Hewlett-Packard 9111A Graphics Tablet Programming Guide

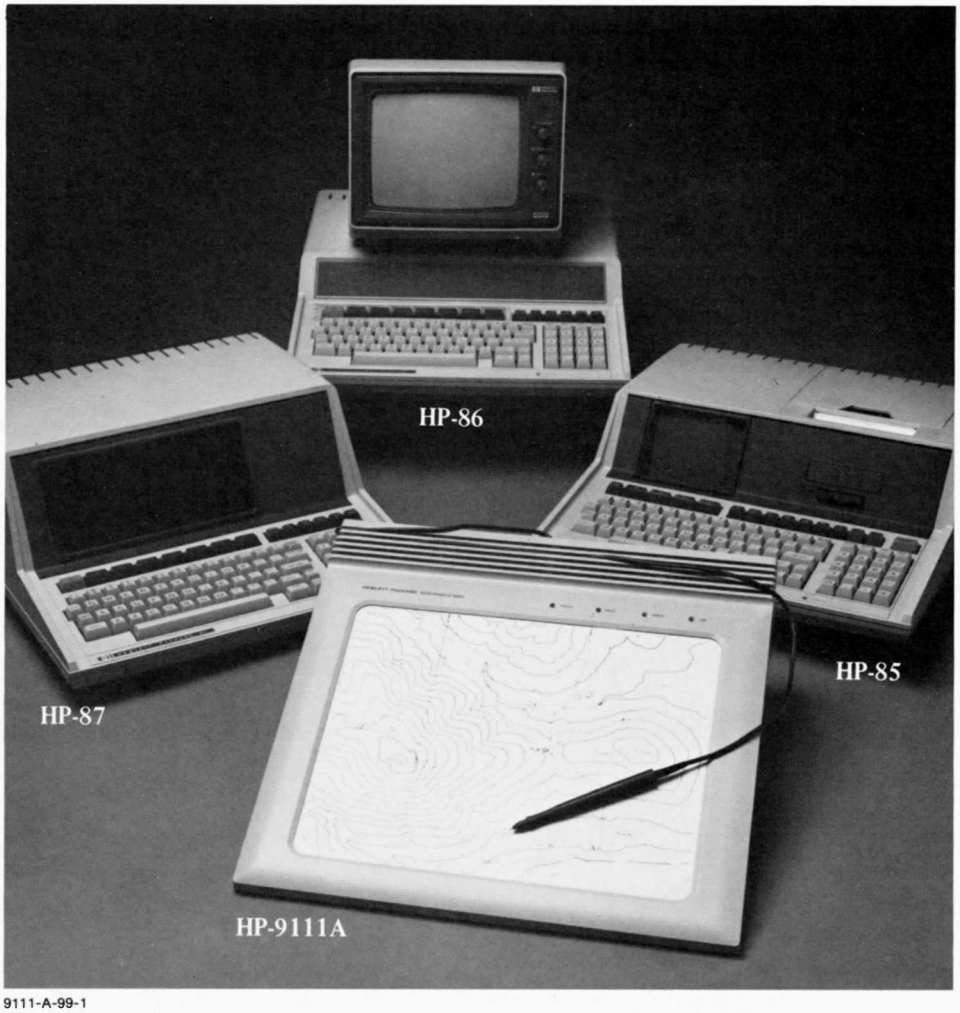
For Use with HP Series 80 Personal Computers

DESIGNED FOR HP-IB SYSTEMS

Hewlett-Packard 9111A Graphics Tablet Programming Guide

For Use with HP Series 80 Personal Computers

DESIGNED FOR HP-IB SYSTEMS



9111-A-99-1

16399 W. Bernardo Drive, San Diego, California 92127 Nov 1982

Table of Contents

How to Use This Programming Guide

This programming guide introduces you to some basic interactive graphics programming techniques using the HP 9111A Graphics Tablet and the HP-85, HP-86, and HP-87 Personal Computers. These techniques are demonstrated in short example programs that you can later modify or expand to meet your application requirements.

The first example is a simple program that illustrates the basic concept of generating a cursor ("+" symbol) on the CRT of your personal computer. This cursor will track the location of the stylus tip as you move the stylus around the active digitizing area of the tablet. Each subsequent example shows how to modify the program to utilize additional capabilities of the tablet. For example, you will learn how to define the tablet's menu softkeys, assigning different digitizing modes to each key.

Although the logic of each program is explained, detailed explanations of individual statements and instructions are not included. Therefore, you will probably wish to use this manual in conjunction with your other computer and graphics tablet documentation. You should first be familiar with the operation of the tablet, and the operation and BASIC programming of the computer. Then you might wish to "walk through" the examples in this manual, keying them in and running them. Once you see a few of the things you can do with the tablet, you can spend more time learning the HP-GL (Hewlett-Packard Graphics Language) instructions that are defined in the HP 9111A Graphics Tablet Programming Manual. Then, if you'd like, you can modify or expand these programs to suit your particular applications.

As you work through this programming guide, you will sometimes see notes that serve as reminders or to tell you of special considerations. These notes apply to all Series 80 personal computers unless indicated by the phrase "Applies to (computer model number) only."

If you are interested in learning more about how to send HP-GL instructions to the tablet using your Series 80 personal computer, refer to Appendix A. If you find that you are unfamiliar with some of the computer-graphics terminology used in this manual, refer to Appendix B, where you will find a glossary.

For more specific information about the BASIC statements and HP-GL instructions used in this manual, refer to the documents in the next table.

Reference Documents

Manual Title	Part Number	Contents
HP 9111A Graphics Tablet		
HP 9111A Graphics Tablet Operator's Manual	09111-90003	How to operate the HP 9111A.
HP 9111A Graphics Tablet Programming Manual	09111-90004	How to program in HP-GL.*
HP 9111A Graphics Tablet Reference Card	09111-90007	Summary of HP-GL instructions, error messages, tablet default conditions, status word, and binary default response.
HP-85 Personal Computer		
HP-85 Owner's Manual and Programming Guide	00085-90002	How to operate and program the HP-85.
HP-83/85 Plotter/ Printer ROM Owner's Manual	00085-90519	How to create graphics using AGL.*
HP-85 I/O Programming Guide	00085-90142	How to program status bit checking, end-of-line branching, interrupts, service requests, and binary data transfers.
HP-86 Personal Computer		
Introduction to the HP-86	00086-90014	How to operate the HP-86.
HP-86/87 Operating and BASIC Programming Manual	00087-90017	How to program the HP-86.
HP-86/87 Plotter ROM Owner's Manual	00087-90109	How to create graphics using AGL.*
HP-85 I/O Programming Guide with supplement	00085-90142	How to program status bit checking, end-of-line branching, interrupts, service requests, and binary data transfers.
HP-IB Interface Owner's Manual	82937-90017	How to change the HP-IB select code; theory of operation.

Manual Title	Part Number	Contents
HP-87 Personal Computer		
Introduction to the HP-87	00087-90002	How to operate the HP-87.
HP-86/87 Operating and BASIC Programming Manual	00087-90017	How to program the HP-87.
HP-86/87 Plotter ROM Owner's Manual	00087-90109	How to create graphics using AGL.*
HP-85 I/O Programming Guide with supplement	00085-90142	How to program status bit checking, end-of-line branching, interrupts, service requests, and binary data transfers.

***Reference Documents** (Continued)*

*AGL stands for A Graphics Language. AGL is an extension of the BASIC programming language; it provides statements for designing graphics. HP-GL stands for Hewlett-Packard Graphics Language. This is the instruction set that is actually understood by the tablet. Each AGL statement is translated by the computer into a defined series of HP-GL instructions; the tablet then receives these instructions. You can communicate with the tablet using certain AGL statements, such as CURSOR and DIGITIZE. However, the tablet provides you with a complete set of HP-GL instructions. These instructions are a powerful tool for programming the tablet because they allow you to communicate directly with the tablet to obtain specific data or results that are not available from AGL statements.

Minimum System Configuration

In order to run the program examples given in this manual, your computer system must include specific components. Refer to the following table for the computer equipment that you need in addition to the HP 9111A Graphics Tablet.

Equipment	Model Number
HP-85 Personal Computer	
ROM Drawer Plotter/Printer ROM I/O ROM HP-IB Interface	HP 82936A HP 00085-15002 HP 00085-15003 HP 82937A
HP-86 Personal Computer	
Display Monitor ROM Drawer Plotter ROM I/O ROM HP-IB Interface *Disc Drive (optional)	HP 82912A, HP 82913A, or equivalent HP 82936A HP 00087-15002 HP 00087-15003 HP 82937A HP 9130A or equivalent

Minimum System Configuration

*The disc drive is optional. However, it is useful for storing the example programs for later use and modification. Ask your dealer or local HP Sales and Support Office for details on available disc drives.

Minimum System Configuration *(Continued)*

Equipment	Model Number
HP-86 Personal Computer (Continued)	
*Printer (optional) *Printer Cable	HP 82905B or equivalent HP 82957A
HP-87 Personal Computer	
ROM Drawer Plotter ROM I/O ROM *Disc Drive (optional) *Printer (optional) HP-IB Interconnection Cables (one for each peripheral)	HP 82936A HP 00087-15002 HP 00087-15003 HP 82900 Series or equivalent HP 82905B or equivalent HP 10833 Series

*The disc drive and printer are optional. However, the disc drive is useful for storing the example programs for later use and modification. The printer can be used in the example programs for obtaining printed reproductions of the computer's CRT screen. Ask your dealer or local HP Sales and Support Office for details on available disc drives and printers.

Connect the Graphics Tablet to Your Computer

The Series 80 personal computers all operate in a similar manner, using the Hewlett-Packard Interface Bus (HP-IB) to communicate with peripheral devices. However, each computer has a specific method for connecting peripheral devices using the HP-IB interface. The next two paragraphs contain general information about HP-IB; for more specific information about your computer, refer also to the appropriate subsection below. Only the computer/tablet connections are discussed in the subsections. Refer to your computer's documentation for detailed procedures such as installing the I/O and Plotter/Printer or Plotter ROMs in a ROM drawer, and inserting a ROM drawer into the computer.

An HP-IB Overview

The HP-IB is Hewlett-Packard's implementation of the IEEE 488-1978* interface standard. The purpose of the HP-IB is to provide for mechanical, electrical, timing, and data compatibilities between all devices adhering to the standard. On the HP-85 and HP-86, the HP-IB interface is a card that you insert into a port on the rear panel of the computer. Attached to this card is a cable that connects to a peripheral device. On the HP-87, the HP-IB interface is integral to the computer; you do not actually see the interface. Separate cables connect the HP-87 to peripheral devices.

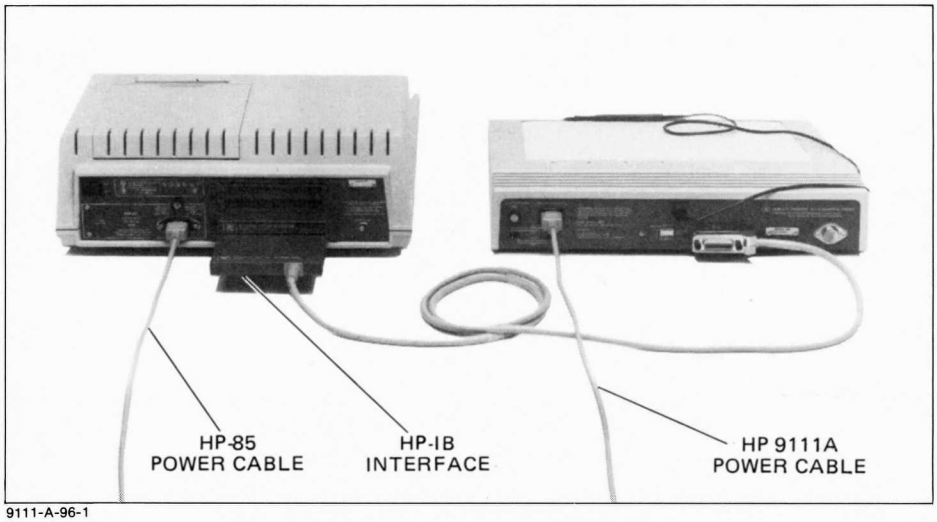
Each HP-IB interface can interconnect as many as 15 devices. Because of this, the interface must use an addressing technique to ensure that each device receives only the data that are intended for it. With this addressing technique, alternate devices can be instructed to talk (output) or listen (input). More than one device can listen at the same time, but only one device can be designated as the talker at any given time. The addressing technique requires assigning a "select code" to the HP-IB interface and an

*Institute of Electrical and Electronics Engineers 488-1978 standard for a general-purpose interfacing bus.

"address code" to each peripheral device. The two codes are then combined in certain BASIC statements to address different devices. In the "OUTPUT 706" statement, 7 is the interface select code, and 06 is the address code; the device with the address of 06 will receive the information sent by the OUTPUT statement. Select codes are discussed in the subsections for each computer; the address code is discussed in the subsection for the graphics tablet.

The Tablet and the HP-85

In order to interface the HP-85, you must have an HP-IB interface card. The select code in this card is preset to 7 and need not be changed for the purposes of this manual. The following illustration shows how the tablet is connected to the HP-85.



9111-A-96-1

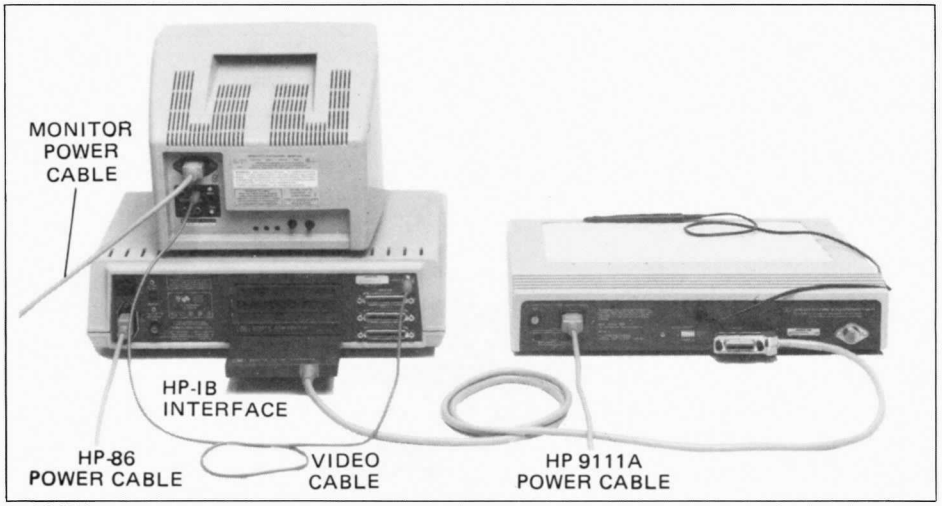
HP-85 and Tablet Connection

The Tablet and the HP-86

In order to connect the tablet to the HP-86, you must have an HP-IB interface card. To avoid electrical damage, you need to check the select code of the interface card to be sure it is not the same select code that is used by the HP-86 printer/disc interface. The HP-86 printer/disc interface uses the select code 7; the interface card can use any select code between 3 and 10 *except 7*. The examples in this manual use the select code 6. Refer to the following instructions to disassemble the HP-IB interface card and change the select code to 6. (In addition, the HP-IB Interface Owner's Manual provides more detailed information.)

After you have changed the select code in the HP-IB interface card, use the following illustration to see how the tablet is connected to the HP-86.

HP-86 and Tablet Connection



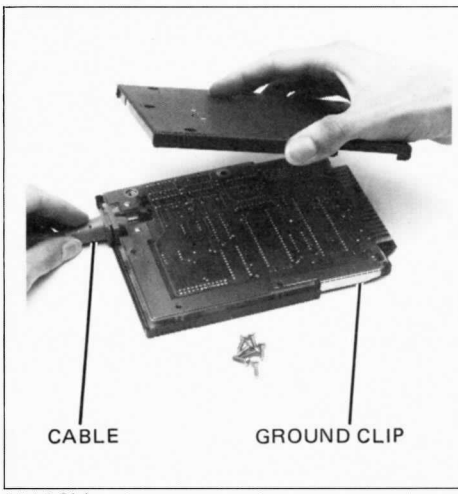
9111-A-97-1

Disassemble the HP-IB Interface Card

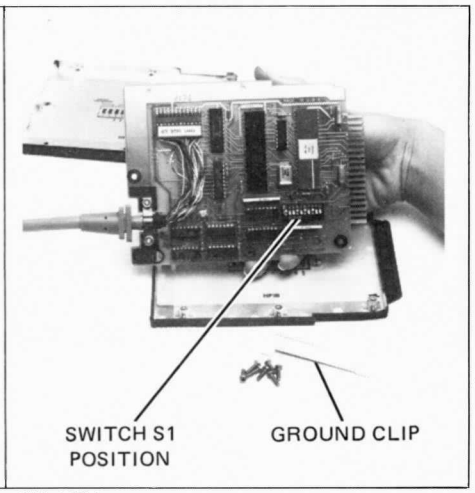
Follow these steps to disassemble the HP-IB interface card.

1. Place the interface card on a flat surface; the side with the screws should face upward and the cable should come out to the left.
2. Using a phillips-head screwdriver, remove all of the screws and set them aside.
3. Hold the cable in place and remove the top half of the interface card. (Refer to the photograph on the left.)
4. Slide the ground clip off and set it aside. Then remove the circuit board and turn it over. (Refer to the photograph on the right.)

HP-IB Interface Disassembly.
The photograph on the left illustrates step 3; the photograph on the right illustrates step 4.



9111-A-94-1



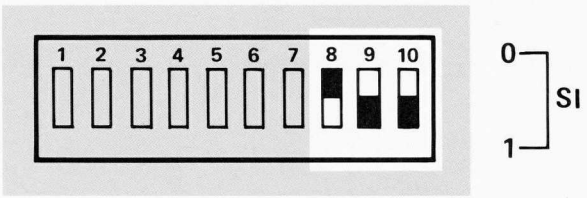
9111-A-95-1

Change the HP-IB Select Code

Switch segments **8**, **9**, and **10** of switch **S1** (located in the lower right of the circuit board, as shown above) are used for setting the HP-IB select code. The select code may be set to any number between 3 and 10; however, the program examples in this manual assume that you have set the select code to 6. Follow these steps:

1. Refer to the following figure to identify the **0** and **1** switch positions.

2. Using a pencil tip or similar object, place switch segment **8** in the **0** position. Then place switch segments **9** and **10** in the **1** position. Be careful not to disturb the settings of adjoining switch segments. When you have set the address to 6, switch **S1** should look the same as the following figure.



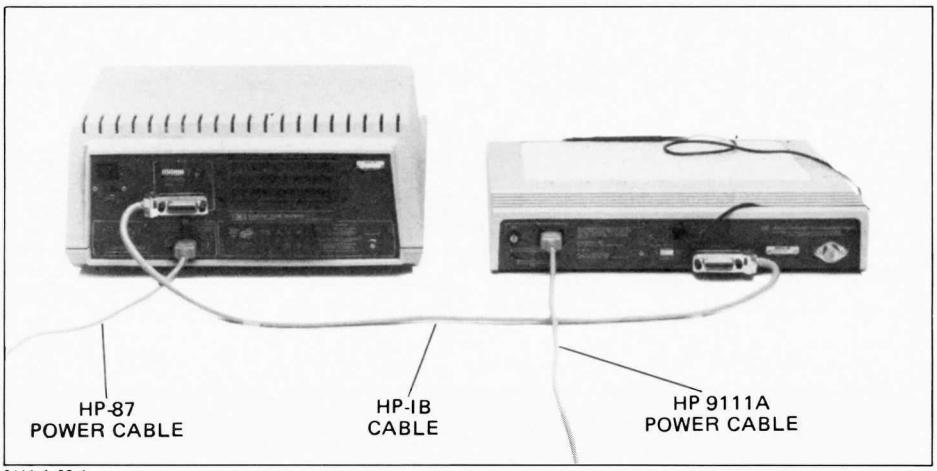
HP-IB Select Code Set to 6

Reassemble the HP-IB Interface Card

To reassemble the HP-IB interface card, reverse the disassembly procedures listed above. Make sure that the ground clip is in place; it should be on the side of the circuit board that does *not* have any components attached.

The Tablet and the HP-87

The HP-IB interface select code on the HP-87 is set at the factory to 7 and need not be changed for the purposes of this manual. The illustration below shows how the tablet is connected to the HP-87.



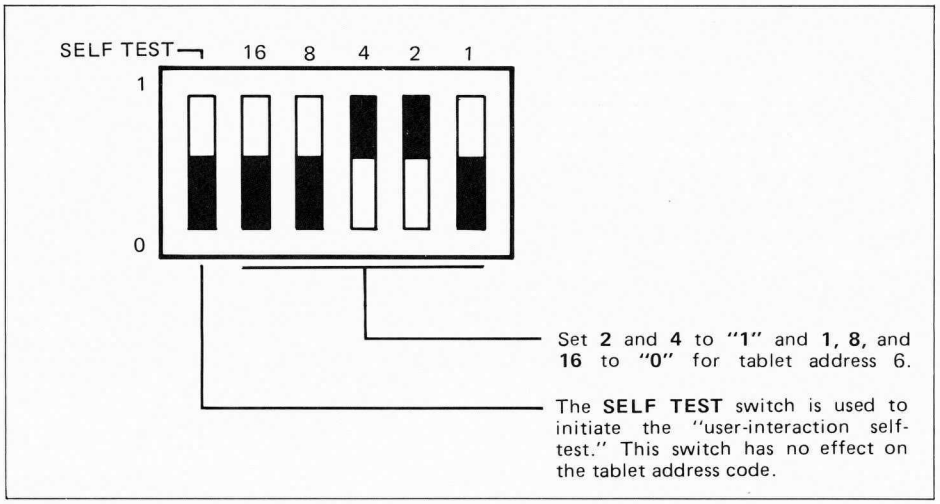
9111-A-98-1

HP-87 and Tablet Connection

Check the Tablet's Address Switches

The five **ADDRESS** switches on the tablet are set to 6 when shipped from the factory. Check the following figure to be sure the **ADDRESS** switches are set to the indicated positions. (The HP 9111A Operator's Manual describes the switches in more detail.)

The Tablet's Factory-Set Address



Turn On the System

Applying ac power to your tablet and computer initiates internal self-tests that verify that your system is ready to use. Satisfactory completion of these self-tests is indicated as follows:

- The tablet **LINE** light turns on and stays on, the other three lights turn on momentarily, and an ascending series of tones is generated.
- The HP-85 displays an underscore dash (_) in the upper-left corner of the CRT.
- The HP-86 and HP-87 display a rectangular box (▮) in the upper-left corner of the CRT.

NOTE: If the tablet or computer does not perform the self-test correctly, refer to the HP 9111A Operator's Manual or to the computer's operating manual for instructions on how to obtain servicing assistance.□

How to Use the Example Programs

In each of the following sections, you will develop example programs that build on each other. That is, you will start by entering a small program, then add more lines to that program in each section.

Some of the program lines vary slightly depending upon which Series 80 personal computer you have. When the lines are the same for each computer, the lines are listed once, and are identified in the margin with the caption "Program Lines for HP-85, HP-86, and HP-87." If the program lines are different for any computer, the lines are listed separately and identified with captions that specify the appropriate model numbers. *Use the captions to aid you in entering the correct lines for your computer.* At the end of each section, the entire program is listed separately for each computer so that you can verify your program.

If you have a tape drive or a disc drive, you might find it convenient to store each example program at the end of a section so that you can later retrieve and modify that program. Use the STORE statement as described in your computer's programming manual.

Generate a Cursor to Track the Stylus

The program that you will develop in this section creates a symbol on the CRT of your computer; this symbol is known as a cursor. In addition to creating a cursor, the program will allow the cursor to follow the movement of the stylus on the tablet. That is, as you "draw" on the surface of the tablet, the symbol on the CRT will move in the same directions. This is called "using a cursor to track the stylus."

Here is an outline that breaks down the basic operations involved in tracking the stylus.

1. Setup.
   a. Set up the computer's CRT.
   b. Create a CRT cursor ("+" symbol).
   c. Set up the tablet.
2. Tracking loop (repeated as long as necessary).
   a. Read the stylus location from the tablet.
   b. Display the cursor on the computer's CRT to represent the location of the tip of the stylus.

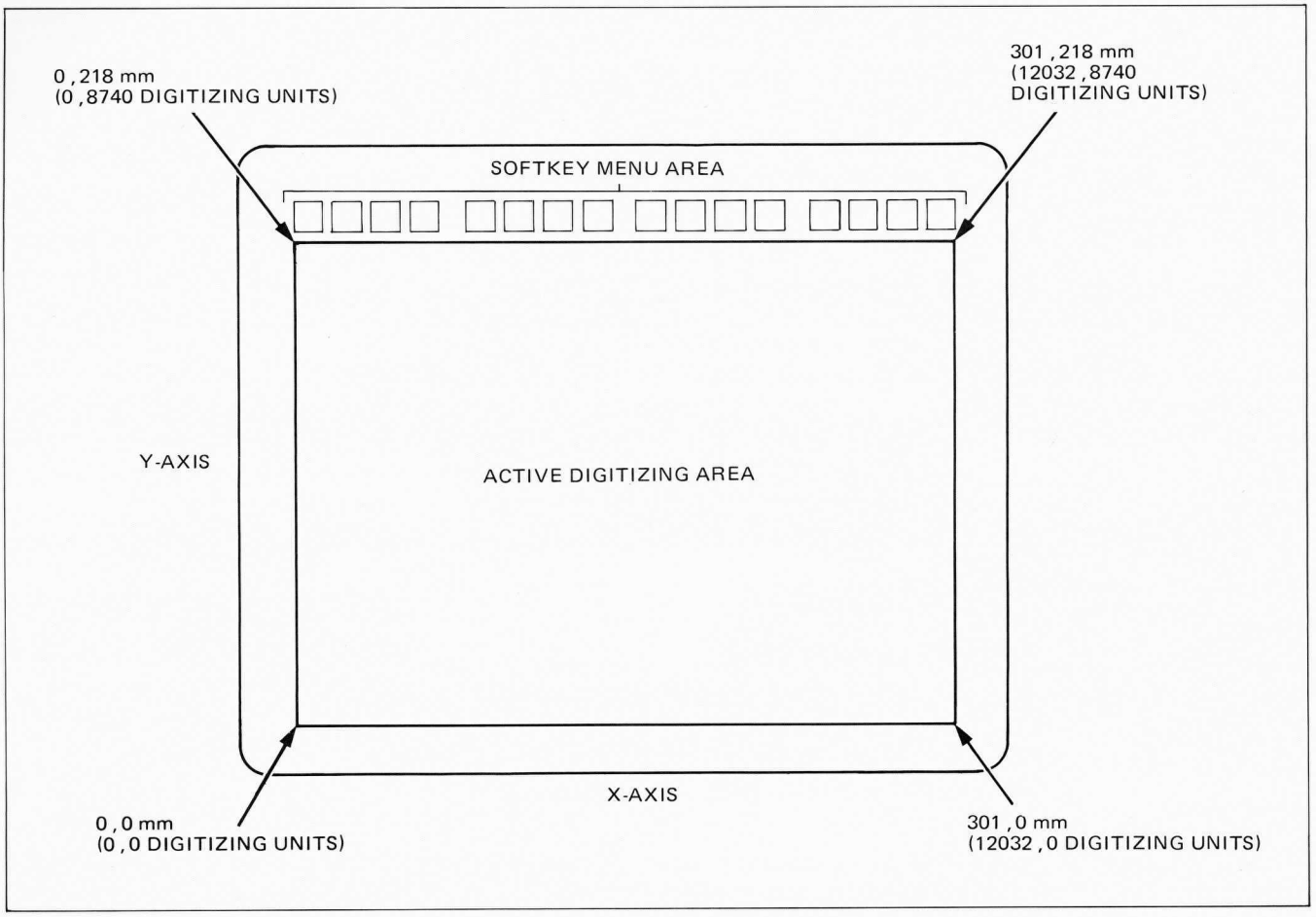
If you key in the program lines when they are presented after the text that describes them, you will have a cursor-tracking program when you finish this section. The whole program is listed at the end of the section so that you can make sure you have entered all of the lines.

Set Up the Computer and the Tablet

To set up the computer's CRT as a drawing area that reflects the movements of the stylus on the tablet, you must first declare that all graphics statements will be directed to the CRT (line 10 on the next page). Then clear the CRT of any previous graphics and specify the graphics mode (lines 20 and 30).

Next, you need to frame an area on the CRT that is proportional to the active digitizing area of the tablet; this will allow you to create a true representation of what you are drawing on the tablet. That is, if you are tracing a square on the tablet with the stylus, you want the cursor on the CRT to trace a square, too (as opposed to a rectangle). You can do this by specifying an area that has the same width-to-height ratio as the tablet's active digitizing area (refer to the following figure). Once you have defined a proportional area on the CRT, you can draw a line around that area and assign a scale that uses the tablet's dimensions in millimetres (lines 40 through 60).

Dimensions of Tablet's Active Digitizing Area.
The width-to-height ratio (X/Y) is 1.4.



Now enter the following lines into your computer. Be careful to enter the correct lines for your specific computer.

Program Lines for HP-85

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,97,0,70
50 FRAME
60 SCALE 0,301,0,218
```

Program Lines for HP-86

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,125,0,51
50 FRAME
60 SCALE 0,301,0,218
```

NOTE: *(Applies to HP-86 only.)* Line 40 assumes that you have an HP 82912A monitor; the CRT dots on this and other monitors are compressed in the X-axis direction, so that there are more dots per millimetre along the X-axis than there are along the Y-axis. You must compensate for this in order to create a drawing area that will be proportional to the tablet.

The following discussion states how the parameters in line 40 were determined for the HP 82912A monitor, using a formula from Appendix B in the "Introduction to the HP-86" manual. If you have a different monitor (such as the HP 82913A), you can use the formula in a similar manner. Here is the formula:

$$\text{LIMIT parameter} = \text{desired boundary (in mm)} \times \frac{\text{HP-87 screen dimension}}{\text{monitor screen dimension}}$$

(The HP-87 screen dimension is 125 mm along the X-axis and 75 mm along the Y-axis.)

To make use of the full size of the HP 82912A monitor in GRAPH mode, the desired boundaries are 103 mm by 75 mm (width-to-height ratio = 1.4). The monitor screen dimension is 103 mm by 110 mm. Substituting in the above formula for the X-axis, LIMIT parameter = $103 \times 125/103 = 125$; for the Y-axis, LIMIT parameter = $75 \times 75/110 = 51$.

For the HP 82913A monitor, the desired boundaries in GRAPH mode are 144 mm by 104 mm. The monitor screen dimension is 144 mm by 150 mm. Substituting in the above formula, the LIMIT parameters would be 0,125,0,52.□

Program Lines for HP-87

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,104,0,75
50 FRAME
60 SCALE 0,301,0,218
```

Now create a cursor (+) for the CRT. Use a FOR . . . NEXT loop to read five numeric values, convert them into string characters, and store them in the string variable T$ (lines 70 through 110). These characters will later be used by the BPLOT statement to define which CRT dots will be turned on to plot the cursor.

Program Lines for HP-85, HP-86, and HP-87

```
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
```

Next, initialize the tablet (line 120). This is a good habit to get into, as it sets the tablet to its power-on condition. That way, your program will run from a known cleared state and will not be affected by any previously set conditions, such as a digitizing mode. This statement completes the setup of the computer's CRT and the tablet.

Program Line for HP-85 and HP-87

```
120 OUTPUT 706;"IN"
```

Program Line for HP-86

```
120 OUTPUT 606;"IN"
```

NOTE: *(Applies to HP-86 only.)* Line 120 assumes that you changed the HP-IB interface card to have a select code of 6 (refer to "Connect the Graphics Tablet to Your Computer"). If you used a different select code,

substitute that code for the first "6" in all OUTPUT, ENTER, ENTER USING, STATUS, and INTR statements throughout this manual.□

Create a Tracking Loop

Start the tracking loop by sending the output cursor instruction, OC, to the tablet (line 130). This step informs the tablet that you will be requesting the X,Y location of the stylus. Next, enter the stylus location from the tablet and store this location in the variables X and Y (line 140). The tablet always outputs the stylus location in digitizing units. However, we scaled the CRT to millimetres (line 60) since you are probably more familiar with millimetres than with digitizing units. So, convert the stylus location to millimetres (line 150); divide the X and Y variables by 40 (1 digitizing unit = 0.025 mm, or 1/40 mm).

Program Lines for HP-85 and HP-87

```
130 OUTPUT 706;"OC"
140 ENTER 706;X,Y
150 X=X/40 @ Y=Y/40
```

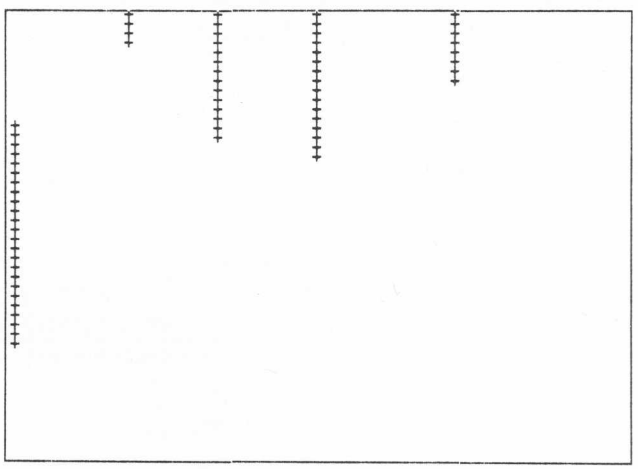
Program Lines for HP-86

```
130 OUTPUT 606;"OC"
140 ENTER 606;X,Y
150 X=X/40 @ Y=Y/40
```

Now it's time to make the cursor appear on the CRT at the correct X,Y position. In order to "draw" the cursor on the CRT, you need to use the BPLOT statement. However, BPLOT has two minor limitations when used in a cursor-tracking program.

1. When plotting very close to the FRAME limits, the BPLOT character repeats itself on the CRT (refer to the following illustration). To avoid this, check the X and Y values and do not plot them if they approach the FRAME limits (lines 160 and 170).

BPLOT Character Too Close to FRAME.
The cursor is drawn several times when it gets too close to the FRAME limits.



2. The X- and Y-coordinate values define the upper-left corner of the character cell in which the BPLOT character (cursor) is generated. This causes the cursor to appear to the lower right of the actual stylus location. Adjust the location of the cursor with the MOVE statements (lines 180 and 200). The cursor will then be plotted at the position on

the CRT that best represents the location of the stylus on the tablet's surface (lines 190 and 210). Notice that plotting T$ once draws the cursor and that plotting it a second time erases the cursor. This causes the cursor to blink on and off as it tracks the stylus. The GOTO statement in line 220 completes the cursor-tracking loop. (For more information on BPLOT characters, refer to your computer's programming manual.)

Now add the following lines to your program.

Program Lines for HP-85, HP-86, and HP-87

```
160 IF X<5 OR X>300 THEN 130
170 IF Y<1 OR Y>216 THEN 130
180 MOVE X-5,Y+2
190 BPLOT T$,1
200 MOVE X-5,Y+2
210 BPLOT T$,1
220 GOTO 130
230 END
```

Complete Program Listings

Now press **RUN** on the computer, and move the stylus around on the tablet. The small blinking cursor on the CRT should follow your motion. When you are finished and want to return to the listing, press **PAUSE** and then any alphanumeric key. On the HP-86 and HP-87, you can also press **SHIFT A/G** to return to the alpha screen after pressing **PAUSE**.

Here is the whole program.

Complete Program for HP-85

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,97,0,70
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 706;"IN"
130 OUTPUT 706;"OC"
140 ENTER 706;X,Y
150 X=X/40 @ Y=Y/40
160 IF X<5 OR X>300 THEN 130
170 IF Y<1 OR Y>216 THEN 130
180 MOVE X-5,Y+2
190 BPLOT T$,1
200 MOVE X-5,Y+2
210 BPLOT T$,1
220 GOTO 130
230 END
```

Complete Program for HP-86

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,125,0,51
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 606;"IN"
130 OUTPUT 606;"OC"
140 ENTER 606;X,Y
150 X=X/40 @ Y=Y/40
160 IF X<5 OR X>300 THEN 130
170 IF Y<1 OR Y>216 THEN 130
180 MOVE X-5,Y+2
190 BPLOT T$,1
200 MOVE X-5,Y+2
210 BPLOT T$,1
220 GOTO 130
230 END
```

NOTE: *(Applies to HP-86 only.)* Lines 120, 130, and 140 assume that you changed the HP-IB interface card to have a select code of 6 (refer to "Connect the Graphics Tablet to Your Computer"). If you used a different select code, substitute that code for the first "6" in all OUTPUT, ENTER, ENTER USING, STATUS, and INTR statements throughout this manual.□

Complete Program for HP-87

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,104,0,75
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 706;"IN"
130 OUTPUT 706;"OC"
140 ENTER 706;X,Y
150 X=X/40 @ Y=Y/40
160 IF X<5 OR X>300 THEN 130
170 IF Y<1 OR Y>216 THEN 130
180 MOVE X-5,Y+2
190 BPLOT T$,1
200 MOVE X-5,Y+2
210 BPLOT T$,1
220 GOTO 130
230 END
```

Draw the Path That the Cursor Tracks

Now that you can track the stylus, it might be nice to draw some lines on the CRT. The first method to look at is the simplest. If you modify line 140 in the previous program, you can read in the pen parameter (P) associated

with the output cursor instruction, OC. This parameter contains a "1" if the stylus is pressed down and a "0" if the stylus is not pressed down. If you branch to a plot routine everytime P = 1, you can draw on the CRT as well as track the cursor (lines 155 and 215 through 226). Modify line 140, and then add the next four lines to your program.

Program Lines for HP-85 and HP-87

```
140 ENTER 706;X,Y,P

155 IF P=1 THEN 225
215 MOVE X,Y
225 DRAW X,Y
226 GOTO 130
```

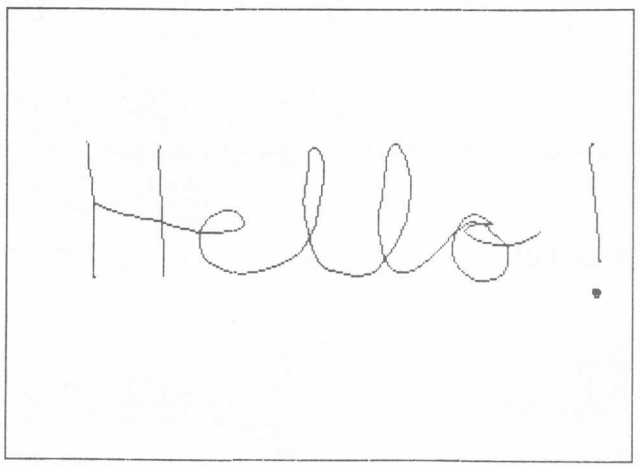
Program Lines for HP-86

```
140 ENTER 606;X,Y,P

155 IF P=1 THEN 225
215 MOVE X,Y
225 DRAW X,Y
226 GOTO 130
```

Now press **RUN**, and the cursor tracking should be operating normally. In addition, you should be able to "draw" on the CRT by pressing down on the stylus while you move the stylus on the platen. Try writing (digitizing) your name or a message!

Copy of CRT after Digitizing the Word "Hello"



Here is the whole program (after renumbering it by entering the REN statement from your computer keyboard).

Complete Program for HP-85

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,97,0,70
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 706;"IN"
```

Complete Program for HP-85
(Continued)

```
130 OUTPUT 706;"OC"
140 ENTER 706;X,Y,P
150 X=X/40 @ Y=Y/40
160 IF P=1 THEN 250
170 IF X<5 OR X>300 THEN 130
180 IF Y<1 OR Y>216 THEN 130
190 MOVE X-5,Y+2
200 BPLOT T$,1
210 MOVE X-5,Y+2
220 BPLOT T$,1
230 MOVE X,Y
240 GOTO 130
250 DRAW X,Y
260 GOTO 130
270 END
```

Complete Program for HP-86

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,125,0,51
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 606;"IN"
130 OUTPUT 606;"OC"
140 ENTER 606;X,Y,P
150 X=X/40 @ Y=Y/40
160 IF P=1 THEN 250
170 IF X<5 OR X>300 THEN 130
180 IF Y<1 OR Y>216 THEN 130
190 MOVE X-5,Y+2
200 BPLOT T$,1
210 MOVE X-5,Y+2
220 BPLOT T$,1
230 MOVE X,Y
240 GOTO 130
250 DRAW X,Y
260 GOTO 130
270 END
```

Complete Program for HP-87

```
10 PLOTTER IS 1
20 GCLEAR
30 GRAPH
40 LIMIT 0,104,0,75
50 FRAME
60 SCALE 0,301,0,218
70 FOR I=1 TO 5
80 READ V
90 T$[I,I]=CHR$(V)
100 NEXT I
110 DATA 4,4,31,4,4
120 OUTPUT 706;"IN"
130 OUTPUT 706;"OC"
140 ENTER 706;X,Y,P
150 X=X/40 @ Y=Y/40
160 IF P=1 THEN 250
170 IF X<5 OR X>300 THEN 130
```

```
180 IF Y<1 OR Y>216 THEN 130
190 MOVE X-5,Y+2
200 BPLOT T$,1
210 MOVE X-5,Y+2
220 BPLOT T$,1
230 MOVE X,Y
240 GOTO 130
250 DRAW X,Y
260 GOTO 130
270 END
```

Set a Digitizing Mode and Create a Data File

The limitation of the program we just developed is that the image exists only on the CRT screen; this means you can't do anything but look at it. The real power is gained when you save data so that the computer can manipulate it. One of the easiest ways to save data is in an array, as you will see in this section.

You will also learn how to set the switch-follow form of the continuous-sample digitizing mode. With this mode, you can draw in the same way as you did with the previous program (i.e., while the stylus is pressed down in an active digitizing area on the tablet, its location will be drawn on the CRT.) However, there are two main differences with specifying a digitizing mode. First, the tablet's **DIGITIZE** light turns on to tell you that the tablet is ready to accept digitized points. Second, setting a digitizing mode allows you to use the output digitized point instruction, OD, to obtain the X,Y stylus location whenever a point has been digitized. If you check the proper bit of the status word (which appears in the parameter S that is output as part of the OC instruction), you can branch to the OD instruction only when a point has been digitized.

Here is an expanded outline to help you understand how the new program works.

1. Setup.
   a. Set up the computer display.
   b. Create a CRT cursor (+).
   c. Set up the tablet.
   d. Dimension arrays for storing digitized-point locations (refer to item 4).
2. Tracking loop (repeated as long as necessary).
   a. Using the OC instruction, read the stylus location from the tablet.
   b. Display the cursor on the CRT to represent the stylus location.
3. Digitizing mode.
   a. Using the SF and CN instructions, set the switch-follow form of the continuous-sample digitizing mode.
   b. Check bit 2 of the status word to determine whether a digitized point is available.

4. Data file and digitizing routine (whenever a digitized point is available).

   a. Using the OD instruction, read the digitized-point location from the tablet. Then store the location in the arrays (the arrays were dimensioned during the setup, item 1).

   b. Draw the digitized-point location on the CRT.

5. When the arrays are full, erase and redraw the CRT by accessing the data stored in the arrays.

Before you start modifying your program, it might help to know a little about the status word and how it is useful in a digitizing routine.

The Status Word

The status word is the primary mechanism for communicating the inner workings of the tablet to the outside world. Conveniently, the status word is included as one of the parameters of the output cursor instruction, OC. If you look at all of the parameters passed back from the tablet after the OC instruction is received, the data would look like this:

X-value	Y-value	Pen	Menu	Status	Error	
00000,	00000,	0,	00,	00000,	000	CR/LF

The group of digits labeled "status" contains the decimal equivalent of the bit values in the status word. The status word is 11 bits long; 9 of the bits represent significant conditions within the tablet. A bit map of the status word looks like this:

Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Pen Press	New Stylus Position	Pen Proximity	Softkey Selected	SRQ*	Error	Ready	Initialized	Digitized Point Available	Clear	Clear

*service request

Bit 2 of the status word is set each time a point is digitized and it is cleared each time this point is read from the tablet. By checking bit 2 of the status word, it is possible to determine when a new point has been digitized. This information is useful in a program such as the one we are developing, because the program can sit in the cursor-tracking loop and wait for you to digitize a point. When you set bit 2 by digitizing a point, the program can branch to a routine that updates the data file and draws the point on the CRT. The program can then return to the cursor-tracking loop to wait for another digitized point.

Set the Digitizing Mode

Now modify the program to set a digitizing mode and to check bit 2 of the status word.

Change line 120 to set the switch-follow form (SF) of the continuous-sample mode (CN). Also change line 140 to allow your computer to read

all of the parameters output with the OC instruction. These parameters are listed here:

X — contains the X-coordinate value
Y — contains the Y-coordinate value
P — contains the pen parameter (1 or 0)
M — contains the menu or softkey value
S — contains the value of the status word
E — contains any error values if they exist

Finally, add line 145, which uses the BIT function to check bit 2 of the status word. If bit 2 is not set (S = 0), the program keeps cycling through the cursor-tracking loop. When a point is digitized, setting bit 2 (S = 1), the program branches to the digitizing routine (this routine starts on line 280).

Program Lines for HP-85 and HP-87

```
120 OUTPUT 706;"IN;CN;SF"
140 ENTER 706;X,Y,P,M,S,E
145 IF BIT (S,2) THEN 280
```

Program Lines for HP-86

```
120 OUTPUT 606;"IN;CN;SF"
140 ENTER 606;X,Y,P,M,S,E
145 IF BIT (S,2) THEN 280
```

Set Up the Data File

Before you add the digitizing routine to the program, set up a data file so that the points collected during the digitizing routine can be stored by the program.

A data file is simply an orderly method for storing data. In this program, the data file consists of three large arrays. The array X(100) will hold the X-coordinates of each digitized point, Y(100) will hold the Y-coordinates of each point, and P(100) will hold the pen status for each point. In order to accept data, the arrays must first be dimensioned.

Insert lines 1 through 4 at the beginning of the program to dimension the arrays to 101 elements each, set the first element of each array to zero, and set the array element counter (J) to zero. This counter will keep track of how many digitized points have been stored in the arrays. Using SHORT and INTEGER (lines 2 and 3) to dimension the arrays conserves computer memory.

Program Lines for HP-85, HP-86, and HP-87

```
1 OPTION BASE 0
2 SHORT X(100),Y(100)
3 INTEGER P(100)
4 X(0),Y(0),P(0),J=0
```

Set Up the Digitizing Routine

Now add the digitizing routine to the end of the program. Start by incrementing the array element counter (J) by one (line 280); element 0 has already been set to 0 for each array, so digitized points will be stored in elements 1 through 100.

Then obtain the X- and Y-coordinate values (S and T) and the pen value (U) of the digitized point (lines 290 and 300). (Reading these values also clears bit 2 of the status word so that the tablet will be able to accept another digitized point.) Next, convert the X- and Y-coordinate values to millimetres, and store these values along with the pen value in arrays X, Y, and P (line 310). (The pen value is 1 while the stylus is pressed and you are digitizing. When you lift the stylus, the value changes to 0 to mark the last point that you digitized before lifting the stylus.) Finally, draw the digitized point on the CRT (line 320) and return to the cursor-tracking loop to await the next digitized point (line 330). If the stylus remains pressed down, the program will immediately branch back to this routine to store and plot another point.

Now add the following lines to your program.

Program Lines for HP-85 and HP-87

```
280 J=J+1
290 OUTPUT 706;"OD"
300 ENTER 706;S,T,U
310 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
320 PLOT X(J),Y(J),P(J)
330 GOTO 130
340 END
```

Program Lines for HP-86

```
280 J=J+1
290 OUTPUT 606;"OD"
300 ENTER 606;S,T,U
310 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
320 PLOT X(J),Y(J),P(J)
330 GOTO 130
340 END
```

NOTE: Before you run this program, you need to delete the lines that are no longer used. These lines are 160, 230, 250, 260, and 270.□

Now run the program. It should operate much as it did before, except that it will eventually overflow the arrays, halting operation of the program. "That's an improvement?" you might say. Well, actually it is, but not because you can overflow the arrays. Rather, it is an improvement because you have stored the data in the arrays. (The computer will tell you that the arrays have overflowed with the "ERROR 55 ON LINE 310: SUBSCRIPT" message. You can ignore the message for now; in the next subsection you'll see how to use the arrays before they overflow.)

Use the Data File to Redraw the CRT

To see why stored data is so important, add another routine to the program to show one way the stored data can be reused by the computer. Add line 285; this line causes the program to branch when the arrays are full. Then add a routine that erases and frames the CRT (lines 350 and 360) and redraws the digitized points that are stored in the arrays (lines 370 through 400). Finally, add line 410 and delete line 340 to complete the program.

Program Lines for HP-85, HP-86, and HP-87

```
285 IF J>100 THEN 350

350 GCLEAR
360 FRAME
370 MOVE X(1),Y(1)
380 FOR J=1 TO 100
390 PLOT X(J),Y(J),P(J)
400 NEXT J
410 END
```

NOTE: Remember to delete line 340.□

Complete Program Listings

Now run the program again. This time, when the arrays are filled, the CRT is automatically cleared and the data points are replotted.

This is only one way that you can use data files. You can also use data files to store the data on your magnetic tape or disc, plot the data on a plotter, transfer the data to other programs or computers, and so on. You may wish to experiment with routines to manipulate your data files. However, there is more to learn about the graphics tablet — in the next section we'll turn to programming the softkeys so that you can tell the computer what to do without using the keyboard.

If you renumber the program, it should look like this:

Complete Program for HP-85

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,97,0,70
90 FRAME
100 SCALE 0,301,0,218
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 706;"IN;CN;SF"
170 OUTPUT 706;"OC"
180 ENTER 706;X,Y,P,M,S,E
190 IF BIT (S,2) THEN 280
200 X=X/40 @ Y=Y/40
210 IF X<5 OR X>300 THEN 170
220 IF Y<1 OR Y>216 THEN 170
230 MOVE X-5,Y+2
240 BPLOT T$,1
250 MOVE X-5,Y+2
260 BPLOT T$,1
270 GOTO 170
280 J=J+1
290 IF J>100 THEN 350
300 OUTPUT 706;"OD"
310 ENTER 706;S,T,U
320 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
```

Complete Program for HP-85 *(Continued)*

```
330 PLOT X(J),Y(J),P(J)
340 GOTO 170
350 GCLEAR
360 FRAME
370 MOVE X(1),Y(1)
380 FOR J=1 TO 100
390 PLOT X(J),Y(J),P(J)
400 NEXT J
410 END
```

Complete Program for HP-86

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,125,0,51
90 FRAME
100 SCALE 0,301,0,218
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 606;"IN;CN;SF"
170 OUTPUT 606;"OC"
180 ENTER 606;X,Y,P,M,S,E
190 IF BIT (S,2) THEN 280
200 X=X/40 @ Y=Y/40
210 IF X<5 OR X>300 THEN 170
220 IF Y<1 OR Y>216 THEN 170
230 MOVE X-5,Y+2
240 BPLOT T$,1
250 MOVE X-5,Y+2
260 BPLOT T$,1
270 GOTO 170
280 J=J+1
290 IF J>100 THEN 350
300 OUTPUT 606;"OD"
310 ENTER 606;S,T,U
320 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
330 PLOT X(J),Y(J),P(J)
340 GOTO 170
350 GCLEAR
360 FRAME
370 MOVE X(1),Y(1)
380 FOR J=1 TO 100
390 PLOT X(J),Y(J),P(J)
400 NEXT J
410 END
```

Complete Program for HP-87

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,104,0,75
90 FRAME
100 SCALE 0,301,0,218
```

Complete Program for HP-87
(Continued)

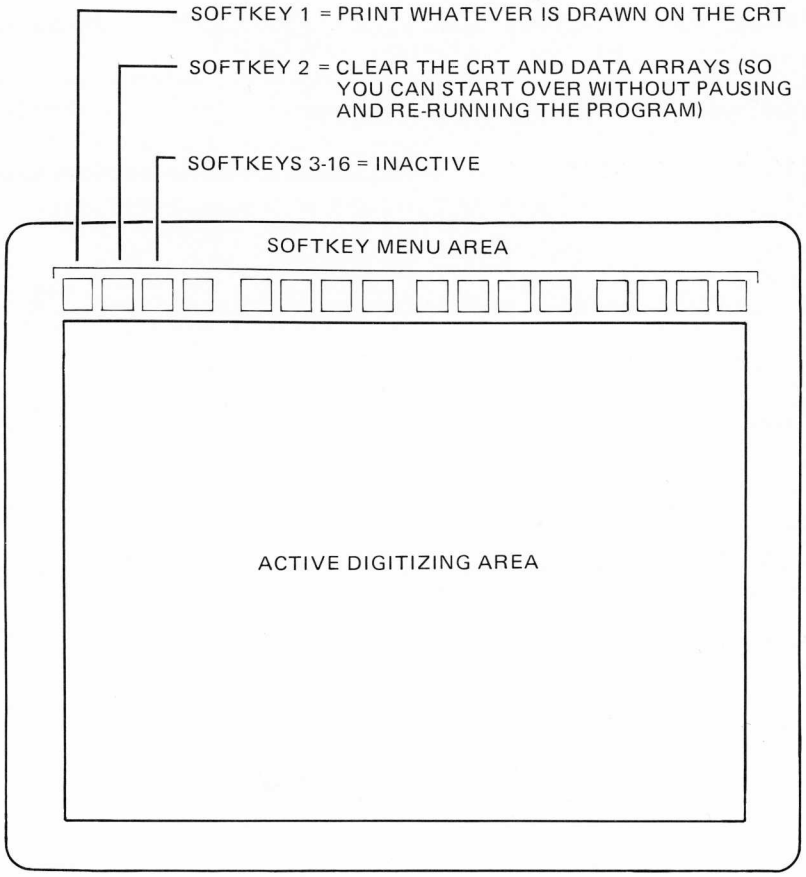
```
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 706;"IN;CN;SF"
170 OUTPUT 706;"OC"
180 ENTER 706;X,Y,P,M,S,E
190 IF BIT (S,2) THEN 280
200 X=X/40 @ Y=Y/40
210 IF X<5 OR X>300 THEN 170
220 IF Y<1 OR Y>216 THEN 170
230 MOVE X-5,Y+2
240 BPLOT T$,1
250 MOVE X-5,Y+2
260 BPLOT T$,1
270 GOTO 170
280 J=J+1
290 IF J>100 THEN 350
300 OUTPUT 706;"OD"
310 ENTER 706;S,T,U
320 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
330 PLOT X(J),Y(J),P(J)
340 GOTO 170
350 GCLEAR
360 FRAME
370 MOVE X(1),Y(1)
380 FOR J=1 TO 100
390 PLOT X(J),Y(J),P(J)
400 NEXT J
410 END
```

Define Some Menu Softkeys

One big advantage of the tablet is the built-in softkey structure. The sixteen numbered boxes along the upper edge of the platen can be interpreted by the tablet as "keys," and used to control the execution of a program on your computer. This allows you to control your program completely from the tablet without having to use the computer's keyboard. Because the softkeys offer you a choice of options, they are often referred to as "menu" softkeys.

You can define the functions of the softkeys by assigning subroutines in your program; then, when you want the program to execute the subroutine, all you have to do is press the stylus on the appropriate softkey box. In this section, we will define softkeys **1** and **2** to perform the routines listed in the following illustration.

Softkey Locations and Assignments



Monitor the Softkeys

In order to make use of the routines that you assign to the softkeys, you must have your program continually monitor the softkeys to determine whether one has been selected. If you use the OC instruction, this is easy.* One of the parameters output by the OC instruction is the variable M (refer to line 180 in your current program). When a softkey is selected, the number of that softkey (an integer from 1 to 16) appears in the variable M; M contains the number 0 until a softkey is selected.

You can use the ON M GOSUB statement (line 186 on the next page) to set up your program so that it branches to the appropriate subroutine whenever a certain softkey has been selected. In our program, M = 1 causes the program to branch to the subroutine beginning on line 420, and M = 2 causes the program to branch to the subroutine beginning on line 440. Softkeys **3** through **16** are inactive; if you select any of these keys, the program encounters a RETURN statement (line 470). This allows the program to remain in the cursor-tracking loop until another softkey is selected. Of course, if you wish to assign more subroutines to softkeys **3** through **16**, it will be very simple for you to change the appropriate line numbers to cause the program to branch to new subroutines.

You need to add a couple more lines to complete the setup of this program segment. First, you need to allow the program to remain in the cursor-tracking loop whenever M does not contain a valid softkey value

*You can also use the read softkey instruction, RS, as you will see in the section "Control Program Execution with Service-Request Interrupts."

(line 182). Second, you need to clear the softkey value with the set key instruction, SK (line 184); this allows the computer to recognize when you have selected another softkey.

Now add lines 182 through 186 to your program.

Program Lines for HP-85 and HP-87

```
182 IF M<1 OR M>16 THEN 190
184 OUTPUT 706;"SK"
186 ON M GOSUB 420,440,470,470,470,470,470,470,470,
    470,470,470,470,470,470,470
```

Program Lines for HP-86

```
182 IF M<1 OR M>16 THEN 190
184 OUTPUT 606;"SK"
186 ON M GOSUB 420,440,470,470,470,470,470,470,470,
    470,470,470,470,470,470,470
```

NOTE: *(Applies to HP-86 and HP-87 only.)* If you do *not* have a printer, use "470" instead of "420" for the first line number in the ON M GOSUB statement. This will cause softkey **1** to be inactive in case you accidentally press the stylus down on it. For the purposes of this manual, do *not* delete lines 420 and 430 (the print routine) from your program. If you do, the lines in your program will be in a different sequence and references in text to line numbers might then be confusing.□

Set Up the Softkey Routines

Now add the softkey routines. Lines 420 and 430 cause the drawing on the CRT to be printed whenever softkey **1** is selected; lines 440 through 470 cause the CRT and arrays to be cleared whenever softkey **2** is selected.

Program Lines for HP-85

```
420 COPY
430 RETURN
440 GCLEAR
450 FRAME
460 J=0
470 RETURN
```

Program Lines for HP-86

```
420 PRINTER IS 701 @ DUMP GRAPHICS
430 RETURN
440 GCLEAR
450 FRAME
460 J=0
470 RETURN
```

NOTE: *(Applies to HP-86 only.)* Line 420 assumes that you have an HP 82905B printer. If you do not have a printer, key in lines 420 and 430, for the reasons noted previously.

If you have a different printer, you may need to add parameters to the DUMP GRAPHICS statement; refer to the HP-86/87 Plotter ROM Owner's Manual. In addition, if your printer connects to the HP-86 via the HP-IB interface that the tablet is using (instead of connecting via the **PRINTER** receptacle on the computer), you need to change the select code in line 420. Using the select code "6" and printer address "01," line 420 should read PRINTER IS 601 @ DUMP GRAPHICS.□

Program Lines for HP-87

```
420 PRINTER IS 701 @ DUMP GRAPHICS
430 RETURN
440 GCLEAR
450 FRAME
460 J=0
470 RETURN
```

NOTE: *(Applies to HP-87 only.)* Line 420 assumes that you have an HP 82905B printer. If you do not have a printer, key in lines 420 and 430, for the reasons noted previously. If you have a different printer, you may need to add parameters to the DUMP GRAPHICS statement; refer to the HP-86/87 Plotter ROM Owner's Manual.□

Complete Program Listings

Now run the program. Draw a picture, and then press the stylus in the square labeled **1**. The picture on the CRT should be copied onto your printer (if you have one). You can also clear the CRT and reset the array-element counter by pressing the stylus in the square labeled **2**.

NOTE: *(Applies to HP-86 only.)* If you have an HP-86, the printed copy of the CRT might appear slightly compressed along the Y-axis. This is because there are more dots per unit along the Y-axis on the printer than there are on the CRT. In other words, the spaces that you see between the dots on the CRT are eliminated on the printer.□

Here's the current program as it looks after it is renumbered.

Complete Program for HP-85

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,97,0,70
90 FRAME
100 SCALE 0,301,0,218
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 706;"IN;CN;SF"
170 OUTPUT 706;"OC"
180 ENTER 706;X,Y,P,M,S,E
190 IF M<1 OR M>16 THEN 220
200 OUTPUT 706;"SK"
210 ON M GOSUB 450,470,500,500,500,500,500,500,500,
    500,500,500,500,500,500,500
220 IF BIT (S,2) THEN 310
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 170
250 IF Y<1 OR Y>216 THEN 170
260 MOVE X-5,Y+2
270 BPLOT T$,1
280 MOVE X-5,Y+2
290 BPLOT T$,1
300 GOTO 170
310 J=J+1
320 IF J>100 THEN 380
```

Complete Program for HP-85
(Continued)

```
330 OUTPUT 706;"OD"
340 ENTER 706;S,T,U
350 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
360 PLOT X(J),Y(J),P(J)
370 GOTO 170
380 GCLEAR
390 FRAME
400 MOVE X(1),Y(1)
410 FOR J=1 TO 100
420 PLOT X(J),Y(J),P(J)
430 NEXT J
440 END
450 COPY
460 RETURN
470 GCLEAR
480 FRAME
490 J=0
500 RETURN
```

Complete Program for HP-86

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,125,0,51
90 FRAME
100 SCALE 0,301,0,218
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 606;"IN;CN;SF"
170 OUTPUT 606;"OC"
180 ENTER 606;X,Y,P,M,S,E
190 IF M<1 OR M>16 THEN 220
200 OUTPUT 606;"SK"
210 ON M GOSUB 450,470,500,500,500,500,500,500,500,
    500,500,500,500,500,500,500
220 IF BIT (S,2) THEN 310
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 170
250 IF Y<1 OR Y>216 THEN 170
260 MOVE X-5,Y+2
270 BPLOT T$,1
280 MOVE X-5,Y+2
290 BPLOT T$,1
300 GOTO 170
310 J=J+1
320 IF J>100 THEN 380
330 OUTPUT 606;"OD"
340 ENTER 606;S,T,U
350 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
360 PLOT X(J),Y(J),P(J)
370 GOTO 170
380 GCLEAR
390 FRAME
400 MOVE X(1),Y(1)
410 FOR J=1 TO 100
420 PLOT X(J),Y(J),P(J)
```

Complete Program for HP-86 *(Continued)*

```
430 NEXT J
440 END
450 PRINTER IS 701 @ DUMP GRAPHICS
460 RETURN
470 GCLEAR
480 FRAME
490 J=0
500 RETURN
```

NOTE: *(Applies to HP-86 only.)* If you do not have a printer, line 210 should read ON M GOSUB 500,470,500,500,500,500,500,500,500,500,500, 500,500,500,500,500.

Also remember that if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 450. If your printer is not connected via the **PRINTER** receptacle on the computer, line 450 might also need a different select code in the PRINTER IS statement. Refer to the note in the text for this section.□

Complete Program for HP-87

```
10 OPTION BASE 0
20 SHORT X(100),Y(100)
30 INTEGER P(100)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,104,0,75
90 FRAME
100 SHOW 0,301,218
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 OUTPUT 706;"IN;CN;SF"
170 OUTPUT 706;"OC"
180 ENTER 706;X,Y,P,M,S,E
190 IF M<1 OR M>16 THEN 220
200 OUTPUT 706;"SK"
210 ON M GOSUB 450,470,500,500,500,500,500,500,500,
    500,500,500,500,500,500,500
220 IF BIT (S,2) THEN 310
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 170
250 IF Y<1 OR Y>216 THEN 170
260 MOVE X-5,Y+2
270 BPLOT T$,1
280 MOVE X-5,Y+2
290 BPLOT T$,1
300 GOTO 170
310 J=J+1
320 IF J>100 THEN 380
330 OUTPUT 706;"OD"
340 ENTER 706;S,T,U
350 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
360 PLOT X(J),Y(J),P(J)
370 GOTO 170
380 GCLEAR
390 FRAME
400 MOVE X(1),Y(1)
410 FOR J=1 TO 100
```

Complete Program for HP-87
(Continued)

```
420 PLOT X(J),Y(J),P(J)
430 NEXT J
440 END
450 PRINTER IS 701 @ DUMP GRAPHICS
460 RETURN
470 GCLEAR
480 FRAME
490 J=0
500 RETURN
```

NOTE: *(Applies to HP-87 only.)* If you do not have a printer, line 210 should read ON M GOSUB 500,470,500,500,500,500,500,500,500,500,500, 500,500,500,500,500.

Also remember that if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 450. Refer to the note in the text for this section.□

Control Program Execution with Service-Request Interrupts

Now that you have covered the basics of operating the tablet, it's time to explore using advanced I/O (input/output) programming techniques to simplify and speed up your program. Specifically, you will next be modifying the program to operate under "interrupt-driven" processing.

Overview of Interrupt-Driven Programming

Interrupt-driven operations eliminate the need for constantly checking the bits in the status word. Instead, they make use of the tablet's ability to generate a service request (SRQ) whenever the tablet requires attention. An interrupt-driven program is set up to demand and recognize the SRQ under certain defined conditions. When an SRQ occurs, it interrupts the HP-IB interface, causing an end-of-line branch in the operation of the program. A service routine then responds to the interrupt before allowing normal program execution to continue. Finally, the interrupts are "re-enabled," so that the program can recognize future interrupts. The HP-85 I/O Programming Guide (with supplements for the HP-86 and HP-87) provides more details on interrupt-driven programming.

Here is an outline for an interrupt-driven program, based on the above discussion. This program will be added to your current program.

1. Set up interrupt conditions.
   a. Define end-of-line branch (to the service routine).
   b. Set SRQ-generating conditions in the tablet.
   c. Enable interrupts.
2. Set up interrupt response (i.e., the service routine).
   a. Determine why the tablet generated the SRQ.
   b. Respond appropriately to the SRQ.
   c. Reenable the interrupts.

In your program, an SRQ will be generated whenever a point has been digitized or a softkey has been selected. If a point has been digitized, the service routine will direct program execution to the routine that plots the point on the CRT; if a softkey has been selected, the service routine will direct program execution to the routine that defines different responses for each softkey.

Set Up the Interrupt Conditions

First, tell the computer what to do when an interrupt occurs in the HP-IB interface (change line 160 as shown below). Then set up the tablet to generate the SRQ whenever a softkey is selected or a point is digitized (line 161). You can do this with the input mask instruction, IM, using a parameter of 132. This parameter is obtained by adding the S-mask bit values for digitized points (4) and selected softkeys (128). Finally, enable the computer to interrupt program execution when an SRQ occurs (line 162).

Program Lines for HP-85 and HP-87

```
160 ON INTR 7 GOSUB 301

161 OUTPUT 706;"IM,132"
162 ENABLE INTR 7;8
```

Program Lines for HP-86

```
160 ON INTR 6 GOSUB 301

161 OUTPUT 606;"IM,132"
162 ENABLE INTR 6;8
```

When an interrupt (SRQ) occurs, the computer completes the line it is currently executing before processing the interrupt. There are some places where an interrupt can cause some problems, such as in the middle of an I/O sequence. Using the "@" symbol to combine the I/O sequence into a one-line operation eliminates any problem an interrupt might create. Change line 170 to read as shown below; then delete line 180. Another place where an interrupt can cause a problem is in the BPLOT routine; change line 260 as shown next to put the BPLOT routine into one line. Then delete lines 270, 280, and 290.

Program Lines for HP-85 and HP-87

```
170 OUTPUT 706;"OC" @ ENTER 706;X,Y

260 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
```

Program Lines for HP-86

```
170 OUTPUT 606;"OC" @ ENTER 606;X,Y

260 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
```

NOTE: Remember to delete lines 180, 270, 280, and 290. And don't run the program yet! You still need to add the interrupt response and make some other modifications.□

Set Up the Interrupt Response

Now insert the service routine that responds to the interrupt (SRQ). First, read the interrupt cause in status register 1 of the HP-IB interface (line 301). This is necessary to ensure that program execution is not resumed

until the service routine for the SRQ is performed. In addition, it clears the interrupt from the status register so that the computer can recognize future interrupts.

Next, use the output status instruction, OS, to obtain the status word from the tablet (line 302). (This instruction is also useful because it disables the SRQ and resets the tablet's status-word bits to their default conditions.) Now test the status word with the BIT function to determine why the tablet generated the SRQ (lines 303 through 305). If the digitized-point bit (bit 2) is set, branch to the routine that plots the digitized point on the CRT; if the selected-softkey bit (bit 7) is set, branch to the softkey routine. Finally, reenable the SRQ interrupts so that the program will recognize future interrupts (line 306).

Now add the following lines to your program.

Program Lines for HP-85 and HP-87

```
301 STATUS 7,1;A
302 OUTPUT 706;"OS"
303 ENTER 706;S1
304 IF BIT (S1,2) THEN GOSUB 310
305 IF BIT (S1,7) THEN GOSUB 371
306 ENABLE INTR 7;8 @ RETURN
```

Program Lines for HP-86

```
301 STATUS 6,1;A
302 OUTPUT 606;"OS"
303 ENTER 606;S1
304 IF BIT (S1,2) THEN GOSUB 310
305 IF BIT (S1,7) THEN GOSUB 371
306 ENABLE INTR 6;8 @ RETURN
```

Additional Modifications

Now modify the softkey routine that you added under "Defining the Menu Softkeys." The modified routine uses the read softkey instruction, RS, which outputs a decimal number that corresponds to the softkey box that was selected (lines 371 and 372). The ON K GOSUB statement (line 373) then causes the program to branch to the appropriate routine for the softkey that was selected.

Now add the following lines to your program.

Program Lines for HP-85 and HP-87

```
371 OUTPUT 706;"RS1"
372 ENTER 706;K
373 ON K GOSUB 450,470,500,500,500,500,500,500,500,
    500,500,500,500,500,500,500
374 RETURN
```

Program Lines for HP-86

```
371 OUTPUT 606;"RS1"
372 ENTER 606;K
373 ON K GOSUB 450,470,500,500,500,500,500,500,500,
    500,500,500,500,500,500,500
374 RETURN
```

NOTE: Delete lines 190, 200, and 210; these lines are no longer used.

Also, if you do not have a printer, line 373 should read ON K GOSUB 500,470,500,500,500,500,500,500,500,500,500,500,500,500,500,500.□

Since you changed line 160 to start the setup of interrupt conditions, you need to redefine the digitizing modes set up by the old line 160; do this by adding line 55 as shown below. Also delete line 220 since this statement is now part of the service routine (i.e., it is now on line 304). In addition, the statements that plot the digitized point on the CRT are now part of the subroutine called by line 304, so change line 370 from a GOTO to a RETURN, as shown below.

Program Lines for HP-85 and HP-87

```
55 OUTPUT 706;"IN;CN;SF"

370 RETURN
```

Program Lines for HP-86

```
55 OUTPUT 606;"IN;CN;SF"

370 RETURN
```

NOTE: Remember to delete line 220.□

Finally, increase the data file so that you can digitize more points. Simply change the following program lines.

Program Lines for HP-85, HP-86, and HP-87

```
20 SHORT X(1000),Y(1000)
30 INTEGER P(1000)
320 IF J>1000 THEN 380
410 FOR J=1 TO 1000
```

Complete Program Listings

Now run the program. It should operate as before, but the execution will be faster and you will be able to digitize more points before the data arrays are filled.

You made quite a few modifications to the program in this section. You might want to check the text to be sure you completed all steps before you renumber the program. Here is the current program as it looks after it is renumbered.

Complete Program for HP-85

```
10 OPTION BASE 0
20 SHORT X(1000),Y(1000)
30 INTEGER P(1000)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 OUTPUT 706;"IN;CN;SF"
70 GCLEAR
80 GRAPH
90 LIMIT 0,97,0,70
100 FRAME
110 SCALE 0,301,0,218
120 FOR I=1 TO 5
130 READ V
140 T$[I,I]=CHR$(V)
150 NEXT I
160 DATA 4,4,31,4,4
170 ON INTR 7 GOSUB 260
180 OUTPUT 706;"IM,132"
```

Complete Program for HP-85
(Continued)

```
190 ENABLE INTR 7;8
200 OUTPUT 706;"OC" @ ENTER 706;X,Y
210 X=X/40 @ Y=Y/40
220 IF X<5 OR X>300 THEN 200
230 IF Y<1 OR Y>216 THEN 200
240 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
250 GOTO 200
260 STATUS 7,1;A
270 OUTPUT 706;"OS"
280 ENTER 706;S1
290 IF BIT (S1,2) THEN GOSUB 320
300 IF BIT (S1,7) THEN GOSUB 390
310 ENABLE INTR 7;8 @ RETURN
320 J=J+1
330 IF J>1000 THEN 430
340 OUTPUT 706;"OD"
350 ENTER 706;S,T,U
360 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
370 PLOT X(J),Y(J),P(J)
380 RETURN
390 OUTPUT 706;"RS1"
400 ENTER 706;K
410 ON K GOSUB 500,520,550,550,550,550,550,550,550,
    550,550,550,550,550,550,550
420 RETURN
430 GCLEAR
440 FRAME
450 MOVE X(1),Y(1)
460 FOR J=1 TO 1000
470 PLOT X(J),Y(J),P(J)
480 NEXT J
490 END
500 COPY
510 RETURN
520 GCLEAR
530 FRAME
540 J=0
550 RETURN
```

Complete Program for HP-86

```
10 OPTION BASE 0
20 SHORT X(1000),Y(1000)
30 INTEGER P(1000)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 OUTPUT 606;"IN;CN;SF"
70 GCLEAR
80 GRAPH
90 LIMIT 0,125,0,51
100 FRAME
110 SCALE 0,301,0,218
120 FOR I=1 TO 5
130 READ V
140 T$[I,I]=CHR$(V)
150 NEXT I
160 DATA 4,4,31,4,4
170 ON INTR 6 GOSUB 260
180 OUTPUT 606;"IM,132"
190 ENABLE INTR 6;8
200 OUTPUT 606;"OC" @ ENTER 606;X,Y
210 X=X/40 @ Y=Y/40
220 IF X<5 OR X>300 THEN 200
```

Complete Program for HP-86 *(Continued)*

```
230 IF Y<1 OR Y>216 THEN 200
240 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
250 GOTO 200
260 STATUS 6,1;A
270 OUTPUT 606;"OS"
280 ENTER 606;S1
290 IF BIT (S1,2) THEN GOSUB 320
300 IF BIT (S1,7) THEN GOSUB 390
310 ENABLE INTR 6;8 @ RETURN
320 J=J+1
330 IF J>1000 THEN 430
340 OUTPUT 606;"OD"
350 ENTER 606;S,T,U
360 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
370 PLOT X(J),Y(J),P(J)
380 RETURN
390 OUTPUT 606;"RS1"
400 ENTER 606;K
410 ON K GOSUB 500,520,550,550,550,550,550,550,550,
    550,550,550,550,550,550,550
420 RETURN
430 GCLEAR
440 FRAME
450 MOVE X(1),Y(1)
460 FOR J=1 TO 1000
470 PLOT X(J),Y(J),P(J)
480 NEXT J
490 END
500 PRINTER IS 701 @ DUMP GRAPHICS
510 RETURN
520 GCLEAR
530 FRAME
540 J=0
550 RETURN
```

NOTE: *(Applies to HP-86 only.)* If you do not have a printer, line 410 should read ON K GOSUB 550,520,550,550,550,550,550,550,550,550,550, 550,550,550,550,550.

Also remember that if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 500. If your printer is not connected via the **PRINTER** receptacle on the computer, line 500 might also need a different select code in the PRINTER IS statement.□

Complete Program for HP-87

```
10 OPTION BASE 0
20 SHORT X(1000),Y(1000)
30 INTEGER P(1000)
40 X(0),Y(0),P(0),J=0
50 PLOTTER IS 1
60 OUTPUT 706;"IN;CN;SF"
70 GCLEAR
80 GRAPH
90 LIMIT 0,104,0,75
100 FRAME
110 SCALE 0,301,0,218
120 FOR I=1 TO 5
130 READ V
140 T$[I,I]=CHR$(V)
150 NEXT I
```

Complete Program for HP-87
(Continued)

```
160 DATA 4,4,31,4,4
170 ON INTR 7 GOSUB 260
180 OUTPUT 706;"IM,132"
190 ENABLE INTR 7;8
200 OUTPUT 706;"OC" @ ENTER 706;X,Y
210 X=X/40 @ Y=Y/40
220 IF X<5 OR X>300 THEN 200
230 IF Y<1 OR Y>216 THEN 200
240 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
250 GOTO 200
260 STATUS 7,1;A
270 OUTPUT 706;"OS"
280 ENTER 706;S1
290 IF BIT (S1,2) THEN GOSUB 320
300 IF BIT (S1,7) THEN GOSUB 390
310 ENABLE INTR 7;8 @ RETURN
320 J=J+1
330 IF J>1000 THEN 430
340 OUTPUT 706;"OD"
350 ENTER 706;S,T,U
360 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
370 PLOT X(J),Y(J),P(J)
380 RETURN
390 OUTPUT 706;"RS1"
400 ENTER 706;K
410 ON K GOSUB 500,520,550,550,550,550,550,550,550,
    550,550,550,550,550,550,550
420 RETURN
430 GCLEAR
440 FRAME
450 MOVE X(1),Y(1)
460 FOR J=1 TO 1000
470 PLOT X(J),Y(J),P(J)
480 NEXT J
490 END
500 PRINTER IS 701 @ DUMP GRAPHICS
510 RETURN
520 GCLEAR
530 FRAME
540 J=0
550 RETURN
```

NOTE: *(Applies to HP-87 only.)* If you do not have a printer, line 410 should read ON K GOSUB 550,520,550,550,550,550,550,550,550,550,550, 550,550,550,550,550.

Also remember that if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 500.□

Make Use of Audible Prompts

Now that you have been selecting softkeys to perform various routines, you can probably see that it would be nice if the tablet somehow acknowledged to you that a softkey was indeed selected. This is possible with the beep instruction, BP, which allows you to generate musical tones of a specific duration and amplitude. These musical tones can be used to indicate various program conditions or to prompt you when the program is ready for a digitized point or menu selection. Ascending frequency

sequences are recommended to indicate inquiry, and descending sequences to indicate acknowledgement. Add descending sequences to your program by inserting the following lines.

Program Lines for HP-85 and HP-87

```
11   DIM K$[25]
12   K$="BP36,50,3;BP34;BP32;BP30"
401  OUTPUT 706;K$
```

Program Lines for HP-86

```
11 DIM K$[25]
12 K$="BP36,50,3;BP32;BP30"
401 OUTPUT 606;K$
```

Now run the program and select a softkey. You should get the "softkey-recognized" prompt.

Instead of renumbering the program now, continue modifying your current program by proceeding to the next section. There you will define a few more softkeys to represent different digitizing modes.

Define More Digitizing Modes

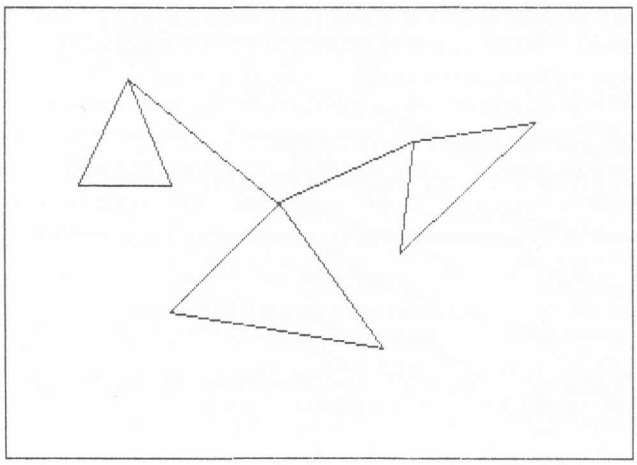
The tablet has two types of digitizing modes: the single-sample mode, set by the SG instruction, and the continuous-sample mode, set by the CN instruction. The continuous-sample mode has two variations that interpret the press of the stylus differently; one variation is set by the switch follow instruction, SF, and the other by the switch normal instruction, SN. In this section, you will modify the program to define softkeys **3**, **4**, and **5** to set the single-sample mode, the switch-follow mode, and the switch-normal mode, respectively. You will not be able to run the program until the end of the section, when all modifications have been made.

The Single-Sample Mode

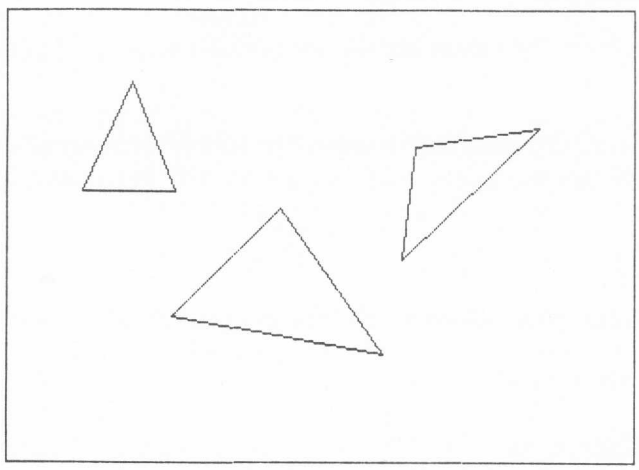
In the single-sample mode, a single point is digitized each time the stylus is pressed against the platen. This mode is useful for defining angular shapes. For example, rather than digitize the entire outline of a triangle using one of the continuous-sample modes, you can digitize the three endpoints using the single-sample mode. Then your program can draw lines between those points. You can also use this mode to digitize single points from strip charts or other graphs.

In this program, you will define softkey **3** to draw lines between individual digitized points as long as you remain in the single-sample mode (refer to the first figure on the next page). If you press softkey **3** again, or press any other softkey, no line will be drawn between the last digitized point and the next digitized point. This allows you to draw separate shapes or contours without connecting them to each other, as shown in the second figure on the next page.

NOTE: Do not try to reproduce these examples until you have entered the entire program for this section.□



Triangles Drawn while Remaining in the Single-Sample Mode.
Press softkey **3** *and digitize each of the endpoints for all of the triangles.*



Triangles Drawn by Selecting the Single-Sample Mode for Each One.
Press softkey **3** *and digitize the endpoints of the first triangle. Then press softkey* **3** *again and digitize the endpoints of the second triangle. Repeat for the third triangle.*

Now add the following routine to set the single-sample mode. First, send the single-sample mode instruction, SG, to the tablet (line 560). Then set a flag using the variable F1 (line 570); this flag will be explained in the next paragraph. Finally, force the pen parameter P(J) to zero so that the last digitized point will be terminated in the data file (line 580). This is what allows you to digitize a new point without drawing a line from the last point (as described in the previous paragraph). Finally, terminate the single-sample mode routine (line 590).

Program Lines for HP-85 and HP-87

```
560 OUTPUT 706;"SG"
570 F1=1
580 P(J)=0
590 RETURN
```

Program Lines for HP-86

```
560 OUTPUT 606;"SG"
570 F1=1
580 P(J)=0
590 RETURN
```

Your program tracks the stylus with the CRT cursor whenever digitized points are not being plotted on the CRT. In the single-sample mode, only one point is digitized at a time. You can move the stylus (and the cursor) anywhere between each point that you digitize. Because of this, the

program does not remember where the last digitized point was. If you want to draw a line from the last point when you digitize a new point, you must send the program back to the original point. To do this, you can use a flag, F1; set this flag only when you are in the single-sample mode (line 570, shown previously). (In all other modes, lines are drawn continuously while you are digitizing, so the program knows where the last digitized point was.) When this flag is set, send the cursor to the last digitized point; add line 311 as shown below.

Program Line for HP-85, HP-86, and HP-87

```
311 IF F1=1 THEN PLOT X(J),Y(J),P(J)
```

NOTE: Remember that you cannot run the program yet — you need to add a few more lines.□

The Continuous-Sample Modes

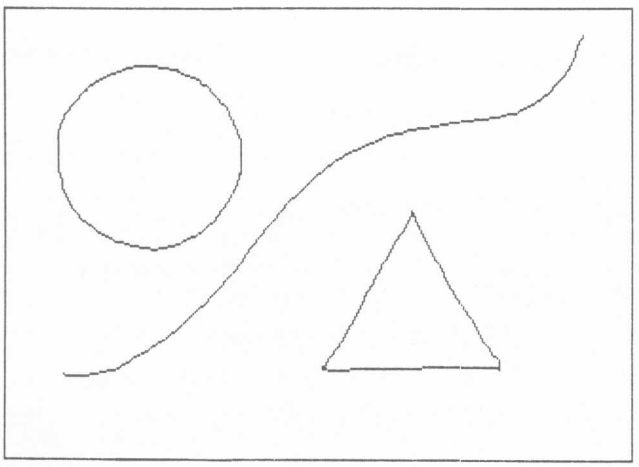
In the continuous-sample modes, points are loaded into the tablet's I/O register at a steady rate (60 points per second). If the computer does not input a point from the tablet before the tablet loads a new point into the register, the old point is replaced by the new point. Therefore, some interim points may not be saved in the arrays.* The continuous-sample modes are useful for tracing line contours such as circles or curves. The flow of points can be started and stopped in different ways by pressing the stylus, as described in the following subsections.

The Switch-Follow Mode

The switch-follow mode lets you digitize very intuitively: while you press down on the stylus, points are read into the tablet's I/O register (digitized). When you release pressure from the stylus, points are no longer digitized. This process is very similar to drawing with a pen or pencil. If you apply pressure again, you can start digitizing a new contour. (You do not need to press the softkey again, as you do when you are in the single-sample mode, described above. Refer to the following figure.)

NOTE: Do not try to reproduce this example until you have entered the entire program for this section.□

Contours Drawn with the Switch-Follow Mode. *Press softkey 4. Then press the stylus down and trace the contour while exerting pressure on the stylus. When finished, release the stylus. Repeat for each new contour (you do not need to press the softkey again).*



*This problem can be alleviated by setting a slower rate with the cursor rate instruction, CR. The rate you choose depends upon how many devices are connected on the HP-IB. You will need to experiment to choose the cursor rate; you might start with a rate of 30 points per second and then vary the rate up or down depending upon how smoothly the computer seems to be tracking the cursor.

Now add the following routine to set the switch-follow mode. This routine is set up in the same way as described previously for the single-sample mode, except that the flag F1 is not necessary, so it is cleared.

```
600 OUTPUT 706;"CN;SF"
610 F1=0
620 P(J)=0
630 RETURN
```

Program Lines for HP-85 and HP-87

```
600 OUTPUT 606;"CN;SF"
610 F1=0
620 P(J)=0
630 RETURN
```

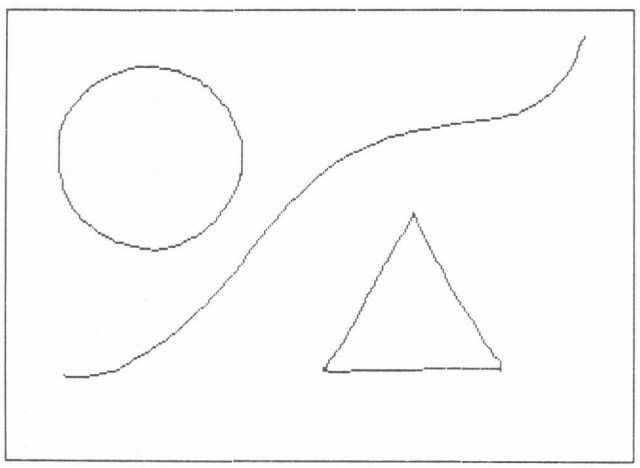
Program Lines for HP-86

NOTE: Remember that you cannot run the program yet.□

The Switch-Normal Mode

The switch-normal mode is a convenient way to digitize continuous line contours. Pressing the stylus down once starts the flow of digitized points; pressing the stylus again stops the flow. This way you can trace a contour without constantly exerting pressure on the stylus (refer to the following figure). This mode is less tiring, plus if you are tracing a contour on paper, you are less likely to tear the paper.

NOTE: Do not try to reproduce this example until you have entered the entire program for this section.□



Contours Drawn with the Switch-Normal Mode. *Press softkey* 5. *Next press the stylus down and then release pressure with the tip still touching the tablet. Trace the contour. When finished with the contour, press the stylus down again. Repeat for each new contour (you do not need to press the softkey again).*

Now add the following routine to set the switch-normal mode. This routine is set up in the same way as described previously for the single-sample mode, except that the flag F1 is not necessary, so it is cleared.

```
640 OUTPUT 706;"CN;SN"
650 F1=0
660 P(J)=0
670 RETURN
```

Program Lines for HP-85 and HP-87

Program Lines for HP-86

```
640 OUTPUT 606;"CN;SN"
650 F1=0
660 P(J)=0
670 RETURN
```

NOTE: Remember that you cannot run the program yet.□

Additional Modifications

You must make a few changes to your program before it will operate properly. First, initialize the flag F1 so that it is always cleared in the default digitizing mode (line 40). (The default mode is the switch-follow mode, as defined in line 60 of your current program.) Then change line 290 so that the GOSUB starts with line 311 instead of 320. Finally, change line 410 to allow access to the new softkey routines. You only need to change the third, fourth, and fifth line numbers (560, 600, and 640) in this statement.

Program Lines for HP-85, HP-86, and HP-87

```
40 X(0),Y(0),P(0),J,F1=0

290 IF BIT (S1,2) THEN GOSUB 311

410 ON K GOSUB 500,520,560,600,640,550,550,
    550,550,550,550,550,550,550,550,550
```

NOTE: *(Applies to HP-86 and HP-87 only.)* If you do not have a printer, line 410 should read ON K GOSUB 550,520,560,600,640,550,550,550,550, 550,550,550,550,550,550,550,550.□

Summary of the Softkey Definitions

Now run the program. It should operate as it did before, in the default switch-follow mode. Now select softkey **3** and press the stylus down in the active area of the tablet. A small dot should appear on the CRT. Press the stylus at a new location on the tablet, and a line will appear between the last point and the new one. This interconnecting will continue until you select a new softkey. Selecting the mode already in effect continues operation in the mode, but starts a new contour. Experiment with each of the modes — you might first try to recreate the figures shown on the previous pages.

Here is a summary of the operations that are under softkey control:

Softkey **1** — copy the CRT to the printer
Softkey **2** — erase the CRT and clear the data files
Softkey **3** — single-sample mode
Softkey **4** — switch-follow, continuous-sample mode
Softkey **5** — switch-normal, continuous-sample mode

Complete Program Listings

Here is the current program as it looks after it is renumbered. Before renumbering the program, you might want to check to be sure that you completed all of the steps listed in the text.

```
10 OPTION BASE 0
20 DIM K$[25]
30 K$="BP36,50,3;BP34;BP32;BP30"
40 SHORT X(1000),Y(1000)
50 INTEGER P(1000)
60 X(0),Y(0),P(0),J,F1=0
70 PLOTTER IS 1
80 OUTPUT 706;"IN;CN;SF"
90 GCLEAR
100 GRAPH
110 LIMIT 0,97,0,70
120 FRAME
130 SCALE 0,301,0,218
140 FOR I=1 TO 5
150 READ V
160 T$[I,I]=CHR$(V)
170 NEXT I
180 DATA 4,4,31,4,4
190 ON INTR 7 GOSUB 280
200 OUTPUT 706;"IM,132"
210 ENABLE INTR 7;8
220 OUTPUT 706;"OC" @ ENTER 706;X,Y
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 220
250 IF Y<1 OR Y>216 THEN 220
260 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
270 GOTO 220
280 STATUS 7,1;A
290 OUTPUT 706;"OS"
300 ENTER 706;S1
310 IF BIT (S1,2) THEN GOSUB 340
320 IF BIT (S1,7) THEN GOSUB 420
330 ENABLE INTR 7;8 @ RETURN
340 IF F1=1 THEN PLOT X(J),Y(J),P(J)
350 J=J+1
360 IF J>1000 THEN 470
370 OUTPUT 706;"OD"
380 ENTER 706;S,T,U
390 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
400 PLOT X(J),Y(J),P(J)
410 RETURN
420 OUTPUT 706;"RS1"
430 ENTER 706;K
440 OUTPUT 706;K$
450 ON K GOSUB 540,560,600,640,680,590,590,590,590,
    590,590,590,590,590,590,590
460 RETURN
470 GCLEAR
480 FRAME
490 MOVE X(1),Y(1)
500 FOR J=1 TO 1000
510 PLOT X(J),Y(J),P(J)
520 NEXT J
530 END
540 COPY
550 RETURN
560 GCLEAR
570 FRAME
580 J=0
590 RETURN
600 OUTPUT 706;"SG"
610 F1=1
```

Complete Program for HP-85 *(Continued)*

```
620 P(J)=0
630 RETURN
640 OUTPUT 706;"CN;SF"
650 F1=0
660 P(J)=0
670 RETURN
680 OUTPUT 706;"CN;SN"
690 F1=0
700 P(J)=0
710 RETURN
```

Complete Program for HP-86

```
10 OPTION BASE 0
20 DIM K$[25]
30 K$="BP36,50,3;BP34;BP32;BP30"
40 SHORT X(1000),Y(1000)
50 INTEGER P(1000)
60 X(0),Y(0),P(0),J,F1=0
70 PLOTTER IS 1
80 OUTPUT 606;"IN;CN;SF"
90 GCLEAR
100 GRAPH
110 LIMIT 0,125,0,51
120 FRAME
130 SCALE 0,301,0,218
140 FOR I=1 TO 5
150 READ V
160 T$[I,I]=CHR$(V)
170 NEXT I
180 DATA 4,4,31,4,4
190 ON INTR 6 GOSUB 280
200 OUTPUT 606;"IM,132"
210 ENABLE INTR 6;8
220 OUTPUT 606;"OC" @ ENTER 606;X,Y
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 220
250 IF Y<1 OR Y>216 THEN 220
260 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
270 GOTO 220
280 STATUS 6,1;A
290 OUTPUT 606;"OS"
300 ENTER 606;S1
310 IF BIT (S1,2) THEN GOSUB 340
320 IF BIT (S1,7) THEN GOSUB 420
330 ENABLE INTR 6;8 @ RETURN
340 IF F1=1 THEN PLOT X(J),Y(J),P(J)
350 J=J+1
360 IF J>1000 THEN 470
370 OUTPUT 606;"OD"
380 ENTER 606;S,T,U
390 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
400 PLOT X(J),Y(J),P(J)
410 RETURN
420 OUTPUT 606;"RS1"
430 ENTER 606;K
440 OUTPUT 606;K$
450 ON K GOSUB 540,560,600,640,680,590,590,590,590,
    590,590,590,590,590,590,590
460 RETURN
470 GCLEAR
480 FRAME
490 MOVE X(1),Y(1)
```

Complete Program for HP-86
(Continued)

```
500 FOR J=1 TO 1000
510 PLOT X(J),Y(J),P(J)
520 NEXT J
530 END
540 PRINTER IS 701 @ DUMP GRAPHICS
550 RETURN
560 GCLEAR
570 FRAME
580 J=0
590 RETURN
600 OUTPUT 306;"SG"
610 F1=1
620 P(J)=0
630 RETURN
640 OUTPUT 306;"CN;SF"
650 F1=0
660 P(J)=0
670 RETURN
680 OUTPUT 306;"CN;SN"
690 F1=0
700 P(J)=0
710 RETURN
```

NOTE: *(Applies to HP-86 only.)* If you do not have a printer, line 450 should read ON K GOSUB 590,560,600,640,680,590,590,590,590,590,590, 590,590,590,590,590.

Also remember that if you have a printer that is not an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 540. If your printer is not connected via the **PRINTER** receptacle on the computer, line 540 might also need a different select code in the PRINTER IS statement.□

Complete Program for HP-87

```
10 OPTION BASE 0
20 DIM K$[25]
30 K$="BP36,50,3;BP34;BP32;BP30"
40 SHORT X(1000),Y(1000)
50 INTEGER P(1000)
60 X(0),Y(0),P(0),J,F1=0
70 PLOTTER IS 1
80 OUTPUT 706;"IN;CN;SF"
90 GCLEAR
100 GRAPH
110 LIMIT 0,104,0,75
120 FRAME
130 SCALE 0,301,0,218
140 FOR I=1 TO 5
150 READ V
160 T$[I,I]=CHR$(V)
170 NEXT I
180 DATA 4,4,31,4,4
190 ON INTR 7 GOSUB 280
200 OUTPUT 706;"IM,132"
210 ENABLE INTR 7;8
220 OUTPUT 706;"OC" @ ENTER 706;X,Y
230 X=X/40 @ Y=Y/40
240 IF X<5 OR X>300 THEN 220
250 IF Y<1 OR Y>216 THEN 220
260 MOVE X-5,Y+2 @ BPLOT T$,1 @ MOVE X-5,Y+2 @
    BPLOT T$,1
270 GOTO 220
```

Complete Program for HP-87
(Continued)

```
280 STATUS 7,1;A
290 OUTPUT 706;"OS"
300 ENTER 706;S1
310 IF BIT (S1,2) THEN GOSUB 340
320 IF BIT (S1,7) THEN GOSUB 420
330 ENABLE INTR 7;8 @ RETURN
340 IF F1=1 THEN PLOT X(J),Y(J),P(J)
350 J=J+1
360 IF J>1000 THEN 470
370 OUTPUT 706;"OD"
380 ENTER 706;S,T,U
390 X(J)=S/40 @ Y(J)=T/40 @ P(J)=U
400 PLOT X(J),Y(J),P(J)
410 RETURN
420 OUTPUT 706;"RS1"
430 ENTER 706;K
440 OUTPUT 706;K$
450 ON K GOSUB 540,560,600,640,680,590,590,590,590,
    590,590,590,590,590,590,590
460 RETURN
470 GCLEAR
480 FRAME
490 MOVE X(1),Y(1)
500 FOR J=1 TO 1000
510 PLOT X(J),Y(J),P(J)
520 NEXT J
530 END
540 PRINTER IS 701 @ DUMP GRAPHICS
550 RETURN
560 GCLEAR
570 FRAME
580 J=0
590 RETURN
600 OUTPUT 706;"SG"
610 F1=1
620 P(J)=0
630 RETURN
640 OUTPUT 706;"CN;SF"
650 F1=0
660 P(J)=0
670 RETURN
680 OUTPUT 706;"CN;SN"
690 F1=0
700 P(J)=0
710 RETURN
```

NOTE: *(Applies to HP-87 only.)* If you do not have a printer, line 450 should read ON K GOSUB 590,560,600,640,680,590,590,590,590,590,590, 590,590,590,590,590.

Also remember that if you have a printer that is not an HP 82905B, you might need parameters in the DUMP GRAPHICS statment in line 540.□

Add Speed with the Binary Data Transfer Mode

As more and more flexibility is designed into a program, certain speed penalties are encountered. You can see this with the program that you have developed up to this point. If you require speed and limited flexibility, this section presents a somewhat faster program that digitizes only when the stylus is pressed down.

Overview of Binary Data Transfer

This program is faster primarily because it uses the tablet's binary data transfer mode. This mode is the default response of the tablet when no other data transfer is being undertaken. In other words, if no instruction that requires data to be loaded into the tablet's I/O register has been sent to the tablet, certain binary data are always available. (For example, the OC and OD instructions load data into the I/O register; if your program sends these instructions, you cannot access the tablet's binary data.)

The binary data consist of six bytes, represented in two's complement form. The first two bytes are the binary representation of the X-location of the stylus, the next two bytes are the Y-location, and the last two bytes are the current status. You can access these data using the binary image specifier "W" available with your I/O ROM (see line 160 in the following listing). For more detailed information, refer to the HP-85 I/O Programming Guide (with supplements for the HP-86 and HP-87) and the HP 9111A Programming Manual.

Program Description

This program operates very similarly to the other programs developed in this manual, except that only one type of digitizing is allowed (equivalent to the switch-follow, continuous-sample mode). Whenever you are not digitizing, the program cycles through the cursor-tracking loop.

If you press the stylus down in the active digitizing area of the tablet, the stylus location is plotted on the CRT. If you continue to move the stylus while pressing it down, lines are plotted to represent its movement. When you lift the stylus, the program returns to the cursor-tracking loop until you press down on the stylus again. Two softkeys are functional: softkey **1** copies the CRT to your printer, and softkey **2** clears the CRT and the data files.

The following paragraphs explain the important lines in the program. The line numbers refer to the program, which is listed in the next subsection, "Complete Program Listings."

Lines 10 through 150 set up the computer, with the exception of line 40, which sets up the tablet. Line 100, the SCALE statement, scales the CRT using the tablet's default unit of measure, digitizing units. Using digitizing units speeds up execution since the program does not have to convert digitizing units to millimetres.

Line 160 accesses the stylus' X,Y position and the tablet's current status using the binary data transfer mode. The integer variables (X, Y, and P) used with this transfer are initialized in line 20. The image specifier "#" suppresses the final end-of-line sequence that is output with the last byte; this prevents the computer from interpreting the end-of-line characters as binary data. The image specifier "W" causes the binary data to be output in sets of two 8-bit bytes.

Lines 170 and 180 check bits in the status word (appearing in the variable P). If bit 10 (the pressed-pen bit) is set, the program branches to the "plot data" routine (lines 240 through 280). This routine uses the same data obtained in line 160 to plot on the CRT and add to the data file. If bit 7 (the selected-softkey bit) is set, the program branches to the routine that checks to see which softkey was selected (lines 290 through 320); the ON K GOSUB routine then branches to the proper line. If bit 7 or bit 10 are

not set, the cursor-tracking loop is performed over and over (lines 190 through 230).

Once the data file of 1000 is full (line 250), the program branches to the ending routine. In this routine, the CRT is cleared, the data is replotted, and the program is ended (lines 330 through 390).

Complete Program Listings

This is a new program, although many lines are similar to previous programs. Therefore, before you enter the following lines into your computer, enter the SCRATCH statement from your keyboard. This clears the computer memory for the new program.

Complete Program for HP-85

```
10 OPTION BASE 0
20 INTEGER X(1000),Y(1000),P(1000),X,Y,P
30 X(0),Y(0),P(0),J=0
40 OUTPUT 706;"IN"
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,97,0,70
90 FRAME
100 SCALE 0,12032,0,8740
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 ENTER 706 USING "#,W,W,W";X,Y,P
170 IF BIT (P,10) THEN 240
180 IF BIT (P,7) THEN GOSUB 290
190 IF X<200 OR X>11992 THEN 160
200 IF Y<40 OR Y>8660 THEN 160
210 MOVE X-200,Y+80 @ BPLOT T$,1 @ MOVE X-200,Y+80 @
    BPLOT T$,1
220 P(J)=0
230 GOTO 160
240 J=J+1
250 IF J>1000 THEN 330
260 X(J)=X @ Y(J)=Y @ P(J)=1
270 PLOT X,Y,1
280 GOTO 160
290 OUTPUT 706;"RS1"
300 ENTER 706;K
310 ON K GOSUB 400,420,450,450,450,450,450,450,450,
    450,450,450,450,450,450,450
320 RETURN
330 GCLEAR
340 FRAME
350 MOVE X(1),Y(1)
360 FOR J=1 TO 1000
370 PLOT X(J),Y(J),P(J)
380 NEXT J
390 END
400 COPY
410 RETURN
420 GCLEAR
430 FRAME
440 J=0
450 RETURN
```

```
10 OPTION BASE 0
20 INTEGER X(1000),Y(1000),P(1000),X,Y,P
30 X(0),Y(0),P(0),J=0
40 OUTPUT 606;"IN"
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,125,0,51
90 FRAME
100 SCALE 0,12032,0,8740
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 ENTER 606 USING "#,W,W,W";X,Y,P
170 IF BIT (P,10) THEN 240
180 IF BIT (P,7) THEN GOSUB 290
190 IF X<200 OR X>11992 THEN 160
200 IF Y<40 OR Y>8660 THEN 160
210 MOVE X-200,Y+80 @ BPLOT T$,1 @  MOVE X-200,Y+80 @
    BPLOT T$,1
220 P(J)=0
230 GOTO 160
240 J=J+1
250 IF J>1000 THEN 330
260 X(J)=X @ Y(J)=Y @ P(J)=1
270 PLOT X,Y,1
280 GOTO 160
290 OUTPUT 606;"RS1"
300 ENTER 606;K
310 ON K GOSUB 400,420,450,450,450,450,450,450,450,
    450,450,450,450,450,450,450
320 RETURN
330 GCLEAR
340 FRAME
350 MOVE X(1),Y(1)
360 FOR J=1 TO 1000
370 PLOT X(J),Y(J),P(J)
380 NEXT J
390 END
400 PRINTER IS 701 @ DUMP GRAPHICS
410 RETURN
420 GCLEAR
430 FRAME
440 J=0
450 RETURN
```

NOTE: *(Applies to HP-86 only.)* If you do not have a printer, line 310 should read ON K GOSUB 450,420,450,450,450,450,450,450,450,450,450, 450,450,450,450,450. (Refer to the notes in "Define Some Menu Softkeys.")

Also, if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 400. (Refer to the HP-86/87 Plotter ROM Owner's Manual.) If your printer is *not* connected via the **PRINTER** receptacle on the computer, line 400 might also need a different select code in the PRINTER IS statement. (Refer to the notes in "Define Some Menu Softkeys.")

In addition, this program assumes that you changed the HP-IB interface card to have a select code of "6." (Refer to "Connect the Graphics Tablet

to Your Computer.") If you used a different select code, substitute that code for the first "6" in all OUTPUT, ENTER, and ENTER USING statements.□

Complete Program for HP-87

```
10 OPTION BASE 0
20 INTEGER X(1000),Y(1000),P(1000),X,Y,P
30 X(0),Y(0),P(0),J=0
40 OUTPUT 706;"IN"
50 PLOTTER IS 1
60 GCLEAR
70 GRAPH
80 LIMIT 0,104,0,75
90 FRAME
100 SCALE 0,12032,0,8740
110 FOR I=1 TO 5
120 READ V
130 T$[I,I]=CHR$(V)
140 NEXT I
150 DATA 4,4,31,4,4
160 ENTER 706 USING "#,W,W,W";X,Y,P
170 IF BIT (P,10) THEN 240
180 IF BIT (P,7) THEN GOSUB 290
190 IF X<200 OR X>11992 THEN 160
200 IF Y<40 OR Y>8660 THEN 160
210 MOVE X-200,Y+80 @ BPLOT T$,1 @ MOVE X-200,Y+80 @
    BPLOT T$,1
220 P(J)=0
230 GOTO 160
240 J=J+1
250 IF J>1000 THEN 330
260 X(J)=X @ Y(J)=Y @ P(J)=1
270 PLOT X,Y,1
280 GOTO 160
290 OUTPUT 706;"RS1"
300 ENTER 706;K
310 ON K GOSUB 400,420,450,450,450,450,450,450,450,
    450,450,450,450,450,450,450
320 RETURN
330 GCLEAR
340 FRAME
350 MOVE X(1),Y(1)
360 FOR J=1 TO 1000
370 PLOT X(J),Y(J),P(J)
380 NEXT J
390 END
400 PRINTER IS 701 @ DUMP GRAPHICS
410 RETURN
420 GCLEAR
430 FRAME
440 J=0
450 RETURN
```

NOTE: *(Applies to HP-87 only.)* If you do not have a printer, line 310 should read ON K GOSUB 450,420,450,450,450,450,450,450,450,450,450,450,450,450,450,450. (Refer to the notes in "Define Some Menu Softkeys.")

Also, if you have a printer that is *not* an HP 82905B, you might need parameters in the DUMP GRAPHICS statement in line 400. (Refer to the HP-86/87 Plotter ROM Owner's Manual.)□

Appendix A
HP-GL Syntax

How to Send HP-GL Instructions to the Tablet

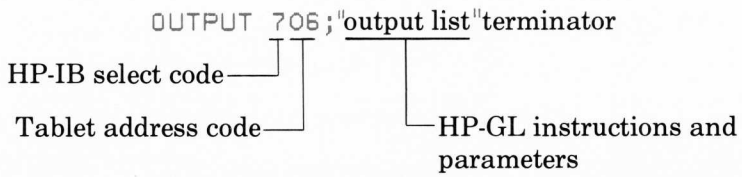
HP-GL instructions are sent to the tablet as literal strings contained within certain BASIC statements, such as PRINT and OUTPUT. In the OUTPUT 706;"IN" statement, the HP-GL mnemonic IN (two-letter code for the initialize instruction) is the literal string that is sent to the tablet.

Your computer must have an I/O ROM in order to read data from the tablet. Although you can send data to the tablet without an I/O ROM (using PRINT statements), the example programs in this manual use the OUTPUT statements, which can only be used with the I/O ROM. Only the OUTPUT statements are described in this appendix.

NOTE: The PRINT statements work similarly to the OUTPUT statements. If you wish to use PRINT statements, read through this appendix and refer to your Series 80 BASIC programming manual for information on sending literal strings using the PRINTER IS, PRINT, and PRINT USING statements. You have to use PRINTER IS with the tablet's address to declare that all PRINT statements will go to the tablet. Then you can send HP-GL instructions in the PRINT statements, following the same format guidelines presented in this appendix for OUTPUT statements.□

OUTPUT Statement Format

The OUTPUT statement sends HP-GL instructions to the device whose address is specified. The format of the statement follows.



The HP-IB select code and tablet address code are assumed to have factory settings of 7 and 06, respectively. If you have an HP-86, you need to change the select code to a number other than 7, between 3 and 10. This manual uses the select code 6; refer to "Connect the Graphics Tablet to Your Computer."

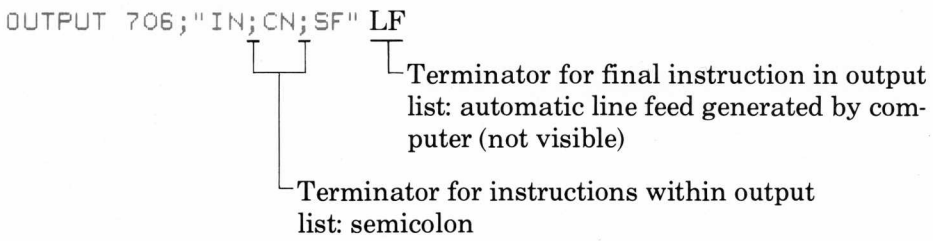
The output list consists of the HP-GL instructions and their parameters. However, you must format the statement differently, depending upon whether you have constant or variable parameters. These are discussed separately after the next two paragraphs.

The output list can also consist of string variables, if these string variables have been assigned with HP-GL instructions that have constant parameters. Refer to the following example.

```
A$="BP5,75,5;BP;BP"
OUTPUT 706;A$
```

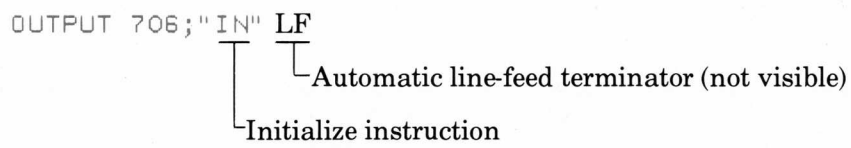
Assign the string variable A$ with the beep instruction, BP, and its constant parameters. Then send A$ to the tablet.

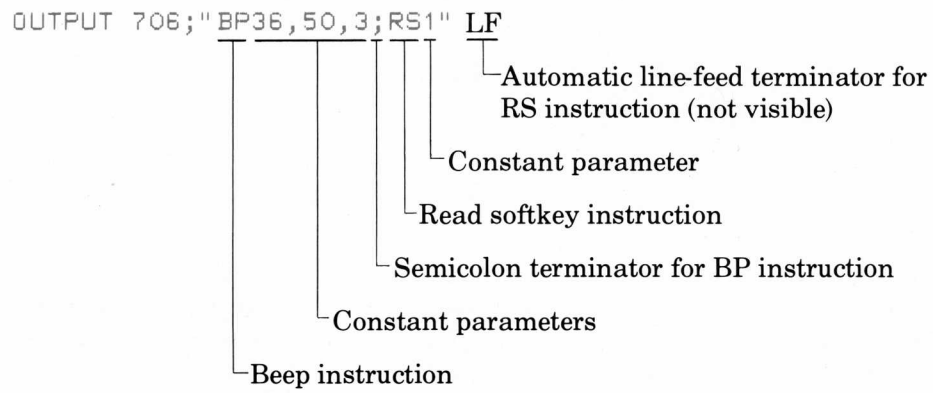
Each HP-GL instruction must be followed by a terminator; only one terminator can be used for each instruction. A semicolon is used to terminate an instruction when that instruction is followed by another instruction within one output list. An automatic line feed (LF) is used to terminate the last instruction in the output list; this is accomplished by pressing the **END LINE** key on the computer after entering the OUTPUT statement into a program. Refer to the following examples.

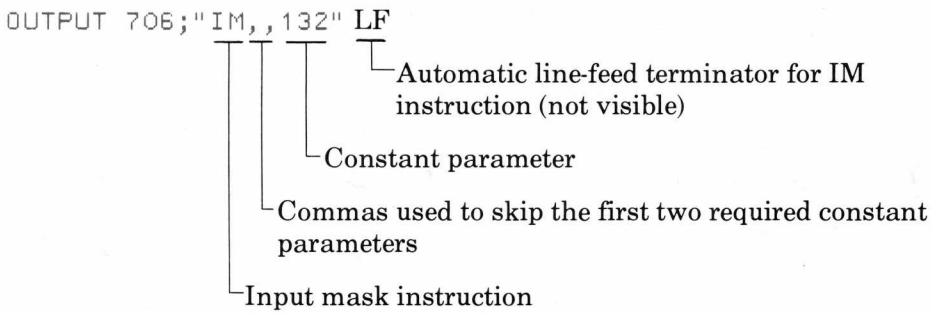


Instructions with Constant Parameters

Instructions with constant parameters must be entirely enclosed in quotation marks. As shown in the examples below, HP-GL instructions are separated by semicolon terminators. When an instruction has more than one parameter, commas separate the parameters. If you do not include all parameters, use commas to replace the parameters not included. Refer to the following examples.







Instructions with Variable Parameters

Instructions with variable parameters require a little more complex format. For example, suppose you want to use the variables X2 and Y2 to represent the X- and Y-coordinates of the scaling point P2 for the input points instruction, IP. In addition, you want to use the constant value of 100 for both the X- and Y-coordinates of the scaling point P1. If you enclose the entire instruction in quotation marks as follows

```
OUTPUT 706;"IP100,100,X2,Y2"
```

you are telling the tablet that the X- and Y-coordinates of P2 are literally X2 and Y2, which is *not* acceptable to the tablet.

You have three choices for sending HP-GL instructions with variable parameters. The method you choose will depend on the considerations discussed below. Three different ways of transmitting variable parameters are explained:

- using a comma to separate variables,
- using a semicolon to separate variables, and
- using the image specifier "K" with the OUTPUT USING statement.

Commas as Separators for Variables

Using a comma to separate variable parameters causes the computer to send data in a free-field format with up to 20 extra spaces in each field. The instructions

```
X2=800
Y2=600
OUTPUT 706;"IP",100,100,X2,Y2
```

cause the following data to be sent with 17 spaces in each field, except the first, which has 18 spaces:

```
IP              100             100             800             600             CRLF
```

The disadvantage of this method is that a lot of unnecessary traffic gets sent over the bus. Also, you can quickly exceed the specified line length and find automatic line feeds (LF) and carriage returns (CR) inserted in unwanted places. Remember that a line feed terminates an HP-GL instruction.

Semicolons as Separators for Variables

Using a semicolon to separate variable parameters produces a compact field with no more than a leading and trailing space. The instructions

```
X2=800
Y2=600
OUTPUT 706;"IP";100;100;X2;Y2
```

cause the following to be sent to the tablet:

```
IP  100  100  800  600    CRLF
```

This method eliminates many of the excess spaces and is relatively simple to enter.

The OUTPUT USING "K" Statement

Another way to control the format of information sent to the tablet is with image specifiers. Using "K" as an image specifer causes output to be sent to the tablet with no leading or trailing blanks. The instructions

```
X2=800
Y2=600
OUTPUT USING "K";"IP",100,",",100,",",X2,",","Y2"
```

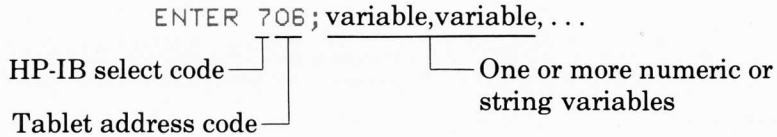
cause the following to be sent to the tablet:

```
IP100,100,800,600CRLF
```

Note that you must specify all separators in quotation marks so that the parameters do not run together. The advantage of using this statement is that it eliminates the extra spaces found between parameters and before the CR and LF when using a semicolon. This speeds up your data transfer. The disadvantage is the additional data entry time and possibility for error that can occur when entering more quotation marks and separators.

How to Receive the Tablet's Output Data

The ENTER statement (available with the I/O ROM) must be used to obtain output data from the tablet. The ENTER statement tells the computer to read the tablet's output into one or more variables. The format of the statement follows.



The HP-IB select code and tablet addess code are assumed to have factory settings of 7 and 06, respectively. As noted in the previous section, your HP-IB select code will be different if you have an HP-86.

A semicolon is used to separate the device selector code from the list of items being entered; commas are used to separate items within the list. The items in the list are variable names that correspond to the data you expect the tablet to output. (If you do not provide a variable name, the tablet will not know where to put its data.) You cannot use commas to skip over any of the items you expect the tablet to output. However, you do not have to read all of the data items. Refer to the examples below.

```
OUTPUT 706;"OI"
ENTER 706;A$
```

Use a string variable. The output identifier instruction, OI, causes the tablet to output the tablet's model number as a string variable.

```
OUTPUT 706;"OC"
ENTER 706;X,Y,P,M,S,E
```

Use numeric variables. The output cursor instruction, OC, causes the tablet to output location and status data as numeric variables.

```
OUTPUT 706;"OC"
ENTER 706;,,,M,S,E
```

Do not use commas to skip over variables. The tablet will not recognize this format.

```
OUTPUT 706;"OC"
ENTER 706;X,Y,P
```

You can read only a few of the expected data items if you do not skip any. For example, if you only want to read P, you must also read X and Y; you do not have to read M, S, and E.

Appendix B
Glossary

Address — The characters sent by the computer to specify which device will receive information along a particular interface bus.

AGL — A Graphics Language. An extension of BASIC that provides graphics statements.

BASIC — Beginner's All-Purpose Symbolic Instruction Code. The computer language understood by the Series 80 personal computers.

Binary — A number system with a base of 2, using the numerals 0 and 1.

Binary default response — The format in which the tablet sends data unless otherwise requested. The binary default response consists of the stylus location and status word sent as six 8-bit bytes.

Binary data transfer mode — The method of utilizing the tablet's binary default response.

Bit — The smallest part of a byte that contains intelligible information. A binary digit.

Bus — A signal line or a set of signal lines used by an interface system to which a number of devices are connected and over which messages are carried.

Byte — A character sent over data lines, normally consisting of eight bits.

CRT — Cathode Ray Tube. Provides the display on a video terminal.

Carriage return (CR) — A special character that causes movement back to the beginning of an existing line.

Continuous digitizing — A digitizing mode in which X,Y coordinates are automatically picked up at regular time intervals as you move the stylus on the tablet's surface.

Contour — A continuous line or other shape (such as a circle).

Cursor — An indicator that guides data entry or locates existing data on the CRT.

Cursor tracking — The use of the cursor on the CRT to track the movement of the stylus on the tablet's surface.

Data rate — The rate at which the tablet checks the stylus position in points per second.

Default — A value that is automatically assumed if no other value is specified.

Digitizing — The process of pressing the stylus on the tablet's surface and the tablet sending the X,Y coordinates of that location to the computer.

Digitizing limits — The physical active digitizing area on the tablet's surface. Nothing can be digitized outside these limits.

Digitizing unit (DU) — A digitizing unit is 0.025 mm (approximately 0.001 in.) in length and is the unit of measure used to express physical locations in the tablet's Cartesian coordinate system.

Execute — To cause program statements to perform their particular functions. To run a program.

Graphics tablet — A data entry device that converts a physical location on its surface into digital values usable by the computer.

HP-GL — Hewlett-Packard Graphics Language. The instruction set sent to and understood by the graphics tablet.

HP-IB — Hewlett-Packard Interface Bus. Hewlett-Packard's implementation of the IEEE 488-1978 Instrumentation Bus used to connect multiple devices together with a well-defined hardware protocol.

Initialize — To set certain conditions to a known state (usually the default state).

Input — A process of transferring information into a device.

Interactive graphics — A two-way communication between the user and the computer to produce graphics. The graphics tablet facilitates this process.

Interface status check — The process of checking a bit in an interface register to see if a service request has been sent.

Interrupt — A signal that suspends ordinary operation of a computer so that some immediate need, such as a service request, can be met.

Line feed (LF) — A character that causes the printing or display position to be moved to the next printing or display line. It is used as a terminator for the tablet's HP-GL instructions.

Listener — A device that receives information sent on the bus when it is addressed.

Menu — A list or table of available choices.

Mnemonic — A code that assists your memory. For example, OS suggests output status. Each HP-GL instruction begins with a two-letter mnemonic.

Output — A process of transferring information out of a device.

Parameter — A value that is used with certain HP-GL instructions to define specific conditions.

Pen status — A code from the tablet that tells whether the stylus is pressed, whether a digitized point is the last in a series while in the continuous digitizing mode, or whether a digitizing mode is set.

Peripheral — A device used to enhance the performance or a computer; for example, a disc drive, printer, plotter, or graphics tablet.

Platen — The white ceramic surface of the graphics tablet.

Program — An organized set of statements that tells the computer and tablet to perform certain tasks.

Read — A process of transferring information into a device.

Register — A place reserved in a device's memory for storing information.

ROM — Read-Only Memory. A memory device in which the memory locations are set to fixed patterns when the device is manufactured.

Routine A portion of a computer program that performs a specific task.

Run A single, complete execution of a computer program.

Separator Punctuation separating numerical parameters in an HP-GL instruction, usually a comma or semicolon.

Service request (SRQ) The message a device sends to signify that it needs interaction with the computer.

Single-point digitizing A digitizing mode in which one point is digitized at a time.

Softkeys The sixteen boxes at the top of the tablet's platen which the user can personalize for specific job functions and which, when digitized, send an integer corresponding to the box selected.

Status mask (S-mask) Specifies when HP-IB service requests are to be sent.

Status word Eleven bits of information that communicate significant conditions within the tablet.

String A linear sequence of alphanumeric characters.

Stylus The digitizing or "pointing" device on the tablet that looks much like a ball-point pen.

Switch-follow mode The continous-digitizing mode in which the tablet takes points at regular time intervals when the stylus remains pressed.

Switch-normal mode The continuous-digitizing mode in which the tablet begins taking points at regular time intervals when the stylus is pressed, and then stops when the stylus is pressed again.

Syntax The rules governing the structure of a language.

Talker A device that sends information on the bus when it is addressed.

Terminator A special character or signal that is required to end an HP-GL instruction. A semicolon (;) and a line feed (LF) are valid terminators.

Variable The name of something that can take one value or a succession of values in a program.

Subject Index

Subject Index (Continued)

i

m

o

p

r

s