

Errata

Title & Document Type: Using Instrument Basic with the
3577A

Manual Part Number: 03577-90030

Revision Date: March 1990

HP References in this Manual

This manual may contain references to HP or Hewlett-Packard. Please note that Hewlett-Packard's former test and measurement, semiconductor products and chemical analysis businesses are now part of Agilent Technologies. We have made no changes to this manual copy. The HP XXXX referred to in this document is now the Agilent XXXX. For example, model number HP8648A is now model number Agilent 8648A.

About this Manual

We've added this manual to the Agilent website in an effort to help you support your product. This manual provides the best information we could find. It may be incomplete or contain dated information, and the scan quality may not be ideal. If we find a better copy in the future, we will add it to the Agilent website.

Support for Your Product

Agilent no longer sells or supports this product. You will find any other available product information on the Agilent Test & Measurement website:

www.tm.agilent.com

Search for the model number of this product, and the resulting product page will guide you to any available information. Our service centers may be able to perform calibration if no repair parts are needed, but no other support from Agilent is available.

Operating Manual
Using HP Instrument BASIC
with the HP 3577B



HP Part Number 03577-90030
Microfiche Part Number 03577-90230

Printed in U.S.A.
Print Date: March 1990

©Hewlett-Packard Company, 1989, 1990. All rights reserved.
8600 Soper Hill Road, Everett, Washington 98205-1298

Table of Contents

Chapter 1: Introduction

Welcome to HP Instrument BASIC	1-1
Overview of HP Instrument BASIC	1-1
Using HP Instrument BASIC	1-1
How to Use This Manual	1-2
For Experienced Programmers	1-2
Typographical Conventions	1-3
Other Sources of Information:	1-3

Chapter 2: Recording and Running Programs

Recording a Measurement Sequence	2-1
Exercise 2.1 A Simple Recording Session	2-2
Running a program	2-5
HP Instrument BASIC Programs and the HP-IB Buffer	2-5
Error Message Display	2-5
Limitations of Keystroke Recording	2-6
Stopping a Program	2-7
Adding Program Lines	2-7
Printing and Viewing a Program	2-8
How to View a Program	2-8
How to Print a Program	2-9
Other Available Front-Panel Features	2-10
Scratching Your Program	2-10
Securing Your Program	2-12
Examining and Allocating Memory	2-14
Initializing Disks	2-16
General Information about Recorded Programs	2-17
What's in a Recorded Program	2-17
Rules of Recording	2-20

Chapter 3: Saving and Recalling Programs

Overview	3-1
Using an External Disk	3-2
Saving a Program to a Disk	3-4
Exercise 3.1 Using the [SAVE] Hardkey	3-4
Special Notes on Disk Storage	3-5
Recalling a Program from a Disk	3-7
Exercise 3.2 Recalling a Program from Disk	3-7
Autoloading a Program	3-8

Table of Contents

Chapter 4: Developing Programs Using an External Computer

Introduction	4-1
Using the HP BASIC Controller	4-2
Overview	4-2
Transferring Programs Via Disk	4-2
Downloading and Uploading Programs	4-3
Downloading Incompatible Code	4-4
Using an MS-DOS Computer	4-7
Overview	4-7
Transferring Programs Via Disk	4-7

Chapter 5: Program I/O

Introduction	5-1
Graphics and Display Techniques	5-2
Overview	5-2
Setting Up and Using Softkeys	5-2
The Waiting Loop	5-2
Drawing Graphics	5-3
Printing to the Screen	5-4
Clearing the Display	5-4
Interfacing with the General Purpose Input/Output Port	5-5
Overview	5-5
8-Bit Input	5-5
8-Bit Output	5-5
Using the DOS file System	5-6
Disk Formats and File Systems	5-6
Disk Volumes	5-6
Formatting Disks	5-6
File Types	5-7
More About DOS	5-7
Using a DOS disk to Exchange Data with a PC	5-8

Chapter 6: Interfacing with the HP-IB

Introduction	6-1
Communicating with Devices	6-2
HP-IB Device Selectors	6-2
Moving Data Through the HP-IB	6-3
General Structure of the HP-IB	6-3
Examples of Bus Sequences	6-5
General Bus Management	6-6
REMOTE	6-7
LOCAL LOCKOUT	6-8
LOCAL	6-9
TRIGGER	6-10
CLEAR	6-11
ABORT	6-12
HP-IB Service Requests	6-13
Passing and Regaining Control	6-15

The HP Instrument BASIC HP-IB Model 6-16

- External and Internal Busses 6-16
- Service Request Indicators 6-16
- HP Instrument BASIC as the Active Controller 6-17
- HP Instrument BASIC as a Non-Active Controller 6-19

Interfacing with an External Controller 6-20

- Transferring Data Between Programs 6-21
- Downloading and Uploading Programs 6-24

Chapter 7: Instrument-Specific HP Instrument BASIC Features

Introduction 7-1

Global Exceptions 7-2

- HP-IB, GPIO and RS-232 Interfaces 7-2
- CRT and Keyboard Interfaces 7-3
- Disk I/O 7-5
- Miscellaneous Command Differences 7-11
- Commands Not Implemented 7-11

Specific Exceptions 7-12

Appendix A: Extended HP-IB Command Reference

Appendix B: Example Programs

Introduction B-1

MSE B-1

PLOTf B-9

Index

Sales Offices

Introduction

Welcome to HP Instrument BASIC

Welcome to HP Instrument BASIC

This guide will help you learn about setting up and using your HP Instrument BASIC software. It will help you perform typical operations involving creating, executing, saving and recalling programs.

If you are new to programming or to HP BASIC languages, take the time to read this guide and perform the exercises. For many, this will provide all the information that is needed to create and run measurement sequence programs. If you are familiar with an HP Series 200/300 BASIC language, be sure to read the section "For Experienced Programmers" in this chapter.

Overview of HP Instrument BASIC

When installed in your HP 3577B, HP Instrument BASIC can be used for a wide range of applications, from simple recording and playback of measurement sequences to remote control of other instruments.

HP Instrument BASIC is a complete system controller residing inside your analyzer. It communicates with your analyzer via HP-IB commands and can also communicate with other instruments, computers and peripherals over the HP-IB interface or the general purpose I/O port.

Using HP Instrument BASIC

You need not be proficient in a programming language to successfully use HP Instrument BASIC. In recording mode, HP Instrument BASIC automatically builds an executable program by capturing measurement sequences as they are performed. You can immediately put your program to work controlling and automating your HP 3577B analyzer.

The HP Instrument BASIC command set is similar to the command set of HP Series 200/300 BASIC. In fact, HP Instrument BASIC programs can be run on any HP BASIC workstation with few if any changes. Refer to chapter 4 ("Using an External Controller") for information on transferring programs between the HP Series 200/300 BASIC and HP Instrument BASIC environments. Porting information can be found in chapter 10 of the "HP Instrument BASIC Programming Techniques" section of the *HP Instrument BASIC Users Handbook*. Also refer to chapter 7 ("Instrument-Specific HP Instrument BASIC Features") of this manual.

How to Use This Manual

If you want to use HP Instrument BASIC with a minimum of editing and programming you will find all the information you need to record, run, save and recall programs in chapters 2 and 3.

The tasks in each chapter, when performed in sequential order, demonstrate a typical use of HP Instrument BASIC and cover the most common tasks. Read the overview and try the sample tasks in each chapter to get you started. For more background information, you can read on in each chapter; otherwise, go to the next exercise and continue the session. You can refer back to the individual chapters for more information as necessary.

If you want to develop and download HP Instrument BASIC programs outside the instrument, continue on into chapters 4, 5, and 6. This will provide you with a good foundation for getting the most out of your HP Instrument BASIC programs. You will find that the *HP Instrument BASIC Users Handbook* also contains information to help acquaint you with the details and structure of the language.

For Experienced Programmers

If you are familiar with HP Series 200/300 BASIC, this manual is a good starting point to introduce you to the operating and programming environment and to provide you with examples of intermediate and advanced HP Instrument BASIC programs.

You will find detailed information on the language in the *HP Instrument BASIC Users Handbook*, including keyword descriptions, error messages, and programming techniques.

Here is a brief guide to help you locate the necessary information in this manual and the *HP Instrument BASIC Users Handbook* you will need:

- Review chapter 2 on recording programs to learn the front-panel interface
- If you plan to save and load recorded programs to an external mass storage drive review chapter 3
- If you plan to develop programs outside the instrument, read chapters 4, 5, 6, and 7
- Study the example programs listed and described in chapter 6 and appendix B
- If you want to port HP Series 200/300 BASIC programs to HP Instrument BASIC refer to chapter 10, "Keyword Guide to Porting," in the *HP Instrument BASIC Users Handbook*, and chapter 7, "Instrument-Specific HP Instrument BASIC Features," in this manual.

Typographical Conventions

The following conventions are used in this manual when referring to various parts of the HP Instrument BASIC and HP 3577B operating environments:

[HARDKEY]	Brackets [] surrounding a bold-faced name indicate the name of a hardkey on the front panel of the HP 3577B.
[SOFTKEY]	Brackets [] surrounding a name indicate the name of a softkey.
[SOFTKEY ON OFF]	Bolded selection in a softkey indicates the state after the softkey is pressed.
[HARDKEY] [SOFTKEY] [SOFTKEY]	A series of hardkeys and softkeys represents the path to a given softkey or menu.
Bold	A bold typeface is used to emphasize a particular word or phrase.
<i>Italic</i>	Italic typeface is used when referring to the name of a different manual.
<element>	Angle brackets are used to signify a syntax element in a statement.

Other Sources of Information:

- *HP 3577B Network Analyzer Operating Manual*
- *HP Instrument BASIC Users Handbook:*
 - *HP Instrument BASIC Programming Techniques*
 - *HP Instrument BASIC Interfacing Techniques*
 - *HP Instrument BASIC Language Reference*

Recording and Running Programs

Recording a Measurement Sequence

Keystroke recording is the easiest way to create HP Instrument BASIC programs. It requires only a couple of steps to set up and run, and can be accomplished with very little knowledge of programming.

HP Instrument BASIC programs communicate with the HP 3577B over an internal bus, using the same set of commands used by external controllers for remote operation of the instrument. Keystroke recording works by finding the bus command (called an HP-IB mnemonic) that fits each operation you perform from the front-panel and then building a program line that, when executed, duplicates that operation.

After a program is recorded, it may be saved to an external disk drive or uploaded to an external controller. Once saved, it may be recalled into instrument memory at a later time. See chapter 3 to learn how to save and recall programs and chapter 4 to learn how to upload and download programs. After you have recorded a program, or recalled it into the instrument, you can run it from the instrument front-panel.

This chapter describes not only how to record, run and stop your programs but also how to list, scratch and secure them. It also covers the rules and limits of using keystroke recording to generate HP Instrument BASIC programs.

Recording and Running Programs

Exercise 2.1 A Simple Recording Session

In this first exercise, we will walk through a typical recording session, making a tuned stub notch filter response measurement. To build this simple filter and set up the configuration refer to your *HP 3577B Network Analyzer Operating Manual*, chapter 2, "Making Measurements" for the Tuned Stub Notch Filter example.

This exercise will store the keystrokes as HP Instrument BASIC commands. You will then be able to run, list, or examine the program.

1. Press [INSTR PRESET]
2. Press [SPCL FCTN]

The following menu will appear:

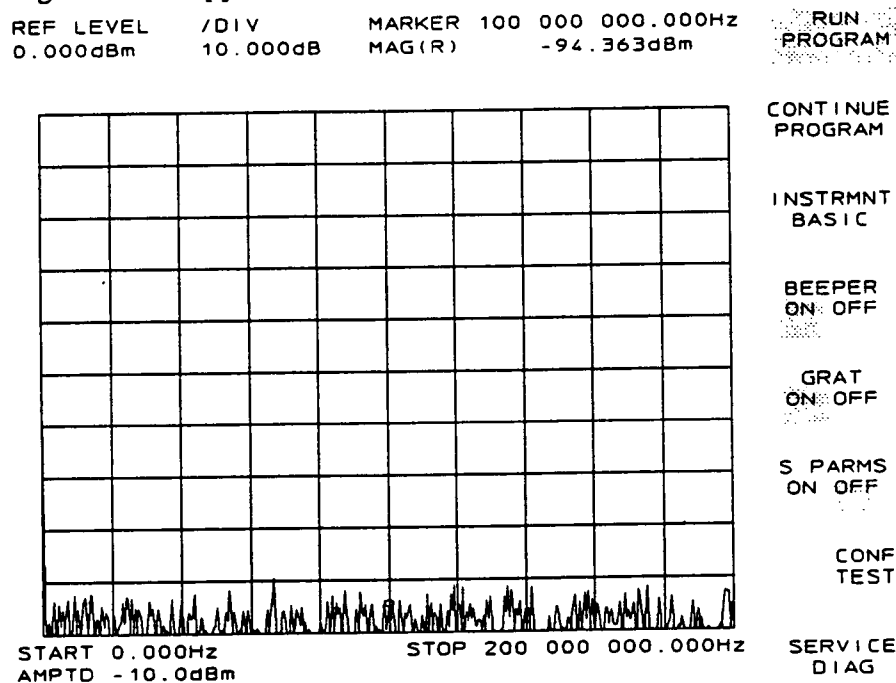


Figure 2-1. The [SPCL FCTN] Menu

1. Press [INSTRMNT BASIC]

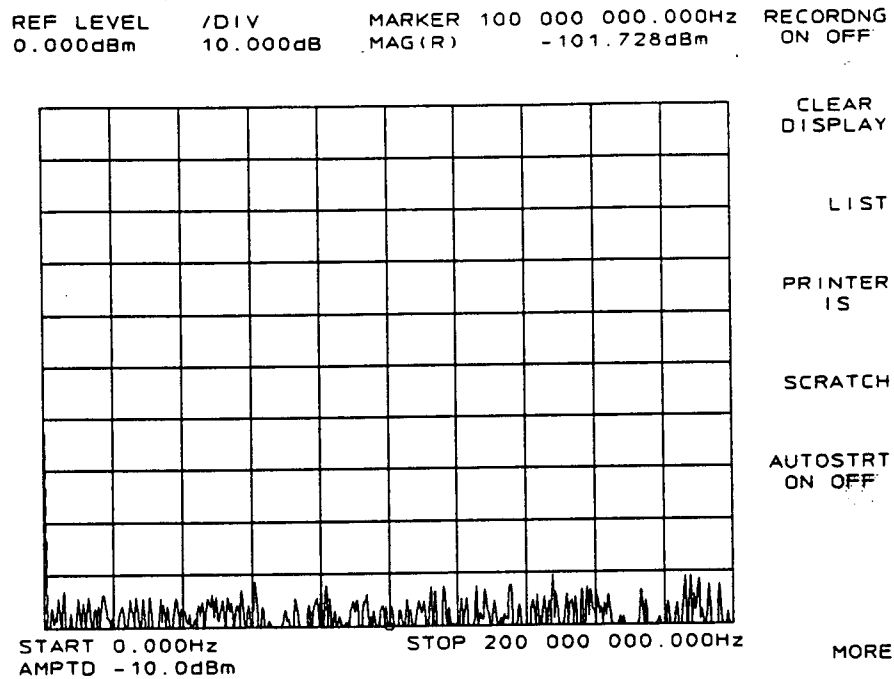


Figure 2-2. The [INSTRMNT BASIC] Menu

2. Press [RECORDING ON OFF]-

The display will indicate "Keystroke Recording ON" for a few seconds to show that subsequent keystrokes will be recorded into program memory.

This is all that is required to start the measurement sequence recording. The next steps comprise the actual measurement sequences to be recorded.

3. In the DISPLAY FORMAT area press

- [INPUT]
- [A]
- [TRACE2]
- [DISPLAY FCTN]
- PHASE
- [INPUT]
- [A]

4. In the SOURCE AREA press

- [FREQ]
- 1 [MHz]
- [STOP FREQ]
- 150 [MHz]

Recording and Running Programs

5. Return to the DISPLAY FORMAT area and press

[SCALE]
225 [deg]

Now you have completed a measurement sequence and simultaneously recorded the keystrokes as a program. The next step will terminate the keystroke recording.

6. Press [SPCL FCTN]

7. Press [INSTRMNT BASIC]

You will know that your program is recorded by the display "Keystroke Recording OFF."

You have just created a recorded program. In a moment we will learn more about various program management features but first let's try running this program.

Running a program

Running a program can be a simple two key procedure. Press [INSTR PRESET] to return to default states. Then press [SPCL FCTN] and the [RUN PROGRAM] softkey. If you look at the bottom right corner of the display, you will notice an R character in a box, indicating that the program is running. This indicator will disappear whenever the program stops, terminates, or is paused.

You should now see the exact sequence being performed that you recorded earlier.

HP Instrument BASIC Programs and the HP-IB Buffer

Recorded programs work by sending HP-IB commands to the instrument, which are queued into an input buffer by the instrument. An HP Instrument BASIC program generally outputs the commands much faster than the instrument can execute them, often causing the program to complete while the instrument is still executing the commands in the input buffer. The instrument then continues to process these commands until the buffer is empty.

This may have some side-effects if you are not aware of this interaction. For example, it may not be immediately obvious that the program has actually finished, since the instrument is still functioning "remotely." This could cause confusion if you try to pause and continue a program that has actually completed.

The other side-effect you may encounter involves synchronization between the buffered commands and the instrument sweep time. Some functions require that a sweep be completed before the ensuing function can perform the desired operation. These functions, whether keystroke recorded or programmed, include autoscaling, marker movement, normalization, and storing registers. In these cases the buffer may output the command before the analyzer has been able to complete a sweep so that the desired outcome may or may not occur. To ensure that a valid sweep is performed before these functions it is advisable to precede them with a "take measurement" command. In a keystroke recorded program this can be accomplished with the [TRIG/RESET] key.

Error Message Display

Any errors generated by your program will result in the same error messages as documented in appendix A of the *HP Instrument BASIC User's Handbook*. Error messages displayed on the HP 3577B HP Instrument BASIC display may be truncated if they exceed 26 characters in length.

Recording and Running Programs

Running a program

Limitations of Keystroke Recording

Many measurement sequences can be performed using only front-panel keys. For these measurements, keystroke recording will be adequate to create programs that can be stored and accessed for repeated use.

You will have to develop your program on an external computer if your applications require any of the following:

- Editing, adding, inserting or deleting lines
- Using the analyzer display for text, graphics or user defined softkeys
- Creating and filling data structures (variables, arrays, or COM statements)
- Communicating with other devices
- Analyzing data
- Error and timeout trapping

For assistance in developing programs to perform these tasks see chapters 4 through 7 and the example programs in Appendix B.

Stopping a Program

It may become necessary from time to time to stop or pause the operation of an HP Instrument BASIC program. You can do this by pressing front-panel keys while the program is running.

Pausing

A program can be suspended temporarily by pressing the [SPCL FCTN] hardkey. You will know that the program is paused because the [CONTINUE] softkey will be highlighted. The program line at which the pause occurred will also be displayed on the bottom line of the HP Instrument BASIC display area until a front-panel operation is performed. To continue the program, press [CONTINUE].

You can also pause a program by inserting a PAUSE statement in your program using an editor on an external computer (see chapter 4, "Developing Programs Using an External Computer"). When a program pauses itself, the instrument front-panel state remains in its current condition. This allows you to develop programs that pause and leave the front-panel in a pre-determined state to allow an operator direct access to desired menus. In this case, to continue the program the operator must press [SPCL FCTN] and then [CONTINUE] to continue the program.

Aborting

To abort a program completely, press the [LCL] hardkey at any time while the program is running. This causes an HP Instrument BASIC RESET. This does not clear the HP-IB buffer, and any commands sent to the instrument by the program that have not yet been executed will continue to be processed. If the instrument is in remote mode (if an external controller has started the HP Instrument BASIC program, for example), pressing the [LCL] key once will return the analyzer to local mode. This does clear the HP-IB buffer. The second [LCL] keypress will reset HP Instrument BASIC.

If the HP 3577B is not in remote mode you may press [INSTR PRESET] to abort the running program, reset HP Instrument BASIC, and reset the analyzer.

Adding Program Lines

If you turn keystroke recording on with a recorded program already in memory, additional program lines will be added at the end of the program (before the END statement). However, if you recall a program from disk then turn keystroke recording on, the lines will be added at the beginning of the program.

Therefore, it is better to add program lines with an external controller. You can then record multiple program segments, save them on an external disk, then combine them.

Printing and Viewing a Program

How to View a Program

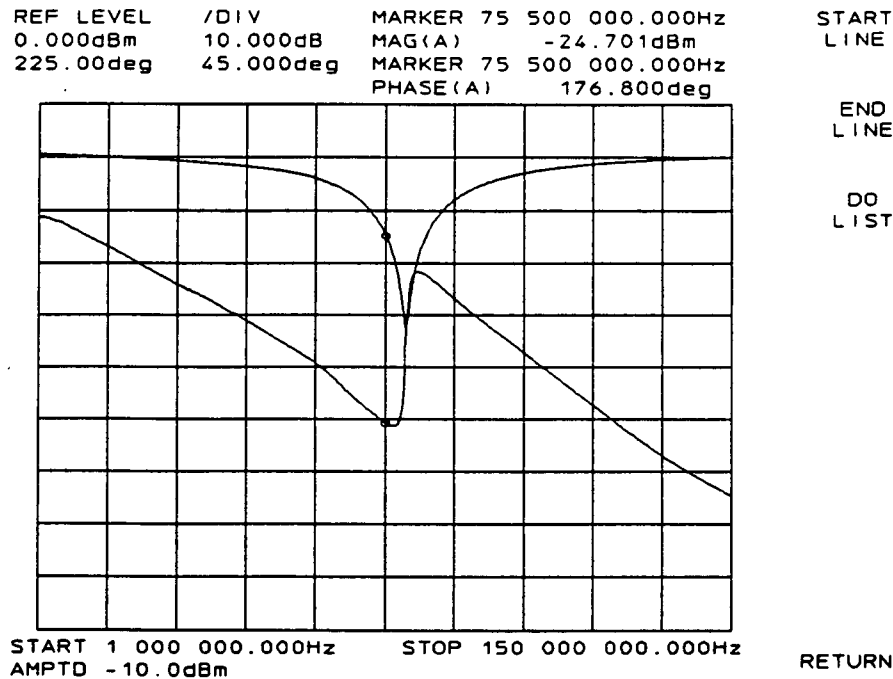


Figure 2-3. The [LIST] Menu

If you want to look at your program without a printer, you can list it to the analyzer's display. Press [SPCL FCTN] then [INSTRMNT BASIC] then [LIST] then [DO LIST]. This will scroll through the entire program. See figure 2-10 for an example of a program listed in the display area. The program listing area is 46 characters wide with lines exceeding 46 characters wrapping to the next line. The instrument's display area is 10 lines long, but only the last 9 lines will remain in view (a blank line is generated at after a display output, unless suppressed).

In order to view other parts of the program, you may use the [START LINE] and [END LINE] keys to select a block of lines. For example to select lines 1 through 7 press [START LINE], enter "1" with the keypad, then press [ENTER]. Next press [END LINE], enter "7" with the keypad, and press [ENTER]. To view the selected lines press [DO LIST]. Entering the value "0" for START LINE or END LINE will list from the beginning or to the end respectively. The block of lines selected here will remain in effect until you change it or press [INSTR PRESET]. The values selected here are also shared with the SECURE function so you may want to redefine the block before using the secure function, explained later in this chapter.

Listed lines will remain in the display memory unless you press [CLEAR DISPLAY]. If you wish run a program which uses the HP Instrument BASIC display area you should clear the display before running the program to avoid unwanted output in the display.

You may terminate a program listing by pressing [ABORT]. You may exit the [LIST] menu at any time by pressing [RETURN]

How to Print a Program

You may also list your program to an external HP-IB printer. Connect the printer to the HP 3577B through the HP-IB port and specify the printer's address. To specify the printer's address, press [SPCL FCTN] [INSTRMNT BASIC] [PRINTER IS], then use the keypad to enter the external printer select code and address (for example "701"), and press [ENTER]. The HP 3577B must be the system controller in order to list to the external printer. To change system control status press [LCL] [SYS CTRL ON OFF].

Note



If you have an external controller connected to the analyzer, you must either disconnect it or pass active control of the external interface to the analyzer before continuing.

You can then return to the [LIST] menu and press [DO LIST] to obtain a printed listing of the HP Instrument BASIC program.

The default printer address will be "1," which is the HP 3577B display. You may return to this address by pressing [INSTR PRESET] or by specifying "1" in the [PRINTER IS] function.

Other Available Front-Panel Features

Scratching Your Program

A program will remain in memory unless you power cycle the instrument or scratch the program. This means that if you turn recording on for keystroke recording with a program already in memory the new keystrokes will be added the existing program. If you want to record a new program you will have to scratch the old one.

Pressing [SCRATCH] in the [INSTRMNT BASIC] menu brings up the [SCRATCH] menu.

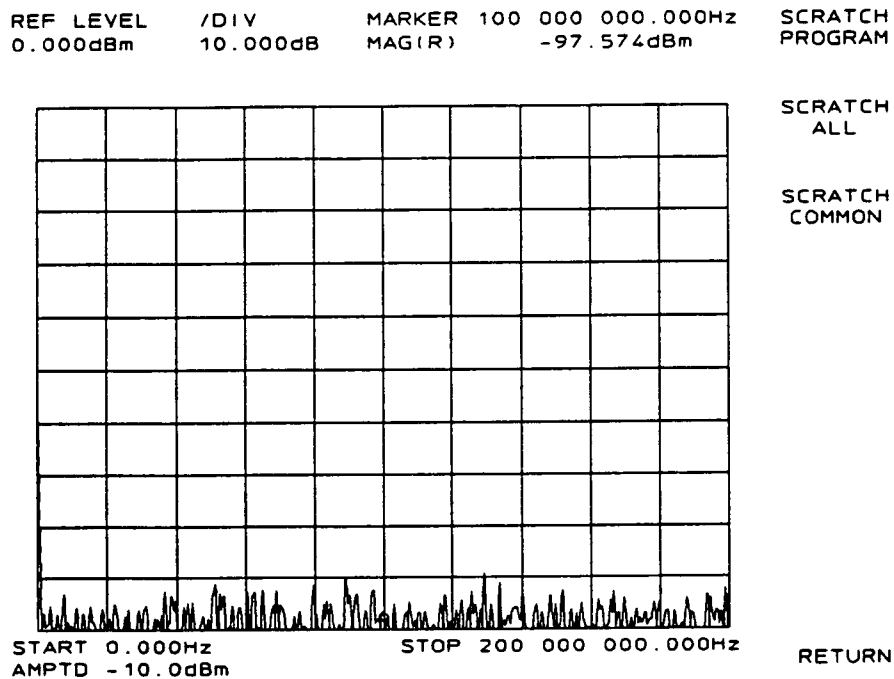


Figure 2-4. The [SCRATCH] Menu

The [SCRATCH PROGRAM] softkey clears the HP Instrument BASIC program and all variables not in COM

The [SCRATCH ALL] softkey clears the HP Instrument BASIC program memory, and all variables, including those in COM

The [SCRATCH COMMON] softkey clears all variables including those in COM, but leaves the program intact. This can only apply to programs developed on an external controller.

Select either [SCRATCH PROGRAM], [SCRATCH ALL], or [SCRATCH COMMON], and then press [RETURN] to get back to the [INSTRMNT BASIC] menu.

Additional features are available by pressing [INSTRMNT BASIC][MORE] which displays this menu:

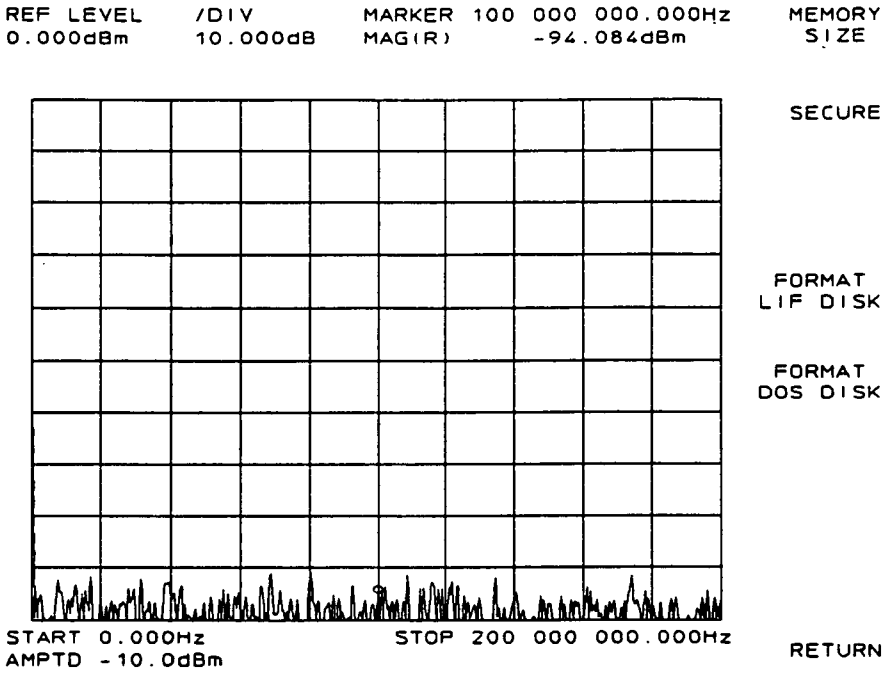


Figure 2-5. The [INSTRMNT BASIC] [MORE] Menu

The [MEMORY SIZE] softkey allows you to view and adjust memory allocation for your program.

The [SECURE] softkey allows you to secure your program from being viewed or edited.

The [FORMAT LIF DISK] and [FORMAT DOS DISK] softkeys allow you to initialize a disk in an external disk drive.

Each of the softkeys in this menu is described in the remainder of this section.

Securing Your Program

Pressing [SECURE] in the [INSTRMT BASIC] [MORE] menu brings up a menu allowing you to protect program lines so that they cannot be viewed with the [LIST] command.

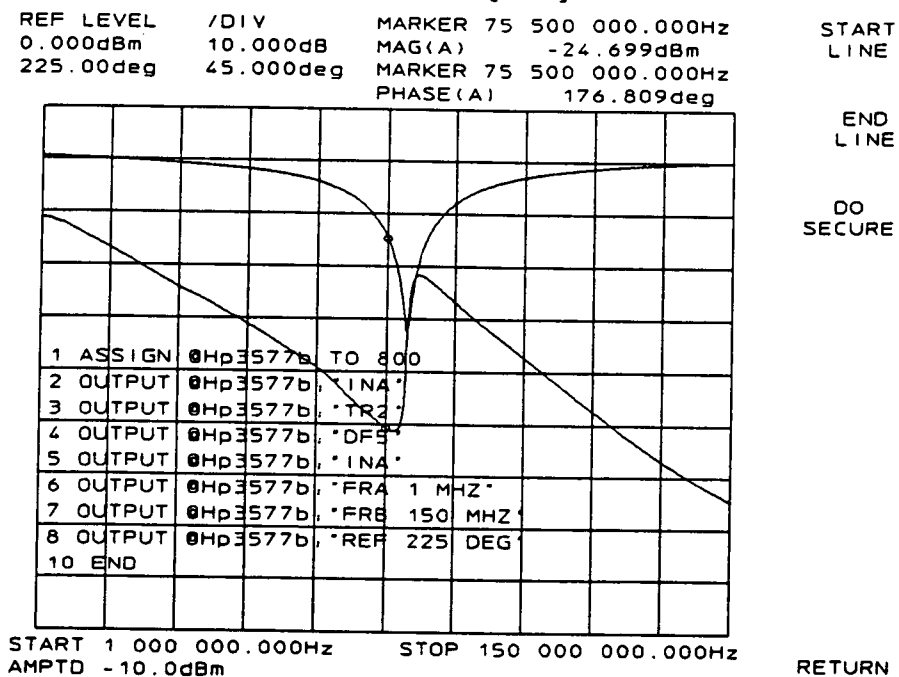


Figure 2-6. The [SECURE] Menu

You can secure all or part of a program by selecting line numbers encompassing the whole program or a block of program lines. Select the desired block of lines in the same way as described in the section titled "How to View a Program," earlier in this chapter. Then press [DO SECURE] to secure the selected block of program lines. You may exit the [SECURE] menu at any time by pressing [RETURN] before pressing [DO SECURE].

When you list a program with secured lines the secured line numbers will appear followed by an asterisk.

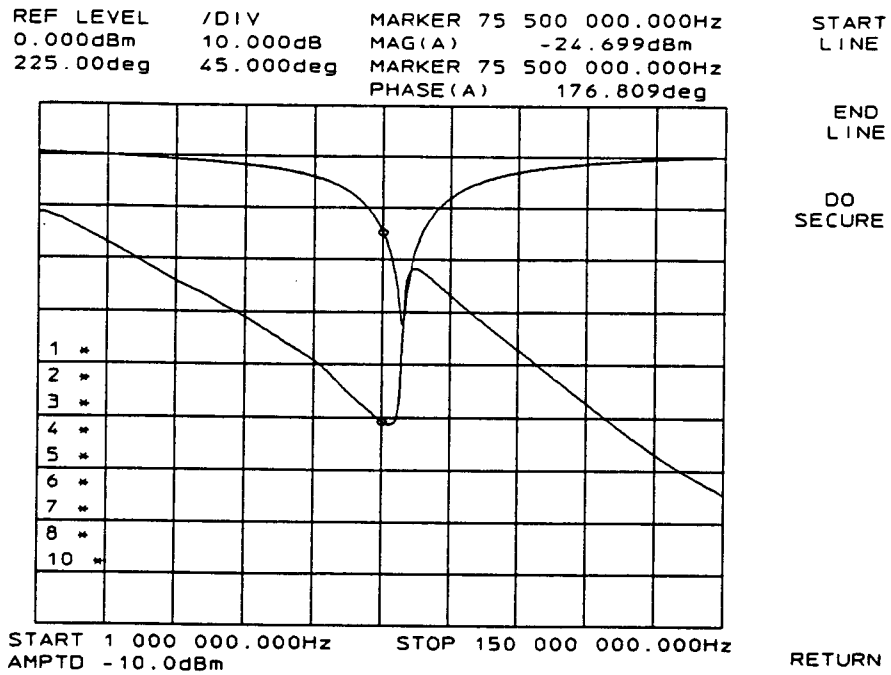


Figure 2-7. A secured program

Caution



Once secured, a program cannot be unsecured. Make sure the program is saved in separate file before securing it.

Recording and Running Programs
 Other Available Front-Panel Features
Examining and Allocating Memory

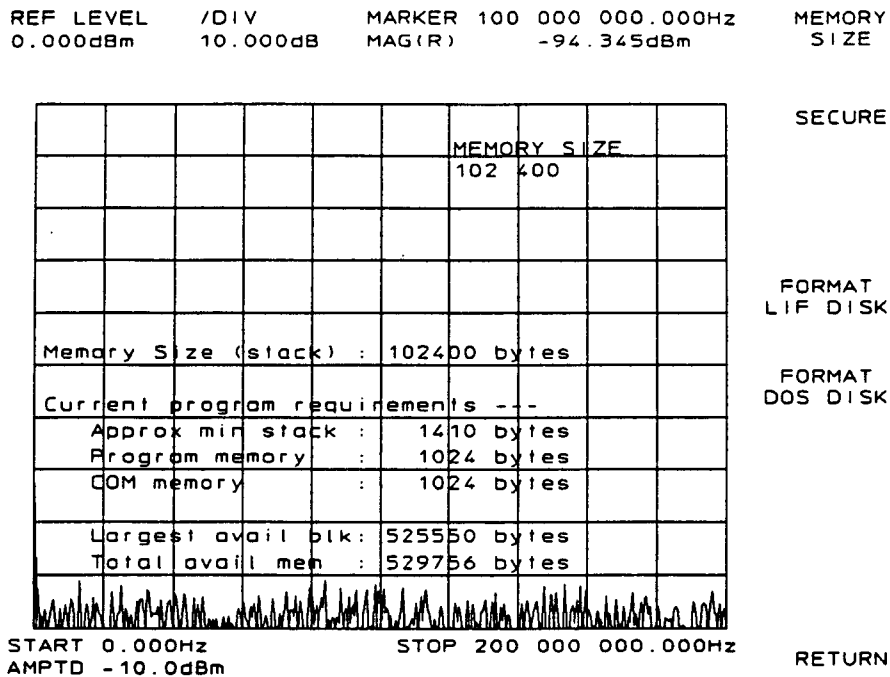


Figure 2-8. The [MEMORY SIZE] Menu

Pressing [MEMORY SIZE] in the [INSTRMNT BASIC][MORE] menu allows you to check and change available memory. HP Instrument BASIC programs require memory (RAM) for three reasons:

- A program “stack” for return addresses and variables not in COM.
- The program code.
- Variables defined by COM statements.

You can change the amount of memory allocated to the “stack” by pressing [MEMORY SIZE] and entering a new value. The other two uses of memory are automatically allocated before a program starts running.

If the stack is too small, an “out of memory” error may occur while a program is running. If the stack is too large, you may not be able to load a large program.

Memory Size (stack) indicates the amount of working space allocated for use by the program and HP Instrument BASIC overhead. Minimum size is 1152 bytes. Attempting to execute a program that is too large for the allocated memory will result in a memory overflow error (error 2). You can change the memory size by using the up and down arrow keys, or the keypad and [ENTER] softkey to enter a new memory size. Memory size will be allocated to the next largest multiple of 128. An approximation of the memory space required for the current program may be estimated from "Approx. min stack," which is explained below.

Approx. min stack reflects the amount of stack size which will be required by the current program if all local variables in the main program and subprograms, but not those in COM, are allocated simultaneously. This total amount may never be required during program execution if all local variables are not allocated simultaneously. Recursive subprograms may require more space than indicated here.

Program memory is automatically allocated to reflect the memory space required by the current program code.

COM memory is automatically allocated to reflect the memory space required for the COM variables in the current program.

Largest Available Block indicates the largest memory segment which can be allocated to stack memory. Previous memory allocation may cause available memory to be segmented and, therefore, not available for allocation as a block.

Available memory reflects the total unused memory space. Available memory decreases relative to increased stack memory size.

Recording and Running Programs
 Other Available Front-Panel Features
Initializing Disks

You can initialize disks in an external disk drive to prepare them for program storage. See chapter 3, "Saving and Recalling Programs," for information on connecting a disk drive.

To initialize a disk press [SPCL FCTN][INSTRMNT BASIC][MORE] then [FORMAT LIF DISK] or [FORMAT DOS DISK]. The following menu will appear:

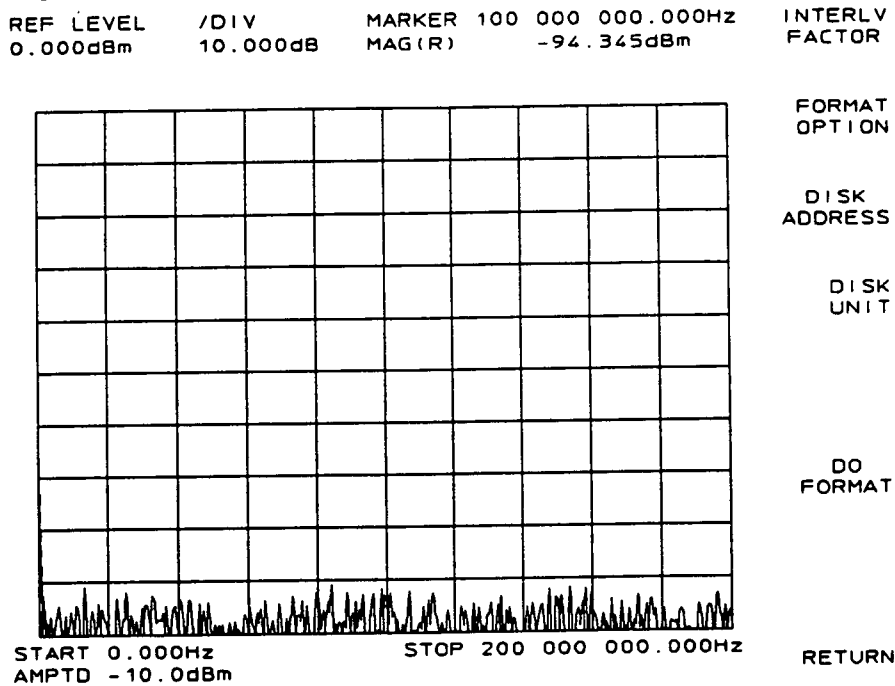


Figure 2-9. The [FORMAT DISK] Menu

Use the softkeys and keypad to select interleave and format options consistent with your disk drive operating manual. Select the disk drive address and disk unit number which contains the disk you want to initialize. Next press [DO FORMAT] to start the initialize operation.

Note



If you are formatting a DOS disk, the correct FORMAT OPTION must be specified, otherwise the disk cannot be read by other DOS systems. See "Using the DOS File System," in chapter 5 for more information.

General Information about Recorded Programs

What's in a Recorded Program

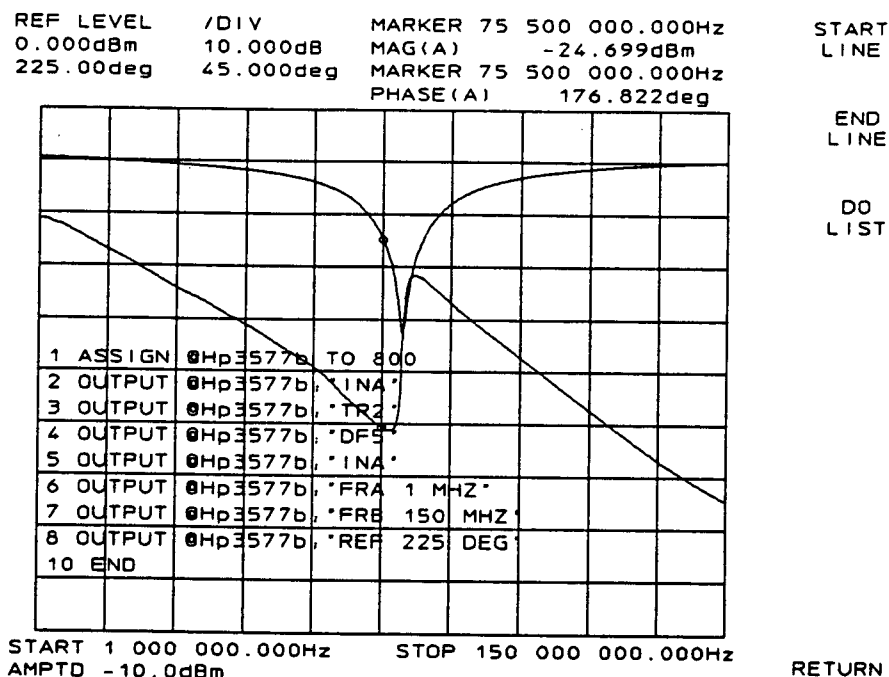


Figure 2-10. Listing of a Recorded Program

If you look at any program created using recording you will find that it is composed of three fundamental HP Instrument BASIC statements: ASSIGN, OUTPUT, and END. The program recorded in Exercise 2.1 demonstrates these statements:

```

1 ASSIGN @Hp3577b TO 800
2 OUTPUT @Hp3577b;"INA"
.
.
10 END
    
```

There will be only one ASSIGN statement at the beginning of the program and one END statement at the end. These statements will be automatically added to the program by HP Instrument BASIC unless you are recording into an existing program. In a typical program there will be many OUTPUT statements between the ASSIGN and END statements. Since the OUTPUT statement does the actual work of controlling the HP 3577B, let's take a closer look at how it is used.

Recording and Running Programs
General Information about Recorded Programs
The OUTPUT Statement

The HP Instrument BASIC statement:

```
OUTPUT <destination>;<data>
```

essentially tells the internal computer to send some information (data) to a device at specific address (destination). The destination can be a device selector (a number), or a name representing a number, called a path name. The data can take several forms but in recorded HP Instrument BASIC programs it is a string containing instructions to the instrument.

Although the OUTPUT command is very flexible it is used only one way when generated by a recording. The following represents a typical OUTPUT command from a recording session:

```
2 OUTPUT @Hp3577b;"INA"
```

Notice that the OUTPUT command is followed by a name representing a device selector (@Hp3577b), followed by a semicolon and the data ("INA").

The ASSIGN Statement

The destination in an OUTPUT statement specifies the address of the device. In recorded programs this address is represented by the I/O path name "@Hp3577b." The following line appears in all recorded programs before any OUTPUT statements:

```
ASSIGN @Hp3577b TO 800
```

The ASSIGN statement allows you to substitute an I/O path name (a variable preceded by the @ symbol) for a device selector number. Therefore, our example program line is equivalent to:

```
2 OUTPUT 800;"INA"
```

The device selector 800 specifies the host instrument as the destination of any data sent by the OUTPUT command. The program communicates with the HP 3577B via select code 8, the internal HP-IB interface, which is used solely for communication between HP Instrument BASIC programs and the analyzer. The analyzer will respond to any address on the internal interface from 800 to 899 (800 is typically used).

The HP-IB Mnemonics

The data sent to the HP 3577B by the OUTPUT command is called an HP-IB mnemonic and is found in quotes following the device selector path name and semicolon:

```
2 OUTPUT @3577b;"INA"
```

The HP-IB mnemonics used in HP Instrument BASIC are the same ones used to control the instrument remotely via an external computer. External computers communicate with the HP 3577B over the external HP-IB bus while HP Instrument BASIC programs communicate with it over the internal bus. In our example, the mnemonic "INA" tells the instrument to set the display format to INPUT A.

For more information on HP-IB interfacing using HP Instrument BASIC, refer to the "HP Instrument BASIC Interfacing Techniques" section of the *HP Instrument BASIC Users Handbook*). Additional commands specific to the HP 3577B are documented in Appendix A of this manual and the *HP 3577B Operating Manual*.

Rules of Recording

How Recording Works

To fully understand HP Instrument BASIC recording, it is important to understand the relationship between front-panel instrument operation and the program that is generated to emulate that operation.

Note



HP-IB mnemonics entered in a program during a recording session do not necessarily have a one-to-one correlation with the actual keys that are pressed during that session.

The fact that HP-IB mnemonics do not exactly correspond to the keys actually pressed is important to remember. As you press a sequence of keys to perform an operation, the corresponding HP-IB mnemonic for that operation is generated. The operation may take one keystroke or several, but the mnemonic is not generated until after a valid sequence of keystrokes is completed.

In other words, it is the functional operation of the instrument that is recorded as a mnemonic, not the keystrokes that it takes to perform that operation.

For example, when recording our simple example in Exercise 2.1, the key sequence [**FREQ**] [**STOP FREQ**] 150 [**MHz**] required six keystrokes and generated one mnemonic, "FRB 150 MHZ", after the sequence was completed. This was then automatically formatted into the command:

```
OUTPUT @Hp3577B;"FRB 150 MHZ"
```

and inserted into the program.

This means that if you accidentally press the wrong key in a sequence, it may not show up in the recorded program. It also means that you cannot exactly mimic keystrokes to leave the instrument in a specific front-panel state, unless of course it is a state that appears as a natural consequence of a completed operation.

For example, pressing [**SCALE**] in a recording session has the effect of bringing up the [**SCALE**] menu but does not, by itself, generate a program line. You could not therefore leave the instrument with the [**SCALE**] menu displayed.

Operations That Do Not Record

Although keystroke recording works automatically in most situations, there are some operations that cannot be captured or can only partially be captured using this method. These generally fall into one of the following areas:

- Front-panel operations with no corresponding HP-IB mnemonic (for example the [**LCL**] key)
- HP Instrument BASIC front-panel operations; that is, any front-panel operation added to the instrument by the HP Instrument BASIC option
- Operations requiring extra programming, such detecting SRQ conditions
- HP-IB operations with no front-panel equivalence (for example, HP-IB query commands)

Operations That Do Not Generate Mnemonics

Front-panel operations that do not have corresponding HP-IB mnemonics cannot be recorded. These are described in this section.

The [**LCL**] hardkey cannot be recorded, although functions in the [**LCL**] menu can be recorded.

The continuous entry key (the unmarked key above the knob) cannot be recorded.

Any parameter set using the knob will not be recorded. You must always use the numeric key pad to specifically enter numeric values.

The [**BACK SPACE**] key has no corresponding mnemonic and cannot be recorded.

The [**SYS CTRL ON OFF**] softkey cannot be recorded.. There are a few operations that require the analyzer to be the active controller on the external bus. This generally means that the analyzer must be set as System Controller before the program is run, or active control must be passed to it from an external controller.

Recording and Running Programs
General Information about Recorded Programs
HP Instrument BASIC Operations

Operations relating to HP Instrument BASIC itself cannot be keystroke recorded. Softkeys under [SPCL FCTN][INSTRMNT BASIC] cannot be recorded since that key sequence turns keystroke recording off. It is also true that the menus added to the [SAVE] and [RECALL] menus by HP Instrument BASIC do not record. You can, however, record all other save and recall operations not having to do with HP Instrument BASIC programs.

In addition the [RUN PROGRAM] and [CONTINUE PROGRAM] softkeys under [SPCL FCTN] cannot be keystroke recorded.

If you attempt to record an invalid keystroke the message "KEY IGNORED: Recording ON " will appear and the keystroke will be ignored.

Although HP Instrument BASIC operations cannot be recorded, many do have corresponding HP-IB mnemonics that allow an external controller to control and communicate with the internal HP Instrument BASIC program. For more information on these refer to "HP Instrument BASIC Interfacing Techniques" in the *HP Instrument BASIC Users Handbook*. Also see Appendix A, "Extended HP-IB Command Reference" in this manual.

Operations Requiring Additional Programming

Some operations that work well when performed from the front-panel have special circumstances that require attention when they are operated from within a program. This mostly refers to measurements that require synchronization with the program. For example, say you wish to have the measurement perform a single sweep and then detect when the sweep has finished. You will need to set the trigger mode to line trigger, sweep mode to single and then press [TRIG / RESET]. These steps can be recorded. However, to detect when the instrument is waiting for trigger (has completed a sweep), you will need to edit the program and setup a service request routine to detect the proper condition.

Mnemonics With No Corresponding Front-Panel Operation

Several of the available HP-IB mnemonics for the instrument perform operations that are not available from the front-panel and therefore cannot be recorded. These include operations such as querying instrument status, transferring data over HP-IB, setting and clearing status registers and general HP-IB housekeeping, to mention a few.

These operations are useful for the more advanced HP-IB programming using HP Instrument BASIC, however because they fall outside the direct operating realm of the HP 3577B, they cannot be recorded. (See the "HP Instrument BASIC Language Reference" section of the *HP Instrument BASIC User's Handbook*.)

Avoiding Recording Errors

By Using Preset

In most cases, you should perform a preset before recording a sequence and again before running the recorded program, or record an instrument preset as the first keystroke of the recorded program. This sets the instrument to its default state and avoids the risk of encountering a setting that defaulted to one setting when recorded but has been changed to another prior to running the program. For cases where a preset is undesirable see the following section, "Specifically Selecting Parameters."

You can perform a preset automatically in your program by pressing PRESET immediately after turning recording on. This inserts the following line prior to all other HP-IB OUTPUT statements in your program:

```
OUTPUT @3577b;"IPR"
```

By Specifically Selecting Parameters

If you do not want the instrument reset before a recorded program is run (for example, you may be recording a section of a larger measurement sequence), be sure to specifically activate every instrument setting that you will need in your automated sequence. For example, if you want the format to be LOG MAG, press [DISPLAY FCTN] and then [LOG MAG] even though LOG MAG is already the default setting. This will generate a program line to specifically set the format to LOG MAG.

Saving and Recalling Programs

Overview

An HP Instrument BASIC program can reside in memory, where it can be run repeatedly until it is scratched by the operator or the instrument is powered down. If you want to save programs for future use you can store them directly to an external disk. In this chapter we will show you how to:

- Connect an external disk
- Save programs from HP Instrument BASIC memory to disk
- Recall programs from disk to HP Instrument BASIC memory
- Autoload a self-starting program

Using an External Disk

HP Instrument BASIC can communicate with any CS-80 compatible disk drive. The drive must be single volume. Valid disk units numbers are 0 through 14.

Initialize, or "format," a disk by using a computer and following the operating instructions provided with the disk drive. Or you can initialize a disk using the HP 3577B front-panel function by following the procedure outlined in the section "Initializing Disks" in chapter 2. Disks must be formatted as either LIF or DOS disks. For storing and recalling keystroke recorded programs from the analyzer, it makes little difference whether the disk is formatted as LIF or DOS. If you plan on using the disk to transfer programs between the analyzer and an external computer for editing, you must select the proper format. LIF-formatted disks are used by HP BASIC controllers and DOS-formatted disks are used by MS-DOS computers. For more information, see chapter 4, "Developing Programs Using an External Computer."

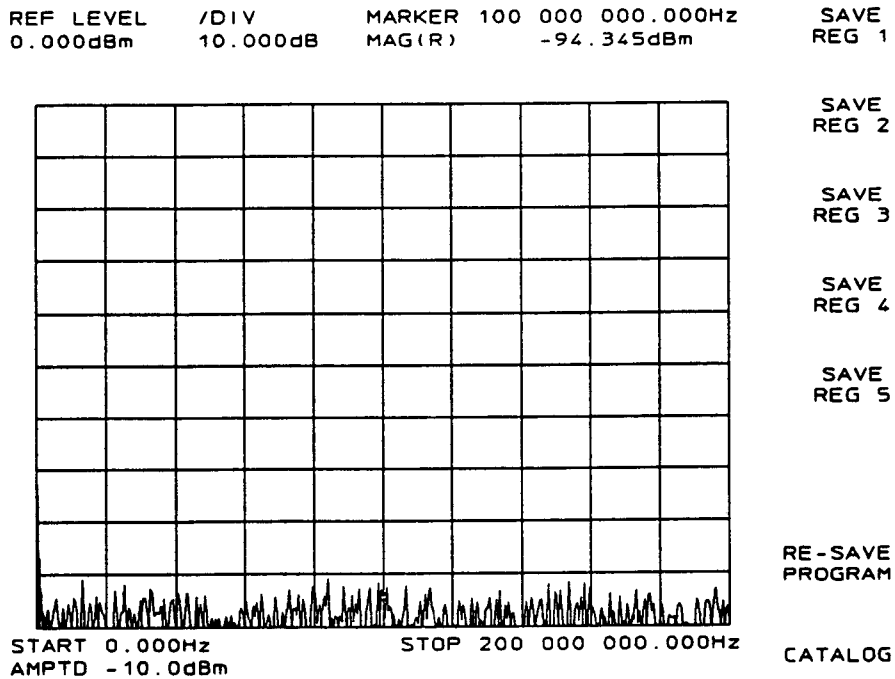


Figure 3-1. The [SAVE] Menu

Connect the disk drive to the HP 3577B through the HP-IB port and specify the drive's address and disk unit in the HP 3577B's [LCL] menu to match the disc drive's address switches. Refer to the disk drive's operating manual for information relating specifically to your drive. Disc functions performed by the HP 3577B require that the instrument be the active controller. Active control is selected in the [LCL] menu by pressing [SYS CTLR ON OFF].

Note



If you have an external controller connected to the analyzer, you must either disconnect it or pass active control of the external interface to the analyzer before continuing. See chapter 6 for more information on interfacing with external controllers.

Saving a Program to a Disk

The following exercise illustrates a typical procedure for saving a program using the [SAVE] menu. For this example it is assumed that you have finished recording a program into the HP 3577B HP Instrument BASIC memory and that the disk address and unit have been set.

Exercise 3.1 Using the [SAVE] Hardkey

1. Insert an unprotected formatted disk in the drive.
2. Press [SAVE]

You will see the following menu:

3. Press [RE-SAVE PROGRAM]

The following menu will appear:

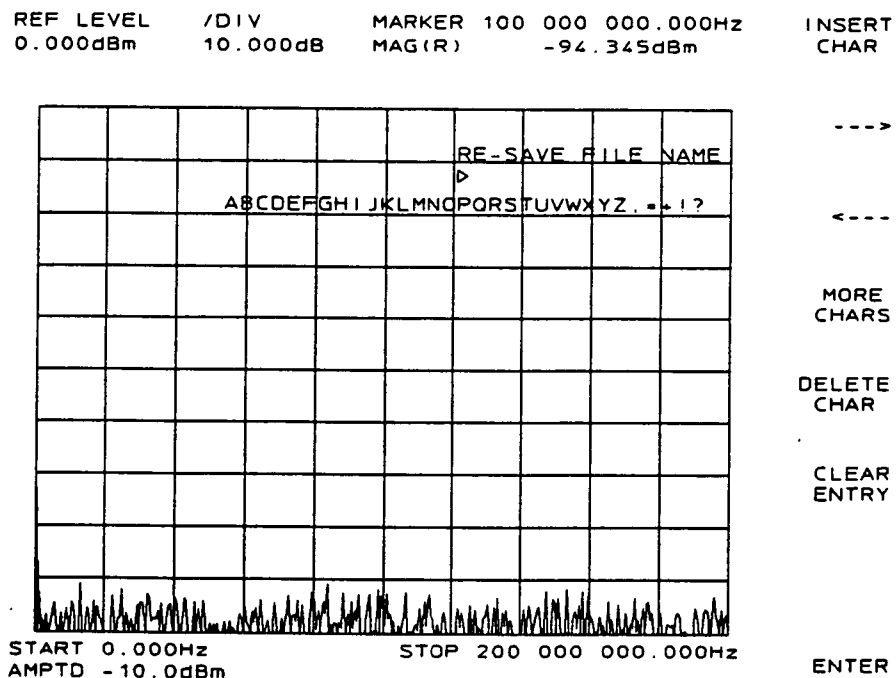


Figure 3-2. The [RE-SAVE] Menu

4. To enter a file name, use the softkeys in conjunction with the knob (or up and down arrow keys) and the numeric keypad to select the alpha characters for your file name. For this example we will use "Prog1."
 - a. Use the knob (or up arrow key) to scroll right until "P" is highlighted and press [INSERT CHAR]. Characters you insert will appear in the display line.
 - b. To change to lower case press [MORE CHARS] and use the knob to scroll to "r" and press [INSERT CHAR].
 - c. Use the knob and [INSERT CHAR] to select "o" and "g."
 - d. To enter the number "1," just press "1" on the numeric keypad.
 - e. Press the [ENTER] softkey to terminate the entry and perform the disk function.
"RESAVE program in progress" will be displayed until program storage is completed.

Special Notes on Disk Storage

LIF file names are limited to ten characters in length. DOS file names are limited to eight characters plus a 3 character extension. If the DOS-formatted disk contains directories, you may specify the directory in the name of the file by separating the directory or directories by backward ("\") or forward ("/") slashes. For example, to save the file "PROG1" under the directory "/MYDIR," enter "/MYDIR/PROG1". Also, for both LIF and DOS disks, disk drives at addresses other than those specified in the [LCL] menu may be accessed by appending the mass storage volume specifier. For example, to save a file named "Prog1" to a LIF disk in drive 1 of a drive at address 700, you could enter: "Prog1:,700,1".

To alter the file name before pressing [ENTER] use the arrow softkeys to move the highlight to the desired position and use [INSERT CHAR] or [DELETE CHAR] to modify the file name.

If you want to leave the [RE-SAVE PROGRAM] menu without storing a program, you may press [CLEAR ENTRY] and [ENTER]. You may also press [ABORT] while the disk operation is in progress but incomplete.

If the knob does not scroll on the displayed characters, check to see that the unmarked key above the knob is selecting ENTRY.

If you press the [MORE CHARS] key again after selecting the lower case selections, a symbol menu will appear. If you enter an underscore in your program name, it appears on the display line as a left arrow (←) but is interpreted by HP Instrument BASIC as an underscore.

Saving and Recalling Programs

Saving a Program to a Disk

Press [CATALOG] in the [SAVE] menu to catalog the external disk to the HP Instrument BASIC display area. The catalog function causes the display to scroll through all the disk entries and only the last ten disk entries remain in view. If you have more than ten entries on the disk and want to view those recorded prior to the last ten, press [ABORT] during the catalog function to halt the display at any point. Otherwise, you may choose to catalog the disk to an external printer by selecting the external printer as described in "Printing and Viewing a Program" in chapter 2. Subsequent [CATALOG] key presses catalog the selected disk to the printer.

If a program exists on the disk it will be overwritten by the [RE-SAVE PROGRAM] operation and will maintain its file type. For example, an existing ASCII file on a DOS disk remains ASCII but if no file exists, the program is saved as an untyped file, which is editable on an MS-DOS computer.

The HP 3577B front-panel disk functions do not include purging or renaming files. These functions must be performed from an external controller or a program (see the example program "MSE" in appendix B).

Caution



The RE-SAVE operation will overwrite an existing file with the same name and there will be no warning to indicate that a file with the same name exists.

Recalling a Program from a Disk

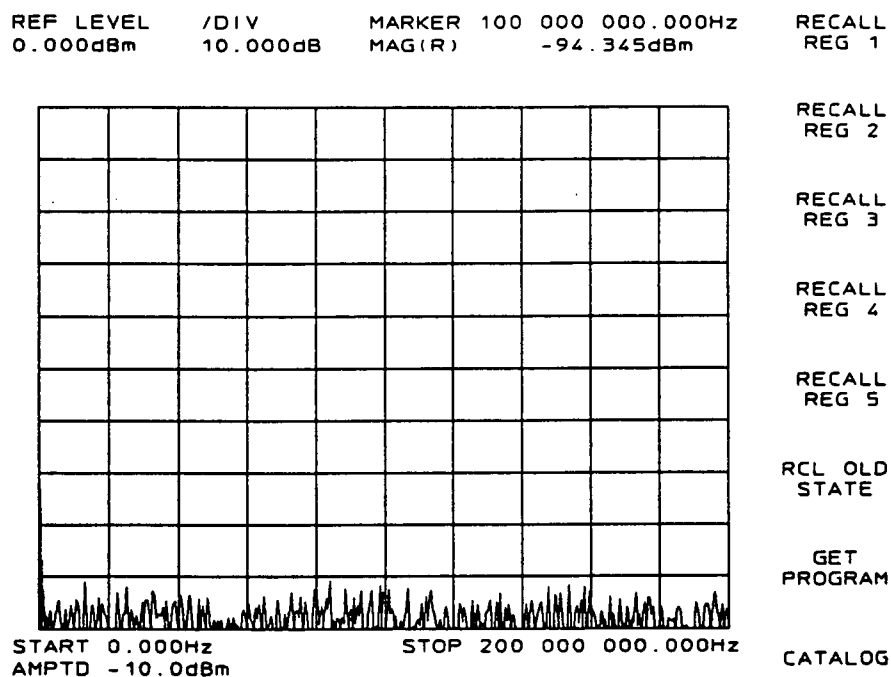


Figure 3-3. The [RECALL] Menu

To load an HP Instrument BASIC program from an external disk you can use the [RECALL] menu. Exercise 3.2 demonstrates the steps used to recall a file. This exercise assumes that you have performed Exercise 3.1 so that a program named "Prog1" is on the external disk drive.

Exercise 3.2 Recalling a Program from Disk

1. Press the [INSTR PRESET] key to return the instrument to its default state
2. Press the [RECALL] key
3. Press [CATALOG] to see what programs are on your disk
4. Press [GET PROGRAM]
5. Enter the file name, "Prog1," in the same way as you entered the file name in Exercise 3.1
6. Press [ENTER] to recall the program from disk

Autoloading a Program

HP Instrument BASIC allows you to designate a program to be automatically loaded and run when the instrument is first powered up. This requires that a disk drive be connected to the HP 3577B and that a disk containing an autoloading program be inserted in the drive.

To make an autoloading program, save it on the disk with the file name "AUTO_BAS". Enable autoloading in the HP 3577B by pressing [AUTOLOAD ON OFF] in the [INSTRMNT BASIC] menu. Check to see that the disk drive's address and unit number are selected in the [LCL] menu at powerdown. Also be sure that System Controller is ON. At the subsequent powerup HP Instrument BASIC will search the designated disk for a file named "AUTO_BAS." If found, it will be loaded and run.

Note that if the disk in the external drive is formatted as DOS, the "AUTO_BAS" program must be at the root directory to be found at powerup.

Developing Programs Using an External Computer

Introduction

For many applications, you can easily record and run programs without any need to alter the program code that is generated. However, with some knowledge of the HP Instrument BASIC language and a computer you can add immeasurable power to your recorded programs or create your own programs from the ground up. This chapter shows you how to make the best use of an external computer to take full advantage HP Instrument BASIC in your HP 3577B.

There are two common types of computers that interface easily with HP Instrument BASIC programs: HP BASIC controllers and MS-DOS computers. HP BASIC controllers, such as the HP Series 200/300 workstations, can be used to edit or write HP Instrument BASIC programs using the HP BASIC language environment. Likewise, program files saved to, and recalled from, a DOS-formatted disk by the HP 3577B are fully editable in the MS-DOS environment. Therefore, MS-DOS computers can use any standard editor program to write or edit an HP Instrument BASIC file. However, a standard MS_DOS editor will not provide the syntax checking built into HP BASIC.

A third type of editing environment is somewhat of a hybrid of these two. For example, HP BASIC Language Processor is an HP BASIC controller on a card that resides in an MS-DOS computer and uses its file system.

This chapter describes how to transfer programs between the HP 3577B and all of these external computer environments for the purpose of program development.

Using the HP BASIC Controller

Overview

Using a computer that runs HP Series 200/300 BASIC, you can easily edit or write your own HP Instrument BASIC programs and then transfer the programs into the analyzer. Likewise, you can take advantage of the keystroke recording capabilities of HP Instrument BASIC to record HP-IB sequences then transfer those programs to an external computer for further program development.

Once the HP Instrument BASIC program is in the HP BASIC editor, you can write and edit it the same way as you would alter any BASIC program. Refer to the *HP Instrument BASIC Users Handbook* for guidance in this area. Also see chapter 3, "Remote Programming" in the *HP 3577B Network Analyzer Operating Manual* for an understanding of the mnemonics used in programming the HP 3577B. Programs edited or developed in this way may be downloaded to the HP 3577B or stored on an external disk for direct access by the HP 3577B. These programs can then be executed by HP Instrument BASIC in the HP 3577B at any time without the use of an external controller.

For more complex programs and systems you may use programs running in both the external controller and the HP 3577B (see "Interfacing with an External Controller" in chapter 6).

Transferring Programs Via Disk

To transfer a program file to or from the HP 3577B you must have an external disk drive connect to the analyzer. **When you are transferring files to be used in an HP BASIC editor, the disk must be a LIF-formatted disk.** To transfer a keystroke recorded program generated in the HP 3577B, simply save the program using the procedure described in the section "Saving a Program to a Disk," in chapter 3.

HP BASIC Editor

Program files saved to a LIF-formatted disk are saved as ASCII files and may be recalled from an HP BASIC editor using the GET statement. After editing or writing an HP Instrument BASIC program in the HP BASIC editor, SAVE or RE-SAVE the program onto a disk (do not "STORE" the program, as this creates a PROG type file). Then transfer the file to the external disk drive connected to the HP 3577B and recall the program (see "Recalling a Program from a Disk" in chapter 3.)

HP BASIC Language Processor Editor

If you are using the HP BASIC Language Processor to edit programs and you have access to an external disk drive (for example, the external drive used by the HP 3577B to save and recall programs), you can use the preceding method to transfer program files. Or, you can upload or download programs over HP-IB as described next. If you want to use a DOS-formatted disk to transfer programs between the HP 3577B and the HP BASIC Language Processor, refer to the section titled "Using an MS-DOS Computer" for more information.

Downloading and Uploading Programs

Transferring programs over the HP-IB is referred to as "uploading" or "downloading" programs. At the end of this section you will find program listings for "UPLOAD" and "DOWNLOAD" programs. Enter each of these programs in your HP BASIC editor and store them on a disk.

To transfer programs between HP Instrument BASIC and another computer, the computer must be connected to the HP 3577B via the HP-IB interface. Make sure that the HP 3577B is set to [SYS CTLR ON OFF] (in [LCL] hardkey menu). Also, make sure that either the local address of the analyzer is set to 11, or that you change the address in the upload or download program to match whatever address the HP 3577B is currently set to. After the program transfer is complete you may need to press the [LCL] key to return the HP 3577B to local control from remote.

Note



Remember that the external controller is System Controller when running either the upload or download programs. You must pass control to the HP 3577B (or disconnect the external controller), and set the HP 3577B to System Controller, before running any HP Instrument BASIC program that uses the external HP-IB interface.

If you want more information on the details of downloading and uploading programs, see the section "Downloading and Uploading Programs" at the end of chapter 6, "Interfacing with the HP-IB." For information on the HP-IB upload and download mnemonics refer to the Appendix A.

Developing Programs Using an External Computer Using the HP BASIC Controller

Uploading Programs

You can upload a program from the HP 3577B program memory to a file on your HP BASIC controller by loading and running the "UPLOAD" program in the external controller. The program prompts you for a local file name then transfers the contents of the HP 3577B program memory to an ASCII disk file. You can then GET the file and modify it using all applicable BASIC commands. See the *HP Instrument BASIC Users Handbook* for more information on program development. You can SAVE the modified program as an ASCII file and download it to the HP 3577B.

Downloading Programs

After writing a program using the HP BASIC editor on your external controller, SAVE it to disk as an ASCII file. Now load the "DOWNLOAD" program and run it. The program will prompt you for the name of the file that you want to download. After entering the name of the ASCII file you saved, the file will be transferred to the HP Instrument BASIC program memory.

The upload and download programs may leave the HP 3577B in remote mode so you will have to press [LCL] to regain front-panel control.

Downloading Incompatible Code

Each line of program code is syntax-checked by HP Instrument BASIC as it is download, the same as it would be if recalled into an HP BASIC editor. Attempting to download BASIC program lines that are not compatible with HP Instrument BASIC will result in "IBASIC syntax ERROR" being displayed on the HP 3577B screen. The program line will appear in the program listing but will be commented out (it will have a "!" before the line) so that the program will execute without this line.

If you attempt to download unnumbered program lines the error message "IBasic MISSING LINE#" will appear on the instrument display. Remaining program lines will be downloaded omitting the unnumbered line.

Attempting to download a null line will terminate the download.

Developing Programs Using an External Computer
Using the HP BASIC Controller

```

10 ! "UPLOAD"
20 ! BASIC program to transfer an Instrument BASIC program
30 ! from the HP 3577B to a file. This program is not intended
40 ! to be executed with HP Instrument BASIC.
50 !
60   Scode=7           !Select code for interface
70   Address=11       !Address of HP 3577B
80   Na=Scode*100+Address
90   DIM Line$(256)   !String to hold program lines
100  DIM Filename$(100) !Destination file name
110  File_size=1      !Destination file size in bytes
120  !
130  Get_filename: !
140    LINPUT "Enter file name: ",Filename$
150    !
160  Assign_file: !
170    ON ERROR GOTO File_error !set up error trapping.
180    ASSIGN @File TO Filename$;FORMAT ON !Try to open an existing file
190    !will error if it does not exist.
200    OFF ERROR
210    CLEAR Na
220    OUTPUT Na;"DPR" !Send "Dump program" code to the HP 3577B
230    ENTER Na USING "%,2A";Pound_i$ !Read the #I block header
240    Line_count=0
250    LOOP
260      ENTER Na;Line$ !Read a line from the HP 3577B
270      EXIT IF Line$="" !An empty line signals the end.
280      OUTPUT @File;Line$ !Write the line to the file
290      Line_count=Line_count+1
300      DISP "Lines transferred: ";Line_count
310    END LOOP
320    ASSIGN @File TO * !Close the file
330    LOCAL Na
340    DISP "Transfer complete"
350    STOP
360    !
370    !
380  File_error: !jumps here when an error occurs during ASSIGN @file
390    OFF ERROR
400    IF ERRN=56 THEN !error 56 is "file name undefined"
410      INPUT "Enter file size in bytes: ",File_size
420      CREATE ASCII Filename$,((File_size-1) DIV 256)+1
430    ELSE
440      PRINT ERRM$
450      GOTO Get_filename
460    END IF
470    GOTO Assign_file !jump back to file assign
480    !
490    END

```

Developing Programs Using an External Computer
Using the HP BASIC Controller

```

10      ! RE-STORE "DOWNLOAD"
20      !BASIC program:  DOWNLOAD -- Transfer BASIC program to HP 3577B
30      !
40      !This program demonstrates how to download an HP Instrument
50      !BASIC program to the HP 3577B.  This program is not intended
60      !to be executed with HP Instrument BASIC.
70      !
80      Scode=7                               !Select code for interface
90      Address=11                             !Address for HP 3577B
100     Na=Scode*100+Address
110     DIM Line$(255)                         !String to hold program lines
120     DIM Filename$(100)
130     CLEAR Na                               !Abort any HP-IB activity
140     !
150     Get_filename:  !
160     LINPUT "Program to be transfered?",Filename$
170     ON ERROR GOTO No_file                   !Trap errors opening file
180     DISP "Checking file..."
190     ASSIGN @Basic_prog TO Filename$        !Open the file
200     OFF ERROR                               !Disable error branching
210     !
220     OUTPUT Na;"SCA"                        !SCRATCH A to clear prog space
230     !
240     Transfer:  !Transfer the BASIC program to the HP 3577B
250     ! The next line demonstrates the data format for program transfers.
260     ! LPR #I10 GOTO 10<cr><lf>20 END<cr><lf><cr><lf>
270     ! The last line-feed character is sent with EOI asserted
280     !
290     ASSIGN @Prog TO Na
300     Line_count=0                           !Initialize variable Line_count
310     !
320     ON END @Basic_prog GOTO End_block      !Set up branching on end of file
330     OUTPUT @Prog;"LPR #I";                !Initiate data transfer
340     LOOP
350     ENTER @Basic_prog;Line$                !Read program line from file
360     OUTPUT @Prog;Line$                    !Send to HP 3577B
370     Line_count=Line_count+1
380     DISP "Lines transfered: ";Line_count
390     END LOOP                               !Loop until end of file (EOF)
400     End_block:  !Terminate the data block and close the file
410     OUTPUT @Prog;CHR$(10) END              !Output a line-feed with EOI
420     ASSIGN @Basic_prog TO *                !Close the file
430     !
440     DISP "Transfer complete"
450     LOCAL Na                               !Return HP 3577B to LOCAL mode
460     STOP                                   !Logical end of this program
470     !
480     No_file:  !error_handler for opening file
490     BEEP
500     DISP "CAN'T OPEN: """;Filename$;"" -- ";
510     GOTO Get_filename                      !Go back and get new filename
520     !
530     END

```

Using an MS-DOS Computer

Overview

Because the HP 3577B and HP Instrument BASIC both recognize the MS-DOS file system on an external disk, the task of editing program data in an MS-DOS computer has become extremely simplified. There are essentially four ways to edit HP Instrument BASIC programs on a PC:

- Transfer the program on a DOS-formatted disk and use an ASCII editor
- Transfer the program on a DOS-formatted disk and use the HP BASIC Language Processor editor
- Transfer the program on a LIF-formatted disk and use the HP BASIC Language Processor editor
- Transfer the program over HP-IB and use the HP BASIC Language Processor editor

The first two of these options are covered in this section. The last two have been described in the section "Using the HP BASIC controller," earlier in this chapter.

Transferring Programs Via Disk

To transfer a program file to or from the HP 3577B you must have an external disk drive connected to the analyzer. **When you are transferring files to be used in an MS-DOS editor, the disk must be a DOS-formatted disk.** To learn how to format a DOS disk from the analyzer, see "Disk I/O" in chapter 5, "Program I/O." As a general rule, it is often easier to format the disk on the external PC first.

MS-DOS Editors

To transfer a keystroke recorded program generated in the HP 3577B, simply save the program using the procedure described in the section "Saving a Program to a Disk," in chapter 3. Program files saved to a DOS-formatted disk are saved as "untyped" DOS files and may be edited by any standard MS-DOS editor. Although this is generally referred to as an "ASCII" editor in the MS-DOS environment, do not confuse this with the ASCII file type used to transfer files between the analyzer and an HP BASIC controller.

After editing or writing an HP Instrument BASIC program in the MS-DOS editor, resave the file. Then transfer the file to the external disk drive connected to the HP 3577B and recall the program (see "Recalling Programs from a Disk" in chapter 3.)

Note that, if you save a program under a different directory than the root, you must provide the full pathname when recalling the program into the HP 3577B.

Developing Programs Using an External Computer Using an MS-DOS Computer

Also, remember that if you try to use the HP 3577B [CATALOG] softkey to view the program file entry on the DOS disk, the HP 3577B will only catalog one level of DOS directories. This is usually the root of the directory tree, since there is no front-panel provision to change the MSI (MASS STORAGE IS) specification to a subdirectory. Therefore, the file will not appear in a catalog of the root directory, if it was saved in a subdirectory.

The only way to change the MSI to a different directory on a DOS disk is to run an HP Instrument BASIC program that changes the MSI. In this case, the HP 3577B will catalog the directory specified in the last MSI statement.

HP BASIC Language Processor Editor

If you are using the HP BASIC Language Processor to edit programs and wish to use the DOS file transfer method just described, you must SAVE or RE-SAVE the program file to the DOS file system. The HP BASIC Language Processor SAVE and RE-SAVE commands create an HP BASIC ASCII file on the DOS medium, which cannot easily be edited by an MS-DOS editor. However, the HP 3577B recognizes this file type and will bring it in as a program file from the DOS disk. If the program is later re-saved to that file, it will maintain the same file type rather than creating an "untyped" DOS file as it would if the file did not previously exist.

Program I/O

Introduction

This chapter describes how to write programs that make use of the CRT and the GPIO interface in the HP 3577B and how to understand and use the DOS file system on an external drive.

The HP Instrument BASIC environment can be viewed as a complete computer residing inside the HP 3577B. Like any computer, it requires I/O devices to communicate with the "outside world." Most computers require a CRT for displaying text and graphics, a keyboard for entering data, and a disk drive for saving and recalling data.

HP Instrument BASIC in the HP 3577B uses the analyzer's CRT for program output, and can take advantage of an external drive connected to the HP 3577B for saving and recalling data. Since there is no keyboard on the analyzer, HP Instrument BASIC commands (INPUT and ENTER) bring up the same text entry menu used to GET and RE-SAVE HP Instrument BASIC programs from the front panel. This menu is described in chapter 3, "Saving and Recalling Programs."

In addition to these "standard" I/O features, the HP 3577B contains a General Purpose I/O port, which can be programmed from an HP Instrument BASIC program using special HP-IB mnemonics.

HP Instrument BASIC programs recognize both LIF and DOS file systems on a disk drive connected to the HP-IB. The LIF file system is described in the "Data Storage and Retrieval" chapter of the *HP Instrument BASIC Programming Techniques* manual. The way that HP Instrument BASIC in the HP 3577B utilizes the DOS file system on an external disk is described in this chapter.

Graphics and Display Techniques

Overview

HP Instrument BASIC allows you to customize the display for your program needs. The programmable display features include softkeys, graphics, and text. This section will show you how to:

- Set up and label softkeys for user interaction and program branching during program execution
- Use graphic commands PEN, MOVE and DRAW to use the screen for graphic display
- Use the PRINT and DISP commands to write to the screen

Setting Up and Using Softkeys

HP Instrument BASIC allows you to define the eight softkeys on the right side of the display for your own programming needs. The ON KEY statement sets up softkeys with labels and specifies the branch to be taken when that softkey is pressed. For example, the statement:

```
150 ON KEY 1 LABEL "PLOT IT" GOSUB Plot
```

displays the label [PLOT IT] next to softkey number 1 and specifies a branch to the subroutine at program label "Plot:". As with all on interrupt statements, once the ON KEY statements are executed, they stay in effect until the program stops or an OFF KEY statement is executed.

The Waiting Loop

After setting up the ON KEY statements, a common technique is to enter an infinite loop where the program waits for one of the softkeys to be pressed. This is an example of a waiting loop:

```
200 Waiting: GOTO Waiting
```

With the ON KEY statements in effect, an interrupt from a pressed softkey will cause the program to break the loop and branch to the label, line number or subprogram specified in the ON KEY statement. In the case of the ON KEY...GOSUB statement, the label begins a subroutine and a RETURN statement in a subroutine returns to the waiting loop. In the case of an ON KEY...GOTO statement, the program does not return to the waiting loop, but continues to its termination. For example:

```
160 ON KEY 9 LABEL "QUIT" GOTO Quit
```

If you are writing a program that idles in a loop while the instrument is busy, add a WAIT statement to the loop to make sure the program doesn't steal too much CPU time from the instrument. For example, the waiting loop above could be expanded to include a half-second wait on each iteration as follows:

```
200 Waiting:WAIT .5  
210 GOTO Waiting
```

Drawing Graphics

The entire screen can be used for graphics display. The MOVE and DRAW commands can be used to draw vectors anywhere on the display. Coordinates range from 0,0 (lower left corner) to 2047,2047 (upper right corner). Vectors drawn to points beyond the coordinate range will be truncated. Display memory is limited to 460 MOVEs and DRAWs.

There are three graphics commands available to you:

MOVE <x>,<y>

This command moves the pen to the point at "x" units on the horizontal axis and "y" units on the vertical axis.

DRAW <x>,<y>

This command draws a line from the current pen location to the point at "x" units on the horizontal axis and "y" units on the vertical axis.

As an example using MOVE and DRAW, the following program lines will draw a box around the outside of the screen:

```
500 MOVE 1,1
510 DRAW 1,2047
520 DRAW 2047,2047
530 DRAW 2047,1
```

PEN <number>

Selecting a pen number ≥ 1 will cause DRAW commands to display vectors. Selecting a pen number ≤ 0 is used to remove a previously drawn vector from the display. Because the HP 3577B display uses a vector display, all DRAW commands are stored in a vector table. To remove a vector you must issue the command "PEN 0" and then issue the exact DRAW statement that was used to initially draw the vector. For example, the following program segment will draw a line from 0,0 to 100,100:

```
100 PEN 1
110 MOVE 0,0
120 DRAW 100,100
```

To erase that line use:

```
200 PEN 0
210 MOVE 0,0
220 DRAW 100,100
```

Program I/O
Graphics and Display Techniques
Printing to the Screen

HP Instrument BASIC PRINT statements use a 10 line by 46 character area in the lower left portion of the graticule (the same area used by the [INSTRMNT BASIC][LIST] command). OUTPUT statements to the CRT use the same area.

The DISP statement prints to a one line area just to the right and above the PRINT area. The display line can display up to 17 characters. Prompts for INPUT statements appear in the same area and allow a string to be constructed on this line. See the alpha entry example in Exercise 3.1 to learn how to enter letters, numbers, and symbols.

The following program segment demonstrates the use of both the print and display areas.

```
600 PRINT "In order to continue testing"  
610 PRINT "place the next part in the"  
620 PRINT "test fixture and enter the"  
630 PRINT "part number."  
640 INPUT "Part number ",Part_no$
```

Clearing the Display

HP Instrument BASIC and the instrument share the screen, so the displays for both will appear simultaneously. The HP Instrument BASIC display, which includes program listing, print output, display line and graphics, can be cleared from the front panel by pressing [SPCL FCTN] [INSTRMNT BASIC][CLEAR DISPLAY]. You can also press [INSTRMNT PRESET] to clear the display and reset the analyzer.

To completely clear the instrument display you must send the instrument HP-IB commands to turn off traces (DF0), graticule (GR0), characters (CH0), etc. For more information refer to chapter 3, "Remote Operation" in the *HP 3577B Operating Manual*.

Interfacing with the General Purpose Input/Output Port

HP Instrument BASIC provides control for the 8-bit input/output port, which communicates with certain peripheral devices. This section describes how to communicate with the interface using HP Instrument BASIC.

Overview

There are two methods of transferring data between an HP Instrument BASIC program and the I/O port. The port has a separate select code (12) at which data can be sent and received. There are also HP-IB commands (LIO and DIO) that transfer values using the instrument address (800). This section describes how to use each type of command to input and output data.

8-Bit Input

The inputs can be read directly from the GPIO select code:

```
ENTER 12 USING "#,B";Value
```

The inputs can also be read using the HP-IB command "DIO":

```
OUTPUT 800;"DIO"  
ENTER 800;Value
```

8-Bit Output

Values can be output directly to the GPIO select code:

```
OUTPUT 12 USING "#,B";Value
```

The outputs can also be set using the HP-IB command "LIO":

```
OUTPUT 800;"LIO 5"
```

Using the DOS file System

Disk Formats and File Systems

HP Instrument BASIC in the HP 3577B recognizes two disk formats: LIF (Logical Interchange Format), and DOS. The format of a disk (or file system) is determined when it is formatted (or initialized). A LIF disk contains only one directory, while a DOS disk can have a "hierarchical" structure of directories.

The LIF format should be used to exchange programs and data with other HP BASIC computers. The DOS format should be used to exchange programs and data with MS-DOS computers.

The *HP Instrument BASIC Users Handbook* refers to a third disk format, HFS (Hierarchical File System). The HP 3577B does not support HFS, however, the DOS file system is similar to the HFS.

Disk Volumes

HP Instrument BASIC in the HP 3577B does not have an internal disk drive, and does not allow Memory Volumes (RAM disks) to be created. HP Instrument BASIC does support external disk CS80 drives connected via HP-IB.

Formatting Disks

The disk format (LIF or DOS) is specified when the media is initialized. For example,

```
INITIALIZE "LIF: ,707,1",0,0
```

will initialize a LIF disk in unit 1 of the drive at HP-IB address 7, using default values for the Interleave Factor and Format Option.

To initialize a DOS disk in the same drive,

```
INITIALIZE "DOS: ,707,1",0,16
```

should be used. The format option, 16 in this example, is important when initializing a DOS disk. An incorrect format option will result in a disk that the HP 3577B can use, but other MS-DOS systems can not. **This potential problem can be avoided by formatting the disk on the MS-DOS system, rather than the HP 3577B.**

Table 5 1. External Drive Format Options

Drive Model	Format Options	
	LIF	DOS
HP 9122C	0	16
HP 9127A	0	2
other	see drive manual	

Once initialization is complete, "LIF" or "DOS" is not specified in other file operations. HP Instrument BASIC will automatically determine the format of the disk.

File Types

In addition to the two disk formats (LIF and DOS), HP Instrument BASIC in the HP 3577B supports 3 file types: ASCII, BDAT, and "untyped." Untyped files are referred to as either "DOS" or "HP-UX" files. "File type" is essentially independent of disk format (or file system): ASCII, BDAT and untyped files can exist on either LIF or DOS disks. Untyped files will appear as "HP-UX" in a catalog of a LIF disk, or as "DOS" in catalog of a DOS disk.

In HP Instrument BASIC, "CREATE ASCII" is used to create an ASCII file, "CREATE BDAT" to create a BDAT file, and "CREATE" to create an untyped file. See the "Data Storage and Retrieval" chapter of the *HP Instrument BASIC Programming Techniques* manual for more information on file types.

Unsupported file types will be printed in the CAT output using the file type number.

More About DOS

On DOS disks, file names must conform to DOS conventions. Files names are limited to 8 characters plus a three character extension. The file name is separated from the extension by a period. The HP 3577B does not use wild cards in file names, and DOS files names are case insensitive. For example, "PROGA.ASC" is the same file as "Proga.ASC."

When using the DOS disk format, the file specifier can include a directory path. For example,

```
ASSIGN @file TO "\DATA\TEST1\BEFORE"
```

opens a the file named "BEFORE" in the directory "TEST1," under the directory "DATA." Either a "\" or a "/" character can be used to separate directories and file names.

The MASS STORAGE IS command can be used to change the current directory. Note that when the program terminates, the directory specified in the last MASS STORAGE IS command of the program will be the default for any catalog operation from the analyzer's front panel.

The CREATE DIR will create a subdirectory on a DOS disk.

Because untyped files on DOS disks are extensible, they can be created with a size of 1, and they will "grow" to the size needed. See the CREATE command for more information.

Using a DOS disk to Exchange Data with a PC

Measurement data from the HP 3577B can be transferred to a personal computer by writing an HP Instrument BASIC program to output the data to a DOS disk. Here are some important things to remember:

- The disk must be formatted correctly so that both the PC and the external drive connected to the HP 3577B can use it. Either format the disk on the PC, or use the HP 3577B and the correct format option.
- Use the CREATE command (without "ASCII" or "BDAT") to create an untyped DOS file. ASCII and BDAT usually cannot be read by other DOS programs because they contain an additional 512-byte header and an internal data structure.
- Use the ASSIGN command with the FORMAT ON option to open the file. FORMAT ON directs HP Instrument BASIC to store the data as ASCII characters. FORMAT OFF (the default) could be used, but the DOS program would need to know the internal data representation used by HP Instrument BASIC.

HP Instrument BASIC programs can also be exchanged with a PC using a DOS disk. The HP 3577B [GET PROGRAM] key and the HP Instrument BASIC GET command can recall HP Instrument BASIC programs from a DOS file. The [RE-SAVE PROGRAM] key and the SAVE command will create an untyped (DOS) file on a DOS disk. See chapter 4 for information on using the [GET PROGRAM] and [RE-SAVE PROGRAM] softkeys and chapter 5 for information on transferring program files to and from an MS-DOS editor.

Interfacing with the HP-IB

Introduction

This section describes the techniques necessary for programming the HP-IB interface. It also describes specific details of how this interface works and how to use it to control or interface with systems containing various HP-IB devices.

The HP-IB interface is Hewlett-Packard's implementation of the IEEE-488.1 Digital Interface for Programmable Instrumentation. The acronym HP-IB stands for "Hewlett-Packard Interface Bus," and is often referred to as the "bus."

The HP-IB Interface is both easy to use and allows great flexibility in communicating data and control information between the HP Instrument BASIC program and external devices.

HP Instrument BASIC is essentially an HP-IB instrument controller residing inside an instrument. It uses the host instrument's HP-IB interface for external communication and an internal HP-IB interface to communicate with the host instrument. This unique arrangement presents a few differences between HP Instrument BASIC's implementation of HP-IB control and the standard HP Series 200/300 BASIC Controller. A description of the interaction of HP Instrument BASIC with the host instrument and the external HP-IB interface is given in the section entitled "The HP-IB Model."

Communicating with Devices

This section describes programming techniques used to transfer data to and from HP-IB devices. General bus operation is also briefly described in this chapter.

HP-IB Device Selectors

Since the HP-IB allows the interconnection of several devices, each device must have a means of being uniquely accessed. Specifying just the interface select code of the HP-IB interface to which a device is connected is not sufficient to uniquely identify a specific device on the bus.

Each device on the bus has an primary address by which it is identified. This address must be unique to allow individual access of each device. Each HP-IB device has a configurable address. Thus, when a particular HP-IB device is to be accessed, it must be identified with both its interface select code and its bus address.

The interface select code is the first part of an HP-IB device selector. HP Instrument BASIC programs run inside a host instrument and communicate with it over the internal bus, which is addressed with select code 8. HP Instrument BASIC programs can also communicate with external devices via the host instrument's HP-IB interface, whose select code is 7.

The second part of an HP-IB device selector is the device's primary address, which is in the range of 0 through 30. For example, to specify the device on the interface at select code 7 with a primary address of 22, use device selector = 722.

Remember that each device's address must be unique. The procedure for setting the address of an HP-IB device is given in the installation manual for each device. Since the host instrument is the only device on the internal interface, its primary address on that interface is arbitrary and the instrument will respond to any primary address with a select code equal to 8 (e.g., 800, 811, 822, etc.).

Moving Data Through the HP-IB

Data is output and entered into the program through the HP-IB with the OUTPUT and ENTER statements, respectively. The only difference between the OUTPUT and ENTER statements for the HP-IB and those for other interfaces is the addressing information within HP-IB device selectors.

Examples

```
100  Hpib=7
110  Device_addr=22
120  Device_selector=Hpib * 100 + Device_addr
130  !
140  OUTPUT Device_selector;"F1R7T2T3"
150  ENTER Device_selector;Reading

320  ASSIGN @Hpib_device TO 702
330  OUTPUT @Hpib_device;"Data message"
340  ENTER @Hpib_device;Number

440  OUTPUT 800;"SAM -10 DBM"

380  ENTER 724;Readings(*)
```

General Structure of the HP-IB

Communications through the HP-IB are made according to a precisely defined set of rules. These rules help to ensure that only orderly communication may take place on the bus. For conceptual purposes, the organization of the HP-IB can be compared to that of a committee. A committee has certain "rules of order" that govern the manner in which business is to be conducted. For the HP-IB, these rules of order are the IEEE 488.1 standard.

On the HP-IB, the System Controller corresponds to the committee chairman. The system controller is generally designated before running a program and cannot be changed under program control. However, it is possible to designate an "acting chairman" on the HP-IB. On the HP-IB, this device is called the Active Controller, and may be any device capable of directing HP-IB activities, such as an instrument (using printing and plotting functions) or a desktop computer.

When the System Controller is first turned on or reset, it assumes the role of Active Controller. Thus, only one device can be designated System Controller. These responsibilities may be subsequently passed to another device while the System Controller tends to other business. This ability to pass control allows more than one computer to be connected to the HP-IB at the same time.

In a committee, only one person at a time may speak. It is the chairman's responsibility to "recognize" which one member is to speak. Usually, all committee members present always listen; however, this is not always the case on the HP-IB. One of the most powerful features of the bus is the ability to selectively send data to individual (or groups of) devices. This allows fast talkers to communicate with fast listeners without having to wait for slower listeners on the bus.

**Interfacing with the HP-IB
Communicating with Devices**

During a committee meeting, the current chairman is responsible for telling the committee which member is to be the talker and which is (are) to be the listener(s). Before these assignments are given, he must get the attention of all members. The talker and listener(s) are then designated, and the next data message is presented to the listener(s) by the talker. When the talker has finished the message, the designation process may be repeated.

On the HP-IB, the Active Controller takes similar action. When talker and listener(s) are to be designated, the attention signal line (ATN) is asserted while the talker and listener(s) are being addressed. ATN is then cleared, signaling that those devices not addressed to listen may ignore all subsequent data messages. Thus, the ATN line separates data from commands; commands are accompanied by the ATN line being true, while data messages are sent with the ATN line false.

On the HP-IB, devices are addressed to talk and addressed to listen in the following orderly manner. The Active Controller first sends a single command that causes all devices to unlisten. The talker's address is then sent, followed by the address(es) of the listener(s). After all listeners have been addressed, the data can be sent from the talker to the listener(s). Only device(s) addressed to listen accept any data that is sent through the bus (until the bus is reconfigured by subsequent addressing commands).

The data transfer, or data message, allows for the exchange of information between devices on the HP-IB. Our committee conducts business by exchanging ideas and information between the speaker and those listening to his presentation. On the HP-IB, data is transferred from the active talker to the active listener(s) at a rate determined by the slowest active listener on the bus. This restriction on the transfer rate is necessary to ensure that no data is lost by any device addressed to listen. The handshake used to transfer each data byte ensures that all data output by the talker is received by all active listeners.

Examples of Bus Sequences

With HP Instrument BASIC, all data transfers through the HP-IB involve a talker and only one listener. For instance, when an OUTPUT statement is used (by the Active Controller) to send data to an HP-IB device, the following sequence of commands and data is sent through the bus.

```
OUTPUT 701;"DATA"
```

1. The unlisten command is sent.
2. The talker's address is sent (here, the address of the active controller), which is also a command.
3. The listener's address (01) is sent, which is also a command.
4. The data bytes "D", "A", "T", "A", CR, and LF are sent; all bytes are sent using the HP-IB's interlocking handshake to ensure that the listener has received each byte.

Similarly, all ENTER statements involve transferring data from a talker to only one listener. For instance, the following ENTER statement invokes the following sequence of commands and data-transfer operations.

```
ENTER 722;Voltage
```

1. The unlisten command is sent.
2. The talker's address (22) is sent, which is a command.
3. The listener's address is sent (here, the active controller's address), also a command.
4. The data is sent by device 22 to the controller using the HP-IB handshake.

General Bus Management

The HP-IB standard provides several mechanisms that allow managing the bus and the devices on the bus. Here is a summary of the statements that invoke these control mechanisms.

ABORT is used to abruptly terminate all bus activity and reset all devices to power-on states.

CLEAR is used to set all (or only selected) devices to a pre-defined, device-dependent state.

LOCAL is used to return all (or selected) devices to local (front panel) control.

LOCAL LOCKOUT is used to disable all devices' front-panel controls.

REMOTE is used to put all (or selected) devices into their device-dependent, remote modes.

SPOLL is used to perform a serial poll of the specified device (which must be capable of responding).

TRIGGER is used to send the trigger message to a device (or selected group of devices).

These statements (and functions) are described in the following discussion. However, the actions that a device takes upon receiving each of the above commands are, in general, different for each device. For external devices, refer to the particular device's manuals to determine how it will respond.

Note that all of the bus management commands, with the exception of **ABORT**, require that the program be the Active Controller on the interface. A running HP Instrument BASIC program is always Active Controller on the internal interface (select code 8). For the program to be active controller on the external interface (select code 7), the host instrument must either be set as System Controller or have control passed to it from an external controller. The program automatically assumes the controller status of the host instrument. For more information refer to "The HP-IB Model" section later in this chapter.

REMOTE

External Devices

Most HP-IB devices can be controlled either from the front panel or from the bus. If the device's front-panel controls are currently functional, it is in the Local state. If it is being controlled through the HP-IB, it is in the Remote state. Pressing the front-panel "Local" key will return the device to Local (front panel) control, unless the device is in the Local Lockout state, or is the host instrument.

The Remote message is automatically sent to all devices whenever the System Controller is powered on, reset, or sends the Abort message. A device also enters the Remote state automatically whenever it is addressed. The REMOTE statement also outputs the Remote message, which causes all (or specified) devices on the bus to change from local control to remote control. The host instrument must be set to System Controller before an HP Instrument BASIC program can execute the REMOTE statement on select code 7.

Examples

```
REMOTE 7
```

```
ASSIGN @Device TO 700  
REMOTE @Device
```

```
REMOTE 700
```

Host Instrument

The REMOTE statement has no effect on the host instrument since it is always in remote control whenever an HP Instrument BASIC program is running. Specifying the internal interface in a REMOTE statement will not generate an error, but will have no effect.

Interfacing with the HP-IB
General Bus Management
LOCAL LOCKOUT

External Devices

The Local Lockout message effectively locks out the "local" switch present on most HP-IB device front panels, preventing a device's user from interfering with system operations by pressing buttons and thereby maintaining system integrity. As long as Local Lockout is in effect, no bus device can be returned to local control from its front panel.

The Local Lockout message is sent by executing the LOCAL LOCKOUT statement. This message is sent to all devices on the external interface.

Examples

```
ASSIGN @HpiB TO 7  
LOCAL LOCKOUT @HpiB
```

```
LOCAL LOCKOUT 7
```

The Local Lockout message is cleared when the Local message is sent by executing the LOCAL statement. However, executing the ABORT statement does not cancel the Local Lockout message.

Host Instrument

The Local Lockout message is not supported for the host instrument since some front panel functionality is always necessary in order to pause or abort the program. Specifying the internal interface in a LOCAL LOCKOUT statement will not generate an error, but will have no effect.

LOCAL

External Devices

During system operation, it may be necessary for an operator to interact with one or more external devices. For instance, an operator might need to work from the front panel to make special tests or to troubleshoot. And, in general, it is good systems practice to return all devices to local control upon conclusion of remote-control operations. Executing the LOCAL statement returns the specified devices to local (front panel) control.

If primary addressing is specified, the Go-to-Local message is sent only to the specified device(s). However, if only the interface select code is specified (LOCAL 7), the Local message is sent to all devices on the external interface and any previous Local Lockout message (which is still in effect) is automatically cleared.

Examples

```
ASSIGN @Hpib TO 7  
LOCAL @Hpib
```

```
ASSIGN @Device TO 700  
LOCAL @Device
```

Host Instrument

The LOCAL statement has no effect on the host instrument since it is always in remote control whenever an HP Instrument BASIC program is running. Specifying the internal interface in a LOCAL statement will not generate an error.

TRIGGER

External HP-IB Devices

The TRIGGER statement sends a Trigger message to a selected device or group of devices. The purpose of the Trigger message is to initiate some device-dependent action; for example, it can be used to trigger a digital voltmeter to perform its measurement cycle. Because the response of a device to a Trigger Message is strictly device-dependent, neither the Trigger message nor the interface indicates what action is initiated by the device.

Examples

```
ASSIGN @Hpib TO 7  
TRIGGER @Hpib
```

```
ASSIGN @Device TO 707  
TRIGGER @Device
```

Specifying only the interface select code outputs a Trigger message to all devices currently addressed to listen on the bus. Including a device address in the statement triggers only the device addressed by the statement.

Host Instrument

The TRIGGER statement is fully compatible on the internal HP-IB interface. Note that the instrument must be set to trigger on the HP-IB for this command to be effective.

CLEAR

External HP-IB Devices

The CLEAR statement provides a means of "initializing" a device to its predefined, device-dependent state. When the CLEAR statement is executed, the Clear message is sent either to all devices or to the specified device, depending on the information contained within the device selector. If only the interface select code is specified, all devices on the specified HP-IB interface are cleared. If primary-address information is specified, the Clear message is sent only to the specified device. Only the Active Controller can send the Clear message.

Examples

```
ASSIGN @Hpib TO 7  
CLEAR @Hpib
```

```
ASSIGN @Device TO 700  
CLEAR @Device
```

Host Instrument

The CLEAR statement is fully compatible on the internal interface.

Interfacing with the HP-IB
General Bus Management

ABORT

External Devices

This statement may be used to terminate all activity on the external bus and return all the HP-IB interfaces of all devices to a reset (or power-on) condition. Whether this affects other modes of the device depends on the device itself. The HP Instrument BASIC program must be either the Active or the System Controller to perform this function. If it is the System Controller and has passed active control to another device, executing this statement causes active control to be returned. Only the interface select code may be specified; primary-addressing information (such as 724) must not be included.

Examples

```
ASSIGN @HpiB TO 7  
ABORT @HpiB
```

```
ABORT 7
```

Aborting the Internal Bus

ABORT is not supported for select code 8. Executing ABORT 8 will not generate an error.

HP-IB Service Requests

Most HP-IB devices, such as voltmeters, frequency counters, and spectrum analyzers, are capable of generating a "service request" when they require the Active Controller to take action. Service requests are generally made after the device has completed a task (such as making a measurement) or when an error condition exists (such as a printer being out of paper). The operating and/or programming manuals for each device describe the device's capability to request service and conditions under which the device will request service.

To request service, the device sends a Service Request message (SRQ) to the Active Controller. The mechanism by which the Active Controller detects these requests is the SRQ interrupt. Interrupts allow an efficient use of system resources, because the system may be executing a program until interrupted by an event's occurrence. If enabled, the external event initiates a program branch to a routine which "services" the event (executes remedial action).

Setting Up and Enabling SRQ Interrupts

In order for an HP-IB device to be able to initiate a service routine in the Active Controller, two prerequisites must be met: the SRQ interrupt event must have a service routine defined, and the SRQ interrupt must be enabled to initiate the branch to the service routine. The following program segment shows an example of setting up and enabling an SRQ interrupt.

```
100 Hpib=7
110 ON INTR Hpib GOSUB Service_routine
120 !
130 Mask=2
140 ENABLE INTR Hpib;Mask
```

Since HP Instrument BASIC recognizes only SRQ interrupts, the value assigned to the mask is meaningless. However, a mask value may be present as a placeholder for compatibility with HP Series 200/300 BASIC programs.

When an SRQ interrupt is generated by any device on the bus, the program branches to the service routine when the current line is exited (either when the line's execution is finished or when the line is exited by a call to a user-defined function). The service routine, in general, must perform the following operations:

1. Determine which device(s) are requesting service (parallel poll)
2. Determine what action is requested (serial poll)
3. Clear the SRQ line
4. Perform the requested action
5. Re-enable interrupts
6. Return to the former task (if applicable)

Interfacing with the HP-IB
General Bus Management
Servicing SRQ Interrupts

The SRQ is a level-sensitive interrupt; in other words, if an SRQ is present momentarily but does not remain long enough to be sensed by the controller, an interrupt will not be generated. The level-sensitive nature of the SRQ line also has further implications, which are described in the following paragraphs.

Example

Assume only one device is currently on the bus. The following service routine first serially polls the device requesting service, thereby clearing the interrupt request. In this case, the controller did not have to determine which device was requesting service because only one device is on the bus. Since only service request interrupts are enabled in HP Instrument BASIC, the type of interrupt need not be determined either. The service is then performed, and the SRQ event is re-enabled to generate subsequent interrupts.

```
500 Serv_rtn: Ser_poll=SPOLL(@Device)
510   ENTER @Device;Value
520   PRINT Value
530   ENABLE INTR 7 ! Use previous mask.
540   RETURN
```

The IEEE standard has defined that when an interrupting device is serially polled, it is to stop interrupting until a new condition arises (or the same condition arises again). In order to "clear" the SRQ line, it is necessary to perform a serial poll on the device. This poll is an acknowledgement from the controller to the device that it has seen the request for service and is responding. The device then removes its request for service (by releasing SRQ).

Had the SRQ line not been released, the controller would have branched to the service routine immediately upon re-enabling interrupts on this interface. This is another implication of the level-sensitive nature of the SRQ interrupt.

It is also important to note that once an interrupt is sensed and logged, the interface cannot generate another interrupt until the initial interrupt is serviced. The controller disables all subsequent interrupts from an interface until a pending interrupt is serviced. For this reason, it was necessary to allow for subsequent branching.

Conducting a Serial Poll

A sequential poll of individual devices on the bus is known as a Serial Poll. One entire byte of device-specific status is returned in response to a Serial Poll. This byte is called the "Status Byte" message and, depending on the device, may indicate an overload, a request for service, or a printer being out of paper. The particular response of each device depends on the device.

The SPOLL function performs a Serial Poll of the specified device; the program must currently be the Active Controller in order to execute this function.

Examples

```
ASSIGN @Device TO 700  
Status_byte=SPOLL(@Device)
```

```
Spoll_724=SPOLL(724)
```

The Serial Poll is meaningless for an interface since it must poll individual devices on the interface. Therefore, primary addressing must be used with the SPOLL function.

Passing and Regaining Control

Passing control can be accomplished in one of two ways: it can be handled by the system, or it can be handled by the program. To handle it programmatically, use the PASS CONTROL statement. Control can only be passed on the external interface (select code 7). The following statements first define the HP-IB Interface's select code and the new Active Controller's primary address and then pass control to that controller.

```
100 Hp_ib=7  
110 New_ac_addr=20  
120 PASS CONTROL 100*Hp_ib+New_ac_addr
```

Once the new Active Controller has accepted active control, the controller passing control assumes the role of a non-Active Controller on the specified HP-IB Interface. **HP Instrument BASIC programs cannot act as a device when in the role of non-Active controller.** Note that active control of the internal HP-IB interface cannot be passed. These concepts are discussed next.

The HP Instrument BASIC HP-IB Model

The fact that HP Instrument BASIC resides in, and co-exists with an instrument poses a large set of possible interactions, both internal to the instrument and externally with other controllers and instruments. This section defines the principal players and rules of order when HP Instrument BASIC executes within the host instrument.

External and Internal Busses

There is physically only one HP-IB port and one HP-IB address for the HP 3577B. HP Instrument BASIC has access to two HP-IB ports: the “real” external port (select code 7) and a “virtual” internal port (select code 8), through which it communicates with the HP 3577B.

The HP 3577B always behaves as if there is only one controller. If an HP Instrument BASIC program is running, it is assumed to be the controller and therefore will receive all SRQ's from the host instrument (via the internal port). An HP Instrument BASIC program cannot generate an SRQ on the external port while running.

Service Request Indicators

An external controller may perform a serial poll (SPOLL) at any time without affecting a running HP Instrument BASIC program. There are two Service Request Indicators (SRI) – one for the external port and one for the internal port. The internal SRI can only be cleared by an HP Instrument BASIC program performing an SPOLL on device 800. The external SRI can only be cleared by an SPOLL from an external controller and can only be set when there is not an active HP Instrument BASIC program.

The two SRI's will be set to their OR'd value when a program starts, and again when it finishes. This assures that any pending SRQ's can be serviced by the instrument's new controller.

The pausing or termination of a program will cause the Program_stopped bit in the Device Status register to be set. This can be used to generate an external SRQ.

HP Instrument BASIC as the Active Controller

The HP Instrument BASIC program is always the Active Controller on the internal interface (select code 8). When a program starts running, the HP-IB controller status of the instrument is automatically passed to the program. For example, if the instrument is set as System Controller, a program running in the instrument automatically becomes System Controller and Active Controller on the external bus and the instrument relinquishes active control. When the program stops, the instrument regains active control.

Also, if an instrument set as System Controller OFF is passed control from an external controller, any program running in the instrument becomes active controller on the external interface.

Thus, there are two cases where a program running in an instrument can be Active Controller on the external interface:

- When the host instrument is set as System Controller ON and the program has not passed control
- When the host instrument is set as System Controller OFF and the instrument has been passed control from an external controller. In this case, the words "SYS CTLR" in the SYS CTLR ON OFF softkey will be bright to indicate that the HP 3577B is the active controller, and "OFF" will be bright to indicate the HP 3577B will not become the system controller at power on.

HP Instrument BASIC as the Active Controller

The HP Instrument BASIC program is always the Active Controller on the internal interface (select code 8). When a program starts running, the HP-IB controller status of the instrument is automatically passed to the program. For example, if the instrument is set as System Controller, a program running in the instrument automatically becomes System Controller and Active Controller on the external bus and the instrument relinquishes active control. When the program stops, the instrument regains active control.

Also, if an instrument set as System Controller OFF is passed control from an external controller, any program running in the instrument becomes active controller on the external interface.

Thus, there are two cases where a program running in an instrument can be Active Controller on the external interface:

- When the host instrument is set as System Controller ON and the program has not passed control
- When the host instrument is set as System Controller OFF and the instrument has been passed control from an external controller. In this case, the words "SYS CTLR" in the SYS CTLR ON OFF softkey will be bright to indicate that the HP 3577B is the active controller, and "OFF" will be bright to indicate the HP 3577B will not become the system controller at power on.

Interfacing with the HP-IB
The HP Instrument BASIC HP-IB Model

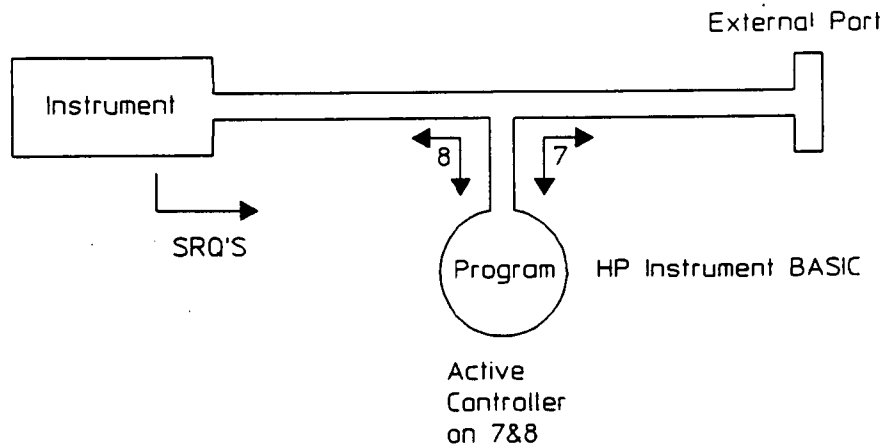


Figure 6-1. The Program as External Active Controller

Note



The HP 3577B always assumes System Controller status whenever control is passed to it from another controller. Therefore it is necessary to pass control back to any external controller that has passed control to the HP 3577B. Issuing an Interface Clear (IFC) from an external controller after it has passed control to the HP 3577B will not return Active Control of the HP-IB to the external controller.

HP Instrument BASIC as a Non-Active Controller

HP Instrument BASIC programs are always the Active Controller on the internal interface. There are two cases where an HP Instrument BASIC program does not have control of the external HP-IB interface:

- When the host instrument is set as System Controller OFF and active control has not been passed from an external device
- When the host instrument is set as System Controller ON and the program has passed control to another device on the external interface. In this case the words "SYS CTLR" in the SYS CTLR ON OFF softkey will be dim to indicate that the HP 3577B is no longer active controller. "ON" will be bright to indicate that the HP 3577B will become the System Controller at power on.

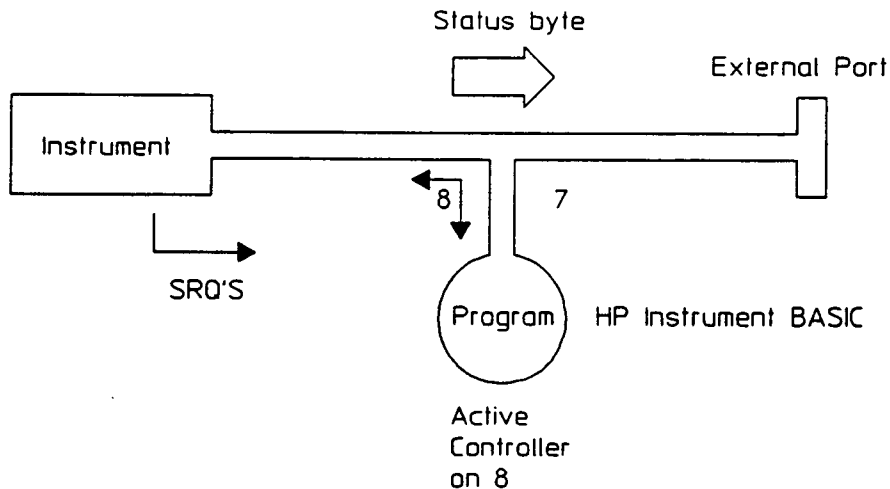


Figure 6-2. Program as Non-active External Controller

In both of these cases, the program cannot perform activities of any kind on the external interface.

Note



An HP Instrument BASIC program cannot act as a device on the external bus. To communicate with an external controller, the HP Instrument BASIC program must be Active Controller and the external controller must act as the device (see "Interfacing with an External Controller").

Interfacing with an External Controller

So far, we have discussed in some detail the ability to interface HP Instrument BASIC programs via HP-IB with a network of external devices. The idea of including another computer in that network and interfacing an HP Instrument BASIC program with a program running in that computer presents some new possibilities.

External controller programs can interface with HP Instrument BASIC programs (hereafter referred to as "internal programs") over HP-IB in two basic ways:

First, the two programs can pass data back and forth using simple OUTPUT and ENTER statements. This requires coordination of both the internal and external programs and also requires that the internal program be the Active Controller during the interaction. To get an internal program and an external program to work together successfully, you should have a good understanding of the HP-IB model, as presented earlier in this chapter.

Second, the external program can make use of the set of HP 3577B HP-IB commands that interface with HP Instrument BASIC programs. These mnemonics allow the external controller to remotely perform many of the HP Instrument BASIC front-panel activities. This includes the ability to run, stop, pause, continue or delete an internal program. You can also remotely set a program's memory size.

Also included in the HP 3577B HP-IB command set are commands that allow you to transfer programs to and from the instrument. That is, programs may be uploaded from and downloaded to the instrument program memory. These HP-IB mnemonics are described in detail in Appendix A in this manual.

Transferring Data Between Programs

Using OUTPUT and ENTER Statements

All data sent from an external controller to the instrument's external port is received by the instrument and not by any program running in it. Therefore, a non-active Controller HP Instrument BASIC program can never enter or output data via the external interface. This means that in order to pass data between an external controller and an internal program using OUTPUT and ENTER statements, the internal program must be given active control and the external controller must become the non-Active Controller. All HP Series 200/300 BASIC controllers have the ability to enter and output data via HP-IB while acting as a non-Active Controller.

Note



Moving data through the HP-IB and running a measurement in the host instrument at the same time can slow both operations significantly. It is recommended that you do not perform these operations concurrently.

One method of passing data between the two controllers is to first set the instrument as System Controller OFF, and then run an HP Series 200/300 BASIC program that starts the HP Instrument BASIC program and then passes control to it. Thereafter, the HP Instrument BASIC program can output data to, and enter data from, the external controller. The following two programs demonstrate how to transfer data between an internal program and an external controller program.

The first program, DTXFRB, is run from an external controller running HP Series 200/300 BASIC. It remotely loads the HP Instrument BASIC program, starts it and then transfers active control to it. The HP Instrument BASIC program DTXFRA, with active control of the interface, queries the external program for name of the drive to catalog, and then OUTPUTs the cataloged string to the external program and passes active control back. After receiving the catalog data, the external program goes into a loop (line 560) executing a command that will always generate an error until the host computer is active controller when control is passed back.

Interfacing with the HP-IB
Interfacing with an External Controller

```
10 ! RE-SAVE "DTXFRB"
20 !
30 ! This program demonstrates how to transfer data between an Instrument
40 ! BASIC program running in the HP 3577B and an external controller.
50 ! In this example, a catalog listing is transferred from an
60 ! Instrument BASIC program (DTXFRA), to this program running in
70 ! an external controller.
80 !
90 ! This program first downloads a program to the HP 3577B, then
100 ! runs it and gives it control of the bus. This program then
110 ! acts as a device on the bus; when addressed to talk it sends
120 ! the mass storage unit specifier of the device to catalog, and
130 ! when addressed to listen it receives the catalog.
140 !
150 Scode=7 ! Select code for HP-IB interface
160 Address=11 ! HP-IB address for HP 3577B
170 ASSIGN @Na TO Scode*100+Address
180 CLEAR @Na
190 DIM Directory$(1:100)[85] ! String Array for CATalog
200 DIM Line$(256) ! String for each program line
210 !
220 DISP "Downloading DTXFRA program to HP 3577B..."
230 ASSIGN @Basic_prog TO "DTXFRA" ! Open file to download
240 OUTPUT @Na;"SCA" ! SCRATCH A to clear prog space
250 ON END @Basic_prog GOTO End_of_file ! Set up branching on end of file
260 OUTPUT @Na;"LPR #I"; ! Initiate data transfer
270 LOOP
280 ENTER @Basic_prog;Line$ ! Read program line from file
290 OUTPUT @Na;Line$ ! Send to HP 3577B
300 END LOOP ! Loop until end of file (EOF)
310 End_of_file: !
320 OUTPUT @Na;CHR$(10) END ! Output a line-feed with EOI
330 ASSIGN @Basic_prog TO * ! Close the file
340 !
350 !
360 DISP "Running the program..."
370 OUTPUT @Na;"PRR" ! Send Program Run command
380 PASS CONTROL @Na ! Pass HP-IB control
390 !
400 DISP "Sending name of device to catalog..."
410 OUTPUT Scode;":,700,0" ! since the interface is not the controller,
420 ! this OUTPUT will wait until addressed
430 ! to talk by the HP 3577B, then send the string.
440 !
450 !
460 DISP "Reading catalog data..."
470 ENTER Scode;Directory$(*) ! ENTER will wait until addressed to listen
480 ! by the HP 3577B, then enter the string array.
490 !
500 DISP "Printing..."
510 FOR I=1 TO 100
520 IF LEN(Directory$(I))>0 THEN PRINT Directory$(I)
530 NEXT I
```

Interfacing with the HP-IB
Interfacing with an External Controller

```
540 !
550 DISP "Waiting for control..."
560 ON ERROR GOTO 560 ! loop until control is passed back.
570 LOCAL @Na
580 OFF ERROR
590 !
600 DISP "Done."
610 END

10 ! RE-SAVE "DTXFRA"
20 !
30 ! This program demonstrates how to transfer data between an Instrument
40 ! BASIC program running in the HP 3577B and an external controller.
50 ! In this example, a catalog listing is transferred from this
60 ! Instrument BASIC program to an external controller running the
70 ! companion program, DTXFRB.
80 !
90 Host=721 ! HP-IB address of external controller
100 DIM Stor_dev$(40) ! Storage device MSUS, sent from host.
110 DIM Directory$(1:100)[85] ! String Array for CATalog
120 !
130 !
140 PRINT "Waiting request..."
150 ON ERROR GOTO 150 ! Loop until control is passed to HP 3577B
160 ENTER Host;Stor_dev$ ! Read host to get device to catalog
170 OFF ERROR
180 !
190 PRINT "Reading catalog..."
200 CAT Stor_dev$ TO Directory$(*) ! Catalog to string array
210 !
220 PRINT "Transferring data..."
230 OUTPUT Host;Directory$(*) ! Address host to listen, write array
240 !
250 PASS CONTROL Host ! Return HP-IB control to host
260 PRINT "Done."
270 END
```

Interfacing with the HP-IB
Interfacing with an External Controller

Downloading and Uploading Programs

Programs can be transferred between an external controller and program memory using the HP-IB download command "LPR" and its converse upload query "DPR". Programs that use these mnemonics are run in the external controller.

The program data is always transferred in what is called an "arbitrary block." An arbitrary block is simply a block of data that begins with the characters "#I" preceding the first line and ends with a line-feed character accompanied by an EOI signal on the HP-IB interface.

When using the mnemonic "LPR" to download program lines, the "#I" must not be followed by a line-feed. Each program line must then have a line number at its beginning and a line-feed at its end. To end the arbitrary block of program lines, a single line-feed must be output with the OUTPUT END parameter, which sends the EOI (End or Identify) signal on the HP-IB control lines. The following program, which runs on an external HP Series 200/300 workstation, demonstrates downloading a short program into the instrument.

```
10    ASSIGN @Na TO 711
20    OUTPUT @Na;"SCP"           ! SCRATCH P to clear prog space
30    !
40    OUTPUT @Na;"LPR #I";       ! send Load Program and #I to start
50    !
60    OUTPUT @Na;"10 BEEP"
70    OUTPUT @Na;"20 END"
80    !
90    OUTPUT @Na;CHR$(10) END    ! Output LF with EOI to terminate
100   END
```

Notice that the OUTPUT statement on line 40 is terminated with a semicolon. This suppresses the line-feed that would otherwise occur.

As each line of the program is downloaded it is checked for syntax. If an error is found, the syntax error message is displayed on the CRT and the line is commented and syntax-checked again. If it still causes an error (for example the line may be too long) the line is discarded.

Any lines that currently exist in the instrument will remain unless they are overwritten by downloaded program lines. This makes it easy to edit lines in an external controller and then download only the edited lines into an existing program. If you want to completely overwrite the current program in memory, you must delete the program first. This can be done remotely using the command "SCP", see Appendix A, "Extended HP-IB Command Reference."

The mnemonic "DPR" is used to upload a program from the HP 3577B. The entire program is then returned as an arbitrary block starting with the characters "#I" and ending with a separate linefeed character. The following program demonstrates an uploading routine run on an external controller:

```
10    DIM A$(100)[256]           !Dimension string array for program lines
20    ASSIGN @Na TO 711
30    !
40    OUTPUT @Na;"DPR"          ! send Dump Program to start
50    !
60    ENTER @Na USING "%,K";A$(*) ! Enter program into string array
70    !
80    END
```

Instrument-Specific HP Instrument BASIC Features

Introduction

The *HP Instrument BASIC Users Handbook* that accompanies this manual includes *HP Instrument BASIC Programming Techniques*, *HP Instrument BASIC Interfacing Techniques*, and *HP Instrument BASIC Language Reference*. These references are included with all Hewlett-Packard instruments that use HP Instrument BASIC. Since each instrument is different, the way that HP Instrument BASIC interfaces and interacts with its host often changes from one instrument to another. For example, some instruments employ editors, while others do not, and front panel interfaces often vary a great deal from one instrument to another. For this reason, many parts of the the *HP Instrument BASIC Users Handbook* are either generic in nature, or apply to only one of many possible instrument interfaces.

This chapter describes how to interpret the *HP Instrument BASIC Users Handbook* for the HP 3577B by listing exceptions and additions. Global Exeptions discusses differences that often apply throughout the manual set and lists differences by category. Specific Exceptions lists all HP Instrument BASIC commands used in the HP 3577B and summarizes differences from the command descriptions in the *HP Instrument BASIC Language Reference* command.

Global Exceptions

In general, the differences can be categorized as

- HP-IB, GPIO and RS-232 interfaces.
- CRT and keyboard interfaces
- Disk I/O
- Miscellaneous command differences
- Commands not implemented

Each of these categories is explained in detail in this section.

HP-IB, GPIO and RS-232 Interfaces

The *HP Instrument BASIC Users Handbook* refers to various interface types, particularly in chapter 2 of *HP Instrument BASIC Interfacing Techniques*. HP Instrument BASIC in the HP 3577B supports the HP-IB and a limited GPIO interface, and not the RS-232 interface.

Also, HP Instrument BASIC in the HP 3577B supports only three interface select codes: 7 for the external bus, 8 for the internal bus, and 12 for the GPIO. This affects the ABORT, ENABLE INTR, OFF INTR, ON TIMEOUT, and OFF TIMEOUT commands, where select codes are limited to 7 and 8. Furthermore, the LOCAL LOCKOUT and REMOTE commands do not apply to the internal HP-IB interface (select code 8).

Finally, the internal default select codes for CRT (1), keyboard (2) apply, but the select code RS-232 (9) does not apply in the HP 3577B. The CRT function returns 1 and the KBD function returns 2 and, when select code 2 is used with ENTER (as in ENTER 2) the same menu as is used by the INPUT command appears to allow text entry from the front panel, however, no prompt is issued.

CRT and Keyboard Interfaces

CRT Differences

References in the *HP Instrument BASIC Users Handbook* to the CRT generally assume a standard 80 column terminal. The HP 3577B has a 46 column display for text. This affects references to the width of the default when the PRINTER IS device is the display for LIST, PRINT and PRINTER IS commands. When PRINTER IS selects external printers the default is 80 characters.

The support of ASCII control codes when PRINTER IS = 1, listed in the PRINT and DISP keyword descriptions, does not apply in the HP 3577B.

For the MOVE and DRAW graphics commands, the maximum X,Y coordinates are 2047. (The minimum coordinates are always 0,0.) See "Graphics and Display Techniques" in chapter 5, for more information on using the MOVE and DRAW commands.

Finally, the PEN command is implemented in the HP 3577B with two parameters, 0 and 1 (or any non-zero number). PEN 1 causes any following DRAW commands to draw a line on the display. PEN 0 erases all line segments that duplicate DRAW commands. Note that PEN 0 does not perform a pixel complement function as it does in HP BASIC.

Keyboard Differences

The *HP Instrument BASIC Users Handbook* assumes the use of a standard HP BASIC Series 200 workstation keyboard. It also assumes that HP Instrument BASIC works in command line execution mode, where individual commands may be entered and executed from the keyboard. HP Instrument BASIC in the HP 3577B has neither a keyboard nor a command line execution mode. However, it does have front-panel softkeys that emulate many of the command line executed keywords (such as RUN, CONTINUE, SCRATCH, etc.). The standard HP BASIC Series 200 Workstation editor uses many of these command line executable commands, and many of these have been remapped to softkeys under the [SPCL FCTN] hardkey or the [SPCL FCTN] [INSTRMNT BASIC] softkey. The following HP Instrument BASIC keywords have corresponding softkeys and hardkeys in the HP 3577B:

CONTINUE

[SPCL FCTN] [CONTINUE PROGRAM]

LIST

[SPCL FCTN] [INSTRMNT BASIC] [LIST] [DO LIST]

PAUSE

[SPCL FCTN]

RUN

[SPCL FCTN] [RUN PROGRAM]

SCRATCH

[SPCL FCTN] [INSTRMNT BASIC] [SCRATCH]

Instrument-Specific HP Instrument BASIC Features
Global Exceptions

SECURE

[SPCL FCTN] [INSTRMNT BASIC] [MORE] [SECURE]

STOP

[LCL]

RESET

[LCL]

The following keywords have descriptions in the *HP Instrument BASIC Language Reference* that use standard keyboard keys and need clarification for use in the HP 3577B:

EDIT

Ignore all documentation in the *HP Instrument BASIC Language Reference* on the EDIT command. All program editing of HP Instrument BASIC programs for the HP 3577B must be performed in an external controller.

ON KEY and OFF KEY

There are 8 softkeys available for use in the HP 3577B. These appear on the right of the display in place of normal instrument softkeys. Key selector values may range from 0 through 7. Other values generate a "improper value or out of range" error.

INPUT

When an INPUT statement is encountered in an HP Instrument BASIC program, an alpha entry menu appears to allow the user to use the alpha and symbol softkeys, the knob, and the keypad to enter a response.

To enter an input response press the [ENTER] softkey in the alpha entry menu. Disregard all keys mentioned in the *HP Instrument BASIC Language Reference* description of this key. You can only terminate the INPUT command by pressing the [ENTER] softkey.

ENTER

For a description of using ENTER from the keyboard (ENTER KBD) see INPUT.

Note



In the *HP Instrument BASIC Language Reference* every command contains a line stating whether or not the command is "keyboard executable." Disregard this information for HP Instrument BASIC in the HP 3577B, since no commands are keyboard executable.

Disk I/O

Note



The *HP Instrument BASIC Users Handbook* is continually revised to support all implementations of HP Instrument BASIC. DOS file support is new to the HP Instrument BASIC environment and, to insure that its implementation in the HP 3577B is well understood, it is described here and in chapter 5, (in the section "Using the DOS File System"). Should you encounter any discrepancy between DOS information in the *HP Instrument BASIC Users Handbook* and this manual, rely on the information in this manual for HP Instrument BASIC in the HP 3577B.

HP Instrument BASIC in the HP 3577B recognizes two disk formats: LIF (Logical Interchange Format), and DOS. The format of a disk (or file system) is determined when it is formatted (or initialized). A LIF disk contains only one directory, while a DOS disk can have a "hierarchical" structure of directories.

HP Instrument BASIC can create three types of files that may exist on either a DOS or a LIF formatted disk: ASCII, BDAT and untyped. If you catalog a DOS disk, these three types will show up as "ASCII," "BDAT," and "DOS." If you catalog a LIF disk, they will show up as "ASCII," "BDAT," and "HP-UX." Table 7-1 illustrates these configurations.

Table 7-1. HP Instrument BASIC File Types

File Type	Appears on a LIF disk as	Appears on a DOS disk as
ASCII	ASCII	ASCII
BDAT	BDAT	BDAT
untyped	HP-UX	DOS

Note



It is important to remember that the ASCII file type described here is specific to HP Instrument BASIC and HP BASIC and is not the same as the standard "ASCII" file type in the MS-DOS environment. Only untyped files are editable by an MS-DOS ASCII editor on a PC. An untyped file is automatically generated whenever an HP Instrument BASIC program is SAVED from the HP 3577B to a DOS-formatted disk. A RE-SAVE maintains the original file type if a file exists, or else acts the same as SAVE.

Instrument-Specific HP Instrument BASIC Features Global Exceptions

The *HP Instrument BASIC Users Handbook* refers to another file system, called Hierarchical File System (HFS). Although this is similar in many cases to DOS, it is not supported in the HP 3577B.

In addition LIF protect codes are ignored by HP Instrument BASIC in the HP 3577B and are neither generated nor observed. They will not generate an error if encountered.

Following is a list of HP Instrument BASIC disk I/O commands. In addition to the descriptions of these commands that appear in the *HP Instrument BASIC Language Reference*, a summarized description of the DOS implementation of each command in HP 3577B HP Instrument BASIC is presented here.

In the entry for each of the following HP Instrument BASIC commands in the *HP Instrument BASIC Language Reference*, the syntax diagram includes a "file specifier." This file specifier is always further described in a syntax diagram labeled "literal form of file specifier," which includes a "directory path" and a "volume specifier." The "directory path" is valid for the DOS file system in every case. Refer to the description of MASS STORAGE IS in this section for a correct syntax diagram and description of "directory path" and "volume specifier" in the HP 3577B.

Also, every command includes a file name and directory name specifier. In all cases, DOS file names may be up to eight characters long with a three character extension (a period followed by up to three characters). DOS directory names may be up to eight characters with no extension. DOS file and directory names are not case sensitive.

ASSIGN

When using files in an ASSIGN statement, you may use BDAT, ASCII or untyped (DOS) files on a DOS disk.

A single I/O path name may not be ASSIGNed to a group of devices in HP Instrument BASIC. It is confined to a single device on the HP-IB or a single mass storage file.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

CAT

A catalog of a DOS disk is much simpler than that of a LIF disk. It simply lists file name, file type, number of records, and the record length. All file types recognized on a LIF disk will be recognized on a catalog of a DOS disk, with the exception that an untyped file will show up as a DOS type file on the DOS catalog and an HP-UX file on a LIF disk.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

COPY

Files may be copied between LIF and a DOS formatted disks. With the exception of untyped files, all file types on a LIF disk, when copied to a DOS disk, maintain their internal structure and have an additional 512 byte header attached to define their original structure. When these type files are copied back from DOS to LIF disks, these headers are removed. Untyped files are not changed at all from one file system to another.

So, for example, an ASCII file copied from a LIF to a DOS disk is not editable by a standard ASCII editor on an MS-DOS computer (only untyped files are). See chapters 4 and 5 for information on transferring files between systems for editing, etc.

You cannot copy entire DOS directories.

See MASS STORAGE IS for a correct syntax diagram of “directory path” and “volume specifier.”

CREATE

CREATE creates a new file of untyped file type. This appears as HP-UX on a LIF disk and DOS on DOS disk. The logical record size is always 1. On LIF disks the number of records is fixed and must be entered in the command. On DOS disks, the number of records is ignored since untyped files are extensible.

See MASS STORAGE IS for a correct syntax diagram of “directory path” and “volume specifier.”

CREATE ASCII

The CREATE ASCII command creates an ASCII file on either LIF or DOS media. ASCII files on a DOS disk include an additional 512 byte header that defines the file type. Unlike ASCII files on an HFS volume, ASCII files on a DOS disk are not extensible; they have a fixed size that must be included in the command syntax.

See MASS STORAGE IS for a correct syntax diagram of “directory path” and “volume specifier.”

CREATE BDAT

The CREATE BDAT command creates an BDAT file on either LIF or DOS media. BDAT files on a DOS disk include an additional 512 byte header that defines the file type. Unlike BDAT files on an HFS volume, BDAT files on a DOS disk are not extensible; they have a fixed size that must be included in the command syntax.

See MASS STORAGE IS for a correct syntax diagram of “directory path” and “volume specifier.”

Instrument-Specific HP Instrument BASIC Features

Global Exceptions

GET

GET reads the specified ASCII or untyped file and attempts to store the strings into memory as program lines. The file may be either ASCII or untyped regardless of disk format (DOS or LIF). Untyped files are HP-UX files on a LIF disk or DOS files on a DOS disk. Any HP Instrument BASIC program created by a standard "ASCII" editor on an MS-DOS computer is an untyped (DOS) file, and may be used in a GET statement. (Note that standard MS-DOS "ASCII" editors do not create HP Instrument BASIC ASCII files.)

The GET command is implemented from the front panel by the [RECALL] [GET PROGRAM] softkey.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

INITIALIZE

HP Instrument BASIC in the HP 3577B does not have an internal disk drive, and does not allow Memory Volumes (RAM disks) to be created. Disregard any references to Memory Volumes, Internal Drives, "MEMORY", and "INTERNAL", in the *HP Instrument BASIC Users Handbook*. HP Instrument BASIC does support external disk CS80 drives connected via HP-IB.

The disk format (LIF or DOS) is specified when the media is initialized. For example,

```
INITIALIZE "LIF: ,707,1" ,0,0
```

will initialize a LIF disk in unit 1 of the drive at HP-IB address 7, using default values for the Interleave Factor and Format Option.

To initialize a DOS disk in the same drive,

```
INITIALIZE "DOS: ,707,1" ,0,16
```

should be used. The format option, 16 in this example, is important when initializing a DOS disk. An incorrect format option will result in a disk that the HP 3577B can use, but other MS-DOS systems can not. **This potential problem can be avoided by formatting the disk on the MS-DOS system, rather than the HP 3577B.**

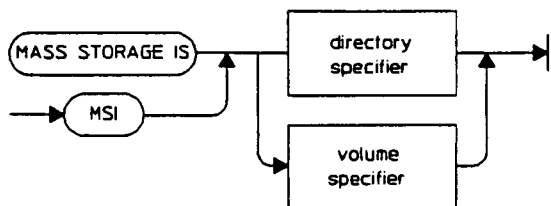
See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

MASS STORAGE IS

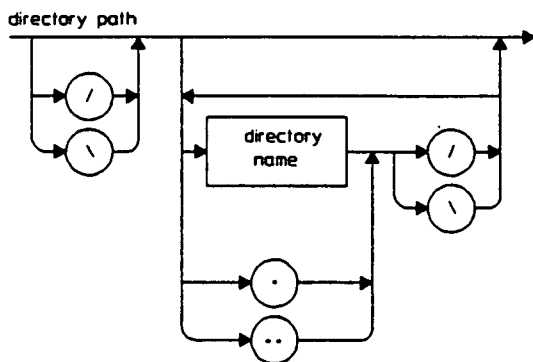
The MASS STORAGE IS statement is consistent with the [LCL] [DISK ADDRESS] and [LCL] [DISK UNIT] keys. The MASS STORAGE IS device is inherited by the program from the front-panel setting when it begins and any change of MSI in the program remains when the program is paused or terminates.

The front panel softkeys, however, can only MSI to the root of a DOS directory, whereas the MASS STORAGE IS command can specify any directory or subdirectory on the DOS media. To reset the MSI to the root of a DOS directory, after a program has designate a subdirectory as the MSI device, simply reset the unit using the [LCL] [DISK UNIT] softkey, or press [INSTRUMENT PRESET].

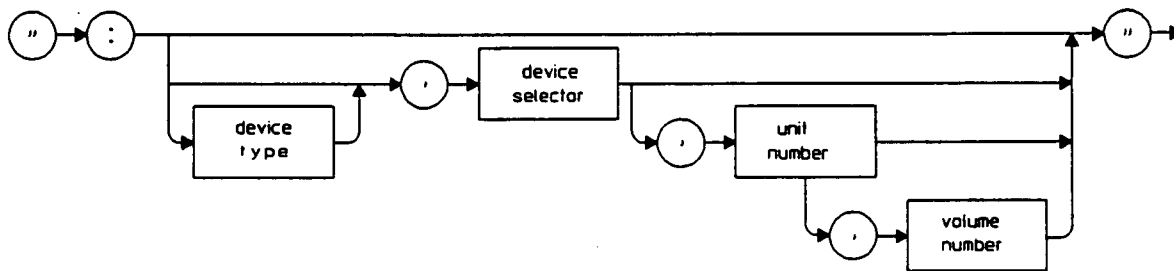
All disk I/O commands in the *HP Instrument BASIC Language Reference* refer to the MASS STORAGE IS description "directory path" and "volume specifier" syntax diagrams. HP Instrument BASIC in the HP 3577B uses the following syntax for MASS STORAGE IS and for all directory path and volume specifier specifications:



literal form of the file specifier (DOS filesystem only):



literal form of volume specifier



Instrument-Specific HP Instrument BASIC Features
Global Exceptions

PRINTER IS

The PRINTER IS statement is consistent with the [SPCL FCTN] [INSTRMNT BASIC][PRINTER IS] softkey. The PRINTER IS device is inherited by the program from the front-panel setting when it begins and any change of PRINTER IS in the program remains when the program is paused or terminates. If a program defines the PRINTER IS device as a file, and pauses or terminates, pressing the [PRINTER IS] softkey will display "PRINTER IS 0."

When the PRINTER IS device is a file, it must be either a BDAT file or an untyped (HP-UX or DOS) file. If an untyped file is created on a DOS disk (see CREATE), it will be extensible; however, since LIF files must be created with a definite size, you risk getting an end-of-file error when the end of a LIF file is encountered.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

PURGE

On a DOS disk, both files and directories may be purged. Files on both DOS and LIF have the same requirements; that is, they must not be open. To remove a directory on a DOS disk, the directory must be empty and must not be the current MASS STORAGE IS designation. There are no permissions on DOS directories to be concerned with as there are on HFS volumes.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

RENAME

Both files and directories may be renamed on DOS disks. Files and directories must be closed before they may be renamed. MSI to another directory to close a directory.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

RE-SAVE

RE-SAVE works the same programmatically as it does from the front panel [SAVE] [RE-SAVE PROGRAM] softkey. If a file exists on either LIF or DOS media, as either an ASCII or untyped file type, the file type will be maintained and the current program contents will be written to the file.

If no file exists with the specified name on a LIF disk, an ASCII file will be created and written to. If no file exists with the specified name on a DOS disk, an untyped (DOS) file will be created and written to.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

SAVE

On a LIF disk, SAVE will create an ASCII type file. However, on a DOS disk, SAVE will create an untyped (DOS) file. This is done to facilitate the use of other utilities and programs available in the MS-DOS operating system on the file.

See MASS STORAGE IS for a correct syntax diagram of "directory path" and "volume specifier."

Miscellaneous Command Differences

COS

The range of the COS command is -298156827.5707960 to 298156824.4292040.

SYSTEMS

The SYSTEMS command accepts only two topic specifiers, SYSTEM PRIORITY and SYSTEM ID. SYSTEM ID returns "HP3577B."

The ON INTR call must precede ENABLE INTR.

Commands Not Implemented

The following commands appear in the *HP Instrument BASIC Language Reference* but are not implemented in HP Instrument BASIC in the HP 3577B:

EDIT

Editing must be performed with an external controller since the HP 3577B does not support editing.

DEL

Deleting lines is associated with editing which is not supported in the HP 3577B.

REN

Renumbering lines is associated with editing which is not supported in the HP 3577B.

Note



Since the *HP Instrument BASIC Users Handbook* is continually revised to support all implementations of HP Instrument BASIC, there may be other commands that appear in that documentation that are not supported in the HP 3577B. Table 7-2 in the following section lists all HP Instrument BASIC keywords supported by HP Instrument BASIC in the HP 3577B.

Specific Exceptions

Table 7.2 presents a summary of the differences between the descriptions of the keywords in the *HP Instrument BASIC Language Reference* and the way they are implemented in the HP 3577B. Where differences are too extensive to be summarized, references to their explanation in the "Global Exceptions" section are given.

Any keywords or functions found in the *HP Instrument BASIC Language Reference* that do not appear in this table do not apply to HP Instrument BASIC in the HP 3577B and should be ignored.

Table 7-2. HP 3577B Keyword Implementation

Command	Front Panel Support	Exceptions
ABORT	None	Interface Select Code = 7 or 8
ABS	None	None
ACS	None	None
AND	None	None
ASN	None	None
ASSIGN	None	LIF protection ignored Only one HP-IB device per ASSIGN
ATN	None	None
BASE	None	None
BEEP	None	Duration and frequency parameters ignored
BINCMP	None	None
BINEOR	None	None
BINIOR	None	None
BIT	None	None
CALL	None	None
CAT	[SAVE] or [RECALL] and [CATALOG]	See "Disk I/O"
CHR\$	None	None
CLEAR	None	None
CLEAR SCREEN	None	None
COM	None	None
CONT	[SPCL FCTN] [CONTINUE]	No line number or label support
COPY	None	LIF protection ignored See "Disk I/O"
COS	None	Absolute range values -298156827.5707960 to 298156824.4292040
CREATE	None	See "Disk I/O"
CREATE ASCII	None	LIF protect code not allowed See "Disk I/O"

Instrument-Specific HP Instrument BASIC Features
Specific Exceptions

Command	Front Panel Support	Exceptions
CREATE BDAT	None	LIF protection ignored See "Disk I/O"
CREATE DIR	None	DOS disks only
CRT	None	ENTER CRT not supported
DATA	None	None
DEF FN	None	None
DEG	None	None
DIM	None	None
DISABLE	None	None
DISABLE INTR	None	None
DISP	None	CRT control codes not supported
DIV	None	None
DEL	None	Keyword not supported
DRAW	None	Maximum x,y coordinates: 2047,2047
DROUND	None	None
DVAL	None	None
DVAL\$	None	None
EDIT	None	Keyword not supported
ENABLE	None	None
ENABLE INTR	None	Interface Select Code = 7 or 8 Must be preceded by ON INTR
END	None	None
ENTER	None	No HFS support
ERRL	None	None
ERRLN	None	None
ERRM\$	None	None
ERRN	None	None
EXOR	None	None
EXP	None	None
FN	None	None
FOR...NEXT	None	None
FRACT	None	None
GCLEAR	None	Clears the graphics buffer
GET	[RECALL] [GET PROGRAM]	See "Disk I/O"
GOSUB	None	None
GOTO	None	None
IF...THEN	None	None
IMAGE	None	None
INITIALIZE	[SPCL FCTN] [INSTRMNT BASIC] [MORE] [FORMAT LIF DISC] or [FORMAT DOS DISK]	See "Disk I/O"
INPUT	None	See INPUT command in "Keyboard Differences" section

Instrument-Specific HP Instrument BASIC Features
Specific Exceptions

Command	Front Panel Support	Exceptions
INT	None	None
INTEGER	None	None
IVAL	None	None
IVAL\$	None	None
KBD	None	Keyboard = front panel softkey menu
LEN	None	None
LET	None	None
LGT	None	None
LIST	[SPCL FCTN] [INSTRMNT BASIC] [LIST]	Default width = 46 or 80 See "CRT Differences"
LOCAL	None	None
LOCAL LOCKOUT	None	Interface Select code = 7 only
LOG	None	None
LOOP	None	None
LWC\$	None	None
MASS STORAGE IS	None	See "Disk I/O"
MAX	None	None
MAXREAL	None	None
MIN	None	None
MINREAL	None	None
MOD	None	None
MODULO	None	None
MOVE	None	Maximum x,y coordinates: 2047,2047
NOT	None	None
NUM	None	None
OFF ERROR	None	None
OFF INTR	None	Interface Select Code = 7 or 8
OFF KEY	None	Key selectors are 0 thru 7
OFF TIMEOUT	None	Interface Select Code = 7 or 8
ON ERROR	None	None
ON INTR	None	Interface Select Code = 7 or 8
ON KEY	None	Key selectors are 0 thru 7
ON TIMEOUT	None	Interface Select Code = 7 or 8
OR	None	None
OUTPUT	None	No HFS support
PEN	None	0 = erase 1 = draw. See "CRT differences."
PASS CONTROL	None	Select code 8 pass control of external bus to analyzer
PAUSE	None	None
POS	None	None

Instrument-Specific HP Instrument BASIC Features
Specific Exceptions

Command	Front Panel Support	Exceptions
PRINT	None	PRINTER IS default width = 46 or 80. See "CRT Differences."
PRINTER IS	[SPCL FCTN] [INSTRMNT BASIC] [PRINTER IS]	default width = 46 or 80 See "CRT Differences" and "Disk I/O"
PROUND	None	None
PRT	None	None
PURGE	None	LIF protection ignored See "Disk I/O"
RAD	None	None
RANDOMIZE	None	None
RANK	None	None
READ	None	None
REAL	None	None
REM	None	None
REMOTE	None	Not supported on select code 8
REN	None	Keyword not supported
RENAME	None	LIF protection ignored See "Disk I/O"
REPEAT...UNTIL	None	None
RE-SAVE	[SAVE] [RE-SAVE PROGRAM]	See "Disk I/O"
RESTORE	None	None
RETURN	None	None
REV\$	None	None
RND	None	None
ROTATE	None	None
RPT\$	None	None
RUN	[SPCL FCTN] [RUN PROGRAM]	None
SAVE	[RE-SAVE PROGRAM]	See "Disk I/O"
SCRATCH	[SPCL FCTN] [INSTRMNT BASIC] [SCRATCH]	None
SECURE	[SPCL FCTN] [INSTRMNT BASIC] [MORE] [SECURE]	None
SELECT...CASE	None	None
SGN	None	None
SHIFT	None	None
SIN	None	None
SIZE	None	None
SPOLL	None	None
SQRT	None	None
STOP	None	None
SUB	None	None
SYSTEM PRIORITY	None	None

Instrument-Specific HP Instrument BASIC Features
 Specific Exceptions

Command	Front Panel Support	Exceptions
SYSTEM\$	None	Only supports " SYSTEM PRIORITY " and " SYSTEM ID "
TAN	None	None
TIMEDATE	None	None
TRIGGER	None	None
TRIM\$	None	None
UPC\$	None	None
VAL	None	None
VAL\$	None	None
WAIT	None	None
WHILE	None	None

Extended HP-IB Command Reference

This appendix describes an extended set of HP-IB mnemonics that are added to the HP-IB command set of the host instrument when HP Instrument BASIC is installed. These commands fall into three general categories. They can be used to:

- Allow an external controller to easily control and interact with HP Instrument BASIC programs
- Allow an external controller to interact with external devices through the HP Instrument BASIC interface control
- Set and read the HP 3577B time clock

These commands augment the HP-IB mnemonics described in Chapter 3, "Remote Operation," in the HP 3577B Network Analyzer Operating Manual. You may wish to move these pages into that volume to consolidate all the available HP-IB commands into one reference.

LPR

Command syntax: LPR<sp>#I

Example Statement: OUTPUT 711;"LPR #I";

Description:

This command will download a HP Instrument BASIC program from an external controller to the instrument. The command prepares the HP 3577B to accept program lines. Any line feed between this command and the first program line must be suppressed. Each program line which follows must end in a line feed. To terminate the download, send an empty line.

DPR

Command syntax: DPR

Example Statement: OUTPUT 711;"DPR"

Description:

This command prepares the HP 3577B to upload the current HP Instrument BASIC program from the instrument to an external controller. The programs lines can then be entered into a string array. Each line will end in a line feed and an EOI will terminate the upload.

Note



To see examples of the above commands see chapter 4, "Downloading and Uploading Programs." For a better understanding of the interface see "Downloading and Uploading Programs" in chapter 6, "Interfacing with the HP-IB."

PRR

Command syntax: PRR

Example Statement: OUTPUT 711;"PRR"

Description:

This command provides remote operation functionality equivalent to the front-panel [RUN PROGRAM] key.

PRP

Command syntax: PRP

Example Statement: OUTPUT 711;"PRP"

Description:

This command will pause the program and is comparable to pressing the [SPCL FCTN] front-panel key during program execution.

PRS

Command syntax: PRS

Example Statement: OUTPUT 711;"PRS"

Description:

This command will stop the program and reset HP Instrument BASIC. This is equivalent to pressing the [LCL] key during program execution.

PRC

Command syntax: PRC

Example Statement: OUTPUT 711;"PRC"

Description:

This command is the remote operation equivalent to the front-panel [CONTINUE PROGRAM] key.

PRM

Command syntax: PRM<sp><size>;

Example Statement: OUTPUT 711;"PRM 10000;"

Description:

This command will allow you to set program stack size remotely and is equivalent to the front panel [MEMORY SIZE] function.

SCP

Command syntax: SCP

Example Statement: OUTPUT 711;"SCP"

Description:

This command provides the remote operation equivalent to the front-panel [SCRATCH PROGRAM]

SCC

Command syntax: SCC

Example Statement: OUTPUT 711;"SCC"

Description:

This command provides the remote operation equivalent to the front-panel [SCRATCH COMMON]

SCA**Command syntax:** SCA**Example Statement:** OUTPUT 711;"SCA"**Description:**

This command provides the remote operation equivalent to the front-panel [SCRATCH ALL]

BA1**Command syntax:** BA1**Example Statement:** OUTPUT 711;"BA1"**Description:**

This command provides the remote program equivalent to the front-panel key sequence [SPCL FCTN][INSTRMNT BASIC][AUTOSTRT][ON] which causes the HP 3577B to load and run a program named "AUTO_BAS" from an external disk at power on.

BA0**Command syntax:** BA0**Example Statement:** OUTPUT 711;"BA0"**Description:**

This command provides the remote program equivalent to the front-panel key sequence [SPCL FCTN][INSTRMNT BASIC][AUTOSTRT][OFF]

BLI

Command syntax: BLI

Example Statement: OUTPUT 711;"BLI"

Description:

This will list the current HP Instrument BASIC program to the analyzer display area.

LBS

Command syntax: LBS<sp><line #>;

Example Statement: OUTPUT 711;"LBS 1;"

Description:

Sets the number of the start line for listing and securing programs. Entering a zero here will cause the start line to default to the first program line.

LBE

Command syntax: LBE<sp><line #>;

Example Statement: OUTPUT 711;"LBE 10;"

Description:

Sets the number of the end line for listing and securing programs. Entering a zero here will cause the end line to default to the last program line.

BDO

Command syntax: BDO

Example Statement: OUTPUT 711;"BDO"

Description:

This command will clear the HP Instrument BASIC display. It is the remote programming equivalent to pressing the [CLEAR DISPLAY] softkey in the [INSTRMNT BASIC] menu.

HBP

Command syntax: HBP<sp><address>;

Example Statement: OUTPUT 711;"HBP 5;"

Description:

This command sets the plotter HP-IB address. It is the remote equivalent to the front-panel [LCL][PLOTTER ADDRESS] operation.

HBD

Command syntax: HBD<sp><address>;

Example Statement: OUTPUT 711;"HBD 1;"

Description:

This command sets the disk HP-IB address and is the remote equivalent to the front-panel [LCL][DISK ADDRESS] operation.

HBU

Command syntax: HBU<sp><unit number>;

Example Statement: OUTPUT 711;"HBU 1;"

Description:

This command sets the HP-IB disk unit specifier and is the remote equivalent to the front-panel [LCL][DISK UNIT] operation.

Note



For more information on system controller issues see "The HP Instrument BASIC HP-IB Model" section in chapter 6, "Interfacing with HP-IB."

STI

Command syntax: STI<sp>"<Y><sep><M><sep><D><sep><h><sep><m><sep><s>" ";"

<sep>	::-	{, : space}
<Y>	::-	year (range 0-32760)
<M>	::-	month (range 1-12)
<D>	::-	day (range 1-31)
<h>	::-	hour (range 0-12)
<m>	::-	minute (range 0-59)
<s>	::-	second (range 0-59)

Example Statement: OUTPUT 711;"STI "1990,6,26,16:19:00";"

Description:

This command allows you to set the HP 3577B internal clock. The time and date clock is used by the Instrument BASIC TIMEDATE function, and for disk catalog entries.

DTI

Command syntax: DTI

Example Statement: OUTPUT 711;"DTI"
 ENTER 711;A\$

Query Syntax: <string variable>

Returned Format: <Y>,<M>,<D>,<h>,<m>,<s>

Description:

This command allows you to read the HP 3577B's time clock. It will return 6 zeros if the time and date have not been set since power on.

Example Programs

Introduction

This chapter contains two example programs that demonstrate HP Instrument BASIC in the HP 3577B. These programs may be used "as is" to provide additional functionality to the HP 3577B, or as example programs to demonstrate specific programming techniques.

The first program, "MSE," is a mass storage example that allows a user added capabilities in handling disk files such as the ability to copy, rename and purge files, the ability to MSI to different mass storage devices or directories on a DOS file system or to catalog a disk, the ability to save and recall state data and more.

The second program, "PLOTf," allows a user to capture the plotting of the display to a file on an external disk. This can be used for generating plots at a later time or for merging with text in a word processor for report generation.

MSE

```

10 ! "MSE"
30 !
40 CALL Ms_example
50 END
60 !-----
70 Ms_example:SUB Ms_example
80 ! -----
90 ! Mass Storage Examples (MSE) subprogram
100 ! for HP 3577B Instrument BASIC.
110 !
120 ! -----
130 IF POS(SYSTEM$("SYSTEM ID"),"3577") THEN
140 !This program running in Instrument BASIC.
150 ASSIGN @Na TO 800 !8- internal HP-IB interface
160 ELSE
170 !This program running in an external controller.
180 ASSIGN @Na TO 711 !7-select code 11-address

```

Example Programs

MSE

```
190     END IF
200     CLEAR SCREEN
210     LOOP
220         ON ERROR RECOVER Handle_error
230         ! FNKey$ is a function that displays 8 softkeys
240         ! and returns the softkey selected.
250         SELECT FNKey$("MSI","CAT","COPY","PURGE","RENAME","", "more", "quit")
260         CASE "MSI"
270             CALL Do_msi
280         CASE "CAT"
290             CAT
300         CASE "COPY"
310             CALL Do_copy
320         CASE "PURGE"
330             CALL Do_purge
340         CASE "RENAME"
350             CALL Do_rename
360         CASE "more"
370             CALL Do_more(@Na)
380         CASE "quit"
390             SUBEXIT
400         END SELECT
410         !
420         IF 0 THEN !start of error handling
430 Handle_error:OFF ERROR
440             PRINT
450             PRINT ERRM$
460             BEEP
470             REPEAT
480             UNTIL "CONTINUE"-FNKey$("CONTINUE","", "", "", "", "", "", "", "")
490             CLEAR SCREEN
500         END IF !error handling
510         !
520     END LOOP
530 SUBEND
540 !
550 !-----
560 Do_more:SUB Do_more(@Na)
570     !This subprogram handles the 2nd level softkey menu.
580     LOOP
590         SELECT FNKey$("INSTRMNT DISP OFF","INSTRMNT DISP ON","SAVE
STATE","RECALL STATE","SAVE X","RECALL X","", "return")
600         CASE "INSTRMNT DISP OFF"
```

```

610     OUTPUT @Na;"GRO; CHO; TR1; DF0; TR2; DF0"
620     CASE "INSTRMNT DISP ON"
630     OUTPUT @Na;"GR1; CH1; TR1; DF7"
640     CASE "SAVE STATE"
650     CALL Do_save_state(@Na)
660     CASE "RECALL STATE"
670     CALL Do_recall_state(@Na)
680     CASE "SAVE X"
690     CALL Do_save_x(@Na)
700     CASE "RECALL X"
710     CALL Do_recall_x(@Na)
720     CASE "return"
730     SUBEXIT
740     END SELECT
750     END LOOP
760     SUBEND
770     !
780     !-----
790 Do_msi:SUB Do_msi
800     ! Subprogram to set MASS STORAGE IS
810     DIM Answer$(30)
820     CLEAR SCREEN
830     PRINT "Enter new Mass Storage Volume Specifier"
840     PRINT "surrounded by quotes (")"
850     PRINT "Example: ",703,1""
860     INPUT "New MSI ?",Answer$
870     CLEAR SCREEN
880     PRINT "MSI ";Answer$
890     MASS STORAGE IS Answer$
900     SUBEND
910     !
920     !-----
930 Do_copy:SUB Do_copy
940     DIM From$(30),To$(30)
950     PRINT
960     PRINT "Enter source file or volume "
970     PRINT "surrounded by quotes (")"
980     INPUT "source ?",From$
990     CLEAR SCREEN
1000    IF From$="" THEN SUBEXIT
1010    PRINT "Enter destination file or volume"
1020    PRINT "surrounded by quotes (")"
1030    INPUT "destination ?",To$

```

Example Programs

MSE

```
1040 CLEAR SCREEN
1050 IF To$="" THEN SUBEXIT
1060 PRINT "COPY ";From$;" TO ";To$
1070 COPY From$ TO To$
1080 SUBEND
1090 !
1100 !-----
1110 Do_purge:SUB Do_purge
1120 ! Subprogram to PURGE a file
1130 DIM Answer${30}
1140 PRINT
1150 PRINT "Enter name of file to PURGE"
1160 INPUT "file ?",Answer$
1170 CLEAR SCREEN
1180 IF Answer$="" THEN SUBEXIT
1190 PRINT "PURGE ";Answer$
1200 PURGE Answer$
1210 SUBEND
1220 !
1230 !-----
1240 Do_rename:SUB Do_rename
1250 ! Subprogram to RENAME a file
1260 DIM From${30},To${30}
1270 PRINT
1280 PRINT "Enter name of file to RENAME"
1290 INPUT "file ?",From$
1300 CLEAR SCREEN
1310 IF From$="" THEN SUBEXIT
1320 PRINT "Enter new file name"
1330 INPUT "file name ?",To$
1340 CLEAR SCREEN
1350 IF To$="" THEN SUBEXIT
1360 PRINT "RENAME ";From$;" TO ";To$
1370 RENAME From$ TO To$
1380 SUBEND
1390 !
1400 !-----
1410 Do_save_state:SUB Do_save_state(@Na)
1420 ! Subprogram to read the HP 3577B instrument setup
1430 ! state and write it to a file.
1440 DIM File${30}
1450 INTEGER State(0:1508) !3018 bytes /2
1460 PRINT
```

```

1470 PRINT "Save the instrument state in a file."
1480 INPUT "File name ?",File$
1490 IF File$="" THEN SUBEXIT
1500 CREATE BDAT File$,(3018 DIV 256)+1
1510 ASSIGN @File TO File$
1520 DISP "dumping state"
1530 CLEAR @Na
1540 OUTPUT @Na;"DCS" !dump complete state
1550 ASSIGN @Na;FORMAT OFF !binary transfer
1560 ENTER @Na;State(*) !get state
1570 ASSIGN @Na;FORMAT ON
1580 DISP "writing file"
1590 OUTPUT @File;State(*) !write state
1600 ASSIGN @File TO * !close file
1610 CLEAR SCREEN
1620 DISP "SAVE done"
1630 SUBEND
1640 !
1650 !-----
1660 Do_recall_state:SUB Do_recall_state(@Na)
1670 ! Subprogram to read a HP 3577B instrument setup
1680 ! state from a file and send it to the instrument.
1690 DIM File${30]
1700 INTEGER State(0:1508) !3018 bytes /2
1710 PRINT
1720 PRINT "Recall the instrument state from a file"
1730 INPUT "File name ?",File$
1740 IF File$="" THEN SUBEXIT
1750 ASSIGN @File TO File$
1760 DISP "reading file "
1770 ENTER @File;State(*) !get state
1780 ASSIGN @File TO * !close file
1790 DISP "loading state"
1800 CLEAR @Na
1810 OUTPUT @Na;"LCS" !load complete state
1820 ASSIGN @Na;FORMAT OFF !binary transfer
1830 OUTPUT @Na;State(*) !send state
1840 ASSIGN @Na;FORMAT ON
1850 CLEAR SCREEN
1860 DISP "RECALL done"
1870 SUBEND
1880 !
1890 !-----

```

Example Programs

MSE

```

1900 Do_save_x:SUB Do_save_x(@Na)
1910     ! Subprogram to save an X register in a file.
1920     DIM File${30}
1930     REAL Reg(0:801)
1940     INTEGER Pound_i,Reg_num
1950     PRINT
1960     PRINT "Save an X register in a file."
1970     INPUT "X register (1-8) ?",Reg_num
1980     IF (Reg_num<1) OR (Reg_num>8) THEN SUBEXIT
1990     INPUT "File name ?",File$
2000     IF File$="" THEN SUBEXIT
2010     CREATE BDAT File$, (802*8 DIV 256)+1
2020     ASSIGN @File TO File$
2030     DISP "dumping X";Reg_num
2040     CLEAR @Na
2050     OUTPUT @Na;"FM2;DX"&VAL$(Reg_num)           !request dump
2060     ASSIGN @Na;FORMAT OFF                         !binary transfer
2070     ENTER @Na;Pound_i,Reg(*)                     !read data
2080     ASSIGN @Na;FORMAT ON
2090     DISP "writing file"
2100     OUTPUT @File;Pound_i,Reg(*)                 !write data
2110     ASSIGN @File TO *                            !close file
2120     CLEAR SCREEN
2130     DISP "SAVE done"
2140 SUBEND
2150 !
2160 !-----
2170 Do_recall_x: SUB Do_recall_x(@Na)
2180     ! Subprogram to recall an X register from a file.
2190     DIM File${30}
2200     REAL Reg(0:801)
2210     INTEGER Pound_i,Reg_num
2220     PRINT
2230     PRINT "Recall an X register from a file"
2240     INPUT "File name ?",File$           !get file name
2250     IF File$="" THEN SUBEXIT
2260     ASSIGN @File TO File$               !open the file
2270     INPUT "X register (1-8) ?",Reg_num
2280     IF (Reg_num<1) OR (Reg_num>8) THEN SUBEXIT
2290     DISP "reading file "
2300     ENTER @File;Pound_i,Reg(*)         !read data from file
2310     ASSIGN @File TO *                   !close file
2320     DISP "loading x1"

```



```

2330 CLEAR @Na
2340 OUTPUT @Na;"FM2;LX"&VAL$(Reg_num) !send "load X" command
2350 ASSIGN @Na;FORMAT OFF !binary transfer
2360 OUTPUT @Na;Pound_i,Reg(*) !send data to instrument
2370 ASSIGN @Na;FORMAT ON
2380 CLEAR SCREEN
2390 DISP "RECALL done"
2400 SUBEND
2410 !-----
2420 Key:DEF FNKey$(K0$,K1$,K2$,K3$,K4$,K5$,K6$,K7$)
2430 ! This function puts up 8 softkeys and waits
2440 ! for one to be pressed, and returns it.
2450 ON KEY 0 LABEL FNFixkey$(K0$) GOTO K0
2460 ON KEY 1 LABEL FNFixkey$(K1$) GOTO K1
2470 ON KEY 2 LABEL FNFixkey$(K2$) GOTO K2
2480 ON KEY 3 LABEL FNFixkey$(K3$) GOTO K3
2490 ON KEY 4 LABEL FNFixkey$(K4$) GOTO K4
2500 ON KEY 5 LABEL FNFixkey$(K5$) GOTO K5
2510 ON KEY 6 LABEL FNFixkey$(K6$) GOTO K6
2520 ON KEY 7 LABEL FNFixkey$(K7$) GOTO K7
2530 LOOP
2540 END LOOP
2550 K0:RETURN K0$
2560 K1:RETURN K1$
2570 K2:RETURN K2$
2580 K3:RETURN K3$
2590 K4:RETURN K4$
2600 K5:RETURN K5$
2610 K6:RETURN K6$
2620 K7:RETURN K7$
2630 FNEND
2640 !
2650 !-----
2660 Fixkey:DEF FNFixkey$(Klabel$)
2670 ! This function adds spaces to softkey labels so
2680 ! words are not split between the two lines of the
2690 ! softkey label. Called by FNKey$
2700 Key_width=8 !Softkey character width
2710 DIM K$(30) !The return string
2720 IF LEN(Klabel$)>17 THEN
2730 K$=Klabel$[1,17] !Truncate if too long
2740 ELSE
2750 K$=Klabel$

```

Example Programs

MSE

```
2760     END IF
2770     IF (LEN(K$)>Key_width) THEN !if longer than 1 width
2780         P=2+POS(K$[3]," ") !find a space (after first 2 chars)
2790         SELECT P
2800         CASE <-2                !no spaces
2810             !do nothing
2820         CASE <Key_width          !space in first line
2830             K$=K$[1,P]&RPT$(" ",Key_width-P)&K$[P+1] !add spaces
2840         CASE Key_width
2850             !do nothing
2860         CASE >Key_width          !space on 2nd line, abbrev 1st line
2870             K$=K$[1,Key_width]&K$[P+1]
2880         END SELECT
2890     END IF
2900     IF LEN(K$)>2*Key_width THEN K$=K$[1,2*Key_width]
2910     RETURN K$
2920 FNEND
2930 !
2940 !-----
```

PLOTF

```

10  !   "PLOTF"
20  CALL Plotf
30  END
40  !-----
50 Plotf:SUB Plotf
60  ! -----
70  ! Subprogram to capture HP 3577B plots in
80  ! a file. The file will contain HP-GL
90  ! plotter commands which can be sent to a
100 ! plotter at a later time.
110 ! -----
120 ! If the destination disk is initialized as DOS, the HP-GL files can
130 ! read by many DOS applications such as desktop publishing programs,
140 ! and the HP 3577B plots can be merged in with reports.
150 !
160 ! Or, a Personal Computer can copy the files to an plotter connected
170 ! to the PC via RS-232.
180 !
190     DIM File$(30)           !string for file name.
200     DIM Pbuf$(18000)       !string to hold plotter commands
210                                 !largest plot is about 17K bytes.
220     !
230     ! Use the SYSTEM ID query to see if runnin in Instrument BASIC
240     IF POS(SYSTEM$("SYSTEM ID"),"3577") THEN
250         ASSIGN @Na TO 800           !8= internal HP-IB interface
260         OUTPUT @Na;"HBP 800"       !set plotter address to 800
270             !Plotter address 800 will cause the HP 3577B to
280             !send plotter data back to the Instrument BASIC
290             !program rather than to an external plotter as it
300             !normally does.
310     ELSE
320         !This program running in an external controller.
330         ASSIGN @Na TO 711           !7=select code 11=address
340     END IF
350     CLEAR SCREEN
360     PRINT "PLOT TO A FILE"
370     INPUT "File name ?",File$
380     CLEAR SCREEN
390     IF File$="" THEN SUBEXIT
400     CREATE File$,18000              !Create a DOS/HP-UX type file

```

Example Programs
PLOTf

```
410    ASSIGN @File TO File$
420    !
430    OUTPUT @Na;"PLA"           !Request plot from HP 3577B
440    ENTER @Na;Pbuf$          !Read plotter commands into array Pbuf$.
450                                !EOI will terminate the ENTER because
460                                !the HP 3577B plotter commands do not
470                                !contain the line-feed character.
480    PRINT "Writing ";LEN(Pbuf$);" bytes to ";File$
490    OUTPUT @File;Pbuf$
500    !
510    OUTPUT @File;END          !set end-of-file pointer
520    ASSIGN @File TO *        !close the file
530    CLEAR SCREEN
540    SUBEND
550    !
560    !
```

Index

A

- Aborting a program 2-7
- Active Controller
 - HP Instrument BASIC as 6-17
- Adding program lines 2-6
- Alpha entry 3-5
- Arbitrary block data 6-24
- ASSIGN 2-18, 5-7
- Autoloading a program 3-8

B

- Buffer, HP-IB 2-5

C

- Cataloging the disk 3-6
 - DOS files 4-8
- Clearing the display 2-8, 5-4
- Creating programs
 - with an external controller 4-1
 - with keystroke recording 2-1

D

- Deleting a program 2-10
- Deleting program lines 2-6
- Device selectors
 - description 6-2
 - HP-IB 6-2
 - primary address 6-2

Disk drives

- cataloging 3-6
 - connecting 3-3
 - recalling programs 3-7
 - saving programs 3-4
- ## Disk IO commands 7-5
- ASSIGN 7-6
 - CAT 7-6
 - COPY 7-7
 - CREATE 7-7
 - CREATE ASCII 7-7
 - CREATE BDAT 7-7
 - GET 7-8
 - INITIALIZE 7-8
 - MASS STORAGE IS 7-8
 - PRINTER IS 7-10
 - PURGE 7-10
 - RE-SAVE 7-10
 - RENAME 7-10
 - SAVE 7-10
- ## Disk volumes 5-6
- Display techniques 5-2 - 5-4
 - Displaying data 5-2 - 5-4
 - DOS file system 4-8, 5-6 - 5-8
 - Downloading programs 4-4, 6-24
 - Drawing graphics 5-3

E

- Editing programs 2-6, 4-2
 - with HP BASIC controller 4-2
 - with HP BASIC Language Processor 4-3
 - with MS-DOS ASCII editor 4-7

Index

ENTER

HP-IB 6-3, 6-5

Error messages

when downloading 4-4

when uploading 4-4

during program execution 2-5

F

File types 4-8

DOS 5-7, 7-5

Formatting disks 2-16, 5-6

G

GPIO 5-5

Graphics 5-3

H

HP Instrument BASIC

as Active Controller 6-17

as Non-Active Controller 6-19

with external controllers 6-20 - 6-25

HP-IB

ABORT statement 6-12

Active Controller 6-3

ATN 6-4

bus 6-1

CLEAR statement 6-11

commands 6-6

example bus sequences 6-5

extended commands 6-20

external port 6-16

general structure 6-3

internal port 6-16

interrupts 6-13

listener 6-4

LOCAL LOCKOUT statement 6-8

LOCAL statement 6-9

model 6-16 - 6-19

PASS CONTROL command 6-15

Primary address 6-2

REMOTE statement 6-7

SPOLL statement 6-15

statement summary 6-6

System Controller 6-3

talker 6-4

TRIGGER statement 6-10

unlisten 6-4

HPIB

mnemonics 2-19

I

Initializing disks 2-16, 5-6

Inserting program lines 2-6

Interrupts

HP-IB(SRQ) 6-14

K

Keystroke recording 2-1

L

LIF file system 4-2, 5-6

Listing a program 2-8 - 2-9

Loading programs

via GET 4-2

via RECALL 3-7

M

Manual Conventions 1-3

MASS STORAGE IS 5-7

Memory

allocating 2-14

examining size 2-14

N

Non-Active Controller
 HP Instrument BASIC as 6-19

O

ON INTR 6-13
OUTPUT 2-18
 HP-IB 6-3, 6-5

P

Pathnames, DOS 4-7
Pausing a program 2-7
Porting 1-2
Printing a program 2-9
Printing data 5-4
Programming
 advanced 1-2
Purging files 3-6

R

Recalling programs 3-7
Recording programs 2-1
 limitations 2-6
Renaming files 3-6
Resaving a program 3-4, 3-6
Resetting a program 2-7
Running programs 2-5 - 2-7

S

Saving programs 3-4, 4-2
Scratching a program 2-10
Securing a program 2-12
Service Request Indicators 6-16
Service routines, HP-IB 6-13

Softkeys

 setting up 5-2

SRQ

 generating external 6-16
 interrupts 6-13
 receiving 6-16

SRQ interrupts 6-14

Stack, program 2-15

Storing programs 3-4

Subdirectories, DOS 4-8

Synchronization 2-5

System Controller 6-3

T**Transferring data**

 between programs 6-21
 to a PC 5-8

Transferring programs

 by recalling 3-7
 via disk, DOS 4-7, 5-8
 via disk, LIF 4-2
 via downloading 4-3
 via uploading 4-3
 by saving 3-4

TRIG/RESET 2-5

U

Uploading programs 4-4, 6-24

V

Viewing a program 2-8