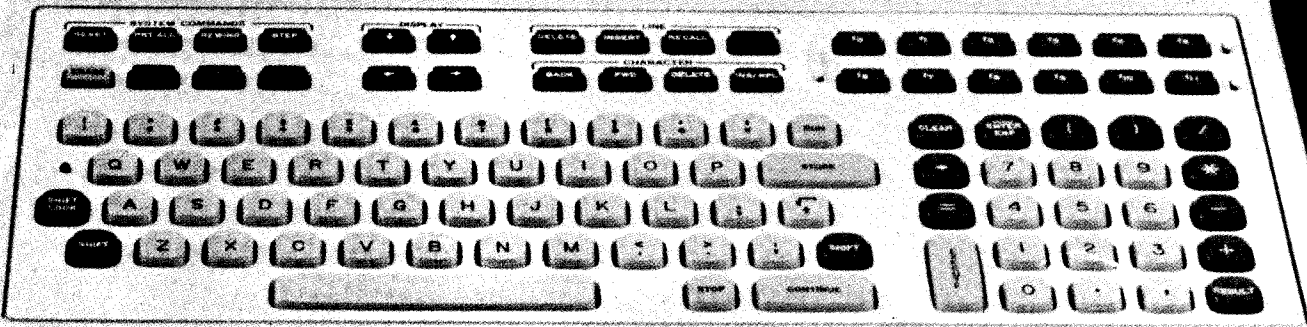


SECURITY ENGINEERING
REFERENCE MATERIAL
DO NOT REMOVE FROM AREA

DO NOT REMOVE FROM AREA

HEWLETT - PACKARD 9825A



HEWLETT  PACKARD

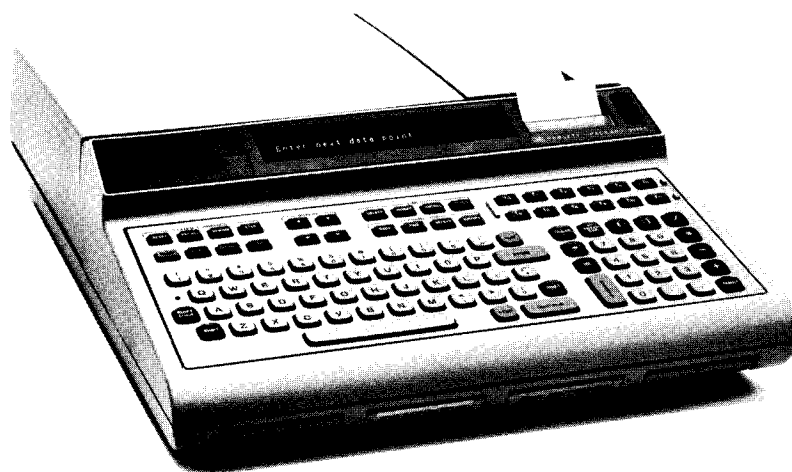
Warranty Statement

Hewlett-Packard products are warranted against defects in materials and workmanship. For Hewlett-Packard Desktop Computer Division products sold in the U.S.A. and Canada, this warranty applies for ninety (90) days from date of delivery.* Hewlett-Packard will, at its option, repair or replace equipment which proves to be defective during the warranty period. This warranty includes labor, parts, and surface travel costs, if any. Equipment returned to Hewlett-Packard for repair must be shipped freight prepaid. Repairs necessitated by misuse of the equipment, or by hardware, software, or interfacing not provided by Hewlett-Packard are not covered by this warranty.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. HEWLETT-PACKARD SHALL NOT BE LIABLE FOR CONSEQUENTIAL DAMAGES.

*For other countries, contact your local Sales and Service Office to determine warranty terms.

General I/O Programming



HP 9825A Desktop Computer

RELIABILITY ENGINEERING
REFERENCE MATERIAL
DO NOT REMOVE FROM AREA

Hewlett-Packard Desktop Computer Division
3404 East Harmony Road, Fort Collins, Colorado 80525
(For World-wide Sales and Service Offices see back of manual.)
Copyright by Hewlett-Packard Company 1976

General I/O ROM Operations

Operation	Description
Write	Output data or character strings to specified device.
Read	Request and input data or character strings.
Format	Specify data specs and edit specs for both read and write statements.
Conversion	Set up a character conversion table for read and write statements.
Write Binary	Output 16-bit binary numbers.
Read Binary	Input 16-bit binary numbers.
Write Control	Output binary status codes to an interface card.
Read Status	Check interface or peripheral status information.
List	Output program listings to an external device.

Preface

An HP 9825A Calculator requires a General I/O ROM to control most peripheral devices. This manual describes the peripheral control statements and functions available with a General I/O ROM. The facing page lists the operations available with the ROM.

The instructions in this manual assume that you are already familiar with operating and programming the calculator. You should also be familiar with the unique operation and functions of each peripheral device to be controlled using the General I/O ROM.

Table of Contents

Chapter 1: General Information

Inspection Procedure	1
Installation	1
Memory Usage	2
Syntax Guidelines	2
Error Messages	2

Chapter 2: Formatted I/O Operations

Calculator I/O Scheme	3
Select Code	4
Input-Output Format	5
Peripheral Interrupt	5
The Write Statement	5
Delimiters	5
Free-Field Output Format	6
The Read Statement	7
Free-Field Input Format	8
Format Statements	10
The Format Syntax	10
Data Output Specifications	11
Text Character Fields	13
Output Edit Specifications	13
Printer Character Set	16
Display Character Set	19
Single Character Output	19
Text in Format and Write Statements	20
Data Input Specifications	21
Input Edit Specifications	23
The Conversion Statement	25
The List Statement	25

Chapter 3: Binary I/O Operations

The Write Binary Statement	27
The Read Binary Function	28
The Read Status Function	29
KDP Status Bits	29
Tape-Drive Status Bits	31
Interface Status	32
The Write Control Statement	33
Alternate Character Set	33

Chapter 4: HP Interface Bus

Overview of the HP-IB	35
Bus Messages	36
The Bus Card	36
HP-IB Addresses	37
Controlling Listeners and Talkers	39
Data Input	41
Interface Status Bits	43

Appendix

Binary Coding and Conversions	45
Select Codes	47
ASCII Character Codes (table)	48
Decimal Key Codes (table)	49
Sales and Service Offices	50
General I/O Syntax	54
Subject Index	56
General I/O Error Messages	(inside-back cover)

Figures

Installing a ROM Card	1
The Calculator I/O Scheme	3
KDP Status Bits	29
Tape-Drive Status Bits	31
98032A Status Bits	32
98032A Control Bits	33
Connecting Bus Cables	37

Tables

General I/O ROM Operations	ii
Data Output Specifications	11
Output Edit Specifications	13
Internal Printer Character Set	16 – 17
Data Input Specifications	21
Input Edit Specifications	23
Factory Set Select Codes	47
ASCII Character Codes	48
Decimal Key Codes	49

Chapter 1

General Information

The General I/O ROM (Read Only Memory) provides a 9825A Calculator with the statements and functions for controlling most external peripheral devices. The ROM also permits formatting printouts on the internal printer and adds many additional characters for use with the printer and display. In addition to direct peripheral control, the General I/O ROM permits controlling individual peripherals via the HP Interface Bus.

Inspection Procedure

The General I/O ROM is packaged with other ROMs in a single ROM card. Please verify that the correct ROM card is supplied and inspect the card for physical damage. To check operation of the ROM, see the 9825A System Test Booklet. If any damage or electrical malfunction is found, contact the nearest HP Sales and Service Office; locations are listed near the back of this manual.

Installation

The ROM card can be plugged into any of the four slots on the front of the calculator (below the keyboard). To install the ROM, first switch off the calculator. Then, with the ROM label right-side-up, slide the card into the slot until it's even with the front of the calculator. Now, switch the calculator on.



Installing a ROM Card

Memory Usage

The General I/O ROM uses 56 bytes of calculator Read/Write Memory. This loss is indicated by the amount of available memory displayed after each list operation.

Syntax Guidelines

The following general conventions apply to the statement and function syntax listed in this manual.

- Dot matrix — All items appearing in `dot matrix` must be entered as shown.
- [] — All items in square brackets are optional and explained in the following text.
- All General I/O statements and functions can be executed either from the keyboard or a program.

A summary of General I/O syntax is listed at the back of this manual.

Error Messages

The General I/O ROM adds error messages G1 through G9 to the calculator error list. Meanings for each error are on the inside back cover.

Chapter 2

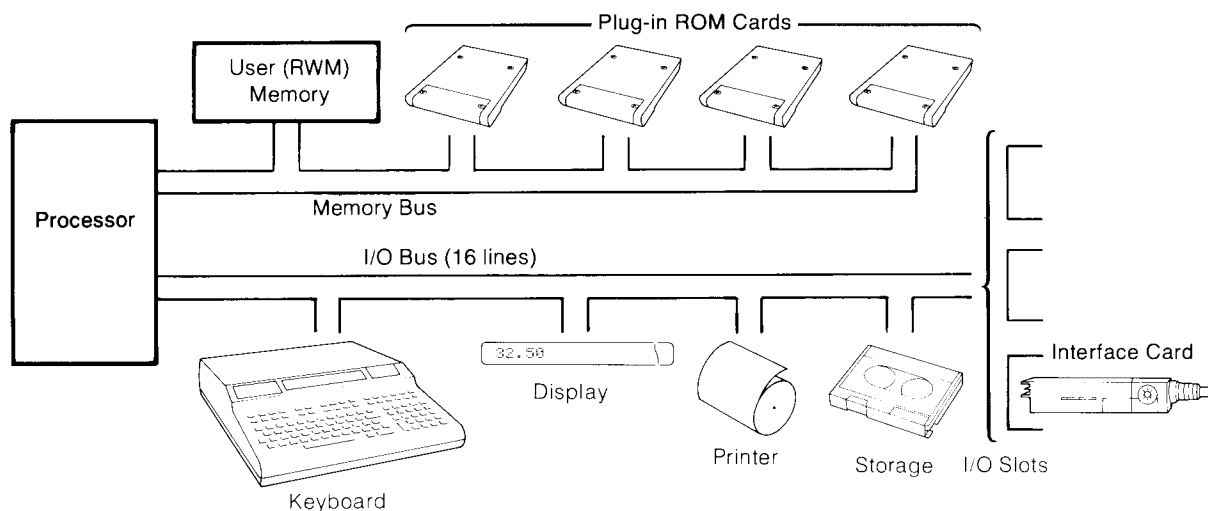
Formatted I/O Operations

This chapter describes the General I/O statements which handle data in specialized formats. If you are using a General I/O ROM to control HP 9800-series peripherals, additional instructions are provided in an operating note furnished with each HP 98032A Interface Card.

Although interfacing a peripheral device to a calculator requires both hardware and software, this manual is concerned primarily with describing the input/output software available. To acquaint you with some interfacing terms, however, a brief discussion of the calculator input/output scheme is provided next.

Calculator I/O Scheme

The general calculator I/O scheme is shown in the next figure. Referring to the figure, notice that the I/O Bus transfers data between the calculator processor and peripheral devices. All incoming data is transferred through the processor before it is stored in memory. In this manual, the term "I/O" is always used with reference to the calculator.



The Calculator Input/Output Scheme

As shown in the figure, each external device must be connected to the calculator via an appropriate interface card and cable. The card plugs into any of the I/O slots in the calculator's back panel. As shown, plug-in ROM cards become a part of the calculator's memory; each card adds up to 8k bytes of ROM. Finally, notice that external devices share the I/O Bus with the internal peripheral devices (printer, display, etc.). So the internal peripherals respond to some General I/O operations, in addition to their specific commands (print, display, etc.).

Select Code

As just described, each external device is connected to the calculator via the same I/O bus. Since all external devices are "party-lined" on the same bus, each device is assigned a unique address, or select code, so that the correct device responds to each I/O operation.

For all external peripherals, the select code is an integer number from 2 through 15, which is specified in each I/O operation and decoded by the corresponding interface card. Each interface card has a switch permitting the user to set any one of many different codes. A list of recommended codes is in the Appendix.

Each internal peripheral has a fixed select code which is automatically specified by standard calculator statements (display, print, etc.). Both the display and the keyboard respond to select code 0, the tape drive responds to select code 1, and the printer responds to select code 16.

The select code can be specified in the form of a constant, a variable, or an expression. Each of these write statements is addressed to the device responding to select code 9 ♦

```
0: wrt 9,A,B,C
1: wrt S,A$ (S = 9)
2: wrt 3r1,A,B (r1 = 3)
```

For some operations, other numbers can be combined with the select code to form a single expression. A general select code syntax is:

```
cc[dd[ee]][.f]  cc= one or two-digit select code.
                  dd= optional HP-IB address code (must be two digits). See Chapter 4.
                  ee= optional HP-IB extended address code (use with Extended I/O
                      ROM only). See The Extended I/O Programming Manual.
                  .f= format number (read and write statements only).
```

Input-Output Format

The I/O bus connecting the processor with internal and external peripherals contains 16 data lines. Data is transmitted in a 16-bit-parallel, character-serial fashion. The I/O operations described in this chapter send and receive data in standard 8-bit (US)ASCII code.¹ The calculator sends and receives one 8-bit character at a time. The parity (most-significant) bit is not used with formatted I/O operations. You'll find a brief introduction to binary coding and conversion in the Appendix. It's important to know the I/O formats available with each interface card; see the card's installation and service manual for details.

Peripheral Interrupt

Since the General I/O ROM is intended for use in systems where the calculator is the controlling device, there is no provision for peripheral interrupt operation (i.e., when an external device can call for an I/O operation). The calculator must be in complete control of each device while that device is involved in data transfer.

To enable the peripheral-interrupt capability available on some interface cards, such as the 98032A Interface, the Extended I/O ROM must be used. Refer to the Extended I/O Programming Manual for more information.

The Write Statement

```
write select code [ , format no. ][ : expression or text1 [ : expression or text2... ] ]
```

The write statement outputs the characters, signs, and decimal point of each item to the specified peripheral. Each item in the list can be a numeric expression, text, or a string name (when the String ROM is in use). The value of each item is output in a free-field format unless a format statement is in effect. The format number can be an integer from 0 to 9 and references a similarly numbered format statement; format statements are described later.

Delimiters

A delimiter is a character that is used either to separate one expression from another inside the list or to terminate the list. The space (Δ) and the carriage-return line-feed (CR/LF) are delimiters that are automatically output during the execution of each write statement. The space is used to separate items within the list, and the CR/LF is used to terminate the list.

¹American Standard Code for Information Interchange.

Free-Field Output Format

The free-field output format is automatically set whenever the calculator is switched on or  is pressed. Free-field is also set whenever run or erase all commands are executed. Each write statement references the free-field format until an appropriate format statement is executed; then the specifications in the format statement override free-field.

The free-field format causes each numeric expression to be output, right-justified, in an 18-character field. A CR/LF is given after every fourth expression output and after the last parameter is output. The form in which expressions appear is determined by the current fixed (`fxd`) setting. Characters within quotes and strings are output as "free text", which means that the 18-character fields are not used.

Here are some examples using free-field. The output device used here is an HP 9866B Printer, connected via a 98032A Interface set to select code 6. For more examples, see "Format Statements" in this chapter; also see Chapter 4.

First, press 

Then load and run this program 

The output is shown below.

```
0: fxd 4;10+A
1: wrt 6,A,A/9,
  A/A1e4
2: end
```

10.0000	1.1111	0.0001 CR/LF
└──────────┘	└──────────┘	└──────────┘
18-char. field	18-char. field	18-char. field

Now change line 1 and run the program again.

```
0: fxd 4;10+A
1: wrt 6,A,A↑2,
  A↑4,A↑6,A↑8
2: end
```

10.0000	100.0000	10000.0000	1000000.0000
100000000.0000 CR/LF			CR/LF

Notice that a CR/LF is automatically output after the last item and after each four items.

To output text between numeric expressions, change line 1 again and run the program ♦

```
0: fxd 4;10→A
1: wrt 6,A,"mete
  rs",A1e2,"cm",
  A1e4,"mm"
2: end
```

```

      10.0000meters      1000.0000cm      100000.0000mm CR/LF
      └──────────┬──────────┬──────────┘
      w = 18      w = 18      w = 18
```

As shown in the printout, text is output between numeric fields with free-field. For complete control of numeric and text outputs, use format statements.

The Read Statement

```
read select code[ , format no.] variable1 [ , variable2...]
```

The read statement inputs and stores data from a specified peripheral¹. The number of variables in the list indicates how many data items to read. String variable names (but not substrings) can be used if the String ROM is in use. Each numeric data item can consist of the digits 0 through 9, plus and minus signs, a decimal point, and an "E" character (upper and lower case). All other characters are treated as input delimiters. The data item itself can assume the same form as any number entered from the keyboard.

The format number can be used to reference any of ten format statements. If a format number is not specified, and if a format statement has not been previously executed, a free-field input format is automatically used.

¹The calculator keyboard can not be used to input data with a read statement. Use an enter statement instead.

Free-Field Input Format

The free-field input format is set whenever the calculator is switched on or  is pressed. Free-field is also set whenever run or erase all commands are executed. Using free-field allows reading numeric data in virtually any form, provided that each item is followed by at least one delimiter (non-numeric character). For each variable in the read list, the calculator ignores all leading non-numeric characters until a numeric character is read. Then, after reading the data item, reading any non-numeric character terminates and stores the data item. Also, reading a LF terminates the entire read operation.

When free-field is used, the calculator cannot input non-numeric characters unless a string variable (String ROM) is specified. Then all characters are input until the dimensioned string length is filled. Reading a LF (during free-field) automatically terminates reading a string. Also see "Format Statements" later in this chapter, and the String Variables Programming Manual.

Here is a brief description of the delimiters used in the free-field input format:

- If the first character is a comma, the corresponding variable in the read statement is skipped and flag 13 is set. The skipped variable keeps its original value.
- All non-numeric characters preceding a data item are ignored. If a continuous string of the same non-numeric character is received, each variable in the read list is skipped and flag 13 is set after 2^{16} characters are read.
- An HT (horizontal tab) character causes the calculator to skip all characters until a LF is read.
- Whenever a LF is read (and it does not correspond to a preceding HT) the read statement is terminated and flag 13 is set.
- An upper- or lower-case "E" character, when part of any of these forms, causes the preceding data item to be multiplied by the power of 10 indicated:

(data item)E(one or two digits)
 (data item)E(+ or - and one or two digits)
 (data item)E(space and one or two digits)

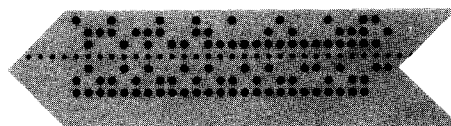
For example, any of these data items will be read as the number "1234" (Δ indicates space):

```
1.234E3
1.234E $\Delta$ 3
1.234E+3
1234000E-3
```

- Spaces read within numeric characters are ignored.
- The CR (carriage return) is always ignored (skipped over) during a read operation.

The following program can be used to input and print the data items on this ASCII-coded paper tape $\blacktriangledown$

1.23,2.34,3.45,4.56,5.67 (CR) (LF)



The program assumes that the tape reader interface is set to select code 3. Run the program twice to obtain both printouts.

```
0: 0+r1+r2+r3
1: red 3,r1,r2,
   r3
2: fxd 2
3: prt r1,r2,r3
4: spc 2!end
```

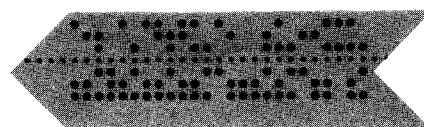
Notice that the read operation is terminated when the required number of items have been read, or when a LF is read. In the second printout, r2 was skipped and so retained its original value (the run command clears all variables).

```
1.23
2.34
3.45
```

```
4.56
0.00
5.67
```

As another example, use the previous program to read this tape $\blacktriangledown$

12.81,13.35,(HT) 200.5,(LF) 25E5



The calculator ignores all data between HT and LF characters. More examples using read statements are in the next section and in Chapter 4.

```
12.80
13.35
2500000.00
```

Format Statements

Use of format statements provides more flexible and complete control of write and read statements. A format statement must be executed before the I/O statement referencing it and provides a list of specifications for use by the I/O statement. Then, as the I/O statement is executed, it references the last-executed format statement rather than free-field.

The Format Syntax

`fnt[format no.][spec1[,spec2...]]`

The format number can be used to identify the statement for successive write or read statements. Each format number can be an integer constant from 0 to 9; if not specified, 0 is assumed. When the calculator is reset (power on, **RESET**, run, or erase all) format number 0 is automatically assigned to specify free-field. Also, the syntax `fnt[format no.]` assigns the format number to free-field.

For example, if the numbered format statements shown here appear in a program, then the write statements in lines 6 and 9 reference format no. 2, and the read statement in line 7 references format no. 9. The last two I/O statements reference free-field, however, since they do not specify a format no. and there is neither a format 0 nor an unnumbered format statement.

Each successive format statement replaces any previous, similarly-numbered format statement. A format statement with no parameters sets free-field. In the example sequence on the right, the first unnumbered format statement specifies free-field for the read statement. The write statements in lines 16 and 18 reference the second unnumbered format (line 14). The last two write statements reference the third unnumbered format (line 19).

```

0: fnt 2,f10.2
1: fnt 5,e15.0
2: fnt 9,f6
.
.
.
6: wrt 6.2,A,B
7: red 3.9,r1
8: for I=1 to 9
9: wrt 16.2,r1
10: next I
11: wrt 16,A,B
12: red 7,A$

13: fnt ;red 3,
    A,B,C,D,E
14: fnt c50
15: fnt 1,5f10.0
16: wrt 6,A$
17: wrt 6.1,A,B,
    C,D,E
18: wrt 6,B$
19: fnt "end",5/
20: wrt 6
21: wrt 16;end

```

Note that if an unnumbered format statement is used, all subsequent input or output operations that do not reference a numbered format statement will reference the unnumbered format statement. To avoid undesired referencing, all format statements should be numbered.

Data Output Specifications

Data specs determine the form in which each data item is output. Most data specs determine whether a number is output in fixed point or floating point, the number of digits to the right of the decimal point, and the character field width in which the number appears.

These data specs are available:

[r]f w. d	Specifies fixed-point format.
[r]e w. d	Specifies exponential (floating-point) format.
[r]fzw. d	Specifies fixed-point format with leading zeros in each field. Negative numbers are not allowed with this format.
[r]b	Specifies binary output of single characters. Examples begin on page 14.
[r]cw	Specifies a character field-width for either text or a string variable (String ROM).

- r is an optional repeat factor (see examples). If r is omitted, 1 is assumed.
- w indicates total field width (in characters). If w is omitted, leading spaces are deleted from the field.
- d indicates the number of digits (0 through 11) to the right of the decimal point. If d is omitted or if d is greater than 11, the current fixed or float setting is used.
- w, d and r parameters must be positive integer constants.

For example, the data spec `f8.2` specifies a fixed-point number with two digits to the right of the decimal point. The number appears (right-justified) in an eight-character field. If d is 0, the decimal point is not output. A number output under a data spec is always rounded according to the number of decimal places specified. The free-field format is equivalent to `4f18`.

Some guidelines should be observed in selecting w and d. The minus sign, decimal point, and exponent are part of each number and must fit in the field width specified by w. For floating-point outputs, w should be greater than or equal to d+7. If the calculator cannot output the data in the field width available, the field is filled with \$s.

The following examples of formatted output use the calculator's internal printer as the output device (select code 16).

Lines 1 and 2 output three numbers. Since the entire format statement is referenced for each number, a CR/LF is output after each.

```
0: 10+A;.5+B;
   16+C
1: fmt f10.2
2: wrt C,A,B,
   AB↑2
```

```
10.00
 0.50
 2.50
```

w = 10

Lines 3 and 4 output the same numbers as above, but use of a repeat factor matches the data spec to the output list – so all numbers appear on the same line.

```
3: fmt 3f5.2
4: wrt C,A,B,
   AB↑2
```

```
10.00 0.50 2.50
└──┬──┬──┘
w = 5 w = 5 w = 5
```

This sequence outputs numbers in various formats. Line 10 increments the format number, so that all formats are referenced. Outputs are shown below.

```
5: fmt f8.1,e8.1
6: fmt 1,4f4.0
7: fmt 2,4f.0
8: fmt 3,fz16.2
9: wrt C,A,2A↑2,
   3A↑3,4A↑4
10: if (C+.1+C) <
    16.4;jmp -1
11: end
```

Format 0: Format 0 is referenced twice in line 9. Note that an upper-case E precedes the exponent.

```
10.0 2.0E 02
3000.0 4.0E 04
```

Format 1: Notice that \$s fill the last field since the number is too large for the field width specified.

```
10 2003000####
```

Format 2: Notice that omitting w deletes leading spaces from each field.

```
10200300040000
```

Format 3: The last format outputs the same numbers in a single column with leading zeros in each field.

```
0000000000010.00
00000000000200.00
000000000003000.00
0000000000040000.00
```

Text Character Fields

The `c` data spec specifies a fixed character field for corresponding text or a string variable in the write statement. The characters are output right-justified in the field. If the field is too small, it's filled with `$s`.

In the example on the right, the first two lines of text fill the field; the third line is too short; the last line is too long.

```
0: fmt c10,f5.1
1: wrt 16,"Funct
   ion =",A
2: wrt 16,"Polar
   ity =",C
3: wrt 16,"Phase
   =",B
4: wrt 16,"Avera
   ge Value =",D+
   E+E
```

```
Function = 0.0
Polarity = 0.0
Phase = 0.0
$$$$$$$$$ 0.0
```

$w = 10$

Output Edit Specifications

These edit specs are used to control the placement of output data and to output character strings:

[r] x	Outputs a blank character space.
[r] /	Outputs a CR/LF.
[r] "TEXT"	Outputs the ASCII characters within quotes. See the following examples and the Appendix.
z	Suppresses the automatic CR/LF output after each write statement.

Any combination of specs can appear in the same format statement; each spec must be separated by a comma. Most of the specs can be duplicated `r` number of times by using the repeat factor.

For example, referencing this statement causes the first fixed-point field to appear twice, followed by four character spaces, and then another fixed-point field.

```
0: fmt 2f6.2,4x,
   f10.1
```

The program on the right shows how to format a three-column table using the internal printer. You can use the same method, of course, with external output printers.

Suppose that the entered values are 5, 9, 15, 25, and 64. When formats 0, 1 and 2 are referenced in line 6, CR/LFs and the column headers are output. Notice that spaces are used to position the characters, and that a `b` data spec causes the binary value of decimal 29 to be output, printing the character `²`. (The complete character set is shown later.) Then line 7 references format 3 for the numeric outputs. Here, spacing is determined by the data field widths.

```

0: 1→A
1: ent r1,r2,r3,
   r4,r5
2: fmt 2/
3: fmt 1,2x,"n",
   3x,"n",b,3x,"1/
   n"
4: fmt 2,16"- "
5: fmt 3,f3.0,
   f6.0,f6.3
6: wrt 16;wrt
   16.1,29;wrt
   16.2
7: wrt 16.3,rA,
   rA↑2,1/rA;jmp
   (A+1→A)=6
8: wrt 16;end

```

n	n²	1/n
5	25	0.200
9	81	0.111
15	225	0.067
25	625	0.040
64	4096	0.016

The next program, which produces a table of trigonometric values, uses a 9866B Printer. Although the program is similar in programming technique to the previous one, here are some exceptions (a sample printout is on the next page):

- Spacing between table columns is achieved by placing blocks of spaces between each number, rather than by adjusting the width of each data field.
- Notice the use of text (in line 5) to enclose the radians entries.

```

0: fmt 1,5;/wrt
  6.1
1: fmt 30x,"Trig
  onometric Table
  ",2;/wrt 6
2: deg;sfa 14;
  0→A→X
3: fmt 2,6x,"Deg
  rees (Radians)"
  ,12x,"Sine",
  10x,"Cosine",
  12x,"Tangent",/
4: wrt 6.2
5: fmt 3,9x,f3.0
  ," (",f4.2,")",
  15x,f6.3,9x,
  f6.3,10x,e10.3
6: wrt 6.3,X;πX/
  180;sin(X);cos(
  X);tan(X)
7: A+1→A
8: if A=3;0→A;
  fmt ;wrt 6
9: if 180>X;X+
  180→X;eto 6
10: wrt 6.1
11: end

```

Trigonometric Table

Degrees (Radians)	Sine	Cosine	Tangent
0 (0.00)	0.000	1.000	0.000E 00
10 (0.17)	0.174	0.985	1.763E-01
20 (0.35)	0.342	0.940	3.640E-01
30 (0.52)	0.500	0.866	5.774E-01
40 (0.70)	0.643	0.766	8.391E-01
50 (0.87)	0.766	0.643	1.192E 00
60 (1.05)	0.866	0.500	1.732E 00
70 (1.22)	0.940	0.342	2.747E 00
80 (1.40)	0.985	0.174	5.671E 00
90 (1.57)	1.000	0.000	9.999E 99
100 (1.75)	0.985	-0.174	-5.671E 00
110 (1.92)	0.940	-0.342	-2.747E 00
120 (2.09)	0.866	-0.500	-1.732E 00
130 (2.27)	0.766	-0.643	-1.192E 00
140 (2.44)	0.643	-0.766	-8.391E-01
150 (2.62)	0.500	-0.866	-5.774E-01
160 (2.79)	0.342	-0.940	-3.640E-01
170 (2.97)	0.174	-0.985	-1.763E-01
180 (3.14)	0.000	-1.000	0.000E 00

Printer Character Set

The program on page 14 shows how to print a new character on the internal printer. As shown in the next program, the `b` data spec can be used to print any character in the printer's character set.

The program on the right prints numbers 0 through 127. It also outputs each number's binary-equivalent value, causing the printer to output its entire character set. A complete printout is shown.

```
0: wrt 16, "CHARA
   CTER SET"
1: fmt f3.0,6x,b
2: wrt 16,I,I
3: if (I+1>I)<12
   8: jmp -1
4: spc 2: end
```

Internal Printer Character Set

CHARACTER SET			
0	4	31	⌘
1	5	32	(space)
2	6	33	!
3	7	34	"
4	8	35	#
5	9	36	\$
6	A	37	%
7	B	38	&
8	C	39	'
9	D	40	(
10	(line feed)	41	)
		42	*
		43	+
11	E	44	,
12	F	45	-
13	(carriage return)	46	.
14	G	47	/
15	H	48	0
16	I	49	1
17	J	50	2
18	K	51	3
19	L	52	4
20	M	53	5
21	N	54	6
22	O	55	7
23	P	56	8
24	Q	57	9
25	R	58	:
26	S	59	;
27	T	60	<
28	U	61	=
29	V	62	>
30	W	63	?

64	@	95	~
65	A	96	
66	B	97	a
67	C	98	b
68	D	99	c
69	E	100	d
70	F	101	e
71	G	102	f
72	H	103	g
73	I	104	h
74	J	105	i
75	K	106	j
76	L	107	k
77	M	108	l
78	N	109	m
79	O	110	n
80	P	111	o
81	Q	112	p
82	R	113	q
83	S	114	r
84	T	115	s
85	U	116	t
86	V	117	u
87	W	118	v
88	X	119	w
89	Y	120	x
90	Z	121	y
91	[	122	z
92	\	123	{
93	]	124	
94	^	125	}
		126	~
		127	

Referring to the printout, notice that most of the characters and decimal numbers correspond to ASCII codes (see the table in the Appendix). But notice that two codes, 10 and 13, do not generate printer characters. Instead, 10 causes the printer to linefeed and 13 is ignored.

Another method of listing the printer character set is to print a string variable (String ROM) containing character values from 0 through 127. Here's the program and printout.

```

0: dim A$(128)
1: 0=I
2: char(I)→A$(I+
  1)
3: if (I+1→I)<12
  8: jmp -1
4: prt A$
5: end

```

10 13

```

#0XNoBΓΠΔσ↓λμτϕ
00δAδAδ0δ000Eε2f%
!"#$%&'()*+,-./
0123456789:;<=>?
@ABCDEFGHIJKLMNO
PQRSTUVWXYZ[ ]^_
`abcdefghijklmnopqrstuvwxyz{|}~

```

Notice that with a print statement, the printer generates characters for string values 10 and 13. But if the string is output to the printer using a write statement (see the next line and printout), the printer does a line feed for 10 and ignores 13.

```

4: wrt 16,A$

```

```

#0XNoBΓΠΔσ
λμτϕ00δAδAδ0δ000E
12 14

```

This last printout shows another important point: The internal printer does line feeds during a print statement to accommodate long strings or text, but it does not automatically do line feeds for write or write binary statements. The second line in this last printout was printed because of the line feed automatically output after the write statement.

Similarly, line feeds must be sent to the display when using write or write binary statements. If not, output data will be lost. For example, execute this line:

```
wrt 0,"Keeper"
```

```
Keeper
```

Now add `fnt zi` and execute the line again:

```
fnt zi:wrt 0,"Keeper"
```

```
␣
```

Display Character Set

The same character set available on the internal printer is available for use in the display by using the `b` data spec. Also, adding 128 to each item output with `b` causes a character-editing cursor (▄ or ▄▄) to flash over each corresponding character. The cursor which appeared last in the display is used for this operation.

For example, this sequence ▄ displays the message "Program Secured!" in a field filled with flashing cursors. The resulting display is shown below.

```
5: 32+128+C
6: fnt 8b,"Progr
   am Secured",9b
7: wrt 0,C,C,C,
   C,C,C,C,C,33,C,
   C,C,C,C,C,C,C
8: end
```



Single Character Output

To output single ASCII characters without CR/LFs, use either the write binary statement described in Chapter 3, or one of these methods:

```
fnt "characters",z;wrt select code
```

or

```
fnt b,z;wrt select code, decimal value
```

For example, the plug-in unit in an HP 3480 Digital Voltmeter can be remotely controlled by using appropriate ASCII characters. The plug-in unit is connected via an HP 98032A I/O Interface Card (select code 2). The DVM mainframe is connected via an HP 98033A BCD Interface (select code 3).

In this program sequence, first the 3484A Plug-in Unit is set to the 100 mV range by sending an ASCII "EOM" (decimal 4); then a data reading is taken. Line 8 sets the plug-in unit to the 1 V range by sending an ASCII "@" (decimal 64) before another reading is taken.

```
5: fnt 1,b,z
6: wrt 2.1,4
7: red 3,A,B
8: wrt 2.1,64
9: red 3,A,C
```

Text in Format and Write Statements

As shown in the preceding examples, text to be output can be included in format or write statements. For programming convenience, however, it's recommended that text **not be placed**: 1) in both format and write statements for a given output, and 2) as the last item in a write statement. The following examples show why.

Suppose that we wish to output this line on an external printer ("DM" indicates Deutsche Marks):

```
Sales =      10DM for the last      4 months.
```

└──────────┘
└────────┘
w = 6
w = 3

This sequence could be used with no problem (assume that A = 10 and B = 4):

```
0: fmt "Sales =",f6.0,"DM for the last ",f3.0," months."
1: wrt 6,A,B
```

Notice that all text is in the format statement. If all text were placed in the write statement:

```
0: fmt f6.0,f3.0
1: wrt 6,"Sales =",A,"DM for the last ",B," months."
```

This output would result:

```
Sales =      10DM for the last      4
months.
```

Since there were more output items than format specs, a CR/LF was output after the last spec was referenced. Then "months" was output, followed by another CR/LF, at the end of the write statement.

Now suppose that we wish to output the same line, but have each write statement determine the units: "DM...months" on one line and "K\$...days" on another line. Here is one method and its results:

```
0: fmt "Sales =",f6.0," for the last ",f3.0,c
1: wrt 6,A,"DM",B," months."
2: wrt 6,A,"k$ ",B," days."
```

```
Sales =      10 for the last DM      4 months.
Sales =      10 for the last k$      4 days.
```

The problem of extra CR/LFs is avoided here by referencing data spec `c`. But notice that text preceding each spec is output **before** text preceding each variable. So the units DM and K\$ are in the wrong place! To output the units in the correct order, use more `c` specs:

```
0: fmt "Sales =",f6.0,c," for the last ",f3.0,c
1: wrt 6,A,"DM",B," months."
2: wrt 6,A,"k$",B," days."
```

```
Sales =    10DM for the last    4 months.
Sales =    10k$ for the last    4 days.
```

When using text in format and write statements, remember:

- Place all "fixed" text in the format statement.
- Place all "variable" text in the write statement and reference `c` data specs.
- Text (and all other edit specs) preceding each data spec are always output **before** text preceding the corresponding item in the write statement.

Data Input Specifications

Data specs can be used to determine which characters are input from a data input string, and in what form the data will appear. When a format statement is referenced by a read statement, the read operation is not terminated until a LF character is read (unless the edit spec `x` is used). A general data input spec syntax is:

`[r]f[w]`

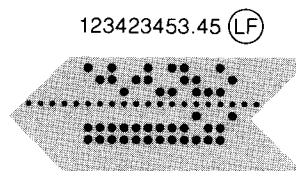
- `r` is the number of consecutive times the spec is to be used (if `r` is 1 it may be omitted). `r` and `w` must be integer constants.
- `w` is the width of the data field to be read. Omitting `w` specifies free-field read for the corresponding item(s).

A data spec like `f10` calls for reading ten numeric characters; all non-numeric characters which precede a numeric are counted but not entered. If an "E" is read within a numeric field, a number of the form `1e dd` is entered.

The following examples use an HP 9883A Tape Reader via a 98032A Interface set to select code 3. The paper tapes shown here are coded in ASCII.

To read this tape containing three data items ♦
run this program.

```
0: fmt 3f4
1: red 3,A,B,C
2: fxd 2
3: prt A,B,C
4: end
```



```
1234.00
2345.00
3.45
```

Notice that four characters were read for each data item.

Now delete line 0, and run the program to read
the same tape again.

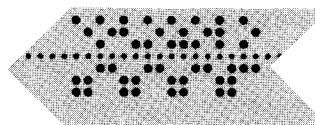
```
123423453.45
0.00
0.00
```

Referencing free-field inputs the entire data string into the first variable; reading the LF terminates the read operation, skipping the last two variables.

To read this tape ♦
run this program.

```
0: 1→A:fmt f3
1: red 3,rA:jnp
(A+1→A)>4
2: fxd 0
3: prt r1,r2,r3,
r4
4: end
```

12 (CR) (LF) 34 (CR) (LF) 56 (CR) (LF) 78 (CR) (LF)



```
12
34
56
78
```

Since there is a LF character after each data item, the read statement had to be repeated to input each item.

Input Edit Specifications

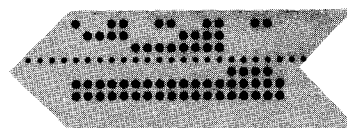
The following edit specs can be used to increase input format flexibility. An example use of each spec follows.

<code>z</code>	Cancel LF as terminator.
<code>[r]x</code>	Skip character.
<code>[r]/</code>	Skip data item.
<code>[r]c w</code>	String field width.

The `z` spec causes the calculator to read only the number of characters specified in the rest of the format statement. The read operation is automatically terminated after the characters are read, without the need for a LF character.

For example, this tape has 9 two-digit numbers, but no LF character ▶

122334455667788990



Run this program to read and print all nine items.

```

0: 1→A                                12
1: fmt z,f2                            23
2: red 3,rA;jmp                        34
   (A+1→A)>9                           45
3: 1→A;fxd 0                          56
4: prt rA;jmp                          67
   (A+1→A)>9                            78
5: end                                89
                                         90

```

The `x` spec causes the calculator to skip (not count) `r` number of characters. For example, to read and skip alternate items on the previous tape, run this program. Notice that the entire format statement is referenced for each data entry.

```

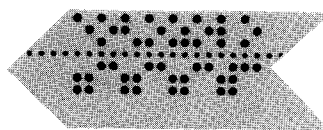
0: fmt z,f2,2x                          12
1: red 3,A,B,C,                          34
   D,E                                  56
2: fxd 0                                78
3: prt A,B,C,D,                          90
   E
4: end

```

The $\circ$ spec causes the calculator to skip all data which precedes r number of CR/LF characters.

For example, to read only the first and last items on this tape $\blacktriangleright$
run the program shown below.

12 (CR) (LF) 34 (CR) (LF) 56 (CR) (LF) 78 (CR) (LF)



```
0: fmt f2,3/,f2
1: red 3,A,B
2: fxd 0
3: prt A,B
4: end
```

12
78

The $\circ$ spec indicates the number of characters to input into a dimensioned string variable (String ROM). All characters are entered until either the dimensioned string is filled, or w characters are read. When $\circ$ is used, reading a LF does not terminate a string input.

Reading into a string variable is a convenient way of storing all characters in a data item, including non-numerics. Later, any character or portion of the string can be evaluated using String ROM operations.

For example, the HP 98033A BCD Interface inputs data from a measurement device, such as an HP 3480 Digital Voltmeter, and transfers ASCII-coded data to the calculator in this 16-character format:

(sign) D₁ D₂ D₃ D₄ D₅ D₆ D₇ D₈ E (sign) D₉, (overload) D₁₀ (LF)

Since a delimiter follows each data item, it's easy to read the numeric items by using a read statement with free-field format.

```
0: red 3,r1,r2
```

By reading into a string variable, however, you can evaluate any portion (substring) of the string later in the program. Running this sequence stores the data reading in A, the overload numeric character in B, and the function code in C.

```
1: dim A#[20]
2: fmt c16
3: red 3,A#
4: val(A#[1,12])
  →A
5: val(A#[14,
  14])→B
6: val(A#[15,
  15])→C
```

See the String Variables Programming Manual for other String ROM operations.

The Conversion Statement

`conv [code1, code2[, code3, code4]...]`

The conversion statement sets up a character replacement table for use with read and write statements. Up to 10 pairs of decimal codes can be specified at a time. Each new conversion statement cancels the previous table and sets up a new one. A conversion statement with no parameters cancels any previous table.

In this program sequence ♦

line 8 sets up a conversion table which changes the spaces (decimal 32) output between numbers in line 10 to asterisks. Then line 11 cancels the previous table.

```
8: conv 32,42
9: fmt 3f10.2
10: wrt 6,A,B,C
11: conv
```

As another example, line 6 on the right specifies that whenever an ASCII "/" (decimal 47) is read in line 7, it will be converted to an ASCII "HT". Also, whenever an ASCII "NULL"

```
6: conv 47,9,0,
  13
7: red 3,A$
```

is read, it will be converted to an ASCII "CR" (decimal 13). Changing these input delimiters causes the calculator to skip all characters read between a / and a LF, and ignore all NULL characters. Input delimiters are described on page 8.

The List Statement

A select code parameter can be used with the list statement when the General I/O ROM is plugged in, enabling program listing on a peripheral output device. The new list syntax is:

`list[#select code][, line nos.]`

The optional line numbers remain as described in the operating and programming manual.

A listing of the same program described on page 15 is shown next, output to a 9866B Printer via a 98032A Interface set to select code 6. Referring to the listing, notice that three CR/LFs are automatically output before and after the listing.

```

0: fmt 1,5;/wrt 6.1
1: fmt 30x,"Trigonometric Table",2;/wrt 6
2: deg;sfa 14;0+A+X
3: fmt 2,6x,"Degrees (Radians)",12x,"Sine",10x,"Cosine",12x,"Tangent",/
4: wrt 6.2
5: fmt 3,9x,f3.0,"(",f4.2,")",15x,f6.3,9x,f6.3,10x,e10.3
6: wrt 6.3,X,πX/180,sin(X),cos(X),tan(X)
7: A+1→A
8: if A=3;0→A;fmt ;wrt 6
9: if 180>X;X+10→X;eto 6
10: wrt 6.1
11: end
*24841 ←checksum

```

If not needed, the CR/LFs and checksum can be suppressed by adding a decimal point and a non-zero digit to the select code. For example, use this statement to list the last half of the program shown above:

list#6.1,5

```

5: fmt 3,9x,f3.0,"(",f4.2,")",15x,f6.3,9x,f6.3,10x,e10.3
6: wrt 6.3,X,πX/180,sin(X),cos(X),tan(X)
7: A+1→A
8: if A=3;0→A;fmt ;wrt 6
9: if 180>X;X+10→X;eto 6
10: wrt 6.1
11: end

```

The checksum and CR/LFs are not suppressed when this list statement is used with the internal printer (select code 16).

Chapter **3**

Binary I/O Operations

Binary I/O operations are available to read and write individual data characters, and to transmit or receive control information using interface status lines. Binary I/O operations do not reference format or conversion statements. The data I/O modes and status line meanings are determined by the interface card; refer to each card's installation and service manual for details.

The Write Binary Statement

`wtb select code ; expression or text1 [ ; expression or text2...]`

This statement outputs the 16-bit, binary-equivalent result of each expression or each character of text. The usable range for each expression is an integer from -32768 through 32767. If the interface handles data in an 8-bit fashion, as does the HP-IB Interface or the 98032A Interface with byte mode, only the 8 least-significant bits of each integer are accepted by the interface. Usable range can be 0 - 65535 if flg 14 is set. Setting flg 14 overrides the overflow error normally encountered. (This is for any operation accepting 16 bit expressions.)

Here is a short program which uses write binary to print the same title and internal-printer character set as shown on pages 16-17. A portion of the printout is also shown below. Notice that a line feed (decimal 10) must be sent at the end of each statement to print the line (line feeds are not automatically sent after binary output operations).

```
0: 0→A
1: wtb 16,"CHARA
  CTER SET",10
2: wtb 16,A,10
3: if (A+1→A)<12
  8;jmp -1
4: end
```

CHARACTER SET

```
4
6
X
N
α
θ
Γ
ñ
Δ
σ
```

The HP 9871A Printer has a variety of functions that are controlled by using sequences of decimal codes. Here are program segments that set the top of form (decimal codes 27 and 84) in line 6 and form length (27,70, int (n/64),int n) in line 7. For a form length of 8 inches, n is 768. Line 15 outputs a form feed instruction. The printer interface is set to select code 6.

```
6: wtb 6,27,84
7: wtb 6,27,70,
  int(768/64),768
```

•
•
•

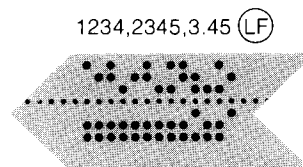
```
15: wtb 6,12
```

The Read Binary Function

rdb (select code)

Read binary is a function that inputs one 16-bit character and stores its integer-decimal value. If the interface handles data in an 8-bit fashion, the eight most-significant bits are read as zeros.

For example, to read this ASCII-coded paper tape ▶



Run the next program ▶

The program assumes that the HP 9883A Tape Reader's interface responds to select code 3.

```
0: fxd 0
1: prt rdb(3)+A
2: if A#10;jmp -
  1
3: end
```

Notice that the read binary function is programmed as part of the print statement. Line 2 continues the program until a LF is seen. Compare the list of decimal numbers with the ASCII characters listed in the Appendix.

49
50
51
52
44
51
52
53
44
51
46
52
53
10

As an interesting exercise, execute this line ▶

```
wrt 16,rdb (0);jmp 0
```

Now press each key, except , to print its decimal equivalent code. Notice that Live Keyboard is disabled while it is involved in an I/O operation. To halt the calculator, press .

The decimal key codes returned by using this method generally do not correspond to ASCII decimal codes. A complete set of decimal key codes is in the Appendix.

The Read Status Function

```
rds (select code)
```

This function reads the current status information transmitted from the specified device and returns a decimal-equivalent number. The number of status bits and their meanings are described in each interface installation and service manual. The status information available from the HP 98032A Interface Card is described later in this section.

KDP Status Bits

Status information from the calculator's keyboard (K), display (D), and printer (P) is combined into an 8-bit byte. Although most of the bits are for internal use only, two bits have programming uses:

7	6	5	4	3	2	1	0
—	Internal Use Only			1	Printer Busy	Out of Paper	0

- Bit 0 - Is always 0.
- Bit 1 - Is 1 whenever the printer is out of paper.
- Bit 2 - Is 1 whenever the printer is busy.
- Bit 3 - Is always 1 (except when  is pressed).

So, if paper is loaded in the calculator, store and run this line (use either select code 0 or 16) ♦

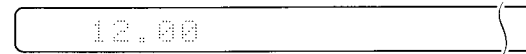
```
0: dsp rds(16);
   wait 100;jmp 0
```

The display should flash continually ♦



Bit 3 will always appear as "1" (binary 8) to the read status function.

Now press **PRT ALL**. Now the display is ♦
since the printer is busy in the print all mode.
Press **PRT ALL** again to switch the mode off. Notice
that status bit 2 (decimal 4) is added to the
byte whenever the printer is busy.




The segment on the right shows how to avoid error 15, out of paper, by watching for the addition of status bit 1 (decimal 2). When a status byte of 10 is returned, the rest of line 0 is executed.

```
1: if rds(16)=10
   ;dsp "out of
   paper+++";stp
2: prt A$
```

The above method works, provided that the printer is not busy. The lines on the right check for both status bytes which indicate "out of paper".

```
3: if rds(16)=10
   ;jmp 3
4: if rds(16)=14
   ;jmp 2
5: jmp 2
6: dsp "Out of
   Paper+++";stp
7: prt A$
```

Fortunately, there's an easy way to isolate one bit of the status byte; this expression can be used to isolate bit 1: $\text{int}(n/2) \bmod 2$. The number returned is either 0 or 1, reflecting the state of bit 1.

Here's a program segment that's equivalent to the one shown above ♦

```
0: int(rds(16)/
   2)mod2→A
1: if A=1;dsp
   "Out of Paper++
   +";stp
2: prt A$
```

Tape-Drive Status Bits

The status bits available with the internal tape drive are shown below.

7	6	5	4	3	2	1	0
Protected Tape	Tape Direction	Tape Motion	Inter-file Gap	0	Cartridge Out	Tape Drive Failure	Internal Use Only

- Bit 0 - Is for internal use only.
- Bit 1 - Is 1 when a tape drive (hardware) failure occurs. Check the tape and then repeat the tape drive operation to verify the failure.
- Bit 2 - Is 1 when a cartridge is not loaded.
- Bit 3 - Is always 0.
- Bit 4 - Is 1 whenever the tape is positioned at an inter-file gap (between files). Bit 4 is usually 1, except when the tape is moving.
- Bit 5 - Is 1 whenever the tape is moving.
- Bit 6 - Is 0 to indicate forward tape movement and 1 to indicate reverse tape movement.
- Bit 7 - Is 1 whenever the tape is protected (RECORD slide).

Tape Drive Status Bits

Here is a program sequence that checks for a protected tape before recording data. If bit 7 is 1 (decimal 128), "4---Protected Tape!" is displayed.

```

7:  if rds(1)<128
   ;jmp 2
8:  fmt b,"--Prot
   ected Tape!";
   wrt 0,0;stp
9:  rcf 5,A[L,P]

```

Interface Status

The 98032A Interface has nine status bits which can be monitored using read status:

8	7	6	5	4	3	2	1	0
Peripheral Status	Interrupt Status	DMA Status	Interface ID 1 0		Invert Input Data	Invert Output Data	Extended Status	

- Bits 0, 1 and 8 - Indicate states of optional peripheral status-input lines.
- Bits 2 and 3 - Indicate states of logic levels preset on the 98032A.
- Bits 4 and 5 - Are preset to 0 and 1, respectively, on the 98032A.
- Bits 6 and 7 - Indicate operating states on the interface card. DMA (direct memory access) and interrupt functions are usable only with an Extended I/O ROM.

98032A Interface Status Bits

Refer to the 98032A manual for complete details on status meanings and status-input lines in use.

The 98032A Option 066 Interface Card uses bit 8 to indicate when the HP 9866B Printer is out of paper. The Peripheral Status bit remains at 1 (decimal 256) except when the printer is out of paper. Here is a program segment that checks for a 1 at bit 8 before sending data to the printer.

```

4:  if rds(6)<257
    ;jmp 2
5:  dsp "9866B
    Out of Paper";
    stp
6:  wrt 6,A$,B

```

The Extended I/O ROM permits using additional parameters in the read status function, enabling other bytes of status information to be read from some interface cards. Refer to the Extended I/O Programming Manual and the interface installation and service manual.

The Write Control Statement

`wtc select code: expression`

This statement outputs a binary number to control functions on most interface cards. One number is allowed with each statement. The available bits and their meanings for the 98032A Interface are shown below.

7	6	5	4	3	2	1	0
Enable Interrupt*	Enable DMA*	Preset	Enable Auto Handshake*	N/A	N/A	CTL 1	CTL 0

*Usable with Extended I/O ROM only.

98032A Interface Control Bits

Control bits 0, 1, and 5 are used to drive interface output lines CTL0, CTL1, and Preset. CTL0 and CTL1 are optional peripheral control lines, while Preset is used to initialize the peripheral to its power-up state. A preset signal is automatically given when the calculator is switched on or when **RESET** is pressed. The interface ignores bits 2 and 3. Bits 4, 6, and 7 are usable only when the Extended I/O ROM is in use. See the interface manual and Extended I/O Programming Manual for more details.

Alternate Character Set

The 9825A calculator has an option which provides an alternate character set for the display and printer. The alternate character set is Katakana which is used in Japan.

If the Katakana option is installed, it can be selected by executing `wtc 0,1`. The normal character set is reselected by pressing **RESET** or executing `wtc 0,0`.

Without the optional alternate character set installed, `wtc 0,1` will disable the printer and the display. This should not be done when the calculator is in the print all mode, because the heating elements that produce the dots on the thermal paper may be left in the ON state, and they could be destroyed within seconds.

CAUTION

IF THE PRINTER IS ACTIVELY PRINTING, `wtc 0,1` SHOULD NOT BE EXECUTED BECAUSE IT COULD CAUSE THE PRINTER TO BURN OUT.

Executing `wait 500` before executing `wtc 0,1` will ensure that a print operation is complete before the printer and display are disabled; thus, avoiding the possibility of destroying the printer's heating elements.

Chapter 4

HP Interface Bus

This chapter describes how to control and exchange data with instruments via the HP Interface Bus (HP-IB). Also included here is a brief description of the HP-IB and the 98034A Interface Card. For more information on the bus card, refer to its installation and service manual.

The General I/O ROM operations described in Chapters 2 and 3 are used to control each instrument via the HP-IB. You should be familiar with those operations before continuing in this chapter.

Overview of the HP-IB

The HP Interface Bus has a serial-byte bus structure which permits bi-directional communication between many instruments. When a controller such as a calculator is used, up to 14 additional HP-IB compatible devices can be controlled via one interface card.

Instruments can be controlled or programmed and data can be transmitted between devices on the bus. This is possible since each instrument connected to the bus has the potential of being a **talker** (send data) or a **listener** (receive data). Each instrument has a unique talk and/or listen address by which a **controller** communicates with the instrument. A unique handshake technique allows the communication to take place at a speed determined only by the specific instruments being addressed. Slower devices will not slow down the communication speed of the bus when they are not addressed.

In addition to the talker, listener, and controller functions, one device can be assigned the role of **system controller** on the bus. This instrument communicates with every other instrument and can halt and reset all bus operation at any time. The calculator is normally set to be the system controller; the function is enabled on the 98034A Interface Card.

Bus Messages

A common set of bus messages are used to transmit all data and control instructions between instruments. The bus messages used with the General I/O ROM are listed briefly here.

HP-IB Messages

Message Name	Description
Data	Transfers data and control information between instruments. To input data, use read statements or the read binary function. To output data, use write or write binary statements.
Remote	Sets all instruments for remote operation, enabling control via the bus. The Remote message is sent whenever the calculator is switched on or  is pressed.
Abort	Halts all bus operation and returns control to the calculator. Press  to send this message.

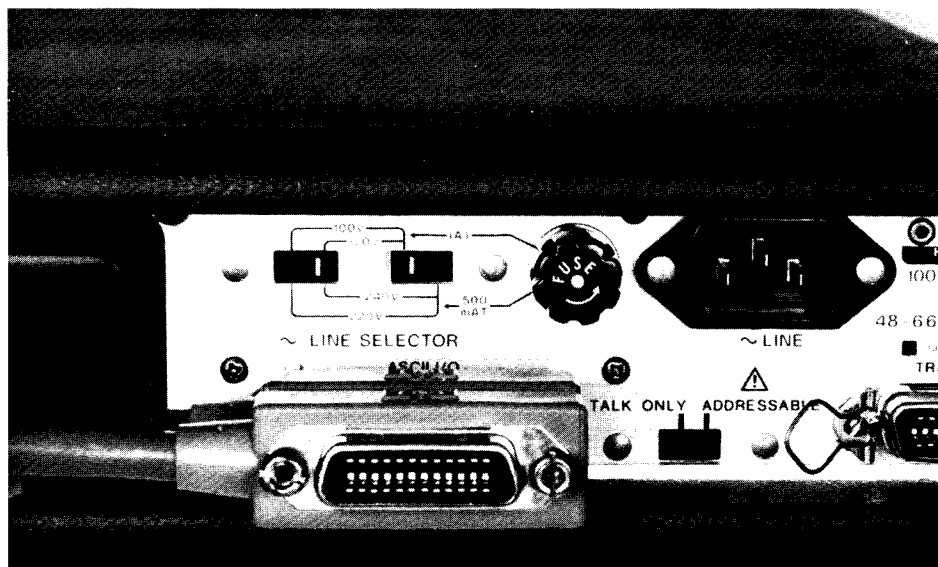
The Data message usually contains a list of talker and listener addresses and is followed by a string of characters representing data or other instructions to the assigned listener(s). The General I/O ROM, however, automatically generates the Data message to communicate between the calculator and the instrument specified by an address parameter (described later). So the General I/O ROM permits control of one instrument at a time on the bus.

When an Extended I/O ROM is in use, many other bus messages are available, enabling complete control of bus functions; refer to the Extended I/O Programming Manual for details.

The HP-IB Card

The HP 98034A Interface provides HP-IB capability for the HP 9825A Calculator. The bus card buffers all data and control messages between the calculator and instruments on the bus. The interface is preset to respond to select code 7.

Each instrument on the bus is connected through a 24-wire cable with a piggy-back connector on each end. Cables are available in 1, 2, and 4 meter lengths. Total cable length for a system can be up to 20 meters. The next photo shows two bus cables connected to an HP 3490A Multimeter.



Connecting Bus Cables

HP-IB Addresses

The General I/O ROM provides simplified control of instruments via the HP-IB by using a select-code parameter containing a three- or four-digit integer. The first one or two digits specify the bus card select code, while the last two digits represent the address of the instrument on the bus.

Instruments having HP-IB capability are assigned unique 7-bit ASCII characters for talker and listener addresses.¹ A controlling instrument, such as the calculator, uses the address characters to indicate which instrument is to talk (send data) or listen (receive data). For example, here are the addresses usually assigned to some instruments:

Instrument	HP-IB Address	
	Talker	Listener
98034A Interface	U	5
3490A Multimeter	V	6
9871A (Opt. 001) Printer	A	!
59309A Digital Clock	P	0

¹Bus addresses for most devices can be changed, if needed; see the manual furnished with each instrument for details.

Now, using the ASCII table in the Appendix, convert the five least-significant bits of each character's binary form to a decimal value:

Address		
Instrument	Character	5-bit Value
98034A Interface	U	21
	5	21
3490A Multimeter	V	22
	6	22
9871A Printer	!	01
	A	01
59309A Clock	P	16
	0	16

Notice that the 5-bit value for talker-listener instruments is the same number.

These numbers are used as the HP-IB address code in select-code parameters of General I/O operations. The HP-IB address code must always contain two digits; if the 5-bit value is a one-digit number (e.g., 9), a leading zero must be used (e.g., 09).

As examples, these write statements are addressed to a 9871A Printer via a bus card set to select code 7 ♦

98034A select code
 ↓
 printer address
 0: wrt 701,A,B,
 C
 1: fmt 5,2c30
 2: wrt 701.5,A\$,
 0\$
 ↑
 format number

This addressing method permits using all appropriate General I/O operations via the HP-IB. Typical example programs and sequences are shown on the following pages.

Controlling Listeners and Talkers

Instruments designated as listeners on the bus are controlled by using write and write binary statements. The address-code parameter just described must be used in each I/O operation.

Here is a short program which prints a listing of r-variables using a 9871A (Opt. 001) Printer. The printer bus address is set to "!" (address code 01). A sample printout and analysis of the program follow.

```

0: 0→A→X;10→B;
   20→C
1: fmt 5/,30x,
   "Data Register
   Listing",/!wrt
   701
2: fmt 3"ΔΔΔΔΔΔΔ
   Reg. ΔΔΔΔΔ ConTen
   ts ΔΔΔΔ",/!wrt
   701
3: fmt f.0,e18.8
   ,f.0,e18.8,f.0,
   e18.8
4: wrt 6,"
   r",A,rA,"
   r",B,rB,"
   r",C,rC
5: A+1→A;B+1→B;
   C+1→C;X+1→X
6: if X=5!0→X;
   wrt 701
7: if A<=9!jmp -
   3
8: fmt 5!/wrt
   701
9: end

```

Data Register Listing

Reg.	Contents	Reg.	Contents	Reg.	Contents
r0	0.00000000E 00	r10	3.44827586E-01	r20	5.00000000E-01
r1	0.00000000E 00	r11	5.00000000E-01	r21	4.76190476E-01
r2	0.00000000E 00	r12	8.33333333E-01	r22	4.54545455E-01
r3	3.33333333E 00	r13	7.69230769E-01	r23	4.34782609E-01
r4	2.50000000E 00	r14	7.14285714E-01	r24	4.16666667E-01
r5	2.00000000E 00	r15	6.66666667E-01	r25	4.00000000E-01
r6	1.66666667E 00	r16	6.25000000E-01	r26	3.84615385E-01
r7	1.42857143E 00	r17	5.88235294E-01	r27	3.70370370E-01
r8	1.25000000E 00	r18	5.55555556E-01	r28	3.57142857E-01
r9	1.11111111E 00	r19	5.26315789E-01	r29	3.44827586E-01

Here is a brief analysis of the program:

- Line 1 – Causes the printer to advance paper five lines and print the program title. Notice that the title is centered on the paper by sending a block of spaces before the title.
- Line 2 – Causes the column headings to be printed. Here, spacing is achieved by including spaces (ΔΔ) within the text format specs.

- Lines 3 and 4 – Specify the form of each table entry and which numbers (variables) are printed in each line of the table. The magnitude of each width (w) parameter in the format statement (line 3) was selected to position the corresponding table column under the appropriate heading.
- Line 5 – Increments program counters.
- Line 6 – Causes the printer to space one line after each five lines printed.
- Line 7 – Determines the maximum number of table entries.
- Line 8 – Causes the printer to space five lines.

As another example, the HP 3490A Multimeter can be connected via the bus and controlled by using strings of ASCII characters. Here is a list of control characters:

HP 3490A Control Characters

Char.	Function	Char.	Function
R	Range Program Identifier	T	Trigger Source Program Identifier
1	10,000 k Ω ; Test 7	0	Internal Sample Rate
2	1,000 k Ω ; 1000 V; Test 6	1	Immediate Internal
3	100 k Ω ; 100 V; Test 5	2	Next External Trigger
4	10 k Ω ; 10 V; Test 4	3	None
5	1 k Ω ; 1 V; Test 3	M	Mode of Operation Program Identifier
6	.1 k Ω ; 1 V; Test 2	0	Addressed Multi with No Output
7	Autorange; Test 1	1	Addressed Multi with Output
F	Function Program Identifier	2	Addressed Single with No Output
0	DC Volts	3	Addressed Single with Output
1	K Ohms	4	Interrupt Multi with No Output
2	AC Volts	5	Interrupt Multi with Output
3	Test	6	Interrupt Single with No Output
S	Sample/Hold Program Identifier	7	Interrupt Single with Output
0	Sample/Hold Off	E	Execute Mode of Operation Program
1	Sample/Hold Off		
2	Track/Hold		
3	Acquire/Hold		

The line on the right shows one method of setting 3490A range, function, etc., and then initiating a data sample. A read statement could be used to input the resulting data.

DC Volts Function
10V Range

```
0: wrt 722, "F0R4
  TIM3"
```

Internal Trigger
Output Mode

Here is a line which sets an HP 3330A Synthesizer to an output frequency of 100.0 Hz. Its listen address is "\$" (decimal code 04).

Frequency
Value
1: wrt 704, "L100
0="
Hertz

Sometimes it may be necessary to program a device using a variable stored in the calculator. Perhaps the variable contains the desired output frequency or amplitude for a signal source. This example outputs the same characters as line 1 above. The data spec `f.1` suppresses leading spaces, which the 3330A cannot accept.

2: fmt "L",f.1,
"="
3: wrt 704,A A = 100.0

Data Input

For most applications, read statements and the free-field format are used to input data via the HP-IB. Many devices output leading non-numeric characters in data messages; format statements must be referenced to read those characters. Chapter 2 describes the data and edit specs usable with format statements.

Examples best illustrate the technique for receiving data via the bus. For example, the HP 3490A Multimeter may send a data message of this form when programmed for DC volts and the 10V range (Δ indicates a space):

N Δ D C + 0 8 3 4 6 2 E - 4 $\text{\textcircled{CR}}$ $\text{\textcircled{LF}}$

By using the sequence on the right, the calculator skips the four leading non-numeric characters and reads the numeric portion of the string.

5: fmt 4x,f
6: red 722.8,A

If the String ROM is in use, either of these sequences will input the same number shown above. The first sequence has the advantage of retaining all characters in the input string; in either case the leading characters can be checked later in the program. See the String Variables Programming Manual for more information.

7: dim A#[20]
8: fmt c16
9: red 722,A#
10: val(A#[5]) $\rightarrow$ A

7: dim A#[4]
8: fmt c4,f
9: red A#,A

The **b** data spec can be used to input non-numeric characters. For example, suppose that this sequence is used to read the 3490A data string shown on page 41. Afterwards, variables A through E will contain ♦

```
0: fmt 4b,f
1: red 722,A,B,
  C,D,E
  A = 78 (N)
  B = 32 (Δ)
  C = 68 (D)
  D = 67 (C)
  E = 8.3462
```

Now consider this more-complex data message transmitted from an HP 3570A Network Analyzer:

- 034.48, - 088.93 **CR** **LF**

This sequence can be used to input and print the readings. The 3570A talker address is "A" (indicated by decimal code 01).

```
6: fmt ;fxd 2
7: red 701,A,P
8: prt "Amplitud
  e=",A,"Phase=",
  P
```

Here is a printout ♦

```
Amplitude=-34.48
Phase=      -88.93
```

As a final example, here is a program which inputs, stores, and computes the average of 50 data samples taken using a 3490A Multimeter.

```
0: dim A$(50,
  201;0→A;1→I→N
1: wrt 722,"FOR6
  TIM3"
2: red 722,A$(I)
3: wrt 722
4: if (I+1→I)<51
  ;jmp -2
5: A+val(A$(N,
  51)→A
6: if (N+1→N)<51
  ;jmp -1
7: prt A/50
8: end
```

- Line 0 – Sets up a 50-element string array and initializes three variables.
- Line 1 – Programs 3490A remote controls and instructs it to take a data sample.
- Line 2 – Inputs the data sample into a string array element.
- Line 3 – Instructs the 3490A to take another data sample.
- Line 4 – Exits the data input loop when 50 samples are stored.
- Lines 5 - 7 – Compute and print an average value from the data.

Interface Status Bits

As described in Chapter 3, the read status function is used to return status information from a specified interface card. These 98034A Interface status bits can be read using the General I/O ROM:

7	6	5	4	3	2	1	0
Service Request	Controller Active	Talker Active	Listener Active	System Controller Set	1	Serial Poll Set	EOI

- Bit 0: Is 1 when the EOI (end of data) line has been set true. The bit is cleared via an rds function.
- Bit 1: Is 1 when the Serial Poll function is set.¹
- Bit 2: Is always 1.
- Bit 3: Is 1 when the System Controller switch is enabled on the 98034A Interface Card.
- Bit 4: Is 1 when the calculator is an active listener.
- Bit 5: Is 1 when the calculator is the active talker.
- Bit 6: Is 1 when the calculator is an active controller.
- Bit 7: Is 1 when an instrument has sent a Require Service message.¹

98034A Interface Status Bits

These interface status bits should not be confused with the bus messages Status Byte and Status Bit¹, which transfers status information on the bus. For examples using the read status function, refer to Chapter 3.

Although the calculator is normally the System Controller in an HP-IB system, the function can be disabled via a switch on the 98034A Interface Card. Bit 3 of the interface status byte is logical 1 when the function is set and 0 when the function is disabled.

When the calculator is not the System Controller, it must be addressed (from the System Controller) to talk, listen, or take active control; the bus message Pass Control is used to transfer control between devices. When the calculator is not in active control, it must monitor interface status bits 4 through 7 to determine when it has been addressed to talk, listen, or take active control.

¹These functions and messages are enabled by using an Extended I/O ROM. See the Extended I/O Programming Manual for details.

When the read status function indicates that the calculator has been addressed to talk (bit 5 is logical 1), it can branch to an appropriate write-type operation to send a Data (output) message. When the calculator is addressed to listen (bit 4 is logical 1), it can execute an appropriate read-type operation to accept a Data (input) message. After executing each I/O operation, the calculator should then return to monitor interface status until it's addressed to talk or listen again.

When the calculator is addressed to talk or listen, the write or read operation must specify device address code 31 (e.g., `wrt 731;...` or `red 731;...`). If any other address code is specified when the calculator is not in active control of the bus, an error message will indicate that the calculator cannot address the bus when it is not the active controller.

When the calculator is addressed to take active control (bit 6 is logical 1), it can perform any appropriate bus operations until the System Controller sends the Abort message. When the Extended I/O ROM is installed, the calculator can pass control back to the System Controller or to another appropriate device by using the pass control (`pct`) statement.

A more complete discussion of calculator operation when it is not the active controller is in the Extended I/O Programming Manual.

Appendix

Binary Coding and Conversions

Binary is a base 2 number system using only 1s and 0s. By giving the 1s and 0s positional value, any decimal number can be represented. For example, this diagram shows how decimal 41 = binary 101001:

Decimal		Binary						
10^1	10^0	2^5	2^4	2^3	2^2	2^1	2^0	
↓	↓	↓	↓	↓	↓	↓	↓	
10	1	32	16	8	4	2	1	
4	1	1	0	1	0	0	1	

Binary-Decimal Conversions

To convert from binary to decimal, the positional values for the 1s are added up. From the above example this would be:

$$2^5 + 2^3 + 2^0 = 32 + 8 + 1 = 41$$

To convert from decimal to binary, the decimal number is repeatedly divided by 2. The remainder is the binary equivalent. For example:

		Remainder (read up)
$2 \overline{)41}$	→	1
$2 \overline{)20}$	→	0
$2 \overline{)10}$	→	0
$2 \overline{)5}$	→	1
$2 \overline{)2}$	→	0
$2 \overline{)1}$	→	1

ASCII

Binary is often used as a code to represent not only numbers, but also alphanumeric characters such as "A" or ",", or "?" or "x" or "2". One of the most common binary codes used is ASCII. ASCII is an eight-bit code, containing seven data bits and one parity bit. The General I/O ROM uses ASCII for most I/O operations. No parity bit is used. For example:

Character	ASCII Binary Code	ASCII Decimal Code
A	01000001	65
B	01000010	66
?	00111111	63

A complete list of ASCII characters and their equivalent binary and decimal representations is on page 48.

Binary Coded Decimal

Another often-used code for representing numeric values is Binary Coded Decimal (BCD). BCD is a four-bit binary code; each four bits represents a decimal digit from 0 through 9.

For example, to convert the BCD 12-bit word 010000110110 to decimal:

BCD representation	0100	0011	0110
	↓	↓	↓
Decimal Value	4	3	6

The decimal equivalent number is found by breaking up the word into 4-bit bytes (starting from the right) and converting the bytes into decimal.

Octal-Binary Conversions

Octal is a base 8 number system. Octal numbers are often used since conversion from binary to octal and vice-versa is easy by using electronic circuits.

To convert from binary to octal, the octal number is broken up into groups of three bits (starting from the right). The groupings of 3 bits represent an octal number.

For example, to convert binary 10110100011001 to octal:

Binary Number	10	110	100	011	001
	↓	↓	↓	↓	↓
Octal Number	2	6	4	3	1

Notice that only values from 0 through 7 are used in octal.

To convert from octal to binary, the process is reversed:

Octal Number	1	4	0	7	2	6
	↓	↓	↓	↓	↓	↓
Binary Number	001	100	000	111	010	110

Select Codes

The recommended select-code assignments for a 9825A Calculator system are listed below. The HP calculator peripherals and interfaces listed are preset at the factory to the indicated select code.

Factory Set Select Codes

Select Code	Assignment	HP Peripheral Device
0	Calculator Keyboard and Display	_____
1	Calculator Tape Drive	_____
2	Paper Tape Punch	9884A*,98032A Interface
3	Paper Tape Reader	9883A*,9863A*,98033A Interface
4	Digitizer	9864A*
5	Plotter	9862A*
6	Printer	9866B*,9871A*
7	HP Interface Bus	98034A Interface
8	Mass Memory	9885A*
9 through 15	Unassigned	Special or Duplicate Peripherals
16	Calculator Printer	_____

*These peripherals should be ordered with Option 025.

ASCII Character Codes

ASCII Char.	EQUIVALENT FORMS			ASCII Char.	EQUIVALENT FORMS			ASCII Char.	EQUIVALENT FORMS			ASCII Char.	EQUIVALENT FORMS		
	Binary	Octal	Dec		Binary	Octal	Dec		Binary	Octal	Dec		Binary	Octal	Dec
NULL	00000000	000	0	space	00100000	040	32	@	01000000	100	64	`	01100000	140	96
SOH	00000001	001	1	!	00100001	041	33	A	01000001	101	65	a	01100001	141	97
STX	00000010	002	2	"	00100010	042	34	B	01000010	102	66	b	01100010	142	98
ETX	00000011	003	3	#	00100011	043	35	C	01000011	103	67	c	01100011	143	99
EOT	00000100	004	4	\$	00100100	044	36	D	01000100	104	68	d	01100100	144	100
ENQ	00000101	005	5	%	00100101	045	37	E	01000101	105	69	e	01100101	145	101
ACK	00000110	006	6	&	00100110	046	38	F	01000110	106	70	f	01100110	146	102
BELL	00000111	007	7	'	00100111	047	39	G	01000111	107	71	g	01100111	147	103
BS	00001000	010	8	(	00101000	050	40	H	01001000	110	72	h	01101000	150	104
HT	00001001	011	9	)	00101001	051	41	I	01001001	111	73	i	01101001	151	105
LF	00001010	012	10	*	00101010	052	42	J	01001010	112	74	j	01101010	152	106
V _{TAB}	00001011	013	11	+	00101011	053	43	K	01001011	113	75	k	01101011	153	107
FF	00001100	014	12	,	00101100	054	44	L	01001100	114	76	l	01101100	154	108
CR	00001101	015	13	-	00101101	055	45	M	01001101	115	77	m	01101101	155	109
SO	00001110	016	14	.	00101110	056	46	N	01001110	116	78	n	01101110	156	110
SI	00001111	017	15	/	00101111	057	47	O	01001111	117	79	o	01101111	157	111
DLE	00010000	020	16	Ø	00110000	060	48	P	01010000	120	80	p	01110000	160	112
DC ₁	00010001	021	17	1	00110001	061	49	Q	01010001	121	81	q	01110001	161	113
DC ₂	00010010	022	18	2	00110010	062	50	R	01010010	122	82	r	01110010	162	114
DC ₃	00010011	023	19	3	00110011	063	51	S	01010011	123	83	s	01110011	163	115
DC ₄	00010100	024	20	4	00110100	064	52	T	01010100	124	84	t	01110100	164	116
NAK	00010101	025	21	5	00110101	065	53	U	01010101	125	85	u	01110101	165	117
SYNC	00010110	026	22	6	00110110	066	54	V	01010110	126	86	v	01110110	166	118
ETB	00010111	027	23	7	00110111	067	55	W	01010111	127	87	w	01110111	167	119
CAN	00011000	030	24	8	00111000	070	56	X	01011000	130	88	x	01111000	170	120
EM	00011001	031	25	9	00111001	071	57	Y	01011001	131	89	y	01111001	171	121
SUB	00011010	032	26	:	00111010	072	58	Z	01011010	132	90	z	01111010	172	122
ESC	00011011	033	27	;	00111011	073	59	[	01011011	133	91	{	01111011	173	123
FS	00011100	034	28	<	00111100	074	60	\	01011100	134	92		01111100	174	124
GS	00011101	035	29	=	00111101	075	61	]	01011101	135	93	}	01111101	175	125
RS	00011110	036	30	>	00111110	076	62	^	01011110	136	94	~	01111110	176	126
US	00011111	037	31	?	00111111	077	63	_	01011111	137	95	DEL	01111111	177	127

Decimal Key Codes

KEY	CODE		KEY	CODE		KEY	CODE	
	UNSHIFT	SHIFT		UNSHIFT	SHIFT		UNSHIFT	SHIFT
A	97	225	0	48	176	f10	75	203
B	98	226	1	123	251	f11	76	204
C	99	227	2	78	206	PRT ALL	19	147
D	100	228	3	79	207	REWIND	2	130
E	101	229	4	80	208	STEP	24	152
F	102	230	5	81	209	←	16	144
G	103	231	6	82	210	→	17	145
H	104	232	7	83	211	DELETE	9	137
I	105	233	8	84	212	INSERT	8	136
J	106	234	9	85	213	RECALL	11	139
K	107	235	·	86	214	FETCH	28	156
L	108	236	·	87	215	ERASE	29	157
M	109	237	=	88	216	LOAD	31	159
N	110	238	→	89	217	RECORD	30	158
O	111	239	CLEAR	61	189	LIST	27	155
P	112	240	ENTER EXP	125	253	←	14	142
Q	113	241	(	18	146	→	15	143
R	114	242	)	96	224	BACK	20	148
S	115	243	/	40	168	FWD	21	149
T	116	244	*	41	169	DELETE	23	151
U	117	245	-	47	175	INS-RPL	22	150
V	118	246	+	42	170	RUN	12	140
W	119	247	RESULT	45	173	STORE	13	141
X	120	248	f0	43	171	STOP	1	129
Y	121	249	f1	7	135	CONTINUE	25	153
Z	122	250	f2	65	193	⏏	10	138
!	49	177	f3	66	194	;	59	187
"	50	178	f4	67	195	↵	94	222
#	51	179	f5	68	196	<	44	172
\$	52	180	f6	69	197	>	46	174
%	53	181	f7	70	198	:	63	191
&	54	182	f8	71	199	?		
@	55	183	f9	72	200	space bar	32	160
	56	184		73	201			
	57	185		74	202			

HEWLETT PACKARD

SALES & SERVICE OFFICES

AFRICA, ASIA, AUSTRALIA

ANGOLA

Electra
Empresa Técnica de
Equipamentos
Eléctricos, S.A.R.L.
R. Barbosa Rodrigues, 42-10DT.
Caixa Postal, 6487
Luanda
Tel: 355156
Cable: ELECTRA Luanda

AUSTRALIA

Hewlett-Packard Australia
Pty. Ltd.
31-41 Joseph Street
Blackburn, Victoria 3130
P.O. Box 36
Doncaster East, Victoria 3109
Tel: 896351
Telex: 31-024
Cable: HEWPARD Melbourne

Hewlett-Packard Australia

Pty. Ltd.

31 Bridge Street

Pymble

New South Wales, 2073

Tel: 4496566

Telex: 21561

Cable: HEWPARD Sydney

Hewlett-Packard Australia

Pty. Ltd.

153 Greenhill Road

Perth, W.A., 5063

Tel: 2725911

Telex: 82536

Cable: HEWPARD Adelaide

Hewlett-Packard Australia

Pty. Ltd.

141 Spring Highway

Needlands, W.A., 5009

Tel: 3865455

Telex: 33859

Cable: HEWPARD Perth

Hewlett-Packard Australia

Pty. Ltd.

121 Wollongong Street

Fyshwick, A.C.T., 2809

Tel: 804244

Telex: 62650

Cable: HEWPARD Canberra

Hewlett-Packard Australia

Pty. Ltd.

5th Floor

Teachers Union Building

495-499 Boundary Street

Spring Hill, Queensland 4000

Tel: 2291544

Cable: HEWPARD Brisbane

BANGLADESH

The General Electric Co.

of Bangladesh Ltd.

Magnet House 72

Dilkusha Commercial Area

Motijheel, Dacca 2

Tel: 252415, 252419

Telex: 734

Cable: GECDAC Dacca

ETHIOPIA

Abdella Abdulmalik

P.O. Box 2635

Addis Ababa

Tel: 11 93 40

GUAM

Medical Only

Guam Medical Supply, Inc.

Suite C, Airport Plaza

P.O. Box 8547

Termining 96911

Tel: 646-4513

Cable: EARMED Guam

HONG KONG

Schmidt & Co. (Hong Kong) Ltd.

Wing On Centre, 28th Floor

Connaught Road, C.

Hong Kong

Tel: 545564

Cable: HEWPARD HX

Tel: 74766 SCHMC HX

INDIA

Blue Star Ltd.

Kasturba Building

Jamshedpur Rd

Bombay 400 020

Tel: 21561

Cable: HEWPARD Sydney

Hewlett-Packard Australia

Pty. Ltd.

153 Greenhill Road

Perth, W.A., 5063

Tel: 2725911

Telex: 82536

Cable: HEWPARD Adelaide

Hewlett-Packard Australia

Pty. Ltd.

141 Spring Highway

Needlands, W.A., 5009

Tel: 3865455

Telex: 33859

Cable: HEWPARD Perth

Hewlett-Packard Australia

Pty. Ltd.

121 Wollongong Street

Fyshwick, A.C.T., 2809

Tel: 804244

Telex: 62650

Cable: HEWPARD Canberra

Hewlett-Packard Australia

Pty. Ltd.

5th Floor

Teachers Union Building

495-499 Boundary Street

Spring Hill, Queensland 4000

Tel: 2291544

Cable: HEWPARD Brisbane

BANGLADESH

The General Electric Co.

of Bangladesh Ltd.

Magnet House 72

Dilkusha Commercial Area

Motijheel, Dacca 2

Tel: 252415, 252419

Telex: 734

Cable: GECDAC Dacca

Blue Star Ltd.

Blue Star House

11/11A Magarath Road

Bangalore 560 025

Tel: 55668

Telex: 043-430

Cable: BLUESTAR

Blue Star Ltd.

Mesakshi Mandiram

xx/1678 Mahatma Gandhi Rd.

Cochin 682 016

Tel: 32069, 32161, 32282

Telex: 0885-514

Cable: BLUESTAR

Blue Star Ltd.

Saropini Devi Road

Secunderabad 500 003

Tel: 70126, 70127

Telex: 015-459

Cable: BLUEFROST

Blue Star Ltd.

Kasturba Building

Jamshedpur High Road

Madrassa 800 034

Tel: 82056

Telex: 041-379

Cable: BLUESTAR

INDONESIA

BERCA Indonesia P.T.

P.O. Box 4964

Jin. Abdul Muhs 62

Jakarta

Tel: 349255, 349886

Telex: 46748 BERSIL IA

Cable: BERSAL

BERCA Indonesia P.T.

P.O. Box 174, Sby.

23 Jin. Jimerto

Surabaya

Tel: 42027

Cable: Bercacoon

ISRAEL

Electronics Engineering Div.

of Motorola Israel Ltd.

15, Kremenetski Street

P.O. Box 25016

Tel: 38973

Telex: 33569, 34164

Cable: BASTEL Tel-Aviv

JAPAN

Yokogawa-Hewlett-Packard Ltd.

Chuo Bldg., 4th Floor

4-20, Nishinakajima 5-chome

Yokogawa-ku, Osaka-shi

Osaka 532

Tel: 06-304-6021

Cable: BLUESTAR

Blue Star Ltd.

Bhandari House

7th & 8th Floor

91 Nehru Place

New Delhi 110 024

Tel: 634770 & 635166

Telex: 031-2463

Cable: BLUESTAR

Yokogawa-Hewlett-Packard Ltd.

Nakamo Building

24 Kam Sasajima-cho

Nakamura-ku, Nagoya, 450

Tel: 052 571-5171

Yokogawa-Hewlett-Packard Ltd.

Tanigawa Building

2-24-1 Tsuruya-cho

Kanagawa-ku

Yokohama, 221

Tel: 045-312-1252

Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.

Mito Mitsui Building

105, 1-chome, San-no-maru

Mito, Ibaragi 310

Tel: 0292-25-7470

Yokogawa-Hewlett-Packard Ltd.

Inoue Building

1348-3, Asahi-cho, 1-chome

Atsugi, Kanagawa 243

Tel: 0462-24-0452

Yokogawa-Hewlett-Packard Ltd.

Kumagaya Asahi

Hachijuni Building

4th Floor

3-4, Tsukuba

Kumagaya, Saitama 360

Tel: 0485-24-6563

KENYA

Advanced Communications Ltd.

P.O. Box 30070

Nairobi

Tel: 22639

Medical Only

International Aeradio (E.A.) Ltd.

P.O. Box 19012

Nairobi Airport

Nairobi

Tel: 336055/56

Telex: 22201/22301

Cable: INTAERIO Nairobi

Medical Only

International Aeradio (E.A.) Ltd.

P.O. Box 95221

Mombasa

KOREA

Samsung Electronics Co., Ltd.

15th Floor, Daeyongak Bldg.,

25-5, 1-KA

Choong Moo-Ro, Chung-Ku,

Seoul

Tel: 231 6811, 778-3401/2,3,4

Telex: 22575

MALAYSIA

Hewlett-Packard Sales SDN BHD

Suite 2, 21/22

Bangunan Angkasa Raya

Jalan Ampang

Kuala Lumpur

Tel: 23320/27491

Protel Engineering

P.O. Box 197

Lot 259, Satok Road

Kuching, Sarawak

Tel: 53544

Cable: PROTELENG

MOZAMBIQUE

A.N. Goncalves, Ltd.

162, 1st Apt. 14 Av. D. Luis

Caixa Postal 107

Maputo

Tel: 27091, 27114

Telex: 6-203 NEGON Mo

Cable: NEGON

NEW GUINEA

Hewlett-Packard Australia

Pty. Ltd.

Development Bank Building

Ground Floor

Wood Strip

Port Moresby, Papua

New Guinea

Tel: 256933

NEW ZEALAND

Hewlett-Packard (N.Z.) Ltd.

4-12 Cruickshank Street

Kilbirnie, Wellington 3

P.O. Box 9443

Courtney Place

Wellington

AUSTRIA
Hewlett-Packard Ges.m.b.H.
Handelskai 52
P.O. Box 7
A-1205 Vienna
Tel: 351621-27
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

BAHRAIN
Medical Only
Wadi Pharmacy
P.O. Box 648
Bahrain
Tel: 54886, 56123
Telex: 6550 WAEI GJ
Cable: WAEI-PHARM
Analytical Only
Al Hamidiya Trading
and Contracting
P.O. Box 20074
Manama
Tel: 259378, 259958
Telex: 8895 KALDIA GJ

BELGIUM
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: (02) 660 50 50
Cable: PALOBER Brussels
Telex: 23-494 palober bru

CYPRUS
Kypros
Irgonios Xenopoulos Street
P.O. Box 1152
Nicosia
Tel: 45628/29
Cable: Kypros Pandeis
Telex: 3018

CZECHOSLOVAKIA
Vývojová a Provozní Závodka
Výzkumných Ústavů v Bechovicích
ČSSR-25997 Bechovice u Prahy
Tel: 89 93 41
Telex: 12133
Institute of Medical Bionics
Výzkumný Ústav Lekáři Bioniky
Jedlova 6
CS-88346
Bratislava-Kramare
Tel: 4251
Telex: 93229

DDR
Entwicklungslabor der TU Dresden
Forschungsinstitut Meinsberg
DDR-7305
Waldheim/Meinsberg
Tel: 37 667
Telex: 518741
Export Contact AG Zurich
Guenther Forger
Schlegelstrasse 15
1040 Berlin
Tel: 42-74-12
Telex: 111889

DENMARK
Hewlett-Packard A/S
Davang 52
DK-3460 Birkerød
Tel: (02) 81 66 40
Cable: HEWPAK AS
Telex: 37409 hpaas dk
Hewlett-Packard A/S
Navejvej 1
DK-8600 Silkeborg
Tel: (06) 82 71 66
Telex: 37409 hpaas dk
Cable: HEWPAK AS

EGYPT
I.E.A.
International Engineering Associates
24 Hussein Hegazi Street
Kasr-el-Aini
Cairo
Tel: 23 829
Telex: 93830
Cable: INTENGASSO
SAMITRO
Sami Amin Trading Office
18 Abdel Aziz Gawish
Abdo-Can
Tel: 24937
Cable: SAMITRO CAIRO

ALABAMA
8130 P.O. Box 4207
San Diego Whitesburg Dr.
Huntsville 35802
Tel: (205) 881-4591
8933 E. Roebuck Blvd.
Birmingham 35206
Tel: (205) 836-2203/2

ARIZONA
2336 E. Magnolia St.
Phoenix 85034
Tel: (602) 244-1361
2424 East Aragon Rd.
Tucson 85705
Tel: (602) 889-4661

ARKANSAS
Medical Service Only
P.O. Box 5646
Bridgman Station
Little Rock 72215
Tel: (501) 376-1844

CALIFORNIA
1579 W. Shaw Ave.
Fresno 93711
Tel: (209) 244-0582
1430 East Orangeflorio Ave.
Fullerton 92631
Tel: (714) 870-1000
3939 Lantershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2671
5400 West Rosencrans Blvd.
P.O. Box 92105
Worldway Postal Center
Los Angeles 90009
Tel: (213) 776-7500
TWX: 910-325-6608

LOUISIANA
3013 Scott Boulevard
Santa Clara 95050
Tel: (408) 988-7000
"Ridgcrest"
Tel: (414) 446-6165
646 W. North Market Blvd
Sacramento 95834
Tel: (916) 929-7222

ALABAMA
9606 Aero Drive
P.O. Box 23333
San Diego 92123
Tel: (714) 279-3200
"Tarzana"
Tel: (213) 705-3344
COLORADO
5600 DTC Parkway
Englewood 80110
Tel: (303) 771-3455
CONNECTICUT
12 Lunar Drive
New Haven 06525
Tel: (203) 389-6551
Tel: (203) 746-0529
FLORIDA
P.O. Box 24210
2727 N.W. 62nd Street
Ft. Lauderdale 33309
Tel: (305) 973-2600
4428 Emerson Street
Unit 103
Jacksonville 32207
Tel: (904) 725-6333
P.O. Box 13910
6177 Lake Ellenor Dr.
Orlando 32809
Tel: (305) 859-2900
P.O. Box 12826
Suite 5, Bldg. 1
Office Park North
Pensacola 32575
Tel: (904) 476-8422
GEORGIA
P.O. Box 105005
450 Interstate North Parkway
Atlanta 30348
Tel: (404) 955-1500
Medical Service Only
"Augusta 30903"
Tel: (404) 736-0592
P.O. Box 2103
1172 N. Davis Drive
Warner Robins 31098
Tel: (912) 922-0449
HAWAII
2875 So. King Street
Honolulu 96826
Tel: (808) 955-4455

FINLAND
Hewlett-Packard OY
Nahkonosuntti 5
P.O. Box 7
SF-00211 Helsinki 21
Tel: (09) 6923031

FRANCE
Hewlett-Packard France
Les Us Us
Boite Postale No. 6
91401 Orsay-Cedex
Tel: (1) 907 78 25
TWX: 600048
Hewlett-Packard France
Chemin des Moulins
Eschestrass 5
69130 Ecullly
Tel: (78) 33 61 25,
TWX: 3106177
Hewlett-Packard France
Périmètre de la Cépide
31081 Toulouse-Le Mirail
Tel: (81) 40 11 12
TWX: 5109577

FRANCE
Le Lignères
Bureau de Vente de Marseille
Plus Route de Villeneuve
13100 Aix-en-Provence
Tel: (42) 58 41 02
Hewlett-Packard France
2, Allée de la Bourgnette
35100 Rennes
Tel: (99) 51 42 44
TWX: 740912F
Hewlett-Packard France
18, rue du Canal de la Marne
67300 Schiltigheim
Tel: (88) 83 08 10
TWX: 890141F
Hewlett-Packard France
Immeuble péricentre
Rue van Gogh
59650 Villeneuve d'Ascq
Tel: (20) 91 41 25
TWX: 160124F
Hewlett-Packard France
Bureau de Vente
Centre d'affaires Paris-Nord
Bâtiment Ampère
Rue de la Commune de Paris
B.P. 300
93153 Le Blanc Mesnil Cedex
Tel: (01) 931 88 50
Hewlett-Packard France
Av. du Pdt. Kennedy
33700 Mérignac
Tel: (66) 97 22 69
Hewlett-Packard France
"France-Evry" Immeuble Lorraine
Boulevard de France
91335 Evry-Cedex
Tel: 077 96 60
Hewlett-Packard France
60, Rue de Metz
57130 Jouy aux Arches
Tel: 871 69 45 32

GERMAN FEDERAL REPUBLIC
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Bernier Strasse 117
Postfach 560 140
D-6000 Frankfurt 56
Tel: (0611) 50-04-1
Cable: HEWPAKCSA Frankfurt
Telex: 04 13249 hpfm d
Hewlett-Packard GmbH
Technisches Büro Böttingen
Herrenberger Strasse 110
D-7030 Böttingen, Württemberg
Tel: (0714) 67-1
Cable: HEWPAK Böttingen
Telex: 0726539 bbn
Hewlett-Packard GmbH
Technisches Büro Düsseldorf
Emanuel-Leube-Str. 1 (Seestern)
Volk-4000 Düsseldorf
Tel: (0211) 59711
Telex: 085,86 533 hppd d
Hewlett-Packard GmbH
Technisches Büro Hamburg
Wendenstrasse 23
D-2000 Hamburg 1
Tel: (040) 24 13 93

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

IRELAND
Hewlett-Packard Ltd.
King Street Lane
Winnereah, Wokingham
Berkshire, RG11 5AR
GB-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewpie London
Hewlett-Packard Ltd.
20 Avonham Industrial Estate
Long Mile Road
Dublin 12, Eire
Tel: (01) 514322
Telex: 30439
Hewlett-Packard Ltd.
Cardiac Services (Ireland) Ltd.
Kilmore Road
Dublin 5, Eire
Tel: (01) 315820
Medical Only
Cardiac Services Co.
55A Frinagh Rd. South
Dublin 12, Eire
GB-Northern Ireland

ITALY
Hewlett-Packard Italiana S.p.A.
Via G. Di Vittorio, 9
20063 Carmusio
Sul Naviglio (MI)
Tel: (2) 903691
Telex: 311046 HEWPAKIT
Hewlett-Packard Italiana S.p.A.
Via Turazza, 14
35100 Padova
Tel: (49) 664888
Telex: 41612 HEWPAKIT
Hewlett-Packard Italiana S.p.A.
Via G. Armellini 10
1-00143 Roma
Tel: (06) 54 69 61
Telex: 61514
Cable: HEWPAKIT Roma
Hewlett-Packard Italiana S.p.A.
Corso Giovanni Lanza 94
I-10133 Torino
Tel: (011) 6822-655908
Medical-Calculators Only
Hewlett-Packard Italiana S.p.A.
Via Principe Nicola 43 G.C.
I-95126 Catania
Tel: (095) 37 05 04
Hewlett-Packard Italiana S.p.A.
Via Nuova San Rocco A.
Capodimonte, 62A
I-80131 Napoli
Tel: (081) 7913544
Hewlett-Packard Italiana S.p.A.
Via E. Masi, 96
I-40137 Bologna
Tel: (051) 70787/300040

JORDAN
Mouassat Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

MOROCCO
Moroccan Cousins Co.
P.O. Box 1387
Amman
Tel: 24907/39907
Telex: SABCO JO 1455
Cable: MOUTASHERCO
KUWAIT
Al-Khaleel Trading & Contracting
P.O. Box 830-Safat
Kuwait
Tel: 42 9104/1 1726
LUXEMBURG
Hewlett-Packard Benelux
S.A. N.V.
Avenue du Col-Vert, 1
(Groenkraslaan)
B-1170 Brussels
Tel: 67 22 40
Cable: PALOBER Brussels
Telex: 23 494

NETHERLANDS
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 12
P.O. Box 667
NL-Amstelveen 1134
Tel: (020) 47 20 21

NORWAY
Hewlett-Packard Norge A/S
Osloveien 15
P.O. Box 34
1345 Osteras
Tel: (02) 1711 80
Telex: 16621 hpnas n
Hewlett-Packard Norge A/S
Nylandsveien 114
5000 Bergen

NETHERLANDS
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 12
P.O. Box 667
NL-Amstelveen 1134
Tel: (020) 47 20 21

NORWAY
Hewlett-Pack

General I/O Syntax

Syntax Conventions

Brackets [] - Items within brackets are optional.

Dot Matrix - Items in `dot matrix` must appear as shown.

Expression - A constant (like 16.4), a variable (like X or B[8] or r3), or an expression (like $8 \uparrow 4$ or $6 < A + B$).

Select code format - `cc[dd[ee]][.f]`

`cc` = device or interface select code.

`dd` = optional HP-IB address code (must be two digits).

`ee` = optional HP-IB extended address code (Extended I/O ROM only)

`.f` = format number, for read and write only.

Text - A series of characters within quotation marks.

Variable - A simple variable (like A or Q), an array variable (like E[5]), an r-variable (like r12), or a string name (like A\$).

... - Dots indicate that successive parameters are allowed. Be sure that a comma precedes each successive parameter.

Conversion Statement

```
conv[code1,code2[,code3,code4][,...]]
```

Sets up a character conversion table for read and write statements. Ten pairs of ASCII-decimal codes are allowed. See page 25.

Format Statement

```
fnt[format no.][spec1[,spec2,...]]
```

Lists data and edit specifications for read and write statements. All numeric parameters must be integer constants. See pages 10 thru 24.

List Statement

```
list[#select code]
```

Output program listings to external devices. See page 25.

Read Statement

`red select code [ , format no.] , variable1 [ , variable2...]`

Input data and string variables, using either free-field or format specs. See page 7.

Read Binary Function

`rdb (select code)`

Input a single 16-bit character and return a decimal number in range: -32768 to 32767. See page 28.

Read Status Function

`rds (select code)`

Input one byte of status information and return a decimal equivalent code. See page 29.

Write Statement

`wrt select code [ , format no.][ , expression or text1 [ , expression or text2...]]`

Output data using free-field or format specs. See page 5.

Write Binary Statement

`wtb select code , expression or text1 [ , expression or text2...]`

Output single 16-bit characters. See page 27.

Write Control Statement

`wtc select code , expression`

Output a single binary number to control functions or lines on an interface card. See page 33.

Subject Index

a

Addressing:
 HP-IB 37
 Select Code 4,47
 Alternate Character Set 33
 ASCII Bus (HP-IB) 35
 ASCII Code 4
 ASCII Table 48

b

Binary I/O Operations 27
 Binary Codes and Conversions 45
 Bus Interface (HP-IB) 35

c

Calculator I/O Scheme 3
 Calculator ROM Cards 3
 Control Bits (98032A) 33
 Conversion Statement 25
 Cursors, Editing 19

d

Data Input Operations 7,28,29
 Data I/O Format 5
 Data Output Operations 5,25,27
 Data Specifications:
 Input Specs 21
 Output Specs 11
 Default (free-field) Output Format 6
 Delimiters:
 Read Statements 8
 Write Statements 5
 Digital Voltmeter Examples 19,24,40
 Display Character Set 19

e

Edit Specifications:
 Input Specs 23
 Output Specs 13
 Error Messages (inside-back cover)
 Extended I/O ROM
 (references) 5,32,33,36,43

f

Free-Field Formats:
 Input Free-Field 8
 Output Free-Field 6
 Format Statements:
 Input Specs 21,23
 Output Specs 11,13
 Syntax 10
 Formatted I/O Operations 3

g

General I/O Operations (table) ii

h

HP Interface Bus (HP-IB) 35

i

Interface Cards 3
 Internal Peripherals 3
 Internal Printer:
 Character Set 16
 Examples 12-14,16-18,27
 Interrupt, Peripheral 5
 I/O Bus 3
 I/O Format 5
 I/O Scheme 3
 Installation Procedure 1

k

Key Codes, Decimal 49

l

Leading Spaces, Suppressing 11
 Leading Zeros 11
 List Statement 25

m

Memory Usage (ROM Card)	2
Messages, HP-IB	36

p

Peripheral Interrupt	5
Peripheral Status Check	29
Printers:	
HP 9866A	6,7,15,20,21,26,32
HP 9871A	38,39
Printer (internal) Status	29
Program Listings	25

r

Read Binary Function	28
Read Statement	7
Read Status Function	29
ROM Block Memory Usage	2

S

Select Code:	
General Syntax	4
List of Settings	47
Single Character Output	19
Status Bits:	
HP-IB Interface	43
KDP	29
Interface (98032A)	32
Tape Drive	31
Status Check Function	29
String ROM (references)	5,7,8,11,24,41
Suppressing Leading Spaces	11
Syntax:	
Guidelines	52
Listing	52-53

t

Tape Drive (internal)	31
Tape Reader Examples	9,22-24,28
Text Character Fields	13
Text in Format and Write	20

W

Write Binary Statement	27
Write Control Statement	33
Write Statement	5

General I/O Error Messages

error G1

Incorrect Format Numbers:

- Format number in format statement not in range of $0 \leq n \leq 9$.
- Referenced format number not executed.

error G2

Referenced Format Statement has an error:

- Incorrect format spec.
- Numeric overflow in format statement.

error G3

Incorrect I/O Parameters:

- Parameter not a number or a string.
- Negative parameter with `f z` numeric spec.
- Numeric parameter with `c` edit spec.
- Binary parameter not in range of $-32768 \leq n \leq 32767$.
- More than one parameter for read binary or read status function.
- Missing or non-numeric parameter for write control statement.

error G4

Incorrect Select Code:

- Select code is non-numeric or greater than 4 digits.
- Select code is greater than 16 for read status.
- Select code is not in range from 0 through 16.
- Select code 1 allowed only for read status.
- HP-IB address code not in range from 0 through 31.
- Read from select code 0 not allowed.

error G5

Incorrect Read Parameter:

- Constant in read list.
- String not filled by read operation.
- Numeric parameter references `c` format spec.

error G6 Incorrect Parameter in Conversion Statement:

- More than 20 parameters.
- Odd number of parameters.
- Non-numeric parameter.
- Parameter not in range of $0 \leq n \leq 127$.

error G7 Unacceptable Input Data:

- More than one decimal point or "E" read.
- 511 characters read without a LF.
- "E" with no leading digit.
- More than 158 numeric characters read.

error G8 Peripheral Device Down:

- Incorrect status bits – device not ready or power is off.
-  cancelled operation.

error G9 Interface Hardware Problem:

- Improper HP-IB operation.
- Empty I/O slot.
- Select code does not match interface card (e.g., wrt 711 when a 98032A is set to 7, or wrt 6 when 98034A is set to 6).
- Write Control addressed to a 98034A HP-IB Card.

Scan Copyright ©
The Museum of HP Calculators
www.hpmuseum.org

Original content used with permission.

Thank you for supporting the Museum of HP
Calculators by purchasing this Scan!

Please to not make copies of this scan or
make it available on file sharing services.