



HP 8592A Portable Microwave Spectrum Analyzer

Programming Manual

HP-IB, HP-IL, RS-232

HP 8592A Documentation Description

Manuals shipped with your analyzer:

Installation Manual*

HP Part Number 08592-90003

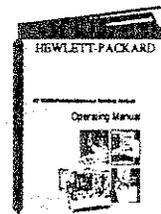
- Tells you how to install the spectrum analyzer
- Tells you what to do in case of a failure



Operating Manual*

HP Part Number 08592-90005

- Tells you how to make measurements with your spectrum analyzer
- Describes analyzer features

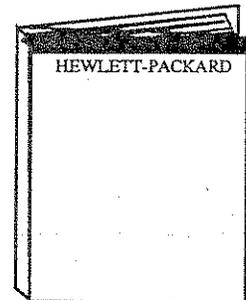


Options:

Support Manual (HP 8592A Option 915)**

HP Part Number 08592-90008

- Describes troubleshooting and repair of the analyzer

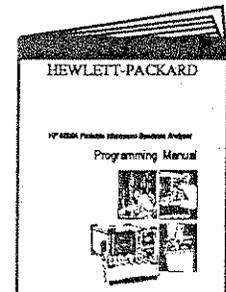


Programming Manual

HP Part Number: 08592-90010 (HP 8592A Option 021, 022, 023)

- Describes analyzer operation via a remote controller (computer)

You are here!



* Additional copies of the Operating Manual and the Installation Manual are not available separately; together, they constitute the HP 8592A Documentation Package and must be ordered by its HP Part Number — 08592-90001.

** Option 915, Service Documentation (08592-90007), consists of one copy each of the Support Manual, the Installation Manual, and the Operating Manual.

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

Words in this manual that appear CAPITALIZED in [BRACKETS] refer to softkeys that appear on the analyzer screen. Keys that appear on the front panel of the instrument appear in ***BOLD ITALICS***.

Printing History

Each new edition of this manual incorporates all material updated since the previous edition. Manual change sheets may be issued between editions, allowing you to correct or insert information in the current edition.

The manual part number changes only when a new edition is published. Corrections or additions may be made as the manual is reprinted between editions.

Part Number 08592-90010
First Printing August 1987

Table of Contents

HP 8592A Documentation Description	2
How to Use This Manual	3
Where to Start	3
Manual Terms and Conventions	3
Printing History	3

1 Preparation for Use

Preparation

What You'll Learn in This Chapter	1-1
Connecting Your Analyzer to a Printer/Plotter	1-1
Connecting Your Analyzer to a Computer	1-3
Configuring Your Computer System	1-3
Computer/Analyzer Interconnections	1-3
The HP-IB	1-3
The RS-232	1-3
The HP-IL	1-4
HP Series 200 Personal Technical Computer	1-5
HP Series 300 Technical Computers	1-7
HP Vectra Personal Computer	1-8
IBM PC/XT/AT and Compatible Computers	1-12
HP-71B Handheld Computer	1-13
HP-75 Handheld Computer	1-15
If There is a Problem	1-17

2 Programming Fundamentals

Fundamentals

What You'll Learn in This Chapter	2-1
Writing Your First Program	2-2
Composing the Program	2-2
HP-IB	2-2
RS-232	2-3
HP-IL	2-4
Modifying the Program	2-5
HP-IB	2-5
RS-232	2-6
HP-IL	2-7
Enhancing the Program with Variables	2-8
HP-IB	2-8
RS-232	2-9
HP-IL	2-10
Getting Information from the Analyzer	2-11
HP-IB	2-11
RS-232	2-12
HP-IL	2-13
Programming Guidelines	2-15

3 Advanced Programming

What You'll Learn in This Chapter	3-1
Controlling Trace Data With a Computer	3-2
Reading Trace Data	3-2
HP-IB	3-2
RS-232	3-3
HP-IL	3-4
Saving Trace Data	3-5
HP-IB	3-5
RS-232	3-6
HP-IL	3-7
Reading Trace Data from a Computer Disc	3-8
HP-IB	3-8
RS-232	3-8
HP-IL	3-9
Saving and Recalling Instrument States	3-10
Saving the Analyzer's State	3-10
HP-IB	3-10
RS-232	3-12
HP-IL	3-14
Returning the Analyzer to its Former State	3-15
HP-IB	3-15
RS-232	3-16
HP-IL	3-17
Measuring Harmonic Distortion	3-18
HP-IB	3-18
RS-232	3-21
HP-IL	3-23

Commands

4 Programming Commands

What You'll Learn in This Chapter	4-1
Syntax Conventions	4-1
Symbols in Syntax Diagrams	4-1
Command Syntax Summary (Table 4-1)	4-5
Syntax Elements Enclosed in Rectangles (Table 4-2)	4-8
Command Parameters (Table 4-3)	4-9
Summary of Compatible Commands (Table 4-4)	4-11
Functional Index (Table 4-5)	4-13
Command Descriptions	4-17

Appendix A
Analyzer Messages

Appendix B
HP Interface Bus

Appendix C
HP Interface Loop

Using the Computer as a Solid State Camera	C-2
Taking A Snapshot	C-4
Viewing A Snapshot	C-5

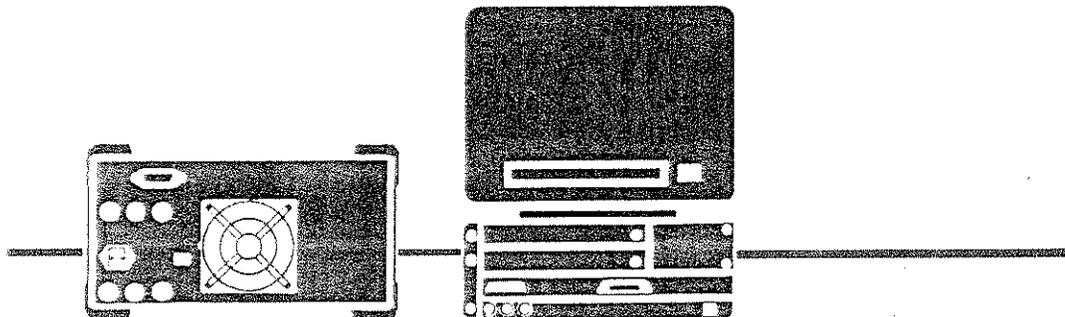
Appendix D
RS-232 Serial Connector

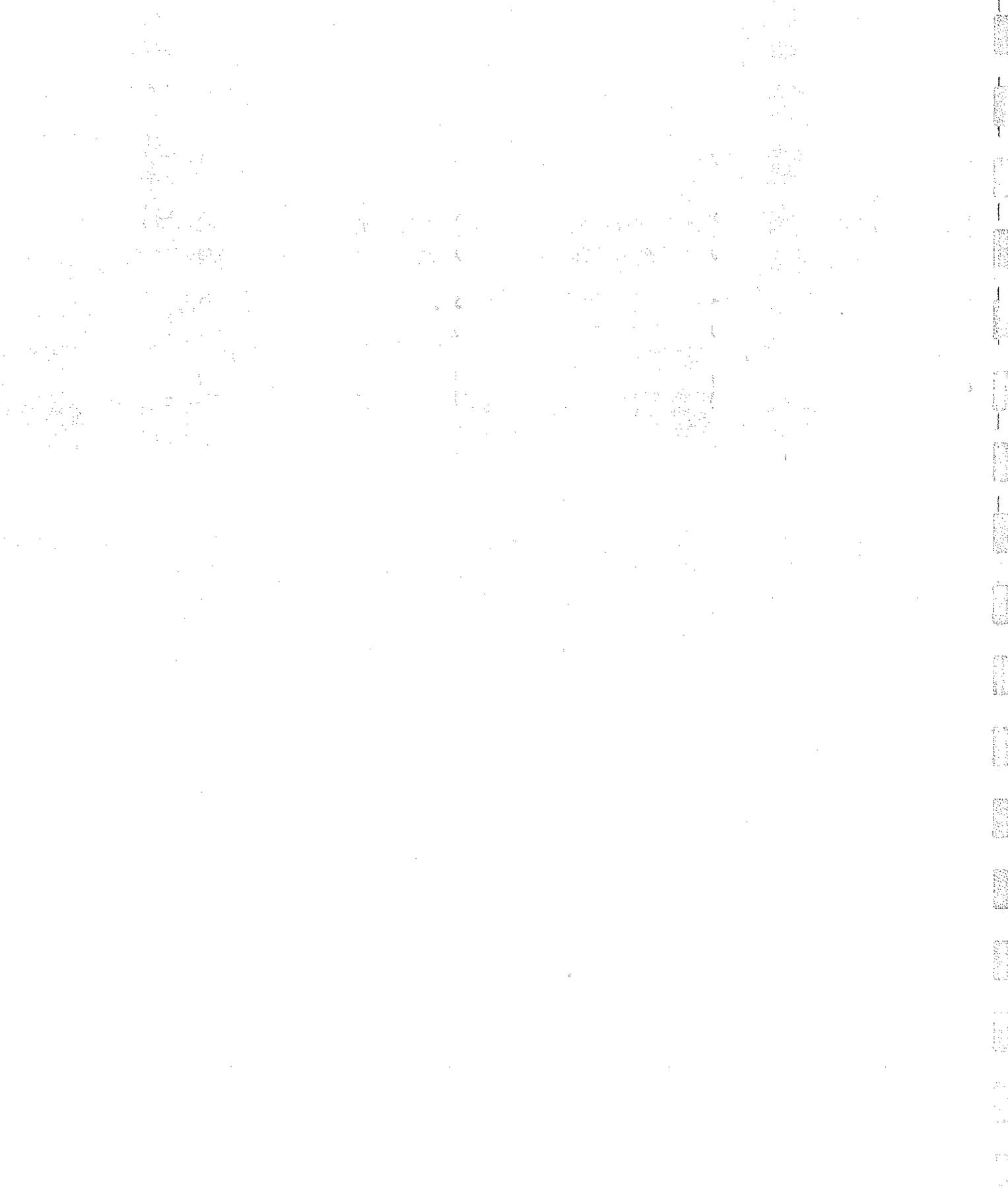
Connecting a ThinkJet Printer	D-4
Connecting a Modem	D-5
Connecting an HP-GL Plotter	D-6
Front-Panel Lockout	D-7
Setting the Baud Rate	D-7

Index

Chapter 1

Preparation for Use





CHAPTER 1

PREPARATION FOR USE

What You'll Find in This Chapter

This chapter explains how to connect your analyzer to a printer or plotter, and how to connect it to a computer using either the Hewlett-Packard Interface Bus (HP-IB), the RS-232 Interface, or the Hewlett-Packard Interface Loop (HP-IL). The remainder of this chapter covers procedures to follow if a problem is encountered.

Connecting Your Analyzer to a Printer or Plotter

You may wish to obtain a permanent record of data that is displayed on the screen of the analyzer. This can be done using the *PRNT* or *PLOT* keys of the analyzer, provided that the following has been done:

1. A printer or plotter, or both, have been connected with any other devices (possibly including a computer) to the analyzer in an HP-IB party-line bus structure, an RS-232 serial data transmission, or in an HP-IL loop. **Note:** The analyzer is **not** likely to print or plot without disconnecting the computer.

To connect the HP-IB:

- a. Turn off all of the devices to be connected.
- b. Connect all devices (including the analyzer) together in a party line using the appropriate HP-IB cable.
- c. Turn on all devices involved.

To connect the RS-232:

- a. Turn off the device to be connected. (**Note:** There should only be one device connected to the RS-232 interface since it is not a party line.)
- b. Connect the device using the appropriate RS-232 cable.
- c. Turn on all devices connected to the serial connector.

To connect the HP-IL loop:

- a. Turn off all of the devices to be connected.
- b. Connect all devices (including the analyzer) together in a loop using the appropriate HP-IL cables.
- c. Turn on all devices on the loop.

2. The analyzer has been told where the printer and plotter are connected by setting the analyzer printer and plotter addresses from the front panel. When the *PRNT* or *PLOT* key is pressed, the analyzer becomes the controller and it configures the transmission.

To set the printer address for the HP-IB:

- a. Turn on the power to the analyzer.
- b. On the analyzer, press **CONFIG** [PRINTER ADDRESS].
- c. The printer usually resides at the first device address. To set the address, press **1 Hz** to enter the address 1 for the printer.

To set the plotter address for the HP-IB:

- a. On the analyzer, press **CONFIG**, [PLOTTER ADDRESS].
- b. The plotter usually resides at the fifth device address. To set this address, press **5 Hz** to enter the address 5 for the plotter.

To set the baud rate for the RS-232:

- a. Turn on the power to the analyzer.
- b. See Appendix D of this manual for the appropriate RS-232 cable connectors.
- c. Press the **CONFIG** key on the analyzer.
- d. Press the [BAUD RATE] softkey.
- e. Press these numerical keys: **9, 6, 0, 0, Hz**. To set the baud rate to 1200 baud, substitute these keypad keys: **1, 2, 0, 0, Hz**.

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

To set the printer address for the HP-IL:

- a. Turn on the power to the analyzer.
- b. On the analyzer, press **CONFIG** [PRINTER ADDRESS].
- c. If the printer is the first device connected to the OUT receptacle of the analyzer, then its address is 1. Press **1 Hz** to enter the address 1 for the printer.

To set the plotter address for the HP-IL:

- a. On the analyzer, press **CONFIG**, [PLOTTER ADDRESS].
- b. If the plotter is the second device connected to the loop from the OUT receptacle of the analyzer, then its address is 2. Press **2 Hz** to enter the address 2 for the plotter.

Note: For the HP-IL Interface, the addresses that are set on the analyzer are the loop positions of the printer or plotter with respect to the OUT receptacle of the analyzer.

To test the connection, press **PRNT** if you have a printer, or **PLOT** if you have a plotter. The printer or plotter should immediately start producing a copy of what is on the display of the analyzer.

The analyzer will remember these addresses, even though the power is turned off, so there is no need to re-enter them when the analyzer is turned off and on. However, when using the HP-IL, if the order of devices on the loop changes, these addresses must be changed to match the new loop configuration. (RS-232 interfaces do not have addresses.)

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 will depend upon which computer you are using. Before turning to the inter-connection instructions for your computer, please read the following general information.

Configuring Your Computer System

Every computer system has a specific configuration, or set of components. Your system configuration may include a printer, external disc drive, and/or plotter. Whenever you add another piece of equipment — for example the analyzer — you may need to configure your computer system so that the computer will know where and how to send information to the newly added device.

Some computers do not require any configuring when an analyzer is connected; other computers require a simple modification. The most common modification is changing the configuration information stored on the computer's operating system disc. A few computers require the insertion of an add-on board or "card." Step-by-step configuration instructions are provided in this chapter for each interface.

All of the test programs for the HP-IB, HP-IL, and the RS-232, 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

HP-IB

The HP-IB interface is described in Appendix B.

Interconnection instructions for the HP-IB interface are provided for use on an HP 200 Series computer, an HP 300 Series computer, and the HP Vectra computer. 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 analyzer. Consult your computer documentation to determine how to connect external devices on the bus.

RS-232

The RS-232 interface is described in Appendix D.

Interconnection instructions for an HP Vectra PC and the IBM PC/XT/AT or compatible computer are provided for the RS-232. If your computer is not listed and it supports a standard RS-232 interface, there is a good possibility that the 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 basic RS-232 connections: Data Terminal Equipment (DTE) and Data Communications Equipment (DCE). Types of DTE devices include CRT terminals. DCE equipment includes modems and, generally, other computer RS-232 connectors. The analyzer RS-232 port is the DCE-type. Connections for the computer (DCE) to the analyzer (DCE) hookup are shown in Appendix D.

HP-IL

The HP-IL interface is described in Appendix C.

When using the HP-IL controller and advanced programming commands, a special input/output read-only memory (I/O ROM; HP Part Number 00075-15001) must be added to the HP-75D to enhance its BASIC language capabilities. The HP-IL interface is built into the HP-75D computer.

The HP-71B computer requires the addition of the HP-IL option (the HP 82401A HP-IL Interface) in order to interface with the analyzer.

If you have a computer other than the HP-71B or HP-75D and it supports an HP-IL interface, it probably can be connected to the analyzer. Consult your computer documentation to determine how to connect external devices on the loop.

The Test Program

To test the system configuration when the analyzer has been installed, a simple test program is provided for each computer interface type listed. After you have connected your computer and analyzer, you should enter and run the appropriate 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, the computer will display a statement when you run the program.

Note: The listed computer and analyzer equipment includes the minimum components necessary to establish communication between your analyzer and computer. If you will be 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.

HP Series 200 Personal Technical Computer

Equipment

- HP 9816, 9826, or 9836
- HP 8592A Spectrum Analyzer, Optional 021
- HP-IB Cable 10833A, B, C, D, or equivalent

Interconnection Instructions

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

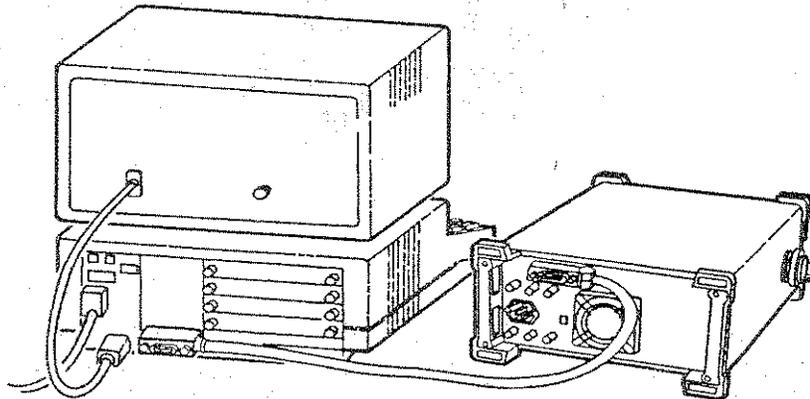


Figure 1-1

2. Check the HP-IB address of the analyzer. Press *CONFIG* and [ANALYZER ADDRESS]. If necessary, reset the address of the analyzer: press *CONFIG*, [ANALYZER ADDRESS], *1*, *8*, *HZ*, (or enter the appropriate address).

Test Program (HP-IB)

To test the computer/analyzer interface, turn on your computer and analyzer and follow the instructions below:

Your HP Series 200 computer may have either a soft-loaded or built-in language system. If your language system is built-in, remove any discs from the drives and press the power switch.

If your language is soft-loaded, install the BASIC language disc into the proper drive.

For further information on booting your system, refer to your BASIC manual.

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 prints CENTER FREQUENCY = 3.0E+8 HZ. The program shows that the computer is able to send instructions to the analyzer and read information from the analyzer.

HP Series 300 Technical Computers

Equipment

- HP 98580A, 98581A, 98582A, or 98583A
- HP 8592A Spectrum Analyzer, Optional 021
- HP-IB Cable 10833A, B, C, D, or equivalent

Interconnection Instructions

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

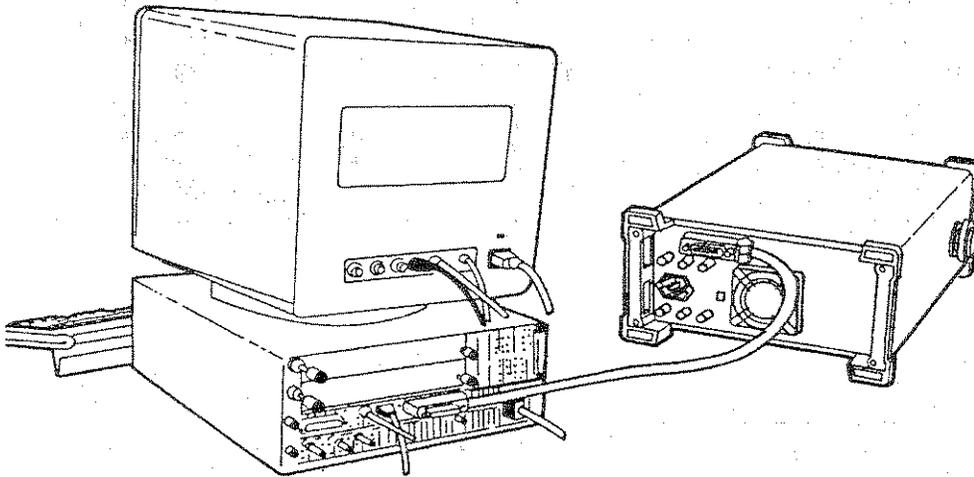


Figure 1-2

2. Check the HP-IB address of the analyzer. Press **CONFIG** and [ANALYZER ADDRESS]. If necessary, reset the address of the analyzer: press **CONFIG**, [ANALYZER ADDRESS], **I**, **8**, **HZ**, or enter the appropriate address.

Test Program (HP-IB)

To test the computer/analyzer interface, follow the test program instructions for HP Series 200 Computers.

HP Vectra Personal Computer

Equipment

HP-IB

- HP Vectra Personal Computer, Required. Option: HP 61062AA or 61062BA Interface and Command Library
- HP 8592A Spectrum Analyzer, Option 021
- HP-IB cable 10833A, B, C, D, or equivalent

Interconnection Instructions

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

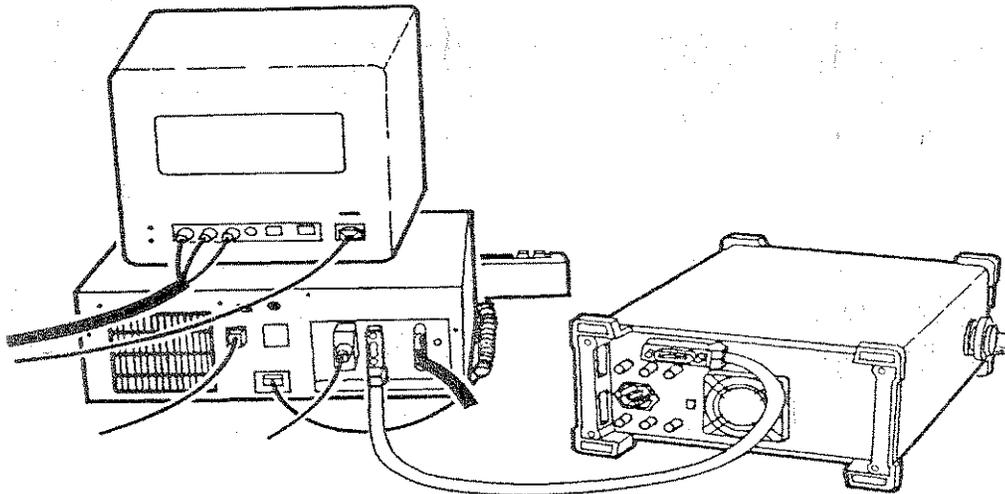


Figure 1-3

Test Program (HP-IB)

To test the computer/analyzer interface, follow the test program instructions for HP Series 200 Computers.

HP Vectra Personal Computer

Equipment

RS-232 (using a 9 pin female port)

- HP Vectra PC with RS-232 Interface
- HP 8592A Spectrum Analyzer, Option 023
- HP 24542G RS-232 Cable

Interconnection Instructions

1. Connect the analyzer to the computer, using the RS-232 cable. Figure 1-4 shows an HP Vectra PC connected to the analyzer.

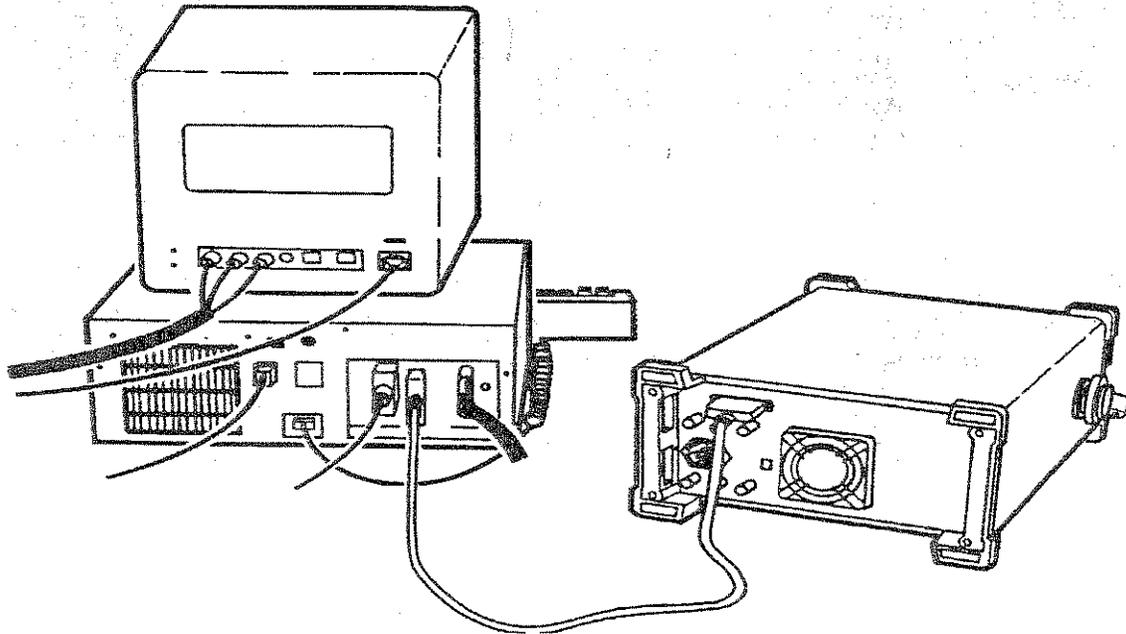


Figure 1-4

The following computers use the BASIC language specified below:

- HP Vectra PC uses Vectra BASIC (HP45952A). The MS BASIC Interpreter (HP 35190A) is compatible with Vectra BASIC.
- HP Series 150 computers use GW BASIC (HP 45450A)
- IBM (or compatible) computers use BASICA version 2.0 or later.

To test the interface: from BASIC, load SETUP.BAS on the computer by typing:

```
LOAD "SETUP"
```

Generally, lines 1 through 999 are reserved for the SETUP.BAS file information; thus, the test program below should begin after line 1000.

```
1010 OPTION BASE 1
1020 MAX.ELEMENTS = 20
1030 DIM READINGS(MAX.ELEMENTS)
1040 CODES$ = SPACES(50)
1050 ANALYZER = 718
1060 CALL IOCLEAR(ANALYZER)
1070 CODES$ = "IP;SNGLS;CF 300MZ;TS;"
1080 LENGTH = LEN(CODES$)
1090 CALL IOOUTPUTS(ANALYZER, CODES$, LENGTH)
1100 CODES$ = "CF?;"
1110 LENGTH = LEN(CODES$)
1120 CALL IOOUTPUTS(ANALYZER, CODES$, LENGTH)
1130 CALL IOENTER(ANALYZER, A)
1140 PRINT "CENTER FREQ = ";A;"Hz";
1150 END
```

When you have entered the program, type:

```
SAVE "PROGRAM"
```

When you are ready to run the program, make sure the analyzer is on, and type:

```
RUN
```

Watch the display on the analyzer. The program tells the analyzer to perform an instrument preset, enter single sweep mode, set the center frequency to 300 MHz, and take a sweep. The program then queries the center frequency value and the computer prints CENTER FREQ = 3.0E+8 Hz. The program proves that the computer is able to send instructions to the analyzer and read information from the analyzer.

Test Program for RS-232 Computers

The test program provided in this section should work with any of the following computers and BASIC language versions:

- HP Vectra PC uses Vectra BASIC (HP45952A). The MS BASIC Interpreter (HP35190A) is compatible with Vectra BASIC.
- IBM PC/XT/AT and compatible computers use BASICA version 2.0 or later or GW BASIC.

To test the interface, 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 D 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.

Watch the display on the analyzer. 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 prints CENTER FREQ = 3.0E+8 Hz. This proves that the computer is able to send instructions to the analyzer and read information from the analyzer.

IBM PC/XT/AT and Compatible Computers

Equipment

- IBM PC/XT/AT or compatible with RS-232 Interface
- HP 8592A Spectrum Analyzer, Option 023
- RS-232 Cable (DCE-DCE), 7 pins used (See Appendix B for wiring of this cable.)

Interconnection Instructions

1. Connect the analyzer to the computer, using the RS-232 cable. Figure 1-5. Note the gender of the connectors. The analyzer uses a female RS-232 connector. The IBM PC/XT/AT computer usually uses a male RS-232 connector. Some compatibles use a female RS-232 connector.

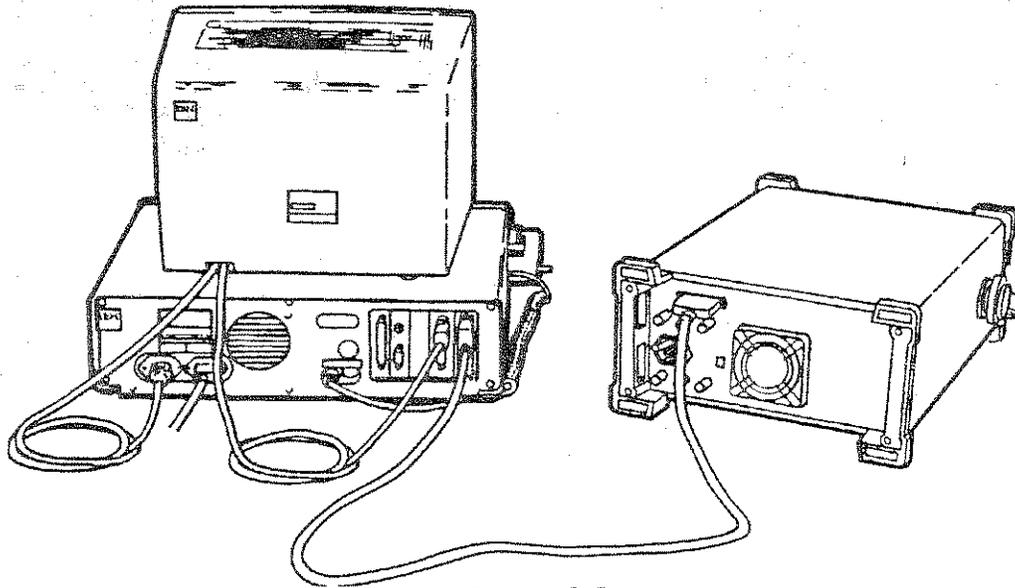


Figure 1-5

Test Program (RS-232)

To test the computer/analyzer interface, follow the test program instructions for the HP Vectra Personal Computer.

HP-71B Handheld Computer

Equipment

- HP-71B Handheld Computer
- HP 82401A HP-IL Interface
- HP 8592A Spectrum Analyzer, Option 022
- HP 82167A, B or D, HP-IL Cables, (2 required)

Interconnection Instructions

1. Turn off the computer and all other devices in the loop (if any).
2. Using the two HP-IL cables, connect the analyzer to the computer, as shown in Figure 1-6.

Note: If more than one device is in the loop, make sure that the analyzer is at position 1 in the loop; the analyzer connects to the HP-71B OUT connector, as shown in the figure.

3. Turn on all peripheral devices in the loop, including the analyzer.
4. Turn on the HP-71B.
5. Type `RESTORE IO` on the HP-71B keyboard. This configures the loop for the set of devices currently connected.

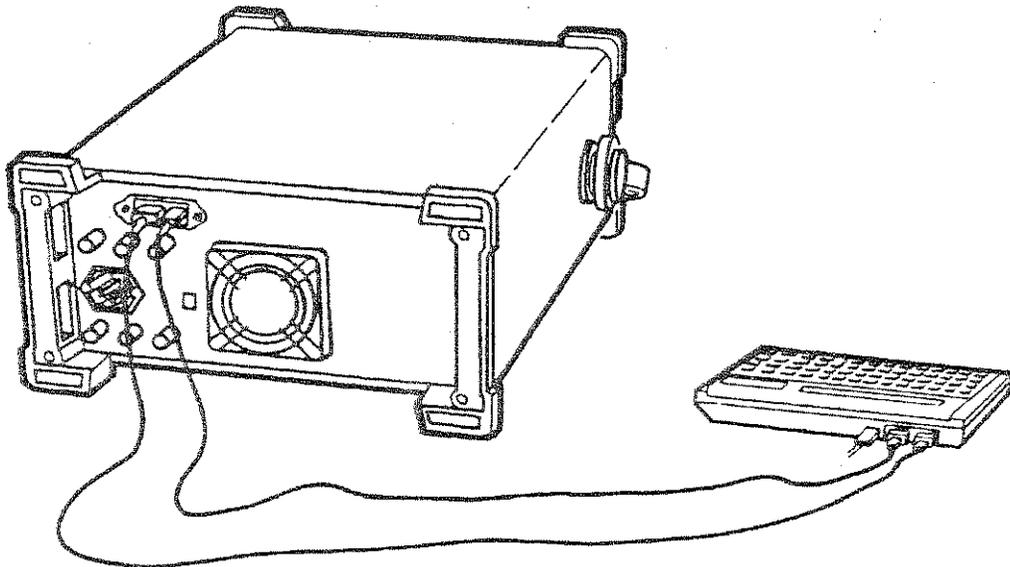


Figure 1-6

Test Program (HP-IL)

Enter the following test program. This program is written in the BASIC language for the HP-71B.

```
10 ! File = TSTPGM71
20 RESET HPIL ! resets HP-IL interface in HP-71B
30 RESTORE IO ! reconfigures the HP-IL loop
40 DESTROY ALL ! set variables to 0
50 REAL C
60 A=1 ! A equals analyzer loop address
70 CLEAR :A
80 OUTPUT :A ;"IP;SNGLS;"
90 OUTPUT :A ;"CF 300MZ;TS;"
100 OUTPUT :A ;"CF?;"
110 ENTER :A ;C
120 FIX 2 ! set the HP-71B display format
130 DISP "CENTER FREQ =";C/1000000;" MHZ"
140 END
```

When you are ready to run the program, make sure the analyzer is on, and that it is the first device in the loop. Press the **RUN** key on the HP-71B keyboard.

Watch the display on the analyzer. The program tells the analyzer to perform an instrument preset, enter single sweep mode, set the center frequency to 300 MHz, and take a sweep. The program then queries the center frequency value and the computer prints **CENTER FREQ = 300.00 MHZ**. This proves that the computer is able to send instructions to the analyzer and read information from the analyzer.

HP-75D Handheld Computer

Equipment

- HP-75D Handheld Computer
- HP 00075-15001 I/O ROM
- HP 8592A Spectrum Analyzer, Option 022
- HP 82167A, B or D, HP-IL Cables, (2 required)

HP-75D Interconnection Instructions

1. Turn off the computer and all other devices in the loop (if any).
2. Using the two HP-IL cables, connect the analyzer to the computer, as shown in Figure 1-7.

Note: If more than one device is in the loop, make sure that the analyzer is at position 1 in the loop; the analyzer connects to the HP-75D OUT connector, as shown in the figure.

3. Turn on all peripheral devices in the loop, including the analyzer.
4. Turn on the HP-75D.

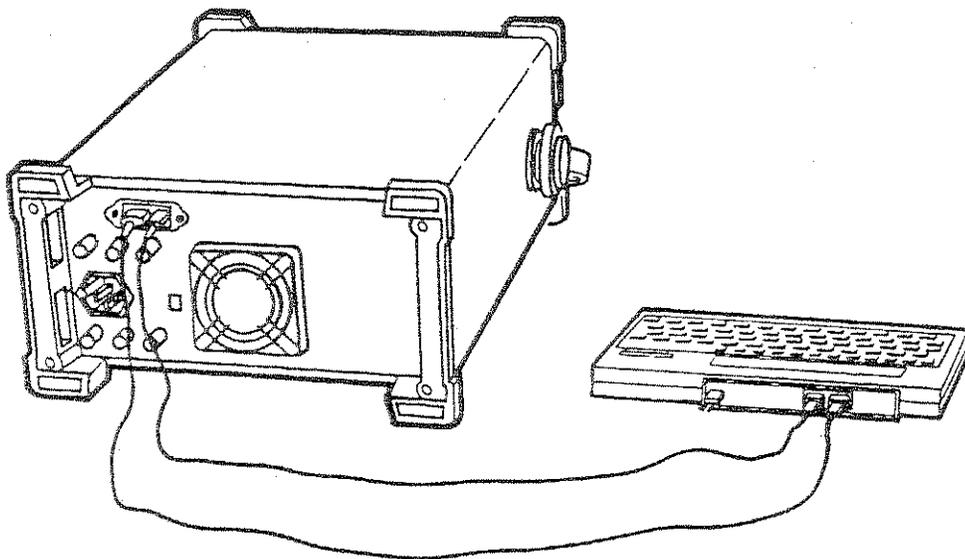


Figure 1-7

Test Program

Enter the following test program. This program is written in the BASIC language for the HP-75D.

```
10 ! File = TSTPGM75
20 ASSIGN LOOP
30 REAL C
40 A=1 ! A equals analyzer loop address
50 ! Send a device clear to the analyzer
60 SENDIO "", "UNT,UNL,LAD"&STR$(A)&",SDC", ""
70 OUTPUT A ; "IP;SNGLS;"
80 OUTPUT A ; "CF 300MZ;TS;"
90 OUTPUT A ; "CF?;"
100 ENTER A ;C
110 DISP USING "K,4D.DD,K" ; "CENTER FREQ =";C/1000000;" MHZ"
120 END
```

When you are ready to run the program, make sure the analyzer is on, and that it is the first device in the loop. Press the *RUN* key on the HP-75D.

Watch the display on the analyzer. The program tells the analyzer to perform an instrument preset, enter single sweep mode, set the center frequency to 300 MHz, and take a sweep. The program then queries the center frequency value and the computer prints CENTER FREQ = 300.00 MHZ. This proves that the computer is able to send instructions to the analyzer and read information from the analyzer.

If There is a Problem

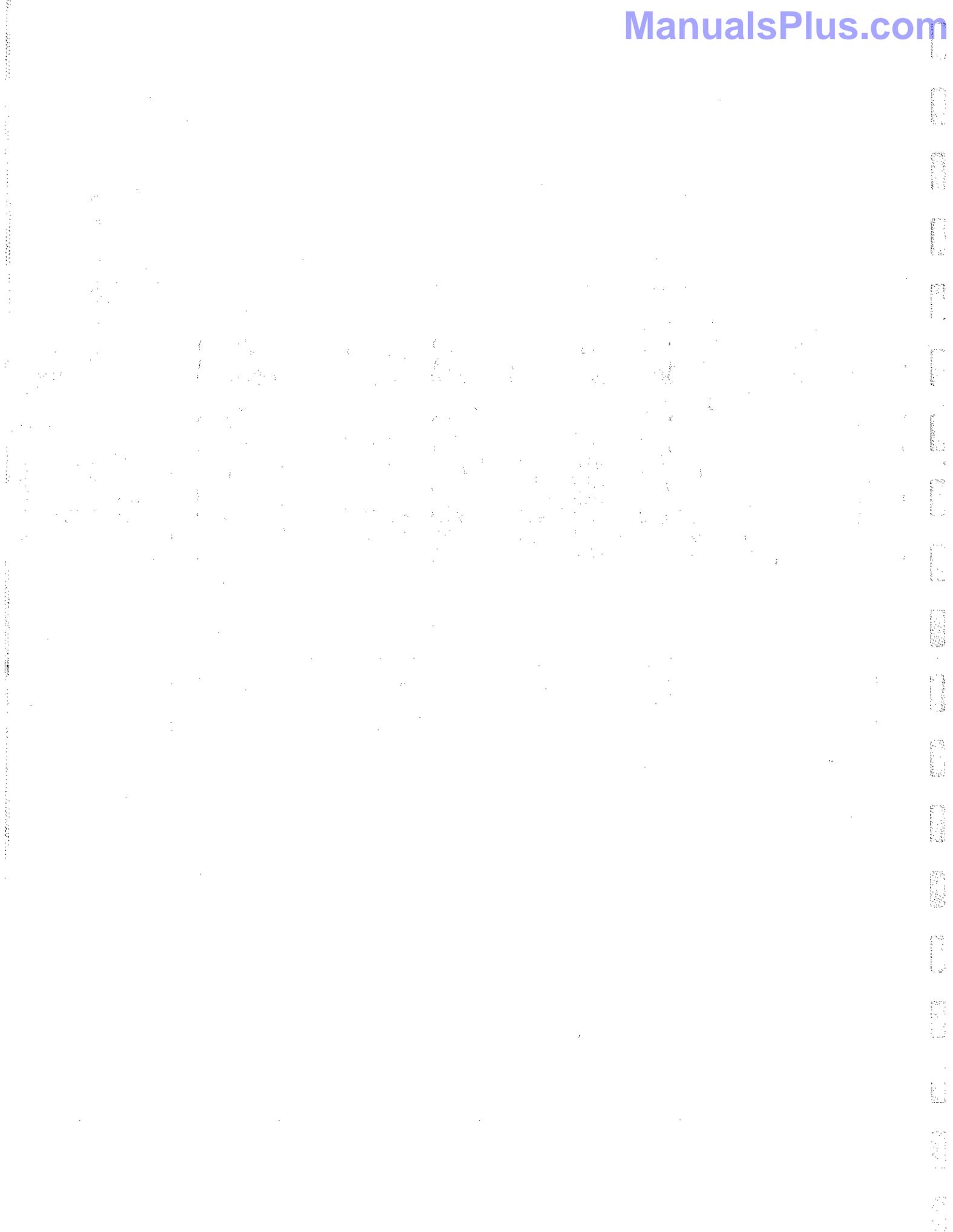
This section offers suggestions to help get your computer and analyzer working as a system. The test programs provided in the remainder of this chapter let you know if the computer/analyzer interface is working properly. If the test program does not run:

1. You may need to modify the program if the program syntax is wrong for the computer. Refer to your BASIC manual for correct syntax.
2. The program must be executed correctly. Refer to your computer manual for help in executing programs.
3. Check for errors in the test program stored in your computer.

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

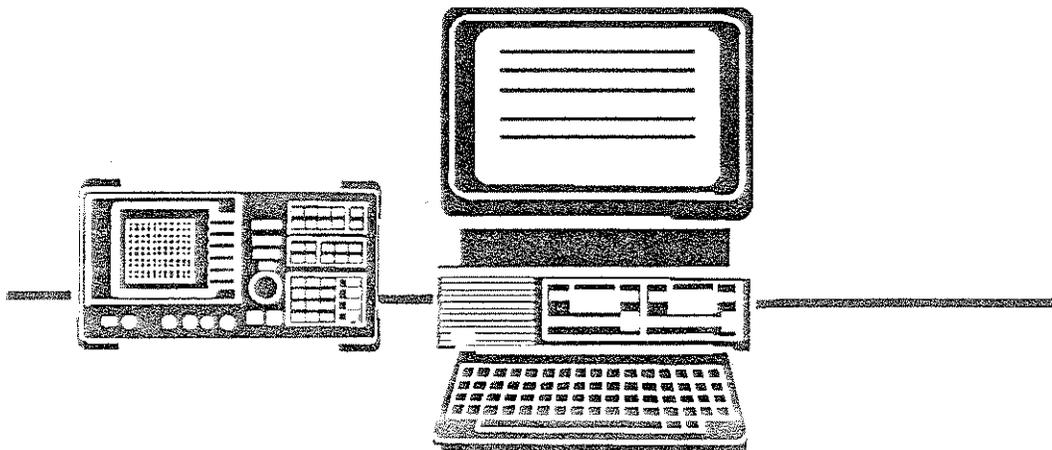
1. Make sure the analyzer is turned on. If the analyzer has power, the green indicator above the line switch will light.
2. Make sure the interface cable is securely connected. Check the interface cable for defects. Make sure the correct cable is used.
3. If you are using the HP-IB or HP-IL interfaces, the analyzer must have the correct address setting. Press *CONFIG* and [ANALYZER ADDRESS].
4. If you are using the HP-IL interface, verify that the analyzer is the first device connected to the OUT receptacle of the computer (address = 1). The analyzer address is its position in the loop.
5. If you are using the RS-232 interface, check the analyzer baud rate. Refer to Appendix D to set 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.



Chapter 2

Programming Fundamentals





CHAPTER 2

PROGRAMMING FUNDAMENTALS

What You'll Learn in This Chapter

This chapter introduces spectrum analyzer programming. The first section of this chapter helps you write your first spectrum analyzer program and introduces programming fundamentals. The second section shows how to extract data from 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."

To best understand this chapter, a general knowledge of the BASIC language is recommended. (Refer to your software documentation manuals.) For reference, Chapter 4 of this manual presents spectrum analyzer commands alphabetically.

Note: All programming examples in this chapter for the HP-IB interface are written in HP BASIC 4.0, using an HP Series 200 Computer. Examples for the HP-IL interface are written in HP BASIC for the HP-71B Handheld Computer. If you use the HP-75D computer, you may need to do some code translation. For the RS-232 interface, examples are written in GWBASIC, using an HP Vectra PC or compatible.

Writing Your First Program

When the analyzer has been connected to a computer via the HP-IB, HP-IL, or RS-232 interfaces, the computer can be used to send instructions to the analyzer. These instructions tell the analyzer such things as the frequency span, the resolution bandwidth, and the 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 analyzer programs contain several common statements that address the analyzer, preset it, and select its sweep mode. As an example, we will write a short program that executes only these common statements.

The finished programs for the three different languages are shown below. 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 quotes). 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.

(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 the program assigns a variable called "Analyzer" to our 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 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 of the program activates 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 analyzer to front-panel control with the "LOCAL 7" command. 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, press *RUN* on the computer, and watch the analyzer as it completes each instruction.

(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, identified as #1, and sets the RS-232 parameters as:

- 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 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, then run the program. Watch the analyzer as it completes each instruction.

Note: When using an HP Series 200 or 300 computer, END commands are necessary. When using an HP Vectra computer, END commands are suggested, but not necessary. Refer to your BASIC manual to determine END statement requirements for your specific computer.

(For the HP-IL Interface)

```
10 ! File = ILPROG1
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 A=1
60 CLEAR :A
70 OUTPUT :A ;"IP;"
80 OUTPUT :A ;"SNGLS;TS;"
90 END
```

Line 20 of the program resets the HP-IL interface in the computer. The next line addresses the peripherals on the loop. Line 40 resets all variables to zero. Placing these three lines at the beginning of a program will insure fewer problems when running programs for the analyzer.

Line 50 of the program assigns a variable called "A" to the analyzer at loop address 1. This instruction is followed by the HP BASIC "CLEAR-HP BASIC Commands" command, which resets the analyzer on the HP-IL. These program lines set up a clear communication path between the computer and the analyzer.

Line 70 introduces the Instrument Preset (IP) command, which corresponds to the *PRESET* key on the analyzer. 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 80 of the program 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. Finally, in the last line, end the program with the "END" command.

Run the program. See that the analyzer does an instrument preset and is then put into single-sweep mode.

Modifying the Program

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

The first chapter in the 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 Your First Measurement" in the Operating 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.

(For the HP-IB Interface)

```
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
```

First, we can set the center frequency to 300 MHz. The CF command corresponds to the **FREQUENCY** key on the analyzer. (All 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 programming line:

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

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

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

When the program is run, the analyzer will take one full sweep before executing the next line. This changes the center frequency to 300 MHz and the span to 200 MHz. Last, we place a marker at the highest peak on the trace. Enter the following line:

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

Run the program to make the measurement. Watch the analyzer 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 will be less likely to make an error than an operator manually entering the same instructions through the front panel.

(For the RS-232 Interface)

```
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
```

First, set the center frequency to 300 MHz. The CF command corresponds to the **FREQUENCY** 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 programming line:

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

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

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

When PROG2 is run, the analyzer will take one full sweep before executing the next line. This changes the center frequency to 300 MHz and the span to 200 MHz. Last, place a marker at the highest peak on the trace.

Enter the following line:

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

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

(For the HP-IL Interface)

```
10 ! File = ILPROG2
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 A=1
60 CLEAR :A
70 OUTPUT :A ;"IP;"
80 OUTPUT :A ;"SNGLS;TS;"
81 OUTPUT :A ;"CF 300MZ;"
82 OUTPUT :A ;"SP 200MZ;"
83 OUTPUT :A ;"TS;"
84 OUTPUT :A ;"MKPK HI;"
90 END
```

First, set the center frequency to 300 MHz. The CF command corresponds to the **FREQUENCY** key on the analyzer. (All analyzer commands, such as CF, are described in Chapter 4.) Insert the following program line between lines 80 and 90 of ILPROG1:

```
81 OUTPUT :A ;"CF 300MZ;"
```

Next, set the span to 200 MHz with the SP command. Add another programming line:

```
82 OUTPUT :A ;"SP 200MZ;"
```

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

```
83 OUTPUT :A ;"TS;"
```

When the program is run, the analyzer will take one full sweep before executing the next line. This changes the center frequency to 300 MHz and the span to 200 MHz. Last, place a marker at the highest peak on the trace.

Enter the following line:

```
84 OUTPUT :A ;"MKPK HI;"
```

Run the program to make the measurement. Watch the analyzer 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 will be less likely to make an error than an operator manually entering the same instructions through 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.

(For the HP-IB Interface)

```
10 !FILE: "IBPROG3"
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(MZ)?",Center_freq
80 !ask for the desired span:
90 INPUT "SPAN(MZ)?",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 in order to include center frequency and span variables. First, using the HP BASIC "REAL" command, we define two variables, Center_freq and Span. These variables will be used to store the frequency and span parameters that will be sent to the analyzer. (See line 20 in the program.)

Second, using the HP BASIC "INPUT" command, we ask the computer operator 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 operator does not enter them on the computer. (See lines 70 to 140.)

Third, we modify the output parameter statements so that the values stored in the Center_freq and Span variables 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. In the program, the "!" allows the words that follow to be ignored by the computer. Thus, they serve as comments in the program.

(For the RS-232 Interface)

```
10 'File = 232PROG3
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, 232PROG2, in order to include center frequency and span variables. We first define two variables, CENTER and SPAN. These variables will be used to store the frequency and span parameters that will be sent to the analyzer.

Second, using the BASIC "INPUT Commands" command, we ask the computer operator to enter the desired center frequency and span from the computer. (See lines 70 and 90 in program 232PROG3.)

Third, 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.

In the program, the single quote (') is used to precede phrases used as program comments. This helps to clarify program operation.

(For the HP-IL Interface)

```
10 ! File = ILPROG3
20 RESET HPIL ! reset the HP-IL interface in the HP-71B
30 RESTORE IO ! reconfigure the HP-IL loop
40 DESTROY ALL ! reset all variables to 0
50 REAL C,S ! define the variables
60 A=1
70 CLEAR :A
80 OUTPUT :A ;"IP;SNGLS;TS;"
90 ! ask for the desired center frequency
100 INPUT "CENTER FREQ(MZ)?" ;C
110 ! ask for the desired span
120 INPUT "SPAN(MZ)?" ;S
130 ! send the center frequency and span to the
140 ! analyzer and take a sweep to update the
150 ! analyzer screen
160 OUTPUT :A ;"CF ";C;"MZ;"
170 OUTPUT :A ;"SP ";S;"MZ;"
180 OUTPUT :A ;"TS;"
190 ! find the signal peak with peak search
200 OUTPUT :A ;"MKPK HI;"
210 END
```

First, using the HP BASIC "REAL" command, define two variables, "C" and "S". These variables will be used to store the frequency and span parameters that will be sent to the analyzer. (See line 50 in the following program, ILPROG3.)

Second, using the HP BASIC "INPUT Commands" command, the program asks the operator 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, do not enter them on the computer. (See lines 100 and 120.)

Third, send the values for center frequency and span to the analyzer (variables "C" and "S"). (See lines 160 and 170.)

Next, a sweep is taken (line 180) to ensure that the analyzer screen is updated before the next command. Last, the marker is placed at the highest signal peak (line 200).

In the program, the "!" allows the words that follow to be ignored by the computer. Thus, they serve to comment the program.

Getting Information from the Analyzer

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

In our second program (earlier in this chapter), we placed a marker at the highest peak on a trace. The value of the marker could be read from the upper-right corner of the analyzer.

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

(For the HP-IB Interface)

```

10 !FILE: "IBPROG4"
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;" MZ"
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. These variables will be used to store the amplitude and frequency values of the marker. (See line 20.)

Second, we set the output format of the 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. 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 ... MZ"
```

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

(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. These variables store the amplitude and frequency values of the marker (lines 170 and 210). The program requests the amplitude and frequency values of the marker (lines 150 and 190). Finally, the amplitude and frequency values from the analyzer are displayed on the computer screen:

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

The program then returns the analyzer to continuous sweep mode and ends.

(For the HP-IL Interface)

```
10 ! File = ILPROG4
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 REAL M1,M2 ! define variables
60 A=1
70 OUTPUT :A ;"IP;"
80 ! set the output format of the analyzer for
90 ! real numbers
100 OUTPUT :A ;"TDF P;"
110 ! set the analyzer parameters
120 OUTPUT :A ;"SNGLS;"
130 OUTPUT :A ;"CF 300MZ;"
140 OUTPUT :A ;"SP 200MZ;"
150 OUTPUT :A ;"TS;"
160 OUTPUT :A ;"MKPK HI;"
170 ! ask the analyzer for the marker's
180 ! amplitude value
190 OUTPUT :A ;"MKA?;"
200 ! send the amplitude value to the computer
210 ENTER :A ;M1
220 ! ask the analyzer for the marker's
230 ! frequency value
240 OUTPUT :A ;"MKF?;"
250 ! send the frequency value to the computer
260 ENTER :A ;M2
270 ! set the HP_71B display format
280 FIX 2
290 ! display the amplitude and frequency
300 DISP "THE SIGNAL PEAK IS ";M1;
310 DISP " dBm AT ";M2/1000000;" MZ"
320 ! set the analyzer to continuous-sweep mode
330 OUTPUT :A ;"CONTS;"
340 END
```

First, using the HP BASIC "REAL" command, define two variables, M1 and M2. These variables will be used to store the amplitude and frequency values of the marker. (See line 50.)

Second, set the output format of the analyzer for real numbers with the analyzer's Trace Data Format (TDF) command. (See line 100.)

As in the original program, set the center frequency and span values. The analyzer takes a sweep, placing the marker on the trace.

The program then asks the analyzer for the amplitude value of the marker. The marker amplitude value is stored in the marker variable, M1. The program also asks the analyzer for the frequency value of the marker, storing the marker frequency value in the frequency variable, M2. (See lines 190 through 260.) Finally, print the values on the computer screen:

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

Before the program ends, the program returns the analyzer to continuous-sweep mode and local control.

Programming Guidelines

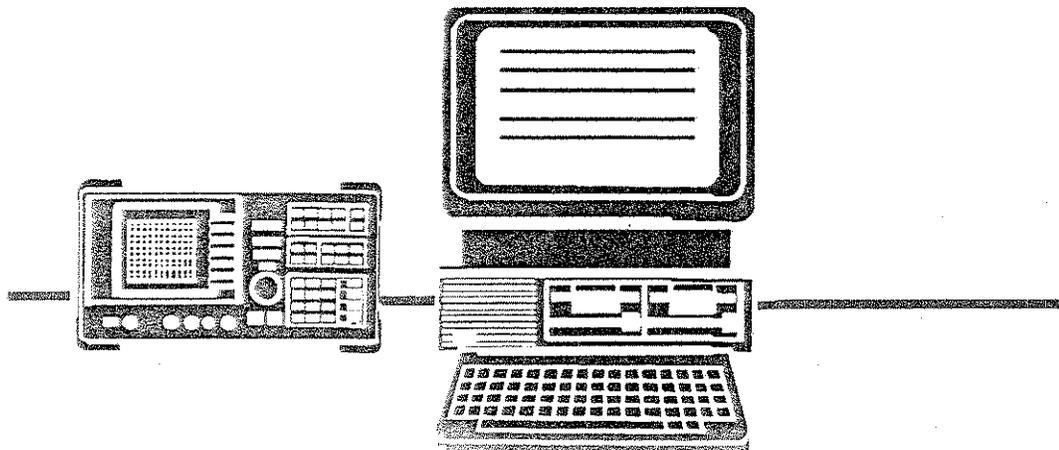
1. Perform the measurement manually, keeping track of the sequence of functions used.
2. In the written program, execute an instrument preset and set single sweep mode before setting other analyzer functions.
3. Use variables for function values. List variables at the beginning of the program.
4. Activate analyzer functions in logical order. Place quotation marks around analyzer commands. Separate commands with semicolons.
5. After setting analyzer functions, execute a take sweep command before reading data or activating markers.
6. Use the "!" to include comment lines in the program when using the HP BASIC connectors.

When using GW BASIC, use the single quote " ' " or REM to include comment lines in the program.

(The "!" and " ' " are controller/language dependent, not interface dependent, and may vary for your system.)

Chapter 3

Advanced Programming



CHAPTER 3

ADVANCED PROGRAMMING

What You'll Learn in This Chapter

This chapter demonstrates advanced programming techniques. In the first section, traces are read from the spectrum analyzer and saved with the computer. In the second section, spectrum analyzer states are saved with the computer, and then the states are placed back into the spectrum analyzer. In the last section, a harmonic distortion program illustrates programming techniques used to develop spectrum analyzer measurements.

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

Note: All programming examples in this chapter for the HP-IB interface are written in HP BASIC 4.0, using an HP Series 200 Computer. Examples for the HP-IL interface are written in HP BASIC for the HP-71B Handheld Computer. If you use the HP-75D computer, you may need to do some code translation. For the RS-232 interface, examples are written in GWBASIC, using an HP Vectra or PC compatible.

To best understand this chapter, a general knowledge of the BASIC language is recommended. (Refer to your software documentation manuals.) Chapter 4 of this manual, "Programming Commands," defines analyzer commands alphabetically.

Controlling Trace Data With a Computer

The analyzer reserves an area in its memory, called an array, for trace information. Whenever a trace is swept, the analyzer updates the array with new data. Two separate traces may be alternately swept, trace A and trace B.

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.

(For the HP-IB Interface)

```

10 !FILE: "IBPROGS"
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 built in Chapter 2. First, we will modify the program to create a 401-point trace array, called "Trace_a," in which the trace data will be stored. Second, the program will use the TRA command to request trace A data. (The MKA and MKF commands are deleted from the original program.) Third, we will have the analyzer send the trace A data into the Trace_a variable.

(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 built in Chapter 2. First, we will modify the program to create a 401-point trace array, called "TRCA" in which the trace data will be stored (line 40). Second, the program will use the TRA command to request trace A data (line 180). (The MKA and MKF commands are deleted from the original program.) Third, we will have the analyzer send the trace A data into the TRCA variable (line 210).

(For the HP-IL Interface)

```
10 ! File = ILPROG5
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 OPTION BASE 1 ! set arrays to begin at index 1
60 ! create a 401-point trace array
70 REAL T(401)
80 ! find the loop address of the analyzer
90 ! and put it in the variable A
100 A=DEVADDR("HP8592A")
110 OUTPUT :A ;"IP;"
120 ! set the output format of the analyzer for
130 ! real numbers
140 OUTPUT :A ;"TDF P;"
150 ! set the analyzer parameters
160 OUTPUT :A ;"SNGLS;"
170 OUTPUT :A ;"CF 300MZ;"
180 OUTPUT :A ;"SP 200MZ;"
190 OUTPUT :A ;"TS;"
200 OUTPUT :A ;"MKPK HI;"
210 ! move peak to center of analyzer screen
220 OUTPUT :A ;"MKCF;"
230 OUTPUT :A ;"TS;"
240 ! ask the analyzer for trace data
250 OUTPUT :A ;"TA;"
260 ! enter the trace data into the computer
270 ! the process takes about 30 seconds
280 FOR I=1 TO 401
290 ENTER :A ;T(I)
300 NEXT I
310 OUTPUT :A ;"CONTS;"
320 END
```

In line 100, notice that the variable A, which contains the analyzer address, is not assigned the value 1. Instead, the variable A is assigned the value of the address of the analyzer using the "DEVADDR" statement. This statement has the computer find the address of the analyzer. The program is no longer dependent on the physical position of the analyzer in the loop. This type of addressing scheme can be applied to any device on the HP-IL loop.

The statement Option Base 1 (line 50) tells the computer that all arrays will start with the index of 1. The trace will be put into the array T, which is declared in line 70. There are 401 data points across the screen, so the array must be 401 points long.

The TA command requests that the trace data be sent to the computer (line 250). The TA command is used instead of the TRA? command, because the HP-71B requires that each number in the trace array sent by the analyzer be separated from each other number by a carriage return/line feed (rather than by commas, as provided by "TRA?,").

A "FOR NEXT" loop (lines 280-300) transfers the trace data from the analyzer to the array T.

Saving Trace Data

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

(For the HP-IB Interface)

```
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 disc in which to store 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, and 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.

The program is shown on the following page. Lines 20 through 210 are identical to the previous program.

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

(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 disc
240 OPEN "TRACEA" FOR OUTPUT AS #2
250 'print the trace data to the disc
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

```

First, we assign an input/output path to the file "TRACEA" (lines 200 through 220).

Using the "OPEN" command, we create an empty file on the disc in which to store the trace. Then we send the trace data to the file. (See lines 230 through 280.)

Lines 20 through 180 of 232PROG6 are identical to the previous program, 232PROG5.

(For the HP-IL Interface)

```
10 ! File = ILPROG6
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 OPTION BASE 1 ! set arrays to begin with an index of 1
60 ! create a 401-point trace array
70 REAL T(401)
80 ! find the address of the analyzer
90 ! and put it in the variable A
100 A=DEVADDR("HP 8592A")
110 OUTPUT :A ;"IP;"
120 ! set the output format of the analyzer for
130 ! real numbers
140 OUTPUT :A ;"TDF P;"
150 ! set the analyzer parameters
160 OUTPUT :A ;"SNGLS;"
170 OUTPUT :A ;"CF 300MZ;"
180 OUTPUT :A ;"SP 200MZ;"
190 OUTPUT :A ;"TS;"
200 OUTPUT :A ;"MKPK HI;"
210 ! move peak to center of analyzer screen
220 OUTPUT :A ;"MKCF;"
230 OUTPUT :A ;"TS;"
240 ! ask the analyzer for trace data
250 OUTPUT :A ;"TA;"
260 ! enter the trace data into the computer
270 ! the process takes about 30 seconds
280 FOR I=1 TO 401
290 ENTER :A ;T(I)
300 NEXT I
310 ! open file TRACEA and assign it to channel 1
320 ASSIGN #1 TO TRACEA
330 ! print the trace data to the file
340 PRINT #1;T()
350 ! close file and channel 1
360 ASSIGN #1 TO *
370 OUTPUT :A ;"CONTS;"
380 END
```

First, using the "ASSIGN" command, the program creates a channel for the file TRACEA (line 320). If the file does not exist, the HP-71B creates the new file. If the file exists, the HP-71B writes over the old file.

Next, send the trace data to the file (line 340). Finally, in line 360, the program closes the file. The program, ILPROG6, is shown below. Lines 20 through 300 are identical to the previous program, ILPROG5.

Reading Trace Data from a Computer Disc

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

(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(*).

(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 disc file TRACEA and stores data in the array variable TRCA.

(For the HP-IL Interface)

```
10 ! File = ILPROG7
20 DESTROY ALL
30 OPTION BASE 1
40 ! create 401-point trace array
50 REAL T(401)
60 ! open file TRACEA and assign it to channel 1
70 ASSIGN #1 TO TRACEA
80 ! read trace from file into T()
90 READ #1;T()
100 ! close file and channel 1
110 ASSIGN #1 TO *
120 END
```

First, in line 50, the program creates a 401-point trace array. Then, in line 70, the program assigns a path to the trace file. Finally, line 90 reads the trace data from the file "TRACEA" into the array variable T().

Saving and Recalling Instrument States

The 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 the nine state registers in the analyzer, in computer memory, or on a computer disc.

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

If you wish to save states in the analyzer, see the descriptions of the Save State (SAVES) and Recall State (RCLS) commands.

Saving the Analyzer's State

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

(For the HP-IB Interface)

```

10 !FILE: "IBPROG8"
20 !define 114 character string:
30 DIM Learn_string$[114]
40 !create 401 point array to store trace:
50 INTEGER Trace_a(1:401)
60 Analyzer=718
70 !set output format for two byte integers:
80 OUTPUT Analyzer;"TDF B;"
90 !ask analyzer for trace data:
100 OUTPUT Analyzer;"TRA?;"
110 !send trace to the computer:
120 ENTER Analyzer USING "#,W";Trace_a(*)
130 !get learnstring from analyzer:
140 OUTPUT Analyzer;"OL;"
150 ENTER Analyzer USING "#,114A";Learn_string$
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 disc.

Using the data stored in STATE, 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 screen.

Line 30 gives the dimensions of the learn string using the HP BASIC "DIM" command. Learn strings for the analyzer require 114 bytes of storage space. (Also see the Output Learn String (OL) command.)

When the trace and state data is sent from the analyzer to the computer, it must be formatted. Lines 120 and 150 format the trace data with the HP BASIC "USING" command. In the formatting statement, "#" indicates the statement is terminated when the last ENTER item is filled. (EOI and line-feed are item terminators, and early termination will result in an inaccurate learn string.) "W" specifies word format. "114A" 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 114-byte learn string is added to give 916 bytes total. This total is divided by 256 bytes per record, resulting in 4 records.)

Note: As in the previous program, if the program containing the CREATE command is run twice, the computer will report an error the second time as the file will already exist. To prevent this, place an exclamation mark before the CREATE command to "comment out" line 170 after the program has been run.

(For the RS 232 Interface)

```
10 'File = 232PROG8
20 OPEN "COM1:1200,N,8,1" AS #1
30 'Define 114_character string
40 DIM LEARN$(114)
50 'Create 802-character string to store trace data
60 DIM TR1$(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$(114,#1)
200 'create file to store trace on disc
210 OPEN "TRACEA" FOR OUTPUT AS #2
220 'change ASCII data to integers for disc storage
230 'because ASC 26 will put EOF on disc
240 DEFINT X,Y,I 'integer variables
250 DIM X1(114),Y1(802) 'arrays for the data
260 'first format the learn string
270 FOR I=1 TO 114
280 'get ASCII character from string
290 L2$=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 580 '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$,2,1)
500 Y1(802)=ASC(L2$)
510 'data is now formatted, write to disc
520 FOR I=1 TO 114
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
```

This program, 232PROG8, reads a trace from the analyzer and stores it in a 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 disc.

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 screen.

Line 40 gives the dimensions of the learn string using the HP BASIC "DIM" command. Learn strings for the analyzer require 114 bytes of storage space. (Also see the Output Learn String (OL) command.)

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.

For the HP-IL Interface)

```

10 ! File = ILPROG8
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 ! dimension trace and learnstring strings
60 DIM T$(802),L$(114)
70 A=DEVADDR("analyzer") ! analyzer address
80 ! set output format for two_byte integers
90 OUTPUT :A ;"TDF B;"
100 ! ask analyzer for trace data
110 OUTPUT :A ;"TRA?;"
120 ! enter trace data into string T$
130 ENTER :A USING "#,802A";T$
140 ! ask analyzer for learnstring
150 OUTPUT :A ;"OL;"
160 ! enter learnstring into string L$
170 ENTER :A USING "#,114A";L$
180 ! open file STATE and assign it to channel 1
190 ASSIGN #1 TO STATE
200 ! print the trace data and learnstring to the file
210 PRINT #1;T$,L$
220 ! close file and channel 1
230 ASSIGN #1 TO *
240 END

```

This program, ILPROG8, saves the state of the analyzer along with trace A in a file named "STATE". Using data stored in STATE, the analyzer settings can be reset according to the saved state. The stored trace data then can be replaced into the analyzer and viewed on the analyzer screen.

In ILPROG8, unlike ILPROG6, the trace data is taken from the analyzer in binary format. The binary format reduces the amount of memory space needed to store trace data by a factor of 4 when compared to real format. If it is necessary to analyze trace data with a computer program, use the technique shown in ILPROG6. Binary output format is set in line 90.

Line 60 dimensions the trace string (T\$) and learnstring string (L\$), using the HP BASIC "DIM" command. A binary-formatted trace from the analyzer takes up 802 bytes of storage space. Learn strings for the analyzer are 114 bytes long. (Also see the Output Learn String (OL) command.)

In line 130, the trace data is entered into the string T\$. The computer is told that there are 802 bytes of binary data to be entered with the USING option of the ENTER command. The format of the incoming data is defined in the format statement "#,802A".

In line 150, the OL command instructs the analyzer to send the analyzer learn string to the computer. In line 170, the learn string is entered into the variable L\$.

Returning the Analyzer to its Former State

The following three programs read a trace stored in a file and load the trace into the variable Trace_a(*).

(For the HP-IB Interface)

```
10 !FILE: "IBPROG9"
20 !define 114 character string:
30 DIM Learn_string$(114)
40 !create 401 point array to store trace:
50 INTEGER Trace_a(1:401)
60 Analyzer=718
70 !assign path to the file:
80 ASSIGN @File TO "STATE"
90 !get values for Learn_string$
100 !and Trace_a(*) from disc:
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" and loads it into the variable Trace_a(*).

The settings in force when the trace was stored are placed into the variable Learn_string\$. 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.

(For the RS 232 Interface)

```
10 'File = 232PROG9
20 OPEN "COM1:1200,N,8,1" AS #1
30 DEFINT X,Y,I 'integer variables
40 'define 114-character string
50 DIM LEARN$(114),X1(114)
60 'create an 802-character string to store disc data
70 DIM TR1$(200),TR2$(200),TR3$(200),TR4$(200)
80 DIM TR5$(2),TR6$(200),Y1(802)
90 'open disc file "TRACEA"
100 OPEN "TRACEA" FOR INPUT AS #2
110 'enter learn array from disc
120 FOR I=1 TO 114
130 INPUT #2,X1(I) 'get integer variable from disc
140 NEXT I
150 'enter trace data from disc
160 FOR I=1 TO 802
170 INPUT #2,Y1(I)
180 NEXT I
190 'close the disc file
200 CLOSE #2
210 'format the integer data into strings
220 'for the analyzer. see 232PROG8 for explanation.
230 LEARN$="" 'null out the learn string
240 FOR I=1 TO 114 'format learn string first
250 LEARN$=LEARN$+CHR$(X1(I))
260 NEXT I
270 'format the trace data
280 I2=1 'set counter
290 GOSUB 500 'do the conversion
300 TR1$=TR6$ 'set the string
310 I2=201 'set the counter
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$;";
470 PRINT #1,"VIEW TRB;"
480 GOTO 560 'end the program
490 'subroutine for converting integer data to ASCII
500 TR6$="" '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 disc in the previous program example, 232PROG8.

First, the settings of the analyzer in force when the trace was stored are placed into the variable LEARN\$. Then, previously stored trace data is returned to the analyzer and the trace is viewed on the analyzer screen.

(For the HP-IL Interface)

```

10 ! File = ILPROG9
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 ! dimension trace and learnstring strings
60 DIM T$(802),L$(114)
70 A=DEVADDR("analyzer") ! ANALYZER ADDRESS
80 ! open the file STATE and assign it to channel 1
90 ASSIGN #1 TO STATE
100 ! read the trace data and learnstring into
110 ! the strings T$ and L$
120 READ #1;T$,L$
130 ! close file and channel 1
140 ASSIGN #1 TO *
150 OUTPUT :A ;"IP DONE;"
160 ENTER :A ;Z
170 ! send learnstring to analyzer
180 OUTPUT :A USING "114A";L$
190 ! send trace data to Trace B of analyzer
200 OUTPUT :A USING "2A,802A";"IB";T$
210 ! view Trace B to see it was sent
220 OUTPUT :A ;"VIEW TRB;"
230 END

```

The HP-IL program, ILPROG9, reads a trace and a learnstring stored in the file "STATE" and loads them into the analyzer.

Line 180 formats and sends the learnstring to the analyzer. The analyzer recognizes that it is a learnstring and performs the appropriate operations. Line 200 formats and appends the command IB to the trace data before it sends it to the analyzer. This puts the trace data into trace B. Line 220 allows us to view the previously sampled and stored data.

(The Input Trace B (IB) command transmits the contents of an array, located in the controller, to trace B memory. This command is used specifically in this HP-IL program and is not documented in Chapter 4.)

Measuring Harmonic Distortion

The harmonic distortion program presented here illustrates how the 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 from a formula. Extensive annotation has been added to help you understand the programs.

The following programs measure percent of total harmonic distortion automatically, quickly, and accurately. The programs operate as if we were to make the measurement manually.

(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 Pront_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),Harmonic_dbc(2:Max_harmonic)
100 GOSUB Clearscreen !clear the alpha screen
110 !ask for the frequency of the fundamental:
120 OUTPUT CRT USING "4/,10X,K,3/";"**** HARMONIC DISTORTION ****"
130 OUTPUT CRT USING "K,/";"CONNECT SOURCE TO ANALYZER INPUT, THEN"
140 OUTPUT CRT USING "K,2/";"ENTER FREQUENCY OF FUNDAMENTAL IN MHZ "
150 OUTPUT CRT USING "10X,K";"WHEN READY => press ENTER "
160 INPUT Fundamental
170 GOSUB Clearscreen !clear the alpha screen
180 Fundamental: !write "measuring fundamental" on screen:
190 DISP "MEASURING FUNDAMENTAL"
200 !preset the analyzer, set single sweep mode, and take sweep:
210 OUTPUT @Sa;"IP; SNGLS; TS;"
220 !tune the analyzer to the fundamental freq and set 20 MZ span:
230 OUTPUT @Sa;"CF ";Fundamental;"MHZ;"
240 OUTPUT @Sa;"SP 20MZ; TS;"
250 !put a marker on signal peak, move marker to reference level:
260 OUTPUT @Sa;"MKPK HI; MKRL; TS;"
270 !find signal peak, activate signal track, and narrow span:
280 OUTPUT @Sa;"MKPK HI; TS;"
290 OUTPUT @Sa;"MKTRACK ON; SP 100KZ; TS;"
300 !turn off signal track:
310 OUTPUT @Sa;"MKTRACK OFF;"
320 !find the peak of the signal; move peak to center of screen:
330 OUTPUT @Sa;"AUNITS V;"! MAKE READOUT UNITS VOLTS
340 !find peak of signal; send amplitude value to computer
350 !enter the amplitude of the fundamental:
360 OUTPUT @Sa;"MKPK HI; MKA?;"
370 ENTER @Sa;Fund_amptd_v
380 !send marker frequency to the computer, enter frequency value:
390 OUTPUT @Sa;"MKF?;"
400 ENTER @Sa;Fundamental

```

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

```
920 !the statement below presses the "CLR SCR" key on the keyboard:
930 OUTPUT KBD USING "#,B";255,75
940 RETURN
950 END
```

The HP-IB 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 percent of harmonic distortion results, plus harmonic amplitudes in dBc (decibels relative to the carrier), are displayed on the computer screen.

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

(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 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 20-MZ 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 reference level
280 PRINT #1,"MKPK HI;MKRL;TS;"
290 'find signal peak, activate signal track, and 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 computer
400 PRINT #1,"MKPK HI;MKA?;"
410 INPUT #1,FUNDAMPTDV
420 'send marker frequency to computer, enter frequency value
430 PRINT #1,"MKF?;"
440 INPUT #1,FUNDAMENTAL
450 'make the fundamental frequency the center freq 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, activate the signal track
580 PRINT #1,"TS;"
585 PRINT #1,"MKPK HI;MKTRACK ON;SP 100KZ;TS;"
590 'turn off signal track
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 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;" dbm"
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 percent of harmonic distortion results, plus harmonic amplitudes in dBc (decibels relative to the carrier), are displayed on the computer screen.

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

(For the HP-IL Interface)

```
10 ! File = THDTEST
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 ! F1 is the frequency of the fundamental
60 ! F2 is the amplitude of the fundamental in volts
70 ! F3 is the amplitude of the fundamental in dBm
80 REAL F1,F2,F3
90 ! P1 is the percent total harmonic distortion
100 ! S1 is the sum of the squares of the harmonics
110 REAL P1,S1
120 ! H1() are the harmonic amplitudes in volts
130 ! H2() are the harmonic amplitudes in dBc
140 REAL H1(10),H2(10)
150 ! M1 is the number of the max harmonic to be measured
160 ! I is the loop index used in the program
170 ! X,Y is the starting position of output data printout
180 INTEGER M1,I,X,Y
190 A=DEVADDR("analyzer")
200 M1=4 ! change this line to change max harmonic measured
210 DISP "HARMONIC DISTORTION"
220 ! turn off the annotation and the graticule
230 ! blank trace A and blank trace B
240 OUTPUT :A ;"ANNOT OFF;GRAT OFF;BLANK TRA;BLANK TRB;"
250 ! set pen to absolute mode
260 ! move the pen to the position -20,120 on the screen
270 OUTPUT :A ;"PU;PA-20,120;"
280 ! print the message at the current pen position
290 OUTPUT :A ;"TEXT/CONNECT SOURCE TO ANALYZER INPUT/"
300 OUTPUT :A ;"PU;PA-20,100;"
310 OUTPUT :A ;"TEXT!ENTER FREQUENCY OF FUNDAMENTAL IN MHZ ON 71B KEYBOARD!"
320 DISP "MHZ?";
330 INPUT "";F1
340 ! erase the screen
350 OUTPUT :A ;"EM;"
360 OUTPUT :A ;"PU;PA-20,120;"
370 OUTPUT :A ;"TEXT!PRESS ENDLINE WHEN READY TO START MEASUREMENT!;"
380 INPUT "";ZS
390 ! FUNDAMENTAL MEASUREMENTS
400 ! preset the analyzer, set single_sweep mode, and take sweep
410 OUTPUT :A ;"IP;SNGLS;TS;"
420 ! put MEASURING FUNDAMENTAL in title block
430 OUTPUT :A ;"TITLE! MEASURING FUNDAMENTAL!"
440 ! tune the analyzer to the fundamental freq and set 20 MZ span
450 OUTPUT :A ;"CF ";F1;"MHZ;"
460 OUTPUT :A ;"SP 20MZ;TS;"
470 ! put marker on signal peak, move marker to reference level
480 OUTPUT :A ;"MKPK HI;MKRL;TS;"
490 ! find signal peak, activate signal track, and narrow the span
500 OUTPUT :A ;"MKPK HI;TS;"
510 OUTPUT :A ;"MKTRACK ON;SP 100KZ;TS;"
520 ! turn off signal track
530 OUTPUT :A ;"MKTRACK OFF;"
540 ! make readout units volts
```

```
550 OUTPUT :A ;"AUNITS V;"
560 ! find peak of signal, ask for maker amplitude
570 ! enter the amplitude of the fundamental
580 OUTPUT :A ;"MKPK HI;MKA?;"
590 ENTER :A ;F2
600 ! ask for the marker frequency
610 ! enter the frequency of the fundamental
620 OUTPUT :A ;"MKF?;"
630 ENTER :A ;F1
640 ! make the fundamental frequency the center freq step size
650 OUTPUT :A ;"MKSS;"
660 ! set the fundamental frequency units in MHZ
670 F1=F1/1000000
680 ! HARMONIC MEASUREMENTS
690 FOR I=2 TO M1
700 OUTPUT :A ;"TITLE! MEASURING HARMONIC # ";STR$(I)[1,1];"!"
710 OUTPUT :A ;"SP 20MZ;"
720 OUTPUT :A ;"CF UP;TS;" ! tune to next harmonic
730 ! take second sweep to allow analyzer to move to the center
740 ! frequency; find the signal peak; activate signal track;
750 ! narrow the span;
760 OUTPUT :A ;"TS;"
770 OUTPUT :A ;"MKPK HI;MKTRACK ON;SP 100KZ;TS;"
780 ! turn off signal track
790 OUTPUT :A ;"MKTRACK OFF;"
800 ! find signal peak; ask for marker amplitude
810 ! enter the amplitude of the harmonic
820 OUTPUT :A ;"MKPK HI;MKA?;"
830 ENTER :A ;H1(I)
840 NEXT I
850 ! set amplitude units to dBm
860 OUTPUT :A ;"AUNITS DBM;"
870 ! calculate the fundamental amplitude in dBm since
880 ! it was measured in volts
890 F3=10*LGT(F2^2/.05)
900 ! calculate the sum of the squares of the amplitudes
910 ! of the harmonics; calculate amplitudes of harmonics (dBm)
920 S1=0
930 FOR I=2 TO M1
940 S1=S1+H1(I)^2
950 H2(I)=20*LGT(F2/H1(I))
960 NEXT I
970 ! calculate the percent distortion
980 P1=SQR(S1)/F2*100
990 ! output the data onto the analyzer screen
1000 ! clear the title block
1010 OUTPUT :A ;"TITLE"
1020 OUTPUT :A ;"ANNOT OFF;GRAT OFF;BLANK TRA;BLANK TRB;"
1030 ! use fixed data format with two decimal places
1040 FIX 2
1050 ! set the starting x and y position of the printed data1060 X=50
1070 Y=150
1080 OUTPUT :A ;"PU;PA";X;" ";Y;" ";
1090 OUTPUT :A ;"TEXT/HARMONIC RESULTS/"
1100 OUTPUT :A ;"PU;PA";X;" ";Y-20;" ";
1110 OUTPUT :A ;"TEXT/FUND FREQ = ";F1;" MHZ/"
```

```
1120 OUTPUT :A ;"PU;PA";X;",";Y-30;";"  
1130 OUTPUT :A ;"TEXT/FUND AMP = ";F3;" dBm/"  
1140 OUTPUT :A ;"PU;PA";X;",";Y-40;";"  
1150 OUTPUT :A ;"TEXT/2nd HARMONIC = ";H2(2);" dBc/"  
1160 OUTPUT :A ;"PU;PA";X;",";Y-50;";"  
1170 OUTPUT :A ;"TEXT/3rd HARMONIC = ";H2(3);" dBc/"  
1180 FOR I=4 TO M1  
1190 OUTPUT :A ;"PU;PA";X;",";Y-(I+2)*10;";"  
1200 OUTPUT :A ;"TEXT/";STR$(I)[1,1];"th HARMONIC = ";H2(I);" dBc/"  
1210 NEXT I  
1220 OUTPUT :A ;"PU;PA";X;",";Y-(I+2)*10;";"  
1230 OUTPUT :A ;"TEXT/TOTAL HARMONIC DISTORTION = ";P1;" %/"  
1240 END
```

The HP-IL program shows how to use the signal track and auto-zoom feature of the analyzer from a computer. It also demonstrates that the 22-character, single-line display of the HP-71B is not a limiting factor. The analyzer can easily display prompts and data.

On the analyzer screen, the program prompts the operator to connect a source to the analyzer INPUT and enter the source frequency. The program then sets the analyzer center frequency to the value entered through the HP-71B.

Next, the program measures and records the frequency and amplitude of the fundamental; then it does the same thing with the second, third, and fourth harmonics. The analyzer displays the name of the signal being measured in the TITLE area of the display. These values are used to compute the percent of harmonic distortion. The harmonic amplitudes in dBc (decibels relative to the carrier), are also displayed on the analyzer screen.

To save memory space, the comments may be deleted from the program. This can reduce the memory taken up by a program by more than 50%.

If necessary, you can change the number of harmonics on line 200.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

Chapter 4

Programming Commands



Vertical line of text on the left margin, possibly a page number or reference code.

Vertical line of text on the right margin, possibly a page number or reference code.

CHAPTER 4

PROGRAMMING COMMANDS

What You'll Learn in This Chapter

This chapter is a reference for the HP 8592A command language. In this chapter are descriptions of commands recognized by the HP 8592A Spectrum Analyzer. This chapter is a command dictionary; commands are listed alphabetically.

To find a programming command which performs a particular function, first refer to the Functional Index (Table 4-5) where commands are categorized by function. Once the desired command is found, refer to the alphabetical listing.

If you are totally unfamiliar with spectrum analyzer programming, do not start with this chapter. New programmers should refer to Chapter 2, "Programming Fundamentals."

This chapter includes several reference tables as listed below:

- Table 4-1 Command Syntax Summary
- Table 4-2 Syntax Elements Enclosed in Rectangles
- Table 4-3 Command Parameters
- Table 4-4 Summary of Compatible Commands
- Table 4-5 Functional Index

Syntax Conventions

Command syntax is represented pictorially. Many commands have identical syntax diagrams. Seven of the most common diagrams are shown in Figures 4-1 through 4-7; these diagrams are referred to throughout the remainder of Chapter 4.

Symbols in Syntax Diagrams

- Narrow ovals surround command names.
- Circles and wide ovals surround secondary keywords, or special numbers and characters.
- Rectangles contain the description of a syntax element defined in Table 4-2.
- Solid lines represent the recommended path.
- Semicolons are the recommended command terminators. Semicolons are inserted at the end of each line and between spectrum analyzer commands. Using semicolons makes programs easier to read, prevents command misinterpretation, and is recommended by IEEE Standard 728. Optional command terminators (line feed, carriage return, space, and comma) can replace semicolons as terminators in most instances.



Figure 4-1

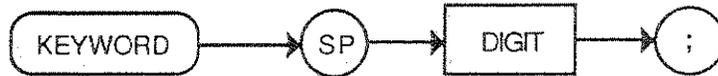


Figure 4-2

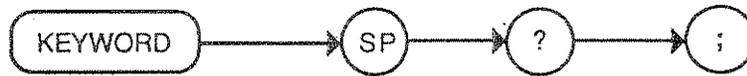


Figure 4-3

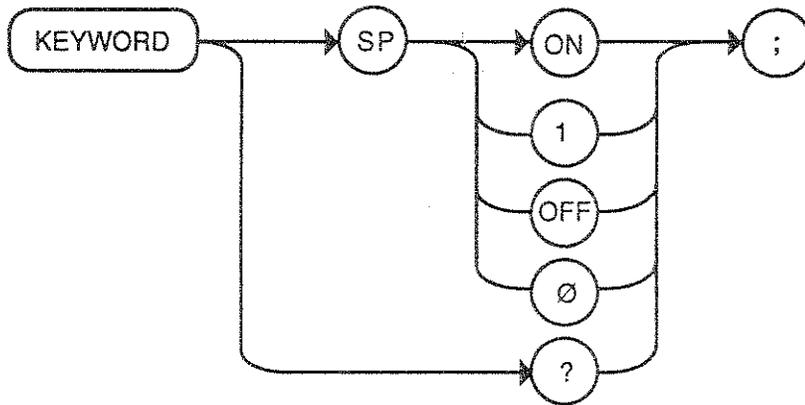


Figure 4-4

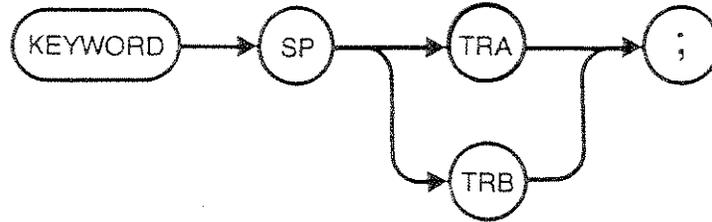


Figure 4-5

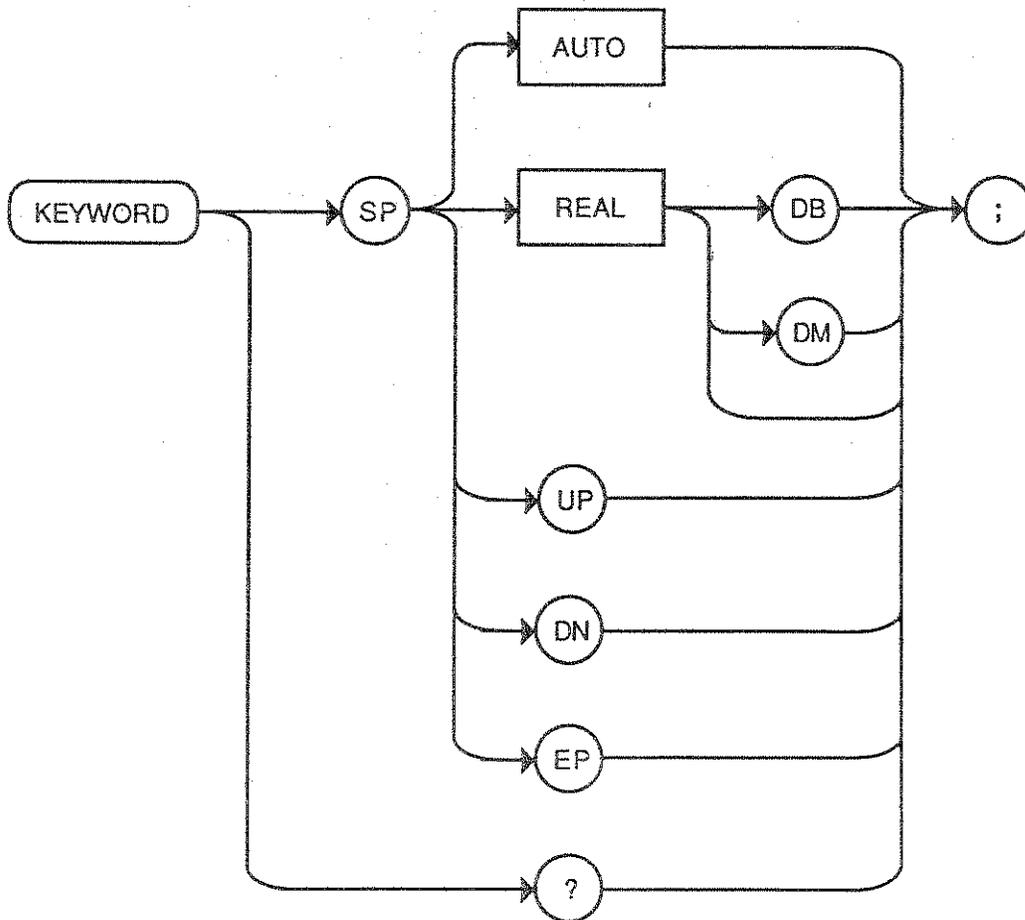


Figure 4-6

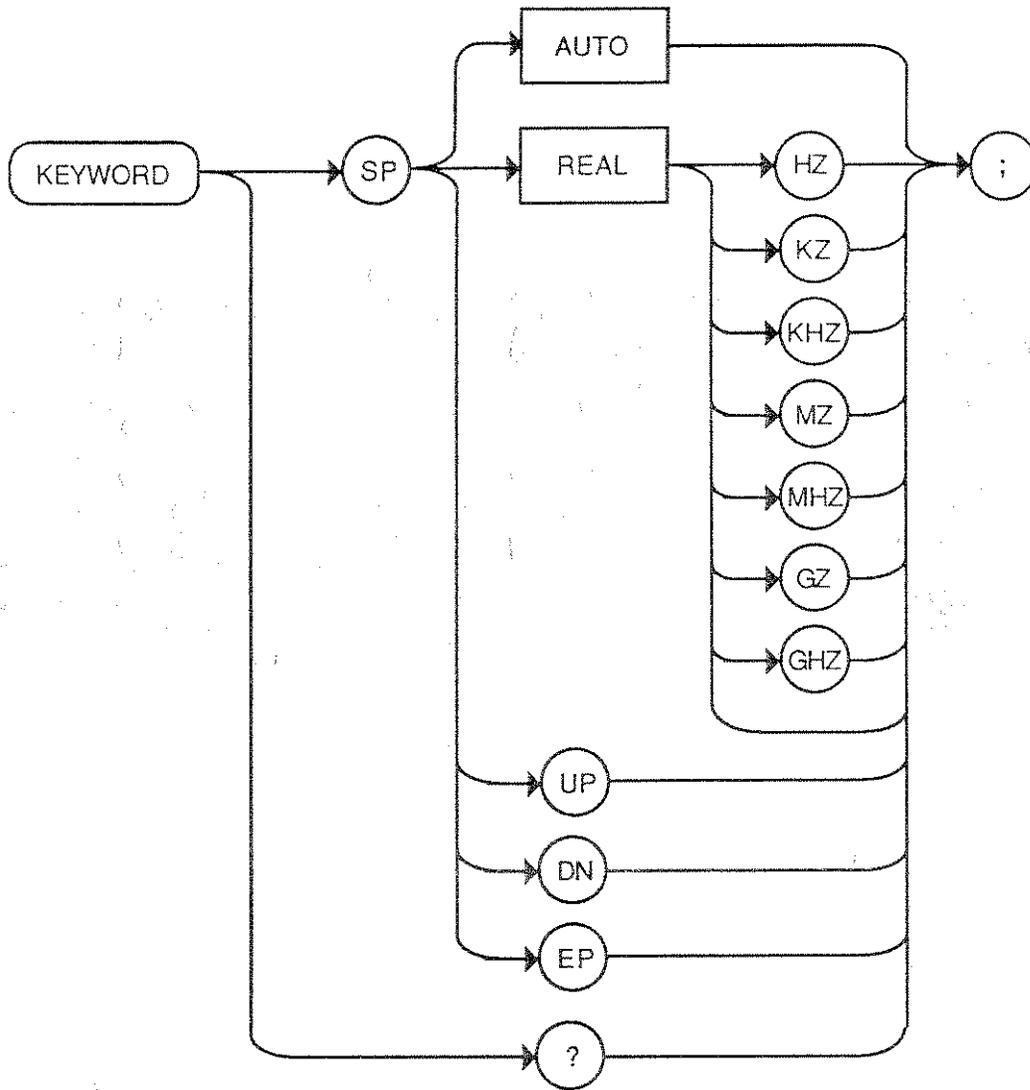


Figure 4-7

Table 4-1. Command Syntax Summary

Command	Syntax Diagram
AMB	Figure 4-4*
AMBPL	Figure 4-4
ANNOT	Figure 4-4
APB	Figure 4-1
AT	Figure 4-6*
AUNITS	See Command Description
AUTO	See Command Description
AXB	Figure 4-1
BLANK	Figure 4-5
BL	Figure 4-1
CAL	See Command Description
CF	Figure 4-7
CLRW	Figure 4-5
CLS	Figure 4-1
CNF	Figure 4-1
CONTS	Figure 4-1
DET	See Command Description
DL	Figure 4-6*
DN	Figure 4-1
DONE	Figure 4-1*
DSPLY	See Command Description
EE	Figure 4-1
EK	Figure 4-1
EM	Figure 4-1
EP	Figure 4-1
FA	Figure 4-7
FB	Figure 4-7
FFT	See Command Description
FOFFSET	See Command Description
GRAT	Figure 4-4
HD	Figure 4-1
HN	Figure 4-3
HNLOCK	See Command Description
HNUNLK	Figure 4-1
ID	Figure 4-1
INZ	See Command Description
IP	Figure 4-1
LG	Figure 4-6*
LN	Figure 4-1
MA	Figure 4-1
MDS	See Command Description
MF	Figure 4-1
MKA	Figure 4-6*
MKCF	Figure 4-1
MKD	Figure 4-7*
MKF	Figure 4-7*
MKMIN	Figure 4-1

Table 4-1. Command Syntax Summary (continued)

Command	Syntax Diagram
MKN	Figure 4-7
MKNOISE	Figure 4-4
MKOFF	See Command Description
MKPAUSE	See Command Description
MKPK	See Command Description
MKPX	Figure 4-6*
MKRL	Figure 4-1
MKSP	Figure 4-1
MKSS	Figure 4-1
MKTRACK	Figure 4-4
MKTYPE	See Command Description
ML	Figure 4-6*
MXMH	Figure 4-5
M4	Figure 4-7*
OA	Figure 4-1
OL	Figure 4-1
PA	See Command Description
PD	Figure 4-1
PEAKS	See Command Description
PLOT	See Command Description
PP	Figure 4-1
PR	See Command Description
PRINT	Figure 4-1
PRSDAC	See Command Description
PSTATE	Figure 4-4
PU	Figure 4-1
RB	Figure 4-7
RCLS	Figure 4-2
RCLT	See Command Description
REV	Figure 4-1
RL	Figure 4-6*
ROFFSET	See Command Description
RQS	See Command Description
SAVES	See Command Description
SAVET	See Command Description
SNGLS	Figure 4-1
SP	Figure 4-7*
SRQ	Figure 4-2
SS	Figure 4-7*
ST	See Command Description
STB	See Command Description
TA	Figure 4-1
TB	Figure 4-1
TDF	See Command Description
TEXT	See Command Description
TH	Figure 4-6*
TITLE	See Command Description

Table 4-1. Command Syntax Summary (continued)

Command	Syntax Diagram
TM	See Command Description
TRA/TRB	See Command Description
TRCMEM	Figure 4-3
TS	Figure 4-1
TWNDOW	See Command Description
VAVG	See Command Description
VB	Figure 4-7
VBR	See Command Description
VIEW	Figure 4-5

*Slight differences may occur in syntax diagrams; refer to specific Syntax descriptions for complete variances.

Table 4-2. Syntax Elements Enclosed in Rectangles

Element	Description
A-Block Data Field	Absolute block data field consisting of #, A, Length
Command List	Alphanumeric character comprising any analyzer command. ASCII characters 0 through 255.
Data Bytes	8-bit bytes representing ASCII command list or trace data.
Delimiter	String Delimiter ! " \$ % & ' / : = @
Detector Mode	Detection mode specifier. See command description for DET.
I-Block Data Field	Indefinite block data field consisting of #, I, and Data Bytes with END or Identity line asserted on Last Data Byte.
Length	Two 8-bit bytes specifying number of 2-bit bytes. A-Block Data Field. The most significant byte is first: MSB LSB.
Output Termination	Carriage-return/line-feed with End-or-identify (EOI) asserted.
P1X and P1Y	Integer representing plotter-dependent values that specify lower-left plotter dimension.
P2X and P2Y	Integer representing plotter-dependent values that specify upper-right plotter dimension.
String Delimiter	! " \$ % & ' / : = @
Trace Element	Any element (point) of trace A or B.

Table 4-3. Command Parameters
(Syntax Elements Enclosed in Circles)

Element	Description
ALL	all
AMP	amplitude
AUTO	auto couple (command)
B	8-bit byte
DB	decibel (unit)
DBM	absolute decibel milliwatt (unit)
DBMV	decibel millivolt
DBUV	decibel microvolt
DM	absolute decibel milliwatt (unit)
DN	decreases the parameter one step size (command)
EP	pauses program for data entry from front panel
EXT	external
FIXED	fixed
FREE	free run
FRQ	frequency
GHZ	gigahertz (unit)
GZ	gigahertz (unit)
HI	highest
HZ	hertz (unit)
KHZ	kilohertz (unit)
KZ	kilohertz (unit)
LINE	line, as power line
MHZ	megahertz (unit)
MS	millisecond (unit)
MV	millivolts (unit)
MZ	megahertz (unit)
NH	next highest
NL	next left
NR	next right
OA	Output active function value (command)
OFF	turn function off
ON	turn function on
PER	period
POS	positive peak detection mode
PSN	position
SC	seconds (unit)
SMP	sample detection mode
SP	space
SWT	sweep time
TRA	trace A
TRB	trace B
UP	increases the parameter one step size (command)
UV	microvolts (unit)

*Table 4-3. Command Parameters (continued)
(Syntax Elements Enclosed in Circles)*

Element	Description
US	microseconds (unit)
V	volts (unit)
VID	video
W	word
?	Returns a query response containing the value or state of the associated parameter. The query response is followed by a carriage-return/line-feed.

Table 4-4. Summary of Compatible Commands

The alternate commands (Alt. Comm.) listed in the left-hand column are supported by the HP 8592A and are provided to allow for backward compatibility with commands in the HP 8566A/B, HP 8568A/B, and HP 71000 instruments. The new equivalent HP 8592A commands, listed in the right-hand column, may also be used.

Alt. Comm.	Description	HP 8592A 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
BL	B — DL → B	BML
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
CA	couple input attenuation	AT AUTO
CR	couple resolution bw	RB AUTO
CS	couple step size	SS AUTO
CT	couple sweep time	ST AUTO
CV	couple video bw	VB AUTO
C1	A — B off	AMB OFF
C2	A — B on	AMB ON
EX	exchange trace A and B	AXB
E1	peak search	MKPK HI
E2	enter marker frequency into center frequency	MKCF
E3	enter marker delta into center frequency step size	MKSS
E4	enter marker amplitude into reference level	MKRL
KSA	dBm amplitude units	AUNITS DBM
KSB	dBmV amplitude units	AUNITS DBMV
KSC	dB μ 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
KS _n	graticule on	GRAT ON

Table 4-4. Summary of Compatible Commands (Continued)

Alt. Comm.	Description	HP 8592A Command
KSm KSo KSp	graticule off annotation on annotation off	GRAT OFF ANNOT ON ANNOT OFF
LO	display line off	DL OFF
MT0 MT1 M1 M2 M3	marker track off marker track on marker off marker normal marker delta	MKTRACK OFF MKTRACK ON MKOFF MKN MKD
O1 O2 O3 O4	output format — real output format — word output format — measurement unit output format — byte	TDF P TDF B MDS W TDF M TDF B MDS B
RC R1 R2 R3 R4	recall states activates illegal command service request only activates end-of-sweep, illegal command activates broken hardware, illegal command activates units-key pressed, illegal command	RCLS RQS 32 RQS 36 RQS 40 RQS 34
SV S1 S2	save state sweep continuous sweep single	SAVES CONTS SNGLS
T0 T1 T2 T3 T4	threshold off trigger mode free run trigger mode line trigger mode external trigger mode video	TH OFF TM FREE TM LINE TM EXT TM VID

Table 4-5. Functional Index

This functional index categorizes the HP 8592A commands by type or function. To find a programming command which performs a particular function, first refer to the index below. The command and a brief definition are shown. Once the desired command is found, refer to the alphabetical listing for further command definition.	
Command	Description
Amplitude Commands	
AT	specifies input attenuation
AUNITS	specifies amplitude units for input, output, and display
INZ	specifies input impedance
LG	selects log scale
LN	selects linear scale
MKRL	moves active marker to reference level
ML	specifies mixer level
RL	specifies reference level
ROFFSET	specifies reference level offset
Bandwidth Commands	
RB	specifies resolution bandwidth
VB	specifies video bandwidth
VBR	specifies coupling ratio of video bandwidth to resolution bandwidth
Calibration and Diagnostic Control	
CAL	calibrate
CNF	confidence test
Coupling Control	
AUTO	recouples specified function or recouples all functions
Display Commands	
ANNOT	turns annotation on or off; preset condition is on
AUNITS	specifies amplitude units for input, output, and display
DL	specifies display line level in dBm
DSPLY	writes the value of a variable on the analyzer screen
GRAT	turns graticule on or off; preset condition is on
HD	holds or disables data entry and blanks active function readout
LG	selects log scale
LN	selects linear scale
TH	specifies display threshold value
TITLE	writes text string to the top line of the analyzer screen

Table 4-5. Functional Index (Continued)

Command	Description
Frequency Commands	
CF	specifies center frequency
FA	specifies start frequency
FB	specifies stop frequency
FOFFSET	specifies frequency offset
HN	specifies harmonic number
HNLOCK	specifies harmonic lock
HNUNLK	specifies harmonic unlock
SP	specifies frequency span
SS	specifies center frequency step size
Graphics Commands	
EM	removes graphics from screen
PA	draws vectors to specified x and y positions
PD	pen down (for drawing graphics)
PR	draws vector from last absolute position (plot relative)
PRINT	prints screen data
PU	pen up (for drawing blank vectors)
TEXT	writes text string to screen at current pen position
TITLE	writes text string to top line of analyzer screen
Information and Service Diagnostic Commands	
CLS	clears status byte
ID	returns the HP model number of the analyzer used (HP 8592A)
REV	returns the firmware revision number
RQS	provides the decimal weighting of service request mask bits which are enabled for service requests
SRQ	sets service request
STB	reads status byte, then clears status byte
Input/Output Commands	
EE	enable entry
EK	enables front-panel knob control
EP	enables front-panel number entry
DONE	sends a message to controller after preceding commands are initiated
HD	holds or disables data entry and blanks active function readout
MA	returns marker amplitude
MDS	specifies measurement data size to byte or word; preset condition is word
MF	returns marker frequency
MKA	specifies amplitude of active marker
MKF	specifies frequency of active marker
OL	returns learn string
RQS	provides the decimal weighting of service request mask bits which are enabled for service requests

Table 4-5. Functional Index (Continued)

Command	Description
Input/Output Commands (continued)	
TA TB TRA/TRB TDF TS	trace A output control trace B output control trace data input/output control selects trace data output format takes a sweep
Instrument State Commands	
IP RCLS SAVES	instrument preset recall state save state
Marker Commands	
MA MF MKA MKCF MKD MKF MKMIN MKN MKNOISE MKOFF MKPAUSE MKPK MKPX MKRL MKSP MKSS MKTRACK MKTYPE M4	returns marker amplitude returns marker frequency specifies amplitude of active marker, returns marker amplitude value enters marker frequency into center frequency moves delta marker to specified frequency or activates delta marker specifies frequency of active marker, returns marker frequency for non-zero span (or time in zero span) moves active marker to minimum signal detected moves active marker to specified frequency (or time in zero span) returns average noise value at marker, normalized to 1 Hz bandwidth connection for res bandwidth and detection mode turns off all markers or turns off active marker pauses sweep at marker for duration of specified delay time (in seconds) moves active marker to maximum signal detected specifies minimum excursion for peak identification; preset value is 6 dB moves active marker to reference level moves marker delta frequency into span moves marker to center frequency step size turns signal track on or off specifies the marker type turns on marker zoom
Operator Entry Commands	
EE EK HD	enables front panel data number entry enables front panel knob control holds or disables entry and blanks active function readout
Plotter/Printer Commands	
PLOT PRINT	plots screen data prints screen data

Table 4-5. Functional Index (Continued)

Command	Description
Sweep and Trigger Commands	
CONTS SNGLS ST TM TS	selects continuous-sweep mode selects single-sweep mode specifies sweep time specifies trigger mode takes a sweep
Synchronization Commands	
TS DONE	takes a sweep sends message to controller after preceding commands are executed
Trace Commands	
AUNITS BLANK CLRW DET FFT PEAKS RCLT SAVET TA TB TRA/TRB TRCMEM TWINDOW VIEW	specifies amplitude units for input, output, and display stores and blanks specified trace register clear-writes specified trace register specifies detection mode initiates Fast Fourier Transform specifies trace peaks recalls a saved trace saves a trace trace A output trace B output trace data input/output control trace memory specifies trace window for FFT views specified trace register
Trace Math Commands	
AMB AMBPL APB AXB BL VAVG	A - B into A A - B + display line into A A + B into A exchanges A and B B - DL into B turns video average on or off, as well as setting the number of counts to average

Command Descriptions

This section is a command dictionary; commands are listed alphabetically by their programming names. Each command entry includes a command description and a syntax diagram or a reference to one of the seven common syntax diagrams shown in Figures 4-1 through 4-7 (at the beginning of this chapter).

To find a programming command which performs a particular function, first refer to the Functional Index in the preceding pages (Table 4-5). The commands are grouped by function. Once the desired command is found, refer to this section for further command definition and syntax information.

Note: Most command descriptions provide an example of command usage. The examples use an HP BASIC "OUTPUT 718" output statement that sends the commands inside the quote fields to the analyzer. If you are using a different computer or language, you will need to substitute a different output statement in your programs. To determine the appropriate output statement, refer to the examples provided earlier in this manual and to your computer documentation.

Example: For HP-IB, HP BASIC, you may use: `OUTPUT 718;"IP;"`
For HP-IL, HP BASIC, you may use: `OUTPUT:A;"IP;"`
For RS-232, GW BASIC, you may use: `PRINT #1,"IP;"`

Note: UPPERCASE is recommended and preferred for entering all commands.

AMB

Trace A Minus Trace B

Description: The AMB command subtracts trace B from trace A, point by point, and sends the difference to trace A.

Syntax: See Figure 4-4, without query.

Example: OUTPUT 718;"AMB ON;"

Note: The functions of the AMB and C1 or C2 commands and the [A — B → A] softkey are identical. AMB returns "ON" or "OFF" if queried.

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 result and sends the answer to trace A.

Syntax: See Figure 4-4

Example: OUTPUT 718;"AMBPL;"

ANNOT

Annotation

Description: The ANNOT command turns on or off all the words and numbers (annotation) on the analyzer screen (except softkey labels). When queried, ANNOT reports the annotation state: ON or OFF.

Syntax: See Figure 4-4

Example: OUTPUT 718;"ANNOT ON;"

Note: Alternate command KSo turns the annotation off and KSp turns the annotation on. The functions of the ANNOT command and [ANOTATN ON OFF] softkey are identical.

APB

Trace A Plus Trace B

Description: The APB command adds trace A and trace B, point by point, and sends the result to trace A.

Syntax: See Figure 4-1

Example: OUTPUT 718;"APB;"

AT

Attenuation

Description: The AT command specifies the input attenuation from 0 to 70 dB, in 10 dB steps. Normally, the input attenuator couples 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 automatically change. If the attenuation is decreased from the coupled value by the AT command, the reference level may decrease. 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 couples.

Executing 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.

When queried with ? or OA, AT returns the attenuation value as a real number, followed by a carriage-return/line-feed.

Only direct entry of "AT 0DB;" will achieve 0-dB attenuation. The step keys, knob, and DN parameter will stop at 10dB.

Syntax: See Figure 4-6

Example:

```
10 OUTPUT 718;"AT 40;"  
20 OUTPUT 718;"AT UP;"
```

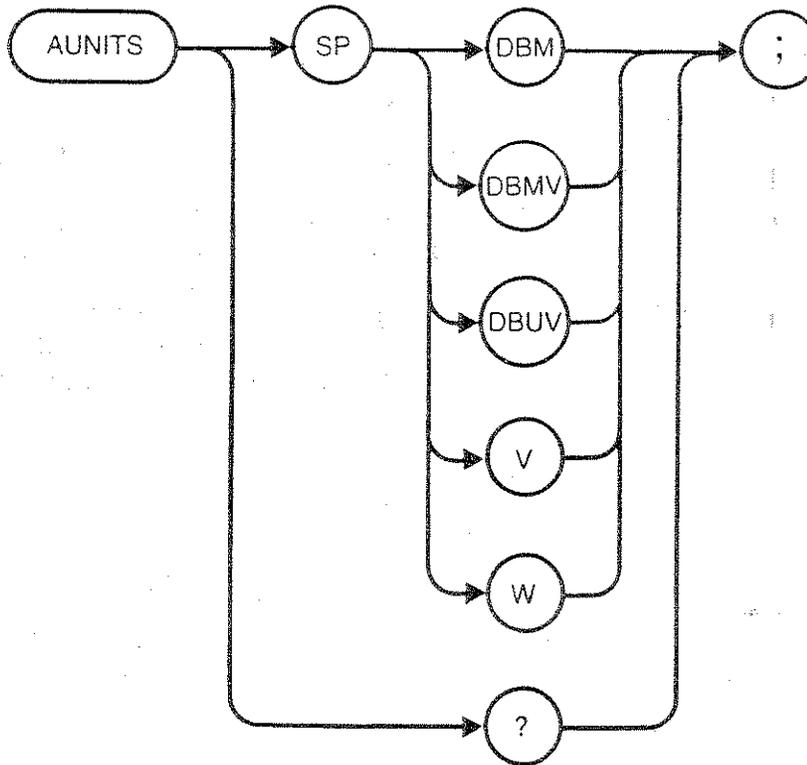
Line 10 sets the attenuation to 40 dB. Line 20 sets the attenuation to 50 dB.

AUNITS

Amplitude Units

Description: The AUNITS command sets the amplitude readouts (reference level, marker, display line, and threshold) to the specified units.

Syntax:



Example: OUTPUT 718; "AUNITS DBMV;"

Note: Alternate commands allow you to set the amplitude units: KSA sets dBm units, KSB sets dBmV units, KSC sets dB μ V units, and KSD sets volts.

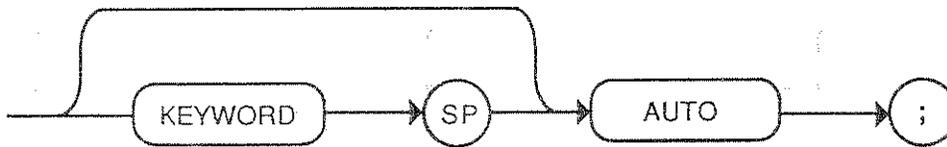
Note: The AUNITS command changes to "DBM" if "IP;" is executed.

AUTO

Auto Couple

Description: The AUTO command couples the attenuation (AT), step size (SS), sweep time (ST), resolution (RB) and video (VB) bandwidths to yield optimum amplitude accuracy. In addition, AUTO deactivates the display line (DL), threshold (TH), video averaging (VAVG), and markers. Individual functions can couple by entering the keyword for the command before AUTO (for example, "AT AUTO;"). To recouple all functions, press HOLD, followed by *AUTO COUPLE*.

Syntax:



Example: OUTPUT 718;"AT AUTO;"
 OUTPUT 718;"HD;AUTO;"
 OUTPUT 718;"AUTO;"

The first line couples the attenuation. The second line couples all functions. The third line couples and deactivates a function if one was active, or couples all functions if none were active. It has no effect if the active function does not couple.

Note: Alternate commands allow you to couple individual functions: CR couples resolution bandwidth, CS couples step size, CT couples sweep time, CV couples video bandwidth, CA and couples input attenuation. The AUTO command is also available for the AT, SS, RB, and VB commands. The functions of the AUTO command and the *AUTO COUPLE* key are identical.

AXB

Exchange Trace A and Trace B

Description: The AXB command exchanges trace A and trace B, point by point.

Syntax: See Figure 4-1

Example: OUTPUT 718;"AXB;"

Note: The functions of the AXB and EX commands and the [A EXCH B] softkey are identical.

BLANK

Blank Trace

Description: The BLANK command disconnects a trace from the input data source. The trace is not displayed.

Syntax: See Figure 4-5

Example: OUTPUT 718;"BLANK TRA;"

Note: The functions of the BLANK command and the [STORE BLANK A] and [STORE BLANK B] softkeys are identical.

BL

Trace B Minus Display Line

Description: The BL command subtracts the display line from trace B and sends the difference to trace B.

Syntax: See Figure 4-1

Example: OUTPUT 718;"BL;"

Note: The functions of the BL and BML commands and the [B — DL → B] softkey are identical.

CAL

Calibration

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

ON turns correction factors on and corresponds to [CORRECT ON off].

OFF turns correction factors off and corresponds to [CORRECT on OFF].

STORE saves the calibration data in the non-volatile memory of the analyzer and corresponds to [CAL STORE]. It is a service function that should not be used without correct calibration data.

FETCH recalls the calibration data that was stored in the non-volatile memory of the analyzer and corresponds to [CAL FETCH].

FREQ initiates the frequency calibration routine and corresponds to [CAL FREQ].

AMP initiates the amplitude calibration routine and corresponds to [CAL AMPTD].

DLY initiates the tuning delay calibration routine and corresponds to [CAL YTO DELAY].

ALL initiates frequency, amplitude, and tuning delay calibration routines.

INIT resets stored calibration data from the non-volatile memory of the analyzer and corresponds to [EEPROM INIT].

Note: When INIT is selected, the CAL ALL routine must be run again.

DISP displays calibration data on the analyzer screen and corresponds to [DISPLAY CAL DATA].

DUMP returns calibration data to the controller.

Syntax:



Example: OUTPUT 718;"CAL ON;"

Note: See CAL under "Analyzer Functions" in the Operating Manual for more information including the correct instrument set-up for each routine. CAL DISP and CAL DUMP do not return all of the calibration data since screen size is limited and CAL DUMP only returns the data that would be displayed on the analyzer screen.

CF

Center Frequency

Description: The CF command specifies the value of the center frequency. When queried with OA or ?, CF returns the center frequency value as a real number followed by a carriage-return/line-feed.

Since span is affected by frequency, change the frequency before changing span. (See HNLOCK.)

Syntax: See Figure 4-7. The AUTO parameter does not apply.

Example: OUTPUT 718;"CF 200MZ;"

Note: The function of the CF command and the *FREQUENCY* key are identical, except CF does not call up the Frequency menu.

CLRW

Clear Write

Description: The CLRW command starts a sequence of events. First, each element in the indicated trace is set to the bottom of screen. Then, new data from the detector is put in the trace with each sweep. Traces in CLRW mode are displayed on the analyzer screen.

Syntax: See Figure 4-5

Example: OUTPUT 718;"CLRW TRA;"

Note: The functions of the CLRW command and the [CLEAR WRITE A] and [CLEAR WRITE B] softkeys are identical. Alternate command A1 is identical to [CLEAR WRITE A] and B1 is identical to [CLEAR WRITE B]

CLS

Clear Status Byte

Description: The Clear Status Byte CLS command clears all status bits. See SRQ for more information on status bytes, status registers, and service requests.

Syntax: See Figure 4-1

CNF

Confidence Test

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

Syntax: See Figure 4-1

Example: OUTPUT 718; "CNF;"

Note: The functions of the CNF command and the [CONF TEST] softkey are identical.

CONTS

Continuous Sweep

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 interface bus.

Syntax: See Figure 4-1

Example: OUTPUT 718; "CONTS;"

Note: The functions of the CONTS and S1 commands and the [CONT SWEEP] softkey are identical.

Note: If the specified center frequency is in band 0 then 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.

DET

Detection Mode

Description: The DET command selects the type of analyzer detection (positive peak or sample) and accesses service-diagnostic detection functions.

POS, the default detector mode, enables positive peak detection, which displays the maximum video signal excursions

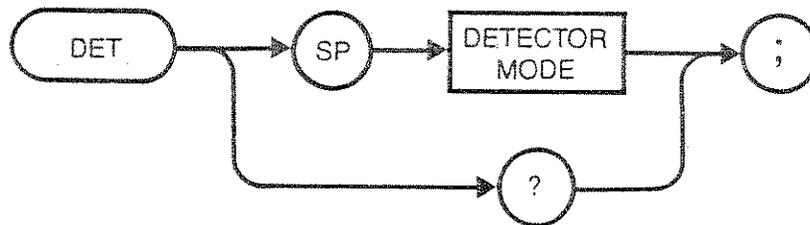
SMP enables sample detection, which uses the instantaneous signal value detected at the analyzer-to-digital converter output. Video averaging and noise-level markers, when activated, automatically activate sample detection.

Service Diagnostic Modes: The detector modes listed below are designed for troubleshooting and diagnostic tests of the HP 8592A. They are described in the HP 8592A Support Manual (HP Part Number 08592-90008), which is not supplied with the instrument. Contact your HP sales representative for more information.

AUXA, CDAC, DROOP, FDAC, FMD, FMSP, GND, MCD, MNSP, MTEN, MXRB, PTEN, REF, SDAC, SWPR, YFD, XFIN, YFL, YFN, YFSP, TEST.

When queried, DET returns the detection mode.

Syntax:



Example: OUTPUT 718;"DET POS;"
OUTPUT 718;"DET SMP;"

DL

Display Line

Description: The DL command defines the level of the display line in dBm and displays it on the analyzer screen. Activating video trigger mode activates the display line. The AUTO command and DL OFF turn off the display line.

Syntax: See Figure 4-6

Example: OUTPUT 718;"DL OFF;"
OUTPUT 718;"DL 5DB;"

Note: Alternate command L0 turns off the display line. The functions of the DL command and the [DISPLAY LINE] softkey are identical.

DN

Down Command

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

Syntax: See Figure 4-1

Example: OUTPUT 718;"SS 3GZ;CF 7GZ;DN;"
OUTPUT 718;"SP 40MZ;MKPK;DN;"

The first line sets center frequency to 4 GZ. The second line sets the span to 20 MHz since peak search is not an active function. The step size for span is one graticule.

DONE

Done

Description: The DONE command is a synchronizing function. When DONE follows a command list, it sends the controller a 1 after the command list is executed. (The command list is defined as any analyzer command from this chapter.)

Syntax: See Figure 4-1

Example:

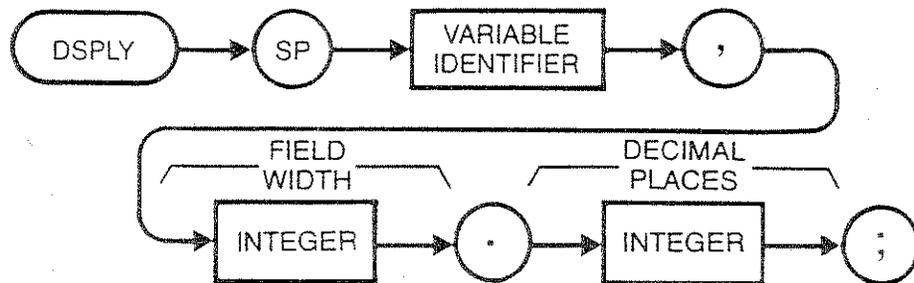
```
05 REAL DONE_VAL
10 ! EXAMPLE TO SHOW WHEN ATTENUATOR HAS COMPLETED CHANGING
20 OUTPUT 718;"AT 30 DB;DONE;"
30 ENTER 718;DONE_VAL
40 ! WHEN ANANLYZER RECEIVES THE DONE VALUE. IT IS DONE EXECUTING THE COMMAND
50 END
```

DSPLY

Display

Description: The DSPLY command displays the value of a variable anywhere on the analyzer display. Use the PU and PA commands to position the variable on the 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 places. For example, the number 123.45 has a field of 7, and 2 decimal places.

Example: OUTPUT 718;"PU;PA 50,50;DSPLY CF,15.0;"

The DSPLY command in the preceding example allows the programmer to check the value of the center frequency. The number, 15.0, is the total field width, 15, and the desired number of decimal places, 0.

The analyzer displays the data independent of PU or PD. The pen does not need to be down for data display.

EE

Enable Entry

Description: The EE command sends values entered by the operator on the analyzer number keyboard to the controller. Generally, the sequence of programmed events is as follows:

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.

Syntax: See Figure 4-1

Example: The following program allows you to enter values on the analyzer. As values are entered, they appear on the computer screen. When 1 MHz is entered, "FINISHED" appears on the computer screen and the program is terminated.

```
10 ASSIGN @Sa TO 718
20 ON INTR 7 GOSUB Process_key
30 OUTPUT @Sa;"RQS 2;EE;"
40 ENABLE INTR 7;2
50 REPEAT
60  DISP "ENTER VALUE AND UNITS (ENTER 1MHZ TO EXIT)"
70  UNTIL New_value=1.E+6
80  OUTPUT @Sa;"HD;"!turn off active function
90  DISP "FINISHED"
100 STOP
110 !
120 Process_key: !key process routine
130 A=SPOLL(@Sa)
140 IF BIT(A,1)=1 THEN
150  OUTPUT @Sa;"?;"
160  ENTER @Sa;New_value
170  PRINT "VALUE IS ";New_value
180  WAIT .1 !wait for value entry
190  OUTPUT @Sa;"EE;RQS 2;"
200  ENABLE INTR 7;2
210 ELSE !handle other interrupts here
220 END IF
230 RETURN
240 END
```

EK

Enable Knob

Description: The EK command allows data entry with the front-panel knob when the analyzer is under remote control. The knob is functional, but other front panel functions remain inoperative.

Syntax: See Figure 4-1

Example: While the program is paused, the operator positions a marker on a signal that needs further analysis.

```
10 OUTPUT 718;"MKN;EK;"
20 PRINT "USE KNOB TO PLACE MARKER"
30 PRINT "PRESS CONTINUE WHEN DONE"
40 PAUSE
50 ! Analysis program inserted here
60 END
```

Press the *CONTINUE* key on the controller to resume program operation.

EM

Erase Graphics Memory

Description: The EM command clears out any user generated graphics and text. It can be used to clear the analyzer screen in preparation for the display of custom graphics. (EM does not remove annotation, graticule, display line, or threshold. If desired, these can be removed and used with EM as below.)

Execute the following line to blank the analyzer screen:

```
OUTPUT 718;"EM;BLANK TRA;BLANK TRB;GRAT OFF;ANNOT OFF;TH OFF;DL OFF;"
```

To reinstate the normal analyzer display, execute the following program line:

```
OUTPUT 718;"EM;CLRW TRA;GRAT ON;ANNOT ON;"
```

Syntax: See Figure 4-1

Example: OUTPUT 718;"EM;"

EP

Enter Parameter Function

Description: The EP command sends values entered by the operator on the analyzer number keyboard to the parser.

Syntax: See Figure 4-1.

Example: OUTPUT 718;"DL EP;"

This example allows you to enter a value manually through the numeric keypad. When the value is terminated by the *dBm* or *ENTER* keys 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.

FA

Start Frequency

Description: The FA command specifies the start frequency value, performing the same function as the [START FREQ] softkey. When queried (? or OA), FA returns the start frequency, a real number, followed by a carriage-return/line-feed. The End-or-identify line (EOI) is asserted.

See HNLOCK for a description of the limits.

Syntax: See Figure 4-7. The AUTO parameter does not apply.

Example: OUTPUT 718;"FA 88MZ;"

FB

Stop Frequency

Description: The FB command specifies the stop frequency value, performing the same function as the [STOP FREQ] softkey. When queried (? or OA), FB returns the stop frequency, a real number, followed by a carriage-return/line-feed. The End-or-identify state (EOI) is asserted.

See HNLOCK for a description of the limits.

Syntax: See Figure 4-7. The AUTO parameter does not apply.

Example: OUTPUT 718;"FB 88MZ;"

FFT

Fast Fourier Transform

Description: 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. The maximum length of any of the traces is 401 points.

If the analyzer is in logarithmic mode, the data in the source trace is assumed to be logarithmic and will be converted to linear before transformed with FFT. Next, the source array is weighted with the function in the window trace (described below) to minimize amplitude inaccuracies, side lobes, etc. Finally, the transform is computed and the results are placed in the destination array.

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. 20 msec divided by 400 equals 50 μ sec, the sampling period. The sample rate is 1/50 μ sec. Fmax equals 1/50 μ sec 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.

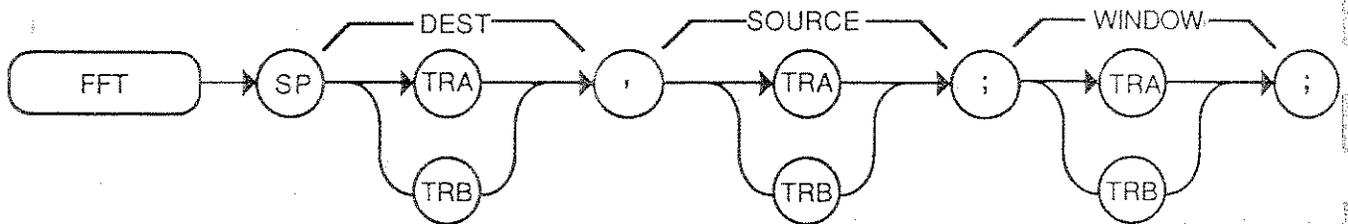
The windowing function stored in the window trace may be selected with the TRACE WINDOW (TWINDOW) command or by the user storing his own values in that trace.

Note: After using the FFT command, the markers remain in FFT mode. For this reason, the markers must be turned off before continuing (see MKOFF).

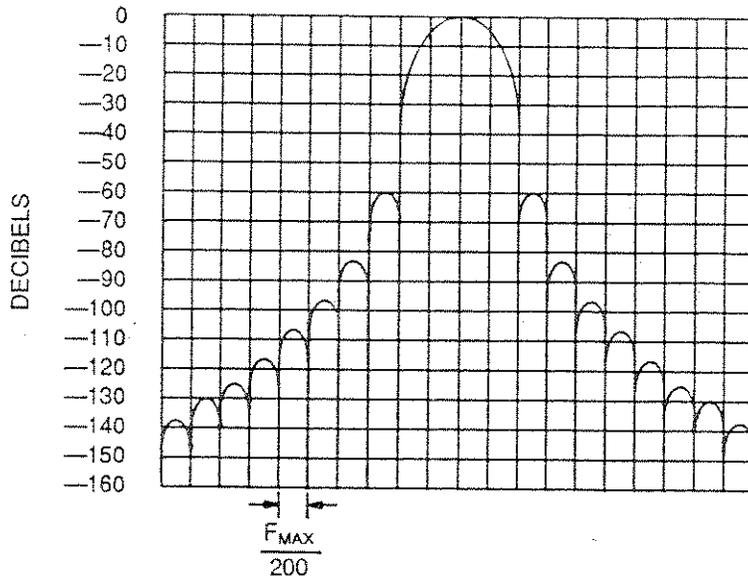
Note: The [FFT MEAS] softkey initiates a routine that first selects sample-detection mode, takes a sweep, and stores and places trace A in view mode (DET SMP;TS;VIEW TRA). Then the routine sets the trace window to flattop (TWINDOW TRB,FLATTOP) and FFT is run (FFT TRA,TRA,TRB). Finally, markers are turned off, a marker is placed at the highest peak, the FFT marker-readout mode is set, and a normal marker is activated (MKOFF ALL; MKPK; MKREAD FFT;MKN).

Note: TWINDOW should always be set before FFT is called.

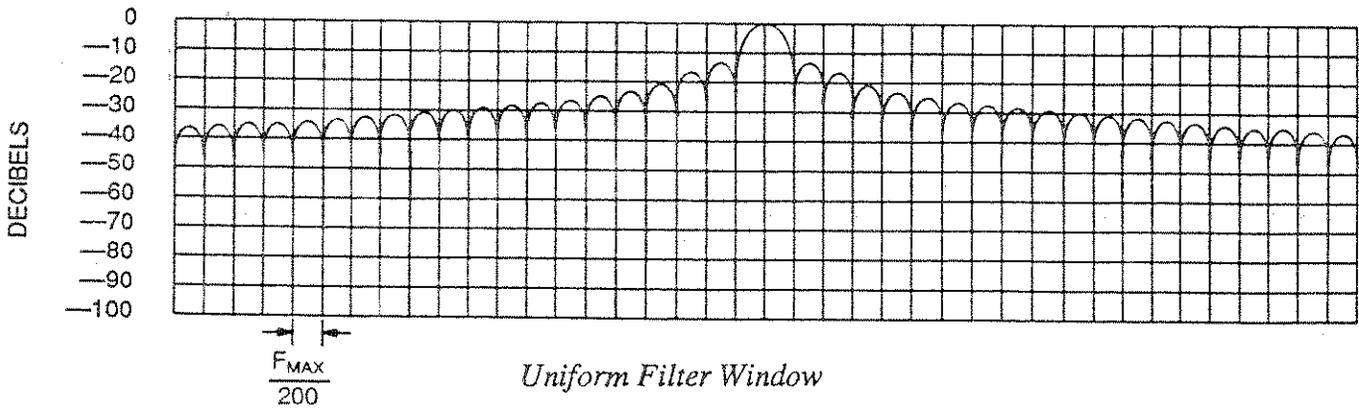
Syntax:



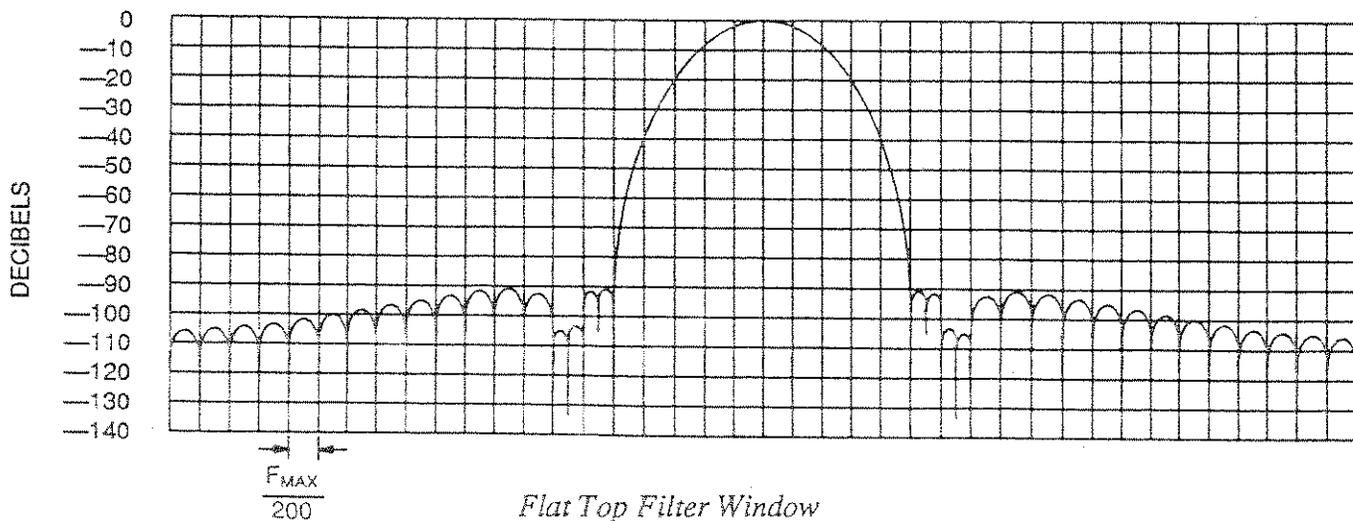
Filter Windows:



Hanning Filter Window



Uniform Filter Window



Flat Top Filter Window

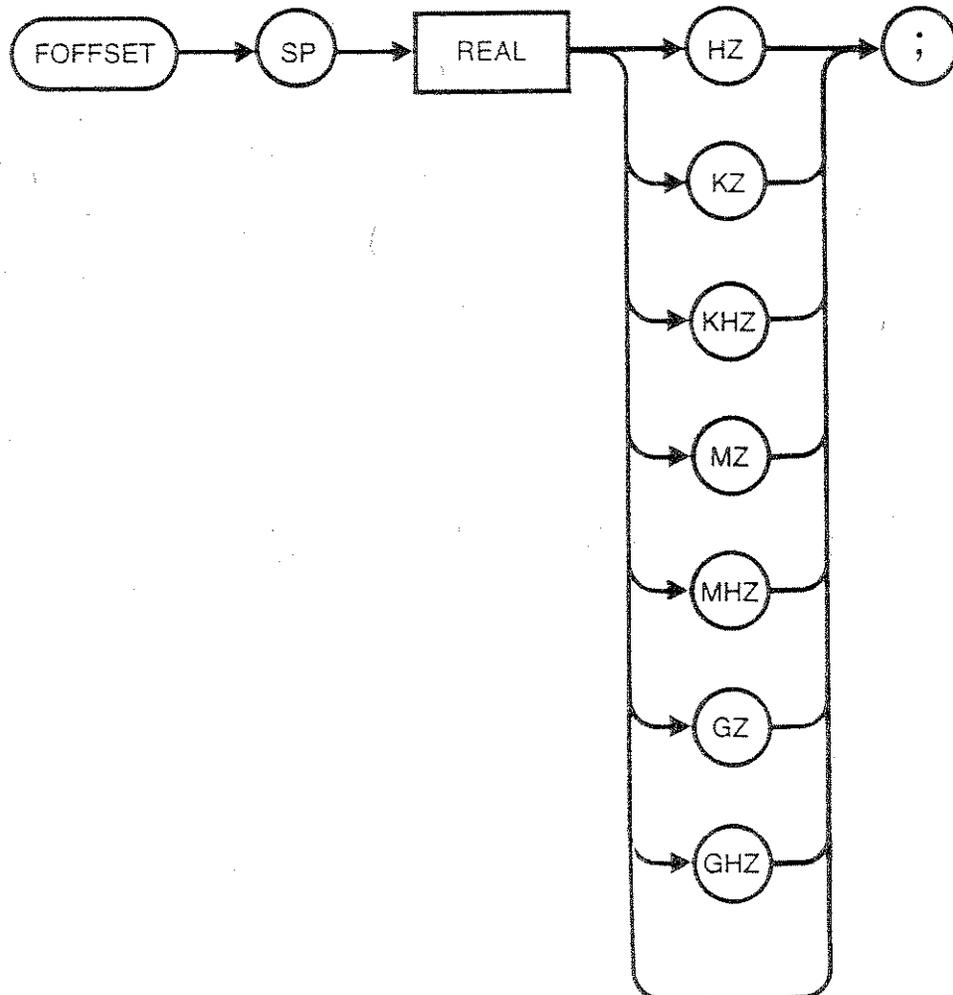
FOFFSET

Frequency Offset

Description: The FOFFSET command selects a value that offsets the frequency scale for all absolute frequency readouts (for example, center frequency). Relative values like span and delta marker 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.

Syntax:



Example: OUTPUT 718;"FOFFSET 100MZ;"

This command replaces the previous value of the offset, if any, with 100 MHz. All displayed frequency values are modified by this amount.

Note: The functions of the FOFFSET command and the [FREQ OFFSET] softkey are identical.

GRAT

Graticule

Description: The GRAT command turns the graticule on and off. When queried (?), GRAT returns the graticule state: ON or OFF.

Syntax: See Figure 4-4

Example: OUTPUT 718;"GRAT OFF;"

Note: Alternate command KSm turns off the graticule and KSn turns it on. The functions of the GRAT command and the [GRAT ON OFF] softkey are identical.

HD

Hold Data Entry

Description: The HD command disables data entry via the front panel number keyboard, blanks the active function readout, and deactivates any active function.

Syntax: See Figure 4-1

Example: OUTPUT 718;"HD;"
OUTPUT 718;"CF 7 GZ;12 GZ;HD;14 GZ;"

The first line disables the active function. The second line causes an error at "14 GZ" and leaves CF at "12 GZ;" since HD deactivates any current function.

Note: The functions of the HD command and the *HOLD* key are identical.

HN

Harmonic Number

Description: The HN command returns an integer.

Single-band sweep: harmonic number 0 to 4
Multi-band sweep: —1

Syntax: See Figure 4-3

Example: OUTPUT 718;"HN?;"

HNLOCK

Harmonic Number Lock

Description: The HNLOCK command forces the analyzer to use only the selected harmonic.

HNLOCK ON locks onto the harmonic necessary for the current center frequency, lowering the span, if necessary, due to the limits of the harmonic.

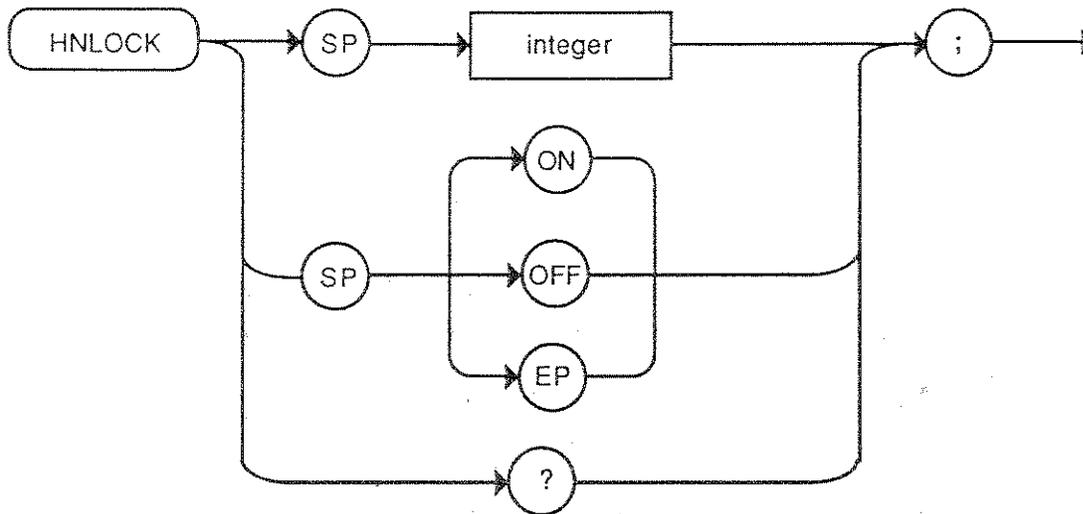
HNLOCK X locks onto harmonic band X and automatically selects the settings shown on Table 4-6.

Table 4-6

REMOTE COMMAND	EQUIVALENT SOFTKEY	FREQUENCY SETTINGS	
HNLOCK 0	BAND 0:	center frequency 1.45 GHz	span 2.9 GHz
HNLOCK 1	BAND 1:	center frequency 4.475 GHz	span 3.45 GHz
HNLOCK 2	BAND 2:	center frequency 9.4 GHz	span 6.8 GHz
HNLOCK 3	BAND 3:	center frequency 15.9 GHz	span 7 GHz
HNLOCK 4	BAND 4:	center frequency 20.55 GHz	span 2.9 GHz

Once HNLOCK is set, only frequencies and spans that fall within the frequency band of the current harmonic may be entered. The span automatically reduces 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.

Syntax:

Note: The HNLOCK command and the softkeys under the BAND LOCK menu are identical.

Note: HNLOCK? returns ON or OFF.

Band Lock 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 will be used for both the start and stop frequency so 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 above) stop frequency will be used for both the start and stop frequency so the span will be set to zero. (Also see FB.)

Center Frequency:

The span will be modified if necessary to get the center frequency requested without crossing the band edges. (Also see CF.)

Span:

The span will be limited as necessary to keep the start and stop within the band edges without changing the center frequency. The maximum span allowed is 3.6 GHz in bands 0 and 1, and 7 GHz in higher bands. (Also see SP.)

Band Lock Off

The STOP and START FREQUENCIES will be 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.)

HNUNLK

Unlock Harmonic Number

Description: The HNUNLK command unlocks the harmonic number, allowing you to select frequencies and spans outside its range.

Syntax: See Figure 4-1

Example: The following program unlocks the harmonic number.

```
10 OUTPUT 718;"HNUNLK;"  
20 END
```

Note: HNUNLK is equivalent to HNLOCK OFF.

ID

Identify

Description: The ID command returns the instrument identity to the controller (for example, "HP 8592A").

Syntax: See Figure 4-1

Example: The following program prints the instrument identifier on the computer screen.

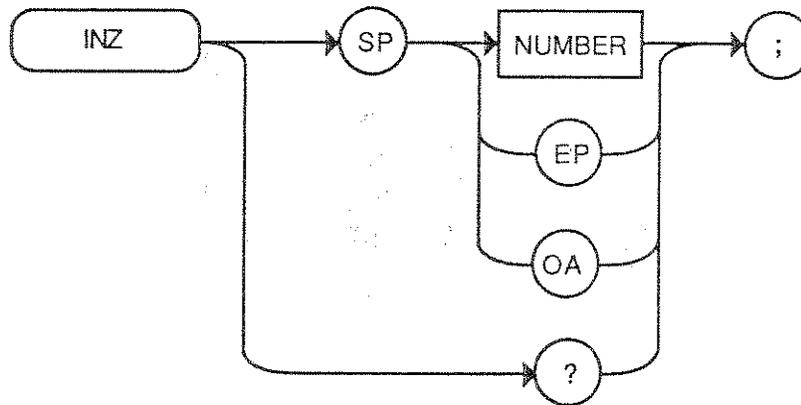
```
10 OUTPUT 718;"ID;"  
20 ENTER 718;A $  
30 PRINTER IS 1  
40 PRINT A $  
50 END
```

INZ

Input Impedance

Description: The actual impedance can only be affected by internal hardware; the hardware only supports 50 ohms. The values allowed are 50 or 75 ohms. The INZ command is used for computation purposes during power/voltage conversions. The INZ command effects the amplitude results reported in any units other than relative power (dBm units). (See AUNITS command.)

Syntax:



Example: OUTPUT 718;"INZ 50;"

Note: The functions of the INZ command and the [INPUT IMPED] softkey are identical.

IP

Instrument Preset

Description: The instrument preset command, IP, executes the following commands:

AMB OFF: Turns off A — B mode.
ANNOT ON: Turns on annotation.
AT: —10 dB
AUNITS DBM: Selects dBm amplitude units.
AUTO: Couples RB, AT, SS, ST, and VB. Turns off display line and threshold.
BLANK B: Blanks trace B.
CLRW A: Clears and writes trace A.
CONTS: Selects continuous sweep.
DET POS: Selects positive peak detection.
DL OFF: Display line off.
EM: Erase user graphics
FA: Sets the start frequency. 3000 MHz
FB: Sets the stop frequency. 22000 MHz
FOFFSET: 0
HD: Hold (Deactivates active function.)
HNLOCK OFF: Harmonic lock off, allowing multiband sweeping.
GRAT ON: Turns on the graticule.
INZ: Loaded from configurable location
LG: Selects 10 dB/DIV log scale.
MDS W: Selects data size of one word, which is two 8-bit bytes.
MKNOISE OFF: Turns off noise markers.
MKOFF: Turns off all markers.
MKPAUSE OFF: Turns off marker pause mode.
MKPX: 6dB minimum excursion for peak identification.
MKTRACK OFF: Turns off marker tracking.
ML: —10 dBm
PD: Puts pen down at current position.
RL: 0 dBm
ROFFSET: 0
RQS 40: Decimal 40 = Illegal Command or Hardware Broken
STATUS BYTE: clear
SS: 100 MHz step size
TH: one division above bottom graticule line, line off.
TITLE: clear
TM FREE: Selects free run trigger mode.
TDF P: Selects parameter units output format.
VAVG OFF: Turns off video averaging.
VAVG LIMIT: 100

Instrument preset automatically occurs when you turn on the analyzer, and is a good starting point for many measurement processes. When IP is executed remotely, the analyzer does not necessarily execute a complete sweep. PRESET sets CF to 12.5 GHz and span to 19 GHz.

Syntax: See Figure 4-1

Example: OUTPUT 718;"IP;TS;"

Note: IP does not clear state or trace registers.

LG

Logarithmic Scale

Description: The LG command specifies the vertical graticule divisions as logarithmic units without changing the reference level. The vertical scale may be specified (in integers) between 1 and 20 dB per graticule division. When queried, LG returns the dB/division value. If "LG?" returns 0, then the analyzer is in linear mode.

Syntax: See Figure 4-6

Example: OUTPUT 718;"LG 1DB;"

Note: The functions of the LG command and the [LOG dB/DIV] softkey are identical.

LN

Linear Scale

Description: The LN command selects the LINEAR mode of operation and 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 mV become 30.16 mV, which equals -17.4 dBm.

Syntax: See Figure 4-1

Example: OUTPUT 718;"LN;"
OUTPUT 718;"LN;RL 30MV;"

Note: The functions of the LN command and the [LINEAR] softkey are identical.

MA

Marker Amplitude Output

Description: If the marker is on the screen, the MA command returns the amplitude level of the active marker to the controller. The marker amplitude is displayed in the upper-right corner of the analyzer screen. If both the delta marker and active marker are on the screen, MA returns the amplitude difference between the two markers. In zero span, MA returns the amplitude difference. To obtain the time difference, see MKD.

The data that is returned by "MA;" depends upon many command conditions including MKN, MKD, TDF, AUNITS, MDS and MKNOISE.

Syntax: See Figure 4-1

Example: MA is used in the following program to determine the amplitude value of the marker.

```
10 PRINTER IS 701
20 OUTPUT 718;"IP;SNGLS;"
30 OUTPUT 718;"CF 300MZ; TS;"
40 OUTPUT 718;"MKN;MKPK HI;"
50 OUTPUT 718;"MA;"
60 ENTER 718;A
70 PRINT A
80 END
```

Line 30 sets the start and stop frequencies. Line 40 activates a normal marker and peak search. Line 60 returns the amplitude to the controller. Lines 60 and 70 assign the amplitude to variable A and print the amplitude.

MDS

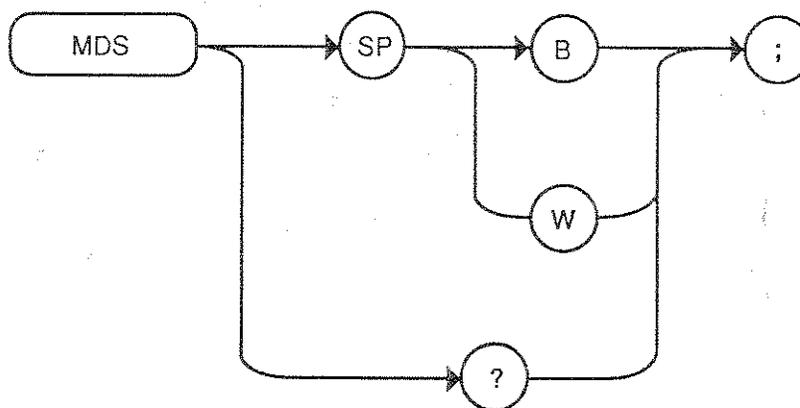
Measurement Data Size

Description: The MDS command formats binary measurements:

B selects a data size of one 8-bit byte.

W selects a data size of one word, which is two 8-bit bytes.

Syntax:



Example: OUTPUT 718;"TDF B;MDS W;TRA?"

The above example transfers trace A in binary, 2 bytes per word.

```
10 INTEGER TRACE_A (1:401)
20 ! READING A TRACE BACK IN "W" FORMAT
30 OUTPUT 718;"SNGLS;TS;"
40 OUTPUT 718;"TDF B;MDS W;TRA?;"
50 ENTER USING "#,401(W);TRACE_A(*)"
```

The above example demonstrates how data is read back to the analyzer.

MF

Marker Frequency Output

Description: If the marker is on the screen, the MF command returns the frequency value of the active marker to the controller. In delta marker mode, non-zero span, "MF;" returns the frequency difference between the two markers. In zero span, "MF;" returns the marker time, or the delta marker time.

Syntax: See Figure 4-1

Example: MF is used in the following program to determine the frequency value of the marker.

```
10 PRINTER IS 701
20 OUTPUT 718;"IP;SNGLS;"
30 OUTPUT 718;"FA 280MZ;FB 320MZ;TS;"
40 OUTPUT 718;"MKN;MKPK HI;"
50 OUTPUT 718;"MF;"
60 ENTER 718;F
70 PRINT F
80 END
```

Line 30 sets the start and stop frequencies. Line 40 activates a normal marker and peak search. Line 60 returns the frequency to the controller. Line 70 prints the frequency.

MKA

Marker Amplitude

Description: The MKA command specifies the amplitude of the active marker in dBm 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.

Syntax: See Figure 4-6

Example:

```
10 OUTPUT 718;"MKA?;"
20 ENTER 718;A
```

"A" will contain the amplitude value of the marker.

MKCF

Marker to Center Frequency

Description: The MKCF command sets the center frequency equal to the marker frequency and moves the marker to the center of the screen.

Syntax: See Figure 4-1

Example: OUTPUT 718;"MKCF;"

Note: The functions of the MKCF and E2 commands and the [MARKER → CF] softkey are identical.

MKD

Marker Delta

Description: In non-zero span, the MKD function computes the frequency and amplitude difference between the active marker and the delta marker. In zero span, MKD computes the time and amplitude difference. To read the amplitude difference or amplitude ratio in LN display mode, use MA. These values are displayed on the analyzer screen.

If a delta marker is not on the screen, MKD places one at the specified frequency if using frequency markers, or at the current active marker. If an active marker is not on the screen, MKD positions an active frequency marker at center screen. 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).

In zero span, MKD?; computes and returns time differences. To attain amplitude, see MA.

Syntax: See Figure 4-7. AUTO parameter turns markers off.

Example: OUTPUT 718;"MKD;"

Note: The functions of the MKD and M3 commands and the [MARKER DELTA] softkey are identical.

MKF

Marker Frequency

Description: The MKF command specifies the frequency value of the active marker. When queried, MKF returns the active marker frequency as a real number. In zero span, "MKF?;" returns the time value.

Syntax: See Figure 4-7. UP moves the marker to the right and DN moves the marker to the left. AUTO does not apply.

MKMIN

Marker Minimum

Description: The MKMIN command moves the active marker to the minimum value detected.

Syntax: See Figure 4-1

Example: OUTPUT 718;"MKMIN;"

MKN

Marker Normal

Description: The MKN command activates and moves the marker to the specified frequency (or CF if none is supplied). In non-zero span, "MKN?;" returns the frequency value. In zero span, "MKN?;" returns the time value.

Syntax: See Figure 4-7. The AUTO parameter turns markers off.

Example: OUTPUT 718;"MKN 300MZ;"

Note: The functions of the MKN and M2 commands and the [MARKER NORMAL] softkey are identical.

MKNOISE

Marker Noise

Description: The MKNOISE command displays the average noise level at the marker and selects DET SMP. The value is normalized to a 1-Hz bandwidth. When queried, MKNOISE returns ON or OFF. 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. So the data values averaged are not situated exactly symmetrically 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 3db resolution bandwidth. The firmware assumes the noise bandwidth is 1.12 * 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 effects 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 value:

In log detector mode

$$(\text{Averaged value over 32 values}) - 10 * (\log[1.12 * \text{Resolution bandwidth}]) + 2.5\text{dB}$$

In Linear detector mode (dBm) units:

$$(\text{Averaged value over 32 values}) - 10 * (\log[1.12 * \text{Resolution bandwidth}]) + 1.05\text{dB}$$

In linear mode with the normal display of voltage units, the noise marker voltage value will be related to the present marker voltage by this relation.

$$(\text{V_noise_marker})^2 = (\text{V_average})^2 * 1.12 * \text{Resolution_bandwidth} * 0.78524$$

$$\text{V_noise_marker} = \text{V_average} / (*1.12 * \text{Resolution_bandwidth} * 0.78524)^{0.5}$$

$$\text{V_noise_marker} = \text{V_average} * 1.06633 / (\text{Resolution_bandwidth})^{0.5}$$

Syntax: See Figure 4-4

Example: OUTPUT 718;"MKNOISE ON;"

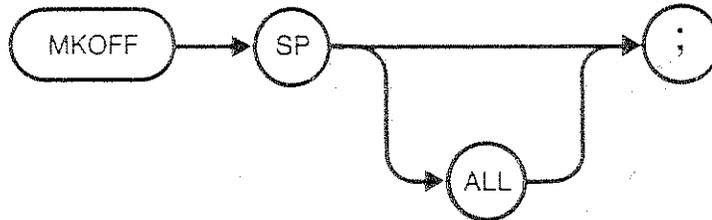
Note: The functions of the MKNOISE and KSM commands and the [MKNOISE on OFF] softkey are identical.

MKOFF

Marker Off

Description: The MKOFF command turns off either the active marker or all markers displayed on the screen. If the ALL parameter is omitted, only the active marker is turned off.

Syntax:



Example: OUTPUT 718;"MKOFF ALL;"

Note: The functions of the MKOFF ALL and M1 commands and the [MARKERS OFF] softkey are identical.

MKPAUSE

Marker Pause

Description: The MKPAUSE command pauses the sweep at the active marker for the duration of the delay period. The "real" parameter refers to the delay period from 2 msec to 100 seconds. To turn pause off, turn off markers (or, send "MKPAUSE 0;").

Syntax: See Syntax for ST. Marker Pause Auto is equivalent to marker pause 0, turning off the pause.

Example: OUTPUT 718;"MKPAUSE 100;"

Note: The functions of the MKPAUSE command and the [MARKER PAUSE] softkey are identical.

MKPK

Marker Peak

Description: The MKPK command positions the active marker on signal peaks. 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.

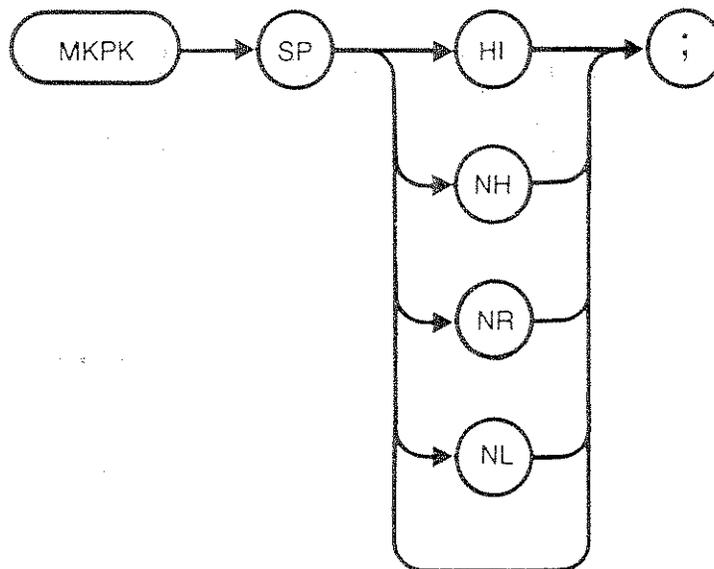
HI (highest) moves the active marker to the highest peak.

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.

Syntax:



Note: Alternate command E1 positions the marker at the highest signal peak. The functions of the MKPK HI command and the *PEAK SEARCH* key are identical. The MKPK parameters correspond to the [NEXT PEAK], [NEXT PK RIGHT], and [NEXT PK LEFT] softkeys.

Note: This function is for use with the frequency markers only.

MKPX

Marker Peak Excursion

Description: The MKPX command specifies the minimum signal excursion for the analyzer's internal peak-identification routine. When queried with ?, MKPX returns the excursion value.

Syntax: See Figure 4-6. AUTO does not apply.

Example: OUTPUT 718;"MKPX 8DB;"

Note: The functions of the MKPX command and the [PEAK EXCURSN] softkey are identical. (See MKPK and PEAKS.)

MKRL

Marker to Reference Level

Description: The MKRL command sets the reference level to the amplitude value of the active marker.

Syntax: See Figure 4-1

Example: OUTPUT 718;"MKRL;"

Note: The functions of the MKRL and E4 commands and the [MARKER → RL] softkey are identical.

MKSP

Marker to Span

Description: The MKSP command sets the start and stop frequencies to the frequencies of the delta markers. (See MKD.) The left marker specifies start frequency, and the right marker specifies stop frequency. If MKD is off, there is no operation.

Syntax: See Figure 4-1

Example: OUTPUT 718;"MKSP;"

Note: The functions of the MKSP and KSO commands and the [MRK Δ → SPAN] softkey are identical.

MKSS

Marker Step Size

Description: The MKSS command sets the center frequency step size to the frequency of the active marker. If in delta mode, the step size is set to the delta frequency (absolute value).

Syntax: See Figure 4-1

Example: OUTPUT 718;"MKSS;"

Note: The functions of the MKSS and E3 commands and the [MARKER → STEP] softkey are identical.

MKTRACK

Marker Track

Description: The MKTRACK command moves the signal with an active marker to the center of the screen and fixes the signal there. To keep a drifting signal at center screen, place the active marker on the desired signal before executing MKTRACK.

Syntax: See Figure 4-4

Example: OUTPUT 718;"MKTRACK ON;"

Note: The functions of the MKTRACK command and the *SIGNAL TRACK* key are identical. Alternate command MT0 turns marker track off, and MT1 turns it on.

MKTYPE

Marker Type

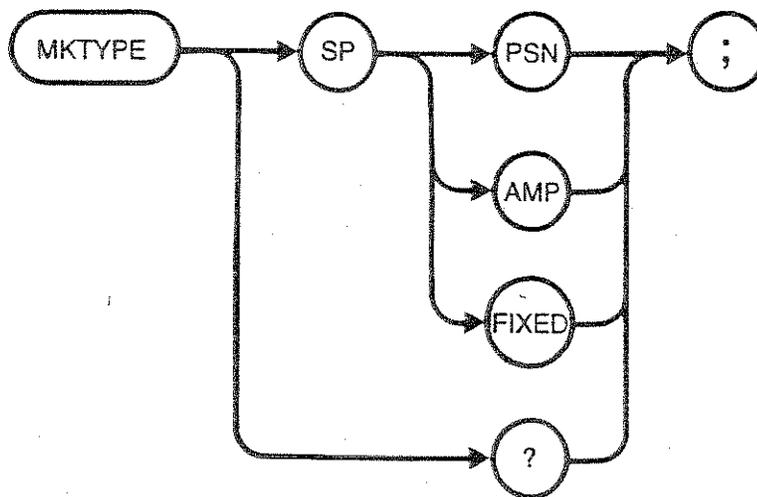
Description: The MKTYPE command changes the type of the current active marker.

AMP allows markers to be positioned according to amplitude, as shown in the example.

PSN allows markers to be positioned according to the horizontal position on the display.

FIXED allows a marker to be placed at any fixed point on the analyzer screen.

Syntax:



Example: OUTPUT 718;"MKTYPE AMP;MKA -3;"

The example positions a marker at the frequency value where the amplitude is closest to -3dBm.

Note: The functions of the MKTYPE AMP and the [MARKER AMPTD] softkey are identical.

ML

Mixer 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. IP selects -10 dBm.

Syntax: See Figure 4-6. AUTO does not apply.

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.

Note: The functions of the ML and KSO commands and the [MAX MXR LEVEL] softkey are identical. (Also see AT.)

MXMH

Maximum Hold

Description: While the trace is active and displayed, the MXMH command updates each trace element with the maximum level detected.

Syntax: See Figure 4-5

Example: `OUTPUT 718;"MXMH TRA;"`

Note: The functions of the MXMH TRA and A2 commands and the [MAX HOLD A] softkey are identical. The functions of the MXMH TRB and B2 commands and the [MAX HOLD B] softkey are identical.

M4

Marker Zoom

Description: The M4 command activates a single marker at center frequency. The knob changes the position of the marker and the step keys change the frequency span; the center frequency is set to the value of the marker frequency with each press of the step keys. Press the *LOCAL* key before using the knob or step keys. The "UP" and "DN" parameters increase and decrease the span remotely.

Syntax: See Figure 4-7. AUTO turns off the function.

Example: OUTPUT 718;"M4;"

OA

Output Active Function Value

Description: The OA command sends the value of the active function to the controller.

Syntax: See Figure 4-1

Example: OUTPUT 718;"ST 3;OA;"

The example produces the value 3.

OL

Output Learn String

Description: The OL command transmits information to the controller that describes the state of the analyzer when the OL command is executed. This information 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 114 bytes of storage space as is shown in Chapter 3, "Advanced Programming." (See "Saving States" in Chapter 3.)

Syntax: See Figure 4-1

Example:

```
10 DIM LEARN_STRING$(114)
20 OUTPUT 718;"OL;" ! ASK FOR FIRST STATE
30 ENTER 718 USING "#,114A"; LEARN_STRING 8
40 DISP"CHANGE ANALYZER TO ALL OTHER STATE, THEN CONTINUE TO RESTORE FIRST
STATE
50 PAUSE
60 OUTPUT 718; LEARN_STRING$
```

Note: 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; therefore, a restore learn string command does not exist. (Also see RC and SAVES.)

PA

Plot Absolute

Description: The PA command specifies in display units a vector location on the analyzer screen relative to reference coordinates 0,0. 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 lower-left coordinate corresponds to (Xmin, Ymin) and the upper-right coordinate corresponds to (Xmax, Ymax).

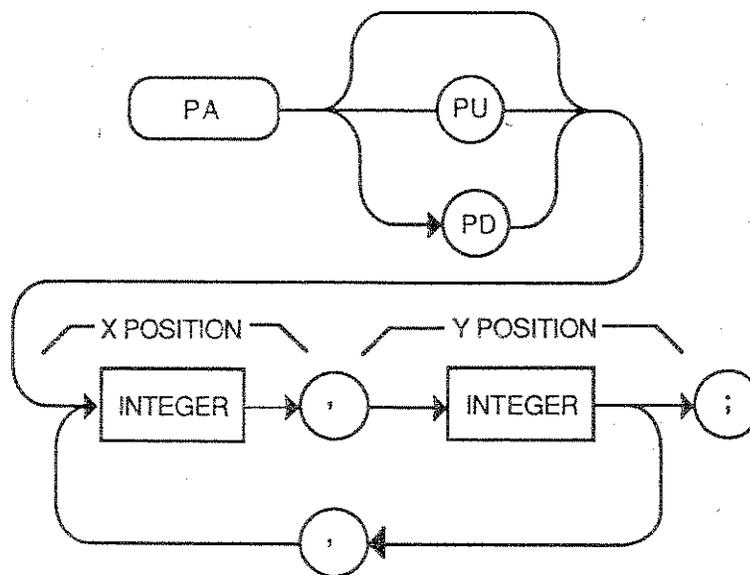
Screen Coordinates:

(Xmin, Ymin) = (-40, -22)
(Xmax, Ymax) = (471, 233)

Graticule Coordinates:

(Xmin, Ymin) = (0,0)
(Xmax, Ymax) = (400,200)

Syntax:



Example: This example draws a rectangle on the analyzer's screen.

```

10 OUTPUT 718;"IP;BLANK TRA;"
20 OUTPUT 718;"ANNOT OFF;GRAT OFF;"
30 OUTPUT 718;"PU;"
40 OUTPUT 718;"PA 100,100;PD 100,150,"
50 OUTPUT 718;"150,150,150,100,100,100;"
60 END
  
```

Line 20 clears the display. Line 30 specifies PU, which prevents the initial vector from being drawn. Line 40 specifies the starting point of the rectangle to be drawn by the program. The PD command causes a vector to be drawn on the screen according to the coordinates specified in the program. Line 50 gives the coordinates for the remaining points of the rectangle.

PD

Pen Down

Description: The PD command determines whether or not the lines drawn by the next set of PA/PR commands will be visible. The PA command, plot absolute, or PR, plot relative, may be used to mark the starting point of the vector. PD may then be used to drop the pen.

Syntax: See Figure 4-1

Example: See the example for the PA command.

PEAKS

Peaks

Description: The PEAKS command sorts signal peaks by frequency or amplitude, stores the results in the destination trace, and returns the number of peaks found.

When sorting by frequency, PEAKS first computes the horizontal position of all peaks. These positions are consecutively loaded into the destination trace, the lowest value occupying the first element. Thus, signal frequencies, from low to high, determine the amplitude of the destination trace from left to right.

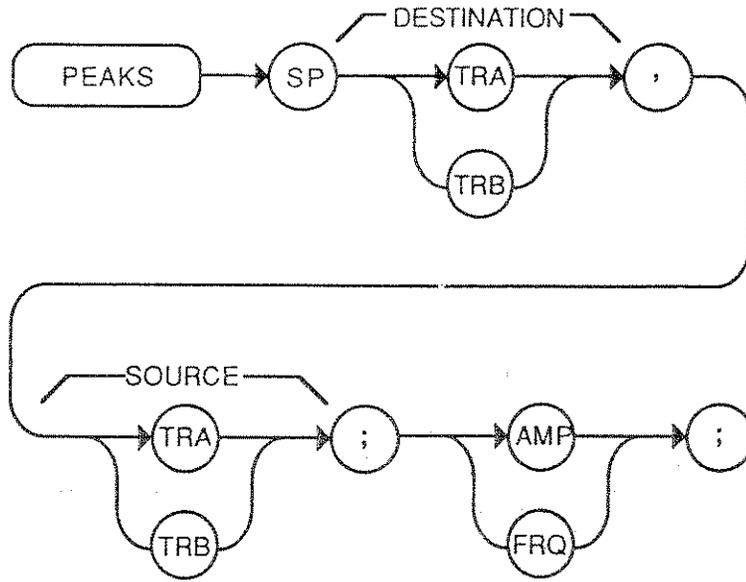
When sorting by amplitude, PEAKS first computes the amplitudes of all peaks in the source trace in measurement units, and sorts these values from high to low. The positions of the peaks are then loaded into the destination trace, with the position of the highest value occupying the first element.

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 rise and fall the MKPX dB value.

Syntax:



PLOT

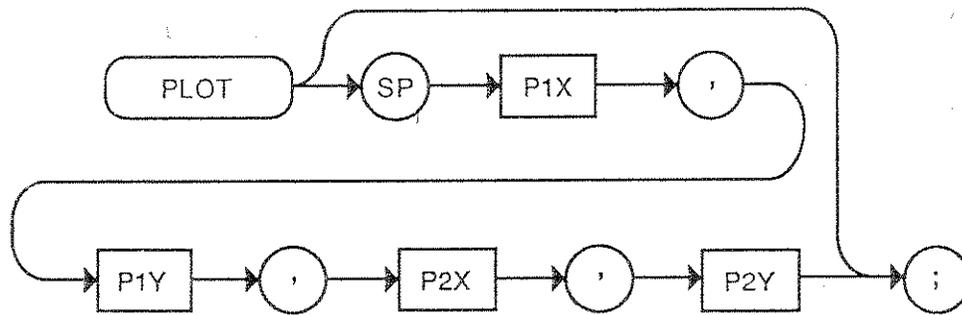
Plot

Description: The trace data, graticule, and annotation of the analyzer's screen can be directly transferred to a plotter via the analyzer's interface. PLOT immediately initiates a plotter dump of the screen data to the specified device (softkey labels excluded).

The example below 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) may 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.

Syntax:



Example: The following computer program is used to plot analyzer data. The plotter is at address 05 and the analyzer is at address 18. (The program is only valid for HP 200 and 300 series computers.)

```

10 !PLOTTER IS AT ADDRESS 705
20 !ANALYZER IS AT 718
30 DIM P$(80)
40 OUTPUT 705;"OP;"
50 ENTER 705;P$
60 OUTPUT 718;"PLOT";P$
70 SEND 7;LISTEN 5 TALK 18 DATA
80 END
  
```

Note: The functions of the PLOT command and the *PLOT* key on the front of the analyzer are identical. Specify the plotter by pressing *CONFIG* and [PLOTTER ADDRESS] and entering an address.

PP

Preselector Peak

Description: The PP command peaks the preselector. Set the desired trace to clear-write mode, place a marker on a desired signal, then execute PP. To read the new preselector peaking number, use the PRSDAC command. Commands following PP are not executed until after the analyzer has finished peaking the preselector.

Preselector peak 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 master. A "CAL: PEAKING" message appears on the active graticule area to indicate operation of the peaking routine. PRESELECTOR PEAK only operates in the 2.7 - 22 GHz preselector bands.

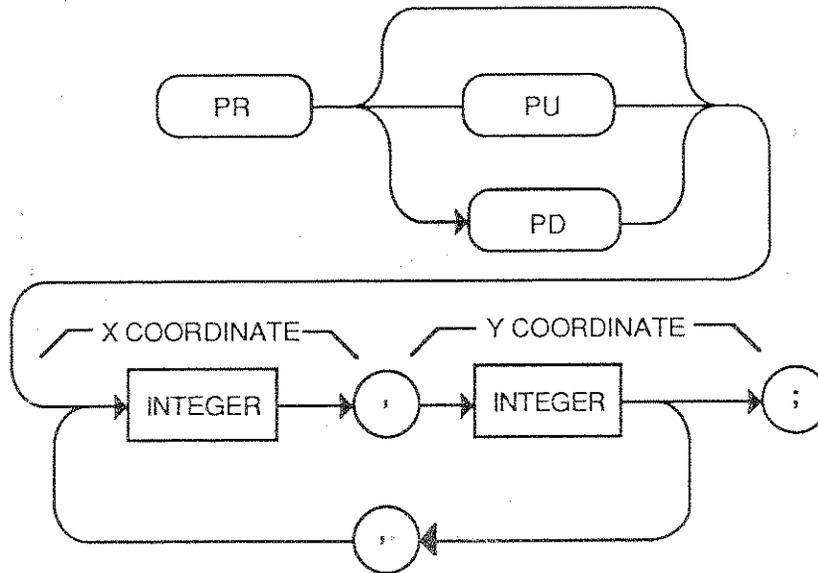
Example: OUTPUT 718;"PP;"

PR

Plot Relative

Description: The PR command specifies a new plot location on the analyzer screen relative to its current coordinates. 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.)

Syntax:



Example:

```

10 OUTPUT 718;"PU;PA 0,100;"
20 OUTPUT 718;"PD;PR100,0,0,-100,-100,0,0,100;"
  
```

PD (pen down) tells the analyzer to display the vectors drawn in accordance with the vector coordinates (x,y pairs) that follow the PR command. Vectors are drawn to the four corners of a rectangle.

PRINT

Print

Description: The PRINT command initiates a print dump of the screen data to the specified printer.

Syntax: See Figure 4-1

Example: The following program is used to print analyzer data. The printer address is 1 and the plotter address is 18. (The program is only valid for HP 200 and 300 series computers with an HP raster graphics printer, such as the HP Thinkjet.)

```
10 OUTPUT 718;"PRINT;"  
20 SEND 7;LISTEN 1 TALK 18 DATA
```

Note: The functions of the PRINT command and the *PRNT* key on the front of the analyzer are identical. Specify the printer address by pressing *CONFIG*, [PRINTER ADDRESS], and entering an address before sending the print command. (See PLOT.)

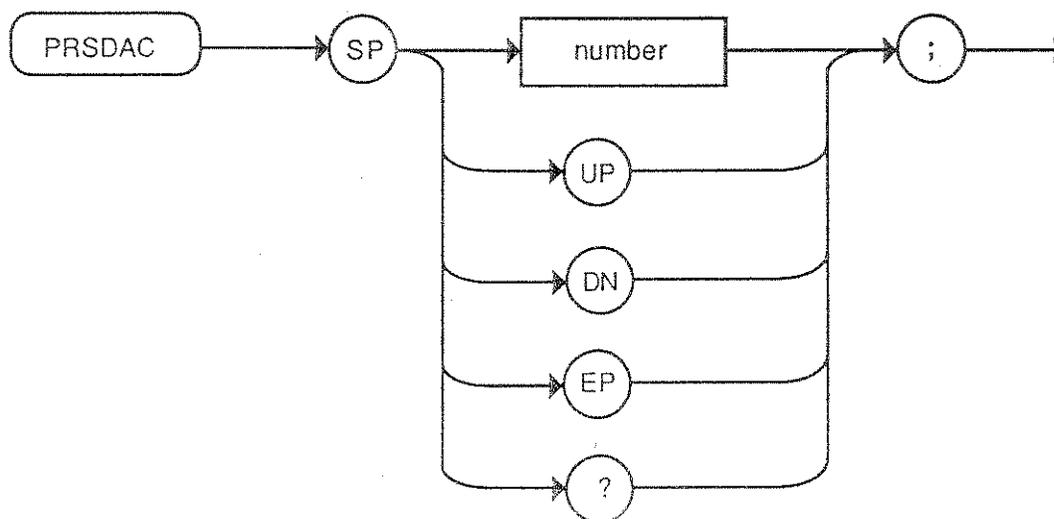
PRSDAC

Preselector DAC Number

Description: The PRSDAC command returns the preselector peak DAC number.

Parameters: number: integer from 0 to 3840.
UP/DN: increments of 1.

Syntax:



PSTATE

Protect State

Description: Enables and disables SAVES and SAVET.

Syntax: See Figure 4-4

Note: The PSTATE command is equivalent to the [SAVE LOCK] softkey.

PU

Pen Up

Description: The PU command instructs the analyzer not to plot vectors on the analyzer screen.

Syntax: See Figure 4-1

Example: See the syntax diagram for PA and PR

RB

Resolution Bandwidth

Description: The RB command specifies the resolution bandwidth. Available bandwidths are 1 kHz, 3 kHz, 10 kHz, 30 kHz, 100 kHz, 300 kHz, 1 MHz, and 3 MHz. The coupling between video bandwidth or sweep time and resolution bandwidth is not changed by this command. Execute RB AUTO to re-establish coupling. (Also see AUTO.)

Syntax: See Figure 4-7

Example: OUTPUT 718;"RB 1KZ;"

Note: The functions of the RB command and the [RES BW] softkey are identical.

RCLS

Recall State

Description: The RCLS command recalls analyzer states from one of the nine state registers.

Syntax: See Figure 4-2

Example: OUTPUT 718;"RCLS 3;"

The example recalls a previously saved state from register 3.

Note: The functions of RCLS and the *RECALL* key are identical. (Also see OL.)

RCLT

Recall Trace

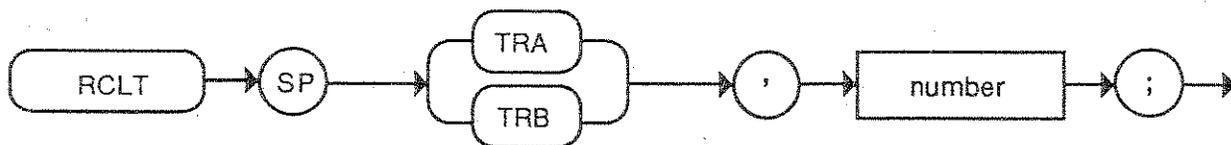
Description: The RCLT command recalls previously-saved trace data to the display along with the corresponding instrument state.

Note: The RCLT TRB command is equivalent to the [RECALL TRACE] softkeys.

Parameters:

TRA: recalls the trace data to Trace A.
 TRB: recalls the trace data to Trace B.
 number: integer from 0 to the TRCMEM—1.

Syntax:



Example: The following program saves trace A and recalls it to trace B.

```

10 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;"
20 OUTPUT 718;"SAVET TRA,1;"
30 OUTPUT 718;"IP;"
40 OUTPUT 718;"RCLT TRB,1;"
50 END
  
```

REV

Revision

Description: The REV command returns the firmware revision number and HP date code (for example, 860910 indicates September 10, 1986). The firmware revision number and HP date code also appear when the instrument is first turned on, but not in the same format.

Syntax: See Figure 4-1

Example: The following program prints the revision date on the computer screen.

```

10 OUTPUT 718;"REV;"
20 ENTER 718;A
30 PRINTER IS 1
40 PRINT A
50 END
  
```

RL

Reference Level

Description: The RL command specifies the amplitude value of the top graticule on the screen, which is called the reference level.

The reference level can be specified from -139.9 dBm to $+50$ dBm. The reference level and input attenuator couple 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.

Syntax: See Figure 4-6. AUTO does not apply.

Example: OUTPUT 718;"RL -10DM;"

Note: The functions of the RL command and the reference level function (accessed when *AMPLITUDE* is pressed) are identical.

Note: RL may affect the attenuation value.

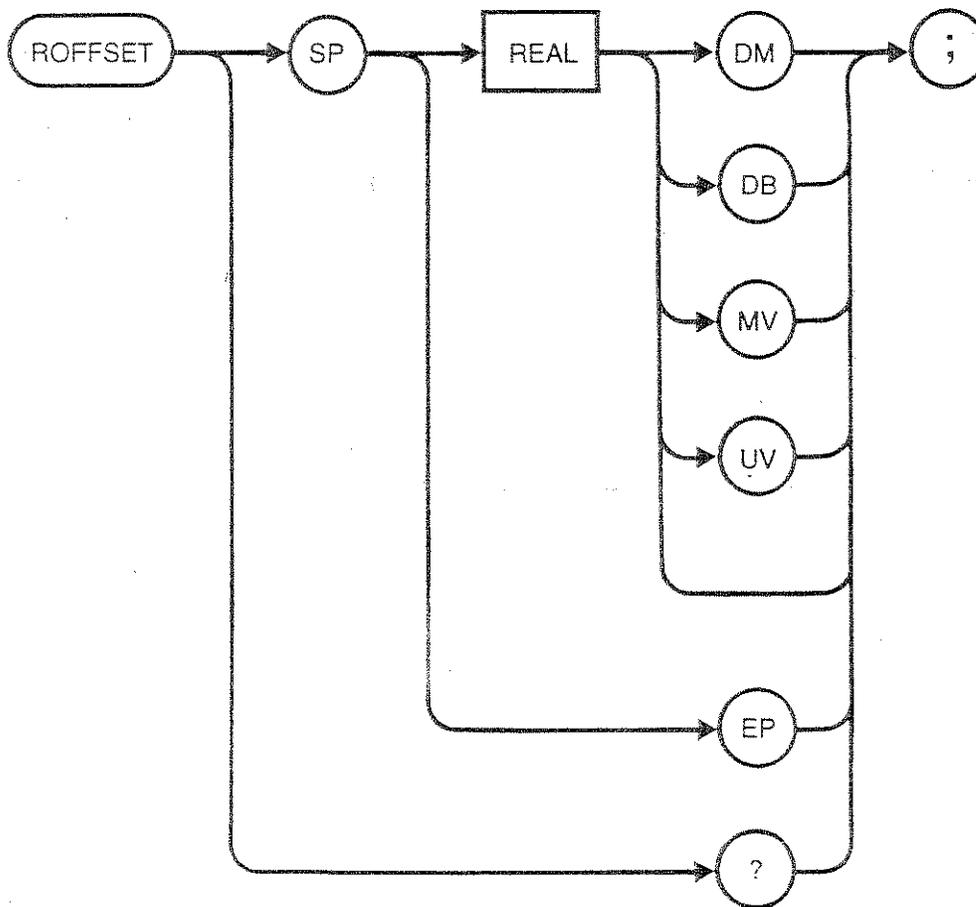
ROFFSET

Reference Level Offset

Description: The ROFFSET command offsets all amplitude readouts on the screen without affecting the trace. 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" eliminates an amplitude offset.

Syntax:



Example: OUTPUT 718;"ROFFSET -12DM;"

Note: The functions of the ROFFSET and KSZ commands and the [REF LVL OFFSET] softkey are identical.

RQS

Service Request Mask

Description: The RQS sets a bit mask for service requests. (See SRQ.) Each bit in the status byte is defined as follows:

- 1 indicates the units key was pressed and "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, only user for Enable Entry, EE mode.)
- 2 indicates end of sweep and "SRQ 104" appears on the analyzer screen. (If you send any RQS value that contains mask value 4, another sweep will be taken. See example.)
- 3 indicates broken hardware and "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 and "SRQ 140" appears on the screen.
- 0 (LSB) 6, and 7 are not used.

Note: The analyzer screen numbers 102, 104, and 110 are the octal values corresponding to the status register values, i.e., SRQ 102 = Bits 6 = octal 100 and bit 2 = octal 2 are both true.

Example: Assignment of values for the mask is as follows:

Illegal command	=	32
Command complete	=	16
Hardware broken	=	8
End of sweep	=	4
Units key pressed	=	2

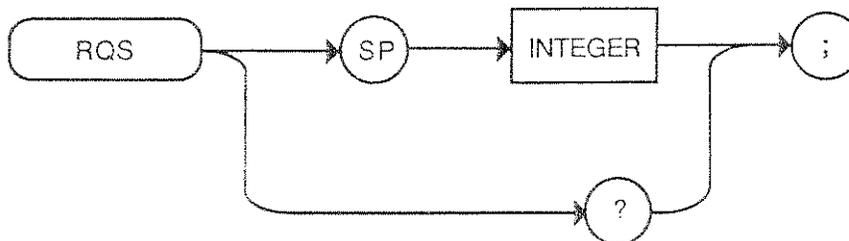
To send a mask with hardware broken and end of sweep, the mask is equal to 10 (8 + 2).

Send: OUTPUT 718;"RQS 10;"

To activate all conditions in the mask, the mask value is equal to 62 (32 + 16 + 8 + 4 + 2)

Send: OUTPUT 718;"RQS 62;"

Syntax:



SAVES

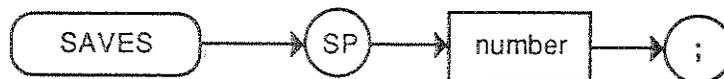
Save State

Description: The SAVES command saves the currently-displayed instrument state in the specified state register if it has not been locked by the PSTATE command.

Parameters:

number: integer from 0 to 9.

Syntax:



Example: The following program saves the instrument state in state register 1.

```
10 OUTPUT 718;"IP;CF 300MZ;SP 20MZ;"  
20 OUTPUT 718;"SAVES 1;"  
30 END
```

Note: The functions of the SAVES command and the *SAVE* key are identical. (Also see OL.)

SAVET

Save Trace

Description: The SAVET command saves the selected trace and state in the specified register if it has not been locked by PSTATE ON. (See SAVES command.)

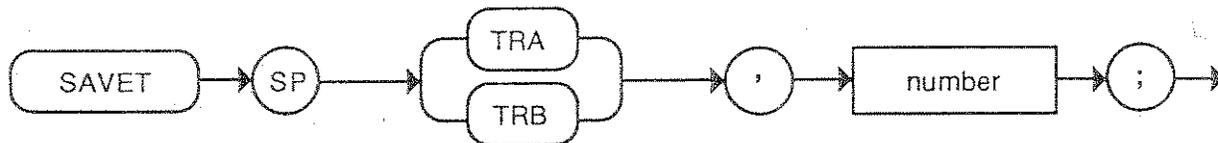
Parameters:

TRA: stores the contents of Trace A.

TRB: stores the contents of Trace B.

number: integer from 0 to the TRCMEM — 1.

Syntax:



Example: The following program saves both the trace and the instrument state.

```

10 INPUT "SELECT THE TRACE YOU WISH TO SAVE (TRA OR TRB)", Save_trace$
20 INPUT "SELECT THE REGISTER NUMBER", Reg_number
30 OUTPUT 718;"SAVET ";Save_trace$;"", ";Reg_number;"";"
40 END
  
```

Note: The SAVET and TRA commands are equivalent to the [SAVE TRACE] softkeys.

SNGLS

Single Sweep

Description: The SNGLS command sets the analyzer to single_sweep mode. Each time TS (take sweep) is sent, one sweep is initiated, as long as the trigger and data entry conditions are met. SNGLS allows a 0 GHz to 22 GHz span. (Also see TS.)

Syntax: See Figure 4-1

Example: OUTPUT 718;"SNGLS;"

Note: The functions of the SNGLS and S2 commands and the [SINGLE SWEEP] softkey are identical.

Note: The values are bounded by the instrument range only; therefore, band 0 can be included in a multiband sweep in single mode.

SP

Span

Description: The SP command changes the total displayed frequency range symmetrically about the center frequency. The frequency span readout refers to the displayed frequency range. Divide the readout by ten to determine the frequency span per division. If span width couples to the resolution and video bandwidths, 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 it couples. All of these functions normally couple, 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.

Syntax: See Figure 4-7. The AUTO parameter does not apply.

Example: OUTPUT 718;"SP 10MZ;"

Note: The functions of the SP command and the *SPAN* key are identical.

SRQ

Service Request

Description: A service request is a method for the analyzer to notify an external controller of an analyzer condition. 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:

1. Units key pressed in enable entry mode
2. End of sweep
3. Hardware broken
4. Command complete
5. Illegal remote command.

A service request is generated only if the proper request mask bit has been set (see RQS) and either the condition itself or the Simulate Service Request (see RQS command) 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 reenabte 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, 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.

Example: OUTPUT 718;"RQS 8;SRQ 8;"

In the previous example, the hardware broken interrupt is generated. A program can then respond to the interrupt exactly in the same way as it would under a true broken hardware condition.

Interface Differences

As implemented on the HP-IB interface, an analyzer service request asserts the SRQ control line on the HP-IB.

On the HP-IL interface, the analyzer can send a Service Request to the loop. This is similar to asserting the SRQ control line.

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** following 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

Additionally, the HP-IB and HP-IL interfaces support interface commands to read the status byte:

On HP-IB in HP series 200 or 300 BASIC, the statement `SPOLL(Device_address)` may be used to read the status byte.

On HP-IL in HP-71 BASIC, the `SPOLL(Device_specifier)` instruction may 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.

Syntax: See Figure 4-2

Table 4-7 Analyzer Status Register

BIT NUMBER	DECIMAL EQUIVALENT	ANALYZER STATE	DESCRIPTION
7	128		Not Used
6	64	SRQ ENABLE	Service Request Line is assigned by the analyzer
5	32	ILLEGAL COMMAND	Set when illegal command present
4	16	COMMAND COMPLETE	Any command is completed
3	8	HARDWARE BROKEN	Hardware Broken
2	4	END OF SWEEP	Set when any sweep is completed
1	2	UNIT KEY	Set when unit key pressed
0	1		Not Used

SS

Center Frequency Step Size

Description: The SS command specifies center frequency step size. This command is designed for use with the UP and DN parameters.

Syntax: See Figure 4-7. The AUTO parameter removes it as an active function but does not have an effect on its value.

Example: OUTPUT 718;"SS 10MZ;"

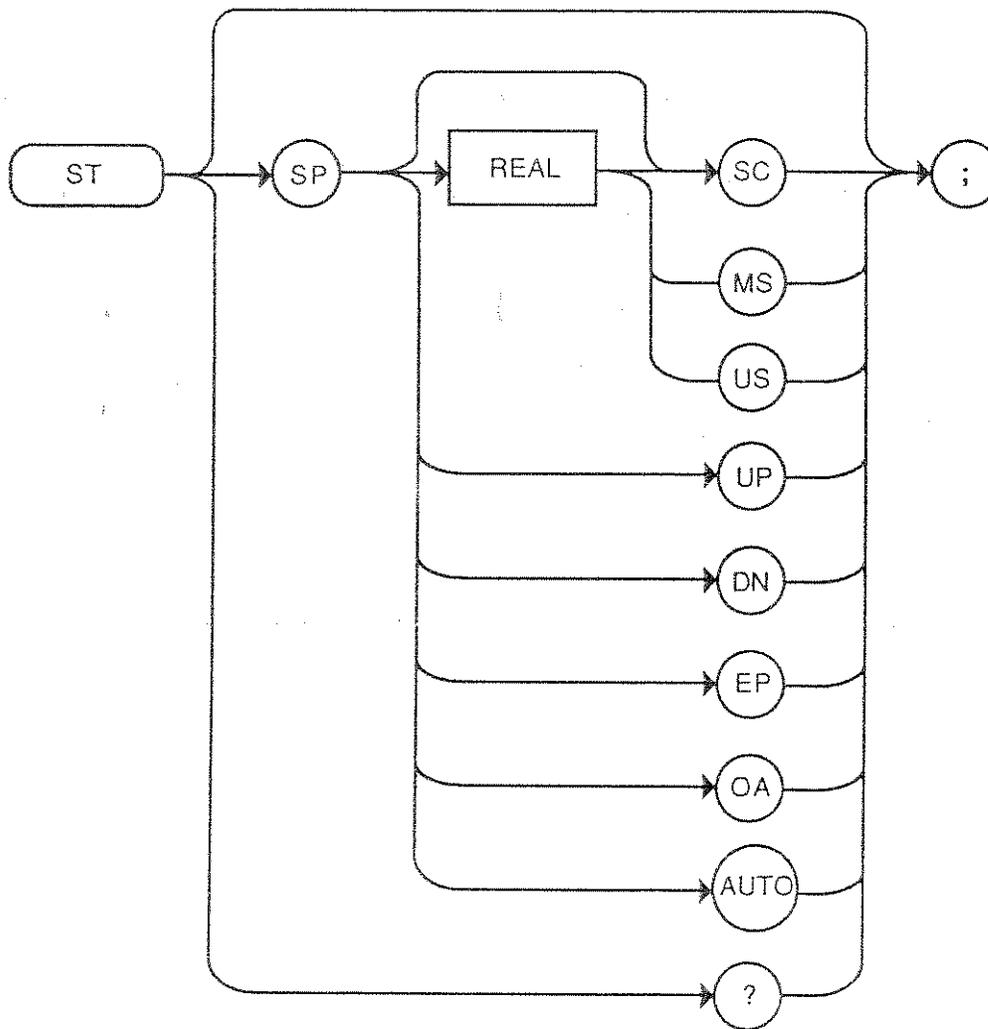
Note: The functions of the SS command and the [CF STEP SIZE] softkey are identical.

ST

Sweep Time

Description: The ST command specifies the time in which the analyzer sweeps the displayed frequency range. The minimum time is 20 msec/GHz.

Syntax:



Example: OUTPUT 718; "ST 100MS;" sets the sweep time to 100 milliseconds.

Note: The functions of the ST command and the [SWEEP TIME] softkey are identical.

STB

Status Byte Query

Description: The STB command returns to the controller the decimal equivalent of the bits set in the status byte (see commands RQS and SRQ). STB 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. For more information, refer to Appendix B of the Installation Manual.

Syntax: See Figure 4-8

TA

Transfer A

Description: The TA command transfers trace A amplitude values from the analyzer to the controller. 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.

Syntax: See Figure 4-1

Note: Items are separated by CR/LR and are dependent upon MDS & TDF commands.

TB

Transfer B

Description: The TB command transfers trace B amplitude values from the analyzer to the controller. The operation of TB is similar to the operation of TA. (Also see TA.)

Syntax: See Figure 4-1

Example: OUTPUT 718;"TB;"

TDF

Trace Data Format

Description: The trace data format, TDF, command formats trace information for return to the controller.

M, measurement units, returns values in display units from -32768 to +32767.

P, parameter units, returns absolute measurement values such as dBm or Hz.

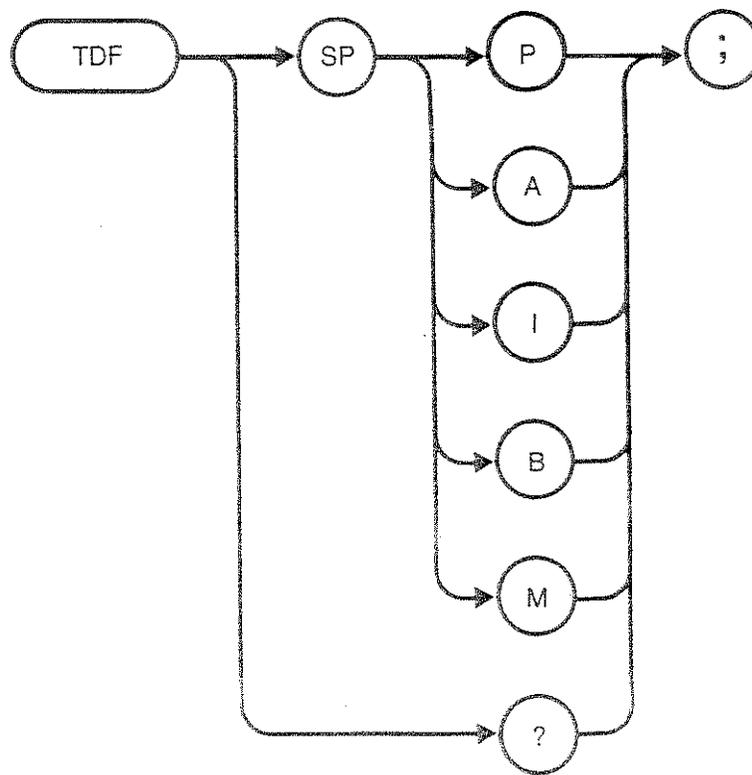
A returns data as an A-block data field. The MDS command determines whether data comprises one or two 8-bit bytes.

I returns data as an I-block data field. The MDS command determines whether data comprises one or two 8-bit bytes.

Specifying **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's internal amplitude units (log: hundredths of dBm; linear volts: 8,000 = top graticule line and 0 = bottom graticule line). **P** is in the current parameter unit specified by AUNITS.

Syntax:



Example: OUTPUT 718; "TDF M;"

Note: Alternate commands O1, O2, O3, and O4 provide backward compatibility for other Hewlett-Packard spectrum analyzers.

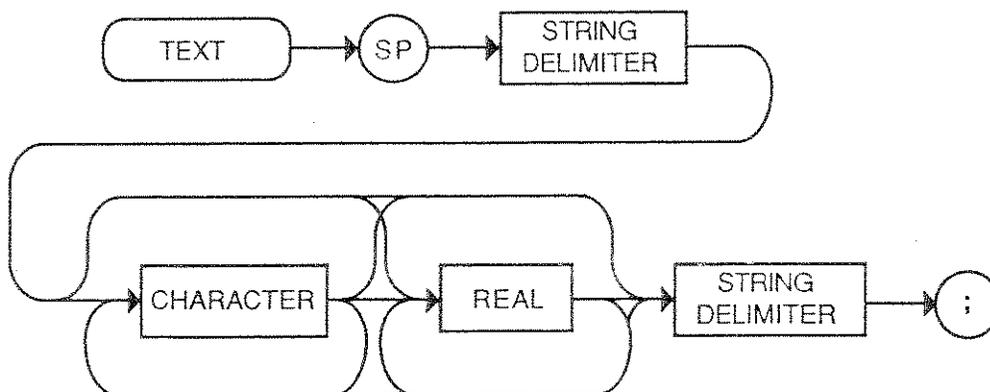
O1 = "TDF P"
 O2 = "TDF B; MDS W;"
 O3 = "TDF M;"
 O4 = "TDF B; MDS B;"

TEXT

Text

Description: The TEXT command writes text on the analyzer screen at the current pen position.

Syntax:



The string delimiters, which mark the beginning and end of the string, must match. Characters are alphanumeric ASCII characters 32 through 126.

Example: OUTPUT 718; "PA 80,80;TEXT%CONNECT ANTENNA%;"

TH

Threshold

Description: Similar to a base line clipper, the TH command blanks signal responses below the threshold level. The threshold level is nine graticule divisions below the reference level unless otherwise specified. The UP and DN commands move the threshold one division. (Also see AUTO.) The threshold level is annotated in reference level units at the lower-left corner of the analyzer screen and appears as a line on the display.

Syntax: See Figure 4-6. The AUTO parameter deactivates the threshold function.

Example: OUTPUT 718;"TH UP;"

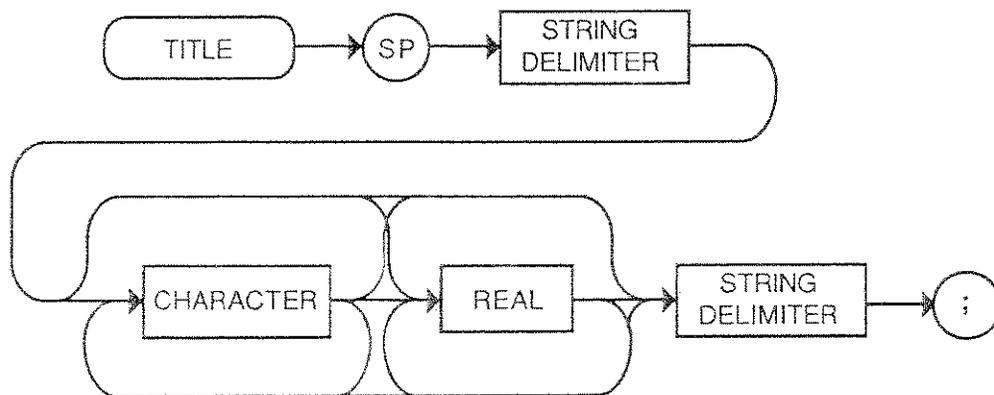
Note: Alternate command T0 turns the threshold off. The functions of the TH command and the [THRESHOLD] softkey are identical.

TITLE

Title

Description: The TITLE command activates the title mode. This function writes a message at the top of the analyzer screen. Any character on the controller keyboard can be written. The full width of the display is available for writing a maximum of 58 characters. However, the marker readout may interfere with the last 21 characters. IP removes the message.

Syntax:



Example: 20 OUTPUT 718;"TITLE%Adjust Antenna%;"

Note: The functions of the TITLE command and the [SCREEN TITLE] softkey are identical.

TM

Trigger Mode

Description: The TM command selects a trigger mode: free, line, video, or external. Query returns the trigger mode. The conditions of the trigger modes are described below.

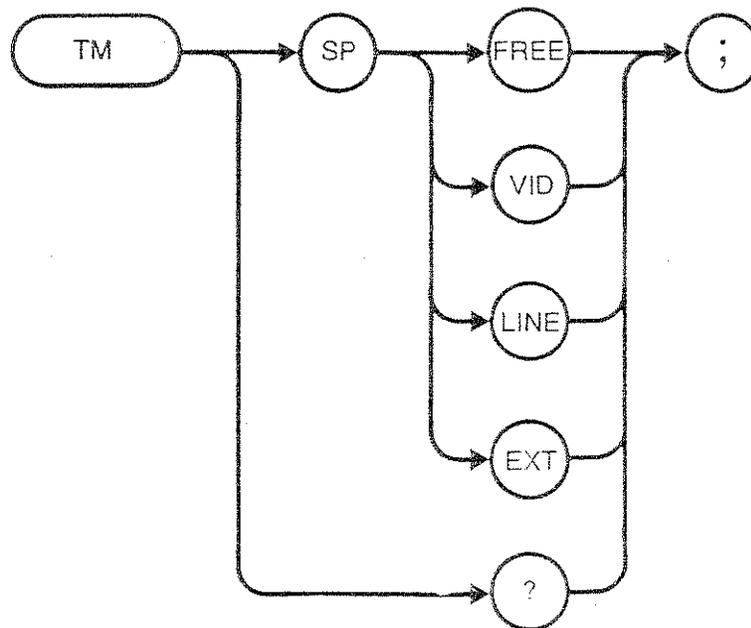
FREE allows the next sweep to start as soon as possible after the last sweep.

LINE allows the next sweep to start when the line voltage passes through zero, going positive.

VID allows the next sweep to start if the detected RF envelope voltage rises across a level set by the display line.

EXT allows the next sweep to start when an external voltage level passes through approximately 1.5 volts, going positive. The external trigger signal level must be between 0 V and +5 V.

Syntax:



Example: OUTPUT 718;"TM EXT;"

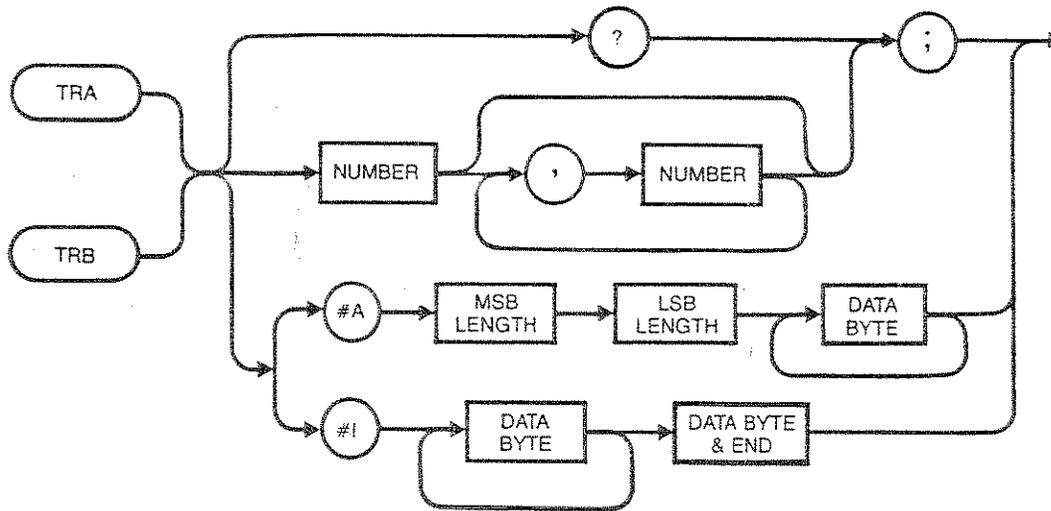
Note: The functions of the TM command and the *TRIG* key are identical. Alternate commands allow you to select the trigger modes: FREE (T1), LINE (T2), EXT (T3), VID (T4).

TRA/TRB

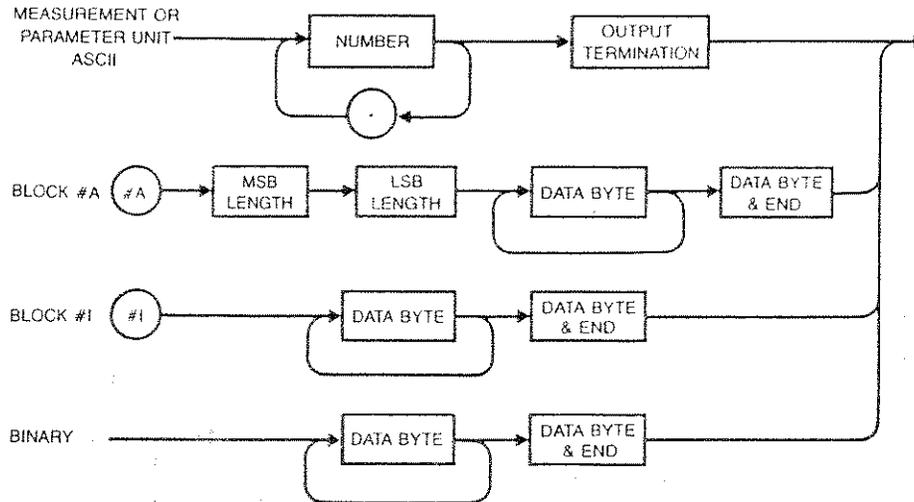
Trace Data Input/Output

Description: This command provides a method for reading or storing values into a trace. Input in #A or #I is considered 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.

Syntax:



The form of the query response is dependent upon the previously used Trace Data Format command as follows:



Example: See "Saving Trace Data" in Chapter 3 which illustrates "TDF P" format. Since TRA/TRB are fixed in length, there are always 401 or 802 bytes to be read from trace A or trace B if they are in binary input or binary output mode.

TRCMEM

Trace Memory

Description: The TRCMEM command returns a non-negative integer that indicates the total number of trace registers available for SAVET and RCLT.

Syntax: See Figure 4-3

Example: OUTPUT 718;"TRCMEM?;"

Note: A TRCMEM value of three indicates that trace registers 0, 1, and 2 are available for use.

TS

Take Sweep

Description: The take sweep command starts and completes one full sweep before the next command is executed. TS 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 previously set by the instrument is displayed in trace A.

```
OUTPUT 718;"IP;CF 11MZ;SP 20MZ;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 11MZ;SP 20MZ;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.

Syntax: See Figure 4-1

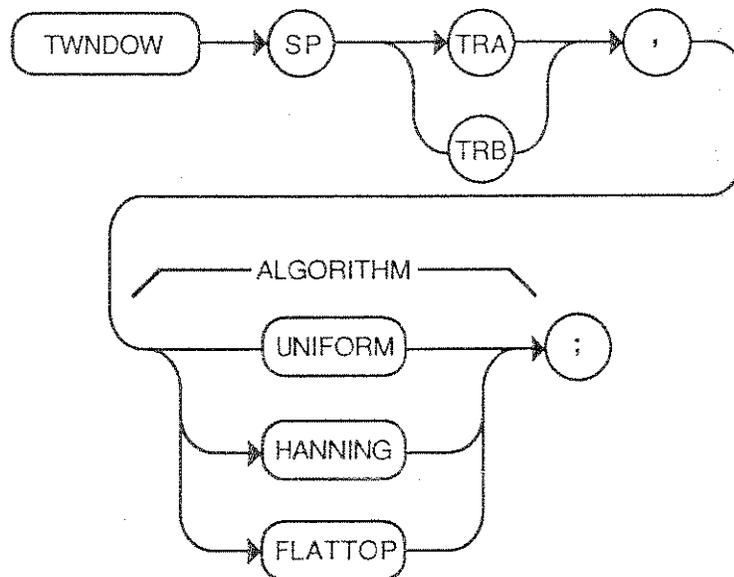
Example: OUTPUT 718;"IP;SNGLS;TS;"

TWNDOW

Trace Window

Description: The TWNDOW command creates a window trace array for the Fast Fourier Transform Function (FFT). 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.

Syntax:



UP

Up Command

Description: Increases value of active function by the applicable step size.

Syntax: See Figure 4-1

Example: OUTPUT 718;"IP;RB1KZ;UP;"

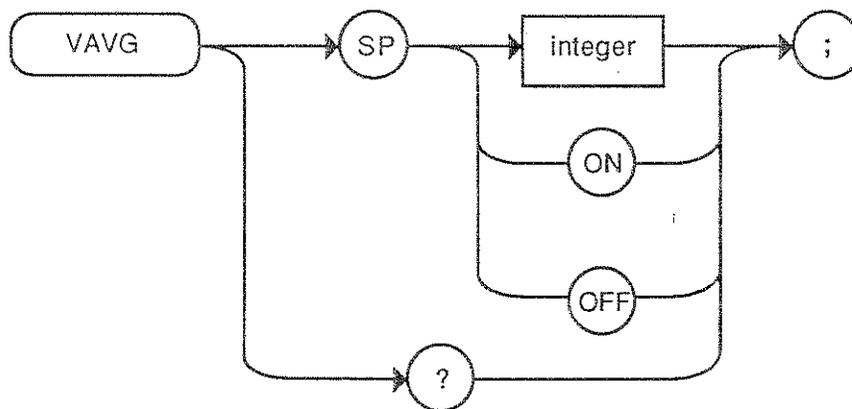
Note: The example sets the resolution bandwidth to 3 kHz.

VAVG

Video Average

Description: The VAVG command enables the video averaging function, which averages trace points to smooth the displayed trace. Use VAVG to view low-level signals without slowing the sweep time. Video averaging can lower the noise floor more than a 1-Hz video bandwidth if a large number of sweeps has been specified for averaging. Video average may also be used to monitor instrument state changes (changing bandwidths, center frequencies, etc.) while maintaining a low noise floor. The active function readout indicates the number of sweeps to be averaged; the default is 100 unless otherwise specified.

Syntax:



Example: OUTPUT 718;"VAVG 150;"

Note: Alternate command KSG turns video averaging on and KSH turns it off. The functions of the VAVG command and the [VIDEO AVERAGE] softkey are identical.

VB

Video Bandwidth

Description: The video bandwidth command specifies the video bandwidth, which is a post-detection filter. VB allows you to change the post-detection filter from 30 Hz to 3 MHz in a 1, 3, 10 sequence. The resolution, video bandwidth, and sweep time normally couple. Executing VB uncouples video bandwidth from res bandwidth. (It does nothing to the sweeptime/res bw coupling.) Executing AUTO recouples it to RBW.

Syntax: See Figure 4-7

Example: OUTPUT 718;"VB 10KZ;"

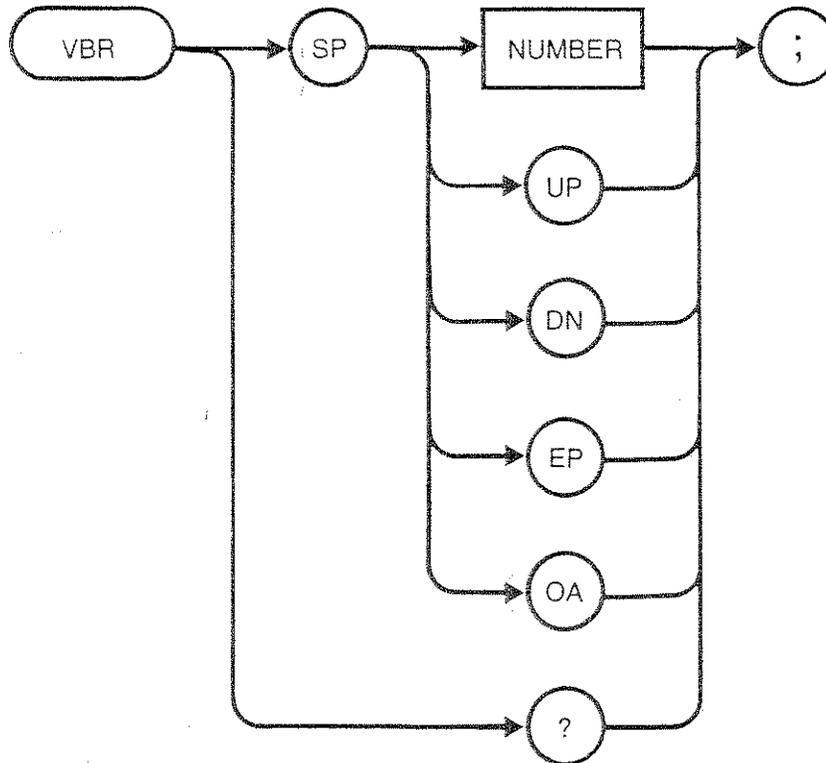
Note: The functions of the VB command and the [VIDEO BW] softkey are identical.

VBR

Video Bandwidth Ratio

Description: The VBR is multiplied by the resolution bandwidth to determine the automatic setting of video bandwidth if it is coupled to the res bandwidth. The VBR is selected in a 0.1, 0.3, 1, 3, 10 sequence.

Syntax:



Example: OUTPUT 718;"VBR 1;"

Note: The functions of the VBR command and the [VBW/RBW RATIO] softkey are identical.

VIEW

View

Description: The VIEW command displays trace A or trace B and stops the sweep when one trace appears on the screen in clear-write mode. The trace is not updated. When VIEW TRA is executed, the contents of trace A are stored in display memory. When VIEW TRB is executed, the contents of trace B are stored in display memory.

Syntax: See Figure 4-5

Note: The functions of the VIEW TRA and A3 commands and the [VIEW TRACE A] softkey are identical. The functions of the VIEW TRB and B3 commands and the [VIEW TRACE B] softkey are identical.

APPENDIX A

Analyzer Messages

The analyzer can generate various messages that appear on its screen during operation to provide an indication of progress through a procedure or 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.
- User-created error messages appear when the analyzer is used incorrectly. They are usually generated during remote operation.
- Informational messages indicate analyzer 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 - A/D $\pm 15V$ supplies can cause ADC-GND and ADC-2V. (H)

ADC-TIME FAIL

Indicates a failure in the processor - A/D A +5V supply can cause this. (H)

ADC-2v FAIL

Indicates a failure in the processor - A/D $\pm 15V$ supplies can cause ADC-GND and ADC-2V. (H)

CAL: FM SPAN SENS FAIL

The analyzer could not set up span sensitivity of the FM coil. (H)

CAL: LINEAR DET FAIL

The linear calibration routine failed. (H)

CAL: RES BW AMPL FAIL

The relative insertion loss of the resolution bandwidth is incorrect. (H)

CAL: SPAN SENS FAIL

The calibration span sensitivity routine failed. (H)

CAL:

During the calibration routine, messages may appear on the display indicating the routine is progressing: MC DELAY, FM DELAY, DONE, SWEEP, SWP DELAY, FREQ, SPAN, AMPTD, 3dB BW, ATTEN, LOG AMP, PEAKING, YTF. (M)

COMMAND ERROR:

The specified command is not recognized by the analyzer. The analyzer will recognize the commands described in Chapter 4. (U)

CONFLICT TABLE OVERFLOW

A command has been used that is not compatible with the analyzer. (U)

FAIL:

An error was discovered during the power-up check. The 4-digit by 8-digit code indicates the type of error. Error codes are described in the Support Manual. (H)

INVALID AUNITS:

The amplitude units are not valid. See the AUNITS command. (U)

INVALID DETECTOR:

The specified detector is not valid. See the DET command. (U)

INVALID ENTER FORMAT

The enter format is not valid. See the appropriate command description to determine the correct format. (U)

INVALID HP-IB ADDRESS OR OPERATION

An HP-IB operation was aborted due to an incorrect address or invalid operation. (U)

INVALID HP-IB OPERATION REN TRUE

The HP-IB operation is not allowed. (Usually caused by print/plot when a calculator is on the interface bus.) (U)

INVALID HP-IL ADDRESS OR OPERATION

An HP-IL operation was aborted due to an incorrect address or invalid operation. (U)

INVALID HP-IL OPERATION REN TRUE

The HP-IL operation is not allowed. (Usually caused by print/plot when a calculator is on the interface bus.) (U)

INVALID OUTPUT FORMAT

The output format is not valid. See the appropriate command description to determine the correct format. (U)

INVALID RS-232 ADDRESS OR OPERATION

An RS-232 operation was aborted due to an incorrect address or invalid operation. (U)

INVALID TRACE:

The specified trace is invalid. See trace commands (VIEW, MXMH, CLRW or BLANK). (U)

INVALID TRIGGER MODE:

The specified trigger mode is invalid. See the TM command. (U)

INVALID WINDOW TYPE:

The specified window is invalid. See the TWINDOW command. (U)

MEAS UNCAL

The measurement is uncalibrated. Check the sweep time, span, and bandwidth settings. (U)

PARAMETER ERROR:

The specified parameter is not recognized by the analyzer. See the appropriate command description to determine the correct parameters. (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 Manual. (M)

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

APPENDIX B

HP-IB Option 021

What You'll Learn in This Appendix

This appendix tells you how to connect a computer to your HP 8592A Spectrum Analyzer with the Hewlett-Packard Interface Bus (HP-IB), if that option was selected.

Your HP 8592A Option 021 has an HP-IB connector on the rear panel, as shown in Figure B-1.

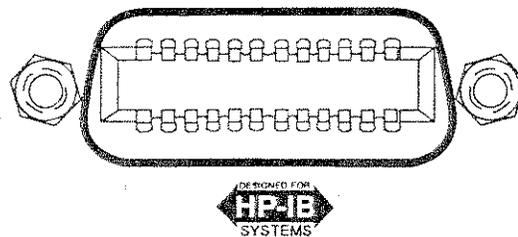


Figure B-1

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.

When you turn on the HP 8592A, the HP-IB address will appear on the screen (for example, HP-IB ADRS: 18). If necessary, you can reset the address of the HP 8592A by pressing **CONFIG** and [ANALYZER ADDRESS]. (You may use any address between 0 and 30. Usually, 1 is set aside for printers, and 5 for plotters.)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

APPENDIX C

HP-IL Option 022

What You'll Learn in This Appendix

This chapter tells you how to connect a computer to your HP 8592A Option 022 with the Hewlett-Packard Interface Loop (HP-IL), if that option was selected.

Introducing the Hewlett-Packard Interface Loop

The analyzer and all devices included in the HP-IL system are connected together in series. They communicate using HP-IL messages, which convey instructions and data. All HP-IL messages travel only in one direction. Each message sent on the loop is passed sequentially from one device to the next. If the message doesn't affect a particular device, the device passes the message on to the next device in the loop. When the message reaches the intended device, the device responds as directed and (usually) passes the message on to the next device. Eventually the message returns to the sender. All components of an HP-IL system have roles. The four roles defined for HP-IL components are:

Controller: The component that is responsible for issuing instructions to other components in the HP-IL system. Only one component can be the controller. The remaining components are often referred to as devices. The role of controller can be transferred to another device, but there can be only one controller at a time.

Talker: A component that has been assigned by the controller to send data to other devices. The controller can be a talker.

Listener: A component that has been assigned by the controller to receive data from the talker. The controller can be a listener.

Idle: A component is idle if it is not a talker or listener. Although a component may be idle, it still responds to instructions issued by the controller and passes all messages on to the next device. Each component of the HP-IL system connects into the loop using two connectors: one male and one female. This arrangement prevents improper connections HP-IL. The analyzer has two HP-IL connectors on the rear panel, as shown in Figure C-1.

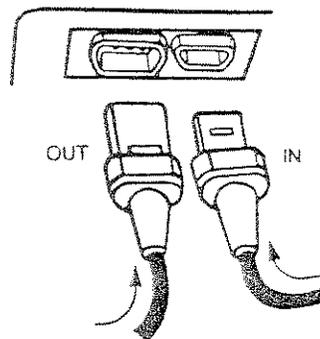


Figure C-1

HP-IL Addressing

When the controller of the HP-IL system configures the loop, it automatically assigns each device an address according to its position on the loop. Addresses are sequential: the first device (connected to the controller's OUT receptacle) is given the address 1. The next position is address 2 and so on. The last device (connected to the controller's IN receptacle) is given the highest address.

The analyzer is located in position 1 in the loop for the test programs in this chapter.

When you turn on the analyzer, its HP-IL address will appear on the screen (for example, HP-IL ADRS: 1). If necessary, you can change the address of the analyzer by pressing **CONFIG** and [ANALYZER ADDRESS]. Remember that this address will be changed when the controller configures the HP-IL loop.

There are several methods of identifying the HP 8592A with the HP-71B Handheld Computer.

- Address: The physical position of the HP 8592A on the HP-IL loop with respect to the present controller.
- Device word: The HP 8592A is of class INSTRMT.
- Device ID: The HP 8592A device ID is HP8592A.
- Accessory ID: The HP 8592A accessory ID is 90.

Refer to your computer HP-IL documentation for more complete HP-IL information.

Using the Computer as a Solid State Camera

The following two programs, "SAVSTATE" and "RCLSTATE," comprise what can be termed Solid-state camera application. The first program, "SAVSTATE," captures the measurement data and settings for Trace A, and saves it, along with the filename, time, and date, in a file in the HP-71B memory. In effect, a snapshot of the analyzer data and settings is stored in the computer's memory.

The "RCLSTATE" program retrieves the data from the HP-71B memory and sends it to the analyzer. The filename, date, and time at which the snapshot was taken are displayed in the title area of the analyzer screen. Trace data is put into trace B and trace B is put into view mode. Trace A remains available to make other measurements, which can be compared with trace B, if desired.

The filename in which the trace A data is stored can be up to eight characters long. The program does not check for the existence of a file with the same filename, and will write new data in place of the old file data. Each file requires about 1 kilobit of computer memory. To find how much memory is left in the HP-71B, enter:

MEM *Endline*

A useful feature of the solid-state camera is that any analyzer equipped with an HP-IL interface (Option 022) can be used to view the stored data. It does not need to be the analyzer from which the data was obtained.

Assigning User-Defined Keys

The two programs, or any programs for that matter, can be assigned a particular computer key. When a key is pressed, the assigned program executes. To assign the "S" key to "SAVSTATE" and the "R" key to "RCLSTATE," enter the following:

```
DEF KEY 'S', 'RUN "SAVSTATE":  
DEF KEY 'R', 'RUN "RCLSTATE":
```

Refer to the "Customizing the HP_71" section of your HP-71B Owner's Manual for more details on key assignment. To run the programs, simply press three keys on the HP-71B:

g 1 USER S for the "SAVSTATE" program
g 1 USER R for the "RCLSTATE" program

This technique can be used with other programs in this manual. For example, note that the first program in Chapter 2 is PROG1. We can assign a key for PROG1. You can run any of these programs with only three key strokes.

Subprograms

These two programs introduce the concept of subroutines, also known as subprograms. Subroutines are programs that perform complex functions and processes and are accessible simply by referencing the subroutine name (for example, GOSUB 'FILESAVE'). Subroutines are located after the "END" statement, and can be referenced anywhere in the main program.

Taking A Snapshot

This programming example stores the data measured by the analyzer, the analyzer state, the date, the time, and the filename in a file in the HP-71B. By deleting the comment lines in the program, the amount of memory taken up by the program can be reduced by more than 50%.

```
10 ! File = SAVSTATE
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 ! dimension Trace and Learnstring strings
60 DIM T$(802),L$(114)
70 ! dimension Prompt, Filename, and Title strings
80 DIM P$(20),F$(15),T1$(30)
90 A=DEVADDR("HP8592A") ! analyzer address
100 GOSUB 'GETSTATE'
110 GOSUB 'FILESAVE'
120 END
130 'GETSTATE': ! loads the current state into T$ & L$
140 ! ask analyzer for Trace A in binary format
150 OUTPUT :A ;"TDF B;TA;"
160 ! enter Trace A into T$
170 ENTER :A USING "#,802A";T$
180 ! ask analyzer for learnstring
190 OUTPUT :A ;"OL;"
200 ! enter learnstring into L$
210 ENTER :A USING "#,114A";L$
220 RETURN
225 ! FILESAVE saves the state in a file in main ram
230 'FILESAVE':
240 ! set up the prompt string P$
250 P$="          "&CHR$(27)&"<"&CHR$(255)&"SAV"
260 ! ask for filename from user
270 INPUT "FILENAME ?",P$,F$
280 ! truncate filename to 8 characters
290 F$=F$(1,8)
300 ! put the filename, date, and time into T1$
310 ! the date is reformatted to read mm-dd-yy
320 T1$=" "&F$&"          "&DATE$(4,5)&"- "&DATE$(7,8)
330 T1$=T1$&"- "&DATE$(1,2)&" "&TIME$(1,5)
340 ! open file F$ and assign it to channel #1
350 ASSIGN #1 TO F$
360 ! print the Title, Trace, and
365 ! Learnstring strings to the file F$
370 PRINT #1;T1$,T$,L$
380 ! close the file F$ and channel #1
390 ASSIGN #1 TO *
400 RETURN
```

Viewing a Snapshot

This program, "RCLSTATE," retrieves the information from a HP-71B file that was created by the program "SAVSTATE." The learnstring is sent to the analyzer. It then transmits the trace data to trace B and puts trace B in view mode. Last, it displays the filename, date, and time of the "snapshot" in the title area of the analyzer screen. Thus, you can view data collected from any number of places and times.

```
10 ! File = RCLSTATE
20 RESET HPIL
30 RESTORE IO
40 DESTROY ALL
50 ! dimension Trace and Learnstring strings
60 DIM T$(802),L$(114)
70 ! dimension Prompt, Filename, and Title strings
80 DIM P$(20),F$(15),T1$(30)
90 A=DEVADDR("HP8592A") ! analyzer address
100 GOSUB 'FILEGET'
110 ! send the learnstring to the analyzer
120 OUTPUT :A USING "114A";L$
130 ! put the analyzer in continuous sweep mode
140 OUTPUT :A ;"CONTS"
150 ! blank Trace A, clear-write Trace B, view Trace B
160 OUTPUT :A ;"BLANK TRA;CLRW TRB;VIEW TRB;"
170 ! send trace data into Trace B of analyzer
180 OUTPUT :A USING "2A,802A";"IB";T$
190 ! view Trace B
200 OUTPUT :A ;"VIEW TRB;"
210 ! put the Title string into the title area
215 ! of the analyzer display
220 OUTPUT :A ;"TITLE!";T1$;"!"
225 ! clean up the screen
230 OUTPUT :A ;"HD;ANNOT OFF;ANNOT ON;"
240 END
250 'FILEGET': ! gets the state from a file in main ram
260 ! set up the prompt string P$
270 P$="          "&CHR$(27)&"<"&CHR$(255)&"RCL"
280 ! ask for filename from user
290 INPUT "FILENAME ?",P$,F$
300 ! truncate filename to 8 characters
310 F$=F$(1,8)
320 ! open file F$ and assign it to channel #1
330 ASSIGN #1 TO F$
340 ! read Title, Trace, and Learnstring strings
345 ! from the file F$
350 READ #1;T1$,T$,L$
360 ! close the file F$ and channel #1
370 ASSIGN #1 TO *
380 RETURN
```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

APPENDIX D

RS-232 Option 023

What You'll Learn in This Appendix

This appendix explains how to connect a computer to the HP 8592A Spectrum Analyzer, using the RS-232 interface and 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, if that option was selected as shown in Figure D-1.

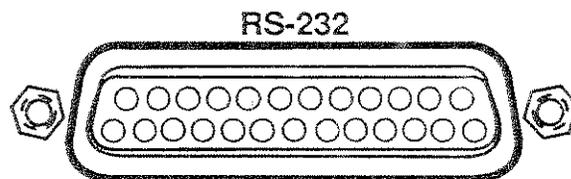


Figure D-1

The RS-232 interface or connection, 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 3-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.

What Is RS-232?

RS-232 is a means of communication between instruments. RS-232 uses serial data transmission where the data is transmitted one bit at a time. There are two data carrying signals:

- Transmit Data (TxD) — the serial data output. This line connects to the RxD input line.
- Receive Data (RxD) — the serial data input. This line connects to the TxD output line.

In addition to the data signals, there are five other signals used to control the flow of data. Listed below are the personal computer handshake signal descriptions:

- Request To Send (RTS) — Output signal indicates the system 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 system is ready to receive data.
- Data Terminal Ready (DTR) — Output signal from the analyzer. When the input buffer is full, this line will go false.
- Data Set Ready (DSR) — Input to the analyzer to lockout the front panel, when true. The front panel is enabled when this line is false.
- Data Carrier Detect (DCD) — Input to the analyzer. If DCD is true, the analyzer will receive data from the PC. 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 goes 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. Baud rates of 1200 and 9600 baud are very common.

The programs contained in this manual will all function at 1200 baud. Some are set to run at 9600 baud. If your RS-232 connection handshake lines perform as described, you can use a 9600-baud data rate. If you need to change the baud rate, refer to the "Setting the Analyzer Baud Rate" section of 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:
 - Xon/Xoff, ENQ/ACK (Enquire, Acknowledge) not supported by analyzer

When BREAK is issued to the analyzer with Option 023, the following occurs:

1. Aborts present command
2. Clears the input buffer
3. Clears output buffer
4. Stops all trace output in process
5. Reinitializes command parser

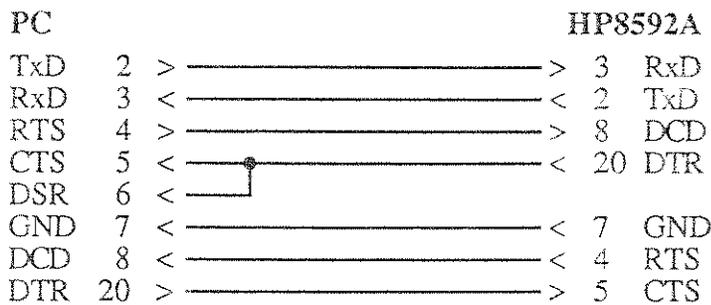
BREAK does not:

1. Invoke instrument preset
2. Clear SRQ off screen
3. 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 D-1 lists the signal connections between a personal computer and the analyzer.

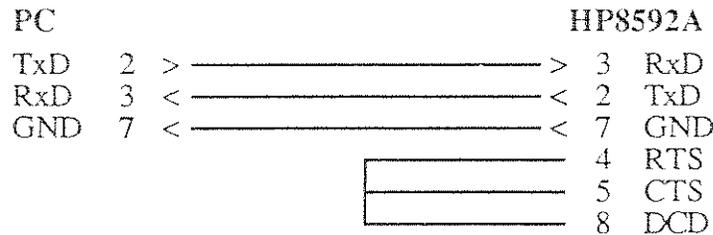
Table D-1. Full Handshaking Connection



If your computer operates with only three wires, you can use the cable connections in Table D-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 PC RTS line.

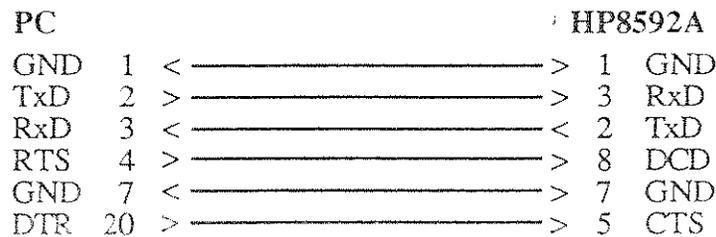
Table D-2. 3-Wire Connection



Connecting a ThinkJet Printer

To connect an HP ThinkJet printer to the analyzer, use the information in Table D-3.

Table D-3. ThinkJet Printer Connection



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 D-4. The connection is for a Hayes 1200 Modem and the analyzer.

Table D-4. Modem Connection

PC				HP8592A
GND	1	< _____ >		1 GND
TxD	2	> _____ >		2 TxD
RxD	3	< _____ <		3 RxD
CTS	5	> _____ >		5 CTS
GND	7	< _____ >		7 GND
CD	8	> _____ >		8 CD
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 D-5.

Table D-5. HP-GL Plotter Connection

PC			HP8592A	
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.

Switch Position

Expand: ←
 Emulate: ←
 Standalone: ←
 Monitor Mode: → (normal)
 Local → (normal)

Switch Position

Parity: → (Off)
 Even/Odd: → (Odd)
 Duplex: → (Full)
 Hardwire: ←
 DTR-Bypass: → (normal)

HP 8592A Front-Panel Lockout

If you wish to lock out the front panel of the analyzer, set the analyzer DSR input true. You can accomplish this by connecting the analyzer DSR input to the PC RTS output.

DSR True — Front-panel lockout (an R will appear on the screen).

DSR False — Front-panel active (you can use the keyboard).

Setting the HP 8592A Baud Rate

The baud rates of the analyzer and the PC must be the same. For example, to set the analyzer to 9600 baud, use the following procedure:

1. Press the *CONFIG* key on the analyzer.
2. Press the [BAUD RATE] softkey.
3. Press these numerical keys: *9, 6, 0, 0, Hz*.

To set the baud rate to 1200 baud, substitute these keypad keys: *1, 2, 0, 0, Hz*.

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

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

Index

AMB	4-5, 4-16, 4-18
AMBPL	4-5, 4-16, 4-18
Amplitude Commands	4-13
Amplitude Units	4-20
Analyzer Messages	A-1
ANNOT	4-5, 4-13, 4-18
Annotation	4-18
APB	4-5, 4-16, 4-19
Array	3-2
AT	4-5, 4-13, 4-19
Attenuation	4-19
AUNITS	4-5, 4-13, 4-20
AUTO	4-5, 4-21
Auto Couple	4-21
AXB	4-5, 4-16, 4-21
Band Lock ON	4-39
Band Lock OFF	4-40
Bandwidth Commands	4-13
BLANK	4-5, 4-16, 4-22
Blank Trace	4-22
BL	4-5, 4-16, 4-22
Baud Rate	D-2
Setting	D-7
CAL	4-5, 4-13, 4-23
Calibration	4-23
Calibration and Diagnostic Control	4-13
Center Frequency	4-24, 4-39
Center Frequency Step Size	4-77
CF	4-5, 4-14, 4-24
Clear Write	4-24
CLRW	4-5, 4-16, 4-24
CLS	4-5, 4-14, 4-24
Clear Status Byte	4-24
CNF	4-5, 4-13, 4-25
Command Parameters	4-9
Command Syntax Summary	4-5
Composing the Program	2-2
Computer/Analyzer Interconnections	1-3
Confidence Test	4-25
Configuring Your Computer System	1-3
Connecting a Modem	D-5
Connecting a ThinkJet Printer	D-4
Connecting an HP-GL Plotter	D-6
Connecting Your Analyzer to a Computer	1-3
Connecting Your Analyzer to a Printer or Plotter	1-2
Continuous Sweep	4-25, 4-40

Controller	C-1
Controlling Trace Data With a Computer	3-2
CONTS	4-5, 4-16, 4-25
Coupling Control	4-13
DET	4-5, 4-16, 4-26
Detection Mode	4-26
Display	4-28
Display Commands	4-13
Display Line	4-27
DL	4-5, 4-13, 4-27
DN	4-9, 4-27
DONE	4-5, 4-14, 4-16, 4-27
Done	4-27
Down Command	4-27
DSPLY	4-5, 4-13, 4-28
EE	4-5, 4-14, 4-15, 4-29
EK	4-5, 4-14, 4-15, 4-30
EM	4-5, 4-14, 4-30
Enable/Disable SAVES and SAVET	4-65
EP	4-5, 4-14, 4-31
Enable Entry	4-29
Enable Knob	4-30
Enhancing the Program with Variables	3-8
Enter Parameter Function	4-31
Erase Graphics Memory	4-30
Exchange Trace A and Trace B	4-21
FA	4-5, 4-14, 4-31
Fast Fourier Transform	4-33
FB	4-5, 4-14, 4-32
FFT	4-5, 4-16, 4-33
Flat Top Filter Window	4-35
FOFFSET	4-5, 4-14, 4-36
Frequency Commands	4-14
Frequency Offset	4-36
Front-Panel Lockout	D-7
Functional Index	4-13
Getting Information from the Analyzer	3-11
Graphics Commands	4-14
GRAT	4-5, 4-13, 4-37
Graticule	4-37

Hanning Filter Window	4-35
Harmonic Distortion	3-20
Harmonic Number	4-37
Harmonic Number Lock	4-38
HD	4-5, 4-13, 4-14, 4-15, 4-37
HN	4-5, 4-14, 4-37
HNLOCK	4-5, 4-14, 4-38
HNUNLK	4-5, 4-14, 4-41
Hold Data Entry	4-37
How to Use This Manual	3
HP-IB	B-1
HP-IL	C-1
Accessory ID	C-2
Address	C-2
Addressing	C-2
Connections	C-2
Device ID	C-2
Device Word	C-2
Messages	C-1
HP Series 200 Personal Technical Computer	1-5
Equipment	1-5
Interconnection Instructions	1-5
Test Program	1-6
HP Series 300 Technical Computers	1-7
Equipment	1-7
Interconnection Instructions	1-7
Test Program	1-7
HP Vectra Personal Computer (HP-IB)	1-8
Equipment	1-8
Interconnection Instructions	1-8
Test Program	1-8
HP Vectra Personal Computer (RS-232)	1-9
Equipment	1-9
Interconnection Instructions	1-9
Test Program	1-11
HP-71B Handheld Computer	1-13
Equipment	1-13
Interconnection Instructions	1-13
Test Program	1-14
HP-75D Handheld Computer	1-15
Equipment	1-15
Interconnection Instructions	1-15
Test Program	1-16
IBM PC/ST/AT and Compatible Computers	1-12
Equipment	1-12
Interconnection Instructions	1-12
Test Program	1-12

ID	4-5, 4-14, 4-41
Identify	4-41
Idle	C-1
Information and Service Diagnostic Commands	4-14
Input Impedance	4-42
Input/Output Commands	4-15
Instrument Preset	4-43
Instrument State Commands	4-15
Interconnection Symptoms and Solutions	1-17
Interface Bus	B-1
Interface Loop	C-1
INZ	4-5, 4-13, 4-42
IP	4-5, 4-15, 4-43
LG	4-5, 4-13, 4-44
Linear Scale	4-44
Listener	C-1
LN	4-5, 4-13, 4-44
Logarithmic Scale	4-44
MA	4-5, 4-14, 4-15, 4-45
Manual Terms and Conventions	3
Marker Amplitude	4-47
Marker Amplitude Output	4-45
Marker Commands	4-15
Marker Delta	4-48
Marker Frequency	4-48
Marker Frequency Output	4-47
Marker Minimum	4-49
Marker Noise	4-50
Marker Normal	4-49
Marker Off	4-51
Marker Pause	4-51
Marker Peak	4-52
Marker Peak Excursion	4-53
Marker Step Size	4-54
Marker to Center Frequency	4-48
Marker to Reference Level	4-53
Marker to Span	4-53
Marker Track	4-54
Marker Type	4-55
Marker Zoom	4-57
Maximum Hold	4-56
MDS	4-5, 4-14, 4-46
Measurement Data Size	4-46
Measuring Harmonic Distortion	3-20
MF	4-5, 4-14, 4-15, 4-47
Mixer Level	4-56
MKA	4-5, 4-14, 4-15, 4-47

MKCF	4-5, 4-15, 4-48
MKD	4-5, 4-15, 4-48
MKF	4-5, 4-14, 4-15, 4-48
MKMIN	4-5, 4-15, 4-49
MKN	4-5, 4-15, 4-49
MKNOISE	4-5, 4-15, 4-50
MKOFF	4-6, 4-15, 4-51
MKPAUSE	4-6, 4-15, 4-51
MKPK	4-6, 4-15, 4-52
MKPX	4-6, 4-15, 4-53
MKRL	4-6, 4-13, 4-15, 4-53
MKSP	4-6, 4-15, 4-53
MKSS	4-6, 4-15, 4-54
MKTRACK	4-6, 4-15, 4-54
MKTYPE	4-6, 4-15, 4-55
ML	4-6, 4-13, 4-56
Modifying the Program	2-5
MXMH	4-6, 4-56
M4	4-6, 4-15, 4-57
OA	4-6, 4-57
OL	4-6, 4-14, 4-58
Operator Entry Commands	4-15
Output Learn String	4-58
PA	4-6, 4-14, 4-59
PD	4-6, 4-14, 4-60
PEAKS	4-6, 4-16, 4-60
Peaks	4-60
Pen Down	4-60
Pen Up	4-66
PLOT	4-6, 4-15, 4-62
Plot	4-62
Plot Absolute	4-59
Plot Relative	4-65
Plotter/Printer Commands	4-15
PP	4-6, 4-63
PR	4-6, 4-14, 4-64
Preselector DAC Number	4-65
PRINT	4-6, 4-14, 4-15, 4-65
Print	4-65
Printing History	3
PROG1	2-2
PROG2	2-5
PROG3	2-8
PROG4	2-12
PROG5	3-2
PROG6	3-5
PROG7	3-8
PROG8	3-10
PROG9	3-15

Programming Guidelines	2-16
Protect State	4-67
Protocol	D-3
PRSDAC	4-6, 4-65
PSTATE	4-6, 4-66
PU	4-6, 4-14, 4-66
RB	4-6, 4-13, 4-66
RCLS	4-6, 4-12, 4-15, 4-66
RCLT	4-6, 4-67
Reading Trace Data	3-2
Reading Trace Data from a Computer Disc	3-8
Recall State	4-66
Recall Trace	4-67
Reference Level	4-68
Reference Level Offset	4-69
Resolution Bandwidth	4-68
Returning the Analyzer to its Former State	3-16
REV	4-6, 4-67
Revision	4-68
RL	4-6, 4-13, 4-68
ROFFSET	4-6, 4-13, 4-69
RQS	4-6, 4-14, 4-70
RS-232 Interface	D-1
3-Wire Connection	D-4
Baud Rate	D-2
Clear to Send (CTS)	D-2
Data Carrier Detect (DCD)	D-2
Data Communications Equipment (DCE)	D-2
Data Set Ready (DSR)	D-2
Data Terminal Equipment (DTE)	D-2
Data Terminal Ready (DTR)	D-2
Full Handshaking Connection	D-3
Handshaking lines	D-1
HP-GL Plotter Connection	D-6
Modem Connection	D-5
Protocol	D-3
Receive Data (RxD)	D-2
Request to Send (RTS)	D-2
Serial Data Lines	D-1
ThinkJet Printer Connection	D-4
Transmit Data (TxD)	D-2

SAVES	4-6, 4-15, 4-71
Save State	4-71
SAVET	4-6, 4-16, 4-72
Save Trace	4-72
Saving and Recalling Instrument States	3-10
Saving the Analyzer's State	3-10
Saving Trace Data	3-5
Service Request	4-74
Service Request Mask	4-68
Single Sweep	4-40, 4-73
SNGLS	4-6, 4-16, 4-73
SP	4-6, 4-14, 4-73
Span	4-39, 4-73
SRQ	4-6, 4-14, 4-74
SS	4-6, 4-14, 4-76
ST	4-6, 4-16, 4-77
Status Byte Query	4-78
STB	4-6, 4-14, 4-78
Start Frequency	4-31, 4-39
Stop Frequency	4-32, 4-39
Sub Programs (HP-IL)	C-3
Summary of Compatible Commands	4-11
Sweep and Trigger Commands	4-16
Sweep Time	4-77
Symbols in Syntax Diagrams	4-1
Synchronization Commands	4-16
Syntax Conventions	4-1
Syntax Elements Enclosed in Rectangles	4-8
TA	4-6, 4-15, 4-16, 4-78
Taking a Snapshot	C-4
Talker	C-1
Take Sweep	4-85
TB	4-6, 4-15, 4-16, 4-78
TDF	4-6, 4-15, 4-79
TEXT	4-6, 4-80
Text	4-80
TH	4-6, 4-13, 4-81
Threshold	4-81
TITLE	4-6, 4-13, 4-81
Title	4-81
TM	4-6, 4-16, 4-82
TRA/TRB	4-6, 4-15, 4-16, 4-83
Trace A Minus Trace B	4-18
Trace A Minus Trace B Plus Display Line	4-18
Trace A Plus Trace B	4-22
Trace B Minus Display Line	4-22
Trace Commands	4-16

Trace Data Format	4-79
Trace Data Input/Output	4-83
Trace Math Commands	4-16
Trace Memory	4-85
Trace Window	4-86
Transfer A	4-78
Transfer B	4-78
TRCMEM	4-6, 4-16, 4-85
Trigger Mode	4-82
TS	4-7, 4-15, 4-16, 4-85
TWNDOW	4-7, 4-16, 4-86
Unlock Harmonic Number	4-40
Uniform Filter Window	4-35
Using the Computer as a Solid State Computer	C-2
VAVG	4-7, 4-16, 4-87
VB	4-7, 4-13, 4-87
VBR	4-7, 4-13, 4-88
Video Average	4-87
Video Bandwidth	4-87
Video Bandwidth Ratio	4-88
VIEW	4-7, 4-16, 4-89
View	4-89
Viewing a Snapshot	C-5
Writing Your First Program	2-2