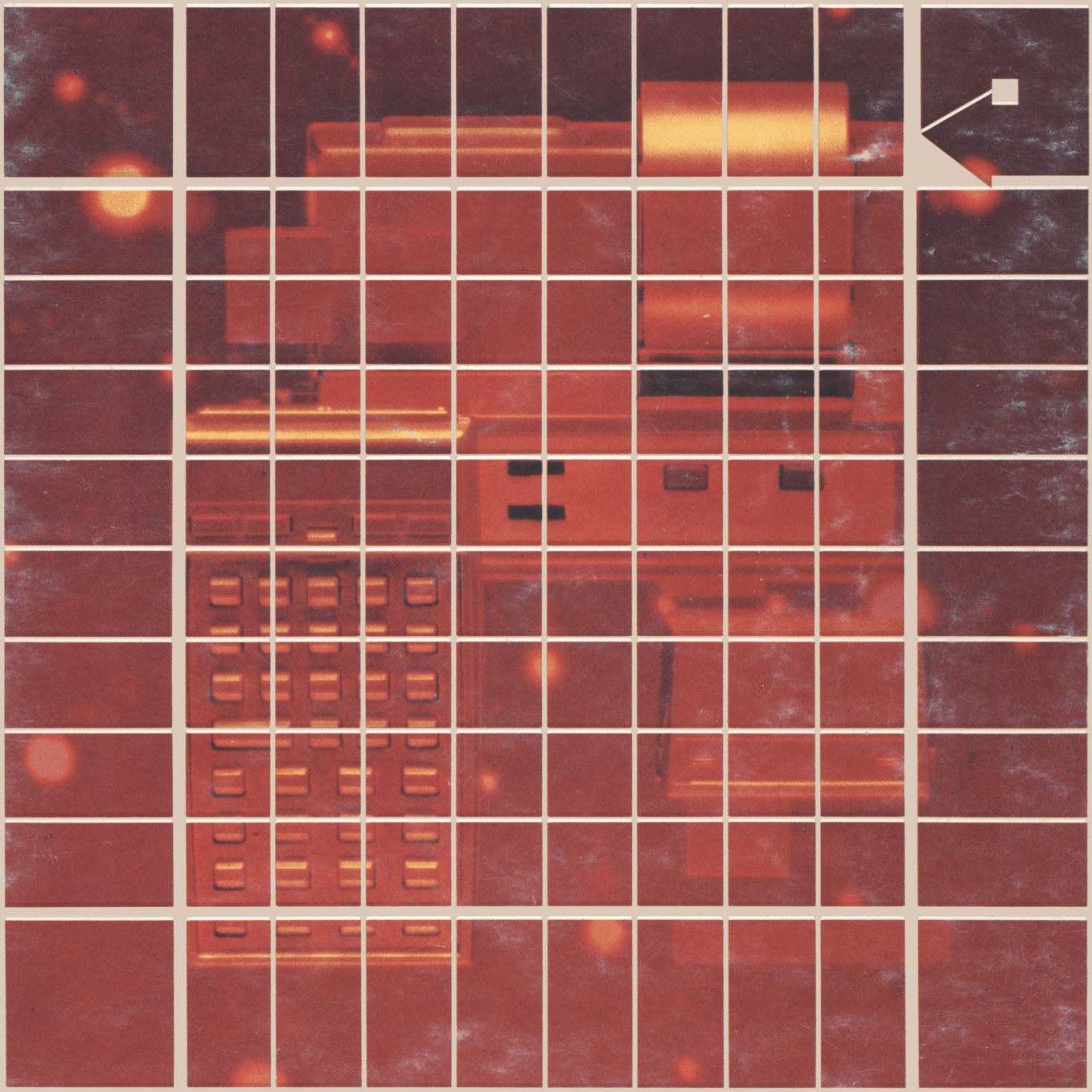
HEWLETT-PACKARD

HP 82160A

HP-IL Module

OWNER'S MANUAL



HP 82160A
HP-IL Module

Owner's Manual

February 1981

82160-90001

Printed in U.S.A.

Contents

Section 1

Getting Started

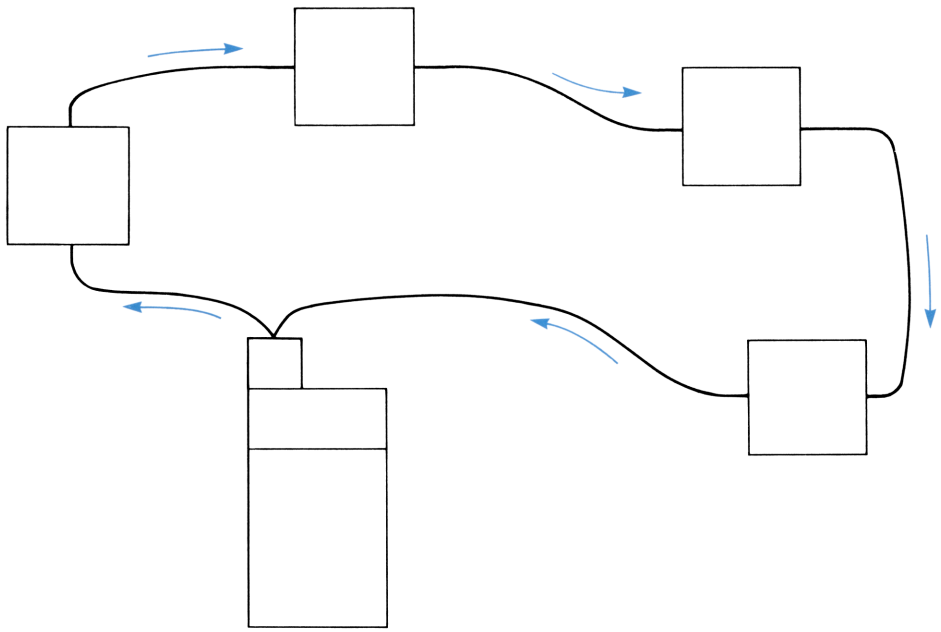
The HP 82160A HP-IL Module enables you to expand your calculator system according to your needs. This interface is compatible with all HP-41 family calculators. It allows you to connect your calculator to the Hewlett-Packard Interface Loop (HP-IL) and perform many versatile printer, mass storage, and interface control operations.

This manual describes how to connect and use your HP 82160A HP-IL Module. It explains all of the interface instructions contained in the interface module and shows how to use them. For information about the special features of each peripheral device, please refer to the owner's manual for that device.

The Hewlett-Packard Interface Loop

The Hewlett-Packard Interface Loop uses an approach that is easy to use and understand. By using the HP 82160A HP-IL Module, your calculator can interact with any compatible HP-IL peripherals—such as printers and mass storage devices.

The calculator and all devices included in the interface loop are connected together in series, forming a "communication circuit." Any information (instructions or data) that is transferred among HP-IL devices is passed from one device to the next around the circuit. If the information is not intended for a particular device, the device merely passes the information on to the next device in the loop. When the information reaches the proper device, that device responds as directed. In this way, the calculator can send information to and receive information from each device in the loop, according to the device's capability.



Connecting the Interface Loop

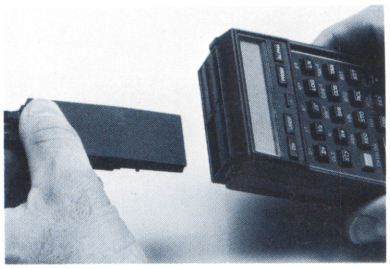
The interface loop consists of your calculator, the HP 82160A HP-IL Module, and up to 30 peripheral devices. These should be connected according to the instructions below.

CAUTION

Be sure the calculator is turned off before connecting or disconnecting the module and cable connectors. If this is not done, the calculator may be damaged or the system's operation may be disrupted.

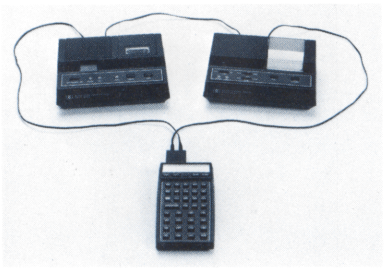
Installing the Interface Module

The HP 82160A HP-IL Module plugs into any of the calculator's ports. (If any HP 82106A Memory Modules are also plugged in, the interface module must be in a higher-numbered port than the memory modules.) Push in the module until it snaps into place. The module's switch should face down.

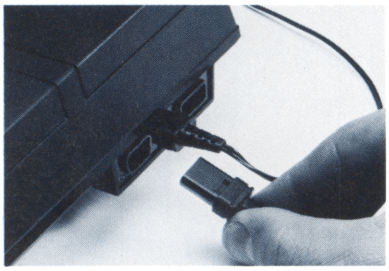


Connecting Peripheral Devices

The peripheral devices in the interface loop may be connected to the interface module in any order—but all of the interface cables must form a continuous loop. All connections are designed to ensure proper orientation.

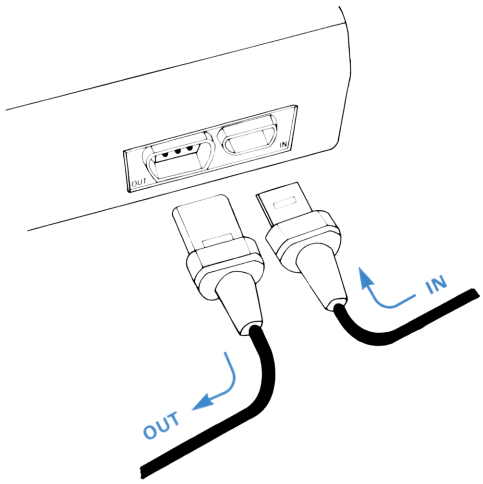


To connect a peripheral device, first turn off the calculator. Then merely disconnect the loop in one place and connect the new device into the loop at that place. *All devices must be turned on for the interface to operate properly.*



Note: If a plug-in HP 82143A Printer is connected to the calculator system, the Print Function Switch on the interface module must be set to DISABLE. Otherwise, the operation of the calculator could be disrupted. With this setting, printer operations will be executed by the plug-in printer only. If you want to print using the HP-IL print functions and an HP-IL printer, the Print Function Switch on the interface module must be set to ENABLE and a plug-in HP 82143A Printer must *not* be connected.

The connectors indicate the direction of information transfer (and the numbering of devices during operation) as shown below:



Disconnecting the Interface Loop

To remove any peripheral from the interface loop, first turn off the calculator. Then unplug the device from the loop and reconnect the loop where the device was removed.

To remove the interface module from the calculator, first turn off the calculator. Then pull the module out of the port and install a port cap on the empty port.

Using This Manual

When the interface loop is connected to your calculator, the functions implemented by the HP 82160A HP-IL Module become available for your use. These functions are grouped and discussed in three general categories: printer operations (section 2), mass storage operations (section 3), and interface control operations (section 4). Printer operations are normally used to control printer-type output devices, such as a printer or video display. Mass storage operations are intended to control devices that store and retrieve information, such as a digital cassette drive. Interface control operations provide control of other types of devices and of the interface loop itself. Press ■ CATALOG 2 to observe a list of the HP-IL functions.

For simplicity, HP-IL functions (and any other functions not on the standard calculator keyboard) are represented by single, colored keys—such as OUTA. When you want to execute a function, you can do it in two ways: by using XEQ ALPHA ***name*** ALPHA, or by assigning the function to a key using ASN and pressing that key in USER mode. (Refer to the owner's handbook for your calculator.)

Before working through any section of this manual, make sure the interface is set to its automatic operating mode. Do this by pressing XEQ ALPHA AUTOIO ALPHA. (Clearing the calculator's memory sets the interface to its automatic mode also.)

In this manual the description of each function is preceded by a summary of information required by that function. This provides a quick, visual summary of how to execute the function. For example:

CREATE X *filesize* ALPHA *filename*

This indicates that a file size must be placed in the X-register and a file name placed in the ALPHA register before you execute CREATE —from the keyboard or in a program.

If at any time an error message is displayed by the calculator, refer to appendix B for an explanation of its cause. For certain conditions the error message may not be displayed until after a short delay.

As you read through the remainder of this manual, you will discover the expansive capabilities provided by your HP 82160A HP-IL Module.

Section 2

Printer Operations

The HP 82160A HP-IL Module permits the calculator to generate printed output by connecting a printer to the interface loop. The interface module adds powerful printing, graphics, special character, and plotting capabilities to your system. To use the printer, all you do is follow the directions given in section 1 for installing the interface module and connecting the HP-IL printer to the loop. Be sure the Print Function Switch (located on the bottom of the interface module) is set to ENABLE and a plug-in HP 82143A Printer is *not* connected. Refer to the owner's manual for the printer to determine any additional preparations required for that device. The system is then ready to perform the operations described in this section.*

> The printer functions described in this section include all of the functions available on the HP 82143A Printer, an earlier, plug-in accessory. Throughout this section, printer operations are illustrated using the HP 82162A Thermal Printer, an HP-IL peripheral. The operation of the interface module and the HP 82162A Thermal Printer is almost identical to the operation of the HP 82143A Printer. In fact, programs written to use the HP 82143A Printer will operate normally using the HP 82162A Thermal Printer.†

Flags and the Printer

When you start using the printer operations discussed in this section, you will find it helpful to know how the calculator flags influence the operations.

The calculator uses up to six flags to control a printer-type device. Five of these flags (flags 12, 13, 15, 16, and 21) are user flags—that is, you can set, clear, and test them. The other flag (flag 55) is a system flag—it can only be tested. The effects of these flags are summarized below.

FLAG	SET	CLEAR
Flag 55: Printer Existence	Indicates a printer is connected to system.	Indicates no printer is connected to system.
Flag 21: Printer Enable	Performs printer operations normally.	Ignores printer operations in programs only.
Flag 12: Double Wide	Prints and accumulates characters double width.	Prints and accumulates characters normal width.
Flag 13: Lowercase	Prints and accumulates letters in lowercase.	Prints and accumulates letters in uppercase (except [a] through [e]).
Flags 15 and 16: Print Mode (not used for HP 82162A Thermal Printer)	Flag 15 / Flag 16 / Print Mode clear / clear / MAN (*manual*) clear / set / NORM (*normal*) set / clear / TRACE set / set / TRACE with stack option	

*Refer to the owner's manual for the printer to determine which types of operations, if any, cannot be performed by your printer. Other printer-type (output) devices, such as video displays, may generate output using the printer operations described in this section.

†The few operational differences between the HP 82162A Thermal Printer and the HP 82143A Printer will be noted throughout this section.

Remember that all special-purpose user flags (flags 11 through 20) are cleared each time the calculator is turned on. However, once you set any of the printer flags, the printer will operate accordingly until you clear the flag or turn off the calculator. Notice that the conditions of the flags do not affect the calculator's display.

The Printer Existence Flag (flag 55) is used to indicate if a printer is connected to the system. Each time the calculator is turned on, flag 55 is set if a printer is detected; flag 55 is cleared if no printer is detected at turn-on. (A printer can be detected only if it is turned on.) After that time, flag 55 is set whenever a printer function is executed in a program or any function is executed from the keyboard *and* a printer is present. (Flag 55 is cleared only when the calculator is turned on and no printer is detected.) Since flag 55 is a system flag, you can only test it.*

The Printer Enable Flag (flag 21) is used to control printing in *programs* that contain specific print functions. Flag 21 has no effect on print functions executed from the keyboard. Generally, while flag 21 is set, printing functions in a program will print normally. While flag 21 is clear, printing is suppressed in a program. (For more detailed information about the effects of flag 21, refer to Printing During Program Execution at the end of this section.) If and when flag 55 is set by the calculator, flag 21 is *automatically* set to match the status of flag 55. If a printer is connected when the calculator is turned on, flags 55 and 21 are automatically set at that time; otherwise, both flags are cleared. If print functions in a program do not operate as expected, check flag 21.*

The Double Wide Flag (flag 12) is a special-purpose user flag that is used to control how characters are printed on the printer paper. While flag 12 is set, all characters are printed double width—all dots in the printed output are printed double. In addition, other printer operations use double-wide characters when flag 12 is set.

The Lowercase Flag (flag 13), another special-purpose user flag, also controls the way characters are printed. While flag 13 is set, all letters are printed in lowercase form. (Other characters are not affected by flag 13.) In addition, other printer operations use lowercase letters.

The Print Mode Flags (flags 15 and 16) determine the output mode of a printer—unless it has a Print Mode Switch, such as the HP 82162A Thermal Printer has. You can control what and how information is printed by selecting the proper print mode:

- In MAN (*manual*) mode, the printer is idle and does not print unless you or a program executes a print function. Listings of programs are printed left-justified in this mode.
- In NORM (*normal*) mode, the printer prints numbers and ALPHA strings that are keyed in, function names as they are executed from the keyboard, and output from print functions. Running programs print only output from print functions and PROMPT. Listings of programs are printed right-justified in this mode.
- In TRACE mode, the printer prints numbers and ALPHA strings that are keyed in, function names, intermediate and final answers, and output from print functions. Program listings are printed in a special "packed" or condensed form in this mode.
- In TRACE mode with stack option, the printer operates the same as in regular TRACE mode—except that the contents of the four stack registers (X, Y, Z, and T) are printed after each operation. (This mode is not available with the HP 82162A Thermal Printer.)

To obtain the printed output shown in examples in this section, set the HP 82162A print mode switch as indicated in each example.

---

*If you turn on an HP-IL printer *after* turning on the calculator and then immediately test either flag 55 or flag 21 from the keyboard, NO will be displayed for that test. However, both flags will be *set* when you execute the test function with a printer connected. A subsequent test will result in a YES display.

Standard Printing Operations

Information stored in your calculator may be printed directly using the printing functions described below. The actual format of the printed output will be determined by the type of printer that you use. Refer to the owner's manual for the printer for a description of the printer's operation.

Using Calculator Functions That Print

There are five standard calculator functions that will automatically print as well as perform their normal functions while the HP-IL printer is connected and set to the proper print mode. These functions are VIEW, AVIEW, PROMPT, ADV, and CATALOG. They perform their normal functions (described in the owner's handbook for your calculator) as well as the printing operations. These five functions are available even when the interface loop is not connected to the calculator.

VIEW *nn* R_{nn} *data*

The VIEW (*view register contents*) function causes the contents of the specified register to be displayed. With a printer connected, VIEW causes the contents of the specified register to be printed as well as displayed. The register may be specified directly or indirectly, as explained in the owner's handbook for your calculator.

AVIEW ALPHA *data*

The AVIEW (*ALPHA view*) function displays the contents of the ALPHA register. When a printer is connected, AVIEW displays and prints the contents of the ALPHA register.

PROMPT ALPHA *prompt*

The PROMPT function displays the contents of the ALPHA register and halts program execution in a running program. While a printer is connected, the execution of PROMPT also prints the ALPHA prompt—but only if the printer is in NORM or TRACE mode.

ADV

The ADV (*advance*) function advances the printer paper a single line each time it is executed, either in a program or from the keyboard. It also prints any information being held in the printer. If no printer is connected to the system, ADV is ignored.

> For the HP 82162A Thermal Printer, pressing the PAPER ADVANCE key on the printer advances the paper a single line. When the calculator is in PRGM mode, pressing the PAPER ADVANCE key on the printer inserts an ADV function into program memory.

CATALOG *n*

The CATALOG function lists the contents of any of the three calculator catalogs: the user catalog (**CAT 1**), the extension catalog (**CAT 2**), and the standard function catalog (**CAT 3**). While a printer is connected and set to TRACE mode, CATALOG displays and prints the catalog. The listing for CATALOG 1 also prints—next to the **END** of each program—the number of "bytes" that each program occupies in program memory. (Refer to the owner's handbook for your calculator for a discussion of "bytes.") The CATALOG function is not programmable—it cannot be keyed in as part of a program.

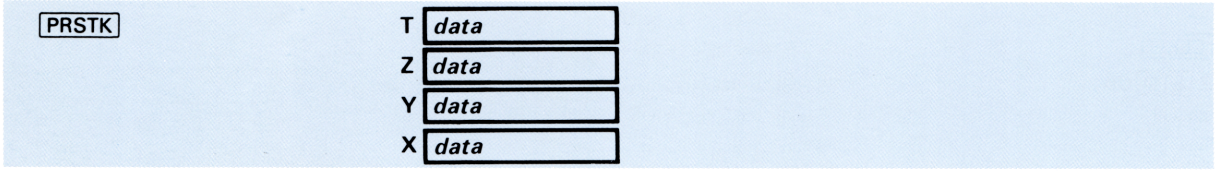
Printing the Display

For the HP 82162A Thermal Printer, the PRINT key on the printer provides a method for printing the contents of the display at any time, in any print mode. While the calculator in not in ALPHA mode, pressing PRINT prints the contents of the X-register (in PRGM mode, pressing PRINT inserts a [PRX] into the program). While the calculator is set to ALPHA mode, pressing PRINT prints the contents of the ALPHA register (in PRGM/ALPHA mode, pressing PRINT inserts a [PRA] into the program).

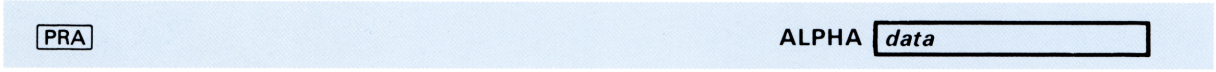
Printing Registers

Six functions allow you to print the contents of certain registers in the calculator. Any listing can be terminated by pressing [R/S].

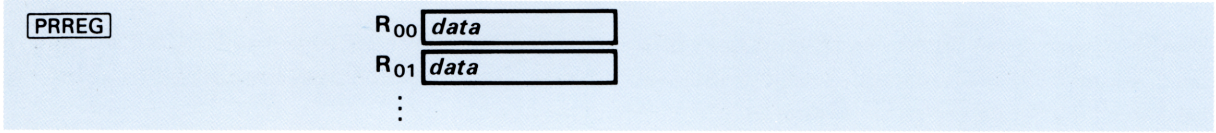
The [PRX] (*print X*) function prints the contents of the X-register and marks it with ***.



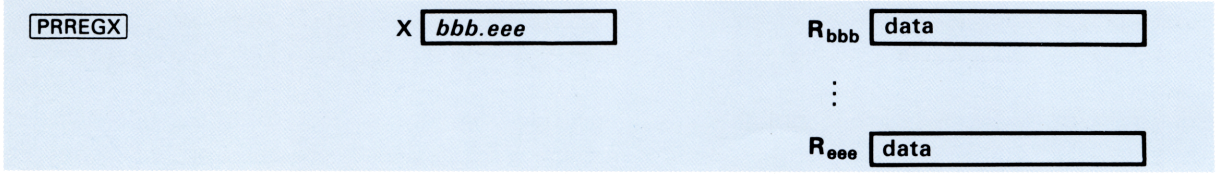
The [PRSTK] (*print stack*) function prints and labels the contents of the automatic memory stack in T, Z, Y, X order.



The [PRA] (*print ALPHA*) function prints the contents of the ALPHA register. The printed output is left-justified—it lines up with the left margin.



The [PRREG] (*print registers*) function prints and labels the contents of all currently allocated data storage registers, beginning with R_{00}. When a register contains ALPHA characters, its contents are enclosed in quotes when printed, like this **R01 = "ABC"**.



The [PRREGX] (*print registers as directed by X*) function provides you with control over which registers you print. [PRREGX] uses a number in the X-register to control the printing operation. Before executing [PRREGX], place a control number in the X-register using the following format:

bbb.eee

where ***bbb*** is the beginning data storage register address and ***eee*** is the ending data storage register address. The ***bbb*** portion can be one to three digits; the calculator uses the first three digits of the ***eee*** portion, which follows the decimal point. For example, specify registers R_{03} thru R_{07} using 3.007 in the X-register.

PR Σ

$R_{\Sigma 1}$ *data*
⋮
$R_{\Sigma 6}$ *data*

The PR Σ (*print statistics registers*) function enables you to print the contents of the currently defined statistics registers. (Refer to the owner's handbook for your calculator for information about Σ REG and how statistics registers are defined.) PR Σ prints the contents of all six statistics registers.

Example of printing registers:

Keystrokes

SIZE 017	Allocates 17 storage registers.
CLRG	Clears all storage registers.
2 STO 05	Stores 2.0000 in R_{05}.
1.005	Specifies R_{01} through R_{05}.
PRREGX	Prints registers.
■ SF 12	Sets double-wide flag.
ALPHA STRETCH ALPHA	Enters ALPHA string.
PRA	Prints ALPHA register.
■ CF 12	Specifies normal width.

HP 82162A: MAN

```
R01=  0.0000
R02=  0.0000
R03=  0.0000
R04=  0.0000
R05=  2.0000

STRETCH
```

Printing Programs

Two functions print programs that are stored in program memory: PRP and LIST. The print mode determines the format in which program lines are printed. You can terminate the printing operation at any time by pressing R/S. These two functions are not programmable.

PRP ***name***

The PRP (*print program*) function prints a specific program stored in program memory. When you execute PRP, the calculator prompts you for the name of the program you wish to print. Simply key in the name of the program (by pressing ALPHA ***name*** ALPHA) and printing will begin at the first line of the named program. If you press ALPHA ALPHA (do not specify a program name) in response to the prompt, the printer will print the program to which the calculator is presently positioned—beginning at its first line.

LIST ***nnn***

The LIST function prints a specified number of lines of a program. First, position the calculator to the desired program and then to the line where you wish printing to begin. Then execute LIST. When prompted, key in a three-digit number specifying the number of lines you wish to print.

Example of printing programs: The program listings below show the three format options specified by the print mode. The program name is CHARS. (You may wish to key in and run this program—it produces a list of all printer characters. Later in this section you will learn more about the functions used in this program.)

Keystrokes

PRP	Prints program CHARS.
ALPHA CHARS ALPHA	

HP 82162A: MAN

```
01♦LBL "CHARS"
02 0.127
03 STO 00
04 FIX 0
05♦LBL 00
06 RCL 00
07 INT
08 ACX
09 ACCHR
10 ADV
11 ISG 00
12 GTO 00
13 FIX 4
14 END
```

TRACE

```
                PRP "CHARS"

 01♦LBL "CHARS"
0.127  STO 00  FIX 0

 05♦LBL 00
RCL 00  INT  ACX  ACCHR
ADV  ISG 00  GTO 00
FIX 4  END
```

NORM

```
                PRP "CHARS"

01♦LBL "CHARS"
   02 0.127
   03 STO 00
   04 FIX 0

   05♦LBL 00
   06 RCL 00
   07 INT
   08 ACX
   09 ACCHR
   10 ADV
   11 ISG 00
   12 GTO 00
   13 FIX 4
   14 END
```

Keystrokes

■ GTO	
ALPHA CHARS ALPHA	Locates CHARS program.
■ GTO · 005	Locates line 005.
LIST 006	Prints six lines.

HP 82162A: MAN

```
05♦LBL 00
06 RCL 00
07 INT
08 ACX
09 ACCHR
10 ADV
```

Printing Status and Key Assignments

Because the calculator system has a reassignable keyboard and many flags, you may wish to review the calculator's internal conditions. Using PRKEYS and PRFLAGS, you can print this information at any time.

PRKEYS

The PRKEYS (*print key assignments*) function, when executed in a program or from the keyboard, prints the keycode of each reassigned key followed by the name of the program or function assigned to each key. A keycode is a row-column code of a key's location on the keyboard. The keycode for shifted key locations are prefixed with a – (minus sign).

PRFLAGS

The PRFLAGS (*print flags*) function, when executed in a program or from the keyboard, prints the following information:

Number of data storage registers (**SIZE** = ***nnn***).
Location of first statistics register (Σ = ***nnn***).
Trigonometric mode (**DEG**, **RAD**, or **GRAD**).
Display format (**FIX** ***n***, **SCI** ***n***, or **ENG** ***n***).
Status of all flags (**F** ***nn*** **SET** or **F** ***nn*** **CLEAR**).

Example of printing calculator information:

Keystrokes		**HP 82162A:** MAN
PRFLAGS	Prints flags and other information.	STATUS: SIZE= 017 Σ= 11 DEG FIX 4 FLAGS: F 00 CLEAR F 01 CLEAR F 55 SET

Accumulating Printer Output

Seven functions allow you to build up—or accumulate—information to be printed, and then print it. Using these functions, you can control the characters that are printed and how they are formatted.

These functions (and the graphics functions discussed later in this section) require a special set of storage registers, or cells, in the printer. These cells are collectively called the *print buffer*.

For the HP 82162A Thermal Printer, the print buffer contains 101 cells. Depending upon the actual data accumulated and the operations performed, the print buffer will hold fewer than 101 actual printing characters.*

The states of the double wide flag (flag 12) and the lowercase flag (flag 13) determine how information is accumulated and later printed. These flags were discussed earlier in this section. For example, whenever flag 12 is set, double-wide characters are accumulated in the print buffer. By using these flags, you can accumulate and print any mix of character types you want.

Accumulating Characters

The following three functions accumulate characters into the print buffer. Each character occupies one cell in the buffer.

ACA ALPHA *data*

The ACA (*accumulate ALPHA*) function copies all of the characters from the ALPHA register and transfers them into the print buffer. Executed in a running program or from the keyboard, ACA adds the ALPHA characters at the end of the characters already in the print buffer.

*The plug-in HP 82143A Printer has 44 cells in its print buffer—57 less than the buffer in the HP-IL HP 82162A Thermal Printer. This will cause the buffer to be filled and printed much later for the HP-IL printer than for the plug-in printer.

Keystrokes **HP 82162A:** MAN

ADV	Advances the paper (and prints whatever is already in the print buffer).
ALPHA ABCDE ALPHA	Places the letters ABCDE in the ALPHA register.
ACA	Accumulates the contents of the ALPHA register into the print buffer without printing.
ALPHA FGHI ALPHA	Places the letters FGHI in the ALPHA register.
ACA	Adds the new contents of the ALPHA register to the contents of the print buffer. Nothing has been printed yet.
ADV	Prints the accumulated contents of the print buffer and advances the paper.

ABCDEFGHI

Here is what happened when you accumulated the ALPHA characters using ACA and then executed ADV.

	ALPHA Register	Print Buffer
ALPHA ABCDE ALPHA	ABCDE	
ACA	ABCDE	A B C D E
ALPHA FGHI ALPHA	FGHI	A B C D E
ACA	FGHI	A B C D E F G H I
ADV	FGHI	

Printer: ABCDEFGHI

ACX X *data*

The ACX (*accumulate X-register*) function operates the same way as ACA except that ACX uses information from the X-register for its accumulation. Whenever you execute ACX, a copy of the characters in the X-register is accumulated into the print buffer.

When you use ACX, the entire formatted number is accumulated into the print buffer—this includes the number (according to the current display format) and the space for the sign of the number (blank for positive, minus sign for negative). If you wish to omit the initial space for positive numbers, use ARCL and ACA to accumulate numbers. You can do this by pressing ■ ARCL · X in ALPHA mode—recalling the contents of the X-register into the ALPHA register—and then using ACA to accumulate those ALPHA characters. ARCL will copy a minus sign, but not a leading blank for a positive number.

X-Register		ALPHA Register		Print Buffer
1.2345			ACX	1.2345
−1.2345			ACX	−1.2345
1.2345	ARCL	1.2345	ACA	1.2345
−1.2345	ARCL	−1.2345	ACA	−1.2345

ACCHR x *code*

The ACCHR (*accumulate character*) function accumulates one standard printer character into the print buffer. The number in the X-register specifies the particular character that is accumulated. Characters are numbered from 0 through 127. ACCHR allows you to accumulate many more characters than are contained on the calculator's keyboard.

For the HP 82162A Thermal Printer, the 128 standard characters and their corresponding numbers are shown in the following listing.* (The program in the example on page 14 produces this list of characters and numbers.)

0.♦	32.	64.@	96.`
1.☺	33.!	65.A	97.a
2.☻	34."	66.B	98.b
3.♥	35.#	67.C	99.c
4.♦	36.$	68.D	100.d
5.♣	37.%	69.E	101.e
6.♠	38.&	70.F	102.f
7.•	39.'	71.G	103.g
8.◘	40.(	72.H	104.h
9.○	41.)	73.I	105.i
10.◙	42.*	74.J	106.j
11.♂	43.+	75.K	107.k
12.♀	44.,	76.L	108.l
13.♪	45.-	77.M	109.m
14.♫	46..	78.N	110.n
15.☼	47./	79.O	111.o
16.►	48.0	80.P	112.p
17.◄	49.1	81.Q	113.q
18.↕	50.2	82.R	114.r
19.‼	51.3	83.S	115.s
20.¶	52.4	84.T	116.t
21.§	53.5	85.U	117.u
22.▬	54.6	86.V	118.v
23.↨	55.7	87.W	119.w
24.↑	56.8	88.X	120.x
25.↓	57.9	89.Y	121.y
26.→	58.:	90.Z	122.z
27.←	59.;	91.[	123.{
28.∟	60.<	92.\	124.|
29.↔	61.=	93.]	125.}
30.▲	62.>	94.↑	126.~
31.▼	63.?	95._	127.⌂

Each printer-type device has a set of standard characters that it normally uses. However, the HP 82160A HP-IL Module automatically replaces some of these characters with other characters. For the HP-IL HP 82162A Thermal Printer, characters 10 and 13 are replaced by characters 0 and 124, respectively. For all other printer-type devices, characters 10, 13, and 126 are replaced by characters 0, 124, and 28, respectively.

*For the plug-in HP 82143A Printer, character 124 is | . For the HP-IL HP 82162A Thermal Printer, character 124 is ¦ ; the HP-IL printer has no ¦ character.

Example of accumulating characters, normal and double-wide:

Keystrokes		HP 82162A: MAN
ADV	Prints whatever is in buffer.	
■ SF 12	Sets double-wide flag.	
15	The number for Φ.	
ACCHR	Accumulates Φ.	
■ CF 12	Specifies normal width.	
2.6 CHS	Places -2.6 in X-register.	
ALPHA		
SPACE = SPACE	ALPHA string.	
■ ARCL · X	Appends X-register.	
ALPHA		
ACA	Accumulates ALPHA register.	
ADV	Prints buffer and advances paper.	Σ = -2.6000

Accumulating Spaces

SKPCHR x *number*

The SKPCHR (*skip character*) function accumulates spaces—skips character positions—in the print buffer. The number of spaces to skip is specified by the number placed in the X-register. This function makes it easier for you to control the format of information you accumulate and then print—without having to enter individual spaces.

Printing the Contents of the Print Buffer

After you have the desired characters accumulated into the print buffer, you can instruct the printer to print the contents of the buffer. The buffer is printed left to right—the first character you place in the buffer is printed on the left and the last character you place in the buffer is printed on the right. Once any information in the print buffer is printed, that information is no longer in the buffer. Flags 12 and 13 (double wide and lowercase flags) have no effect on how the buffer is printed—they only affect how the contents are accumulated.

ADV

The ADV (*advance*) function causes the print buffer to be printed right-justified—that is, the information is lined up with the right margin of the paper.

PRBUF

The PRBUF (*print buffer*) function prints the buffer left-justified—lined up with the left margin.

Other operations that will cause the contents of the print buffer to be printed are, in general, any operations that normally cause printing. This includes executing functions such as PRX, PRA, and VIEW. However, you should remember that these functions may also change the contents of the buffer in the course of their normal execution. Also, in NORM or TRACE modes, the print buffer will be printed by any operation normally listed in these modes.

When the print buffer is filled, the printer *automatically* prints one line of the buffer to provide additional space.

You can clear the contents of the print buffer *without printing* by turning the printer off and then on. All information in the print buffer will be lost.

Keystrokes **HP 82162A:** MAN

ALPHA RIGHT SPACE	
ALPHA	Places characters in ALPHA register.
ACA	Accumulates ALPHA characters.
125 ACCHR	Accumulates character 125.
ADV	Prints right-justified.
ACA	Accumulates ALPHA characters.
ACCHR	Accumulates character 125.
PRBUF	Prints left-justified.

```
                    RIGHT →
RIGHT →
```

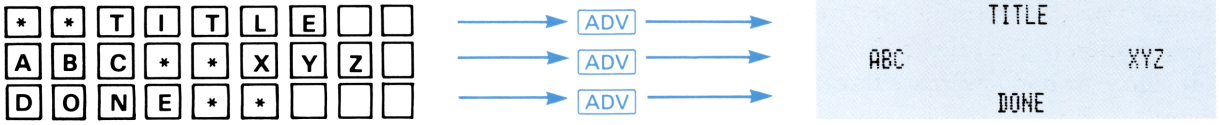
Formatting the Printed Output

The ADV and PRBUF functions normally let you control how accumulated information is printed—right- or left-justified. FMT gives you the ability to center or separate the contents of a printed line.

FMT

The FMT (*format*) function accumulates a *format specifier* into two consecutive cells in the print buffer. The format specifier causes the printing to be centered on the line if the specifier occupies the first or last cells that are printed. The specifier causes the printing to be separated (and left- and right-justified) when it occurs between other accumulated output. The presence of a format specifier supercedes the formatting normally done by the ADV and PRBUF functions—both of these print functions will print a formatted line the same way. For example, if ** represents the format specifier accumulated by FMT:

Print Buffer **Printed Output**



Centered lines are positioned to the nearest dot position in the printed output. If the printed output cannot be exactly centered on the line, the extra space (one dot wide) is placed where the format specifier is accumulated—to the left or right of the characters.

Keystrokes **HP 82162A:** MAN

ALPHA SPAN= ALPHA	Enters ALPHA label.
38.5	Enters number.
FMT	Leading format specifier.
ACA	Accumulates label.
ACX	Accumulates number.
PRBUF	Prints the buffer.
ACA	Accumulates label.
FMT	Internal format specifier.
ACX	Accumulates number.
PRBUF	Prints the buffer.

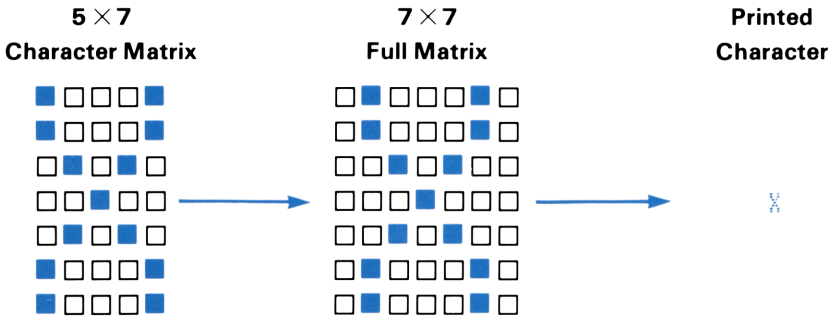
```
SPAN= 38.5000
```

```
SPAN=          38.5000
```

Graphics (HP 82162A Thermal Printer Only)

Special graphics operations allow you to control precisely what is printed—even the exact shape of what is printed. You can build and print your own special characters and noncharacter graphics using the HP 82162A Thermal Printer.

All standard characters are defined in terms of dots in a five-by-seven matrix. To print a specific character, the printer prints the dots that make up that character. In order to provide spaces between characters, each character sits in a seven-by-seven matrix.

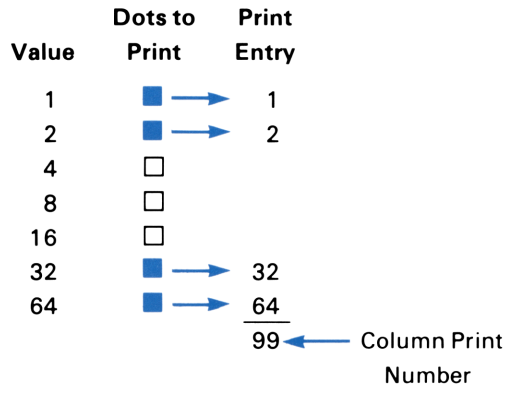


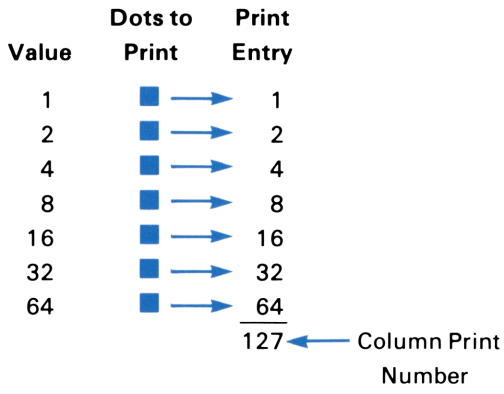
Using the graphics operations, you can actually tell the printer which dots to print in each column of a printed line.

Specifying a Column of Dots

A *column print number* is used to specify which dots are to be printed in a particular column. Column print numbers are used in some of the graphics operations to control the exact form of the printing.

Each dot in the column is assigned a numeric value, as shown below. Simply add the numbers for the dots you want to print in the column—the sum is the column print number. The column print number can be 0 through 127.





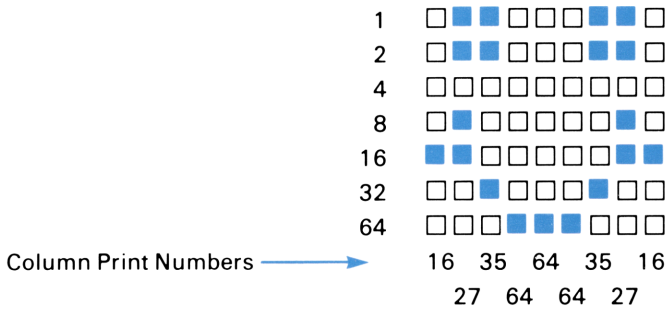
Accumulating Columns

ACCOL X *c-p number*

The ACCOL (*accumulate column*) function accumulates a single dot column into the contents of the print buffer. The specific dot pattern is determined by the column print number in the X-register. Each accumulated column occupies one cell in the print buffer. You can accumulate columns until the buffer is filled—when it is filled, it is automatically printed.

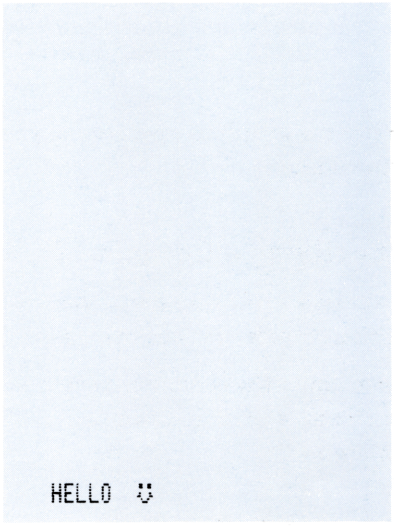
Remember that in NORM and TRACE modes, the print buffer will be printed by any operation normally listed in these modes.

The next example accumulates and prints the nine-column pattern shown below.



Keystrokes **HP 82162A:** MAN

ADV	Clears the print buffer.
ALPHA	
HELLO SPACE SPACE	
ALPHA	
ACA	Accumulates the ALPHA data.
16 ACCOL	Accumulates first column.
27 ACCOL	
35 ACCOL	
64 ACCOL	
ACCOL	Uses same column print number.
ACCOL	Uses same column print number.
35 ACCOL	
27 ACCOL	
16 ACCOL	Accumulates last column.
PRBUF	Prints the buffer.



Skipping Columns

SKPCOL X *number*

The SKPCOL (*skip column*) function accumulates a specified number of skipped columns into the print buffer. This allows you to generate blank columns and obtain the printed output you desire. The number in the X-register indicates the number of columns to skip. You can skip from 0 up to 167 columns using SKPCOL. (For a 24-character line, skipping 168 columns is the same as a paper advance.)

Building Special Characters

If you want to print special characters that are not among the standard character set, you can use the special-character functions to build and store custom characters and symbols. Like other characters, the special characters you create must fit in the standard seven-by-seven dot matrix.

BLDSPEC X *c-p number* Y *reserved*

The BLDSPEC (*build special character*) function uses up to seven column print numbers, one at a time, to define the dot pattern for your special character. The column print numbers are entered in the X-register and processed and combined using the X- and Y-registers of the calculator's stack. Before starting to build

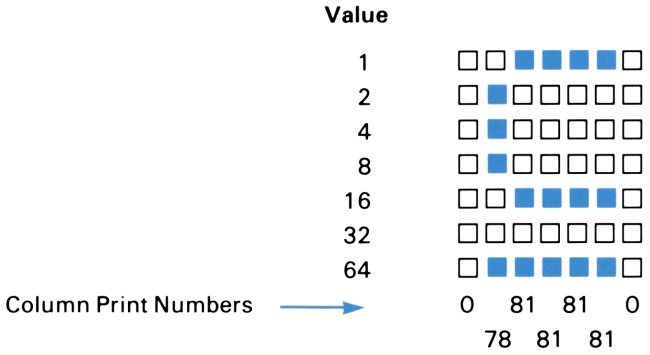
a character, be sure to clear the X- and Y-registers (press 0 ENTER↑). Execute BLDSPEC for each of the seven column print numbers, left to right. The character representation is then present in the X-register—the display on the calculator is otherwise meaningless. This representation can be stored in any register for use at any time. If you specify more than seven print column numbers, earlier numbers are lost—only the last seven numbers will be used to form the character. If you specify fewer than seven numbers, blank columns will remain in the undefined left part of the character.

Note: Standard characters have columns 1 and 7 blank to provide spacing between printed characters. However, by using BLDSPEC, you can build special characters without spacing.

ACSPEC X *representation*

The ACSPEC (*accumulate special character*) function accumulates a special character into the print buffer. The X-register must contain the character representation generated by the BLDSPEC process. However, the representation may be recalled from a register where it has been previously stored. An accumulated special character uses seven cells in the print buffer.

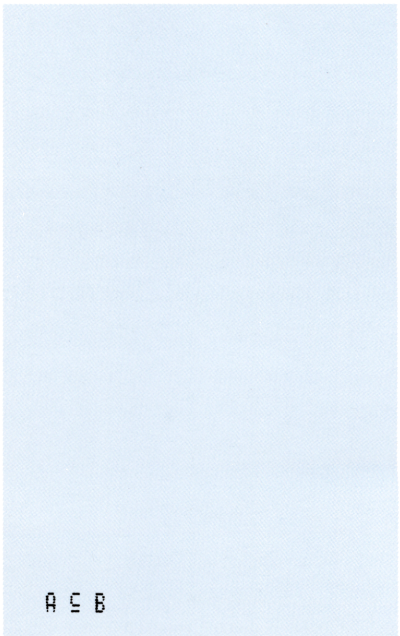
Here is an example showing how to build, store, and use a special character.



Keystrokes

0 ENTER↑	Clears X- and Y-registers.
0 BLDSPEC	Builds column 1.
78 BLDSPEC	
81 BLDSPEC	
81 BLDSPEC	
81 BLDSPEC	
81 BLDSPEC	
0 BLDSPEC	Builds column 7.
STO 01	Stores representation.
←	Clears X-register.
ALPHA A SPACE	
ALPHA	
ACA	Accumulates ALPHA.
RCL 01	Recalls representation.
ACSPEC	Accumulates special character.
ALPHA SPACE B	
ALPHA	
ACA	Accumulates ALPHA.
PRBUF	Prints the buffer.

HP 82162A: MAN



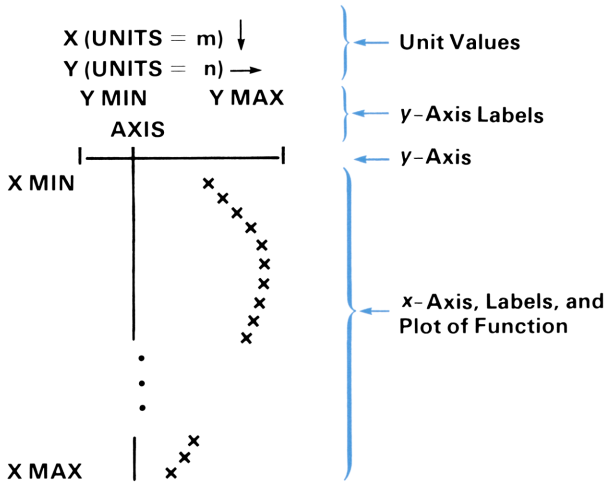
Plotting (HP 82162A Thermal Printer Only)

The HP 82160A HP-IL Module allows you to perform plotting operations with an HP 82162A Thermal Printer. Using the five plotting operations, you can plot any valid single-valued mathematical function. The function that you plot is often specified by a program that you enter into the calculator—refer to the owner's handbook for your calculator for programming information.

The HP 82162A Thermal Printer must be in MAN print mode in order to properly execute the plotting operations. Otherwise, the print buffer will be printed prematurely.

Printer Plots

The two basic plotting operations— PRPLOT and PRPLOTP —are designed to generate plots having the type of format shown below. Other plotting operations produce individual parts of a plot, allowing you to customize your plots according to your needs.



The unit values are automatically determined by certain plotting routines. Unit values permit the axis labels to be printed in a shorter form. The X unit value applies to the numbers down the left side of the plot. The Y unit value applies to the numbers across the top of the plot. For example, a unit value of **E-2.** means the numbers along the axis are expressed in units of 10^{-2}.

The y-axis labels must be specified either from the keyboard or in a program—according to the plotting operation being performed. Y MIN and Y MAX represent the minimum and maximum values of y that will be plotted. AXIS represents the position on the y-axis where the x-axis is to be drawn. (If NONE or any other ALPHA characters are specified for AXIS, no x-axis will be drawn.)

The x-axis labels must be specified using three numbers—X MIN, X MAX, and X INC. X MIN and X MAX represent the minimum and maximum values of x for which the plot will be drawn. X INC determines the size of the step between values of x: if X INC is positive, x increases in steps of X INC; if X INC is negative, x has that many equal increments from X MIN to X MAX. For example:

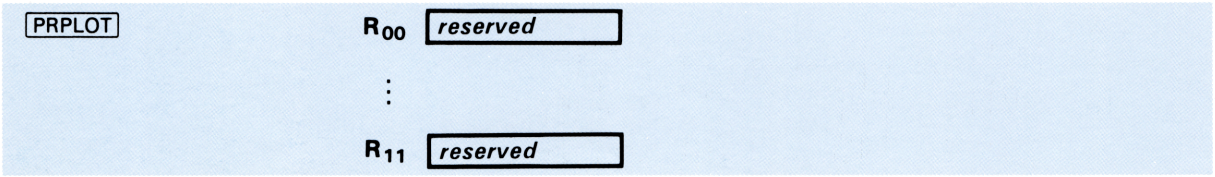
Parameters		Function plotted at these *x* values	
X MIN	0		
X MAX	360		
X INC	10	0, 10, 20, 30, 40, ..., 360	(10-unit increments)
X MIN	0		
X MAX	360		
X INC	-10	0, 36, 72, 108, 144, ..., 360	(10 increments)

The y function values that are plotted are computed by the function program that you provide. It can be *any* program that uses a number in the X-register and returns a corresponding y function value to the X-register.* The function program should not alter registers R_{00} through R_{11}—these registers are used by the plotting routine. (Register R_{06} contains a copy of the x value placed into the X-register.)

Plotting with Special Characters

All but one of the plotting operations use register R_{03} to specify the plotting symbol. If the contents of this register are numeric, the plotting symbol is a small "x." But if R_{03} contains a character representation generated by [BLDSPEC]—as discussed on page 21—that special character is used to plot the function. If you do not specify a special plotting symbol, clear R_{03} (press 0 [STO] 03) to ensure that some unwanted symbol is not used.

Interactive Plotting



The [PRPLOT] (*print plot*) operation prints a plot of your programmed function. [PRPLOT] prompts you for all of the information it needs to construct the axes of the plot. Then it uses the program that you have already stored in program memory—your function program—to plot the actual values.

Example of plotting: Plot the function

$$y = \sin x$$

(This example assumes that the calculator is in degrees mode and the allocation of data storage registers is [SIZE] 017.)

First, enter a program that computes the function.

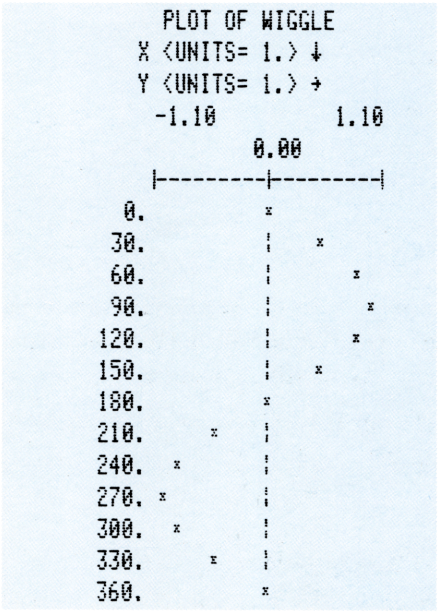
Keystrokes	Display	
0 [STO] 03	0.0000	Clears register R_{03}.
[PRGM]		
■ [GTO] [·] [·]		
■ [LBL]		
[ALPHA] WIGGLE [ALPHA]	01 LBL"WIGGLE	Names the program.
[SIN]	02 SIN	Computes sin x and leaves the result in X.
■ [GTO] [·] [·]		Ends the program.
[PRGM]	0.0000	

When you execute [PRPLOT], it prompts you for plotting information. (Plot the function for x from 0° to 360° in steps of 30°. To allow room for y values close to −1 and +1, draw the y-axis from −1.1 to 1.1. Draw the x-axis at $y = 0$.)

*When executed from a program, the [PRPLOT] routine uses either (a) three subroutine levels or (b) two subroutine levels more than the number of subroutine levels in your function program, whichever is greater.

Prompt	Keystrokes	
	PRPLOT	Executes plotting routine.
NAME?	WIGGLE R/S	Function name (label): WIGGLE
Y MIN?	1.1 CHS R/S	Minimum *y* value: −1.1
Y MAX?	1.1 R/S	Maximum *y* value: 1.1
AXIS?	0 R/S	*x*-axis location: 0
X MIN?	0 R/S	Minimum *x* value: 0
X MAX?	360 R/S	Maximum *x* value: 360
X INC?	30 R/S	*x* increment value: 30

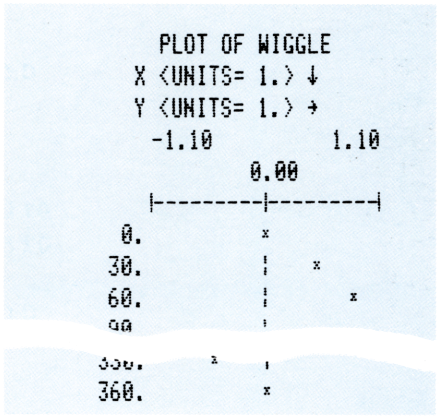
HP 82162A: MAN



PRPLOT automatically supplies each *x* value, runs your function program (WIGGLE), and plots the calculated point using the *y* value returned in the X-register.

Printer

X-Register (*x* values)		X-Register (*y* values)	
PRPLOT	0	WIGGLE	0.0000 PRPLOT
PRPLOT	30	WIGGLE	0.5000 PRPLOT
PRPLOT	60	WIGGLE	0.8660 PRPLOT
⋮			
PRPLOT	360	WIGGLE	0.0000 PRPLOT



The PRPLOT program is written using normal calculator functions. If you wish to look at the program, you can load it into program memory using COPY. The PRPLOT program requires 77 registers of program memory. Once the PRPLOT program is in program memory, you can add new lines and delete existing lines. The changes you make cannot, however, be recorded back into the interface module. Instead, you must use the altered program as it is in memory, or record it on a mass storage device. You can print the complete program on the printer. Appendix C contains an annotated listing of PRPLOT.

If PRPLOT is allowed to execute to completion, the display format will be set to FIX 4, regardless of the format that was in effect when PRPLOT began execution. In addition, flag 12 (double wide) will be cleared.

Registers used by PRPLOT include registers R_{00} through R_{11}. These registers store plotting information. Before you execute PRPLOT, you must be sure that storage registers are allocated to at least SIZE 012. In particular, the contents of register R_{03} specify the character used to plot the function—the plotting symbol. Refer to page 24.

Programmable Plotting

The interactive plotting operation, PRPLOT, can be executed in a running program. It operates just as it does when executed from the keyboard—by prompting you for the required information. However, there is another plotting routine that operates similarly to PRPLOT, but does not get its information by prompting for it. It is more suitable for programmed plotting since it does not require you to key in information during execution.

PRPLOTP

R_{00}	*Y MIN*
R_{01}	*Y MAX*
R_{03}	*plot symbol*
R_{04}	*AXIS*
R_{08}	*X MIN*
R_{09}	*X MAX*
R_{10}	*X INC*
R_{11}	*NAME*

The PRPLOTP (*print plot, programmable*) operation generates a plot of a programmed function. PRPLOTP obtains the needed plotting parameters from registers R_{00} through R_{11}. Before executing PRPLOTP, either from the keyboard or in a running program, store the plotting parameters in the corresponding registers as shown above. These values are used by PRPLOTP to construct the plot, just as PRPLOT uses them. Be sure your function program does not alter the contents of registers R_{00} through R_{11}.

The PRPLOTP program consists of normal calculator functions and can be copied into program memory using COPY. The program requires 77 registers of program memory. Except for having no prompting operations, PRPLOTP is the same program as the PRPLOT program.

Plotting a *y*-Axis

PRAXIS

R_{00}	*Y MIN*
R_{01}	*Y MAX*
R_{02}	*column width*
R_{04}	*AXIS*

The PRAXIS (*print axis*) operation is a part of the PRPLOT program that can be executed separately to print the *y*-axis. PRAXIS scales, prints, and labels the *y*-axis. This operation can be used for special applications where you want to construct your own plotting routines.

As shown above, the PRAXIS operation uses the contents of four registers to construct and label the *y*-axis, as well as determine the unit value for the *y*-axis. The column width stored in R_{02} specifies the number of dot columns to be spanned by the *y*-axis—how many columns wide the plot should be. (This parameter should not be greater than 168.) The PRAXIS operation modifies the column width parameter in R_{02} to include information for plotting the *x*-axis, stores it in R_{02}, and displays it in the X-register. This revised parameter is suitable for use by subsequent plotting operations, discussed next. The other plotting parameters are the same as described earlier.

For the conditions shown below, PRAXIS modifies the column width in R_{02} and plots the y-axis as follows:

R_{00}	**Y MIN**	-2	
R_{01}	**Y MAX**	5	
R_{02}	**Width**	140	→ 140.025
R_{04}	**AXIS**	-0.8	

```
Y (UNITS= 1.) →
-2.00          5.00
 -0.80
|---|---------------|
```

R_{00}	**Y MIN**	100	
R_{01}	**Y MAX**	2000	
R_{02}	**Width**	155	→ 155.074
R_{04}	**AXIS**	1000	

```
Y (UNITS= E 3.) →
0.10              2.00
          1.00
|---------|---------|
```

The PRAXIS operation uses two subroutine levels in its execution. After the axis has been printed, the two additional levels become available again. PRAXIS clears flag 12 (double wide) and leaves the display in FIX 4 format.

Plotting a Function Value

Two functions allow you to plot a single numeric value on one line of printer output. By using one of these functions, you can have complete control over how data points are plotted. One function uses data storage registers that are compatible with the PRAXIS operation. The second function uses the stack registers for specifying plotting information, making all of the storage registers available for your use. The information needed to plot the point is

Function Value
Y MIN
Y MAX
Modified Column Width (*nnn.aaa*)

The function value is the y value that is to be plotted. Y MIN and Y MAX define the limits of the y-axis, as discussed earlier. The modified column width—mentioned briefly in connection with PRAXIS—defines both the dot-column width of the y-axis and the column number for the x-axis. The format for the modified column width is ***nnn.aaa***. The ***nnn*** portion tells how many columns wide the y-axis is—the same as with PRAXIS. The ***aaa*** portion (after the decimal point) specifies which column of the y-axis, from 001 to ***nnn***, the x-axis bar should be printed in. (If ***aaa*** is zero, the x-axis bar is automatically printed at the column closest to $y = 0$.) If the modified column width is negative, the x-axis bar is not printed. You should note that the modified column width parameter is *automatically* calculated and stored in R_{02} when PRAXIS is executed—you don't need to compute it again.

REGPLOT

X *y value*

R_{00}	*Y MIN*
R_{01}	*Y MAX*
R_{02}	*nnn.aaa*
R_{03}	***plot symbol***

The REGPLOT (*register plot*) function uses registers R_{00} through R_{03} to plot the y value in the X-register.

STKPLOT

T	*y value*
Z	*Y MIN*
Y	*Y MAX*
X	*nnn.aaa*

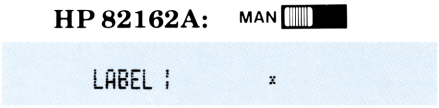
The STKPLOT (*stack plot*) function obtains its plotting information from the stack registers only—no data storage registers are used. The plotting symbol is automatically a small "x."

Labels for single-line plotting can also be printed by [REGPLOT] and [STKPLOT]. You can create labels by first accumulating the label in the print buffer using any of the functions described earlier in this section.

When [REGPLOT] or [STKPLOT] is executed, the buffer is printed immediately to the left of the plot column. Note that the plot is right-justified—lined up at the right edge. For a constant column width on a sequence of plot lines, the right ends of the labels will also line up.

Keystrokes	Display	
5 [CHS] [ENTER↑]	-5.0000	Function value.
10 [CHS] [ENTER↑]	-10.0000	Y MIN
0 [ENTER↑]	0.0000	Y MAX
120.001	120.001_	Modified Column Width with x-axis in column 1.
[ALPHA] LABEL [SPACE]	LABEL _	
[ALPHA]	120.0010	
[ACA]	120.0010	
[STKPLOT]	120.0010	

HP 82162A:



Disabling the Printer Operations

The Print Function Switch, located on the bottom of the interface module, enables or disables the printer operations in the interface loop. When the switch is set to ENABLE, the printer operations described in this section are executed in their normal manner using a printer connected to the interface loop. When the switch is set to DISABLE, no printer operations described in this section are executed by the interface loop.

The printer operations implemented by the HP 82160A HP-IL Module include all of those for the HP 82143A Printer—an earlier, plug-in accessory. In order to achieve this compatibility, the instructions programmed into each of these two devices cannot be distinguished by the calculator.

If a plug-in HP 82143A Printer is connected to the calculator system, the Print Function Switch must be set to DISABLE. Otherwise, the operation of the calculator could be disrupted. With this setting, printer operations will be executed by the plug-in printer only.

If you want to print using an HP-IL printer, the Print Function Switch must be set to ENABLE and a plug-in HP 82143A Printer must *not* be connected.

Programming and the Printer Operations

Printing During Program Entry

While you are keying a program into the calculator in PRGM mode, you can use the printer to print a record of each line you enter. Simply set the print mode to TRACE or NORM—lines are automatically printed as you enter them.

Printing During Program Execution

Flag 21—mentioned earlier in this section—affects printer operation during program execution. Flag 21 doesn't affect printing functions executed from the keyboard. The table below lists the effects of flag 21—the printer enable flag—on operations performed in a running program.

	FLAG 21	
	SET	**CLEAR**
Printer Operations	Operation is normal if printer is present. **NO PRINTER** displayed if no printer is present. (**NONEXISTENT** displayed if no interface is present or if it is set to DISABLE.)	Ignored. (**NONEXISTENT** displayed if no interface is present or if it is set to DISABLE.)
VIEW and AVIEW	Display is printed and execution is not halted if printer is present. Display is not printed and execution is halted if no printer is present.	Display is not printed and execution is not halted.
ADV	Paper is advanced if printer is present. Ignored if no printer is present.	Ignored.

If the print mode is set to TRACE during program execution, the printer will print the operation on each line along with any intermediate and final results calculated.* In this mode, you will notice that program execution slows significantly to allow the printer to keep up with the execution. TRACE mode execution is a good way to debug your program. However, programs will execute faster in NORM or MAN mode. In these modes, printer functions in your program control what is printed and when.† The contents of the print buffer are accumulated and printed normally by your program in NORM and MAN modes.

Almost all printer operations discussed in this section can be executed in a running program—only PRP, LIST, and CATALOG cannot be programmed. Be sure that any parameters needed by a print function are placed in the proper registers before the function is executed.

Further Programming Information

Section 5, Programming and the Interface Loop, discusses other topics related to programs using HP-IL operations—including printer operations.

*For the plug-in HP 82143A Printer, changing the setting of the print mode switch during a running program causes the new print mode to be implemented immediately. For HP-IL printers, a change in the print mode is not implemented until the next printing operation is performed.

†The calculator interacts with the interface module during program execution, even if the program does not contain HP-IL operations. When the interface module is plugged in, program execution will be slightly slower than if the module is unplugged.

Section 3

Mass Storage Operations

The mass storage operations available in the HP 82160A HP-IL Module permit you to store and retrieve information conveniently. By connecting a mass storage device to your Hewlett-Packard Interface Loop, you greatly expand the storage capacity of your calculator system. Follow the directions given in section 1 for installing the interface module and connecting the HP-IL mass storage device. Refer to the owner's manual for that device to determine any additional preparations that may be needed. The system is then ready to perform the operations described in this section.*

Throughout this section, mass storage operations are illustrated using the HP 82161A Digital Cassette Drive.

The Storage Medium

A mass storage device typically stores and retrieves information on a removable, interchangeable storage medium—such as a tape cassette. Each collection of information that is recorded on the storage medium is given an ALPHA name and is called a *file*. Because this information is generally transferred between the medium and registers in the calculator, the basic unit of information within a file is called a *register*. A *record* is a unit of storage capacity that is equal to from 32 to 37 registers, depending upon the type of information stored.

File names can be any string of up to seven ALPHA characters. If longer strings are used, only the first seven characters will be recognized and used by the calculator. No two files on a medium can have the same file name.

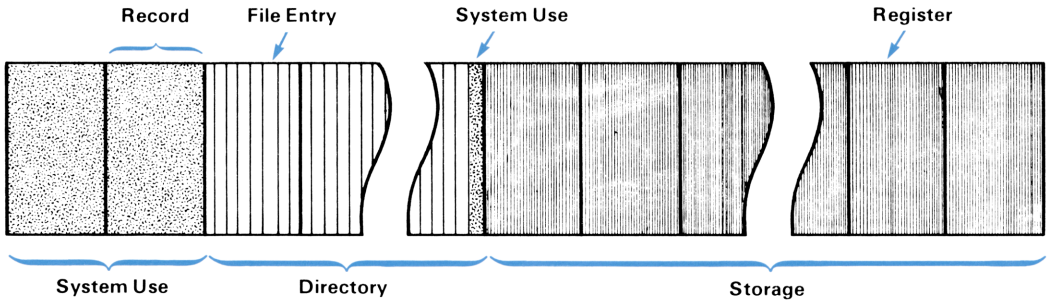
NEWM *nnn*

The NEWM (*new medium*) function initializes the storage medium. Each medium must be initialized at least once to establish on the medium a directory space and a format in which information will be recorded. When you execute NEWM, the calculator prompts you for the number of file entries you want to allocate to the directory space on the medium—any number up to 447. Each file that you record on the medium requires one entry in the directory. The directory space created by NEWM consists of an integral number of records, with each record containing eight entries. However, the last entry in the directory is reserved for the system and is not available. As a result, the directory always accomodates one less than a multiple of eight file entries—and at least as many entries as you specify. In addition, two records on the medium are reserved for system use. All remaining space on the medium is available for information storage.

Note: A smaller directory space permits faster access to files stored on the medium.

Any information previously stored on the medium will be erased when this command is executed. NEWM is not programmable.

*Refer to the owner's manual for the mass storage device to determine which types of operations, if any, cannot be performed by that device.



For the HP 82161A Digital Cassette Drive, for example, the tape cassette has a total capacity of 512 records. If you specify 075 for the number of entries, the directory will occupy 10 records—(75 + 1)/8 = 9.5 ➔ 10. This leaves 500 records (512 − 10 − 2) available for storing about 16,000 registers. For this device, NEWM takes about 3 minutes to execute.

DIR

The DIR (*directory*) function displays (in sequential order) the name of each file currently stored on the medium, the type of information stored in that file, and any special options associated with the file. The type of file is indicated by a two-letter code: **PR** (program), **DA** (data), **KE** (key assignment), **ST** (status), and **WA** ("write-all" information).* Special options are indicated following the file type: **S** (secure), **P** (private program), and **A** (automatic execution). (These file types and options are discussed later in this section.) DIR uses the ALPHA register to display the information for each file. DIR also appends the length of each file (in registers) in the ALPHA register, but does not display this information.

If a printer device is also connected to the system, the DIR function also prints directory information. In addition to the file names, types, and special options, the printed directory contains the length of each file (in registers).

You can stop the directory listing at any time by pressing R/S. The most recent entry is preserved in the ALPHA register.

Example of directory listings: Assume that a tape cassette in the HP 82161A Digital Cassette Drive already has a number of files existing on it. Determine its directory with no printer connected to the system.

Keystrokes	Display		
DIR	EXER	PR	Program file EXER.
	TEST1	DA	Data file TEST1.
	TEST2	DA,S	Data file TEST2 (secure).
	EXKEYS	KE	Key assignment file EXKEYS.
	0.0000		X-register (assumed cleared).
ALPHA	EXKEYS	KE	Last directory entry...
	E	9	...nine registers long.
ALPHA	0.0000		

Alternately, with an HP 82162A Thermal Printer connected:

Keystrokes	Display	
DIR	EXER	PR
	TEST1	DA
	TEST2	DA,S
	EXKEYS	KE
	0.0000	

HP 82162A: MAN

```
   NAME    TYPE     REGS
   EXER    PR         24
   TEST1   DA         32
   TEST2   DA,S       64
   EXKEYS  KE          9
```

*Two other file types may be indicated: **AS** (ASCII) and **??** (unknown type). These types of files can't be created or used by the HP 82160A HP-IL Module.

Storing and Retrieving Programs

Any program stored in your calculator can easily be stored on the storage medium and loaded into the calculator at a later time.

Storing a Program

WRTP	ALPHA *prgmname,filename*
WRTP	ALPHA *,filename*
WRTP	ALPHA *prgmname*

The WRTP (*write program*) function copies a program from the calculator's program memory onto the medium. The contents of the ALPHA register specify the program to be copied and the file name where it will be stored. An existing program file with the same name as the specified file name will be replaced by the WRTP function—otherwise, a new file will be created. If the calculator is positioned in the program you wish to copy, the program name does not have to be specified—and the first ALPHA character should be a comma. If you do not specify a file name in the ALPHA register, the file name will be the same as the program name. Current key assignments for labels in the program will also be recorded in the file.

WRTPV	ALPHA *prgmname,filename*
WRTPV	ALPHA *,filename*
WRTPV	ALPHA *prgmname*

The WRTPV (*write program—private*) function operates much the same as the WRTP function above. However, the WRTPV function sets a privacy flag and stores this information on the medium with the program. Whenever a program is stored with the WRTPV function, it can only be retrieved from the medium and executed—it cannot be listed, edited, or recorded on any medium. A private program is indicated in a directory listing by a P following the file type.

Retrieving a Program

Two functions permit you to copy a program file from the medium into the calculator's program memory. They differ primarily in where the program is placed in memory.

READP	ALPHA *filename*

The READP (*read program*) function copies a program from the storage medium into the calculator's program memory. The contents of the ALPHA register specify the name of the file to be copied from the medium. The program in the specified file *replaces* the last program in program memory.* By replacing the last program, you may not need an excessive number of registers in program memory. When you execute READP, the calculator remains at its former position in program memory, unless it was positioned in the last program—then it is positioned at the first line of the new program. Any key assignments recorded with the program will become active if READP is executed in USER mode.

---

*If you press ■ GTO · · before executing READP, the calculator creates a blank program space at the end of program memory. If you then execute READP, the copied program will replace the last, blank program—leaving other stored programs intact.

READSUB ALPHA *filename*

The READSUB (*read subroutine*) function operates similarly to the READP function, except that the program is placed in program memory *after* the last program. This function is particularly useful in a program that copies a subroutine program from a mass storage device, executes that subroutine, and then continues the main program. READSUB does not change the position of the calculator in program memory.

Example of storing and retrieving a program: After entering a program into your calculator, store it on the medium, clear it from program memory, and then recall it from the medium.

Keystrokes	Display	
■ GTO · ·	0.0000	Creates new program space. (Display assumed cleared.)
PRGM		
■ LBL		
ALPHA AREA ALPHA	01 LBL"AREA	Sample program AREA.
■ x^2	02 X↑2	
■ π	03 PI	
×	04 *	
PRGM	0.0000	X-register.
ALPHA ,ML	,ML_	Specifies current program and filename ML.
ALPHA	0.0000	
WRTP	0.0000	Program AREA stored in file ML.
CLP ALPHA AREA	CLP AREA_	Clears program AREA.
ALPHA	0.0000	
ALPHA ML	ML_	Specifies filename ML.
ALPHA	0.0000	
READSUB	0.0000	Copies program AREA at end of program memory.
■ GTO ALPHA AREA	GTO AREA_	Positions calculator to AREA.
ALPHA	0.0000	
PRGM	01 LBL"AREA	First line of retrieved program.
PRGM	0.0000	

Storing and Retrieving Data

A collection of data can conveniently be stored on the medium and recalled when needed. Using the functions described below, the entire collection—or any portion—can be stored or recalled.

CREATE X *filesize* ALPHA *filename*

The CREATE function allocates a portion of the medium for a data file and fills all registers with zero values. The number in the X-register specifies the number of registers to be allocated in the new data file. The contents of the ALPHA register specify the name of the file. (If a file with the specified name exists on the medium, DUP FL NAME is displayed and no new file is created.)

SEEKR X *register* ALPHA *filename*

The SEEKR (*seek register*) function positions the storage medium to a specific register within a data file. This permits data to be stored and retrieved from individual registers within a file using the WRTRX and

READRX functions described below. The ALPHA register specifies the name of the data file. The X-register specifies which register in the file should be accessed. Registers within a data file are numbered consecutively starting from 0.

Whenever you store or read data, the medium is left positioned after the last file register used. If the medium is left in the desired position for a subsequent WRTRX or READRX operation, you don't need to reposition it.

Storing Data in a File

WRTR ALPHA *filename*

The WRTR (*write registers*) function copies data from registers in the calculator into a data file that already exists on the medium. (Use CREATE to set up the data file.) The contents of the ALPHA register specify the name of the file to be filled with the data. Copying begins from register R_{00} in the calculator and continues sequentially until all of the calculator's registers have been copied. Data is stored in the file starting at file register 0.

WRTRX X *bbb.eee*

The WRTRX (*write registers according to X*) function copies data from specific registers in the calculator onto the medium. The medium must be positioned to the file and register where the copying is to begin—by SEEKR or a previous WRTRX, WRTR, READRX, or READR. Registers are filled sequentially in the file. The number in the X-register is interpreted as ***bbb.eee***, where the ***bbb*** portion indicates the first calculator register to be copied and the ***eee*** portion (following the decimal point) indicates the last register to be copied. For example, specify registers R_{03} thru R_{07} using 3.007 in the X-register. If ***eee*** is less than ***bbb***, only R_{bbb} is copied.

ZERO ALPHA *filename*

The ZERO function fills all registers in an existing data file with zero values. The ALPHA register specifies the name of the file to be filled. This function is useful for initializing a data file that contains unknown values. (Note that the CREATE function establishes a *new* data file that contains all zero values.)

Reading Data From a File

READR ALPHA *filename*

The READR (*read registers*) function copies data from a file on the medium into registers in the calculator. The contents of the ALPHA register specify the data file to be copied. Data is retrieved from the file starting at file register 0. File registers are copied sequentially into registers in the calculator, beginning with register R_{00}. Copying stops when all file registers are copied or all storage registers are filled.

READRX X *bbb.eee*

The READRX (*read registers according to X*) function copies data from a data file into specific registers in the calculator. Before executing READRX, the medium must be positioned to the file and register to be read first—by SEEKR or a previous WRTRX, WRTR, READRX, or READR. The contents of the X-register are interpreted as ***bbb.eee***, where ***bbb*** is the first register in the calculator to be filled and ***eee*** is the last register to be filled. If ***eee*** is less than ***bbb***, only R_{bbb} is copied. Copying stops if the last storage register is reached.

Example of storing and retrieving data: This example copies three numbers from R_{04} thru R_{06} into a data file, and then retrieves one of the numbers into R_{02}.

Keystrokes	Display	
14 [STO] 04	14.0000	Stores 14 in R_{04}.
25 [STO] 05	25.0000	Stores 25 in R_{05}.
36 [STO] 06	36.0000	Stores 36 in R_{06}.
[ALPHA] NUM	NUM_	Specifies file name NUM.
[ALPHA]	36.0000	
10	10_	Specifies 10 registers.
[CREATE]	10.0000	Creates data file NUM with 10 registers.
0	0_	Specifies file register 0.
[SEEKR]	0.0000	Positions medium to file register 0 in file NUM.
4.006	4.006_	Specifies registers R_{04} through R_{06}.
[WRTRX]	4.0060	Copies specified registers into registers 0 thru 2 in current file.
0 [STO] 02	0.0000	Clears R_{02}.
1	1_	Specifies file register 1.
[SEEKR]	1.0000	Positions medium to file register 1 in file NUM.
2	2_	Specifies R_{02} only.
[READRX]	2.0000	Copies one file register (register 1) to R_{02}.
[RCL] 02	25.0000	Recalls contents of R_{02}.

Storing and Retrieving Key Assignments

Any standard calculator functions or any functions contained in any plug-in application module or other system extension can be assigned to the calculator keyboard. Using two mass storage operations, you can store and access these key assignments. (Key assignments for labels in programs you have written and stored in program memory are not included in these operations, although these assignments can be stored in program files, as discussed on page 33.)

[WRTK] **ALPHA** *filename*

The [WRTK] (*write key assignments*) function copies all current key assignments of standard system functions onto the medium. The ALPHA register specifies the name of the file. An existing key assignment file with the same name as the specified file name will be replaced when you execute [WRTK] — otherwise, a new file will be created.

[READK] **ALPHA** *filename*

The [READK] (*read key assignments*) function implements all key assignments recorded in the key assignment file. [READK] cancels existing key assignments for standard functions. The contents of the ALPHA register specify the key assignment file.

Storing and Retrieving Calculator Status

You can set up certain conditions in your calculator by using the status functions. Calculator status consists of:

- The contents of the X, Y, Z, T, LAST X, and ALPHA registers.
- The states of user flags 00 through 43.
- The location of the first statistics register.
- The number of data storage registers allocated.

The two status functions are programmable.

WRTS ALPHA *filename*

The WRTS (*write status*) function copies the current status of the calculator onto the medium. An existing status file with the same name as the specified file name will be replaced when you execute WRTS—otherwise, a new file will be created.

READS ALPHA *filename*

The READS (*read status*) function sets the calculator's status according to the information in the specified status file. If the number of allocated data storage registers is decreased by READS, the contents of the excess registers are lost. (If the calculator does not have enough available registers to allocate to data storage, **SIZE ERR** will be displayed. In this case, the storage allocation is not changed, but all of the other status conditions will be implemented.)

You should use care when using READS in a program. The process of reallocating the data storage registers also eliminates the record of pending subroutine returns. That is, a subroutine will not automatically return to the main program after READS is executed. READS is best used in a main program—one in which no returns are pending.

Storing and Retrieving All Calculator Information

The entire contents of the calculator—and memory modules—can be saved on the storage medium. This allows you to reset your calculator to a known state—just as it was when you recorded this information. All status information, key assignments, data storage registers, and program memory are duplicated using the following operations. For example, you can reproduce all conditions required by a program by using a "write-all" file.

WRTA ALPHA *filename*

The WRTA (*write all*) function copies all calculator information onto the medium. The ALPHA register specifies the name of the "write-all" file. An existing "write-all" file with the same name as the specified file name will be replaced when you execute WRTA—otherwise, a new file will be created.

READA ALPHA *filename*

The READA (*read all*) function sets the calculator to the conditions recorded in the "write-all" file. The contents of the ALPHA register specifies the name of the "write-all" file. Because all conditions of the calculator are duplicated by this operation, the calculator system must be set up exactly as it was when the information was stored—including plug-in extensions and peripherals.

Working With Files

Additional mass storage functions allow you to protect, alter, and check your stored files.

Protecting Files

SEC ALPHA *filename*

The SEC (*secure*) function protects a file from being erased, renamed, or altered, such as by recording other information in its location. The ALPHA register specifies the name of the file to be secured. (The NEWM function, however, initializes the entire medium and does *not* preserve secured files.) A secured file is indicated in a directory listing by an **S** following the file type.

UNSEC ALPHA *filename*

The UNSEC (*unsecure*) function cancels the protection given to a secured file. The ALPHA register specifies the file. If you wish to change a secured file in any way, you must first execute UNSEC.

Altering Files

RENAME ALPHA *oldname,newname*

The RENAME function replaces the former name of a file with a new name. The ALPHA register specifies the old name followed by the new name, with the names separated by a comma.

PURGE ALPHA *filename*

The PURGE function removes a file from the medium. The contents of the ALPHA register specify the name of the file. The information in the file is erased and its entry in the directory is removed. Its physical space on the medium becomes empty and becomes available for storing a new file which can fit into the empty space—otherwise, the space is not used.

Checking Files

VERIFY ALPHA *filename*

The VERIFY function checks a file to ensure that the information in that file can be read—that is, that the medium is good and that no perturbations have caused improper information to be recorded on and read from the medium. The ALPHA register specifies the name of the file that is checked. If the file can be read, the X-register is returned to the display. If the file cannot be read, MEDM ERR or DRIVE ERR is displayed—such a file should be recorded again, possibly on a new medium.

Example of working with files: Assume that a tape cassette in the HP 82161A Digital Cassette Drive already has a number of files existing on it. Verify, rename, and protect the first file and erase the second file.

Keystrokes	Display		
DIR	EXER	PR	Displays directory.
	TEST1	DA	
	TEST2	DA,S	
	EXKEYS	KE	
	0.0000		(Assumes display is cleared.)
ALPHA EXER	EXER_		Specifies filename EXER.
ALPHA	0.0000		
VERIFY	0.0000		File EXER can be read.
ALPHA	EXER		
■ APPEND ,MAG	EXER,MAG_		Specifies old filename EXER and new filename MAG.
ALPHA	0.0000		
RENAME	0.0000		Renames EXER as MAG.
ALPHA MAG	MAG_		Specifies filename MAG.
ALPHA	0.0000		
SEC	0.0000		Protects file MAG.

Keystrokes	Display		
ALPHA TEST ■ 1	TEST1_		Specifies filename TEST1.
ALPHA	0.0000		
PURGE	0.0000		Erases file TEST1.
DIR	MAG	PR,S	Displays new directory.
	TEST2	DA,S	
	EXKEYS	KE	
	0.0000		

Using Multiple Mass Storage Devices

The HP 82160A HP-IL Module can interact with more than one mass storage device as though they were extensions of the first device. That is, all mass storage devices may be accessed when you execute almost any mass storage function. However, this capability is in effect only when the interface loop is in its automatic operating mode. (Interface modes are discussed in section 4.)

For example, if you attempt to copy a program file into your calculator, the calculator will search the medium in each device until it locates a file with the specified name. If you try to store a file, the calculator checks all of the media for the specified file name—and then, for a new file, searches for the first medium with enough space to record the file.

This capability gives you extended mass storage capacity—more capacity than that of any one device—without requiring additional keystrokes. The only mass storage functions that do not access all devices are DIR and NEWM. For these two functions, only the "first" mass storage device is accessed. You can specify the "first" mass storage device by using the SELECT function, discussed in section 4.

Programming and the Mass Storage Operations

Automatic Execution of Programs

By using flag 11 in the calculator, you can store programs so that they automatically begin executing when you copy them back into the calculator *from the keyboard*. That is, when you execute READP, READSUB, or READA from the keyboard, such programs will start running as soon as they are copied.

If flag 11 is clear when programs are stored, those programs are retrieved normally.

If flag 11 is set when a *program* is stored (using WRTP or WRTPV), that program begins executing as soon as you copy it back into the calculator from the keyboard. Execution begins from the beginning of the program. The status of flag 11 is important only when the program is *stored*.

If flag 11 is set when *all information* is stored (using WRTA), a program begins executing after you copy the "write-all" file back into the calculator. Execution begins at the place in program memory where the calculator was positioned when the information was stored. You must be sure to locate the proper program line *before* you store the information on the medium. Flag 11 is important only when the information is *stored*.

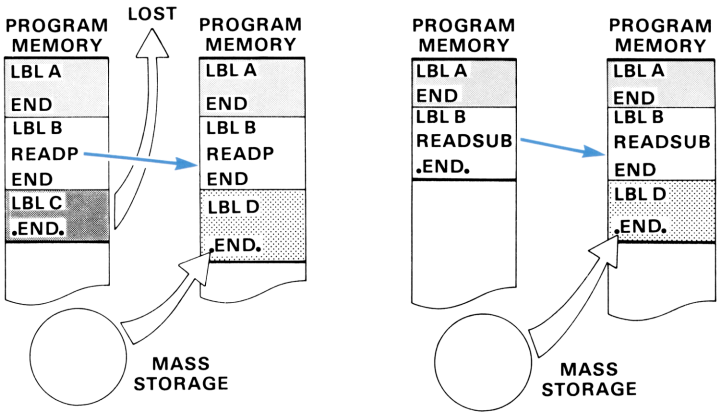
A file that contains a program that executes automatically is indicated in a directory listing by an **A** following the file type.

Executing Mass Storage Functions in Programs

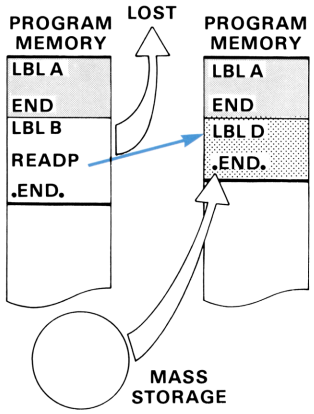
Almost all mass storage functions are programmable—only NEWM is not programmable. All of the programmable functions operate as described earlier in this section. However, the following discussion points out how a running program copies programs from stored files—and possibly executes them.

Three functions copy programs into the calculator: READP, READSUB, and READA. When one of these *read* functions is executed in a program, execution continues according to the following rules:

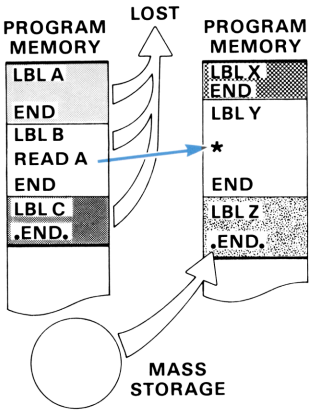
- If the current program is not deleted from program memory, execution continues at the line after the *read* function.



- If the current program is deleted from program memory by READP, execution continues at the first line of the new program.



- If all programs are deleted from program memory by READA, execution continues at the line in the new program memory where the calculator was positioned when the "write-all" file was originally stored.



Further Programming Information

Section 5, Programming and the Interface Loop, discusses other topics related to programs using HP-IL operations—including mass storage operations.

Section 4

Interface Control Operations

You have seen that the HP 82160A HP-IL Module gives you the capability to perform printer and mass storage operations by using functions specifically designed for these applications. However, the Hewlett-Packard Interface Loop is a *general-purpose interface.* A third set of functions—the interface control functions—is designed to give you more complete control of interface activity—for any types of HP-IL devices connected in the loop.

It will be helpful to digress for a moment to give a brief explanation of the operation of the interface loop. This information should give you additional insight into how the devices interact and allow you to work with the loop more effectively—especially when you use the interface control operations described in this section.

Operation of the Hewlett-Packard Interface Loop

When you execute any of the functions described in this manual, the interface module translates the function into a sequence of HP-IL instructions. It then sends them around the interface loop to each device, one at a time. In the discussion that follows, you will learn how the calculator and peripherals communicate using the HP-IL instructions.

Roles of Devices

In order for the interface loop to operate in an orderly manner, the devices in the loop must operate according to their assigned roles. The role of each device is changed to suit the operation being performed. Three different roles are defined for HP-IL devices: controller, talker, and listener. Any device not assigned one of these roles is inactive.

The controller is the *one* device in the loop that can designate the roles of devices and control the loop's operation. The *system* controller (the calculator) is the device that controls and initializes the loop when it is first turned on. It can transfer control to another device, which then becomes the controller of the loop—the *active* controller. Similarly, an active controller can transfer control to another device. For all operations provided by the HP 82160A HP-IL Module, the calculator is always the system controller and the active controller.

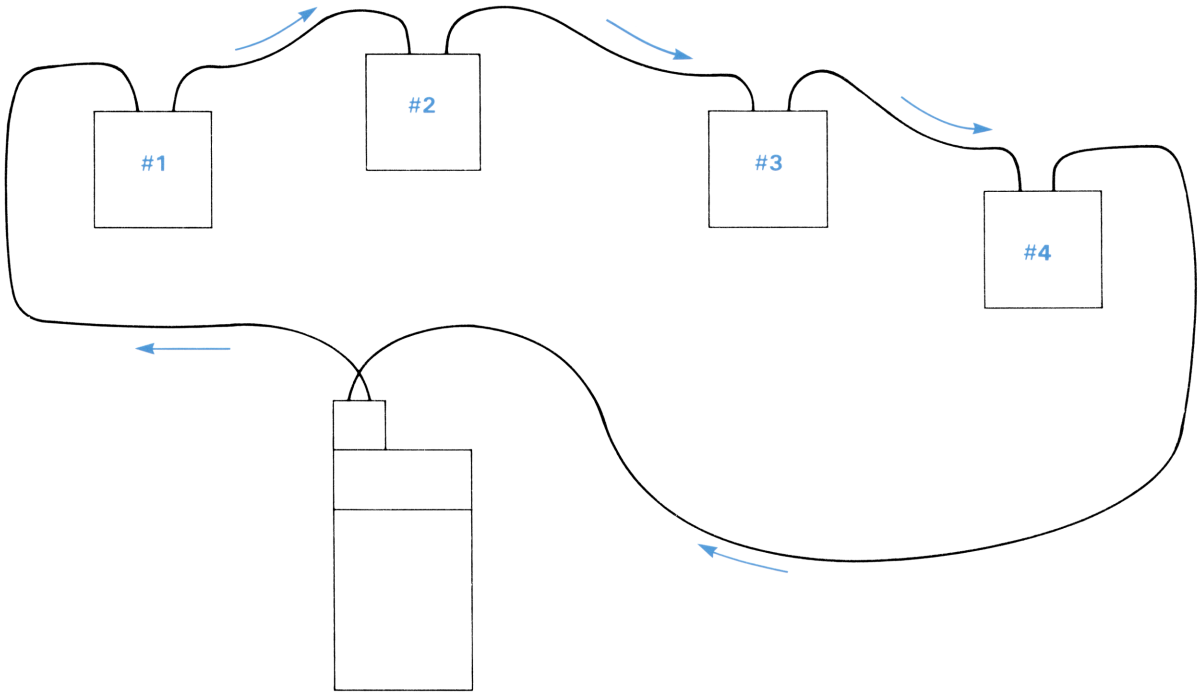
A talker is a device that *sends* information to the interface loop. It is designated and enabled by the controller. At any time, there can be no more than one talker. The controller may be a talker. Examples of talkers are a mass storage device sending data from a stored file and a voltmeter sending voltage measurements.

A listener is a device that *receives* information on the interface loop. There may be more than one listener in the loop at the same time. Listeners are designated by the controller. The controller may be a listener. (A device cannot be a listener and a talker at the same time, although it can have these roles at *different* times.) Examples of listeners are a mass storage device receiving and storing data in a file and a printer receiving and printing information.

Device Addresses

In order to distinguish between devices in the loop, each device must have an address—a number from 1 to 30. The controller uses the addresses to specify and control the devices.

Each HP-IL device has a built-in default address that the device recognizes whenever it is turned on. However, to simplify user operation, the system controller (calculator) always assigns new, sequential addresses to the devices. These addresses start with address 1 for the device next in order after the controller in the direction of information transfer. In this way, each device in the loop has a unique address, which the device stores internally.



Information Flow in the Interface Loop

Information transmitted in the interface loop can be separated into two categories: commands and data.

Commands are initiated by the controller and are monitored by all devices in the loop. Using commands, the controller can initialize the interface loop, assign or unassign devices as talkers and listeners, initiate the transfer of data, and (if it is the system controller) interrupt the operation of the loop.

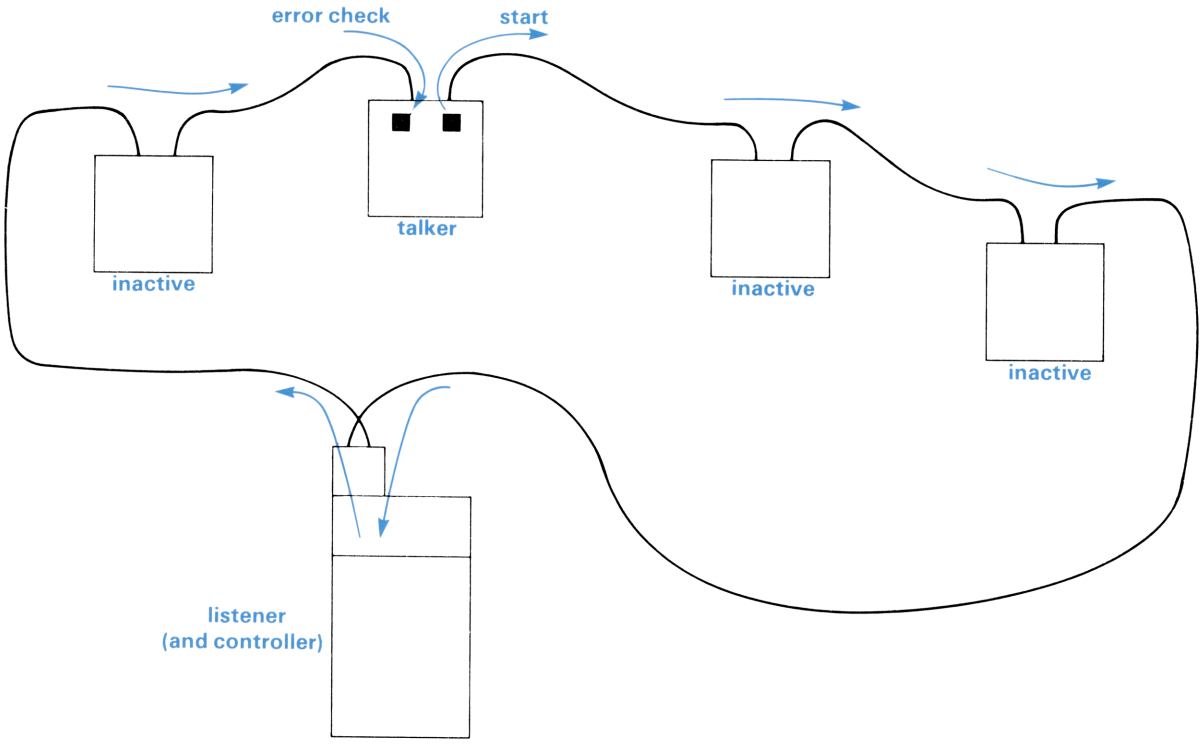
Data is sent by a talker and is processed by all listeners. Data is ignored by the remaining, inactive devices. The talker begins sending data when enabled by the controller; the talker informs the controller when all requested data has been sent. Data can represent numeric and ALPHA information. Data often conveys instructions that control the operation of a peripheral.

Each piece of information—a command or a data byte—is initiated by a device and sent around the interface loop. Each other device, in its turn, receives the information and does one of the following:

- Sends the information to the next device.
- Processes the information and sends the information to the next device.

The action a device takes depends upon the type of information and the role of the device in the loop.

The talker eventually receives the information after it has travelled completely around the loop. It then checks for transmission errors by comparing the received information to the original information. The next piece of information may then be sent around the loop.



Control of the Interface Loop

All functions provided by the HP 82160A HP-IL Module are performed with the calculator as the controller. The calculator *automatically* designates certain devices as talkers and listeners in order to execute each function.

Printer functions are performed by sending instructions and data to the printer. If the printer is one whose accessory type is 32, the instructions are sent as special eight-bit data bytes. For other printer-type devices, the data is sent without any printer instructions. (Refer to the Send Accessory ID message in the owner's manual for the printer.)

Mass storage functions are performed using device-dependent commands—a type of HP-IL command—to control the device. These instructions are automatically sent when the function is executed by the calculator.

By using the interface control functions described below, you can more directly specify the device to perform an operation, send and receive information from a device, and control the interface loop. These functions allow you to use almost *any* HP-IL peripheral—without having to specify the individual HP-IL messages that are sent around the loop.

Selecting an HP-IL Device

For most interface operations, the interface module communicates with one device at a time—the remaining devices merely pass the instructions around the loop. At any time, only one device is designated as the *primary device* in the loop. The primary device is the one that the module will first attempt to communicate with for each function that you execute.

SELECT X *address*

The SELECT function determines the HP-IL device that will be the *primary device* in the loop. The number in the X-register specifies the address of the device that will become the new primary device—from 1 to 30. Whenever you plug in the interface module, the device with address 1 is automatically assigned as the primary device. You can use SELECT to designate any device as the primary device and to change the primary device as often as you want. This is particularly useful when you wish to use more than one printer or mass storage device—or some other type of HP-IL device—in the loop. The primary device is never reset automatically unless you unplug the interface module. The way that the interface loop performs an HP-IL operation will generally depend upon which device is the primary device.

Auto and Manual Modes

The interface module controls the interface loop according to the module's operating mode—either Auto or Manual. In the calculator the status of system flag 32 (one of the peripheral flags) indicates the interface mode. If flag 32 is clear, the interface is in Auto mode. If flag 32 is set, the interface is in Manual mode. You can test flag 32 to determine the interface mode, but you cannot directly set or clear this flag because it is a *system* flag. The status of flag 32 is maintained when the calculator is turned off.

Auto mode is the easier mode to use with almost all HP-IL devices. In Auto mode, when you execute a printer or mass storage function, the interface *automatically* searches the loop for the proper device to carry out the operation. For example, if you execute the PRA function, the printer in the loop performs the operation. If you execute WRTP, the mass storage device performs the operation. If you execute any of the interface control functions that operate with a single device (as described in this section), the primary device—which can be any HP-IL peripheral—performs the operation. Any time the calculator encounters a **MEMORY LOST** condition, the interface begins controlling the loop in Auto mode. This operating mode was assumed to be in effect throughout the discussions in sections 2 and 3.

In Auto mode, the interface searches for a printer or mass storage device starting at the *primary device* and continuing in order around the loop until the proper type of device is found. You can use the SELECT function to specify the primary device and thereby define where the search for a device will start. For example, if you have two or more printer-type devices in the loop, you can choose a particular printer by using SELECT to tell the interface where to start searching for it.

When the interface searches for a printer or mass storage device, it actually looks for a "standard" device. A "standard" printer-type device is one whose accessory type is a number from 32 to 63, inclusive. A "standard" mass storage device is one whose accessory type is 16. (Refer to the Send Accessory ID message in the owner's manual for the device.)

In Manual mode, the calculator's attention focuses on the primary device. In particular, printer and mass storage operations are directed to the primary device—regardless of the type of device. In this mode you can perform printer operations using a nonstandard printer and mass storage operations using a nonstandard mass storage device. If the primary device can't perform an operation, the calculator may display **TRANSMIT ERR** (after a short delay) or the device may set an internal status bit (or indicator) that corresponds to an error condition.

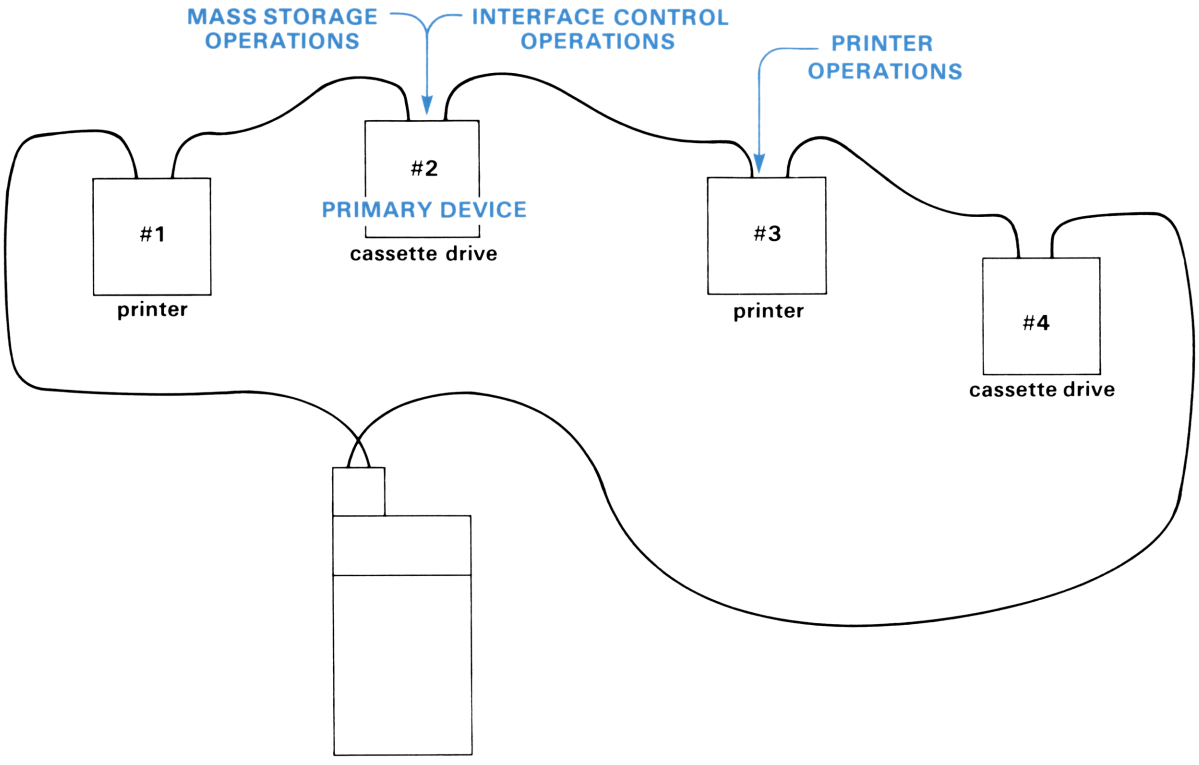
> **Note:** When operating in Manual mode, you should clear flags 15 and 16 if the primary device is not a printer-type device. This prevents the calculator from trying to use the primary device to list the operations being performed.

In both Auto and Manual modes, the interface control functions are performed in the same way. Single-device operations are carried out by the primary device—except for the LISTEN function, which requires an address. Loop-control operations affect *all* devices, regardless of the interface mode.

The following table illustrates how Auto and Manual modes affect the operation of the interface loop:

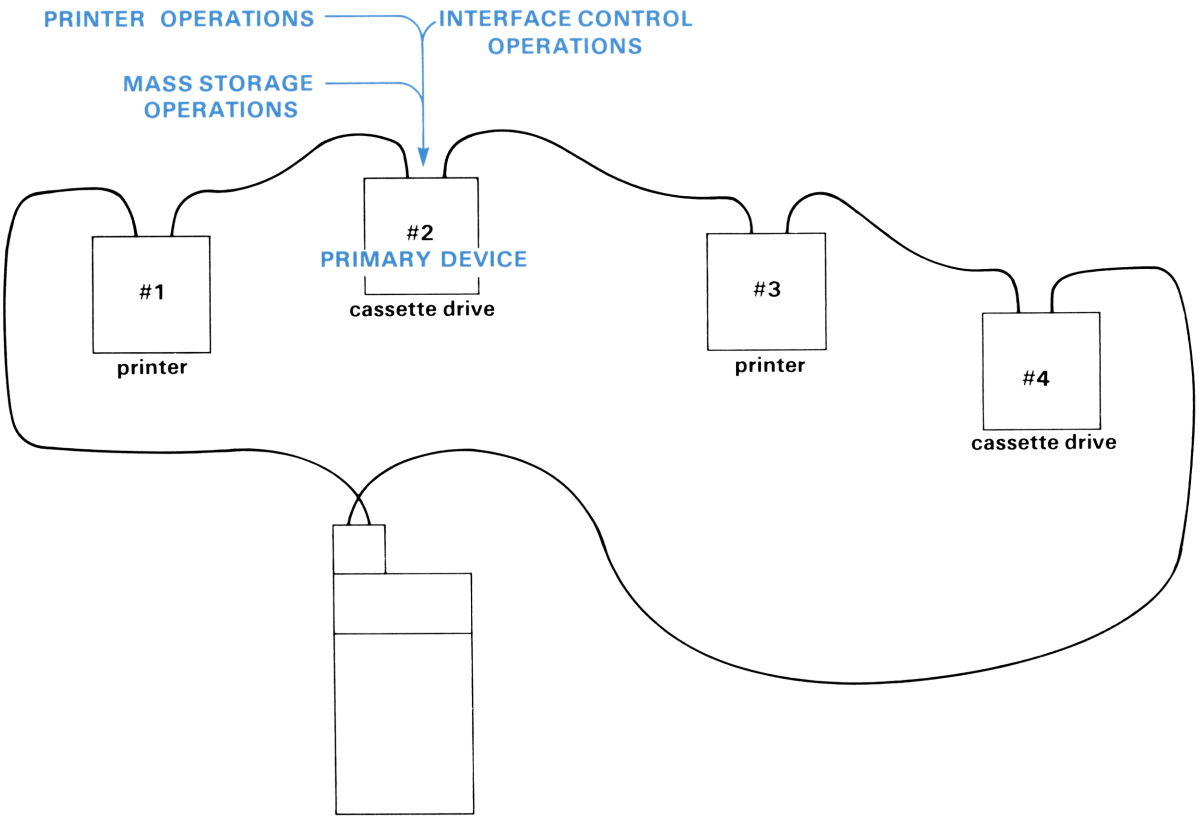
Operation	Auto Mode	Manual Mode
Printer Operations	Performed by first printer-type device in loop starting with primary device.	Performed by primary device, if possible.
Mass Storage Operations	Performed by combination of all mass storage devices in loop starting with primary device.	Performed by primary device, if possible.
Interface Control Operations	Single-device operations performed by primary device, if possible.	Single-device operations performed by primary device, if possible.

As an example, device #2 is selected as the primary device in the interface loop shown below. The loop operates this way:



Auto Mode with Primary Device #2

For the same loop after changing to Manual mode (and device #2 still the primary device), the loop operates this way:



Manual Mode with Primary Device #2

AUTOIO

The AUTOIO (*Auto input/output*) function sets the interface loop to its Auto operating mode and clears flag 32. The loop remains in this mode until it is set to Manual mode. The loop is automatically set to Auto mode only for a **MEMORY LOST** condition.

MANIO

The MANIO (*Manual input/output*) function sets the interface loop to Manual operating mode and sets flag 32. The loop remains in this mode until it is set to Auto mode by executing AUTOIO or by a **MEMORY LOST** condition.

The interface mode is not affected by turning the calculator off and on.

Working With One HP-IL Device

You can control the HP-IL devices in the loop—one at a time—and send and receive information from each device by using the interface control functions described below. To determine the actual response for each function, refer to the appropriate HP-IL message in the owner's manual for the device.

Controlling the Device Mode

You can operate certain HP-IL peripherals in either of two modes—*remote* or *local*. In remote mode the peripheral operates as directed by the interface. In local mode the peripheral is controlled manually and normally does not respond to instructions from the interface loop. (It does, however, monitor the loop for special interface-control messages.) A device may have a control for manually setting its mode. Refer to the owner's manual for the peripheral to determine the device's operating modes and how it operates in these modes.

REMOTE

The REMOTE function sets the primary device to remote mode. This device remains in remote mode until it is manually or automatically set to local mode. (Refer to the Remote Enable message in the owner's manual for the device.)

LOCAL

The LOCAL function sets the primary device to local mode. This device remains in local mode until it is manually or automatically set to remote mode. (Refer to the Go To Local message in the owner's manual for the device.)

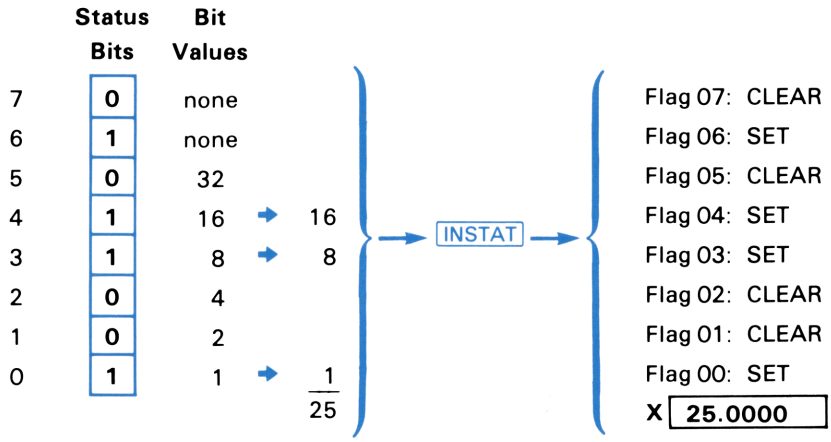
Sending and Retrieving Information

By using interface control functions, you can send ALPHA information to the primary device and you can retrieve three types of information from the device: status information, ALPHA information, and numeric information. ALPHA and numeric information are coded and interpreted by the interface according to standard ASCII (American Standard Code for Information Interchange) conventions. Information is sent and received in the same form if the peripheral uses ASCII coding also.

INSTAT

The INSTAT (*input status*) function retrieves eight status bits from the primary device and places a corresponding number into the X-register. The eight status bits are transferred to user flags 00 through 07 in the calculator. The first, least-significant status bit is transferred to flag 00; the eighth, most-significant status bit is transferred to flag 07.

INSTAT places into the X-register the decimal number that represents the first through sixth status bits (flags 00 through 05)—a number from 0 to 63.



If the primary device sends more than eight bits (one byte) of status, INSTAT uses only the first byte. Refer to the owner's manual for the peripheral to determine the meanings of the status bits. By understanding the meanings of the bits, you can use INSTAT to monitor the condition or state of the peripheral and respond accordingly. (Refer to the Send Status message in the owner's manual for the device.)

OUTA **ALPHA** *information*

The OUTA (*output ALPHA*) function sends the information in the ALPHA register to the primary device in the loop. The ASCII-coded ALPHA string is normally terminated by the ASCII codes for Carriage Return (CR) and Line Feed (LF) when the string is transmitted. The ALPHA register is not altered. The response of the receiving device will depend upon the device. For example, a printer-type device may print the ALPHA characters; a mass storage device may ignore the information. Refer to the owner's manual for the device to determine its response.

You can easily send numerical data using OUTA. Simply execute CLA and ARCL, specifying a register directly or indirectly—then execute OUTA.

You may be able to use OUTA to control certain devices. This is particularly useful for HP-IL devices that are not "standard" devices. By using the ALPHA register and OUTA, you can send the appropriate sequence of commands to the peripheral—causing the peripheral to perform the operations you desire.

INA

The INA (*input ALPHA*) function retrieves an ASCII-coded ALPHA string from the primary device and places it into the ALPHA register. The string normally ends when Carriage Return (CR), Line Feed (LF) are received. However, the calculator stops the primary device if 24 characters are received, filling the ALPHA register. If you execute INA again, the primary device normally sends additional characters to the ALPHA register. (Refer to the Send Data message in the owner's manual for the device.)

IND

The IND (*input decimal*) function retrieves an ASCII-coded numeric value from the primary device and places it into the X-register. The peripheral determines the format in which the number is sent. The interface uses only the first 10 digits of the number and the first 2 digits of an exponent. For example, all of the following formats are valid and are recognized by the interface as the correct number:

```
  2
+0.88253
-.14E2
  6E-22
-427.00766E+12
```

A nonstandard format may produce a meaningless number in the X-register. (Refer to the Send Data message in the owner's manual for the device.)

Suppressing End-of-Line Information

User flag 17 controls how the interface module uses the standard end-of-line indicator—a Carriage Return (CR) and Line Feed (LF). For most peripherals, the end-of-line indicator is normally used, and flag 17 should be clear for this situation.

When flag 17 is clear, the end-of-line indicator signifies the end of a line of data. For OUTA, the calculator sends the ASCII codes for CR and LF at the end of the ALPHA string. For INA, the calculator accepts ALPHA characters only until it receives the CR and LF codes—or until 24 characters are received or the string has been completely transmitted.

When flag 17 is set, the end-of-line indicator is not used. For OUTA, the calculator does not send CR and LF. For INA, the calculator ignores CR and LF, and it accepts characters until 24 characters are received or the string has been completely transmitted.

Triggering a Device

Certain peripherals are designed to perform specific operations when they are activated—or "triggered"—by the interface. For example, a voltmeter could measure a voltage or a solenoid could activate a valve.

TRIGGER

The TRIGGER function activates the primary device. The device's response is determined by its individual capability. (Refer to the Group Execute Trigger message in the owner's manual for the device.)

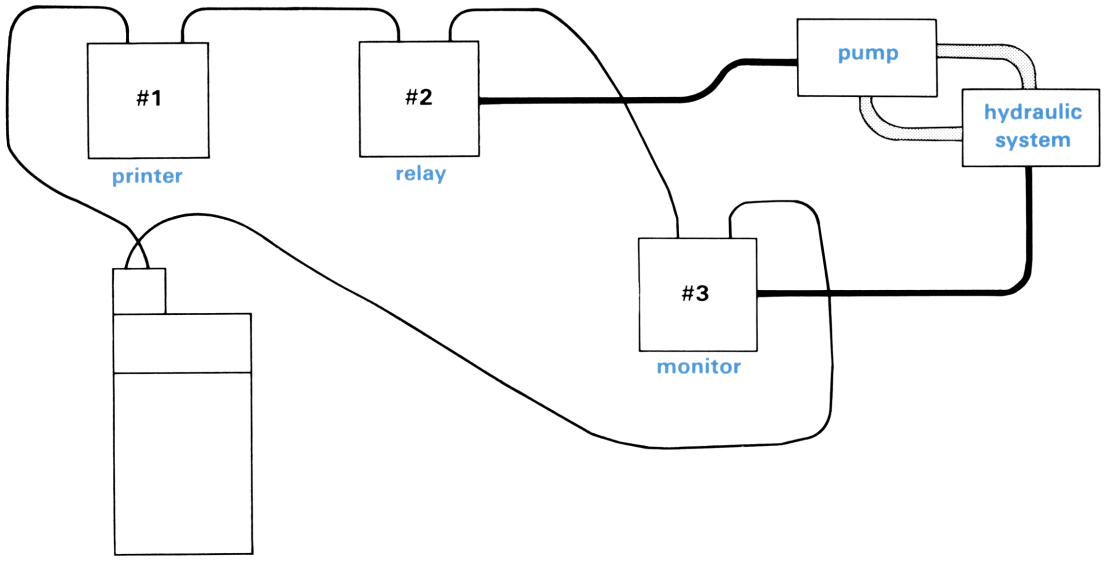
Making a Device a Listener

LISTEN X *address*

The LISTEN function makes the specified device a listener—that is, the device is enabled to receive information. The number in the X-register is the address of the device—a number from 1 to 30. If you execute LISTEN with the number 31 in the X-register, all devices are removed from listener status. (Refer to the Listen Address message in the owner's manual for the device.)

You can set up more than one listener by executing LISTEN for each device. However, when you execute almost any function in the HP-IL module, all devices are automatically removed from listener status. The OUTA and TRIGGER functions are exceptions. Under certain conditions, you can use LISTEN with OUTA or TRIGGER to interact with several devices simultaneously. That is, you can send information to several listeners with a single OUTA function, or you can trigger several listeners with a single TRIGGER function. You can do this from the keyboard if the interface is in Auto mode and either there is no printer-type device in the loop or the Print Function Switch is set to DISABLE. You can do this in a running program in Auto or Manual mode if the loop is set to not trace the operations being performed. Simply execute LISTEN for each device that you want to respond, then execute OUTA or TRIGGER.

Example of working with HP-IL devices: The hydraulic system shown below uses hypothetical HP-IL devices to control and monitor its operation. Use HP-IL functions to turn on the pump by connecting the relay, sample the hydraulic pressure using a 200-psi range on the monitor, turn off the system by disconnecting the relay, and print the pressure.



Assume that the following ALPHA strings perform the indicated functions for these HP-IL devices:

MONITOR		RELAY	
D1	Take one pressure reading.	G1	Connect relay.
F3	Select 200-psi range.	G2	Disconnect relay.

Keystrokes	Display	
AUTOIO	0.0000	Sets loop to Auto mode. (Display assumed cleared.)
■ CF 17	0.0000	Specifies normal end-of-line operation.
2 SELECT	2.0000	Selects relay as primary device.
REMOTE	2.0000	Sets relay to remote mode, allowing it to receive instructions.
ALPHA G ■ 1	G1_	Enters ALPHA relay instruction.
ALPHA	2.0000	
OUTA	2.0000	Sends instruction "G1" to relay.
3 SELECT	3.0000	Selects monitor as primary device.
REMOTE	3.0000	Sets monitor to remote mode.
ALPHA F ■ 3	F3_	Enters ALPHA monitor instruction.
ALPHA	3.0000	
OUTA	3.0000	Sends instruction "F3" to monitor.
ALPHA D ■ 1	D1_	Enters ALPHA monitor instruction.
ALPHA	3.0000	
OUTA	3.0000	Sends instruction "D1" to monitor (and takes pressure reading).
IND	187.2021	Receives decimal input from monitor.
2 SELECT	2.0000	Selects relay as primary device.
ALPHA G ■ 2	G2_	Enters ALPHA relay instruction.
ALPHA	2.0000	
OUTA	2.0000	Sends instruction "G2" to relay.
$x \gtrless y$	187.2021	Restores pressure reading to X-register.
PRX	187.2021	Prints pressure reading.

Working With All Devices

A special group of interface control functions interacts with *all* devices in the interface loop. These functions allow you to control and monitor the loop without repeatedly selecting different primary devices.

Finding a Device Type

For a particular application you may want a program to execute differently according to whether or not a certain device is connected to the interface loop.

For another application you may have a program or procedure that uses a number of HP-IL peripherals in the interface loop. You may not wish to require that the peripherals be connected in exactly the same order in the loop—that is, the devices would not have the same addresses each time you set up the loop.

In both of these cases, you need to check the interface loop to determine if a particular device is connected. You also need to know the device's address.

FINDID ALPHA *identity*

The FINDID (*find identity*) function searches for a device with a specified identity and determines the address of that device. The ALPHA register specifies the identity of the device being sought. The identity is a string of up to seven characters that is used to indicate a model of peripheral. (Refer to the Send Device ID message in the owner's manual for a device to determine its identity, if it has one.) The address of the identified device is returned in the X-register. If the device is not found, the X-register is set to zero.

In Auto mode the interface searches the entire loop starting at the primary device. It continues until it finds the first device with an identity with the same first seven characters as that specified. In Manual mode the interface checks only the primary device.

> The HP 82162A Thermal Printer and the HP 82161A Digital Cassette Drive do *not* have identities that can be used with FINDID.

Controlling Device Power

The power state of some peripherals can be controlled by the interface. By using the functions below, you can set all such devices in the loop to their lowest, standby power conditions. Similarly, you can return these devices to their full operating power conditions. This is particularly useful for applications in which you have a delay between certain operations. However, not all devices have a low-power condition that can be set by the interface. Or you may have to set the power switch to STANDBY for the interface to control the power state. Refer to the owner's manual for a device to determine its power states.

PWRDN

The PWRDN (*power down*) function sets all devices in the loop to their lowest, standby power conditions. Certain devices do not respond to this function; other devices may be manually set to their full-power conditions and will not respond to PWRDN. (Refer to the Loop Power Down message in the owner's manual for the device.)

PWRUP

The PWRUP (*power up*) function sets all devices in the loop to their operating power conditions. PWRUP does not affect devices that are manually turned off. If a device responds to PWRDN, it will normally respond to PWRUP.

Clearing the Interface Loop

STOPIO

The STOPIO (*stop input/output*) function clears the interface loop and sets each device to a known condition. (Refer to the Interface Clear message in the owner's manual for a device.)

Section 5

Programming and the Interface Loop

Information about programming using printer and mass storage operations is contained in sections 2 and 3. More general information about entering and executing programs that use interface operations is described in this section.

Entering Programs That Use HP-IL Operations

The HP 82160A HP-IL Module implements all of the interface operations discussed in this manual. You may enter these functions as part of a program whenever the module is plugged into the calculator—regardless of whether or not the peripheral devices are connected in the loop. (Of course, the peripherals should be connected when the program is executed.) When the interface module is connected, program lines with interface operations are displayed and printed as normal functions.

If the module is disconnected later, these program lines are displayed and printed as XROM functions—with two identification numbers. This indicates that the function belongs to a plug-in accessory. The first number identifies the accessory. (XROM accessory numbers 28 and 29 correspond to the interface module.) The second number identifies the function for that accessory. When the interface module is removed, the interface functions have the following XROM numbers:

Function	XROM Number
ACA	XROM 29,01
ACCHR	XROM 29,02
ACCOL	XROM 29,03
ACSPEC	XROM 29,04
ACX	XROM 29,05
BLDSPEC	XROM 29,06
LIST	Not Programmable
PRA	XROM 29,08
PRAXIS	XROM 29,09
PRBUF	XROM 29,10
PRFLAGS	XROM 29,11
PRKEYS	XROM 29,12
PRP	Not Programmable
PRPLOT	XROM 29,14
PRPLOTP	XROM 29,15
PRREG	XROM 29,16
PRREGX	XROM 29,17
PR Σ	XROM 29,18
PRSTK	XROM 29,19
PRX	XROM 29,20
REGPLOT	XROM 29,21
SKPCHR	XROM 29,22
SKPCOL	XROM 29,23
STKPLOT	XROM 29,24
FMT	XROM 29,25

Function	XROM Number
CREATE	XROM 28,01
DIR	XROM 28,02
NEWM	Not Programmable
PURGE	XROM 28,04
READA	XROM 28,05
READK	XROM 28,06
READP	XROM 28,07
READR	XROM 28,08
READRX	XROM 28,09
READS	XROM 28,10
READSUB	XROM 28,11
RENAME	XROM 28,12
SEC	XROM 28,13
SEEKR	XROM 28,14
UNSEC	XROM 28,15
VERIFY	XROM 28,16
WRTA	XROM 28,17
WRTK	XROM 28,18
WRTP	XROM 28,19
WRTPV	XROM 28,20
WRTR	XROM 28,21
WRTRX	XROM 28,22
WRTS	XROM 28,23
ZERO	XROM 28,24
AUTOIO	XROM 28,27
FINDID	XROM 28,28
INA	XROM 28,29
IND	XROM 28,30
INSTAT	XROM 28,31
LISTEN	XROM 28,32
LOCAL	XROM 28,33
MANIO	XROM 28,34
OUTA	XROM 28,35
PWRDN	XROM 28,36
PWRUP	XROM 28,37
REMOTE	XROM 28,38
SELECT	XROM 28,39
STOPIO	XROM 28,40
TRIGGER	XROM 28,41

If program lines using interface functions are entered *without* the interface module being connnected, the function is recorded, displayed, and printed as **XEQ**$^{\mathsf{T}}$*label*, rather than in the normal form described above. (Also, print functions are recorded this way if the Print Function Switch is set to DISABLE and no HP 82143A Printer is connected.) Program execution will be slowed by lines in this form because the calculator will first search for a *program* or *line* with the specified label.

Executing Programs That Use HP-IL Operations

The interface module and all required HP-IL peripherals must be connected and turned on for normal execution of programs that use these devices.

If the interface module is not connected, **NONEXISTENT** will be displayed when an interface operation is attempted. Similarly, **NONEXISTENT** will be displayed for print functions if the Print Function Switch is set to DISABLE and no HP 82143A Printer is plugged in.

If a peripheral is turned off (or the loop is otherwise not completed), **TRANSMIT ERR** will be displayed when an interface operation is attempted.

If a required device is not connected, improper operation may occur or an error message may be displayed when a corresponding operation is attempted.

Appendix A

Care, Warranty, and Service Information

Interface Care

CAUTION

Always turn off the calculator before connecting or disconnecting the interface module or any peripheral. Failure to do so could result in damage to the calculator or disruption of the system's operation.

- Keep the contact area of the interface module free of obstructions. Should the contacts become dirty, carefully brush or blow the dirt out of the contact area. Do not use any liquid to clean the contacts.
- Store the interface module in a clean, dry place.
- Always turn off the calculator before installing or removing the module or peripherals in the loop. Follow the procedures described in section 1.
- Observe the following temperature specifications:

  Operating: 0° to 45° C (32° to 113° F).

  Storage: −40° to 75° C (−40° to 167° F).

Verifying Proper Operation

If at any time you suspect that your calculator or interface loop is not operating properly, you can verify its operation by doing the following:

1. Check that all peripheral devices are turned on.
2. Check that the interface mode is set properly for your application. If flag 32 is clear, the interface is in Auto mode. If flag 32 is set, the interface is in Manual mode.
3. Turn off the calculator and all peripherals. After disconnecting all plug-in devices from the calculator, check that the calculator turns on and operates properly.
4. Turn off the calculator. After installing the interface module in any port and connecting its two HP-IL leads *to each other,* turn on the calculator.
   - If the calculator display turns on immediately, the interface module is operating properly and is probably good.
   - If the calculator display turns on after a delay of about 2 or 3 seconds, the interface module requires service.
5. Connect any peripheral devices to the loop, one at a time, and observe how the display turns on when you turn on the calculator. (Be sure the devices are turned on *before* you turn on the calculator.)
   - If the calculator display turns on immediately, the peripherals are interacting properly.
   - If the calculator display turns on after a delay of about 2 or 3 seconds, the peripheral either is turned off or requires service.

   If any device causes improper operation of the interface, that device may require service. (Refer to the owner's manual for that device.)

If this procedure indicates proper operation, but you still experience difficulty, write or telephone Hewlett-Packard at an address or phone number listed below under Service.

Limited One-Year Warranty

What We Will Do

The HP 82160A HP-IL Module is warranted by Hewlett-Packard against defects in materials and workmanship for one year from the date of original purchase. If you sell your unit or give it as a gift, the warranty is automatically transferred to the new owner and remains in effect for the original one-year period. During the warranty period, we will repair or, at our option, replace at no charge a product that proves to be defective, provided you return the product, shipping prepaid, to a Hewlett-Packard service center.

What Is Not Covered

This warranty does not apply if the product has been damaged by accident or misuse or as the result of service or modification by other than an authorized Hewlett-Packard service center.

No other express warranty is given. The repair or replacement of a product is your exclusive remedy. **ANY OTHER IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS IS LIMITED TO THE ONE-YEAR DURATION OF THIS WRITTEN WARRANTY.** Some states, provinces, or countries do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you. **IN NO EVENT SHALL HEWLETT-PACKARD COMPANY BE LIABLE FOR CONSEQUENTIAL DAMAGES.** Some states, provinces, or countries do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

This warranty gives you specific legal rights, and you may also have other rights which vary from state to state, province to province, or country to country.

Warranty for Consumer Transactions in the United Kingdom

This warranty shall not apply to consumer transactions and shall not affect the statutory rights of a consumer. In relation to such transactions, the rights and obligations of Seller and Buyer shall be determined by statute.

Obligation to Make Changes

Products are sold on the basis of specifications applicable at the time of manufacture. Hewlett-Packard shall have no obligation to modify or update products once sold.

Warranty Information

If you have any questions concerning this warranty, please contact an authorized Hewlett-Packard dealer or a Hewlett-Packard sales and service office. Should you be unable to contact them, please contact:

- In the United States:

Hewlett-Packard
1000 N.E. Circle Blvd.
Corvallis, OR 97330
Telephone: (503) 758-1010
Toll-Free Number: (800) 547-3400 (except in Oregon, Hawaii, and Alaska)

- In Europe:

Hewlett-Packard S.A.
7, rue du Bois-du-lan
P.O. Box
CH-1217 Meyrin 2
Geneva
Switzerland
Telephone: (022) 83 81 11

Note: Do *not* send units to this address for repair.

- In other countries:

Hewlett-Packard Intercontinental
3495 Deer Creek Rd.
Palo Alto, California 94304
U.S.A.
Telephone: (415) 857-1501

Note: Do *not* send units to this address for repair.

Service

Hewlett-Packard maintains service centers in most major countries throughout the world. You may have your unit repaired at a Hewlett-Packard service center any time it needs service, whether the unit is under warranty or not. There is a charge for repairs after the one-year warranty period.

Hewlett-Packard calculator products are normally repaired and reshipped within five (5) working days of receipt at any service center. This is an average time and could possibly vary depending upon the time of year and work load at the service center. The total time you are without your unit will depend largely on the shipping time.

Obtaining Repair Service in the United States

The Hewlett-Packard United States Service Center for handheld and portable calculator products is located in Corvallis, Oregon:

Hewlett-Packard Company
Corvallis Division Service Department
P.O. Box 999/1000 N.E. Circle Blvd.
Corvallis, Oregon 97330, U.S.A.
Telephone: (503) 757-2000

Obtaining Repair Service in Europe

Service centers are maintained at the following locations. For countries not listed, contact the dealer where you purchased your unit.

AUSTRIA
HEWLETT-PACKARD GmbH
Kleinrechner-Service
Wagramerstr.-Lieblgasse
A-1220 VIENNA
Telephone: (222) 35.16.20

BELGIUM
HEWLETT-PACKARD BELGIUM SA/NV
Boulevard de la Woluwe 100
Woluwelaan
B-1200 BRUSSELS
Telephone: (2) 762 32 00

DENMARK
HEWLETT-PACKARD A/S
Datavej 52
DK-3460 BIRKEROD (Copenhagen)
Telephone: (02) 81 66 40

EASTERN EUROPE
Refer to the address listed under Austria

FINLAND
HEWLETT-PACKARD OY
Revontulentie 7
02100 ESPOO 10 (Helsinki)
Telephone: (90) 455 02 11

FRANCE
HEWLETT-PACKARD FRANCE
Division Informatique Personnelle
S.A.V. Calculateurs de Poche
F-91947 Les Ulis Cedex
Telephone: (6) 907 78 25

GERMANY
HEWLETT-PACKARD GmbH
Kleinrechner-Service
Vertriebszentrale
Berner Strasse 117
Postfach 560 140
D-6000 FRANKFURT 56
Telephone: (611) 50041

ITALY
HEWLETT-PACKARD ITALIANA S.P.A.
Casella postale 3645 (Milano)
Via G. Di Vittorio, 9
I-20063 CERNUSCO SUL NAVIGLIO (Milan)
Telephone: (2) 90 36 91

NETHERLANDS
HEWLETT-PACKARD NEDERLAND B.V.
Van Heuven Goedhartlaan 121
N-1181 KK AMSTELVEEN (Amsterdam)
P.O. Box 667
Telephone: (020) 472021

NORWAY
HEWLETT-PACKARD NORGE A/S
P.O. Box 34
Oesterndalen 18
N-1345 OESTERAAS (Oslo)
Telephone: (2) 17 11 80

SPAIN
HEWLETT-PACKARD ESPANOLA S.A.
Calle Jerez 3
E-MADRID 16
Telephone: (1) 458 2600

SWEDEN
HEWLETT-PACKARD SVERIGE AB
Enighetsvagen 3
Box 205 02
S 161 BROMMA 20 (Stockholm)
Telephone: (8) 730 05 50

SWITZERLAND
HEWLETT-PACKARD (SCHWEIZ) AG
Kleinrechner-Service
Allmend 2
CH-8967 WIDEN
Telephone: (057) 50111

UNITED KINGDOM
HEWLETT-PACKARD Ltd
King Street Lane
GB-WINNERSH, WOKINGHAM
BERKSHIRE RG11 5AR
Telephone: (734) 784774

International Service Information

Not all Hewlett-Packard service centers offer service for all models of HP calculator products. However, if you bought your product from an authorized Hewlett-Packard dealer, you can be sure that service is available in the country where you bought it.

If you happen to be outside of the country where you bought your unit, you can contact the local Hewlett-Packard service center to see if service is available for it. If service is unavailable, please ship the unit to the address listed above under Obtaining Repair Service in the United States. A list of service centers for other countries can be obtained by writing to that address.

All shipping, reimportation arrangements, and customs costs are your responsibility.

Service Repair Charge

There is a standard repair charge for out-of-warranty repairs. The repair charges include all labor and materials. In the United States, the full charge is subject to the customer's local sales tax. In European countries, the full charge is subject to Value Added Tax (VAT) and similar taxes wherever applicable. All such taxes will appear as separate items on invoiced amounts.

Products damaged by accident or misuse are not covered by the fixed repair charges. In these situations, repair charges will be individually determined based on time and material.

Service Warranty

Any out-of-warranty repairs are warranted against defects in materials and workmanship for a period of 90 days from date of service.

Shipping Instructions

Should your unit require service, return it with the following items:

- A completed Service Card, including a description of the problem and system setup when the problem occurred.
- A sales receipt or other documentary proof of purchase date if the one-year warranty has not expired.

The product, the Service Card, a brief description of the problem, and (if required) the proof of purchase date should be packaged in the original shipping case or other adequate protective packaging to prevent in-transit damage. Such damage is not covered by the one-year limited warranty; Hewlett-Packard suggests that you insure the shipment to the service center. The packaged unit should be shipped to the nearest Hewlett-Packard designated collection point or service center. Contact your dealer directly for assistance. (If you are not in the country where you originally purchased the unit, refer to International Service Information above.)

Whether the unit is under warranty or not, it is your responsibility to pay shipping charges for delivery to the Hewlett-Packard service center.

After warranty repairs are completed, the service center returns the unit with postage prepaid. On out-of-warranty repairs in the United States and some other countries, the unit is returned C.O.D. (covering shipping costs and the service charge).

Further Information

Service contracts are not available. Calculator product circuitry and design are proprietary to Hewlett-Packard, and service manuals are not available to customers.

Should other problems or questions arise regarding repairs, please call your nearest Hewlett-Packard service center.

Potential for Radio/Television Interference

The HP 82160A HP-IL Module generates and uses radio frequency energy and, if not installed and used properly (that is, in strict accordance with the instructions in this manual), may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If the module does cause interference to radio or television reception, which can be determined by unplugging the module, you are encouraged to try to correct the interference by one or more of the following measures:

- Reorient the receiving antenna.
- Relocate the calculator with respect to the receiver.
- Move the calculator away from the receiver.
- Plug the calculator's recharger into a different outlet so that the calculator and the receiver are on different branch circuits.

If necessary, you should consult your dealer or an experienced radio/television technician for additional suggestions. You may find the following booklet, prepared by the Federal Communications Commission, helpful: *How to Identify and Resolve Radio—TV Interference Problems*. This booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402, Stock Number 004-000-00345-4.

Programming and Applications Assistance

Should you need technical assistance concerning programming, applications, etc., call Hewlett-Packard Customer Support at (503) 757-2000. This is not a toll-free number, and we regret that we cannot accept collect calls. As an alternative, you may write to:

Hewlett-Packard
Corvallis Division Customer Support
1000 N.E. Circle Blvd.
Corvallis, OR 97330

A number of our users submit program applications or unique program key sequences to Hewlett-Packard. However, we can consider using only ideas given freely to us. Since it is the policy of Hewlett-Packard not to accept suggestions given in confidence, the following statement must be included with your submittal:

"I am voluntarily submitting this information to Hewlett-Packard Company. The information is not confidential and Hewlett-Packard may do whatever it wishes with the information without obligation to me or anyone else."

Dealer and Product Information

For dealer locations, product information, and prices, please call (800) 547-3400. In Oregon, Alaska, or Hawaii, call (503) 758-1010.

Appendix B

Error Messages

This appendix contains a list of messages and errors that are related to interface operations. The messages and errors are grouped into printer, mass storage, and general interface control categories. Errors in the interface control category may occur during printer or mass storage operations. (Refer to the owner's handbook for your calculator for a list of all standard errors and messages.)

Note: For most error conditions, the function being attempted is not performed. However, for those conditions and functions indicated by * below, the operation may be partially performed.

Printer Operations

Display		Functions	Meaning
ALPHA DATA		*-all-*	ALPHA characters are in a register where a number is required—either a stack register or a data storage register.
DATA ERROR		ACCHR, ACCOL, BLDSPEC	$\|x\| \geq 128$.
		PRAXIS	YMAX $\leq$ YMIN, AXIS $>$ YMAX, AXIS $<$ YMIN, or $\|nnn\| > 168$.
	*	PRPLOTP	XMAX $\leq$ XMIN, YMAX $\leq$ YMIN, AXIS $>$ YMAX, or AXIS $<$ Y MIN.
		PRREGX	$\|x\| > 999$.
		REGPLOT, STKPLOT	YMAX $\leq$ YMIN, ***nnn*** $= 0$, or $\|nnn\| > 168$.
		SKPCHR	$\|x\| \geq 24$.
		SKPCOL	$\|x\| \geq 168$.
NO PRINTER		*-all-*	A standard printer-type device is not in the interface loop. (Occurs in Auto mode only.)
NONEXISTENT		*-all-*	Print Function Switch set to DISABLE. Set switch to ENABLE.
		PRP	Specified program or function program does not exist. Check program name.
	*	PRPLOT	
	*	PRPLOTP	
	*	PRREGX	Specified registers exceed highest numbered data storage register. Check ***bbb.eee*** format in X.
PRINTER ERR	*	*-all-*	The printer is out of paper, jammed, or requires service. Reload paper (if required) or turn printer off and on, then check whether error recurs.
PRIVATE		*-all-*	An attempt was made to list, trace, edit, or view a private program.

Mass Storage Operations

Display		Functions	Meaning
ALPHA DATA		CREATE, READRX, SEEKR, WRTRX	The X-register contains ALPHA characters instead of the required numeric data.
DATA ERROR		NEWM	$\|x\| \geq 448$.
DIR FULL		*-all-*	Directory space on storage medium is full. The file is not recorded. Erase a file or use another medium.
DRIVE ERR		*-all-*	Medium stalled (or at end). Try new medium. Drive or medium may be bad.
		VERIFY	Medium stalled (or at end)—file not verified. Medium may be bad.
DUP FL NAME		CREATE	A file with the same name already exists on the medium. The new file is not created. Use the existing file (if it is a data file), erase the existing file, or specify another name.
		WRTA, WRTK, WRTP, WRTPV, WRTS	A file of a different type with same name already exists on the medium. The new file is not created. Erase the existing file or specify another name.
END OF FILE		READRX, SEEKR, WRTR, WRTRX	Execution would require medium to be positioned past the last register in the specified file. Specify a lower file register, use a larger data file or decrease storage memory, or specify fewer registers.
FL NOT FOUND		*-all-*	File with specified name not found on the medium. Check file name, especially for trailing blanks.
FL SECURED		*-all-*	Specified file is secured. Execute UNSEC to cancel the security.
FL TYPE ERR		READA, READK, READP, READR, READS, SEEKR	The specified file is not the type required by the function. Specify a file of the proper type.
		READRX, WRTRX	The medium is not positioned to a data file. Execute SEEKR.
MEDM ERR	*	*-all-*	Medium improperly installed or possibly damaged or worn out.
		VERIFY	Medium improperly installed or possibly damaged or worn out—file not verified.
MEDM FULL		CREATE, WRTA, WRTK, WRTP, WRTPV, WRTS	File storage space on the medium is too small for the file. The file is not recorded. Erase a file or use another medium.

Display		Functions	Meaning
MEMORY LOST		READA	Read error—calculator memory cleared.
NAME ERR		*-all-*	ALPHA register is empty.
		WRTP WRTPV	ALPHA register is empty or program is nonexistent. Check program name, especially for trailing blanks.
NO DRIVE		*-all-*	A standard mass storage device is not in the interface loop. (Occurs in Auto mode only.)
NO KEYS		WRTK	No calculator or accessory functions are assigned to any keys. The file is not recorded.
NO ROOM		READA	Memory is too small for information being read. Memory is not changed. Add memory module(s).
		READK READP READSUB	Executed from a program, memory is too small for information being read. Memory is not changed. Reallocate registers or add memory module(s).
NO MEDM		*-all-*	Medium not installed properly.
NONEXISTENT		READRX WRTRX	Specified registers exceed highest numbered data storage register. Check ***bbb.eee*** format in X or reallocate registers.
PACKING TRY AGAIN		READK READP READSUB	Executed from the keyboard, memory is too small for information being read. Memory is packed. Try the operation again, or reallocate registers or add memory module(s).
PRIVATE		WRTA WRTP WRTPV	An attempt was made to store a private program.
READ ERR	*	READA READP READS READSUB	Invalid data read from medium. Try again or re-record the file. The device may require service.
ROM		WRTP WRTPV	Specified program is in ROM (read-only memory) in a plug-in module. Use COPY first.
SIZE ERR	*	READS	Not enough unused program memory for required increase in storage memory. Register allocation is not changed. Delete programs or program lines, or add memory module(s).

Interface Control Operations

Display		Functions	Meaning
ALPHA DATA		SELECT	Non-numeric data is in the X-register.
ADR ERR		LISTEN	$\|$***address***$\| < 1$ or $\|$***address***$\| \geq 32$.
		SELECT	$\|$***address***$\| < 1$ or $\|$***address***$\| \geq 31$.
TRANSMIT ERR	*	*-all-*	Interface loop not connected, one device turned off, or one device may require service (possibly the active device). Check the loop. In Manual mode, the primary device may not be able to perform the operation; select the proper device.

Annotated PRPLOT Program Listing

Following is a listing of the PRPLOT plotting program. This program can be loaded into your calculator's program memory using the COPY operation. The PRPLOT program requires 77 registers of program memory. Refer to the owner's handbook for your calculator for more information about COPY.

Prompts for inputs, checks data, and stores the input values.

```
01♦LBL "PRPLOT"
02 AON
03 "NAME ?"
04 PROMPT
05 AOFF
06 ASTO 11
07♦LBL 11
08 "Y MIN ?"
09 PROMPT
10 STO 00
11 "Y MAX ?"
12 PROMPT
13 STO 01
14 X<=Y?
15 GTO 11
16♦LBL 12
17 "AXIS ?"
18 CF 23
19 PROMPT
20 STO 04
21 FS? 23
22 ASTO 04
23 RCL 01
24 X<Y?
25 GTO 12
26 CLX
27 RCL 00
28 X>Y?
29 GTO 12
30♦LBL 13
31 "X MIN ?"
32 PROMPT
33 STO 08
34 "X MAX ?"
35 PROMPT
36 STO 09
37 X<=Y?
38 GTO 13
39 "X INC ?"
40 PROMPT
41 STO 10
```

Prints the name of the function being plotted.

```
42♦LBL "PRPLOTP"
43 CF 12
44 ADV
45 6
46 SKPCHR
47 "PLOT OF "
48 ARCL 11
49 ACA
50 PRBUF
```

Calculates and prints the X units.

```
51 RCL 08
52 RCL 09
53 "X"
54 XEQ 09
55 STO 07
56 7
57 ACCHR
58 PRBUF
```

Sets plot field width to 130 columns.

```
59 130
60 STO 02
```

Calculates Y units and completes the plot label.

```
61 XROM "PRAXIS"
```

Checks X INC for positive or negative sign.

```
62 RCL 10
63 X>0?
64 GTO 00
```

Calculates x increment value if X INC is negative (number of increments).

```
65 RCL 09
66 RCL 08
67 -
68 RCL 10
69 ABS
70 /
71 STO 10
```

Determines print format for x-axis labels.

```
72♦LBL 00
73 RCL 09
74 RCL 08
75 ABS
76 X<Y?
77 X<>Y
78 RCL 07
79 /
80 LOG
81 INT
82 2
83 -
84 STO 05
```

Sets first x value equal to X MIN (the initial value).

```
85 RCL 08
86 STO 06
```

Sets print format and accumulates x-axis label.

```
87♦LBL 14
88 FIX IND 05
89 RCL 07
90 /
91 RND
92 ACX
93 3
94 SKPCOL
```

Calculates and prints one point.

```
95 RCL 06
96 XEQ IND 11
97 REGPLOT
```

Increments x value and checks for completion of plot.

```
98 RCL 10
99 ST+ 06
100 RCL 09
```

```
101 RCL 06
102 X<=Y?
103 GTO 14
```

Resets display mode.

```
104 FIX 4
```

```
105 RTN
```

Calculates and prints *y*-axis label.

```
106♦LBL "PRAXIS"
107 CF 12
108 RCL 00
109 RCL 01
110 "Y"
111 XEQ 09
112 STO 06
113 125
114 ACCHR
115 PRBUF
```

Converts **nnn.aaa** to **nnn** and checks **nnn**.

```
116 RCL 02
117 INT
118 ABS
119 STO 02
120 168
121 X<Y?
122 GTO 10
```

Formats and accumulates Y MIN label.

```
123 RCL 00
124 RCL 06
125 /
126 RND
127 ACX
```

Calculates and skips number of columns between the Y MIN and Y MAX labels.

```
128 XEQ 05
129 R↑
130 RCL 01
131 XEQ 04
132 R↑
133 +
134 -
135 7
136 X<=Y?
```

Accumulates the Y MAX label and prints the *y*-axis labels.

```
137 RDN
138 SKPCOL
139 RCL 01
140 RCL 06
141 /
142 RND
143 ACX
144 ADV
```

Calculates axis column location (if printed).

```
145 RCL 04
146 SIGN
147 X=0?
148 GTO 03
149 LASTX
150 RCL 00
151 X>Y?
152 GTO 10
153 -
154 RCL 01
155 RCL 00
156 -
157 X<Y?
158 GTO 10
159 /
160 RCL 02
161 1
162 -
163 *
164 .5
165 +
166 INT
167 STO Y
```

Positions and prints axis label.

```
168 RCL 04
169 RCL 06
170 /
171 RND
172 ACX
173 XEQ 05
174 2
175 /
176 X>Y?
177 GTO 00
178 +
179 RCL 02
180 1
181 -
182 X<Y?
183 ENTER↑
184 -
185 GTO 01
186♦LBL 00
187 ENTER↑
188 +
189 RCL 02
190 -
191♦LBL 01
192 SKPCOL
193 ADV
```

Calculates position of axis mark and prints *y*-axis.

```
194 XEQ 08
195 STO 05
196 X=0?
197 GTO 00
198 RCL 02
199 1
200 -
201 X=Y?
202 GTO 00
203 X<>Y
204 1
205 -
206 XEQ 06
207 RCL 05
208 1
209 +
210 GTO 01
211♦LBL 03
212 XEQ 08
213♦LBL 00
214 RCL 02
215 2
216♦LBL 01
217 -
218 XEQ 06
219 ADV
```

Puts **nnn.aaa** in R_{02}.

```
220 RCL 02
221 RCL 05
222 1
223 +
224 1 E3
225 /
226 +
227 ENTER↑
228 CHS
229 X<>Y
230 RCL 04
231 SIGN
232 X=0?
233 RDN
234 RDN
235 STO 02
```

Resets display format.

```
236 FIX 4
237 RTN
```

Formats label value into actual label.

```
238♦LBL 04
239 RCL 06
240 /
241 RND
```

Calculates number of columns required for a label.

```
242♦LBL 05
243 ABS
244 INT
245 X≠0?
246 GTO 00
247 RDN
248 5
249♦LBL 00
250 LOG
251 INT
252 RCL 05
253 +
254 3
255 +
256 7
257 *
258 RTN
```

Fills in axis with dashes between label marks.

```
259♦LBL 06
260 ENTER↑
261 ENTER↑
262 7
263 MOD
264 2
265 /
266 INT
267 SKPCOL
268 -
269 "-"
270♦LBL 07
271 7
272 X>Y?
273 GTO 00
274 -
275 ACA
276 GTO 07
277♦LBL 00
278 RDN
279 SKPCOL
```

Accumulates a label mark.

```
280♦LBL 08
281 127
282 ACCOL
283 R↑
284 RTN
```

Calculates multiplier value and places that value into X. Accumulates "units" line.

```
285♦LBL 09
286 "⊢ <UNITS="
287 X<=Y?
288 GTO 10
289 X<>Y
290 ABS
291 X<Y?
292 X<>Y
293 LOG
294 X<0?
295 GTO 00
296 INT
297 2
298 X<>Y
299 X>Y?
300 GTO 01
301 -
302 STO 05
303 0
304 GTO 02
305♦LBL 00
306 FRC
307 X≠0?
308 1
309 LASTX
310 INT
311 X<>Y
312 -
313♦LBL 01
314 "⊢ E"
315♦LBL 02
316 4
317 SKPCHR
318 ACA
319 FIX 0
320 RDN
321 X=0?
322 GTO 00
323 ACX
324 10↑X
325 2
326 STO 05
327 FIX 2
328 RDN
329 GTO 01
330♦LBL 00
331 1
332 ACX
333 FIX IND 05
334♦LBL 01
335 ">"
336 ACA
337 RTN
```

Generates **DATA ERROR** message for bad inputs.

```
338♦LBL 10
339 0
340 /
341 END
```

Function Index

The operations listed below are active while the HP 82160A HP-IL Module is plugged into the calculator (and the Print Function Switch is set to ENABLE). These operations and programs containing these operations are executable only when appropriate peripherals are connected to the interface loop.

Printer Operations:

Mass Storage Operations:

Interface Control Operations:

HP 82160A HP-IL Module
Owner's Manual

1000 N.E. Circle Blvd., Corvallis, OR 97330, U.S.A.

82160-90001

Printed in U.S.A.