# HP-41CX Quick Reference Guide

# **Contents**

The HP-41CX Keyboards	
Normal	2
User	4
Alpha	6
Alarm Catalog	8
Stopwatch	10
Text Editor	12
How to Execute Functions	14
Function Set	14
Display Features	27
Organization of Memory	28
Main Memory	28
Extended Memory	29
Storing and Executing Programs	30
Time and Alarm Formats	31
Time Values	31
Alarm Format	32
Acknowledging and Clearing Message Alarms	33
The Catalogs	34
Character Codes	36
The Flags and Their Status	38
List of Errors	39

## The Normal Keyboard



- 1. Primary Function.
- 2. Alternate Function.
- 3. Shift Key. Press first to carry out an alternate function.
- 4. Toggle Keys.
- 5. CLX/A Clear X or Clear Alpha. Clears the entire register.
- 6. Back Arrow. Backspaces and erases one character at a time (if entry has not been terminated).
- 7. XEQ Execute.

Used to execute functions and programs not assigned to keys. See page 14 in this guide.

# The User Keyboard



### 1. USER User.

Activates and deactivates the User keyboard.

#### 2. Assigning a Function or Global Label to a Key.

- 1. Press ASN.
- 2. Press ALPHA.
- 3. Enter the function name or global label.
- 4. Press ALPHA.
- 5. Press the key to which you want the function assigned. (To restore a key to its Normal function, skip step 3.)

#### 3. Executing a User Function.

- 1. Make sure the User keyboard is active.
- 2. Press the redefined key.

Any key *not* redefined retains its Normal function (except in the top two rows).

#### 4. Local Label Searching.

If a key in the top two rows (or shifted top row) is not reassigned, the HP-41CX will perform local label searching if one of those keys is pressed when the User keyboard is active. If a matching local label (A through J or a through e) is found in the current program, execution of the program starts there. If that label is not found in the current program, the Normal function of that key is executed.

The Alpha Keyboard



- 1. Primary Function.
- 2. Alternate Function.

```
3. Shift Key.
Press first to carry out an alternate function.
```

4. 🕞 Append.

Press  $\vdash$  first to have the following Alpha entry be appended to (rather than overwrite) the previous Alpha entry.

- 5. <u>ALPHA</u> Alpha. Activates and deactivates the Alpha keyboard.
- 6. ASTO Store From Alpha. Stores the leftmost six characters of the Alpha register into the specified register.
- 7. ARCL Recall Into Alpha. Recalls the contents of the specified register and appends them to the Alpha register.
- 8. **CLA** Clear Alpha Register.
- 9. 🔸 Back Arrow.

Backspaces and erases one character at a time (if entry has not been terminated).

#### 10. AVIEW View Alpha Register.

Used primarily as a program instruction to display the Alpha register during a running program.

The Alarm Catalog Keyboard



- 1. 📕 🖸 Clear Alarm.
- 2. M Alarm Message.
- R Alarm Repeat Interval.
   R Reset Alarm Interval by Repeat Interval.
- 4. D Alarm Date.
- 5. [SST], [BST] Step Through Catalog Listing.
- 6. Exit Alarm Catalog.
- 7. T Alarm Time.
  T Current Time.
- 8. R/S Run/Stop Catalog Listing.

# The Stopwatch Keyboard



- 1. SPLIT Take Split.
- 2. **ASPLIT Set/Clear Delta Split Mode.**
- 3. Digits. Set new register address.
- 4. Register Address.
- 5. SST, BST Increment/Decrement Register Address.
- 6. Rnnn Set/Cancel Display of Three-Digit Address.
- 7. **EXIT** Exit Stopwatch.
- 8. CLEAR Clear Time to Zero.
- 9. REG# Suppress/Restore Display of Register Address.
- 10. R/S Run/Stop Stopwatch.



- 1. Empty-Record Indicator.
- 2. Cursor (Pointer).
- 3. Record Number.
- 4. **+1**, **+12** Move Cursor Left.
- 5. EXIT Exit Text Editor.
- 6. INSERT Insert/Replace Mode Toggle.
- 7. Insert Mode Active.
- 8. [1+], [12+] Move Cursor Right.
- 9. [ALPHA] Alpha/Numeric Keyboard Toggle.
- 10. GTO Go To Record nnn.
- 11. **REC+** Go To Previous Record.
- 12. REC+ Go To Next Record.
- 13. **\_\_\_\_\_** Delete Record.
- 14. 
   Delete Character.
- 15. **(+REC+)** Insert New Record Before Current Record.
- 16. <u>+REC+</u> Insert New Record After Current Record.

# How to Execute Functions (Alpha Execution)

If a function has its own key (whether on the Normal keyboard or the User keyboard), you can perform its operation by pressing that key—such as for 1/x—or by pressing the shift key and then that key—such as for  $x^2$ . (Remember to supply any necessary numbers or labels first.)

If a function does not appear on the keyboard—such as  $\boxed{\mathsf{TIME}}$ —you can perform it using either Alpha execution or a User-defined key on the User keyboard. How to assign functions to User keys is shown on page 5 of this guide. Alpha execution is shown below:

- 1. Press XEQ.
- 2. Press ALPHA to activate the Alpha keyboard.
- 3. Spell out the Alpha name of the desired function, or the global label of the desired program.
- 4. Press <u>ALPHA</u> to deactivate the Alpha keyboard and end the procedure.

If the function needs a parameter, it will cue for it with the  $\_$  input cue.

# **Function Set**

This is an alphabetical list of the HP-41CX functions, including brief definitions. For a more detailed summary of these functions, refer to the Function Tables in volume 2 of the owner's manual. For page references to the complete descriptions within the owner's manual, refer to the Function Index in either volume of the owner's manual.

Note that usually you supply any needed operands *before* you execute the function (the operator). The exceptions are the *parameter functions*, which cue you for information *after* you execute the function. Parameter functions are shown below with their parameters, such as **ARCL** *nn*.

Function names printed in blue are *Alpha names* and use Alpha execution or User-keyboard execution. Function names printed in black or gold are *keyboard names*, and have keys for execution on the Normal keyboard.

Function

#### Definition

Back arrow. Deletion.
Append to Alpha register.
Plus.
Minus.
Multiplied by.
Divided by.
Reciprocal.
Common exponential.
Absolute value.
Arc cosine.
<i>Alpha date</i> . Append date to Alpha reg.
Advance printer paper.
<i>Alpha length</i> . No. of characters in Alpha reg.
Alarm catalog.
Alarm now. Activate oldest past- due conditional or control alarm.
Alpha keyboard toggle.
<i>Alpha number</i> . Find first digit string in Alpha reg.
Alpha keyboard off.
Alpha keyboard on.

#### Function Definition APPCHR Append characters to record in text file. APPREC Append record to text file. ARCL nn (ARCL nn) Alpha recall. Append reg. nn to Alpha reg. Alpha recall record. Append record to Alpha reg. Alpha rotate n places. AROT ASHE Alpha shift six characters to the left. ASIN (SIN<sup>-1</sup>) Arc sine. ASN name, key Assign function or label to User (ASN) key. ASCII room. Bytes available in ASROOM text file Alpha store. Copy first six ASTO nn (ASTO nn) characters from Alpha reg. into reg. nn. ATAN (TAN-1) Arc tangent. Alpha time. Append time to ATIME Alpha reg. Alpha time 24-hour. Append time ATIME24 to Alpha reg. in CLK24 format. Alpha to X. Shift leftmost char-ATOX acter out of Alpha reg. and convert to its character code. AVIEW (AVIEW) Alpha view. BEEP (BEEP) Beeper. BST (BST) Back step through program lines. CAT n (CATALOG n) List catalog n (1 to 6). CF nn (CF nn) Clear flag nn (00 to 29). CHS (CHS) Change sign.

# Function

# Definition

CLA (CLA)	Clear Alpha.
CLALMA	Clear alarm by Alpha. Clear alarm whose message matches Alpha reg.
CLALMX	Clear alarm by X. Clear $n$ th alarm.
CLD	Clear display of message.
CLFL	<i>Clear file</i> named (text or data file).
CLK12	Clock 12-hour (format).
CLK24	Clock 24-hour (format).
CLKEYS	Clear all User keys.
CLKT	Clock time only (format).
CLKTD	Clock time and date (format).
CLOCK	Display clock.
CLP label	<i>Clear program</i> specified by global label.
CLRALMS	Clear all alarms.
CLRG	Clear all data registers.
CLRGX	Clear registers by X (bbb.eeeii). Clear every <i>ii</i> th reg. from $R_{bbb}$ through $R_{eee}$ .
	Clear summations. Clear statis- tics regs.
CLST	Clear stack.
CLX (CLx)	Clear X-register (the usual display).
COPY	Copy ROM program specified by global label.
CORRECT	Set time and adjust accuracy factor.
COS (COS)	Cosine.
CRFLAS	Create file-ASCII. Create text file of given name and length.

Function	Definition			
CRFLD	<i>Create file-data</i> of given name and length.			
D-R	Degrees to radians conversion.			
DATE	Value for the date.			
DATE+	Add number of days (in X-register) to date (in Y-register) to find new date.			
DDAYS	Delta days. Find number of days between dates in X- and Y-registers.			
DEC	Decimal. Octal to decimal conversion.			
DEG	Degrees mode set.			
DEL	Delete nnn program lines, incl. current line.			
DELCHR	Delete n characters from current text file, starting at pointer.			
DELREC	Delete current record.			
DMY	Day-month-year format.			
DOW	Day of week of the given date $(0=Sun.)$ .			
DSE nn	Decrement and skip if less than or equal. Given $iiiii.fffcc$ in $R_{nn}$ , decrement $iiiii$ by $cc$ and skip next line if $iiiii$ is now $\leq fff$ .			
ED	Text editor.			
EEX	Enter exponent.			
EMDIR	Extended memory directory (catalog 4).			
EMDIRX	Extended memory directory by $X$ . Find <i>n</i> th file's name and type.			
EMROOM	Extended memory room. No. of regs. available.			
END	End of program.			

#### Function Definition ENG n (ENG n) Engineering display. Use n+1digits and powers of $10^{3n}$ . ENTER ( ENTER + ) Separate sequential numbers. $[E \uparrow X] (e^x)$ Natural exponential. E+X-1 For arguments close to zero. Factorial FACT FC? nn Flag nn clear? If not, skip next line. Flag nn clear? FC?C nn Clear flag nn. Fixed-point display with n deci-FIX n (FIX n)mal places. File size (registers) of given file. FLSIZE FRC Fractional part. FS? nn (FS? nn) Flag nn set? If not, skip next line. FS?C nn Flag nn set? Clear flag nn. Get ASCII. Copy mass-storage GETAS text file. After 10 sec., return key code of GETKEY key pressed (0 if none). GETKEYX Get key by X. After given no. of sec., return keycode (Y-register) and character code (Xregister). Get program. Replace last pro-GETP gram with program file named. Get all registers from given data GETR file and copy to main memory. Get record from current text file GETREC

*Get record* from current text file and copy to Alpha reg., starting at pointer *rrr.ccc*.

Function	Definition		
GETRX	Get registers by X (bbb.eee). Copy regs. in current data file (start- ing at pointer) to $R_{bbb}$ through $R_{eee}$ in main memory.		
GETSUB	<i>Get subroutine</i> from named file and copy into main memory.		
GETX	<i>Get X</i> -value from current data- file reg.		
GRAD	Set Grads mode.		
GTO label			
(GTO label)	Go to. Program branch to given label.		
GTO • nnn	Go to (dot). Move current line to line nnn or global label.		
GTO ···	Go to (dot dot). Move current line to end of program memory and pack memory.		
HMS	To hours-minutes-seconds. Con- vert from decimal hours.		
HMS+	Hours-minutes-seconds plus. Add degrees or times.		
HMS-	Hours-minutes-seconds minus. Subtract degrees or times.		
HR	<i>To decimal hours.</i> Convert from HMS.		
INSCHR	<i>Insert characters</i> from Alpha reg. into text file starting at pointer.		
INSREC	<i>Insert record.</i> Copy from Alpha reg. to new record at pointer.		
INT	Integer part.		
ISGNN (ISGNN)	Increment and skip if greater. Given <i>iiiii.fffcc</i> in $\mathbf{R}_{nn}$ , increment <i>iiiii</i> by <i>cc</i> and skip next line if <i>iiiii</i> is now > <i>fff</i> .		

Function	Definition		
LASTX (LASTx)	Recall number from LAST X reg.		
LBL label (LBL label)	Label.		
LN (LN)	Natural log.		
LN1+X	For arguments close to 1.		
LOG (LOG)	Common log.		
MDY	Month-day-year format.		

Means of accumulated *x*- and *y*-values.

 $y \mod x$ .

Octal. Decimal to octal conversion.

Turn off computer.

Continuous on. (Cancels automatic turn-off.)

On/off toggle.

Polar to rectangular conversion. Enter  $\theta$ , then r. Returns x in X-reg., y in Y-reg.

Pack program memory.

Programmable assign. See ASN.

Programmable clear-programs. Clear program named and all following programs.

x percent of y.

Percent change from y to x.

Value of  $\pi$  to nine decimal places.

Position in Alpha. Find position of string (specified in X-register) in Alpha reg.

Position in file. Pointer value of string (specified in Alpha reg.) in text file.

21

POSA

%CH [РІ] ( \_\_\_)

MEAN

MOD

OFF

ON

 $\boxed{\mathsf{P}}_{\mathsf{R}} ( \boxed{\mathsf{P}}_{\mathsf{R}} )$ 

PACK

PASN

PCLPS

**%** (**%**)

POSFL

Program mode toggle.
Display the message in Alpha reg. and stop program (allow- ing input).
<i>Pause.</i> Interrupt program for a second.
Programmable size. See SIZE.
Purge file named.
Roll up stack.
Radians to degrees conversion.
Rectangular to polar conversion. Enter y, then x. Returns r in X-reg., $\theta$ in Y-reg.
Run/stop program.
Radians mode.
Recall (copy) value from $R_{nn}$ .
Recall accuracy factor for clock.
Recall alarm parameters for alarm $n$ .
Recall flag status of flags 00-43.
<i>Recall pointer</i> value for current file.
<i>Recall pointer by Alpha.</i> Recall pointer value for file named.
Recall stopwatch time.
Roll down stack.
Register move. Given sss.dddnnn, copy nnn registers from $R_{sss}$ on, to $R_{ddd}$ on.
Register swap. Given sss.dddnnn, swap nnn registers from $R_{sss}$ on, with $R_{ddd}$ on.
<i>Resize file</i> (text or data) as specified.

Function	Definition			
RND	Round.			
RTN (RTN)	<i>Return</i> program flow from sub- routine to main program.			
RUNSW	Run stopwatch.			
SAVEAS	Save ASCII. Copy text file named to mass-storage file named.			
SAVEP	Save program named to program file named.			
SAVER	Save all registers in the given data file.			
SAVERX	Save registers by X (bbb.eee). Copy $R_{bbb}$ through $R_{eee}$ to the current data file.			
SAVEX	<i>Save x</i> -value in current data-file reg.			
SCIN (SCIN)	Scientific notation with $n$ decimal places.			
SDEV	Standard deviations of accu- mulated x- and y-values.			
SEEKPT	<i>Seek pointer</i> . Set given pointer value for current text or data file.			
SEEKPTA	Seek pointer by Alpha. Set given pointer value for the text or data file named.			
SETAF	Set accuracy factor for clock.			
SETDATE	Set date of clock.			
SETIME	Set time of clock.			
SETSW	Set stopwatch starting time.			
SFnn (SFnn)	Set flag nn (00 to 29).			
Σ+ (Σ+)	Summation plus. Add data val- ue(s) to statistical accumulation.			

Function	Definition			
Σ- (Σ-)	Summation minus. Delete data value(s) from statistical accumulation.			
<b><u>SREG</u></b> nn	Statistics registers set to $Rnn$ through $Rnn+5$ .			
ΣREG?	Find address of first statistics reg.			
SIGN	1 or $-1$ for numbers, 0 for non- numbers, $+1$ for zero.			
SIN (SIN)	Sine.			
SIZE nnn	Allocates <i>nnn</i> regs. to data storage.			
SIZE?	No. of regs. allocated to data storage.			
SQRT (IT)	Square root.			
SST (SST)	Single step to next program line.			
ST+nn (STO(+nn)	Store plus. $R_{nn} + x$ ; result in $R_{nn}$ .			
ST-nn (STO-nn)	Store minus. $R_{nn} - x$ ; result in $R_{nn}$ .			
ST∗nn (STO∣×nn)	Store multiply. $R_{nn} \times x$ ; result in $R_{nn}$ .			
ST/ nn (STO(+ nn)	Store divide. $R_{nn} \div x$ ; result in $R_{nn}$ .			
STOnn (STOnn)	Store copy of $x$ in $R_{nn}$ .			
STOFLAG	Restore flag status of flags 00-43 from X-reg. Or: restore status of flags bb thru ee given bb.ee in X and flag data in Y.			
STOP (R/S)	Stop a running program.			
STOPSW	Stop stopwatch.			
SW	Stopwatch. Activate Stopwatch keyboard.			

#### Function

SWPT

T+X

TAN (TAN) TIME TONE n USER VIEW nn (VIEW nn)  $X \uparrow 2$   $(x^2)$ [X = 0?] (x = 0?)X ≠ 0? X < 0? X <= 0? X > 0?X = Y? (x = y?) $X \neq Y?$ X < Y? $[X < = Y?] ([x \le y?])$ X > Y? (x > y?) X = NN? $X \neq NN?$ X < NN? $X \le NN?$ X > NN?X > = NN?X <> nnX <> F

#### Definition

Stopwatch and pointers. Given sss.rrr, activate Stopwatch kbd. and set storage (sss) and recall (rrr) pointers.

*Time plus X.* Adjust time by increment given.

Tangent.

Value for the current time.

 $0 \leq n \leq 9.$ 

User keyboard toggle.

Display contents of R<sub>nn</sub>.

Square.

Conditional. If not true, skips next program line.

Conditional. Uses contents of  $R_{nn}$  (NN specified in Y-register) for comparison. If not true, skips next program line.

X exchange with  $R_{nn}$  contents. X exchange flags (status of flags

00-07).

#### Function

#### Definition

X<>Y (x & y)

XEQ name (XEQ name)

XTOA

 $\begin{array}{c} XYZALM \\ \hline Y \blacklozenge X & (y^{x}) \end{array}$ 

X exchange Y contents.

Execute given function or label.

X to Alpha. Convert x (a character code) to equiv. character and append to Alpha reg.

XYZ alarm set (see page 32).

y to the x power (enter y, then x).

# **Display Features**



- 1. Display Annunciators.
- 2. Low-Power Condition.
- 3. User Keyboard Active.
- 4. Current Angular Mode.
- 5. Digit Separator and Radix Mark: Flag 28 set. CF 28 reverses them. CF 29 removes the digit separator.
- 6. Shift Set. (To cancel, press again.)
- 7. Flag(s) Set (flags 00 through 04).
- 8. Input Cue.
- 9. Program Mode or program running.
- 10. Alpha Keyboard Active.

The display message **MEMORY LOST** indicates that Continuous Memory has been cleared and reset.

The program execution indicator,  $\succ$ , appears and moves each time the program encounters a label.

# Organization of Memory Main Memory\*

#### **Data Storage Registers**



The *number of registers* currently allocated to data storage is found by executing <u>SIZE?</u>.



The number of uncommitted registers still available for use is displayed at the end of catalog 1 and after pressing GTO . in Program mode.

Whenever Continuous Memory is cleared,  $R_{00}$  through  $R_{99}$  are allocated to data storage. This distribution of registers in main memory exists until you change it by executing SIZE nnn (where nnn is the number of registers to be in data storage).

<sup>\*</sup> This diagram is simplified from the more complete one in section 12 of the owner's manual.



# **Extended Memory**

124 registers for text, data, or program files of various sizes.

The number of registers still available in extended memory is displayed by **EMROOM** and at the end of catalog 4.

# **Storing and Executing Programs**

#### To store a program in main memory:

- 1. Press **PRGM** to activate Program mode.
- 2. Press GTO . to pack memory and move to the end of program memory.
- 3. Key in a global label of up to seven Alpha characters.
- 4. Key in each subsequent instruction.
- 5. Optional: press GTO... to automatically add an END instruction and pack program memory.
- 6. Press **PRGM** to activate Execution mode.

If you make any mistakes, use  $\bullet$  to delete individual characters and entire lines.

#### To execute a program in main memory:

- 1. Make sure Execution mode is active (no **PRGM** annunciator).
- 2. Start the program by executing its global label—by Alpha execution (page 14) or by User key (page 5 in this guide). Program execution *starts* at that global label.

While the program is actually running, the **PRGM** annunciator is on. The  $\succ$  program execution indicator also appears.

Pressing  $\boxed{R/S}$  will either start the current program (from its current line) or stop a running program. If a running program stops to prompt for data, for example, you key in the data and then press  $\boxed{R/S}$  to continue the program.

To run (and re-run) the current program, you can simply press RTN[R/S].

# **Time and Alarm Formats**

# **Time Values**

The computer interprets clock time values that you specify according to the following conventions:

Setting	Clock Time		
0	Midnight		
1	1 (a.m.)		
2	2		
:	:		
10	10		
11	11		
12	Noon		
-1 or 13	1 p.m. or 13:00		
-2 or 14	2 or 14		
:	:		
-10 or 22	10 or 22		
-11 or 23	11 or 23		
0	Midnight		

#### **Time Settings**

Results of clock-time operations ([TIME], [RCLALM]) are always expressed in a 24-hour format in the X-register. Midnight is zero.

## Alarm Format

Message Alarm: sounds tones and displays a message when it goes off.

**Control Alarm:** runs the specified program or programmable catalog-2 function when the alarm comes due.

**Conditional Alarm:** does not interrupt a running program, unlike the other alarms. If the HP-41CX is off or displaying the clock, a conditional alarm becomes a control alarm. If the HP-41CX is on and **not** running a program, a conditional alarm becomes a message alarm. If a program **is** running, the alarm only beeps (twice), and then becomes past due.

To set an alarm (XYZALM), follow these steps:

- 1. Key in the repeat interval (using zero for no repetition). Press [ENTER+].
- 2. Key in the date for the alarm (using zero for today). Press ENTER1.
- 3. Key in the time for the alarm.
- 4. Press ALPHA.

For a message alarm, key in a message or clear the Alpha register. (A clear Alpha register results in an alarm "message" of the time and date.)

For a control alarm, key in  $\textcircled{\bullet} \textcircled{\bullet}$  global label or  $\textcircled{\bullet} \textcircled{\bullet}$  function name.

For a conditional alarm, key in 🛉 global label or 🛉 function name.

Press ALPHA again.

5. Execute XYZALM.



# Acknowledging and Clearing Message Alarms

- To halt a current, flashing alarm, press any key *except* <u>STO</u>. This also clears (deletes) the alarm, unless it is a repeating one. A repeating message alarm is reset.
- To halt and clear a current repeating alarm, press **C**.
- To clear an alarm that is *not* currently active, use **C** on the Alarm Catalog keyboard. (Run the catalog, stop it at the desired alarm, and press **C**.)

You do not acknowledge non-message alarms, that is, ones that run programs.

# The Catalogs

There are six catalogs (press CATALOG n) in the HP-41CX:

- Catalog 1: User Programs. A list of all global labels and END instructions with the byte count for that program, listed in the order in which they were stored. The permanent END (.END.) shows the number of unused registers in uncommitted memory (and therefore still available for programming).
- Catalog 2: External Functions + Time Functions + Extended Functions. A list of all functions and programs currently available to the computer from peripheral devices, plug-in modules, and the time, extended, and extended-memory functions. The list of functions is grouped by source (press ENTER+) to see individual functions).
- Catalog 3: Standard Functions. An alphabetical list of the standard functions.
- Catalog 4: Extended Memory Directory (EMDIR). A list of all files in extended memory. It gives the file name, file type, and the number of registers in the file. It ends with the number of registers left in extended memory.
- Catalog 5: Alarm Catalog (<u>ALMCAT</u>). A list of each alarm, in chronological order, with its time, date, and message. (See the Alarm Catalog keyboard diagram.)
- Catalog 6: User Key Assignments. A list of all User key definitions in order of keycode.

When you execute CATALOG n, the catalog listing begins. You can stop and restart it with <u>R/S</u>. With the automatic listing stopped, you can step through it forwards with <u>SST</u> and backwards with <u>BST</u>, or exit the catalog with •. In catalog 2, press <u>ENTER</u>• to see a list of those functions belonging to the displayed source device.

Most automatic catalog listings speed up when you press an undefined key. If a printer is attached, the catalogs will print out in Trace mode only.

# **Character Codes**

Code	ASCII	Display	Code	ASCII	Display
0		-	32	space	
1		¥	33	!	1
2		<b>88</b>	34	"	10
3		<b>88</b>	35	#	Ħ
4		X	36	\$	ዄ
5		X	37	%	96
6		Τ	38	&	2
7		<b>83</b>	39	,	1
8			40	(	<
9			41	)	>
10			42	*	*
11			43	+	+
12		<i>,</i> 9	44	,	,
13		á	45	-	
14			46		
15		<b>8</b>	47	/	1
16			48	0	
17			49	1	1
18			50	2	2
19		<b>8</b>	51	3	3
20		<b>#</b>	52	4	ч
21			53	5	5
22		<b>8</b>	54	6	6
23		<b>#</b>	55	7	٦
24			56	8	8
25		<b>#</b>	57	9	9
26			58	:	1.1
27			59	;	,
28		<b>8</b>	60	<	2
29		Ľ	61	=	. <u>2</u>
30			62	>	7
31		8	63	?	7

Code	ASCII	Display	Code	ASCII	Display
64	0	e	96		т
65	A	R	97	а	Cal.
66	В	В	98	b	ь
67	С	E	99	С	c
68	D	В	100	d	d
69	Е	E	101	е	٤
70	F	F	102	f	8
71	G	6	103	g	8
72	н	н	104	h	8
73	1	I	105	i	8
74	J	L L	106	j	8
75	К	ĸ	107	k	8
76	L	L	108	I I	8
77	М	11	109	m	8
78	N	N	110	n	8
79	0		111	0	8
80	Р	P	112	р	8
81	Q		113	q	8
82	R	R	114	r	8
83	S	5	115	S	
84	т	T	116	t	
85	U	L	117	u	
86	V	L'	118	v	8
87	W	14	119	w	8
88	Х	×	120	x	8
89	Y	Y	121	у	
90	Z	2	122	z	8
91	[	L	123	{	8
92	\	<u>&gt;</u>	124	1	8
93	]	7	125	}	8
94	^	1	126	~	2
95	-	-	127		F-

# The Flags and Their Status

0 =	clear.	? =	depends on other conditions.	

1		set.	M =	maintained	by	Continuous	Memory.
---	--	------	-----	------------	----	------------	---------

Flag Number	Flag Name	Status at Reset, Turn-On
00-10	User Flags You can test and alter the	0, M
11.20	Control Elogo	coc nago.
11-23	You can test and alter the	ese flags.
11	Automatic Execution	0.0
12-20	External Device Control	0,0
21	Printer Enable	2 2
22	Numeric Data Input	0,0
23	Alpha Data Input	0,0
24	Range-Error Ignore	0,0
25	Error lanore	0.0
26	Audio Enable	1, 1
27	User Keyboard	0, M
28	Radix Mark	1, M
29	Digit Separator Mark	1, M
30-55	System Flags	
	You can test but not alter	these flags.
31	Date Format	0, M
36	Number of Digits	0, M
37	"	1, M
38	"	0, M
39	"	0, M
40	Display Format	1, M
41	"	0, M
42	Grads Mode	0, M
43	Radians Mode	0, M
44	Continuous On	0, 0
48	Alpha Keyboard	0, 0
49	Low Power	?, ?
50	Message	0, 0
55	Printer Existence	?, ?

# List of Errors

Following is a simplified description of each error message. For complete descriptions of the error conditions, refer to appendix A in the owner's manual. The function that caused an error does not get executed. You can clear an error message by pressing  $\frown$ .

#### Error

#### Meaning

ALPHA DATA	Nonnumeric data used.
CHKSUM ERR	Part of file lost.
DATA ERROR	Illegal operand.
DUP FL	A file of that name already exists.
END OF FL	Pointer is at end of file.
END OF REC	Pointer is at end of record.
ERROR :: Dnn	Number not in time format.
ERROR :: Rnn	Number greater than 99.
FL NOT FOUND	Specified file does not exist.
FL SIZE ERR	Invalid file size.
FL TYPE ERR	Invalid file type.
KEYCODE ERR	Nonassignable keycode.
MEMORY LOST	Continuous Memory has been cleared and reset.
NAME ERR	Invalid file name.
NO DRIVE	The necessary device absent.
NONEXISTENT	The register, label, or function specified does not exist.
NO ROOM	Not enough room in memory.
NO SUCH ALM	Alarm does not exist.
OUT OF RANGE	Number too large.
PRIVATE	Program on card or cassette is private.

(table continued next page)

RAM	The global label specified already exists in main memory.
REC TOO LONG	Record too long.
ROM	You cannot modify a program in ROM.



**Portable Computer Division** 1000 N.E. Circle Blvd., Corvallis, OR 97330, U.S.A.

European Headquarters 150, Route Du Nant-D'Avril Geneva-Switzerland

**HP-United Kingdom** (Pinewood) P.O. Box, CH-1217 Meyrin 2 GB-Nine Mile Ride, Wokingham Berkshire RG11 3LL

© Hewlett-Packard Company 1983 00041-90475 English Printed in Singapore 8/83