STORAGE REGISTERS

<table>
<thead>
<tr>
<th>R0</th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
<th>R4</th>
<th>R5</th>
<th>R6</th>
<th>R7</th>
</tr>
</thead>
</table>

**STO n** stores x value in Rn.

**RCL n** recalls value from Rn.

**STO - n** x value subtracted from contents of Rn and difference stored in Rn.

**STO + n** x value added to contents of Rn and sum stored in Rn.

**STO × n** x value multiplied by contents of Rn and product stored in Rn.

**STO ÷ n** Contents of Rn divided by x value and quotient stored in Rn.

COORDINATE CONVERSION

**f ▶R** converts polar coordinates (r,θ) to rectangular coordinates (x,y).

| Y | θ | x | y |

**g ▶P** converts rectangular coordinates (x,y) to polar coordinates (r,θ).

CONTROLLING THE DISPLAY

**f FIX n** shows numbers with “n” places to the right of the decimal point.

**f SCI n** shows numbers in scientific notation with “n” places to the right of the decimal point.

**f ENG n** shows numbers with “3 + n” digits and an exponent of ten that is the nearest multiple of three. For example, after pressing **f ENG 1**, 1.2456 × 104 is displayed **12.4603**.

SUMMATIONS

Press **f REG** to clear storage registers R0 through R7 before using **Σ+**.

**Σ+** stores summations of the numbers in the X- and Y-registers into registers R3 through R7 as shown below:

\[ n \rightarrow R_3 \quad \sum xy \rightarrow R_5 \quad \sum x \rightarrow R_7 \]
\[ \sum y \rightarrow R_4 \quad \sum x^2 \rightarrow R_6 \]

**f Σ-** Subtracts same entries from the summations shown above in registers R3 through R7.

AUTOMATIC MEMORY STACK

<table>
<thead>
<tr>
<th>T</th>
<th>0.00</th>
<th>Top</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>0.00</td>
<td>Always displayed</td>
</tr>
</tbody>
</table>

Calculation Rules to Remember

1. To use any one-number function (e.g., \( \sqrt{x} \), \( \log \), \( \sin \)): a. Key in the number.
   b. Press the function keys.

   For example to calculate \( \frac{1}{4} \), key in 4 and press \( \sqrt{x} \).

2. To use any two-number function (e.g., - , + , × , ÷ , \( \sqrt{x} \), \( xy \)): a. Key in the first number.
   b. Press \( \text{ENTER} \).
   c. Key in the second number.
   d. Press the function keys.

   For example to calculate 2 × 3, key in 2, press \( \text{ENTER} \), key in 3, and press \( \times \).

Printed in U.S.A. © Hewlett-Packard Company 1975
**Program Memory**

When the calculator is switched ON, program memory is filled with \[0\][0] instructions (keycode 13 00).

- 00 → Automatic stop instruction.
- 01 13 00
- 02 13 00
- 03 13 00
- 04 13 00
- 46 13 00
- 47 13 00
- 48 13 00
- 49 13 00

49 steps for your programs.

**Program Mode**

In program mode, only the following three functions are active. Every other function key is recorded in program memory when pressed.

- **SST** Single step. Displays step number and keycode of next program memory step.
- **BST** Back step. Displays step number and keycode of previous program memory step.

**Automatic Run Mode**

The three active keys in program mode operate differently in automatic run mode.

- **SST** Single step. Displays step number and keycode of current program memory step when held down; executes current instruction, displays result, and moves to next step when released.
- **BST** Back step. Moves to previous step and displays step number and keycode of previous program memory step when held down; displays original contents of X-register when released. No instructions are executed.

**Executed In a Program**

Function keys may be executed in a program. Program instructions are described below:

- **R/S** Stops program execution.
- **GTO n n** Branches program execution to step number specified. Execution then continues sequentially downward. Step numbers must be two digits (e.g., \[GTO 0 8\] executes a branch to step 8).

**Pressed from the Keyboard**

Function keys may be pressed from the keyboard. Normally, only two programming instructions are also pressed from the keyboard.

- **GTO n n** Specifies that the step number selected by \[GTO n n\] becomes the current program memory step number. All step numbers must be two digits. (e.g., press \[GTO 0 8\] to branch to step 8).