HP-41C TRAINING MANUAL

HEWLETT-PACKARD CALCULATORS

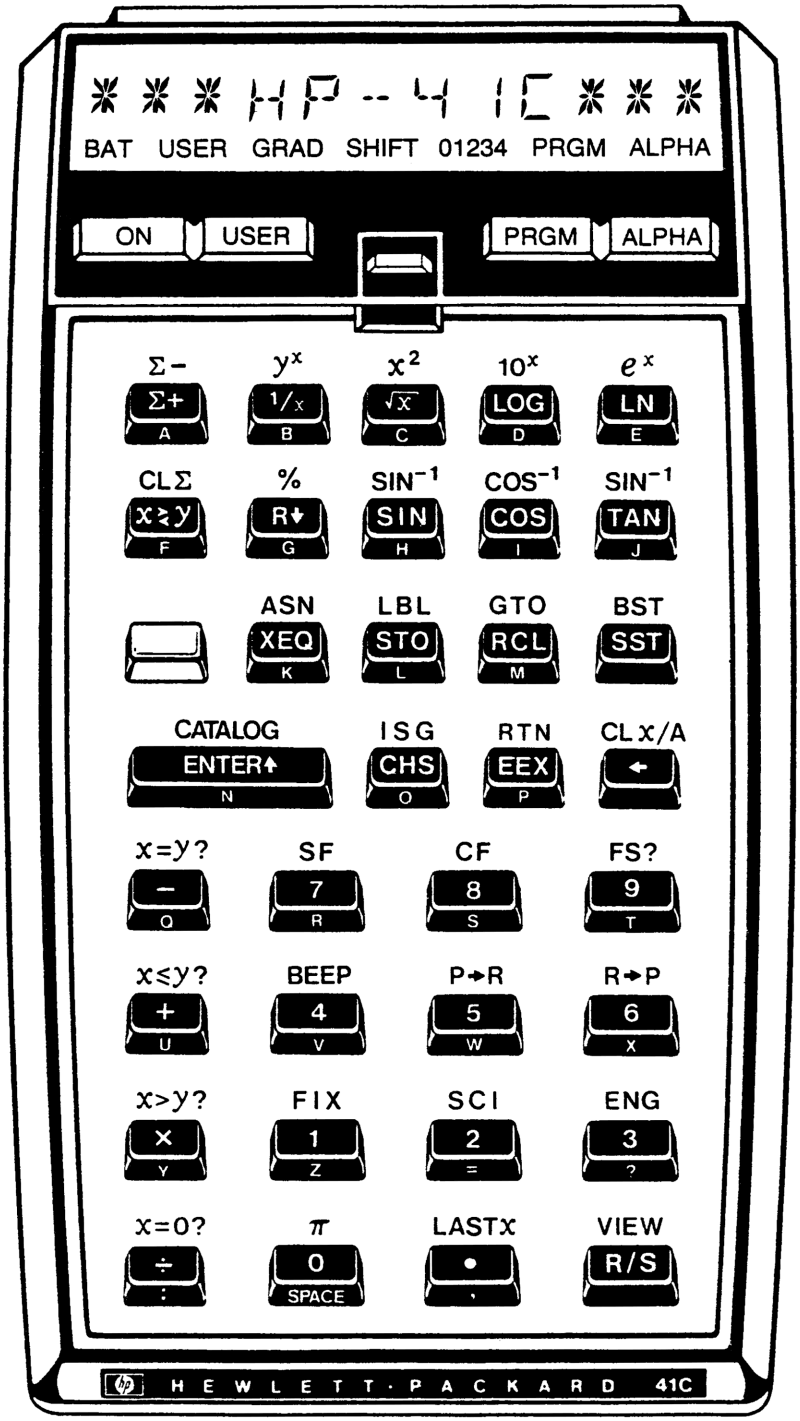

HP-41C
BAT USER GRAD SHIFT 01234 PRGM ALPHA
ON USER PRGM ALPHA
Σ- yˣ x² 10ˣ eˣ
Σ+ A 1/x B √x C LOG D LN E
CLΣ % SIN⁻¹ COS⁻¹ SIN⁻¹
x≷y F R↓ G SIN H COS I TAN J
ASN LBL GTO BST
XEQ K STO L RCL M SST
CATALOG ISG RTN CLx/A
ENTER↑ N CHS O EEX P ←
x=y? SF CF FS?
− 7 R 8 S 9 T
x≤y? BEEP P→R R→P
+ U 4 V 5 W 6 X
x>y? FIX SCI ENG
× Y 1 Z 2 = 3 ?
x=0? π LASTx VIEW
÷ : 0 SPACE • , R/S
HEWLETT·PACKARD 41C

Table of Contents - Training Package

HP-41C SYSTEM TRAINING MATERIALS

We have prepared a complete set of slides and accompanying text for your use when training dealers and/or end users on the HP-41C system. The material ranges from the most basic HP calculator operations (such as HP-RPN) to the most advanced operations of the HP-41C peripherals. This allows you the flexibility to tailor your presentations for any audience.

If you need additional training material, please contact your sales development representative.

HEWLETT-PACKARD 41C

. . . A CALCULATOR . . .

. . . A SYSTEM . . .

. . . A WHOLE NEW STANDARD . . .

INTRODUCING THE HEWLETT-PACKARD 41C

Many words can be used to describe the new HP-41C. A concise yet full description of the HP-41C is:

"A CALCULATOR, A SYSTEM, A WHOLE NEW STANDARD!"

HEWLETT-PACKARD 41C

. . . A CALCULATOR . . .

COMMUNICATES WITH THE USER
CUSTOMIZES TO YOUR OWN DESIGN
HP-RPN - PROGRAMMABLE
CONTINUOUS MEMORY

. . . A SYSTEM . . .

UP TO FIVE TIMES MORE MEMORY
EXTRA SMART CARD READER
WHISPER QUIET PRINTER
DIGITAL BAR CODE WAND
EXTENSIVE SOFTWARE SUPPORT

. . . A WHOLE NEW STANDARD . . .

POWER
FLEXIBILITY
EXPANDABILITY
EASE OF USE

A CALCULATOR, A SYSTEM, A WHOLE NEW STANDARD

Several associated major benefits are highlighted below each segment of the summary description. (See Slide)

HEWLETT-PACKARD 41C

- THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . . IT COMMUNICATES WITH THE USER.
- THE HP-41C HAS AN EXCITING NEW KEYBOARD DESIGN . . . YOU CAN CUSTOMIZE THE CALCULATOR TO YOUR OWN DESIGN.
- THE HP-41C HAS A UNIQUE COMBINATION OF PROGRAMMING EASE AND SOPHISTICATION . . . IT IS EASY TO PROGRAM.
- THE HP-41C HAS PROGRAMMING POWER . . . IT CAN SOLVE PROFESSIONAL PROBLEMS.
- THE HP-41C HAS CONTINUOUS MEMORY . . . IT SAVES DATA, PROGRAMS AND KEY ASSIGNMENTS EVEN WHEN TURNED OFF.
- THE HP-41C IS AN EXPANDABLE SYSTEM . . . IT MEETS YOUR NEEDS TODAY AND IN THE FUTURE.

TRAINING OVERVIEW

As we learn about the HP-41C, we'll focus our attention on the following key HP-41C benefits, and the HP-41C features that provide them to the user. (See Slide)

HEWLETT-PACKARD 41C

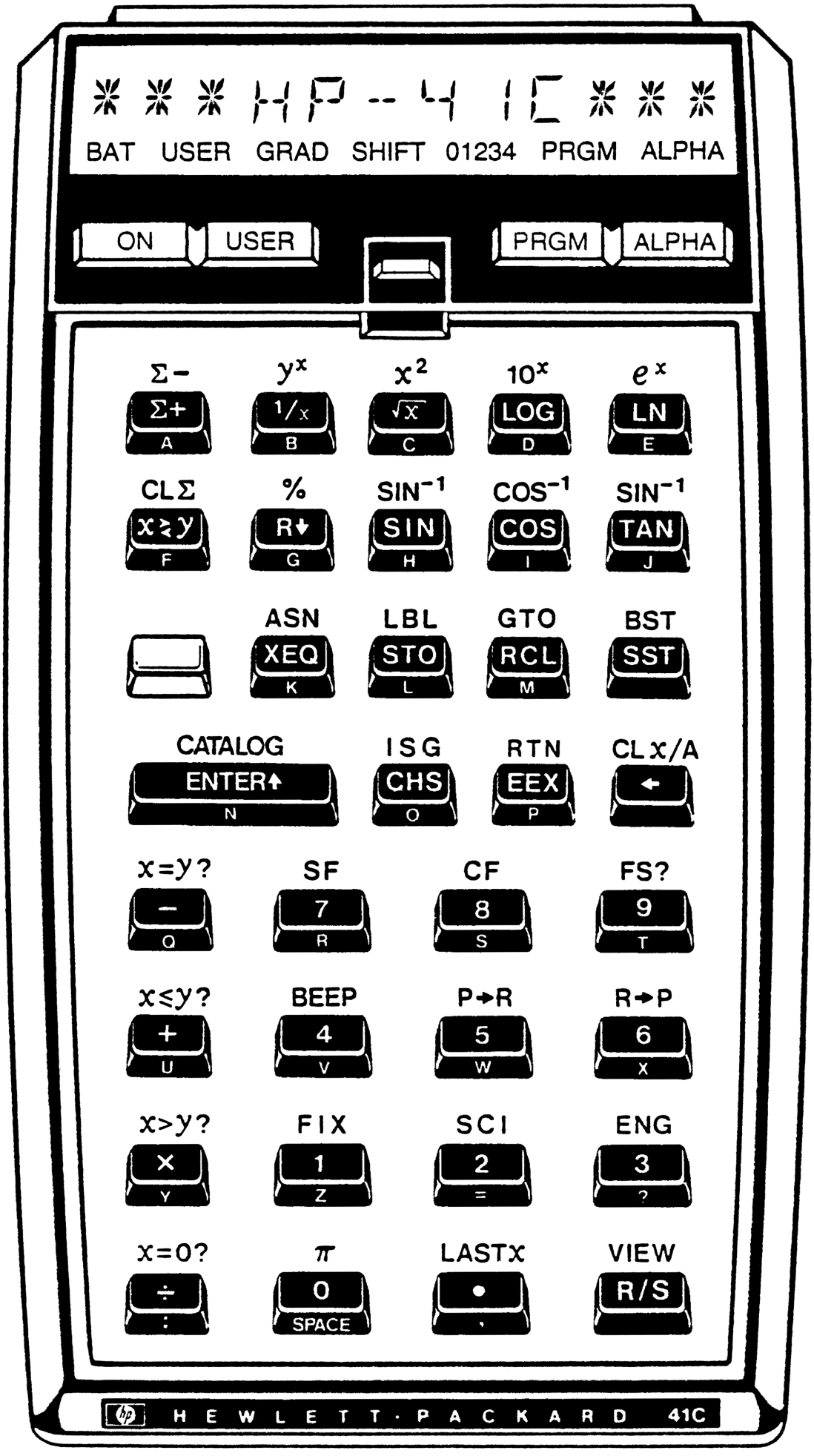
THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . . IT COMMUNICATES WITH THE USER

ON/OFF TOGGLE SWITCH
LIQUID CRYSTAL DISPLAY (LCD)
ANNUNCIATORS
DIGIT ENTRY
ALPHA CHARACTER ENTRY
ERROR CORRECTION
HP-RPN
SIMPLE CALCULATIONS
CHAIN CALCULATIONS
ADVANCED FUNCTIONS
AUTOMATIC MEMORY STACK
DATA STORAGE AND RECALL
DEFINING STORAGE REGISTER CONFIGURATIONS
DISPLAY FORMAT CONTROL
AUDIBLE BEEPER
LONG BATTERY LIFE

SIMPLE TO USE CALCULATOR

The HP-41C is a very simple to use calculator. It communicates with the user.

The first portion of training will center around the following key HP-41C features.


*** HP-41C ***
BAT USER GRAD SHIFT 01234 PRGM ALPHA
ON
USER
PRGM
ALPHA
Σ- y^x x^2 10^x e^x
Σ+ 1/x √x LOG LN
A B C D E
CLΣ % SIN^-1 COS^-1 SIN^-1
x≷y R↓ SIN COS TAN
F G H I J
ASN LBL GTO BST
XEQ STO RCL SST
K L M
CATALOG ISG RTN CLx/A
ENTER↑ CHS EEX ←
N O P
x=y? SF CF FS?
− 7 8 9
Q R S T
x≤y? BEEP P→R R→P
+ 4 5 6
U V W X
x>y? FIX SCI ENG
× 1 2 3
Y Z = ?
x=0? π LASTx VIEW
÷ 0 • R/S
: SPACE ,
HEWLETT·PACKARD 41C

ON/OFF TOGGLE SWITCH

The HP-41C [ON] key is an easy to use On/Off toggle switch. Push it once and the calculator turns ON. Push it again, and the calculator turns OFF.

LIQUID CRYSTAL DISPLAY

The HP-41C has an all new, easy to read Liquid Crystal Display (LCD). It offers better viewing in sunlight, and it helps to conserve battery power.

ANNUNCIATORS

An exciting new feature offered by the HP-41C is a set of annunciators that let you know the status of your machine at all times. (Activate several annunciators, and then turn them all off.)

ANNUNCIATOR	IF DISPLAYED	KEYSTROKES TO ACTIVATE
BATT	Low Batteries	AUTOMATIC
USER	User Mode	[USER]
RAD	Radians Mode	[XEQ] [ALPHA] RAD [ALPHA]
GRAD	Gradians Mode	[XEQ] [ALPHA] GRAD [ALPHA]
SHIFT	Shift Active	[GOLD]
0 - 4	Flag Set	[GOLD] SF 01-04
PRGM	PRGM Mode	[PRGM]
ALPHA	ALPHA Mode	[ALPHA]

NUMBER ENTRY

KEYSTROKES	DISPLAY
[ON] [←]	MEMORY LOST
123	123_
4567890	1,234,567,890
[ENTER]	1,234,567,890.
[GOLD] [CLX]	0.0000

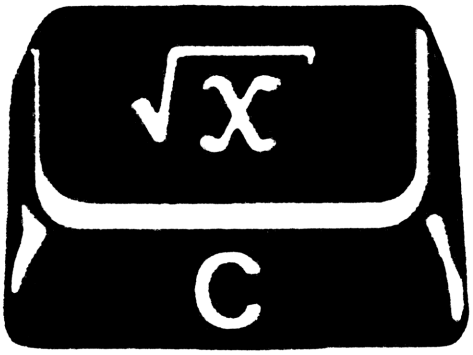
NUMBER ENTRY

One of the most basic functions of calculator operation is NUMBER ENTRY. Before practicing simple Number Entry we'll first standardize our calculators by erasing all memory.
(Turn on calculator while holding down backarrow key.)

While practicing Number Entry, note two important HP-41C communication features, the PROMPT UNDERLINE and AUTOMATIC THOUSANDS SEPARATION.

While in Normal mode, the HP-41C can display up to 10 digits.

NORMAL KEYBOARD



FOR X^2
PRESS [GOLD]
THEN [$\sqrt{X}$]

FOR $\sqrt{X}$
PRESS [$\sqrt{X}$]

ALPHA CHARACTER C
CANNOT BE ACTIVATED
WHILE IN NORMAL MODE

NORMAL KEYBOARD

As with other advanced calculators, each key on the HP-41C serves multiple purposes.

In NORMAL MODE, the function on the face of the key is activated solely by pressing the key. The Gold function above the key is activated by first pressing the GOLD SHIFT KEY, and then the key below the Gold function. The character or function printed in Blue on the lower face of the key is not active when the calculator is in Normal Mode.

If you hold a function key down momentarily, you'll notice that the HP-41C prompts you by identifying the function about to be performed.

If you hold it down for over a second, it will say "NULL", and the function will not be executed. We'll learn more about the value of key prompting later.

NORMAL KEYBOARD PRACTICE

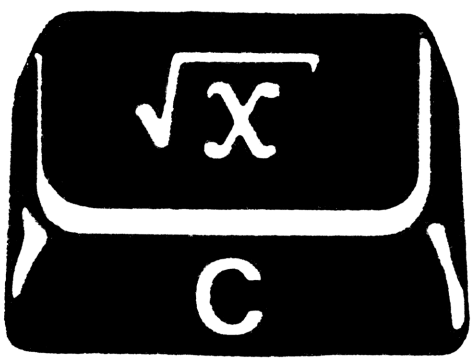
KEYSTROKES	DISPLAY
49	49_
[$\sqrt{X}$]	7.0000
[GOLD] [X^2]	49.0000
5	5_
[LOG]	0.6990
[GOLD] [10^X]	5.0000
[GOLD] [CLX]	0.0000

NORMAL KEYBOARD PRACTICE

To reinforce the previous two slides, practice using squares and square roots.

With HP-RPN, you do not have to ENTER the number before performing an operation.

ALPHA KEYBOARD



x^2 FUNCTION IS NOT ACTIVE WHILE IN ALPHA MODE

$\sqrt{X}$ FUNCTION IS NO LONGER ACTIVE WHILE IN ALPHA MODE. FOR SPECIAL SHIFTED CHARACTER (LOWER CASE C IN THIS EXAMPLE) FIRST PRESS [GOLD] THEN [$\sqrt{X}$]

FOR ALPHA CHARACTER C PRESS [$\sqrt{X}$]

ALPHA KEYBOARD

An exciting new feature offered by the HP-41C is the ALPHA Keyboard. The Alpha keyboard is activated by first pressing the toggle switch designated "ALPHA".

The primary function of each key while the calculator is in Alpha mode is the Alpha character or special symbol printed in Blue on the lower face of each key. The Gold function above the key is no longer active. Instead, a special shifted Alpha character is now associated with the key. To activate this special shifted Alpha character, first press the [Gold] shift key, and then the function key. To identify the shifted Alpha character associated with each key, refer to the keyboard located on the back of your calculator.

ALPHA KEYBOARD PRACTICE

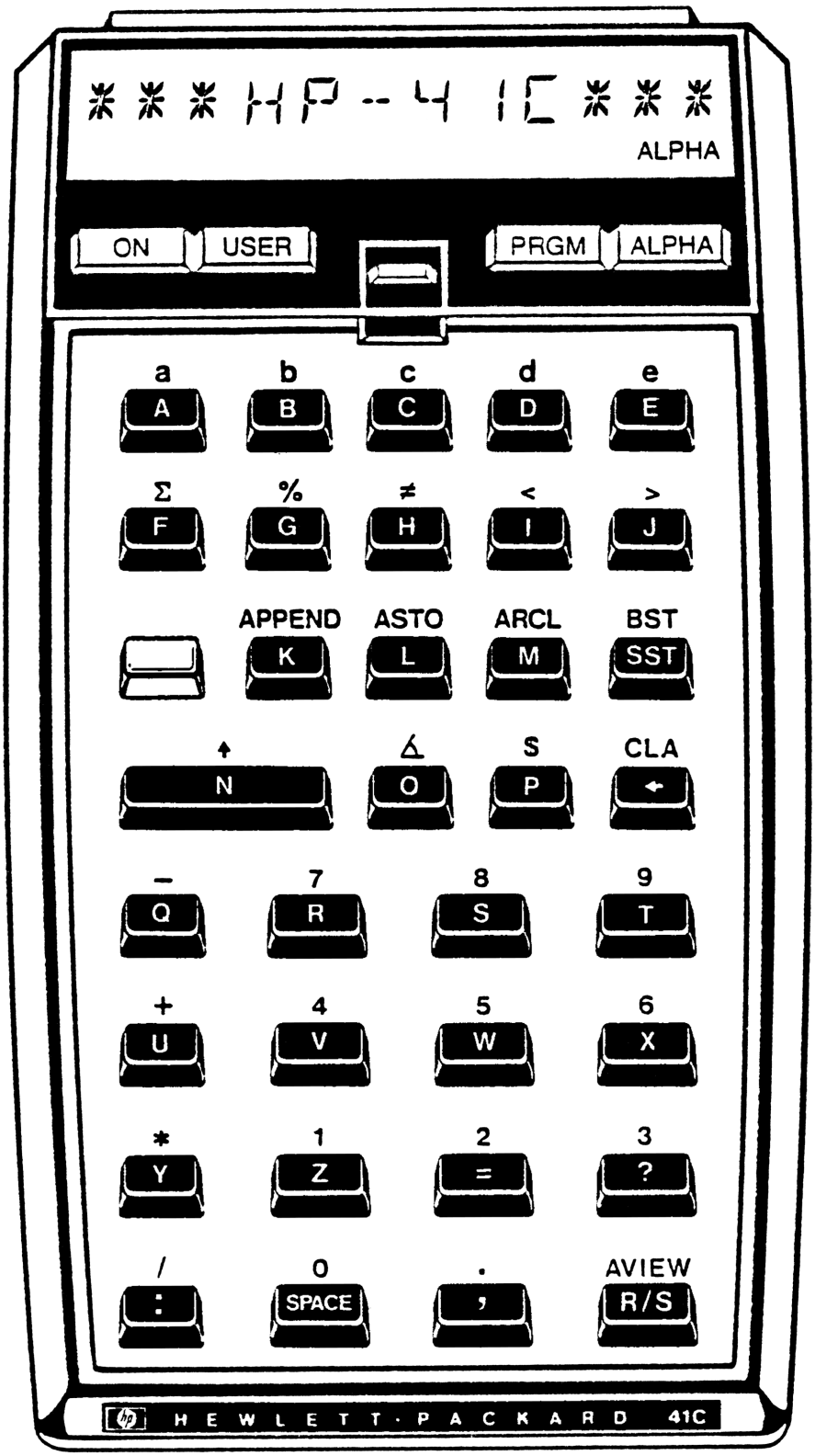
KEYSTROKES	DISPLAY
[ALPHA]	
C	C_
[GOLD] [CLA]	
ABC	ABC_
[GOLD] [CLA]	
NAME	NAME_
[GOLD] [CLA]	
ABCDEFGHIJK	ABCDEFGHIJK_
L	BCDEFGHIJKL_
M	CDEFGHIJKLM_
NOPQRSTUVWX	NOPQRSTUVWX_
[ALPHA]	0.0000
[ALPHA]	ABCDEFGHIJK SCROLLS MNOPQRSTUVWX
[GOLD] [CLA]	

ALPHA CHARACTER ENTRY PRACTICE

To familiarize ourselves with the Alpha capability of the HP-41C, let's first key in a simple example, and then our names.

The HP-41C can display up to 12 characters at one time, although it can retain up to 24 characters. When more than 11 characters are entered, the HP-41C will begin scrolling characters to the left. When 24 characters have been entered, the HP-41C will alert the user via a beep.

If a mistake is made, use [Gold] CLA. Soon, we'll learn how to overcome mistakes with a special error correction feature.


HP-41C
ALPHA
ON
USER
PRGM
ALPHA
a
A
b
B
c
C
d
D
e
E
Σ
F
%
G
≠
H
<
I
>
J
APPEND
K
ASTO
L
ARCL
M
BST
SST
↑
N
Δ
O
S
P
CLA
←
−
Q
7
R
8
S
9
T
+
U
4
V
5
W
6
X
*
Y
1
Z
2
=
3
?
/
:
0
SPACE
.
,
AVIEW
R/S
HEWLETT·PACKARD 41C

SHIFTED ALPHA CHARACTERS

SHIFTED ALPHA CHARACTERS are accessed by first pressing the shift key while the HP-41C is in Alpha mode. The Shifted Alpha Characters assigned to each key are shown on this slide, as well as on a keyboard reference permanently attached to the back of your calculator.

ALPHANUMERIC STRINGS

KEYSTROKES	DISPLAY
SPACE	_
[GOLD] Y (*)	*_
[GOLD] Y (*)	**_
H	**H_
P	**HP_
[GOLD] [-]	**HP-_
[GOLD] 4	**HP-4_
[GOLD] 1	**HP-41_
C	**HP-41C_
[GOLD] Y (*)	**HP-41C*_
[GOLD] Y (*)	**HP-41C**_
[GOLD] [CLA]	

ALPHANUMERIC STRINGS

To this point, we have seen how numeric information is entered, and how Alpha characters are entered. Now, in order to truly appreciate the communicative ability of the HP-41C, let's examine its Alphanumeric capability, or the ability to integrate Alpha characters and numeric characters.

As explained earlier, each key has a special shifted character associated with it, while in Alpha mode. We will access some of these shifted characters as we create an ALPHANUMERIC STRING.

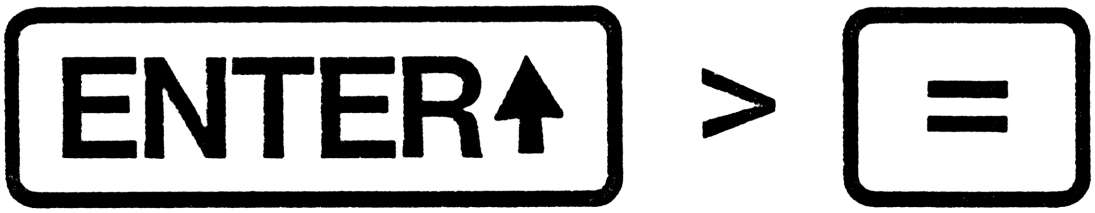
ERROR CORRECTION

KEYSTROKES	DISPLAY
HYDVO	HYDVO_
[←]	HYDV_
[←]	HYD_
RO	HYDRO_
[GOLD] [CLA]	
[ALPHA]	0.0000
12355	12,355_
[←]	1,235_
[←]	123_
45	12,345_
[GOLD] [CLX]	0.0000

ERROR CORRECTION

One of the exciting new features offered by the HP-41C is the ERROR CORRECTION key, [←]. No longer is it necessary to clear an entire entry when a mistake is made. In either Normal or Alpha mode, each press of the [←] key deletes one right-most character. Notice how the " __ " (underscore) prompt moves back as you correct your mistake. Repeated pushes on the [←] will remove more characters.

HEWLETT-PACKARD RPN LOGIC



* CONSISTENT APPROACH TO NUMBER AND FUNCTION ENTRY

* MEANINGFUL INTERMEDIATE AND FINAL DISPLAY FEEDBACK

* KEYSTROKE EFFICIENCY - NO PARENTHESIS

* ERROR CORRECTION WITHOUT STARTING OVER

HP - RPN

The HP-41C employs Hewlett-Packard's powerful yet easy to use RPN logic system. Simply stated, HP-RPN offers the following benefits:

1. CONSISTENCY - The operation ALWAYS follows the argument (or arguments) to be operated on. ALWAYS. This is similar to the way one works with pencil & paper.

2. MEANINGFUL DISPLAY FEEDBACK - With the HP-RPN system, the display ALWAYS shows the results of the function just performed. This allows a user to check for reasonableness all the way through a calculation. The HP-41C ALWAYS executes a function the instant it is pressed. It does not require storing operations for later execution (pending operations).

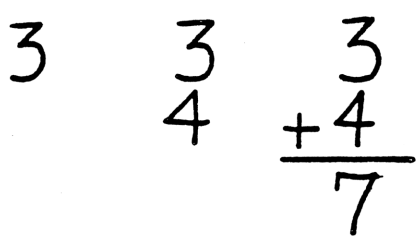
3. KEYSTROKE EFFICIENCY - The HP-RPN system does not require the entry of parenthesis into the calculator, thus resulting in fewer keystrokes. This feature of HP-RPN is also important for the way in which it conserves program memory space when programming the HP-41C.

4. ERROR CORRECTION - Using the LAST X feature, and the power to manipulate an automatic memory stack, HP-RPN gives the user the ability to "Back Up" or even reverse an operation performed by mistake. This eliminates the need to "Start Over" after making a mistake in the middle of a lengthy problem.

SIMPLE CALCULATIONS

3 + 4

WITH PAPER AND PENCIL



WITH THE HP-41C USING HP-RPN

KEYSTROKE	DISPLAY
3	3_
[ENTER]	3.0000
4	4_
[+]	7.0000

SIMPLE CALCULATIONS

Solving problems with HP-RPN is very similar to solving problems as you would with pencil and paper. We can note the similarity by working a simple problem.

With HP-RPN, the only new element is the [ENTER] key. The [ENTER] key is used simply to separate two numbers, or two intermediate calculations, as we'll see next.

CHAIN CALCULATIONS

16 + 30 - 11 + 17 - 14

KEYSTROKE	DISPLAY	ANSWER EQUALS
[GOLD] [CLX]	0.0000	
16	16_	
[ENTER]	16.0000	
30	30	
[+]	46.0000	(16 + 30)
11	11_	
[-]	35.0000	(16 + 30 - 11)
17	17_	
[+]	52.0000	(16 + 30 - 11 + 17)
14	14_	
[-]	38.0000	(16 + 30 - 11 + 17 - 14)
[GOLD] [CLX]	0.0000	

CHAIN CALCULATIONS

You only have to press [ENTER] when putting two numbers into the HP-41C, one after another. If an operation is to be performed after keying in a number, it is not necessary to "ENTER" the number.

COMPLEX CALCULATIONS

$$5\left[\frac{(3\div4) + (4X3)}{(3 X .213}\right]$$

KEYSTROKE	DISPLAY
3	3_
[ENTER]	3.0000
4	4_
[÷]	.7500
4	4_
[ENTER]	4.0000
3	3_
[X]	12.0000
[+]	12.7500
3	3_
[ENTER]	3.0000
.213	.213_
[X]	.6390
[÷]	19.9531
5	5_
[X]	99.7653 FINAL ANSWER
[GOLD] [CLX]	0.0000

COMPLEX CALCULATIONS

Even complex calculations are easily solved using HP-RPN. As we work with more complex problems, you can begin to appreciate the benefits of HP-RPN.

- CONSISTENT APPROACH TO NUMBER ENTRY
- MEANINGFUL INTERMEDIATE AND FINAL DISPLAY FEEDBACK
- KEYSTROKE EFFICIENCY - NO PARENTHESIS
- ERROR CORRECTION WITHOUT STARTING OVER

ADVANCED FUNCTION EXECUTION

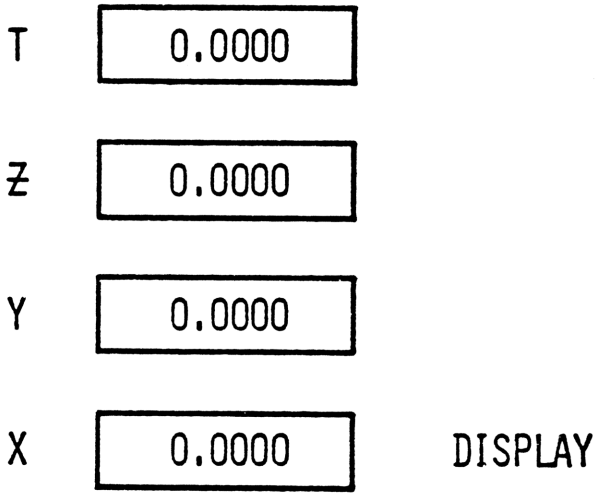
PROBLEM	KEYSTROKES	DISPLAY
$\sqrt{5}$	5	5_
	[$\sqrt{X}$]	2.2361
1/20	20	20_
	[1/X]	0.0500
$\sqrt{\pi}$	[GOLD] [π]	3.1416
	[$\sqrt{X}$]	1.7725
2^2	2	2_
	[GOLD] [X^2]	4.0000
SIN 35^o	35	35_
	[SIN]	.5736
6^3	6	6_
	[ENTER]	6.0000
	3	3_
	[GOLD] [Y^X]	216.0000
	[GOLD] [CLX]	0.0000

ADVANCED FUNCTION EXECUTION

More ADVANCED FUNCTIONS are executed using the same consistent approach.

First key in the argument, then key in the operation. The function (operation) is performed immediately, and the answer appears in the display.

AUTOMATIC MEMORY STACK



AUTOMATIC MEMORY STACK

We have seen that the HP-41C provides an automatic mechanism for temporarily storing arguments and intermediate results, and then making them accessable at the appropriate point in a calculation sequence. This automatic mechanism is called the MEMORY STACK.

The Automatic Memory Stack consists of four storage registers logically arranged to provide a last in, first out storage and retrieval of arguments during calculations.

Each storage register has a letter name. (See Example)

T	0.0000	
Z	0.0000	
Y	0.0000	
X	0.0000	Display

It is important to note that the X register is the display register. This means that the number you see in the calculator's display is always the number you see in the X register.

EXPLORING THE MEMORY STACK

3 + 4

T	0	0	0	0
Z	0	0	0	0
Y	0	3	3	0
X	3	3	4	7

KEYSTROKES 3 ENTER 4 +

EXPLORING THE MEMORY STACK

We can follow the memory stack's automatic mechanisms by again working our simple addition problem. Note how numbers roll up the stack as we enter a new number and roll down the stack as we perform an operation.

$$5\left[\frac{(3\div4)+(4\times3)}{(3\times.2)}\right]$$

T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z	0	0	0	0	0	.75	.75	0	0	0	12.75	12.75	0	0	0	0
Y	0	3	3	0	.75	4	4	.75	0	12.75	3	3	12.75	0	21.25	0
X	3	3	4	.75	4	4	3	12	12.75	3	3	.2	.6	21.25	5	106.25

KEYSTROKE 3 ENTER 4 ÷ 4 ENTER 3 X + 3 ENTER .2 X ÷ 5 X

ANOTHER EXAMPLE OF

AUTOMATIC MEMORY STACK OPERATION

Again, using a problem encountered before, we can follow our intermediate and final calculations through the AUTOMATIC MEMORY STACK.

STACK MANIPULATION

T	0	0	0	0	0	1	1	4	4	4	4	4	4	4	4	4
Z	0	0	0	1	1	2	2	1	1	1	1	4	4	4	4	4
Y	0	1	1	2	2	3	3	2	3	2	2	1	4	4	4	11
X	1	1	2	2	3	3	4	3	2	3	0	2	3	7	11	7

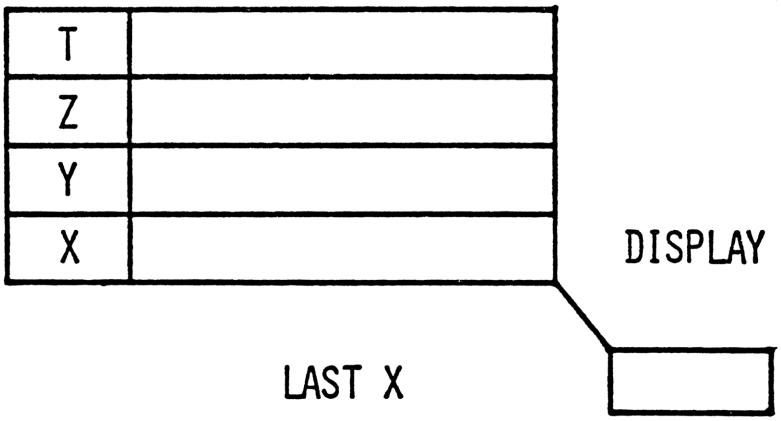
KEYSTROKE 1 ENTER 2 ENTER 3 ENTER 4 R↓ $X \gtrless Y$ $X \gtrless Y$ CLX + + + + LAST X

LAST X	0	0	0	0	0	0	0	0	0	0	0	0	2	3	7	7

STACK MANIPULATION SLIDE

Note to Trainer: This slide is provided for your use if you desire to develop additional stack manipulation examples.

LAST X REGISTER



KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
3	3_
[ENTER]	3.0000
4	4_
[+]	7.0000
[GOLD] [LAST X]	4.0000
[GOLD] [CLX]	0.0000

LAST X REGISTER

There are two added features associated with the automatic memory stack, the LAST X REGISTER and the stack manipulation functions. Both contribute to HP-RPN's problem solving flexibility and both are especially helpful in error recovery.

Think of the LAST X register as being separate from the automatic memory stack. The LAST X register preserves the value that was last in the display (X register) before the execution of a function. To place the contents of the the LAST X register into the display register (X register) press [Gold] LAST X.

ERROR RECOVERY WITH LAST X

SOLVE 3 + 4 - 6

T	0	0	0	0	0	0	0	0	0	0
Z	0	0	0	0	0	0	0	0	0	0
Y	0	3	3	0	7	0	2	0	7	0
X	3	3	4	7	5	2	5	7	6	1

KEYSTROKE	3	ENTER	4	+	5	-	LAST X	+	6	-

LAST X	0	0	0	4	4	5	5	5	5	6

ERROR RECOVERY WITH LAST X

We've seen how one of the HP-41C's new features, the ERROR CORRECTION KEY [←], allows us to correct mistakes made while keying in digits. Using the LAST X feature, we can also recover from incorrect digit entry mistakes even after having performed an operation.

STACK MANIPULATION

T																			
Z																			
Y																			
X																			

KEYSTROKES

LAST X																			

STACK MANIPULATION FUNCTIONS

The contents of the automatic memory stack can be manipulated between registers in several ways.

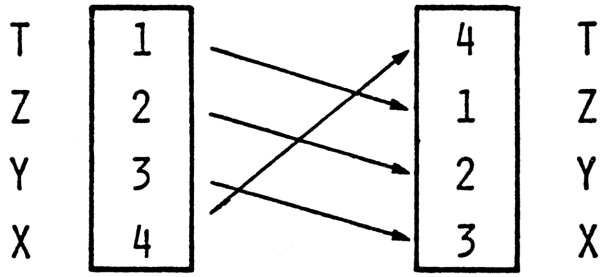
The traditional RPN stack manipulations are ROLL DOWN, ROLL UP, X-Y EXCHANGE, and CLEAR X.

A new stack manipulation feature offered by the HP-41C is the ability to exchange data between the X register and any other register, including the LAST X register.

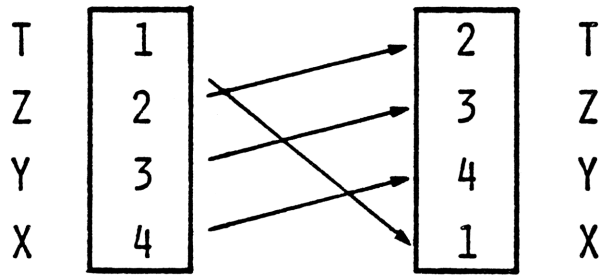
The ability to manipulate stack contents offers user flexibility in all calculator modes.

STACK MANIPULATION

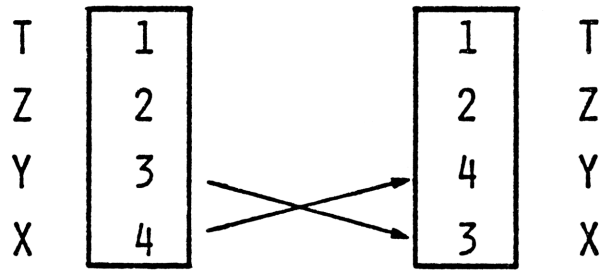
ROLL DOWN



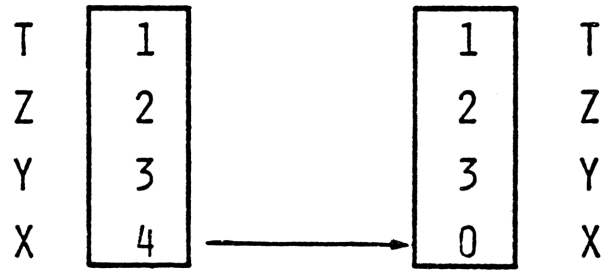
ROLL UP



X≶Y EXCHANGE



CLEAR X



STACK MANIPULATION EXERCISES

Practice STACK MANIPULATION and watch the changes in the display (X) register.

Note also that, as the stack folds down after performing an operation, the number in the T register automatically "fills in" below.

ALPHA REGISTER

DISPLAY | ABC | ALPHA REGISTER

KEYSTROKES	DISPLAY
1	1_
[ENTER]	1.0000
2	2_
[ENTER]	2.0000
3	3_
[ENTER]	3.0000
4	4_
[ALPHA]	
ABC	ABC_
[ALPHA]	4.0000
[ALPHA]	ABC
[ALPHA]	4.0000
[R↓]	3.0000
[R↓]	2.0000
[R↓]	1.0000

ALPHA REGISTER

We have seen how you can execute a function and how the result is placed in the X register and appears in the display.

If you are in ALPHA mode, any characters you key in are placed into a special Alpha register as well as the display. The Alpha register, like the LAST X register, is separate from the automatic memory stack. The automatic memory stack is not disturbed when you key in Alpha characters.

To see what is in the Alpha register, simply place the HP-41C into ALPHA mode. In ALPHA mode, the Alpha register is always displayed.

APPEND FUNCTION

KEYSTROKES	DISPLAY
[ALPHA]	ABC
[GOLD] [CLA]	
ADD	ADD_
[ALPHA]	1.0000
[ALPHA]	ADD
[GOLD] [K] (APPEND)	ADD_
ITION	ADDITION_
[ALPHA]	1.0000
[ALPHA]	ADDITION
RUN	RUN_
[GOLD] [CLA]	
[ALPHA]	1.0000
[GOLD] [CLX]	0.0000

APPEND FUNCTION

The APPEND FUNCTION enables you to build on to a string in the Alpha register. You can add characters to a string already in the Alpha register by placing the HP-41C into Alpha Mode, executing the Append Function, and then keying in the desired additional characters.

If you don't execute the Append Function before adding new characters, the new characters will clear the previous string from the Alpha register.

DATA STORAGE AND RECALL

STO 01 - STORES THE VALUE IN THE DISPLAY IN REGISTER 01

RCL 01 - RECALLS THE VALUE IN REGISTER 01 INTO THE DISPLAY

KEYSTROKES	DISPLAY
.005	.005_
[STO] 02	0.0050
175	175_
[STO] 10	175.0000
[GOLD] [CLX]	0.0000
[RCL] 02	0.0050
[RCL] 10	175.0000

DATA STORAGE AND RECALL

The HP-41C DATA STORAGE registers allow you to manually store and recall numbers and Alpha strings for use in later calculations. These registers are independent of the automatic memory stack and the LAST X register.

Note that the HP-41C prompts for a two digit Storage Register number.

STORING AND RECALLING STACK REGISTER DATA

[STO] [.] Z - STORES THE VALUE IN THE DISPLAY IN THE Z STACK REGISTER

[RCL] [.] Z - RECALLS THE VALUE IN THE Z STACK REGISTER INTO THE DISPLAY

KEYSTROKES	DISPLAY
0	0_
[ENTER]	0.0000
[ENTER]	0.0000
[ENTER]	0.0000
100	100_
[STO] [.] Z	100.0000
[GOLD] [CLX]	0.0000
[RCL] [.] Z	100.0000
[GOLD] [CLX]	0.0000

STORING AND RECALLING STACK REGISTER DATA

The HP-41C also allows you to STORE AND RECALL DATA in the automatic memory stack registers and the LAST X register.

This is a new feature, and it adds a new level of flexibility to programming.

ALPHA STRING STORE AND RECALL

KEYSTROKES	DISPLAY
[ALPHA]	
[GOLD] [CLA]	
ABC	ABC_
[GOLD] L (ASTO)	ASTO _ _
01	ABC
DEF	DEF_
[GOLD] L (ASTO) [.] Z	DEF
[GOLD] [CLA]	
[GOLD] M (ARCL) 01	ABC
[GOLD] M (ARCL) [.] Z	ABCDEF
[GOLD] L (ASTO) 02	ABCDEF
[GOLD] CLA	
[ALPHA]	0.0000
[RCL] 01	ABC
[GOLD] [CLX]	0.0000
[RCL] 02	ABCDEF
[GOLD] [CLX]	0.0000

ALPHA STRING STORE AND RECALL

While in Alpha Mode, ALPHA STRINGS can be STORED AND RECALLED using the shifted Alpha functions of Alpha Store and Alpha Recall.

Alpha data can also be recalled from stack registers and storage registers while in normal mode, using the normal recall [RCL] function.

[ARCL] always adds the recalled strings to whatever is already in the Alpha register. Later, we'll see how this feature is very useful when labeling data output.

STORAGE REGISTER ARITHMETIC

[STO] + 01 NUMBER IN DISPLAY IS ADDED TO VALUE IN REGISTER 01 AND SUM IS PLACED INTO REGISTER 01

[STO] ÷ 03 NUMBER IN REGISTER 03 IS DIVIDED BY VALUE IN DISPLAY AND QUOTIENT IS PLACED INTO REGISTER 03

KEYSTROKES	DISPLAY
100	100
[STO] 01	100.0000
25	25_
[STO] 03	25.0000
25	25_
[STO]	STO _ _
[+]	STO +_ _
01	25.0000
10	10_
[STO]	STO_
[÷]	STO÷_
03	10.0000
[RCL] 01	125.0000
[RCL] 03	2.5000
[GOLD] [CLX]	0.0000

STORAGE REGISTER ARITHMETIC

ARITHMETIC can be performed upon the contents of ALL storage registers by executing [STO] followed by the register address. Notice the user prompting during this process.

DEFINING STORAGE REGISTER CONFIGURATIONS

- TURN-ON CONFIGURATION
  17 DATA STORAGE REGISTERS
  46 REGISTERS OF PROGRAM MEMORY (322 LINES)
  63 REGISTERS

- 1 DATA REGISTER = 7 PROGRAM LINES

- TO REALLOCATE MEMORY

KEYSTROKES	DISPLAY
[XEQ]	XEQ_
[ALPHA] SIZE [ALPHA]	SIZE _ _ _
080	PACKING
	TRY AGAIN
[XEQ]	XEQ_
[ALPHA] SIZE [ALPHA]	SIZE _ _ _
010	0.0000
↓	
CONFIGURATION IS NOW	10 DATA STORAGE REGISTERS
	53 REGISTERS OF PROGRAM MEMORY (371 LINES)
	63 REGISTERS

DEFINING STORAGE REGISTER CONFIGURATIONS

The HP-41C comes standard with 63 storage registers. Memory modules can be added to increase the number of registers to a total of 319.

In the HP-41C, program memory also uses storage registers for the storage of program instructions. One of the exciting features of the HP-41C is its ability to allocate registers between data storage and program memory. The [SIZE] function allows you to specify the number of data storage registers you wish to have allocated. Every register allocated to program memory provides the user with seven additional lines of program memory.

If you attempt to increase the allocation of storage registers and there is not enough unused space in program memory for this increase, the HP-41C will display PACKING and then TRY AGAIN. After you execute Size again, if the HP-41C again displays PACKING and TRY AGAIN, this means that the reallocation is not possible until program instructions are deleted from program memory.

If you decrease the allocation of data storage registers, any information in reallocated data storage registers will be lost.

The HP-41C turns on with a standard allocation of 17 storage registers and 46 registers of program memory. Each storage register can convert to 7 program lines.

Since we'll need more program memory than the standard allocation to complete our training, use [SIZE] to allocate an additional 7 storage registers to program memory. (Execute [SIZE] 010, leaving 53 registers of program memory.)

DISPLAY FORMAT CONTROL

FIXED POINT

KEYSTROKE	DISPLAY
[GOLD] FIX 2	0.00
[GOLD] FIX 9	0.000000000
[GOLD] FIX 4	0.0000

SCIENTIFIC NOTATION

KEYSTROKE	DISPLAY	
150	150_	
[ENTER]	150.0000	
[GOLD] [SCI] 6	1.500000	02
[GOLD] [SCI] 4	1.5000	02
100	100_	
[X]	1.5000	04

ENGINEERING NOTATION

KEYSTROKE	DISPLAY	
[GOLD] [ENG] 4	15.000	03
[GOLD] [ENG] 2	15.0	03
100	100_	
[X]	1.50	06
[GOLD] [CLX]	0.00	00
[GOLD] [FIX] 4	0.0000	

DISPLAY FORMAT CONTROL

As seen earlier, the HP-41C can display up to 10 digits at one time. The user also has his choice of three display modes.

FIXED - Using Fixed Point display, you can specify the number of places to be shown after the decimal point.

SCIENTIFIC - With the HP-41C in Scientific Notation mode, each number is displayed with a single digit to the left of the decimal point. This number is followed by a user specified number of digits (up to 7) to the right of the decimal point and is followed by the appropriate exponent of 10.

ENGINEERING - Engineering Notation is similar to scientific notation except that Engineering Notation shows all exponents in multiples of 3.
(e.g., 10^{3}, 10^{-6}, 10^{12}).

AUDIBLE BEEPER

KEYSTROKE	DISPLAY
[GOLD] [BEEP]	0.0000
[GOLD] [BEEP]	0.0000
[GOLD] [BEEP]	0.0000

AUDIBLE BEEPER

Another exciting new feature of the HP-41C is the AUDIBLE BEEPER. The Beeper is particularly useful while running programs.

1. It can prompt you for data.

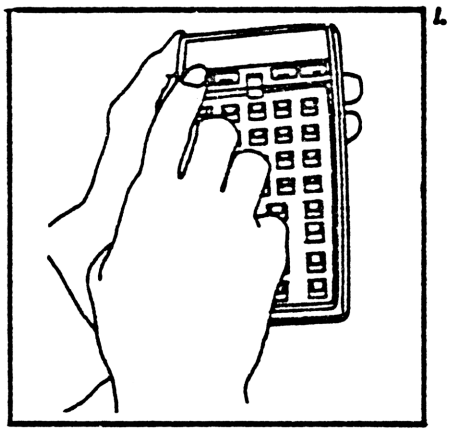
2. It can signal that an intermediate result has been obtained.

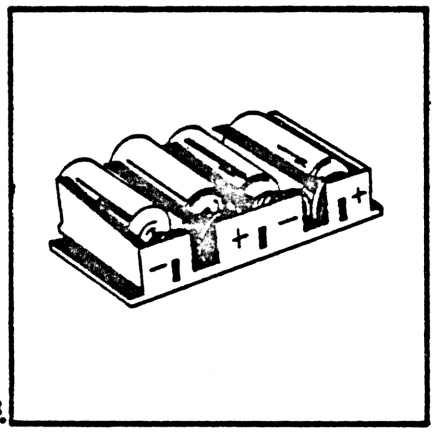
3. It can signal that a program has finished running.

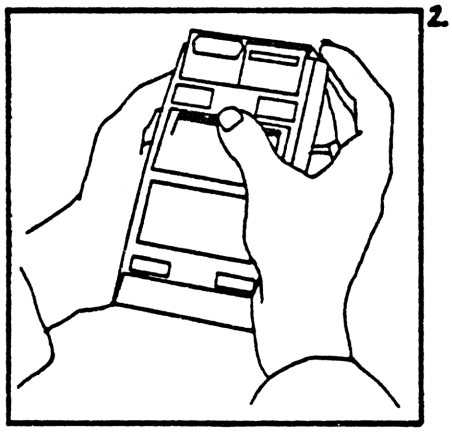
4. It can signal that the HP-41C has started or or completed a particular branch or subroutine.

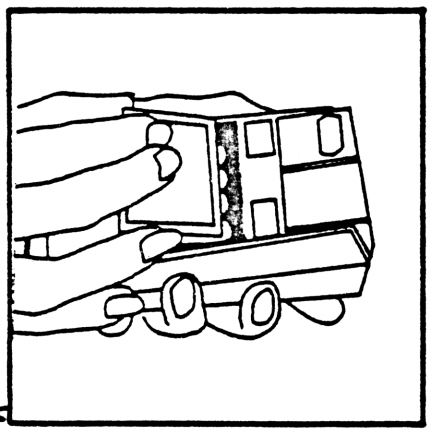
LONG BATTERY LIFE

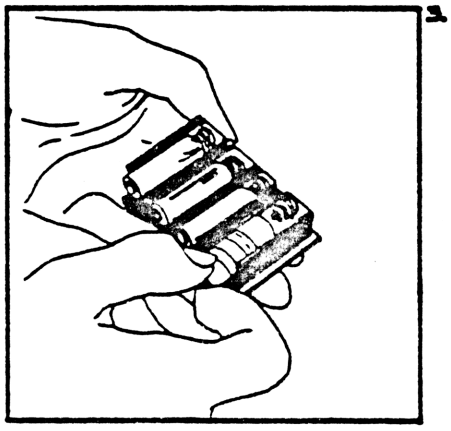
- STANDARD N CELL BATTERIES
- SIX TO TWELVE MONTHS BETWEEN CHANGES
- BAT ANNUNCIATOR SIGNALS LOW BATTERY POWER

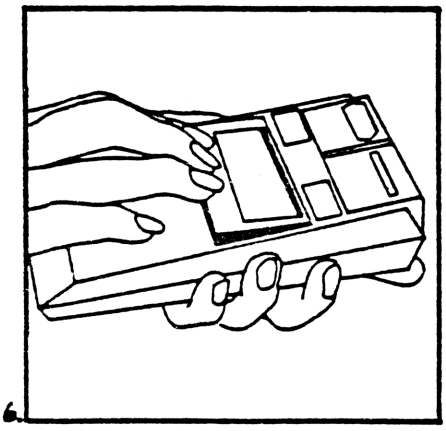












LONG BATTERY LIFE

Even though the HP-41C has continuous memory, its battery requirements are very low, which means LONG BATTERY LIFE.

The HP-41C uses four disposable N-cell Alkaline batteries, which are readily available wherever batteries are sold. A typical user should get 6-12 months of use from a set of batteries.

The BAT ANNUNCIATOR signals when batteries are low on power and should be replaced.

HEWLETT-PACKARD 41C

- THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . . IT COMMUNICATES WITH A USER

- THE HP-41C HAS AN EXCITING NEW KEYBOARD DESIGN . . . YOU CAN CUSTOMIZE THE CALCULATOR TO YOUR OWN DESIGN

- 130 STANDARD FUNCTIONS
- FUNCTION CATALOGS
- EXECUTE FUNCTION
- ASSIGN FUNCTION
- USER MODE KEYBOARD
- KEY PROMPTING
- KEYBOARD OVERLAYS

CUSTOMIZES TO YOUR OWN DESIGN

We've seen how the HP-41C communicates with the user.
Now we'll see how you can CUSTOMIZE it to your own design.

COMPLETE FUNCTION SET

- OVER 130 STANDARD FUNCTIONS, INCLUDING EXTENSIVE
  - GENERAL MATHEMATICS FUNCTIONS
  - TRIGONOMETRIC FUNCTIONS
  - LOGARITHMIC AND EXPONENTIAL FUNCTIONS
  - STATISTICAL FUNCTIONS
  - OPERATIONAL AND GENERAL FUNCTIONS

COMPLETE FUNCTION SET

130 STANDARD FUNCTIONS

As you may have noticed by now, not all of the functions available in the HP-41C are printed on the keyboard.

The HP-41C has a very complete function set, including over 130 standard functions, 68 of which are immediately accessable by pressing keys on the keyboard.

FUNCTIONS CATALOGS

CATALOG 1 - THE USER CATALOG
CATALOG 2 - THE EXTENSION CATALOG
CATALOG 3 - THE STANDARD FUNCTION CATALOG

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[GOLD] [CATALOG]	CAT_
3	CAT 3
	+
	-
	.
	.
	.
	X↗2
	Y↗X
	0.0000
[GOLD] [CATALOG]	CAT_
1	.END. REG 53
	0.0000
[GOLD] [CATALOG]	CAT_
2	0.0000

FUNCTIONS CATALOGS

Most of the standard functions (117) are conveniently stored away for the user in what is called CATALOG 3. In order to review the standard functions contained in Catalog 3, press [Gold] Catalog 3. The HP-41C will then list the functions in alphabetical order. To slow down the listing press and hold [←], to stop the listing press [R/S]. After pressing [R/S] you can single step forward or backward through the catalog listing. To continue the listing, press [R/S] again.

Besides Catalog 3, which contains standard functions, the HP-41C has two other catalogs.

CATALOG 1, the user catalog, contains a listing of all user programs currently stored in the HP-41C. If no user programs are stored in the HP-41C, Catalog 1 remains empty.

CATALOG 2, the extension catalog, contains all functions that become active when you plug extensions into the HP-41C, such as the Card Reader, Printer, Application Modules, or other accessories. If no extensions are plugged into the HP-41C, Catalog 2 remains empty. (If an extension is available, review Catalog 2 while it is plugged into the HP-41C.)

EXECUTE FUNCTION

KEYSTROKES	DISPLAY
6	6_
[XEQ]	XEQ _ _
[ALPHA]	XEQ_
FACT	XEQ FACT_
[ALPHA]	720.0000
[XEQ]	XEQ _ _
[ALPHA]	XEQ_
SQRT	XEQ SQRT
[ALPHA]	26.8328
[XEQ]	XEQ _ _
[ALPHA]	XEQ_
INT	XEQ INT_
[ALPHA]	26.0000

EXECUTE FUNCTION

All 130 standard functions can be executed from the display using the EXECUTE FUNCTION.

Before using the Execute Function, you must first place the argument to be executed upon in the display. Then press the [XEQ] key. The HP-41C will then prompt you for a function to be executed. The HP-41C will execute the desired function immediately after receiving it, just as it does when executing a keyboard function.

ASSIGN FUNCTION

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA]	ASN_
FACT	ASN FACT_
[ALPHA]	ASN FACT _ _
[Σ+]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA]	ASN_
SQRT	ASN SQRT_
[ALPHA]	ASN SQRT _ _
[1/X]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA]	ASN_
INT	ASN INT_
[ALPHA]	ASN INT _ _
[$\sqrt{X}$]	0.0000

ASSIGN FUNCTION

You may find that you use functions not on the keyboard as often as those permanently on the keyboard. The HP-41C allows you the flexibility to correct for this by providing the capability to assign any function in any catalog to any key. This includes Standard Functions, Extension Programs, or User Programs. This flexibility to customize your calculator to your own design is made possible with the ASSIGN FUNCTION.

Once a function or a program is assigned to a key, merely press the key while in User Mode, and it is immediately executed.

If you attempt to assign a function whose name does not exist in the calculator, the HP-41C will display non-existent. Use Catalog 3 to help you with function names.

USER MODE KEYBOARD

KEYSTROKES	DISPLAY
[USER]	0.0000
6	6_
[Σ+] (FACT)	720.0000
[1/X] (SQRT)	26.8328
[$\sqrt{X}$] (INT)	26.0000
[USER]	26.0000
[$\sqrt{X}$]	5.0990
[GOLD] [CLX]	0.0000

USER MODE KEYBOARD

User assigned functions are active while the calculator is in USER MODE.

If you have assigned a function or a program to a key, that function or program will execute immediately when you press the key (while in User Mode). If you have not assigned anything to a key, its normal function continues to be active even in User Mode.

KEY PROMPTING

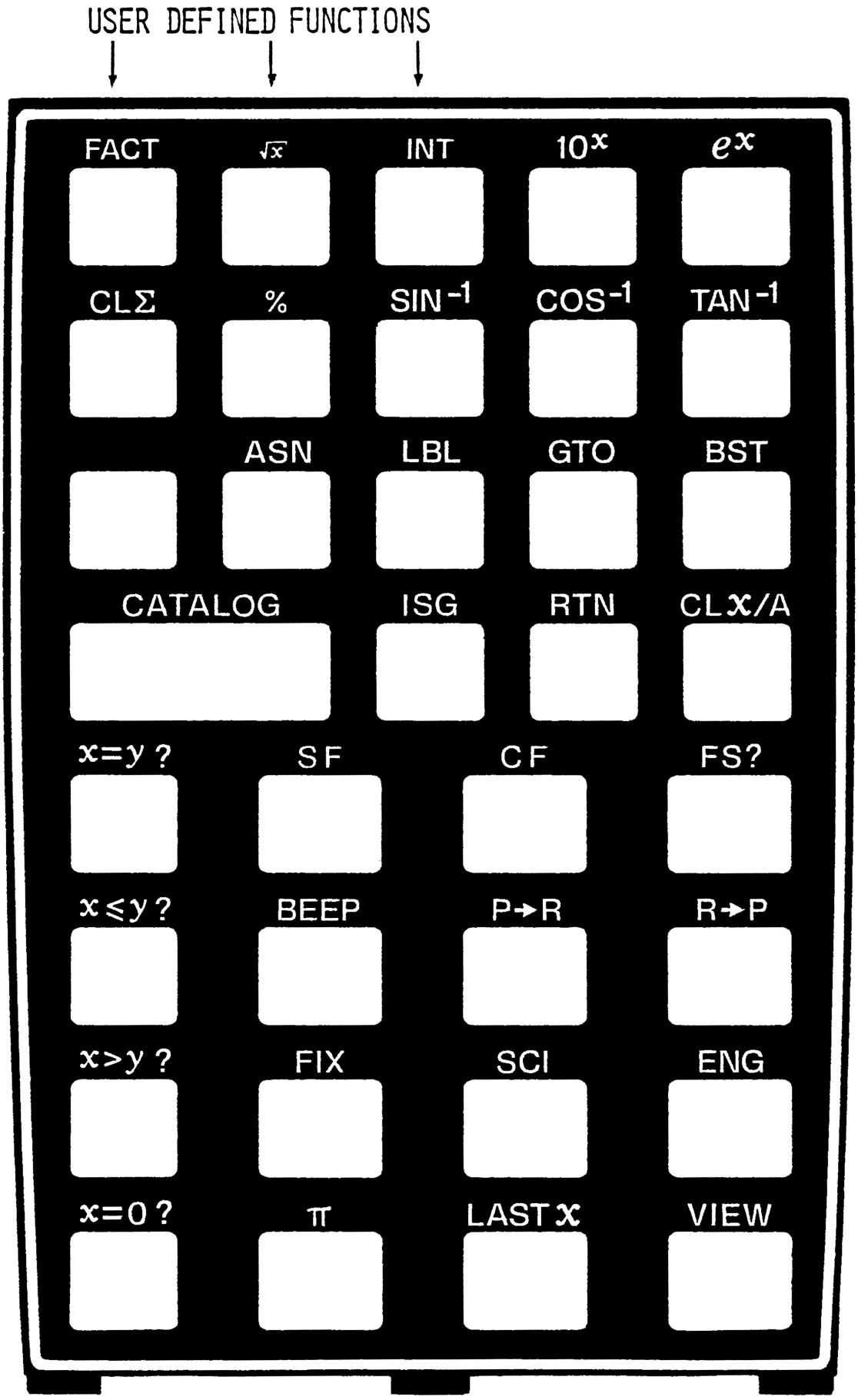
KEYSTROKES	DISPLAY
[Σ+]	Σ+
	NULL
[1/X]	1/X
	NULL
[√X]	SQRT
	NULL
[USER]	
[Σ+]	FACT
	NULL
[1/X]	SQRT
	NULL
[√X]	INT
	NULL
[USER]	0.0000

KEY PROMPTING

The HP-41C will tell you which functions are active in both modes for each key through a special feature called KEY PROMPTING.

If you press down on a key and hold it down, it will prompt you with the function assigned to that key. The same is true while in User Mode. If you release the key, the function will execute. If you continue to hold the key down, the word NULL will appear, and the function will not execute.

KEYBOARD OVERLAYS



KEYBOARD OVERLAYS

We assigned special functions to three keys, using the assign function. We have the ability, though, to assign special functions to every key except [Gold], [ON], [USER], [PRGM], and [ALPHA].

Thus, in User Mode, the keyboard is a very personalized problem solver. To further enhance this personalization, KEYBOARD OVERLAYS may be used, thus completing the customization of the keyboard while providing the user with a convenient means of identifying all user defined functions at once.

HEWLETT-PACKARD 41C

- THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . .
  IT COMMUNICATES WITH THE USER

- THE HP-41C HAS AN EXCITING NEW KEYBOARD DESIGN . . . YOU CAN CUSTOMIZE THE CALCULATOR TO YOUR OWN DESIGN

- THE HP-41C HAS A UNIQUE COMBINATION OF PROGRAMMING EASE AND SOPHISTICATION . . .
  IT IS EASY TO PROGRAM

  KEYSTROKE PROGRAMMING
  ALPHA KEY MNEMONICS
  COMPLETE PROGRAM LINES
  ALPHANUMERIC PROGRAM NAMES
  COMPREHENSIVE EDITING
  ALPHA PROMPTING AND LABELING

THE HP-41C HAS A UNIQUE COMBINATION OF PROGRAMMING EASE AND SOPHISTICATION . . . IT IS EASY TO PROGRAM

We've seen how the HP-41C communicates with the user, and how you can customize the calculator to your own design.

The HP-41C also has a unique combination of programming ease and sophistication. It is easy to program. We'll learn just how easy it is as we discover the following HP-41C features.

SIMPLE KEYSTROKE PROGRAMMING

WHEN SOLVED MANUALLY

KEYSTROKES	DISPLAY
5	5_
[GOLD] [X^2]	25.0000
[GOLD] [π]	3.1416
[X]	78.5398

WHEN ENTERED AS A PROGRAM

KEYSTROKES	DISPLAY
[PRGM]	END REG NN
[GOLD] [GTO]. .	00 REG NN
[GOLD] [LBL]	01 LBL _ _
[ALPHA] CIRCLE [ALPHA]	1 LBL ^{T} CIRCLE
[GOLD] [X^2]	02 X↑2
[GOLD] [π]	03 PI
[X]	04 *
[GOLD] [GTO]. .	00 REG NN
[PRGM]	78.5398
5	5_
[XEQ]	XEQ _ _
[ALPHA] CIRCLE [ALPHA]	78.5398

SIMPLE KEYSTROKE PROGRAMMING

A program is little more than a series of keystrokes that you would press to solve a problem manually. Except that when you program, the calculator REMEMBERS the keystrokes as you enter them, then executes all of the specified keystrokes in the proper sequence whenever you wish.

Before keying in a program, press [Gold] GTO .. This positions the calculator to the end of program memory (after the last existing program in program memory), and is now ready for you to begin keying in the instructions of your program. The display will show 00 Reg nn. The nn indicates the number of registers that are unused in program memory (more about this later).

In addition to positioning the calculator to the end of program memory, [Gold] GTO .., also checks to see if the last program you keyed in was terminated with an [END] instruction. If an [END] was not keyed in as the last instruction of the preceding program, [Gold] GTO .. automatically inserts one. In this way, the HP-41C automatically maintains program memory for you.

You can see that [Gold] GTO .. is extremely useful. Before and after keying in a program, simply press [Gold] GTO .. and the calculator will automatically manage program memory.

HP-41C SPECIAL PROGRAMMING FEATURES

* ALPHA PROGRAM NAMES
* ALPHA KEYCODES
* COMPLETE PROGRAM LINES
* USER PROGRAM PROMPTING

KEYSTROKES	DISPLAY
[GOLD] [GTO]	GTO_
[ALPHA] CIRCLE [ALPHA]	78.5398
[PRGM]	1 LBL T CIRCLE
[SST]	02 X↗ 2
[SST]	03 PI
[SST]	04 *
[SST]	05 END
[PRGM]	78.5398

HP-41C SPECIAL PROGRAMMING FEATURES

The HP-41C offers special features to the user to help make programs easier to edit and remember.

1. ALPHA PROGRAM NAMES - Alphabetic names can be used to identify programs. This makes them both more meaningful and easier to remember.

2. ALPHA KEY CODES - An alphabetic name identifies each function used in a program, rather than complicated key codes as with other calculators.

3. COMPLETE PROGRAM LINES - Each complete instruction forms one program line so program listings are easier to read and understand.

4. USER PROGRAM PROMPTING - The HP-41C identifies user programs with a little T. This prompt helps to differentiate user programs from HP-41C standard functions.

PROGRAM EDITING

TOTAL SURFACE AREA
OF A CYLINDER

$S = (2\pi r) + (2\pi rh)$

KEYSTROKES	DISPLAY
[PRGM]	05 END
[GOLD] [GTO] . .	00 REG NN
[GOLD] [LBL]	01 LBL _ _
[ALPHA] AREA [ALPHA]	01 LBL T AREA
[STO] 01	02 STO 01
[GOLD] [X^2]	03 X↑2
[GOLD] [π]	04 PI
[X]	05 *
2	06 2_
[X]	07 *
[X≷Y]	08 X<>Y
[RCL] 01	09 RCL 01
[X]	10 *
[GOLD] [π]	11 PI
[X]	12 *
2	13 2_
[X]	14 *
[+]	15 +
[GOLD] [GTO] . .	00 REG NN
[PRGM]	78.5398

PROGRAM EDITING

Often you may wish to alter or add to a program that you have keyed into the calculator. The HP-41C has several editing functions that permit you to easily change any lines in any of your programs without reloading the entire program.

Using the program "Area", we'll learn about the following edit functions:

RESETTING TO THE BEGINNING OF A PROGRAM

SINGLE LINE EXECUTION

SINGLE LINE VIEWING WITHOUT EXECUTION

MODIFYING EXECUTION

DELETING AND CORRECTING INSTRUCTIONS

EXECUTING AREA

KEYSTROKE	DISPLAY
50	50_
[ENTER]	50.0000
11	11_
[GOLD] [ASN]	ASN_
[ALPHA] AREA [ALPHA]	ASN AREA_
[LOG]	11.0000
[USER]	11.0000
[LOG] (AREA)	4,216.0173
[USER]	4,216.0173

EXECUTING AREA

Before reviewing the program editing capabilities of the HP-41C, let's make sure we loaded the program correctly by running a sample problem.

Let's also assign "Area" to the [Log] key.

SINGLE-STEP PROGRAM EXECUTION

KEYSTROKE	DISPLAY
[GOLD] [GTO]	GTO _ _
[ALPHA] AREA [ALPHA]	4,216.0173
[PRGM]	01 LBL ^{T} AREA
[PRGM]	4,216.0173
50	50_
[ENTER]	50.0000
11	11_
[SST]	11.0000
[SST]	11.0000
[SST]	121.0000
[SST]	3.1416
[SST]	380.1327
[SST]	2.0000
[SST]	760.2654
[SST]	50.0000
[SST]	11.0000
[SST]	550.0000
[SST]	3.1416
[SST]	1,727.8760
[SST]	2.0000
[SST]	3,455.7519
[SST]	4,216.0173

RESET TO BEGINNING OF PROGRAM

AND EXECUTE LINE BY LINE

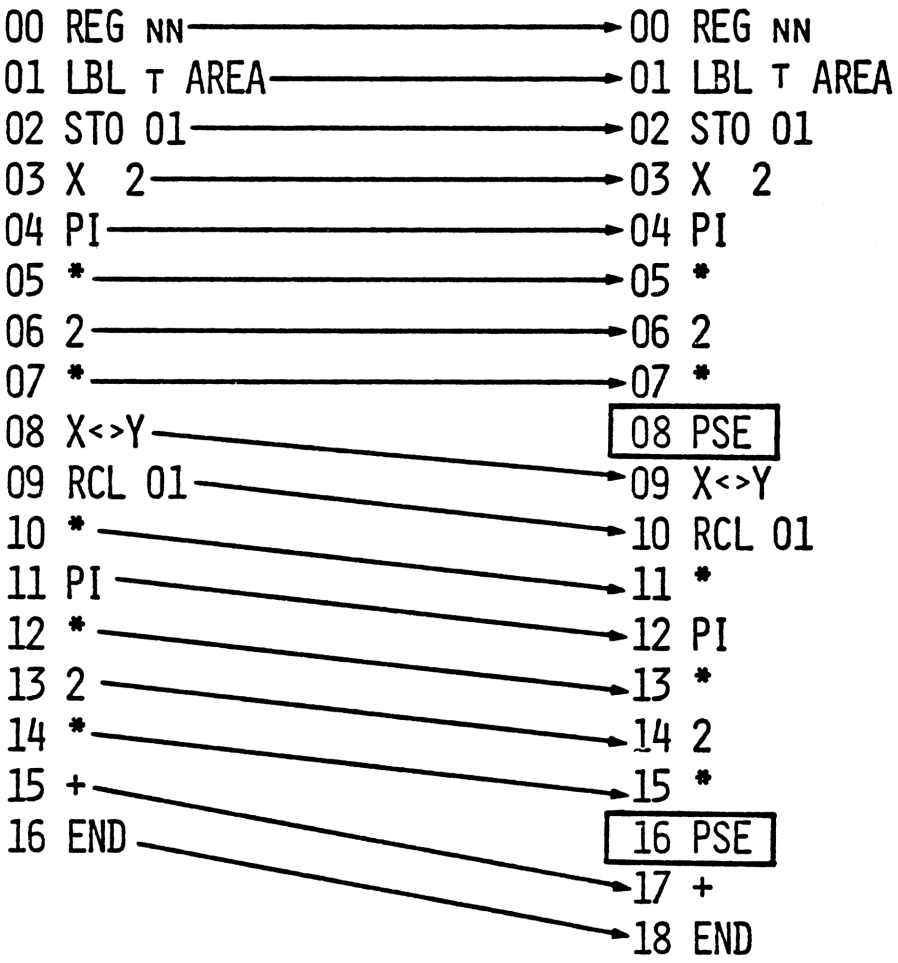
In order to reset the calculator to the beginning of a program, use the following keystrokes.

IF CURRENTLY POSITIONED - [Gold] RTN or
[GOLD] GTO .000

IF NOT CURRENTLY POSITIONED - [Gold] GTO [ALPHA]
AREA [ALPHA]

After you have initialized the program, you can execute it line by line using the [SST] function.

MODIFYING A PROGRAM



MODIFYING A PROGRAM

You can modify a program by first resetting to its beginning and then single stepping through the program while in program mode to the point in the program where you would like to make an addition.

MODIFYING A PROGRAM USING [SST]

KEYSTROKES	DISPLAY
[GOLD] [GTO]	GTO_
[ALPHA] AREA [ALPHA]	4,216.1073
[PRGM]	01 LBL T AREA
[SST]	02 STO 01
[SST]	03 X/2
[SST]	04 PI
[SST]	05 *
[SST]	06 2
[SST]	07 *
[XEQ]	08 XEQ_
[ALPHA] PSE [ALPHA]	08 PSE
[SST]	09 X<>Y
[SST]	10 RCL 01
[SST]	11 *
[SST]	12 PI
[SST]	13 *
[SST]	14 2
[SST]	15 *
[XEQ]	16 XEQ_
[ALPHA] PSE [ALPHA]	17 +
[SST]	18 END
[PRGM]	4,216.0173

MODIFYING A PROGRAM USING [SST]

Try modifying a program by adding the Pause function to the area program so that it will momentarily display intermediate results during execution. Remember to use the execute function when inserting an HP-41C function.

1ST INTERMEDIATE RESULT = Area of the cylinder ends

2ND INTERMEDIATE RESULT = Area of the cylinder without the ends.

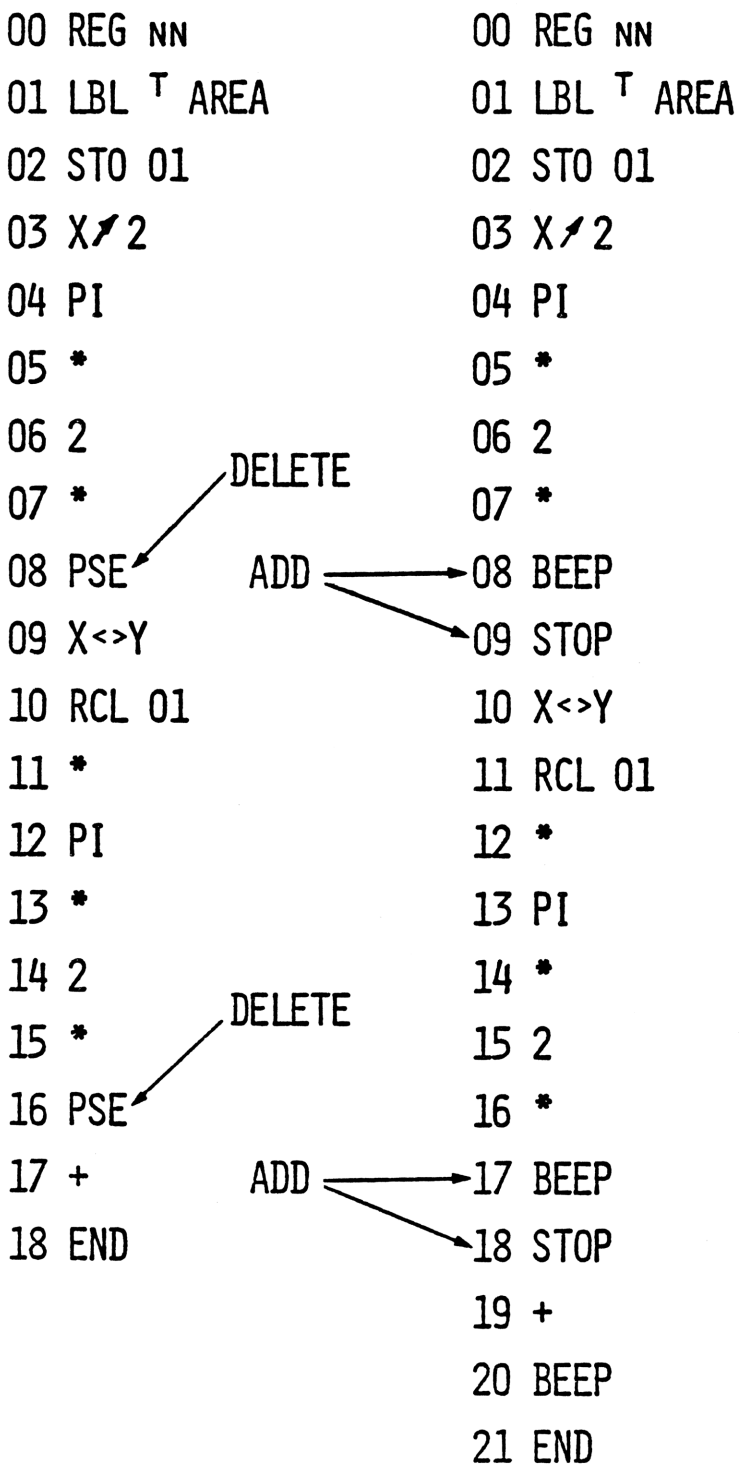
RUNNING THE MODIFIED PROGRAM

KEYSTROKES	DISPLAY
[USER]	4,216.0173
50	50_
[ENTER]	50.0000
11	11_
[LOG] (AREA)	760.2654
	3,455.7519
	4,216.0173

RUNNING THE MODIFIED PROGRAM

When you run the modified program, you can check intermediate results momentarily before the final result is displayed.

DELETING AND CORRECTING PROGRAM LINES



DELETING AND CORRECTING PROGRAM LINES

Suppose that you wanted to write down your intermediate results, and wanted to be notified by your HP-41C when it had them calculated.

You could do this by first DELETING the PAUSE functions, and then INSERTING first the BEEPER function and then the STOP function. The BEEPER function will alert you that the HP-41C has obtained your intermediate results. The STOP function halts program execution, allowing the user to record results. The user can continue program execution by pressing [R/S].

DELETING PAUSE AND INSERTING BEEP AND STOP

KEYSTROKES	DISPLAY
[GOLD] [GTO]	GTO_
[ALPHA] AREA [ALPHA]	4,216.0173
[PRGM]	01 LBL ^{T} AREA
[GOLD] [GTO] [.]	GTO._ _ _
008	08 PSE
[←]	07 *
[GOLD] [BEEP]	08 BEEP
[R/S]	09 STOP
[GOLD] [GTO] [.]	GTO._ _ _
017	17 PSE
[←]	16 *
[GOLD] [BEEP]	17 BEEP
[R/S]	18 STOP
[SST]	19 +
[GOLD] [BEEP]	20 BEEP
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	4,216.0173

DELETING PAUSE AND INSERTING BEEP AND STOP

Now, let's modify our Area Program again, this time by DELETING our PAUSE instructions and inserting the BEEP and STOP functions. Rather than single stepping down to the point where you would like to make a change, use the GTO .XXX editing feature.

To DELETE a program line, press the backarrow key while the calculator is positioned at the line to be deleted. Then, simply insert the correct line.

RUNNING THE MODIFIED PROGRAM

KEYSTROKES	DISPLAY
50	50_
[ENTER]	50.0000
11	11_
[LOG] (AREA)	BEEP
	760.1654
[R/S]	BEEP
	3,455.7519
[R/S]	BEEP
	4,216.0173

RUNNING THE MODIFIED PROGRAM

After modifying the Area Program again, try running it. Note the BEEP, alerting you that an intermediate result has been obtained.

To restart program execution, simply press [R/S].

ALPHA PROMPTING

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[GOLD] [GTO]	GTO _ _
[ALPHA] CIRCLE [ALPHA]	4,216.0173
[PRGM]	1 LBL ᵀ CIRCLE
[ALPHA] RADIUS? [ALPHA]	02 ᵀ RADIUS?
[XEQ]	03 XEQ _ _
[ALPHA] PROMPT [ALPHA]	03 PROMPT
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA] CIRCLE [ALPHA]	ASN CIRCLE _ _
[X≷Y]	0.0000
[X≷Y] (CIRCLE)	RADIUS?
5	5_
[R/S]	78.5398

ALPHA PROMPTING

Some of the greatest benefits of the HP-41C ALPHA capabilities are realized during program execution. ALPHA Strings (a series of Alpha characters) in your programs can prompt you for information, inform you of the status of a program, and even label output.

Using the "Circle" program we wrote earlier, we'll see how to prompt for DATA Input using the Prompt function.

DATA LABELING

KEYSTROKES	DISPLAY
[GOLD] [GTO]	GTO_
[ALPHA] CIRCLE [ALPHA]	78.5398
[PRGM]	1 LBL ^{T} CIRCLE
[GOLD] [GTO].006	06 *
[ALPHA] AREA =	07 ^{T} AREA =
[GOLD] M (ARCL) [.]X	08 ARCL X
[GOLD] [AVIEW]	09 AVIEW
[ALPHA]	09 AVIEW
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	78.5398
[X≷Y] (CIRCLE)	RADIUS?
5	5_
[R/S]	AREA-78.5398
[X≷Y] (CIRCLE)	RADIUS?
10	10_
[R/S]	AREA=314.1593
[USER]	AREA=314.1593

DATA LABELING

DATA LABELING can make your program outputs much more meaningful. Labeled output leaves no doubt as to which result is displayed.

To label output, first key in the ALPHA line as a line in the program, then recall the result to be labeled into the display using [ARCL]. Remember that [ARCL] simply adds to whatever is already in the Alpha register. Then use the [AVIEW] function to place the contents of the Alpha register into the display.

HEWLETT-PACKARD 41C

- THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . . IT COMMUNICATES WITH THE USER

- THE HP-41C HAS AN EXCITING NEW KEYBOARD DESIGN . . . . YOU CAN CUSTOMIZE THE CALCULATOR TO YOUR OWN DESIGN

- THE HP-41C HAS A UNIQUE COMBINATION OF PROGRAMMING EASE AND SOPHISTICATION . . . IT IS EASY TO PROGRAM

- THE HP-41C HAS PROGRAMMING POWER . . . IT CAN SOLVE PROFESSIONAL PROBLEMS

  - CONDITIONAL BRANCHING
  - CONTROLLED LOOPING
  - SUBROUTINES
  - LOCAL LABELS
  - INDIRECT ADDRESSING
  - FLAGS

THE HP-41C HAS PROGRAMMING POWER . . . IT CAN SOLVE PROFESSIONAL PROBLEMS

The HP-41C offers new capabilities in controlled looping, indirect addressing, and program labeling.

It also offers a complete set of user definable and system control flags.

UNCONDITIONAL BRANCHING

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[PRGM]	00 REG NN
[GOLD] [GTO] [.] [.]	00 REG NN
[GOLD] [LBL]	01 LBL_
[ALPHA] ROOT [ALPHA]	01 LBL T ROOT
0	02 0_
[STO] 01	03 STO 01
[GOLD] [LBL] 05	04 LBL 05
1	05 1_
[STO] [+] 01	06 ST + 01
[RCL] 01	07 RCL 01
[XEQ]	08 XEQ_
[ALPHA] PSE [ALPHA]	08 PSE
[$\sqrt{X}$]	09 SQRT
[XEQ]	10 XEQ_
[ALPHA] PSE [ALPHA]	10 PSE
[GOLD] [GTO] 05	11 GTO 05
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	0.0000

UNCONDITIONAL BRANCHING

You have used the [GTO] instruction earlier to position the calculator to a particular place in program memory. You can also use the [GTO] instruction in your programs, followed by an Alpha or numeric label to transfer execution to any part of the program you desire.

A [GTO] instruction used in this way is known as an UNCONDITIONAL BRANCH. It always branches execution to the specified label.

EXECUTING AN UNCONDITIONAL BRANCH

KEYSTROKES	DISPLAY
[XEQ]	XEQ _ _
[ALPHA] ROOT [ALPHA]	1.0000
	1.0000
	2.0000
	1.4142
	3.0000
	1.7321
	4.0000
	2.0000
	5.0000
[R/S]	2.2361
[GOLD] [CLX]	0.0000

EXECUTING AN UNCONDITIONAL BRANCH

Now that we've loaded the program "Root", let's execute it. Notice that it will run until we halt its execution with [R/S]. This is because the Unconditional Branch to LBL 05 created what is known as an infinite loop.

CONDITIONAL BRANCHING

CONDITIONAL BRANCHING ALLOWS THE CALCULATOR TO MAKE DECISIONS WHILE A PROGRAM IS RUNNING

$X = Y$?	$X < 0$?
$X = 0$?	$X \leq Y$?
$X > Y$?	$X \leq 0$?
$X > 0$?	$X \neq Y$?
$X < Y$?	$X \neq 0$?

CONDITIONAL BRANCHING

CONDITIONAL BRANCHING allows the HP-41C to make decisions while the program is running.

It does this by testing the value in the X register against the value in the Y register, or the value in the X register against Ø.

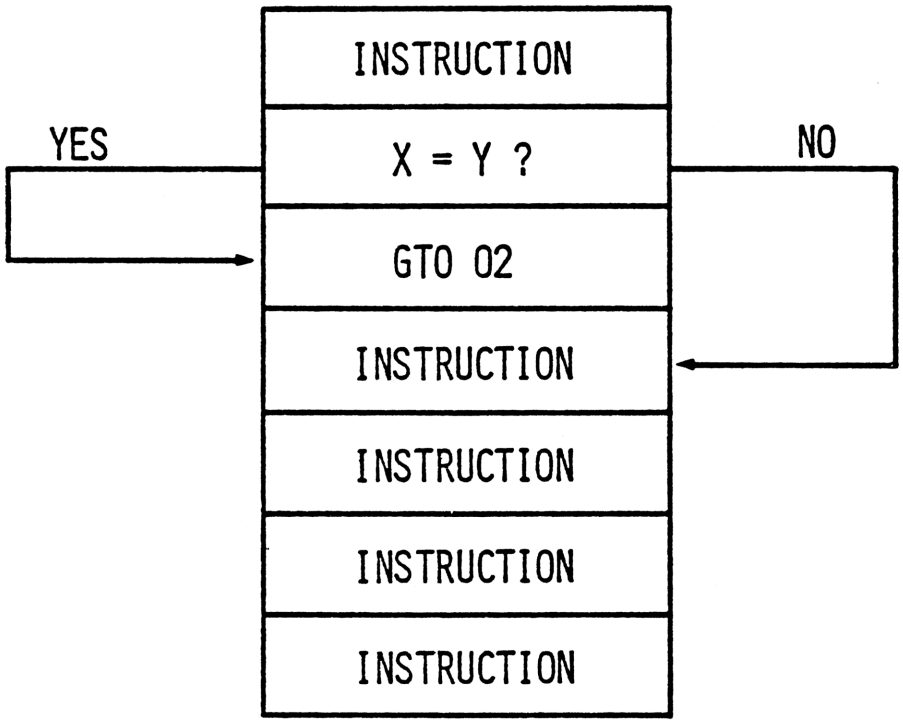
The HP-41C offers 10 different CONDITIONAL BRANCHING "TESTS".

X = Y ?	X < Y ?	X ≠ Y ?
X = 0 ?	X < 0 ?	X ≠ 0 ?
X > Y ?	X ≤ Y ?	
X > 0 ?	X ≤ 0 ?	

Two of these conditionals, [X = Y ?] and [X ≠ Y ?], can also be used to compare Alpha strings as well as numbers. All the other conditionals compare only numbers. For two Alpha strings to be equal, they must be Exactly equal in length and have identical characters.

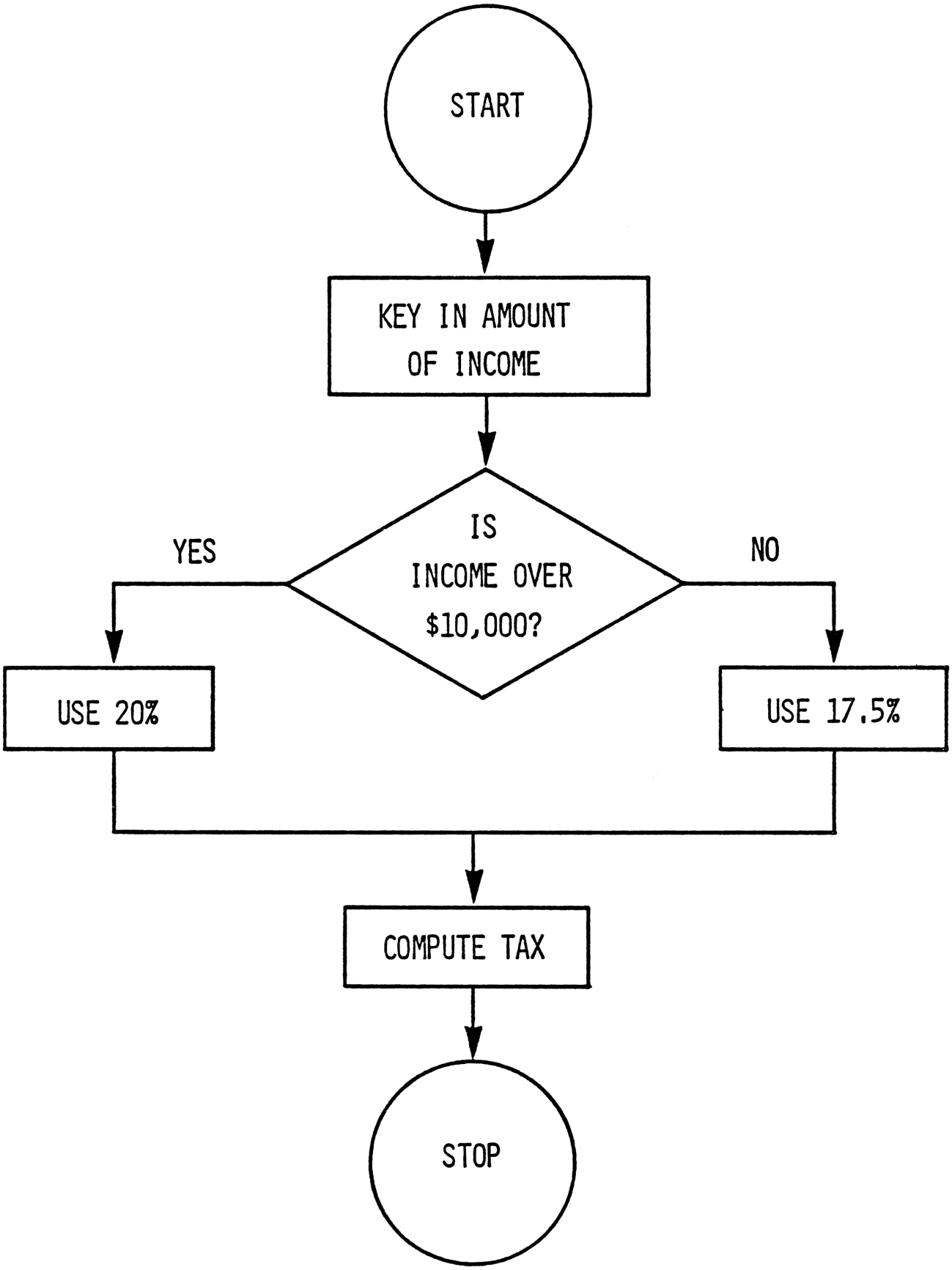
CONDITIONAL TESTS

THE RULE TO REMEMBER IS "DO IF TRUE"



CONDITIONAL TESTS

To see how a decision is made while a program is running, look at the block of program lines. If the answer to a CONDITIONAL TEST is true, the HP-41C continues to the next program line. This highlights the "DO IF TRUE" rule of CONDITIONAL TESTS. If the answer to a CONDITIONAL TEST is NO, the calculator skips a program line and continues on.


START
KEY IN AMOUNT OF INCOME
IS INCOME OVER $10,000?
YES
NO
USE 20%
USE 17.5%
COMPUTE TAX
STOP

FLOWCHART EXAMPLE

Let's assume that we have to compute the Federal income tax to be paid on several incomes, and that the rate of tax differs for incomes above and below $10,000. For incomes of $10,000 or less, the tax rate is 17.5%. For incomes above $10,000, the tax rate is 20%.

In order to understand this problem, we can lay it out in FLOWCHART form. When we do this, we see that a decision must be made each time we solve the problem.

USING A CONDITIONAL TEST

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[PRGM]	00 REG NN
[GOLD] [GTØ] [.] [.]	00 REG NN
[GOLD] [LBL]	01 LBL_
[ALPHA] TAX [ALPHA]	01 LBL T TAX
[ALPHA] INCOME? [ALPHA]	02 T INCOME?
[XEQ]	03 XEQ_
[ALPHA] PROMPT [ALPHA]	03 PROMPT
10000	04 10000_
[X≷Y]	05 X≷Y
[GOLD] [X>Y?]	06 X>Y?
[GOLD] [GTO] 02	07 GTO 02
17.5	08 17.5
[GOLD] [GTO] 03	09 GTO 03
[GOLD] [LBL] 02	10 LBL 02
20	11 20_
[GOLD] [LBL] 03	12 LBL 03
[GOLD] [%]	13 %
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	0.0000

FIGURING TAXES FOR DIFFERENT INCOMES

USING A CONDITIONAL TEST

Once we've flowcharted our problem, it is easy to write a program using the right CONDITIONAL TEST.

Name your program "TAX", and have it prompt for income.

USING THE "TAX" PROGRAM

FIGURE FEDERAL INCOME TAXES USING THE TAX PROGRAM FOR INCOMES OF $8,000, $10,000, AND $15,000

KEYSTROKES	DISPLAY
[GOLD] [ASN]	ASN_
[ALPHA] TAX [ALPHA]	ASN TAX_
[TAN]	0.0000
[USER]	0.0000
[TAN] (TAX)	INCOME?
8000	8,000_
[R/S]	1,400.0000
[TAN] (TAX)	INCOME?
10000	10,000_
[R/S]	1,750.0000
[TAN] (TAX)	INCOME?
15000	15,000_
[R/S]	3,000.0000
[USER]	3,000.0000
[GOLD] [CLX]	0.0000

USING THE TAX PROGRAM

Once you've written and loaded your TAX PROGRAM, figuring Federal income taxes for different incomes is easy.

Figure Federal tax for incomes of $8,000, $10,000, and $15,000. Notice how the Conditional Test "DECIDES" which tax rate to use.

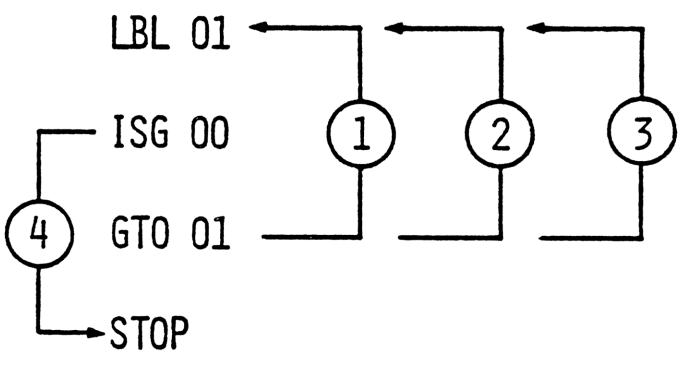
CONTROLLED LOOPING

- ISG - INCREMENT AND SKIP IF GREATER
- DSE - DECREMENT AND SKIP IF EQUAL
- CONTROL NUMBER FORMAT iiii.fffcc

  iiii - CURRENT COUNTER VALUE
  fff - COUNTER TEST VALUE
  cc - INCREMENT/DECREMENT VALUE

- ASSUME AN ORIGINAL CONTROL NUMBER IN R_{00} = 4.01002

DIAGRAM OF PROGRAM EXECUTION IS



AS COUNTER VALUE INCREMENTS TO

(1) 6.01002
(2) 8.01002
(3) 10.01002
(4) 12.01002

CONTROLLED LOOPING

The HP-41C has two powerful functions that make program looping control very easy. These functions are [ISG] (increment and skip if greater), and [DSE] (decrement and skip if equal). Both functions contain integral counters that allow you to control the number of executions of a loop.

These two functions use a number that is stored in a special way to CONTROL THE LOOPING process. The Control number may be stored in any storage register (even the stack).

The format of the control number is:

iii.fffcc

where iiii is the current counter value
fff is the current counter value
cc is the increment/decrement value

A new feature offered by the HP-41C is the ability to increment or decrement by any integer value between 1 and 99, rather than just incrementing or decrementing by 1.

USING A CONTROLLED LOOP

WRITE A PROGRAM THAT GENERATES A TABLE OF SQUARES OF EVEN NUMBERS FROM 2 THROUGH 20

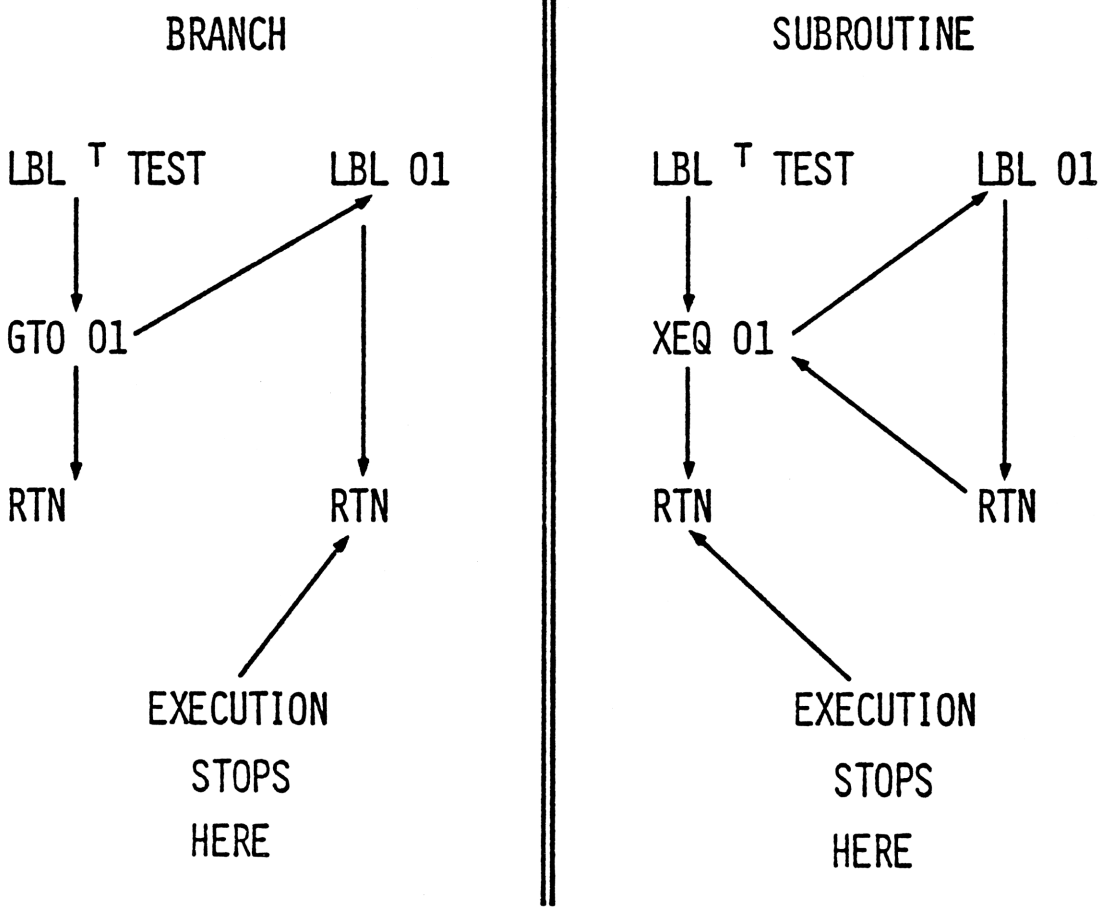
KEYSTROKES	DISPLAY
[PRGM]	00 REG NN
[GOLD] [GTO] [.] [.]	00 REG NN
[GOLD] [LBL]	01 LBL_
[ALPHA] EVENS [ALPHA]	01 LBL T EVENS
2.02002	02 2.02002_
[STO] 01	03 STO 01
[GOLD] [LBL] 01	04 LBL 01
[RCL] 01	05 RCL 01
[XEQ]	06 XEQ_
[ALPHA] INT [ALPHA]	06 INT
[XEQ]	07 XEQ_
[ALPHA] PSE [ALPHA]	07 PSE
[GOLD] [X^2]	08 X 2
[XEQ]	09 XEQ_
[ALPHA] PSE [ALPHA]	09 PSE
[GOLD] [ISG] 01	10 ISG 01
[GOLD] [GTO] 01	11 GTO 01
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	3,000.0000
[XEQ]	XEQ_
[ALPHA] EVENS [ALPHA]	2.0000
	4.0000
	16.0000
	6.0000
	36.0000
	.
	.
	.
	20.0000
	400.0000
[GOLD] [CLX]	0.0000

USING A CONTROLLED LOOP

To fully appreciate CONTROLLED LOOPING and the ability to increment or decrement, by more than one, let's write a program.

Write a program that generates a table of squares of even numbers from 2 to 20. Name the program "EVENS".

SUBROUTINES



SUBROUTINES

Often a program contains a certain series of instructions that are executed several times in several places in a program, or a program requires a set of instructions that are included in another program. These instructions can be executed by a program as a SUBROUTINE.

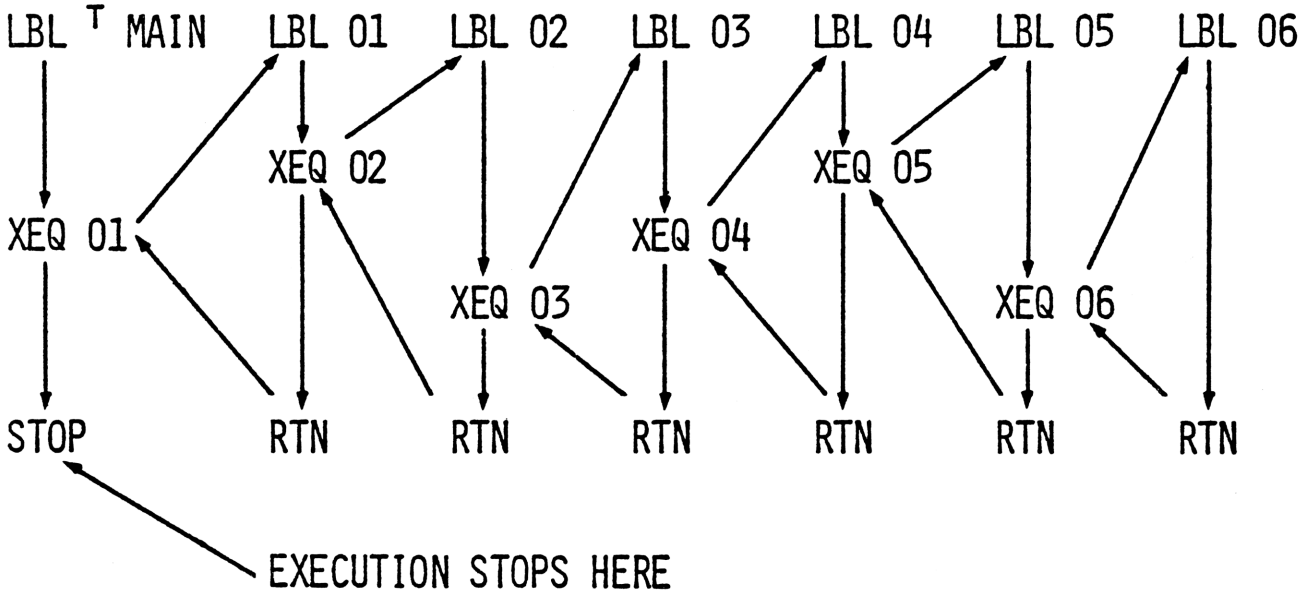
A SUBROUTINE is selected and executed in a program by the [XEQ] function. Using [XEQ], you can select either Alpha labeled or numeric labeled SUBROUTINES.

In a program, [XEQ] transfers execution to the program label specified by the [XEQ] function. After the SUB-ROUTINE has been executed, and the running program executes an [END] or [RTN], execution is transferred back to the main program. Execution then continues with the next instruction after the [XEQ] and sequentially down through the program.

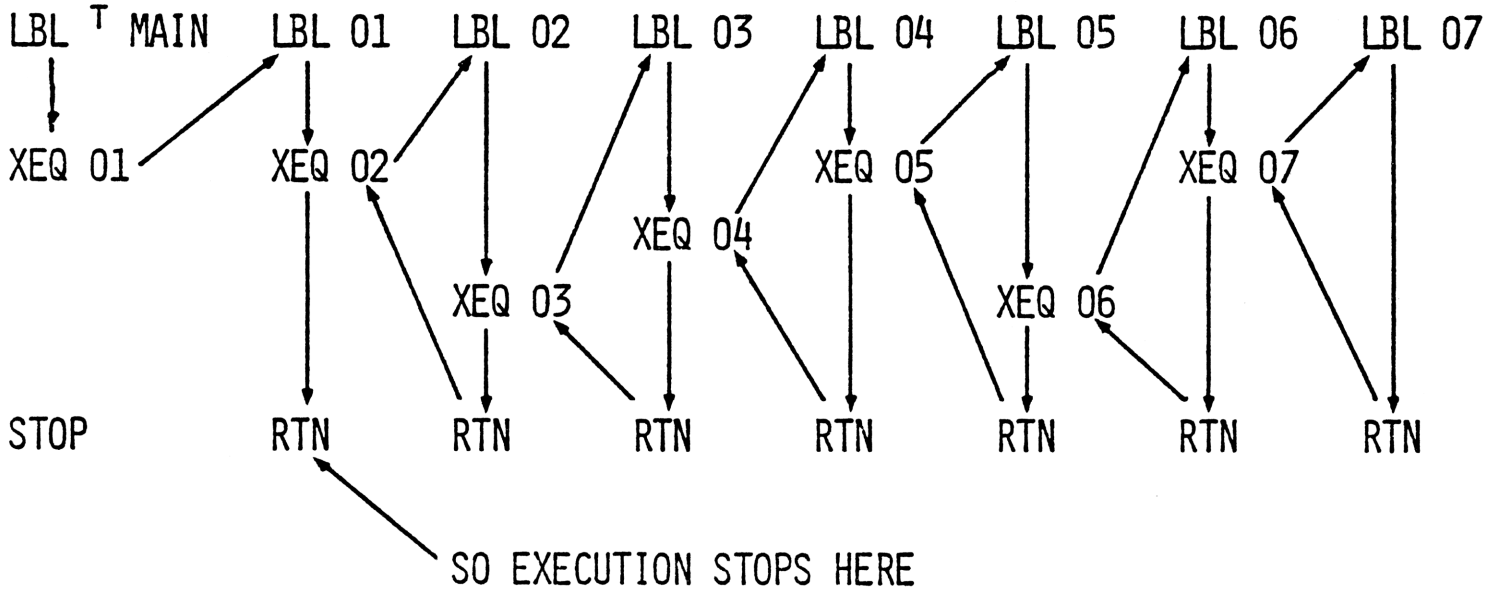
It is very important to note how this differs from branching, for the [GTO] instruction merely transfers execution to the specified label but does not return execution to the main program.

SUBROUTINE LIMITS

- UP TO SIX SUBROUTINE LIMITS CAN BE HELD PENDING



- ONLY SIX SUBROUTINE LIMITS CAN BE HELD PENDING THOUGH



SUBROUTINE LIMITS

A subroutine can call up another subroutine, and that subroutine can call up yet another. In fact, you can have up to six subroutine branches before returning to the first program. SUBROUTINE BRANCHING IS LIMITED ONLY BY the number of [END]s or [RTN]s that can be held pending by the calculator, which for the HP-41C is six.

LOCAL AND GLOBAL SUBROUTINE LABELS

LOCAL LABELS - NUMERIC LABELS 00 - 99
ALPHA LABELS A-J AND a-e

* CALCULATOR SEARCHES INSIDE CURRENT PROGRAM SPACE ONLY

GLOBAL LABELS - ALL ALPHA LABELS OTHER THAN A-J AND a-e

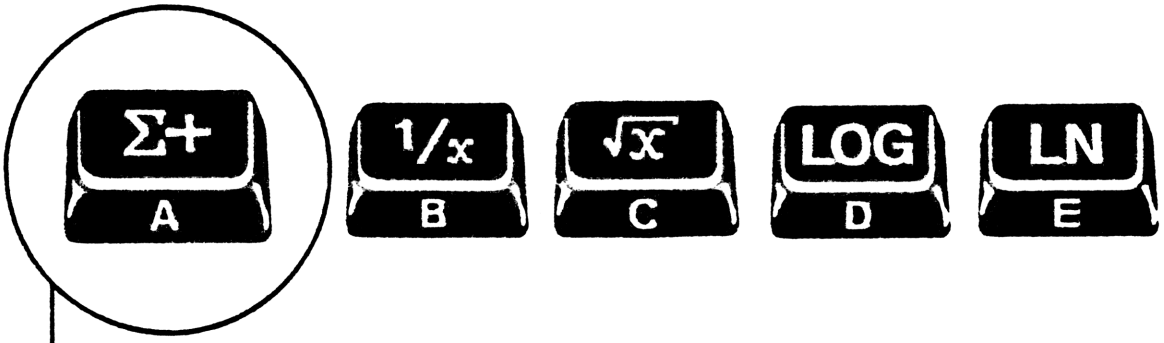
* CALCULATOR SEARCHES ALL OF PROGRAM MEMORY

LOCAL AND GLOBAL SUBROUTINE LABELS

The HP-41C features two different types of subroutine labels, LOCAL AND GLOBAL.

The HP-41C will search only inside the current program file for Local Labels, while it will search its entire memory for Global Labels. Program execution time can be optimized with proper use of Local and Global Labels.

ALPHA LOCAL LABELS



WHEN YOU PRESS [Σ+] IN USER MODE, THE HP-41C WILL SEARCH FOR LABEL A WITHIN THE CURRENT PROGRAM.

IF ANY FUNCTION WAS ASSIGNED TO THE [Σ+] KEY FOR EXECUTION IN USER MODE, THE CALCULATOR WOULD NOT SEARCH FOR LABEL A.

ALPHA LOCAL LABELS

Earlier, we learned how to label or name a program with a string of Alpha characters. There are 15 Alpha labels on the HP-41C that have special functions called "LOCAL LABELS". These 15 labels are [LBL] A through [LBL] J and [LBL] a through [LBL] e (shifted A through E). Any time you label a portion of a program or a subroutine with one of these Alpha labels, it is a LOCAL LABEL.

When the HP-41C is in user mode and you press one of the keys in the top two rows (or [GOLD] and a top row key) the calculator immediately begins searching for the corresponding (A through J, a through e) local label within the current program. If the local label is not found, the calculator executes the function printed on the face of, or above, the key.

It is very important to note that when you reassign any other function to the top row locations for execution in user mode, the HP-41C will not perform a local label for that particular location. As an example, if the function FACT was assigned to the [Σ+] key for execution in user mode, the calculator will not search for local label A even if it comes across a GTO LBL A instruction in a program.

CLEARING USER ASSIGNED FUNCTIONS

KEYSTROKES	DISPLAY
[GOLD] [CLX]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA] [ALPHA]	ASN_
[Σ+]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA] [ALPHA]	ASN_
[1/X]	0.0000
[GOLD] [ASN]	ASN_
[ALPHA] [ALPHA]	ASN_
[$\sqrt{X}$]	0.0000

CLEARING USER ASSIGNED FUNCTIONS

Using a program called SPEED, we can better understand the way subroutines and Alpha Local labels can be used.

Our Speed program will compute distance (given rate and time), rate (given distance and time), or time (given distance and rate).

Since we will be using local labels, we must first make sure that no functions are assigned to any top row locations we plan to use. We'll be using Local Labels A, B, and C, so clear fact from [Σ+], SQRT from [1/x], and INT from [$\sqrt{x}$].

WRITING A PROGRAM USING SUBROUTINES AND LOCAL LABELS

KEYSTROKES	DISPLAY	COMMENTS
[GOLD] [ASN]	ASN_	
[ALPHA] PROMPT [ALPHA]	ASN PROMPT_	
[CHS]	0.0000	
[USER]	0.0000	
[PRGM]	00 REG NN	
[GOLD] [GTO] [.] [.]	00 REG NN	
[GOLD] [LBL]	01 LBL _ _	
[ALPHA] SPEED [ALPHA]	01 LBL T SPEED	MAIN PROGRAM
[ALPHA] A, B, OR C? [ALPHA]	02 T A, B, OR C?	
[CHS] (PROMPT)	03 PROMPT	
[GOLD] [LBL]	04 LBL _ _	
[ALPHA] A [ALPHA]	04 LBL A	LOCAL LABEL A
[ALPHA] RATE? [ALPHA]	05 T RATE?	
[CHS] (PROMPT)	06 PROMPT	
[ALPHA] TIME? [ALPHA]	07 T TIME?	
[CHS] (PROMPT)	08 PROMPT	
[X]	09 X	
[GOLD] [RTN]	10 RTN	END OF SUBROUTINE A
[GOLD] [LBL]	11 LBL _ _	
[ALPHA] B [ALPHA]	11 LBL B	LOCAL LABEL B
[ALPHA] DISTANCE? [ALPHA]	12 T DISTANCE?	
[CHS] (PROMPT)	13 PROMPT	
[ALPHA] TIME? [ALPHA]	14 T TIME?	
[CHS] (PROMPT)	15 PROMPT	
[÷]	16 /	
[GOLD] [RTN]	17 RTN	END OF SUBROUTINE B
[GOLD] [LBL]	18 LBL_	
[ALPHA] C [ALPHA]	18 LBL C	LOCAL LABEL C
[ALPHA] DISTANCE? [ALPHA]	19 T DISTANCE?	
[CHS] (PROMPT)	20 PROMPT	
[ALPHA] RATE? [ALPHA]	21 RATE?	
[CHS] (PROMPT)	22 PROMPT	
[÷]	23 /	END OF SUBROUTINE
[GOLD] [GTO] [.] [.]	00 REG NN	END OF MAIN PROGRAM
[PRGM]	0.0000	

WRITING A PROGRAM USING

SUBROUTINES AND LOCAL LABELS

Now that we've cleared the top row locations of the user defined functions, we can load in the "SPEED" program.

Note the use of LOCAL Labels and SUBROUTINES.

RUNNING THE "SPEED" PROGRAM

A = DISTANCE B = RATE C = TIME

INSTRUCTIONS	KEYSTROKES	DISPLAY
DISTANCE = 300	[XEQ]	XEQ _ _
RATE = 60	[ALPHA] SPEED [ALPHA]	A, B OR C?
TIME = ?	[$\sqrt{\overline{X}}$] (C)	DISTANCE?
	300	300_
	[R/S]	RATE?
	60	60_
	[R/S]	5.0000
RATE = 120	[XEQ]	XEQ _ _
TIME = 7	[ALPHA] SPEED [ALPHA]	A, B OR C?
DISTANCE = ?	[Σ+] (A)	RATE?
	120	120_
	[R/S]	TIME?
	7	7_
	[R/S]	840.0000
DISTANCE = 400	[XEQ]	XEQ _ _
TIME = 8	[ALPHA] SPEED [ALPHA]	A, B OR C?
RATE = ?	[1/X] (B)	DISTANCE?
	400	400_
	[R/S]	TIME?
	8	8_
	[R/S]	50.000

RUNNING THE PROGRAM

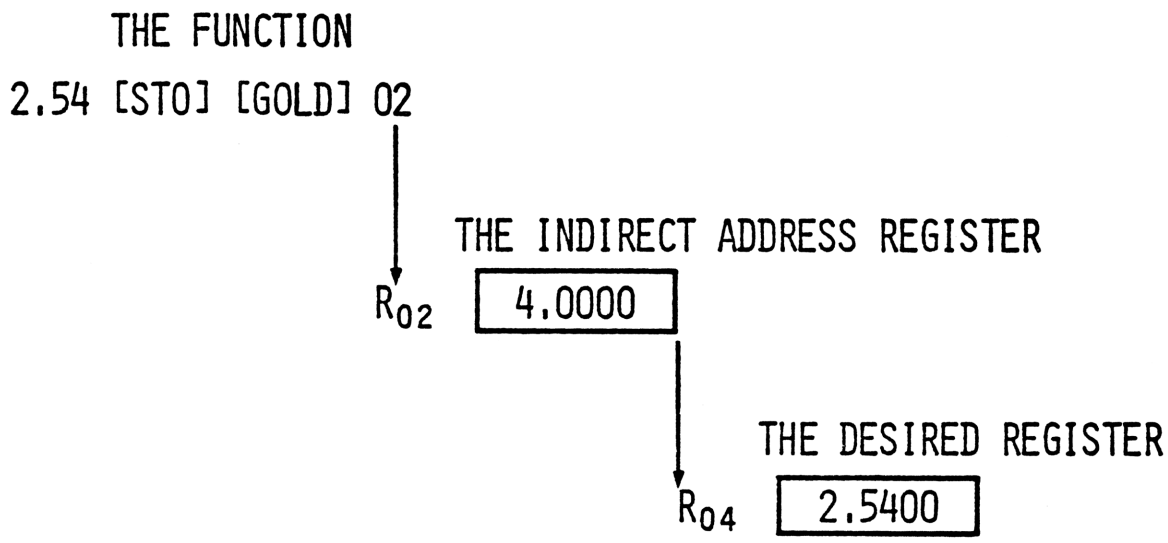
Now that we've written and entered the program "SPEED", try several sample problems.

To compute distance, use Subroutine A ([Σ+] in user mode).

To compute rate, use Subroutine B ([1/x] in user mode).

To compute time of travel, use Subroutine C ([$\sqrt{x}$] in user mode).

INDIRECT ADDRESSING



REGISTER 02 (R_{02}) CONTAINS THE REGISTER ADDRESS (R_{04}) ON WHICH THE FUNCTION IS TO BE PERFORMED.

THE HP-41C PERFORMS THE FUNCTION (STORE 2.54) ON THE INDIRECTLY ADDRESSED REGISTER (R_{04}).

INDIRECT OPERATIONS

An important feature of the HP-41C is the numerous INDIRECT OPERATIONS the calculator can perform.

Any storage register in the HP-41C can be used for indirect operations. This capability greatly expands the power and utility of your HP-41C.

An Indirect Address is selected by following a function, store or recall for example, with the shift key [GOLD] and then a register address. The function is then performed on the indirect, or addressed register.

INDIRECT STORE AND RECALL

STORE 2.54 IN REGISTER 04 USING INDIRECT ADDRESSING

KEYSTROKES	DISPLAY
4	4_
[STO] 02	4.0000
2.54	2.54
[STO] [GOLD]	STO IND _ _
02	2.5400
[GOLD] [CLX]	0.0000
[RCL] 04	2.5400
[GOLD] [CLX]	0.0000

RECALL 2.54 FROM REGISTER 04 USING INDIRECT ADDRESSING

[RCL] [GOLD]	RCL IND _ _
02	2.5400
[GOLD] [CLX]	0.0000

RECALL INDIRECT ADDRESS FROM REGISTER 02

[RCL] 02	4.0000
[GOLD] [CLX]	0.0000

INDIRECT STORE AND RECALL

To better understand the power of indirect operations, let's practice using INDIRECT STORE AND INDIRECT RECALL.

We'll use numerical information in our example, but it should be noted that Alpha characters can also be indirectly stored and recalled.

In all cases, only the absolute value of the integer portion of the register address is used by the calculator. As an example, if -4.9366 was stored in the address register, the calculator would perform the indirect operation on register 04.

INDIRECT FUNCTION CONTROL

KEYSTROKES	DISPLAY
[PGRM]	00 REG NN
[GOLD] [GTO] [.] [.]	00 REG NN
[GOLD] [LBL]	01 LBL_
[ALPHA] SONG [ALPHA]	01 LBL T SONG
.009	02 .009_
[STO] 01	03 STO 01
9	04 9_
[STO] 02	05 STO 02
[GOLD] [LBL] 01	06 LBL 01
[XEQ]	07 XEQ _ _
[ALPHA] TONE [ALPHA]	07 TONE_
[GOLD] 01	TONE IND 01
[GOLD] [ISG] 01	08 ISG 01
[GOLD] [GTO] 01	09 GTO 01
[GOLD] [LBL] 02	10 LBL 02
[XEQ]	11 XEQ _ _
[ALPHA] TONE [ALPHA]	11 TONE_
[GOLD] 02	TONE IND 02
[XEQ]	12 XEQ _ _
[ALPHA] DSE [ALPHA]	12 DSE _ _
02	12 DSE 02
[GOLD] [GTO] 02	13 GTO 02
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	0.0000
[XEQ]	XEQ _ _
[ALPHA] SONG [ALPHA]	9.0000

INDIRECT FUNCTION CONTROL

Now that we've seen how indirect addressing works, let's progress a little and see how some of the other indirect features work in programs.

The HP-41C has a function called [TONE], which emits a single beep upon execution. The user can choose from 10 different tonal sounds.

Functions requiring the input of an operating specification, like [TONE], can use indirect addressing to specify how the function is to operate.

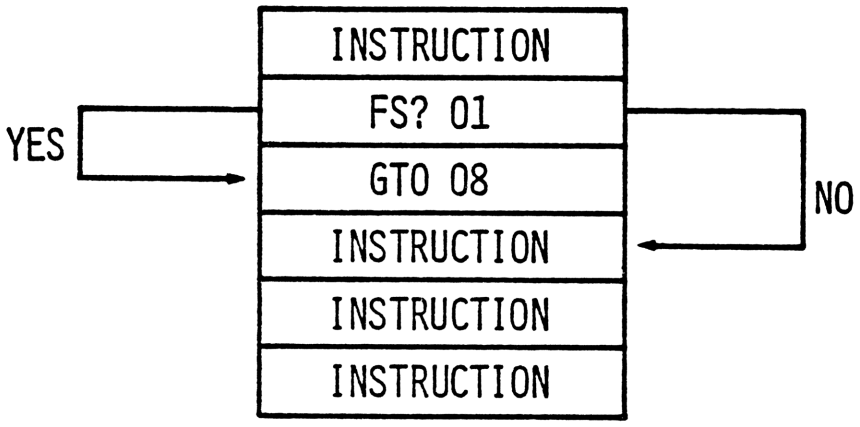
[TONE] requires a number from 0 to 9 to specify the desired tonal sound. Using indirect addressing, you can store the tone specification number in a register, and then use increment and decrement to help you run through all 10 tonal sounds.

Key in the program "Song" and then execute it. The program uses two controlled loops to specify a number to be used with the [TONE] function. The program counts from 0 to 9 and controls the second loop using the decrement function [DSE].

FLAGS

FUNCTION		EXECUTION
FS?	_	FLAG SET?
FS?C	-	FLAG SET? CLEAR
FC?	-	FLAG CLEAR?
FC?C	-	FLAG CLEAR? CLEAR

IS FLAG 01 SET?



REMEMBER THE "DO IF TRUE" RULE

FLAGS

FLAGS are an important programming tool in your calculator.

A flag actually is a memory that can either be set or clear. A running program can then test the flag later in the program and make a decision, depending upon whether the flag was set or clear.

There are 30 "USER" flags (numbered 00 through 29) available in your HP-41C. In addition, there are 26 "SYSTEM" flags (numbered 30 through 55) that have limited uses in programs, but which when set can control various system functions, such as printer or card reader operations.

A flag's operation is similar to a conditional test. When the HP-41C comes across a flag in a program, it tests the flag's status in one of four ways: FLAG SET?, FLAG SET? AND CLEAR, FLAG CLEAR?, AND FLAG CLEAR? AND CLEAR. If the answer to the test question is true, the calculator executes the next program line (this is the do if true rule again). If the answer to the test question is false, the calculator skips a program step and then continues executing.

FLAG USE IN A PROGRAM

KEYSTROKES	DISPLAY
[PRGM]	00 REG NN
[GOLD] [GTO] [.] [.]	00 REG NN
[GOLD] [LBL]	01 LBL _ _
[ALPHA] FLAG [ALPHA]	01 LBL T FLAG
[GOLD] [LBL] 01	02 LBL 01
[ALPHA] SET	03 T SET
[GOLD] [AVIEW] [ALPHA]	04 AVIEW
[XEQ]	05 XEQ _ _
[ALPHA] PSE [ALPHA]	05 PSE
[GOLD] [CF] 00	06 CF 00
[GOLD] [LBL] 02	07 LBL 02
[GOLD] [FS?] 00	08 FS? 00
[GOLD] [GTO] 01	09 GTO 01
[ALPHA] CLEAR	10 T CLEAR
[GOLD] [AVIEW] [ALPHA]	11 AVIEW
[XEQ]	12 XEQ _ _
[ALPHA] PSE [ALPHA]	12 PSE
[GOLD] [SF] 00	13 SF 00
[GOLD] [GTO] 02	14 GTO 02
[GOLD] [GTO] [.] [.]	00 REG NN
[PRGM]	9.0000
[XEQ] [ALPHA] FLAG [ALPHA]	SET
	CLEAR
	SET
	CLEAR
	.
	.
	.
	.
[R/S]	9.0000
[GOLD] [CLX]	0.0000

FLAG USE IN A PROGRAM

To illustrate the operation of a FLAG, let's write a program with an infinite loop that will alternately display SET and CLEAR by changing and testing the status of Flag 00.

Notice that the annunciator for Flag 00 turns ON and OFF as the flag status changes. Flags 00-04 all have annunciators, which turn ON as the flag is set.

HEWLETT-PACKARD 41C

- THE HP-41C IS A SIMPLE-TO-USE HANDHELD CALCULATOR . . . IT COMMUNICATES WITH THE USER
- THE HP-41C HAS AN EXCITING NEW KEYBOARD DESIGN . . . . YOU CAN CUSTOMIZE THE CALCULATOR TO YOUR OWN DESIGN
- THE HP-41C HAS A UNIQUE COMBINATION OF PROGRAMMING EASE AND SOPHISTICATION . . . IT IS EASY TO PROGRAM
- THE HP-41C HAS PROGRAMMING POWER . . . IT CAN SOLVE PROFESSIONAL PROBLEMS
- THE HP-41C HAS CONTINUOUS MEMORY . . . IT SAVES DATA, PROGRAMS, AND KEY ASSIGNMENTS EVEN WHEN TURNED OFF
    - SAVES DATA AND PROGRAMS
    - SAVES PERSONALIZED ASSIGNMENTS
    - LOW POWER CONSUMPTION

THE HP-41C HAS CONTINUOUS MEMORY . . . IT SAVES DATA, PROGRAMS, AND KEY ASSIGNMENTS EVEN WHEN TURNED OFF

The value of CONTINUOUS MEMORY is truly appreciated when we have many programs and keyboard assignments contained within the calculator.

HEWLETT-PACKARD CONTINUOUS MEMORY

SAVES DATA AND PROGRAMS . . . SAVES PERSONALIZED ASSIGNMENTS . . . LOW POWER CONSUMPTION

Hewlett-Packard Continuous Memory

Hewlett-Packard continuous memory assures memory retention while consuming little battery power.

HEWLETT-PACKARD 41C

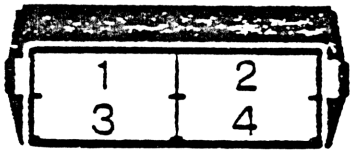
- THE HP-41C IS AN EXPANDABLE SYSTEM . . . IT WILL MEET YOUR NEEDS TODAY AND IN THE FUTURE

    - MEMORY MODULES
    - APPLICATION MODULES
    - "EXTRA-SMART" CARD READER
    - PRINTER
    - WAND

THE HP-41C IS AN EXPANDABLE SYSTEM

To this point, we have learned about the HP-41C as a calculator. We will now learn about the HP-41C as a system. Besides the calculator itself, the HP-41C system features memory modules, application modules, an "extra smart" cardreader, a whisper quiet printer, and an optical wand.

HP-41C PERIPHERAL IO PORTS



- ANY COMBINATION OF PERIPHERALS MAY BE USED
- TURN CALCULATOR OFF BEFORE REMOVING OR INSERTING ANY PERIPHERAL

HP-41C PERIPHERAL IO PORTS

The HP-41C features four identical IO Ports. Any combination of HP-41C peripherals can be plugged into these ports, thus allowing the user to customize a system to fit personal needs.

The calculator must be turned off before removing or inserting any peripheral.

It is Hewlett-Packard policy not to release any information regarding the internal operations of our products. We therefore cannot provide any information regarding IO Port circuitry.

HEWLETT-PACKARD 82106A

MEMORY MODULES ADD MEMORY CAPACITY

- 64 REGISTERS PER MODULE
- UP TO 319 REGISTERS TOTAL
- CONTINUOUS MEMORY

MEMORY	MAXIMUM DATA STORAGE REGISTERS	INITIAL REGISTER ALLOCATION		MAXIMUM PROGRAM MEMORY REGISTERS
		DATA STORAGE	PROGRAM MEMORY	
INITIAL CONFIGURATION	63	17	46	63 (445 BYTES)
+1 MEMORY MODULE	127	81	46	127 (893 BYTES)
+2 MEMORY MODULES	191	145	46	191 (1341 BYTES)
+3 MEMORY MODULES	255	209	46	255 (1789 BYTES)
+4 MEMORY MODULES	319	273	46	319 (2237 BYTES)

MEMORY MODULES ADD MEMORY CAPACITY

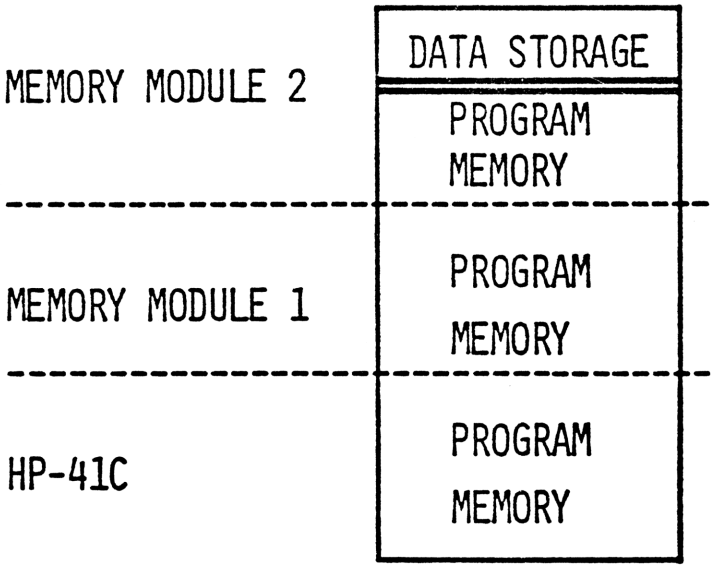
To meet expanded programming and data storage needs, Hewlett-Packard designed the HP-82106A memory module. Once plugged into the HP-41C, each memory module provides an additional 64 registers of program memory or data storage registers, or any combination up to four memory modules can be added to the HP-41C system at one time, thus providing a total system capacity of 319 registers (over 2,000 lines of program memory).

All of the additional memory provided by memory modules, like the internal memory of the HP-41C, is continuous memory. As long as the memory module is plugged into the HP-41C, its contents are preserved for later use, even while the HP-41C is turned off.

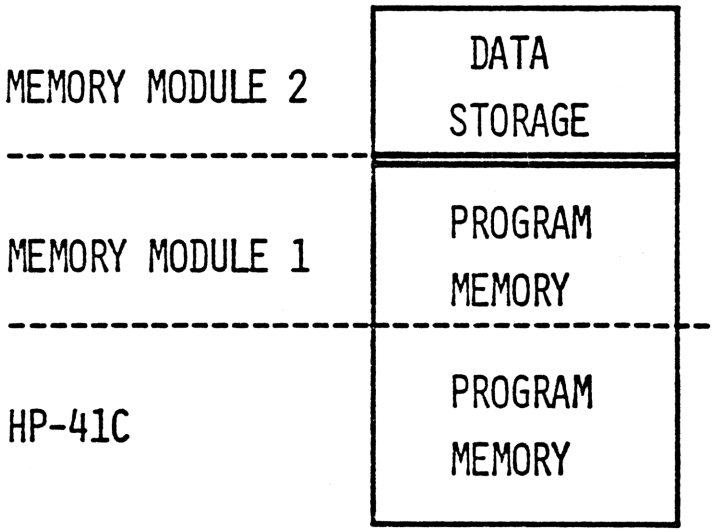
As in the internal memory of the HP-41C, the plug-in memory can be allocated in any combination of program memory and data storage, using the [SIZE] function. When you first plug in memory modules and then turn on the HP-41C, the registers in those modules become allocated as data storage registers. The chart on the slide identifies initial memory configuration and maximum data storage and program memory capacity given any number of memory modules. Remember that data storage registers $R_{(100)}$ through $R_{(319)}$ are addressed indirectly.

SYSTEM MEMORY PARTITIONING

"PERMANENT END" PARTITION DENOTED BY ═══



SIZE 017 - PARTITION RESIDES IN MEMORY MODULE 2. REMOVAL OF EITHER MEMORY MODULE WILL CLEAR THE ENTIRE SYSTEM.



SIZE 064 - PARTITION RESIDES AT TOP OF MEMORY MODULE 1. MEMORY MODULE 2 CAN BE REMOVED. REMOVAL OF MEMORY MODULE 1 WILL CLEAR THE ENTIRE SYSTEM.

WHEN REMOVING 1 MEMORY MODULE, EXECUTE [SIZE] 064
WHEN REMOVING 2 MEMORY MODULES, EXECUTE [SIZE] 128
WHEN REMOVING 3 MEMORY MODULES, EXECUTE [SIZE] 192
WHEN REMOVING 4 MEMORY MODULES, EXECUTE [SIZE] 256

SYSTEM MEMORY PARTITIONING

It is very important to understand user system memory management while using memory modules.

As we've learned, memory can be allocated between data storage and program memory, using the [SIZE] function. When we execute [SIZE], we establish a "Permanent End" at the bottom of allocated memory. This "Permanent End" or partition between data storage and program memory, cannot be deleted, nor can program instructions be inserted beyond it.

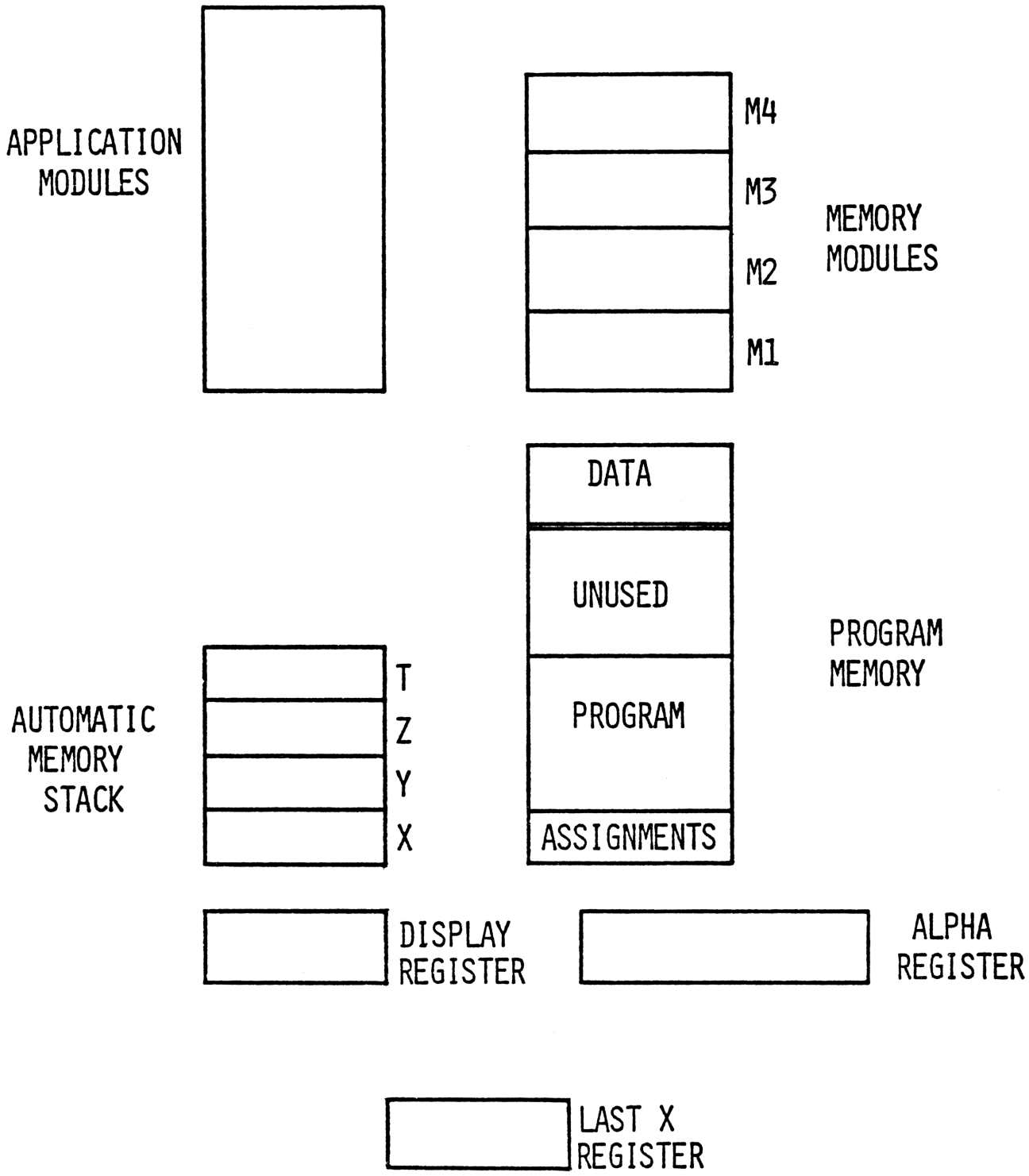
It is important that the user know the location of the "Permanent End" partition when removing memory modules, for if it resides in a module that is removed, all system memory will be lost. In order to assure that this won't happen, the following should be done:

If you wish to remove 1 memory module, execute [SIZE] 064
If you wish to remove 2 memory module, execute [SIZE] 128
If you wish to remove 3 memory module, execute [SIZE] 192
If you wish to remove 4 memory module, execute [SIZE] 256

If a portion of the memory module to be removed currently contains program instructions, the calculator will display "PACKING", "TRY AGAIN". If this happens, removal of the memory module(s) will result in losing all system memory.

When removing memory modules, always remove the module in the highest numbered port first. Remember also that the calculator must be turned off when inserting or removing memory modules.

MEMORY MAP


APPLICATION MODULES
M4
M3
M2
M1
MEMORY MODULES
DATA
UNUSED
PROGRAM
ASSIGNMENTS
PROGRAM MEMORY
AUTOMATIC MEMORY STACK
T
Z
Y
X
DISPLAY REGISTER
ALPHA REGISTER
LAST X REGISTER

MEMORY MAP

Now that we've learned about a number of HP-41C display and memory retention capabilities, we can use the memory map to piece together their relationships with one another.

HEWLETT-PACKARD 82104A

- "EXTRA-SMART" CARD READER FOR PERMANENT STORAGE OF PROGRAMS AND DATA

    - PROMPTS FOR CARDS
    - READ IN ANY ORDER
    - RECORDS DATA OR PROGRAMS
    - HP-67/97 COMPATIBILITY
    - PRIVATE PROGRAMS
    - RECORDS KEY ASSIGNMENTS

HP-82104A - AN "EXTRA SMART" CARD READER

One of the HP-41C's most exciting peripherals is the HP-82104A, an "extra smart" card reader.

The 82104A has a number of outstanding features (see slide).

READING A PROGRAM

- TO AVOID OVERWRITING PROGRAM MEMORY
  KEY IN [GTO] [.] [.] BEFORE READING A CARD

- CARDREADER WILL PROMPT FOR LOWEST TRACK

READY KK OF NN

KK - LOWEST UNREAD TRACK

NN - TOTAL NUMBER OF TRACKS

READING A PROGRAM

Programs read in will replace the last program in program memory. To avoid overwriting key GTO [.] [.] before loading card.

Card reader will prompt for the lowest track not yet read. It will also tell you how many tracks are to be read.

Ready kk of nn

kk = lowest unread track

nn = number of tracks

WRITING ON A CARD

- CARD READER WILL PROMPT FOR CARDS

  READY KK OF NN

  KK - TRACK BEING RECORDED

  NN - TOTAL NUMBER OF TRACKS REQUIRED

- HP-41C WILL DISPLAY "CARD ERROR" WHEN CARD IS IMPROPERLY LOADED

- KEY ASSIGNMENTS BECOME ACTIVE IF HP-41C IS IN [USER] MODE

WRITING ON A CARD

Key in a program

While in program mode, insert card. The HP-41C will prompt with

Ready kk of nn

kk - track being recorded
nn - total number of tracks required

If the card is not read because of a card reader error, the HP-41C will prompt with "CARD ERROR".

If the HP-41C is in [USER] mode, all key assignments in place when the card was recorded become active.

WRITING ON PROTECTED CARDS

- A CARD IS PROTECTED BY CLIPPING A CORNER

- TO OVERWRITE A PROTECTED CARD, SET FLAG 14 BEFORE WRITING ON CLIPPED CARD

WRITING ON PROTECTED CARDS

To overwrite a protected card (corner clipped) you set flag 14 before writing.

READING OR WRITING DATA

- DATA IS RECORDED ON CARDS WITH [WDTA] FUNCTION
- DATA MAY BE POSITIONED IN MEMORY USING [WDTAX] FUNCTION

bbb.eee - CONTROL NUMBER FORMAT (PLACED IN X REGISTER)

bbb - BEGINNING STORAGE REGISTER

eee - ENDING STORAGE REGISTER

KEYSTROKES	DISPLAY
1	1_
[STO] 00	1.0000
2	2_
[STO] 01	2.0000
3	3_
[STO] 02	3.0000
4	4_
[STO] 03	4.0000
.003	.003_
[XEQ] [ALPHA] WDTAX [ALPHA]	RDY 01 OF 01
(LOAD CARD)	0.0030
[XEQ] [ALPHA] CLRG [ALPHA]	0.0030
(READ CARD)	0.0030
[RCL] 00	1.0000
[RCL] 01	2.0000
[RCL] 02	3.0000
[RCL] 03	4.0000
[RCL] 04	0.0000

READING OR WRITING DATA

Data may be recorded on cards just like programs using the [WDTA] function.

A new function is the [WDTAX] function. This function takes the number in the X register as a control number when reading or writing data. The number in X must be formatted like bbb.eee where bbb is the beginning register and eee is the ending register. This function allows you to position data anywhere in available data storage memory.

CARD VERIFICATION

- UPON EXECUTION OF [VER], HP-41C WILL DISPLAY

  TYPE *t* TR NN

  t- ONE OF THE FOLLOWING

P	PROGRAM CARD
D	DATA CARD
S	STATUS CARD
A	"WRITE ALL" CARD
7P	HP-67/97 PROGRAM CARD
7D	HP-67/97 DATA CARD

  NN - TRACK NUMBER

- HP-41C WILL DISPLAY "CARD ERR" TO INDICATE IMPROPER OR BLANK CARD

- HP-41C WILL DISPLAY "CHECKSUM ERR" WHEN IT FINDS AN ERROR ON A "PROPER" CARD

CARD VERIFICATION

The "VER" function allows the user to verify the contents of program cards.

After executing the verify function, and reading a card, the 41C responds with

Type t TR nn
t - see slide for code
nn - track number of card

If the card is blank or not a proper card, the 41C will say CARD ERR.

If it analyzes the card and finds an error, it will respond with

CHECKSUM ERR

WALL - W STS - W PRV

WALL - THE WRITE ALL FUNCTION ALLOWS USER TO WRITE ENTIRE CONTENTS OF HP-41C ONTO A SERIES OF CARDS

W STS - THE WRITE STATUS FUNCTION ALLOWS THE USER TO WRITE ENTIRE STATUS OF HP-41C ONTO CARDS

W PRV - THE WRITE PRIVATE FUNCTION ALLOWS THE USER TO PROTECT SOFTWARE

WALL - WRITE ALL

This allows the user to write the entire contents of the 41C memory onto a series of cards. This includes program and data memory, flag status, all mode statuses, the stack and Last X, and key assignments.

WSTS - WRITE STATUS

This allows the user to write all of the calculator status onto cards.

WPRV - WRITE PRIVATE

This allows the user to protect software. Once a card is made Private, it can only be read and then executed. It cannot be reviewed or single stepped. This includes the user's own calculator.

AUTO EXECUTION

- FOR AUTO EXECUTION, SET FLAG 11, THEN RECORD CARD

- CALCULATOR BEGINS EXECUTION AT LINE 000 AS SOON AS CARD READING IS COMPLETE

- HP-41C WILL "BEEP" UPON AUTO START

AUTO EXECUTION

By recording a card with flag 11 set, the calculator will begin execution at line 000 as soon as the read is complete.

This allows the program to initialize the calculator before calculations begin.

The calculator beeps to let the user know that it is Auto Starting.

MERGE INSTRUCTION

```
000 LBL T TIME
    .
    .
    .
    .
082 MRG
083 T DAYLIGHT TIME=
    .
    .
    .
100     END
```

```
083 T STANDARD TIME=
    .
    .
    .
    .
    .
120     END
```

MERGE INSTRUCTION

The MRG instruction is a programmable function. When encountered, the calculator prompts with CARD. The new program is added after the MRG and replaces the program lines up to the first END statement. The MRG statement must be in the last program in memory.

HP-67/97 SOFTWARE COMPATIBILITY

- ALL HP-67/97 SOFTWARE CAN BE USED WITH THE HP-41C AND HP-82104A

- HP-82104A MANUAL DOCUMENTS IMPORTANT POINTS REGARDING COMPATIBILITY

HP-67/97 SOFTWARE COMPATIBILITY

The HP-41C card reader can translate programs written on any HP-67/97. In some cases minor changes to the program will be required to make them run conveniently.

The card reader manual lists any problems with HP developed 67/97 software.

OTHER CARD READER KEY FEATURES

- PRIVATE PROGRAMS PROVIDE SOFTWARE SECURITY
- RECORDING KEY ASSIGNMENTS SAVES TIME
- CARD READER ENHANCES USER ABILITY TO SOLVE PROBLEMS

OTHER CARD READER KEY FEATURES

Software Security

An important new feature offered by the HP-82106A is the ability to maintain software security, through the use of the "PRIVATE" function. Once recorded, the card can only be read and then executed. The program cannot be reviewed or modified.

Recorded Key Assignments

The card reader automatically records key assignments for any label within a program. The user simply has to read in a program while in [USER] mode, and key assignments become active.

Custom Solutions

Together, the card reader features greatly enhance the ability of the user to solve problems.

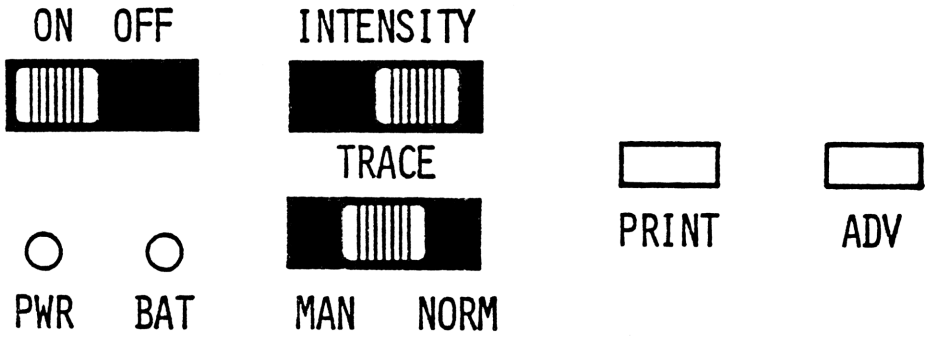
HEWLETT-PACKARD 82143A

- "WHISPER-QUIET" THERMAL PRINTER FOR HARD COPIES OF RESULTS
    - BATTERY PORTABLE
    - ADJUSTABLE PRINT INTENSITY
    - FULL ALPHANUMERIC CAPABILITY
    - DOUBLE WIDE CHARACTERS
    - CHARACTER PLOTTING
    - SPECIAL CHARACTERS

HP-82143A - A "WHISPER QUIET" THERMAL PRINTER

The HP-82143A is an exceptional printer that greatly enhances the power and flexibility of the HP-41C system. (See slide for key features)

PRINTER CONTROLS



PRINTER CONTROLS

ON/OFF Switch turns on the Printer. When turned on the P.WR light should be lit. If the BAT light is on, the batteries are low. You should immediately plug in the A.C. Adapter.

Print Intensity is a five position switch which allows the user to adjust for different paper and variable ambient temperatures.

Man, Norm, Trace is a mode switch; it will be discussed later.

ADV is the paper advance key. When the HP-41C is in PRGM mode pushing this key will place an ADV into program memory.

PRINT is the print key. When the HP-41C is in ALPHA mode the [PRINT] acts as a print alpha; if in normal mode the [PRINT] acts as PRTX; if in PRGM mode it puts a PRT into program memory.

MANUAL MODE

TRACE

MAN  NORM

CALCULATOR KEYSTROKES	PRINTER OUTPUT
25	
(PRESS PRINT ON PRINTER)	25.0000 ***
[XEQ] [ALPHA] PRX [ALPHA]	25.0000 ***
[GOLD] [VIEW] [.] X	25.0000
(R/S)	
[ALPHA] ABC	
(PRESS PRINT ON PRINTER)	ABC
[GOLD] [AVIEW]	ABC
(R/S)	
[ALPHA]	

MANUAL MODE

When in manual mode the Printer will only respond to Print functions. Program listings appear left-justified.

NORMAL MODE

TRACE

MAN  NORM

HP-41C KEYSTROKES	PRINTER OUTPUT	
25		
[ENTER ↑]	25.0000	ENTER↑
5		
X	5.0000	*
100		
+	100.0000	+
(PRESS PRINT ON PRINTER)	225.0000	***

NORMAL MODE

The printer keeps track of numbers keyed in, functions executed from the keyboard, and any output from print functions. Program listings are printed right-justified.

TRACE MODE

TRACE

MAN  NORM

HP-41C KEYSTROKES	PRINTER OUTPUT
25	25.0000 ENTER↑
[ENTER↑]	5.0000 *
5	125.0000 ***
X	-PRINTER-
CATALOG 2	ACA
	ACCHR
	ACCOL
	.
	.
	.
	.
	SKPCOL
	STKPLOT

TRACE MODE

The printer keeps complete records of input, output, and functions. Programs are printed in a unique "packed" form.

Trace mode also provides a simple way to list any of the catalogs.

FULL ALPHANUMERIC PRINTING

TRACE

MAN  NORM

HP-41C KEYSTROKES	PRINTER OUTPUT
[GOLD] [ASN] [ALPHA] PRA [ALPHA] [LN]	
[USER]	
[ALPHA] PRINTER [ALPHA]	PRINTER
[LN] (PRA)	
[GOLD] [SF] 13	printer
[LN] (PRA)	
[GOLD] [CF] 13	
[GOLD] [SF] 12	PRINTER
[LN] (PRA)	
[GOLD] [SF] 13	printer
[LN] (PRA)	
[GOLD] [CF] 13	
[GOLD] [CF] 12	

FULL ALPHA-NUMERIC PRINTING

The HP-82143A offers full alphabetic printing for messages.

Upper and lower case printing

SF 13 prints lower case

Double wide printing is available by SF 12.

PLOT FUNCTION

TRACE

MAN  NORM

HP-41C KEYSTROKES

[GTO] [.] [.]

[PRGM]

[LBL] [ALPHA] SINE [ALPHA]

SIN

[RTN]

[PRGM]

[XEQ]

[ALPHA] PRPLOT [ALPHA]

SINE
[R/S]

1 [.] 1 [CHS]
[R/S]

0
[R/S]

0
[R/S]

360
[R/S]

18
[R/S]

PRINTER OUTPUT

```
          XROM "PRPLOT"
NAME ?
SINE
                    RUN
Y MIN ?
         -1.1000    RUN
Y MAX ?
          1.1000    RUN
AXIS ?
          0.0000    RUN
X MIN ?
          0.0000    RUN
X MAX ?
        360.0000    RUN
X INC ?
         18.0000    RUN

     PLOT OF SINE
   X (UNITS= 1.) →
   Y (UNITS= 1.) ↑
    -1.10          1.10
           0 00
    |--------|--------|
  0          *
 18.         |  *
 36.         |     *
 54.         |        *
 72.         |          *
 90.         |          *
108.         |          *
126          |        *
144          |     *
162.         |  *
180.         *
198        *  |
216.     *    |
234    *      |
252.  *       |
270  *        |
288  *        |
306.   *      |
324.     *    |
342.       *  |
360.         *
```

PLOT FUNCTION

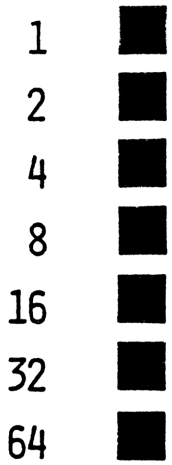
The [PRPLOT] function provides simple plots.

First load program to be plotted.

Execute PRPLOT

```
                    XROM        *PRPLOT*
NAME?
SINE                              RUN
Y MIN?
              -1.1000             RUN
Y MAX?
               1.1000             RUN
AXIS?
               0.0000             RUN
X MIN?
               0.0000             RUN
X MAX?
             360.000              RUN
X INC?
              18.000              RUN
```

SPECIAL CHARACTERS



SPECIAL CHARACTERS

An exciting new feature offered by the HP-41C and its "Whisper Quiet" printer is the ability to create and print special characters.

Each print column contains seven dots, and a special code indicates which dots are to be printed (see slide).

ACCUMULATE COLUMN

HP-41C KEYSTROKES

[USER]
[GOLD] [ASN] [ALPHA] PRBUF [ALPHA] [Σ+]
[GOLD] [ASN] [ALPHA] ACCOL [ALPHA] [1/X]
[PRBUF] (Σ+)
0 [ACCOL] (1/X)
127 [ACCOL] (1/X)
65 [ACCOL] (1/X)
93 [ACCOL] (1/X)
85 [ACCOL] (1/X)
93 [ACCOL] (1/X)
65 [ACCOL] (1/X)
127 [ACCOL] (1/X)
0 [ACCOL] (1/X)
[PRBUF] (Σ+)

ACCUMULATE COLUMN

The printer has a print buffer (register which can hold up to 43 columns). You can use the ACCOL function to accumulate columns into the buffer. The PRBUF prints out the buffer.

BUILD SPECIAL

TRACE

MAN  NORM

HP-41C KEYSTROKES

[USER]
[GOLD] [ASN] [ALPHA] BLDSPEC [ALPHA] [1/X]
[GOLD] [ASN] [ALPHA] ACSPEC [ALPHA] [$\sqrt{X}$]
0 [ENTER]
0 [BLDSPEC] (1/X)
28 [BLDSPEC] (1/X)
85 [BLDSPEC] (1/X)
127 [BLDSPEC] (1/X)
85 [BLDSPEC] (1/X)
28 [BLDSPEC] (1/X)
0 [BLDSPEC] (1/X)
[ACSPEC] ($\sqrt{X}$)
[PRBUF] (Σ+)
[STO] 01
[CLX]
[RCL] 01
[ACSPEC] ($\sqrt{X}$)
[PRBUF] (Σ+)

REPEATED SPECIAL CHARACTER USE

You can also build your own special characters for repeated use.

The basic method is the same as ACCOL except you use two new functions and they can only be seven characters long.

BLDSPEC is a function which builds up a special character.

ACSPEC is a function which puts the special character into the printer buffer.

You can use regular STO and RCL to store special characters for use later.

HP-41C APPLICATION MODULES

APPLICATION MODULES PROVIDE SOLUTIONS TO PROFESSIONAL PROBLEMS

HP-41C APPLICATION MODULES

HP-41C Application Modules provide solutions to professional problems.

APPLICATION ROMS

- AVIATION
- CLINICAL LAB AND NUCLEAR MEDICINE
- CIRCUIT ANALYSIS
- FINANCIAL DECISIONS
- GAMES
- HOME MANAGEMENT
- MACHINE DESIGN
- MATHEMATICS
- NAVIGATION
- REAL ESTATE
- SECURITIES
- STATISTICS
- STRESS ANALYSIS
- STRUCTURAL ANALYSIS
- SURVEYING
- THERMAL AND TRANSPORT SCIENCE

APPLICATION ROMS

Every preprogrammed application module turns the HP-41C into an answer machine for a particular discipline: a specific calculator for specific needs (over 15 application modules are available - see slide).

HEWLETT-PACKARD 82153A

- THE OPTICAL WAND FOR QUICK ERROR-FREE ENTRY OF PROGRAMS OR DATA
    - BAR CODE
    - PAPER KEYBOARD
    - PAPER ROMS

HP-82153A - OPTICAL WAND

The Optical Wand allows the user to quickly load programs without errors.

HP-41C FUNCTION LIST

The following is a list of all functions available on the HP-41C (not including printer and card reader functions).

FUNCTION NAME	DISPLAY	KEYSTROKES
* PLUS	+	[+]
* MINUS	−	[-]
* TIMES	*	[x]
* DIVIDE	/	[÷]
* RECIPROCAL	1/X	[1/x]
* 10^x	10↑X	f/g 10^x
* ABSOLUTE VALUE	ABS	[XEQ] [ALPHA] ABS [ALPHA]
* ARC COSINE (COS^{-1})	ACOS	f/g COS^{-1}
PAPER ADVANCE	ADV	[XEQ] [ALPHA] ADV [ALPHA]
ALPHA OFF	AOFF	[XEQ] [ALPHA] AOFF [ALPHA]
ALPHA ON	AON	[XEQ] [ALPHA] AON [ALPHA]
* ALPHA RECALL	ARCL	[ALPHA] f/g [RCL]
ALPHA SHIFT	ASHF	[XEQ] [ALPHA] ASHF [ALPHA]
* ARC SINE (SIN^{-1})	ASIN	f/g SIN^{-1}
* ASSIGN	ASN	f/g ASN
* ALPHA STORE	ASTO	[ALPHA] f/g [STO]
* ARC TANGENT (TAN^{-1})	ATAN	f/g TAN^{-1}
* ALPHA VIEW	AVIEW	[ALPHA] f/g VIEW
* BEEP	BEEP	f/g BEEP
* BACK STEP	BST	f/g BST
* CATALOG	CAT	f/g CATALOG
* CLEAR FLAG	CF	f/g CF
* CHANGE SIGN	CHS	[CHS]
* CLEAR ALPHA	CLA	[ALPHA] f/g CLX/A
CLEAR DISPLAY	CLD	[XEQ] [ALPHA] CLD [ALPHA]
CLEAR PROGRAM	CLP	[XEQ] [ALPHA] CLP [ALPHA]
CLEAR REGISTERS	CLRG	[XEQ] [ALPHA] CLRG [ALPHA]
* CLEAR STATISTICAL REGISTERS	CLΣ	f/g CLΣ
CLEAR STACK	CLST	[XEQ] [ALPHA] CLST [ALPHA]
* CLEAR X	CLX	f/g CLX/A
COPY	COPY	[XEQ] [ALPHA] COPY [ALPHA]
* COSINE	COS	[COS]
DEGREES INTO RADIANS	D-R	[XEQ] [ALPHA] D f/g [-] R [ALPHA]
OCTAL TO DECIMAL	DEC	[XEQ] [ALPHA] DEC [ALPHA]
DEGREES MODE	DEG	[XEQ] [ALPHA] DEG [ALPHA]
DELETE	DEL	[XEQ] [ALPHA] DEL [ALPHA]
DECREMENT, SKIP IF EQUAL	DSE	[XEQ] [ALPHA] DSE [ALPHA]
END	END	[XEQ] [ALPHA] END [ALPHA]
* ENGINEERING NOTATION	ENG	f/g ENG
* ENTER	ENTER↑	[ENTER↑]
* e^x (EXPONENTIAL FUNCTION)	E↑X	f/g e^x
$e^{(x-1)}$	E↑X-1	[ALPHA] E f/g [ENTER↑] X f/g [-] f/g [1] [ALPHA]
FACTORIAL	FACT	[XEQ] [ALPHA] FACT [ALPHA]
IS FLAG CLEAR?	FC?	[XEQ] [ALPHA] FC? [ALPHA]
IS FLAG CLEAR? CLEAR	FC?C	[XEQ] [ALPHA] FC?C [ALPHA]
* FIX DISPLAY	FIX	f/g FIX
FRACTIONAL PORTION	FRC	[XEQ] [ALPHA] FRC [ALPHA]
* IS FLAG SET?	FS?	f/g FS?
IS FLAG SET? CLEAR	FS?C	[XEQ] [ALPHA] FS?C [ALPHA]

FUNCTION NAME	DISPLAY	KEYSTROKES
GRADIANS MODE	GRAD	[XEQ] [ALPHA] GRAD [ALPHA]
* GO TO	GTO	ℓ/∫ GTO
CONVERT TO HOURS, MINUTES, SECONDS	HMS	[XEQ] [ALPHA] HMS [ALPHA]
HOURS, MINUTES, SECONDS PLUS	HMS+	[XEQ] [ALPHA] HMS ℓ/∫ + [ALPHA]
HOURS, MINUTES, SECONDS MINUS	HMS-	[XEQ] [ALPHA] HMS ℓ/∫ - [ALPHA]
CONVERT TO DECIMAL HOURS	HR	[XEQ] [ALPHA] HR [ALPHA]
INTEGER PORTION	INT	[XEQ] [ALPHA] INT [ALPHA]
* INCREMENT, SKIP IF GREATER	ISG	ℓ/∫ ISG
* LAST X	LASTX	ℓ/∫ LAST X
* LABEL	LBL	ℓ/∫ LBL
* NATURAL LOG (BASE e)	LN	[LN]
NATURAL LOG OF (1+x)	LN1+X	[XEQ] [ALPHA] LN ℓ/∫ [1] ℓ/∫ [+] X [ALPHA]
* COMMON LOG (BASE 10)	LOG	[LOG]
MEAN	MEAN	[XEQ] [ALPHA] MEAN [ALPHA]
MODULO	MOD	[XEQ] [ALPHA] MOD [ALPHA]
DECIMAL TO OCTAL	OCT	[XEQ] [ALPHA] OCT [ALPHA]
* OFF	OFF	[XEQ] [ALPHA] OFF [ALPHA]
* ON	ON	[XEQ] [ALPHA] ON [ALPHA]
* POLAR TO RECTANGULAR	P-R	ℓ/∫ P→R
PACK	PACK	[XEQ] [ALPHA] PACK [ALPHA]
* PERCENT	%	ℓ/∫ %
PERCENT CHANGE	%CH	[XEQ] [ALPHA] ℓ/∫ G CH [ALPHA]
* PI	PI	ℓ/∫ π
PROMPT	PROMPT	[XEQ] [ALPHA] PROMPT [ALPHA]
PAUSE	PSE	[XEQ] [ALPHA] PSE [ALPHA]
ROLL UP STACK	R↑	[XEQ] [ALPHA] R ℓ/∫ [ENTER↑] [ALPHA]
RADIANS TO DEGREES	R-D	[XEQ] [ALPHA] R ℓ/∫ [-] D [ALPHA]
* RECTANGULAR TO POLAR	R-P	ℓ/∫ R→P
RADIANS MODE	RAD	[XEQ] [ALPHA] RAD [ALPHA]
* RECALL	RCL	[RCL]
* ROLL DOWN STACK	RDN	[R↓]
ROUND OFF	RND	[XEQ] [ALPHA] RND [ALPHA]
* RETURN	RTN	ℓ/∫ RTN
STANDARD DEVIATION	SDEV	[XEQ] [ALPHA] SDEV [ALPHA]
* SCIENTIFIC NOTATION	SCI	ℓ/∫ SCI
* SET FLAG	SF	ℓ/∫ SF
* SIGMA PLUS (STAT. REG.)	Σ+	[Σ+]
* SIGMA MINUS	Σ-	ℓ/∫ [Σ-]
SPECIFY SIGMA REGISTERS	ΣREG	[XEQ] [ALPHA] ℓ/∫ F REG [ALPHA]
* SINE	SIN	[SIN]
SIGN	SIGN	[XEQ] [ALPHA] SIGN [ALPHA]
SIZE	SIZE	[XEQ] [ALPHA] SIZE [ALPHA]
* SQUARE ROOT	SQRT	[$\sqrt{x}$]
SINGLE STEP	SST	[SST]
* STORE PLUS	ST+	[STO] [+]
* STORE MINUS	ST-	[STO] [-]
* STORE TIMES	ST*	[STO] [x]
* STORE DIVIDE	ST/	[STO] [÷]
* STORE	STO	[STO]
* RUN/STOP	STOP	[R/S]
* TANGENT	TAN	[TAN]
TONE	TONE	[XEQ] [ALPHA] TONE [ALPHA]

FUNCTION NAME	DISPLAY	KEYSTROKES
VIEW	VIEW	[XEQ] [ALPHA] VIEW [ALPHA]
* IS X EQUAL TO ZERO?	X=0?	⟦/⟧ X=0?
IS X NOT EQUAL TO ZERO?	X≠0?	[XEQ] [ALPHA] X ⟦/⟧ H ⟦/⟧ 0? [ALPHA]
IS X LESS THAN ZERO?	X<0?	[XEQ] [ALPHA] X ⟦/⟧ I ⟦/⟧ 0? [ALPHA]
IS X LESS THAN OR EQUAL TO ZERO?	X<=0?	[XEQ] [ALPHA] X ⟦/⟧ I = ⟦/⟧ 0? [ALPHA]
IS X GREATER THAN ZERO?	X>0?	[XEQ] [ALPHA] X ⟦/⟧ J ⟦/⟧ 0? [ALPHA]
* IS X EQUAL TO Y?	X=Y?	⟦/⟧ X=Y?
IS X NOT EQUAL TO Y?	X≠Y?	[XEQ] [ALPHA] X ⟦/⟧ H Y ? [ALPHA]
IS X LESS THAN Y?	X<Y?	[XEQ] [ALPHA] X ⟦/⟧ I Y ? [ALPHA]
* IS X LESS THAN OR EQUAL TO Y?	X<=Y?	⟦/⟧ X≤Y?
* IS X GREATER THAN Y?	X>Y?	⟦/⟧ X>Y?
X EXCHANGE WITH (ANY REGISTER)	X< >	[XEQ] [ALPHA] X ⟦/⟧ I ⟦/⟧ J [ALPHA]
* X EXCHANGE WITH Y	X< >Y	[X≷Y]
* EXECUTE	XEQ	[XEQ]
* X SQUARED	X↑2	⟦/⟧ x^2
* Y TO THE X POWER	Y↑X	⟦/⟧ y^x

* Function defined on the keyboard.

Functions on the face of the keys are contained within brackets [ ]. Functions following a ⟦/⟧ symbol indicate a shifted function in yellow on the keyboard, and characters between [ALPHA] symbols are alpha characters in blue on the keyboard.