variable.*

*ON is displayed on the computer screen.*

#### Example 2

```

10 OUTPUT 718;"IP;SNGLS;"
20 OUTPUT 718;"MOV TRA, 8000;"

30 OUTPUT 718;"VIEW TRA;"
40 OUTPUT 718;"MOV TRB,7000;"

50 OUTPUT 718;"VIEW TRB;"
60 OUTPUT 718;"AMB ON;"
70 OUTPUT 718;"BLANK TRB;VIEW TRA;"

```

*Each element of trace A is at 8000 measurement units, which is equal to 0 dBm.*

*Each element of trace B is at 7000 measurement units, which is equal to -10 dBm.*

*Subtracts trace B from trace A.*

*The result is displayed at 1000 measurement units, which is equal to -70 dBm.*

80 END

**Example 3**

10 OUTPUT 718;"IP;SNGLS;RL 20DB;"	
20 OUTPUT 718;"MOV TRA,5000;"	<i>Sets trace A to 5000 measurement units which is equal to -10 dBm.</i>
30 OUTPUT 718;"VIEW TRA;"	
40 OUTPUT 718;"MOV TRB,4000;"	<i>Sets trace B to 4000 measurement units which is equal to -20 dBm.</i>
50 OUTPUT 718;"VIEW TRB;"	
60 OUTPUT 718;"DL ODB;"	<i>Sets display line to 0 dBm which is at 6000 measurement units.</i>
70 OUTPUT 718;"BML;"	<i>Subtracts trace B minus display line. Result is <math>4000 - 6000 = -2000</math> measurements units (off screen).</i>
80 OUTPUT 718;"AMB ON;"	<i>Subtracts trace A minus modified trace B (<math>5000 - (-2000) = 7000</math> or 10 dBm. Note that this has resulted in a subtraction of amplitude in dBm, <math>-10 \text{ dBm} - (-20 \text{ dBm}) = 10 \text{ dBm}</math>.</i>
90 OUTPUT 718;"BLANK TRB;VIEW TRA;"	
100 END	

**Description**

The AMB command subtracts trace B from trace A, point by point, and sends the result to trace A. The AMB function remains in effect until it is turned off by executing "AMB OFF;".

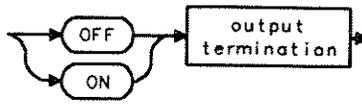
The AMB command is a trace math command and subtracts trace A from trace B in measurement units (see "CTA" for information about measurement units). Since subtracting trace B from trace A can cause the result in trace A to be displayed off-screen, the trace A minus trace B plus display line (AMBPL) command can be used. As shown in example 2, if the trace data value of trace A is 0 dBm (8000 measurement units), and trace B is -10 dBm (7000 measurement units), the result of executing AMB is 1000 measurement units. If the AMBPL command is used instead of AMB, and the display line is set at -50 dBm (mid-screen), the result in trace A is kept at mid-screen.

A common use of trace subtraction is to normalize one trace with respect to another. For example, traces are frequently subtracted to normalize the analyzer response when a tracking generator is used. In such applications, amplitude units in dBm should be subtracted. Note from example 2 that subtraction of measurement units is not equivalent to subtraction of amplitude units. Correct results are obtained if the display line is set to 0 dBm using DL, and BML is used to subtract the display line from trace B. (The AMBPL command performs this function more easily.) See example 3.

The functions of the AMB command and **A - B -> A ON OFF** are identical.

AMB Trace A Minus Trace B

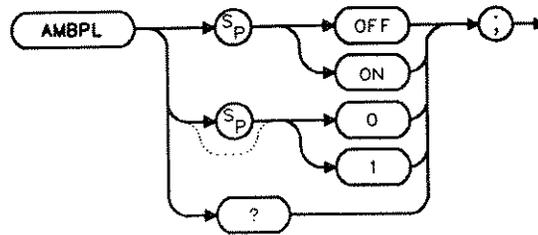
Query Response



## AMBPL Trace A Minus Trace B Plus Display Line

The AMBPL command subtracts trace B from trace A, adds the display line value to the difference, and sends the result to trace A during every sweep of the analyzer.

### Syntax



Preset State: AMBPL OFF.

Related Commands: AMB, CONTS, CLRW, DL, MXMH, SNGLS, TS, VAVG, VIEW.

### Example

```
10 OUTPUT 718;"IP;SNGLS;RL 20DB;"
20 OUTPUT 718;"MOV TRA,5000;"
```

*Initializes analyzer, activates single-sweep mode. Sets trace A to 5000 measurement units which is equal to -10 dBm.*

```
30 OUTPUT 718;"VIEW TRA;"
40 OUTPUT 718;"MOV TRB,4000;"
```

*Sets trace B to 4000 measurement units which is equal to -20 dBm.*

```
50 OUTPUT 718;"VIEW TRB;"
60 OUTPUT 718;"DL ODM;"
```

*Sets display line to 0 dBm which is at 6000 measurement units.*

```
70 OUTPUT 718;"AMBPL ON;"
```

*Performs trace A - trace B + display line. The result is  $5000 - 4000 + 6000 = 7000$  or 10 dBm. Note that this has resulted in a subtraction of amplitude in dBm,  $-10 \text{ dBm} - (-20 \text{ dBm}) = 10 \text{ dBm}$ .*

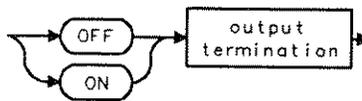
```
80 OUTPUT 718;"BLANK TRB;VIEW TRA;"
90 END
```

**AMBPL Trace A Minus Trace B Plus Display Line****Description**

The AMBPL command subtracts trace B from trace A, point by point, adds the display line value to the difference, and sends the difference to trace A. The AMBPL function remains in effect until it is turned off by executing "AMBPL OFF;".

A common use of trace subtraction is to normalize one trace with respect to another. For example, traces are frequently subtracted to normalize the analyzer response when a tracking generator is used. In such applications, amplitude units in dBm should be subtracted. To accomplish this, the display line should be set to 0 dBm using DL as shown in the example. Also see example 2 and 3 in "AMB" for comparison.

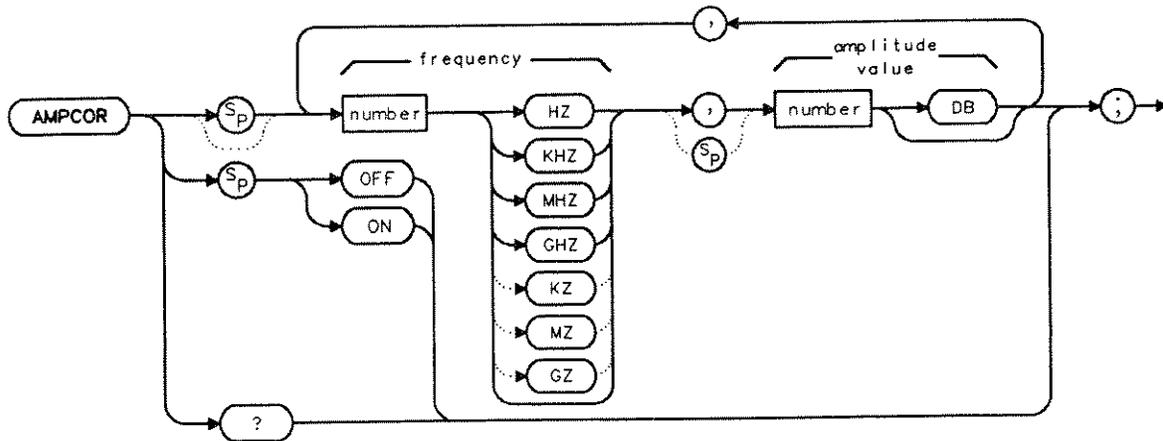
The functions of AMBPL, SRCNORM, and ~~NORMLIZE ON OFF~~ are identical.

**Query Response**

## AMPCOR Amplitude Correction

The AMPCOR command applies amplitude corrections at specified frequencies.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dB.	Frequency: 0 to 1000 GHz, amplitude: $\pm 327$ dB.

Preset State: AMPCOR OFF.  
Related Commands: AT, RL.

### Example

Compensate for frequency-dependent amplitude inaccuracies at the input.

```

10  DIM A$(200)
20  OUTPUT 718;"CF 1GHZ;SP 200MHZ;"  Sets center frequency and span.
30  OUTPUT 718;"AMPCOR 100MHZ,5DB,  Stores frequency-amplitude pairs in analyzer.
    1GHZ,-5DB,1.5GHZ,10DB;"         Notice that frequencies are in ascending order.
40  OUTPUT 718;"AMPCOR?;"          Returns correction values to computer.
50  ENTER 718;A$
60  PRINT A$                        Displays the frequency-amplitude pairs.
70  OUTPUT 718;"AMPCOR OFF;"       Turns off the amplitude correction constants.
80  OUTPUT 718;"AMPCOR?;"
90  ENTER 718;A$
100 PRINT A$                          0,0 is displayed since AMPCOR is off.

```

**AMPCOR Amplitude Correction**

110 END

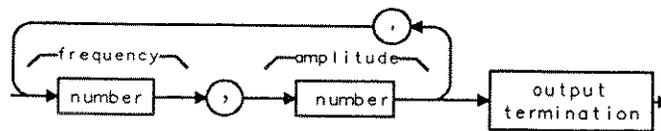
**Description**

Use AMPCOR to compensate for frequency-dependent amplitude variations at the input port. Up to 80 pairs of frequency-amplitude correction points can be entered. The frequency values entered must be equal or in increasing order, or an error condition results. Whenever AMPCOR is on, the correction values are added to all measurement results. Executing AMPCOR with frequency and amplitude pairs as parameters or ON as the parameter turns the amplitude correction factors on. Instrument preset (IP) or turning off the analyzer sets AMPCOR to OFF.

The values of the correction points are applied across the active measurement range. Between points, the correction values are interpolated. When measuring at frequencies outside the first and last correction points, these values are used as the correction value.

Because AMPCOR adjusts IF gain or attenuation, AMPCOR should be used with care when measuring signals near the compression level that are also at a frequency that is corrected by the AMPCOR command.

Amplitude correction factors can be stored in analyzer memory with the SAVET or SAVRCLN commands, or on the memory card with the STOR or SAVRCLN commands.

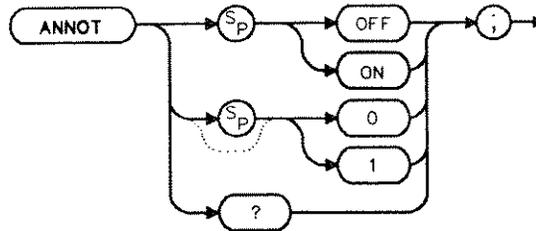
**Query Response**

Returned values are 0,0 when AMPCOR is set to OFF.

## ANNOT Annotation

The ANNOT command turns the display annotation on or off.

### Syntax



Preset State: ON.

Related Commands: GRAT, TITLE.

### Example

```

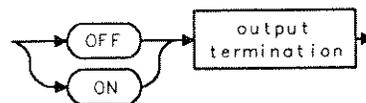
10 OUTPUT 718;"ANNOT ON;"  Turns the annotation on.
20 OUTPUT 718;"ANNOT?;"   Queries state of the annotation function.
30 ENTER 718;Reply$       Query response is placed in a variable.
40 DISP Reply$            Response is displayed on the computer screen.
50 END
  
```

### Description

The ANNOT command turns on or off all the words and numbers (annotation) on the analyzer screen (except softkeys).

The functions of the ANNOT command and **ANNOTATN ON OFF** are identical.

### Query Response



## APB Trace A Plus Trace B

The APB command adds trace A to trace B and sends the result to trace A.

### Syntax



Related Commands: CLRW, SNGLS, TS, VIEW.

### Example

10 OUTPUT 718;"IP;SNGLS;"	<i>Initializes analyzer, changes to single-sweep mode.</i>
20 OUTPUT 718;"TS;"	<i>Updates the trace.</i>
30 OUTPUT 718;"VIEW TRA;RL -20DM;CLRW TRB;"	<i>Changes the reference level.</i>
40 OUTPUT 718;"TS;VIEW TRB;"	<i>Takes a measurement sweep.</i>
50 OUTPUT 718;"APB;"	<i>Activates APB command.</i>
60 OUTPUT 718;"BLANK TRB;VIEW TRA;"	<i>Views result of APB.</i>
70 END	

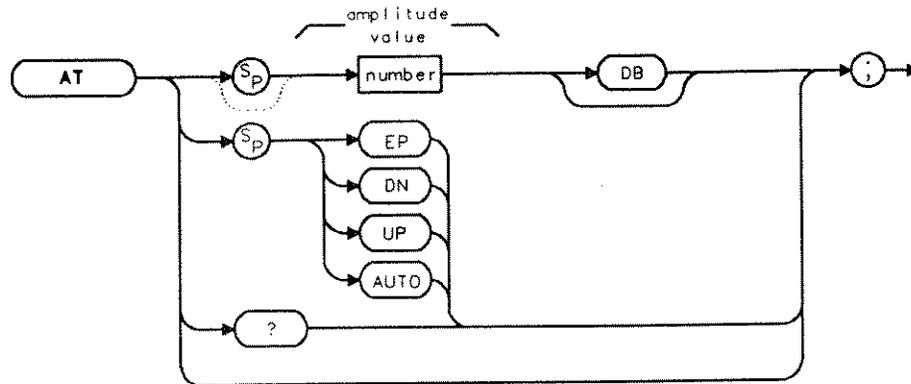
### Description

The traces are added as 16-bit integers. Negative numbers are represented in two's complement format. The two's complement representation of a negative number is obtained by changing the 1s to 0s of the binary representation of the number, and then binarily adding 1.

## AT Attenuation

The AT command specifies the RF input attenuation.

### Syntax



Item	Description/Default	Range
Number	Any real or integer. Default units are dB.	Input attenuator range of analyzer.

Preset State: 10 dB.

Input attenuator range (HP 8590B or HP 8591A): 0 to 60 dB.

Input attenuator range (HP 8592B or HP 8593A): 0 to 70 dB.

Step Increment: in 10 dB steps.

Related Commands: AUTO, ML, RL.

### Example

OUTPUT 718;"AT 40DB;" *Sets the attenuation to 40 dB.*

OUTPUT 718;"AT UP;" *Increases the attenuation to 50 dB.*

### Description

The AT command specifies the input attenuation in 10 dB steps. Normally, the input attenuator is coupled to the reference level. When a continuous wave signal is displayed with its peak at or below the reference level, the coupling keeps the mixer input level at or below the specified level (also see "ML"). Instrument preset sets the attenuation to 10 dB. The AT command allows less than the specified value at the mixer input.

When the attenuation is increased with the AT command, the reference level does not change. If the attenuation is decreased from the coupled value using the AT command, the reference

## AT Attenuation

level will be decreased. When the reference level is changed with the RL command, the input attenuation changes to maintain a constant signal level on the screen if attenuation is coupled.

Using AUTO (auto-couple) resets the attenuation value so that a continuous wave signal displayed at the reference level yields  $-10$  dBm (or the specified mixer level) at the mixer input.

The step keys, knob, and DN parameter do not allow an attenuation entry below 10 dB. Only direct entry of "AT 0DB;" will achieve 0 dB attenuation.

---

**Caution** Signal levels above +30 dBm will damage the analyzer.



---

The functions of AT and **ATTEN AUTO MAN** are similar.

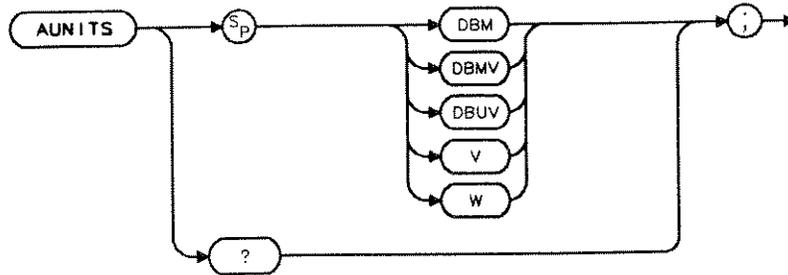
## Query Response



## AUNITS Amplitude Units

The AUNITS command specifies the amplitude units for input, output and display.

### Syntax



Related Commands: CAL, DL, MKA, RL, TH.

### Example

OUTPUT 718;"LN;"	<i>Changes analyzer to linear mode.</i>
OUTPUT 718;"AUNITS DBMV;"	<i>Changes the linear amplitude units to DBMV.</i>
OUTPUT 718;"AUNITS?;"	<i>Queries current amplitude units.</i>
ENTER 718;Reply\$	<i>Puts response in a variable.</i>
DISP Reply\$	<i>Displays response on the computer screen.</i>

### Description

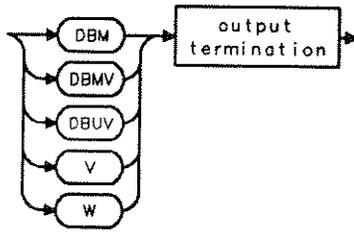
The AUNITS command sets the amplitude readouts (reference level, marker, display line, and threshold) to the specified units. Different amplitude units can be set for log and linear amplitude scales.

The functions of AUNITS and AMPTD UNITS are similar.

**AUNITS Amplitude Units**

**Query Response**

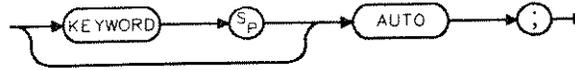
The query response returns the current amplitude units for the current amplitude scale.



## AUTO Auto Couple

The AUTO parameter automatically couples the active functions.

### Syntax



Related Commands: AT, DL, HD, MKA, MKD, MKF, MKFCR, MKN, RB, SRCPSTP, SRCPSWP, SRCPWR, SS, ST, TH, VAVG, VB, VBR.

### Example

OUTPUT 718;"AT AUTO;" *Couples the attenuation.*  
 OUTPUT 718;"HD;AUTO;" *Couples all functions.*  
 OUTPUT 718;"AUTO;" *Couples and deactivates a related function (if one was active), or couples all functions (if no functions were active).*

### Description

The result of the AUTO command depends on the active function it acts upon. The following are the functions that are affected by the auto parameter:

AT	couples attenuation to the reference level.
DL	turns off display of line but does not change the value of the display line.
MKA	turns off marker.
MKD	turns off marker.
MKF	turns off marker.
MKFCR	deactivates use of user-supplied counter resolution value, however the value remains unchanged (HP 8591A and HP 8593A only).
MKN	turns off marker.
RB	couples resolution bandwidth to frequency span.
SRCPSTP	sets source power step to 0 (it may value may be displayed at 10 however).
SRCPSWP	turns off power sweep.
SRCPWR	turns off source power.
SS	couples step size to frequency span.
ST	couples sweep time to frequency span.
TH	turns off display of threshold but does not change its value or prevent usage in peak searching.
VAVG	stops averaging.
VB	couples video bandwidth to resolution bandwidth.
VBR	sets video to bandwidth ratio to 0.3.

Individual functions can be coupled by entering the keyword for the command before AUTO, (for example, "AT AUTO;").

**AUTO Auto Couple**

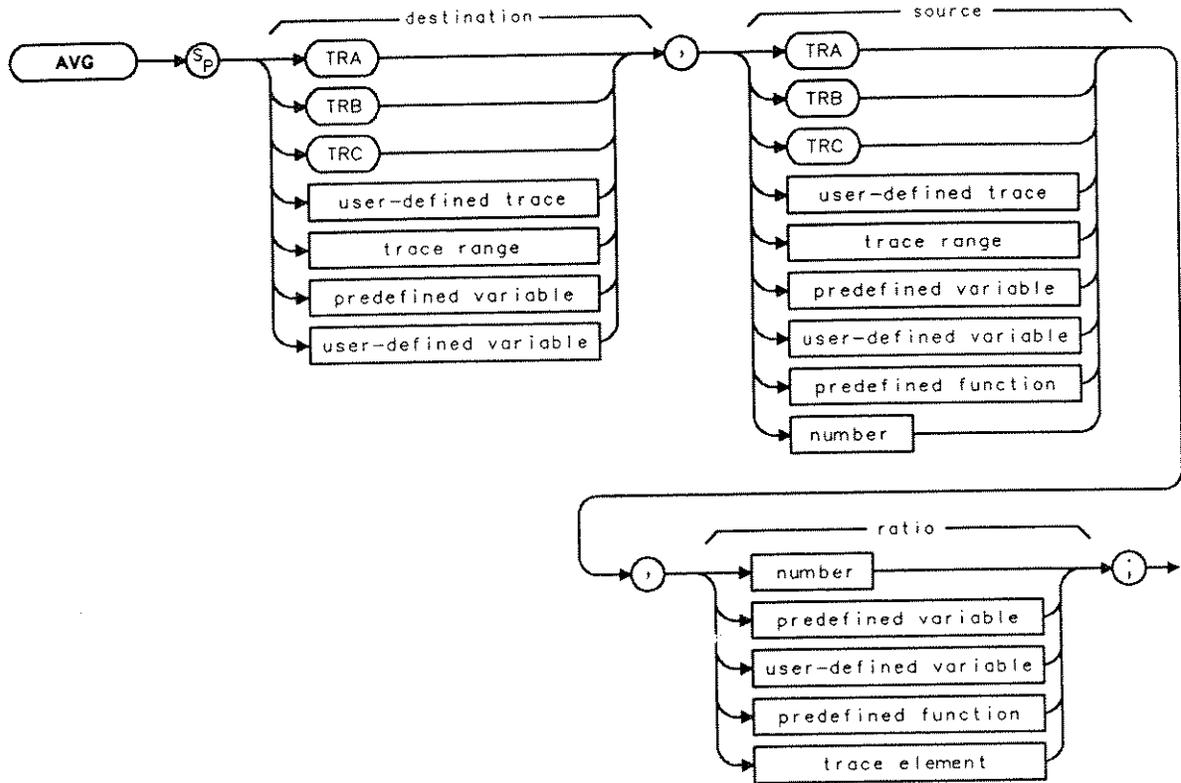
AUTO has no effect if the active function is not in the above list.

Executing "AUTO;" if no functions were active couples all functions. Executing "HD;AUTO;" couples all functions, turns off the reference position, and turns off normalization.

## AVG Average

The AVG command averages the source and the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

**AVG Average**

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

Related Commands: CLRAVG, TS.

**Example**

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
30 OUTPUT 718;"DET POS;TS;"	<i>Activates the positive-peak detector and update the trace.</i>
40 OUTPUT 718;"VIEW TRA;"	<i>Stores results and view trace A.</i>
50 OUTPUT 718;"DET SMP;"	<i>Activates sample detection.</i>
60 OUTPUT 718;"CLRWB TRB;TS;"	<i>Activates trace B and take sweep of trace B.</i>
70 OUTPUT 718;"VIEW TRB;"	<i>Views trace B.</i>
80 OUTPUT 718;"AVG TRA,TRB,2;"	<i>Averages traces B and A with ratio of 2 and stores the result in trace A.</i>
90 OUTPUT 718;"BLANK TRB;"	<i>Blanks trace B and displays result (in trace A).</i>
100 END	

**Description**

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

$$\text{Average} = [((\text{ratio} - 1) \times \text{destination}) + \text{source}] / \text{ratio}$$

The results of AVG are invalid if ratio is equal to zero.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## AXB Exchange Trace A and Trace B

The AXB command exchanges trace A and trace B.

### Syntax



Related Commands: CLRW, SNGLS, TS, VIEW.

### Example

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
30 OUTPUT 718;"DET POS;TS;"	<i>Activates positive-peak detection of trace A .</i>
40 OUTPUT 718;"VIEW TRA;"	<i>Stores results, displays trace A.</i>
50 OUTPUT 718;"DET SMP;"	<i>Activates sample detection.</i>
60 OUTPUT 718;"CLRW TRB;TS;"	<i>Clear-writes trace B and takes sweep.</i>
70 OUTPUT 718;"VIEW TRB;"	<i>Stores results of sweep in trace B.</i>
80 OUTPUT 718;"AXB;"	<i>Exchanges trace A with trace B.</i>
90 OUTPUT 718;"BLANK TRB;"	<i>Blanks trace B leaving only trace A on screen.</i>
100 END	

### Description

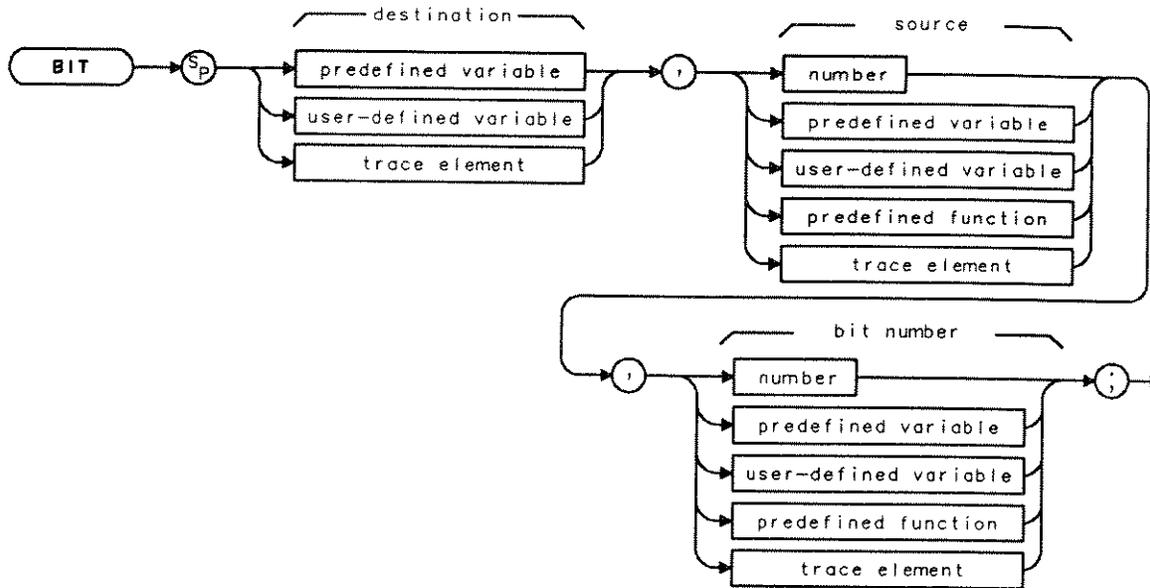
The AXB command exchanges trace A and trace B, point by point. AXB sets trace A and trace B to the view mode.

The functions of the AXB command and `A <- -> B` are similar.

## BIT Bit

The BIT command returns the state ("1" or "0") of a bit.

### Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer number.	-32,768 to +32,767.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: INT, STB.

**Example**

10 CLEAR 718	<i>Clears HP-IB bus.</i>
20 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
30 OUTPUT 718;"VARDEF E_ERROR,0;"	<i>Defines user-defined variable E_ERROR equal to 0.</i>
40 OUTPUT 718;"BIT E_ERROR,STB,5;"	<i>Stores value of bit 5 of status byte in E_ERROR.</i>
50 OUTPUT 718;"E_ERROR?;"	<i>Returns value of E_ERROR.</i>
60 ENTER 718;Err	<i>Assigns value to computer variable, Err.</i>
70 IF NOT Err THEN DISP "NO ";	<i>If Err equals zero, displays "NO" on computer screen.</i>
80 DISP "ERROR PRESENT";	<i>Displays "ERROR PRESENT" on computer screen.</i>
90 END	

**Description**

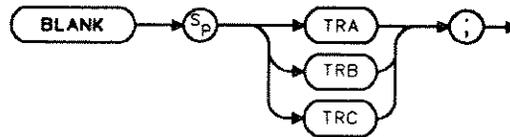
The least significant bit is bit 0. Floating-point numbers are changed to a 64-bit integer before BIT is executed.

## BLANK

### Blank Trace

The BLANK command blanks trace A, trace B, or trace C, and stops taking new data into the specified trace.

### Syntax



Preset State: BLANK TRB, BLANK TRC.

Related Commands: CLRW, MXMH, TRDSP, VIEW.

### Example

```
OUTPUT 718;"BLANK TRA;"
```

### Description

The functions of the BLANK command and **BLANK A**, **BLANK B**, and **BLANK C** are identical.

## BML

### Trace B Minus Display Line

The BML command subtracts the display line from trace B and sends the result to trace B.

#### Syntax



Related Commands: BLANK, CLRW, TS.

#### Example

10 OUTPUT 718;"IP;SNGLS"	<i>Initializes analyzer, activates single-sweep mode.</i>
20 OUTPUT 718;"BLANK TRA;"	<i>Blanks trace A.</i>
30 OUTPUT 718;"CLRW TRB;TS;"	<i>Clear-writes trace B, takes sweep.</i>
40 OUTPUT 718;"DL -70DM;"	<i>Set displays line to -70 dBm.</i>
50 OUTPUT 718;"BML;"	<i>Activates BML function.</i>
60 END	

#### Description

The BML command subtracts the display line from trace B, point by point, and sends the difference to trace B. The functions of the BML command and  $B - DL \rightarrow B$  are identical.

## BTC

### Transfer Trace B to Trace C

The BTC command transfers trace B to trace C.

#### Syntax



Related Commands: BLANK, CLRW, SNGLS, TS, VIEW.

#### Example

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
30 OUTPUT 718;"BLANK TRA;"	<i>Blanks trace A.</i>
40 OUTPUT 718;"CF 300 MHZ;SP 1MHZ;"	<i>Sets up measurement range.</i>
50 OUTPUT 718;"CLRW TRB;TS;"	<i>Takes measurement sweep.</i>
60 OUTPUT 718;"BTC;"	<i>Moves trace B to trace C.</i>
70 OUTPUT 718;"BLANK TRB;VIEW TRC;"	<i>Displays result in trace C.</i>
80 END	

#### Description

The BTC command moves trace B into trace C, then stops updating trace C by placing it in the view mode. Trace B is unchanged by BTC. Trace B must contain a complete sweep of measurement information.

The functions of BTC and `B -> C` are identical.

## BXC

### Trace B Exchange Trace C

The BXC command exchanges trace B and trace C.

#### Syntax



Related Commands: BLANK, CLRW, SNGLS, TS.

#### Example

10 OUTPUT 718;"IP;BLANK TRA"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
30 OUTPUT 718;"DET POS;CLRW TRB;TS;"	<i>Activates positive-peak detection of trace B.</i>
40 OUTPUT 718;"VIEW TRB;"	<i>Stores results and view trace B.</i>
50 OUTPUT 718;"DET SMP;CLRW TRC;"	<i>Activates sample detection.</i>
60 OUTPUT 718;"TS;"	
70 OUTPUT 718;"VIEW TRC;"	<i>Stores results of sweep in trace C.</i>
80 OUTPUT 718;"BXC;"	<i>Exchanges trace B with trace C.</i>
90 OUTPUT 718;"BLANK TRB;"	<i>Blanks trace B leaving only trace C on screen.</i>
100 END	

#### Description

The BXC command exchanges the contents of traces B and C, then stops updating them by placing them in the view mode.

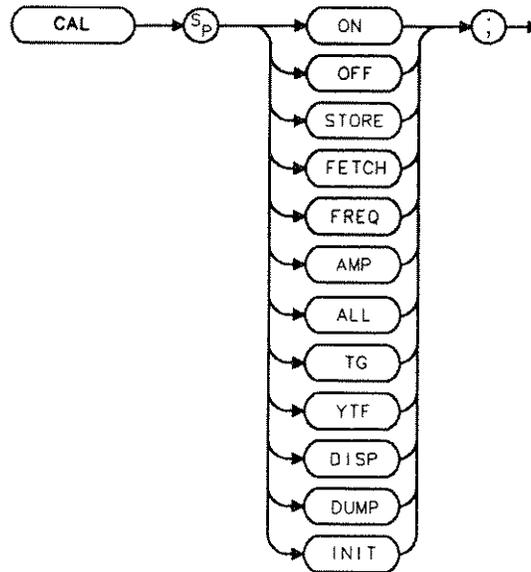
To retain all data, trace B and trace C should contain a complete sweep of measurement data before executing BXC.

The functions of the BXC command and **B <--> C** are identical.

## CAL Calibration

The CAL command controls the calibration functions.

### Syntax



### Example

OUTPUT 718;"CAL FREQ;" *Performs the frequency correction routine.*

### Description

The CAL command controls calibration functions. CAL initiates action according to the CAL parameters. The various parameters correspond to spectrum analyzer softkeys as shown below.

**ON** turns correction factors on and corresponds to **CORRECT ON OFF** (ON).

**OFF** turns correction factors off and corresponds to **CORRECT ON OFF** (OFF).

**STORE** moves the correction factors from the "working RAM" area of memory to the area of memory accessed at power up. CAL STORE corresponds to **CAL STORE**. CAL STORE is not allowed unless the correction data is valid. Correction data is valid if **CORRECT ON OFF** (ON) causes CORR to be displayed on the analyzer screen. Executing CAL STORE with invalid data generates an SRQ 110.

**FETCH** recalls the correction factors that were stored in the area of analyzer memory accessed at power-up into the area of memory that is "working RAM," and corresponds to **CAL FETCH**.

---

**Note**  Connect a cable between CAL OUT and analyzer input before running the **FREQ**, **AMP**, or **ALL** correction routines.

---

**FREQ** initiates the frequency correction routine and corresponds to **CAL FREQ**.

**AMP** initiates the amplitude correction routine and corresponds to **CAL AMPTD**.

**ALL** initiates frequency and amplitude correction routines. It corresponds to **CAL FREQ & AMPTD**.

---

**Note**  A cable must be connected between COMB OUT and analyzer input to use the comb signal for CAL YTF. The HP part number of the cable is 8120-5148.

Execute **CAL STORE** after successful completion of the self-calibration routines.

---

**YTF** initiates the correction routine for the YIG-tuned filter. It corresponds to **CAL YTF**. (HP 8592B or HP 8593A only.)

**DISP** displays some of the correction factors on the analyzer screen.

**DUMP** returns correction factors to the controller.

**INIT** sets the calibration data back to predetermined values. **CAL INIT** corresponds to **DEFAULT CAL DATA**. A center frequency of -37 Hz is required to access this function.

---

**Note**  A cable must be connected between the tracking generator output and the analyzer input for **CAL TG**.

---

**TG** performs the calibration routines if the tracking generator is installed (Option 010 or 011 for the HP 8590B or HP 8591A only).

Refer to the Installation, Verification, and Operation Manual for more information, including the correct instrument set-up for each routine. **CAL DISP** and **CAL DUMP** do not return all of the correction factors, since screen size is limited, and **CAL DUMP** only returns the factors that would be displayed on the analyzer screen. **CAL DUMP** returns the correction factors as ASCII values. Refer to the Service Manual for more information about **CAL DISP**, **CAL DUMP**, and **CAL INIT**.

If the calibration data has been corrupted or is obviously inaccurate, use **CAL FETCH** to retrieve the calibration data that has previously been saved. If the calibration data fetched is corrupt, the following example can be used to set the calibration data back to predetermined values. Execute **OUTPUT 718;"CF -37HZ;CAL INIT;"**, then perform the **CAL AMP**, **CAL FREQ**, or **CAL ALL** calibration routines. Use **CAL STORE** if the calibration routines have finished successfully.

**CAL Calibration****Note**

---

Using the default calibration data may cause the calibration routine to fail. If this occurs, execute "OUTPUT 718;CF -37HZ;" before performing the CAL FREQ or CAL ALL calibration routines.

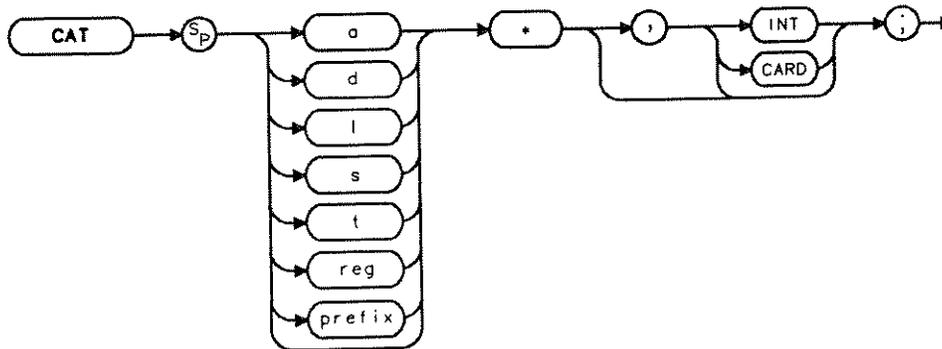
---

Executing take sweep (TS) and DONE commands after the correction routine can be used to indicate that the correction routine has finished. Use CORREK to check that the calibration routines have finished successfully.

## CAT Catalog

The CAT command returns directory information from the specified or current mass storage device.

### Syntax



Related Commands: LOAD, MSI, STOR.

### Example

This example returns the catalog information for the states stored on the memory card. Catalog information is sent with the individual catalog lines separated by a carriage return and a line feed. A carriage return, a line feed, and an EOI is asserted after the last item (equivalent to a carriage return, line feed, line feed).

<pre> 10  DIM User\$[2000],Catalog\$(1:100)[80] 20  INTEGER I,Pos_lf 30  OUTPUT 718;"CAT s*,CARD;" 40  ENTER 718 USING "#,-K";User\$ 50  I=0 60  WHILE LEN(User\$)&gt;1 70    I=I+1 80    Pos_lf=POS(User\$,CHR\$(10))  90    Catalog\$(I)=User\$[1,Pos_lf-2] 100  OUTPUT CRT;Catalog\$(I) </pre>	<p><i>Dimensions strings to store the catalog information. User\$ stores the entire string of catalog information. Catalog\$ stores the catalog information line by line (up to 80 lines and 100 characters long). I and Pos_lf are used to search through User\$ string.</i></p> <p><i>Analyzer sends catalog information for all the states stored on the memory card. Reads the catalog information into the User\$ string.</i></p> <p><i>Loops until the User\$ string is empty.</i></p> <p><i>Checks for line feeds. CHR\$(10) represents the line feed. The ASCII code for a line feed is 10.</i></p> <p><i>Extracts catalog line.</i></p> <p><i>Displays catalog line.</i></p>
---	---

**CAT Catalog**

```

120     User$=User$[Pos_lf+1]
130     END WHILE
140     END

```

**Description**

The CAT command displays the directory information for the memory card in a series of strings. Each string is within a specific field of the CAT entry as follows:

String	Position within CAT Entry	Description
Filename	1—10	
File Type	11—16	ASCII or binary data (in LIF -2 format).
Data Type	17—22	a = amplitude, d = downloadable program, l = limit line, s = state, t =trace.
Start Record	23—30	Physical record number of the start of file.
Length	31—38	Number of records in the file.
Time of creation	39—52	Time file was created in YYMMDDHHMMSS format.

Each string is right-justified within its field, with spaces separating the fields.

Guidelines for cataloging the memory card are as follows:

- CAT returns the directory information only if a file has been found.
- The asterisk (\*) character is interpreted as a wild card. The asterisk can be used with the data type to catalog all files of that data type, or alone to catalog all files on the memory card.

Cataloging internal memory returns the following information:

- MSI device (internal).
- Bytes of available analyzer memory used.
- Kilobytes or analyzer memory available.
- Name of the stored data, and the number of bytes used.

Use the asterisk, "prefix", and "reg" to catalog internal memory. For example, OUTPUT 718;"CAT \*,INT;" catalogs all the programs and variables stored in analyzer memory. OUTPUT 718;"CAT prefix\*,INT;" catalogs the programs and variables in analyzer memory according to the prefix. (Use **CHANGE PREFIX** or **PREFIX** to change the current prefix.) OUTPUT 718;"CAT reg\*,INT;" catalogs the contents of the state and trace registers. The center frequency and span of the state registers are displayed; the title and date of the trace registers are displayed.

If the mass storage device parameter (CARD or INT) is omitted, the default is the current mass storage device (memory card or analyzer memory).

**Note**

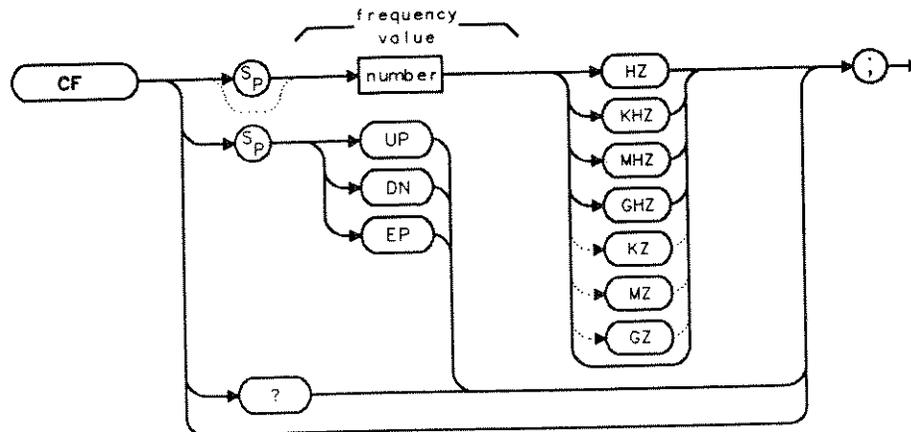
When executed from a remote port, the catalog information is sent to the remote port.



## CF Center Frequency

The CF command specifies the center frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	Frequency range of the analyzer.

Preset State HP 8590B or HP 8591A: 900 MHz.

Preset State HP 8592B or HP 8593A: 12.38 GHz.

Frequency Range HP 8590B or HP 8591A: 9 kHz to 1.8 GHz.

Frequency Range HP 8592B or HP 8593A: 9 kHz to 22 GHz, or 26.5 GHz for Option 026.

Step Increment: If uncoupled, step size is determined by the SS command. If coupled, step size is 10% of span.

Related Commands: HNLOCK, FA, FB, FOFFSET, FS, MKCF, MKSS, SP, SS.

### Note



Although the spectrum analyzer allows entry of frequencies not in the specified frequency range, using frequencies outside the frequency span of the analyzer is not recommended and is not guaranteed to meet specifications.

**CF Center Frequency****Example**

OUTPUT 718;"CF 300MHZ;" *Sets the center frequency to 300 MHz.*

**Description**

The CF command specifies the value of the center frequency. The functions of the CF command and **CENTER FREQ** are identical.

**Query Response**

## CLRAVG

### Clear Average

The CLRAVG command restarts video averaging.

### Syntax



Related Commands: AMB, CLRW, MINH, MXMH, VAVG.

### Example

```
OUTPUT 718;"IP;"           Initializes the analyzer.
OUTPUT 718;"VAVG 100;"     Initializes video averaging.
WAIT 30
OUTPUT 718;"CLRAVG;"       Restarts video averaging.
```

### Description

The CLRAVG command restarts the VAVG command by resetting the number of averaged sweeps to be one. The video averaging routine resets the number of sweeps but does not stop video averaging. Use "VAVG OFF;" to stop video averaging.

---

## CLRDSP Clear Display

The CLRDSP command erases user-generated graphics.

### Syntax



Related Commands: DSPLY, PA, PD, PR, PU, TEXT.

### Example

```
OUTPUT 718;"CLRDSP;BLANK TRA;BLANK TRB;   Blanks the analyzer screen except for trace  
GRAT OFF;TH OFF;DL OFF;"                 C and annotation.  
OUTPUT 718;"CLRW TRA;GRAT ON;ANNOT ON;"   Reinstates the default analyzer display.
```

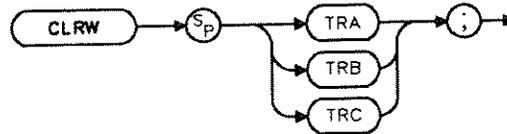
### Description

The CLRDSP command removes all user-created graphics from the analyzer screen and memory.

## CLRW Clear Write

The CLRW command clears the specified trace and enables trace data acquisition.

### Syntax



Preset State: CLRW TRA.

Related Commands: BLANK, DET, MINH, MXMH, VAVG, VIEW.

### Example

```
OUTPUT 718;"CLRW TRA;"
```

### Description

The CLRW command places the indicated trace in the clear-write mode. Data acquisition begins at the next sweep. (See "TS.")

The functions of the CLRW command and **CLEAR WRITE A**, **CLEAR WRITE B**, and **CLEAR WRITE C** are identical.

---

## CLS

### Clear Status Byte

The CLS command clears all status bits.

#### Syntax



Related Commands: RQS, SRQ, STB.

#### Example

```
OUTPUT 718;"CLS;"
```

#### Description

The CLS command clears all status bits (see "SRQ" for more information on status byte, status register, and service requests).

---

## CNF Confidence Test

The CNF command performs the confidence test.

### Syntax



### Example

```
OUTPUT 718; "CNF;"
```

### Description

The CNF command initiates a confidence test of the resolution bandwidth, video bandwidth, and step gain.

---

### Note



Connect a cable between CAL OUT to analyzer input before initiating the CNF function.

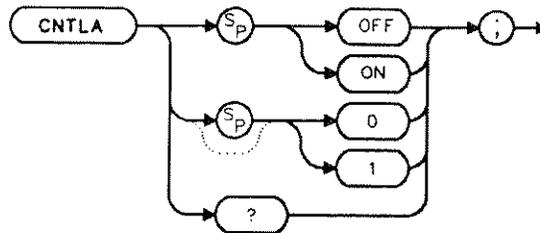
---

The functions of the CNF command and **CONF TEST** are identical.

## CNTLA Auxiliary Control Line A

The CNTLA sets the control line A of the auxiliary interface high or low.

### Syntax



Related Commands: CNTLB, CNTLC, CNTLD, CNTLI.

### Example

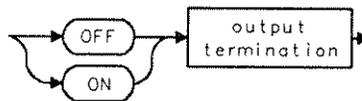
```
OUTPUT 718;"CNTLA ON;"  Sets control line A high.
OUTPUT 718;"CNTLA?;"   Queries the state of control line A.
ENTER 718;A$           Gets the response from the analyzer.
DISP A$                Displays the response.
```

### Description

The auxiliary connector outputs use transistor-transistor logic.

The functions of the CNTLA command and `CNTL A 0 1` are identical.

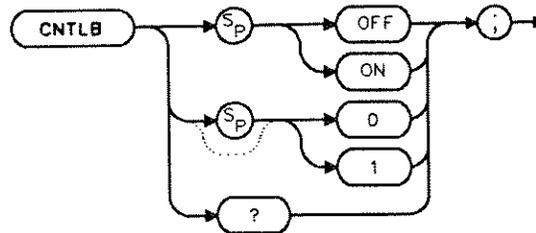
### Query Response



## CNTLB Auxiliary Control Line B

The CNTLB sets the control line B of the auxiliary interface high or low.

### Syntax



Related Commands: CNTLA, CNTLC, CNTLD, CNTLI.

### Example

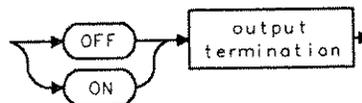
OUTPUT 718;"CNTLB ON;"	<i>Sets control line B high.</i>
OUTPUT 718;"CNTLB?;"	<i>Queries the state of control line B.</i>
ENTER 718;A\$	<i>Gets the response from the analyzer.</i>
DISP A\$	<i>Displays the response.</i>

### Description

The auxiliary connector outputs use transistor-transistor logic.

The functions of the CNTLB command and **CNTL B 0 1** are identical.

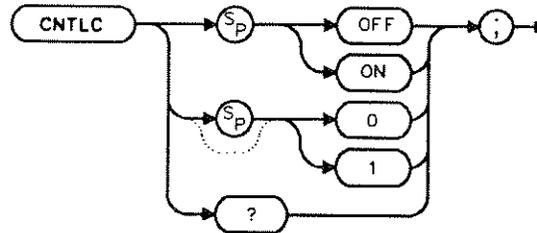
### Query Response



## CNTLC Auxiliary Control Line C

The CNTLC sets the control line C of the auxiliary interface high or low.

### Syntax



Related Commands: CNTLA, CNTLB, CNTLD, CNTLI.

### Example

```

OUTPUT 718;"CNTLC ON;"  Sets control line C high.
OUTPUT 718;"CNTLC?;"   Queries the state of control line C.
ENTER 718;A$           Gets the response from the analyzer.
DISP A$                Displays the response.

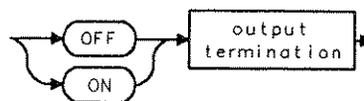
```

### Description

The auxiliary connector outputs use transistor-transistor logic.

The functions of the CNTLC command and **CNTL C 0 1** are identical.

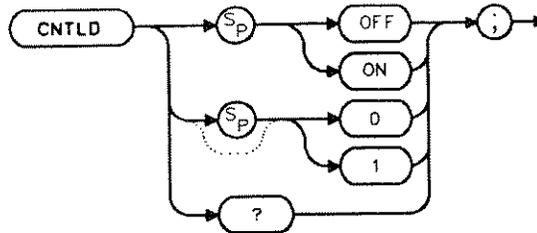
### Query Response



## CNTLD Auxiliary Control Line D

The CNTLD sets the control line D of the auxiliary interface high or low.

### Syntax



Related Commands: CNTLA, CNTLB, CNTLC, CNTLI.

### Example

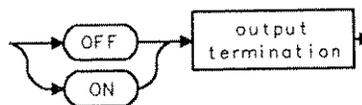
OUTPUT 718;"CNTLD ON;"	<i>Sets control line D high.</i>
OUTPUT 718;"CNTLD?;"	<i>Queries the state of control line D.</i>
ENTER 718;A\$	<i>Gets the response from the analyzer.</i>
DISP A\$	<i>Displays the response.</i>

### Description

The auxiliary connector outputs use transistor-transistor logic.

The functions of the CNTLD command and **CNTL D 0 1** are identical.

### Query Response

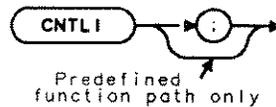


## CNTLI

### Auxiliary Control Line Input

The CNTLI returns a "1" when the control line I of the auxiliary interface is high, and "0" if the line input is low.

### Syntax



Related Commands: CNTLA, CNTLB, CNTLC, CNTLD.

### Example

```
OUTPUT 718;"CNTLI;"  Gets the status of control line I.  
ENTER 718;A  
DISP A              Displays status.
```

### Description

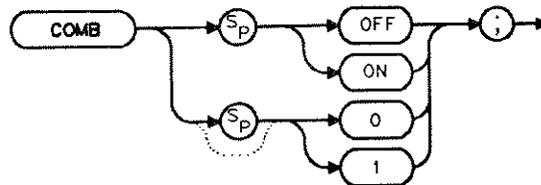
The functions of CNTLI and **DISPLAY CNTL I** are similar.

## COMB Comb

The COMB command turns the comb generator on or off.

Model Required: HP 8592B or HP 8593A

### Syntax



### Example

OUTPUT 718;"COMB ON;" *Turns the comb generator on.*

### Description

#### Note



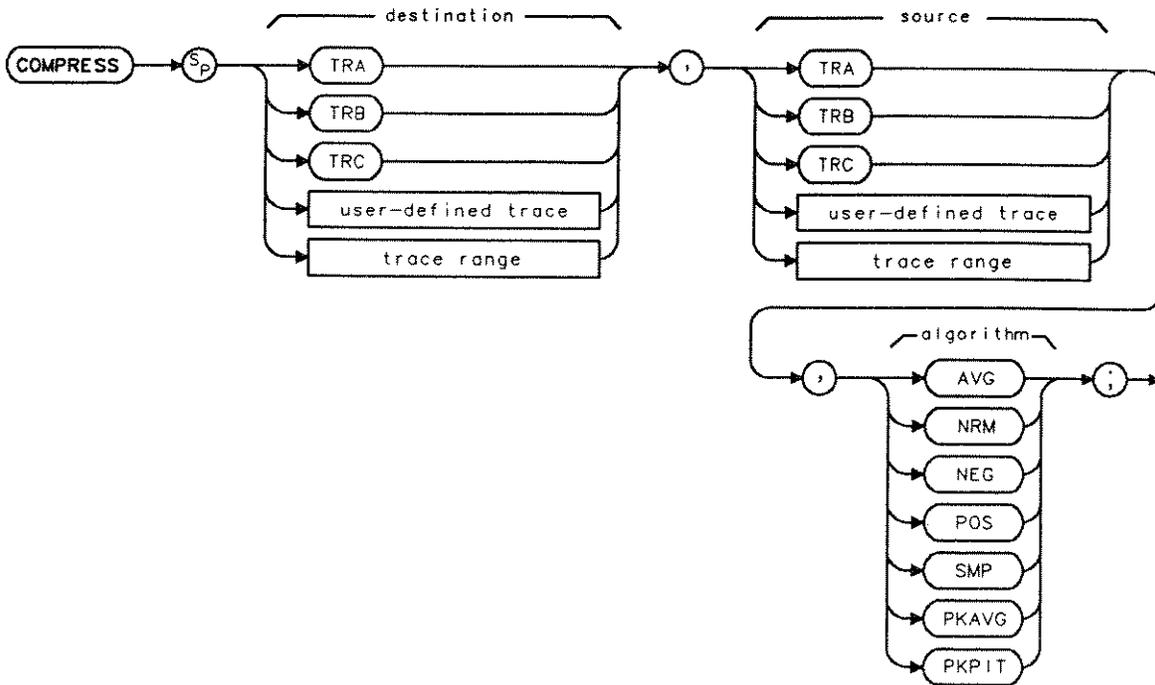
A cable must be connected between COMB OUT and analyzer input to use the comb signal.

The functions of the COMB command and **COMB GEN ON OFF** are identical.

## COMPRESS Compress Trace

The COMPRESS command reduces the number of trace elements while retaining the relative frequency and amplitude characteristics of the trace data.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command	Any valid trace name.
Trace Range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using user-defined trace. TS when using trace data.

## Example

This example compresses trace B into "C.OMPTRAC" using POSITIVE algorithm.

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"TRDEF C_OMPTRAC,100;"	<i>Creates a trace called C.OMPTRAC with a length of 100 elements.</i>
30 OUTPUT 718;"BLANK TRA;SNGLS;"	<i>Blanks trace A, activates single-sweep mode.</i>
40 OUTPUT 718;"CLRW TRB;TS;"	<i>Measures with trace B.</i>
50 OUTPUT 718;"COMPRESS C_OMPTRAC,TRB,POS;"	<i>Compresses trace B into C.OMPTRAC.</i>
60 OUTPUT 718;"BLANK TRB;"	<i>Blanks trace B.</i>
70 OUTPUT 718;"MOV TRA,C_OMPTRAC;"	<i>Moves C.OMPTRAC into trace A.</i>
80 OUTPUT 718;"VIEW TRA;"	<i>Displays the result.</i>
90 END	

## Description

The COMPRESS command stores a compressed copy of the source trace into a smaller destination trace. The source trace is divided into the same number of intervals as there are points in the destination trace, and the data within each interval are compressed into the value for the corresponding destination trace point. The algorithm used to compress the data is given as an argument to the command.

The algorithms available are:

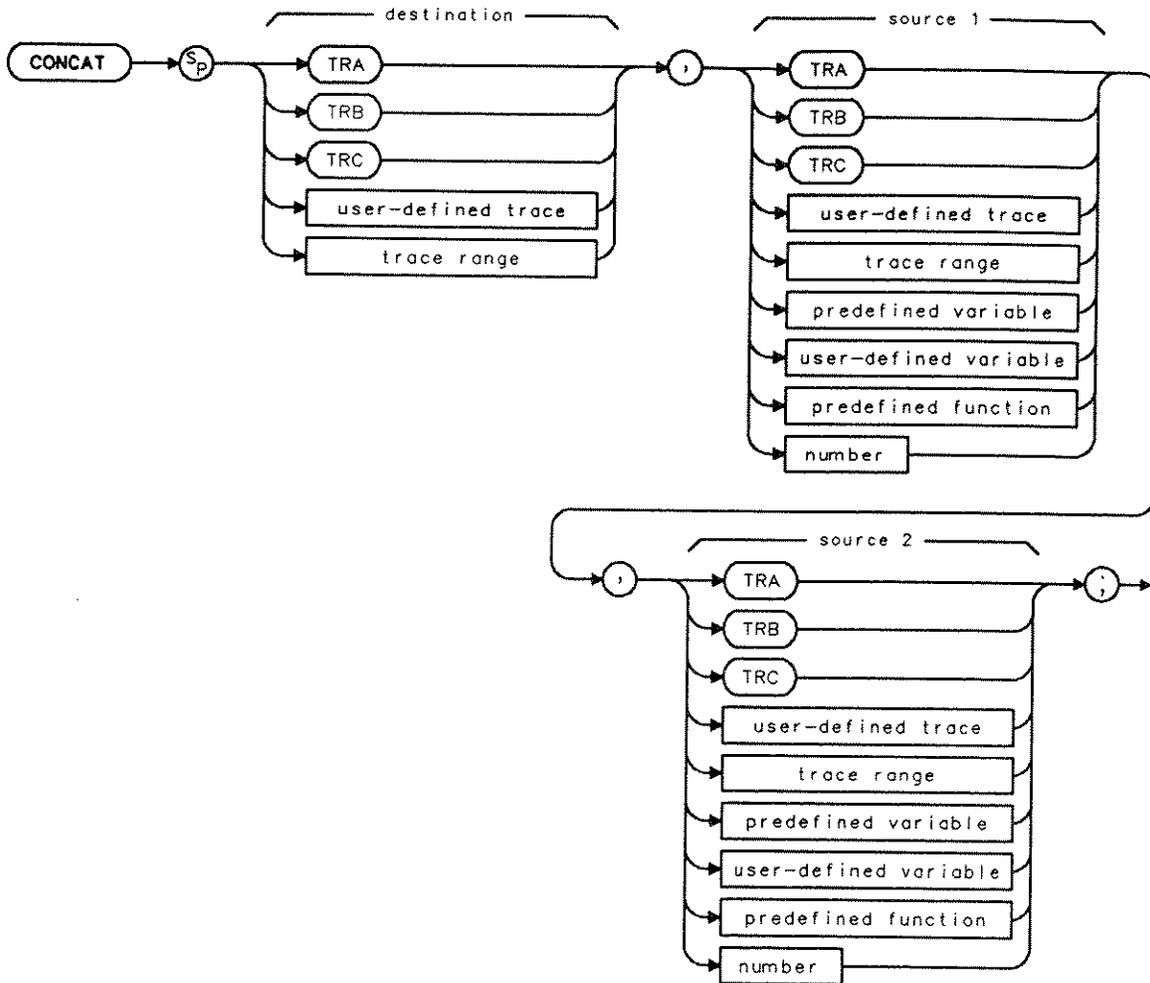
AVERAGE (AVG)	The average of the points within an interval is used.
NORMAL (NRM)	NRM computes the compressed value of the interval by using a Rosenfell algorithm. The Rosenfell algorithm is a mathematical operation defined in analyzer firmware. It compresses a locally continuously rising or falling signal into the peak value detected in each interval. If the detected signal is not continuously rising or falling, then the data value alternates between minimum and maximum in the compressed interval. This shows the peak-to-peak noise variations better. The Rosenfell option is useful for accurately displaying noise, since peak detection can give misleading representation of noise.
NEGATIVE (NEG)	The lowest value in each interval is used.
POSITIVE (POS)	Specifying POS selects the highest point in the interval as the compressed value.
SAMPLE (SMP)	Specifying SMP selects the last point in the interval as the compressed value.
PEAK AVERAGE (PKAVG)	The PKAVG algorithm selects the difference between the peak and the average value of the interval as the compressed value.
PEAK PIT (PKPIT)	The PKPIT algorithm returns the difference between the positive and negative peaks within the interval.

# CONCAT

## Concatenate

The CONCAT command combines two traces.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that return a value. Refer to Table 4-1.	
Trace Range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.  
 Related Commands: MOV, VIEW.

**Example**

<pre>10 OUTPUT 718;"IP;" 20 OUTPUT 718;"SNGLS;" 30 OUTPUT 718;"TS;VIEW TRA;" 40 OUTPUT 718;"CLRW TRB;TS;" 50 OUTPUT 718;"CONCAT TRC,TRA[1,200],TRB[201,401];"  60 OUTPUT 718;"BLANK TRA;BLANK TRB;" 70 OUTPUT 718;"VIEW TRC;" 80 END</pre>	<p><i>Initializes the analyzer, activates the single-sweep mode.</i></p> <p><i>Stores and view trace A.</i></p> <p><i>Measures with trace B.</i></p> <p><i>Concatenates the last half of trace B to the first half of trace A.</i></p> <p><i>Blanks trace A and trace B.</i></p> <p><i>Displays the result.</i></p>
--	---

**Description**

The CONCAT command concatenates source 2 to the end of source 1, then stores the new array in the destination. If the destination length is less than the length of source 1 and source 2, the list is truncated. For example, executing "CONCAT TRA[1,15],TRB[3,6],TRB[7,207];" results in trace A elements 1—4 being replaced by trace B elements 3—6, and trace A elements 5—15 being replaced by trace B elements 7—17. Trace B elements 18—207 are ignored. If necessary, reduce trace lengths with COMPRESS, TRA, TRB, TRC, or TRDEF.

If the length of the destination is greater than the length of source 1 and source 2, the last value is repeated. For example, executing "CONCAT TRA[1,15],TRB[3,6],TRB[7,9];" results in trace A elements 1—4 being equal to trace B elements 3—6, trace A elements 5 and 6 being replaced by trace B elements 7 and 8, and trace A elements 7—15 being replaced by trace B element 9.

---

## CONTS

### Continuous Sweep

The CONTS command sets the analyzer to the continuous sweep mode.

#### Syntax



Preset State: CONTS.

Related Commands: SNGLS, ST, TM.

#### Example

```
OUTPUT 718;"CONTS;"
```

#### Description

The CONTS command sets the analyzer to continuous sweep mode. In the continuous sweep mode, the analyzer takes its next sweep as soon as possible after the current sweep as long as the trigger conditions are met. A sweep may temporarily be interrupted by data entries made from the front panel or over the remote interface.

The functions of the CONTS command and **SWEEP CONT SGL** (CONT) are identical.

## CORREK Correction Factors On

The CORREK command returns a "1" if the correction factors are on, a "0" if they are off.

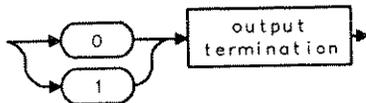
### Syntax



### Example

```
OUTPUT 718;"CORREK;"  
ENTER 718;A  
DISP A
```

### Query Response

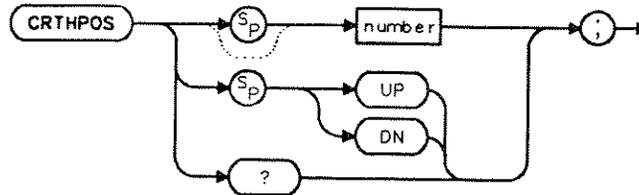


## CRTHPOS

### Horizontal Position of CRT Display

The CRTHPOS command specifies the horizontal position of the analyzer's display.

#### Syntax



Item	Description/Default	Range
Number	Any valid integer number.	1 to 34.

Default Value: 10.

Step Increment: by 1.

Related Commands: CRTVPOS.

#### Example

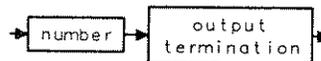
```
OUTPUT 718;"CRTHPOS 15;"
```

#### Description

The CRTHPOS command specifies the horizontal position of the analyzer's display. Each CRTHPOS number represents four screen pixels. The analyzer display is 512 screen pixels wide by 256 screen pixels high. Execute CAL STORE to change the default value of the horizontal position.

The functions of the CRTHPOS command and **CRT HORZ POSITION** are identical.

#### Query Response

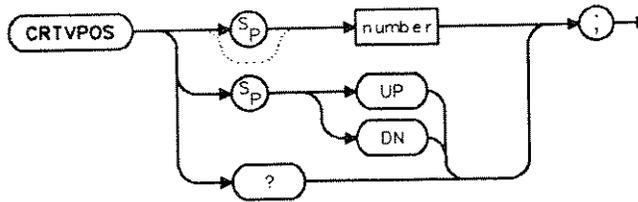


# CRTVPOS

## Vertical Position of CRT Display

The CRTVPOS command specifies the vertical position of the analyzer's display.

### Syntax



Item	Description/Default	Range
Number	Any valid integer number.	10 to 58.

Default Value: 48.  
 Step Increment: by 1.  
 Related Commands: CRTHPOS.

### Example

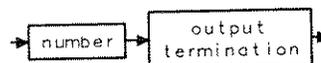
```
OUTPUT 718;"CRTVPOS 50;"
```

### Description

The CRTVPOS command specifies the vertical position of the analyzer's display. Each CRTVPOS number represents four screen pixels. The analyzer display is 512 screen pixels wide by 256 screen pixels high. Execute CAL STORE to change the default value of the vertical position.

The functions of the CRTVPOS command and **CRT VERT POSITION** are identical.

### Query Response

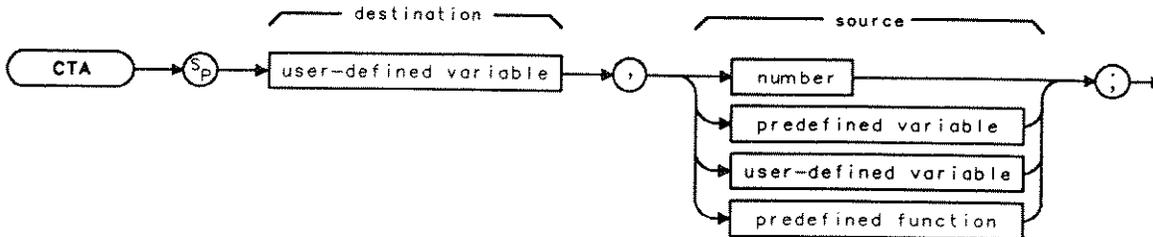


## CTA

### Convert to Absolute Units

The CTA command converts the source values from measurement units to the current absolute amplitude units and stores this result in the destination.

#### Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Number	Any valid integer number.	-32,768 to +32,767.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that return a value. Refer to Table 4-1.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: AUNITS, LG, LN, RL.

#### Example

```

10 OUTPUT 718;"VARDEF C_ONLV,0;"   Declares a variable called C_ONLV.
20 OUTPUT 718;"LG;"               Puts the analyzer in log mode.
30 OUTPUT 718;"RL -10DB;"         Changes the reference level.
40 OUTPUT 718;"CTA C_ONLV, 8000;"  -10 is stored in C_ONLV.
50 OUTPUT 718;"RL -20DB;"         Changes the reference level.
60 OUTPUT 718;"CTA C_ONLV,8000;"  -20 is now stored in C_ONLV.
70 OUTPUT 718;"C_ONLV?;"
80 ENTER 718;A
90 DISP A
100 END
  
```

## Description

The result of the CTA command depends on the reference level, the current amplitude units, and the amplitude scale (log or linear).

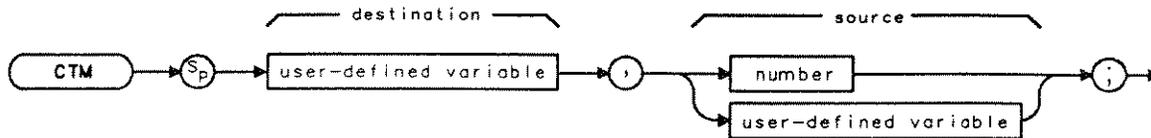
Measurement units are the 16-bit amplitude values stored in traces. The values range from -32,768 to 32,767. The value of 8000 corresponds to an amplitude equal to the reference level. In log mode, each count represents 0.01 dBm. A signal 0.01 dBm above the reference level is at 8001, and a signal 1.0 dBm below the reference level is at 8000 minus 100, or 7900. In linear mode, 8000 is the reference level and 0 is the 0 volt level. If the reference level is at 80 mV, each count would represent  $0.080/8000$  or  $10 \mu\text{V}$ , but a reference level of 2.4 volts would represent  $2.4/8000$  or  $300 \mu\text{V}$  per count.

## CTM

### Convert to Measurement Units

The CTM command converts the source values to measurement units and places the result in the destination.

#### Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands	Any valid variable name.
Number	A number expressed in integer, decimal, or exponential form.	Real number range.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.

Related Commands: AUNITS, CTA, LG, LN, RL.

#### Example

```
OUTPUT 718;"VARDEF A_MPV,0;"    Declares a variable called A_MPV.
OUTPUT 718;"LG;"              Puts the analyzer in log mode.
OUTPUT 718;"RL -10DB;"        Changes the reference level.
OUTPUT 718;"CTM A_MPV,-10;"    8000 is stored in A_MPV.
OUTPUT 718;"DSPLY A_MPV,13.5;" Displays A_MPV.
```

#### Description

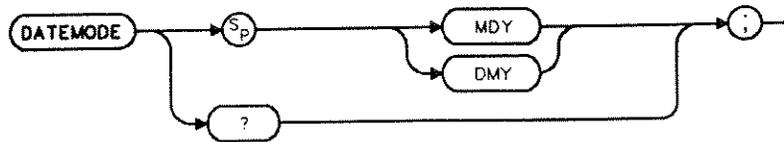
The result of the CTM command depends on the reference level, the current amplitude units, and the amplitude scale (log or linear).

Measurement units are the 16-bit amplitude values stored in traces. The values range from -32,768 to 32,767. The value of 8000 corresponds to an amplitude equal to the reference level. In log mode, each count represents 0.01 dBm. A signal 0.01 dBm above the reference level is at 8001, and a signal 1.0 dBm below the reference level is at 8000 minus 100, or 7900. In linear mode, 8000 is the reference level and 0 is the 0 volt level. If the reference level is at 80 mV, each count would represent 0.080/8000 or 10  $\mu$ V, but a reference level of 2.4 volts would represent 2.4/8000 or 300  $\mu$ V per count.

## DATEMODE Date Mode

The DATEMODE command allows you to set the format for displaying the real-time clock in the month, day, year format or the day, month, year format.

### Syntax



### Example

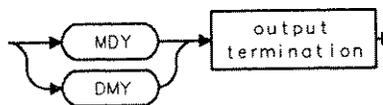
```

OUTPUT 718;"DATEMODE DMY;"  Sets the date mode to day, month, year format.
OUTPUT 718;"DATEMODE?;"    Queries the format of the display of the real-time clock.
ENTER 718;A$
DISP A$
  
```

### Description

The functions of the DATEMODE command and `DATEMODE MDY DMY` are identical.

### Query Response



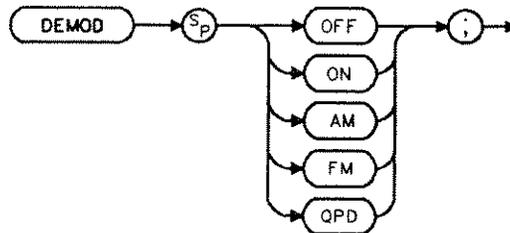
## DEMODO

### Demodulation

The DEMOD command turns the demodulator on or off, and selects between AM, FM, or quasi-peak demodulation.

**Option and Model Required:** Option 102 with HP 8591A or HP 8593A

### Syntax



Related Commands: DET, FMGAIN, HAVE, MKPAUSE, SP, SPEAKER, SQLCH.

### Example

This example demonstrates FM demodulation in a span greater than 0.

10 OUTPUT 718;"IP;FA 90MHZ;"	<i>Sets start frequency.</i>
20 OUTPUT 718;"FB 110MHZ;"	<i>Sets stop frequency.</i>
30 OUTPUT 718;"TS;MKPK HI;MKCF;"	<i>Places marker on the highest peak and brings the peak to center frequency.</i>
40 OUTPUT 718;"DEMODO ON;DEMODO FM;"	<i>Turns on FM demodulation.</i>
50 OUTPUT 718;"MKPAUSE 500MS;"	<i>Turns on marker pause. The detector is automatically switches to the FMV detector during the dwell time.</i>
60 END	

### Description

Executing "DEMODO AM;" or "DEMODO FM;" does not turn on demodulator, it just selects the AM or FM demodulation mode.

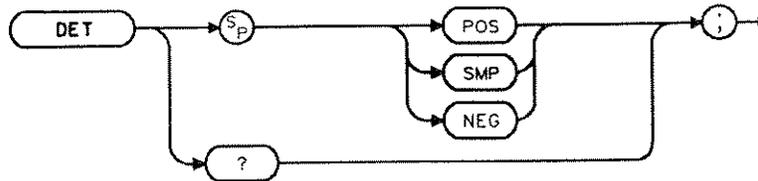
For demodulation in nonzero frequency spans, use MKPAUSE to set the dwell time of the marker.

Quasi-peak demodulation is available with Option 103. See the documentation accompanying Option 103 for more information.

## DET Detection Mode

The DET command selects the analyzer detection mode.

### Syntax



Preset State: DET POS.

Related Commands: MKNOISE, MXMH, VAVG.

### Example

10 OUTPUT 718;"IP;"	<i>Initializes the analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
30 OUTPUT 718;"DET POS;TS;"	<i>Activates the positive-peak detection of trace A .</i>
40 OUTPUT 718;"VIEW TRA;"	<i>Stores results in trace A.</i>
50 OUTPUT 718;"DET SMP;"	<i>Activates sample detection for trace B.</i>
60 OUTPUT 718;"CLRWB TRB;TS;"	<i>Measures with trace B.</i>
70 OUTPUT 718;"VIEW TRB;"	<i>Stores results in trace B.</i>
80 OUTPUT 718;"AVG TRA,TRB,2;"	<i>Averages trace A and B with a ratio of 2, and store results in trace A.</i>
90 OUTPUT 718;"BLANK TRB;"	<i>Blanks trace B, leaving only averaged results on-screen.</i>
100 END	

### Description

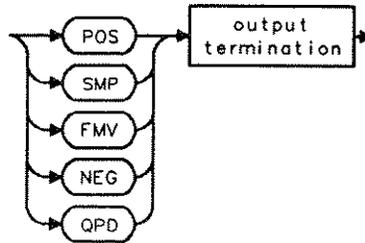
The DET command selects the type of analyzer detection (positive-peak or sample) and accesses service-diagnostic detection functions. The functions of DET POS, DET SMP and DETECTOR SAMPL PK are identical.

- POS enables positive-peak detection, which displays the maximum video signal detected over a number of instantaneous samples for a particular frequency.
- SMP enables sample detection, which uses the instantaneous signal value detected at the analyzer-to-digital converter output. Video averaging and noise-level markers, when activated, automatically activate sample detection.
- NEG enables negative peak detection in sweep times of less than or equal to 200 ms (HP 8591A or HP 8593A with Option 101 only).

**DET Detection Mode**

**Query Response**

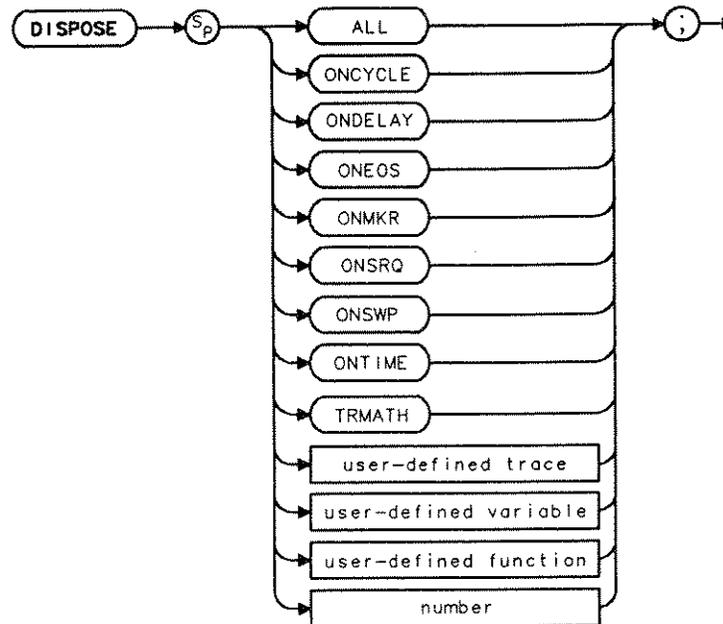
FMV or QPD may be returned. DEMOD FM uses the FMV detector, DEMOD QPD uses the QPD detector. The DEMOD command is available for Option 102 or 103.



## DISPOSE Dispose

The DISPOSE command allows the user to free user memory which has been allocated previously for user-defined operands.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by the VARDEF or ACTDEF commands.	Any valid variable name.
User-defined function	A subroutine defined by the FUNCDEF command.	Any valid function name.
Number	Any valid softkey number.	1—6, 601—1200.

Prerequisite Commands: TRDEF when using a user-defined trace. VARDEF or ACTDEF when using a user-defined variable.

Related Commands: ERASE, KEYCMD, KEYDEF, KEYLBL, LIMIDEL, ONCYCLE, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, ONTIME, TRMATH.

**DISPOSE** Dispose**Example**

```
OUTPUT 718;"VARDEF T_EMP,10;" Defines a variable for use in a program. When the
variable is no longer needed, it can be purged using the
DISPOSE command
```

```
OUTPUT 718;"DISPOSE T_EMP;"
```

**Description**

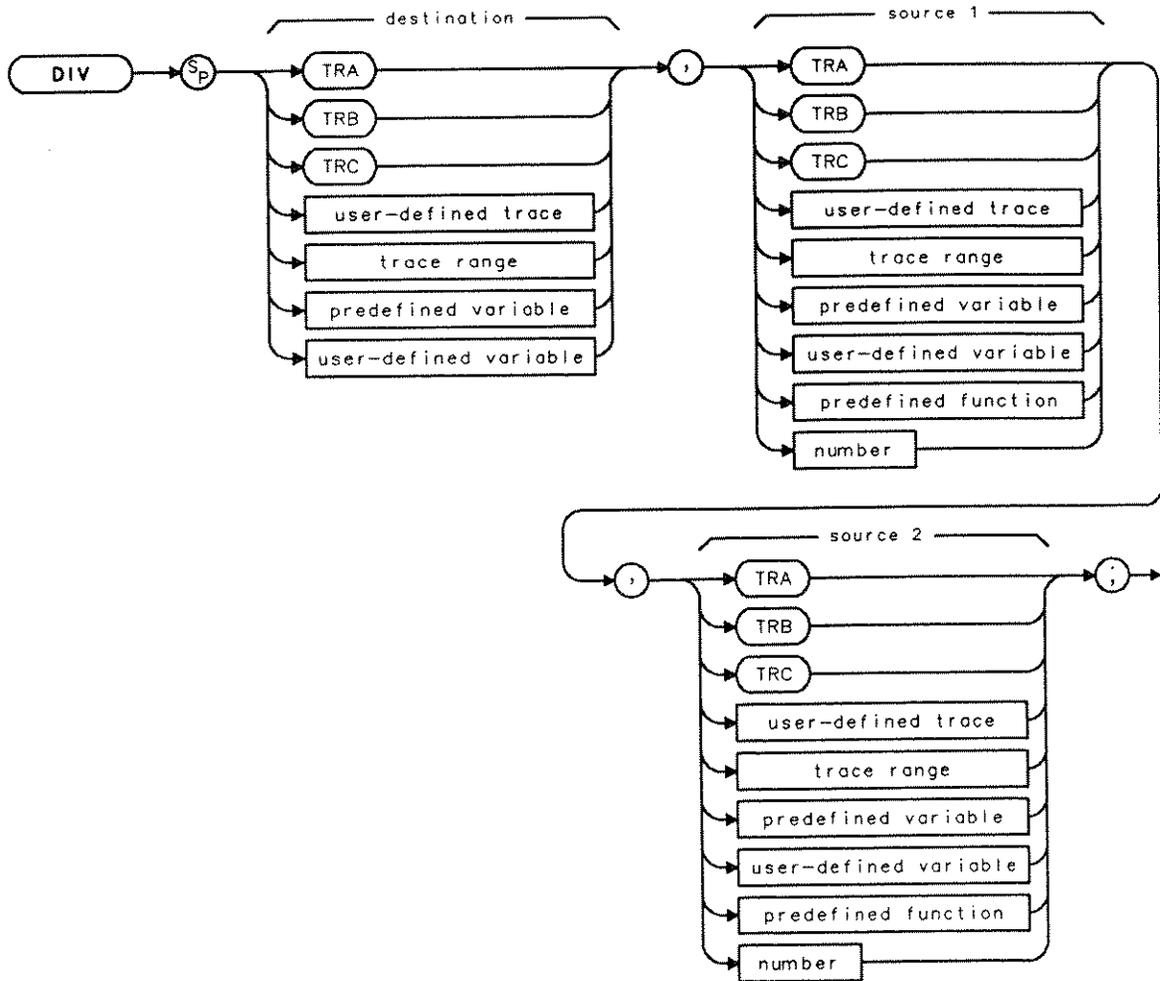
The DISPOSE ALL function frees user memory (except the state registers and predefined traces), to make the total available user-allotted memory the maximum size.

The functions of DISPOSE ALL and **DISPOSE USER MEM** are identical.

# DIV Divide

The DIV command divides source 1 by source 2 and places the result in the destination.

## Syntax



**DIV Divide**

Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

Related Commands: MPY, SNGLS, TS.

**Example**

OUTPUT 718;"IP;"	<i>Initializes the analyzer.</i>
OUTPUT 718;"SNGLS;"	<i>Activates single-sweep mode.</i>
OUTPUT 718;"TS;"	<i>Updates trace information.</i>
OUTPUT 718;"DIV TRB,TRA,2;"	<i>Divides trace A by two and place in trace B.</i>
OUTPUT 718;"VIEW TRB;"	<i>Displays the result.</i>

**Description**

Integer values are used when a trace is the destination or one of the sources. If trace data is used as the source and destination, the DIV function is done with 32-bit arithmetic on 16-bit integer data. If a user-defined variable or predefined variable is used as the source or destination, the DIV function is done in floating-point format. If a real number is used as a source, but the destination is an integer value, the result is truncated. If traces are used, be sure each contains a complete sweep of measurement information before executing DIV.

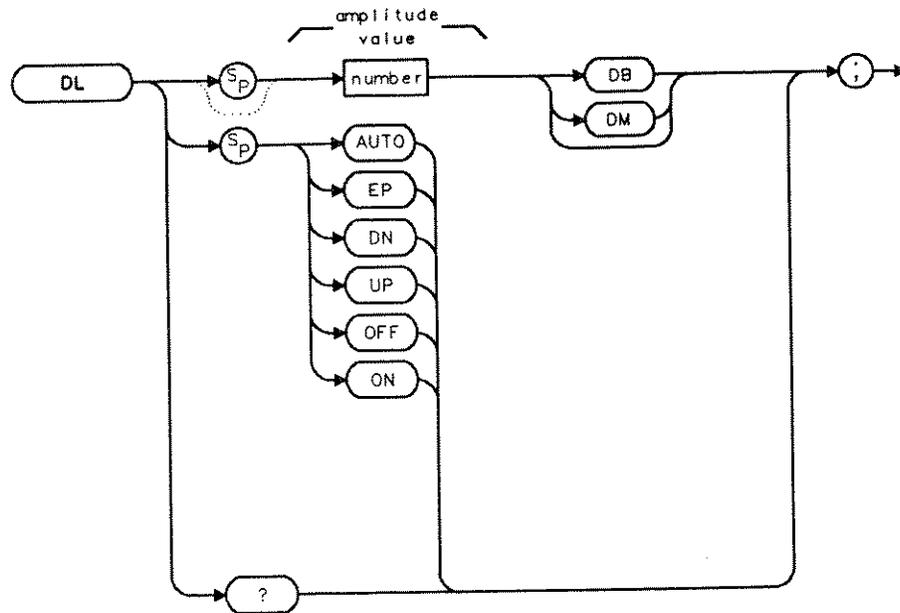
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

The results of the DIV function are invalid if source 2 is equal to zero.

## DL Display Line

The DL command defines the level of the display line in the active amplitude units and displays it on the analyzer screen.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default units are dBm.	Dependent on reference level.

Preset State: DL OFF.

Step Increment: 1 major division.

Related Commands: AUNITS, AUTO, LG, LN, NRL, RL, ROFFSET, TH.

### Example

```
OUTPUT 718;"AUNITS DBM;"  Changes the active amplitude units to dBm.
OUTPUT 718;"DL ON;"      Turns display line on.
OUTPUT 718;"DL -5DM;"    Changes display line to -5 dBm.
```

## DL Display Line

### Description

Activating video trigger mode activates the display line. The AUTO command and “DL OFF;” turn off the display line. The functions of the DL command and **DSP LINE ON OFF** are identical. See “AUNITS” for more information on changing the active amplitude units.

### Query Response



## DN Down

The DN command reduces the active function by the applicable step size.

### Syntax



Related Commands: ACTDEF, AT, CF, DL, FA, FB, LG, MKA, MKD, MKN, MKF, MKSS, NRL, RB, RL, RLPOS, SP, SRCAT, SRCPOFS, SRCPWR, SRCPSTP, SRCPSWP, SRCTK, SS, ST, TH, UP, VB, VBR.

### Example

```
OUTPUT 718;"SS 1MHZ;CF 1GHZ;DN;" Sets center frequency to 1 GHz.  
OUTPUT 718;"SP 40MHZ;MKPK;DN;" Decreases the frequency span.
```

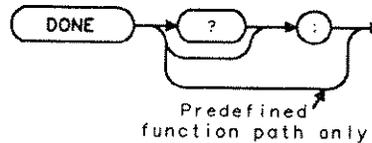
The second line of the example decreases the span, since marker peak (MKPK) is not an active function.

## DONE

### Done

The DONE command is a way of remotely determining when the analyzer has parsed and started all commands prior to and including DONE. It returns a value of "1" when all commands in a command string entered before DONE have been started.

### Syntax



Related Commands: TS.

### Example

```
OUTPUT 718;"IP;CF 1GHZ;SP 1GHZ;DONE;"
```

*Since TS does not precede the DONE command in this line, the center frequency and span values are set before 1 is returned, but functions coupled to SP, such as RB, have not been changed, and there is no trace data associated with the new frequency settings.*

```
OUTPUT 718;"IP;CF 1GHZ;SP 1GHZ;TS;DONE;"
```

*Since the take sweep must be completed before the DONE command is executed, the autocoupled functions and trace data have been changed before the DONE command is executed.*

```
ENTER 718;Done
```

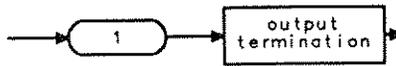
*Stores 1 in computer variable, called Done.*

```
DISP Done
```

### Description

As shown by the example, if a take sweep (TS) precedes the DONE command, DONE is executed after all the preceding commands have been completed. Use a take-sweep (TS) to ensure all previous commands have completed before DONE returns a "1."

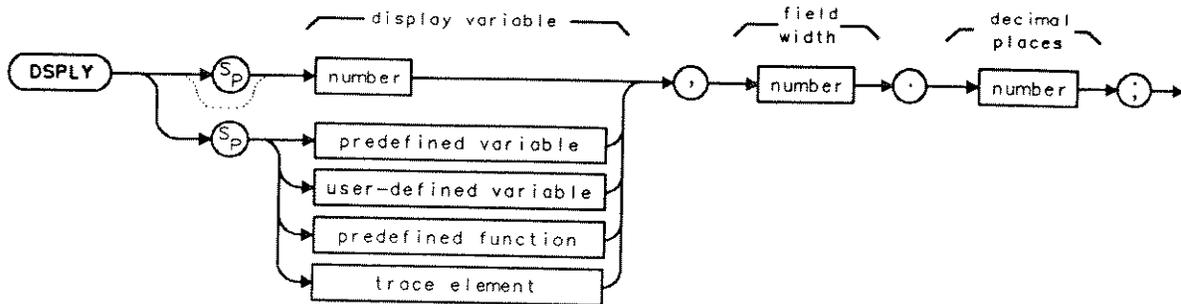
### Query Response



## DSPLY Display

The DSPLY command displays the value of a variable on the analyzer screen.

### Syntax



“Field width” specifies the number of digits displayed, including sign and decimal point. (Places to the right of the decimal point are limited by decimal places.) For example, the number 123.45 has a field of 7 (even though the sign is not displayed), and 2 decimal places.

Item	Description/Default	Range
Number	Any real or integer number.	Real number range.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: PA, PD, PR, PU.

### Example

```

    OUTPUT 718;"CF 300MHZ;"           Sets center frequency to 300 MHz.
    OUTPUT 718;"PU;PA 50,150;DSPLY CF, Center frequency is displayed to three decimal
    13.3;"                             places in a field width of 13.
    
```

## Description

The DSPLY command displays the value of a variable anywhere on the spectrum analyzer display. Use the PU, PR, and PA commands to position the variable on the screen.

The analyzer displays the data independent of PU or PD (PU and PD do not have to be specified).

The coordinates given relate to the lower left-hand corner of the first character. For example, "PU;PA 50,100;DSPLY CF,13.3;" would display the center frequency with the first digit to the right and above the point represented by 50, 100. Position the pen before executing the DSPLY command. If the number to be displayed is larger than the field width, the number is displayed in scientific notation.

## DT Define Terminator

The DT command defines any character as a label terminator. The label terminator is only used with the LB command.

### Syntax



Item	Description/Default	Range
Character	Any valid character.	

Related Commands: LB.

### Example

```
OUTPUT 718;"DT@";
```

*Defines the @ symbol as the text terminator.*

```
OUTPUT 718;"CF 600MHZ;"
```

*Sets center frequency to 600 MHz.*

```
OUTPUT 718;"LB CAL OUT 2ND HARMONIC@RL ODM;"
```

*Displays the label "CAL OUT 2ND HARMONIC" on the analyzer screen, then executes the RL command.*

### Description

The DT command is used by the LB command to separate text from analyzer commands.

## EE Enable Entry

The EE command sends values entered by the operator on the analyzer numeric keypad to the controller.

### Syntax



Related Commands: RQS.

### Example

The following example works with RS-232 and HP-IB interfaces; however, the address (718) must be changed for RS-232 operation.

The program polls the analyzer for its status byte.

```
10 PRINTER IS 1
20 DISP "Enter new value with analyzer
number pad and terminate with units key"
30 OUTPUT 718;"RQS 2;CF EE;"

40 REPEAT
50  OUTPUT 718;"STB?;"
60  ENTER 718;A
70  UNTIL BIT(A,1)=1

80 DISP
90  OUTPUT 718;"?;"
100 ENTER 718;New_value
110 PRINT "You entered the value of ";
    New_value
120 OUTPUT 718;"HD;"
130 END
```

*Outputs to the computer screen.*

*Sets up SRQ mask, then uses the enable entry mode.*

*Polls the analyzer until entry is complete. Asks the analyzer for status byte.*

*Does REPEAT UNTIL loop until user presses terminating key.*

*Asks for entry.*

*Displays value.*

*Turns off enable entry mode.*

**EE Enable Entry****Description**

Unlike enter parameter (EP), EE does not terminate the entry mode when the operator completes an entry.

The EE command is used generally in a sequence of programmed events:

1. A program loop prevents the controller from using the entered value until the operator signals that the entry is complete.
2. The operator makes a data entry, which is stored in the analyzer internal data register.
3. The operator indicates completion of the entry.
4. The controller reads the value of the entry and continues to the next program step.

## EK Enable Knob

The EK command allows data entry with the front-panel knob when the analyzer is under remote control.

### Syntax



### Example

1 PRINTER IS 1	
10 OUTPUT 718;"MKN;EK;"	<i>Marker is activated. Front-panel knob is enabled.</i>
20 PRINT "USE KNOB TO PLACE MARKER"	<i>While the program pauses, the operator positions a marker on a signal that needs further analysis.</i>
30 PRINT "PRESS CONTINUE WHEN DONE"	<i>Prompts user.</i>
40 PAUSE	<i>The operator positions a marker on a signal that needs further analysis.</i>
:	<i>Analysis program inserted here.</i>
100 END	

### Description

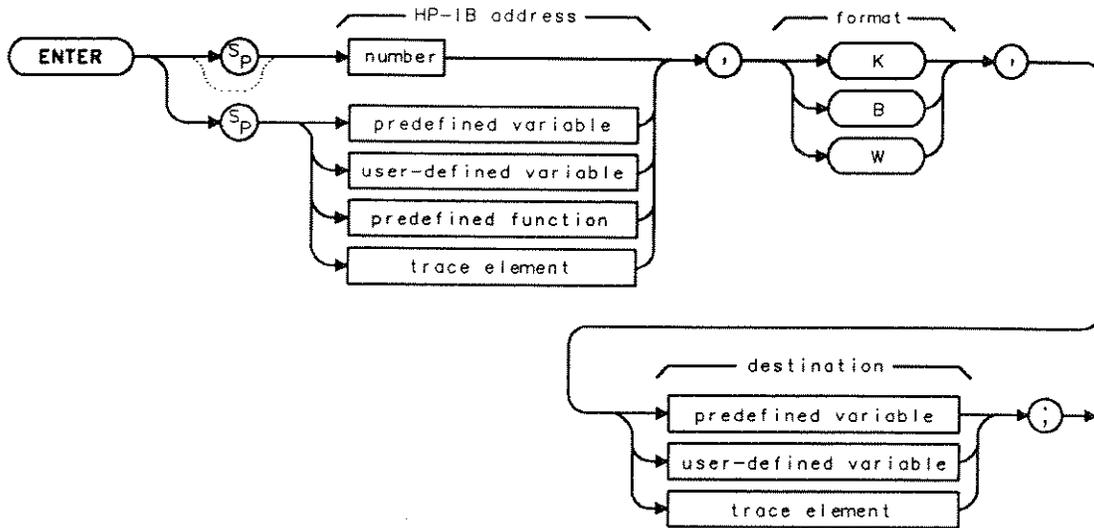
With the EK command, the knob is functional but other front-panel functions remain inoperative. Moving the knob changes the active function. If no function is active, moving the knob has no effect.

# ENTER

## Enter From HP-IB

The ENTER command establishes the analyzer as a controller on the HP-IB.

### Syntax



Item	Description/Default	Range
Number	Any valid integer number.	0 to 30.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: FUNCDEF, OUTPUT, RELHPIB.

### Example

#### Note



The plotter is at address 5 and the analyzer is at address 18. (The program is only valid for HP 9000 Series 200 and 300 computers.)

The example uses the analyzer to send the ASCII code for OP; (output parameter) to the plotter. The ENTER command is used to receive the coordinates from the plotter. The

coordinates are displayed on the analyzer screen with program lines 110 to 140. Finally, softkey 1 displays the plotter coordinates when pressed. Disconnect the computer before executing softkey 1.

10 OUTPUT 718;"FUNCDEF P_OP,!";	<i>Defines a function called P_OP.</i>
20 OUTPUT 718;"VARDEF P_ONEX,1,VARDEF P_ONEY,1;";	<i>Declares the variables used to hold the plotter coordinates.</i>
30 OUTPUT 718;"VARDEF P_TWOX,1;VARDEF P_TWOY,1;";	<i>Declares the variables used to hold the plotter coordinates.</i>
40 OUTPUT 718;"OUTPUT 5,B,79;";	<i>Sends ASCII code for "O".</i>
50 OUTPUT 718;"OUTPUT 5,B,80;";	<i>Sends ASCII code for "P".</i>
60 OUTPUT 718;"OUTPUT 5,B,59;";	<i>Sends ASCII code for ",".</i>
70 OUTPUT 718;"ENTER 5,K,P_ONEX;";	<i>Gets plotter coordinates from plotter.</i>
80 OUTPUT 718;"ENTER 5,K,P_ONEY;";	<i>Gets Y coordinate from plotter.</i>
90 OUTPUT 718;"ENTER 5,K,P_TWOX;";	<i>Gets X coordinate from plotter.</i>
100 OUTPUT 718;"ENTER 5,K,P_TWOY;";	<i>Gets Y coordinate from plotter.</i>
110 OUTPUT 718;"PU;PA 200,190;DSPLY P_ONEX,10.2;";	<i>Displays coordinate on analyzer screen.</i>
120 OUTPUT 718;"PU;PA 200,180;DSPLY P_ONEY,10.2;";	<i>Displays coordinate on analyzer screen.</i>
130 OUTPUT 718;"PU;PA 200,170;DSPLY P_TWOX,10.2;";	<i>Displays coordinate on analyzer screen.</i>
140 OUTPUT 718;"PU;PA 200,160;DSPLY P_TWOY,10.2;";	<i>Displays coordinate on analyzer screen.</i>
150 OUTPUT 718;"RELHPIB;";	<i>Releases analyzer control of the HP-IB.</i>
160 OUTPUT 718;"!;";	<i>Marks the end of the function, P_OP.</i>
170 OUTPUT 718;"KEYDEF 1,!P_OP;!;!DSP OP;!;";	<i>Puts P_OP function in a soft-key under the user menu.</i>
180 END	

## Description

Use the ENTER command to request data from an HP-IB device. The ENTER command causes the analyzer to assume controller capabilities on HP-IB. The RELHPIB command may be used to disable these capabilities. The returned data is formatted according to the format specified in the format field:

K: Free field. ASCII real number format.

B: One byte binary.

W: One word (2 bytes) binary.

---

## EP

### Enter Parameter Function

The EP command sends values entered by the operator on the analyzer number keyboard to the present active function value. It must be invoked each time a complete new value is entered.

#### Syntax



Related Commands: ACTDEF, AT, DL, EE, MKA, MKD, MKF, MKN, NRL, RB, RLPOS, SRCAT, SRCPOFS, SRCPSTP, SRCPSWP, SRCPWR, SRCTK, SS, ST, TH, VAVG, VB, VBR.

#### Example

OUTPUT 718;"ST;EP;" *The sweep time can be entered by using the front-panel keys.*

#### Description

The EP command allows the user to enter a value manually through the numeric keypad. When the value is terminated by a unit or **ENTER** key on the analyzer, the analyzer will be ready to execute more remote commands.

---

#### Note

Because the EP command uses the current active function, confirm that the correct active function is being used before executing the statement.



## ERASE

### Erase

The ERASE command clears trace A and trace B, disposes of the contents of the user memory, and resets the state registers and the analyzer to the instrument preset state.

### Syntax



Related Commands: DISPOSE, ONCYCLE, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, ONTIME, PSTATE, TRDEF, TRMATH, VARDEF.

### Example

```
OUTPUT 718;"ERASE;"
```

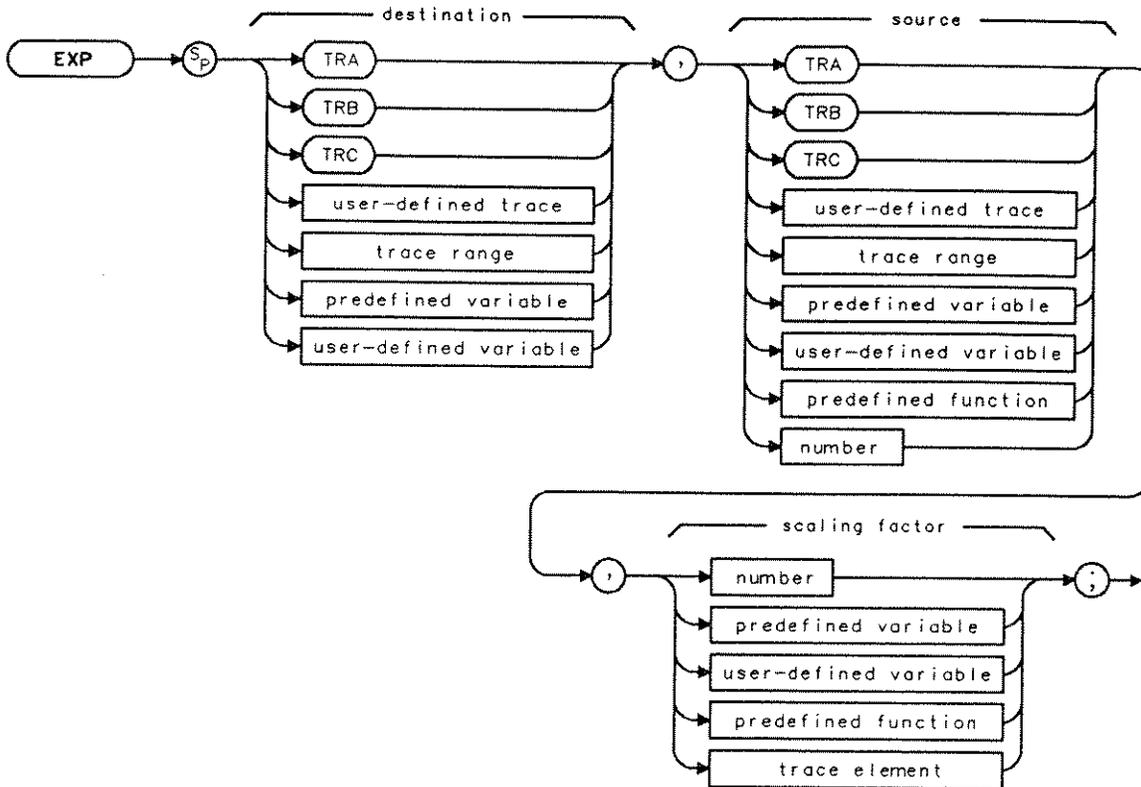
### Description

The ERASE command erases *all* of the user memory by performing a DISPOSE ALL. All user-defined functions, traces, variables, softkeys, and on-event algorithms are erased. The on-event algorithms are ONCYCLE, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, and ONTIME.

## EXP Exponent

The EXP command places the exponential of the source in the destination. The EXP command is useful for converting log values to linear values.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name. Any valid variable name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	Real number range.
Number	Any real or integer number.	

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.  
Related Commands: FUNCDEF, LOG.

### Example 1

This example converts the marker amplitude to power units.

10 OUTPUT 718;"IP;"	<i>Initializes the analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates the single-sweep mode.</i>
30 OUTPUT 718;"VARDEF P_MW,0;"	<i>Initializes variable P_MW to 0.</i>
40 OUTPUT 718;"TS;MKPK HI;"	<i>Finds highest peak of sweep.</i>
50 OUTPUT 718;"EXP P_MW,MKA,10;"	<i>Divides the marker amplitude by 10, raises it to the power of 10, and stores the results in P_MW.</i>
60 OUTPUT 718;"P_MW?;"	<i>Returns the value of P_MW to computer.</i>
70 ENTER 718;Number	<i>Assigns value to computer variable.</i>
80 DISP Number;"mW"	<i>Displays result on the computer screen.</i>
90 END	

### Example 2

This example finds the natural exponential of a number and uses the LOG function to return the original source value of the EXP function.

10 OUTPUT 718;"VARDEF E_XP,0;"	<i>Defines a variable called E_XP.</i>
20 OUTPUT 718;"EXP E_XP,2,2.30259;"	<i>Finds the natural exponential of 2.</i>
30 OUTPUT 718;"E_XP?;"	<i>Returns the natural exponent of 2.</i>
40 ENTER 718;Value	
50 PRINT Value	<i>Prints the value of the exponential.</i>
60 OUTPUT 718;"LOG E_XP,E_XP,2.30259;"	<i>Uses the log function on the exponential value.</i>
70 OUTPUT 718;"E_XP?;"	<i>The log of the exponential value is approximately 2.</i>
80 ENTER 718;Logvalue	
90 PRINT Logvalue	<i>Prints value.</i>
100 OUTPUT 718;"VARDEF E_XPY,0;"	<i>Declares a variable called E_XPY.</i>
110 OUTPUT 718;"EXP E_XPY,-5,2.30259;"	<i>Finds the natural exponential of -5.</i>
120 OUTPUT 718;"E_XPY?;"	<i>Returns the value of the natural exponential of -5.</i>
130 ENTER 718;Value2	
140 PRINT Value2	<i>Prints the value of the exponential.</i>
150 OUTPUT 718;"LOG E_XPY,E_XPY,2.30259;"	<i>Uses the log function on the exponential value.</i>
160 OUTPUT 718;"E_XPY?;"	<i>The log of the exponential value is approximately -5.</i>
170 ENTER 718;Logval	
180 PRINT Logval	
190 END	

**EXP Exponent****Description**

The EXP command first divides the source by the scaling factor, then uses the quotient as an exponent of 10:

$10^{\text{source/scaling factor}}$  is placed into the destination.

The EXP command can be used to evaluate a natural exponential function by using 2.30259 as the scaling factor. Since EXP and LOG are inverse functions, the EXP command has a scaling factor that may be used to “undo” the scaling factor of the LOG command. (See example 2.)

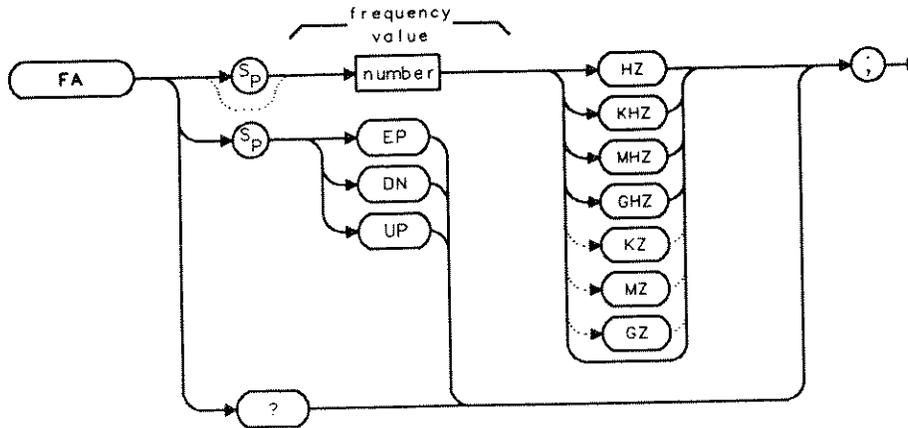
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

See “LOG” for more information on the scaling factor.

## FA Start Frequency

The FA command specifies the start frequency value.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	Frequency range of the analyzer.

Preset State HP 8590B or HP 8591A: 0 Hz.

Preset State HP 8592B or HP 8593A: 2.750 GHz.

Step Increment: (Span)/10.

Related Commands: CF, FB, FOFFSET, FS, HNLOCK, HNUNLK, MKF, SP.

### Example

OUTPUT 718;"FA 88MHZ;FB 108MHZ;" *Sets the start frequency to 88 MHz, the stop frequency to 108 MHz.*

OUTPUT 718;"FA?;" *Returns the start frequency.*

ENTER 718;Freq *Stores the response from the analyzer.*

DISP Freq *Displays the frequency on the computer screen.*

## FA Start Frequency

### Description

The FA command specifies the start frequency value, performing the same function as **START FREQ**. The start frequency is equal to the center frequency minus the span divided by two ( $FA = CF - SP/2$ ).

### Note



Changing the start frequency changes the center frequency and span. For the HP 8592B or HP 8593A only, the start frequency may be limited by the harmonic band if harmonic band locking is used.

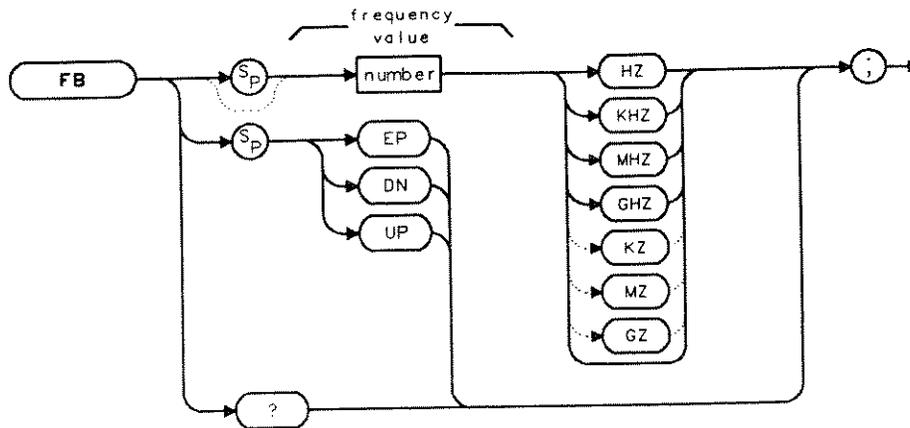
### Query Response



## FB Stop Frequency

The FB command specifies the stop frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	Frequency range of the analyzer.

Preset State HP 8590B or HP 8591A: 1.8 GHz.

Preset State HP 8592B or HP 8593A: 22 GHz.

Step Increment: (Span)/10.

Related Commands: CF, FA, FOFFSET, FS, HNLOCK, MKF, SP.

### Example

OUTPUT 718;"FA 88MHZ;FB 108MHZ;" *Sets the start frequency to 88 MHz, the stop frequency to 108 MHz.*

OUTPUT 718;"FB?;" *Returns the start frequency.*

ENTER 718;Freq *Stores the response from the analyzer.*

DISP Freq *Displays the frequency on the computer screen.*

## FB Stop Frequency

### Description

The FB command specifies the stop frequency value, performing the same function as **STOP\_FREQ**. The stop frequency is equal to the center frequency plus the span divided by two ( $FA = CF + SP/2$ ).

### Note



Changing the stop frequency changes the center frequency and span. For the HP 8592B or HP 8593A only, the stop frequency may be limited by the harmonic band if harmonic band locking is used.

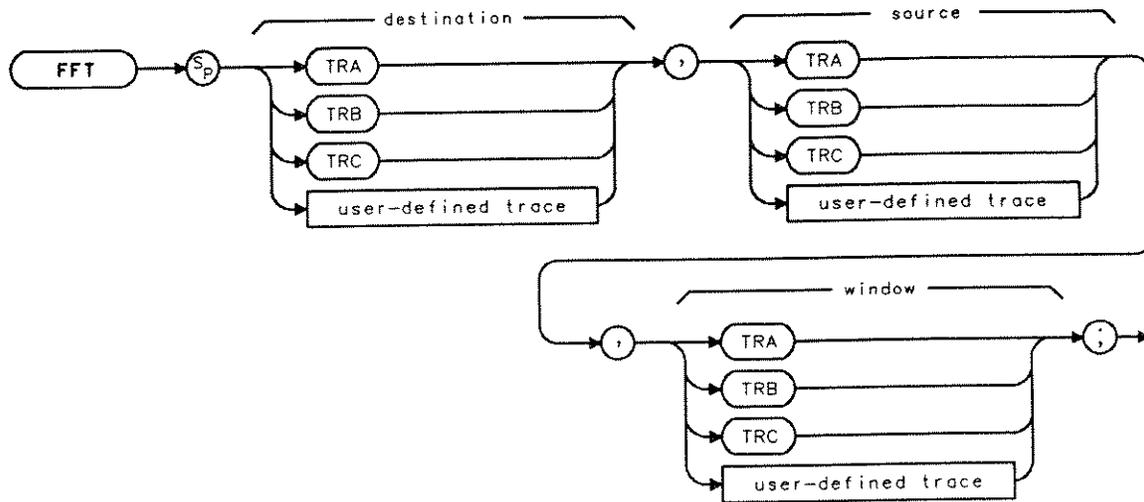
### Query Response



## FFT Fast Fourier Transform

The FFT command performs a discrete Fourier transform on the source trace array and stores the logs of the magnitudes of the results in the destination array.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.

Prerequisite Commands: TWINDOW, or a user-defined trace array used as a window.  
Related Commands: TWINDOW.

### Example

OUTPUT 718;"IP;SNGLS;"	<i>Initializes analyzer.</i>
OUTPUT 718;"CF 300MHZ;TS;MKPK HI;MKTRACK ON;"	<i>Tunes center frequency to carrier and decreases span.</i>
OUTPUT 718;"CONTS;SP 200KHZ;RB 100KHZ;"	<i>Changes resolution bandwidth to capture modulation.</i>
OUTPUT 718;"MKTRACK OFF;SP OHZ;"	<i>Reduces span to zero Hz to demodulate carrier.</i>
OUTPUT 718;"MKPK HI;MKRL;LN;SNGLS;"	<i>Changes to linear amplitude scale.</i>
OUTPUT 718;"ST 800MS;TS;"	<i>Sets sweep time to correspond to modulation frequency, execute FFT function.</i>
OUTPUT 718;"TWINDOW TRB,FLATTOP;"	<i>The TWINDOW must be defined before using the FFT function.</i>

**FFT Fast Fourier Transform**

```
OUTPUT 718;"FFT TRA,TRA,TRB;VIEW TRA;"
```

*Executes the FFT function with the trace window function.*

**Description**

The maximum length of any of the traces is 401 points for a predefined trace and 2047 points for a user-defined trace.

The source array is weighted with the function in the window trace (described below) to minimize amplitude inaccuracies, side lobes and so on. Finally, the transform is computed and the results are placed in the destination array.

The analyzer should be in linear mode when the FFT measurement is made. The FFT results are displayed on the analyzer in logarithmic scale. For the horizontal dimension, the frequency at the left side of the graph is 0 Hz, and at the right side is Fmax. Fmax can be calculated using a few simple equations and the sweep time of the analyzer. The sweep time divided by the number of trace array elements containing amplitude information is equal to the sampling period. The reciprocal of the sampling period is the sampling rate. The sampling rate divided by two yields Fmax.

For example, let the sweep time of the analyzer be 20 ms and the number of trace elements be 400. Twenty ms divided by 400 equals 50  $\mu$ s, the sampling period. The sample rate is 1/50  $\mu$ s. Fmax equals 1/50  $\mu$ s divided by 2, or 10 kHz.

FFT is designed to be used in transforming zero span information into the frequency domain. Performing FFT on a frequency sweep will not provide time-domain results.

The windowing function stored in the window trace may be selected with the trace window (TWNDOW) command or by the user storing his own values in that trace. The trace window function modifies the contents of a trace array according to one of three built-in algorithms: UNIFORM, HANNING, or FLATTOP. See Figure 4-2, Figure 4-3, and Figure 4-4. The TWNDOW command multiplies a trace array with one of these windows. The three algorithms simulate passband shapes that represent a give-and-take between amplitude uncertainty, sensitivity, and frequency resolution. The values in the window trace range from -32,768 to 32,767 and are treated as fractional numbers. No offset is used. When FFT is called, the average window value is computed and used to correct the results in absolute units.

The Fourier transforms of the window functions (created with TWNDOW) are shown under the heading "Filter Windows." Use the graphs to estimate resolution and amplitude uncertainty of a Fourier transform display. Each horizontal division of the graphs equals 1/sweep-time or Fmax/200, and represents two trace array elements.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

**Note**

TWNDOW should always be set before FFT is executed.



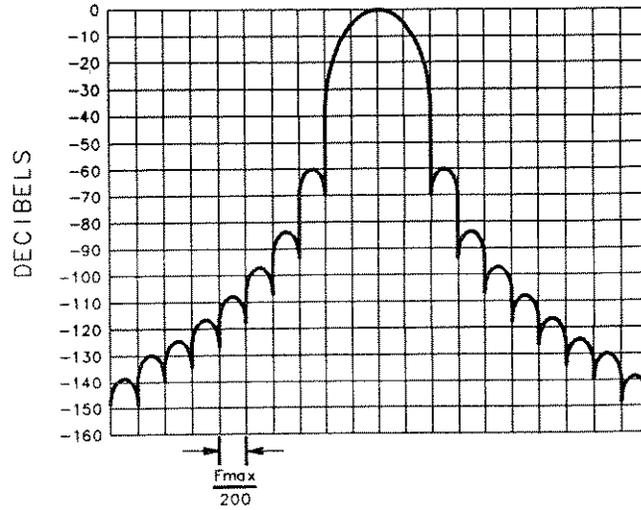


Figure 4-2. Hanning Filter Window

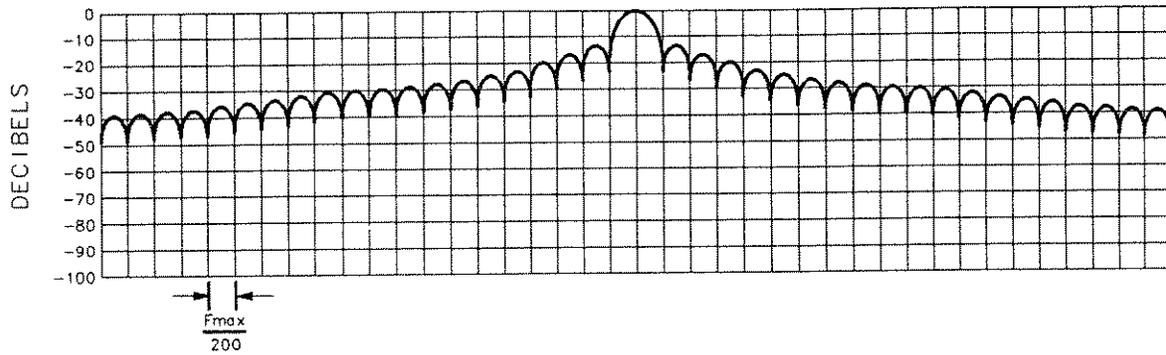


Figure 4-3. Uniform Filter Window

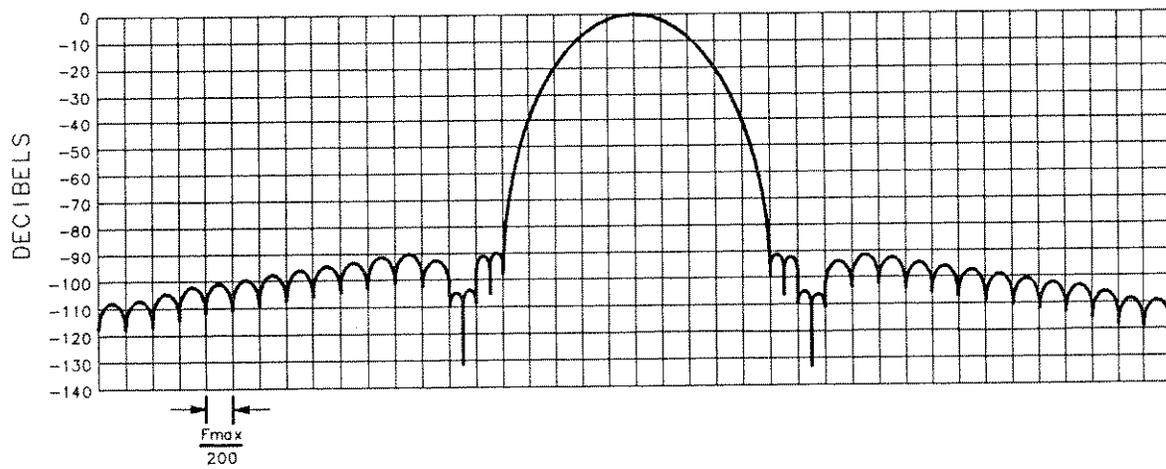


Figure 4-4. Flat Top Filter Window

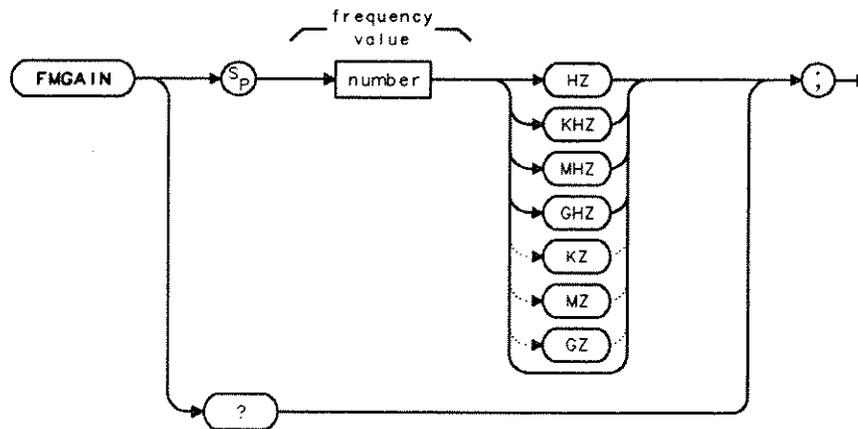
## FMGAIN

### FM Gain

The FMGAIN command provides direct access to the FM gain. FM gain sets the total FM frequency deviation for full screen demodulation.

**Option and Model Required:** Option 102 with HP 8591A or HP 8593A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	10 kHz to 500 kHz.

Preset Value: 100 kHz.

Related Commands: DEMOD, SPEAKER, SQLCH

### Example

OUTPUT 718;"FMGAIN 10KHZ;" *Sets the FM gain.*

### Description

The value of FMGAIN divided by eight yields the FM gain per division. The functions of FMGAIN and **FM GAIN** are identical.

### Query Response

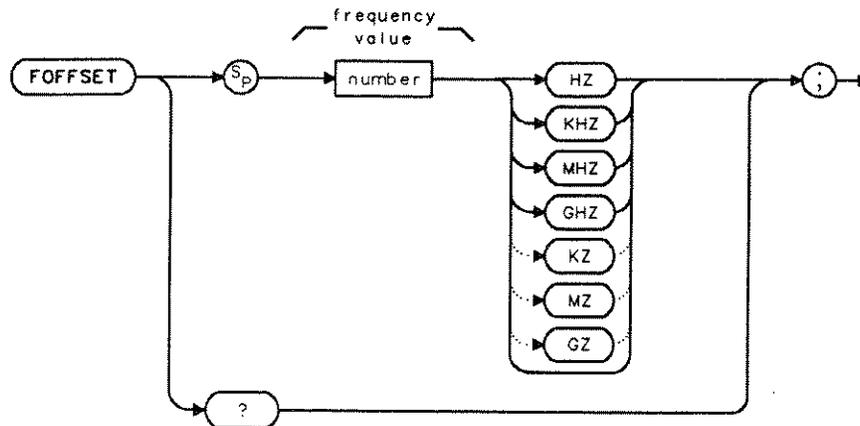


## FOFFSET

### Frequency Offset

The FOFFSET command specifies the frequency offset for all absolute frequency readouts such as center frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	

Preset State: 0 Hz.

Related Commands: CF, FA, FB, MKN, MKF, MKSP, MKSS.

### Example

```

10 OUTPUT 718;"IP;FA 200MZ;"      Initializes analyzer. Sets start frequency.
20 OUTPUT 718;"FB 1GZ;"          Sets stop frequency.
30 OUTPUT 718;"TS;MKPK HI;"      Places marker on signal peak.
40 OUTPUT 718;"MF;"              Finds frequency of marker.
50 ENTER 718;A
60 PRINT A                        Prints frequency of marker.
70 OUTPUT 718;"FOFFSET 500MZ;"    Adds a frequency offset.
80 OUTPUT 718;"TS;MF;"           The frequency of the marker now is the frequency of
                                   the signal peak plus the frequency offset.

90 ENTER 718;A
100 PRINT A                       The displayed frequency is 500 MHz greater than the
                                   frequency displayed in line 60.

110 END

```

## Description

The FOFFSET command selects a value that offsets the frequency scale for all absolute frequency readouts (for example, center frequency). Relative values such as span and marker delta are not offset.

After execution, the FOFFSET command displays the frequency offset in the active function readout. When an offset is in effect, it is displayed beneath the bottom graticule line on the analyzer screen.

Execute "FOFFSET 0;" or "IP;" to turn off the offset.

The functions of the FOFFSET command and **FREQ OFFSET** are identical.

## Query Response



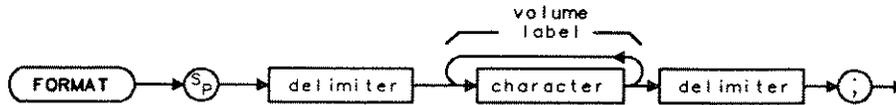
## FORMAT

### Format Card

The FORMAT command formats a memory card in logical interchange format (LIF).

Option Required: Option 003 with HP 8590B or HP 8592B

### Syntax



Item	Description/Default	Range
Character	Any valid character.	0 to 6 characters long, A through Z and the underscore (the underscore cannot be the first character of the label).
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	! "\$ % & ' / : = @

### Example

OUTPUT 718;"FORMAT %MYCARD%;" *Formats a memory card with the label MYCARD.*

### Description

If a label is not specified, the label defaults to HP859X. The functions of FORMAT and **FORMAT CARD** are similar.

## FS Full Span

The FS command selects the full-span mode of the analyzer.

### Syntax



Related Commands: CF, FA, FB, HNLOCK, SP, SS.

### Example

OUTPUT 718;"FS;" *Puts the analyzer in full-span mode.*

### Description

The FS command selects a start frequency of 0 Hz and a stop frequency of 1.8 GHz (for the HP 8590B or HP 8591A), or a start frequency of 2.750 GHz and a stop frequency of 22 GHz (for the HP 8592B and HP 8593A in harmonic unlock). For the HP 8592B or HP 8593A in harmonic lock, "FS;" sets the center frequency and span according to the harmonic band:

**Center Frequency and Span Settings**

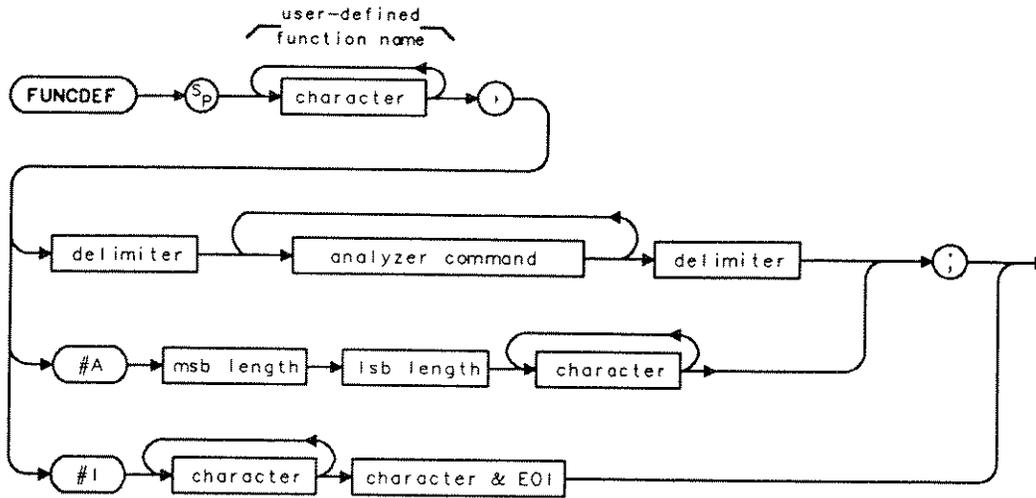
Harmonic Band	Center Frequency	Span
0	1.45 GHz	2.9 GHz
1	4.575 GHz	3.6 GHz
2	9.4 GHz	6.8 GHz
3	15.9 GHz	7.0 GHz
4	20.55 GHz	2.9 GHz

The functions of FS and **FULL SPAN** are identical.

## FUNCDEF Define Function

The FUNCDEF command defines a routine consisting of analyzer commands, assigns it a label, and stores the routine and its label in the user memory.

### Syntax



Item	Description/Default	Range
Character (function name)	Any valid character.	2 to 11 characters long, A through Z and the underscore (the underscore should be used as the second character of the label).
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Analyzer command	Any valid analyzer command.	
Msb length	Most significant byte of a two-byte word that describes the number of bytes transmitted.	
Lsb length	Least significant byte of a two-byte word that describes the number of bytes transmitted.	
Character (data)	Any valid character.	
Character & EOI	Any valid character and END.	

Restriction: User-defined function name cannot be a reserved word (see Table 4-2).

Related Commands: ABORT, DISPOSE, KEYDEF, RETURN.

## Example

Connect CAL OUT to the analyzer input.

OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
OUTPUT 718;"VARDEF H_SPAN,0;"	<i>Defines user-defined variable with an initial value of 0.</i>
OUTPUT 718;"FUNCDEF S_HIFT,^";	<i>Creates user-defined function, called S_HIFT. Shift divides the span by two and adds the results to the center frequency. The "^" delimits the definition.</i>
OUTPUT 718;"DIV H_SPAN,SP,2;"	<i>Puts half of span value into H_SPAN.</i>
OUTPUT 718;"ADD CF,CF,H_SPAN;"	<i>Adds H_SPAN to CF.</i>
OUTPUT 718;"^";	<i>Marks the end of the FUNCDEF declaration.</i>
OUTPUT 718;"KEYDEF 1,S_HIFT,%SHIFT_UP%";	<i>Assigns the function S_HIFT to the user-defined key, called SHIFT_UP.</i>
OUTPUT 718;"CF 300MHZ;"	<i>Displays the calibrator signal.</i>
OUTPUT 718;"SP 1MHZ;"	
LOCAL 718	<i>Returns control to local mode.</i>

## Description

After FUNCDEF is executed, the command list is executed whenever the function label is encountered.

The function label and the list of commands it executes form a user-defined function. Once a user-defined function is stored, it can be executed by invoking the function name in user-defined softkeys, other user-defined functions, or computer programs.

Delete user-defined functions from the analyzer memory with the DISPOSE command. Use ABORT to stop a user-defined function operation. Use IF/THEN/ELSE/ENDIF, REPEAT/UNTIL, or RETURN to alter the user-defined function operation.

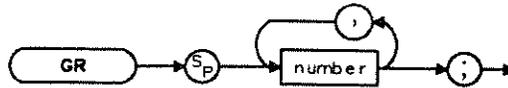
The following are general rules and limitations of FUNCDEF:

- Do not use existing function names or secondary key words (reserved words) as labels for user-defined functions.
- Do not execute a user-defined function within its FUNCDEF statement (recursion).
- Do not nest definitions of FUNCDEF commands.

## GR Graph

The GR command graphs the given  $y$  coordinate by incrementing the  $x$  coordinate by 1.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	-22 to 233.

### Example

This example graphs a diagonal line on the analyzer display.

```

OUTPUT 718;"BLANK TRA;"
OUTPUT 718;"PU;PA 0,0;GR;"  Positions the pen.
FOR I = 0 TO 400             I represents the Y value in graticule coordinates.
OUTPUT 718;I DIV 2          Graphs the Y values, incrementing the X value by 1.
NEXT I
  
```

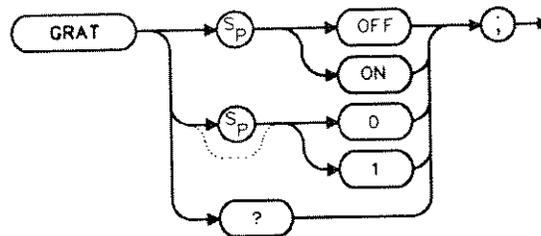
### Description

The GR command plots a graph at the amplitude point indicated by the next  $y$  coordinate. The  $x$  coordinate is always incremented by 1; it cannot be decremented.

## GRAT Graticule

The GRAT command turns the graticule on and off.

### Syntax



Preset State: GRAT ON.  
 Related Commands: ANNOT.

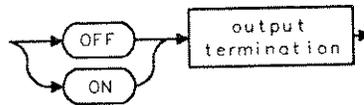
### Example

OUTPUT 718;"GRAT OFF;"	<i>Turns the graticule off.</i>
OUTPUT 718;"GRAT?;"	<i>Queries graticule status.</i>
ENTER 718;Grat\$	<i>Gets response from the analyzer.</i>
DISP Grat\$	<i>Displays OFF on the computer screen.</i>

### Description

The functions of the GRAT command and **GRAT ON OFF** are identical.

### Query Response

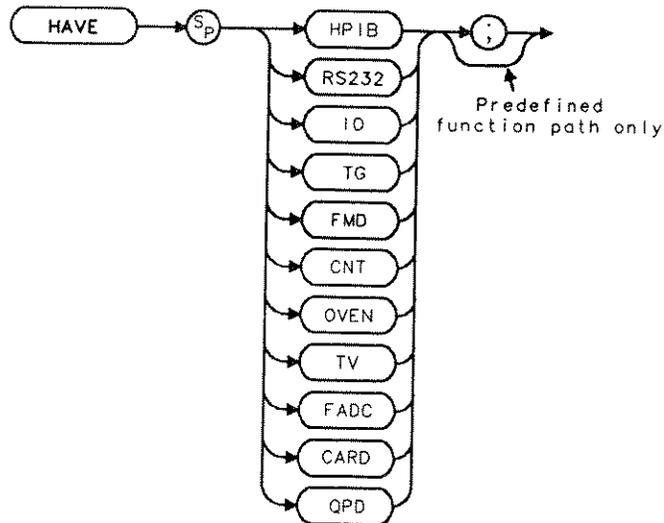


## HAVE

### Have

The HAVE command returns a "0" if the specified option or device is *not* installed.

### Syntax



### Example

```
OUTPUT 718;"HAVE HPIB;"  Queries if Option 021 is installed.
ENTER 718;A              Receives response from analyzer.
DISP A                   Displays response.
```

### Description

The parameters are:

HPIB	HP-IB interface, Option 021.
RS232	RS-232 interface, Option 023.
IO	Either the HP-IB interface (Option 021) or RS-232 interface (Option 023).
TG	Tracking generator, Option 010 or 011. (HP 8590B or HP 8591A only.)
FMD	FM demodulator, Option 102. (HP 8591A or HP 8593A only.)
CNT	Counter-lock. (HP 8591A or HP 8593A only.)
OVEN	Precision frequency reference, Option 004. (HP 8591A or HP 8593A only.)
TV	TV synch trigger, Option 102. (HP 8591A or HP 8593A only.)
FADC	Fast ADC, Option 101. (HP 8591A or HP 8593A only.)
CARD	Memory card. (Option 003 for the HP 8590B or HP 8592B, standard for the HP 8591A or HP 8593A.)

QPD Quasi-peak detector, Option 103. (HP 8591A or HP 8593A only.)

“HAVE CARD;” returns additional information about the memory card. By checking the bit status of the byte returned from “HAVE CARD;”, you can determine the following:

Bit Position	Bit Status = 0	Bit Status = 1
0	Memory card reader is not installed.	Memory card reader is installed.
1	Memory card is write protected.	Memory card is not write protected.
2	Memory card is a random access card (RAM).	Memory card is a read only memory card (ROM).
3	Memory card not inserted into memory card reader.	Memory card is inserted into memory card reader.

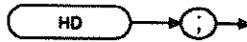
Use the BIT command to determine the status of the bits. For example,

OUTPUT 718;"VARDEF R_RESULT,0;"	<i>Stores the bit status in R_RESULT.</i>
OUTPUT 718;"BIT R_RESULT,HAVE CARD,3;"	<i>Checks the status of bit 3.</i>
OUTPUT 718;"R_RESULT?;"	<i>Gets the result.</i>
ENTER 718;A	
DISP A	<i>Displays the bit status of bit 3.</i>

## HD Hold Data Entry

The HD command disables data entry via the analyzer numeric keypad, knobs, or step keys. The active function readout is blanked, and any active function is deactivated.

### Syntax



Related Commands: ACTDEF, AT, CF, DL, EE, EK, EP, FA, FB, FOFFSET, INZ, LG, MKA, MKACTION, MKD, MKF, MKFC, MKNOISE, MKN, MKP, MKPAUSE, MKPX, MKTRACK, ML, NRL, RB, RL, RLPOS, ROFFSET, SRCAT, SRCPOFS, SRCPSTP, SRCPSWP, SRCPWR, SRCTK, SS, ST, TH, VAVG, VB, VBR.

### Example

OUTPUT 718;"HD;"

*Disables the active function.*

OUTPUT 718;"CF 500 MHZ;600 MHZ;HD;700 MHZ;"

*This will leave the center frequency at 600 MHz, since HD deactivates any current function.*

### Description

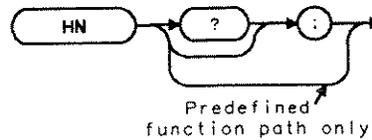
The functions of the HD command and **HOLD** are identical.

## HN Harmonic Number

The HN command returns the harmonic number of the current analyzer is tuning.

**Model Required:** HP 8592B or HP 8593A

### Syntax



Related Commands: FS, HNLOCK, HNUNLK.

### Example

OUTPUT 718;"HN?;" *Queries harmonic band of analyzer.*  
ENTER 718;Number *Gets response from the analyzer.*  
DISP Number *Displays the result on computer screen.*

### Query Response

The HN command returns the number of the harmonic band (0 to 4), if the analyzer is sweeping single-band. A "-1" is returned if the analyzer is sweeping multiband.

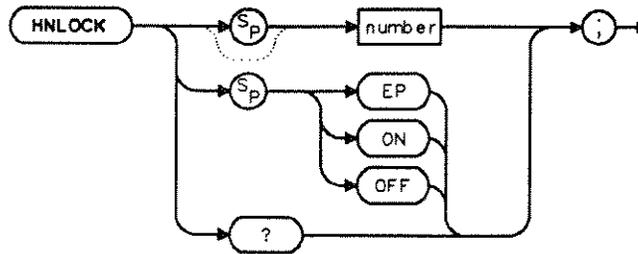


## HNLOCK Harmonic Number Lock

The HNLOCK command forces the analyzer to use only the selected harmonic band.

Model Required: HP 8592B or HP 8593A

### Syntax



Item	Description/Default	Range
Number	Any valid integer number.	0 to 4.

Preset State: HNLOCK OFF.

Related Commands: CF, FA, FB, FOFFSET, HN, HNUNLK, SNGLS, SP.

### Example

```
INPUT "SELECT THE DESIRED FREQUENCY BAND",Harm  Gets harmonic band from user.
OUTPUT 718;"HNLOCK ";Harm;";"                Locks harmonic band.
```

### Description

HNLOCK ON locks onto the harmonic necessary for the current center frequency, lowering the span, if necessary, due to the limits of the harmonic band.

HNLOCK x, where x is an integer, locks onto harmonic band x and automatically selects the settings shown in the following table.

Remote Commands	Equivalent Softkey	Frequency Settings	
HNLOCK 0	BAND 0	Center frequency 1.450 GHz	Span 2.900GHz
HNLOCK 1	BAND 1	Center frequency 4.575 GHz	Span 3.600GHz
HNLOCK 2	BAND 2	Center frequency 9.400 GHz	Span 6.800GHz
HNLOCK 3	BAND 3	Center frequency 15.90 GHz	Span 7.000GHz
HNLOCK 4	BAND 4	Center frequency 20.55 GHz	Span 2.900GHz

Once HNLOCK is set, only frequencies and spans that fall within the frequency band of the current harmonic may be entered. The span is automatically reduced to accommodate a center frequency specified near the end of the band range.

**Note**



Before changing the frequency range to another harmonic, unlock the band with the harmonic unlock command, "HNLOCK OFF;" or "HNUNLK;".

The HNLOCK command and the softkeys under the **BAND LOCK** menu are identical.

**BAND LOCK ON (HNLOCK ON)**

**Start Frequency**

If a start frequency is entered that is outside of the current band boundaries, it will be set to the nearest band edge instead. If a start frequency greater than the current stop frequency is entered, the (possibly modified) start frequency is used for both the start and stop frequency; hence the span will be set to zero. If the start frequency and stop frequency specify too large a span, they will be modified. (Also see "FA.")

**Stop Frequency**

If a stop frequency is entered that is outside of the current band boundaries, it will be set to the nearest band edge instead. If a stop frequency less than the current start frequency is entered, the (possibly modified) stop frequency will be used for both the start and stop frequency; hence the span will be set to zero. (Also see "FB.")

**Center Frequency**

The span will be modified if necessary to get the center frequency specified without crossing the band edges. (Also see "CF.")

**HNLOCK Harmonic Number Lock****Span**

The span will be limited as necessary to keep the start and stop frequencies within the band edges without changing the center frequency. The maximum span allowed is 2.943 GHz in band 0, 3.600 GHz in band 1, 6.957 GHz in band 2, and 7 GHz in higher bands. (Also see "SP.")

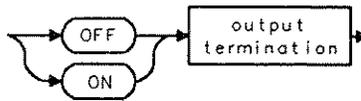
**BAND LOCK OFF (HNLOCK OFF)**

The start and stop frequencies are bounded by the range of the instrument.

The frequency not specified will be bounded by the following scheme:

**Continuous Sweep Mode:** If the specified frequency is in band 0, the values will be contained by the bounds of band 0. If the value is in the harmonic band range, the values will be bounded by the top of the instrument range and the lower end of band 1. (Also see "CONTS.")

**Single Sweep Mode:** The values are bounded by the instrument range only; therefore, band 0 can be included in a multiband sweep in single mode. (Also see "SNGLS.")

**Query Response**

---

## HNUNLK Unlock Harmonic Number

The HNUNLK command unlocks the harmonic band.

Model Required: HP 8592B or HP 8593A

### Syntax



Related Commands: CF, FA, FB, FOFFSET, FS, HN, HNLOCK, SP.

### Example

```
OUTPUT 718;"HNUNLK;"
```

### Description

The HNUNLK command allows you to select frequencies and spans outside the range of a single harmonic band.

HNUNLK is equivalent to HNLOCK OFF and **BND LOCK ON OFF** (OFF).

## IB Input B

The IB command provides a method for putting values into trace B. The analyzer expects 401 two-byte integers. The data values can represent the range of integer numbers.

### Syntax



Item	Description/Default	Range
Data byte	8-bit byte containing numeric or character data.	

### Example

```

10 ASSIGN @Sa TO 718;FORMAT ON
20 ASSIGN @Sa_bin TO 718;FORMAT OFF
30 INTEGER Binary(1:401)
40 OUTPUT @Sa;"CF 300MZ;CLRW TRB;BLANK TRA;SP
10MZ;SNGLS;TS;"
50 OUTPUT @Sa;"TDF B;MDS W;TB;"
60 ENTER @Sa_bin;Binary(*)
70 OUTPUT @Sa;"CF 100MZ;RB 30KZ;SP 1MZ;TS;"
80 PAUSE
90 OUTPUT @Sa;"IB";

100 OUTPUT @Sa_bin;Binary(*)
110 END

```

*Dimensions array Binary.*

*Takes a measurement sweep.*

*Outputs trace B (in binary) to computer.*

*Stores trace data in array.*

*Changes the analyzer settings.*

*Prepares analyzer to receive trace B data stored in array.*

*Sends trace B data to analyzer.*

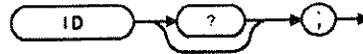
### Description

The IB command sends trace B data as binary data only; it is independent of the measurement data size (MDS) command. The IB command expects 802 data bytes (401 data points, two bytes each).

## ID Identify

The ID command returns the instrument identity to the controller (for example, "HP8593A").

### Syntax



Related Commands: REV, SER.

### Example

10 ALLOCATE A\$[50]	<i>Allocates string to hold model number.</i>
20 OUTPUT 718;"ID;"	<i>Gets model number.</i>
30 ENTER 718;A\$	<i>Transfers number to computer.</i>
40 DISP A\$	<i>Displays model number.</i>
50 END	

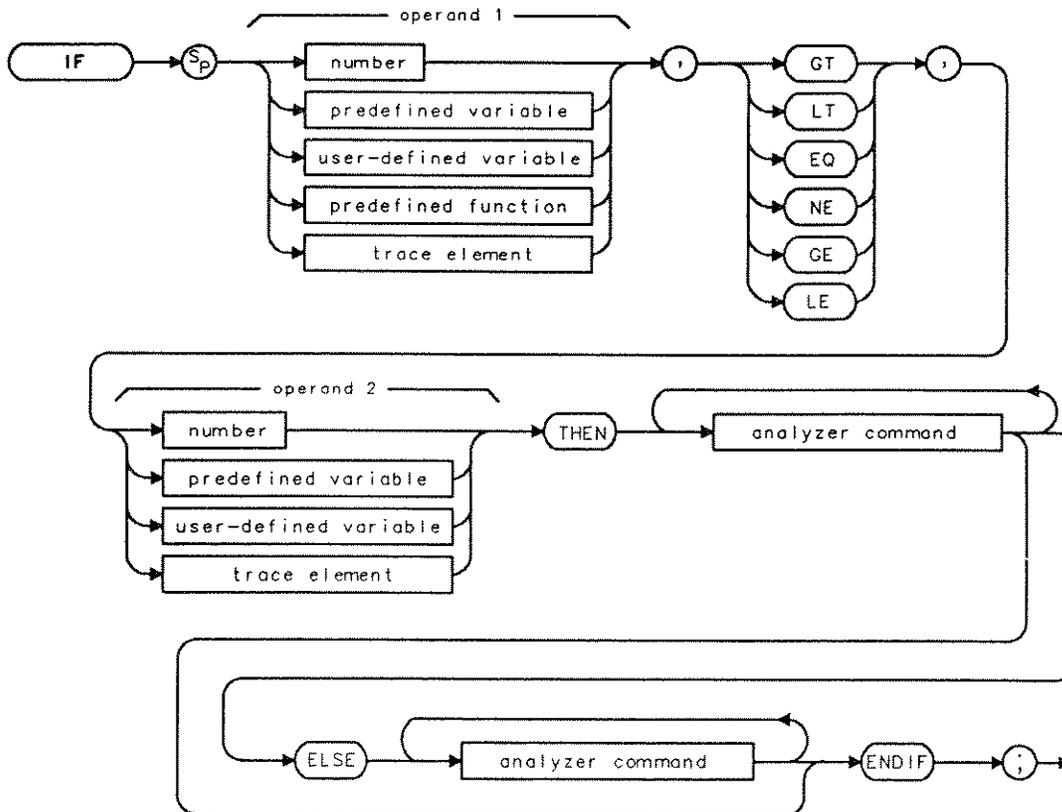
### Query Response



## IF THEN ELSE ENDIF If Then Else Endif

The IF/THEN/ELSE/ENDIF commands form a decision and branching construct.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	Real number range.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Analyzer command	Any spectrum analyzer command.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
Related Commands: REPEAT/UNTIL.

### Example

The following example uses the IF/THEN/ELSE/ENDIF command to pick a center frequency.

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"TH -35DM;"	<i>Sets threshold level.</i>
30 OUTPUT 718;"TS;MKPK HI;MA;"	<i>Finds highest peak.</i>
40 OUTPUT 718;"IF MA,GT,TH"	<i>Compares peak to threshold.</i>
50 OUTPUT 718;"THEN CF 20MHZ"	<i>Changes center frequency to 20 MHz if peak amplitude is greater than -35 dB.</i>
60 OUTPUT 718;"ELSE CF 100MHZ;TS;MKPK HI;"	<i>This line is executed if peak is less than or equal to the threshold level.</i>
70 OUTPUT 718;"ENDIF;"	<i>End of IF/THEN/ELSE/ENDIF.</i>
80 END	

The example below does not include the ELSE portion of the IF/THEN/ELSE/ENDIF command. This example lowers any signal positioned above the analyzer screen.

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;TS;MKPK HI;MA;"	<i>Finds peak of trace.</i>
30 OUTPUT 718;"IF MA,GT,RL THEN"	<i>Compares peak amplitude and reference level.</i>
40 OUTPUT 718;"MKRL;"	<i>Performs line 40 if the marker amplitude is greater than the reference level.</i>
50 OUTPUT 718;"ENDIF;"	<i>Ends IF/THEN/ENDIF structure.</i>
60 END	

### Description

The IF portion compares operands 1 and 2 with the following operators:

Operator	Description
GT	Greater than.
LT	Less than.
EQ	Equal to.
NE	Not equal to.
GE	Greater than or equal to.
LE	Less than or equal to.

If the condition is true, the command list following the IF statement is executed and commands between ELSE and ENDIF are skipped. If the condition is false, the commands after the ELSE statement are executed. If there is no ELSE statement, program execution resumes after the ENDIF statement.

**IF THEN ELSE ENDIF If Then Else Endif**

The "equal to" (EQ) operator is not recommended if value 1 or value 2 represents a real number. When checking for equality with real numbers, the difference between the numbers is useful. For example:

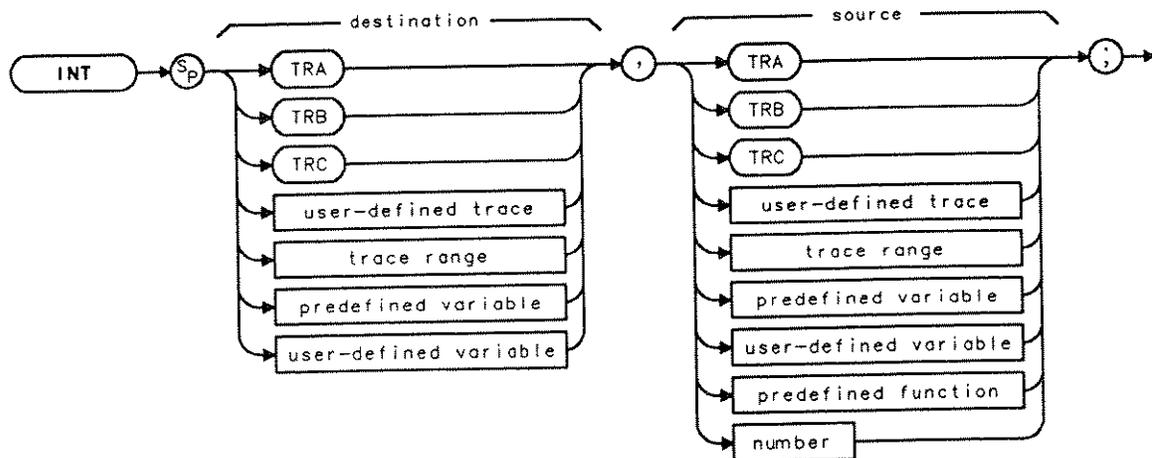
```
OUTPUT 718;"SUB T_EST,V_ARA,V_ARB;" Places the difference of V_ARA and V_ARB into
T_EST. V_ARA, V_ARB, and T_EST are user-
defined variables.

OUTPUT 718;"ABS T_EST,T_EST;" Finds the absolute value of T_EST.
OUTPUT 718;"IF T_EST,LT,.01 THEN" Does a comparison.
OUTPUT 718;"CF UP;"
OUTPUT 718;"ENDIF;"
```

## INT Integer

The INT command places the greatest integer that is less than or equal to the source value into the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

Related Commands: REPEAT/UNTIL.

INT Integer

### Example

OUTPUT 718;"IP;"	<i>Initializes the analyzer.</i>
OUTPUT 718;"ST 2.55SC;TS;"	<i>Sets sweep time to 2.55 seconds. Takes a sweep.</i>
OUTPUT 718;"INT ST,ST;"	<i>Resets the sweep time using the integer value of the previous sweep time.</i>
OUTPUT 718;"TS;"	<i>Sweeps with the 2 second sweep time.</i>

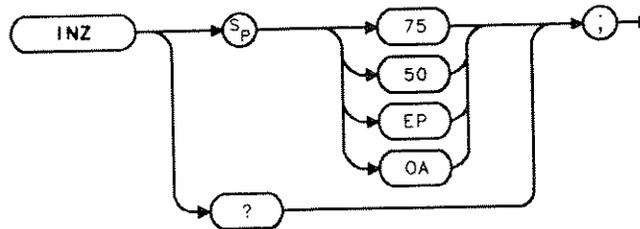
### Description

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## INZ Input Impedance

The INZ command specifies the value of input impedance expected at the active input port.

### Syntax



Preset Value: Returns impedance to the power-on value.  
 Related Commands: AUNITS.

### Example

```
OUTPUT 718;"INZ 75;"    Changes input impedance to 75 ohms.
OUTPUT 718;"AUNITS V;"  Changes amplitude units to volts.
```

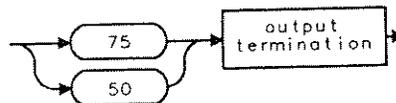
### Description

The actual impedance can be affected by internal hardware only. With the exception of Option 001 or 011 (HP 8590B or HP 8591A only), the analyzer hardware supports 50Ω only. The INZ command is used for computation purposes during power/voltage conversions.

The INZ command only affects the amplitude results reported in absolute relative power (dBm units or watts). (See "AUNITS.")

The functions of the INZ command and **INPUT Z 50 75** are identical.

### Query Response



## IP Instrument Preset

The IP command performs an instrument preset.

### Syntax



### Example

```
OUTPUT 718;"IP;"
```

### Description

The instrument preset command, IP, executes the following commands:

- AMB OFF: Turns off A – B mode.
- AMPCOR OFF: Turns off amplitude correction factors.
- ANNOT ON: Turns on annotation.
- AT: Sets attenuation to 10 dB.
- AUNITS: Loaded from configuration location.
- AUTO: Couples RB, AT, SS, ST, and VB. Turns off display line and threshold.
- BLANK B, C: Blanks trace B and trace C.
- CF (HP 8590B or HP 8591A): Sets center frequency to 900 MHz.
- CF (HP 8592B or HP 8593A): Sets center frequency to 12.38 GHz.
- CLRDSP: Erase user graphics.
- CLRW A: Clears and writes trace A.
- CONTS: Selects continuous sweep.
- DET POS: Selects positive peak detection.
- DL OFF: Display line off.
- FA (HP 8590B or HP 8591A): Sets the start frequency (0 Hz).
- FA (HP 8592B or HP 8593A): Sets the start frequency (2.750 GHz).
- FB (HP 8590B or HP 8591A): Sets the stop frequency. (1.8 GHz).
- FB (HP 8592B or HP 8593A): Sets the stop frequency. (22 GHz).
- FMGAIN: Sets FM gain to 100 kHz. (HP 8591A or HP 8593A with Option 102 only.)
- FOFFSET: Sets the frequency offset to 0 Hz.
- HD: Hold (deactivates active function).
- HNLOCK OFF (HP 8593A only.) Unlocks harmonic band, allowing multiband sweeping.
- GRAT ON: Turns on the graticule.
- INZ: Loaded from configurable location.
- LG: Selects 10 dB/Div log scale.
- LIMITEST OFF: Turns off limit-line testing.
- MDS W: Selects data size of one word, which is two 8-bit bytes.
- MKFCR: Marker counter resolution is set to 0, but a calculated value of 1800 is returned if the marker counter resolution is queried. (HP 8591A or HP 8593A only.)
- MKNOISE OFF: Turns off noise markers.
- MKOFF: Turns off all markers.

**MKPAUSE OFF:** Turns off marker pause mode.  
**MKPX:** Minimum excursion for peak identification is set to 6 dB.  
**MKREAD FRQ:** Sets marker readouts to frequency.  
**MKTRACK OFF:** Turns off marker tracking.  
**MKTYPE PSN:** Sets the marker type as position type.  
**ML:** Sets mixer level to -10 dBm.  
**MODE:** Sets the operating mode to 0.  
**MSI:** Selects the analyzer memory as the mass storage device.  
**PD:** Puts pen down at current position.  
**RL:** Sets reference level to 0 dBm.  
**ROFFSET:** Sets reference offset to 0.  
**RQS 40:** Allows SRQ 110, SRQ 140 for illegal commands or broken hardware.  
**SP (HP 8590B and HP 8591A):** Sets span to 1.8 GHz.  
**SP (HP 8592B and HP 8593A):** Sets span to 19.25 GHz.  
**SQLCH:** Sets the squelch level to 0. (HP 8591A or HP 8593A with Option 102 only.)  
**SS:** Sets the center frequency step size to 100 MHz.  
**STATUS BYTE:** Clear the status byte.  
**TH:** One division above bottom graticule line, line off.  
**TITLE:** Clears the title from the analyzer screen.  
**TM FREE:** Selects free run trigger mode.  
**TDF P:** Selects parameter units output format.  
**TRA:** Sets the trace values to 0.  
**TRC:** Sets the trace values to 8000.  
**TVLINE:** Sets TV line number to 17. (HP 8591A or HP 8593A with Options 101 and 102 only.)  
**VAVG OFF:** Turns off video averaging.  
**VAVG LIMIT:** Sets the video averaging limit to 100.

Instrument preset automatically occurs when you turn on the analyzer. It is a good starting point for many measurement processes. When IP is executed remotely, the analyzer does not necessarily execute a complete sweep.

IP does not clear state or trace registers. IP clears user graphics.

The functions of IP and **PRESET** are similar.

---

**Note**



IP clears all on-event algorithms. The on-event algorithms are ONCYCLE, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, and ONTIME.

---

## KEYCLR

### Key Clear

The KEYCLR command clears softkeys 1 through 6 of menu 1.

### Syntax



Related Commands: DISPOSE, KEYDEF, KEYEXC, KEYLBL, SAVEMENU.

### Example 1

OUTPUT 718;"MENU 1;" *Displays menu 1.*  
 OUTPUT 718;"KEYCLR;" *Erases keys 1-6 of menu 1.*

### Example 2

OUTPUT 718;"MENU 1;" *Displays menu 1.*  
 OUTPUT 718;"SAVEMENU 101;" *Copies the key functions from menu 1 into menu 101.*  
 OUTPUT 718;"KEYCLR;" *Erases the key functions of menu 1.*  
 PAUSE  
 OUTPUT 718;"MENU 101;" *Displays menu 101.*

### Description

The KEYCLR command clears softkeys 1 through 6. Use the DISPOSE command to clear a single softkey.

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position  
 (The key position range is 1 through 6.)

For example:

Menu 1 contains key numbers 1 through 6  
 Menu 101 contains key numbers 601 to 606  
 Menu 200 contains key numbers 1195 to 1200

Menu 1 can be accessed by pressing **MEAS/USER**, **USER MENU(S)**.

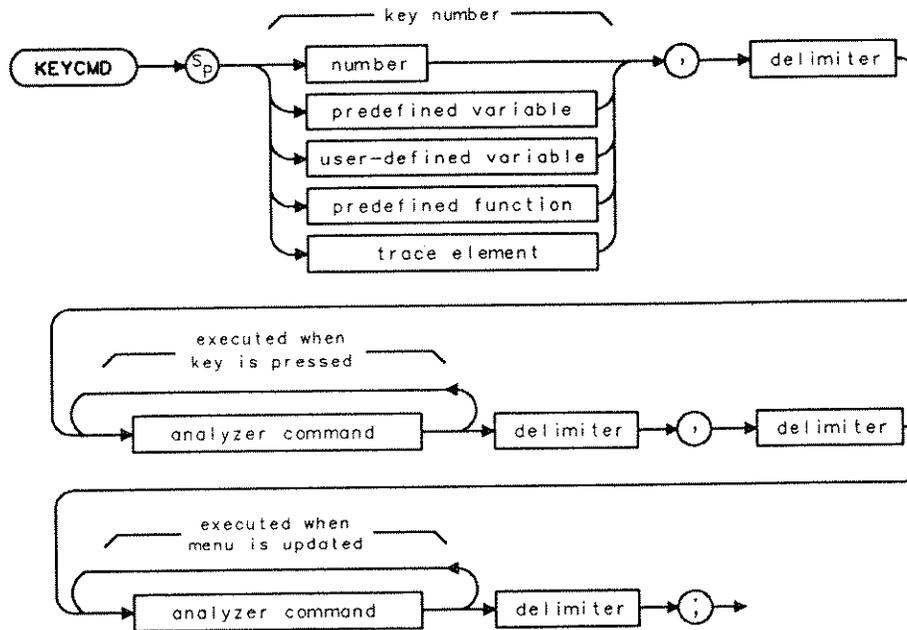
Menus 101 through 200 as well as menu 1 can be accessed using the MENU command. For example, executing "MENU 200;" accesses menu 200.

The SAVEMENU command can be used to save keys 1 through 6 in the desired menu number. (See "SAVEMENU.")

## KEYCMD Key Command

The KEYCMD command allows you define the function and label of a softkey based on a condition. The softkey label is updated whenever a key is pressed.

### Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	1—6, 601—1200
Predefined variable	A command that act as a variable. Refer to Table 4-1.	
Analyzer command	Any valid analyzer command.	
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	! "\$ % & ' / : = @

Related Commands: DISPOSE, ERASE, FUNCDEF, KEYENH, KEYEXC, KEYLBL.

## KEYCMD Key Command

## Example 1

This example defines softkey 1. Softkey 1 has the label "QSAD" until it is pressed. When it is pressed, the following changes occur: center frequency changes to 105.3 MHz, span changes to 300 kHz, and the label for softkey 1 changes from "QSAD" to "KSAD". If softkey 1 is pressed again, the center frequency changes to 100.1 MHz and the label changes back to "QSAD". Note that both lines 50 and 60 comprise the KEYCMD declaration; the limitation on the length of a BASIC program line requires that it be divided into two program lines.

<pre> 10 OUTPUT 718;"IP;" 20 UL\$=CHR\$(16)  30 Off\$=CHR\$(15)  40 OUTPUT 718;"VARDEF C_COUNTRY,1;" 50 OUTPUT 718;"KEYCMD 1,%IF C_COUNTRY,EQ,      1 THEN CF 105.3MHZ;SP 300KHZ;MOV C_COUNTRY,     0;ELSE CF 100.1MHZ;SP 300KHZ;MOV C_COUNTRY,     1;ENDIF;%, "; 60 OUTPUT 718;"@IF C_COUNTRY,EQ,0 THEN     KEYLBL 1,%KSAD% ELSE KEYLBL 1,%QSAD%;     ENDIF;@" 70 END </pre>	<p><i>Initializes analyzer.</i></p> <p><i>UL\$ will be used to turn underline on.</i></p> <p><i>Off\$ will be used to turn the inverse video and underlining off.</i></p> <p><i>Defines variable called C_COUNTRY.</i></p> <p><i>This portion of KEYCMD assigns a function to softkey 1.</i></p> <p><i>This portion of KEYCMD assigns the label to softkey 1.</i></p>
---	---

## Example 2

This example includes an additional programming line, line 70. Line 70 uses KEYCMD to define softkey 2. Softkey 2 has a softkey label, "COUNTRY YES NO". If the value of the variable C\_COUNTRY is 1, then YES is underlined, and label of softkey 1 is "QSAD". Pressing softkey 2 moves the underline to NO, and the softkey label of softkey 1 is changed to "KSAD".

<pre> 10 OUTPUT 718;"IP;" 20 UL\$=CHR\$(16)  30 Off\$=CHR\$(15)  40 OUTPUT 718;"VARDEF C_COUNTRY,1;" 50 OUTPUT 718;"KEYCMD 1,%IF C_COUNTRY,EQ,     1 THEN CF 105.3MHZ;SP 300KHZ;MOV C_COUNTRY,     0;ELSE CF 100.1MHZ;SP 300KHZ;MOV C_COUNTRY,     1;ENDIF;%, "; 60 OUTPUT 718;"@IF C_COUNTRY,EQ,0 THEN     KEYLBL 1,%KSAD% ELSE KEYLBL 1,%QSAD%;ENDIF;@" 70 OUTPUT 718;"KEYCMD 2,%IF C_COUNTRY,EQ,     1 THEN MOV C_COUNTRY,0;ELSE MOV,C_COUNTRY,     1;ENDIF;%,%KEYENH 2,@COUNTRY ";UL\$;"YES";Off\$;"     NO@,,!IF C_COUNTRY,EQ,0!;%, "; 80 END </pre>	<p><i>Initializes analyzer.</i></p> <p><i>UL\$ will be used to turn the underline on.</i></p> <p><i>Off\$ will be used to turn the inverse video and underlining off.</i></p> <p><i>Defines variable called C_COUNTRY.</i></p> <p><i>This portion of KEYCMD assigns a function of softkey 1.</i></p> <p><i>This portion of KEYCMD assigns the label to softkey 1.</i></p> <p><i>Defines the softkey label of softkey 2 according to the value of the variable, C_COUNTRY.</i></p>
---	---

## Description

The KEYCMD command updates the softkey label whenever a key is pressed (as shown in example 2).

The functions of KEYCMD and KEYDEF are similar. The advantage of KEYCMD is that the label of the key can change dependent on a conditional statement without the key itself being pressed. KEYCMD is useful for indicating the state of a function in the analyzer by changing the appearance of the key.

---

### Note



Using other than the recommended key numbers can cause undesirable results. The recommended key numbers are 1–6, 601–1200.

---

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position  
(The key position range is 1 through 6.)

For example:

- Menu 1 contains key numbers 1 through 6
- Menu 101 contains key numbers 601 to 606
- Menu 200 contains key numbers 1195 to 1200

Menu 1 can be accessed by pressing **MEAS/USER**, **USER MENU(S)**.

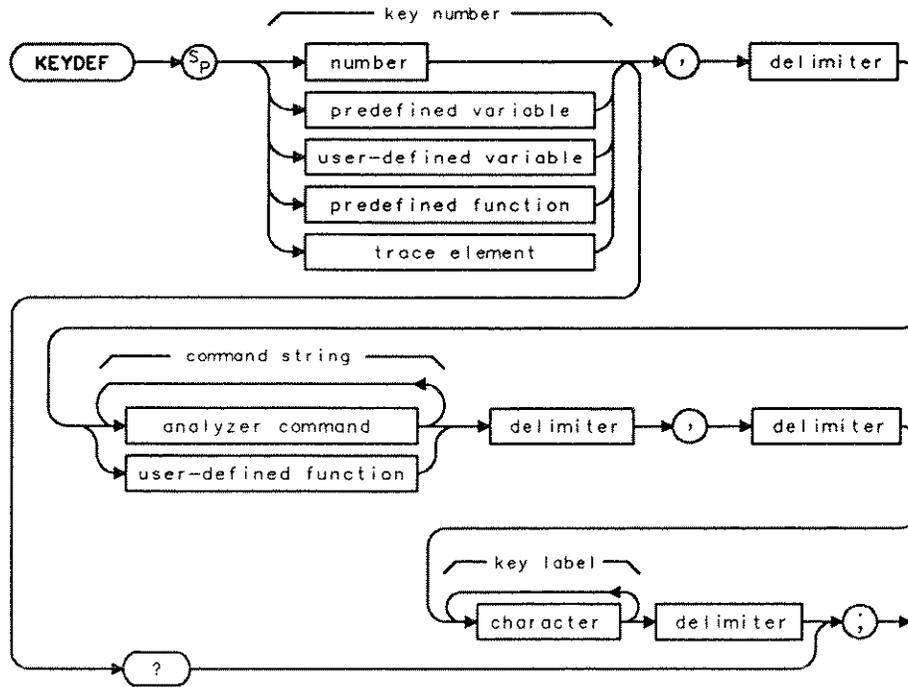
Menus 101 through 200 as well as menu 1 can be accessed using the MENU command. For example, executing "MENU 200;" accesses menu 200.

# KEYDEF

## User-Defined Key Definition

The KEYDEF command assigns a label and user-defined function to a softkey.

### Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined function	Function that returns a value. Refer to Table 4-1.	Any valid predefined function that returns a value within the key number range.
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	1—6, 601—1200.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	Any valid predefined variable that returns a value within key number range.
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!"\$%&'/:=@
Analyzer command	Any valid analyzer command.	
User-defined function	A subroutine defined by the FUNCDEF command.	Any valid function name.
Character	Any valid character. See Table 4-6 and Table 4-7 for additional characters and label functions.	1 to 8 characters per label line, use the ( ) symbol or blank spaces to separate softkey label lines.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: DISPOSE, ERASE, FUNCDEF, KEYEXC, KEYLBL.

### Example 1

Connect CAL OUT to the analyzer input.

<pre>OUTPUT 718;"IP;" OUTPUT 718;"FUNCDEF DLP,^"; OUTPUT 718;"CF 300MHZ;";  OUTPUT 718;"SP 1MHZ^";  OUTPUT 718;"KEYDEF 1,DLP,%SHOW CAL%"; OUTPUT 718;"KEYEXC 1;"</pre>	<p><i>Initializes analyzer.</i></p> <p><i>Defines a function called DLP.</i></p> <p><i>Changes center frequency to measure the calibration signal.</i></p> <p><i>Measures the calibration signal in narrow span. The ^ signifies the end of the function declaration.</i></p> <p><i>Key 1 will now have the "Show Cal" label and perform the function DLP.</i></p> <p><i>Executes key 1.</i></p>
--	--

## KEYDEF User-Defined Key Definition

## Example 2

To redefine the command string without changing the label, enter a single blank space for the command string parameter. If you want to delete the command string and the key label, enter two or more blank spaces for the key label parameter.

```
OUTPUT 718;"KEYDEF 2,%IP;CF 300MHZ;%,%MY KEY|LABEL%;" Defines key 2.
OUTPUT 718;"KEYDEF 2, ,%MY KEY|LABEL%;" Removes the command string
                                           functions IP and CF 300MHZ,
                                           but the key label remains
                                           on screen.

OUTPUT 718;"KEYDEF 2, , ;" Removes the command string
                                           functions and the key label
                                           "MY KEY LABEL".
```

## Description

The recommended key numbers are 1 through 6, and 601 through 1200.

**Note** Using other than the recommended key numbers can cause undesirable results.



The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position  
(The key position range is 1 through 6.)

For example:

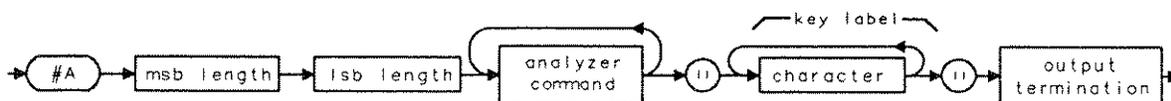
Menu 1 contains key numbers 1 through 6  
Menu 101 contains key numbers 601 to 606  
Menu 200 contains key numbers 1195 to 1200

Menu 1 can be accessed by pressing **MEAS/USER**, **USER MENU(S)**.

Menus 101 through 200 as well as menu 1 can be accessed using the MENU command. For example, executing "MENU 200;" accesses menu 200.

The key label and the command string can be deleted by entering blank spaces in the key label or command string. See example 2.

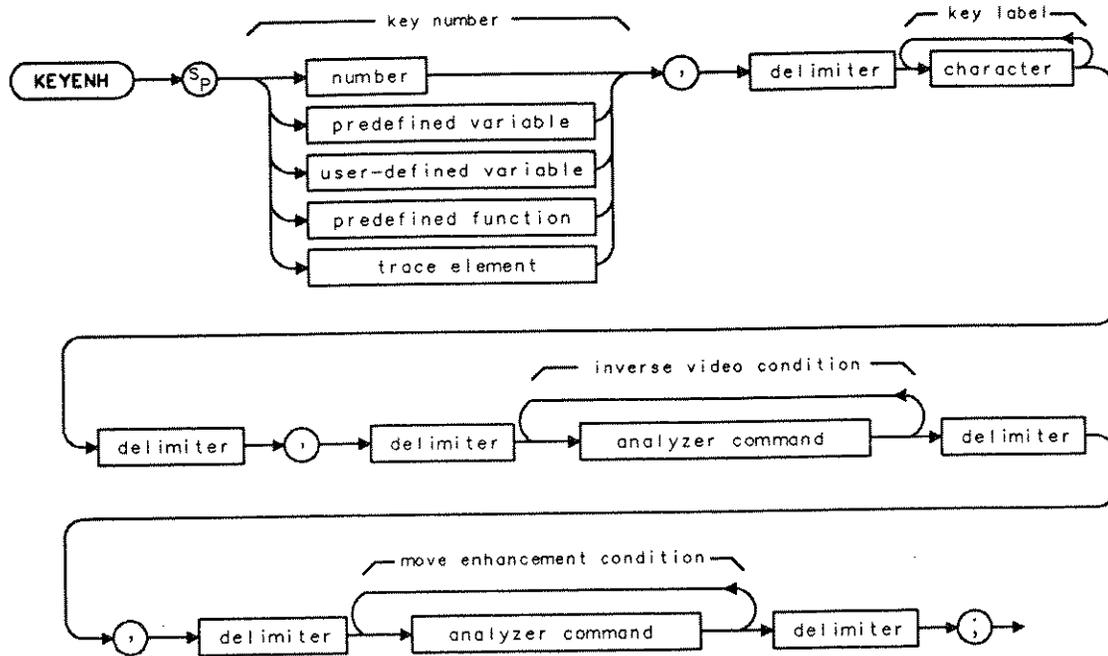
## Query Response



## KEYENH Key Enhance

The KEYENH command allows you to activate part or all of the key label in the inverse video mode, or move the underline from one section of the label to another.

### Syntax



## KEYENH Key Enhance

Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined function	Function that returns a value. Refer to Table 4-1.	Any valid predefined function that returns a value within the key number range.
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	1—6, 601—1200.
Predefined variable	A command that act as a variable. Refer to Table 4-1.	Any valid predefined variable that returns a value within key number range.
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Character	Any valid character. See Table 4-6 and Table 4-7 for additional characters and label functions.	1 to 8 characters per line, use the ( ) symbol or blank spaces to separate softkey label lines.
Analyzer command	Any valid analyzer command.	

Related Commands: DISPOSE, ERASE, FUNCDEF, KEYEXC, KEYLBL.

**Example 1**

Use the KEYENH command to turn on inverse video for the upper label line of key number 1.

```
UL$=CHR$(16)
```

*Defines UL\$ as the character to turn the underline on.*

```
Off$=CHR$(15)
```

*Defines OFF\$ as the character to turn inverse video and underlining off.*

```
OUTPUT 718;"VARDEF I_NV,0;"
```

*The value of LNV acts as a inverse video condition.*

```
OUTPUT 718;"VARDEF M_OVENH,0;"
```

*M\_OVENH acts as the move enhancement condition.*

```
OUTPUT 718;"KEYENH 1,%MY_KEY|";UL$;"ON";Off$;" OFF%,@IF 0,
EQ,I_NV@,@IF 0,EQ,M_OVENH@;"
```

*The MY\_KEY label line changes to inverse video, and OFF (in the lower label line) is underlined.*

**Example 2**

Changing the condition of the inverse video mode to false turns the inverse video off.

```
OUTPUT 718;"MOV I_NV,1;"
OUTPUT 718;"MOV M_OVENH,0;"
```

```
OUTPUT 718;"KEYENH 1,%MY_KEY|";U1$;"ON";Off$;" OFF%,@IF 0,
EQ,I_NV@,@IF 0,EQ,M_OVENH@;"
```

*Change the conditional value of M\_OVENH. The ON portion of the lower label is underlined, the MY\_KEY portion of the label is still in inverse video.*

**Example 3**

Omitting U1\$ and Off\$ turns on inverse video for the upper and lower softkey label lines.

```
OUTPUT 718;"MOV I_NV,0;"
OUTPUT 718;"MOV M_OVENH,1;"
OUTPUT 718;"KEYENH 1,%MY_KEY|ON OFF%,@IF 0, EQ,I_NV@,@IF
0,EQ,M_OVENH@;"
```

*MY\_KEY and ON OFF are displayed in inverse video.*

**Example 4**

Changing the inverse video condition to false turns off the inverse video.

```
OUTPUT 718;"MOV I_NV,1;"
```

```
OUTPUT 718;"KEYENH 1,%MY_KEY|";U1$;"ON";Off$;" OFF%,@IF 0
EQ,I_NV@,@IF 0,EQ,M_OVENH@;"
```

*Changes the conditional value of I\_NV. The inverse video around MY\_KEY of the softkey label is turned off, the ON portion of the softkey label is underlined.*

**Example 5**

Changing the position of U1\$ and Off\$ changes the section of the label that is underlined. In this example, the lower softkey label "FUNCTION" is underlined.

```
OUTPUT 718;"VARDEF I_NV,1;"
OUTPUT 718;"VARDEF M_OVENH,1;"
OUTPUT 718;"KEYENH 1,%MY_KEY|";U1$;"FUNCTION";Off$;"%,@IF
0,EQ,I_NV@,@IF
0,EQ,M_OVENH@;"
```

**KEYENH Key Enhance****Description**

KEYENH turns on the inverse video for the softkey label if the inverse video condition is true. The softkey label is underlined if the move-enhancement mode condition is true.

If you want to use inverse video and underline different sections of the softkey label, use code CHR\$(16) and CHR\$(15). CHR\$(16) is required to turn the underlining on. CHR\$(15) is required to turn off the inverse video and underlining. (See Table 4-7 for more information about label functions.)

The analyzer functions use softkey labels in inverse video for an active function, underlined labels to show a choice.

The recommended key numbers are 1 through 6, and 601 through 1200.

---

**Note** Using other than the recommended key numbers can cause undesirable results.



---

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position

(The key position range is 1 through 6, the top analyzer softkey is key 1, the bottom softkey is key 6.)

For example:

Menu 1 contains key numbers 1 through 6

Menu 101 contains key numbers 601 to 606

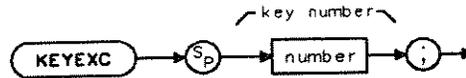
Menu 200 contains key numbers 1195 to 1200

The SAVEMENU command can be used to save keys 1 through 6 in the desired menu number. (Refer to "SAVEMENU".)

## KEYEXC Key Execute

The KEYEXC command executes the specified defined key.

### Syntax



Item	Description/Default	Range
Number	Any valid integer	1—6, 601—1200.

Related Commands: DISPOSE, ERASE, FUNCDEF, KEYDEF, KEYLBL.

### Example

```
OUTPUT 718;"FUNCDEF DLP,^";
OUTPUT 718;"CF 300MHZ";
```

*Defines a function called DLP.  
Changes center frequency to measure the calibration signal.*

```
OUTPUT 718;"SP 1MHZ^";
```

*Measures the calibration signal in narrow span. The ^ signifies the end of the function declaration.*

```
OUTPUT 718;"KEYDEF 1,DLP,%SHOW CAL%";
```

*Key 1 will now have the "Show Cal" label and perform the function DLP.*

```
OUTPUT 718;"KEYEXC 1";
```

*Executes key 1.*

### Description

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position  
(The key position range is 1 through 6.)

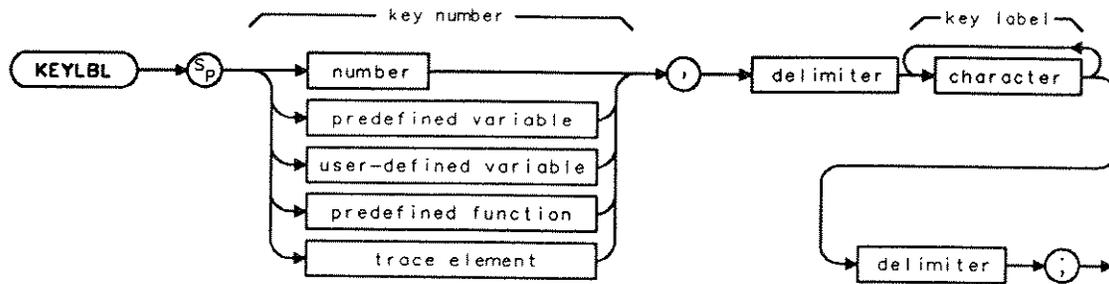
For example:

- Menu 1 contains key numbers 1 through 6
- Menu 101 contains key numbers 601 to 606
- Menu 200 contains key numbers 1195 to 1200

# KEYLBL Key Label

The KEYLBL command renames a key without changing its function.

## Syntax



Item	Description/Default	Range
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	1-6, 601-1200.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Character	Any valid character. See Table 4-6 and Table 4-7 for additional characters and label functions.	1 to 8 characters per label line, use the ( ) symbol or blank spaces to separate softkey label lines.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: DISPOSE, ERASE, FUNCDEF, KEYDEF, KEYEXC.

## Example

Connect CAL OUT to the analyzer input.

```
10 OUTPUT 718;"FUNCDEF DLP,^";
20 OUTPUT 718;"CF 300MHZ";

30 OUTPUT 718;"SP 1MHZ^";
```

*Defines a function called DLP.  
 Changes center frequency to measure the calibration signal.  
 Measures the calibration signal in narrow span. The ^ signifies the end of the function declaration.*

```
40 OUTPUT 718;"KEYDEF 1,DLP,%SHOW CAL%;" Key 1 is now called "SHOW CAL" and  
                                         performs the function DLP.  
50 OUTPUT 718;"KEYEXC 1;"                Executes key 1.  
60 OUTPUT 718;"KEYLBL 1,%NEW NAME%;"      Key 1 is now labeled "NEW NAME", but  
                                         performs the same function.  
  
70 END
```

## Description

The recommended key numbers are 1 through 6, and 601 through 1200.

---

### Note

Using other than the recommended key numbers can cause undesirable results.



---

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position

(The key position range is 1 through 6.)

For example:

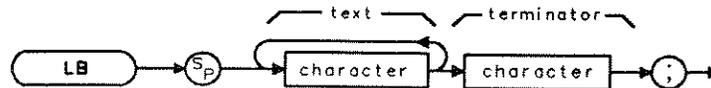
- Menu 1 contains key numbers 1 through 6
- Menu 101 contains key numbers 601 to 606
- Menu 200 contains key numbers 1195 to 1200

To remove a key label, enter a single blank space for the key label parameter. For example, "KEYLBL 1, ;".

## LB Label

The LB command writes text (label) at the current pen position with alphanumeric characters specified in the character field.

### Syntax



Item	Description/Default	Range
Character (text)	Any valid character. See Table 4-6 and 4-7 for additional characters available.	
Character (delimiter)	Any valid character declared as a delimiter by the DT command.	

Prerequisite Command: DT.

Related Commands: TEXT, TITLE.

### Example

```

10 OUTPUT 718;"IP;"
20 OUTPUT 718;"BLANK TRA;ANNOT OFF;"

30 OUTPUT 718;"DT@";

40 OUTPUT 718;"PU;PA 75,175;LB LABEL@";
50 OUTPUT 718;"PU;PA 75,150;LB";
   CHR$(36);"@";

60 OUTPUT 718;"PU;PA 75,125;LB";CHR$(16);
   "AN UNDERLINED LABEL";CHR$(15);"@";
70 OUTPUT 718;"PU;PA 75,100;LB";CHR$(14);
   "A LABEL IN INVERSE VIDEO";CHR$(15);"@";
80 END

```

*Initializes analyzer.*

*Clears trace and annotation from the analyzer screen.*

*Establishes @ as the termination of the label text.*

*Displays LABEL on the analyzer screen.*

*Displays the dollar sign (\$) on the analyzer screen. The semicolons before and after CHR\$(36) prevent the computer from performing a line feed.*

*Displays underlined text.*

*Displays text in inverse video.*

## Description

Each text character is specified by 8 bits in an 8 bit data byte which immediately follows the LB command. Additional characters can be displayed using CHR\$(code) where code represents the ASCII code (see line 50 of the example). Refer to Table 4-6 and Table 4-7 for additional characters and label functions available.

The TITLE command displays the text in the title area only; LB displays the text at the current pen position. When using LB, the end of the text characters must be terminated. If the text is not terminated, instructions and other text following the actual label's statement are displayed on the CRT. The label mode can always be terminated with an ASCII end-of-text code (decimal code 3), or with a character specified previously by the DT command. The terminator character itself must immediately follow the label.

To remove the text written by the LB command, write spaces over the text or use the CLRDSP command.

Table 4-6. Character Set

Code	Char	Code	Char	Code	Char	Code	Char	Code	Char
32	(space)	73	I	114	r	190	$\geq$	237	$\mu$
33	!	74	J	115	s	191	$\sim$	238	$\nu$
34	"	75	K	116	t	192	$\angle$	239	$\omicron$
35	#	76	L	117	u	193	$\Delta$	240	$\pi$
36	\$	77	M	118	v	194	$\int$	241	$\theta$
37	%	78	N	119	w	195	$\neq$	242	$\rho$
38	&	79	O	120	x	196	$\nabla$	243	$\sigma$
39	'	80	P	121	y	199	g	244	$\tau$
40	(	81	Q	122	z	200	h	245	$\upsilon$
41	)	82	R	123	{	201	$\uparrow$	246	
42	*	83	S	125	}	204	L	247	$\omega$
43	+	84	T	126	$\sim$	205	n	248	$\Gamma$
44	,	85	U	160	$\wedge$	206	n	249	$\Delta$
45	-	86	V	162	"	207	o	250	$\Omega$
46	.	87	W	163	$\neq$	208	P	251	$\Sigma$
47	/	88	X	164	$\pounds$	209	$\infty$	252	$\Lambda$
48	0	89	Y	165	$\alpha$	210	r	253	$\Upsilon$
49	1	90	Z	166	$\oplus$	211	s	254	$\Xi$
50	2	91	[	167	.	212	T		
51	3	92	\	168	$\uparrow$	213	$\delta$		
52	4	93	]	169	$\rightarrow$	214	$\vee$		
53	5	94	^	170	$\S$	216	=		
54	6	95	-	171	$\pm$	217	"		
55	7	96	`	172	$\downarrow$	218			
56	8	97	a	173	-	219	$\Pi$		
57	9	98	b	174	$\times$	220	$\Theta$		
58	:	99	c	175	$\div$	221	$\Psi$		
59	;	100	d	176	o	222	$\Phi$		
60	<	101	e	177	1	224	$\varphi$		
61	=	102	f	178	2	225	$\alpha$		
62	>	103	g	179	3	226	$\beta$		
63	?	104	h	180	$^{-1}$	227	$\chi$		
64	@	105	i	181	2	228	$\delta$		
65	A	106	j	182	3	229	$\epsilon$		
66	B	107	k	183	$\sqrt{\quad}$	230	$\phi$		
67	C	108	l	184	$\sim$	231	$\gamma$		
68	D	109	m	185	$\approx$	232	$\eta$		
69	E	110	n	186	$\parallel$	233	L		
70	F	111	o	187	$\therefore$	234	$\zeta$		
71	G	112	p	188	$\leq$	235	r		
72	H	113	q	189	$\equiv$	236	$\lambda$		

Table 4-7. Label Functions

Code	Function	Description
8	back space	Positions the pen back one character width.
10	line feed	Positions the pen position down one character height.
11	vertical tab	Positions the pen up one character height.
12	form feed	Position the pen to the upper-left corner of the display.
13	carriage return	Positions the pen at the far left side of the display.
14	inverse video on	Turns on inverse video.
15	enhancements off	Turns off underlining or inverse video.
16	underlining on	Turns on underlining.

---

## LF

### Base Band Instrument Preset

The LF command performs an instrument preset into base band (band 0).

#### Syntax



Related Commands: IP.

#### Example

```
OUTPUT 718;"LF;"
```

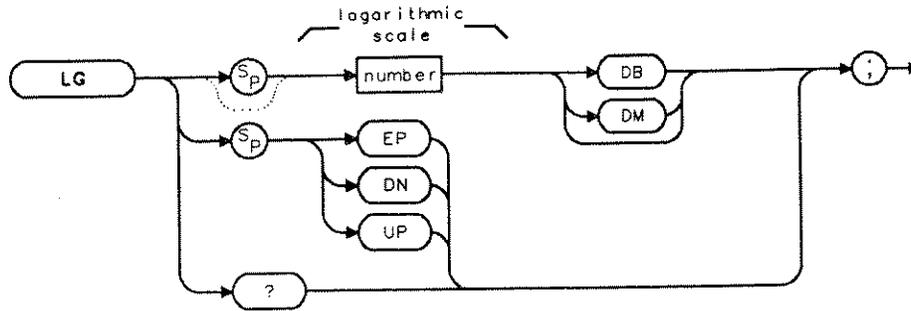
#### Description

For the HP 8592B or the HP 8593A analyzer, use of LF instead of instrument preset (IP) reduces wear on the RF switch (if the analyzer is in base 0 before executing LF).

## LG Logarithmic Scale

The LG command specifies the vertical graticule divisions as logarithmic units without changing the reference level.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default units are dB.	1 to 20 dB.

Preset State: 10 dB.  
 Step Increment: in a 2, 5, 10 sequence.  
 Related Commands: LN.

### Example

OUTPUT 718;"LG 1DB;"

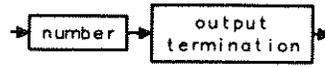
### Description

The vertical scale may be specified (in integers) between 1 and 20 dB per graticule division. If LG is used as the destination in a MOV command, it changes the log scale but does not change the scale from linear to logarithmic.

The functions of the LG command and the **SCALE LOG LIN** (LOG) are identical.

LG Logarithmic Scale

Query Response



---

## LIMIDEL

### Delete Limit-Line Table

The LIMIDEL command deletes all upper and lower segments in the current limit-line table.

#### Syntax

LIMIDEL → ; →

Related Commands: DISPOSE, LIMISEG, SEGDEL, SENTER.

#### Example

```
OUTPUT 718;"LIMIDEL;"
```

#### Description

Use LIMIDEL before entering a new limit line.

---

#### Note



Use STOR, SAVET, or SAVRCLW if you want to save the current limit-line table. LIMIDEL does not affect stored limit-line data.

---

The functions of LIMIDEL, **NEW LIMIT**, and **PURGE LIMITS** are equivalent. LIMIDEL sets the type of limit line to fixed. See "LIMILINE."

## LIMIFAIL

### Limits Failed

The LIMIFAIL command returns a "0" if the last measurement sweep is equal to or within the limit-line bounds.

### Syntax



Related Commands: LIMILINE, LIMITEST.

### Example

10	OUTPUT 718;"IP;SNGLS;CF300MHZ;SP100MHZ;"	<i>Initializes analyzer and changes the frequency and span settings.</i>
20	OUTPUT 718;"LIMIDEL;"	<i>Deletes any limit-line tables, sets the table type to fixed.</i>
30	OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
40	OUTPUT 718;"LIMISEG 250MHZ,-60DB,FLAT;"	<i>Creates an entry to the upper limit-line table.</i>
50	OUTPUT 718;"LIMISEG 290MHZ,-60DB,SLOPE;"	
60	OUTPUT 718;"LIMISEG 295MHZ,-15DB,FLAT;"	
70	OUTPUT 718;"LIMISEG 305MHZ,-15DB,SLOPE;"	
80	OUTPUT 718;"LIMISEG 310MHZ,-60DB,FLAT;"	
90	OUTPUT 718;"LIMISEG 910MHZ,-60DB,FLAT;"	
100	OUTPUT 718;"LIMITEST ON;TS;"	<i>Turns on limit-line testing.</i>
110	OUTPUT 718;"LIMIFAIL?;"	<i>Returns the status of the limit-line testing.</i>
120	ENTER 718;A	
130	DISP A	<i>Displays the result.</i>
140	END	

### Description

LIMIFAIL returns one of the following values:

- 0 indicates the measurement sweep was within the limit-line bounds.
- 1 indicates the measurement sweep failed the lower limit.
- 2 indicates the measurement sweep failed the upper limit.
- 3 indicates the measurement sweep failed both the lower and upper limits.
- 4 indicates that no test was performed. A "4" is returned if LIMITEST is off.

## Query Response

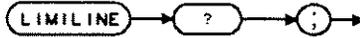


## LIMILINE

### Limit Lines

The LIMILINE command outputs the current limit-line table definitions.

### Syntax



Related Commands: LIMIFAIL, LIMIMODE, LIMIREL, LIMISEG, LOAD, SENTER.

### Example

10	DIM States\$[2000]	<i>Dimensions an array to store the limit-line information.</i>
20	OUTPUT 718;"IP;CF300MHZ;SP100MHZ;"	<i>Initializes analyzer.</i>
30	OUTPUT 718;"LIMIDEL;"	<i>Deletes any limit-line tables, sets the table type to fixed.</i>
40	OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
50	OUTPUT 718;"LIMISEG 250MHZ,-60DB,FLAT;"	<i>Enters a value for the upper limit-line table.</i>
60	OUTPUT 718;"LIMISEG 290MHZ,-60DB,SLOPE;"	
70	OUTPUT 718;"LIMISEG 295MHZ,-15DB,FLAT;"	
80	OUTPUT 718;"LIMISEG 305MHZ,-15DB,SLOPE;"	
90	OUTPUT 718;"LIMISEG 310MHZ,-60DB,FLAT;"	
100	OUTPUT 718;"LIMISEG 910MHZ,-60DB,FLAT;"	
110	OUTPUT 718;"LIMILINE?;"	<i>Gets the current limit-line table definitions.</i>
120	ENTER 718 USING "#,-K";States\$	<i>Enters information into array.</i>
130	PRINT States\$	<i>Prints the current limit-line table definitions.</i>
140	END	

### Description

LIMILINE is used to query the current limit line. Executing LIMILINE returns an ASCII string containing the commands needed to create the limit line.

Use these commands in the order given to build a limit line:

1. LIMIDEL clears the limit-line table.
2. LIMIREL determines whether the values of the limit line are absolute values or positioned relative to the reference-level and center-frequency settings.
3. Use LIMIMODE UPPER or LIMIMODE LOWER to treat the upper and lower amplitude values for a frequency separately. Use LIMISEG to define the amplitude and frequency characteristics of each of the limit-line segments. Or, use LIMIMODE UPLOW to enter upper and lower amplitude values for a frequency together. Use LIMIMODE DELTA

to enter the mid-amplitude value and delta. Use `SENDER` to define the amplitude and frequency characteristics of each of the limit-line segments.

The current limit-line table can be stored in a file on the memory card with the `STOR` or `SAVRCLW` commands. Use `SAVET` or `SAVRCLW` to store the limit-line table in analyzer memory.

When limit testing is enabled, the segments in the current table are interpolated into traces B and C according to the current span and center frequency of the analyzer. After the sweep, each value in trace A is compared to its corresponding bucket in traces B and C.

If the current limit-line table is empty, (for instance after using the command `LIMIDEL`), and limit testing is enabled, trace B and trace C are blanked and set to out-of-range values.

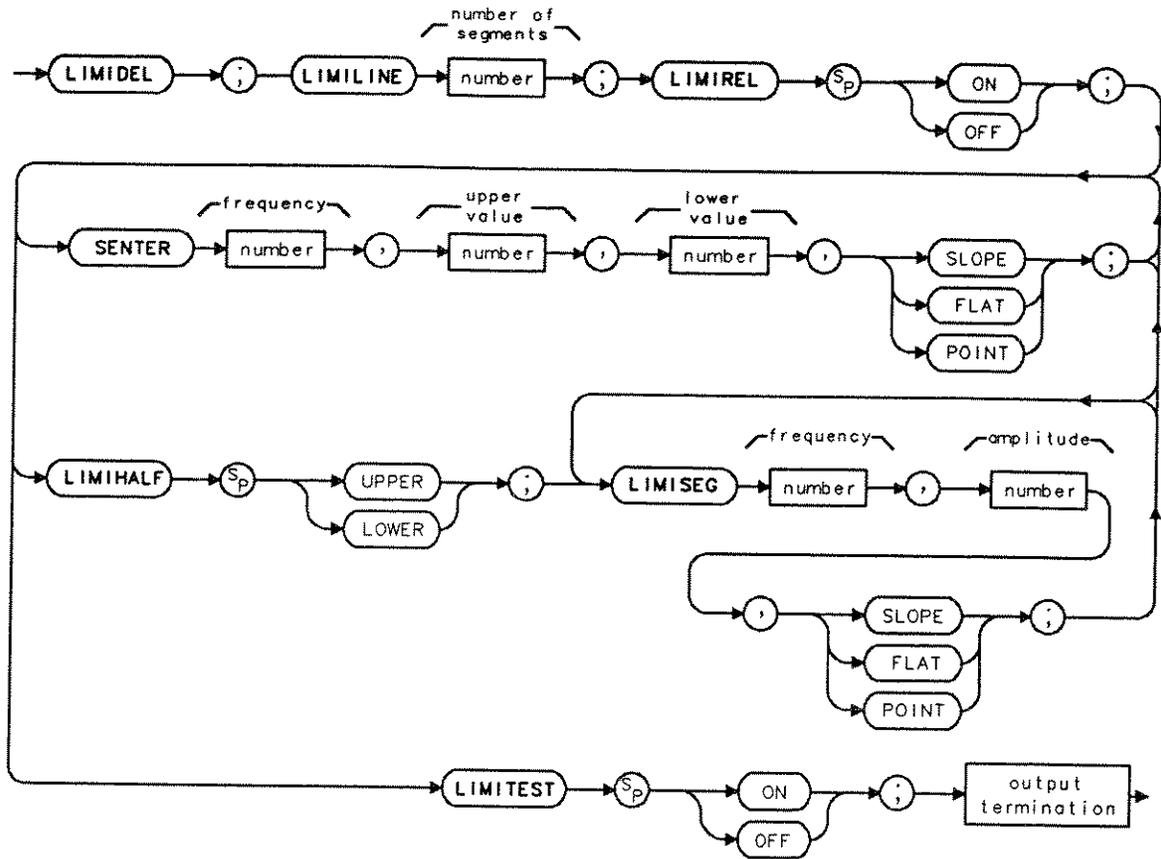
Using `SUB`, `MKPK HI`, and `MKF?`, one can read the point of greatest difference between the trace and limit line.

See `LIMITEST` for more information.

LIMILINE Limit Lines

Query Response

The query response is a character string consisting of LIMIDEL, LIMIREL, LIMIMODE, LIMIHALF, LIMISEG or SENTER commands, which restore the limit-line table. The number of segments indicates the number of segments for the current limit-line table. The LIMIHALF command is for backward compatibility, it is not used as an HP 8590 Series programming command.



## LIMIMIRROR Mirror Limit Line

The LIMIMIRROR command reflects the current definition about the amplitude axis at the largest frequency in the definition.

### Syntax

LIMIMIRROR ;

Related Commands: LIMILINE, LIMISEG, SENTER.

### Example

10	OUTPUT 718;"IP;CF300MHZ;SP100MHZ;"	<i>Initializes analyzer, changes the frequency and span.</i>
20	OUTPUT 718;"LIMIDEL;"	<i>Deletes any limit-line tables, sets the table type to fixed.</i>
30	OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
40	OUTPUT 718;"LIMISEG 250MHZ,-60DB,FLAT;"	<i>Enters a value into the upper limit-line table.</i>
50	OUTPUT 718;"LIMISEG 290MHZ,-60DB,SLOPE;"	
60	OUTPUT 718;"LIMISEG 295MHZ,-15DB,SLOPE;"	
70	OUTPUT 718;"LIMISEG 300MHZ,-10DB,SLOPE;"	
80	OUTPUT 718;"LIMIMIRROR;"	<i>Mirrors the upper limit-line entries.</i>
90	OUTPUT 718;"LIMITEST ON;"	<i>Turns on the limit-line testing and displays the limit-lines.</i>
100	END	

The example results in the following limit-line table:

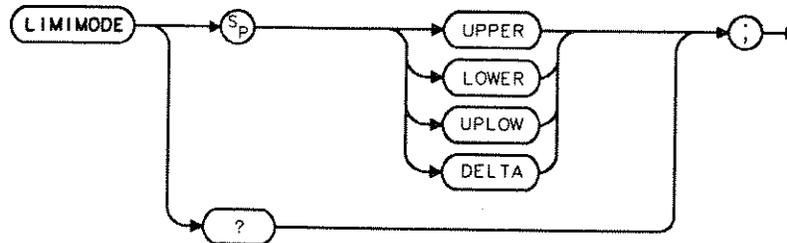
SEG	START_FREQ	UPPER_AMP	TYPE
1	250.0 MHz	-60.0 dBm	FLAT
2	290.0 MHz	-60.0 dBm	SLOPE
3	295.0 MHz	-15.0 dBm	SLOPE
4	300.0 MHz	-10.0 dBm	SLOPE
5	305.0 MHz	-15.0 dBm	SLOPE
6	310.0 MHz	-60.0 dBm	SLOPE
7	350.0 MHz	-60.0 dBm	FLAT

## LIMIMODE

### Limit-Line Entry Mode

The LIMIMODE command determines if the limit-line entries are treated as upper amplitude values, lower amplitude values, upper and lower amplitude values, or mid-amplitude and delta values.

#### Syntax



Related Commands: LIMILINE, LIMISEG, SEGDEL, SENTER.

#### Example

This example uses LIMIMODE for entering segments into the upper limit-line table, then entering a segment into the lower limit-line table (upper and lower limit lines are treated as separate tables). Line 60 demonstrates entering a segment into a combined upper and lower limit-line table.

```
10 OUTPUT 718;"LIMIDEL;"
```

*Deletes the current limit-line table, sets the table type to fixed.*

```
20 OUTPUT 718;"LIMIMODE UPPER;"
```

*Specifies the upper limit-line table.*

```
30 OUTPUT 718;"LIMISEG 300MHZ,-30DB,SLOPE;"
```

*Enters a segment into the upper limit-line table.*

```
40 OUTPUT 718;"LIMIMODE LOWER;"
```

*Specifies the lower limit-line table.*

```
50 OUTPUT 718;"LIMISEG 300MHZ,-70DB,SLOPE;"
```

*Enters a segment into the lower limit-line table.*

```
60 OUTPUT 718;"LIMIMODE UPLow;"
```

*Specifies the upper and lower limit-line tables.*

```
70 OUTPUT 718;"SENER 350MHZ,-30DB,-80DB,FLAT;"
```

*Enters a segment into the upper and lower limit-line tables.*

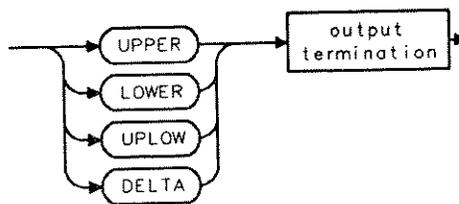
```
80 END
```

## Description

The LIMIMODE command should be used before entering limit-line data with LIMISEG or SENTER: specify LIMIMODE UPPER or LIMIMODE LOWER before using LIMISEG; specify LIMIMODE UPLOW or LIMIMODE DELTA before using SENTER.

The LIMIMODE command determines if the limit-line table entries are to be treated separately (upper or lower) or together (upper and lower) when deleting a segment with SEGDEL (see "SEGDEL"). If limit-line table entries are entered with LIMISEG, they are treated as entries to separate tables even if LIMIMODE UPLOW or LIMIMODE DELTA had been previously specified.

## Query Response

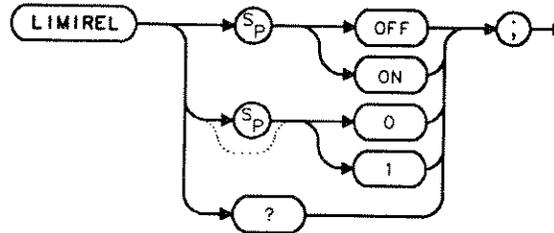


## LIMIREL

### Relative Limit Lines

The LIMIREL command specifies whether the current limit lines are fixed or relative.

#### Syntax



Related Commands: LIMIDEL, LIMILINE.

#### Example

```
OUTPUT 718;"LIMIREL ON;"
```

#### Description

The LIMIREL command determines whether the frequency and amplitude parameters in the limit-line table represent absolute values or relative values referenced to the reference-level and center-frequency settings.

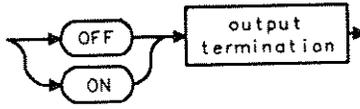
Executing "LIMIREL OFF;" specifies limit-line table parameters as fixed values, and positions the limit line accordingly.

Executing "LIMIREL ON;" specifies limit-line table parameters as relative values, positioning the limit line relative to the reference-level and center-frequency settings.

Fixed limit lines are specified in absolute frequency and amplitude and do not depend upon CF (center frequency) or RL (reference level). Relative limit lines are specified in relative frequency and amplitude, and are positioned with respect to CF and RL.

When the current limit line reference is changed, the segment frequencies and amplitudes are converted according to the current CF and RL.

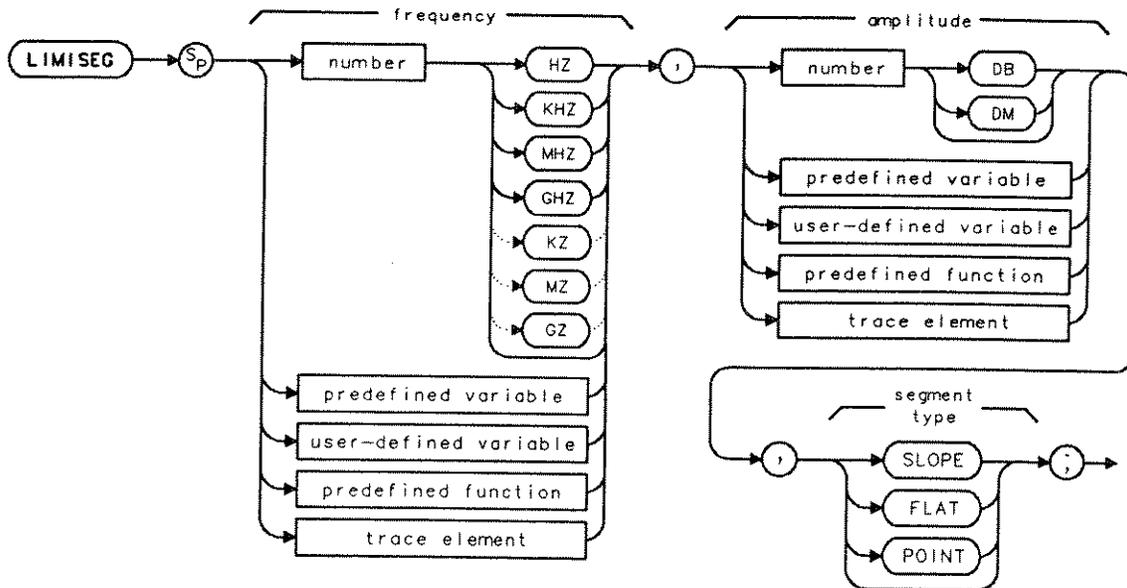
### Query Response



## LIMISEG Enter Limit-Line Segment

The LIMISEG command adds new segments to the current limit line in the upper limit line or the lower limit line.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dBm.	Varies with FOFFSET and ROFFSET.

Related Commands: LIMIDEL, LIMILINE, LIMIMODE, LIMIREL, SEGDEL, SENTER.

### Example

```

10  OUTPUT 718;"IP;CF300MHZ;SP100MHZ;RB 3MHZ;"
20  OUTPUT 718;"LIMIDEL;"
30  OUTPUT 718;"LIMIMODE UPPER;"
40  OUTPUT 718;"LIMISEG 250MHZ,-60DB,FLAT;"
50  OUTPUT 718;"LIMISEG 290MHZ,-60DB,SLOPE;"
60  OUTPUT 718;"LIMISEG 295MHZ,-15DB,FLAT;"
    
```

*Initializes analyzer, changes the frequency, span, and bandwidth. Deletes the current limit-line table, sets the table type to fixed. Specifies the upper limit-line table. Adds segment to the upper limit-line table.*

70	OUTPUT 718;"LIMISEG 305MHZ,-15DB,SLOPE;"	
80	OUTPUT 718;"LIMISEG 310MHZ,-60DB,FLAT;"	
90	OUTPUT 718;"LIMISEG 910MHZ,-60DB,FLAT;"	
100	OUTPUT 718;"LIMIMODE LOWER;"	<i>Specifies the lower limit-line table.</i>
110	OUTPUT 718;"LIMISEG 250MHZ,-75DB,FLAT;"	<i>Adds segment to the lower limit-line table.</i>
120	OUTPUT 718;"LIMISEG 910MHZ,-75DB,FLAT;"	
130	OUTPUT 718;"LIMITEST ON;TS;"	<i>Enables limit-line testing.</i>
140	OUTPUT 718;"LIMIFAIL?;"	<i>Returns the result of limit-line testing.</i>
150	ENTER 718;A	
160	DISP A	<i>Displays the result.</i>
170	END	

## Description

Each limit-line segment is specified with a starting frequency, an amplitude, and a segment type. The segment type defines how the line segment is to extend from its starting point to the next segment.

Three types are available:

- **FLAT** draws a zero-slope line between the coordinate point of the current segment and the coordinate point of the next segment, producing limit-line values equal in amplitude for all frequencies between the two points. If the amplitude values of the two segments differ, the limit-line will "step" to the value of the second segment.
- **SLOPE** draws a straight line between the coordinate point of the current segment and the coordinate point of the next segment, producing limit-line values for all frequencies between the two points.
- **POINT** specifies a limit value for the coordinate point, and no other frequency points, so that a POINT segment specifies a limit value for a single frequency. For an upper limit-line, a POINT segment is indicated by a line drawn from the coordinate point, vertically off the top of screen. For a lower limit-line, a POINT segment is indicated by a line drawn from the coordinate point, vertically off the bottom of screen. The POINT segment type should be used as the last segment in the limit-line table. However, if the last segment in the table is not specified as the POINT segment type, an implicit point is automatically used. If a visible POINT segment at the right-hand edge of the display is not desired, add an explicit last point segment to the limit-line table that is higher in frequency than the stop frequency.

Segments are sorted as they are entered according to starting frequency. A maximum of 20 segments can be defined in each of the upper and lower halves of a limit line. When the type is omitted, the last type given (or SLOPE if no previous type has been given), is used.

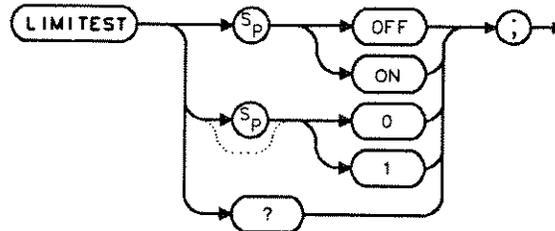
Use LIMISEG if you want to enter amplitude data in the upper or lower limit lines. If you want to enter amplitude data as upper/lower amplitude pairs or as mid/delta pairs, use the SENTER command instead of LIMISEG. Use LIMIMODE to specify entry into the upper limit-line table or the lower limit-line table (see line 30 of example).

## LIMITEST

### Enable Limit Line Testing

The LIMITEST command compares the active trace data with the current limit-line data.

#### Syntax



Preset State: OFF.

Related Commands: LIMIFAIL.

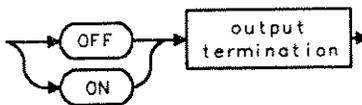
#### Example

10	OUTPUT 718;"IP;CF300MHZ;SP100MHZ;"	<i>Initializes analyzer and changes the frequency and span settings.</i>
20	OUTPUT 718;"LIMIDEL;"	<i>Deletes any limit-line tables, sets the table type to fixed.</i>
30	OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
40	OUTPUT 718;"LIMISEG 250MHZ,-60DB,FLAT;"	<i>Creates an entry to the upper limit-line table.</i>
50	OUTPUT 718;"LIMISEG 290MHZ,-60DB,SLOPE;"	
60	OUTPUT 718;"LIMISEG 295MHZ,-15DB,FLAT;"	
70	OUTPUT 718;"LIMISEG 305MHZ,-15DB,SLOPE;"	
80	OUTPUT 718;"LIMISEG 310MHZ,-60DB,FLAT;"	
90	OUTPUT 718;"LIMISEG 910MHZ,-60DB,FLAT;"	
100	OUTPUT 718;"LIMITEST ON;TS;"	<i>Turns on limit-line testing.</i>
110	OUTPUT 718;"LIMIFAIL?;"	<i>Returns the status of the limit-line testing.</i>
120	ENTER 718;A	
130	DISP A	<i>Displays the result.</i>
140	END	

### Description

A test is made of the data in TRA (trace A) and the result can be read using LIMIFAIL after each sweep.

### Query Response



## LN

### Linear Scale

The LN command specifies the vertical graticule divisions as linear units without changing the reference level.

#### Syntax



Related Commands: LG, RL.

#### Example

```
OUTPUT 718;"LN;"           Selects linear mode.
OUTPUT 718;"LN;RL 30MV;"
```

#### Description

The LN command scales the amplitude (vertical graticule divisions) proportionally to the input voltage, without changing the reference level. The bottom graticule line represents a signal level of zero volts.

Voltage entries are rounded to the nearest 0.1 dB. Thus, 30.16 mV becomes  $-17.4$  dBm for a  $50\Omega$  analyzer system.

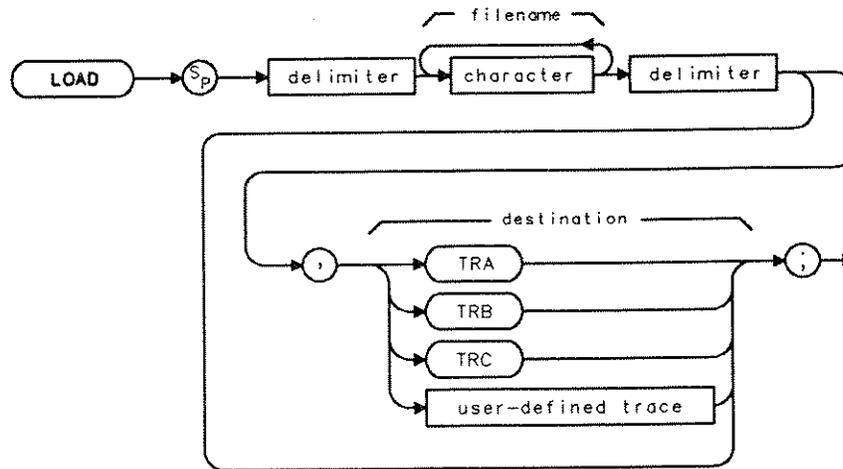
The functions of the LN command and the **SCALE LOG LIN** (LIN) are identical.

## LOAD Load

The LOAD command loads a file from the memory card.

**Option Required: Option 003 with HP 8590B or HP 8592B**

### Syntax



Item	Description/Default	Range
Character	Any valid character.	Any valid filename.
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@

Related Commands: CAT, STOR.

### Example

OUTPUT 718;"LOAD %tMYTRA%,TRA;" *Loads MYTRA from the memory card into trace A.*  
 OUTPUT 718;"LOAD %dMY\_PROG%;" *Loads a program from the memory card into analyzer memory.*

**LOAD Load****Description**

Use TRA, TRB, TRC, or a user-defined trace as the destination when recalling trace data. Omit the destination parameter when recalling downloadable programs, states, limit-line tables, or amplitude correction factors. The result of the LOAD operation is dependent on the type of data loaded. The data type is indicated by the lowercase letter preceding the file name. The file types are as follows:

File Type	Description	Result
a	Amplitude factors	Loads the amplitude correction factors.
d	Downloadable program	Loads the downloadable program defined by FUNCDEF or KEYDEF.
l	Limit lines	Loads the limit-line values.
s	Machine State	Loads new state, current analyzer state is changed to the state that was loaded.
t	Trace	Loads new trace and state, current analyzer trace and state is changed to the state that was loaded.

**Note**

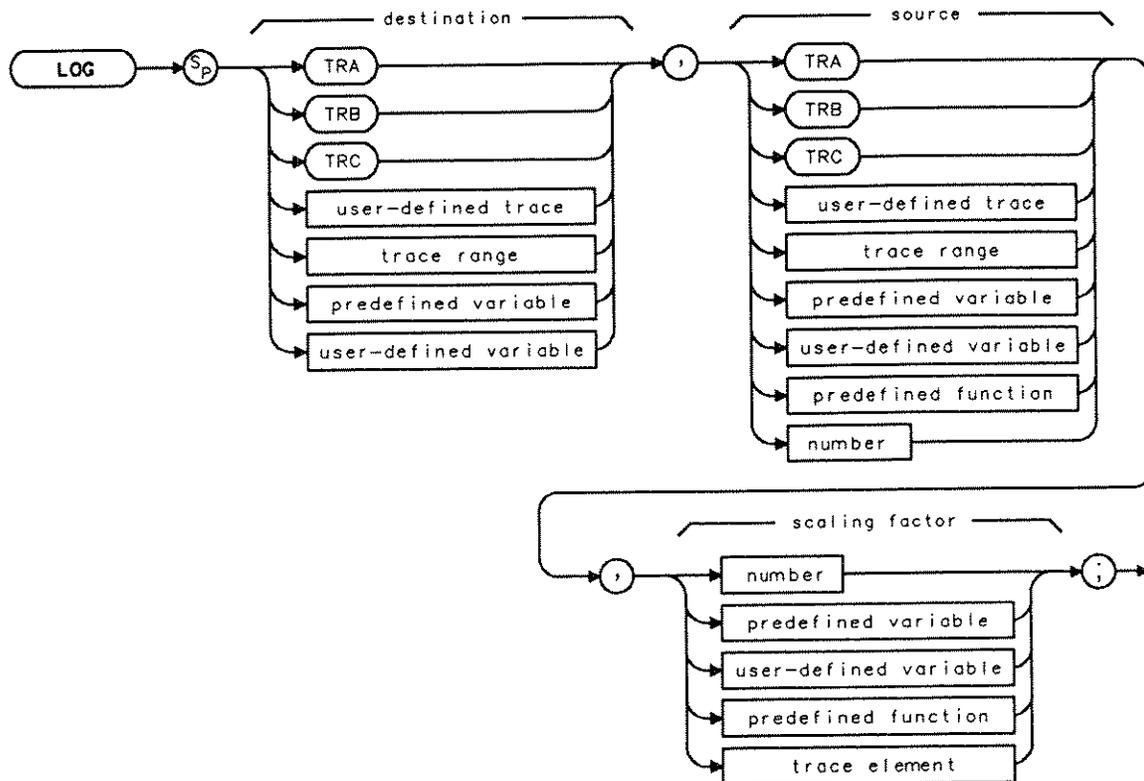
The LOAD command recalls data from the memory card. See "SAVRCLN," "RCLT," or "RCLS" to recall data from analyzer memory.

The functions of LOAD and **LOAD FILE** are similar.

## LOG Logarithm

The LOG command takes the logarithm (base 10) of the source, multiplies the result by the scaling factor, then stores it in the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

**LOG Logarithm**

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

Related Commands: EXP.

**Example 1**

OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
OUTPUT 718;"VARDEF P_POWER,0;"	<i>Defines user-defined variable, called P_POWER, equal to 0.</i>
OUTPUT 718;"LOG P_POWER,5,10;"	<i>P_POWER = 10 × LOG 5.</i>
OUTPUT 718;"P_POWER?;"	<i>Returns value to computer.</i>
ENTER 718;N	<i>Assigns value to computer variable.</i>
DISP USING "D.DD,K";N;"dB"	<i>Displays value on the computer screen.</i>

**Example 2**

This example finds the natural exponential of a number and uses the LOG function to return the original source value of the EXP function.

10 OUTPUT 718;"VARDEF E_XP,0;"	<i>Defines a variable called E_XP.</i>
20 OUTPUT 718;"EXP E_XP,2,2.30259;"	<i>Finds the natural exponential of 2.</i>
30 OUTPUT 718;"E_XP?;"	<i>Returns the natural exponential of 2.</i>
40 ENTER 718;Value	
50 PRINT Value	<i>Prints the value of the exponential.</i>
60 OUTPUT 718;"LOG E_XP,E_XP,2.30259;"	<i>Uses the log function on the exponential value.</i>
70 OUTPUT 718;"E_XP?;"	<i>The log of the exponential value is approximately 2.</i>
80 ENTER 718;Logvalue	
90 PRINT Logvalue	
100 OUTPUT 718;"VARDEF E_XPY,0;"	<i>Declares a variable called E_XPY.</i>
110 OUTPUT 718;"EXP E_XPY,-5,2.30259;"	<i>Finds the natural exponential of -5.</i>
120 OUTPUT 718;"E_XPY?;"	<i>Returns the value of the natural exponential of -5.</i>
130 ENTER 718;Value2	
140 PRINT Value2	<i>Prints the value of the exponential.</i>
150 OUTPUT 718;"LOG E_XPY,E_XPY,2.30259;"	<i>Uses the log function on the exponential value.</i>
160 OUTPUT 718;"E_XPY?;"	<i>The log of the exponential value is approximately -5.</i>
170 ENTER 718;Logval	
180 PRINT Logval	
190 END	

## Description

The scaling factor may be used to improve numerical computations when calculating logarithms of integer trace data. For example, the log of a trace value of 8000 is 3.9, which would be stored as the value 4 in a trace.

The log of trace value of 1 is 0, so the log of a trace containing values from 1 to 8000 would be compressed to values 0, 1, 2, 3, 4. Computational accuracy can be improved by using the scaling factor to scale up the log values before they are stored. In this case, since 3.903 is the log of 8000 and the largest positive trace value is 32,767, a scaling factor of  $32,767/3.903$  or 8,395 may be applied to the data. Since EXP and LOG are inverse functions, the EXP command has a scaling factor that may be used to “undo” the scaling factor of the LOG command.

The LOG command can be used to calculate the natural logarithm by using 2.30259 as the scaling factor.

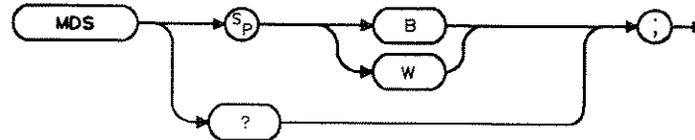
The LOG function returns an invalid result if the source is zero or a negative number.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MDS Measurement Data Size

The MDS command formats binary output data size.

### Syntax



Related Commands: MKA, TDF, TRA.

### Example

These commands transfer trace A in binary, 2 bytes per word.

<code>INTEGER TRACE_A (1:401)</code>	<i>Declares variable, Trace_A.</i>
<code>OUTPUT 718;"SNGLS;TS;"</code>	<i>Activates single-sweep, update trace A.</i>
<code>OUTPUT 718;"TDF B;MDS W;TRA?;"</code>	<i>Reads Trace A in "W" format.</i>
<code>ENTER 718 USING "#,401(W)";TRACE_A(*)</code>	<i>Formats trace A output using data size of one word.</i>
<code>PRINT TRACE_A(*)</code>	<i>Prints out trace A.</i>

### Description

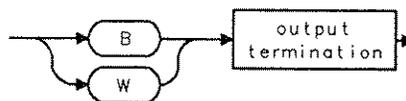
The MDS command formats binary data in one of the following formats:

**B** selects a data size of one 8-bit byte.

**W** selects a data size of one word, which is two 8-bit bytes.

See "TDF" for information on formatting in binary measurements.

### Query Response:



## MDU Measurement Data Units

The MDU command returns values for the analyzer baseline and reference level.

### Syntax



Related Commands: TDF.

### Example

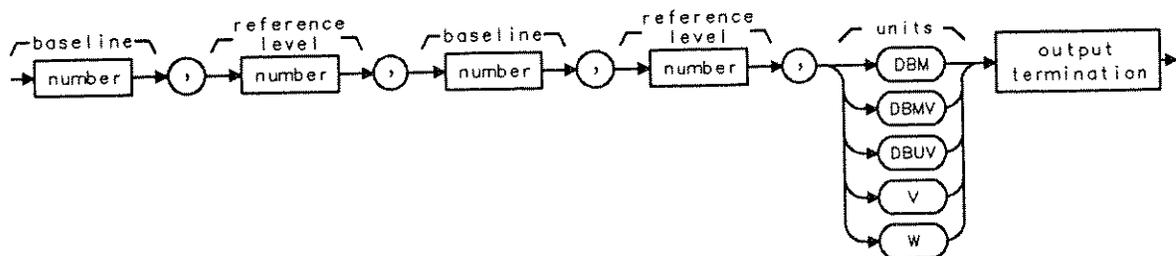
<pre>10 OUTPUT 718;"IP;TDF M;" 20 OUTPUT 718;"RL -10DM;" 30 OUTPUT 718;"MDU?;" 40 ENTER 718;A,B,C,D,A\$ 50 PRINT A,B,C,D,A\$ 60 END</pre>	<p><i>Initializes the analyzer and formats the trace data in measurement units.</i></p> <p><i>Changes the reference level to -10 dBm.</i></p> <p><i>Queries the position of the analyzer baseline and reference level.</i></p> <p><i>Moves the analyzer response to the computer.</i></p> <p><i>Displays the results on the computer screen.</i></p>
---	--

The example returns the following to the controller: 0, 200, -90, -10 dBm. The first two numbers received indicate that the vertical scale spans from 0 to 200 plotter units. The third and fourth number received indicate that the baseline is at -90 dBm, and the reference level is at -10 dBm. So, the baseline value of -90 dBm is equal to 0 plotter units. The reference level of -10 dBm is equal to 200 plotter units.

### Description

The MDU command returns values for the analyzer baseline and reference level, in plotter units and measurement units.

### Query Response

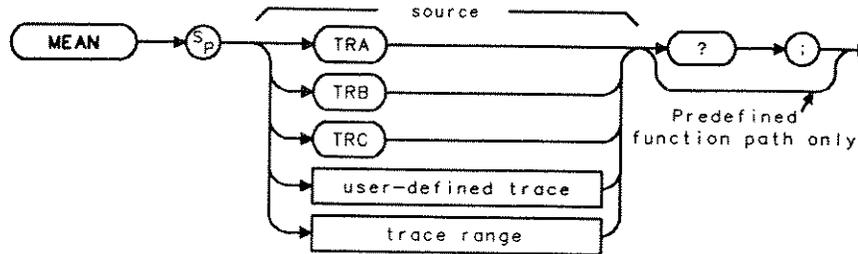


## MEAN

### Trace Mean

The MEAN function returns the mean value of the given trace in measurement units.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command	Any valid trace name.
Trace Range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: RMS, STDEV, VARIANCE.

### Example

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates the single-sweep mode.</i>
30 OUTPUT 718;"CF 300MHZ;SP 1MHZ;"	<i>Sets measurement range.</i>
40 OUTPUT 718;"TS;"	<i>Sweeps trace A.</i>
50 OUTPUT 718;"MEAN TRA?;"	<i>Returns the mean value of trace A to the computer.</i>
60 ENTER 718;Number	<i>Assigns value to computer variable, Number.</i>
70 DISP "MEAN OF TRACE A IS ";Number	<i>Displays result on the computer screen.</i>
80 END	

## Query Response



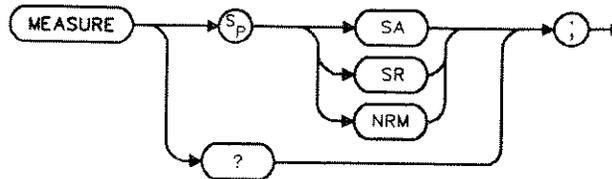
## MEASURE

### Measure Mode

The MEASURE command determines what kind of measurements the analyzer makes: signal analysis, stimulus response, or signal normalization.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



**Related Commands:** Commands affecting amplitude, such as AUNITS, DL, INZ, LN, MKA, MKREAD, MKRL, RL, RLPOS, ROFFSET, TH.  
**Preset State:** MEASURE SA.

### Example

Activate the tracking-generator source output.

OUTPUT 718;"MEASURE SR;"	<i>Sets analyzer to stimulus-response mode.</i>
OUTPUT 718;"SRCPWR -10DM;"	<i>Makes the tracking generator source power the active function.</i>

### Description

"MEASURE SA;" activates spectrum analysis and auto couples the analyzer functions. If AMB ON or AMBPL ON and RLPOS have been executed prior to MEASURE SA, MEASURE SA turns off the reference level position. When normalization is off, all amplitude units are specified in absolute values determined by:

- Amplitude units (AUNITS).
- Reference level (RL).
- Reference-level offset (ROFFSET).
- Input impedance (INZ).
- Logarithm scale (LG).
- Linear scale (LN).
- Amplitude Correction (AMPCOR).

"MEASURE SR;" activates stimulus-response measurements and uncouples the sweep time for faster sweep times. If AMB ON or AMBPL ON and RLPOS are executed, MEASURE SR activates the reference level position. When normalization is used, amplitude units are specified relative to the display level.

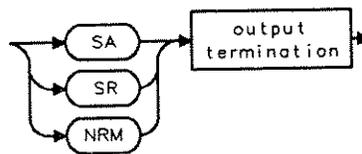
During this relative-amplitude mode, the following parameters are in dB:

- Trace data (TRA, TRB, TRC).
- Display line (DL).
- Threshold (TH).
- Marker amplitude (MKA).

“MEASURE NRM;” recouples the sweep time for accurate signal analysis measurements. If AMB ON or AMBPL ON and RLPOS are executed, MEASURE NRM activates the reference level position.

See “RLPOS” for more information about changing the reference level position.

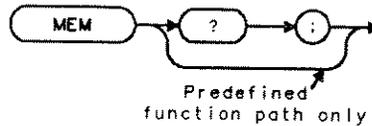
### Query Response



## MEM Memory Available

The MEM command returns the amount of unused analyzer memory.

### Syntax



Related Commands: ACTDEF, DISPOSE, ERASE, FUNCDEF, LOAD, ONDELAY, ONEOS, ONTIME, ONSRQ, ONSWP, TRCMEM, TRDEF, TRMATH, VARDEF.

### Example

```
10 OUTPUT 718;"MEM?;"           Queries the amount of user-allotted memory available.
20 ENTER 718;How_much_memory    Sends response from analyzer to the computer.
30 DISP How_much_memory         Displays the amount of available memory.
40 END
```

### Description

Functions that affect the amount of user-allotted memory include: ACTDEF, FUNCDEF, ONDELAY, ONEOS, ONSRQ, ONSWP, ONTIME, TRDEF, TRMATH, VARDEF.

The MEM command returns the number of available bytes to the controller.

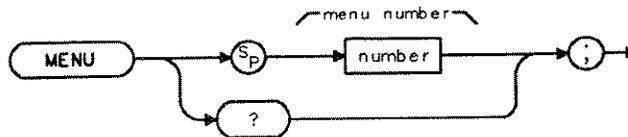
### Query Response



## MENU Menu

The MENU command selects and displays the softkey menus on the analyzer screen.

### Syntax



Item	Description/Default	Range
Number	Any valid integer number.	1, 101—200.

Related Commands: DISPOSE, ERASE, KEYDEF, KEYLBL, SAVEMENU.

### Example

OUTPUT 718; "MENU 1;" *Displays menu 1 on the analyzer screen.*

### Description

Executing "MENU 0;" blanks the softkey menu.

The recommended key numbers are 1 through 6, and 601 through 1200. The key number corresponds to the menu number as follows:

$$\text{key number} = (\text{menu number} - 1) \times 6 + \text{key position}$$

The key position range is 1 through 6. Key 1 is the top softkey, key 6 is the bottom softkey.

For example:

- Menu 1 contains key numbers 1 through 6
- Menu 101 contains key numbers 601 to 606
- Menu 200 contains key numbers 1195 to 1200

The menu numbers 1 and 101 through 200 are the recommended menus available for the user to modify.

**MENU Menu**

**Query Response**



## MF Marker Frequency Output

The MF command returns the frequency (or time) of the on-screen active marker.

### Syntax



Related Commands: MA, MKA, MKCF, MKD, MKF, MKN, MKPK, MKREAD.

### Example

Connect CAL OUT to the analyzer input.

<code>OUTPUT 718;"IP;SNGLS;"</code>	<i>Initializes the analyzer, activates single-sweep mode.</i>
<code>OUTPUT 718;"FA 280MHZ;FB 320MHZ;TS;"</code>	<i>Sets up measurement range.</i>
<code>OUTPUT 718;"MKN;MKPK HI;"</code>	<i>Places marker on peak of calibrator signal.</i>
<code>OUTPUT 718;"MF;"</code>	<i>Takes frequency of marker.</i>
<code>ENTER 718;A</code>	<i>Returns frequency to the computer.</i>
<code>PRINT A</code>	<i>Prints the frequency on the computer screen.</i>

### Description

The MF command returns the frequency of the active marker to the controller if the marker is on-screen. In delta marker mode, nonzero span, "MF;" returns the frequency difference between the two markers. In zero span, "MF;" returns the marker time, or the delta marker time.

The data that is returned by "MF;" depends on many command conditions including TDF, MKREAD, and MDS.

If the trace data format P is used with MF, the result is one real value in time units or frequency units, depending on the marker readout format. (See "MKREAD".)

### Example

<code>OUTPUT 718;"TDF P;MKREAD FRQ;MF;"</code>	<i>This returns a frequency value (in Hz) if not in zero-span.</i>
<code>OUTPUT 718;"TDF P;MKREAD FRQ;MF;"</code>	<i>This returns a time value (in seconds) if in zero-span.</i>
<code>OUTPUT 718;"TDF P;MKREAD PER;MF;"</code>	<i>This returns the time value (in seconds) of 1/(marker frequency).</i>
<code>OUTPUT 718;"TDF P;MKREAD SWT;MF;"</code>	<i>This returns the marker time value (in seconds).</i>
<code>OUTPUT 718;"TDF P;MKREAD IST;MF;"</code>	<i>This returns the frequency value (in Hz) for 1/(marker time).</i>
<code>OUTPUT 718;"TDF P;MKREAD FFT;MF;"</code>	<i>This returns the frequency value (in Hz).</i>

**MF Marker Frequency Output**

If the trace data format is used with trace data format A, the result depends on the setting of the MDS command.

**Example**

OUTPUT 718;"TDF A;MDS B;MF;" *Returns one byte representing the marker position. The byte can assume values 1 to 101.*

OUTPUT 718;"TDF A;MDS W;MF;" *Returns two bytes in a binary word format that has a value from 1 to 401.*

If the trace data format is used with trace data format M, the result is the marker horizontal position value, from 1 to 401, in ASCII.

**Example**

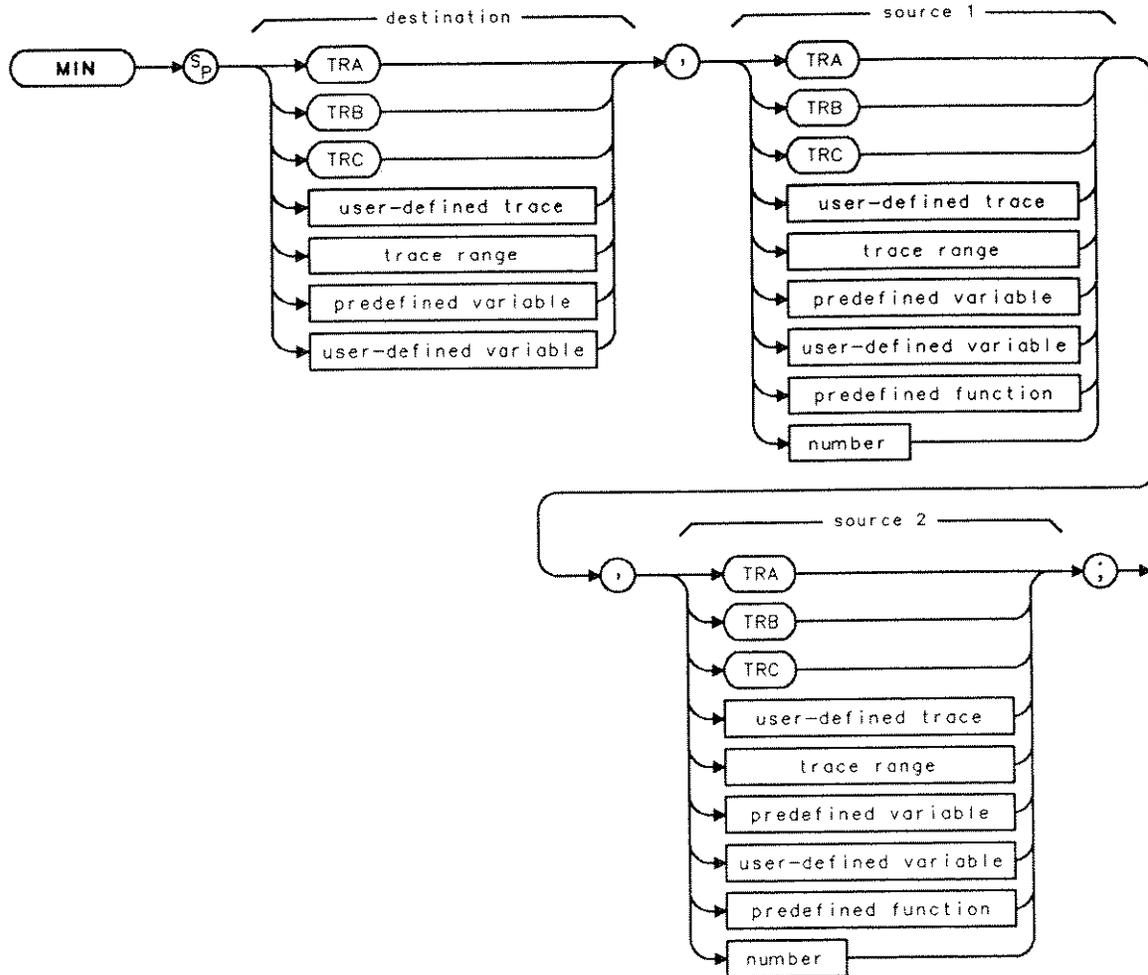
OUTPUT 718;"TDF M;MF;" *Returns marker horizontal position value in ASCII.*

# MIN

## Minimum

The MIN command compares source 1 and 2, point by point, and stores the lesser of the two in the destination.

### Syntax



**MIN Minimum**

Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

Related Commands: MINPOS, MXM, TS.

**Example**

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;"	<i>Activates the single-sweep mode.</i>
30 OUTPUT 718;"VARDEF M_INIMUM,0;"	<i>Defines variable with an initial value of 0.</i>
40 OUTPUT 718;"TS;MKPK HI;"	<i>Sweeps trace A and places the marker at the highest peak.</i>
50 OUTPUT 718;"MIN M_INIMUM,MKA,-20;"	<i>Compares the marker amplitude to -20 dBm. Stores the lesser of the two in M_INIMUM.</i>
60 OUTPUT 718;"M_INIMUM?;"	<i>Returns the result to the computer.</i>
70 ENTER 718;Number	<i>Puts the result in the computer variable, Number.</i>
80 DISP Number	<i>Displays the result.</i>
90 END	

**Description**

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. The values larger than the threshold are replaced by the threshold value in the destination.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MINH Minimum Hold

The MINH command updates trace C with the minimum level detected.

### Syntax



Related Commands: CLRW, MXMH, VAVG, VIEW.

### Example

```
OUTPUT 718;"CLRW TRC;CONTS;" Clears trace C and begin taking data.  
OUTPUT 718;"MINH TRC;" Updates trace C with the minimum level detected.
```

### Description

MINH updates trace C with a new value from a detector only if the new value is smaller than the previous trace data value.

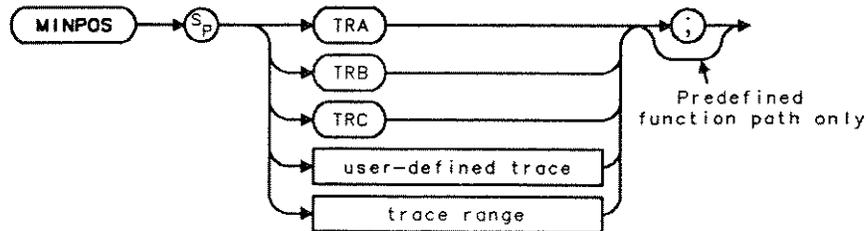
The function of MINH TRC and **MIN HOLD C** are identical.

## MINPOS

### Minimum Position

The MINPOS returns a value which is the  $x$ -axis position (in display units) of the minimum amplitude value in trace A, trace B, trace C, or user-defined trace.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace ACTDEF. TS when using trace data.

Related Commands: MIN, MKMIN, PKPOS.

### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"MINPOS TRA;"   Finds the minimum value of trace A.
30 ENTER 718;Number          Returns value to the computer.
40 DISP Number                Displays result.
50 END
  
```

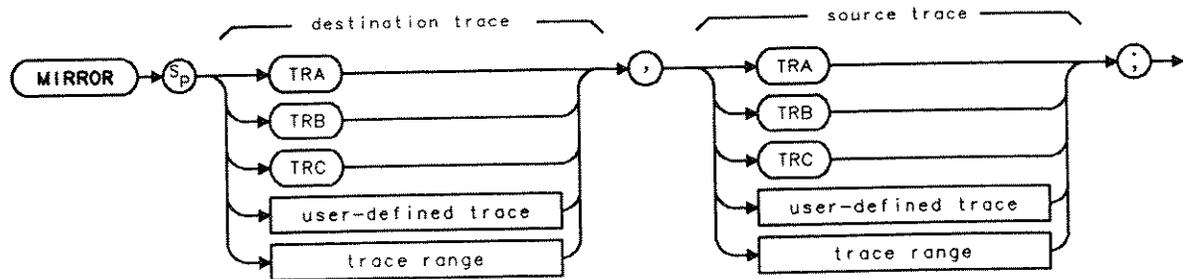
### Description

If a trace range is used with MINPOS, MINPOS returns a value relative to the first element of the trace range. For example, if a trace has a range of 150 to 300 elements, and the minimum value is element 200, MINPOS will return the value of 51.

## MIRROR Mirror Image

The MIRROR command returns the mirror image of the trace.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"SNGLS;"       Activates the single-sweep mode.
30 OUTPUT 718;"TS;"         Takes sweep.
40 OUTPUT 718;"BLANK TRA;"   Blanks trace A from analyzer screen.
50 OUTPUT 718;"MIRROR TRB,TRA;" Moves the mirrored image of trace A into trace B.
60 OUTPUT 718;"VIEW TRB;"   Displays the result.
70 END

```

### Description

The MIRROR command stores the mirror image (with respect to the frequency axis) of a source trace in a destination trace.

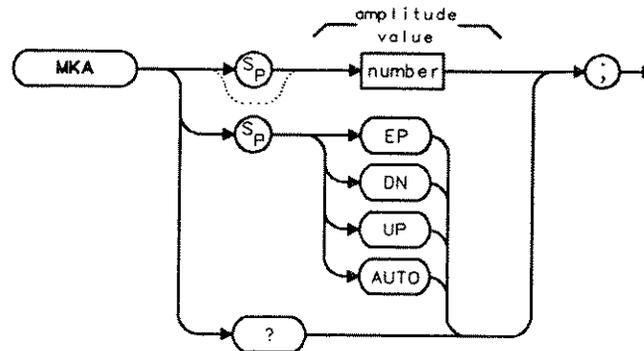
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MKA

### Marker Amplitude

The MKA command specifies the amplitude of the active marker in the current amplitude units when marker type is of fixed or amplitude type. When queried, MKA returns the marker amplitude independent of marker type.

#### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Unit is current amplitude type.	Amplitude range of analyzer.

Step Increment: by 1 dB.

Related Commands: AUNITS, AUTO, MA, MKD, MKN, MKTYPE, TDF.

#### Example

```
OUTPUT 718;"IP;"           Initialize analyzer.
OUTPUT 718;"MKTYPE AMP;"   Change marker type to amplitude.
OUTPUT 718;"MKA -50;"      Places the marker at -50 dBm.
```

#### Description

The MKA command specifies the amplitude of the active marker in current units when the marker is the fixed or amplitude type (see "MKTYPE"). If both the delta marker and active marker are on the screen, "MKA?" returns the amplitude difference between the two markers. Specifying the marker amplitude moves the marker to the point of the trace closest to the given marker amplitude.

If the trace data format P (TDF P), is used with MKA, the result is one real value in the current amplitude units (AUNITS can be used to change the current amplitude units).

**Example**

OUTPUT 718;"TDF P;AUNITS DBM;MKA?;"      *This returns the amplitude value of the marker (amplitude unit is dBm).*

If the trace data format is used with trace data format A, the result depends on the setting of the MDS command.

**Example**

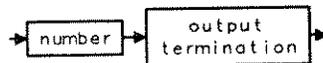
OUTPUT 718;"TDF A;MDS B;MKA?;"      *Returns one byte representing the marker vertical position (–32,768 to 32,767) divided by 32 and then ANDed with 255.*

OUTPUT 718;"TDF A;MDS W;MKA?;"      *Returns two bytes in a binary word format that has a value from –32,768 to 32,767. The value represents the binary trace amplitude value.*

Using the trace data format I is equivalent to the TDF A format. If the trace data format is used with trace data format M, the result is returned in ASCII measurement units (–32,768 to 32,767).

**Example**

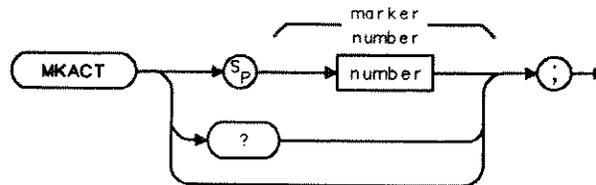
OUTPUT 718;"TDF M;MKA?;"      *Returns one vertical position value in measurement units.*

**Query Response**

## MKACT Activate Marker

The MKACT command establishes the active marker.

### Syntax



Item	Description/Default	Range
Number	Any valid integer. Default value is 1.	1 to 4.

Related Commands: MA, MKA, MKF.

### Example

OUTPUT 718;"MKACT 4;" *Marker 4 becomes marker 1.*

### Description

There can be four different markers, but only one marker can be active at any time.

When this command is used the following results occur:

- The marker number supplied by the command is made marker 1, the active marker.
- If the marker number is not already on, the marker is turned on with preset type (position), and the marker is placed at center screen. The trace chosen is the first displayed trace found in order: trace A, trace B, trace C.

### Note

Using MKACT replaces marker 1 with the new marker function. The contents of marker 1 before MKACT are not saved.



### Query Response

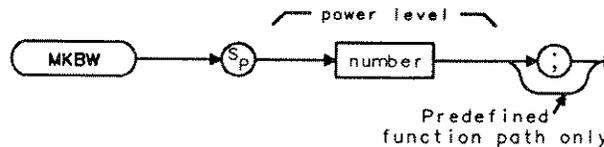
“MKACT?;” returns the marker number.



## MKBW Marker Bandwidth

The MKBW command returns the bandwidth at the specified power level relative to an on-screen marker (if present) or the signal peak (if no on-screen marker is present).

### Syntax



Item	Description/Default	Range
Number	Any valid negative integer.	0 to the amplitude of the noise floor.

### Example

OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
OUTPUT 718;"CF 300MHZ;SP 100MHZ;SNGLS;"	<i>Changes the center frequency and span, then activates the single-sweep mode.</i>
OUTPUT 718;"TS;MKPK HI;"	<i>Updates sweep, places marker on signal peak.</i>
OUTPUT 718;"MKBW -3;"	<i>Uses the MKBW function to find the signal bandwidth at -3 dB below the marker.</i>

### Description

The MKBW command also displays (in the message area) the bandwidth at the power level in dB below the current marker position or the current signal peak.

## MKCF Marker to Center Frequency

The MKCF command sets the center frequency equal to the marker frequency and moves the marker to the center of the screen.

### Syntax



Related Commands: CF, MKF.

### Example

This example provides a quick way to center the desired frequency on the analyzer screen.

```

10 OUTPUT 718;"IP;SP 1MHZ;SNGLS;"      Initializes analyzer, activates single-sweep mode.
20 INPUT "ENTER IN DESIRED STATION
    FREQUENCY, IN MHZ",Freq
30 OUTPUT 718;"CF ";Freq;"MHZ;"      Changes analyzer center frequency.
40 OUTPUT 718;"TS;MKPK HI;MKCF;TS;"    Updates trace, places marker at the signal peak
                                        and centers it.

60 END

```

### Description

This command is performed only if an active marker is present on-screen.

The functions of the MKCF and **MARKER -> CF** are identical.

## MKCONT

### Marker Continue

The MKCONT command resumes the sweep after execution of a MKSTOP command.

#### Syntax



Related Commands: MKSTOP.

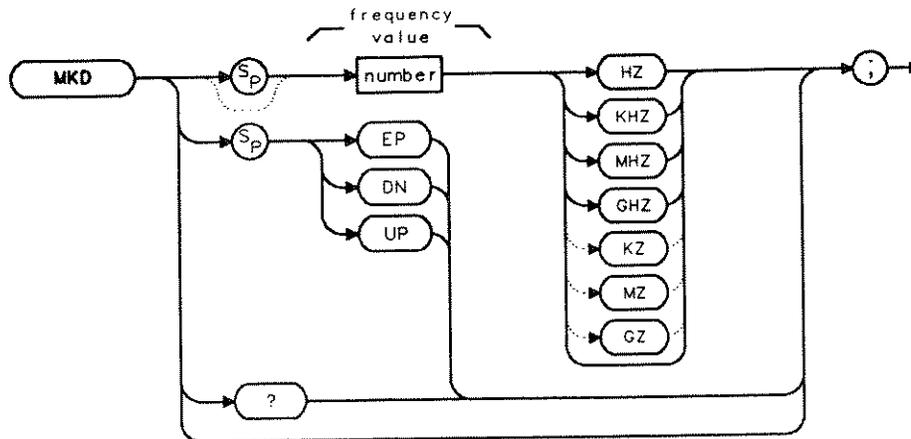
#### Example

```
10 OUTPUT 718;"IP;TS;"      Initializes analyzer.
20 OUTPUT 718;"MKPK HI;"   Creates an active marker.
30 OUTPUT 718;"MKSTOP;"    Stops sweep at marker.
40 OUTPUT 718;"MKCONT;"    Resumes sweep.
50 END
```

## MKD Marker Delta

The MKD function activates the delta marker.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz, default value is value of the active marker.	Start frequency to stop frequency of analyzer.

Step Increment: by 1/10 of the frequency span.

Related Commands: AUTO, MA, MKCF, MKF, MKN, MKSP, MKSS, MKPK.

### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"MKMIN;"       Places a marker at the minimum amplitude of trace.
30 OUTPUT 718;"MKD;"        Activates marker delta.
40 OUTPUT 718;"MKPK HI;"    Places marker at highest amplitude of trace.
50 OUTPUT 718;"MKSP;"       Changes span to the values of the left and right markers.
60 END

```

## MKD Marker Delta

### Description

The MKD command computes the frequency and amplitude difference of the active marker and a special marker, called the delta or differential marker. These values are displayed in the display readout.

The differential value of the frequency is equal to the active marker frequency minus the delta marker frequency. The differential value of the amplitude is equal to the active marker amplitude minus the delta marker amplitude.

If an active marker is not on the screen, MKD positions an active marker at center screen. If a delta marker is not on the screen, MKD places one at the specified frequency, or at the current active marker. If the active marker is in amplitude mode, the delta marker is placed at the same amplitude as the active marker (or a specified value).

To read the amplitude, use "MKA?". The results are displayed on the analyzer screen.

In linear mode, MKD computes the ratio of the amplitudes of the active and delta markers rather than the difference. This results in similar treatment for logarithmic and linear data because the delta of the difference of two logarithmically generated numbers results in the logarithmically generated value of the ratio of the two numbers.

In zero span, "MKD?;" computes and returns time differences.

The functions of MKD and **MARKER DELTA** are identical.

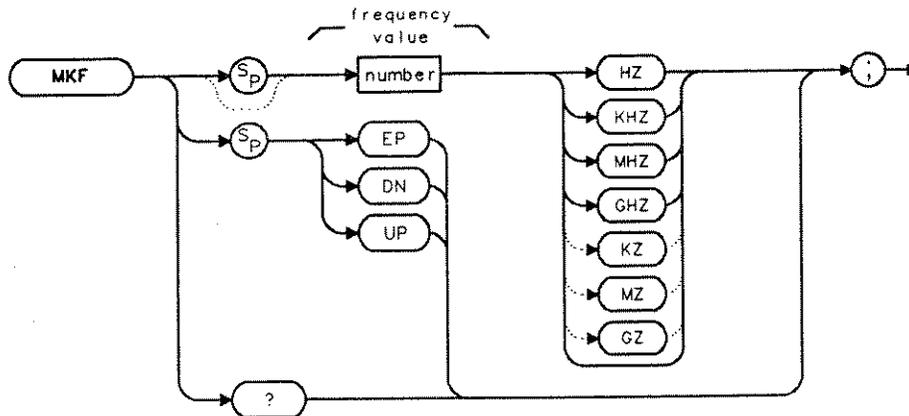
### Query Response



## MKF Marker Frequency

The MKF command specifies the frequency value of the active marker.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	Start frequency to stop frequency of analyzer.

Step Increment: 10% of the screen.

Related Commands: AUTO, MKA, MKD, MKCF, MKPK.

### Example

OUTPUT 718; "MKF 600MHZ;" *Places an active marker at 600 MHz.*

### Description

When queried, MKF returns the active marker frequency as a real number. In zero span, "MKF?;" returns the time value.

The data that is returned by MKF depends on many command conditions, including TDF, MKREAD, and MDS.

If the trace data format P is used with MKF, the result is one real value in time units or frequency units, depending on MKREAD.

**MKF Marker Frequency****Example**

OUTPUT 718;"TDF P;MKREAD FRQ;MKF?;" *This returns a frequency value (in Hz) if not in zero-span.*

OUTPUT 718;"TDF P;MKREAD FRQ;MKF?;" *This returns a time value (in seconds) if in zero-span.*

OUTPUT 718;"TDF P;MKREAD PER;MKF?;" *This returns the time value (in seconds) of 1/(marker frequency).*

OUTPUT 718;"TDF P;MKREAD SWT;MKF?;" *This returns the marker time value (in seconds).*

OUTPUT 718;"TDF P;MKREAD IST;MKF?;" *This returns the frequency value (in Hz) for 1/(marker time).*

OUTPUT 718;"TDF P;MKREAD FFT;MKF?;" *This returns the frequency value (in Hz).*

If the trace data format is used with trace data format A, the result depends on the setting of the MDS command.

**Example**

OUTPUT 718;"TDF A;MDS B;MKF?;" *Returns one byte representing the marker position.*

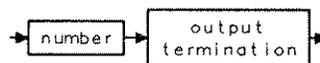
OUTPUT 718;"TDF A;MDS W;MKF?;" *Returns two bytes in a binary word format that has a value from 1 to 401.*

Using the trace data format I is equivalent to the TDF A format.

If the trace data format is used with trace data format M, the result is the marker horizontal position value, from 1 to 401, in ASCII.

**Example**

OUTPUT 718;"TDF M;MKF?;" *Returns marker horizontal position value in ASCII.*

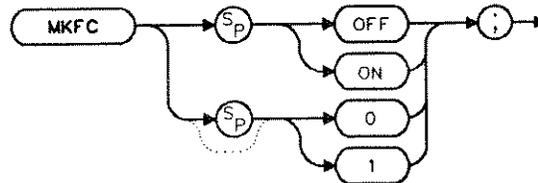
**Query Response**

## MKFC Marker Counter

The MKFC command turns the marker counter on or off.

Model Required: HP 8591A or HP 8593A

### Syntax



Related Commands: MKFCR.

### Example

OUTPUT 718; "MKFC ON;" *Turns on the marker counter.*

### Description

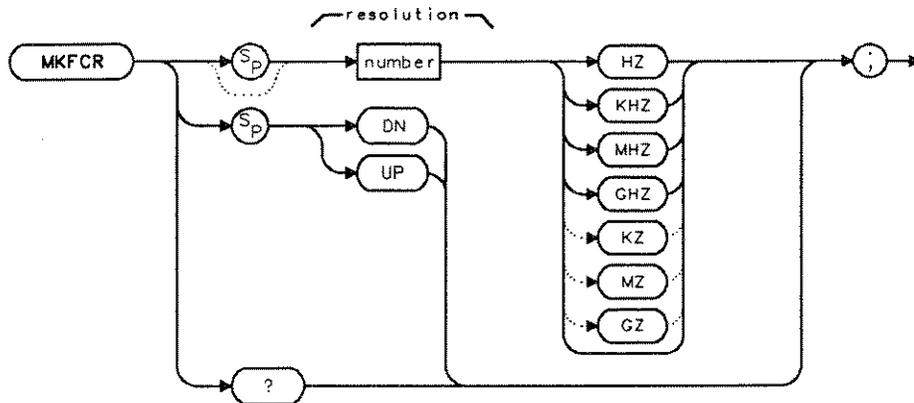
The functions of the MKFC command function and **MKR CNT ON OFF** are identical.

## MKFCR Marker Counter Resolution

The MKFCR commands sets the resolution of the marker counter.

Model Required: HP 8591A or HP 8593A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	0 Hz to 100 kHz.

Preset State: Marker counter resolution is set to 0, but a calculated value of 1800 Hz is returned if the marker counter resolution is queried.

Related Commands: AUTO, MKFC.

### Example

```
OUTPUT 718;"MKFCR 10KHZ;"  Sets the marker counter resolution to 10 kHz.
OUTPUT 718;"MKFCR?;"      Queries the marker counter resolution.
ENTER 718;A                Gets the query response.
DISP A                     Displays the marker counter resolution.
```

### Description

The MKFCR command function is identical to `CNT RES AUTO MAN`. "MKFCR 0;" auto-couples the marker counter resolution to the frequency span.

## Query Response



## MKMIN

### Marker Minimum

The MKMIN command moves the active marker to the minimum value detected.

#### Syntax



Related Commands: MKPK, SMOOTH, TH, VAVG.

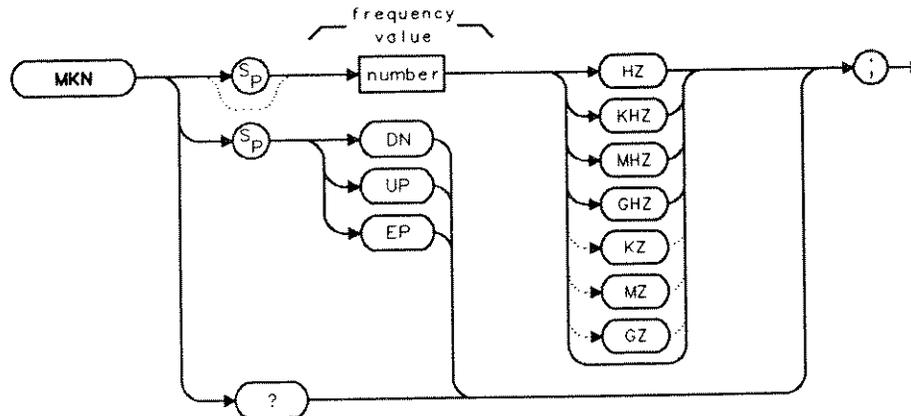
#### Example

<pre> 10 OUTPUT 718;"IP;SNGLS;" 20 INPUT "ENTER IN THE START FREQUENCY,     IN MHZ",Start_freq 30 INPUT "ENTER IN THE STOP FREQUENCY,     IN MHZ",Stop_freq 40 OUTPUT 718;"FA ";Start_freq;"MHZ" 50 OUTPUT 718;"FB ";Stop_freq;"MHZ" 60 OUTPUT 718;"TS;MKPK HI;MKD;MKMIN;MKF?;" 70 ENTER 718;Delta_freq 80 PRINT "DIFFERENCE IN FREQUENCY IS ",     Delta_freq,"HZ" 90 END </pre>	<p><i>Initializes analyzer, activates single-sweep mode.</i></p> <p><i>Sets the start frequency.</i></p> <p><i>Sets the stop frequency.</i></p> <p><i>Updates trace, finds trace peak, turns on marker delta function, finds the minimum value of trace, and return the frequency delta.</i></p> <p><i>Gets the result from analyzer.</i></p>
---	---

## MKN Marker Normal

The MKN command activates and moves the marker to the specified frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz. Default value is center frequency.	Start frequency to stop frequency of analyzer.

Step Increment: by 1/10 of the frequency span.  
Related Commands: AUTO, MKA, MKD, MKF, MKPK.

### Example

```

10 INPUT "ENTER IN THE START FREQUENCY, IN MHZ",Start_freq
20 INPUT "ENTER IN THE STOP FREQUENCY, IN MHZ",Stop_freq
30 OUTPUT 718;"IP;FA ";Start_freq;"MHZ"

40 OUTPUT 718;"FB ";Stop_freq;"MHZ"
50 OUTPUT 718;"MKN EK;"

60 PRINT "PLACE MARKER ON THE DESIRED SIGNAL"
70 PRINT "PRESS HOLD THEN PRESS CONTINUE"
80 PAUSE
90 OUTPUT 718;"MKN?;"

100 ENTER 718;Mkr
    
```

*Initializes analyzer and changes the start frequency.  
Changes the stop frequency.  
Enables the front-panel knob.*

*Gets the frequency of the marker.  
Puts it into the computer variable, Mkr.*

**MKN Marker Normal**

```
110 PRINT "MARKER FREQUENCY IS ",Mkr,"Hz"  
120 END
```

*Prints the result.*

**Description**

In nonzero span, "MKN?;" returns the frequency value. In zero span, "MKN?;" returns the time value. Using the AUTO parameter turns off the marker.

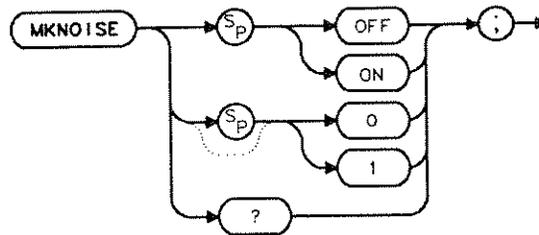
The functions of the MKN command and **MARKER NORMAL** are identical.

**Query Response**

## MKNOISE Marker Noise

The MKNOISE command displays the average noise level at the marker.

### Syntax



Related Commands: MKA, MKF, MKMIN, MKN.

### Example

```
OUTPUT 718;"IP;CF 300MHZ;SNGLS;"
```

*Initializes analyzer, changes center frequency, activates single-sweep mode.*

```
OUTPUT 718;"SP 10MHZ;DET SMP;TS;"
```

*Changes span, activates sample detector, updates trace.*

```
OUTPUT 718;"MKPK HI;MKA?;"
```

*Places marker on highest point of trace, queries marker amplitude.*

```
ENTER 718;Amp_one
```

*Puts the analyzer response in the computer variable, Amp\_one.*

```
OUTPUT 718;"MKD UP;UP;MKNOISE ON;MKA?;"
```

*Moves marker and turns on the marker noise function.*

```
ENTER 718;Amp_two
```

*Puts the analyzer response in the computer variable, Amp\_two.*

```
OUTPUT 718;"MKNOISE OFF;"
```

*Turns off the marker noise function.*

```
DISP Amp_two
```

*Displays the result.*

```
C_to_n=Amp_one - Amp_two
```

*Calculates the carrier to noise ratio.*

```
PRINT "CARRIER TO NOISE RATIO IN 1 HZ
```

```
BANDWIDTH IS ";C_to_n
```

*Outputs result.*

```
PRINT " DB"
```

**MKNOISE Marker Noise****Description**

The value is normalized to a 1 Hz bandwidth. Use “MKA?;” to read the noise marker value.

The noise marker averages 32 trace data values about the location of the marker on the frequency/time scale. The 32 values averaged, except for the first 15 or last 14 values in the trace, commence with the 16th point to the left of the marker, and end with the 15th point to the right of the marker. Note that the data values averaged are not exactly symmetrical with respect to marker position. At the trace end points, the analyzer uses the nearest 32 data values. So while the marker may be moved to trace position 1 to 15, the actual amplitude value returned will be the same value for any marker position from 1 to 15. A similar situation applies for markers at the end of the trace.

A nominal correction for equivalent noise bandwidths is made by the firmware based on a nominal 3 dB resolution bandwidth. The firmware assumes the noise bandwidth is 1.12 times the resolution bandwidth. This means the shape of the resolution bandwidth filters cause the noise power to be overstated by 1.12 times. The detection mode also affects the measurement. If in log mode, the log detector understates the noise response. To compensate, 2.5 dB is added to the measurement. If the detector is in linear mode, the firmware uses 1.05 dB as a correction value.

The final reported value will then be, with the result reported in dBm in a 1 Hz bandwidth:

(in log detector mode)

$$(\text{Averaged value over 32 values}) - 10 \times (\log[1.12 \times \text{Resolution bandwidth}]) + 2.5 \text{ dB}$$

(in linear detector mode (dBm) units)

$$(\text{Averaged value over 32 values}) - 10 \times (\log[1.12 \times \text{Resolution bandwidth}]) + 1.05 \text{ dB}$$

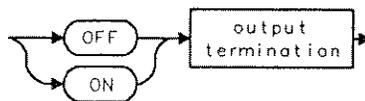
In linear detector mode with the normal display of voltage units, the noise marker voltage value will be related to the present marker voltage by this relation.

$$(V_{\text{noise\_marker}})^2 = (V_{\text{average}})^2 \times 1.12 \times \text{Resolution bandwidth} \times 0.7824$$

$$V_{\text{noise\_marker}} = V_{\text{average}} / (1.12 \times \text{Resolution bandwidth} \times 0.7824)^{0.5}$$

$$V_{\text{noise\_marker}} = V_{\text{average}} \times 1.06633 / (\text{Resolution bandwidth})^{0.5}$$

The functions of MKNOISE and **MKNOISE ON OFF** are identical.

**Query Response**

## MKOFF Marker Off

The MKOFF command turns off either the active marker or all markers.

### Syntax



Related Commands: MKA, MKACTION, MKCF, MKD, MKF, MKN, MKPK.

### Example

OUTPUT 718; "MKOFF ALL;" *Turns off all the on-screen markers.*

### Description

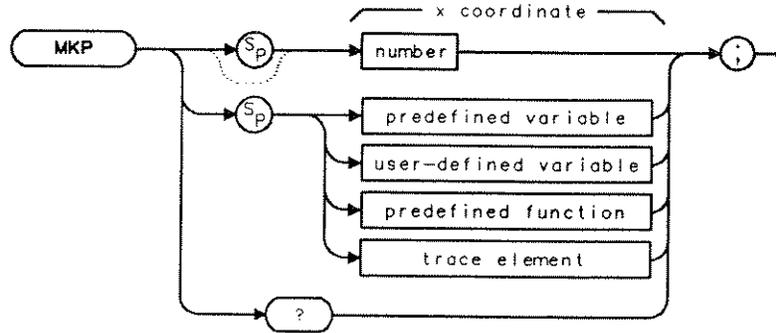
If the ALL parameter is omitted, only the active marker is turned off.

The functions of MKOFF ALL and **MARKERS OFF** are identical.

## MKP Marker Position

This command places the active marker at the given *x*-coordinate.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	1 to 401.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: MKA, MKCF, MKD, MKMIN.

### Example

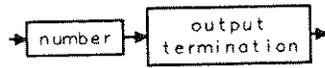
OUTPUT 718;"IP;"            *Initializes analyzer.*  
 OUTPUT 718;"MKP 100;"    *Moves the active marker to a element 100 of trace A.*

### Description

If no marker is active, the marker is turned on with preset type (position) and marker is placed at the given screen position. The trace chosen is the first displayed trace found in order: trace A, trace B, trace C.

Note that MKP and MKCF commands perform different functions. MKCF sets the center frequency equal to the marker frequency and moves the marker to the center of the screen. MKP places the marker to the position of the element specified.

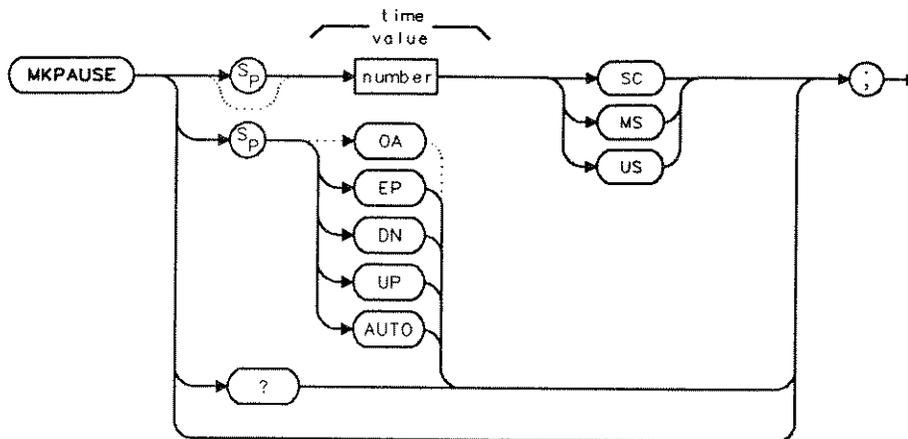
## Query Response



## MKPAUSE Marker Pause

The MKPAUSE command pauses the sweep at the active marker for the duration of the delay period.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	2 ms to 100 s.

Step Decrement: Time value/2.

Step Increment: Time value × 2.

Related Commands: DEMOD, MKA, MKF, MKN, MKOFF, ST.

### Example

OUTPUT 718; "MKPAUSE 10SC;" *Changes the marker pause time to 10 seconds.*

### Description

To turn MKPAUSE off, turn off markers (or, send "MKPAUSE 0;").

The MKPAUSE command requires a sweep time of 50 ms or longer. The sweep time is changed to 50 ms if MKPAUSE is used with a shorter sweep time (less than 50 ms).

After MKPAUSE is executed, the sweep must be completed before another command will be executed.

The functions of the MKPAUSE command and **MKPAUSE ON OFF** are identical.

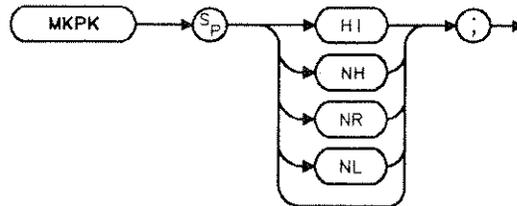
### Query Response



## MKPK Marker Peak

The MKPK command positions the active marker on signal peaks.

### Syntax



Related Commands: MKCF, MKF, MKOFF, MKPX, TH.

### Example

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"SNGLS;TS;MKPK HI;"	<i>Places active marker on highest peak.</i>
30 OUTPUT 718;"MKA?;"	<i>Returns amplitude value of marker to the computer.</i>
40 ENTER 718;A	<i>Puts the analyzer response in the computer variable, A.</i>
50 DISP A	<i>Displays amplitude value.</i>
60 END	

### Description

Executing MKPK HI, or simply MKPK, positions the active marker at the highest signal detected. If an active marker is on the screen, the parameters move the marker accordingly.

HI (highest) moves the active marker to the highest peak.

NH (next highest) moves the active marker to the next signal peak of lower amplitude.

NR (next right) moves the active marker to the next signal peak of higher frequency.

NL (next left) moves the active marker to the next signal peak of lower frequency.

The functions of the MKPK HI command and the **PEAK SEARCH** are identical. The MKPK parameters correspond to **NEXT PEAK**, **NEXT PK RIGHT**, and **NEXT PK LEFT**.

---

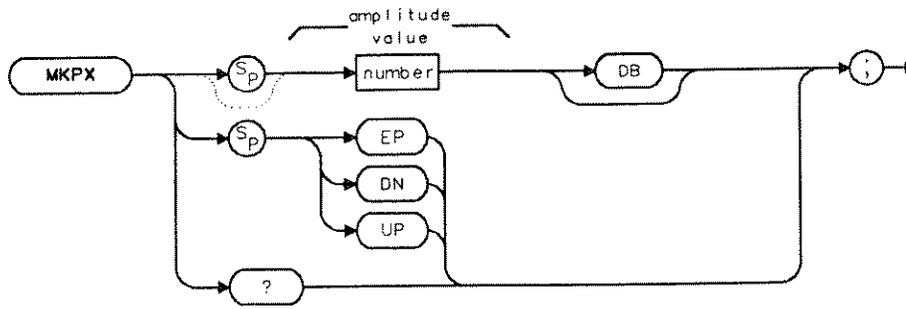
**Note** This function is for use with the frequency markers only.



## MKPX Marker Peak Excursion

The MKPX command specifies the minimum signal excursion for the analyzer internal peak-identification routine.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dB.	0 to 100 dB.

Preset State: 6 dB.  
Step Increment: by 1 dB.  
Related Commands: MKPK.

### Example

```

10 OUTPUT 718;"IP;CF 300MHZ;SP 1GHZ;"
20 INPUT "ENTER IN PEAK EXCURSION, IN DB ",Excursion
30 OUTPUT 718;"MKPX ";Excursion;"DB;"
40 OUTPUT 718;"TS;MKPK HI;MKPK NH;"
50 OUTPUT 718;"MKD?;"
60 ENTER 718;Freq
70 IF Freq <> 0 THEN
80 PRINT "PEAK FOUND"
90 ELSE

```

*Initializes analyzer, changes start and stop frequencies.*

*Changes peak excursion level.*

*Searchs for highest peaks of trace.*

*Finds frequency difference between peaks.*

*Puts the analyzer response in the computer variable, Freq.*

*Outputs results if marker amplitude was not 0.*

*Prints "NO PEAKS FOUND" if Freq = 0.*

**MKPX Marker Peak Excursion**

```
100 PRINT "NO PEAKS FOUND"  
110 END IF  
120 END
```

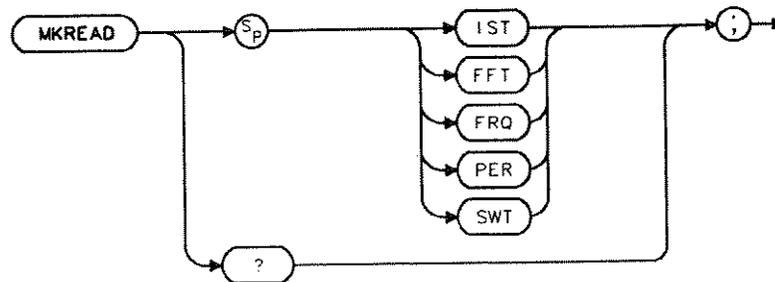
The functions of the MKPX command and **PEAK EXCURSN** are identical (see "MKPK" and "PEAKS").

**Query Response**

## MKREAD Marker Readout

The MKREAD command selects the type of active trace information displayed by the analyzer marker readout.

### Syntax



Related Commands: FFT, MKF, MKTYPE.

### Example

```
OUTPUT 718;"MKREAD FFT;"
```

### Description

The MKREAD command can select the following types of active trace information:

**FRQ:** marker frequency.

**PER:** period.

**SWT:** sweep time.

**IST:** inverse sweep time.

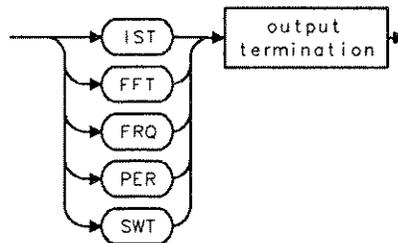
**FFT:** fast Fourier transform readout.

The results of the data depend on the MKREAD parameter, the frequency span, and using the marker delta.

## MKREAD Marker Readout

MKREAD Type	Non-Zero Span	Non-Zero Span Delta	Zero Span	Zero Span Delta
FRQ	Reads frequency	Reads delta frequency	Reads time	Reads delta time
PER	Period of frequency	(Pulse measurement) delta time	N/A	N/A
SWT	Reads time since the start of sweep	Reads delta time between end points	Waveform measurements of detected modulation	Waveform measurements of detected modulation
Ist	N/A	N/A	N/A	Computes frequency corresponding to delta of markers. Performs $1/(T_1 - T_2)$
FFT	N/A	N/A	Reads frequency corresponding the to FFT bucket	Reads delta frequency corresponding to delta FFT bucket

## Query Response



## MKRL Marker to Reference Level

The MKRL command sets the reference level to the amplitude value of the active marker.

### Syntax



Related Commands: MKOFF, RL.

### Example

10 OUTPUT 718;"IP;SNGLS;CF 300MHZ;SP 20MHZ;"	<i>Initialize analyzer, change center frequency and span.</i>
20 OUTPUT 718;"TS;MKPK HI;MKRL;TS;"	<i>Place marker on trace peak, set the reference level to the amplitude of the active marker, update the sweep.</i>
30 OUTPUT 718;"RL?;"	<i>Get the reference level.</i>
40 ENTER 718 USING "K";Ref_level	<i>Put the analyzer response in the computer variable, Ref_level.</i>
50 OUTPUT 718;"AUNITS?;"	<i>Get the current amplitude units.</i>
60 ENTER 718;Aunits\$	
50 PRINT "REFERENCE LEVEL IS",Ref_level,Aunits\$	
60 END	

### Description

The functions of MKRL and `MARKER -> REF LVL` are identical.

## MKSP Marker to Span

The MKSP command sets the start and stop frequencies to the values of the delta markers.

### Syntax



Related Commands: MKD, SP.

### Example

10 OUTPUT 718;"IP;"	<i>Initialize analyzer.</i>
20 OUTPUT 718;"MKMIN;"	<i>Place a marker at the minimum amplitude of trace.</i>
30 OUTPUT 718;"MKD;"	<i>Activate marker delta.</i>
40 OUTPUT 718;"MKPK HI;"	<i>Place marker at highest amplitude of trace.</i>
50 OUTPUT 718;"MKSP;"	<i>Change span to the values of the left and right markers.</i>
60 END	

### Description

The left marker specifies start frequency, and the right marker specifies stop frequency. If MKD is off, no operation is performed.

The functions of MKSP and `MKR -> SPAN` are identical.

## MKSS Marker to Step Size

The MKSS command sets the center-frequency step-size to the marker frequency.

### Syntax



Related Commands: CF, MKA, MKCF, MKD, MKF, SS.

### Example

This example measures a harmonic of the CAL OUT signal.

```
10 DISP "CONNECT THE CAL OUT TO THE INPUT"
20 OUTPUT 718;"IP;SNGLS;CF 300MHZ;SP 20MHZ;TS;"

30 OUTPUT 718;"MKPK HI;MKSS;MKD;CF UP;TS;MKPK HI;"

40 OUTPUT 718;"MKA?;"

50 ENTER 718;Delta_amp

60 OUTPUT 718;"MKF?;"

70 ENTER 718;Delta_freq

80 PRINT "DIFFERENCE IN AMPLITUDE IS ",Delta_amp,"dB"
90 PRINT "DIFFERENCE IN FREQUENCY IS ",Delta_freq,"Hz"
100 END
```

*Initialize analyzer, activate single-sweep mode, change center frequency and span, update trace. Place the marker on the highest point of the trace, change the step size to the marker frequency, activate marker delta, increase center frequency, update trace, place marker at highest point of the trace.*

*Get the amplitude of the marker.*

*Put the analyzer response in the computer variable, Delta\_amp.*

*Get the frequency of the marker.*

*Put the analyzer response in the computer variable, Delta\_freq.*

**MKSS Marker to Step Size****Description**

Sets the center-frequency step-size equal to the marker frequency. If in the delta mode, the step size is set to the delta frequency (absolute value).

The functions of the MKSS command and **MARKER -> CF STEP** are identical.

## MKSTOP Marker Stop

The MKSTOP command stops the sweep at the active marker.

### Syntax



Related Commands: MKCONT.

### Example

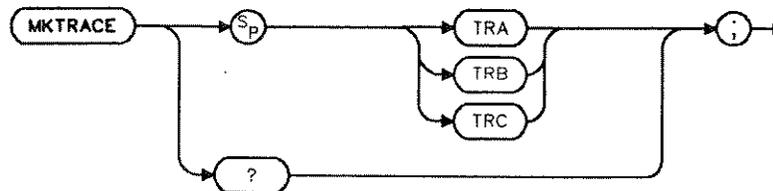
```
10 OUTPUT 718;"IP;TS;"  
20 OUTPUT 718;"MKPK HI;"  
30 OUTPUT 718;"MKSTOP;"  
40 OUTPUT 718;"MKCONT;"  
50 END
```

*Initialize analyzer.  
Create an active marker.  
Stop sweep at marker.  
Resume sweep.*

## MKTRACE Marker Trace

The MKTRACE command moves the active marker to a corresponding position in trace A, trace B, or trace C.

### Syntax



### Example

```

10 OUTPUT 718;"IP;"
20 OUTPUT 718;"MKMIN;"
30 OUTPUT 718;"MKTRACE TRB;"

40 OUTPUT 718;"BLANK TRA;CLRW TRB;"
50 END
  
```

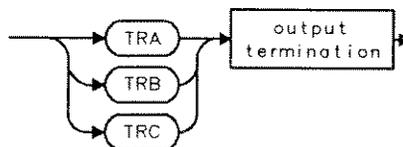
*Initialize analyzer.*

*Find the lowest amplitude of trace.*

*Move marker to corresponding position on trace B.*

*Blank trace A and view trace B.*

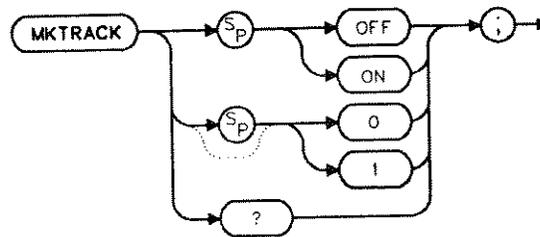
### Query Response



## MKTRACK Marker Track

The MKTRACK command moves the signal with an active marker to the center of the analyzer display and fixes the signal peak there.

### Syntax



Related Commands: MKA, MKCF, MKF.

### Example

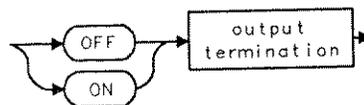
OUTPUT 718;"IP;"	<i>Initialize analyzer.</i>
OUTPUT 718;"CF 300MHZ;TS;"	<i>Change the center frequency.</i>
OUTPUT 718;"MKTRACK ON;"	<i>Activate the marker track.</i>
OUTPUT 718;"SP 10MHZ;TS;"	<i>Change the span.</i>
OUTPUT 718;"MKTRACK OFF;"	<i>Turn off the marker track.</i>

### Description

To keep a drifting signal at center screen, place the active marker on the desired signal before executing MKTRACK.

The functions of the MKTRACK command and the **SIGNAL TRACK** key are identical.

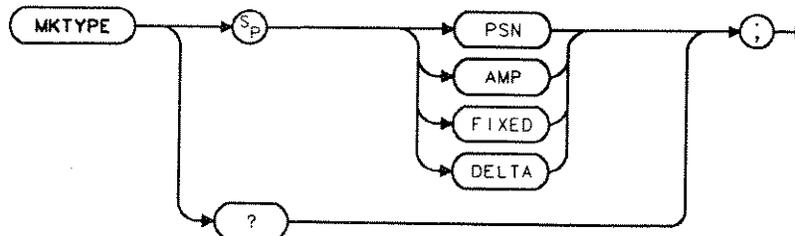
### Query Response



## MKTYPE Marker Type

The MKTYPE command changes the type of the current active marker.

### Syntax



Preset State: MKTYPE PSN.  
Related Commands: MKA, MKBW.

### Example

OUTPUT 718;"MKTYPE AMP;MKA -5;" *The marker is positioned at -5 dBm.*

### Description

The type choices are the following:

**PSN** allows markers to be positioned according to the horizontal position on the display. This is the setting after an instrument preset.

**AMP** allows markers to be positioned according to amplitude, as shown in the example. If two or more points on the trace are at the same amplitude, the marker is moved to the closest point on the trace with the correct amplitude. If no point on the trace is at the specified amplitude, the marker is placed at the specified amplitude and not on the trace.

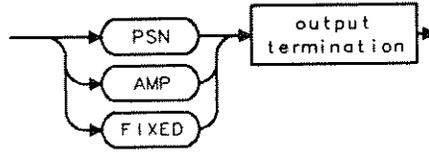
**FIXED** allows a marker to be placed at any fixed point on the analyzer screen. The position of the marker cannot be changed unless another marker type is used.

**DELTA** allows the marker frequency to be positioned with respect to another marker.

Use "MKTYPE PSN" to return from using the AMP, FIXED, or DELTA types.

The functions of MKTYPE AMP and **MARKER AMPTD** are identical.

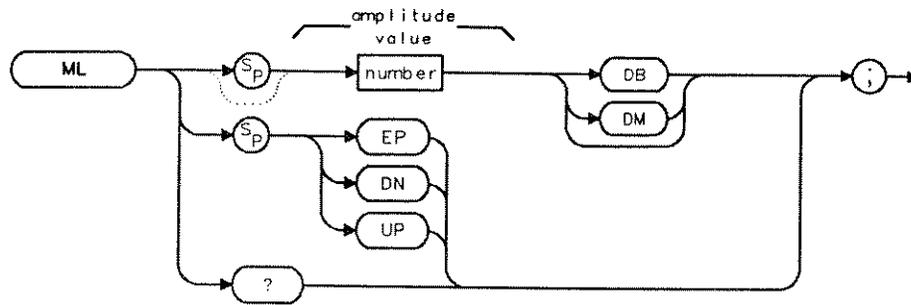
### Query Response



## ML Mixer Level

The ML command specifies the maximum signal level that is applied to the input mixer for a signal that is equal to or below the reference level.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dBm.	-10 to -60 dBm.

Preset State: -10 dBm.

Step Increment: by 10 dBm.

Related Commands: AT, ROFFSET.

### Example

```
OUTPUT 718;"ML -40DM;"
```

*As the reference level is changed, the coupled input attenuator automatically changes to limit the maximum signal at the mixer input to -40 dBm for signals less than or equal to the reference level.*

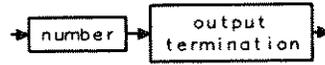
### Description

The ML command specifies the maximum signal level that is applied to the input mixer for a signal that is equal to or below the reference level.

The effective mixer level is equal to the reference level minus the input attenuator setting. When ML is activated, the effective mixer level can be set from -10 dBm to -60 dBm in 10 dB steps.

The functions of the ML command and **MAX MXR LEVEL** are identical.

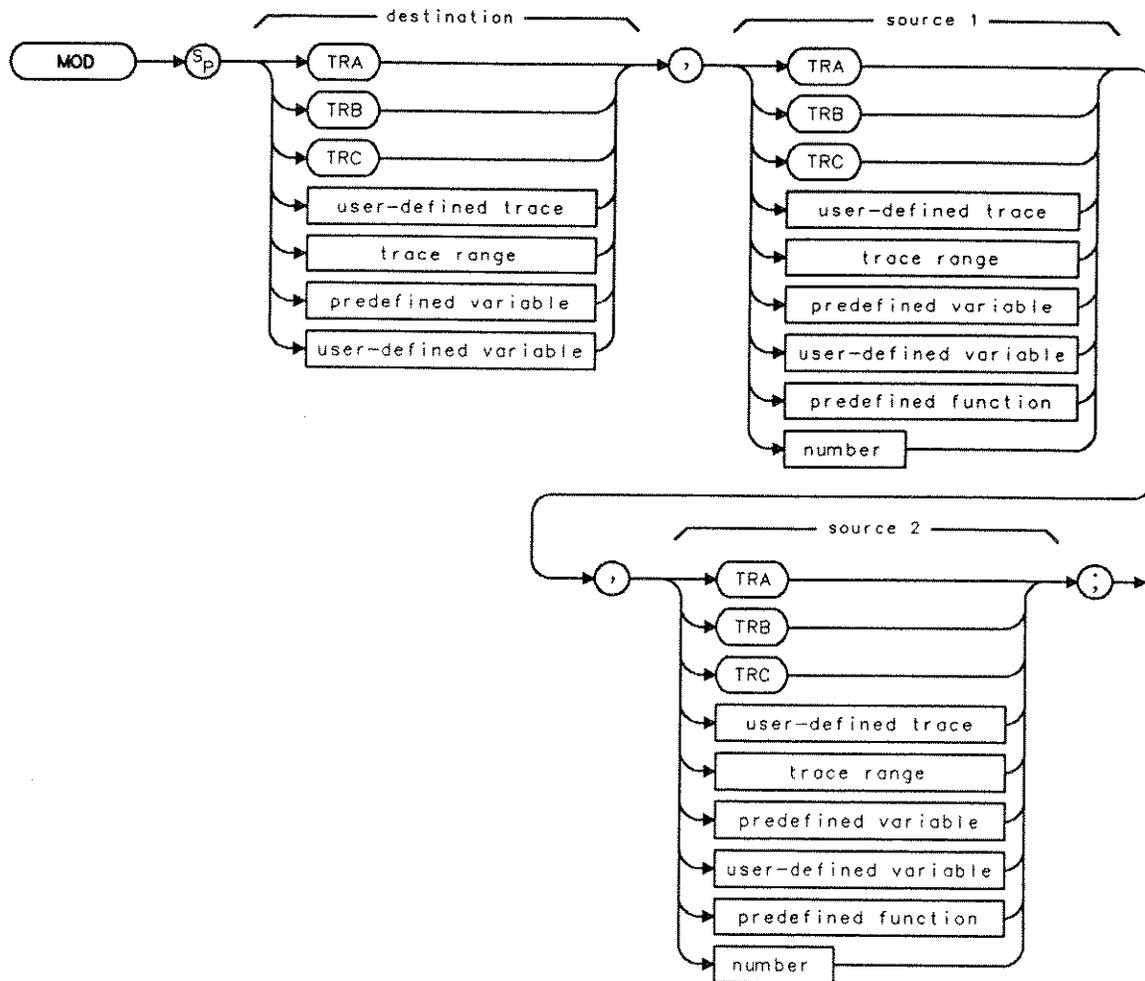
## Query Response



## MOD Modulo

The MOD command stores the remainder from the division of source 1 by source 2 in the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.  
Related Commands: DIV.

### Example

10 OUTPUT 718;"VARDEF S_ONE,15;"	<i>Places 15 into S_ONE.</i>
20 OUTPUT 718;"VARDEF S_TWO,4;"	<i>Places 4 into S_TWO.</i>
30 OUTPUT 718;"VARDEF D_EST,0;"	<i>D_EST holds the result.</i>
40 OUTPUT 718;"MOD D_EST,S_ONE,S_TWO;"	
50 OUTPUT 718;"D_EST?;"	<i>Moves the result to the computer.</i>
60 ENTER 718;Number	<i>Puts the analyzer response in the computer variable, Number.</i>
70 DISP Number	<i>Displays a 3.</i>
80 END	

### Description

If source 1 is a negative number and source 2 is a positive number, the MOD function returns a negative remainder. If both sources are negative, the MOD function returns a negative remainder.

Integer values are used when a trace is the destination or one of the sources. If trace data is used as the source and destination, the MOD function is done with 32-bit arithmetic on 16-bit integer data. If a user-defined variable or predefined variable is used as the source or destination, the MOD function is done in floating point format. If a real number is used as a source, but the destination is an integer value, the result is truncated. If traces are used, be sure each contains a complete sweep of measurement information before executing MOD.

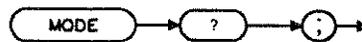
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MODE

### Mode

The **MODE** command returns a "0" if the mode of operation is spectrum analysis. A number other than "0" is returned if the operating mode is other than spectrum analysis.

### Syntax



### Example

```
OUTPUT 718;"MODE?;"
```

### Description

All analyzers have the spectrum analyzer mode of operation. If a program (also called a downloadable program or personality) has been loaded into the analyzer's memory, and the analyzer is using the personality mode of operation, the number that is returned by **MODE** may be modified.

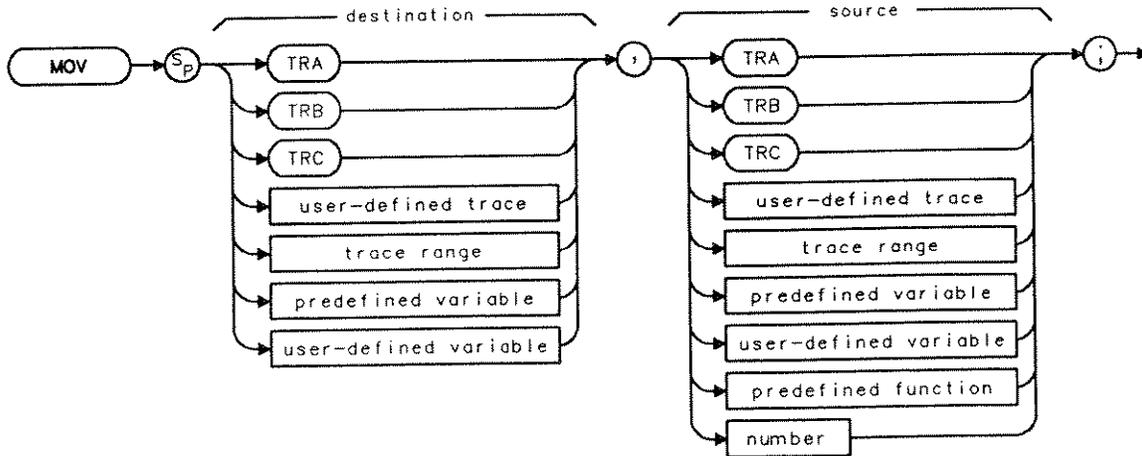
### Query Response



## MOV Move

The MOV command copies the source values into the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

**MOV Move****Example**

10 CLEAR 718	<i>Clears the HP-IB bus and analyzer.</i>
20 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
30 OUTPUT 718;"FA 100MHZ;FB 1100MHZ;"	<i>Sets up measurement range.</i>
40 OUTPUT 718;"TS;MKPK HI;"	<i>Places a marker at the highest peak.</i>
50 OUTPUT 718;"DL ON;"	<i>Turns on display line.</i>
60 OUTPUT 718;"MOV DL,MKA;"	<i>Sets 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.</i>
70 END	

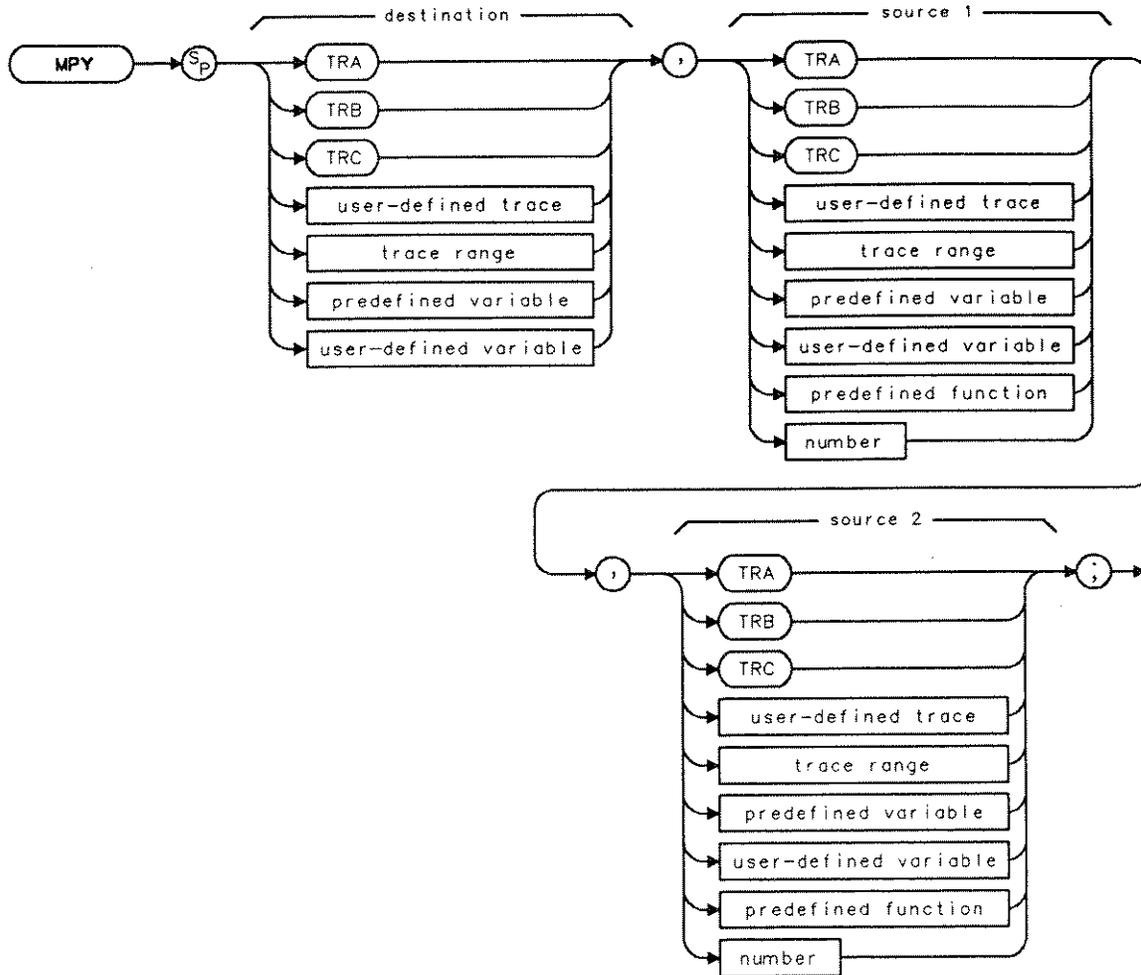
**Description**

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MPY Multiply

The MPY command multiplies the sources, point by point, and places the result(s) in the destination.

### Syntax



**MPY Multiply**

Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.

**Example**

OUTPUT 718; "MPY CF,CF,2;" *Doubles the center frequency.*

**Description**

Traces, user-defined traces, and trace ranges are multiplied as 16-bit integers. Negative numbers are represented in two's complement format. Single variables and numbers are treated as floating point numbers and must be within the real number range as defined in Table 4-1.

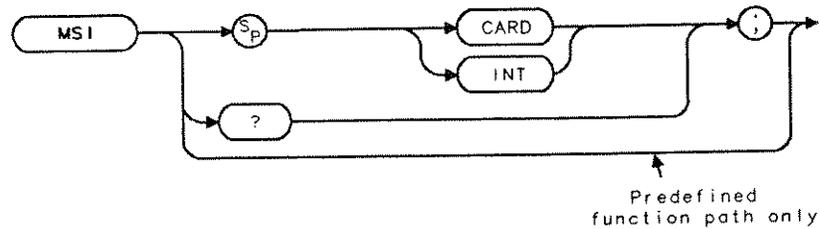
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MSI Mass Storage Is

The MSI command allows you to specify the current mass storage device as analyzer memory or a memory card.

**Option Required:** Option 003 with HP 8590B or HP 8592B

### Syntax



Related Commands: CAT, SAVRCLN.

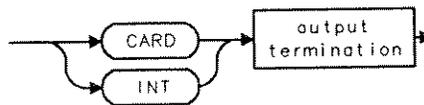
### Example

OUTPUT 718;"MSI CARD;" *Selects the memory card as the current mass storage device.*

### Description

If MSI is used as a predefined function, it returns a "0" if the mass storage device is the analyzer memory and a "1" if it is the memory card.

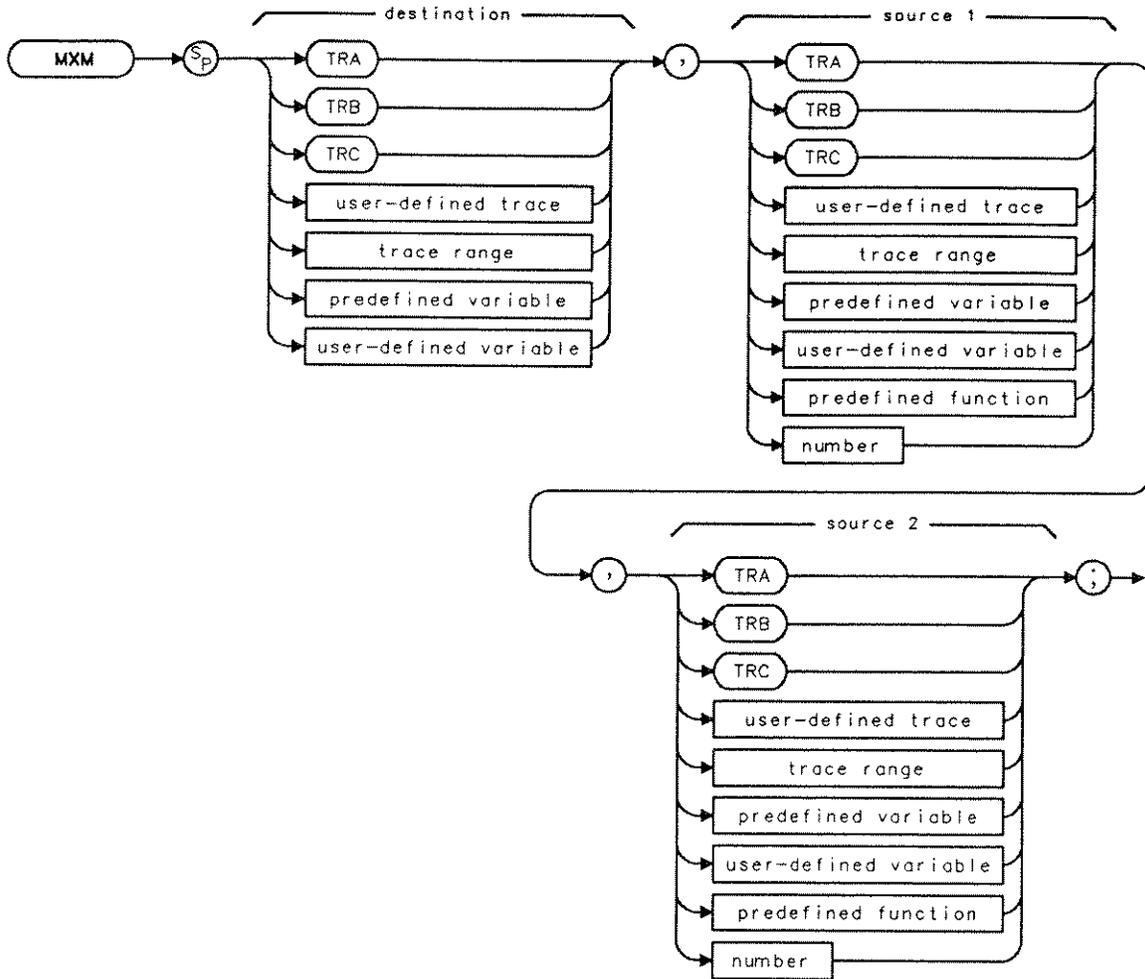
### Query Response



# MXM Maximum

The MXM command compares source 1 and source 2, point by point, sending the greater value of each comparison to the destination.

## Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF when using a user-defined variable. TS when using trace data.  
 Related Commands: MIN, PKPOS, TS.

### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"CF 300MHZ;SNGLS;" Changes the center frequency and activate single-sweep mode.
30 OUTPUT 718;"TS;VIEW TRA;" Updates trace. Displays and stores the results of trace A.
40 OUTPUT 718;"MXM TRB,TRA,4000;" Moves elements of trace A that exceed 4000 trace data points (above center screen) to trace B.
50 OUTPUT 718;"BLANK TRA;VIEW TRB;" Displays result.
60 END

```

### Description

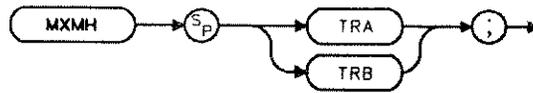
If one of the sources is a single value, it acts as a threshold; all values equal to or greater than the threshold pass to the destination.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## MXMH Maximum Hold

The MXMH command updates each trace element with the maximum level detected.

### Syntax



Related Commands: BLANK, CLRW, MINH, VAVG, VIEW.

### Example

```
OUTPUT 718;"MXMH TRA;"
```

### Description

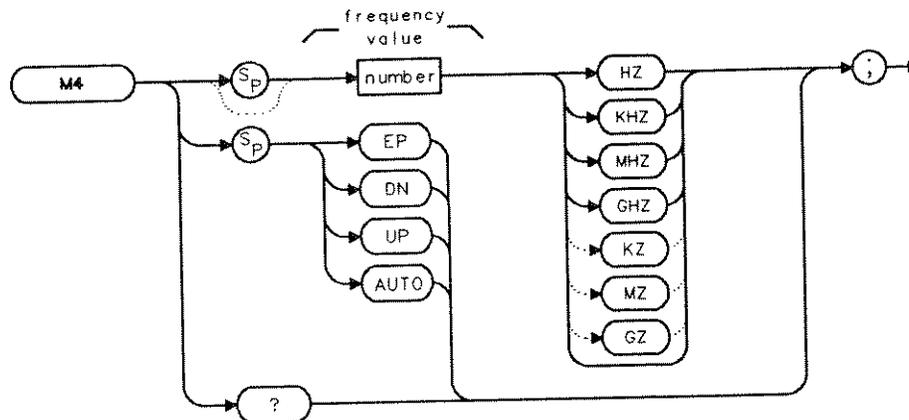
MXMH updates the trace with a new value from a detector only if the new value is larger than the previous trace data value.

The functions of MXMH TRA and MXMH TRB are identical to **MAX HOLD A**, and **MAX HOLD B** respectively.

## M4 Marker Zoom

The M4 command activates a single marker on the trace and enables the knob to change the position of the marker. When changing the frequency span, M4 keeps the marker at center frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	Start frequency to stop frequency of analyzer.

Step Decrement:  $\text{Span}/2$ .  
 Step Increment:  $\text{Span} \times 2$ .  
 Related Commands: MKCF, MKN, SP.

### Example

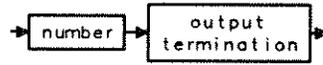
OUTPUT 718;"M4;"

### Description

Used with a marker on the signal peak or at a position of interest, M4 keeps the marker at center screen while the frequency span is decreased.

M4 Marker Zoom

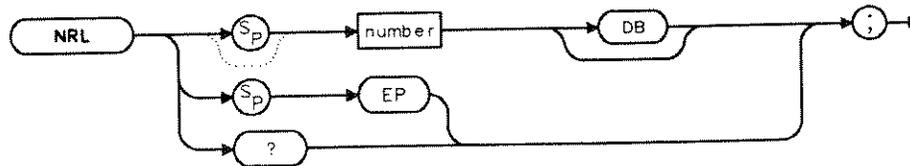
Query Response



## NRL Normalized Reference Level

The NRL command sets the normalized reference level.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dBm.	

Preset State: 0 dB.

Related Commands: MEASURE, RL, RLPOS.

### Example

<pre> 10  OUTPUT 718;"IP;" 20  OUTPUT 718;"SRCPWR -10DB;" 30  PRINT "CONNECT TRACKING GENERATOR OUTPUT TO INPUT" 40  PRINT "THEN PRESS CONTINUE" 50  PAUSE 60  OUTPUT 718;"MEASURE SR;" 70  OUTPUT 718;"CLRWB TRB;TS;" 80  OUTPUT 718;"BLANK TRB;" 90  PRINT "CONNECT DUT TO ANALYZER" 100 PRINT "THEN PRESS CONTINUE" 110 PAUSE 120 OUTPUT 718;"CLRWB TRA;TS;" 130 OUTPUT 718;"AMBPL ON;" 140 OUTPUT 718;"RLPOS 4;" 150 OUTPUT 718;"NRL 40DB;" 160  END </pre>	<p><i>Initializes analyzer. Sets tracking generator output.</i></p> <p><i>Sets the measurement mode to stimulus-response. Take a measurement sweep of the flatness of the system.</i></p> <p><i>Take a measurement sweep of the DUT in the system. Normalize the response. Changes the location of the reference level to the fourth graticule from the bottom. Offsets trace A so it is on- screen.</i></p>
---	--

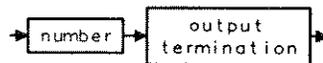
## NRL Normalized Reference Level

### Description

This function is a trace-offset function enabling the user to offset the displayed trace without introducing hardware-switching errors into the stimulus-response measurement. The input attenuator and IF step gains are not affected when using NRL.

In absolute power mode (dBm), reference level affects the gain and RF attenuation settings of the instrument, which affects the measurement or dynamic range. In normalized mode (relative power or dB-measurement mode), NRL offsets the trace data on-screen and does not affect the instrument gain or attenuation settings. This allows the displayed normalized trace to be moved without decreasing the measurement accuracy due to changes in gain or RF attenuation. If the measurement range must be changed to bring trace data on-screen, then the range level should be adjusted. Adjusting the range-level normalized mode has the same effect on the instrument settings as does reference level in absolute power mode (normalize off).

### Query Response



## OA Output Active Function Value

The OA command sends the value of the active function to the controller.

### Syntax



### Example

10 OUTPUT 718;"ST 3SC;OA;"	<i>Changes the sweep time, activates the "OA" command.</i>
20 ENTER 718;Number	<i>Gets the response from the analyzer.</i>
30 DISP Number	<i>The number 3 is displayed on the computer screen.</i>
40 END	

## OL

### Output Learn String

The OL command transmits information to the controller that describes the state of the analyzer when the OL command is executed.

#### Syntax



Related Commands: RCLS, SAVES.

#### Example

```
10 DIM Learn_String$[202]
20 OUTPUT 718;"OL;"
30 ENTER 718 USING "#,202A"; LEARN_STRING$
40 DISP "CHANGE ANALYZER TO ANOTHER STATE,
    THEN CONTINUE TO RESTORE FIRST STATE;"
50 PAUSE
60 OUTPUT 718; LEARN_STRING$
70 END
```

*Allocates memory space for the Learn\_String.*

*Asks for first state.*

*Computer receives the information.*

*Returns the analyzer to its former state (when OL was first activated in line 20).*

#### Description

The information received from the analyzer is called the learn string. The learn string can be sent from the controller back to the analyzer to restore the analyzer to its original state.

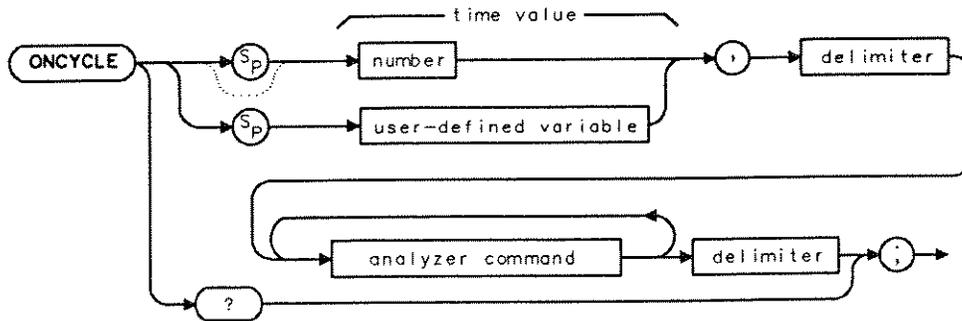
The learn string requires 202 bytes of storage space as is shown in Chapter 3, "Advanced Programming" (see "Saving States" in Chapter 3).

To restore the analyzer to the learn string state, you simply return the learn string to the analyzer. The learn string contains information to tell the analyzer that it is learn string data. (Also see "RCLS" and "SAVES".)

## ONCYCLE On Cycle

The ONCYCLE command periodically executes the command string.

### Syntax



Item	Description/Default	Range
Number	A valid number.	0 to 2,147,483 seconds.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!"\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	

Related Commands: DISPOSE, ERASE, IP, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, ONTIME.

### Example

OUTPUT 718;"ONCYCLE 86400,!MKPK HI;!;" *Places a marker on the highest signal peak every 24 hours (86,400 equals 24 hours × 60 minutes × 60 seconds).*

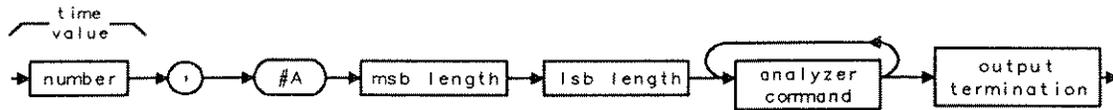
### Description

The ONCYCLE command performs the analyzer command list periodically; the ONDELAY command performs the analyzer command list once after the elapsed time interval. The first execution of the analyzer command(s) does not occur until the time value has elapsed.

Instrument preset (IP) clears ONCYCLE functions.

**ONCYCLE On Cycle****Note**

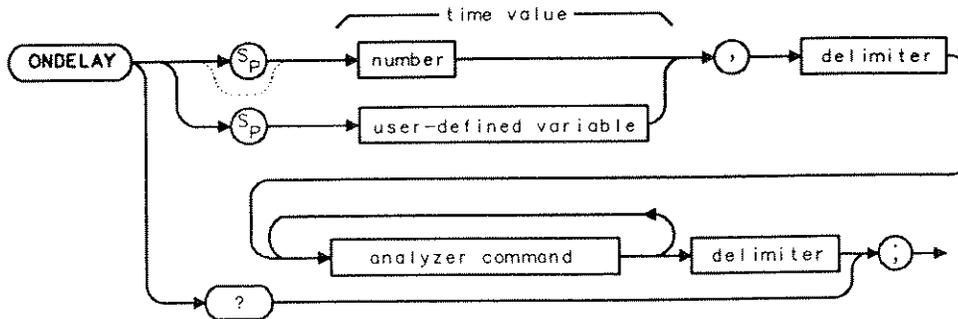
The state of the analyzer is not restored to its original state after the ONCYCLE function is finished. If the list of analyzer commands performed by ONCYCLE changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONCYCLE function, not its pre-ONCYCLE state (for example, continuous-sweep mode).

**Query Response**

## ONDELAY On Delay

The ONDELAY command executes the command string after the time value has elapsed.

### Syntax



Item	Description/Default	Range
Number	A valid number.	0 to 2,147,483 seconds
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!"\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	

Related Commands: DISPOSE, ERASE, IP, ONCYCLE, ONEOS, ONMKR, ONSRQ, ONSWP, ONTIME.

### Example

OUTPUT 718;"ONDELAY 000030,!CF 1.2GHZ;!;" *Changes the center frequency after 30 seconds.*

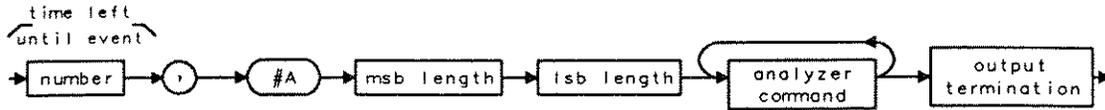
### Description

The ONDELAY command performs the list of analyzer commands once after the elapsed time interval; the ONCYCLE command performs the list of analyzer commands periodically.

Instrument preset (IP) clears ONDELAY functions.

**ONDELAY On Delay****Note**

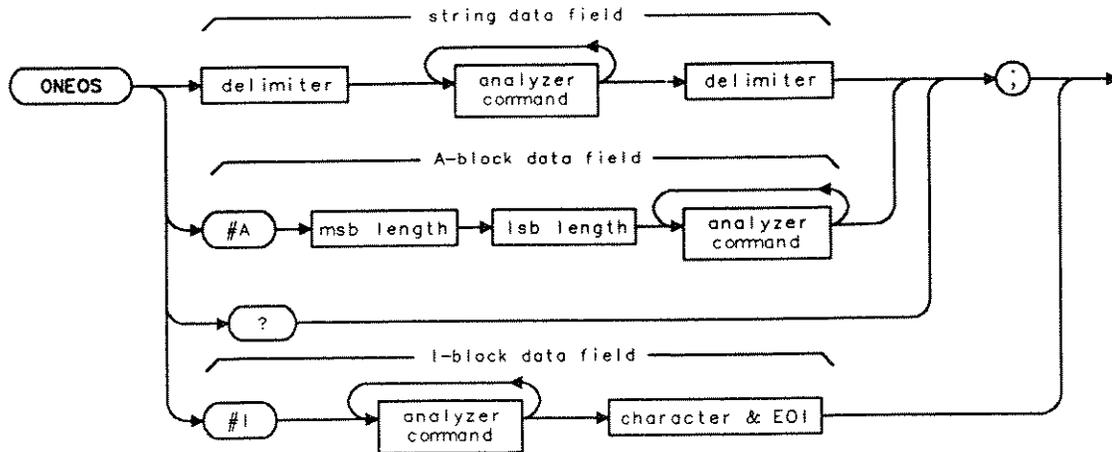
The state of the analyzer is not restored to its original state after the ONDELAY function is finished. If the list of analyzer commands performed by ONDELAY changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONDELAY function, not its pre-ONDELAY state (for example, continuous-sweep mode).

**Query Response**

## ONEOS On End of Sweep

The ONEOS command executes the contents of the data field after the end of the sweep.

### Syntax



Item	Description/Default	Range
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	
Msb length	Most significant byte of a two-byte word that describes the number of bytes transmitted.	
Lsb length	Least significant byte of a two-byte word that describes the number of bytes transmitted.	
Character & EOI	Any valid character and END.	

Related Commands: DISPOSE, ERASE, IP, ONSWP.

### Example

OUTPUT 718;"ONEOS!CF 100MHZ;!"  
*Center frequency is changed at the end of the sweep.*

## ONEOS On End of Sweep

### Description

Instrument preset (IP) clears ONEOS functions. The command string should not include a take sweep (TS).

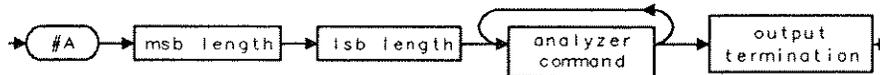
### Note



The state of the analyzer is not restored to its original state after the ONEOS function is finished. If the list of analyzer commands performed by ONEOS changes the state of the analyzer (for example, activating single-sweep mode), the analyzer will still be in the state dictated by the ONEOS function, not its pre-ONEOS state (for example, continuous-sweep mode).

The #, A, msb length, LSB length, and character data form a A-block data field. It is used when the length of the character data is known. The #, I, character data, and EOI (END) form an I-block data field. The I-block data field is used when the length of the character data is unknown. The I-block data field is available for HP-IB interface only.

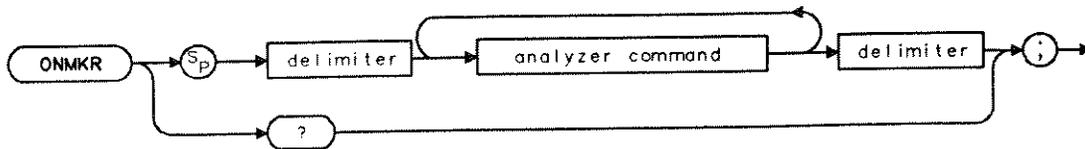
### Query Response



## ONMKR On Marker

The ONMKR command performs the command list when the sweep reaches the marker position.

### Syntax



Item	Description/Default	Range
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!'\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	

Related Commands: DISPOSE, ERASE, IP, ONCYCLE, ONDELAY, ONEOS, ONSRQ, ONSWP, ONTIME.

### Example

```
OUTPUT 718;"ONMKR !PU,PA 100,100;TEXT@CONNECT CAL
OUT TO INPUT@;!;"
```

*The text is displayed on the analyzer screen when the sweep reaches the marker position.*

### Description

The ONMKR command performs the list of analyzer commands when the sweep reaches the marker. The sweep resumes after the list of analyzer commands is executed, provided the list of analyzer commands does not halt execution.

Instrument preset (IP) clears ONMKR functions.

### Note



The state of the analyzer is not restored to its original state after the ONMKR function is finished. If the list of analyzer commands performed by ONMKR changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONMKR function, not its pre-ONMKR state (for example, continuous-sweep mode).

ONMKR On Marker

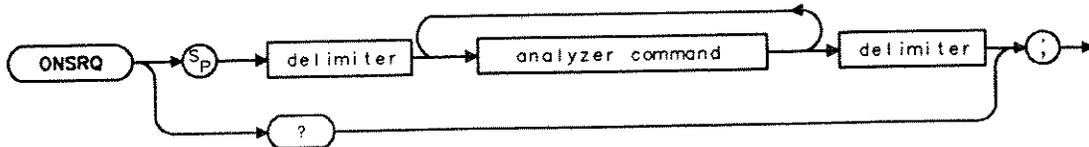
Query Response



## ONSRQ On Service Request

The ONSRQ command executes the command string whenever a service request occurs.

### Syntax



Item	Description/Default	Range
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!"\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	

Related Commands: DISPOSE, ERASE, IP, ONDELAY, ONEOS, ONMKR, ONSWP, ONTIME, SRQ.

### Example

OUTPUT 718;"ONSRQ !PU;PA 100,100;TEXT @SRQ OCCURRED@;!;" *SRQ OCCURRED is displayed on the analyzer screen if an SRQ is encountered.*

### Description

Instrument preset (IP) clears ONSRQ functions.

#### Note



The state of the analyzer is not restored to its original state after the ONSRQ function is finished. If the list of analyzer commands performed by ONSRQ changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONSRQ function, not its pre-ONSRQ state (for example, continuous-sweep mode).

**ONSRQ On Service Request**

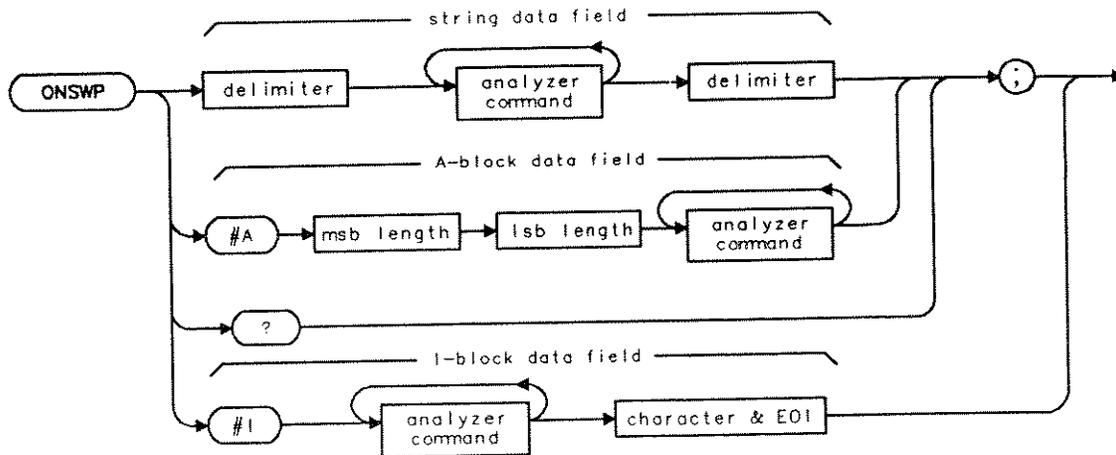
**Query Response**



## ONSWP On Sweep

The ONSWP command executes the command list at the beginning of the sweep.

### Syntax



Item	Description/Default	Range
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	!"\$%&'/:=@
Analyzer command	Any spectrum analyzer command.	
Msb length	Most significant byte of a two-byte word that describes the number of bytes transmitted.	
Lsb length	Least significant byte of a two-byte word that describes the number of bytes transmitted.	
Character & EOI	Any valid character and END.	

Related Commands: DISPOSE, ERASE, IP, ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP.

### Example

OUTPUT 718;"ONSWP!CF 100MHZ;!"  
*The center frequency is changed at the beginning of a sweep.*

## ONSWP On Sweep

### Description

Instrument preset (IP) clears ONSWP functions. The command string should not include a take sweep (TS).

### Note



The state of the analyzer is not restored to its original state after the ONSWP function is finished. If the list of analyzer commands performed by ONSWP changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONSWP function, not its pre-ONSWP state (for example, continuous-sweep mode).

The #, A, msb length, lsb length, and character data form an A-block data field. It is used when the length of the character data is known. To use the A-block data format, #A must precede the msb length and lsb length. The msb length and lsb length represent the length of the character data. The #, I, character data, and EOI (END) form an I-block data field. The I-block data field is used when the length of the character data is unknown. The I-block data field is available for HP-IB interfaces only.

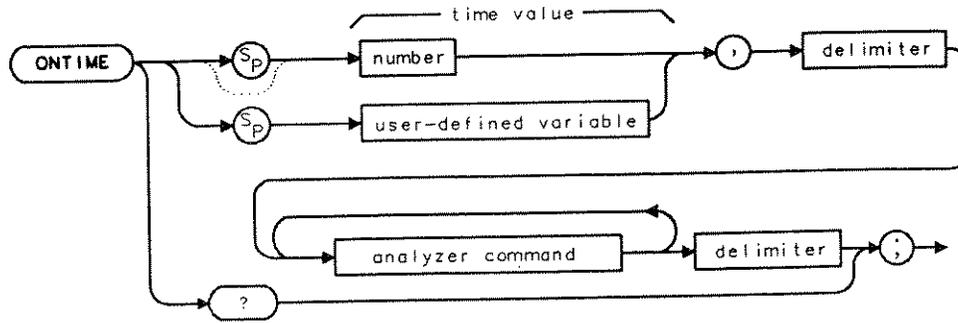
### Query Response



## ONTIME On Time

The ONTIME command executes the command list at the specified time.

### Syntax



Item	Description/Default	Range
Number	A valid number in the YYMMDDHHMMSS (24 hour) format.	A valid year, month, day, and time.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Delimiter	Matching characters marking the beginning and end of the list of analyzer command(s).	! "\$ % & ' / : = @
Analyzer command	Any spectrum analyzer command.	

Related Commands: DISPOSE, ERASE, IP, ONDELAY, ONEOS, ONMKR, ONSWP, ONSRQ.

### Example

```
OUTPUT 718;"ONTIME 890212080000,!CF 600MHZ;!;"
```

*Changes the center frequency on February 12, 1989, at 8 AM.*

```
OUTPUT 718;"ONTIME 080000,!CF 600MHZ;!;"
```

*If the YYMMDD is omitted from the time value parameter, the command list is executed at the next occurrence of the time value given.*

```
OUTPUT 718;"ONTIME 890212150000,!CF 600MHZ;!;"
```

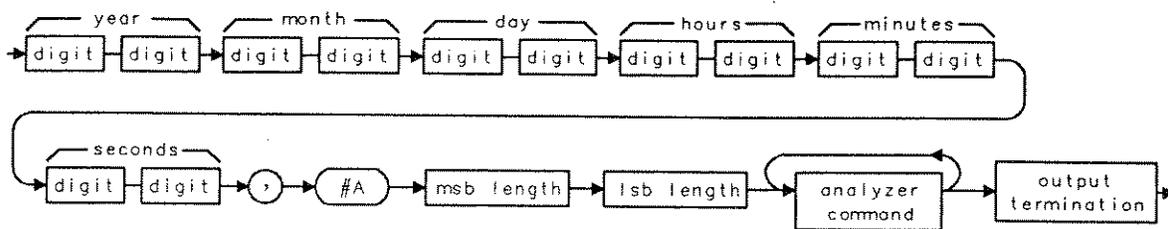
*Changes the center frequency on February 12, 1989, at 3 PM.*

**ONTIME On Time****Description**

Instrument preset (IP) clears ONTIME functions.

**Note**

The state of the analyzer is not restored to its original state after the ONTIME function is finished. If the list of analyzer commands performed by ONTIME changes the state of the analyzer (for example, activating the single-sweep mode), the analyzer will still be in the state dictated by the ONTIME function, not its pre-ONTIME state (for example, continuous sweep mode).

**Query Response**

## OP Output Parameter

The OP command returns parameter values P1 and P2, which represent the  $x$  and  $y$  coordinates of the lower-left and upper-right analyzer display.

### Syntax



### Example

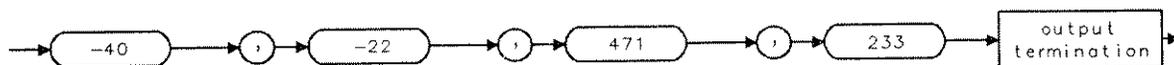
10 DIM A\$[20]	<i>Allocates memory space for result.</i>
30 OUTPUT 718;"OP?;"	<i>Gets the lower-left and the upper-right coordinates of the analyzer display.</i>
40 ENTER 718;A\$	<i>Moves result to the computer.</i>
50 DISP A\$	<i>Displays the result.</i>
60 END	

### Description

The values returned represent  $x$  and  $y$  screen coordinates of the analyzer display. The values returned are Xmin, Ymin, Xmax, and Ymax in plotter units.

The screen coordinates designate the total on-screen area. The lower-left coordinate corresponds to (Xmin, Ymin) and the upper-right coordinate corresponds to (Xmax, Ymax).

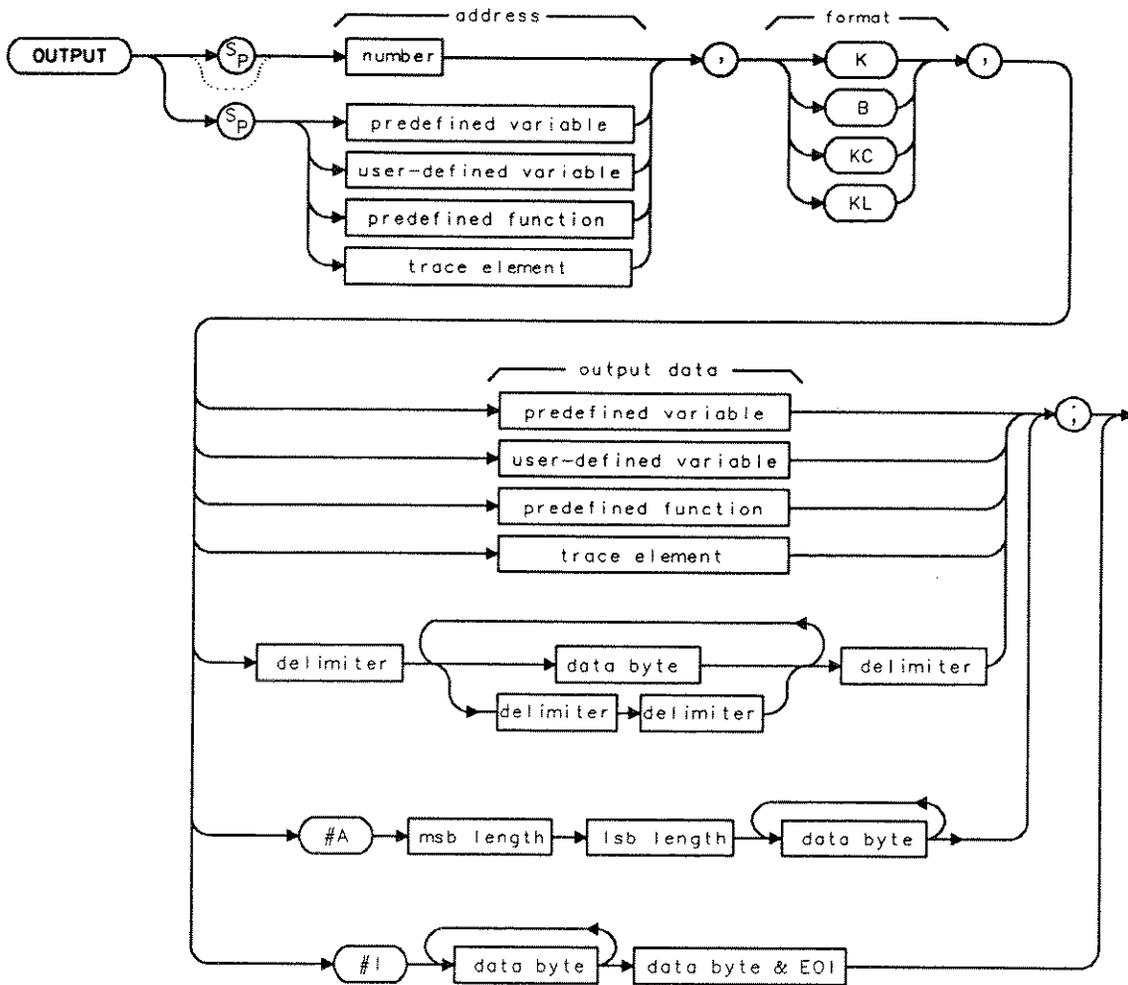
### Query Response



## OUTPUT Output to HP-IB

The OUTPUT command establishes the analyzer as a controller on the HP-IB.

### Syntax



Item	Description/Default	Range
Number	An integer number.	0 to 30.
Predefined function	Function that returns a value. Refer to Table 4-1.	
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Msb length	Most significant byte of a two-byte word that describes the number of bytes transmitted.	
Lsb length	Least significant byte of a two-byte word that describes the number of bytes transmitted.	
Data byte	8-bit byte containing numeric or character data.	
Data byte & EOI	8-bit byte containing numeric or character data followed by END.	

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.  
 Related Commands: ENTER, RELHPIB.

### Example

#### Note



The plotter is at address 5 and the analyzer is at address 18. (The program is only valid for HP 9000 Series 200 and 300 computers.)

This example gets the P1, P2 values from a plotter and displays the coordinates on the analyzer display. Disconnect the computer before executing softkey 1.

10 OUTPUT 718;"FUNCDEF P_OP,!";	<i>Defines a function called P_OP.</i>
20 OUTPUT 718;"VARDEF P_ONEX,1,VARDEF P_ONEY,1;";	<i>Declares the variables used to hold the plotter coordinates.</i>
30 OUTPUT 718;"VARDEF P_TWOX,1;VARDEF P_TWYOY,1;";	<i>Declares the variables used to hold the plotter coordinates.</i>
40 OUTPUT 718;"OUTPUT 5,B,79;";	<i>Sends ASCII number for "O".</i>
50 OUTPUT 718;"OUTPUT 5,B,80;";	<i>Sends ASCII number for "P".</i>
60 OUTPUT 718;"OUTPUT 5,B,59;";	<i>Sends ASCII number for ",".</i>
70 OUTPUT 718;"ENTER 5,K,P_ONEX;";	<i>Gets plotter coordinates from plotter.</i>
80 OUTPUT 718;"ENTER 5,K,P_ONEY;";	<i>Gets Y coordinate from plotter.</i>
90 OUTPUT 718;"ENTER 5,K,P_TWOX;";	<i>Gets X coordinate from plotter.</i>
100 OUTPUT 718;"ENTER 5,K,P_TWYOY;";	<i>Gets Y coordinate from plotter.</i>
110 OUTPUT 718;"PU;PA 200,190;DSPLY P_ONEX,10.2;";	<i>Displays coordinate on analyzer screen.</i>
120 OUTPUT 718;"PU;PA 200,180;DSPLY P_ONEY,10.2;";	<i>Displays coordinate on analyzer screen.</i>
130 OUTPUT 718;"PU;PA 200,170;DSPLY P_TWOX,10.2;";	<i>Displays coordinate on analyzer screen.</i>

**OUTPUT Output to HP-IB**

140 OUTPUT 718;"PU;PA 200,160;DSPLY P_TWOY,10.2;"	<i>Displays coordinate on analyzer screen.</i>
150 OUTPUT 718;"RELHPIB;"	<i>Releases analyzer control of the HP-IB.</i>
160 OUTPUT 718;"!;"	<i>Marks the end of the function, P_OP.</i>
170 OUTPUT 718;"KEYDEF 1,!P_OP;!;!DSP OP;!;"	<i>Puts P_OP function in a soft-key under the user menu.</i>
180 END	

**Description**

Use OUTPUT to send data or instructions to an HP-IB device using the following output formats.

- K** Outputs in free-field ASCII format with no terminator.
- B** Outputs in a free-field format with no terminator, but in a single 8-bit byte.
- KC** Outputs in free-field ASCII with "CR" and "LF" terminator.
- KL** Outputs in free-field ASCII with "LF" and "END" terminator.

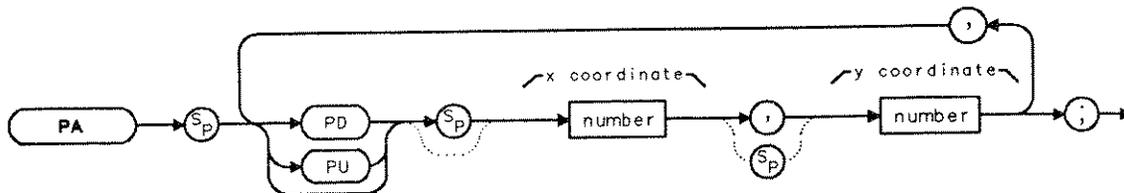
Since HP-IB allows only one controller on the HP-IB, OUTPUT must be synchronized with a controller operation or else incorporated into user-defined functions that are executed with softkeys when the analyzer is under manual control. If another controller is detected on the HP-IB, the OUTPUT function is aborted.

Execute RELHPIB (RELEASE HP-IB) to discontinue analyzer control of HP-IB.

## PA Plot Absolute

The PA command moves the pen to a vector location on the analyzer screen relative to reference coordinates (0,0) in display units.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	Within screen or graticule coordinates.

Related Commands: CLRDSP, DSPLY, ERASE, TEXT, PD, PLOT, PR, PRINT, PU.

### Example

OUTPUT 718;"IP;BLANK TRA;"	<i>Initializes the analyzer and blanks trace A.</i>
OUTPUT 718;"ANNOT OFF;GRAT OFF;"	<i>Clears the analyzer screen.</i>
OUTPUT 718;"PU;"	<i>Pen up.</i>
OUTPUT 718;"PA 100,100;PD 100,150;"	<i>PU and PA commands prevent an initial vector from being drawn before the pen is positioned at (100, 100). PD draws a vector to (100, 150).</i>
OUTPUT 718;"150,150,150,100,100,100;"	<i>Draws the last three sides of the rectangle.</i>

### Description

The vector is drawn on the screen if the pen-down (PD) command is in effect. If the pen-up (PU) command is in effect, the vector does not appear on the screen.

The drawing area within the graticule area is restricted by the graticule coordinates. The screen coordinates designate the total on-screen area. The lower-left coordinate corresponds to (Xmin, Ymin) and the upper-right coordinate corresponds to (Xmax, Ymax). (See "OP.")

Screen Coordinates:

(Xmin, Ymin) = (-40, -22)

(Xmax, Ymax) = (471, 233)

**PA Plot Absolute**

Graticule Coordinates:

(Xmin,Ymin) = (0,0)

(Xmax, Ymax) = (400, 200)

Since PA is an active function, executing PA causes the active function area on the analyzer screen to blank. To prevent the text following PA from being written in the active function area, execute hold (HD) after PA.

PA should be executed before executing TEXT, PD, or DSPLY commands.

## PD Pen Down

The PD command instructs the analyzer to plot vectors on the analyzer screen until a PU command is received.

### Syntax



Related Commands: DSPLY, PA, PLOT, PR, PU, TEXT.

### Example

OUTPUT 718;"IP;BLANK TRA;"	<i>Initializes the analyzer and blanks trace A.</i>
OUTPUT 718;"ANNOT OFF;GRAT OFF;"	<i>Clears the analyzer screen.</i>
OUTPUT 718;"PU;"	<i>Pen up.</i>
OUTPUT 718;"PA 100,100;PD 100,150;"	<i>PU and PA commands prevent an initial vector from being drawn before the pen is positioned at (100, 100). PD draws a vector to (100, 150).</i>
OUTPUT 718;"150,150,150,100,100,100;"	<i>Draws the last three sides of the rectangle.</i>

### Description

The command PD is used to enable drawing of all vectors specified by the commands PA (plot absolute), or PR (plot relative). It remains in effect until a PU command is received. PD does not need to be executed before using the TEXT or DSPLY commands.

The drawing area within the graticule area is restricted by the graticule coordinates. The screen coordinates designate the total on-screen area. The lower-left coordinate corresponds to (Xmin, Ymin) and the upper-right coordinate corresponds to (Xmax, Ymax). (See "OP.")

Screen Coordinates:

(Xmin, Ymin) = (-40, -22)  
(Xmax, Ymax) = (471, 233)

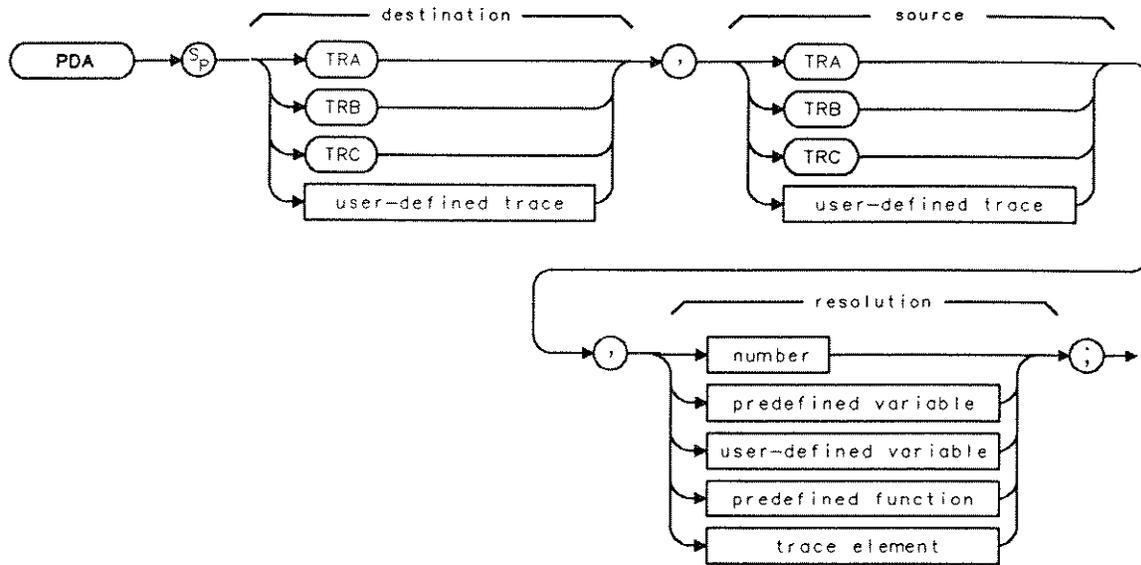
Graticule Coordinates:

(Xmin, Ymin) = (0,0)  
(Xmax, Ymax) = (400, 200)

## PDA Probability Distribution of Amplitude

The PDA command sums the probability distribution of amplitude in the destination trace with the amplitude distribution function of the source trace.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Number	Any real or integer number.	Real number range.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.

Related Commands: PDF, RMS, STDEV.

**Example**

• OUTPUT 718;"IP;SNGLS;"	<i>Initializes analyzer and activates the single-sweep mode.</i>
OUTPUT 718;"VB 10KHZ;HD;TS;"	<i>Changes video bandwidth, updates trace.</i>
OUTPUT 718;"MOV TRB,0;"	<i>Replaces trace B data with all zeros.</i>
OUTPUT 718;"PDA TRB,TRA,1;"	<i>Determines the distribution of trace A and sums results into trace B.</i>
OUTPUT 718;"MPY TRB,TRB,5;"	<i>Multiplies values in trace B by 5 to make the results more visible.</i>
OUTPUT 718;"VIEW TRB;"	<i>Displays the result.</i>

**Description**

The PDA command takes the data in the source trace on a point-by-point basis. Each amplitude value is divided by 100 times resolution value, and the result of the division is rounded to an integer. If the result falls within the range of the buckets of the destination trace, the content of the corresponding destination trace element is increased by one. For example, to show the distribution of amplitudes on a trace with values ranging from 0 to 8000, a resolution value of 1 dB would result in 81 buckets ( $((8000/(1 \times 100)) + 1)$ ). Amplitude values ranging from 0 to 99 would go to bucket 1, values from 100 to 199 would go into bucket 2, and so forth. Finally, values from 7900 to 7999 would go to bucket 400. An amplitude value of 8000 would fall into bucket 81.

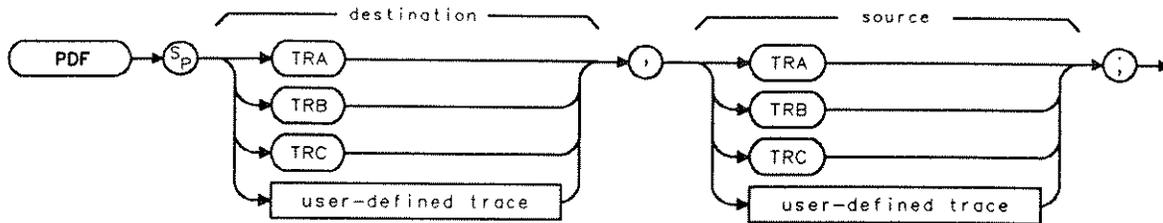
Due to the summing nature of the PDA command, the destination trace should always be initialized to all zeros.

The PDA function is similar to the probability density function in statistics. The probability density function has the  $y$ -axis as the probability of an occurrence, where the PDA function of the 8590 Series has the number of occurrences as its  $y$ -axis. The PDA could be converted to a probability density function by dividing, in an external controller, the value of each bucket by the total number of source elements. Note that performing the divide inside the analyzer would not be appropriate since the result is less than 1, which would be truncated to 0.

## PDF Probability Distribution of Frequency

The PDF command increments an element of the destination trace whenever the corresponding element of the source trace exceeds a threshold. This is useful for constructing a frequency probability density function.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: PDA, TH.

### Example

This example finds the portions of the frequency band where no signals above  $-50$  dBm are observed in an hour time frame.

```

10 OUTPUT 718;"IP;";
20 OUTPUT 718;"SP 100MZ;CF 300MZ;";
30 OUTPUT 718;"TH -50 DM;TS;";

40 OUTPUT 718;"VIEW TRB;CLR TRA;MOV TRB,0;";
50 OUTPUT 718;"ST?;";
60 ENTER 718;Sweep_time
70 Swp_retrace = Sweep_time+.1

80 Num_sweeps = 3600/Swp_retrace

90 FOR I = 1 TO Num_sweeps

```

*Initializes analyzer.*  
*Changes the span and center frequency.*  
*Activates the threshold level, take a sweep.*  
*Sets trace B to zeros.*  
*Gets the sweep time.*  
*Returns the sweep time to the controller.*  
*Calculates the total sweep time, including the retrace time.*  
*Calculates the number of sweeps in one hour.*  
*This finds the maximum number of sweeps that can be taken before numerical overflow (greater than 32,767.)*

## PDF Probability Distribution of Frequency

```
100 OUTPUT 718;"TS;PDF TRB,TRA;"
```

*When  $I = \text{Num\_sweeps}$ , trace B contains the number of sweep that had amplitudes at or above the threshold level of  $-50$  dBm.*

```
110 NEXT I  
120 END
```

**Description**

The TH command permits the user to set an amplitude threshold value. When PDF is performed, measurement buckets of the source trace that exceed the threshold increment the corresponding frequency bucket in the destination trace.

The destination trace should be set to zeros before PDF is executed for the first time. Subsequent calls to PDF increment the destination trace.

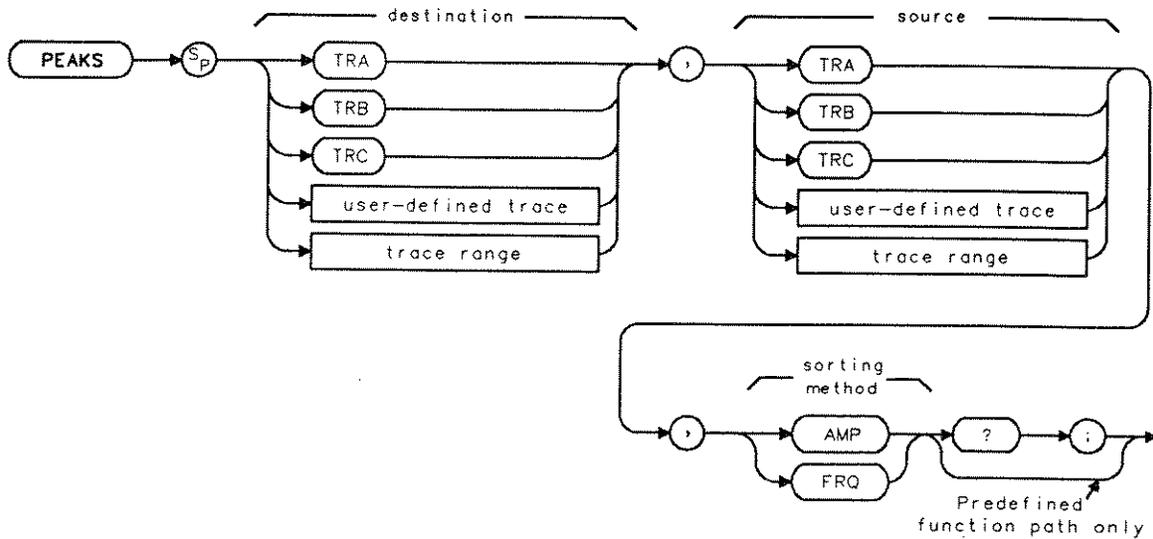
When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination. The PDF function is similar to the probability density function in statistics. The probability density function has the  $y$ -axis as the probability of an occurrence, where the PDF function of the 8590 Series has the number of occurrences as its  $y$ -axis. The PDF could be converted to a probability density function by dividing, in an external controller, the value on each bucket by the total number of buckets. Note that performing the division inside the analyzer would not be appropriate since the result is less than 1, which would be truncated to 0.

# PEAKS

## Peaks

The PEAKS command sorts signal peaks by frequency or amplitude, stores the results in the destination trace, and returns the number of peaks found.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: MKPX, TH.

### Example

OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
OUTPUT 718;"CF 300MHZ;SP 1500MHZ;RB 30KHZ;SNGLS;"	<i>Changes the center frequency, span, bandwidth. Activates single-sweep mode.</i>
OUTPUT 718;"TH -60DM;MKPX 10DB;TS;"	<i>Sets up threshold, sets minimum peak excursion.</i>
OUTPUT 718;"PEAKS TRB,TRA,FRQ?;"	<i>Returns the number of peaks in trace A above the threshold.</i>

ENTER 718;Number

DISP Number

*Gets the number of peaks from the analyzer.*

*Displays the result on the computer screen.*

## Description

When sorting by frequency (FRQ), PEAKS first computes the horizontal position of all peaks. These positions are consecutively loaded into the destination trace, the lowest frequency value occupying the first element. Thus, signal frequencies, from low to high, determine the amplitude of the destination trace from left to right.

When sorting by amplitude (AMP), PEAKS first computes the amplitudes of all peaks in the source trace in measurement units, and sorts these values from high to low. The positions of the peaks are then loaded into the destination trace, with the position of the highest amplitude value occupying the first element.

PEAKS sorts only signals that are above the threshold value. To change the threshold, use the TH command before PEAKS is executed.

If necessary, the last sorted value is repeated to fill remaining elements of the destination trace.

To be classified as a signal peak, a signal must be MKPX above the threshold value and it must rise and fall at least the peak excursion (MKPX value).

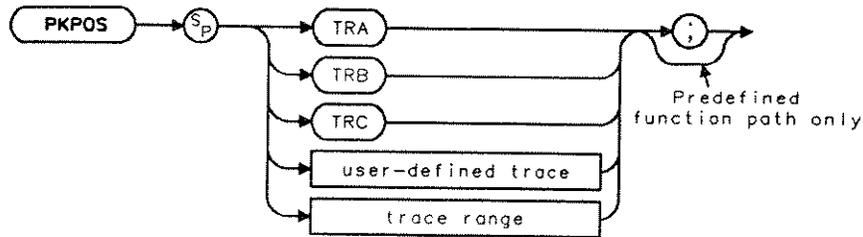
## Query Response



## PKPOS Peak Position

The PKPOS command returns a value which is the index of the maximum value in trace A, trace B, trace C, or user-defined trace.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: MINPOS, MXM.

### Example

```

OUTPUT 718;"IP;"           Initializes analyzer.
OUTPUT 718;"SNGLS;TS;"    Activates single-sweep mode, takes sweep.
OUTPUT 718;"PKPOS TRA;"   Finds the position of the highest peak.
ENTER 718;Pkresult        Outputs result to the computer.
DISP Pkresult             Displays the result.
  
```

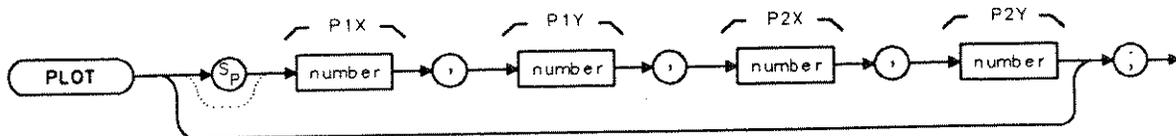
### Description

If a trace range is used with PKPOS, PKPOS returns a value relative to the first element of the trace range. For example, if a trace has a range of 150 to 300 elements, and the maximum value is element 200, PKPOS will return the value of 51.

## PLOT Plot

The PLOT command initiates a plotter output of the screen data to the remote interface. With the appropriate HP-IB commands, the HP-IB can be configured to route the data to an external plotter.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	Number within the plotter coordinates.

Related Commands: PRINT, SNGLS, TS.

### Example for the HP-IB Interface

The plotter is at address 5 and the analyzer is at address 18 in this example. (The program is only valid for HP 9000 series 200 and 300 computers.)

This example illustrates how an external controller can initiate the sending of print data to an external printer.

10 DIM P\$(80)	<i>Allocates room in memory.</i>
20 OUTPUT 705;"OP;"	<i>Plotter outputs lower-left and upper-right display dimensions.</i>
30 ENTER 705;P\$	<i>Puts the plotter response in the computer string.</i>
40 OUTPUT 718;"PLOT";P\$	<i>Plots the analyzer display according to the dimensions stored in the computer string.</i>
50 SEND 7;LISTEN 5 TALK 18 DATA	<i>Configures the interface to output data from analyzer to plotter.</i>
60 END	

**PLOT Plot****Description**

The PLOT command transfers the trace data, graticule, and annotation of the analyzer screen to a plotter via the analyzer interface (softkey labels excluded). The data is transferred in ASCII, HP-GL format.

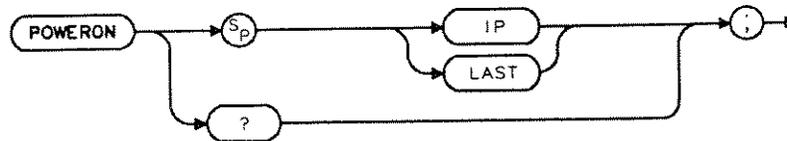
The example routes the data to an external plotter; however, the controller can read the data into a string if desired.

When using the PLOT command, the scaling points (P1x, P1y; P2x, P2y) can be specified. These scaling points specify the (x,y) coordinates which determine the size of the plot. (P1x,P1y) refers to the lower-left plotter coordinates. (P2x,P2y) refers to the upper-right plotter coordinates.

## POWERON Power-On State

The POWERON command selects the state of the analyzer when it is turned on: the IP state (same state as an instrument preset command) or last state (the state the analyzer was in when it was turned off).

### Syntax



### Example

```
OUTPUT 718;"POWERON LAST;"
```

### Description

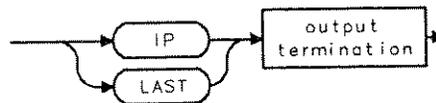
The selected POWERON state can only be changed by the POWERON command.

### Note



The last state of the analyzer is not retained in the case of battery power failure of the analyzer's internal battery.

### Query Response



## PP Preselector Peak

The PP command peaks the preselector.

**Model Required: HP 8592B or HP 8593A**

### Syntax



PROJECT: 8590B/PROG FIG: X-X (FILE: CD2\_M09X) WRITER: LG ILLUS: CE DATE: 12/6/89 FINAL:

Related Commands: MKA, MKCF, MKD, MKF, MKN, MKPK.

### Example

OUTPUT 718;"IP;CF 3GHZ;SP 500KHZ;" *Initializes analyzer, changes center frequency, span.*

OUTPUT 718;"TS;MKPK HI;MKCF;PP;" *Peaks the highest on-screen signal.*

### Description

Set the desired trace to clear-write mode, place a marker on a desired signal, then execute PP. Commands following PP are not executed until after the analyzer has finished peaking the preselector.

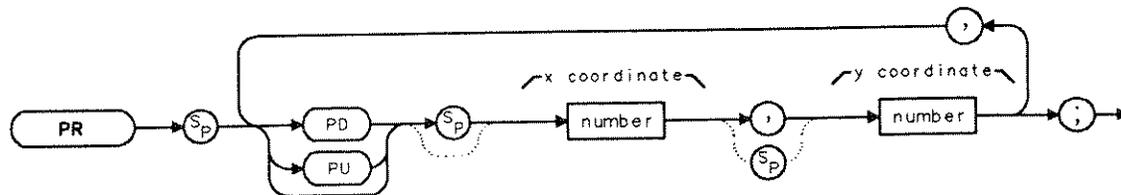
PP automatically adjusts the preselector tracking to peak the signal at the active marker. (When the marker is tuned to a signal, and **PRESEL PEAK** is pressed, an internal routine searches for the peak response of the preselector and adjusts the tracking accordingly.) Using preselector peak prior to measuring a signal yields the most accurate amplitude reading.

Preselector peak operates with the **MARKER NORMAL** or **MARKER DELTA** markers. If the marker is OFF, pressing **PRESEL PEAK** initiates a peak search routine and then peaks the response at that marker; otherwise, it peaks around the active marker. A **CAL:PEAKING** message appears on the active graticule area to indicate operation of the peaking routine. Preselector peak only operates in the 2.75 to 22 GHz preselector bands.

## PR Plot Relative

The PR command moves the pen to a new plot location on the analyzer screen relative to the current coordinates in display units.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	Dependent on the current pen position.

Related Commands: DSPLY, PA, PLOT, PRINT, PU, TEXT.

### Example

```

OUTPUT 718;"IP;"           Initializes analyzer.
OUTPUT 718;"BLANK TRA;ANNOT OFF;"  Clears the display.
OUTPUT 718;"GRAT OFF;"      Turns off graticule.
OUTPUT 718;"PU;PA 0,100;"   Positions pen.
OUTPUT 718;"PD;PR 100,0,0,-100,-100,0,0,100;"  Draws a rectangle.

```

### Description

Vector coordinate sets (x,y pairs) following the PR command can be either positive or negative, depending on the direction of the individual vectors to be drawn. PU (pen up) and PD (pen down) commands tell the analyzer to draw or not draw the vectors on the screen. (See the PU and PD commands.)

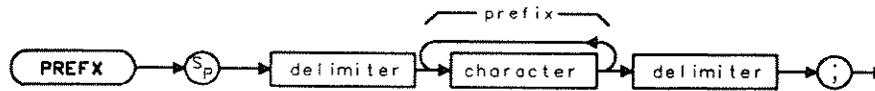
Since PR is an active function, executing PR causes the active function area on the analyzer screen to blank. To prevent the text following PR from being written in the active function area, execute hold (HD) after PR.

## PREFIX

### Prefix

The PREFIX command specifies or changes the prefix used in save and recall operations.

### Syntax



Item	Description/Default	Range
Character	Any valid character.	0 to 6 characters long, A through Z and the underscore (the underscore cannot be the first character of the prefix).
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	! "\$ % & ' / : = @

Related Commands: CAT, SAVRCLN, STOR.

### Example

```
OUTPUT 718;"PREFIX %FRED%;"
```

### Description

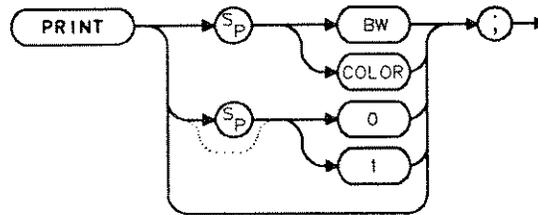
The functions of PREFIX and CHANGE PREFIX are similar.

## PRINT

### Print

The PRINT command initiates a output of the screen data to the remote interface. With appropriate HP-IB commands, the HP-IB can be configured to route the data to an external printer.

### Syntax



Related Commands: PLOT.

### Example for the HP-IB Interface

The printer usually resides at address 1 and the plotter at address 5. (The program is only valid for HP 9000 Series 200 and 300 computers and HP Vectra personal computer with an HP raster graphics printer, such as the HP Thinkjet.)

This example illustrates how an external controller can initiate the sending of print data to an external printer.

```

OUTPUT 718;"PRINT;"
SEND 7;UNT UNL LISTEN 1 TALK 18 DATA  Sends data to printer.
  
```

### Note



Enter the following BASIC commands to allow printing without disconnecting the external controller:

ABORT 7

LOCAL 7

Press **COPY**.

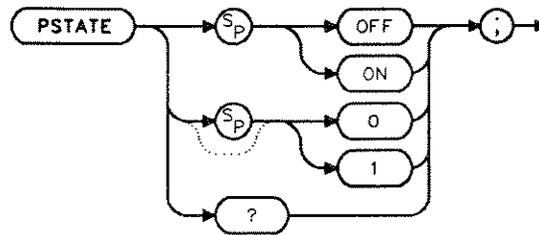
### Description

The data is output in HP raster graphics format. PRINT, PRINT 0, or PRINT BW produces a monochrome printout. PRINT 1 and PRINT COLOR produces a "color format" output for an HP PaintJet printer. Execute "MENU 0;" before printing to blank the softkeys.

## PSTATE Protect State

The PSTATE command protects all internal user state and trace registers from being changed.

### Syntax



Related Commands: ERASE, RCLS, SAVES.

### Example

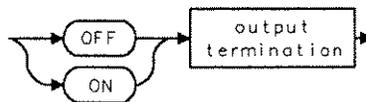
```
OUTPUT 718;"PSTATE ON;"
```

### Description

The PSTATE command is equivalent to **SAV LOCK ON OFF**.

The ERASE command will clear the PSTATE condition and will overwrite the state registers' contents.

### Query Response



## PU Pen Up

The PU command instructs the analyzer not to plot vectors on the analyzer screen until a PD command is received.

### Syntax



Related Commands: DSPLY, PA, PD, PLOT, PR, TEXT.

### Example

OUTPUT 718;"IP;BLANK TRA;"	<i>Initializes analyzer, blanks trace A.</i>
OUTPUT 718;"ANNOT OFF;GRAT OFF;"	<i>Turns off annotation, graticule.</i>
OUTPUT 718;"PU;"	<i>Prevents initial vector from being drawn.</i>
OUTPUT 718;"PA 100,100;PD 100,150;"	<i>Positions pen, pen down.</i>
OUTPUT 718;"150,150,150,100,100,100;"	<i>Draws remaining 3 sides of rectangle.</i>

### Description

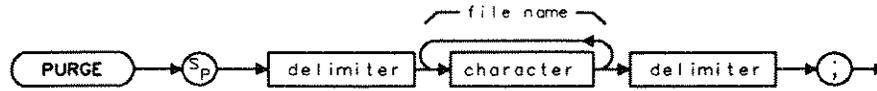
The PU command is used before the commands PA (plot absolute), or PR (plot relative), to suppress drawing while moving to the starting point of a vector. It remains in effect until a PD command is received.

## PURGE

### Purge File

The PURGE command deletes the file from the current mass storage device.

#### Syntax



Item	Description/Default	Range
Delimiter	Matching characters marking the beginning and end of the list of analyzer commands.	!"\$%&'/:=@
Character	Any valid character.	Any valid filename.

Related Commands: MSI.

#### Example

```
OUTPUT 718;"PURGE %dMYFILE%;"
```

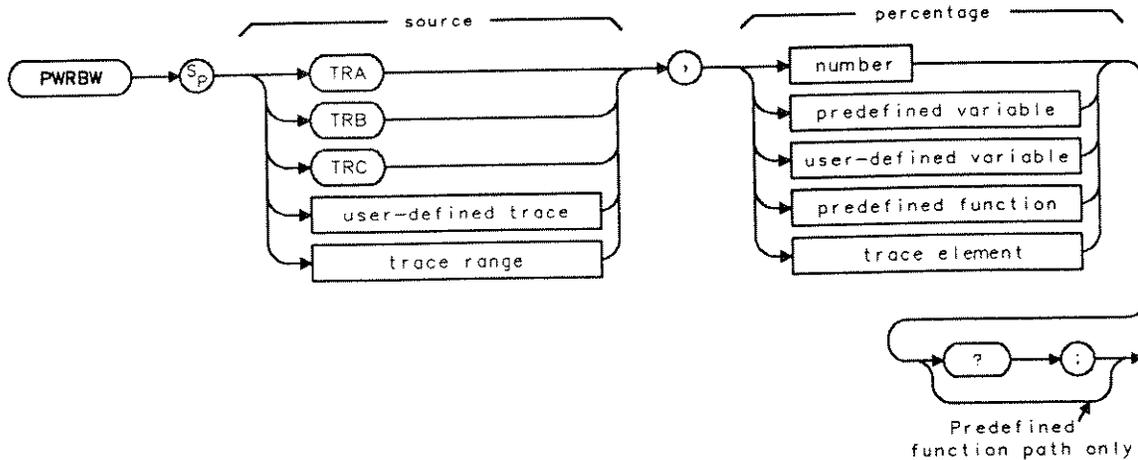
#### Description

The memory card files are repacked automatically after a PURGE command is executed.

## PWRBW Power Bandwidth

The PWRBW command computes the bandwidth around the trace center, which includes signals whose total power is a specified percentage of the total trace signal power.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	0 to 100.

**Parameter Values:** The field used for the percentage must use a value between 0 and 100.  
**Prerequisite Commands:** TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.  
**Related Commands:** RB, SP.

**PWRBW Power Bandwidth****Example**

```
DISP "CONNECT CAL OUT TO INPUT"
```

```
OUTPUT 718;"IP;"
```

```
OUTPUT 718;"SNGLS;"
```

```
OUTPUT 718;"CF 300MHZ;SP 1MHZ;RB 300KHZ;"
```

```
OUTPUT 718;"MXMH TRA;TS;TS;TS;TS;"
```

```
OUTPUT 718;"PWRBW TRA, 99.0;"
```

```
ENTER 718;P
```

```
DISP "THE POWER BANDWIDTH AT 99 PERCENT IS";P/1.0E+3;"kHz"
```

*User prompt.*

*Initializes analyzer.*

*Activates single-sweep mode.*

*Changes center frequency, span, and bandwidth.*

*Activates the maximum hold of trace A, sweep 4 times.*

*Returns the 99% power bandwidth.*

*Gets the result from the analyzer.*

*Displays the frequency of the power bandwidth specified on the computer screen.*

**Description**

If trace A is the source, a delta marker is set at the start and stop frequencies.

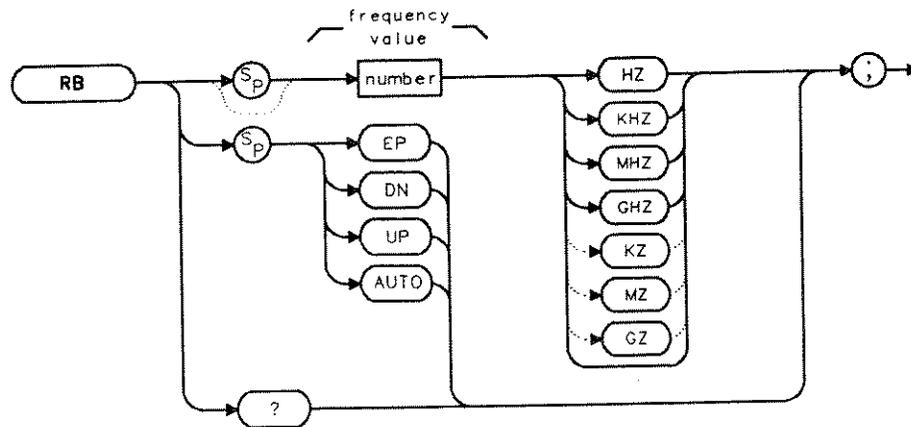
If 100% is specified, the power bandwidth equals the frequency range of the screen display. If 50% is specified, trace elements are eliminated from either end of the array, until the combined power of the remaining signal responses equals half of the original power computed. The frequency span of these remaining trace elements is the power bandwidth returned.

**Query Response**

## RB Resolution Bandwidth

The RB command specifies the resolution bandwidth.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	1 kHz to 3 MHz.

Preset State: 3 MHz.

Step Increment: In a 1, 3, 10.

Related Commands: AUTO, SP, ST, VB, VBR.

### Example

OUTPUT 718;"RB 1KHZ;" Sets the resolution bandwidth to 1 kHz.

### Description

The coupling between video bandwidths or between sweep time and resolution bandwidth is not terminated by this command. Execute RB AUTO to reestablish coupling. (Also see "AUTO.")

The functions of the RB command and **RES BW AUTO MAN** are identical.

**RB Resolution Bandwidth****Note**

The 9 kHz and 120 kHz resolution bandwidths (used for EMI testing) are available by specifying 9 kHz or 120 kHz as the frequency value; the front-panel knob, step increment keys, and auto-coupled settings provide the 1, 3, 10 resolution bandwidth sequence only. Frequencies other than 9 kHz, 120 kHz, 5 MHz or in the 1, 3, 10 sequence are rounded to the nearest permissible value.

**Note**

The analyzer provides uncalibrated bandwidths of 300 Hz and 5 MHz.

**Query Response**

## RCLS Recall State

The RCLS command recalls analyzer state data from one of the nine state registers in analyzer memory.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	1 to 9

Related Commands: LOAD, POWERON, RCLT, SAVES.

### Example

```

OUTPUT 718;"IP;CF 300MHZ;SP 1MHZ;" Changes center frequency, span.
OUTPUT 718;"SAVES 3;" Saves state in register 3.
OUTPUT 718;"IP;"
OUTPUT 718;"RCLS 3;" Recalls the contents of register 3.
  
```

### Description

Registers one through eight are reserved for the user, and contain instrument state information that has been stored with the SAVES command. State register nine contains the previous state data, state register zero contains the current state.

The function of the RCLS command is identical to **INTRNL -> STATE**.

### Note

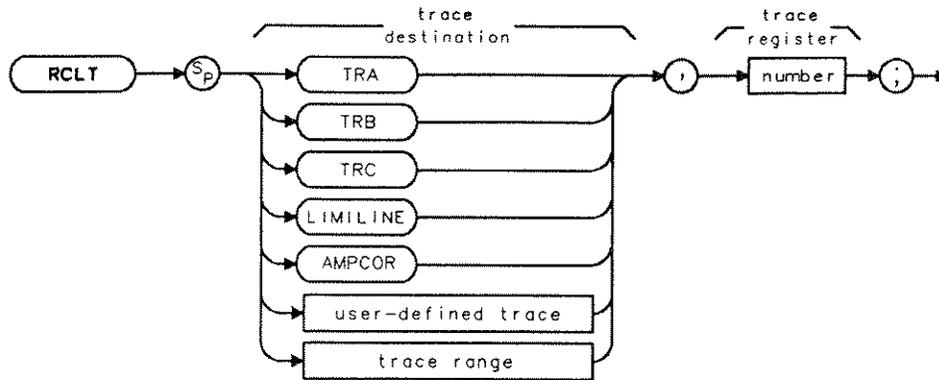


The RCLS recalls state data from analyzer memory. See "LOAD" or "SAVRCLN" to recall state data from the memory card.

## RCLT Recall Trace

The RCLT command recalls previously saved trace data, amplitude factors, or limit-line data from the trace registers in analyzer memory. Trace data is recalled with instrument state, date, and screen title.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	0 to TRCMEM - 1.

Parameter Value: 0 to TRCMEM - 1.

Prerequisite Commands: TRDEF when using a user-defined trace.

Related Commands: CAT, CLRW, LOAD, SAVET, SNGLS, TRCMEM, TS, VIEW.

### Example

```

10 OUTPUT 718;"IP;CF 300MHZ;SP 20MHZ;TS;" Initializes analyzer, changes the center
                                     frequency, span.
20 OUTPUT 718;"SAVET TRA,1;"           Saves analyzer state and trace A data in
                                     register 1.
30 OUTPUT 718;"IP;"                   Initializes analyzer.
40 OUTPUT 718;"RCLT TRA,1;VIEW TRA;"   Recalls analyzer state, trace data; dis-
                                     plays result.
50 END
    
```

## Description

The state and trace data are recalled when the trace destination is trace A, trace B, or trace C. When using a user-defined trace or a trace range for the trace destination, only the trace data is recalled. Use AMPCOR when recalling amplitude correction factors, LIMILINE when recalling limit line tables.

---

### Note



To avoid overwriting the recalled trace data, the VIEW command should be performed immediately after the RCLT command when recalling trace data (see line 40 in the example).

---

The RCLT command and the corresponding VIEW command are equivalent to `INTRNL -> TRACE`.

---

### Note



The RCLT recalls trace data from analyzer memory. See "LOAD" or "SAVRCLN" to recall trace data from the memory card.

---

---

## RELHPIB

### Release HP-IB

The RELHPIB command discontinues analyzer control of HP-IB.

**Option Required:** Option 021

#### Syntax



Related Commands: ENTER, OUTPUT.

#### Example

OUTPUT 718;"RELHPIB;" *The analyzer releases control of HP-IB so that another device can control the bus.*

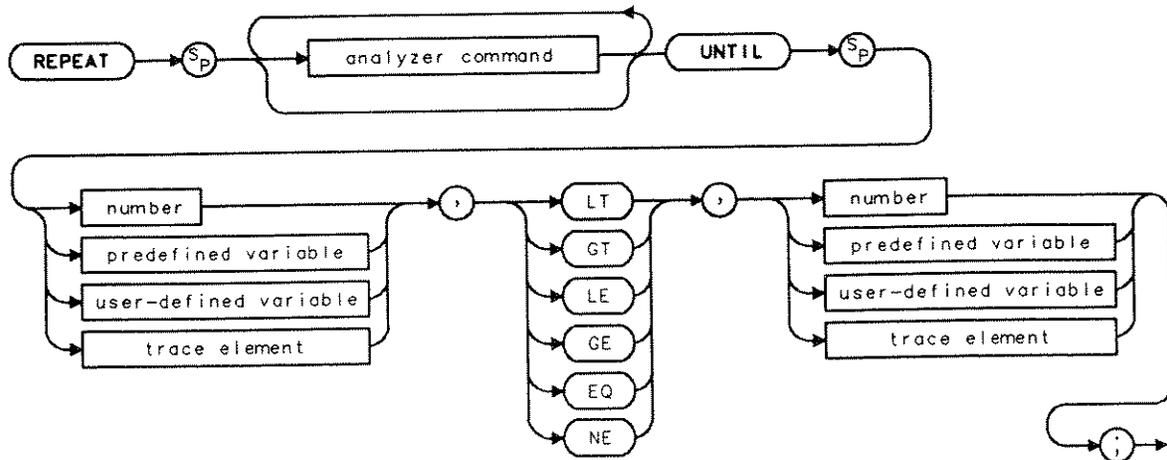
#### Description

The RELHPIB command causes the device that is acting as the controller on the HP-IB (for example, the analyzer) to relinquish control of the bus.

## REPEAT UNTIL Repeat Until

The REPEAT and UNTIL commands form a looping construct.

### Syntax



Item	Description/Default	Range
Analyzer command	Any valid complete analyzer command.	
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Number	Any real or integer number.	Real number range.
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: FUNCDEF when using a user-defined function. ACTDEF or VARDEF when using a user-defined variable.

Related Commands: ABORT, IF (IF/THEN/ELSE/ENDIF).

### Example

The following program lowers any off-screen signal.

```
10 OUTPUT 718;"IP;"
20 OUTPUT 718;"SNGLS;TS;MKPK HI;"

30 OUTPUT 718;"IF MA,GT,RL THEN"
```

*Initializes analyzer.*

*Activates single-sweep mode, takes sweep, places marker on signal peak.*

*Performs lines 40 and 50 if the marker amplitude is greater than the reference level.*

**REPEAT UNTIL Repeat Until**

```
40 OUTPUT 718;"REPEAT RL UP;TS;MKPK HI;" Increases reference level, takes sweep, places  
marker on signal peak.  
50 OUTPUT 718;"UNTIL MA,LE,RL" Does line 40 until peak amplitude is less  
than or equal to the reference level.  
60 OUTPUT 718;"ENDIF CONTS;" Ends the IF THEN construct.  
70 END
```

**Description**

All commands following the REPEAT command are executed until the comparison specified after the UNTIL command is true.

The following can be used to compare condition 1 and condition 2:

**GT** Greater than

**LT** Less than

**LE** Less than or equal to

**GE** Greater than or equal to

**EQ** Equal to

**NE** Not equal to

## RESETRL Reset Reference Level

The RESETRL command resets the reference level to its instrument preset value.

### Syntax



Related Commands: AUNITS, IP, ML, RL.

### Example

```
OUTPUT 718;"RESETRL;"
```

## RETURN

### Return

The RETURN command stops the operation of a user-defined command and returns program operation to the point where the user-defined function was called.

### Syntax



Related Commands: ABORT, FUNCDEF, IF (IF/THEN/ELSE/ENDIF), REPEAT (REPEAT/UNTIL).

### Example

10 OUTPUT 718;"IP;"	<i>Initializes analyzer.</i>
20 OUTPUT 718;"CLRDSB;"	<i>Clears graphics from memory.</i>
30 OUTPUT 718;"TRDSP TRA,OFF;"	<i>Turns off trace A.</i>
40 OUTPUT 718;"ANNOT OFF;GRAT OFF;"	<i>Blanks annotation and graticule.</i>
50 OUTPUT 718;"VARDEF C_OUNT,0;"	<i>Defines variable with the initial value of 0.</i>
60 OUTPUT 718;"FUNCDEF DLP,^;"	<i>Declares a user-defined function.</i>
70 OUTPUT 718;"REPEAT;"	<i>Begins repeat loop.</i>
80 OUTPUT 718;"ADD C_OUNT,C_OUNT,100;"	<i>Add 100 to C_OUNT.</i>
90 OUTPUT 718;"PU;PA 100,100;PD;"	
100 OUTPUT 718;"DSPLY C_OUNT,4.0;"	<i>Writes value of C_OUNT.</i>
110 OUTPUT 718;"IF C_OUNT,EQ,300;THEN;RETURN;ENDIF;"	<i>Executes return when C_OUNT is equal to 300.</i>
120 OUTPUT 718;"UNTIL C_OUNT,EQ,400;"	
130 OUTPUT 718;"^;"	<i>Marks end of DLP.</i>
140 !	<i>Defines second user-defined function called SHELL.</i>
150 OUTPUT 718;"FUNCDEF SHELL,^;"	<i>Reinitializes C_OUNT.</i>
160 OUTPUT 718;"CLRDSB;MOV C_OUNT,0;"	<i>Executes DLP and writes text on screen.</i>
170 OUTPUT 718;"DLP;TEXT@INSIDE SHELL ... @;^;"	<i>This line executes the SHELL program which in turn executes the program called DLP.</i>
180 OUTPUT 718;"SHELL;"	
190 END	

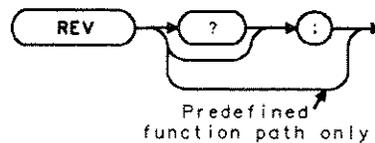
**Description**

The example contains a user-defined function, called DLP, nested within another function, called SHELL. The innermost function, DLP, contains RETURN. When RETURN is encountered, the DLP operation is interrupted, then program operation resumes at the first analyzer command following the function call of DLP in line 170.

## REV Revision

The REV command returns the date code of the firmware revision number in YYMMDD format (for example, 860910 indicates September 10, 1986).

### Syntax



Related Commands: ID, SER, TIMEDATE.

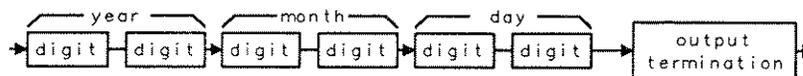
### Example

OUTPUT 718;"REV;" *Gets the firmware revision date of analyzer.*  
 ENTER 718;A *Puts the analyzer response in the computer variable, A.*  
 DISP A *Displays the firmware revision date on the computer screen.*

### Description

The date code of the firmware revision also appears when the instrument is first turned on, but in the day, month, year format.

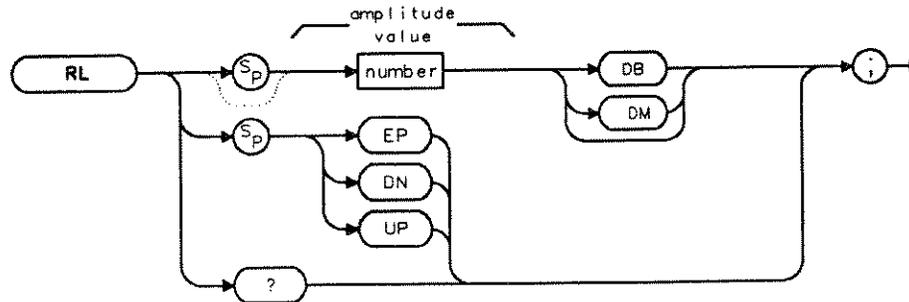
### Query Response



## RL Reference Level

The RL command specifies the amplitude value of the reference level.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is the current amplitude unit.	Amplitude range of the analyzer.

Amplitude Range HP 8590B or HP 8591A: -139.9 to +50 dBm.

Amplitude Range HP 8592B or HP 8593A: -139.9 to +60 dBm.

Preset State: 0 dBm.

Step Increment: by 10 dBm.

Related Commands: AT, MKRL.

### Example

```

10 OUTPUT 718;"IP;SNGLS;CF 300MHZ;SP 20MHZ;" Initializes analyzer, activates single-sweep mode, changes center frequency, span.
20 OUTPUT 718;"TS;MKPK HI;MKRL;TS;" Takes sweep, places marker on signal peak, sets reference level to marker level.
30 OUTPUT 718;"RL?;" Queries reference level.
40 ENTER 718;Ref_level Puts the analyzer response in the computer variable, Ref_level.
50 PRINT "REFERENCE LEVEL IS",Ref_level,"DM"
60 END

```

## RL Reference Level

### Description

The reference level and input attenuator are coupled to prevent gain compression. Signals with peaks at or below the reference level are not affected by gain compression.

---

**Caution** Signal levels above +30 dBm will damage the analyzer.



---

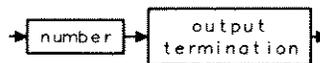
The functions of the RL command and **REF LVL** are identical.

---

**Note** RL may affect the attenuation value.



### Query Response

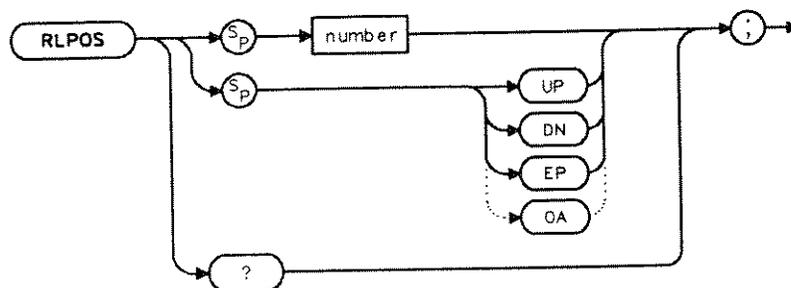


## RLPOS Reference-Level Position

The RLPOS command selects the position of reference level.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	0 to 8.

Step Increment: 1.

Related Commands: AMBPL, IP, MEASURE, ML, NRL, RL.

Preset State: RLPOS 8.

### Example

OUTPUT 718;"IP;"

OUTPUT 718;"MEASURE NRM;"

OUTPUT 718;"AMBPL ON;"

OUTPUT 718;"RLPOS 7;"

*Initializes analyzer.*

*Changes the measurement mode to normalized.*

*Activate trace normalization.*

*Positions 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 8 and 0, respectively. RLPOS must be used with MEASURE NRM or MEASURE SR and AMBPL ON or AMB ON. Arrows appear on the left and right side of the screen graticule when the reference level position is changed.

**RLPOS Reference-Level Position**

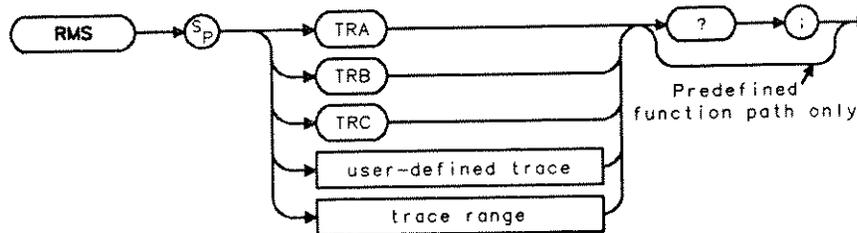
**Query Response**



## RMS Root Mean Square Value

The RMS command returns the root mean square value of the trace in measurement units.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: MEAN, PDA, PDF, STDEV, VARIANCE.

### Example

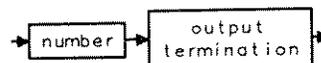
```

OUTPUT 718;"IP;SNGLS;TS;"
OUTPUT 718;"RMS TRA?;"
ENTER 718;Number
DISP Number
  
```

### Description

Trace data, user-defined trace data, and trace range data are treated as 16-bit integers.

### Query Response

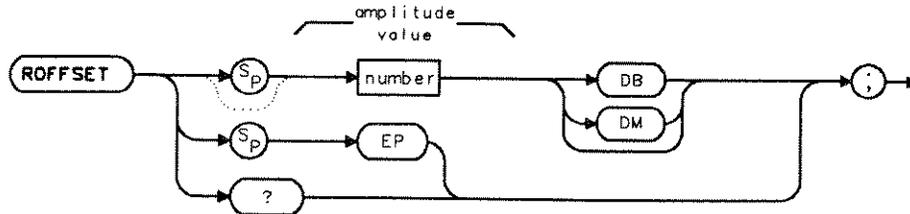


## ROFFSET

### Reference Level Offset

The ROFFSET command offsets all amplitude readouts without affecting the trace.

#### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dBm.	-200 dBm to +200 dBm.

Preset State: 0 dBm.

Related Commands: AT, RL.

#### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"RL -20DB;"     Changes the reference level.
30 OUTPUT 718;"ROFFSET -10;"  Changes analyzer reference offset value.
40 OUTPUT 718;"RL?;"         Gets the reference value from analyzer.
50 ENTER 718;Ref             Puts the analyzer response in the computer variable, Ref.

60 DISP "THE NEW REFERENCE LEVEL IS ",Ref Displays -30 as the new reference level.
70 END
  
```

#### Description

Once activated, the ROFFSET command displays the amplitude offset in the active function block. And, as long as the offset is in effect, the offset is displayed on the left side of the screen.

Entering ROFFSET 0 or presetting the analyzer eliminates an amplitude offset.

The function of the ROFFSET command and `REF LVL OFFSET` are identical.

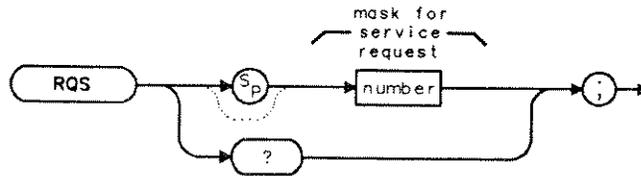
## Query Response



## RQS Service Request Mask

The RQS sets a bit mask for service requests (see "SRQ").

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	0 to 62.

Related Commands: SRQ, STB.

### Example

OUTPUT 718;"RQS 12;" *Sends a mask bit for hardware broken and end of sweep.*

### Description

Assignment of values for the mask is as follows:

- 32 = Illegal command
- 16 = Command complete
- 8 = Hardware broken
- 4 = End of sweep
- 2 = Units key pressed

As shown in the example, a mask with hardware broken and end of sweep is equal to 12 (8 + 4). It also disables command complete and illegal command interrupts.

To activate all conditions in the mask, the mask value is equal to 62 (32 + 16 + 8 + 4 + 2).

Send:

OUTPUT 718;"RQS 62;"

Each bit in the status byte is defined as follows:

1 indicates that the units key was pressed. SRQ 102 appears on the analyzer screen. If you activate the units key bit, it will remain active until you activate "EE" and press a units key. (See "EE.")

**2** indicates end of sweep. SRQ 104 appears on the analyzer screen. If you send any RQS value that contains mask value 4, another sweep will be taken.

**3** indicates broken hardware. SRQ 110 appears on the analyzer screen.

**4** indicates completion of a command. It is triggered by EOI at the end of a command string or the completion of a print or plot.

**5** indicates an illegal analyzer command was used. SRQ 140 appears on the analyzer screen.

**0** (LSB), **6**, and **7** are not used.

The analyzer screen numbers 102, 104, and 110 are the octal values corresponding to the status register values; that is, SRQ 102 = bit 6 = octal 100 and bit 2 = octal 2 are both true.

A service request is generated only if the proper request mask bit has been set, and either the condition itself or the Simulate Service Request (see "SRQ") is sent. To set the request mask, choose the desired interrupt conditions and sum their assigned values. Executing the RQS command with this value sets the bit mask. After setting the bit mask, only the chosen conditions can produce an interrupt. Generally, you must set the bit mask using the RQS command. However, the "hardware broken" and "illegal remote command" conditions are automatically enabled after presetting or sending the IP command. Pressing **PRESET** or sending the IP command, then, produces the same interrupt bit mask as sending "RQS 40;" (decimal 40 is the sum of the assigned values of these two interrupt bits, 32 = Bit 5 and 8 = Bit 3).

For most conditions, the RQS mask bit stays set until the next preset, IP, or RQS command is executed. The only condition to which this does not apply is the Units Key Pressed bit. When this bit (bit 1) is set in the RQS mask, a Units Key Pressed interrupt occurs if EE (enable entry mode) is executed and a front-panel units key such as Hz, kHz, MHz, or GHz is pressed.

When a units key is pressed, the interrupt occurs and the Units Key Pressed bit in the RQS mask is reset. To reenable the Units Key Pressed interrupt, you must send a new RQS mask.

## Query Response

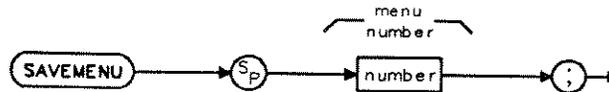


## SAVEMENU

### Save Menu

The SAVEMENU command saves menu 1 under the menu number given.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	1, 101—200.

### Example

```

OUTPUT 718;"MENU 1;"           Displays menu 1.
OUTPUT 718;"SAVEMENU 101;"     Copies the key functions from menu 1 into menu 101.
OUTPUT 718;"KEYCLR;"          Erases the key functions of menu 1.
PAUSE
OUTPUT 718;"MENU 101;"        Displays menu 101.
  
```

### Description

The key number corresponds to the menu number as follows:

key number = (menu number - 1) × 6 + key position  
 (The key position range is 1 through 6.)

For example:

- Menu 1 contains key numbers 1 through 6
- Menu 101 contains key numbers 601 to 606
- Menu 200 contains key numbers 1195 to 1200

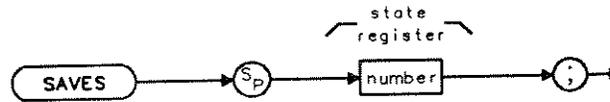
Menu 1 is can be accessed by pressing **MEAS/USER**, **USER MENU(S)**.

Menus 101 through 200 as well as menu 1 can be accessed using the MENU command. For example, executing "MENU 200;" accesses menu 200.

## SAVES Save State

The SAVES command saves the currently displayed instrument state in analyzer memory.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	1 to 8.

Related Commands: OL, PSTATE, RCLS, SAVET, STOR.

### Example

OUTPUT 718;"IP;CF 300MHZ;SP 20MHZ;" *Initializes analyzer, changes center frequency, span.*

OUTPUT 718;"SAVES 1;" *Saves analyzer state in register 1.*

### Description

The state data is saved in the specified state register if it has not been locked by the PSTATE command.

State register nine contains the previous state data, state register zero contains the current state.

The functions of the SAVES command and `STATE -> INTRNL` are identical. (Also see "OL".)

### Note

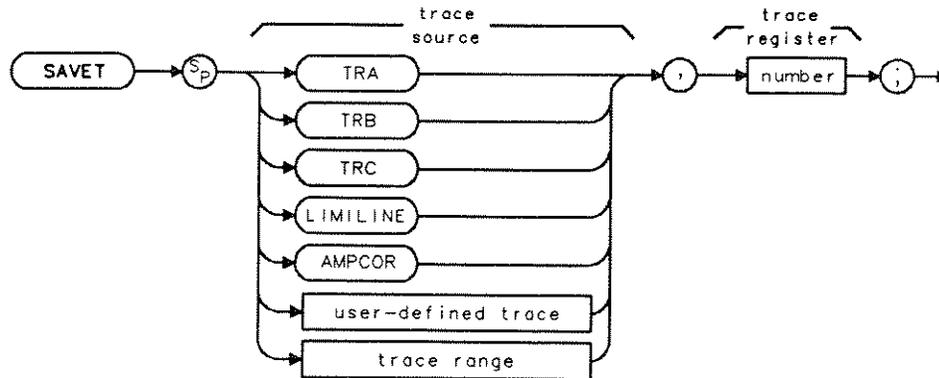


The SAVES saves state data in analyzer memory. See "STOR" or "SAVRCLN" to save state data on the memory card.

## SAVET Save Trace

The SAVET command saves the selected trace data and state information, amplitude correction factors, or limit-line tables in analyzer memory.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command with a length of 401 elements.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any valid integer.	0 to TRCMEM - 1.

Prerequisite Commands: TRDEF when using a user-defined trace.

Related Commands: CAT, CLRW, PSTATE, RCLT, SNGLS, TS, VIEW.

### Example

```

OUTPUT 718;"IP;CF 300MHZ;SP 20MHZ;TS;" Initializes analyzer, changes the center frequency, span.
OUTPUT 718;"VIEW TRA;SAVET TRA,1;" Puts trace A in the view mode, saves analyzer state and trace A data in register 1.
OUTPUT 718;"IP;" Initializes analyzer.
OUTPUT 718;"RCLT TRA,1;VIEW TRA;" Recalls analyzer state, trace data.

```

### Description

The trace data is saved in the specified register if it has not been locked by PSTATE ON (see "SAVES"). The SAVET command is equivalent to **TRACE -> INTRNL**. Use AMPCOR to save amplitude correction factors, LIMILINE to save limit-line tables.

---

**Note**            The TS and VIEW commands should be executed prior to saving trace data.



---

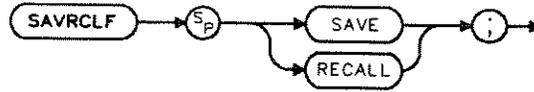
The SAVET saves trace data, amplitude correction factors, or limit-line tables in analyzer memory. See "STOR" or "SAVRCLN" to save data on the memory card.

## SAVRCLF

### Save or Recall Flag

The SAVRCLF command indicates a save or recall operation.

#### Syntax



Related Commands: MSI, PREFX, RCLS, RCLT, SAVES, SAVET, SAVRCLN, SAVRCLW.

#### Example

This example allows trace A to be saved on the memory card.

```

OUTPUT 718;"SAVRCLF SAVE;"
OUTPUT 718;"SAVRCLW TRA;"
OUTPUT 718;"MSI CARD;"

```

*Specifies a save operation.  
Specifies the source as trace A.  
Specifies the card as the mass storage device.*

```

OUTPUT 718;"PREFX %FRED%;"

```

*Specifies the prefix to store the trace data under.*

```

OUTPUT 718;"PU;PA 0,160;TEXT%ENTER TEST NUMBER%;"

```

*Prompts the user for the number to append to the prefix.*

```

OUTPUT 718;"SAVRCLN EP;"

```

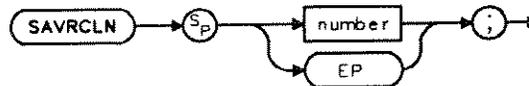
*After the user enters the number, the number is to appended to the prefix and the data transfer is initiated.*

The memory card now contains a file called tFRED\_(register number).

## SAVRCLN Save or Recall Number

The SAVRCLN command specifies the number to append to the prefix for a save or recall operation, and initiates the transfer of data.

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	Dependent on mass storage device.

Related Commands: MSI, PREFIX, RCLS, SAVES, SAVRCLF, SAVRCLW.

### Example

This example allows trace A to be saved on the memory card.

OUTPUT 718;"SAVRCLF SAVE;"	<i>Specifies a save operation.</i>
OUTPUT 718;"SAVRCLW TRA;"	<i>Specifies the source as trace A.</i>
OUTPUT 718;"MSI CARD;"	<i>Specifies the card as the mass storage device.</i>
OUTPUT 718;"PREFIX %FRED%;"	<i>Specifies the prefix to store the trace data under.</i>
OUTPUT 718;"PU;PA 0,160;TEXT%ENTER TEST NUMBER%;"	<i>Prompts the user for the number to append to the prefix.</i>
OUTPUT 718;"SAVRCLN EP;"	<i>After the user enters the number, the number is appended to the prefix and the data transfer is initiated.</i>

### Description

SAVRCLN is used to save or recall data in analyzer memory or on the memory card. The SAVRCLN command takes SAVRCLF flag information, SAVRCLW information, prefix, and mass storage device to form a complete command to transfer data.

Since the SAVRCLN command initiates the transfer of data, it should be the last command specified in the sequence to save or recall data.

The sequence to save or recall data is:

1. Specify either a save or recall operation with SAVRCLF.

**SAVRCLN Save or Recall Number**

2. Indicate the type of data to be saved or recalled using SAVRCLW.
3. Specify the memory card or analyzer memory as the mass storage device with MSI.
4. When saving to or recalling from a memory card, specify the prefix to be used with PREFX. The prefix is ignored when saving or recalling from analyzer memory.
5. Enter the number to append to the prefix and initiate the data transfer with SAVRCLN.

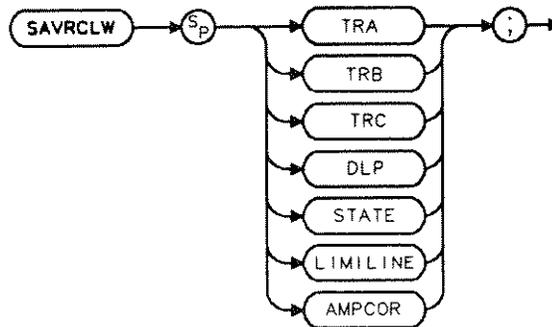
When saving trace data, amplitude correction factors, or limit-line tables in analyzer memory, specify a number within the trace register number range (0 to TRCMEM - 1). When saving state data in analyzer memory, specify a number within the state number range (1 to 8).

When saving data on the memory card, the number plus the number of characters in the prefix must not exceed eight characters.

## SAVRCLW Save or Recall Data

The SAVRCLW command specifies the data to be transferred—trace A, trace B, trace C, program, amplitude correction factors, limit line, or state.

### Syntax



Related Commands: MSI, PREFX, RCLS, RCLT, SAVES, SAVRCLF, SAVRCLN.

### Example

This example allows the current analyzer state to be saved on the memory card.

OUTPUT 718;"SAVRCLF SAVE;"	<i>Specifies a save operation.</i>
OUTPUT 718;"SAVRCLW STATE;"	<i>Specifies the source as the current analyzer state.</i>
OUTPUT 718;"MSI CARD;"	<i>Specifies the card as the mass storage device.</i>
OUTPUT 718;"PREFX %FRED%;"	<i>Specifies the prefix to store the state data under.</i>
OUTPUT 718;"SAVRCLN 34;"	<i>Appends the register number 34 to the prefix and initiates the data transfer.</i>

The memory card now has a file called sFRED\_34 that contains the instrument state.

### Description

SAVRCLW is used to save or recall data in analyzer memory or on the memory card. See SAVRCLN for the sequence of commands to initiate a data transfer.

**SAVRCLW Save or Recall Data**

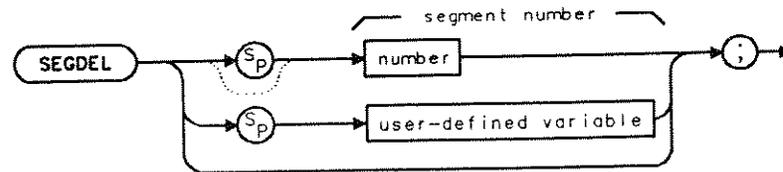
The SAVRCLW parameters correspond to the type of data transferred as follows:

<b>Parameter</b>	<b>Type of Data Transferred</b>
TRA	trace A
TRB	trace B
TRC	trace C
DLP	downloadable programs
STATE	instrument state
LIMLINE	limit lines
AMPCOR	amplitude correction factors

## SEGDEL Segment Delete

The SEGDEL command deletes the specified segment from the limit-line table(s).

### Syntax



Related Commands: LIMIMODE, LIMISEG, SENTER.

### Example

#### Example 1

This example uses LIMIMODE for entering segments into the upper limit-line table, then entering a segment into the lower limit-line table (upper and lower limit lines are treated as separate tables). Line 60 demonstrates the effect of deleting a segment when the upper and lower limit-line tables are treated separately.

10 OUTPUT 718;"LIMIDEL;"	<i>Deletes the current limit-line table, sets the table type to fixed.</i>
20 OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
30 OUTPUT 718;"LIMISEG 300MHZ,-30DB,FLAT;"	<i>Enters a segment into the upper limit-line table.</i>
40 OUTPUT 718;"LIMIMODE LOWER;"	<i>Specifies the lower limit-line table.</i>
50 OUTPUT 718;"LIMISEG 300MHZ,-70DB,FLAT;"	<i>Enters a segment into the lower limit-line table.</i>
60 OUTPUT 718;"SEGDEL 1;"	<i>Deletes the segment from the lower limit-line table.</i>
70 END	

#### Example 2

With the addition of line 60, the upper and lower limit-line tables are no longer treated as separate tables, but as one table. The segment is deleted from the upper and lower limit-line tables (for the given frequency).

10 OUTPUT 718;"LIMIDEL;"	<i>Deletes the current limit-line table.</i>
20 OUTPUT 718;"LIMIMODE UPPER;"	<i>Specifies the upper limit-line table.</i>
30 OUTPUT 718;"LIMISEG 300MHZ,-30DB,FLAT;"	<i>Enters segment into the upper limit-line table.</i>

**SEGDEL Segment Delete**

40 OUTPUT 718;"LIMIMODE LOWER;"	<i>Specifies the lower limit-line table.</i>
50 OUTPUT 718;"LIMISEG 300MHZ,-70DB,FLAT;"	<i>Enters a segment into the lower limit-line table.</i>
60 OUTPUT 718;"LIMIMODE UPLOW;"	<i>Specifies both the upper and lower limit-line tables.</i>
70 OUTPUT 718;"SEGDEL 1;"	<i>Deletes the segment from the upper and lower limit-line tables.</i>
80 END	

**Description**

The result of SEGDEL depends on the setting of the LIMIMODE command:

LIMIMODE Setting	Result of SEGDEL
LIMIMODE UPPER	Deletes specified segment from the upper limit-line table.
LIMIMODE LOWER	Deletes specified segment from the lower limit-line table.
LIMIMODE UPLOW	Deletes specified segment from the upper and lower limit-line table.
LIMIMODE DELTA	Deletes specified segment from the mid/delta limit-line table.

**Note**

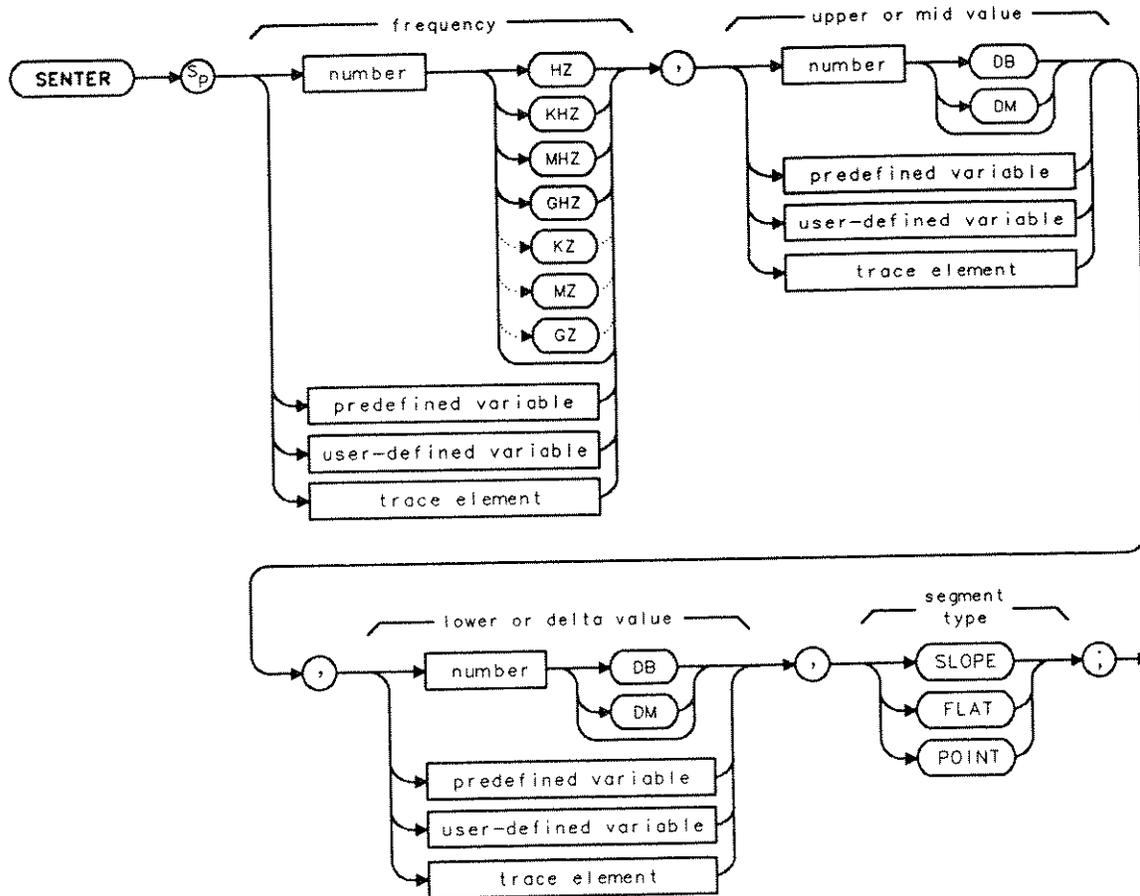
Query LIMIMODE before using SEGDEL if you are unsure of the LIMIMODE setting.

Display the limit-line table to determine the segment number you want to delete. (Limit-line entries are sorted according to frequency.)

## SENER Segment Entry

The SENER command enters the limit-line data in the upper and lower limit-line table or the mid/delta table.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is dBm.	Varies with FOFFSET and ROFFSET.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

**SENDER Segment Entry**

Related Commands: LIMIMODE, LIMIREL, LIMISEG, SEGDEL.

**Example 1**

This example enters limit-line values into the upper and lower limit-line tables.

```
OUTPUT 718;"LIMIMODE UPLow;LIMIREL OFF;"           Specifies the upper and lower limit-
                                                    line table as fixed.
OUTPUT 718;"SENDER 300MHZ,-10DB,-50DB,FLAT;"      Enters in values for a segment.
OUTPUT 718;"SENDER 350MHZ,-15DB,-60DB,FLAT;"      Enters in values for a segment.
```

**Example 2**

```
OUTPUT 718;"LIMIMODE DELTA;LIMIREL OFF;"          Specifies the mid/delta table format
                                                    and fixed type.
OUTPUT 718;"SENDER 300MHZ,-20DB,10DB,FLAT;"      Enters in values for a segment.
OUTPUT 718;"SENDER 350MHZ,-30DB,20DB,FLAT;"      Enters in values for a segment.
```

**Description**

Each limit-line segment is specified with a starting frequency, an upper or median amplitude value, a lower or delta amplitude value, and a segment type. The segment type defines how the line segment is to extend from its starting point to the next segment.

Three types are available:

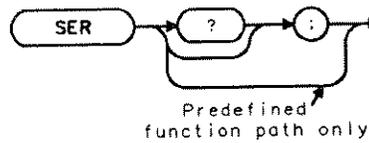
- FLAT draws a zero-slope line between the coordinate point of the current segment and the coordinate point of the next segment, producing limit-line values equal in amplitude for all frequencies between the two points. If the amplitude values of the two segments differ, the limit-line will "step" to the value of the second segment.
- SLOPE draws a straight line between the coordinate point of the current segment and the coordinate point of the next segment, producing limit-line values for all frequencies between the two points.
- POINT specifies a limit value for the coordinate point, and no other frequency points, so that a POINT segment specifies a limit value for a single frequency. For an upper limit-line, a POINT segment is indicated by a line drawn from the coordinate point, vertically off the top of screen. For a lower limit-line, a POINT segment is indicated by a line drawn from the coordinate point, vertically off the bottom of screen. The POINT segment type should be used as the last segment in the limit-line table. However, if the last segment in the table is not specified as the POINT segment type, an implicit point is automatically used. If a visible POINT segment at the right-hand edge of the display is not desired, add an explicit last point segment to the limit-line table that is higher in frequency than the stop frequency.

Segments are sorted as they are entered according to starting frequency. A maximum of 20 segments can be defined using SENDER. When the type is omitted, the last type given (or SLOPE if no previous type has been given) is used.

## SER Serial Number

The SER query returns the serial number suffix of the analyzer.

### Syntax



Related Commands: ID, REV.

### Example

<code>DIM Serial\$[24]</code>	<i>Reserves memory space for a string.</i>
<code>OUTPUT 718;"SER;"</code>	<i>Gets the serial number from the analyzer.</i>
<code>ENTER 718;Serial\$</code>	<i>Puts the analyzer response in the computer variable.</i>
<code>DISP Serial\$</code>	<i>Displays the serial number on the computer screen.</i>

### Query Response

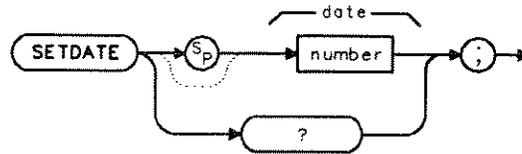
The last five digits of the serial number are returned.



## SETDATE Set Date

The SETDATE command allows you to set the date of the real-time clock of the analyzer.

### Syntax



Item	Description/Default	Range
Number	A number in the YYMMDD format.	Valid year, month, and day.

Related Commands: SETTIME, TIMEDATE, TIMEDSP.

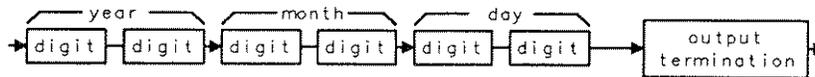
### Example

OUTPUT 718;"SETDATE 890212;" *Sets the date to February 12, 1989.*

### Description

The functions of SETDATE and SET DATE are identical.

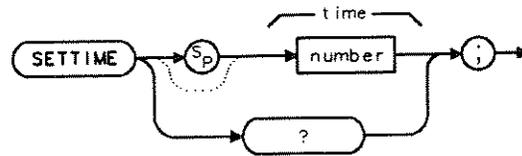
### Query Response



## SETTIME Set Time

The SETTIME command allows you to set the time of the real-time clock of the analyzer.

### Syntax



Item	Description/Default	Range
Number	A number in the HHMMSS (24 hour) format.	0 to 235959.

Related Commands: SETDATE, TIMEDATE, TIMEDSP.

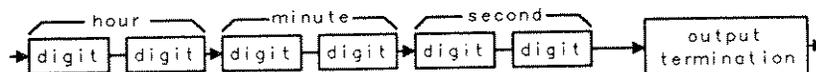
### Example

OUTPUT 718;"SETTIME 135501;" *Sets the time to 1:55:01 PM.*

### Description

The functions of SETTIME and SET TIME are identical.

### Query Response

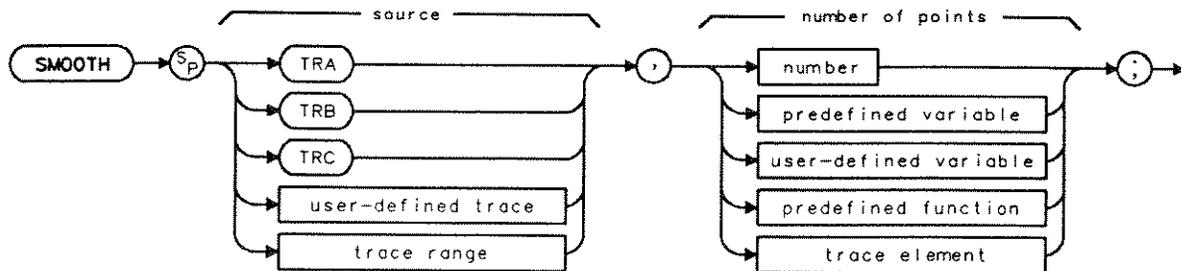


## SMOOTH

### Smooth Trace

The SMOOTH command smooths the trace according to the number of points specified for the running average.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	.
Number	Any real or integer number.	Real number range.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace element	An element of trace A, trace B, trace C, or a user-defined trace.	

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.

Related Commands: SNGLS, TS, VAVG.

### Example

```

10 OUTPUT 718;"IP;"           Initializes analyzer.
20 OUTPUT 718;"SNGLS;TS;"     Activates single-sweep mode, takes a sweep.
30 OUTPUT 718;"VIEW TRA;"     Stores results of trace A.
40 OUTPUT 718;"SMOOTH TRA,10;" Smooths trace A.
50 OUTPUT 718;"VIEW TRA;"     Displays the result.
60 END

```

## Description

Each point value is replaced with the average of the values (in measurement units) of the given number of points centered on it. Increasing the number of points increases smoothing at the cost of decreasing resolution. If the number of points is an even number, then the number of points is increased by one. If the number of points is larger than the size of SOURCE, then the size of SOURCE is used (unless size of SOURCE is even, in which case the size of SOURCE minus one is used). Smoothing decreases at the endpoints.

The purpose of this function is to perform a spatial video averaging as compared to the temporal version supplied by the video-average (VAVG) command. By replacing the value of each point in a trace with the average of the values of a number of points centered about that point, any rapid variations in video noise or signals are smoothed into more gradual variations. It thereby performs a function similar to reducing the video bandwidth without the corresponding changes in sweep time. As such, it does result in a reduction of frequency resolution.

This command requires user memory for its execution. Memory is not permanently allocated, so the largest amount of memory is available for the functions that are used in a particular application. When the command is complete, memory is returned to the free user memory.

## SNGLS Single Sweep

The SNGLS command sets the analyzer to single-sweep mode.

### Syntax



Related Commands: CLRW, CONTS, TM, TS.

### Example

```
OUTPUT 718;"SNGLS;"
```

### Description

Each time TS (take sweep) is sent, one sweep is initiated, as long as the trigger and data entry conditions are met.

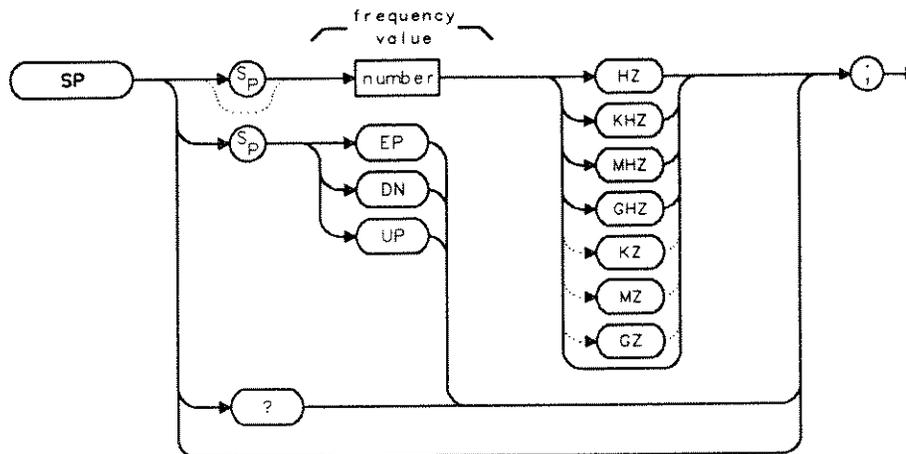
The functions of the SNGLS command, **SGL SWP**, and **SWEEP CONT SGL** (SGL) are identical.

For the HP 8592B or the HP 8593A only: The frequency span that can be viewed with a single-sweep is bounded by the instrument range only; therefore, band 0 can be included in a multiband sweep in single sweep mode. This allows a 0 GHz to 22 GHz span with an HP 8592B or HP 8593A (also see "TS").

## SP Span

The SP command changes the total displayed frequency range symmetrically about the center frequency.

### Syntax



Item	Description/Default	Range
Number	Any real or integer number. Default unit is Hz.	0 to 1.8 GHz for the HP 8590B/8591A. 0 to 19.39 GHz for the HP 8592B/8593A.

Preset State HP 8590B and HP 8591A: 1.8 GHz.

Preset State HP 8592B and HP 8593A: 19.25 GHz.

Step Increment: 1, 2, 5, 10 sequence (up to the stop frequency of the analyzer).

Related Commands: CF, FA, FB, FOFFSET, HNLOCK, HNUNLK, RB, ST, VB.

### Example

```
OUTPUT 718;"IP;SP 20MHZ;"  Initializes analyzer, changes frequency span.
OUTPUT 718;"SP?;"        Gets the span value from the analyzer.
ENTER 718;Span            Puts the analyzer response in the computer variable, Span.
PRINT Span                Displays the span value.
```

## SP Span

### Description

The frequency span readout refers to the displayed frequency range. Dividing the readout by 10 yields the frequency span per division.

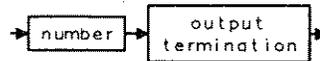
If resolution and video bandwidths are coupled to the span width, the bandwidths change with the span width to provide a predetermined level of resolution and noise averaging. Likewise, the sweep time changes to maintain a calibrated display, if coupled. All of these functions are normally coupled, unless RB, VB, or ST have been executed.

Since span is affected by frequency, change the frequency before changing span (see "HNLOCK").

Specifying 0 Hz enables zero span mode, which configures the analyzer as a fixed-tuned receiver.

The functions of the SP command and the **SPAN** are identical.

### Query Response

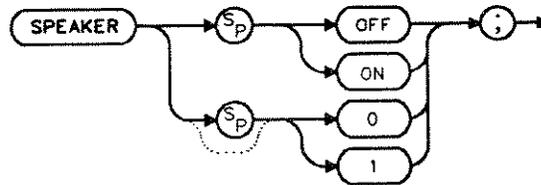


## SPEAKER Speaker

The SPEAKER command turns the internal speaker on or off.

Option and Model Required: Option 102 with HP 8591A or HP 8593A

### Syntax



Preset State: ON.

Related Commands: DEMOD, FMGAIN, SQLCH.

### Example

```
OUTPUT 718;"SPEAKER OFF;"
```

## SPZOOM

### Span Zoom

The SPZOOM command places a marker on the highest on-screen signal (if an on-screen marker is not present), turns on the signal track function, and activates the span function.

### Syntax



### Example

```
OUTPUT 718;"IP;CF 300MZ;TS;"  
OUTPUT 718;"SPZOOM;"
```

### Description

If a marker is present before SPZOOM is executed, SPZOOM turns on the signal track function and activates the span function.

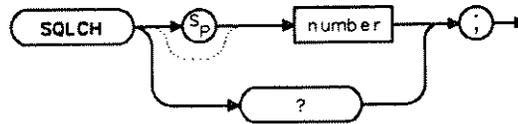
The function of the SPZOOM command and **SPAN ZOOM** are identical.

## SQLCH Squelch

The SQLCH command sets the squelch threshold by setting the squelch level.

**Option and Model Required:** Option 102 with HP 8591A or HP 8593A

### Syntax



Item	Description/Default	Range
Number	Any valid integer.	0 to 100.

Preset Value: 0.

Related Commands: DEMOD, FMGAIN, SPEAKER.

### Example

```
OUTPUT 718;"SQLCH 100;"
```

### Description

SQLCH mutes weak signals and passes strong signals. The function of the SQLCH command is identical to SQUELCH.

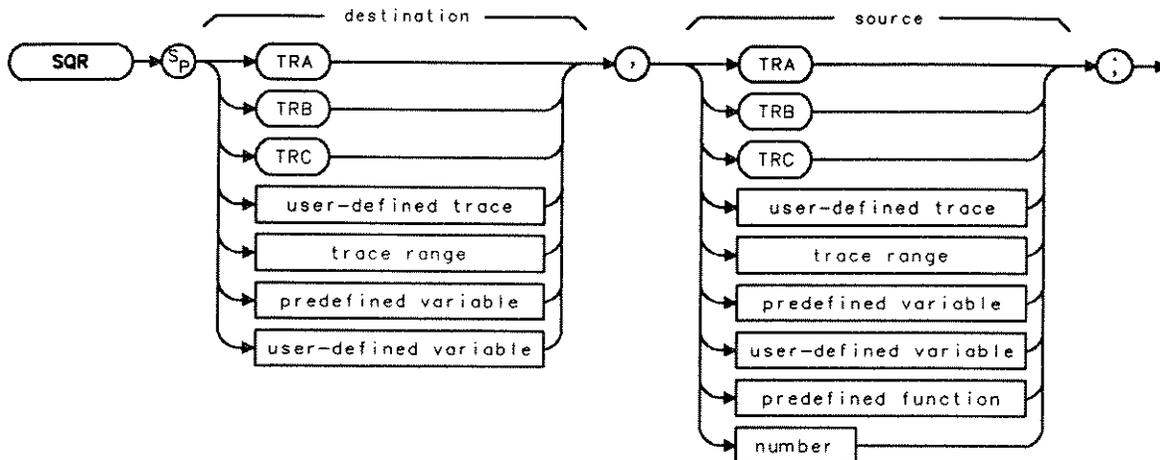
### Query Response



## SQR Square Root

The square root of the source is placed in the destination.

### Syntax



Item	Description/Default	Range
User-defined trace	A trace defined by the TRDEF command.	Any valid trace name.
User-defined variable	A variable defined by VARDEF or ACTDEF commands.	Any valid variable name.
Predefined variable	A command that acts as a variable. Refer to Table 4-1.	
Predefined function	Function that returns a value. Refer to Table 4-1.	
Trace range	A segment of trace A, trace B, trace C, or a user-defined trace.	
Number	Any real or integer number.	Real number range.

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.

Related Commands: PDA, PDF, STDEV.

**Example**

OUTPUT 718;"SQR SP,1E8;" *Changes the span to 10 kHz.*

**Description**

If the source is negative, the square root of the absolute value will be returned.

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

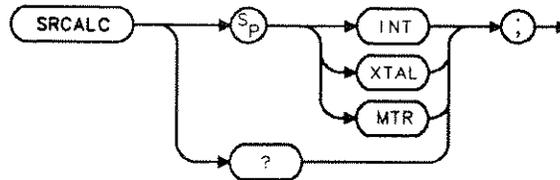
## SRCALC

### Source Leveling Control

The SRCALC command selects internal or external leveling for use with the built-in tracking generator.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Preset State: SRCALC INT.

Related Commands: CF, FA, FB, FS, HNLOCK, SP.

### Example

```
OUTPUT 718;"SRCALC XTAL;"
```

### Description

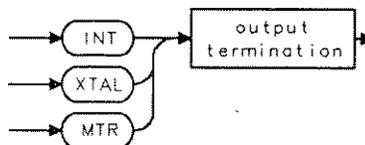
SRCALC INT activates internal leveling.

SRCALC XTAL activates external leveling. The external leveling input is located on the rear panel of the analyzer. Positive- or negative-polarity detectors are supported. External leveling increases the amplitude accuracy by improving the effective source match.

SRCALC MTR narrows loop bandwidth so Hewlett-Packard power meters can be used for external leveling.

The functions of SRCALC and `ALC MTR INT XTAL` are identical.

### Query Response

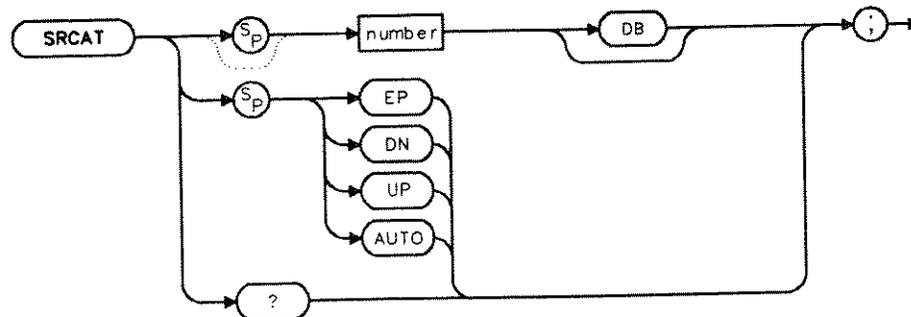


## SRCAT Source Attenuator

The SRCAT command attenuates the source output level.

Option and Model Required: Option 010 or 011 with HP 8591A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	0 to 60 dB.

Coupling: Coupled to power level of the source output (SRCPWR) when set to auto (SRCAT AUTO).

Related Commands: SRCPSTP.

Preset State: SRCAT AUTO.

### Example

Use the SRCAT command to attenuate the source output.

```

OUTPUT 718;"SRCAT AUTO;"      Activates source-attenuation coupling.
OUTPUT 718;"SRCPWR -20DB;"    Activates source output.
OUTPUT 718;"SRCAT 20DB;"      Sets attenuator to 20 dB. This decouples the attenuator
                                from the source power-level setting.
  
```

### Description

The SRCAT command attenuates the output level of the source. Use SRCAT to attenuate the power level of the source manually, from 0 to 60 dB in 10 dB steps.

“SRCAT AUTO;” automatically adjusts the attenuator to yield the source amplitude level specified by the SRCPWR command.

The functions of SRCAT and `SRC ATN MAN AUTO` are identical.

**SRCAT Source Attenuator**

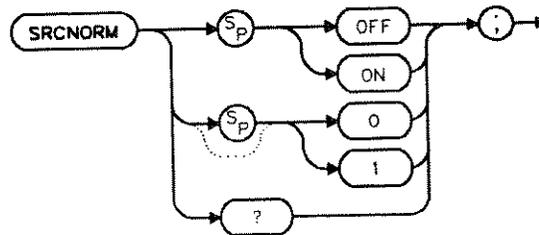
**Query Response**



## SRCNORM Source Normalization

The SRCNORM command subtracts trace B from trace A, adds the display line value to the difference, and sends the result to trace A during every sweep of the analyzer.

### Syntax



Preset State: SRCNORM OFF.

Related Commands: AMB, CONTS, CLRW, DL, MXMH, SNGLS, TS, VAVG, VIEW.

### Example

```
10 OUTPUT 718;"IP;SNGLS;RL 20DB;"
20 OUTPUT 718;"MOV TRA,5000;"
```

*Initializes analyzer, activates single-sweep mode.  
 Sets trace A to 5000 measurement units which  
 is equal to -10 dBm.*

```
30 OUTPUT 718;"VIEW TRA;"
40 OUTPUT 718;"MOV TRB,4000;"
```

*Sets trace B to 4000 measurement units which  
 is equal to -20 dBm.*

```
50 OUTPUT 718;"VIEW TRB;"
60 OUTPUT 718;"DL ODM;"
```

*Sets display line to 0 dBm which is at 6000  
 measurement units.*

```
70 OUTPUT 718;"SRCNORM ON;"
```

*Performs trace A - trace B + display line. The  
 result is 5000 - 4000 + 6000 = 7000 or 10 dBm.  
 Note that this has resulted in a subtraction of  
 amplitude in dBm, -10 dBm - (-20 dBm) =  
 10 dBm.*

```
80 OUTPUT 718;"BLANK TRB;VIEW TRA;"
90 END
```

## SRCNORM Source Normalization

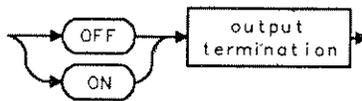
### Description

The SRCNORM command subtracts trace B from trace A, point by point, adds the display line value to the difference, and sends the difference to trace A. The SRCNORM function remains in effect until it is turned off by executing "SRCNORM OFF;".

A common use of trace subtraction is to normalize one trace with respect to another. For example, traces are frequently subtracted to normalize the analyzer response when a tracking generator is used. In such applications, amplitude units in dBm should be subtracted. To accomplish this, the display line should be set to 0 dBm using DL as shown in the example. Also see example 2 and 3 in "AMB" for comparison.

The functions of SRCNORM, AMBPL; and **NORMLIZE ON OFF** are identical.

### Query Response

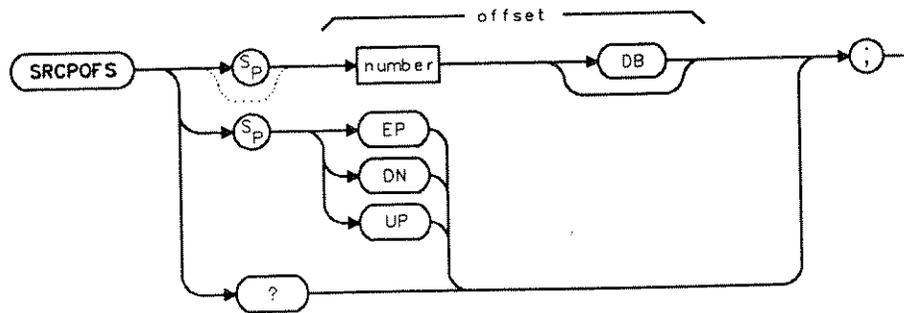


## SRCPOFS Source Power Offset

The SRCPOFS command offsets the source power level readout.

Option and Model Required: Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	

Related Commands: SRCPWR, SRCPSWP.  
 Step Increment: Determined by SRCPSTP.  
 Preset State: 0 dB.

### Example

Use SRCPOFS to offset the power-level readout for the tracking-generator source.

```

140 OUTPUT 718;"MEASURE SR;"      Sets spectrum analyzer to stimulus-response mode.
150 OUTPUT 718;"SRCPWR -10DB;"    Turns on source output.
160 OUTPUT 718;"SRCPOFS 13DB;"    Offsets power-level readout for source by 13 dB.
  
```

### Description

The SRCPOFS command offsets the displayed power of the built-in tracking generator so that it is equal to the measured power at the input of the spectrum analyzer. This function may be used to take into account system losses (for example, cable loss) or gains (for example, preamplifier gain) reflecting the actual power delivered to the device under test.

The functions of SRCPOFS and SRC PWR OFFSET are identical.

**SRCPOFS Source Power Offset**

**Query Response**

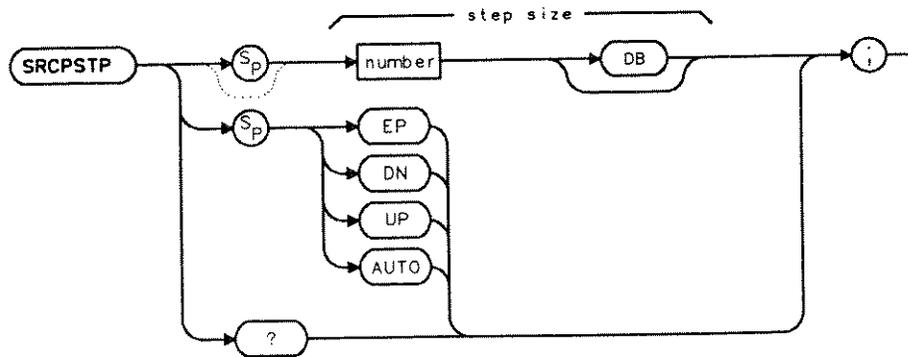


## SRCPSTP Source Power-Level Step Size

The SRCPSTP selects the source-power step size.

Option and Model Required: Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	

Step Increment: 0.1 dB.

Related Commands: SRCPWR, SRCPOFS, SRCPSWP.

Preset State: SRCPSTP AUTO (one major vertical scale division).

### Example

Select incremental changes of power effected by "SRCPWR UP;," "SRCPWR DN;," commands, or the step keys.

OUTPUT 718;"MEASURE SR;"	<i>Activates stimulus-response mode.</i>
OUTPUT 718;"SRCPWR -10DB;"	<i>Turns on the source output.</i>
OUTPUT 718;"SRCPSTP .3DB;"	<i>Sets power-level step size to .3 dB.</i>
OUTPUT 718;"SRCPWR UP;"	<i>Increases the power level.</i>

## SRCPSTP Source Power-Level Step Size

### Description

The SRCPSTP command selects the step size for the following source commands:

- Power offset (SRCPOFS).
- Power sweep (SRCPSWP).
- Power (SRCPWR).

Use SRCPSTP to set the step size to a specific value.

“SRCPSTP AUTO;” sets the step size to one vertical scale division.

The functions of SRCPSTP and **SRC PWR STP SIZE** are identical.

### Query Response

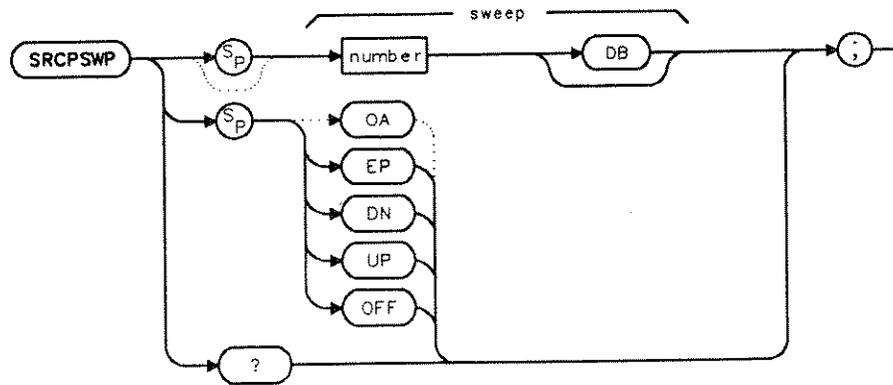


## SRCPSWP Source Power Sweep

The SRCPSWP command selects sweep range of source output.

Option and Model Required: Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Item	Description/Default	Range
Number	Any real or integer number.	

Step Increment: Determined by SRCPSTP.

Related Commands: SRCPSWP, SRCPOFS, SRCPSTP.

Preset State: SRCPSWP OFF.

### Example

Use SRCPSWP to sweep the power level of the source output.

- ```

130 OUTPUT 718;"MEASURE SR;"      Activates stimulus-response mode.
140 OUTPUT 718;"SRCPWR -10DB;"    Sets power level of source output to -10 dBm.
150 OUTPUT 718;"SP 0;"           Sets span to 0 Hz.
160 OUTPUT 718;"SRCPSWP 10DB;"   Sweeps source output from -10 dBm to 0 dBm.
  
```

## SRCPSWP Source Power Sweep

### Description

The SRCPSWP command works in conjunction with the SRCPWR (source power) command to sweep the amplitude level of the source output. The SRCPWR setting determines the amplitude level at the beginning of the sweep. The SRCPSWP command determines the change in amplitude level of the sweep.

For example, if SRCPWR and SRCPSWP are set to  $-15$  dBm and 4 dB respectively, the source sweeps from  $-15$  dBm to  $-11$  dBm.

---

**Note** Power is swept from low to high.



---

The functions of SRCPSWP and **PWR SWP ON OFF** are identical.

### Query Response

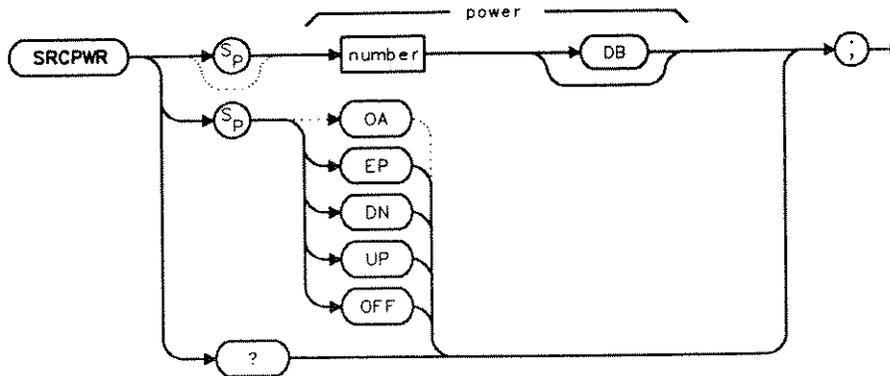


## SRCPWR Source Power

The SRCPWR command selects the source power level.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



| Item   | Description/Default                                                     | Range                               |
|--------|-------------------------------------------------------------------------|-------------------------------------|
| Number | Any real or integer number. Default unit is the current amplitude unit. | Actual range is hardware dependent. |

Step Increment: Set by SRCPSTP.

Related Commands: SRCAT, SRCPSTP, SRCPSWP.

Preset State: SRCPWR -10DM.

### Example

Use SRCPWR to turn on the source and adjust its power level.

```
OUTPUT 718;"SRCPWR -20DB;"      Changes power level to -20 dBm.
OUTPUT 718;"AUNITS DBMV;"      Changes the current amplitude unit.
OUTPUT 718;"SRCPWR 37;"        The source power is now 37 dBmV.
```

**SRCPWR Source Power****Description**

The SRCPWR command sets the power level of the source or turns it off or on. The source is turned on automatically whenever its value is specified with SRCPWR. See SRCPSTP.

The functions of SRCPWR and **SRC PWR ON OFF** are identical.

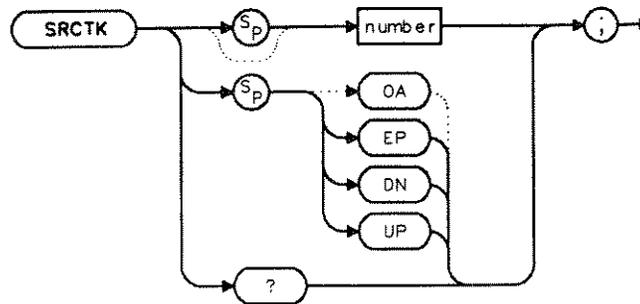
**Query Response**

## SRCTK Source Tracking

The SRCTK command adjusts tracking of source output with spectrum-analyzer sweep.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



| Item   | Description/Default         | Range        |
|--------|-----------------------------|--------------|
| Number | Any real or integer number. | 0 to 16,383. |

Step Increment: 1.

Related Commands: Commands that change bandwidth, such as RB, VB, ST, and commands that change frequency, such as SP, CF, FA, FB, SP, FS.

### Example

|                            |                                                                  |
|----------------------------|------------------------------------------------------------------|
| OUTPUT 718;"MEASURE SR;"   | <i>Activates the stimulus-response mode.</i>                     |
| OUTPUT 718;"SRCPWR -20DB;" | <i>Turns on the power at the source output.</i>                  |
| OUTPUT 718;"SP 1MHZ;"      | <i>Sets measurement range.</i>                                   |
| OUTPUT 718;"RB 1KHZ;"      |                                                                  |
| OUTPUT 718;"TS;"           | <i>Takes sweep.</i>                                              |
| OUTPUT 718;"SRCTK EP;"     | <i>Allows entry of from front-panel keys to adjust tracking.</i> |

## SRCTK Source Tracking

### Description

The SRCTK command adjusts the tracking of the tracking-generator output with the center frequency of the spectrum-analyzer. SRCTK is used typically for bandwidths less than 300 kHz. Bandwidths greater than 300 kHz do not require tracking adjustment. Use SRCTK to improve amplitude accuracy and maximize signal response. Use SRCTKPK to adjust tracking automatically. See "SRCTKPK."

The functions of SRCTK and **MAN TRK ADJUST** are identical.

### Query Response



## SRCTKPK Source Tracking Peak

The SRCTKPK automatically adjust tracking of source output with spectrum-analyzer sweep.  
**Option and Model Required: Option 010 or 011 with HP 8590B or HP 8591A**

### Syntax



Related Commands: Commands that change bandwidth, such as RB, VB, ST, and commands that change frequency, such as SP, CF, FA, FB, SP, FS.

### Example

Automatically adjust the tracking of the tracking-generator module output.

|     |                            |                                                                        |
|-----|----------------------------|------------------------------------------------------------------------|
| 150 | OUTPUT 718;"MEASURE SR;"   | <i>Activates the stimulus-response mode.</i>                           |
| 160 | OUTPUT 718;"SRCPWR -10DB;" | <i>Turns on the power at the source output to its current setting.</i> |
| 170 | OUTPUT 718;"SP 1MHZ;"      | <i>Sets measurement range.</i>                                         |
| 180 | OUTPUT 718;"RB 1KHZ;"      |                                                                        |
| 190 | OUTPUT 718;"TS;"           | <i>Takes sweep.</i>                                                    |
| 200 | OUTPUT 718;"SRCTKPK;"      | <i>Automatically adjusts the tracking.</i>                             |

### Description

The SRCTKPK command automatically adjusts the tracking of the tracking-generator source output to maximize responses for measurements made with resolution bandwidths less than 300 kHz.

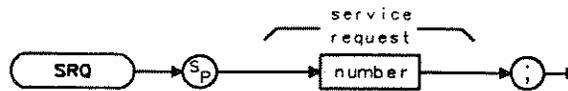
SRCTKPK maximizes the amplitude of the displayed active trace.

The functions of SRCTKPK and **TRACKING PEAK** are identical.

## SRQ Force Service Request

The SRQ command is used by an external controller to simulate interrupts from the analyzer.

### Syntax



| Item   | Description/Default | Range     |
|--------|---------------------|-----------|
| Number | Any valid integer.  | 2 to 126. |

Related Commands: CLS, EE, RQS, STB.

### Example

`OUTPUT 718;"RQS 8;SRQ 8;"` Sets bit mask for a hardware broken service request, generates a hardware broken interrupt.

### Note



A program can respond to the interrupt in the same way it would under a true service request condition.

### Description

The service request condition is also displayed on the analyzer screen with the annotation SRQ XXX, where XXX is a three-digit octal number.

The conditions that can generate a service request are as follows:

- 32 = Illegal command
- 16 = Command complete
- 8 = Hardware broken
- 4 = End of sweep
- 2 = Units key pressed

A service request is generated only if the proper request mask bit has been set (see "RQS"), and either the condition itself or the Force Service Request is sent. To set the request mask, choose the desired interrupt conditions and sum their assigned values. Executing the RQS command with this value sets the bit mask. After setting the bit mask, only the chosen conditions can produce an interrupt.

Each bit in the status byte is defined in the following table:

Table 4-8. Analyzer Status Byte

| Bit Number | Decimal Equivalent | Analyzer State                          | Description                                                                                                                                                  |
|------------|--------------------|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5          | 32                 | Set when an illegal command is present. | SRQ 140 appears on the analyzer screen.                                                                                                                      |
| 4          | 16                 | Set when any command is completed.      | It is triggered by EOI at the end of a command string or the completion of a print or plot.                                                                  |
| 3          | 8                  | Indicates hardware broken condition.    | SRQ 110 appears on the analyzer screen.                                                                                                                      |
| 2          | 4                  | Indicates end of sweep.                 | SRQ 104 appears on the analyzer screen. If you send any RQS value that contains mask value 4, another sweep will be taken.                                   |
| 1          | 2                  | Indicates a units key was pressed.      | SRQ 102 appears on the analyzer screen. (If you activate the units key bit, it will remain active until you activate "EE" and press a units key. (See "EE.") |

Bit numbers 0 (LSB), 6, and 7 are not used.

The analyzer screen numbers 102, 104, and 110 are the octal values corresponding to the status register values; that is, SRQ 102 = bit 6 = octal 100 and bit 2 = octal 2 are both true.

Generally, you must set the bit mask using the RQS command. However, the "hardware broken" and "illegal remote command" conditions are automatically enabled after presetting or sending the IP command. Pressing **PRESET** or sending the IP command, then, produces the same interrupt bit mask as sending "RQS 40;" (decimal 40 is the sum of the assigned values of these two interrupt bits, 32 = bit 5 and 8 = bit 3).

For most conditions, the RQS mask bit stays set until the next preset, IP, or RQS command is executed. The only condition to which this does not apply is the Units Key Pressed bit. When this bit (bit 1) is set in the RQS mask, a Units Key Pressed interrupt occurs if EE (enable entry mode) is executed and a front-panel units key such as Hz, kHz, MHz, or GHz is pressed.

When a units key is pressed, the interrupt occurs and the Units Key Pressed bit in the RQS mask is reset. To reenable the Units Key Pressed interrupt, you must send a new RQS mask. See the RQS command for detailed information.

As mentioned, you can simulate a service request condition. Choose the desired interrupt conditions from the RQS command table (see "RQS"), and sum their assigned values. Use the RQS command with this value to set the bit mask. By setting the corresponding bits in the SRQ command and sending the SRQ command to the analyzer, the desired interrupt occurs. This allows the user to verify proper operation of software routines designed to handle infrequent or unlikely interrupts.

## SRQ Force Service Request

### Interface Differences

As implemented on the HP-IB interface, an analyzer service request asserts the SRQ control line on the HP-IB.

On the RS-232 interface, the analyzer does not have a way of signaling the interrupt condition to a controller. In this case, the controller must operate in a polled mode if it requires interrupt information (see "Polled Mode of Operation" below for a discussion of the polled mode).

#### Interrupt-Related Commands Common to All Interfaces:

- CLS Clear status byte, without read
- RQS Request mask
- SRQ Force service request
- STB Read then clear status byte

The HP-IB interface supports interface commands to read the status byte.

On HP-IB in HP 9000 Series 200 or 300 BASIC, the statement `SPOLL (Device_address)` can be used to read the status byte.

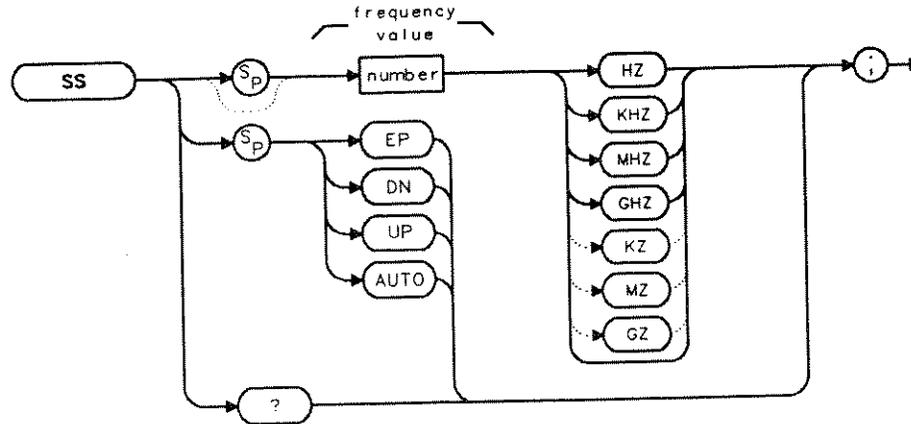
### Polled Mode of Operation

The polled mode of operation is probably most applicable to an RS-232 interface user. Since there is no interrupt signal to the RS-232 controller, the user must periodically ask the analyzer, via the "STB?" command, for the contents of its status register. For example, the RS-232 controller could periodically check for the hardware-broken condition by executing the "STB?" command and reading the results.

## SS Center Frequency Step Size

The SS command specifies center frequency step size.

### Syntax



| Item   | Description/Default                             | Range                                                                                        |
|--------|-------------------------------------------------|----------------------------------------------------------------------------------------------|
| Number | Any real or integer number. Default unit is Hz. | 0 to $\leq 1.8$ GHz (HP 8590B or 8591A), 0 to $\leq 26.5$ GHz (HP 8592B or 8593A). analyzer. |

Preset State: 100 MHz.

Step Increment: 1, 2, 5, 10 sequence.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.

Related Commands: AUTO, CF, FOFFSET, SP.

### Example

```
10 CLEAR 718
```

```
20 OUTPUT 718;"IP;SNGLS;CF 300MHZ;SP 20MHZ;TS;"
```

```
30 OUTPUT 718;"MKPK HI;MKRL;TS;MKF?;"
```

*Clears the HP-IB.*

*Initializes the analyzer, activates single-sweep mode, changes the center frequency, span, takes sweep. Finds the highest peak, changes the reference level to the marker, takes sweep, returns the frequency of the marker.*

**SS Center Frequency Step Size**

```

40 ENTER 718 USING "K";Mk_freq
50 OUTPUT 718;"MKA?;"
60 ENTER 718 USING "K";Mk_amp
70 OUTPUT 718;"SS ";Mk_freq;"HZ"
80 OUTPUT 718;"CF UP;TS;MKPK HI;MKA?;"

90 ENTER 718;Mk_ampl

100 PRINT "THE FUNDAMENTAL IS ";Mk_amp-Mk_ampl
110 PRINT "dB ABOVE THE SECOND HARMONIC"
120 END

```

*Puts the analyzer response in the computer variable, Mk\_freq.*

*Returns the amplitude of the marker.*

*Puts the analyzer response in the computer variable, Mk\_amp.*

*Changes the step size to the marker frequency.*

*Increases the center frequency, takes sweep, puts the marker on the highest peak and returns the amplitude of the marker.*

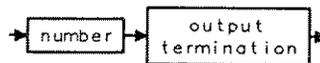
*Puts the analyzer response in the computer variable, Mk\_ampl.*

*Outputs the result.*

**Description**

The functions of the SS command and **CF STEP AUTO MAN** are identical.

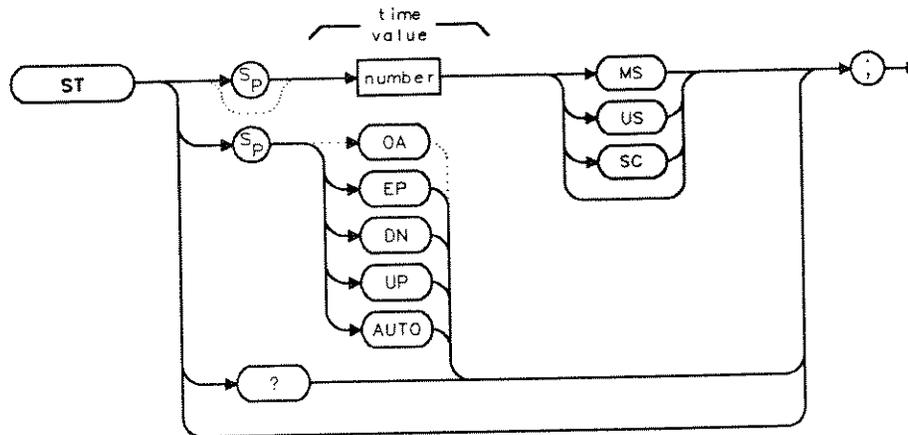
The AUTO parameter removes SS as an active function but does not have an effect on its value.

**Query Response**

## ST Sweep Time

The ST command specifies the time in which the analyzer sweeps the displayed frequency range.

### Syntax



| Item   | Description/Default                                  | Range                                        |
|--------|------------------------------------------------------|----------------------------------------------|
| Number | Any real or integer number. Default unit is seconds. | Within the sweep time range of the analyzer. |

Sweep Time Range in Zero Span: 15 ms to 100s.

Sweep Time Range in Zero Span, Option 101 only: 20  $\mu$ s to 100s.

Sweep Time Range in Non-zero Span: 20 ms to 100 s.

Preset State for HP 8590B and HP 8591A: 20 ms/GHz.

Preset State for HP 8592B and HP 8593A: 385 ms/GHz.

Step Increment: 2, 3, 5, 7.5, 10, 15 sequence.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.

Related Commands: AUTO, CONTS, HNLOCK, HNUNLK, RB, SNGLS, SP, TS.

## ST Sweep Time

### Example

OUTPUT 718;"ST 100MS;" *Sets the sweep time to 100 milliseconds.*

### Description

The function of the ST command and **SWP TIME AUTO MAN** are identical.

---

**Note** When used as a predefined variable, ST returns a real number in seconds.



---

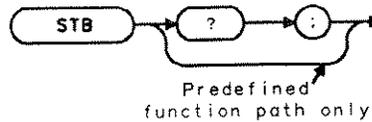
### Query Response



## STB Status Byte Query

The STB command returns to the controller the decimal equivalent of the status byte.

### Syntax



Related Commands: RQS, SRQ.

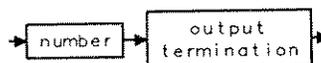
### Example

|                          |                                                                          |
|--------------------------|--------------------------------------------------------------------------|
| 10 OUTPUT 718;"IP;"      | <i>Initializes analyzer.</i>                                             |
| 20 OUTPUT 718;"SNGLS;"   | <i>Activates single-sweep mode.</i>                                      |
| 30 OUTPUT 718;"CLS;"     | <i>Clears the status bits.</i>                                           |
| 40 OUTPUT 718;"TS;"      | <i>Takes sweep.</i>                                                      |
| 50 OUTPUT 718;"STB?;"    | <i>Returns the status bits.</i>                                          |
| 60 ENTER 718;Status_Byte | <i>Puts the analyzer response in the computer variable, Status_Byte.</i> |
| 70 PRINT Status_byte     | <i>Displays the result.</i>                                              |
| 80 END                   |                                                                          |

### Description

The STB command is equivalent to a serial poll command. The RQS and associated bits are cleared in the same way that a serial poll command would clear them. The bits in the status byte are explained under the RQS command.

### Query Response

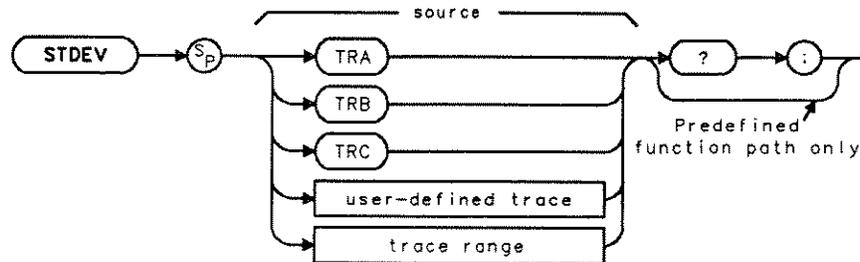


## STDEV

### Standard Deviation of Trace Amplitudes

The STDEV command returns the standard deviation of the trace amplitude in measurement units.

#### Syntax



| Item               | Description/Default                                              | Range                 |
|--------------------|------------------------------------------------------------------|-----------------------|
| User-defined trace | A trace defined by the TRDEF command.                            | Any valid trace name. |
| Trace range        | A segment of trace A, trace B, trace C, or a user-defined trace. |                       |

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: MEAN, PDA, PDF, VARIANCE.

#### Example

The following program segment finds the standard deviation of the amplitude of trace A.

|                                                            |                                                                      |
|------------------------------------------------------------|----------------------------------------------------------------------|
| OUTPUT 718;"IP;"                                           | <i>Initializes analyzer.</i>                                         |
| OUTPUT 718;"SNGLS;"                                        | <i>Activates single-sweep mode.</i>                                  |
| OUTPUT 718;"CF 300MHZ;SP 2MHZ;RB 100KHZ;"                  | <i>Changes the center frequency, span, and resolution bandwidth.</i> |
| OUTPUT 718;"TS;"                                           | <i>Takes sweep.</i>                                                  |
| OUTPUT 718;"STDEV TRA?;"                                   | <i>Finds the standard deviation of trace A.</i>                      |
| ENTER 718;Number                                           | <i>Get the response from the analyzer.</i>                           |
| PRINT "THE STANDARD DEVIATION OF TRACE A ";Number/100;"DB" |                                                                      |

## Description

The formula to calculate the standard deviation is as follows:

$$\sqrt{\frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n - 1}}$$

- $n$  represents the number of data points
- $x_i$  represents a data point
- $\bar{x}$  represents the mean of data

## Query Response

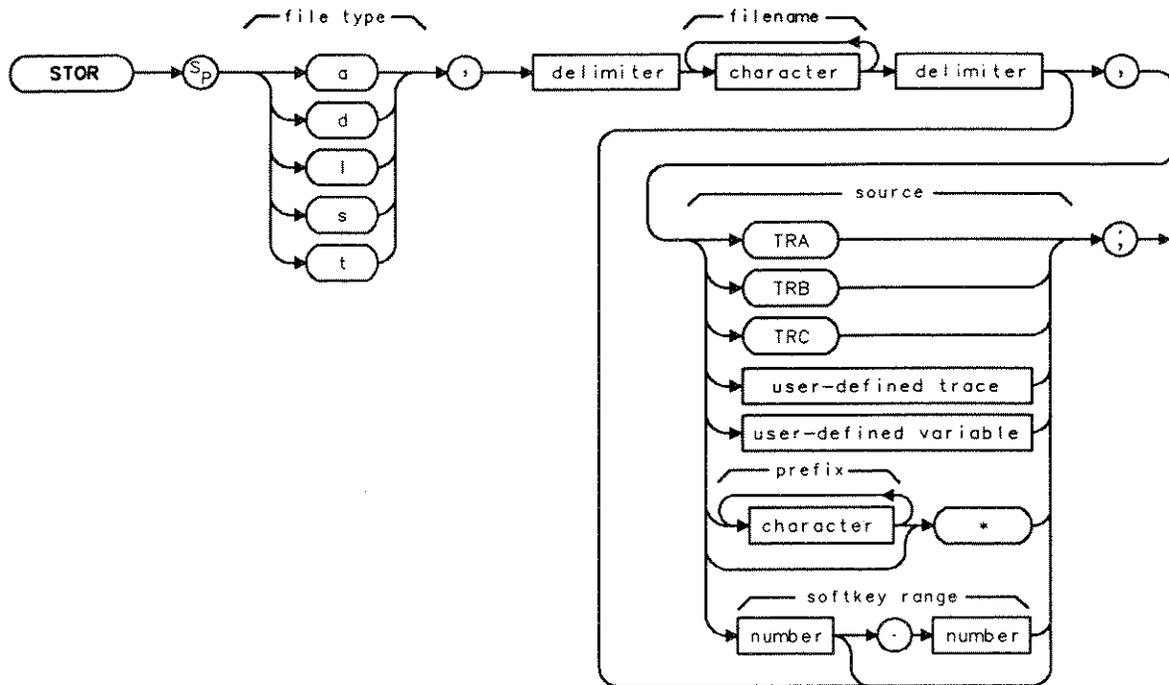


## STOR Store

The STOR command stores an individual function on the memory card.

**Option Required: Option 003 with HP 8590B or HP 8592B**

### Syntax



| Item                  | Description/Default                                                                                                | Range                                                                                                                                                                                                           |
|-----------------------|--------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| User-defined trace    | A trace defined by the TRDEF command.                                                                              | Any valid trace name.                                                                                                                                                                                           |
| User-defined variable | A variable defined by VARDEF or ACTDEF commands.                                                                   | Any valid variable name.                                                                                                                                                                                        |
| Character (filename)  | Any valid character. Characters form the file type and filename.                                                   | File type (lowercase a, c, d, l, s, or t) should precede the filename. Filename is 0 to 6 characters long, A through Z and the underscore (the underscore should be used as the second character of the label). |
| Character (prefix)    | Any valid character.                                                                                               |                                                                                                                                                                                                                 |
| Number                | A valid softkey number range. Use a decimal point to separate the softkey numbers when specifying a softkey range. | 1 to 6, 601 to 1200.                                                                                                                                                                                            |
| Delimiter             | Matching characters marking the beginning and end of the list of analyzer commands.                                | !"\$%&'/:=@                                                                                                                                                                                                     |

Related Commands: CAT, LOAD, PREFX.

### Example

```
OUTPUT 718;"TRDEF N_EW,400;"
OUTPUT 718;"MOV N_EW,TRA;"
OUTPUT 718;"STOR t,%tFUNCX%,N_EW;"
```

*Defines a trace.*

*Moves the contents of trace A into N\_EW.*

*Stores N\_EW under the filename of tFUNCX as trace data.*

```
OUTPUT 718;"STOR d,%dPROG%,*;"
```

*Stores all downloadable programs in the analyzer memory on the memory card.*

```
OUTPUT 718;"STOR a,%aAMPDATA%;"
```

*Stores amplitude correction factors.*

```
OUTPUT 718;"STOR l,%lLIMITS%;"
```

*Stores limit-line tables.*

```
OUTPUT 718;"STOR s,%sSTATE%;"
```

*Stores the analyzer state.*

```
OUTPUT 718;"STOR d,%dKEYS%,601.606;"
```

*Stores softkey functions 601 through 606 in the file dKEYS.*

```
OUTPUT 718;"STOR d,%dKEY%,601;"
```

*Stores softkey function 601 in the file dKEY.*

```
OUTPUT 718;"STOR d,%dFILES%,KEN*;"
```

*Stores all downloadable functions with the prefix "KEN" on the memory card.*

## STOR Store

### Description

The STOR command stores the source file on the memory card under the specified filename and file type.

---

### Note



The STOR saves data on the memory card. See "SAVET," "SAVES," or "SAVRCLN" to save data in analyzer memory.

---

Use trace A, trace B, trace C, or user-defined trace when storing trace data. There are several ways to store downloadable programs. For storing all downloadable programs in analyzer memory on the memory card, use an asterisk as the source. For storing all downloadable with a certain prefix, use an asterisk and the prefix as the source. For storing a range of softkey functions, specify the softkey numbers separated with a decimal. Use the downloadable file type when storing user-defined variables.

| File Type | Description              |
|-----------|--------------------------|
| a         | Amplitude factors.       |
| d         | Downloadable program.    |
| l         | Limit-line table(s).     |
| s         | Machine state.           |
| t         | Trace and machine state. |

---

### Note



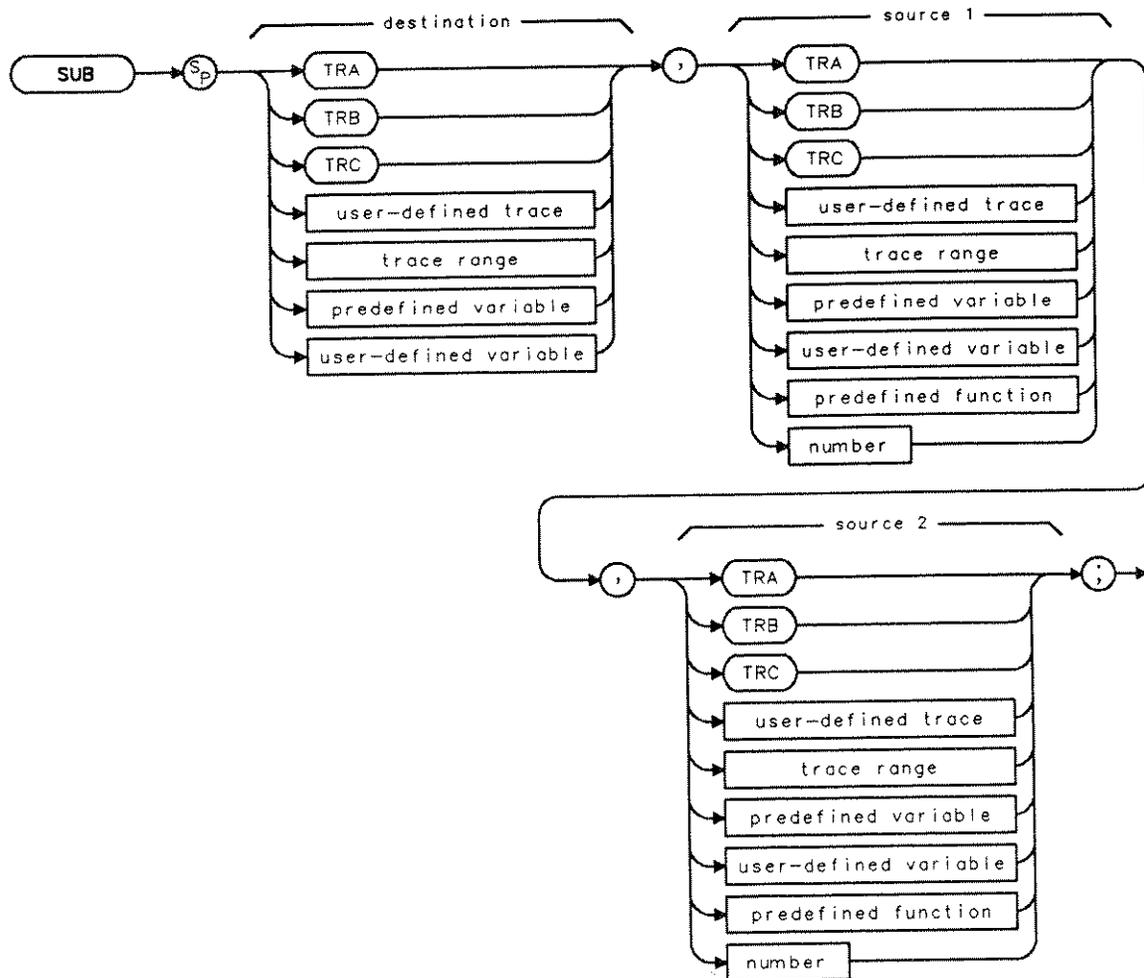
Use the file type as the first character of the filename in order to catalog it by the file type. For example, use tFUNCX instead of FUNCX to catalog it by traces. If the file type is not specified as the first character of the filename, the file is stored as an ASCII file. Use the correct file type to load the file into analyzer memory correctly.

---

## SUB Subtract

The SUB command subtracts source 2 from source 1, point by point, and sends the difference to the destination.

### Syntax



**SUB Subtract**

| Item                  | Description/Default                                              | Range                    |
|-----------------------|------------------------------------------------------------------|--------------------------|
| User-defined trace    | A trace defined by the TRDEF command.                            | Any valid trace name.    |
| User-defined variable | A variable defined by VARDEF or ACTDEF commands.                 | Any valid variable name. |
| Predefined variable   | A command that acts as a variable. Refer to Table 4-1.           |                          |
| Predefined function   | Function that returns a value. Refer to Table 4-1.               |                          |
| Trace range           | A segment of trace A, trace B, trace C, or a user-defined trace. |                          |
| Number                | Any real or integer number.                                      | Real number range.       |

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.

Related Commands: AMB, BML, LIMITEST, SUM.

**Example**

OUTPUT 718; "SUB TRA,TRB,TRC;" *Subtracts trace C from trace B and places the result in trace A.*

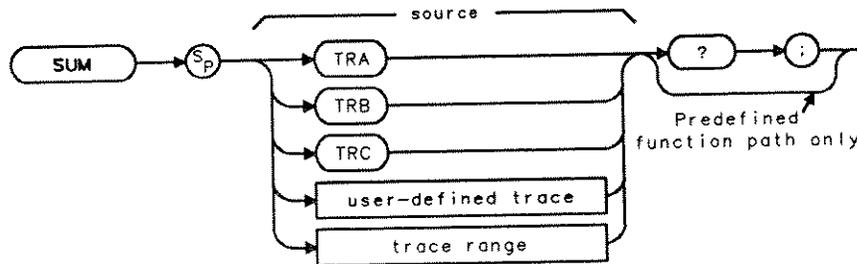
**Description**

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.

## SUM Sum of Trace Amplitudes

The SUM command returns the sum of the amplitudes of the trace elements in measurement units.

### Syntax



| Item               | Description/Default                                              | Range                 |
|--------------------|------------------------------------------------------------------|-----------------------|
| User-defined trace | A trace defined by the TRDEF command.                            | Any valid trace name. |
| Trace range        | A segment of trace A, trace B, trace C, or a user-defined trace. |                       |

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: ADD, DIV, MPY, SUB, TS.

### Example

|                                       |                                                                        |
|---------------------------------------|------------------------------------------------------------------------|
| 10 OUTPUT 718;"IP;"                   | <i>Initializes analyzer.</i>                                           |
| 20 OUTPUT 718;"SNGLS;TS;"             | <i>Activates single-sweep mode, takes sweep.</i>                       |
| 30 OUTPUT 718;"SUM TRA?;"             | <i>Gets the result.</i>                                                |
| 40 ENTER 718;Trace_sum                | <i>Puts the analyzer response in the computer variable, Trace_sum.</i> |
| 50 DISP Trace_sum;"MEASUREMENT UNITS" | <i>Displays the result.</i>                                            |
| 60 END                                |                                                                        |

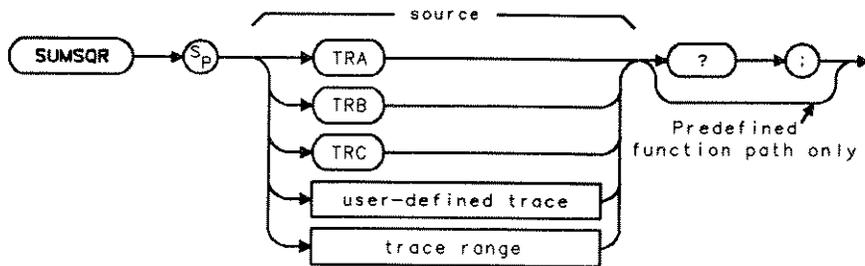
### Query Response



## SUMSQR Sum of Squared Trace Amplitudes

The SUMSQR command returns the sum of the squares of the amplitude of each trace element in measurement units.

### Syntax



| Item               | Description/Default                                              | Range                 |
|--------------------|------------------------------------------------------------------|-----------------------|
| User-defined trace | A trace defined by the TRDEF command.                            | Any valid trace name. |
| Trace range        | A segment of trace A, trace B, trace C, or a user-defined trace. |                       |

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

### Example

|                                       |                                                                           |
|---------------------------------------|---------------------------------------------------------------------------|
| OUTPUT 718;"IP;"                      | <i>Initializes analyzer.</i>                                              |
| OUTPUT 718;"SNGLS;TS;"                | <i>Activates single-sweep mode, takes sweep.</i>                          |
| OUTPUT 718;"SUMSQR TRA?;"             | <i>Gets the result.</i>                                                   |
| ENTER 718;Trace_sqrsum                | <i>Puts the analyzer response in the computer variable, Trace_sqrsum.</i> |
| DISP Trace_sqrsum;"MEASUREMENT UNITS" | <i>Displays the result.</i>                                               |

### Query Response

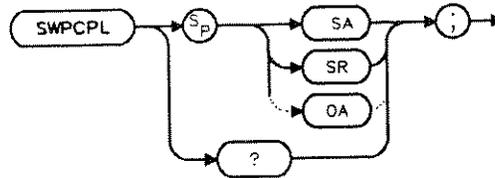


## SWPCPL Sweep Couple

The SWPCPL command selects either a stimulus-response (SR) or spectrum-analyzer (SA) auto-coupled sweep time.

**Option and Model Required:** Option 010 or 011 with HP 8590B or HP 8591A

### Syntax



Preset State: SWPCPL SA.  
Related Commands: SRCPWR.

### Example

```

10 OUTPUT 718;"IP;SNGLS;"
20 OUTPUT 718;"FA 300KHZ;FB 1GHZ;"
30 OUTPUT 718;"SRCPWR -10DB;"
40 OUTPUT 718;"SWPCPL SR;"
50 OUTPUT 718;"SRCTKPK;DONE?;"
60 ENTER 718;Done
70 LOCAL 718
80 END

```

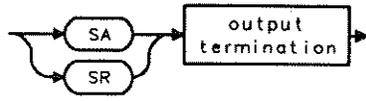
### Description

In stimulus-response mode, auto-coupled sweep times are usually much faster for swept-response measurements. Stimulus-response auto-coupled sweep times are typically valid in stimulus-response measurements when the system's frequency span is less than 20 times the bandwidth of the device under test.

The functions of SWPCPL and **SWP CPLG SR SA** are identical.

**SWPCPL Sweep Couple**

**Query Response**



## TA Transfer A

The TA command returns trace A amplitude values from the analyzer to the controller.

### Syntax



Related Commands: MDS, TB, TDF.

### Example

This example stores the TA results in array A.

```

DIM A(401)
OUTPUT 718;"IP;"
OUTPUT 718;"SNGLS;CF 300MHZ;SP 2MHZ;TS;"

OUTPUT 718;"TDF P;TA;"

FOR N = 1 TO 401

  ENTER 718;A(N)
NEXT N
FOR N = 1 TO 401

  PRINT A(N)
NEXT N
  
```

*Reserves memory area for array.*

*Initializes analyzer.*

*Activates single-sweep mode, changes center frequency, span, takes sweep.*

*Changes trace data format, outputs trace A.*

*FOR NEXT loop moves each element of trace A to the computer.*

*FOR NEXT loop moves the trace values from the computer to the printer.*

*Prints out the results.*

### Description

The display unit values are transferred in sequential order (from left to right) as seen on the screen.

Transfer of trace amplitude data should be done only as follows:

1. Select single sweep mode (SNGLS).
2. Select desired analyzer settings.
3. Take one complete sweep (TS).
4. Transfer data (TA).

This procedure ensures that the current settings of the analyzer are reflected in the transferred data.

See Chapter 3, "Trace Data Transfers," for more information.

TA Transfer A

---

**Note**

Items are separated by CR/LF when in TDF P format.



## TB Transfer B

The TB command transfers trace B amplitude values from the analyzer to the controller.

### Syntax



Related Commands: MDS, TA, TDF.

### Example

```

DIM A(401)
OUTPUT 718;"IP;"
OUTPUT 718;"SNGLS;CF 300MHZ;SP 2MHZ;TS;"

OUTPUT 718;"TDF P;TB;"

FOR N = 1 TO 401

  ENTER 718;A(N)
NEXT N
FOR N = 1 TO 401

  PRINT A(N)
NEXT N
  
```

*Reserves memory area for array.  
Initializes analyzer.  
Activates single-sweep mode, changes center frequency, span, takes sweep.  
Changes trace data format, outputs trace B.  
FOR NEXT loop moves each element of trace A to the computer.*

*FOR NEXT loop moves the trace values from the computer to the printer.  
Prints out the results.*

### Description

The operation of TB is similar to the operation of TA.

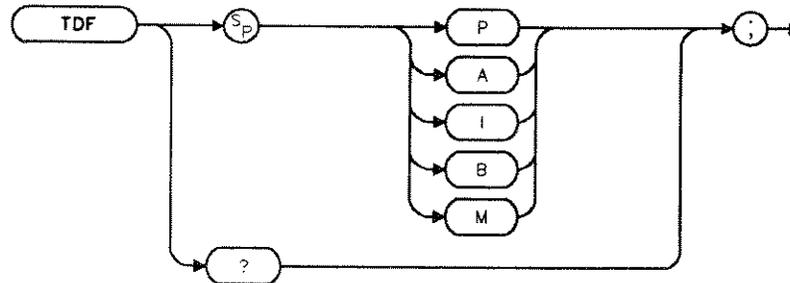
See Chapter 3, "Trace Data Transfers," for more information.

## TDF

### Trace Data Format

The TDF command formats trace information for return to the controller.

#### Syntax



Related Commands: MDS, MKA, TA, TB, TRA.

#### Example

|                                                       |                                                                     |
|-------------------------------------------------------|---------------------------------------------------------------------|
| <code>DIM A(401)</code>                               | <i>Holds trace data.</i>                                            |
| <code>OUTPUT 718;"IP;"</code>                         | <i>Initializes analyzer.</i>                                        |
| <code>OUTPUT 718;"BLANK TRA;CLRWB TRB;"</code>        | <i>Views trace B.</i>                                               |
| <code>OUTPUT 718;"SNGLS;CF 300MHZ;SP 2MHZ;TS;"</code> | <i>Activates single-sweep mode, changes center frequency, span.</i> |
| <code>OUTPUT 718;"TDF P;TB;"</code>                   | <i>Formats trace data.</i>                                          |
| <code>FOR N = 1 TO 401</code>                         | <i>Transfers trace data to array A, one element at a time.</i>      |
| <code>ENTER 718;A(N)</code>                           |                                                                     |
| <code>NEXT N</code>                                   |                                                                     |
| <code>FOR N = 1 TO 401</code>                         | <i>Loop prints out trace B data.</i>                                |
| <code>PRINT A(N)</code>                               | <i>Prints out the results.</i>                                      |
| <code>NEXT N</code>                                   |                                                                     |

#### Description

The different trace data formats are as follows:

M, measurement units, returns values from  $-32768$  to  $+32767$ .

P, parameter units, returns absolute values such as dBm or Hz.

A returns data as an A-block data field. The MDS command determines whether data comprises one or two 8-bit bytes.

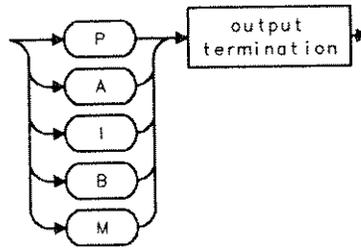
I returns data as an I-block data field. The MDS command determines whether data comprises one or two 8-bit bytes.

**B** enables binary format. The MDS command determines whether data comprises one or two 8-bit bytes.

**A**, **B**, **I**, and **M** are defined in the analyzer internal amplitude units (log: hundredths of dBm; linear: 8,000 = top of screen and 0 = bottom of screen). **P** is in the current parameter unit specified by AUNITS.

Refer to Chapter 3, "Trace Data Transfers," for more information.

### Query Response



## TEXT

### Text

The TEXT command writes text on the analyzer screen at the current pen position.

### Syntax



| Item      | Description/Default                                                                 | Range       |
|-----------|-------------------------------------------------------------------------------------|-------------|
| Delimiter | Matching characters marking the beginning and end of the list of analyzer commands. | !"\$%&'/:=@ |
| Character | Any valid character. See Table 4-6 for additional characters available.             |             |

Related Commands: DSPLY, LB, PA, PD, PU, PR.

### Example

OUTPUT 718;"PU;PA 80,80;TEXT%CONNECT ANTENNA%;" *Displays CONNECT ANTENNA on the analyzer screen.*

OUTPUT 718;"PU;PA 100,100;TEXT%50";CHR\$(250);"%;" *Displays 50Ω on the analyzer screen.*

### Description

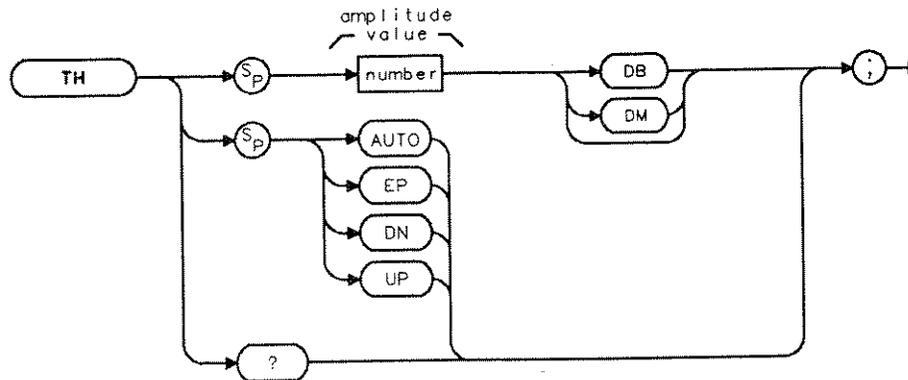
The TEXT origin is at the lower-left corner of the first character. The pen is placed to the right of and behind, the last character position after the text characters. Line feeds are not automatically generated for lines that extend past the edge of the screen.

See the label (LB) command and Table 4-6 for the additional characters available by specifying the ASCII character code. (See second line of example.)

## TH Threshold

The TH command clips signal responses below the threshold level.

### Syntax



| Item   | Description/Default                              | Range                          |
|--------|--------------------------------------------------|--------------------------------|
| Number | Any real or integer number. Default unit is dBm. | Range dependent on RL setting. |

Preset State: Clip off, positioned one division above bottom graticule line.

Step Increment: One division.

Related Commands: AUTO, DL, MKPK, PEAKS, RL.

### Example

OUTPUT 718;"TH UP;" *Increases the threshold level.*

### Description

The TH command is similar to a baseline clipper. The threshold level is eight graticule divisions below the top of the screen unless otherwise specified. The threshold level is annotated in reference level units at the lower-left corner of the analyzer screen. AUTO deactivates clipping. The TH level is used for next peak marker movements (see "MKPK") and the PEAKS command even if the display clipping is off.

The function of the TH command and **THRESHLD ON OFF** are identical.

TH Threshold

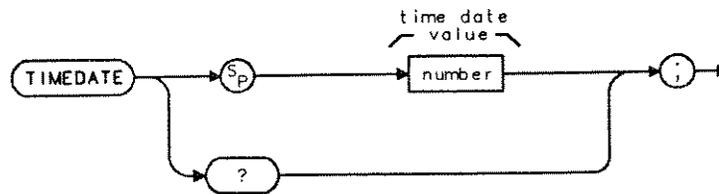
Query Response



## TIMEDATE Time Date

The TIMEDATE command allows you to set the time and date for the analyzer real-time clock in the YYMMDDHHMMSS format.

### Syntax



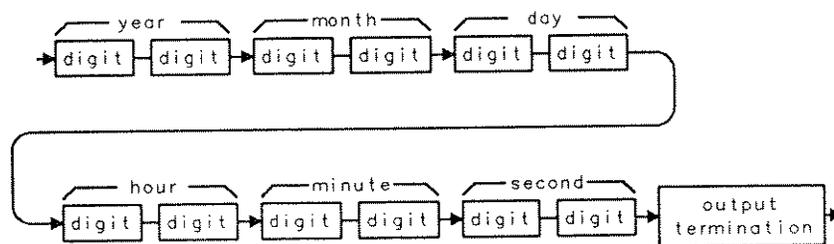
| Item   | Description/Default                                                           | Range                  |
|--------|-------------------------------------------------------------------------------|------------------------|
| Number | A number representing the date and time in the YYMMDDHHMMSS (24 hour) format. | A valid date and time. |

Related Commands: SETDATE, SETTIME, TIMEDSP.

### Example

OUTPUT 718;"TIMEDATE 881231135501;" *Sets the analyzer time and date to 1:55:01 PM on December 31, 1988.*

### Query Response

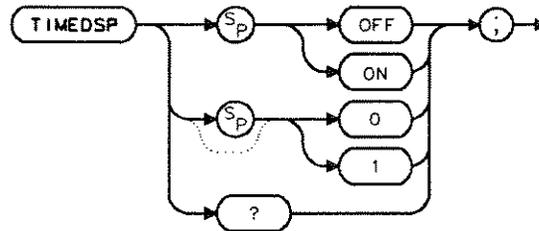


## TIMEDSP

### Time Display

The TIMEDSP command enables the display of the time and date on the analyzer screen.

#### Syntax



Related Commands: ANNOT, SETDATE, SETTIME, TIMEDATE.

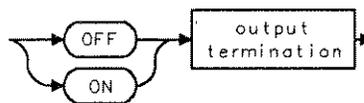
#### Example

```
OUTPUT 718;"TIMEDSP OFF;"
```

#### Description

The functions of TIMEDSP and **TIMEDATE ON OFF** are identical.

#### Query Response

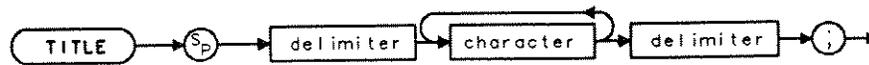


## TITLE

### Title

The TITLE command activates the screen title mode. The title is displayed above the top graticule and is left justified.

### Syntax



| Item      | Description/Default                                                                   | Range                |
|-----------|---------------------------------------------------------------------------------------|----------------------|
| Delimiter | Matching characters marking the beginning and end of the list of analyzer command(s). | !"\$%&'/:=@          |
| Character | Any valid character. See Table 4-6 for additional characters available.               | Up to 53 characters. |

Related Commands: IP, LB, SAVES, TEXT.

### Example

OUTPUT 718;"TITLE %ADJUST ANTENNA%;" *Displays ADJUST ANTENNA on the analyzer screen.*

### Description

This function writes a message at the top of the analyzer screen. The full width of the display is available for writing a maximum of 53 characters. However, the marker readout may interfere with the last 26 characters. IP removes the message.

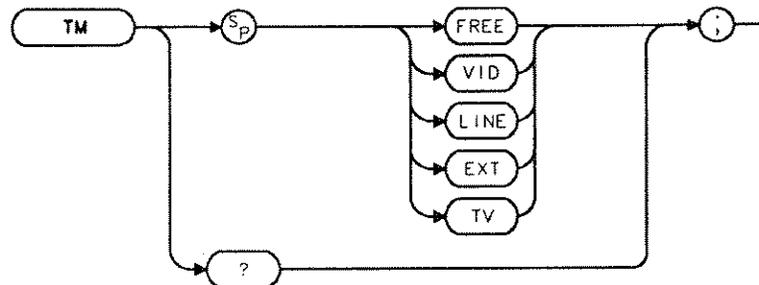
The screen title is saved in the trace register with the SAVET command.

The functions of TITLE and CHANGE TITLE are similar.

## TM Trigger Mode

The TM command selects a trigger mode: free, line, video, TV, or external.

### Syntax



Related Commands: DL.

### Example

OUTPUT 718;"TM EXT;" *Activates the external trigger mode.*

### Description

The conditions of the four trigger modes are as follows:

- FREE** allows the next sweep to start as soon as possible after the last sweep. The functions of TM FREE and **FREE** are identical.
- VID** allows the next sweep to start if the detected RF envelope voltage rises across a level set by the display line. The functions of TM VID and **VIDEO** are identical.
- LINE** allows the next sweep to start when the line voltage passes through zero, going positive. The functions of TM LINE and **LINE** are identical.
- EXT** allows the next sweep to start when an external voltage level passes through approximately 1.5 V, going positive. The external trigger signal level must be between 0 V and +5 V. Connect the external trigger to the EXT TRIG INPUT. The functions of TM EXT and **EXTERNAL** are identical.
- TV** allows TV triggering if Options 101 and 102 are installed (HP 8591A and HP 8593A only). The functions of TM TV and **TV TRIG** are similar. TM TV does not select the TV line number, set up the amplitude level, change the span, change the bandwidth, or change the sweep time.

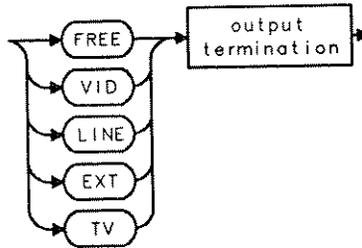
---

**Note**      Some analyzer functions are not performed until the analyzer is triggered.

---



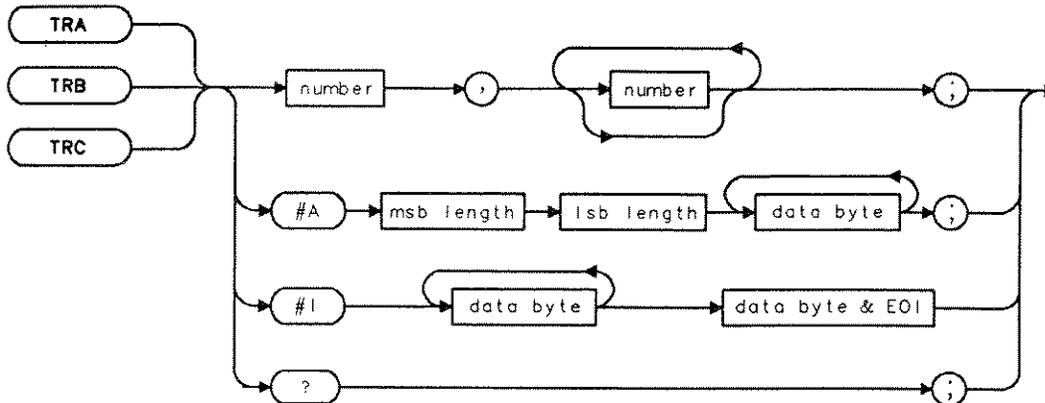
**Query Response**



## TRA/TRB/TRC Trace Data Input/Output

The TRA/TRB/TRC commands provide a method for returning or storing 16-bit trace values.

### Syntax



| Item            | Description/Default                                                                       | Range                |
|-----------------|-------------------------------------------------------------------------------------------|----------------------|
| Number          | Any real or integer number.                                                               | Integer number range |
| Msb length      | Most significant byte of a two-byte word that describes the number of bytes transmitted.  |                      |
| Lsb length      | Least significant byte of a two-byte word that describes the number of bytes transmitted. |                      |
| Data byte       | 8-bit byte containing numeric or character data.                                          |                      |
| Data byte & EOI | 8-bit byte containing numeric or character data followed by END.                          |                      |

Amplitude Range HP 8590B or HP 8591A: -139.9 to +50 dBm.

Amplitude Range HP 8592B or HP 8593A: -139.9 to +60 dBm.

Related Commands: LOAD, ONEOS, RCLT, SAVET, STOR, TDF.

### Example

|                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> 10 REAL Trace_a(1:401) 20 OUTPUT 718;"IP;" 30 OUTPUT 718;"TDF P;" 40 OUTPUT 718;"SNGLS;" 50 OUTPUT 718;"CF 300MHZ;" 60 OUTPUT 718;"SP 200MHZ;" 70 OUTPUT 718;"TS;" 80 OUTPUT 718;"MKPK HI;" 90 OUTPUT 718;"MKCF;" 100 OUTPUT 718;"TS;" 110 OUTPUT 718;"TRA?;" 120 ENTER 718;Trace_a(*) 130 OUTPUT 718;"CONTS;" 140 END                 </pre> | <p><i>Creates a 401-point trace array.</i></p> <p><i>Initializes analyzer.</i></p> <p><i>Changes the format for real numbers.</i></p> <p><i>Sets up analyzer for desired trace output.</i></p><br><p><i>Moves peak to center of analyzer screen.</i></p> <p><i>Updates measurement trace.</i></p> <p><i>Gets the trace data.</i></p> <p><i>Sends the trace data to the computer.</i></p> <p><i>Activates continuous sweep mode.</i></p> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

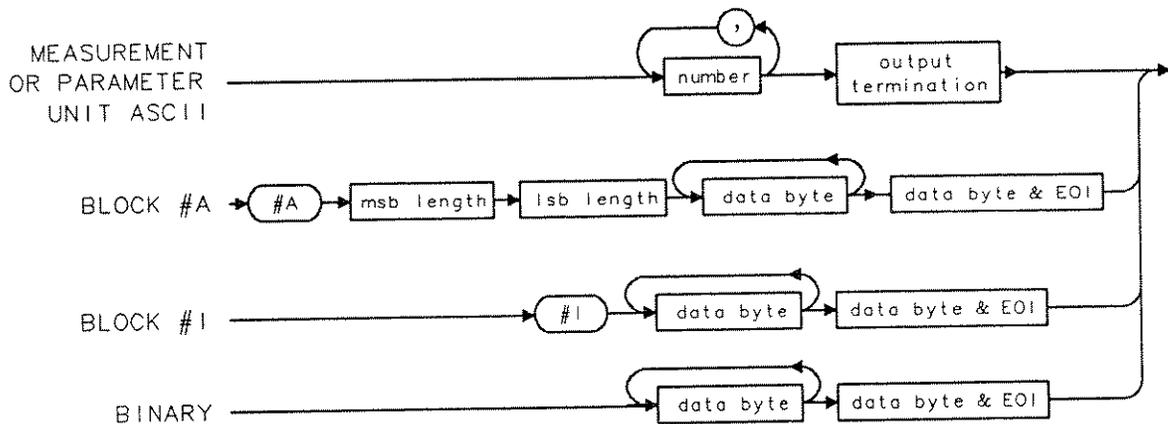
### Description

Input in A-block or I-block format is treated as measurement units independent of trace data format (TDF). Enter words in measurement units only. The output format is specified according to TDF and MDS.

The command may be used to input integer data to traces. See "Saving Trace Data" in Chapter 3. Since the lengths of trace A, trace B, and trace C are fixed, there are always 401 or 802 bytes transferred during binary input or binary output mode, respectively.

### Query Response

The form of the query response is dependent upon the previously used TDF and MDS commands as follows:

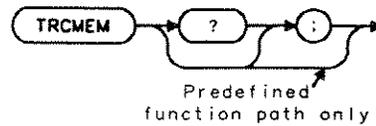


## TRCMEM

### Trace Memory

The TRCMEM command returns a nonnegative integer that indicates the total number of trace registers available for SAVET and RCLT.

### Syntax

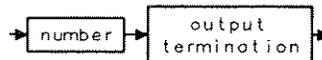


Related Commands: ACTDEF, DISPOSE, FUNCDEF, RCLT, SAVET, TRDEF.

### Example

```
OUTPUT 718;"TRCMEM?;"   Gets the total number of trace registers.  
ENTER 718;Number  
DISP Number
```

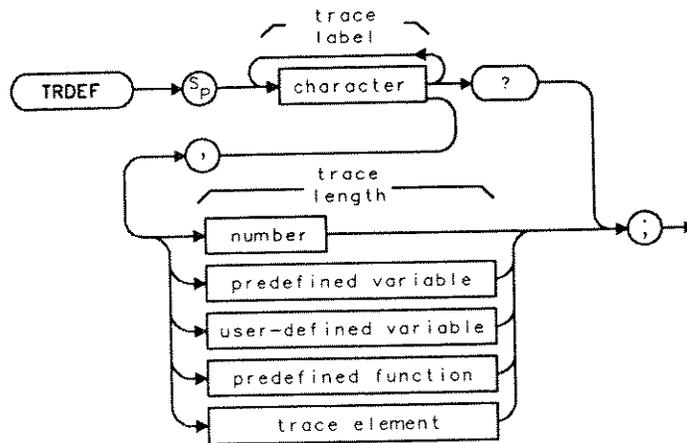
### Query Response



## TRDEF Trace Define

The TRDEF command creates a user-defined trace.

### Syntax



| Item                  | Description/Default                                               | Range                                                                                                                         |
|-----------------------|-------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Character             | Any valid character.                                              | 2 to 11 characters long, A through Z and the underscore (the underscore should be used as the second character of the label). |
| User-defined variable | A variable defined by VARDEF or ACTDEF commands.                  | Any valid variable name.                                                                                                      |
| Predefined variable   | A command that acts as a variable. Refer to Table 4-1.            |                                                                                                                               |
| Predefined function   | Function that returns a value. Refer to Table 4-1.                |                                                                                                                               |
| Trace element         | An element of trace A, trace B, trace C, or a user-defined trace. |                                                                                                                               |
| Number                | Any real or integer number.                                       | 2 to 2047.                                                                                                                    |

Parameter Value : 2 to 2047.

Prerequisite Commands: ACTDEF or VARDEF when using a user-defined variable.

Related Commands: DISPOSE.

**TRDEF Trace Define****Example**

OUTPUT 718;"TRDEF NEW,100;" *Defines a trace called NEW.*

**Description**

The TRDEF command defines a trace and the number of points the trace will contain.

**Query Response**

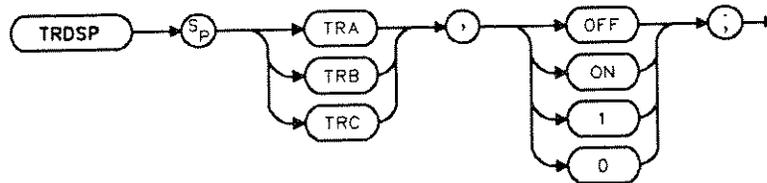
The query response returns the number of trace elements in the trace.



## TRDSP Trace Display

The TRDSP command controls the display of trace A, B, or C without clearing the trace (measurements can still be taken).

### Syntax



Related Commands: TRPRST, TRSTAT.

### Example

```
OUTPUT 718;"TRDSP TRA,OFF;"
```

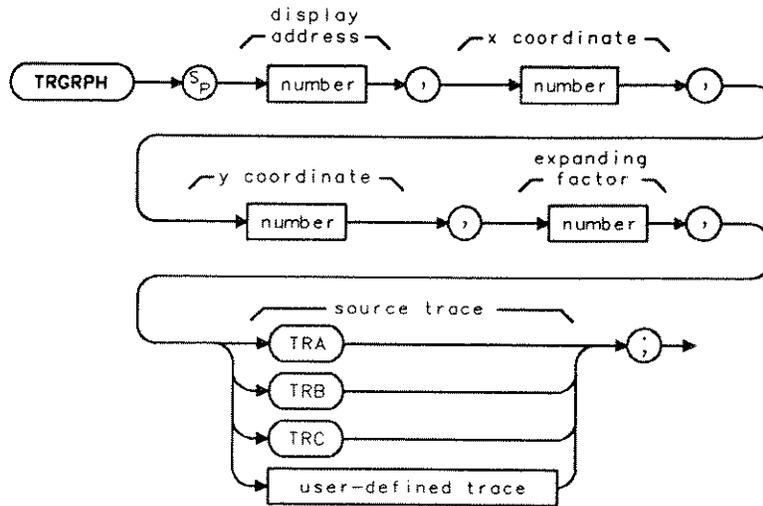
### Description

TRDSP OFF reduces the time duration between sweeps.

## TRGRPH Trace Graph

The TRGRPH command displays a compressed trace on the analyzer display.

### Syntax



| Item                      | Description/Default                   | Range                                                                         |
|---------------------------|---------------------------------------|-------------------------------------------------------------------------------|
| Number                    | Any valid integer.                    | 0 to 4000 for the <i>x</i> coordinate, 0 to 8000 for the <i>y</i> coordinate. |
| Number (expanding factor) | Any valid integer.                    | 0 to 100.                                                                     |
| User-defined trace        | A trace defined by the TRDEF command. | Any valid trace name.                                                         |

Prerequisite Commands: TRDEF when using a user-defined trace.

Related Commands: COMPRESS, DSPLY.

**Example**

|                                    |                                                                                               |
|------------------------------------|-----------------------------------------------------------------------------------------------|
| OUTPUT 718;"IP;"                   | <i>Initializes analyzer.</i>                                                                  |
| OUTPUT 718;"TRDEF NEW,100;"        | <i>Defines a trace called NEW with 100 trace elements.</i>                                    |
| OUTPUT 718;"CF 300MHZ;SNGLS;"      | <i>Changes the center frequency, activates single-sweep mode.</i>                             |
| OUTPUT 718;"TS;"                   | <i>Takes sweep.</i>                                                                           |
| OUTPUT 718;"COMPRESS NEW,TRA,POS;" | <i>Compresses the contents of trace A into trace NEW.</i>                                     |
| OUTPUT 718;"BLANK TRA;"            |                                                                                               |
| OUTPUT 718;"TRGRPH 0,0,400,4,NEW;" | <i>Redraws trace NEW 400 measurement units above the baseline, expanded by a factor of 4.</i> |

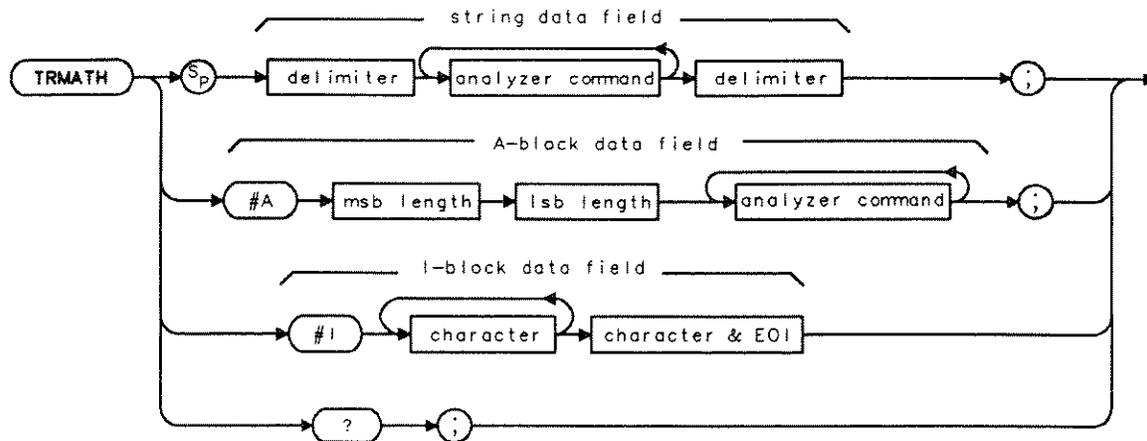
**Description**

The  $x$  and  $y$  coordinates specify the position of the first trace element. Each horizontal division represents 40  $x$  coordinates. Each vertical division represents 1000  $y$  coordinates. The display address is inoperative; it is specified for backward compatibility only. The trace can be expanded according to the scale determined by the expanding factor.

## TRMATH Trace Math

The TRMATH command executes a command list at the end of each sweep.

### Syntax



| Item             | Description/Default                                                                       | Range       |
|------------------|-------------------------------------------------------------------------------------------|-------------|
| Delimiter        | Matching characters marking the beginning and end of the list of analyzer command(s).     | !"\$%&'/:=@ |
| Analyzer Command | Any spectrum analyzer command.                                                            |             |
| Msb length       | Most significant byte of a two-byte word that describes the number of bytes transmitted.  |             |
| Lsb length       | Least significant byte of a two-byte word that describes the number of bytes transmitted. |             |
| Character        | Any valid character.                                                                      |             |
| Character & EOI  | Any valid character and END.                                                              |             |

Related Commands: ONDELAY, ONEOS, ONMKR, ONSRQ, ONSWP, ONTIME.

### Example

The program below halves the amplitude of trace A and moves it to trace B.

|                                       |                                                        |
|---------------------------------------|--------------------------------------------------------|
| OUTPUT 718;"CLRW TRA;VIEW TRB;"       | <i>Views trace B.</i>                                  |
| OUTPUT 718;"DISPOSE TRMATH;"          | <i>Disposes of existing TRMATH commands.</i>           |
| OUTPUT 718;"TRMATH! DIV TRB,TRA,2! ;" | <i>Divides trace A by 2 and moves it into trace B.</i> |

### Description

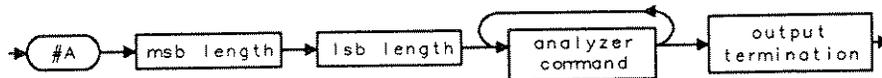
The TRMATH command executes a command list at the end of each sweep. Any analyzer command except TS is allowed within the command list.

The operands and results of trace math are truncated if they are not within certain limits. If operating on traces A, B, or C, trace lengths must be less than or equal to 401. If operating on user-defined traces, results must be less than or equal to 2047.

If an ONEOS command is encountered, it is executed after the contents of the TRMATH command are executed.

Instrument preset (IP) clears TRMATH functions.

### Query Response



## TRPRST

### Trace Preset

The TRPRST command sets the trace operations to their preset values (see "IP").

#### Syntax



Related Commands: AMB, BLANK, CLRDSP, CLRW, DISPOSE, DL, IP, TH.

#### Example

```
OUTPUT 718;"TRPRST;"
```

#### Description

TRPRST executes these commands:

```
CLRW TRA  
BLANK TRB  
BLANK TRC  
DET AUTO  
AMB OFF  
AMBPL OFF  
VAVG OFF  
TRDSP TRA, ON  
TRDSP TRB, OFF  
TRDSP TRC, OFF
```

## TRSTAT Trace Status

The TRSTAT command returns the status of traces A, B, and C: clear write, blank, view, minimum hold, or maximum hold.

### Syntax



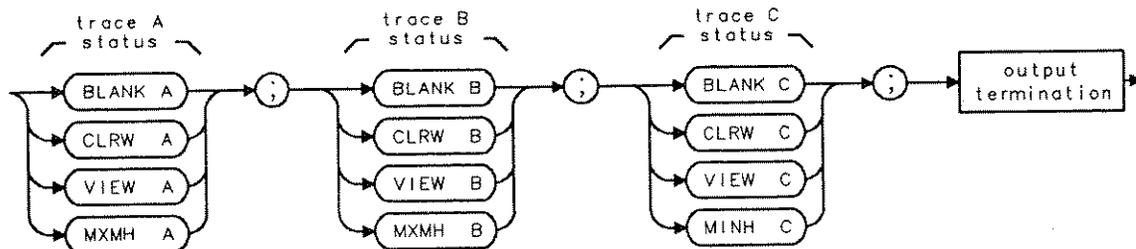
Related Commands: BLANK, CLRW, DET, MINH, TRDSP, VIEW.

### Example

This example returns the measurement state of traces A, B, and C.

|                                            |                                                    |
|--------------------------------------------|----------------------------------------------------|
| <code>DIM States\$[40]</code>              | <i>Declares array for results.</i>                 |
| <code>OUTPUT 718;"TRSTAT?;"</code>         | <i>Returns the status results to the computer.</i> |
| <code>ENTER 718 USING "-K";States\$</code> |                                                    |
| <code>PRINT States\$</code>                | <i>Prints out status of traces.</i>                |

### Query Response



---

## TS

### Take Sweep

The TS command starts and completes one full sweep before the next command is executed.

#### Syntax



Related Commands: SNGLS, TM.

#### Example

```
OUTPUT 718;"SNGLS;TS;" Activate the single-sweep mode, take sweep.
```

#### Description

A take sweep is required for each sweep in the single-sweep mode. TS prevents further input from the interface bus until the sweep is completed to allow synchronization with other instruments.

In the example below, the command sequence does not allow sufficient time for a full sweep of the specified span before VIEW is executed. Therefore, only the span set by the instrument is displayed in trace A.

```
OUTPUT 718;"IP;CF 400MHZ;SP 20KHZ;VIEW TRA;"
```

A TS command inserted before VIEW makes the analyzer take one complete sweep before displaying trace A. This allows the analyzer sufficient time to respond to each command in the sequence.

```
OUTPUT 718;"IP;CF 400MHZ;SP 20MHZ;TS;VIEW TRA;"
```

TS is recommended before transmission of marker data and before executing marker operations such as peak search. This is because the active marker is repositioned at the end of each sweep. When the analyzer receives a TS command, it is not ready to receive any more data until one full sweep has been completed. However, when slow sweep speeds are being used, the controller can be programmed to perform computations or address other instruments while the analyzer completes its sweep.

On-event commands do not interrupt a take sweep.

---

#### Note



When MKPAUSE or MKSTOP are activated, TS considers the sweep complete when it reaches the active marker.

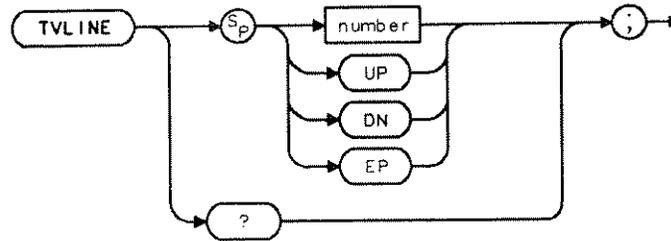
---

## TVLINE TV Line

The TVLINE command sets the line number of the horizontal line of video on which to trigger.

**Option and Model Required:** Options 101 and 102 with HP 8591A or HP 8593A

### Syntax



| Item   | Description/Default                            | Range     |
|--------|------------------------------------------------|-----------|
| Number | Any valid integer number. Default value is 17. | 1 to 101. |

Preset Value: 17.

Related Commands: HAVE, TVSFRM, TVSTND.

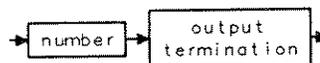
### Example

```
OUTPUT 718;"TVLINE 20;"
```

### Description

The functions of TVLINE and TV LINE \* are identical.

### Query Response

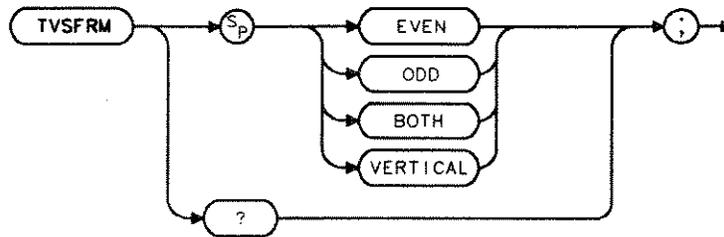


## TVSFRM TV Frame

The TVSFRM command selects the type of video frame to trigger on.

**Option and Model Required:** Options 101 and 102 with HP 8591A or HP 8593A

### Syntax



Related Commands: HAVE, TVLINE, TVSYNC.

### Example

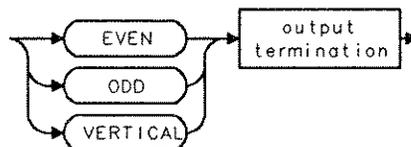
```
OUTPUT 718;"TVSFRM BOTH;"
```

### Description

The ODD and EVEN trigger are for interlaced formats, and VERTICAL is for noninterlaced display formats.

The functions of TVSFRM ODD, TVSFRM EVEN, and TVSFRM BOTH are identical to **TV TRIG ODD FLD**, **TV TRIG EVEN FLD**, and **TV TRIG VERT INT**, respectively. TVSFRM BOTH is the same as TVSFRM VERTICAL.

### Query Response

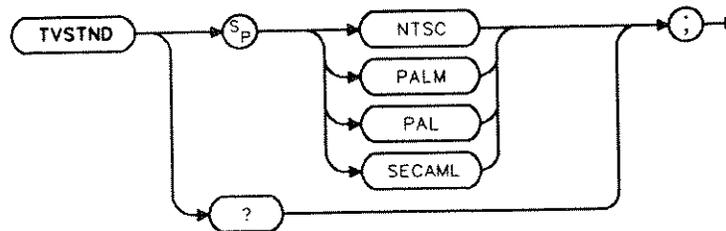


## TVSTND TV Standard

The TVSTND command selects the triggering for NTSC, PAL, PAL-M, and SECAM-L formats.

**Option and Model Required:** Options 101 and 102 with HP 8591A or HP 8593A

### Syntax



Related Commands: TM, TVLINE, TVSYNC.

### Example

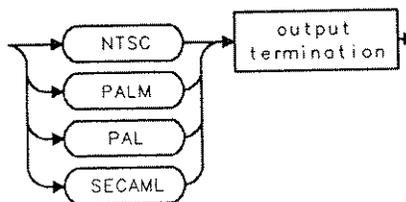
```
OUTPUT 718;"TVSTND PAL;"
```

### Description

TVSTND sets corrections for TVLINE for each standard. TVSTND sets the type of video frame to trigger on (positive or negative) automatically; it is necessary to use TVSYNC after using TVSTND only if you require a non-standard trigger.

The functions of TVSTND and TVSTND are similar.

### Query Response

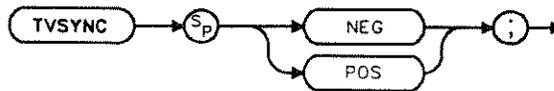


## TVSYNC TV Sync

The TVSYNC command selects the polarity of video modulation to trigger on.

**Option and Model Required:** Options 101 and 102 with HP 8591A or HP 8593A

### Syntax



Preset Value: TVSYNC NEG

Related Commands: HAVE, TVLINE, TVSFRM, TVSTND.

### Example

```
OUTPUT 718;"TVSYNC POS;"
```

### Description

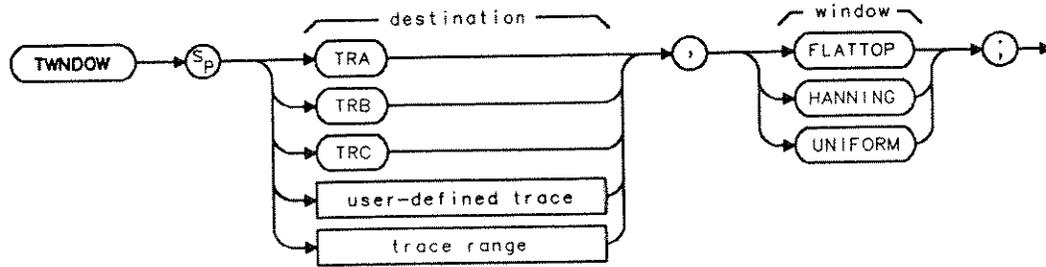
TVSYNC allows you to trigger on the negative or positive modulation video format. TVSTND changes the triggering polarity for the selected video format.

The functions of TVSYNC and `TV SYNC NEG POS` are identical.

## TWNDOW Trace Window

The TWNDOW command creates a window trace array for the fast Fourier transform (FFT) function.

### Syntax



| Item               | Description/Default                                             | Range                 |
|--------------------|-----------------------------------------------------------------|-----------------------|
| User-defined trace | A trace defined by the TRDEF command.                           | Any valid trace name. |
| Trace Range        | A segment of trace A, trace B, trace C, or a user-defined trace |                       |

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: FFT.

### Example

Connect calibrator signal to the analyzer input.

|                                  |                                                                                      |
|----------------------------------|--------------------------------------------------------------------------------------|
| OUTPUT 718;"IP;"                 | <i>Initializes analyzer.</i>                                                         |
| OUTPUT 718;"CF 300MHZ;"          | <i>Changes the center frequency.</i>                                                 |
| OUTPUT 718;"SP OHZ;ST 800MS;"    | <i>Changes span, sweep time.</i>                                                     |
| OUTPUT 718;"TRDEF NEW,401;"      | <i>Defines a trace called NEW.</i>                                                   |
| OUTPUT 718;"TWNDOW NEW,UNIFORM;" | <i>Trace NEW holds the window algorithm, UNIFORM.</i>                                |
| OUTPUT 718;"CLRWB TRB;"          |                                                                                      |
| OUTPUT 718;"SNGLS;TS;TS;"        | <i>Activates single-sweep mode and updates trace.</i>                                |
| OUTPUT 718;"FFT TRA,TRB,NEW;"    | <i>Performs fast Fourier transform on trace B and stores the results in trace A.</i> |
| OUTPUT 718;"BLANK TRB;"          |                                                                                      |
| OUTPUT 718;"VIEW TRA;"           | <i>Displays the result.</i>                                                          |

**TWINDOW Trace Window****Description**

The trace window function creates a trace array according to three built-in algorithms: UNIFORM, HANNING, or FLATTOP. When used with the FFT command, the three algorithms give resultant passband shapes that represent a give-and-take between amplitude uncertainty, sensitivity, and frequency resolution. See "FFT" for more information about these algorithms and the FFT function.

**UP****Up**

The UP command increases the value of the active function by the applicable step size.

**Syntax**

Related Commands: ACTDEF, AT, CF, DL, DN, FA, FB, LG, MKA, MKD, MKN, MKF, MKSS, RB, RL, SP, SS, ST, TH, VAVG, VB, VBR.

**Example**

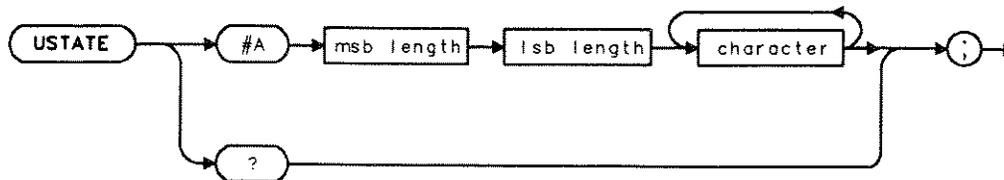
```
OUTPUT 718;"IP;MKN;RB 10KHZ;MKPK NH;UP;"
```

*Increases the resolution bandwidth to 30 kHz because MKPK NH does not change the active function.*

## USTATE User State

The USTATE command transmits information that has been stored in the analyzer by the user.

### Syntax



| Item       | Description/Default                                                                       | Range |
|------------|-------------------------------------------------------------------------------------------|-------|
| Msb length | Most significant byte of a two-byte word that describes the number of bytes transmitted.  |       |
| Lsb length | Least significant byte of a two-byte word that describes the number of bytes transmitted. |       |
| Character  | Any valid character.                                                                      |       |

Related Commands: FUNCDEF, KEYDEF, ONEOS, SAVES, TRDEF, VARDEF.

### Example

```

DIM User$[20000]           Dimensions a string to hold the user state data.
OUTPUT 718;"USTATE?;"     Sends contents of user state memory to the computer.
ENTER 718 USING "#,-K";User$ Stores contents in string. "-K" allows control codes to be treated as characters.

```

### Description

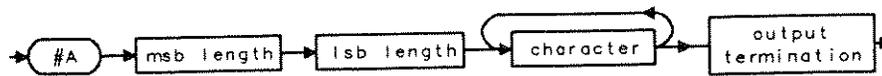
The user state information contains the contents of user memory, which contains:

- User-defined functions (FUNCDEF).
- User-defined variables (VARDEF).
- On-cycle algorithm (ONCYCLE).
- On-delay algorithm (ONDELAY).
- On-end-of-sweep algorithm (ONEOS).
- On-marker algorithm (ONMKR).
- On-SRQ algorithm (ONSRQ).
- On-time algorithm (ONTIME).
- User-defined keys (KEYDEF).
- User-defined trace arrays (TRDEF).

- Values stored in user-defined variables and user-defined traces.

### Query Response

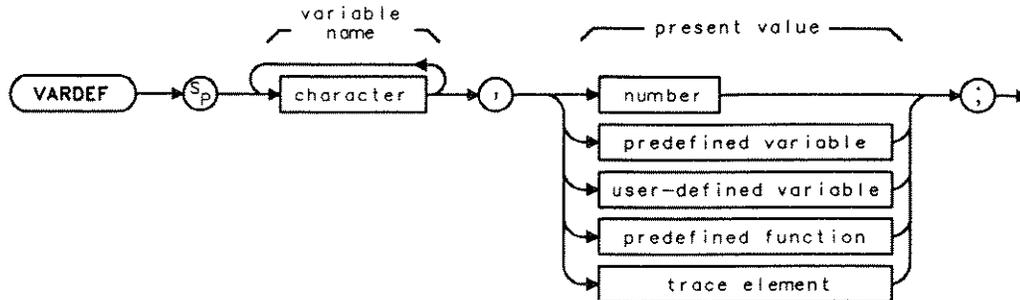
USTATE? returns the user state to the computer. Also returned are instructions required by the analyzer when the user state information is transmitted to the analyzer. The contents of user memory can be restored by executing USTATE followed by the A-block data retrieved by a previous "USTATE?;" command.



## VARDEF Variable Definition

The VARDEF command creates a user-defined variable and assigns it a value.

### Syntax



| Item                  | Description/Default                                               | Range                                                                                                                         |
|-----------------------|-------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Character             | Any valid character.                                              | 2 to 11 characters long, A through Z and the underscore (the underscore should be used as the second character of the label). |
| Number                | Any real or integer number.                                       | Real number range.                                                                                                            |
| User-defined variable | A variable defined by VARDEF or ACTDEF commands.                  | Any valid variable name.                                                                                                      |
| Predefined variable   | A command that acts as a variable. Refer to Table 4-1.            |                                                                                                                               |
| Predefined function   | Function that returns a value. Refer to Table 4-1.                |                                                                                                                               |
| Trace element         | An element of trace A, trace B, trace C, or a user-defined trace. |                                                                                                                               |

Related Commands: DISPOSE, TRDEF.

### Example

OUTPUT 718;"VARDEF V\_AR,0;"

*Defines variable called V\_AR and assigns it a value of 0.*

DISP "ENTER THE VALUE OF THE VARIABLE USING THE ANALYZER KEYS"

OUTPUT 718;"V\_AR EP;"

*The value of V\_AR is changed by using the front-panel controls.*

OUTPUT 718;"V\_AR?;"  
 ENTER 718;N

*Returns entered value of V\_AR to the computer.*

PRINT N

*Displays value on the computer screen.*

OUTPUT 718;"IP;"

*Initializes analyzer.*

OUTPUT 718;"V\_AR?;"

*The value of V\_AR changes to its initial value after an IP.*

ENTER 718;N

*Displays 0.*

PRINT N

### Description

User-defined variables can be substituted for real numbers in many of the analyzer remote-control processes.

Do not use another programming command name as the variable name.

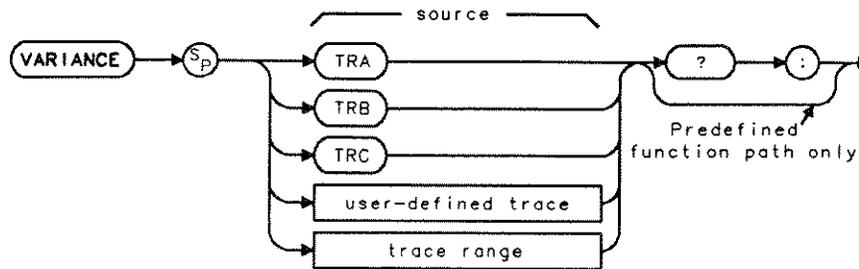
User-defined variables occupy the portion of the analyzer memory reserved for operator use. Use the DISPOSE command to clear user-defined variables from memory to free memory for other functions.

## VARIANCE

### Variance of Trace Amplitudes

The VARIANCE command returns the amplitude variance of the specified trace, in measurement units.

#### Syntax



| Item               | Description/Default                                              | Range                 |
|--------------------|------------------------------------------------------------------|-----------------------|
| User-defined trace | A trace defined by the TRDEF command.                            | Any valid trace name. |
| Trace range        | A segment of trace A, trace B, trace C, or a user-defined trace. |                       |

Prerequisite Commands: TRDEF when using a user-defined trace. TS when using trace data.

Related Commands: MEAN, RMS, STDEV.

#### Example

|                                 |                                                |
|---------------------------------|------------------------------------------------|
| OUTPUT 718;"IP;"                | <i>Initialize analyzer.</i>                    |
| OUTPUT 718;"SNGLS;TS;"          | <i>Activate single-sweep mode.</i>             |
| OUTPUT 718;"VARIANCE TRA?;"     | <i>Return variance of trace A to computer.</i> |
| ENTER 718;Number                | <i>Store value in computer variable.</i>       |
| DISP Number;"MEASUREMENT UNITS" | <i>Display the results on computer screen.</i> |

#### Description

Taking the square root of a variance yields the standard deviation value.

The formula to calculate the variance is as follows:

$$\frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n - 1}$$

$n$  represents the number of data points

$x_i$  represents a data point

$\bar{x}$  represents the mean of data

## Query Response

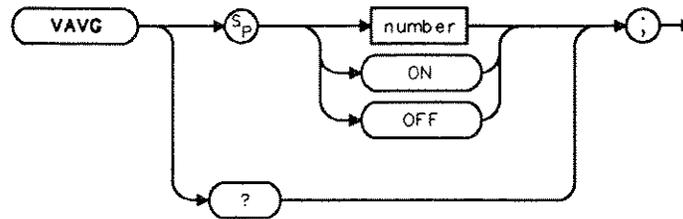


## VAVG

### Video Average

The VAVG command enables the video-averaging function, which averages trace points to smooth the displayed trace.

#### Syntax



| Item   | Description/Default                | Range       |
|--------|------------------------------------|-------------|
| Number | Any valid integer. Default is 100. | 2 to 16384. |

Related Commands: AUTO, CLRAVG, IP, SMOOTH.

#### Example

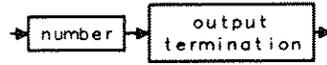
OUTPUT 718;"VAVG 150;"      *Video averages the trace 150 times.*

#### Description

Use VAVG to view low-level signals without slowing the sweep time. Video averaging can lower the noise floor by more than a 30 Hz video bandwidth if a large number of sweeps has been specified for averaging. VAVG may also be used to monitor instrument state changes (for example, changing bandwidths, center frequencies) while maintaining a low noise floor. The active function readout indicates the number of sweeps to be averaged; the default for the number of sweeps is 100 unless otherwise specified. Executing VAVG 0 turns video averaging off. Executing VAVG 1 turns video averaging on.

The functions of the VAVG command and **VID AVG ON OFF** are identical.

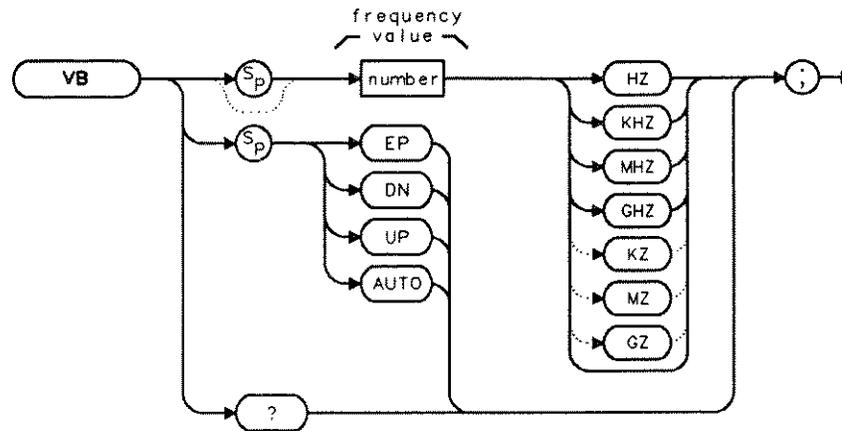
## Query Response



## VB Video Bandwidth

The VB command specifies the video bandwidth, which is a post-detection filter.

### Syntax



| Item   | Description/Default                             | Range           |
|--------|-------------------------------------------------|-----------------|
| Number | Any real or integer number. Default unit is Hz. | 30 Hz to 3 MHz. |

Preset State: 1 MHz.

Step Increment: In a 1, 3, 10 sequence.

Related Commands: AUTO, RB, SP, ST, VBR.

### Example

OUTPUT 718;"VB 10KHZ;" *Changes the video bandwidth to 10 kHz.*

### Description

The resolution bandwidth, video bandwidth, and sweep time are normally coupled to the span. Executing VB uncouples video bandwidth from resolution bandwidth (it does nothing to the sweep-time/resolution-bandwidth/span coupling). Executing AUTO recouples video bandwidth to the resolution bandwidth.

Frequency values other than the values in the 1, 3, 10 sequence are rounded to the nearest permissible value.

The functions of the VB command and **VID BW AUTO MAN** are identical.

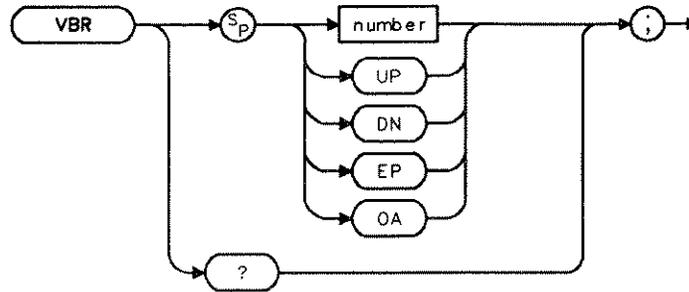
## Query Response



## VBR Video Bandwidth Ratio

The VBR parameter is multiplied by the resolution bandwidth to determine the automatic setting of video bandwidth.

### Syntax



| Item   | Description/Default    | Range        |
|--------|------------------------|--------------|
| Number | Any valid real number. | 0 to 3000000 |

Preset State: 0.300.

Step Increment: 1, 3, 10 sequence.

Related Commands: AUTO, RB, SP, VB.

### Example

```
OUTPUT 718;"VBR 1;"
```

### Description

Ratio values other than the values in the 1, 3, 10 sequence are rounded to the nearest permissible value.

The functions of the VBR command and **VBW/RBW RATIO** are identical.

**Note** VBR returns a real number when used as a predefined variable.



### Query Response

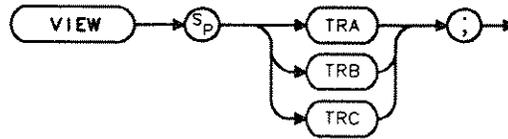


## VIEW

### View Trace

The VIEW command displays trace A, trace B, or trace C, and stops taking new data into the viewed trace.

### Syntax



Preset State: OFF.

Related Commands: BLANK, CLRW, MINH, MXMH.

### Example

```
OUTPUT 718;"VIEW TRA;"
```

### Description

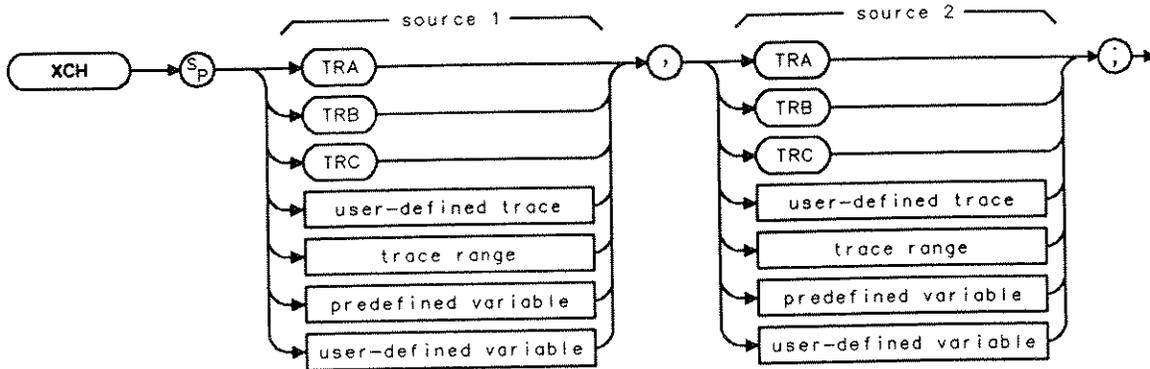
In the VIEW mode the trace is not updated. When VIEW is executed, the contents of the trace are stored in display memory.

The functions of the VIEW TRA, VIEW TRB, and VIEW TRC are identical to **VIEW A**, **VIEW B**, and **VIEW C**, respectively.

## XCH Exchange

The XCH command exchanges the contents of sources 1 and 2.

### Syntax



| Item                  | Description/Default                                              | Range                    |
|-----------------------|------------------------------------------------------------------|--------------------------|
| User-defined trace    | A trace defined by the TRDEF command.                            | Any valid trace name.    |
| User-defined variable | A variable defined by VARDEF or ACTDEF commands.                 | Any valid variable name. |
| Predefined variable   | A command that acts as a variable. Refer to Table 4-1.           |                          |
| Trace range           | A segment of trace A, trace B, trace C, or a user-defined trace. |                          |

Prerequisite Commands: TRDEF when using a user-defined trace. ACTDEF or VARDEF for a user-defined variable. TS when using trace data.

Related Commands: AXB, BXC.

### Example

OUTPUT 718; "XCH TRA,TRB;" *Exchange the contents of trace A with trace B.*

### Description

When the source is longer than the destination, it is truncated to fit. When the source is shorter than the destination, the last element is repeated to fill the destination.



## Index

---

### A

A1, 4-10  
 A2, 4-10  
 A3, 4-10  
 A4, 4-10  
 A-block format, 3-18  
 ABORT, 4-31  
 ABS, 4-33  
 absolute, 4-33  
 ACTDEF, 4-35  
 activate marker, 4-216  
 active function, 4-39, 4-140  
 active function definition, 4-35  
 active functions, 4-39  
 ACTVF, 4-39  
 ADD, 4-40  
 alternate commands, 4-10-11  
 AMB, 4-42  
 AMBPL, 4-45  
 AMPCOR, 4-47  
 amplitude correction, 4-47  
 amplitude units, 4-4, 4-53  
 analyzer command, 4-3  
 analyzer error messages, A-1  
 analyzer state, 3-6  
 analyzers with earlier firmware revisions, vi  
 ANNOT, 4-49  
 annotation, 4-49  
 APB, 4-50  
 AT, 4-51  
 attenuation, 4-51  
 AUNITS, 3-17, 4-53  
 AUTO, 4-55  
 auto couple, 4-55  
 auxiliary control line A, 4-78  
 auxiliary control line B, 4-79  
 auxiliary control line C, 4-80  
 auxiliary control line D, 4-81  
 auxiliary control line I, 4-82  
 average, 4-57  
 AVG, 4-57  
 AXB, 4-59

### B

B1, 4-10  
 B2, 4-10  
 B3, 4-10  
 B4, 4-10  
 base band instrument preset, 4-174  
 BASIC  
   INPUT command, 2-6  
   REAL command, 2-7  
 B format, 3-17  
 BIT, 4-60  
 bit checking, 4-60  
 BL, 4-10  
 BLANK, 4-62  
 blank trace, 4-62  
 BML, 4-63  
 BREAK, C-2  
 BTC, 4-64  
 BXC, 4-65

### C

C1, 4-10  
 C2, 4-10  
 CA, 4-10  
 CAL, 4-66  
 calibration, 4-66  
 CAT, 4-69  
 catalog, 4-69  
 center frequency, 2-3-4, 4-71  
 center frequency step size, 4-377  
 CF, 2-3-4, 4-71  
 character, 4-3  
 character & EOI, 4-3  
 characters and secondary keywords, 4-1, 4-6-9  
 character set, 4-172  
 clear average, 4-73  
 clear display, 4-74  
 clear status byte, 4-76  
 clear write, 4-75  
 CLRAVG, 4-73  
 CLRDSP, 4-74  
 CLRW, 4-75  
 CLS, 4-76  
 CNF, 4-77  
 CNTLA, 4-78

- CNTLB, 4-79
  - CNTLC, 4-80
  - CNTLD, 4-81
  - CNTLI, 4-82
  - COMB, 4-83
  - COM command, 2-2
  - command cross reference, 4-1, 4-25-30
  - command mnemonic, 4-1
  - commands, 2-1
  - command terminators, 4-2
  - commenting programming lines, 2-9
  - compatible commands, 4-10-11
  - compatible function, 4-3
  - COMPRESS, 4-84
  - compress trace, 4-84
  - CONCAT, 4-86
  - concatenate, 4-86
  - confidence test, 4-77
  - configuring your computer system, 1-1
  - connecting your analyzer to a printer or plotter, 1-12
  - continuous sweep, 4-88
  - controlling trace data with a computer, 3-1
  - CONTS, 4-88
  - convert to absolute units, 4-92
  - convert to measurement units, 4-94
  - correction factors on, 4-89
  - CORREK, 4-89
  - CR, 4-10
  - CRTHPOS, 4-90
  - CRTVPOS, 4-91
  - CS, 4-10
  - CT, 4-10
  - CTA, 4-92
  - CTM, 4-94
  - current units, 4-4
  - CV, 4-10
- D**
- data byte, 4-3
  - data byte & EOI, 4-3
  - data entry
    - disabling, 4-140
  - date mode, 4-95
  - DATEMODE, 4-95
  - define function, 4-134
  - define terminator, 4-110
  - delete limit-line table, 4-177
  - delimiter, 4-3
  - DEMOD, 4-96
  - demodulation, 4-96
  - DET, 4-97
  - detection mode, 4-97
  - digit, 4-3
  - display, 4-108
  - display line, 4-103
  - DISPOSE, 4-99
  - DIV, 4-101
  - divide, 4-101
  - DL, 4-103
  - DN, 4-105
  - DONE, 4-106
  - dotted lines, 4-2
  - down, 4-105
  - DSPLY, 4-108
  - DT, 4-110
- E**
- E1, 4-10
  - E2, 4-10
  - E3, 4-10
  - E4, 4-10
  - EE, 4-111
  - EK, 4-113
  - ELSE, 4-148
  - EM, 4-10
  - enable entry, 4-111
  - enable knob, 4-113
  - enable limit line testing, 4-190
  - ENDIF, 4-148
  - end-or-identify, 3-7
  - END statement, 3-19
  - ENTER, 4-114
  - enter From HP-IB, 4-114
  - enter limit-line segment, 4-188
  - enter parameter function, 4-116
  - EOI, 3-7, 3-19
  - EP, 4-116
  - ERASE, 4-117
  - EX, 4-10
  - exchange, 4-439
  - exchange trace A and trace B, 4-59
  - EXP, 4-118
  - exponent, 4-118
- F**
- FA, 4-121
  - fast Fourier transform, 4-125
  - FB, 4-123
  - FFT, 4-125
  - firmware revisions, vi
  - FM gain, 4-128
  - FMGAIN, 4-128
  - FOFFSET, 4-130
  - force service request, 4-374
  - FORMAT, 4-132
  - format card, 4-132
  - frequency offset, 4-130
  - frequency units, 4-4

FS, 4-133  
 full span, 4-133  
 FUNCDEF, 4-134  
 functional index, 4-1, 4-12-24

**G**

GR, 4-136  
 graph, 4-136  
 GRAT, 4-137  
 graticule, 4-137  
 graticule coordinates, 4-289  
 GW BASIC  
   OPEN command, 3-4  
 GW BASIC DIM, 3-9

**H**

hardware error messages, A-1  
 harmonic number, 4-141  
 harmonic number lock, 4-142  
 HAVE, 4-138  
 HD, 4-140  
 Hewlett-Packard interface bus, B-1  
 HN, 4-141  
 HNLOCK, 4-142  
 HNUNLK, 4-145  
 hold data entry, 4-140  
 horizontal position of CRT display, 4-90  
 How to Use This Manual, v  
 HP 9000 Series 200 Technical Computers, 1-3  
 HP 9000 Series 300 Technical Computers, 1-5  
 HP BASIC  
   CLEAR command, 2-2  
   CREATE command, 3-3  
   DIM command, 3-7  
   END command, 2-2  
   ENTER, 3-7  
   LOCAL command, 2-2  
   REAL command, 2-5  
   USING command, 3-7  
 HP-IB interface, 1-1  
 HP Vectra Personal Computer  
   HP-IB interface, 1-7  
   RS-232 interface, 1-8

**I**

IB, 4-146  
 I-block format, 3-18  
 IBM PC/AT and compatible computers, 1-10  
 ID, 4-147  
 identify, 4-147  
 IF THEN ELSE ENDIF, 4-148  
 impedance units, 4-4  
 informational messages, A-1  
 input B, 4-146

input impedance, 4-153  
 instrument preset, 2-2, 4-154  
 INT, 4-151  
 integer, 4-3, 4-151  
 instrument preset, 2-3  
 INZ, 4-153  
 IP, 2-2-3, 4-154

**K**

key clear, 4-156  
 KEYCLR, 4-156  
 KEYCMD, 4-157  
 key command, 4-157  
 KEYDEF, 4-160  
 KEYENH, 4-163  
 key enhance, 4-163  
 KEYEXC, 4-167  
 key execute, 4-167  
 key label, 4-168  
 KEYLBL, 4-168  
 KSA, 4-10  
 KSB, 4-10  
 KSc, 4-10  
 KSC, 4-10  
 KSE, 4-10  
 KSG, 4-10  
 KSH, 4-10  
 KSi, 4-10  
 KSI, 4-10  
 KSm, 4-10  
 KSM, 4-10  
 KSn, 4-10  
 KSo, 4-10  
 KSO, 4-10  
 KSZ, 4-10

**L**

L0, 4-11  
 label, 4-170  
 label functions, 4-173  
 LB, 4-170  
 LF, 3-7, 4-174  
 LG, 4-175  
 LIMIDEL, 4-177  
 LIMIFAIL, 4-178  
 LIMILINE, 4-180  
 LIMIMIRROR, 4-183  
 LIMIMODE, 4-184  
 LIMIREL, 4-186  
 LIMISEG, 4-188  
 LIMITEST, 4-190  
 limit line  
   entry mode, 4-184  
 limit lines, 4-180  
   enable limit line testing, 4-190

- enter limit-line segment, 4-188
- limits failed, 4-178
- mirror limit line, 4-183
- relative limit lines, 4-186
- limits failed, 4-178
- linear scale, 4-192
- line-feed, 3-7
- LN, 4-192
- LOAD, 4-193
- LOG, 4-195
- logarithm, 4-195
- logarithmic scale, 4-175
- lsb length, 4-3
- LSB length, 3-18

**M**

- M1, 4-11
- M2, 4-11
- M3, 4-11
- M4, 4-265
- Manual Terms and Conventions, v
- Marker Amplitude, 4-214
- marker bandwidth, 4-218
- marker continue, 4-220
- marker counter, 4-225
- marker counter resolution, 4-226
- marker delta, 4-221
- marker frequency, 4-223
- marker frequency output, 4-207
- marker minimum, 4-228
- marker noise, 4-231
- marker normal, 4-229
- marker off, 4-233
- marker pause, 4-236
- marker peak, 4-238
- marker peak excursion, 4-239
- marker position, 4-234
- marker readout, 4-241
- Marker Step Size, 4-245
- Marker Stop, 4-247
- marker to center frequency, 4-219
- Marker to Reference Level, 4-243
- Marker to Span, 4-244
- Marker Trace, 4-248
- Marker Track, 4-249
- Marker Type, 4-250
- marker zoom, 4-265
- mass storage is, 4-261
- maximum, 4-262
- maximum hold, 4-264
- MC, 4-11
- MDS, 3-17, 4-198
- MDU, 4-199
- MEAN, 4-200
- MEASURE, 4-202

- measurement data size, 3-17, 4-198
- measurement data units, 4-199
- measurement units, 3-19, 4-93
- measure mode, 4-202
- measuring harmonic distortion
  - HP-IB, 3-11-14
  - RS-232, 3-14-16
- MEM, 4-204
- memory available, 4-204
- MENU, 4-205
- MF, 4-207
- M format, 3-19-20
- MIN, 4-209
- MINH, 4-211
- minimum, 4-209
- minimum hold, 4-211
- minimum position, 4-212
- MINPOS, 4-212
- MIRROR, 4-213
- mirror image, 4-213
- mirror limit line, 4-183
- mixer level, 4-252
- MKA, 3-2, 4-214
- MKACT, 4-216
- MKBW, 4-218
- MKCF, 4-219
- MKCONT, 4-220
- MKD, 4-221
- MKF, 3-2, 4-223
- MKFC, 4-225
- MKFCR, 4-226
- MKMIN, 4-228
- MKN, 4-229
- MKNOISE, 4-231
- MKOFF, 4-233
- MKP, 4-234
- MKPAUSE, 4-236
- MKPK, 4-238
- MKPK HI, 2-4
- MKPX, 4-239
- MKREAD, 4-241
- MKRL, 4-243
- MKSP, 4-244
- MKSS, 4-245
- MKSTOP, 4-247
- MKTRACE, 4-248
- MKTRACK, 4-249
- MKTYPE, 4-250
- ML, 4-252
- MOD, 4-254
- MODE, 4-256
- modulo, 4-254
- MOV, 4-257
- move, 4-257
- MPY, 4-259

msb length, 4-3  
 MSB length, 3-18  
 MSI, 4-261  
 MT0, 4-11  
 MT1, 4-11  
 multiply, 4-259  
 MXM, 4-262  
 MXMH, 4-264

**N**

normalized reference level, 4-267  
 NRL, 4-267  
 number, 4-3

**O**

O1, 4-11  
 O2, 4-11  
 O3, 4-11  
 O4, 4-11  
 OA, 4-269  
 OL, 3-9, 4-270  
 on cycle, 4-271  
 ONCYCLE, 4-271  
 on delay, 4-273  
 ONDELAY, 4-273  
 on end of sweep, 4-275  
 ONEOS, 4-275  
 on marker, 4-277  
 ONMKR, 4-277  
 on service request, 4-279  
 ONSRQ, 4-279  
 on sweep, 4-281  
 ONSWP, 4-281  
 on time, 4-283  
 ONTIME, 4-283  
 OP, 4-285  
 OUTPUT, 4-286  
 output active function value, 4-269  
 output learn string, 3-9, 4-270  
 output parameter, 4-285  
 output termination, 4-3  
 output to HP-IB, 4-286

**P**

PA, 4-289  
 parameter units, 3-17  
 PD, 4-291  
 PDA, 4-292  
 PDF, 4-294  
 peak position, 4-298  
 PEAKS, 4-296  
 pen down, 4-291  
 pen up, 4-307  
 P format, 3-17

PKPOS, 4-298  
 PLOT, 4-299  
 plot absolute, 4-289  
 plot relative, 4-303  
 plotting  
   HP-IB interface, 1-13  
   RS-232, 1-15  
 power bandwidth, 4-309  
 POWERON, 4-301  
 power-on state, 4-301  
 PP, 4-302  
 PR, 4-303  
 predefined function, 4-4  
 predefined variable, 4-4  
 prefix, 4-304  
 PREFX, 4-304  
 preselector peak, 4-302  
 PRINT, 4-305  
 printing  
   HP-IB interface, 1-12  
   RS-232, 1-14  
 probability distribution of amplitude, 4-292  
 probability distribution of frequency, 4-294  
 problems, 1-17  
 programming guidelines, 2-9  
 protect state, 4-306  
 PSTATE, 4-306  
 PU, 4-307  
 PURGE, 4-308  
 purge file, 4-308  
 PWRBW, 4-309

**R**

R1, 4-11  
 R2, 4-11  
 R3, 4-11  
 R4, 4-11  
 RB, 4-311  
 RC, 4-11  
 RCLS, 4-313  
 RCLT, 4-314  
 reading trace data, 3-1  
 recall state, 4-313  
 recall trace, 4-314  
 recommended path, 4-2  
 records, 3-7  
 reference level, 4-323  
 reference level offset, 4-328  
 reference-level position, 4-325  
 relative limit lines, 4-186  
 release HP-IB, 4-316  
 RELHPIB, 4-316  
 repeating syntax element, 4-2  
 REPEAT UNTIL, 4-317  
 reserved words, 4-1

- reset reference level, 4-319
- RESETRL, 4-319
- resolution bandwidth, 4-311
- RETURN, 4-320
- returning the analyzer to its former state, 3-9
- REV, 4-322
- revision, 4-322
- RL, 4-323
- RLPOS, 4-325
- RMS, 4-327
- ROFFSET, 4-328
- root mean square value, 4-327
- RQS, 4-330
- RS-232
  - 3-wire connection, C-3
  - baud rate, C-2
  - connecting a modem, C-4
  - connecting an HP-GL plotter, C-4
  - connecting a thinkJet printer, C-3
  - data lines, C-1
  - handshaking connection for personal computer, C-2
  - handshaking lines, C-1
  - protocol, C-2
  - setting the baud rate, C-5
- RS-232 interface, I-2
- RS-232 interface bus, C-1
  
- S**
- S1, 4-11
- S2, 4-11
- save menu, 4-332
- SAVEMENU, 4-332
- save or recall data, 4-339
- save or recall flag, 4-336
- save or recall number, 4-337
- SAVES, 4-333
- save state, 4-333
- SAVET, 4-334
- save trace, 4-334
- saving and recalling instrument states, 3-6
- saving trace data, 3-3
- SAVRCLF, 4-336
- SAVRCLN, 4-337
- SAVRCLW, 4-339
- screen coordinates, 4-289
- secondary keywords, 4-1
- SEGDEL, 4-341
- segment delete, 4-341
- segment entry, 4-343
- SENDER, 4-343
- SER, 4-345
- serial number, 4-345
- service request mask, 4-330
- set date, 4-346
- SETDATE, 4-346
- set time, 4-347
- SETTIME, 4-347
- single sweep, 4-350
- single-sweep mode, 2-2
- SMOOTH, 4-348
- smooth trace, 4-348
- SNGLS, 2-2-3, 4-350
- source attenuator, 4-359
- source leveling control, 4-358
- source normalization, 4-361
- source power, 4-369
- source power-level step size, 4-365
- source power offset, 4-363
- source power sweep, 4-367
- source tracking, 4-371
- source tracking peak, 4-373
- SP, 2-3-4, 4-351
- span, 2-3-4, 4-351
- span zoom, 4-354
- SPEAKER, 4-353
- special numbers and characters, 4-1
- SPZOOM, 4-354
- SQLCH, 4-355
- SQR, 4-356
- square root, 4-356
- squelch, 4-355
- SRCALC, 4-358
- SRCAT, 4-359
- SRCNORM, 4-361
- SRCPOFS, 4-363
- SRCPSTP, 4-365
- SRCPSWP, 4-367
- SRCPWR, 4-369
- SRCTK, 4-371
- SRCTKPK, 4-373
- SRQ, 4-374
- SS, 4-377
- ST, 4-379
- standard deviation of trace amplitudes, 4-382
- start frequency, 4-121
- status byte query, 4-381
- STB, 4-381
- STDEV, 4-382
- stop frequency, 4-123
- STOR, 4-384
- store, 4-384
- SUB, 4-387
- subtract, 4-387
- SUM, 4-389
- summary of compatible commands, 4-1
- sum of squared trace amplitudes, 4-390
- sum of trace amplitudes, 4-389
- SUMSQR, 4-390
- SV, 4-11

Sweep Couple, 4-391  
 sweep time, 4-379  
 SWPCPL, 4-391  
 syntax elements, 4-1-5

## T

T0, 4-11  
 T1, 4-11  
 T2, 4-11  
 T3, 4-11  
 T4, 4-11  
 TA, 4-393  
 take sweep, 4-418  
 TB, 4-395  
 TDF, 3-16, 4-396  
 test program, 1-2  
 TEXT, 4-398  
 TH, 4-399  
 THEN, 4-148  
 threshold, 4-399  
 time date, 4-401  
 TIMEDATE, 4-401  
 time display, 4-402  
 TIMEDSP, 4-402  
 time units, 4-4  
 TITLE, 4-403  
 TM, 4-404  
 TRA, 3-2, 4-406  
 trace A minus trace B, 4-42  
 trace A minus trace B plus display line, 4-45  
 trace A plus trace B, 4-50  
 trace B exchange trace C, 4-65  
 trace B minus display line, 4-63  
 trace data format, 4-396  
   binary, 3-7, 3-9  
   TDF, 2-7  
 trace data formats, 3-16  
 trace data input/output, 4-406  
 trace data transfers, 3-16  
 trace define, 4-409  
 trace display, 4-411  
 trace element, 4-4  
 trace graph, 4-412  
 trace math, 3-19, 4-414  
 trace mean, 4-200  
 trace memory, 4-408  
 trace preset, 4-416  
 trace range, 4-4  
 trace status, 4-417  
 trace window, 4-423  
 transfer A, 4-393  
 transfer B, 4-395  
 transfer trace B to trace C, 4-64  
 TRA/TRB/TRC, 4-406  
 TRB, 4-406

TRC, 4-406  
 TRCMEM, 4-408  
 TRDEF, 4-409  
 TRDSP, 4-411  
 TRGRPH, 4-412  
 trigger mode, 4-404  
 TRMATH, 4-414  
 TRPRST, 4-416  
 TRSTAT, 4-417  
 TS, 2-2-3, 4-418  
 TV frame, 4-420  
 TV line, 4-419  
 TVLINE, 4-419  
 TVSFRM, 4-420  
 TV Standard, 4-421  
 TVSTND, 4-421  
 TV sync, 4-422  
 TVSYNC, 4-422  
 TWNDOW, 4-423  
 two's complement, 4-50

## U

units, 4-4  
 unlock harmonic number, 4-145  
 UNTIL, 4-317  
 UP, 4-425  
 user-created error messages, A-1  
 user-defined function, 4-5  
 user-defined key definition, 4-160  
 user-defined trace, 4-5  
 user-defined variable, 4-5  
 user state, 4-426  
 USTATE, 4-426

## V

VARDEF, 4-428  
 variable definition, 4-428  
 variables, 2-5  
 VARIANCE, 4-430  
 variance of trace amplitudes, 4-430  
 VAVG, 4-432  
 VB, 4-434  
 VBR, 4-436  
 vertical position of CRT display, 4-91  
 video average, 4-432  
 video bandwidth, 4-434  
 video bandwidth ratio, 4-436  
 VIEW, 4-438

## W

writing your first program, 2-1

## X

XCH, 4-439



## A

## Analyzer Error Messages

---

The analyzer can generate various messages that appear on its screen during operation to indicate a problem.

There are three types of messages: hardware error messages (H), user-created error messages (U), and informational messages (M).

- Hardware error messages indicate the analyzer hardware is probably broken. Refer to Chapter 8 in the *HP 8591A/8593A Installation, Verification, and Operation Manual* or the *HP 8590B/8592B Installation, Verification, and Operation Manual* for more information.
- User-created error messages appear when the analyzer is used incorrectly. They are usually generated during remote operation (entering programming commands using a controller or the external keyboard).
- Informational messages indicate the analyzer's progress within a specific procedure.

The messages are listed in alphabetical order on the following pages; each message is defined, and its type is indicated by an (H), (U), or (M).

### ADC-GND FAIL

Indicates a failure in the processor. (H)

### ADC-TIME FAIL

Indicates a failure in the processor. (H)

### ADC-2V FAIL

Indicates a failure in the processor. (H)

### CAL: - - -

During the self-calibration routine, messages may appear on the display indicating the routine is progressing: SWEEP, FREQ, SPAN, AMPTD, FM GAIN + OFFSET, 3dB BW, ATTEN, LOG AMP, PEAKING, YTF. FREQ UNCAL appears briefly during CAL FREQ. This is normal and does not indicate a problem. (M)

### CAL: DATA NOT STORED

#### CAL AMP NEEDED

The correction factors are corrupt and cannot be stored. Perform the CAL FREQ & AMPTD routine. (U) (H)

### CAL: cannot execute CALAMP

enter: 0 dB PREAMP GAIN

The preamp gain should be set to 0 dB before the CAL AMPTD routine is performed. The preamp gain is set by using EXT PREAMP. (U) (H)

### CAL: FM SPAN SENS FAIL

The analyzer could not set up span sensitivity of the FM coil. (H)

**CAL: GAIN FAIL**

Indicates the signal amplitude is too low during the **CAL AMPTD** routine. (H)

**CAL: LOST COMB SIGNAL**

Indicates the amplitude of the comb generator signal is insufficient to complete the CAL YTF. Be sure to use a low-loss cable (SMA-to-Type N cable) to connect the comb generator output the analyzer input before when using **CAL YTF**. (U) (H)

**CAL: NO YTF IN 8590/1**

The CAL YTF programming command is available for the HP 8592B and the HP 8593A only. (U)

**CAL: NO YTO AVAILABLE**

The CAL DLY programming command is no longer necessary. (U)

**CAL: PASSCODE NEEDED**

Indicates that the function cannot be accessed without the pass code. (M)

**CAL: RES BW AMPL FAIL**

The relative insertion loss of the resolution bandwidth is incorrect. (H)

**CAL SIGNAL NOT FOUND**

Indicates the CAL OUT signal cannot be found. Check that the CAL OUT is connected to the analyzer input connector using an appropriate cable. If the CAL OUT signal is connected to the analyzer input but cannot be found, press **FREQUENCY**, **-37 Hz** before performing the **CAL FREQ** or **CAL FREQ & AMPTD**. (U) (H)

**CAL: SPAN SENS FAIL**

The self-calibration span sensitivity routine failed. (H)

**CAL: USING DEFAULT DATA**

Indicates the calibration data is corrupt and default correction factors are being used. Interruption of the self-calibration routines or an error can cause this problem. (M)

**COMB SIGNAL NOT FOUND**

The comb signal cannot be found. Check that 100 MHz COMB OUT is connected to the analyzer input. The comb generator is available with the HP 8592B or HP 8593A only. (U) (H)

**COMMAND ERROR: \_ \_ \_**

The specified programming command is not recognized by the analyzer. (U)

**CONFLICT TABLE OVERFLOW**

Indicates that too many two-letter compatible commands have been used. See Table 4-3 for information about substituting alternate commands for two-letter compatible commands. (U)

**CONF TEST FAIL**

Indicates that the confidence test failed. (H)

**DECR SPAN**

Indicates the resolution bandwidth to span ratio is too small to use the marker count function. Check the span and bandwidth settings. (U)

**FAIL: \_ \_ \_**

An error was discovered during the power-up check. The 4-digit by 10-digit code indicates the type of error. Error codes are described in the analyzer Service Manual. (H).

**FREQ UNCAL**

Indicates a YTO-tuning failure. This may occur when using default correction factors. Performing the **CAL FREQ** routine may eliminate the failure. The FREQ UNCAL message appears briefly during the **CAL FREQ** routine or when changing the frequency value with the knob (it does not indicate a problem). (U) (H)

**INVALID ACTDEF: - - -**

The specified ACTDEF name is not valid. See the ACTDEF programming command. (U)

**INVALID AMPCOR: FREQ**

For the AMPCOR command, the frequency data must be in increasing order. See the AMPCOR programming command. (U)

**INVALID AUNITS: - - -**

The amplitude units are not valid. See the AUNITS programming command. (U)

**INVALID BLOCK FORMAT: IF STATEMENT**

An invalid block format appeared within the IF statement. (U)

**INVALID CARD: DIRECTORY**

Indicates the memory card has not been formatted. (U)

**INVALID CARD: NO CARD**

Indicates a memory card has not been inserted. (U)

**INVALID CARD**

Indicates a card reader is not installed, the memory card is write-protected, the memory card is a read-only card, or a memory card has not been inserted. This message can occur if remote programming commands for the memory card capability are executed with an HP 8590B or HP 8592B without Option 003. (U)

**INVALID CARD: TYPE**

Indicates a card reader is not installed, the memory card is write-protected, the memory card is a read-only card, or a memory card has not been inserted. This message can occur if remote programming commands for the memory card capability are executed with an HP 8590B or HP 8592B without Option 003. (U)

**INVALID CHECKSUM: USTATE**

The user-defined state does not follow the expected format. (U)

**INVALID COMPARE OPERATOR**

An IF/THEN or REPEAT/UNTIL routine is improperly constructed. Specifically, the IF or UNTIL operands are incorrect. (U)

**INVALID DETECTOR: - - -**

The specified detector is not valid. See the DET programming command. (U)

**INVALID ENTER FORMAT**

The enter format is not valid. See the appropriate programming command description to determine the correct format. (U)

**INVALID FILE: NO ROOM** Indicates that there is not enough available space on the memory card to store the data. (U)

**INVALID HP-IB ADDRESS/OPERATION**

An HP-IB operation was aborted due to an incorrect address or invalid operation. Check that there is only one controller (the analyzer) connected to the printer. (U)

**INVALID HP-IB OPERATION REN TRUE**

The HP-IB operation is not allowed. (This is usually caused by print/plot when a controller is on the interface bus.) (U)

**INVALID ITEM:**

Indicates an invalid parameter has been used in a programming command. (U)

**INVALID KEYNAME: \_ \_ \_**

The specified key name is not allowed. (The key name may have conflicted with an analyzer programming command.) Use an underscore as the second character in the key name, or avoid beginning the key name with the following pairs of letters: LB, OA, OL, TA, TB, TR, MA, MF, TS, OT, and DR. (U)

**INVALID OUTPUT FORMAT**

The output format is not valid. See the appropriate programming command description to determine the correct format. (U)

**INVALID REGISTER NUMBER**

The specified trace register number is invalid. (U)

**INVALID REPEAT MEM OVFL**

Memory overflow occurred due to a REPEAT routine. This occurs if the repeat statements are too long. (U)

**INVALID REPEAT NEST LEVEL**

The nesting level in the REPEAT routine is improperly constructed. This can occur if too many REPEAT routines are nested. (U)

**INVALID RS-232 ADDRESS/OPERATION**

An RS-232 operation was aborted due to an incorrect address or invalid operation. (U)

**INVALID SAVEREG**

Data has not been saved in the specified state or trace register, or the data is corrupt. (U)

**INVALID STORE DEST: \_ \_ \_**

The specified destination field is invalid. (U)

**INVALID SYMTAB ENTRY: SYMTAB OVERFLOW**

There is a symbol table overflow. This can occur if there are too many user-defined items (functions, variables, key definitions) or downloadable programs in analyzer memory. Use **DELETE FILE** or **DISPOSE USER MEM** to delete unnecessary items. This can also occur when the processor board has failed. See the analyzer's Service Manual for more information. (U)

**INVALID TRACE: \_ \_ \_**

The specified trace is invalid. (U)

**INVALID TRACE NAME: \_ \_ \_**

The specified trace name is not allowed. Use an underscore as the second character in the trace name, or avoid beginning the trace name with the following pairs of letters: LB, OA, OL, TA, TB, TR, MA, MF, TS, OT, and DR. (U)

**INVALID TRIGGER MODE: \_ \_ \_**

The specified trigger mode is invalid. See the TM programming command. (U)

**INVALID VALUE PARAMETER: \_ \_ \_**

The specified value parameter is invalid. (U)

**INVALID VARDEF: - - -**

The specified variable name is not allowed. Use an underscore as the second character in the variable label, or avoid beginning the variable label with the following pairs of letters: LB, OA, OL, TA, TB, TR, MA, MF, TS, OT, and DR. (U)

**INVALID WINDOW TYPE: - - -**

The specified window is invalid. See the TWINDOW programming command. (U)

**MEAS UNCAL**

The measurement is uncalibrated. Check the sweep time, span, and bandwidth settings. (U)

**NO CARD FOUND**

Indicates that the memory card is not inserted. (U)

**NO COUNTERLOCK AVAILABLE**

The programming command is available for the HP 8591A or the HP 8593A only. (U)

**OVEN COLD**

Indicates that the analyzer has been powered up for less than 5 minutes. (Option 004 only.) (M)

**PARAMETER ERROR: - - -**

The specified parameter is not recognized by the analyzer. See the appropriate programming command description to determine the correct parameters. (U)

**POS-PK FAIL**

Indicates the positive-peak detector has failed. (H)

**RES-BW SHAPE FAIL**

Indicates the 3 dB bandwidth is not within specifications. (H)

**REF UNLOCK**

Indicates that the frequency reference is not locked to the external reference input. Check that the 10 MHz REF OUT is connected to the EXT REF IN, or that an external 10 MHz reference source is connect to the EXT REF IN (when using an external reference). (M)  
(H)

**RES-BW NOISE FAIL**

Indicates the noise floor level is too high at the indicated bandwidth. (H)

**SAMPLE FAIL**

Indicates the sample detector has failed. (H)

**SOFTKEY OVFL**

Softkey nesting exceeds the maximum number of levels. (U)

**SRQ - - -**

The specified service request is active. Service requests are a form of informational message and are explained in Appendix B of the Installation, Verification, and Operation Manual for the analyzer. (M)

**STEP GAIN ATTEN FAIL**

Indicates the step gain has failed. (H)

**TABLE FULL**

Indicates the upper or lower table of limit lines contains the maximum number of entries allowed. Additional entries to the table are ignored. (U)

**TG SIGNAL NOT FOUND**

Indicates the tracking generator output signal cannot be found. Check that the tracking generator output (RF OUT 50 $\Omega$  or RF OUT 75 $\Omega$ ) is connected to the analyzer input connector using an appropriate cable. (U)

**TG UNLVL**

Indicates that the source power is set higher or lower than the analyzer can provide (HP 8591A with Option 010 or 011 only). See "Stimulus-Response Measurements" in Chapter 6 for more information.

**UNDEF KEY**

A softkey referred to is not recognized by the analyzer. (U)

**VID-BW FAIL**

Indicates the video bandwidth(s) have failed. (H)

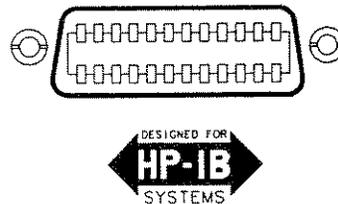
**B**

## HP-IB Option 021

---

This appendix tells you how to connect a computer to your HP 8590 Series Option 021 Spectrum Analyzer with the Hewlett-Packard Interface Bus (HP-IB).

Your analyzer has an HP-IB connector on the rear panel, as shown in Figure B-1.



**Figure B-1. HP-IB Connector**

The HP-IB system utilizes a party-line bus structure. Devices such as the analyzer are connected on the party line with HP-IB cables. A computer gives instructions and is the “controller.” The analyzer takes orders and is the “listener.” The analyzer is also capable of transmitting data over the party line. Devices that transmit data back to the computer are “talkers.”

Each device on the party line has an address. Device addresses are used by the controller to specify who talks and who listens. A device’s address is usually set at the factory.

The number 7 preceding the device’s address (for example, **Analyzer=718**), signifies that the HP-IB interface is selected.

When you turn on the analyzer, the HP-IB address appears on the screen (for example, HP-IB ADRS: 18). If necessary, you can reset the address of the analyzer by pressing **CONFIG** and **ANALYZER ADDRESS**. You may use any address between 0 and 30. (Usually, 1 is reserved for printers and 5 for plotters.)



## RS-232 Option 023

---

### What You'll Learn in This Appendix

This appendix explains how to connect a computer to your HP 8590 Series Option 023 Spectrum Analyzer using the RS-232 interface. It contains information pertaining to RS-232 signals, cable connections, and baud rate.

### Introducing the RS-232 Interface

Your analyzer has an RS-232 connector on the rear panel, as shown in Figure C-1.

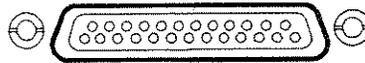


Figure C-1. RS-232 Connector

The RS-232 interface utilizes serial data transmission. Data is sent, one bit at a time, in groups of 10 to 12 data-bits.

Two devices, such as the analyzer and a computer, can exchange commands and data over the RS-232 connection. This interface uses two serial data lines and five handshaking lines. Handshaking signals are required for full hardware control of the information exchange. It is possible to use a three-wire connection, in some situations.

Another parameter for the RS-232 interface is the “baud,” or data, rate. This is the speed at which the computer and analyzer exchange data. The baud rate of each of the two RS-232 devices must be the same.

#### The RS-232 Data Lines

RS-232 uses serial data transmission, meaning that data is transmitted one bit at a time. There are two data lines carrying signals:

- Transmit data (TxD)—the serial data output. This line is connected to the RxD input line.
- Receive data (RxD)—the serial data input. This line is connected to the TxD output line.

#### The RS-232 Handshaking Lines

In addition to the data signals, there are five other signals lines (called handshaking lines), used to control the flow of data. Listed below are the handshake signal descriptions:

- Request to send (RTS)—Output signal indicates that the analyzer is ready to communicate. This line is true at power-up and stays true while power is on.

- Clear to send (CTS)—Input signal indicates that the external controller is ready to receive data.
- Data terminal ready (DTR)—Output signal from the analyzer. When the input buffer is full, this line goes false.
- Data set ready (DSR)—Is not available.
- Data carrier detect (DCD)—Input to the analyzer. If DCD is true, the analyzer will receive data from the controller. If false, no data will be input. The data will be ignored.

The analyzer checks its CTS input before transmitting data to the computer. If the CTS line is false, the analyzer will not transmit data. The analyzer transmits data when the CTS line is true.

The analyzer sets the DTR line (PC CTS) false when its input buffer is full.

## Baud Rate

The speed at which data is exchanged is called the baud rate or data rate. This is usually expressed in baud or bits per second. Common rates are 1200 and 9600.

### Note



Some of the programs in this manual use 1200 baud for proper operation. If your system uses the RS-232 handshake lines, you can use 9600 baud for all of the programs.

If you need to change the baud rate, refer to the “Setting the Analyzer Baud Rate” in this appendix.

## Protocol

The RS-232 protocol is as follows:

- Baud rate 300 to 57,000 baud.
- 8 bits per character.
- 1 stop bit.
- No parity.
- Software handshake—none.
- Xon/Xoff and ENQ/ACK not supported by the analyzer.

When BREAK is issued to the analyzer, the following occurs:

1. The present command is aborted.
2. The input buffer is cleared.
3. The output buffer is cleared.
4. All trace output is stopped.
5. The command parser is reinitialized.

BREAK does not perform any of the following:

- Invoke instrument preset.
- Clear SRQ off screen.
- Clear illegal command off screen.

The RTS signal goes true on power-up and does not go false during any communication. It stays true while power is on.

Table C-1 lists the signal connections between a personal computer and the analyzer.

**Table C-1. Full Handshaking Connection**

| Personal Computer |         | Analyzer |
|-------------------|---------|----------|
| TxD 2             | >—————> | 3 RxD    |
| RxD 3             | <—————< | 2 TxD    |
| RTS 4             | >—————> | 8 DCD    |
| CTS 5             | <—————< | 20 DTR   |
| DSR 6             | <—————< |          |
| GnD 7             | <—————< | 7 GnD    |
| DCD 8             | <—————< | 4 RTS    |
| DTR 20            | >—————> | 5 CTS    |

If your computer operates with only three wires, you can use the cable connections in Table C-2.

Some computers require that the CTS, DSR, and DCD inputs be true before serial transmission can occur. To solve this problem, you can wire these three signals to the personal computer RTS line.

**Table C-2. 3-Wire Connection**

| Personal Computer |         | Analyzer |
|-------------------|---------|----------|
| TxD 2             | >—————> | 3 RxD    |
| RxD 3             | <—————< | 2 TxD    |
| GND 7             | <—————< | 7 GND    |
|                   |         | 4 RTS    |
|                   |         | 5 CTS    |
|                   |         | 8 DCD    |

### Connecting a ThinkJet Printer

To connect an HP ThinkJet printer to the analyzer, use the information in Table C-3.

**Table C-3. ThinkJet Printer Connection**

| ThinkJet |         | Analyzer |
|----------|---------|----------|
| GND 1    | <—————> | 1 GND    |
| TxD 2    | >—————> | 3 RxD    |
| RxD 3    | <—————< | 2 TxD    |
| RTS 4    | >—————> | 8 DCD    |
| GND 7    | <—————> | 7 GND    |
| DTR 20   | >—————> | 5 CTS    |

#### ThinkJet Printer Mode Switches:

HP MODE.  
 Perf Skip.  
 PL 8 1/2 x 11.  
 USASCII.  
 BAUD up to 9600.  
 No Parity.  
 8 bits.

DTR.

## Connecting a Modem

To connect a modem to the analyzer, use the information in Table C-4. The connection is for a Hayes 1200 Modem and the analyzer.

**Table C-4. Modem Connection**

| Modem  | Analyzer |
|--------|----------|
| GND 1  | 1 GND    |
| TxD 2  | 2 TxD    |
| RxD 3  | 3 RxD    |
| CTS 5  | 5 CTS    |
| GND 7  | 7 GND    |
| DCD 8  | 8 DCD    |
| DTR 20 | 20 DTR   |

## System Settings

Select 1200 baud for both the modem and the analyzer.

## Connecting an HP-GL Plotter

To connect an HP-GL plotter to the analyzer, use the information in Table C-5.

**Table C-5. HP-GL Plotter Connection**

| Plotter | Analyzer |
|---------|----------|
| GND 1   | 1 GND    |
| TxD 2   | 3 RxD    |
| RxD 3   | 2 TxD    |
| RTS 4   | 8 DCD    |
| CTS 5   | 20 DTR   |
| DSR 6   |          |
| GND 7   | 7 GND    |
| DTR 20  | 5 CTS    |

## Switch Settings

Set the switches on the HP-GL plotter to the following settings. Set the baud rate of the plotter and analyzer to the same value. After setting the switch positions, turn plotter off, then on again.

|                          |                        |
|--------------------------|------------------------|
| Expand: ◀                | Parity: ▶ (Off)        |
| Emulate: ◀               | Even/Odd: ▶ (Odd)      |
| Standalone: ◀            | Duplex: ▶ (Full)       |
| Monitor Mode: ▶ (normal) | Hardwire: ◀            |
| Local ▶ (normal)         | DTR-Bypass: ▶ (normal) |

## Setting the Analyzer Baud Rate

The baud rates of the analyzer and the personal computer must be the same. For example, to set the analyzer to 9600 baud, use the following procedure:

1. Press the **CONFIG**, **MORE 1 of 2**.
2. Press the **BAUD RATE** softkey.
3. Press these keys: 9600 **Hz**. To set the baud rate to 1200 baud, press these keys: 1200 **Hz**.

---

**Note**

Some of the programs in this manual use 1200 baud for proper operation. If your system uses the RS-232 handshake lines, you can use 9600 baud for all of the programs.

---

