

## HP 9826 HPL Quick Reference



Part No. 09826-90045  
Microfiche No. 09826-99045

Printed in U.S.A.  
First Edition June 1981



## NOTICE

---

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another program language without the prior written consent of Hewlett-Packard Company.

# HP 9826 HPL Quick Reference

Manual Part No. 09826-90045  
Microfiche No. 09826-99045

Hewlett-Packard Desktop Computer Division  
3404 East Harmony Road, Fort Collins, Colorado 80525  
Copyright by Hewlett-Packard Company 1981

# Index

<b>HPL Syntax</b> .....	1
Introduction .....	1
Operators .....	2
Math Hierarchy.....	3
MSUS Table .....	3
<b>Alphabetical Syntax Listing</b> .....	6
<b>Error Codes</b> .....	40
<b>Option ROM Syntax</b> .....	54
<b>Interface Card Registers</b> .....	56
<b>ASCII Table</b> .....	64
<b>ASCII Control Code</b> .....	66

# HPL Syntax

## Introduction

The following pages are a compilation of all current HPL syntax. More information on each operation can be found by referring to the indicated manual and page. The manual titles are abbreviated here:

D Disc Programming (09825-90220).  
I/O I/O Control Reference.  
M Matrix Programming.  
O&P Operating & Programming Reference.

The HPL programming language utilizes four basic types of syntax constructions: **statements**, **functions**, **operators** and **commands**. Operators, such as + and mod, are used with numbers and variable names to construct **expressions** (like A + 5). Expressions can be included in many statements and executed from the keyboard. Each statement can also be preceded by a line number and stored as a program line (like 10: prt A). Most functions can include expressions, and can be executed from the keyboard. Functions can also be treated as expressions when constructing a statement (like prt sinA). Commands are operator aids that can only be executed from the keyboard; they're not programmable.

- New 9826 HPL statements and extensions are denoted with a black dot to the side.

## Operators

The available operators are summarized here. For more details see page 3-19 in your Operating and Programming Reference.

Arithmetic	
+	Add
-	Subtract, unary -
*	Multiply
/	Divide
↑	Exponentiate
mod	Modulus
Logical	
and	inclusive OR
ior	exclusive OR
xor	
not	
Relational	
=	Equal to
→	Assign
>	Greater than
<	Less than
> = or = >	Greater than or equal to
< = or = <	Less than or equal to
# or < > or > <	Not equal to
String	
&	Concatenation

## Math Hierarchy

highest priority functions, flag references, r-variables

↑ (exponentiation)  
implied multiply  
- (unary minus)  
\*, /, mod  
+, -  
all relational operators (=, >, <, <, =, >, =, #, →)  
not  
and  
lowest priority or, xor

Operators of the same level in an expression are executed from left to right. Any operations within parentheses, however, are performed first. For more details, see page 3-18 in your Operating and Programming Reference.

## MSUS Table

The syntax of a mass storage unit specifier (msus) is as follows:

:[device format[controller select code]] [,unit code]

The msus specification can be added to the "file name" parameter of any disc programming statement except the files statement.

Code	Disc Drive	Format
I	Internal	LIF
M	8290x	LIF
F	9885	9825 - compatible
G	9885	LIF
H	9895	9825 - compatible
J	9895	LIF

## Syntax Conventions

These terms and conventions are used in the following listing:

**bold type** — all key words and characters appearing in bold type must appear exactly as shown. These items are shown in dot matrix in the referenced manuals.

[ ] — elements enclosed in brackets (not key characters or parentheses) are optional.

... — an ellipsis indicates that the preceding parameter or sequence in the syntax can be repeated.

**variable name** — a numeric or string variable name (like A or r5 or A\$). Subscripts are allowed (like A [7] ).

**array name** — an array variable name, with or without subscripts.

**string variable** — a string variable name (like A\$ or B\$ [1,4] ).

**string** — either a string variable or text within quotes ("text").

**line number** — an expression from 0 through 32767 referring to a program line.

**line label** — a unique name assigned to a program line. It's enclosed in quotes, follows the line number, and is followed by a colon. For example: 5: "print": ...

**expression** — a logical combination of numeric variable names, constants, operators and functions (including user-defined functions) grouped within parentheses as needed. The evaluated expression yields a numeric result.

**constant** — a fixed number within the computer's range, like 2.23467.

**character** — a letter, number or symbol.

**item** — a series of constants, expressions and/or strings separated by commas, for example:  
prt 5,A,"was",A+7.

**subscripts** — numbers within brackets which are attached to variable names to designate a particular variable element or boundary. For example: A [10,5] or B\$ [1,10]

**file number** — an expression indicating the tape or disc file.

**file name** — a string indicating the disc file name.

**select code** — an expression indicating the device's interface select code setting (an integer from 0 through 16). For example: wrt 6

These select codes are assigned to internal devices:

- 0 Keyboard/Display Line
- 7 Internal HP-IB
- 16 Printer/CRT

**device address** — a two-digit number appended to the select code, indicating a device's HP-IB address. Device address range is from 00 through 31. For example: wrt 711 outputs to device 11 via the HP-IB interface set to select code 7.

**format no.** — a number from .1 through .9 appended to the select code to reference a corresponding fmt statement. For example: wrt 7.3 references fmt 3.

**return variable** — a simple numeric variable name (A or R4) where information is stored after the operation.

**flag no.** — an expression from 0 through 15 (0 to 31 for the 9826) indicating a programmable flag.

# A

**abs** expression

Returns the absolute value of the expression. O&P, 3-22.

**• aclr** [number of pages]

Clears screen, allocates specified number of scrolling pages.

**acs** expression

Returns the principal value of the arccosine of the expression in the current angular units. O&P, 3-25.

**add** ( expression , expression )

Returns the sum of the expressions, added in the current numeric mode, decimal (mdec) or octal (moct). I/O, 3-15.

**• adump** [select code[,number of lines] ]

Dumps alpha screen to printer.

**• aoff**

Alpha off. Turns the alpha display off.

**• aon**

Alpha on. Turns the alpha display on.

**aprt** array variable [, array variable [,...] ]

Prints the specified array's elements on the system printer. M, 8.

**ara** array variable<sub>1</sub> [  $\begin{cases} + \\ * \\ / \end{cases}$  ] array variable<sub>2</sub> ]  $\rightarrow$  array variable<sub>3</sub>

Performs the arithmetic operation, element by element, on arrays 1 and 2. The result is stored in array 3. (Example: ara A+B→C). Arithmetic operations can be performed on arrays in place (ara A+B→A), arrays can be copied (ara A→B) and implied multiplication is allowed (ara AB→C). M, 11.

**asc** expression

Returns the ASCII equivalent of the specified keycode. O&P, 7-25.

**asgn** file name, file number[, drive number

[, return variable ] ]

Assigns a number (1 through 10) to an existing disc file name and indicates optional drive number and a return variable (values below). D, 3-5.

0 Typed data file available and assigned.

1 File doesn't exist.

2 Program file.

3 Special function key file.

4 String/mixed BDATA file available and assigned.

5 Memory file.

6 Binary program file.

7 Numeric BDATA file available and assigned.

8 File number out of range.

9 Data file, but logical records not 256 bytes long.

10 ASCII file available and assigned.

11 Other mainframe file.

12 NULL file available and assigned.

**asn** expression

Returns the principal value of the arcsine of the expression in the current angular units. O&P, 3-26.

**expression** → variable name<sub>1</sub>

[ → variable name<sub>2</sub> [ → ... ] ]

Assigns the value of the expression to the variable(s). O&P, 3-19.

**atn** expression

Returns the principal value of the arctangent of the expression in the current angular units. O&P, 3-26.

**avd**

Disables automatic tape verification. O&P, 5-24.

## HPL Syntax

### ave

Enables automatic tape verification (default setting). O&P, 5-25.

### avm

Returns the size (bytes) of unused read/write memory. O&P, 4-27.

### axe X coordinate , Y coordinate [ , X tic [ , Y tic ] ]

Draws axes through the X,Y point, drawing optional tic marks at X tic and Y tic intervals. (9825 only.) I/O, 7-18.

## B

### band ( expression , expression )

Returns the 16-bit result of ANDing the expressions. I/O, 3-12.

### beep

Sounds the computer's beeper. O&P, 3-16.

### bit ( expression<sub>1</sub>, expression<sub>2</sub> )

Returns the binary value of the bit position in expression 2 indicated by expression 1. I/O, 3-15.

### boot

Loads 98217A Disk ROM bootstraps from a disc tape to an initialized disc. D, 4-4. (9825 only.)

### bplt string, # bytes per line [ ,function ]

Plots binary data in string to the graphics screen. Function: 0 = OR, 1 = AND, 2 = EOR, 3 = STORE

### bred ( buffer name )

Returns the contents of the specified, active, interrupt buffer. O&P, 7-10.



### buf "name" [ ,buffer size or string variable ,

buffer type ]

Sets up and names a data buffer of either type read/write (no type specified) or the specified type (see below). I/O, 6-6.

Buffer Type	Word	Byte
interrupt	0	1
fast read/write	2	3
DMA	4	5

## C

### cap ( string )

Returns an equivalent string of uppercase characters. O&P, 6-24.

### cat [ select code or buffer name ]

Prints a catalog of the files on the default disc to the specified printer/buffer or to the system printer. Single letter file type mnemonics (9825 compatible discs) and multiple letter mnemonics (LIF discs) are shown below.

9825	LIF	Description
Z	NULL	null file
D	TDATA	typed data file
P	PROGRAM	program file
K	KEYS	special function keys file
S	SBDATA	string/mixed binary data file
M	MEMORY	memory file
B	BINARY	binary program file
N	NBDATA	numeric binary data file
O	OTHER	other mainframe file
—	ASCII	ASCII data file (LIF only)
—	SYSTEM	9826 system file (LIF only)

## HPL Syntax

**cfg** [ flag no. ] , ... ] ]

Clears either all program flags or only the specified flags. O&P, 3-29.

**chain** file name [ , 1st line number

[ , 2nd line number ] ]

Loads a program from the specified disc file. Same optional line numbers as get. D, 2-7.

**char** (expression [ ,expression,... ] )

Returns the ASCII equivalent character(s). O&P, 6-20.

**cli** select code

Sends the abort message to all devices on the HP-IB, I/O, 2-27.

**cll** 'name' [ (expression<sub>1</sub> [ , expression<sub>2</sub> ] , ... ] ) ]

Calls the subroutine having the specified label, passing the value of any optional expressions as pass-parameters. O&P, 4-10.

**cln**

Returns the current program line number. O&P, 7-28.

**clr** select code

Sends the clear message, either the all devices or to only a selected device by including the device address in the select code. I/O, 2-17.

**cmd** select code , “address parameters” [ , “string” ]

**cmd** “device name(s)” or select code [ , “string” ]

Sends the string of data characters to the specified HP-IB device. I/O, 2-31.

**cmf** [ flag no. , ... ] ]

Complements either all program flags or only the specified flags. O&P, 3-29.

**cmp** ( expression )

Returns the 16-bit binary one's complement of the expression. I/O, 3-13.

**cont** [ line number or line label ]

This command continues program execution, either from the current point or from the specified point. O&P, 2-24.

**conv** [ expression<sub>1</sub> , expression<sub>1</sub>

[ , expression<sub>2</sub> , expression<sub>2</sub> ] ... ]

Sets up a conversion table (up to 10 sets of expressions) referenced by red and wrt statements. Each expression represents an ASCII character. conv (no parameters) cancels any existing table. I/O, 1-23.

**copy** source file [ , drive number [ , select code ] ]

destination file [ , drive number [ , select code ] ]

Copies a file to another location. D, 4-7.

**copy** [ source drive number[ , select code ], ]

“to” [ , destination drive number [ , select code ] ]

• **copy** complete source msus,“to”, complete

destination msus

Duplicates the contents of the source disc onto the destination disc. D, 4-7.

**copy** source file number , record number ,

destination file number , record number , records

Copies only the specified number of records, beginning at the specified record numbers. D, 4-10.

**cos** ( expression )

Returns the cosine of the expression. O&P, 3-25.

**cplt** [ character-space widths ,

character-space heights ]

Moves the pen the specified distance away from the current point. I/O, 7-41.

• **cret**

Returns to the program from an on-cycle routine.

## HPL Syntax

**csiz** [ height [ , aspect ratio [, paper ratio

[, angle of rotation] ] ] ]

Specifies the size, shape and lettering direction for **lbl** statements. Defaults are: height = 1.5% of paper height; aspect ratio = 1; paper ratio = 1; angle = 0 (left to right lettering). I/O, 7-38.

**csv**

Clears simple variables A through Z. O&P, 3-39.

**ctbl** [ string variable ]

Sets up a conversion table; the value of each string character represents ASCII; the character position represents the foreign code + 1. **ctbl** with no parameters cancels the table. I/O, 4-6.

• **cycle**

Returns the number of on cycle interrupts that have occurred since the previous cycle function call.

## D

• **data** numeric or string constant

[ , numeric or string constant ] ...

Provides constants for read statement variables.

**deg**

Sets degrees units for angular calculations. O&P, 3-25.

**del** line number [ , 2nd line number [,\*] ]

This command deletes either the specified program line or all lines through the optional 2nd line number specified. Including the \* changes all remaining references to the deleted lines to the next remaining program line, preventing error 36. O&P, 2-25.

**dev** "name" , select code

Assigns a name for use in place of the select code in I/O operations. I/O 2-9.

**dig** X , Y [ , return variable ]

Reads, computes and stores the current pen position in user units. Return variables: 0 = pen up; 1 = pen down. External plotter only. I/O, 7-48.

**dim** variable name [ , variable name [ , ... ] ]

Reserves memory for specified variables. Use subscripts to indicate size of each variable. O&P, 3-37.

**dir**

Copies the spare disc directory (default drive) to the main directory. (9825 only.) D, 4-16.

• **dret**

Returns to the program from an on-delay routine.

**drive** unit no. [ , select code ]

Sets the default unit and, optionally, the select code for disc operations. D, 1-14.

**drnd** ( expression , expression )

Returns the value of the first expression, rounded to the number of digits indicated by the second expression. O&P, 3-22.

**dsp** item list

Displays the items listed. To display quotes use double quotes within the string (e.g., 1: **dsp** "Display" "test" "in quotes."). O&P, 3-12.

**dto** ( expression )

Returns the octal equivalent of the decimal value expressed. I/O, 3-12.

**dtrk** [ tape file number ]

Dumps a bad track during the 98217A error recovery routine. (9825 only.) D, 4-15.

## HPL Syntax

### **dtype**

Returns a code indicating the type of drive, disc and data format at the default disc address. See Disc Programming page 1-15. Return values are:

- 0 Unable to access default disc controller.
- 1 Drive door is open or drive not present.
- 2 Drive door closed, but door was opened since last disc operation. File pointers are cleared.
- 3 9895 drive, single-sided disc, HP format
- 4 9895 drive, double-sided disc, HP format.
- 5 9895 drive, single-sided disc, unknown format.
- 6 9895 drive, double-sided disc, unknown format.
- 7 9895 drive, single-sided disc, IBM 3740 format.
- 8 9885 drive, single-sided disc.
- 9 9826 internal drive or 8290X drive

### **dump** [ file name , tape file name ] [ , expression ]

Transfers the contents of the default disc to a tape cartridge. The optional file names indicate to only dump a specified file. The expression can be 1 or 10, indicating the number of disc records to put in each tape file. A positive expression automatically marks the tape. A negative expression suppresses marking the tape. (9825 only.) D, 4-12.

## E

### **edit** [ line number or key ]

Selects edit mode, edit screen format at selected line number, or SFK definition.

### **eir** select code [ , byte ]

Enables an interrupt from the specified select code. Specifying byte = 0 disables the interrupt. I/O, 5-6.

### **end**

Halts program execution and sets the program counter to 0. O&P, 3-17.

### **enp** [ "prompt" , ] variable name

Enters and prints data entered from the keyboard. O&P, 3-15.

### **ent** [ "prompt" , ] variable name

Enters data from the keyboard. O&P, 3-13.

### • **eol** code [ , [ , ... ] ] [ , – delay in milliseconds ]

Specifies up to seven optional ASCII characters for an end-of-line sequence for wrt operations (replaces CR/LFs). The optional delay occurs after the last eol character in the sequence. O&P, 7-12.

### **eor** ( expression , expression )

Returns the 16-bit binary result of the exclusive ORing of the expressions. I/O, 3-13.

### **equ** "name<sup>1</sup>" , "string<sup>1</sup>"

[ , "name<sup>2</sup>" , "string<sup>2</sup>" ] , ... ]

Equates the ASCII character string with the name, for use with cmd. I/O, 2-33.

### **erase** [ letter or key ]

Erases either all programs and variables or the specified areas listed below. O&P, 2-26.

a Erase entire memory.

k Erase all special function keys and sets to defaults.

v Erase all variables and flags.

fn Erase specified key definitions.

### **erl**

Returns the line number in which the error occurred.

### **ern**

Returns the error number for an on err routine.

## HPL Syntax

### ert file number

Erases the current tape track, beginning with the specified file. O&P, 5-15.

### exp ( expression )

Returns  $e$  (2.71828...) raised to the expressed power. O&P, 3-24.

## F

### fdf file number

Positions the tape at the specified file on the current track. O&P, 5-9.

### fetch [ line number or key ]

Displays the specified program line or special function key definition. O&P, 2-27. (9825 only.)

### files file name<sub>1</sub> [ : unit no. ] [ , file name<sub>2</sub>

[ : unit no. ] [ , ... ]

Assigns names up to 10 disc files. Substituting an \* for a file name allows deferring the assignment for that particular file number.

### flg ( flag no. )

Returns flag status: 1 = set; 0 = clear. O&P, 3-30.

### flt [expression]

Sets floating point notation; from 0 through 11 places allowed. O&P, 3-10.

### fmt [ format no. , ] [ spec<sub>1</sub> [ , spec<sub>2</sub> ... ] ]

Sets up a list of format specs for red and wrt operations. Format number can be from 0 through 9. Format specs are listed below. Omitting specs cancels specified format. Omitting format no. sets format 0. A repeat factor can precede each spec. I/O, 1-8.

**b** Single-character binary output.



**cw** String character data.

**ew.d** Exponential format.

**fw.d** Fixed-point.

**fzw.d** Fixed point with leading zeroes.

**x** Blank space.

**z** Suppresses auto CR/LF.

**/** Outputs CR/LF.

**"text"** Outputs text.

w = field width.

d = number of digits to right of decimal point.

**for** simple variable = initial value **to** value

[ **by** step value ]

Defines start of a for-next loop. O&P, 4-3.

**frc ( expression )**

Returns the fractional part of the expression. O&P, 3-22.

**fti ( expression )**

Rounds and changes the expression to integer precision. The result can be stored in a two-character field. O&P, 4-26.

**fts ( expression )**

Changes the expression to split precision for storage in a four-character field. O&P, 4-20.

**fxd [expression]**

Sets the fixed-point format; from 0 through 11 places are allowed. O&P, 3-9.

## G

### • gclr

Clears the graphics screen.

### • gdump [ select code[,number of lines ] ]

Dumps the graphics screen to a raster-scan standard printer.

## HPL Syntax

**get** file name [ , 1st line no. [ , 2nd line no. ] ]

Loads the program from the specified disc file at the 1st line number, and begins execution at the 2nd line number. D, 2-4.

**getb** file name

Loads the specified disc binary program file. D, 2-11.

**getk** file name

Loads the special function keys disc file. D, 2-9.

**getm** file name

Loads the specified disc memory file. D, 2-10. (9825 only.)

• **gload** string

Loads the graphics screen from the specified string.

• **goff**

Turns the graphics display off.

• **gon**

Turns the graphics display on.

• **gptr** xcoord, ycoord [ ,type ]

Draws a graphics cursor at the specified location. (type: 0 = off, #0 = on)

**grad**

Sets the grads units for angular calculations. O&P, 3-25.

**gsb** line number or line label

Branches program execution to the specified subroutine. O&P, 3-34.

**gsb** + or - no. of lines

Branches to the subroutine beginning the number of lines relative to the current line. O&P, 3-34.



• **gstore** string variable

Store graphics screen to the specified string.



**gto** line number or line label

Sends program execution to the specified line. O&P, 3-31.



**gto** + or - no. of lines

Sends execution to specified line relative to the current line. O&P, 3-31.



## I

**idf** file number [ , file type [ , current size

[ , absolute size [ , track ] ] ] ]

Returns info on the current tape file. See tlist for file types. O&P, 5-7.

**idn** array name [ , array name [ , ... ] ]

Creates identity (square) matrices. All elements are 0 except major diagonal elements which are 1. M, 22.

**if** expression<sub>1</sub> = expression<sub>2</sub>

If the equation is true, the rest of the line is executed. If false, execution immediately branches to the next line. Any relational operator can be used ( <, #, >=, etc.). When both expressions are strings, the characters are compared using ASCII values. O&P, 3-36.

**ina** array variable [ : value ]

[ , array variable [ : value ] ... ]

Initializes each element of the array to the specified value (number or variable). Omitting the value initializes each element to 0. M, 8.

**init** drive numbers select code [ , interleave factor ]

Initializes discs in either 9885 or 9895 drive. The interleave can be an integer from 1 thru 29. See page 4-3, Disc Programming.

## HPL Syntax

- **init** complete msus [ ,interleave factor

[ ,# directory records ] ]

Initializes the specified disc with the specified (or default) interleave factor and the specified or default number of directory records.

- int** ( expression )

Returns the integer value of the expression. O&P, 3-22.

- inv** array variable<sub>1</sub> → array variable<sub>2</sub>

[ , simple variable ]

Stores the inverse matrix of array 1 in array 2. If the simple variable is specified, the determinant of array 1 is returned. M, 24.

- iof** select code

Returns interface flag state: 0 if peripheral busy; 1 if ready. I/O, 4-12.

- ior** ( expression , expression )

Returns the 16-bit result of the inclusive OR operation on the expression. I/O, 3-13.

- ios** select code

Returns interface status: 0 if in error condition; 1 if operational. I/O, 4-12.

- iplt** X increment , Y increment [, expression]

Moves the pen the number of X and Y units from its current position. The expression is for pen control; see plt. I/O, 7-29.

- iret**

Ends an interrupt service routine and returns to main program. I/O, 5-7.

- itf** ( string variable )

Returns a full-precision number from the packed, integer-precision number (a two-character string). O&P, 7-26.

## J

- jmp** expression

Jumps program execution the relative number of lines forward (+ expression) or back (- expression). jmp 0 returns execution to the beginning of the current line. O&P, 3-33.

## K

- key**

Returns the earliest, unprocessed keycode in the keyboard buffer. 0 indicates no keycodes in the buffer. O&P, 7-8.

- kill** file name

Purges the specified disc file. D, 1-18.

- killall** drive number , select code

Purges all user files from the specified disc. See page 1-18, Disc Programming.

- **killall** complete msus

Purges all user files on the specified disc.

- kloff**

Turns special function key labels off.

- klon**

Turns special function key labels on.

- knob**

Returns the accumulated knob count. CCR rotation is negative valued, CR rotation is positive valued.

- **kret**

Returns execution to the main program after the key buffer is emptied or after knob count is zeroed. O&P, 7-9.

- **kstat**

Returns knob status: Bit 5 = Control, 4 = Shift.



## HPL Syntax

### llo select code

Sends the local lockout message to all HP-IB devices. I/O, 2-19.

### ln ( expression )

Returns the natural log ( $\log_e$ ) of the expression. O&P, 3-24.

### load [ disc file name , tape file number ]

Loads files previously dumped to a tape back onto the disc. Omitting all parameters loads the entire dump back onto the disc. Including parameters loads only selected data files back onto the disc. D, 4-13. (9825 only.)

### log ( expression )

Returns the common log ( $\log_{10}$ ) of the expression. O&P, 3-24.

### ltr X coordinate , Y coordinate [ , HWD ]

Moves the 9862 plotter pen to the specified point and specifies dimensions for lettering. H and W can be from 1 through 9. D is lettering direction and can be from 1 through 4. (9825 only.) I/O, 7-47.

### ltrk

Returns corrected data to a reinitialized track during disc error-recovery routine. (9825 only.) D, 4-15.

## M

### mat array variable<sub>1</sub> \* array variable<sub>2</sub> → array variable<sub>3</sub>

Array multiplication (arrays must have correct dimensions). M, 19.

### max ( expression [ , expression [ , ... ] ] )

Returns the largest value in the list. O&P, 3-22.

## mdec

Sets the decimal mode (default) for binary operations. I/O, 3-11.

### min ( expression [ , expression [ , ... ] ] )

Returns the smallest value in the list. O&P, 3-22.

### moct

Sets the octal mode for binary operations. I/O, 3-11.

### • mret

Returns to the program from an on-match routine.

### mrk number of files , file size [ ,return variable ]

Marks the number of files, beginning at the tape's current position. The last file number marked is returned in the optional return variable. O&P, 5-10.

### msi [ “:device format [ controller select code

[ ,unit number ]”]

Sets the current system disc drive and format. Drive formats: **I**=internal disc, **M**=8290x mini-floppy, **F**=9885(9825), **G**=9885(LIF), **H**=9895(9825), **J**=9895(LIF)

## N

### nal

Returns the last program line number plus one; used with store to store strings. O&P, 7-24.

### next simple variable

Terminates for-next loop and tests for loop completion. O&P, 4-3.

### nor [ line number [ , line number ] ]

Clears the master program flag, either while executing all lines (omit all parameters) or only for the specified line numbers. Clears debug flags of the specified program lines. O&P, 3-44.

## HPL Syntax

### **num ( “character” or substring )**

Returns the ASCII-decimal value of the character.  
O&P, 6-21.

## O

### **ofs X coordinate , Y coordinate**

Offsets the origin (0,0) to point X,Y. I/O, 7-27.

### **on end file number , line number or label**

Enables a branch to the specified line or label when a disc EOF or EOR mark is encountered during read and write operations. D, 3-19.

### **on cycle time [ ,line label ]**

Sets up clock periodic interrupt service routine. on cycle with no label specified cancels clock-cycle interrupt service. Time is in seconds.

### **• on delay time [ ,line label ]**

Sets up clock delay interrupt service routine. On delay with no label specified cancels clock-delay interrupt service. Time is in seconds.

### **on err [ line label ]**

Enables an error-trapping routine. The program branches to the label and the erl, ern and rom functions are assigned values when an error occurs. Executing on err with no line label cancels on-error trapping. I/O, 4-4.

### **on key [ line label [, flag no. ] ]**

Enables a keyboard interrupt routine. The program branches to the label and optionally sets the flag when the keyboard buffer overflows. Omitting all parameters disables the keyboard interrupt. O&P, 7-6.

### **• on knob [ line label ]**

Sets up Knob interrupt service routine. on knob with no label cancels Knob interrupt service.

### **on match time [ ,line label ]**

Sets up clock match interrupt service routine. on match with no label cancels clock-match interrupt service. Time is in seconds.

### **oni select code [ , line label ]**

References an interrupt service routine associated with the peripheral's select code. Executing oni with no label cancels interrupt service. I/O, 5-5.

### **open file name , number of records [ ,file type ]**

Creates a disc data file of the specified size. D, 3-2. Optional file types are “ASCII”, “NULL”, or “TDATA”.

### **otd ( expression )**

Returns the decimal equivalent of the octal value expressed. I/O, 3-12.

## P

### **par ( expression )**

Sets the parity type (listed below) used for I/O checking. I/O, 4-9.

0	Parity disabled.	2	Even parity.
1	Parity = 1	3	Odd parity.

### **• pbeep [ frequency [ ,duration ] ]**

Programmable beep with frequency (0-5167 Hz) and duration (0-2.56 seconds).

### **• pclr**

Sets default plotter values except scale units, select code, P1, P2, pen location and pen#. 9872 Plotter ROM only. I/O, 7-10.

### **pct select code**

Passes active control to the specified HP-IB device. I/O, 2-26.

### **pen**

Raises the plotter pen. I/O, 7-22.

## HPL Syntax

### pen# [ expression ]

Selects the plotter pen (external plotters). I/O, 7-22. Graphics pens are:

- > 0 Normal
- = 0 Off
- 1 Erase
- 2 EOR

### % string [ ; ]

The % free-text prefix allows storing text without syntax checking. Free text is terminated with a semicolon or end of line. O&P, 7-25.

- **pi**

Returns value of pi.

- **pkbd** [ string ]

Executes ASCII string (as if it were pushing keys).

### plt X coordinate , Y coordinate [ , expression ]

Move plotter pen to specified X,Y point. Optional expression controls pen (see below). I/O, 7-22.

even lowers pen.

odd raise pen.

positive action before plotting.

negative action after plotting.

### pol select code

Conducts a parallel poll on the HP-IB. I/O, 2-25.

### polc select code , byte

Sets parallel poll bits on the specified HP-IB device. I/O, 2-26.

### polu select code

Clears parallel poll bits on the specified device. I/O, 2-26.

### pos ( string<sub>1</sub> , string<sub>2</sub> )

Returns the character position of the second string within the first. O&P, 6-16.

### prnd ( expression , expression )

Returns the first expression rounded to the power of ten indicated by the second expression. O&P, 3-22.

### prt expression or string [ , expression or string [ , ... ] ]

Prints the list of items on the system printer. To print quotes, use double quotes within the string. (e.g., 3: prt "print""text""in quotes."). O&P, 3-12

- **prtsc** select code [ ,width ]

Sets system printer select code (and width).

### psc select code

Sets the select code for all plotter ROM operations. psc 0 causes the program to ignore all plotter operations. psc 16 selects CRT graphics. I/O, 7-5.

### ptyp

Sets a plotter lettering mode. Press STOP key to terminate mode. I/O, 7-45.

## R

### rad

Sets radians units for angular calculations. O&P, 3-25.

### ✓ ( expression )

Returns the square root of the expression. O&P, 3-22.

### rcb file number

Records the binary program in memory on the specified file.

## HPL Syntax

**rcf** [ file number [ , line number

[ , line number ] ] [ , "SE" or "DB" ] ]

Records either all program lines onto the specified tape file (no line numbers) or only the specified block of lines. Including SE prevents the program from being listed or displayed when reloaded. Including DB records all trace and stop flags with the program for debugging. O&P, 5-16.

**rcf** file number , variable list

Records the listed variables onto the tape file. O&P, 5-16.

**rck** file number

Records the special function key definitions on the tape file. O&P, 5-22.

**rcm** file number

Records the entire computer memory on the specified tape file. (9825 only) O&P, 5-22.

**rdb** ( select code )

Returns one 16-bit binary character code from the specified device. I/O, 3-4.

**rdi** ( register number )

Returns a status byte from the interface specified by wti 0. I/O, 4-12.

**rdm** array variable [ , array variable [ , ... ] ]

Redimensions the array(s) to the specified dimensions. M, 16.

**rds** ("name") [ , type [ , empty

[ , fill [ , dim ] ] ] ] ) → status

Extended buffer-status function.

**rds** ( select code )

Returns the current status word from the specified interface. I/O, 3-5.

**rds** (select code[ ,A[ ,B[ ,C ] ] ] )→D

Returns HP-IB extended read status. I/O, 2-34.

• **read** variable name [ ,variable name ]...

Reads data statement constants into variables.

**red** select code [ . format no. ] , variable list

Reads and stores data from the specified device. I/O, 1-5.

**rem** select code

Sends the remote message to either all HP-IB devices or only one device when its address is included in the select code. I/O, 2-18.

**renm** old file name , new file name

Renames a disc file. D, 28.

**repk**

Repacks user files on the default disc. D, 4-5.

**res**

Returns the result of the last keyboard operation not stored in a variable. O&P, 2-20.

**resave** file name [ , 1st line number

[ , 2nd line number ] ] [ , "SE" or "ND" ]

Stores a program (or only the specified lines) in an existing disc file. D, 2-9.

**ret**

Ends a subroutine and returns program execution to the main program (line after gsb). O&P, 3-34.

**rew**

Rewinds the tape. O&P, 5-6.

**rkbd** select code [ , expression ]

Enables a remote keyboard to control the computer. The expression indicates the keycode interpretation: 0 = ASCII (default) or 1 = hardware keycodes. O&P, 7-24.

## HPL Syntax

### **rnd ( expression )**

Returns a pseudo-random number from 0 to (less than) 1. A negative expression is used as a new seed. O&P, 3-22.

### **rom**

Returns the ROM in which the error occurred. 0 = mainframe error. Other ASCII-decimal numbers indicate letter of plug-in ROM.

### **rot ( expression<sub>1</sub> , expression<sub>2</sub> )**

Returns the result of binary rotation of the 16-bit equivalent of expression 1, rotated the number of bits indicated by expression 2. I/O, 3-13.

### **rprt file number , record number [ , data list ] [ , "end" or "ens" ]**

Prints the list of data items on the disc file, starting at the specified record. Including "end" prints an EOF mark after the data. Including "ens" suppresses the automatic EOR mark printed after data. D, 43.

### **rqs select code , byte**

Requests service from the HP-IB system controller and sends the serial status byte upon response to a serial poll. I/O, 2-21.

### **rread file number , record number [ , variable list ]**

Reads data from the disc file, starting at the specified record. D, 3-15.

### **rss ( select code )**

Returns the 98036 Interface status register byte. O&P, 7-16.

- rstr [ line label ]**

Resets data pointer either to line 0, or to "label" if specified.

- rtime**

Returns internal clock value in elapsed seconds.

### **run [ line number or line label ]**

Begins program execution, either at line 0 or at the specified line. O&P, 2-9.

## S

### **save file name [ , 1st line number**

**[ , 2nd line number ] ] [ , "SE" or "ND" ]**

Stores either the entire program on the disc file or only the specified block of lines. D, 2-2.

- saveb file name**

Saves the binary program in memory to the specified file.

### **savek file name**

Stores all special function key definitions on the disc file. D, 2-9.

### **savem file name**

Stores the entire read/write memory on the disc file. (9825 only) D, 2-10.

### **scl Xp1 , Xp2 , Yp1 , Yp2**

Locates the origin and specifies user units for plotting operations. I/O, 7-7.

### **sfg [ ,flag no. [ , flag no. [ ,... ] ] ]**

Sets either all program flags to 1 or only the specified flags. O&P, 3-28.

- sfk [ key number [ ,definition string [ ,label string ] ] ]**

Defines SFK (0 to 31) and optional soft label. No parameters sets all to defaults. Key number only erases sfk.

### **sgn ( expression )**

Returns sign of expression: 0 = zero; 1 = positive; -1 = negative. O&P, 3-22.

## HPL Syntax

**shf ( expression<sub>1</sub>, expression<sub>2</sub> )**

Returns the result of right-shifting the 16-bit binary equivalent of expression 1, the number of places indicated by expression 2. A negative expression 2 shifts the byte to the left. I/O, 3-14.

**sin ( expression )**

Returns the sine of the expression. O&P, 3-2.

**smpy** scalar number or simple variable

[\*] array variable<sub>1</sub> → array variable<sub>2</sub>  
Multiplies each element of array 1 by the scalar number. The \* can be omitted. M, 13.

**spc [ expression ]**

Outputs the expressed number of line feeds on the system printer. O&P, 3-16.

**sprt** file number , data list [ , “end” or “ens” ]

Prints the list of data items on the disc file. Including “end” prints an EOF mark after the data. Including “ens” suppresses the automatic EOR mark printed after data. D, 38.

• **sqr** expression

Returns the square root of the expression variable.

**sread** file number , variable list

Reads data from the disc file. D, 3-10.

**stf ( string )**

Unpacks and returns a split-precision number from its four-character string. O&P, 4-20.

• **stime** seconds

Sets the internal clock to the specified number of seconds.

**store** string [ ,line number ]

Stores program lines from an executing program. O&P, 7-21.

**stp [ line number<sub>1</sub>[ ,line number<sub>2</sub> ] ]**

Stops program execution either immediately or, optionally, at the specified line (line 1). Specifying both line numbers indicates a block of lines to stop at. O&P, 3-17.

• **str ( expression [ ,base ] )**

Returns the ASCII character equivalent to the expression. If the optional base parameter is specified, the numeric expression is converted to ASCII characters of the specified base (2-31). O&P, 6-19.

• **sysboot [ system name ]**

Boots language system specified from disc or ROM.

## T

• **tabxy** x col,y row

Moves print position to column x (0..49), row y (0...17).

**tan ( expression )**

Returns the tangent of the expression. O&P, 3-25.

**tfr** source name ,destination name

[ ,expression [ ,last character ] ]

Transfers data between an I/O buffer and a peripheral device. Optional expression indicates the total number of bytes to transfer. Optional last character expression is the decimal value of the character to terminate the transfer. I/O, 6-8.

**time ( expression )**

Causes an I/O operation to wait for a device to become ready for the specified number of milliseconds. I/O, 4-4.

**tinit**

Reinitializes a bad track during 9885 error recovery. (9825 only.) D, 4-15.

## HPL Syntax

### tlist

Catalogs tape files on the internal printer (file types below). O&P, 5-9.

- 1 Non-tape file type (OTHER)
- 0 Null file. (NULL)
- 1 Binary program. (BINARY)
- 2 Numeric data file. (NBDATA)
- 3 String or string/data. (SBDATA)
- 4 Memory file. (MEMORY, 9825 only)
- 5 Special function key file. (KEYS)
- 6 Program file. (PROGRM)
- 7 Track dump error recovery (9825 only)
- 8 Single file dump (9825 only)
- 9 Entire disc dump (9825 only)

### tn ↑ ( expression )

Returns 10 raised to the specified power. O&P, 3-24.

### trc [ 1st line number [, last line number ] ]

Sets the master flag and, optionally, trace flags for specified program lines. O&P, 3-44.

### trg select code

Sends the trigger message to the specified HP-IB device. I/O, 2-17.

### trk expression

Specifies the tape track (0 or 1) for successive operations. O&P, 5-6.

### trn array name → array name

Transposes rows and columns between arrays. M, 23.

### type ( [ - ] expression )

Returns the next item-type (types listed below) in a disc data file. A negative expression indicates a search for an EOR mark. D, 3-20.

- 0 Unidentified type.
- 1 Full-precision number.
- 2 String (within record).
- 3 EOF mark or physical end of file.
- 4 EOR mark.

Indicates string overlapping record boundaries:

- 2.1 Start of string.
- 2.2 Middle of string.
- 2.3 End of string.

## U

### units

Displays the currently-set angular units. O&P, 3-25.

## V

### • val ( string [ ,base ] )

Returns the numeric value of the string. If the base parameter is specified, the string is converted from the specified base (2-36). O&P, 6-17.

### vfy [ return variable ]

Verifies the contents of a tape file with the original in memory. Return variable: 0 = no error; 1 = error. O&P, 5-25.

### vfyb

Verifies disc bootstraps. (9825 only.) D, 4-15.

### voff

Disables disc data verification (default). D, 4-6.

### von

Enables disc data verification. D, 4-6.

## W

**wait** expression

The program waits for the specified time in milliseconds (from 1 to 32767). O&P, 3-16.

**wrt** select code [ . format no. ] [ , item list ]

Outputs the items to the specified device. I/O, 1-3.

**wscl** select code , expression

Outputs a control word (expression) to the specified interface. O&P, 7-14.

**wsm** select code , mode word [ , control word ]

Outputs a mode word and, optionally a control word (second expression) to the specified 98626A Interface. O&P, 7-15.

**wtb** select code , expression [, expression [ , ... ] ]

Outputs the byte representing each number or character to the specified device. I/O, 3-3.

• **wtc** buffer name [ , type[ , empty[ , fill] ] ]

Writes buffer pointers to specified buffer name.

• **wtc** HP-IB select code, value

Resets HP-IB interface. If "value" is <31, sets new bus address = "value". If "value" = 31, no further action. If "value" >31, "value" configures Parallel Poll response.

**wtc** select code , expression

Outputs a control byte to the specified interface. I/O, 3-9.

**wti 0, select code**

Specifies an interface for successive wti or rdi operations. I/O, 4-11.

**wti expression<sub>1</sub> , expression<sub>2</sub>**

Outputs a control byte (expression 2) to a specified interface register (expression 1). I/O, 4-11.

## X

**xax** Yoffset [ , tic interval [ , start [ , end

[ , no. of tics/label ] ] ] ]

Draws an X axis with optional tic marks and labels. I/O, 7-11.

**xref**

Prints a cross reference of program variables and line numbers, using the current program in memory. O&P, 4-32.

## Y

**yax** Xoffset [ ,tic interval [ , start [ , end

[ ,no. of tics/label ] ] ] ]

Draws a Y axis with optional tic marks and labels. I/O 7-11.

## Error Codes

An error in a program sets the program line counter to line 0. Press the continue key to continue the program from line 0. Execute the continue command with a line number to continue at any desired line (such as: cont 50).

- 00** System error.
- 01** Unexpected peripheral interrupt.
- 02<sup>1</sup>** Unterminated text.
- 03<sup>1</sup>** Mnemonic is unknown.  
Mnemonic not found because disc may be down. (9825 only)
- 04** System is secured.
- 05** Operation not allowed; line cannot be stored or executed with line number.
- 06<sup>1</sup>** Syntax error in number.
- 07<sup>1</sup>** Syntax error in input line.
- 08** Internal representation of the line is too long (gives cursor sometimes).
- 09** gto, gsb, or end statement not allowed in present context.  
Attempt to execute a next statement either from keyboard while for/next loop using same variable is executed in program or from program while for/next loop using same variable is executed from keyboard. Attempt to call function or subroutine from keyboard.
- 10<sup>1</sup>** gto or gsb statement requires an integer.
- 11** Integer out of range or integer required; must be from -32768 thru +32767.
- 12<sup>1</sup>** Line cannot be stored; can only be executed.
- 13** ent statement not allowed in present context.



- 14** Program structure destroyed.
- 15** Printer out of paper or printer failure.
- 16** String Variables ROM not present for the string comparison. Argument in relational comparison not allowed.
- 17** Parameter out of range.
- 18** Incorrect parameter.
- 19** Bad line number.
- 20** Missing ROM or binary program. The second number indicates the missing ROM. In the program mode, the line number is given instead of the ROM number. Displayed number and missing item:
  - 1 Binary Program
  - 4 Systems Programming ROM
  - 5 9826 HPL Extension
  - 6 Strings ROM
  - 8 Extended I/O ROM
  - 9 Advanced Programming ROM
  - 10 Matrix ROM
  - 11 Plotter ROM
  - 12 General I/O ROM
  - 17 Disk ROM
- 21** Line is too long to store.
- 22** Improper dimension specification.
- 23** Simple variable already allocated.
- 24** Array already dimensioned.
- 25** Dimensions of array disagree with number of subscripts.
- 26** Subscript of array element out of bounds.
- 27** P-number reference is negative.
- 28** Undefined array.
- 29** ret statement has no matching gsb statement.
- Cannot execute line because a ROM or binary program is missing.

<sup>1</sup> Press the RECALL key to position the cursor at the location of the error.

## Error Codes

30	Special function key not defined.	47	Read error in file head.
31	Non-existent program line.	48	End-Of-Tape (EOT) encountered before specified number of files were marked.
32	Improper data type. Non-numeric value in for statement or in fts or fti function.	49	File too small.
33	Data types do not match in an assignment statement.	50	ldf statement for a program file must be last statement in the line. get or chain statement should be the last statement in a line.
34	Display overflow due to pressing a special function key.	51 or 52	Memory configuration error for attempted ldm statement. For example, a ROM present when memory was recorded is now not present (see error 20), or attempting to load a memory file recorded on a 9825A into a 9825B.
35	Improper flag reference (no such flag).		Memory files are not compatible between the 9825A and 9825B. Only the program portion can be recovered by loading the memory file into the original machine and doing a rcf. This program file can then be loaded into any 9825 with the ldf statement.
36	Attempt to delete destination of a gto or gsb statement.	53	Negative parameter in cartridge statement.
37	Display buffer overflow caused by dsp statement.	54	Binary program to be loaded is larger than present binary program and variables have been allocated.
38	Insufficient memory for subroutine return pointer. Memory overflow during function or subroutine call.	55	Illegal or missing parameter in a cartridge statement.
39	Insufficient memory for variable allocation or binary program.	56	Data list is contiguous in memory for a cartridge statement.
40	Insufficient memory for operation. Memory overflow while using for statement or while allocating local p-numbers.	57	Improper file type.
41	No cartridge in tape transport.	58	Invalid parameter in rcf statement; "SE" or "DB" expected.
42	Tape cartridge is write protected. (Slide record tab to other position for recording.)	59	Attempt to record a program or special function keys which do not exist.
43	Unexpected Beginning-Of-Tape (BOT) or End-Of-Tape (EOT) marker encountered. Tape transport failure.	60	Attempt to load an empty file or the null file (type = 0).
44	Verify has failed.		
45	Attempted execution of idf statement without parameters or mrk statement when tape position is unknown.		
46	Read error in file body.		

## Error Codes

61	The line referenced in an ldf or ldp statement does not exist. If the line containing the ldf or ldp statement has been overlaid by the load operation, the line number in the display may be incorrect.	71	Asn or acs of number less than - 1 or greater than + 1.
62	Specified memory space is smaller than cartridge file size.	72	Negative base to non-integer power.
63	Cartridge load operation would overlay subroutine return address in program; load not executed.	73	Zero to the zero power ( $0 \uparrow 0$ ).
	Disk load operation would overlay gsb return address; load not executed.	74	Storage range overflow.
64	Attempt to execute ldk, ldf (program file), or ldp during live keyboard statement.	75	Storage range underflow.
	get, chain or getk not allowed from live keyboard mode or during an ent statement.	76	Calculation range overflow.
65	File not found.	77	Calculation range underflow.
	File specified in the previous fdf statement does not exist.	A0	Relational operator in for statement not allowed. No closing apostrophe.
	Default values associated with errors 66 thru 77 when flag 14 is set are explained in the programming chapter of the operating and programming manual.	A1	A for statement has no matching next statement.
66	Division by zero.	A2	A next statement encountered without a previous for statement.
	A mod B, with B equal to zero.	A3	Non-numeric parameter passed as a p-number.
67	Square root of negative number.	A4	No return parameter for a function call.
68	Tan (n * $\pi/2$ radians).	A5	No functions or subroutines running.
	Tan (n * 90 degrees).	A6	Improper p-number.
	Tan (n * 100 grads).	A7	Attempt to allocate local p-numbers from the keyboard.
	where n is an odd integer.	A8	Wrong number of parameters in fts, stf, fti, or itf function. stf or itf parameter must be a string (not a numeric). stf or itf parameter contains too few characters.
69	Ln or log of a negative number.	A9	Overflow or underflow in fts function.
70	Ln or log of zero.		Overflow in fti function.
			String Variables ROM missing for stf or itf functions.

## Error Codes

Errors B0 thru B8 may result during the binary disc initialization and disc error recovery routines. (9825 only)

- B0** Wrong syntax, argument out of range or variable not properly dimensioned.
- B1** Are than six defective tracks on the disc.
- B2** Verify error. Boots on the disc not identical to boots on the cartridge.
- B3** dtrk or tinit not allowed because error information lost or error not d5, d6, d7 or d9.
- B4** Attempt to access record for error correction which isn't part of data file.
- B5** Improper string length (inconsistent with length given in header).
- B6** Not enough space in computer buffer for data item. Item can't be placed in this part of buffer.
- B7** Missing Disk or String ROM.
- B8** Track still bad after tinit.
- C0** Missing General I/O or Extended I/O ROM.
- C1** Incorrect number of parameters.
- C2** Improper parameter specified.
- C3** Wrong parameter type.
- C4** Illegal buffer type for bred statement.
- C5** Key buffer overflow.
- C6** Too large or wrong sign of parameter.
- C7** Improper execution of store statement.
- C8** Illegal use of kret, mret, cret, or dret.
- C9** Missing 98626A Interface card.
- D0** Improper argument in disc statement.
- D1** Disk argument out of range.
- D2** Improper file size (must be from 1 thru 32767). No lines to store for save or savek.

<b>D3</b>	Invalid file name.
<b>D4</b>	File not found.
<b>D5</b>	Duplicate file name. Attempting to copy a non-data file to an existing file.
<b>D6</b>	Wrong file type.
<b>D7</b>	Directory overflow.
<b>D8</b>	Insufficient storage space on disc.
<b>D9</b>	Verify error due to cable, computer or drive problem. Bad data (reprint data).
<b>DISK IS DOWN</b> (9825 only)	
<b>UNABLE TO ACCESS DISK CONTROLLER</b> (9825 only)	
<b>d0</b>	Computer cannot access the disc controller.
<b>d1</b>	Firmware/driver out of synchronization.
<b>d2</b>	Too many defective tracks with init.
<b>d3</b>	All drives in system not powered on.
<b>d4</b>	Door opened while disc being accessed.
<b>d5</b>	Disk not in drive or drive not present.
<b>d6</b>	Write not allowed to protected disc.
<b>d7</b>	Record header error.
<b>d8</b>	Track not found.
<b>d9</b>	Data checkword error.
<b>E0</b>	Hardware failure (Press the RESET key).
<b>E1</b>	Verify error. Data not readable under reduced margins (reprint data).
<b>E2</b>	General I/O ROM missing.
<b>E3</b>	HP-IB error under interrupt.
<b>E4</b>	Wrong number of parameters.
<b>E5</b>	Improper buffer device or equate table usage.
<b>E6</b>	Multiple-listeners error.
<b>E7</b>	Buffer busy.
<b>E8</b>	Wrong parameter type.
<b>E9</b>	Timeout error.

## Error Codes

E5	Buffer underflow or overflow.	f3	Directory entry field overflow. Attempted file copy not possible.
E6	Parameter value out of range.	f4	Illegal structure on LIF format disc. The disc cannot be repacked.
E7	Parity failure.	f5	Disc copy attempted to a significantly larger disc. Use file copy to back up contents of disc.
E8	Improper use of iret statement. Attempt to DMA with HP-IB. Buffer or select code is busy.	f6	Disc copy attempted from 9825-compatible disc to LIF disc, or vice-versa. Only file copy is allowed across media formats.
E9	Illegal HP-IB operation.	f7	System record is not valid for LIF disc.
F0	File overflow when read or print executed.	f8	System record is not valid for 9825-compatible disc.
F1	98217A bootstraps not found (reload bootstraps). Wrong memory configuration for 98228A ROM. (9825 only).	f9	Statement not implemented on 9826A. (See Disc Programming Technical Appendix for a list of disc statements that are not implemented.)
F2	String read but wrong data type encountered.	G1	Incorrect format numbers.
F3	Attempt to read data item but type doesn't match.	G2	Referenced format statement has an error.
F4	Availability table overflow (repack).	G3	Incorrect I/O parameters.
F5	Attempt on end branch from other than running program.	G4	Incorrect select code.
F6	Unassigned data file pointer.	G5	Incorrect read parameter.
F7	Disk is down; line cannot be reconstructed. (9825 only).	G6	Improper conv statement parameters.
F8	Disk is down and <b>STOP</b> pressed. (9825 only).	G7	Unacceptable input data.
F9	System error (save files individually and reinitialize).	G8	Peripheral device down.
f0	Unable to access disc controller. This error has the same cause as the error which issued the "DISK IS DOWN" and "UNABLE TO ACCESS DISK CONTROLLER" messages, except the error is now trappable by on err.	G9	Interface hardware problem.
f1	No DMA card present for 9885 disc controller.	M1 <sup>1</sup>	Syntax error.
f2	Invalid msus syntax. Probable illegal device/format specifier.	M2	Improper dimensions. Array dimensions incompatible with each other or incompatible with the stated operation.
		M3	Improper redimension specification. New number of dimensions must equal original number; new size cannot exceed original size.

<sup>1</sup> Press the RECALL key to position the cursor at the location of the error.

## Error Codes

**M4<sup>1</sup>** Operation not allowed. An array which appears to the left of → cannot also appear on the right.

**M5** Matrix cannot be inverted. Computed determinant = 0.

## 9862A Plotter ROM Error Codes (9825 only)

**P1** Wrong state.  
Statements executed out of order.

**P2** Wrong number of parameters.

**P3** Wrong type of parameters. Parameters for a label statement must be expressions, text, or string variables.

**P4** Scale out of range. Maximum value is less than or equal to the minimum value.

**P5** Integer out of range. Pen control parameter is out of the range - 32768 thru - 32767 or the select code is not 0 or in the range 2 thru 15.

**P6** Character size out of range. Width or height in letter statement is zero or there is an integer overflow in csize calculations or results.

**P7** Not used.

**P8** Axes origin off-scale. X, Y specified for axis statement doesn't fall on plotter surface.

**PLT DOWN** Check interface connection and select code setting; be sure LINE and CHART HOLD are on.

## Graphics/Plotter Error Codes

**P1** Attempt to store into constant. Occurs when one or more parameters in a dig statement are constants rather than variables.

**P2** Wrong number of parameters. Occurs on instructions with numeric-only parameter lists (scl, ofs, plt, iplt, cplt, xax, yax, lim, dig, csiz, line, pen#, and psc). In certain unusual cases where a parameter list contains user-level function calls, an instruction having an incorrect number of parameters may be executed.

**P3** Wrong type of parameter or illegal parameter value.

**P4** No HP-IB device number specified. Occurs when psc parameter is from 0 thru 14 and an HP-IB card is at the corresponding select code.

**P5** Pen control value not from - 32768 thru 32767. Hardware transmission error occurs between plotter and computer.

**P6** No HP-IB card at specified select code.

**P7** axe or ltr statement encountered; these are 9862 Plotter commands only.

**P8** Computer **STOP** key cancelled operation. Occurs when the plotter fails to respond for three seconds after the **STOP** key has been pressed.

**P9** No graphics hardware present. If you have a 9826A, you should not experience this error. Consult your HP field sales and service office for advice.

**p0** Transmission error. The calculator has received an illegal ASCII input from the plotter.

**p1** Instruction not recognized. The plotter has received an illegal character sequence.

<sup>1</sup> Press the RECALL key to position the cursor at the location of the error.

## Error Codes

**p2** Wrong number of parameters. Too many or too few parameters have been sent with an instruction.

**p3** Bad parameter. The parameters sent to the plotter with an instruction are out of range for that instruction.

**p4** Illegal character. The character specified as a parameter is not in the allowable set for that instruction.

**p5** Unknown character set. A character set out of the range 0 thru 4 has been designated as either the standard or alternate character set.

**p6** Position overflow. An attempt to draw a character or perform a cplot that is located outside of the plotters numeric limit of -32768 thru +32767.

Errors generated by write (wrt) and read (red) statements to an external plotter will typically be displayed in the next executed plotter ROM statement. This can be avoided by using an output error command (wrt select code, "OE";) followed by a read statement (red select code, variable) to check for errors after read or write statements that address the plotter.

**S0** Invalid set of strings in data list of ldf statement.

**S1** Improper argument for string function or string variable.

**S2** More parameters than expected for string function or string variable.

**S3** Accessing or assigning to non-contiguous string, num function of null string.

**S4** Trying to find the value of non-numeric string or null string.  
Exponent too large.  
Exponent format invalid (e.g., 1e+ +).

**S5** Invalid destination type for string assignment.

**S6** Parameter is zero, or negative, exceeded dimensioned size.

**S7** Invalid sequence of parameters for string variable.

**S8** String not yet allocated.

**S9** String previously allocated.

**SPARE DIR.** Maximum string length exceeded; additional string length must be specified in dim statement.

## 9826 HPL Extended Errors

**X0** No memory or I/O card present at specified address. This error should not be encountered when programming from HPL. Consult your HP field sales and service office for advice concerning this error should you receive it.

**X1** A read statement was executed with no data remaining. Either a data statement must be added or a rstr statement must be added to reset the data pointer to the desired data statement in the program.

## Option ROM Syntax and Errors

The following syntax and error messages require the appropriate option ROM be plugged into your Desktop Computer. See the alphabetical listing for syntax and error descriptions. Darkly printed mnemonics are 9826 extensions to HPL.

### 9826 HPL Extensions

Syntax: aclr, aoff, aon, data, get, kloff, klon,  
pbeep, pi, read, rstr, sqr

Errors: X0 thru X2

### Advanced Programming

Syntax: cll, for, fti, fts, itf, next, stf, xref

Errors: A0 through A9

### Disk Programming

Syntax asgn, cat, chain, copy, drive, dtype dump,  
files, get, getb, getk, getm, init, kill, killall,  
load, **msi**, on end, open, **rcb**, renm, repk,  
resave, rpt, rread, save, **saveb**, savek,  
savem, sprt, sread, type, von, voff.

Errors: D0 thru D9, d0 thru d9, F0 thru F9, f0  
thru f9.

### Extended I/O Programming

Syntax: add, band, bit, buf, cli, clr, cmd, cmp, ctbl,  
dev, dto, eir, eor, equ, iof, ior, ios, iret, lcl,  
llo, mdec, moct, on err, oni, otd, par, pct,  
pol, polc, polu, rdi, rds, rem, rot, rqs, shf,  
tfr, time trg, wti

Errors: E0 through E9

### General I/O Programming

Syntax: conv, fmt, list#, **prtsc**, rdb, rds, red,  
**rtime**, **stime**, wrt, wtb, wtc

Errors: G1 through G9

### Matrix Programming

Syntax: aprt, ara, dim, idn, ina, inv, ldf, mat, rcf,  
rdm, smpy, trn

Errors: M1 through M5

### Plotter Programming

Syntax: **bplt**, **cplt**, **csiz**, **dig**, **gclr**, **gdump**, **gload**,  
**goff**, **gon**, **gptr**, **gstore**, **iplt**, **lbl**, **lim**, **line**,  
**ofs**, **pclr**, **pen**, **pen#**, **plt**, **psc**, **ptyp**, **scl**,  
**xax**, **yax**

Error: P1 through P9, p0 through p6

### String Programming

Syntax: cap, char, dim, dsp, enp, ent, if, ldf, len,  
num, pos, prt, rcf, str, val

Errors: S0 through S9

### Systems Programming

Syntax: asc, bred, **cret**, **cycle**, **dret**, eol, key,  
**knob**, **kstat**, kret, **mret**, nal, on **cycle**,  
on **delay**, on key, on **knob**, on **match**,  
**pkbd**, rkbd, rss, **sfk**, store, **sysboot**,  
wsc, wsm

Errors: C0 through C9

### 9862A Plotter ROM (9825 only)

Syntax: axe, cplt, csiz, iplt, lbl, ltr, ofs, pen, plt,  
psc, ptyp, scl

Errors: P1 through P8

## Interface Card Registers

This section describes the register bit maps for the various interface cards that are available for the 9825/9826. See the associated Installation and Service Manual for a complete description of a specific interface.

### 98622A/98032A Register Map

	IN	OUT
R4	DATA IN	DATA OUT
R5	STATUS	CONTROL
R6	HIGH BYTE DATA	HIGH BYTE DATA
R7	(not used)	TRIGGER

R4-IN: Read 16 bits (lower 8 bits if jumper B is not installed) of data from the input data latches. Sets I/O line to input.

R4-OUT: Write 16 bits (lower 8 bits if jumper F is not installed) of data to the output data latches. Sets I/O line to output.

R5-IN: Read 98032A interface status byte.

#### R5 Status (R5-IN)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INT	DMA	1	0	IID	IOD	STI1	STI0

R5-OUT: Write 98032A interface control byte.

#### R5 Control (R5-OUT)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INT	DMA	RESET	AH	—	—	CTL1	CTL0

R6-IN: Read 16 bits (upper 8 bits if jumper B is not installed) of data from the input data latches. Does not affect I/O line.

R6-OUT: Write 16 bits (upper 8 bits if jumper F is not installed) of data to the output data latches. Does not affect I/O line.

R7-OUT: Sets PCTL to initiate an input/output handshake, depending on the state of the I/O line from the last R4 access.

### 98623A/98033A Register Map

	IN	OUT
R4	DATA IN	DATA OUT
R5	STATUS	CONTROL
R6	(not used)	(not used)
R7	(not used)	TRIGGER

R4-IN: Read one 8-bit ASCII character from the 98033A BCD-to-ASCII translator.

R4-OUT: Latch one byte of data to the 8-bit output port. (98623 only.)

R5-IN: Read 98033A interface status byte.

#### R5 Status (R5-IN)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INT	0	1	0	0	0	0	0

R5-OUT: Write 98033A interface control byte.

#### R5 Control (R5-OUT)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INT	—	RESET	—	—	—	—	—

## 98624A/98034A Read Status

R7-OUT: An output to R7 (actual value output is a "don't care") causes the 98033A to place the next ASCII character in the sequence representing the reading into the R4-IN register. After 16 characters have been so placed, the next R7-OUT causes a new reading to be taken (i.e., the card sets CTLA and CTLB to start a data handshake with the BCD device) and places the first character of that reading in the R4-IN register.

## 98624A/98034A Register Map

	IN	OUT
R4	DATA IN	DATA OUT
R5	STATUS	CONTROL
R6	STATUS/DATA	COMMANDS
R7	PARALLEL POLL	DIRECT BUS CONTROL

R4-IN: Initiates a data byte input sequence.

R4-OUT: Transfers one byte of data to the bus.

R5-IN: Initiates a status read sequence.

R5-OUT: Outputs a control byte to enable the 98034A for various interrupt conditions.

### R5 Control (R5-OUT)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SRQ	ACT	TLK	LST	IRF	ORE	SEE NOTE 2	SEE NOTE 1

Note 1: Bit 0, when set, causes the STS line to be cleared when EOI is received.

Note 2: 9825: DCL, SDC, Error

9826 HPL: DCL, SDC, IFC, GET

R6-IN: Completes a data byte input sequence.

Clears ATN.

Delivers 98034A status bytes.

Completes a parallel poll input sequence.

R6-OUT: Sets the ATN line true and outputs a byte of command or addressing information.

R7-IN: Initiates a parallel poll byte request.

R7-OUT: Direct<sup>1</sup> bus control.

### R7 Out, Bit 7 Set

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	0	0	EOI	IFC	ATN	REN	SRQ

### R7 Out, Bit 7 Clear

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	SRQ	X	X	X	X	X	X

R7-OUT: Service Request control and serial-poll response byte.

X = user definable.

## 98624A/98034A Read Status Sequence

rds(7, A, B, C, D) → E

### Status Byte 1 (A)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	REM/ LOC	LLO	GET	DCL	IFC	ERROR

<sup>1</sup> After executing this R7-OUT instruction, the 98034A will clear the STS line if an illegal operation (e.g., specifying ATN if the 98034A is not active controller) is indicated.

## 98623A/98033A Registers

### Status Byte 2 (B)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>

### Status Byte 3 (C)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EOI	REN	SRQ	ATN	IFC	NDAC	NRFD	DAV

### Status Byte 4 (D)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SRQ	ACT	TLK	LST	SAC	1	0	EOR

## 98626A/98036A Register Map

### IN

R3 (not used)	CONFIGURATION
R4 DATA IN, R4E	DATA OUT, R4C, R4D
R5 STATUS	CONTROL
R6 LINE STATUS	LINE CONTROL
R7 (not used)	TRIGGER

### R3 OUT

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Number of Stop Bits 00=not valid 01=1 bit 10=1.5 bits 11=2 bits	Parity Type 0=Odd 1=Even	Parity Enable 0=Disable 1=Enable		Character Length 00=5 bits 01=6 bits 10=7 bits 11=8 bits		98036A Bit Rate Factor Not Used for 9826A	

## R4C Mode Word

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

## R4D USART Control Word

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Always 0	USART Reset	No Connect <sup>1</sup> (Standard) Request To Send Pin 4 (Option 001)	Reset Status Bits of USART Status Word	Send Break Character	Enable Data Receiver	Data Set Ready Pin 6 (Standard) Data Terminal Ready Pin 20 (Option 001)	Data Enable Transmitter

<sup>1</sup>The CTS line is controlled by (DTE) RTS line, not by R4D, bit 5.

## R4E USART Status Word

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Request to Send Pin 4 (Standard) Data Set Ready Pin 6 (Option 001)	Break Indicator	Framing Error	Overrun Error	Parity Error	Transmitter Empty	Receiver Ready	Transmitter Ready

Registers are on the following pages.

**R5 OUT Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Interface Interrupt Enable		Programmed Interface Reset			Interrupt Control 2 Receiver Control	Interrupt Control 2 Transmitted Control	R4 Control 0=Data IN/OUT 1=Control/Status

**R5 IN Register**

Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Peripheral Status 1 Mode	Interface Interrupt Enable Status	0	Interface I.D. 0	Interface I.D. 1	0	0	Control Status 2 Receiver	Control Status 1 Transmitter Mode

**R6 OUT Register (standard cable)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
			Half/Full Speed Control (Interface)	Ring Indicator Pin 22	No Connect	Secondary Carrier Detect Pin 12	Data Carrier Detect Pin 8

**R6 IN Register (standard cable)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Always 1	Always 0	No Connect	Secondary Request To Send Pin 19				

**R6 OUT Register (option 001 cable)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
			Half/Full Speed Control	No Connect	No Connect	Data Signal Rate Select Pin 23	Secondary Request To Send Pin 19

**R6 IN Register (option 001 cable)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Always 1	Secondary Carrier Detect Pin 12	Ring Indicator Pin 22	Data Carrier Detect Pin 8				

# ASCII Table

ASCII Char.	EQUIVALENT FORMS			
	Binary	Oct	Hex	Dec
NUL	00000000	000	00	0
SOH	00000001	001	01	1
STX	00000010	002	02	2
ETX	00000011	003	03	3
EOT	00000100	004	04	4
ENQ	00000101	005	05	5
ACK	00000110	006	06	6
BEL	00000111	007	07	7
BS	00001000	010	08	8
HT	00001001	011	09	9
LF	00001010	012	0A	10
VT	00001011	013	0B	11
FF	00001100	014	0C	12
CR	00001101	015	0D	13
SO	00001110	016	0E	14
SI	00001111	017	0F	15
DLE	00010000	020	10	16
DC1	00010001	021	11	17
DC2	00010010	022	12	18
DC3	00010011	023	13	19
DC4	00010100	024	14	20
NAK	00010101	025	15	21
SYN	00010110	026	16	22
ETB	00010111	027	17	23
CAN	00011000	030	18	24
EM	00011001	031	19	25
SUB	00011010	032	1A	26
ESC	00011011	033	1B	27
FS	00011100	034	1C	28
GS	00011101	035	1D	29
RS	00011110	036	1E	30
US	00011111	037	1F	31

ASCII Char.	EQUIVALENT FORMS			
	Binary	Oct	Hex	Dec
space	00100000	040	20	32
!	00100001	041	21	33
"	00100010	042	22	34
#	00100011	043	23	35
\$	00100100	044	24	36
%	00100101	045	25	37
&	00100110	046	26	38
'	00100111	047	27	39
(	00101000	050	28	40
)	00101001	051	29	41
*	00101010	052	2A	42
+	00101011	053	2B	43
,	00101100	054	2C	44
-	00101101	055	2D	45
.	00101110	056	2E	46
/	00101111	057	2F	47
0	00110000	060	30	48
1	00110001	061	31	49
2	00110010	062	32	50
3	00110011	063	33	51
4	00110100	064	34	52
5	00110101	065	35	53
6	00110110	066	36	54
7	00110111	067	37	55
8	00111000	070	38	56
9	00111001	071	39	57
:	00111010	072	3A	58
;	00111011	073	3B	59
<	00111100	074	3C	60
=	00111101	075	3D	61
>	00111110	076	3E	62
?	00111111	077	3F	63

ASCII Char.	EQUIVALENT FORMS			
	Binary	Oct	Hex	Dec
@	01000000	100	40	64
A	01000001	101	41	65
B	01000010	102	42	66
C	01000011	103	43	67
D	01000100	104	44	68
E	01000101	105	45	69
F	01000110	106	46	70
G	01000111	107	47	71
H	01001000	110	48	72
I	01001001	111	49	73
J	01001010	112	4A	74
K	01001011	113	4B	75
L	01001100	114	4C	76
M	01001101	115	4D	77
N	01001110	116	4E	78
O	01001111	117	4F	79
P	01010000	120	50	80
Q	01010001	121	51	81
R	01010010	122	52	82
S	01010011	123	53	83
T	01010100	124	54	84
U	01010101	125	55	85
V	01010110	126	56	86
W	01010111	127	57	87
X	01011000	130	58	88
Y	01011001	131	59	89
Z	01011010	132	5A	90
[	01011011	133	5B	91
\	01011100	134	5C	92
]	01011101	135	5D	93
^	01011110	136	5E	94
-	01011111	137	5F	95

ASCII Char.	EQUIVALENT FORMS			
	Binary	Oct	Hex	Dec
~	01100000	140	60	96
a	01100001	141	61	97
b	01100010	142	62	98
c	01100011	143	63	99
d	01100100	144	64	100
e	01100101	145	65	101
f	01100110	146	66	102
g	01100111	147	67	103
h	01101000	150	68	104
i	01101001	151	69	105
j	01101010	152	6A	106
k	01101011	153	6B	107
l	01101100	154	6C	108
m	01101101	155	6D	109
n	01101110	156	6E	110
o	01101111	157	6F	111
p	01110000	160	70	112
q	01110001	161	71	113
r	01110010	162	72	114
s	01110011	163	73	115
t	01110100	164	74	116
u	01110101	165	75	117
v	01110110	166	76	118
w	01110111	167	77	119
x	01111000	170	78	120
y	01111001	171	79	121
z	01111010	172	7A	122
{	01111011	173	7B	123
	01111100	174	7C	124
}	01111101	175	7D	125
~	01111110	176	7E	126
DEL	01111111	177	7F	127

## 9826A ASCII Control Codes

CTRL of	ASCII Value	ASCII Character	9826A Key Pressed (1)	Displayed Character (3)
@	0	NUL	reserved	
A	1	SOH	PAUSE	
B	2	STX	REWIND	
C	3	ETX	HOME LEFT	
D	4	EOT	HOME RIGHT	
E	5	ENQ	TO TOP	
F	6	ACK	TO BOTTOM	
G	7	BEL	RESULT	
H	8	BS	INSERT LINE	
I	9	HT	DELETE LINE	
J	10	LF	EXECUTE	
K	11	VT	RECALL	
L	12	FF	RUN	
M	13	CR	ENTER	
N	14	SO	CLR TO END	
O	15	SI	CLR SCREEN	
P	16	DLE	DOWN ARROW	
Q	17	DC1	UP ARROW	
R	18	DC2	CLEAR LINE	
S	19	DC3	PRINT ALL	
T	20	DC4	LEFT ARROW	
U	21	NAK	RIGHT ARROW	
V	22	SYN	INSERT CHAR	
W	23	ETB	DELETE CHAR	
X	24	CAN	STEP	
Y	25	EM	CONTINUE	
Z	26	SUB	DUMP GRAPHICS	
[	27	ESC	DISPLAY FUNCTIONS	
\(2)	28	FS	EDIT	
]	29	GS	CAPS LOCK	
.	30	RS	ALPHA	
-	31	US	GRAPHICS	

(1) This is the 9826A pkbd keypress and the key pressed from a remote ASCII keyboard.

(2) This is the shift ")" key on the Numeric Keypad.

(3) This is the displayed character if "DISPLAY FUNCTIONS IS ON".

## Notes

Scan Copyright ©  
The Museum of HP Calculators  
[www.hpmuseum.org](http://www.hpmuseum.org)

Original content used with permission.

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

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