

HEWLETT-PACKARD

**HP-55**

Quick  
Reference  
Guide



This guide is primarily intended for reference use after you have read the *HP-55 Owner's Handbook*. However, if you are familiar with HP calculators, you may find that this guide is sufficient. Whenever you require additional information about a particular operation, you can refer to the main handbook.

Before using the calculator, you may need to charge its battery pack as described in appendix A of the owner's handbook. The calculator can then be operated while the battery is charging or, later, on battery power alone.

**Note:** If you reset the mode switch when the calculator is in the middle of a calculation or a running program, the calculator may *blank out*. To regain control, you must switch the calculator OFF and then ON.

## RUN Mode

### Power ON

0.00 is displayed when the calculator is switched ON.

## Flashing Display

The display flashes whenever an improper operation has been attempted. Stop the flashing by pressing any key. Improper operations are listed on the inside back cover of this guide.

## Prefix Keys

### Gold Operations

Press the gold **f** prefix key before an operation written in gold or before a conversion in the direction of the gold arrow.

### Blue Operations

Press the blue **g** prefix key before an operation written in blue or before a conversion in the direction of the blue arrow.

### Clearing Prefix Keys

Press **BSI** to clear any of these keys:

**f**, **g**, **STO**, **RCL**, **STO**  $\square$ , **RCL**  $\square$ , **FIX**,  
**SCI**, **GTO**

In RUN mode, **BSI** also sets the program pointer to line 00 of program memory.

**Example.** Add 45 and 55.

Press: 45 ENTER\* 55 +

**Example.** Calculate  $\frac{(4 \times 5)}{(2 + 3)} - 6 = -2$ .

T											
Z					20	20					
Y		4	4		20	2	2	20		4	
X	4	4	5	20	2	2	3	5	4	6	-2
Key	4	4	5	X	2	2	3	+	÷	6	-

Note: ENTER\* is abbreviated here as \*

RUN Mode

## Manipulating Numbers in the Stack

The **ROT** key rotates the contents of the stack downward one position.

Press:

**ROT**

Before

4.00
3.00
2.00
1.00

After

1.00
4.00
3.00
2.00

The **X↔Y** key exchanges the contents of the X and Y registers.

Press:

**X↔Y**

Before

4.00
3.00
2.00
1.00

After

4.00
3.00
1.00
2.00

## Storage Register Operations

The HP-55 has 20 data storage registers in addition to the stack.

### Registers $R_0$ thru $R_9$

To store a displayed number in registers  $R_0$  thru  $R_9$ , press **STO** and the applicable register number key (**0** thru **9**).

To recall a number from registers  $R_0$  thru  $R_9$ , press **RCL** and the applicable register number key (**0** thru **9**).

To clear registers  $R_0$  thru  $R_9$  and the stack, press **f** **CL R**.

### Registers $R_{.0}$ thru $R_{.9}$

To store a displayed number in registers  $R_{.0}$  thru  $R_{.9}$ , press **STO** **.**, and the applicable register number key (**0** thru **9**).

To recall a number from registers  $R_{.0}$  thru  $R_{.9}$ , press **RCL** **.**, and the applicable register number key (**0** thru **9**).

To clear registers  $R_{.0}$  thru  $R_{.9}$  and the stack, press **g** **CL R**.

RUN Mode

### Storage Register Arithmetic

Possible in registers  $R_0$  thru  $R_9$ .

**Subtraction** ( $r_n - x \rightarrow R_n$ ): **STO** **-** **[n]**

**Addition** ( $r_n + x \rightarrow R_n$ ): **STO** **+** **[n]**

**Multiplication** ( $r_n \times x \rightarrow R_n$ ): **STO** **x** **[n]**

**Division** ( $r_n \div x \rightarrow R_n$ ): **STO** **÷** **[n]**

where **[n]** is a number key (**0** thru **9**),  $x$  is the number in the X-register, and  $r$  is the number in the storage register.

### Clearing Operations

**CLX** clears the X-register. A new number entry then writes over the zero in X.

**f** **CLR** clears the stack registers and data storage registers  $R_0$  thru  $R_9$ .

**g** **CLR** clears the stack registers and data storage registers  $R_{00}$  thru  $R_{99}$ .

**BSI** clears these prefix keys:

**f**, **g**, **STO**, **RCL**, **FIX**, **□**, **GTO**

To cancel a wrong prefix key, simply press the correct one. In RUN mode, **BSI** also sets the program pointer to line 00 of program memory.

## Display Format

When the calculator is switched ON, two digits are displayed to the right of the decimal point.

**Note:** Numbers are always maintained internally to 10 significant digits.

To specify fixed decimal display, press **FIX** **□**

To specify scientific notation display, press **f** **RCL** **□**.

In each case **□** is a digit key (**0** thru **9**) that determines the number of digits displayed to the right of the decimal point.

## RUN Mode

If a number is too large or too small for the "FIX" format specified, the calculator automatically displays the number in scientific notation ( $\text{f}$   $\text{SCI}$   $\text{9}$ ) displaying all 10 significant digits.

**Note:** Values having a magnitude greater than or equal to 9.99999999 99 are approximated by  $\pm 9.99999999 99$ . Values having a magnitude less than  $10^{-99}$  are approximated by zero.




## Basic Operations

### Pi

Press  $\text{f}$   $\pi$  to insert the constant  $\pi$  into the X-register. The stack considers  $\pi$  a new number and automatically lifts unless the X-register was prepared for a new number.

### Last X



The number last displayed before a calculation is automatically stored in the LAST X register. To recall its value, press  $\text{f}$   $\text{LAST X}$ .

If a number is too large or too small for the "FIX" format specified, the calculator automatically displays the number in scientific notation (  ) displaying all 10 significant digits.



**Note:** Values having a magnitude greater than or equal to 9.999999999 99 are approximated by  $\pm 9.999999999\ 99$ . Values having a magnitude less than  $10^{-99}$  are approximated by zero.

## Basic Operations

### Pi

Press   to insert the constant  $\pi$  into the X-register. The stack considers  $\pi$  a new number and automatically lifts unless the X-register was prepared for a new number.

### Last X

The number last displayed before a calculation is automatically stored in the LAST X register. To recall its value, press  .

**DSI** clears these prefix keys:

**f**, **0**, **STO**, **RCL**, **FIX**, **SCI**, **GT0**

To cancel a wrong prefix key, simply press the correct one. In RUN mode, **DSI** also sets the program pointer to line 00 of program memory.

## Display Format

When the calculator is switched ON, two digits are displayed to the right of the decimal point.

**Note:** Numbers are always maintained internally to 10 significant digits.

To specify fixed decimal display, press **FIX** **n**

To specify scientific notation display, press **f** **SCI** **n**.

In each case **n** is a digit key (**0** thru **9**) that determines the number of digits displayed to the right of the decimal point.

## RUN Mode

Register values are incremented when you press  $\Sigma+$  and decremented when you press  $\Sigma-$  as shown below:

$$\begin{aligned} r_{\bullet 0} \pm 1 &\rightarrow R_{\bullet 0} \rightarrow \text{X-register} \\ r_{\bullet 1} \pm x &\rightarrow R_{\bullet 1} \\ r_{\bullet 2} \pm x^2 &\rightarrow R_{\bullet 2} \\ r_{\bullet 3} \pm y &\rightarrow R_{\bullet 3} \\ r_{\bullet 4} \pm y^2 &\rightarrow R_{\bullet 4} \\ r_{\bullet 5} \pm xy &\rightarrow R_{\bullet 5} \end{aligned}$$

**Note:** You can sum X values alone by clearing the Y-register to zero before starting the summation calculations.

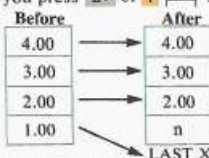
The diagrams below show you what happens to the stack when you press  $\Sigma+$  or  $\Sigma-$ :

Press:

$\Sigma+$

or

$\Sigma-$



This operation prepares the X-register for a new number. The new number then writes over the number in the X-register without lifting the stack.

**RUN Mode**

Press **RCL**  **$\Sigma^+$**  to recall  $\Sigma x$  and  $\Sigma y$  to the X- and Y-registers respectively. The previous value in Y is lost; the previous value in X goes to LAST X.

**Note:** Press **STO**  **$\Sigma^+$**  to accumulate the first data pair following **RCL**  **$\Sigma^+$** . Then continue accumulations using the  **$\Sigma^+$**  key as you did before.

## Functions and Other Operations

### Logarithms

**1**  **$\ln$**  calculates the natural logarithm of the displayed X value.

**9**  **$e^x$**  calculates e raised to the power of the number displayed.

**1**  **$\log$**  calculates the common logarithm of the displayed X value.

**9**  **$10^x$**  calculates 10 raised to the power of the number displayed.

### Exponentials

**$y^x$**  calculates the base number in the Y-register raised to the power in the displayed X-register using the formula  $y^x = e^{x \ln y}$ .

**$\sqrt[x]{y}$**   **$y^x$**  calculates roots since  $\sqrt[x]{y} = y^{1/x}$ .

In either case, the stack drops during the  **$y^x$**  operation.

### Factorial

**$n!$**   **$\frac{1}{n!}$**  Calculates the factorial of the integer in the displayed X-register. The integer (n) must be within the range  $0 \leq n \leq 69$ .

## Angular Functions

### Angular Modes

**$\frac{1}{D}$**   **$\frac{1}{DEC}$**  Sets decimal degrees mode. (This is preset when calculator is switched ON.)

**$\frac{1}{R}$**   **$\frac{1}{RAD}$**  Sets radians mode.

**$\frac{1}{G}$**   **$\frac{1}{GRD}$**  Sets grads mode.

$360 \text{ degrees} = 2\pi \text{ radians} = 400 \text{ grads}$ .

RUN Mode

### Angular Mode Conversion


 Converts from decimal degrees to radians.


 Converts from radians to decimal degrees.

.9  Converts from decimal degrees to grads.

.9  Converts from grads to decimal degrees.

### Trigonometric Functions


 Calculates the sine of the value in the X-register.


 Calculates the cosine of the X value.

 Calculates the tangent of the X value.


**Note:** All trigonometric functions assume the angle is expressed in the decimal form of the prevailing angular mode.


 Calculates the arc sine of the X value.

 Calculates the arc cosine of the X value.


 Calculates the arc tangent of the X value.


### Angle Conversion

 Converts an angle (or time) in decimal degrees (hours) to an angle in degrees, minutes, and seconds in the following form ...d.mmsshh... .

 Converts an angle (or time) in the following form ...d.mmsshh... to an angle in decimal degrees (hours).



### Angle Arithmetic

 Adds angles (or times) in the following form ...d.mmsshh... regardless of the angular mode. The stack drops as in any two number operation.

 Subtracts angles (or times) in the same format as above. Stack also drops.



## RUN Mode

## Polar/Rectangular Coordinate Conversion

  Converts x and y values in the X- and Y-registers respectively to polar coordinates  $\theta$  and r.

$$\theta = \tan^{-1} \frac{y}{x} \rightarrow Y$$

$$r = \sqrt{x^2 + y^2} \rightarrow X$$



  Converts r and  $\theta$  values in the X- and Y-registers to rectangular coordinates x and y.

$$y = r \sin \theta \rightarrow Y$$

$$x = r \cos \theta \rightarrow X$$

## English/Metric Conversions

The arrows on the conversions are color-coded to the  and  keys.

1. To convert from English to metric units, press the  key and then the appropriate digit key.
2. To convert from metric to English units, press the  key and then the appropriate digit key.

## Statistics

All statistical calculations in the HP-55 are based on the data in registers  $R_{00}$  thru  $R_{99}$ . This data can be stored or accumulated automatically using the  $\Sigma^+$  key.

### Mean

$\frac{f}{n}$  Calculates the mean of both X and Y values using the formulas:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \qquad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

The illustrations below represent what happens in the stack when you press  $\frac{f}{n}$  :

Press:



Before

1.00
2.00
3.00
4.00

After

1.00
2.00
$\bar{y}$
$\bar{x}$

→ lost

LAST X

RUN Mode

## Standard Deviation

**9**  $\sqrt{\square}$  Calculates the sample standard deviation of both X and Y values using the formulas:

$$s_x = \sqrt{\frac{\sum x^2 - \frac{(\sum x)^2}{n}}{n-1}} \quad s_y = \sqrt{\frac{\sum y^2 - \frac{(\sum y)^2}{n}}{n-1}}$$

The population standard deviation is related to the sample standard deviation by the formula below:

$$s' = \sqrt{\frac{n-1}{n}} \times s$$

The illustrations below represent what happens in the stack when you press **9**  $\sqrt{\square}$

**Press:**

**9**  $\sqrt{\square}$

**Before**

1.00
2.00
3.00
4.00



**After**

1.00
2.00
$s_y$
$s_x$

→ lost

LAST X

### Linear Regression



  Calculates the coefficients of the linear equation  $y = A + Bx$  using the least square method.

The y intercept (A) is displayed and is calculated using the formula:

$$A = \frac{\sum y \sum x^2 - \sum x \sum xy}{n \sum x^2 - (\sum x)^2}$$

The slope (B) is returned to the Y-register and is calculated using the formula:

$$B = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2}$$

The illustrations below represent what happens in the stack when you press  .

Press:



Before		After
4.00	→	4.00
3.00	→	3.00
2.00	→ lost	B
10.00	↘	A
		LAST X

RUN Mode

### Linear Estimate

**9** Calculates a predicted  $y$  ( $\hat{y}$ ) based on the data already in registers  $R_{\bullet 0}$  thru  $R_{\bullet 5}$ .

**Note:** For all statistical calculations, the data need be accumulated only once. The summations are kept intact in registers  $R_{\bullet 0}$  thru  $R_{\bullet 5}$ .

## TIMER Mode

**R/S** Starts and stops the timer.

**CLX** Clears the timer to 0.00.00 00.

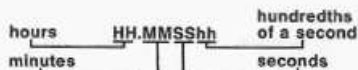
**EEY** Switches the display of the hundredths digits on and off.

**0** thru **9** When the timer is running, pressing a digit key will store the elapsed time in registers  $R_0$  thru  $R_9$  respectively. When the timer is stopped, pressing a digit key will recall values in  $R_0$  thru  $R_9$  respectively.

**Note:** The timer has a range from 0 to 100 hours. When running, the X-register increments every .01 seconds with .01% accuracy.

## Timer Operations in RUN Mode

Starting times within the 0 to 100 hour range must be keyed into the X-register in RUN mode using the following format:



If the initial time is negative or has more than 99 hours in the hours field, or 59 minutes in the minutes field, or 599999 in the seconds field when you switch to TIMER mode, the display will show 0.00.00 00.

### Adding and Subtracting Times



**[F]  $\frac{H.MMSShh}{+}$** , Adds times (or angles) in the X- and Y-registers that are in the following format: ...H.MMSShh....




**[B]  $\frac{H.MMSShh}{-}$** , Subtracts times (or angles) in the X-register from times in the Y-register in this same format.

### Converting Times

**[T]  $\frac{H.MMSShh}{\rightarrow}$** , Converts times (or angles) in HH.MMSShh format to decimal hours (degrees) in the direction of the gold arrow.

## TIMER Mode

  Converts times (or angles) in decimal hours (degrees) to HH.MMSShh format in the direction of the blue arrow.

**Note:** The timer does not stop by resetting the mode switch. Pressing  stops the timer in any mode. When switching from TIMER to RUN mode, times are converted to HH.MMSShh format and the display format changes to  .

## PRGM Mode

### Program Memory

Program memory consists of 50 lines (00 thru 49). Line 00 cannot contain an instruction. Executing line 00 in a program halts program execution automatically. When the calculator is switched ON, program memory is filled with instructions to branch to line 00 (→00 in the display).

### Program Pointer

The calculator executes the instruction at which the program pointer is pointing. The pointer then increments one step downward (in the case of a direct branch, it jumps to the line specified) and points at a new instruction. After executing line 49, the pointer automatically returns to line 00 and program execution stops.

The program pointer can be incremented one step at a time by pressing **SSY** or decremented one step at a time by pressing **BSI** in PRGM mode. (Refer also to Program Operations in RUN Mode.)

### Keycodes

Keys are stored in program memory as codes: each key writing over the previous instruction in memory at that line number. The codes for the digit keys are 00 thru 09. The code for the **GTO** key is a dash in the middle of the display. The **GTO** key must always be followed by two digit keys, the first of which must be less than 5. These three keys are merged to take only one line of memory. For the other keys, the code denotes the row and column of the key.

### Using **RS**

**RS** is used in RUN mode to start program execution. (Refer to Program Operations in RUN Mode.)

**RS** is used in PRGM mode as an instruction to halt program execution. This is useful when data needs to be entered to continue program operation or when checking intermediate results.

### Writing and Executing a Program

1. Set the program pointer to the line at which you want the program to start (usually at line 00).

## PRGM Mode

2. In PRGM mode, key in the same keys you would key in to solve the problem manually. Special branching instructions are covered later.
3. End the program with **RS** (to stop execution at the end of the program) or **GTO 0 0** (to stop execution at the beginning of the program).
4. Switch to RUN mode, set the pointer to line 00, key in any needed data, and press **RS** to run the program.

## Editing a Program

Each key pressed in PRGM mode writes over the instruction already contained in that line of memory.

To change or correct an instruction in memory, position the pointer at the line *preceding* the line to be changed by:

1. Pressing **SST** or **BST** in PRGM mode.
2. Pressing **GTO** (line number) in RUN mode.

Then press the correct key(s).

## PRGM Mode

## Branching

**GTO** (line number) moves the program pointer to the line specified. The line number must be two digit keys between 00 and 49 or the instruction is rejected. In PRGM mode all three keys are merged to take only one line of memory.

**IF<REL>** (line number) When preceded by the **IF** prefix key, the **GTO** instruction is executed only if the value in the X-register is less than or equal to the value in the Y-register. The **IF** prefix key is not merged with the other three keys.

**EQ** (line number) When preceded by the **EQ** prefix key, the **GTO** instruction is executed only if the value in the X-register is equal to the value in the Y-register.

## Program Stops

1. Execution of a **R/S** in a program.
2. Execution of line 00 in a program.
3. Pressing any key.

**Note:** If a program has been stopped by pressing a key, be careful not to restart it in the middle of a digit entry key sequence in the program or between a prefix key and an operation. Use **BSF** or **SSF** to reposition the program pointer in either of these cases.

**PRGM Mode**

4. Overflow in a register (9.99999999 99).
  5. Execution of an improper operation.
- These operations are listed on the inside back cover of this guide.

## Program Operations in RUN Mode

**R/S** Terminates digit entry, clears prefix keys and then starts program execution from the current position of the program pointer with two exceptions:

1. If the program pointer is at line 00, **R/S** starts program execution at line 01.
2. If the program pointer is stopped at a **R/S** instruction in a program, program execution resumes at the next line.

**BSF** Sets the program pointer to line 00 of program memory in RUN mode and also clears prefix keys.

**SSF** Executes a program one line at a time. This is useful when debugging your programs.

**GTO** (line number) Moves the program pointer to the line specified.

**Note:** When an invalid address is used in RUN mode, the program pointer does not move and the display is also invalid.

## General Information

### Underflow

If a result develops that is too small in magnitude to be carried in a register ( $<10^{-99}$ ), the register is set to zero; a running program continues execution.

### Overflow

If a result develops a magnitude that exceeds the capacity of a register ( $>9.999999999\ 99$ ), the register is set to all 9's (with appropriate sign); a running program halts execution.

### Multiple Decimal Point Display

All decimal points but one light to warn you that you have 2 to 5 minutes of operating time left on battery power.

### Flashing Display

The display flashes when any of several improper operations (listed on the inside back cover of this guide) are attempted. Pressing any key stops the flashing.

## Index

**+**, **=**, **×**, **÷** (arithmetic operations), 5

**.** (decimal), 3

**0** thru **9** (digits), 3  
in TIMER mode, 22

**%** (percentage), 11

**π** (pi), 10

## A

---

Addition, 5

angles, 17

storage register, 8

times, 23

Angle

addition, 17

conversion, 17

subtraction, 17

Angular modes, 15

conversions, 16

Arc cosine, 17

Arc sine, 17

Arc tangent, 17

Arithmetic, 5

## B

---

**BST** (back-step), 9, 25, 26, 27, 28

Branching, 27


## C


---

**CHS** (change sign), 3

## Clear operations

 (clear x), 3, 8, 22

 (clear  $R_0 - R_9$ ), 8

 (clear  $R_{+0} - R_{+9}$ ), 8

 (clear prefix), 9

OFF-ON (clear entire machine), 3

## Conversions

angles, 17

coordinate, 18

metric, 18

modes, 16

times, 23, 24

  (cosine, arc cosine), 16, 17

## D

 (degrees), 15

Degrees, minutes, seconds addition, 17

conversion, 17

  (degrees  $\rightleftharpoons$  radians conversion), 16

Deleting data, 12

Display format, 9, 10

fixed decimal display, 9, 10

flashing, 2, 29

multiple decimal point, 29

scientific notation, 9, 10

## E

 (natural antilogarithm), 14

Editing a program, 26

 (enter exponent), 3

in TIMER mode, 22

English/metric conversions, 18

  (feet  $\rightleftharpoons$  meters)

  (gallons  $\rightleftharpoons$  liters)

  (inches  $\rightleftharpoons$  millimeters)

## Index

ENTER  $\frac{1}{1000}$ , 4, 5  
 Exponent entry, 3  
 Exponentials, 15

## F

(prefix key), 2, 9  
 Factorial, 15  
 (fixed decimal display), 9, 10  
 Flashing display, 2, 29

## G

(prefix key), 2, 9  
 (go to), 25, 26, 27, 28  
 (grads), 15

## H

(angle addition, angle subtraction), 17  
 (time addition, time subtraction), 23  
  
 (angle conversion), 17  
 (time conversion), 23, 24




## I

Improper operations, 36

**K**

Keycodes, 25  
Keying in numbers, 3  
Keying in times, 23


**L**

Last x, 10  
Linear estimate, 22  
 (linear regression), 21  
 (natural logarithm), 14  
 (common logarithm), 14

**M**

Mean, 19  
Multiple decimal point display, 29  
Multiplication, 5  
    storage register, 8

**N**

 (factorial), 15  
Negative numbers, 3  
Number building keys, 3  
Number terminating keys, 3

**O**

OFF-ON switch, 1, 2  
Overflow, 29

**P**

Percentage, 11  
Polar to rectangular conversion, 18

**Index**

Prefix keys

clearing, 9

 , 2

 , 2

Program

memory, 24

branching, 27

editing, 26

operations in RUN mode, 28


pointer, 24

stops, 27

writing, 25


**R**


 (radians), 15

 (recall), 7

Reciprocal, 11


 (rectangular ↔ polar conversion), 18


 (roll down), 6

 (run/stop), 25, 26, 27, 28

in TIMER mode 22

**S**

 (scientific notation display), 9, 10


 (Sigma minus), 12

See also 19–22.

 (Sigma plus), 12

See also 19–22.

 ,  (sine, arc sine) 16, 17

 (single-step) 25, 26, 27, 28

Square, 11

Square root, 11

Stack

clearing, 8

exchange x and y, 6

manipulating, 6

roll down, 6

See also 5, 11, 13, 19, 20, 21.

$\sigma^2$  (standard deviation), 20

Statistics

linear estimate, 22

linear regression, 21

mean, 19

standard deviation, 20

summations, 12

**STO** (store), 7, 8

Storage registers, 7

arithmetic, 8

Subtraction, 5



angles, 17

storage registers, 8

times, 23

Summations, 12

## T

  (tangent, arc tangent) 16, 17

Time

addition, 23

conversion, 23, 24

subtraction, 23





Trigonometric functions, 16

## X



  (conditional branches), 27

**XCH** (exchange x and y), 6
















# Index

-  (mean), 19
-  (reciprocal), 11
-  (square), 11
-  (square root), 11

## Y

-  (linear estimate), 22
-  (exponential), 15

## Improper Operations

1. Pressing  with  $x = 0$ .
2. Pressing  with  $x = 0$ .
3. Pressing  with  $x \leq 0$ .
4. Pressing  with  $x \leq 0$ .
5. Pressing  with either:  $y < 0$   
or:  $y = 0$  and  $x < 0$ .
6. Pressing  with  $x$  a non-integer value  
or  $x < 0$ .
7. Pressing  with  $|x| > 1$ .
8. Pressing  with  $|x| > 1$ .
9. Pressing  with  $x < 0$ .
10. Pressing  with  $n \leq 0$ .
11. Pressing  with  $n \leq 1$ .
12. Pressing  or   
with  $n \sum x^i - (\sum x)^i = 0$ .
13. Pressing  or  with  $n \leq 0$ .



Sales and Service from 172 offices in 65 countries.  
19310 Pruneridge Avenue, Cupertino, CA 95014

For additional Sales and Service Information Contact Your Local Hewlett-Packard Sales Office or Call 408/996-0100 (Ask for Calculator Customer Service).



00055-90002 Rev. B 2/75 Printed in U.S.A.

