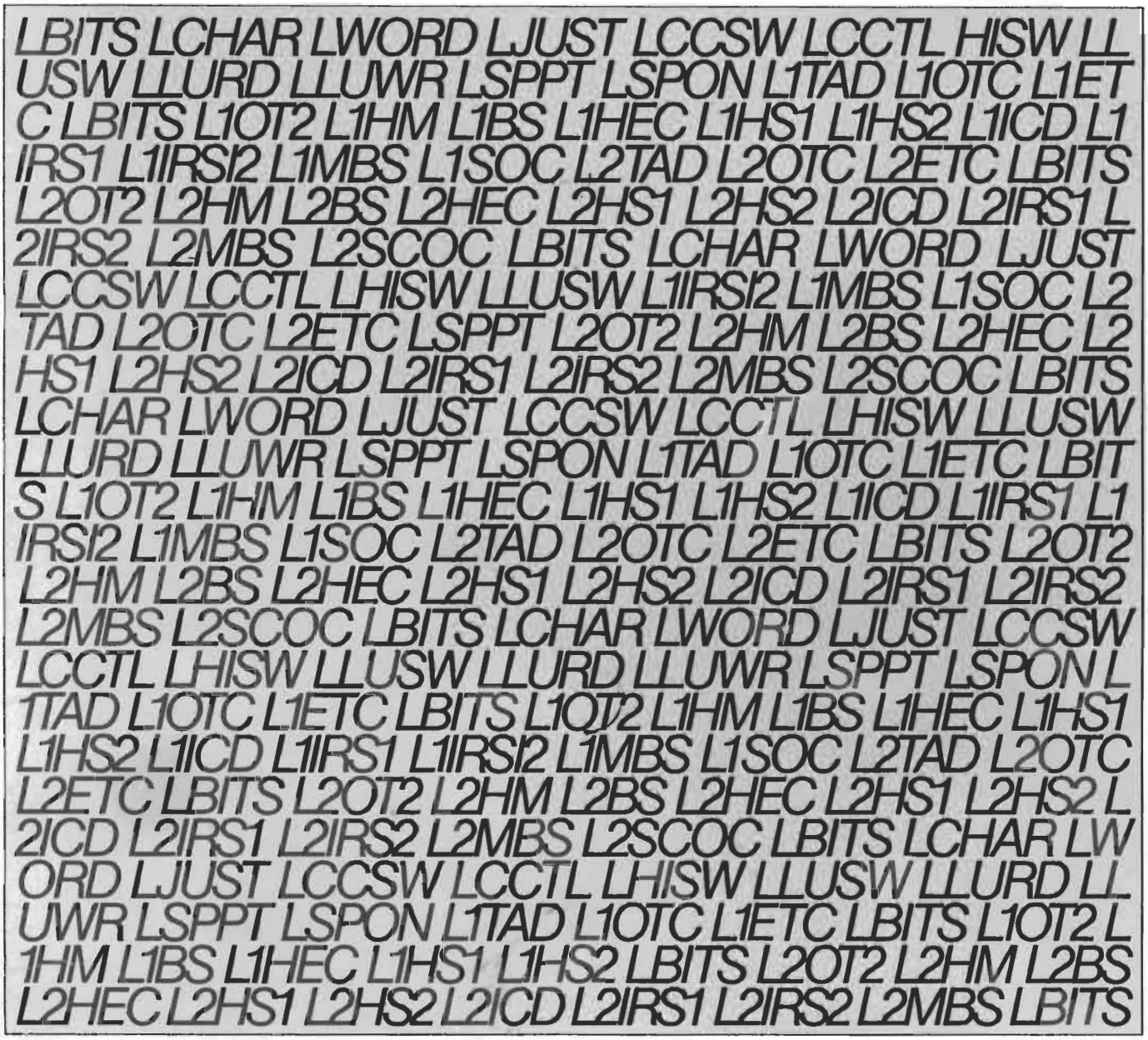
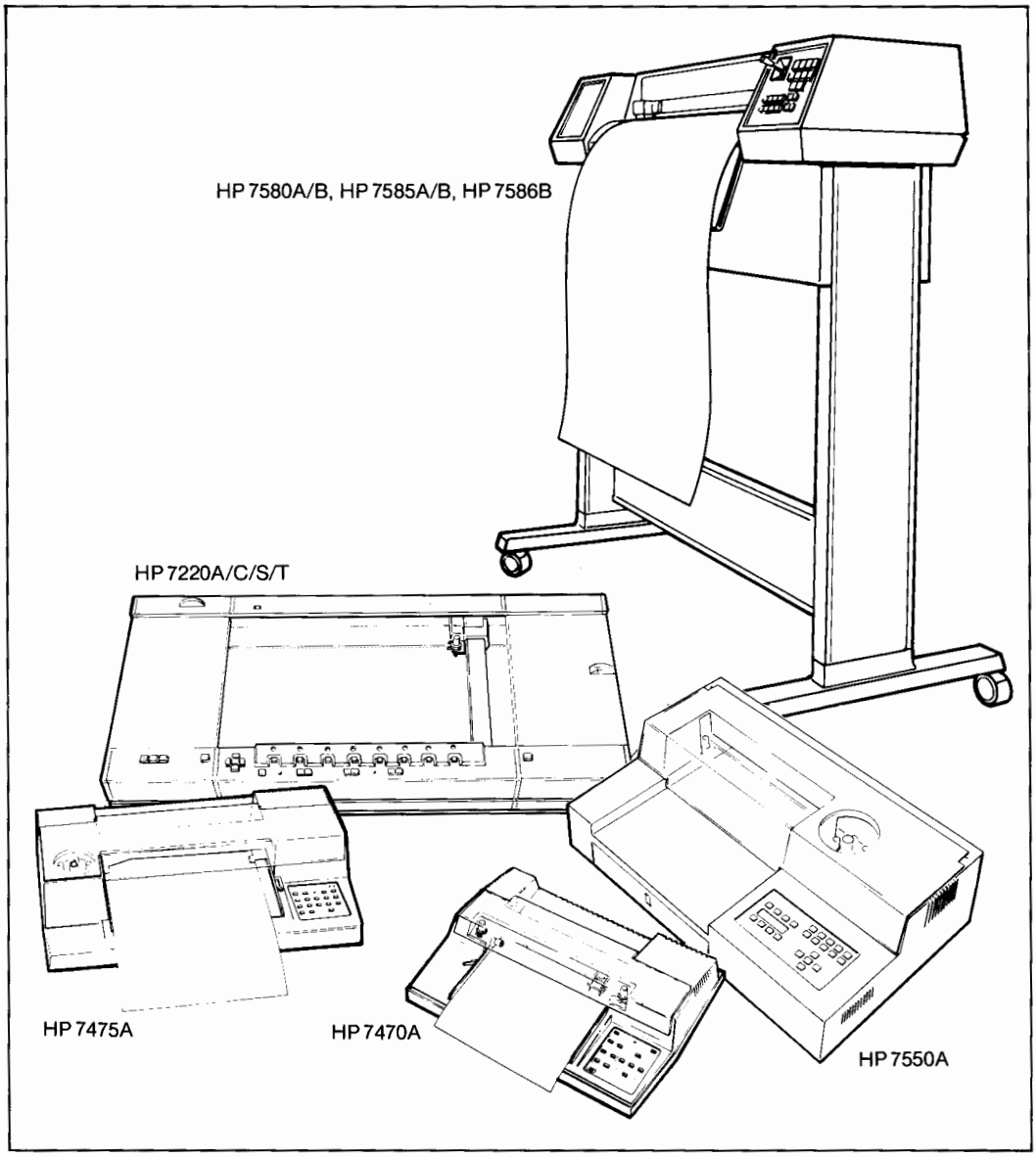
Hewlett-Packard
Industry Standard Plotting Package
User's Manual
for
HP 17580B



Hewlett-Packard
Industry Standard Plotting Package
User's Manual
for
HP 17580B.00.00



16399 W. Bernardo Drive, San Diego, California 92127-1899

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Table of Contents

Chapter 1
Introduction

Becoming Familiar with the HP Industry Standard Plotting Package

The Industry Standard Plotting Package (HP-ISPP, HP 17580B) is a set of FORTRAN subroutines. The subroutines provide high-level control of common graphic functions. Such functions as axis generation, data scaling, and pen selection are part of HP-ISPP (HP Industry Standard Plotting Package).

HP-ISPP has 25 FORTRAN subroutines. The following plotters are supported by HP-ISPP:

RS-232-C/CCITT V.24 Interface

Model	Paper Size		Number of Pens
	English	Metric	
HP 7220A/S	11 × 17 in.	297 × 420 mm	four
HP 7220C/T			eight
HP 7470A*	8½ × 11 in.	210 × 297 mm	two
HP 7475A*	8½ × 11 in. 11 × 17 in.	210 × 297 mm 297 × 420 mm	six
HP 7580A* HP 7585A*	8½ × 11 in. 22 × 34 in.	210 × 297 mm 594 × 841 mm	eight

*Option 001

1

HP-IB (IEEE-488-1978) Interface

Model	Paper Size		Number of Pens
	English	Metric	
HP 7470A*	8½ × 11 in.	210 × 297 mm	two
HP 7475A*	8½ × 11 in. 11 × 17 in.	210 × 297 mm 297 × 420 mm	six
HP 7580A* HP 7585A*	8½ × 11 in. 22 × 34 in.	210 × 297 mm 594 × 841 mm	eight
HP 9872B/S	11 × 17 in.	297 × 420 mm	four
HP 9872C/T			eight

*Option 002

Dual Interfaces

Model	Paper Size		Number of Pens
	English	Metric	
HP 7550A	8½ × 11 in. 11 × 17 in.	210 × 297 mm 297 × 420 mm	eight
HP 7580B	8½ × 11 in. 11 × 17 in. 17 × 22 in. 22 × 34 in.	210 × 297 mm 297 × 420 mm 420 × 594 mm 594 × 841 mm	eight
HP 7585B HP 7586B	8½ × 11 in. 11 × 17 in. 17 × 22 in. 22 × 34 in. 34 × 44 in.	210 × 297 mm 297 × 420 mm 420 × 594 mm 594 × 841 mm 841 × 1189 mm	eight

This manual describes the functional capabilities of HP-ISPP. It will acquaint the user with the syntax needed to call the subroutines in the package. This is a reference document for experienced graphics programmers. The descriptions and examples used here were prepared in an ANSI FORTRAN subset (X3.9-1966). Users should be familiar with FORTRAN syntax requirements and conventions and know how to use them on their computer systems.

Instructions for implementing the protocol or system control language for a host computer are not given here. Instructions are not given for entering, compiling, or running a FORTRAN program. Operating instructions for plotters which use HP-ISPP are not given. Those instructions can be found in the following manuals:

HP 7220A/C/S/T Graphics Plotters Operating and Programming Manuals Using HP-GL Instructions, Part Number 07220-90002 (HP 7220A/S) or 07220-90003 (HP 7220C/T).

HP 7470A Plotter Interfacing and Programming Manual, Part Number 07470-90001, and HP 7470A Operator's Manual, Part Number 07470-90002.

HP 7475A Graphics Plotter Interfacing and Programming Manual, Part Number 07475-90001, and HP 7475A Graphics Plotter Operation and Interconnection Manual, Part Number 07475-90002.

HP 7550A Graphics Plotter Interfacing and Programming Manual, Part Number 07550-90001, and HP 7550A Graphics Plotter and Interconnection Guide, Part Number 07550-90002.

HP 7580A/B, HP 7585B, and HP 7586B Drafting Plotter Interfacing and Programming Manual, Part Numbers 07580-90024 and 07580-90034.

HP 9872B/C/S/T Operating and Programming Manuals, Using HP-GL Instructions, Part Number 09872-90008 (HP 9872B/S) or 09872-90011 (HP 9872C/T).

Philosophy of HP-ISPP

The package is a set of callable subroutines arranged into modules. This is what they do:

1. Scale user data to fit a graph of a given size.
2. Draw an annotated axis for the graph.
3. Connect a set of data points using straight lines.
4. Plot a floating point number using a FORTRAN "F" type format.
5. Scale the size of a plot.
6. Format and transmit arbitrary and user-specified HP-GL commands. The associated parameters used to access additional plotter features while under HP-ISPP software control are also formatted and transmitted.

The package translates the user-oriented mnemonics and numeric values into the language required by the plotters. This language is the Hewlett-Packard Graphics Language (HP-GL) and is discussed in detail in the operating/programming manuals for each supported plotter. The software can handle the necessary handshaking between the host computer and the plotter.

Internal subroutines are called by other subroutines. The called subroutines manage global information and perform specific tasks. They should not be called directly by the user.

The HP-ISPP software has 25 subroutines. You can call 15 of them. Table 1-1 lists the name and function of each subroutine. The functions of 14 subroutines are indicated by their names. Many existing applications programs use these

subroutines. BUFF, the fifteenth subroutine, converts real and integer variables into the proper form for plotter use. BUFF also coordinates host buffer management and plotter output requests.

Data input and output is initiated by BIN and BOUT, two internal subroutines. Actual input and output is performed by eight machine-dependent subroutines. The subroutines in this group provide information about host computer attributes, pack and unpack character strings, and provide the plotter with P1 for deferred plotting. The names of these subroutines, which should not be called directly, are: ZZHOST, ZZINIT, ZZGET, ZZPUT, ZZLOW, ZZPACK, ZZEXTR, and XXORG.

Table 1-1. HP-ISPP Subroutines

Subroutine	Description of Function
User-callable subroutines:	
AXIS	Produce an axis with annotated divisions and a centered label.
DPINIT	Initialize the software for deferred plotting.
FACTOR	Make the graph larger or smaller.
FPINIT	Initialize the plotter and the software to plot from a file.
HPINIT	Initialize the software and the plotter.
LINE	Draw a line plot of the pairs of data values in two arrays.
NEWPEN	Select another pen or return the current pen to the stable.
NUMBER	Draw a floating point number in FORTRAN "F" type format.
OFFSET	Set special offset and scale factors for a graph.
PLOT	Move with the pen up or down to a new position.
PLOTS	Set default values for global variables in the software and turn the plotter logically on.
SCALE	Provide scale factors for an array of floating point numbers which are used by AXIS to draw a single axis of the specified length, and by LINE to plot the array values.
SYMBOL	Draw ASCII characters or special symbols at a specified location.
WHERE	Obtain the current actual physical pen location and scale factor.
BUFF	Perform host buffer management, converting integer and real values into proper form.

Table 1-1. HP-ISPP Subroutines (Continued)

Subroutine	Description of Function
Internal subroutines:	
BIN	Perform the equivalent of a free-field read from the plotter.
BOUT	Output the host data buffer to the plotter.
XXORG	Provide the plotter with P1 for deferred plotting.
ZZHOST ZZINIT	Provide information regarding the attributes of the host computer.
ZZGET	Read a record from the plotter.
ZZPUT	Write a record to the plotter.
ZZLOW	Return the ASCII code of the character stored in the low-order position of an integer.
ZZPACK	Pack the characters for use by subroutine SYMBOL.
ZZEXTR	Unpack characters for output.

NOTE

HP-ISPP internal subroutines perform certain data conversion functions and maintain global information. They should not be called directly from the main program. Their names, and the name of the internal named common block, "HP" and "ZZCOM," should be considered as reserved names and avoided by the programmer when naming other subroutines and global variations.

Graphic Unit Systems

HP-ISPP supports both the inch and the metric unit systems. The default parameter uses inches. Metric units can be used by setting the LUNIT parameter to 1. You do this by calling subroutine HPINIT or DPINIT. All references to subroutine unit systems in this manual use inches. However, if the unit system invoked is metric, centimetres may be assumed.

Device Considerations

Three modes of operation are possible: "eavesdrop," "direct" (or "noneavesdrop"), and "monitor."

In the eavesdrop mode, the plotter is interfaced between the terminal and the computer with an RS-232-C/CCITT V.24 interface. Information from the terminal flows

through the plotter to reach the computer. Information from the computer flows through the plotter to reach the terminal. The plotter observes all the data transmitted from the computer to the terminal. The plotter looks for a "plotter on" command. The plotter interprets all characters received after a "plotter on" command as plotter instructions. It does this until a "plotter off" command is received. No characters pass to the terminal between the "plotter on" and "plotter off" commands.

Plotters that operate in eavesdrop mode can also support monitor mode, useful in debugging interfacing problems. All data transmitted or received by the plotter appears on the terminal CRT in receive monitor mode.

The plotter RS-232-C/CCITT V.24 interface is connected directly to the host computer in direct (noneavesdrop) mode. The plotter is interfaced as a separate logical unit.

Plotters that use the HP-IB interface connect directly to the computer.

Differences among devices supported by ISPP software are:

HP 7220A/C/S/T

Number of pens supported = 4 if an A or S model.
8 if a C or T model.

The centered symbols are generated by software.

The plotter can operate in both eavesdrop and noneavesdrop configurations.

The plotter supports monitor mode in the eavesdrop configuration.

The default X- and Y-minimum and maximum values for the HP 7220 series plotters vary from model to model. For models equipped with paper advance mechanisms, the state of the rear-panel ENGLISH/METRIC switch determines these values.

HP 7470A

The number of pens supported = 2.

The centered symbols are generated by the software.

The RS-232-C/CCITT V.24 interface option may operate in either the eavesdrop or noneavesdrop mode.

The HP 7470A supports monitor mode.

The default X- and Y-minimum values (P1) are 250, 259 plotter units respectively. The default maximum X and Y values (P2) are 10 250, 7479 plotter units.

HP 7475A

The number of pens supported = 6.

The centered symbols are generated by the software.

The RS-232-C/CCITT V.24 interface option may operate in either the eavesdrop or noneavesdrop mode.

The HP 7475A supports monitor mode.

The default X- and Y-minimum and maximum values are dependent on the paper size (A, B, A4, or A3).

HP 7550A

The number of pens supported = 8.

The centered symbols are generated by the firmware.

The RS-232-C/CCITT V.24 interface option may operate in either the eavesdrop or noneavesdrop mode. The HP-IB option has a monitor mode.

The HP 7550A supports monitor mode.

The default X- and Y-minimum and maximum values are dependent on the paper size (A, B, A4, or A3).

HP 7580A/B and 7585A/B

The number of pens supported = 8.

The centered symbols are generated by firmware.

The RS-232-C/CCITT V.24 interface option on the A models operates in a non-eavesdrop configuration only and does not support monitor mode.

The default X- and Y-minimum and maximum values are dependent on the paper size and the setting of the plotter rear-panel EXPAND/NORMAL switch. In the recommended NORMAL position, the plotter pen is allowed to come within about 15 mm of three edges of the paper and to within 39 mm of the fourth edge, the edge that is holding the paper when it is extended out in what is called the VIEW position. Because it might be desirable to plot closer to the edges, the pen can come to within 5 mm of the three edges and to within 29 mm of the fourth edge if the switch is in the EXPAND position.

1

HP 7586B

The number of pens supported = 8.

The centered symbols are generated by the firmware.

The RS-232-C/CCITT V.24 interface option may operate in either the eavesdrop or noneavesdrop mode.

The HP 7586B supports monitor mode.

The default X- and Y-minimum and maximum values are dependent on the paper size and the setting of the plotter rear-panel EXPAND/NORMAL switch. In the recommended NORMAL position, the plotter pen is allowed to come within about 15 mm of three edges of the paper and to within 39 mm of the fourth edge, the edge that is holding the paper when it is extended out in what is called the VIEW position. Because it might be desirable to plot closer to the edges, the pen can come to within 5 mm of the three edges and to within 29 mm of the fourth edge if the switch is in the EXPAND position. However, if the plotter is loaded with roll media instead of sheet media, then the limits are 15 mm for all edges or 5 mm for all edges in EXPAND.

HP 9872B/C/S/T

The number of pens supported is 4 if a B or S model and 8 if a C or T model.

The centered symbols are generated by software.

The plotter operates only with the HP-IB interface and data handshake method.

For B/C models, the default X- and Y-minimum values are 520, 380 plotter units respectively. The default X- and Y-maximum values are 15720, 10380 plotter units respectively. The default X- and Y-minimum and maximum values for S/T models are determined by the setting of the rear-panel ENGLISH/METRIC switch.

S/T models default X- and Y-minimum values with the switch set in the ENGLISH position are, respectively, 520, 1020 plotter units. The English setting default X-and Y-maximum values are 15760, 10380, respectively. With the ENGLISH/METRIC switch set to METRIC, default X- and Y-minimum values are 520, 1140 plotter units. The Metric setting default X- and Y-maximum values are, respectively, 15760, 11140 plotter units.

Chapter 2
The HP Industry Standard Plotting Package Subroutines

This chapter describes user-callable subroutines in the HP-ISPP package. The subroutines are alphabetically arranged.

Each description lists a subroutine. The purpose is stated and the format of the calling sequence is given. Calling parameters are explained and a statement of function is given. Any pertinent notes on the subroutine's use are also stated. Illustrations accompany some command descriptions. The illustrations are reproductions of plots by the HP 7580 plotter using HP-ISPP software. Illustrations are provided to demonstrate the concepts and effect of certain subroutines. In some cases other subroutines may also be required to reproduce the labeling and drawing shown. Many of the illustrations use the HP 7580 arc-generated character set.

Conventions

HP-ISPP subroutines are accessed from your user program through FORTRAN CALL statements. The general format for an HP-ISPP call to a subroutine is:

CALL subnam (parm1,parm2, . . . ,parmn)

"Subnam" is a subroutine name. "Parm1,parm2, . . . ,parmn"are the parameters, as required, that are passed to the called subroutine. Table 1-1 lists subroutine names. In this manual, subroutine names and parameters indicate their function or application. Each name uses a maximum of six characters. A maximum of six characters is used for all subprogram and variable names in HP-ISPP software. The first five characters of each subprogram name are unique.

Subroutine parameter names also follow these additional conventions:

1. Any parameter which refers to a specific axis (X or Y) begins with either X or Y, or if an integer variable, with IX or IY.

2. Integer parameters begin with the letters I through N. Real (floating point) parameters begin with the letters A through H or O through Z.

3. Arrays end with the letters "ARAY." If the array is not specific to one axis, or could be used for either axis, only the letters ARAY are used.

4. All arrays containing labels or character strings begin with the letter L. If the array refers to a specific axis, then the name begins with LX or LY.

Any set of ASCII characters intended as a label is referred to as a "string." A character string may be specified in one of three ways:

Example 1: By using the nH specification in a FORTRAN DATA statement:

```
DIMENSION LARAY(3)
DATA LARAY(1), LARAY(2), LARAY(3) /2HAB, 2HCD, 2HEF/
CALL SYMBOL (X, Y, HEIGHT, LARAY, ANGLE, ICHAR)
```

Example 2: By using the EQUIVALENCE statement as shown in the following:

```
DIMENSION LARAY(3)
CHARACTER*6 STRING
EQUIVALENCE (LARAY(1), STRING)
STRING = 'ABCDEF'
CALL SYMBOL (X, Y, HEIGHT, LARAY, ANGLE, ICHAR)
```

Example 3: By using a Hollerith literal in the call statement itself:

```
CALL SYMBOL (X, Y, HEIGHT, 6HABCDEF, ANGLE, ICHAR)
```

Examples 1 and 3 conform to ANSI X3.9-1966 (FORTRAN) and to ANSI X3.9-1978 (FORTRAN-77) with the recommended extension to support Hollerith data types. Example 2 conforms to FORTRAN-77 standards and some other compilers which support type CHARACTER data.

The examples in this manual have been prepared for use on a computer which uses the ASCII character code for internal storage and stores two characters per integer word. Some programs in this manual will have to be changed in order to execute properly on a processor with a different word size. Refer to the HP-ISPP Installation Guide, P/N 17580-90005, for details.

The Call to AXIS

Purpose: Draw one labeled axis for a previously scaled data array.

Format: CALL AXIS(X, Y, LARAY, NCHAR, AXLEN, ANGLE, FIRSTV, DELTAV)

Parameters:

X, Y The coordinates, in floating point inches (centimetres if metric scaling has been specified) of the starting point of the axis, relative to the current origin. The origin of the coordinate system is the current lower-left hard-clip limit position, unless the origin has been redefined by a call to PLOT with parameter IPEN < 0. The entire axis should be at least one-half inch from any side to allow space for the scale annotation.

Normally, both the X- and Y-axis are joined at the origin of the graph, where parameters X and Y are zero; however, other starting points may be used. When using the LINE subroutine to plot data on an axis, at least one of the coordinates must be zero.

LARAY An integer array containing the characters for the axis label. The characters are stored as Hollerith data or, if the compiler permits, as a Hollerith literal. The characters have a fixed height of 0.14 inches or 0.35 centimetres if metric units are used. Scaling can be changed by a call to FACTOR.

The characters have a fixed height of 0.14 inches. If metric scaling is specified, the characters are drawn at a fixed height of 0.35 centimetres.

NCHAR An integer whose absolute value specifies the number of characters in the array LARAY. The sign of the integer designates whether the label is drawn along the clockwise side or the counterclockwise side of the axis:

If the sign is positive, all annotation appears on the positive (counterclockwise) side of the axis.

If the sign is negative, all annotation appears on the negative (clockwise) side of the axis.

AXLEN The length of the axis, in floating point inches, or in centimetres if metric scaling is specified.

ANGLE The angle in degrees positive or negative, relative to the horizontal 3 o'clock position, at which the axis is to be drawn. The value is normally 0.0 degrees for the X-axis and 90.0 degrees for the Y-axis.

FIRSTV The starting value (either minimum or maximum) which will be labeled at the first tick mark on the axis. This value may either be computed by the SCALE routine and stored at subscripted location ARAY(NSCALE * |NPOINT| + 1), or the value may be determined by the user and stored at any location.

Refer to SCALE for an explanation of the scaling process and the parameters ARAY, NSCALE, and NPOINT.

DELTAV The number of data points per inch of axis. This value (increment or decrement) is added to FIRSTV to determine the value of each succeeding one-inch division along the axis and may either be computed by the user or by subroutine SCALE. If by SCALE, the value is stored beyond FIRSTV at ARAY(NSCALE * |NPOINT| + |NPOINT| + 1).

The value of DELTAV is adjusted to be within the range of 0.01 to 100.0 so that a standard format of two decimal places may be used in labeling. As a result, the decimal point of axis tick mark labels may be shifted left or right when drawn. The axis title is then followed by "*10^n" where n is the power-of-ten adjustment factor.

Function: An axis with tick marks is drawn from the specified (X,Y) starting coordinate, at the specified angle, for the specified length. The tick marks are drawn at one-inch intervals (one-centimetre intervals if metric scaling is specified) along the axis, with a floating point numeric label at each tick mark (every other tick mark if metric scaling is specified). An axis label is drawn along one side of the axis.

Comments:

1. A previous call to SCALE is required if the FIRSTV and DELTAV parameters are specified as the values stored in ARAY(NSCALE * |NPOINT| + 1) and ARAY(NSCALE * |NPOINT| + |NPOINT| + 1).

2. The AXIS subroutine calls NUMBER and SYMBOL to draw the tick mark labels with a 0.105-inch character height and the axis label with a 0.14-inch character height. When metric scaling is specified, the axis label is drawn with a 0.35-centimetre height and the tick mark labels with a 0.19-centimetre height.

3. The starting point of the axis is defined as the left end of an X-axis or the lower end of a Y-axis. The ANGLE parameter is generally specified as 0.0 for an X-axis and 90.0 for a Y-axis.

The Call to BUFF

Purpose: Coordinate all I/O and convert integer and real values to the proper HP-GL syntax for use by the plotter.

Format: CALL BUFF(MODE, IBUFF, XBUFF, INUM)

Parameters:

MODE An integer whose value indicates the type of action to be taken by the subroutine.

1 Insert characters stored as ASCII decimal equivalents in the array IBUFF into the host output buffer. The host output buffer is a COMMON array within HP-ISPP that is used to hold plotter data. If INUM is negative, a semicolon terminator will follow the output string.

2 Insert characters stored as decimal equivalents in the array IBUFF into the host output buffer, then transmit the output buffer to the plotter regardless of the amount of remaining space in the array. If INUM is negative, a semicolon terminator will follow the output string.

3 Read data from the plotter. An array of one or more integer variables is returned in IBUFF, except when issuing an OI call (for plotter identification). In this case, the IBUFF array will contain an integer variable in IBUFF(1), plus the decimal equivalent for the model designation in IBUFF(2). For example, an OI call to an HP 7220C plotter would return IBUFF(1) = 7220 and IBUFF(2) = 67 (where 67 is the ASCII code for a 'C').

4 Convert the integer value(s) contained in the array IBUFF to ASCII characters and insert them into the output buffer. Transmit the host output buffer if it is then full. The permissible range of integer variables is −32 768 to +32 767. The form of the output is:

INT SEP INT SEP INT TERM

where: INT = signed/unsigned integers

SEP = a comma or semicolon

TERM = a semicolon or colon

If INUM is negative, the separator is a semicolon and the terminator is a colon. If INUM is positive, the separator is a comma and the terminator is a semicolon.

5 Same as MODE 4, but transmit the host output buffer regardless of the remaining space. An inquiry is not made regarding the available space in the plotter data buffer.

6 Convert the floating point number or numbers in the array XBUFF to ASCII and insert them into the output buffer. The permissible range of floating point variables is $\pm 2^{31} - 1$. The form of the output is:

DEC SEP DEC TERM

where: DEC = signed/unsigned real variables

SEP = comma

TERM = semicolon

7 Same as MODE 6 but transmit the host buffer to the plotter regardless of the remaining space. An inquiry is not made regarding available space in the plotter data buffer.

IBUFF Integer array containing data to be placed in the buffer or, if MODE = 3, the data returned by the plotter.

XBUFF Real array containing data to be placed in the buffer.

INUM The number of data elements in array IBUFF or XBUFF to be processed by the subroutine, or which have been returned from the plotter and stored in IBUFF.

Function: This subroutine coordinates all of the I/O activities. The HP-GL commands and associated parameters are placed in the host output buffer along with any required separators and/or terminators. Integer and real values are converted to the proper format. BUFF calls internal subroutines BIN and BOUT to perform the actual input and output functions.

Comments:

1. Subroutine BUFF acts on data in either IBUFF or XBUFF as specified by the value of MODE. Only one type of action can be taken at a time. Thus a call to transmit an HP-GL command with one or more parameters will require at least two calls to subroutine BUFF.

2. Chapter 3 contains an illustration using BUFF to send HP-GL commands to a plotter.

The Call to DPINIT

Purpose: To initialize the plotter software and enter into deferred plotting.

Format: CALL DPINIT (LIO, LMODEL, LVER, LPSIZE, LCHARS, LUNIT, LFILE)

Parameters:

LIO Handshake method:

LIO = 0 For the Primary Handshaking method (if this is a software-checking handshake, determined at installation, it cannot be used).
LIO = 1 For the Alternate Handshaking method (if this is a software-checking handshake, determined at installation, it cannot be used).
LIO = 2 For the HP-IB (IEEE-488-1978) Interface and Handshake.
LIO = 3 For no handshake.

LMODEL Model number of plotter that will be used (i.e., LMODEL = 7580).

LVER Model version of plotter.

LVER = ASCII value of version letter — A, B, C, S, or T (i.e., 65, 66, 67, 83, or 84).

LPSIZE Paper size to be used.

LCHARS Character set selection (see HPINIT).

LPSIZE Values for Various Paper

Paper Size	LPSIZE Value	Expand Mode LPSIZE Value*
Metric		
A0	0	100
A1	1	101
A2	2	102
A3	3	103
A4	4	104
English		
E	10	110
D	11	111
C	12	112
B	13	113
A	14	114
Metric Roll**	1003	
English Roll**	1013	

*HP 758X models
**HP 7220 and 9872 models

LUNIT Unit system desired.

LUNIT = 0 For inches unit system.
LUNIT = 1 For metric unit system. (Centimetres are used in the metric system.)

LFILE The FORTRAN unit number assigned to the output file.

Function: The software is initialized and placed in the deferred plotting mode. In deferred plotting, all output is directed to a file instead of to a plotter. The Primary Handshake method is not allowed because it requires responses from a plotter. Selecting LIO = 3 (no handshake) keeps the hardware from putting handshake commands into the output file.

Comments: If LIO = 0 or LIO = 1, then the handshaking commands are placed in the file sent to the plotter via the RS-232-C/CCITT V.24 interface. In this case, handshaking *must* be controlled by the operating system and *not* by the software. If LIO = 2 or LIO = 3, then *no* handshake commands are placed in the file sent to the plotter. The Primary and Alternate Handshake methods are determined when the software is installed. At this time, either system control or software checking control is established.

The user is responsible for assigning a FORTRAN unit number to the file and, if necessary, to open the file before calling DPINIT. The user then continues with the program, creating a plot in the file. The user must close the file afterwards if the system requires this.

The Call to FACTOR

Purpose: Alter the scaling factor of the plot.

Format: CALL FACTOR(FCT)

Parameters:

FCT The plot scaling factor desired. Initialization value established by HPINIT is 1.0. A value for FCT ≤ 0 sets FCT to a default value of 1.0.

Function: The entire plot (drawing of axes, labels, and plotting of data) is scaled according to the value of FCT. A scaling factor of 1.0 yields a plot of "normal" size.

CALL FACTOR (0.5) yields a plot of half the normal size in each dimension.

CALL FACTOR (2.0) yields a plot twice the normal size in each dimension.

The scaling factor affects pen positioning, drawing, axes, and labeling in all unit systems. Clipping at the hard-clip boundaries defined by the lower-left and upper-right limit positions is performed if the scaling factor is such that the pen would be moved outside these boundaries.

The Call to FPINIT

Purpose: To initialize the HP-ISPP software and plotter and to enter the file plot mode. The file plot mode involves reading from the file and outputting to the plotter.

Format: CALL FPINIT (LIO, LMON, LLONG, LDEV, LFILE)

Parameters:

LIO Handshake Method (see HPINIT).

LMON Monitor Mode Control (see HPINIT).

LLONG Long-axis plot:

LLONG = 0 For regular plot.
LLONG = 1 For HP 7586B roll feed long-axis plot.

LDEV The FORTAN unit number assigned to the plotter.

LFILE The FORTRAN unit number assigned to the plot file.

Function: The plotter is turned logically "on" and the input plot file created earlier is sent to the plotter.

Comments:

1. The plot file must be created by DPINIT with LIO = 3, no handshaking.

2. The input file must be opened and assigned a FORTRAN unit number before calling FPINIT.

3. HP 7586B long-axis plots require deferred plotting. To set up a deferred plot file, initialize the software with DPINIT and set the LIO parameter to 3 so that no handshake is used. Then write a program using FPINIT to read from the file. FPINIT sets the number of frames needed and outputs them until the file is plotted.

   Data outside the media's limits is clipped in the positive Y-direction (the pen movement direction). Data outside the limits in the negative Y-direction isn't clipped; the software moves the entire plot in the positive Y-direction. FPINIT can be changed to make negative Y-direction clipping work this way also. Change these lines:

From:	To:
XBUFF(3)=YMIN	XBUFF(3)=P1Y
XBUFF(4)=YMAX	XBUFF($)=P2Y

The Call to HPINIT

Purpose: Initialize the plotter software and the plotter.

Format: CALL HPINIT(LIO, LMON, LCHARS, LUNIT, LDEV)

Parameters:

LIO Handshake Method:

At the time your program is written, you must select one of three handshake methods using the LIO parameter.

LIO = 0 For the Primary Handshake method.
LIO = 1 For the Alternate Handshake method.
LIO = 2 For the HP-IB (IEEE-488-1978) Interface and Handshake.

The Primary Handshake method is the mode most frequently used for plotting applications at an installation. The Alternate Handshake method is for infrequent or unique applications. This method may be specified in applications where the configuration of hardware is different from the primary mode.

The HP-IB controlled handshake must be used with all HP-IB interface plotters. It CANNOT be used with an RS-232-C/CCITT V.24 interfaced device.

Contact your system installer if you have questions about the appropriate handshake mode for your application.

LMON Monitor Mode control:

LMON = 0 No monitor mode desired.
LMON = 1 Invoke parse monitor mode.
LMON = 2 Invoke receive monitor mode.

Monitor mode is used with eavesdrop configuration to aid in program debugging. The receive mode allows all data to be displayed on the terminal that is sent to the computer system or received by the plotter from the computer. Parse mode stores computer commands in the plotter's buffers and displays them on the terminal as they are executed.

LCHARS The standard character set desired for all labeling by the plotter. The value of LCHARS is the designation of the plotter's firmware character set which is to be used as the standard character set for the plot. If a value is specified which requests a character set not available in the plotter, then the error light will come on.

LUNIT Unit system desired:

LUNIT = 0 Inches unit system desired.
LUNIT = 1 Metric unit system desired. (In the metric system, centimetres are used.)

LDEV The FORTRAN unit number assigned to the plotter.

Function: The plotter is turned logically "on," initialized to a known state, and default values are assigned to the software global variables. The HP-ISPP global variables are listed and defined in Appendix A.

Comments: The FORTRAN unit number is the unit to which all software input and output requests are directed. The number must be equated to the physical device connection through a specific operating system command or user program call. For example, the HP 3000 environment requires:

```
:FILE FTNnn;DEV=lu;CCTL
```

where: nn = the value of LDEV above, and specified as a two-digit FORTRAN unit number in a control record within the user's program.

lu = the device or equipment number that the plotter is physically attached to, whether in eavesdrop or noneavesdrop configuration.

CCTL = Indicates that the carriage-control characters are being supplied in WRITE statements to the unit.

This sequence provides the necessary relationship between the logical unit and the FORTRAN file requirements.

The Call to LINE

Purpose: Plot a line and/or symbols through successive data points in arrays previously scaled by the SCALE routine.

Format: CALL LINE(XARAY, YARAY, NSCALE, NPOINT, LINTYP, INTEQ)

Parameters:

XARAY The name of the array containing the abscissa (X) values and the scaling parameters for the X-array.

YARAY The name of the array containing the ordinate (Y) values and the scaling parameters for the Y-array.

NSCALE The number of data points in each of the two arrays XARAY and YARAY to be plotted. This number does not include the two locations storing the scaling parameters calculated by SCALE. The number of points in the two arrays must be the same.

NPOINT The increment that the LINE subroutine is to use in selecting data values from the two arrays, as described in the SCALE subroutine.

LINTYP The control parameter which indicates how lines and/or centered symbols are to be drawn through the data points. The magnitude of LINTYP determines the frequency of plotted symbols. For example, if LINTYP = 4, a special symbol (denoted by INTEQ) is plotted at every fourth data point. The symbols are drawn at a fixed height of 0.08 inches (0.20 centimetres).

If LINTYP is zero, the points are connected by straight lines, but no symbols are plotted.

If LINTYP is positive, a straight line connects every data point plotted from the array. The pen is up when moving from its current position to the first point. The centered symbol specified by INTEQ will be drawn at the interval specified by the value of LINTYP.

If LINTYP is negative, no connecting lines are drawn, and the symbol specified by INTEQ is drawn at each data point.

INTEQ The integer equivalent of the special plotting symbol centered at each data point. This value is 0 through 14 and has meaning only when LINTYP is not zero. Figure 2-1 is an illustration of the available centered symbols.

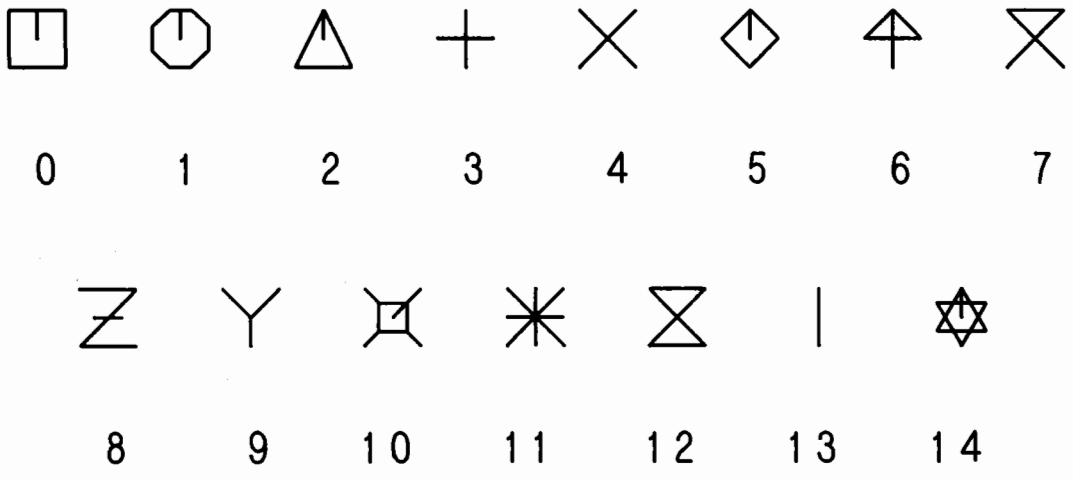


Figure 2-1. Centered Symbols

Function: The subroutine produces a line plot of the pairs of data values in the two arrays (XARAY, YARAY). The coordinates of each point are computed according to the data values in each array and the respective scaling parameters. The data points may be represented by centered symbols and/or connecting lines between points.

The subroutine uses scaling parameters stored in each array at locations (NSCALE * NPOINT + 1) and (NSCALE * NPOINT + NPOINT + 1) by subroutine SCALE. If data scaling is done by the user, the necessary scaling parameters FIRSTV and DELTAV must be calculated and stored in the proper location before calling LINE. Refer to SCALE for a description of the values.

Figure 2-2 illustrates the types of lines drawn by various combinations of LINTYP and INTEQ. The dummy axes are present for reference only.

Comments: See the SCALE subroutine regarding the use of the NPOINT parameter for arrays containing both X- and Y-axis values. At least one of the coordinates must be zero when using the LINE subroutine to plot data on an axis. Redefine the origin in the PLOT subroutine if another location is desired.

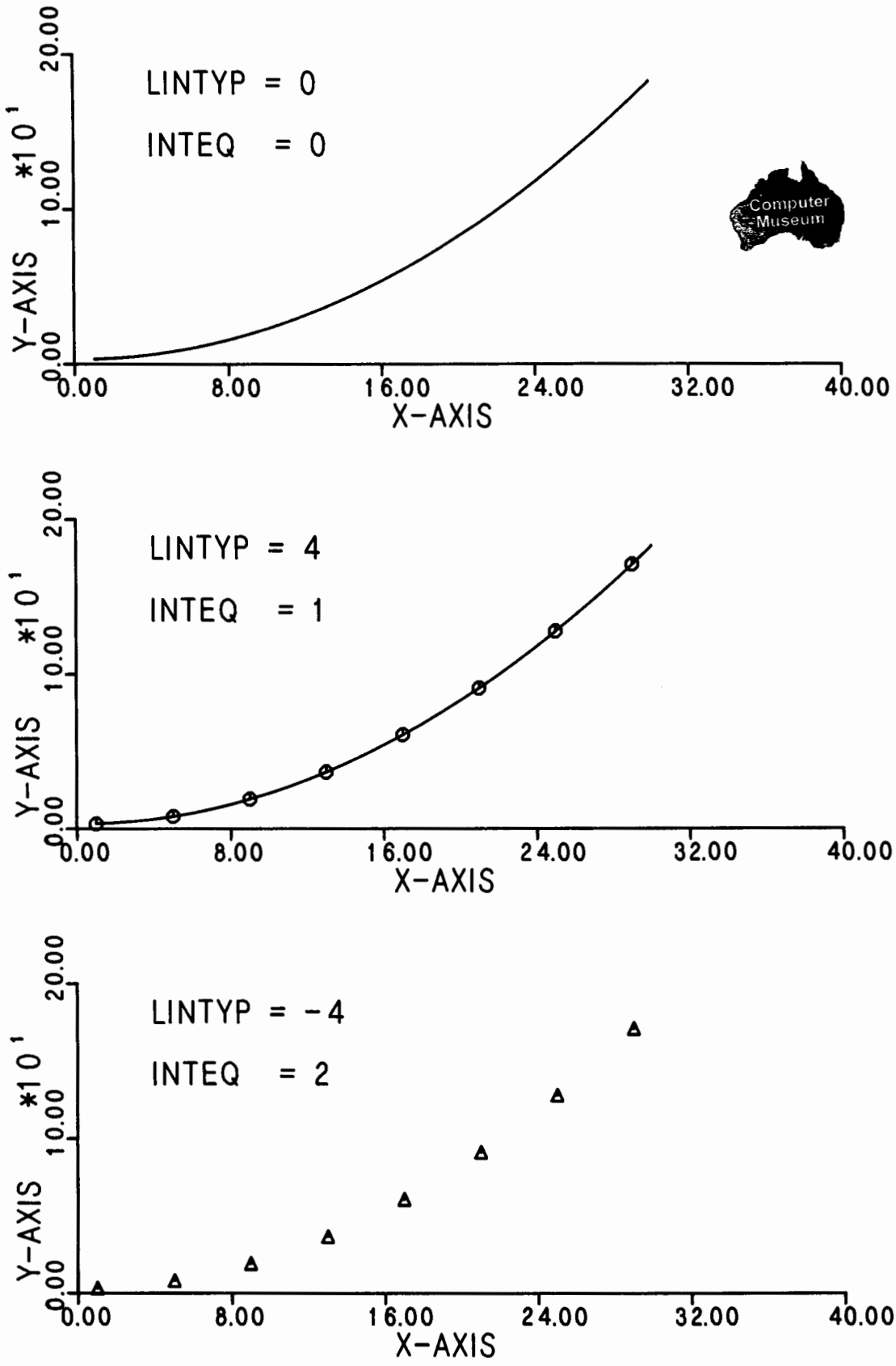


Figure 2-2. Examples of Symbol Plotting Using Subroutine LINE

The Call to NEWPEN

Purpose: Select a different pen to be used for all subsequent drawing, or return the current pen to its pen stable.

Format: CALL NEWPEN(IPEN)

2

Parameters:

IPEN The integer number of the new pen to be selected or a value of 0 to indicate the current pen is to be returned to its stable.

If the plotter is a multiple pen model and the value of IPEN is not greater than the number of pens supported, the pen selected is equal to the absolute value of IPEN.

If the value of IPEN is greater than the number of pens supported, then the pen selected is determined by the equation:

$$IPEN = (ABS(IPEN) - 1, \text{ modulo } n+1)$$

where n is the number of pens supported. For a 2-pen model, use a value of 2; for a 6-pen model, use a value of 6.

Function: The current pen is returned to its original stall in the plotter's pen stable, and the specified new pen is loaded into the pen holder. The pen returns to its previous graphic position with the pen up. If the original stall from which the pen was removed has subsequently been filled, or if the plotter power has been turned off and back on, the pen is returned to the lowest numbered empty stable. If there are no empty stables available, the plotter will ignore the request.

The Call to NUMBER

Purpose: Draw a floating point number, with or without the decimal point, at a specified height, location, and angle.

Format: CALL NUMBER(X, Y, HGT, FPN, THETA, ND)

Parameters:

X, Y Coordinates of the starting point in floating point inches, or if metric scaling has been specified, in centimetres.

HGT The actual height of the number in floating point inches, or if metric scaling has been specified, in centimetres.

If the value of X and/or Y is 999.0, annotation will be continued at the current X and/or Y coordinate. If only one coordinate is 999.0, the specified value of the other coordinate is used.

FPN The floating point number to be drawn.

THETA The angle at which the number is to be drawn, expressed in floating point degrees relative to the horizontal, 3 o'clock position.

ND The number of digits to the right of the decimal point.

- ND = 0 Draw as an integer with a decimal point.
- ND = −1 Suppress the decimal point and draw only the integer portion.
- ND < −1 Suppress the decimal point and truncate ABS(ND) −1 digits from the least significant portion of the value.
- ND > 0 Draw ND digits to the right of the decimal point.

Function: NUMBER converts a floating point number to the appropriate decimal equivalent so that the number may be plotted in FORTRAN "F" type format.

Comments:

1. The magnitude of ND should not exceed 7. If a value of ND greater than 7 is specified, a value of 7 is assumed.

2. The floating point number (FPN) will be drawn to the precision of the host processor's single precision variable. Or, it will be drawn to a total of 18 digits — including sign, decimal point, fractional part, and exponent — whichever is less.

   If an attempt is made to draw an FPN with greater than single precision accuracy, the least significant digits are undetermined and will vary.

The Call to OFFSET

Purpose: Set special scale factors and offsets to be used by calls to subroutine PLOT.

Format: CALL OFFSET(XOFF, XFCT, YOFF, YFCT)

2

Parameters:

XOFF The offset to be applied to the abscissa (X) values when subroutine PLOT is called.

XFCT The scaling factor to be applied to the abscissa (X) values when subroutine PLOT is called.

YOFF The offset to be applied to the ordinate (Y) values when subroutine PLOT is called.

YFCT The scaling factor to be applied to the ordinate (Y) values when subroutine PLOT is called.

Function: OFFSET allows the user to set special scale factors and offsets to be used by subroutine PLOT when it is called with an IPEN value of 12 or 13. These values are applied to the incoming X,Y coordinates prior to regular scale factors (i.e., those set by a call to FACTOR).

Offset and scaling factor values are expressed in the unit system defined by subroutine HPINIT, that is, inches or centimetres.

Comments:

1. When calling PLOT with IPEN codes 12 or 13, the input coordinates are adjusted as follows:

   $$X\ new = (X\ data - XOFF)/XFCT$$
   $$Y\ new = (Y\ data - YOFF)/YFCT$$

2. For normal coordinate orientation, i.e., increasing axis values from left to right and bottom to top, positive XOFF and YOFF values produce negative (down or to the left) offsets. Values of XFCT and YFCT greater than 1.0 produce a decrease in the size of a plot. Values less than 1.0 produce an increase in the size of the plot.

The Call to PLOT

Purpose: Perform absolute data plotting with pen control.

Format: CALL PLOT(X, Y, IPEN)

Parameters:

X, Y Coordinates in floating point inches (centimetres if metric scaling has been specified), of the vector endpoint with respect to the current origin.

IPEN An integer pen control parameter. Except when IPEN = 999, the sign of IPEN has the following meaning:

IPEN > 0 The pen is moved to the specified coordinate point relative to the current origin.

IPEN < 0 The pen is moved to the specified coordinate point relative to the current origin, and then the origin is redefined to that point.

The tens digit of IPEN may be 0 or 1 and has the following meaning:

0 = Move to the specified point as modified only by the scale factor specified by a call to FACTOR.

1 = Move to the specified point as modified by the scale and offset factors specified by a call to OFFSET. The scale factor specified by a call to FACTOR is applied after that specified by a call to OFFSET.

The units digit of IPEN has the following meaning:

2 = Move to the specified point with the pen down.

3 = Move to the specified point with the pen up.

If IPEN = 999, all data in the host buffer will be transmitted to the plotter and executed. The pen will be raised and moved to the location specified by parameters X and Y, and the plotter will be turned logically "off." The point specified by X,Y will be defined as the new logical origin by another call to PLOT with IPEN < 0 or by a call to HPINIT.

The Call to SCALE

Purpose: Scale an array of floating point numbers for one axis of a specified size graph.

Format: CALL SCALE(ARAY, AXLEN, NSCALE, NPOINT)

2

Parameters:

ARAY An array of real values to be scaled to one axis. The array must be dimensioned to a size sufficient to hold the scaling factors calculated by this subroutine. Refer to the definition of NPOINT and comments below for an explanation.

AXLEN Length of the axis to which the data is to be scaled. The value must be greater than 1.0 inch.

NSCALE The integer number of data values contained in ARAY to be used in calculating scaling factors, i.e., the number of data points to be plotted.

NPOINT An integer whose magnitude is the increment used to select the data values in the array to be scaled:

NPOINT = 1 Scale every point.
NPOINT = 2 Scale every other point.
NPOINT = n Scale every nth point.

If NPOINT is positive, the selected axis starting value (FIRSTV) approximates the minimum data value, and the scale factor calculated by subroutine SCALE will be positive.

If NPOINT is negative, the selected axis starting value (FIRSTV) approximates the maximum data value, and the scale factor calculated by subroutine SCALE (DELTAV) will be negative.

If NPOINT is ±1, the array must be dimensioned at least two elements larger than the actual number of data values it contains. If the magnitude of NPOINT is greater than 1, the computed values are stored at (NPOINT) elements and (2 * NPOINT) elements beyond the last data point. The subscripted element for FIRSTV is ARAY(NSCALE * |NPOINT| + 1); for DELTAV, it is ARAY(NSCALE * |NPOINT| + |NPOINT| + 1).

Refer to AXIS for an explanation of FIRSTV and DELTAV. See the comment below regarding the use of NPOINT in subroutines SCALE and LINE.

Function: The SCALE subroutine is used to examine the data values in an array and determine a starting value (minimum or maximum) and a scaling factor (positive or negative) such that:

1. The scale annotation drawn by the AXIS subroutine at each division will properly represent the range of real data values in the array.

2. The data points, when plotted by the LINE subroutine, will fit in a given plotting area.

These two values are computed and stored by SCALE at the end of the array.

The user's program will typically accumulate plotting data in two arrays:

An array of independent variables, X_i.

An array of dependent variables, $Y_i = f(X_i)$.

It would be unusual if the range of values in each array corresponded exactly with the number of inches available in the actual plotting area. For many problems, the range of the data is predictable. The programmer can predetermine the suitable conversion factors used in drawing the axis scale values and plotting the data points on the graph. Normally, however, these factors are not known in advance.

The scaling factor (DELTAV) that is computed represents the number of data units per inch of axis (centimetres if metric scaling is specified). The calculated value is adjusted up or down as necessary so that it is always a multiple of 1, 2, 4, 5, or $8 * 10^n$, where n is an exponent consistent with the calculated scaling factor. This results in more uniform axis labeling.

The starting value (FIRSTV), which will appear as the first annotation on the axis, is computed as some multiple of DELTAV that is equal to or outside the limits of the data in the array.

The integers NSCALE and NPOINT are used by the SCALE routine to determine which of the elements (data values) in ARAY are to be scaled. These elements are scanned for their minimum and maximum data values, and two adjusted scale values are calculated and stored in ARAY as follows:

1. The adjusted minimum value is a number less than or equal to the minimum data value. It is stored in location ARAY(NSCALE * |NPOINT| + 1).

2. The adjusted "delta" value is the result of subtracting the minimum data value from the maximum data value, dividing by the length of the axis (ALEN), and adjusting to provide one-inch increments that will span the data. This adjusted delta value is stored in location ARAY(N), where N = (NSCALE * |NPOINT| + |NPOINT| + 1).

The adjusted scale values are used by the LINE and AXIS routines.

Comments:

1. Separate calls to SCALE are required for the X- and Y-axes. ARAY contains the values pertinent to only one of the axes.

2. The array must be dimensioned two locations larger than NSCALE * NPOINT, which is the number of locations necessary for the data values in ARAY. Typically, NPOINT = 1, so that an array Z of 40 data values should be dimensioned with 42 locations: Z(42).

The following examples illustrate some typical uses of SCALE:

EXAMPLE 1

Assume an array containing 30 data values is to be plotted using a five-inch axis. The minimum data value is 1.0 and the maximum value is 30.0. The following statement may be used to scale the array:

```
CALL SCALE (ARAY, 5.0, 30, 1)
```

The subroutine will calculate the following values:

Results:

Units/Inch = (30.0 − 1.0)/5.0 = 5.8

Therefore:

DELTAV (next higher axis increment value) = 8.0

FIRSTV (minimum axis value) = 0.0

The FIRSTV value is stored in ARAY(31).

The DELTAV value is stored in ARAY(32).

Using these values, AXIS would draw the following axis line:

```
CALL AXIS (X, Y, LARAY, -NCHARS, 5.0, 0.0, ARAY(31), ARAY(32) )
```

EXAMPLE 2

Assume that the array in Example 1 is to be plotted on a four-inch axis, starting with the maximum value.

```
CALL SCALE (ARAY, 4.0, 30, -1)
```

would produce:

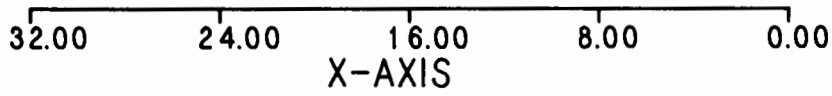
DELTAV = (1.0 − 32.0)/4.0 = −7.75, which is adjusted to −8.0

Minimum axis increment value = 0.0

FIRSTV = Minimum + (AXLEN * ABS(DELTAV))

= 0.0 + (4.0 * ABS(−8)) = 32

Again, the FIRSTV is stored in ARAY(31), and DELTAV is stored in ARAY(32). The call above would produce the following axis:



EXAMPLE 3

Assume a data array contains 200 values, ranging from a minimum of −9.0 to a maximum of 22.0. One hundred of the values will make a satisfactory plot, so every other value is to be plotted on an axis four inches long using a negative increment (declining axis values from left to right). The necessary FORTRAN DIMENSION statement in the program must dimension the array to contain a total of 204 locations. The necessary call to SCALE is:

```
CALL SCALE (ARAY, 4.0, 100, -2)
```

Therefore:

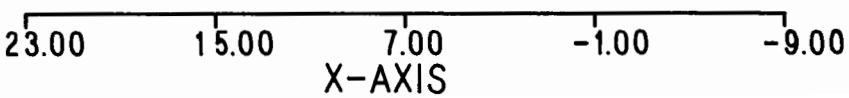
Initial DELTAV = (22.0 − (−9.0))/4.0 = 7.75

Since NPOINT is negative, DELTAV must also be negative and adjusted to the next available axis increment. Therefore, DELTAV = −8.0.

FIRSTV = Minimum Data Value + (AXLEN * ABS(DELTAV))

= −9.0 + (4.0 * 8.0) = 23.0

The resulting axis would appear as follows:



3. Since the SCALE subroutine is used in conjunction with the LINE subroutine, it is important to understand the use of the parameter NPOINT by both. NPOINT refers to the subscripting increment between significant entries in the array. Generally, this value is one, but there are occasions when the value might be greater than one. For example, if two or more arrays of data are intermixed, NPOINT equals the sum of the number of arrays.

EXAMPLE:

A DIMENSION statement may define a single array for convenience in storing data as sequential (X,Y) coordinate pairs, i.e., X1, Y1, X2, Y2, . . . Xn, Yn. In this case, NPOINT equals 2.

The statement:

```
DIMENSION ARAY(24)
```

reserves 20 locations for the data (10 X,Y pairs) and 4 locations for the scaling parameters to be computed by SCALE (2 locations for the X values and 2 locations for the Y values).

The following calls:

```
CALL SCALE (ARAY(1), XLEN, 10, 2)

CALL SCALE (ARAY(2), YLEN, 10, 2)
```

compute the scaling parameters. Observe that ARAY(1) is the location of the first X-data entry, and ARAY(2) is the location of the first Y-data entry. Since the value of NPOINT is 2, every other value in the array ARAY is scanned, starting with the first one named, either ARAY(1) or ARAY(2).

The computed scaling parameters are placed at:

ARAY(21) = FIRSTV for X
ARAY(22) = FIRSTV for Y
ARAY(23) = DELTAV for X
ARAY(24) = DELTAV for Y

The data is then plotted by a call to subroutine LINE as follows:

```
CALL LINE (ARAY(1), ARAY(2), 10, 2, LINTYP, INTEQ)
```

4. When a two-dimension array is used to store the data, the call to SCALE must specify the starting location for the data to be scaled. Generally, data is stored in such arrays dimensioned ARAY(2,n) where n is the number of X,Y data pairs in the array. In such cases, the necessary calls to SCALE are:

   For the X-axis:

   ```
   CALL SCALE (ARAY(1,NSCALE), XLEN, NSCALE, 2)
   ```

   For the Y-axis:

   ```
   CALL SCALE (ARAY(2,NSCALE), YLEN, NSCALE, 2)
   ```

   ARAY(i,1) specifies the start of each data for each axis, and NPOINT is equal to 2.

The Call to SYMBOL

Purpose: Draw ASCII characters or symbols at a specified location.

Format: CALL SYMBOL(X, Y, HEIGHT, LARAY, ANGLE, ICTL)

Parameters:

X, Y The coordinates in floating point inches (centimetres if metric scaling is specified) of the starting point of the character(s) or symbols to be drawn. If the value of X and/or Y is 999.0, annotation will be continued at the current X- and/or Y-coordinate. If only one coordinate is 999.0, the specified value of the other coordinate is used.

HEIGHT The actual height of the characters to be drawn in floating point inches (centimetres if metric scaling is used).

LARAY An array containing the characters to be drawn, stored as Hollerith data. Alternately, the first element of LARAY may contain any integer ranging from 0 through 14 representing the special centered symbol to be drawn (see Figure 2-3).

ANGLE The floating point number of degrees, counterclockwise from the horizontal, that the symbol or line of text is rotated.

ICTL An integer which indicates how the contents of LARAY are to be interpreted.

ICTL > 0 The number of ASCII characters to be drawn from array LARAY.

ICTL = 0 Draw only the single character contained in the low-order character position of the first element in element of LARAY.

ICTL < 0 Plot the special symbol represented by the integer in the first element of LARAY. If ICTL = −1, move to the starting location with the pen up. If ICTL < −1, move to the starting location with the pen down.

Function: The SYMBOL subroutine prduces plot annotation at any angle and in practically any size. There are two SYMBOL call formats:

1. The "standard" call which can be used to draw text such as titles, captions, and legends.

2. The "special" call, which is used to draw special centered symbols (a box, octagon, triangle, etc.) when plotting data points.

Comments:

1. Annotation may be continued from the current pen position or the position at which the last annotation ended. Continuing occurs when X and/or Y equals 999.0 and may be applied to X or Y independently. If only one coordinate value is specified as 999.0, the current value of that coordinate and the specified value of the other coordinate are used.

2. SYMBOL causes the characters to be drawn in the current standard plotter firmware set. Exceptions to this occur when: the characters to be drawn are centered symbols (where ICTL < 0); the HP 7550A and the HP 758X models have centered symbols in their firmware; all other plotters will have the centered symbol generated by software.

3. The starting point is the coordinates of the lower-left corner of an imaginary box drawn around the first character to be drawn. This point is expressed in inches or centimetres from the currently defined logical origin. On first initialization, the starting point is the 'P1' location on the plotter.

4. The maximum number of characters allowed in an LARAY is 68.

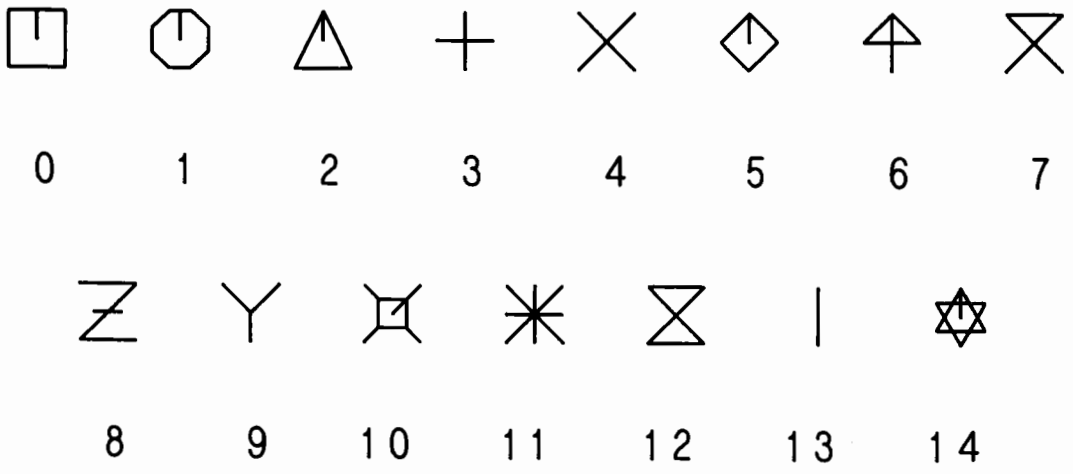


Figure 2-3. Centered Symbols

The Call to WHERE

Purpose: Report to the user program the current actual physical position of the plotter pen and the current scale factor.

Format: CALL WHERE(X, Y, FCT)

2

Parameters:

X, Y Coordinates of the current actual physical pen position.

FCT The current scaling factor (refer to FACTOR).

Function: The host data buffer is emptied to the plotter and all data in the plotter is executed. The plotter is then interrogated for its actual pen location. The values returned are converted to inches (centimetres if metric scaling is used) relative to the currently defined origin together with the scaling factor specified by the last program call to FACTOR.

Chapter 3
Accessing Advanced Plotter Features

The user may access advanced plotter features by transmitting the appropriate HP-GL commands through HP-ISPP software by calling subroutine BUFF from the main program. The BUFF subroutine provides seven I/O modes which allow the user to send HP-GL commands, parameters, and label strings to the plotter. The subroutine handles all data handshake operations. Subroutine BUFF does not perform any checks to assure the data to be sent is a valid HP-GL command or that any parameters are within a valid range (other than as specified for MODES 4 through 7 on the following page). It is the user's responsibility to assure that the command and parameter(s) specified are appropriate for the device used.

The following description of the BUFF subroutine outlines the function of the calling parameters as well as the seven I/O modes available.

The calling sequence is:

```
CALL BUFF (MODE, IBUFF, XBUFF, INUM)
```

Parameters:

MODE (Input/Output Code)

1 Insert characters stored as ASCII decimal equivalents in the array IBUFF into the host output buffer. The host output buffer is a COMMON array within HP-ISPP that is used to hold plotter data.

2 Insert characters stored as decimal equivalents in the array IBUFF into the host output buffer, then transmit the output buffer to the plotter regardless of the amount of remaining space.

3 Read data from the plotter. An array of one or more integer variables is returned in IBUFF, except when issuing an OI call (for plotter identification). In this case, the IBUFF array will contain an integer variable in IBUFF(1), plus the decimal equivalent for the model designation in IBUFF(2). For example, an OI call to an HP 7220C plotter would return IBUFF(1) = 7220 and IBUFF(2) = 67 (where 67 is the ASCII code for a 'C').

4 Convert the integer value(s) contained in the array IBUFF to ASCII characters and insert them into the output buffer. Transmit the host output buffer if it is then full. The permissible range of integer variables is −32 768 to +32 767. The form of the output is:

INT SEP INT SEP INT TERM

where: INT = signed/unsigned integers

SEP = a comma or semicolon

TERM = a semicolon or colon

If INUM is negative, the separator is a semicolon and the terminator is a colon. If INUM is positive, the separator is a comma and the terminator is a semicolon.

5 Same as MODE 4, but transmit the host output buffer regardless of the remaining space. An inquiry is not made regarding the available space in the plotter data buffer.

6 Convert the floating point number or numbers in the array XBUFF to ASCII and insert them into the output buffer. The permissible range of floating point variables is $\pm 2^{31} - 1$. The form of the output is:

DEC SEP DEC TERM

where: DEC = signed/unsigned real variables

SEP = comma

TERM = semicolon

7 Same as MODE 6, but transmit the host buffer to the plotter regardless of the remaining space. An inquiry is not made regarding available space in the plotter data buffer.

IBUFF Integer array for transferring data to the buffer or, if MODE = 3, the data returned by the plotter.

XBUFF Real array for transferring data to the buffer.

INUM The number of data elements in array IBUFF or XBUFF to be processed by the subroutine or which have been returned from the plotter and stored in IBUFF.

MODES 3, 5, and 7 are used when the commands and parameters transmitted will result in output from the plotter. The plotter data is then read by a succeeding call to BUFF with MODE = 3. When calling BUFF to send HP-GL commands, the normal sequence of events would be as follows:

1. Initialize the HP-ISPP software and plotter by either calling subroutine DPINIT, HPINIT, or PLOTS.

2. Issue a call to BUFF for a particular HP-GL command with MODE = 1. If no parameters accompany the command, INUM should be negative. This insures that the HP-GL command is transmitted with the appropriate terminator.

3. Issue a call to BUFF with the necessary parameters which accompany the command, if required.

4. Repeat steps 2. and 3. for as many command sequences as required.

5. Issue a call to PLOT with IPEN = 999 to terminate the plot.

The following sample program illustrates how the user would send HP-GL command sequences in order to take advantage of some of the special features of the HP 7220S/T plotter. In this example, the program establishes a user-unit scaling system, draws eight circles with different line types, and then puts the pen away. The necessary HP-GL commands are contained in the array IA, expressed as decimal equivalents of the desired characters. Two characters are required per HP-GL command. Array IA contains the decimal equivalent of one character in each location. Array IBUFF is used to hold the necessary command parameters. Array XBUFF would be used for real variable parameters, however, none are required in this example.

```
C
C   PROGRAM TO GENERATE CIRCLES WITH DIFFERENT LINE TYPES
C
C   A 16-BIT HOST COMPUTER IS ASSUMED
C
      DIMENSION LABL1(10), LABL2(4), MDES(1),ANUM(1)
      DIMENSION IA(14),IBUFF(4),XBUFF(1)
      DATA LABL1(1)/2HPL/, LABL1(2)/2HOT/, LABL1(3)/2H P/,
     +     LABL1(4)/2HRO/, LABL1(5)/2HDU/, LABL1(6)/2HCE/,
     +     LABL1(7)/2HD /, LABL1(8)/2HBY/, LABL1(9)/2H H/,
     +     LABL1(10)/2HP /
      DATA LABL2(1)/2H P/, LABL2(2)/2HLO/, LABL2(3)/2HTT/,
     +     LABL2(4)/2HER/
      DATA IA(1)/83/,IA(2)/67/
C                 S          C       'SCale'
      DATA IA(3)/80/,IA(4)/65/
C                 P          A       'Plot Absolute'
      DATA IA(5)/76/,IA(6)/84/
C                 L          T       'Line Type'
      DATA IA(7)/67/,IA(8)/73/
C                 C          I       'CIrcle'
      DATA IA(9)/73/,IA(10)/80/
C                 I           P      'Input Points P1 and P2'
      DATA IA(11)/79/,IA(12)/73/
C                  O           I     'Output Identification'
C
      DATA IA(13)/79/, IA(14)/67/
C                 O             C      'Output Commanded Position'
C
C   INITIALIZE THE ISPP SOFTWARE AND THE PLOTTER.  NOTE THAT
C   MONITOR MODE IS INVOKED FOR DISPLAY OF PLOTTER OUTPUT.
C
      CALL HPINIT(0, 1, 0, 0, 20)
C
C   REQUEST, THEN READ PLOTTER I.D.  (OI;)
C
      CALL BUFF(1,IA(11), XBUFF(1),-2)
      CALL BUFF(3,IBUFF(1), XBUFF(1), 2)
      ANUM(1) =IBUFF(1)
C
C   DETERMINE MODEL DESIGNATION FROM IBUFF(2)
C   WILL BE AN  A     B     C     S   OR   T
C   ASCII =    '65'  '66'  '67'  '83' OR  '84'
C
       MDES(1) = IBUFF(2)
C
C   IF MODEL NUMBER IS 7580 THEN SET NEW PLOTTER LIMITS IN
C   USER UNITS (IP-7500, -5000, 7500, 5000;)  OTHERWISE GO TO
C   SCALING
C
      IF (ANUM(1) .NE. 7580) GO TO 5
      CALL BUFF(1, IA(9), XBUFF(1), 2)
      IBUFF(1)=-7500
      IBUFF(2)=-5000
      IBUFF(3)=7500
      IBUFF(4)=5000
      CALL BUFF(4,IBUFF(1),XBUFF(1),4)
```

```
C
C   SCALE THE USER UNITS (SC-1500,1500,-1000,1000; )
C
  5   CONTINUE
      CALL BUFF(1, IA(1), XBUFF(1), 2)
      IBUFF(1)=-1500
      IBUFF(2)=1500
      IBUFF(3)=-1000
      IBUFF(4)=1000

      CALL BUFF(4, IBUFF(1),XBUFF(1), 4)
C
C   MOVE TO ARC STARTING LOCATION (PA0,0;)
C
      CALL BUFF(1, IA(3), XBUFF(1), 2)
      IBUFF(1)=0
      IBUFF(2)=0
      CALL BUFF(4, IBUFF(1), XBUFF(1), 2)
C
C   PRODUCE EIGHT CIRCLES WITH DIFFERENT LINE TYPES, EACH WITH
C   THE DEFAULT PATTERN LENGTH.
C
      LINE=-1
      IRAD=100
      DO 30 I = 1,8
          IF  (LINE .GE. 0) GO TO 10
C
C   INVOKE THE DEFAULT SOLID LINE TYPE  (LT; )
C
          CALL BUFF(1, IA(5), XBUFF(1), -2)
          GO TO 20
C
C   INVOKE THE OTHER SEVEN LINE TYPES (LTn;)
C
 10       CALL BUFF(1, IA(5), XBUFF(1), 2)
          IBUFF(1)=LINE
          CALL BUFF(4, IBUFF(1), XBUFF(1), 1)
C
C    DRAW A CIRCLE  ( CIirad, 5; )
C
 20       CALL BUFF(1, IA(7), XBUFF(1), 2)
          IBUFF(1)=IRAD
          IBUFF(2)=5
          CALL BUFF(4,  IBUFF(1), XBUFF(1), 1)
          LINE =LINE + 1
          IRAD=IRAD+100
 30   CONTINUE
      CALL NEWPEN(2)
C
C   REPOSITION PEN IN USER UNITS (PA -430, -900)
C
      CALL BUFF(1,IA(3),XBUFF(1),2)
      IBUFF(1)=-430
      IBUFF(2)=-900
      CALL BUFF(4,IBUFF(1),XBUFF(1),2)
```

```
C
C   LABEL PLOT WITH DEVICE I.D.
C   OUTPUT 1ST LABEL, MODEL NUMBER, MODEL DESIGNATION AND 2ND LABEL
C
      CALL SYMBOL (999.0,999.0,.15,LABL1,0.0,20)
      CALL NUMBER(999.0,999.0,.15,ANUM(1),0.0,-1)
      CALL SYMBOL (999.0,999.0,.15,MDES,0.0,2)
      CALL SYMBOL (999.0,999.0,.15,LABL2,0.0,8)
C
C   PUT THE PEN AWAY AND TERMINATE THE PLOT
C
      CALL NEWPEN(0)
      CALL PLOT(0.,0., 999)
      STOP
      END
```

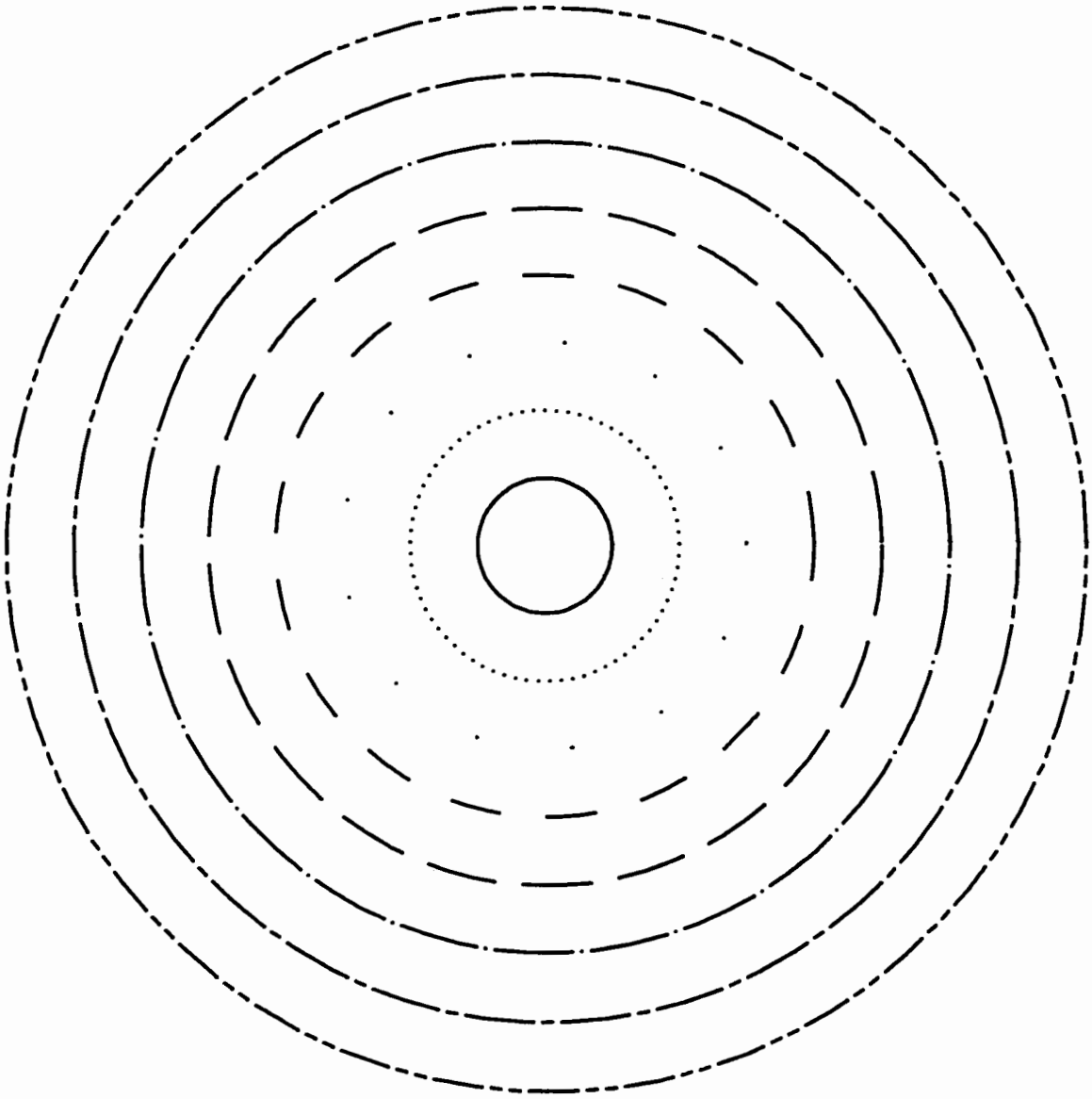


Figure 3-1. Illustration Using Subroutine BUFF to Format HP-GL Instructions

Appendix A
Global Variables

The HP Industry Standard Plotting Package contains a group of global variables that are used for communication between subroutines. The global variables reside in a "labeled common" block called "HP." The global variables and their functions are listed below.

Real Variables:

RSI1 — Tracks the character size width attribute.

RSI2 — Tracks the character size height attribute.

RDI1 — Tracks the character direction run attribute.

RDI2 — Tracks the character direction rise attribute.

RES1 — Tracks the character spaces attribute for the HP 7550A and the HP 758X models.

RES2 — Tracks the character lines attribute for the HP 7550A and the HP 758X models.

XMINPU — Contains the minimum X-coordinate point in plotter units in X.

YMINPU — Contains the minimum Y-coordinate point in plotter units in Y.

XORG — Contains the current X-origin in plotter units.

YORG — Contains the current Y-origin in plotter units.

XOFF — Contains the current X-offset in plotter units.

YOFF — Contains the current Y-offset in plotter units.

XFACT — Contains the current X-scale factor.

YFACT — Contains the current Y-scale factor.

XYUNIT — Contains the current number of plotter units per inch or centimetre (1016 plotter units per inch; 400 plotter units per centimetre).

Integer Variables:

MDEV — Contains the HP device type (7220, 7470, 7475, 7550, 7580, 7585, 7586, or 9872).

MVER — Contains the ASCII value of the device version (A, B, C, S, T).

MUNIT — Contains the code for the unit system enabled (0 = inches/ 1 = metric).

MPENS — Contains the number of pens supported (2, 4, 6, or 8).

MLU — Contains the logical unit number for I/O.

MCNTRL — Contains the code used by subroutines SYMBOL and PLOT to signal annotation continuation (X or Y was 999.0 in a call to subroutine SYMBOL).

MINIT — Contains the code for software system initialization (0 = not initialized/1 = initialized).

MSHAKE — Contains the handshake method code (0 = primary/1 = secondary/2 = HP-IB (IEEE-488-1978) Alternate Interface/3 = no handshake.

MPTR — Contains the pointer to the next location in MOUT.

MCODE — Contains the pen status code (0 = pen is up/1 = pen is down).

MPLOT — Contains the parameter concatenation code used by subroutine BUFF (0 = don't concatenate/1 = concatenate).

MFLUSH — Contains the code used to signal that a forced transmission of the output buffer must be made (0 = no forced transmission/999 = forced transmission).

MSIZE — Contains the size of the output array MOUT.

MSPOOL — Determines spooling mode (0 = normal/1 = deferred mode).

MOUT — An array containing the data to be sent to the plotter. MOUT is dimensioned MOUT(72).

Some FORTRAN systems require that any common or named common blocks referenced in a subroutine must also be referenced in the main program. The following COMMON declaration may be used to satisfy this requirement:

```
COMMON/HP/RDUM1(11),IDUM1(86),RDUM2(6),/ZZCOM/IDUM2(42)
```

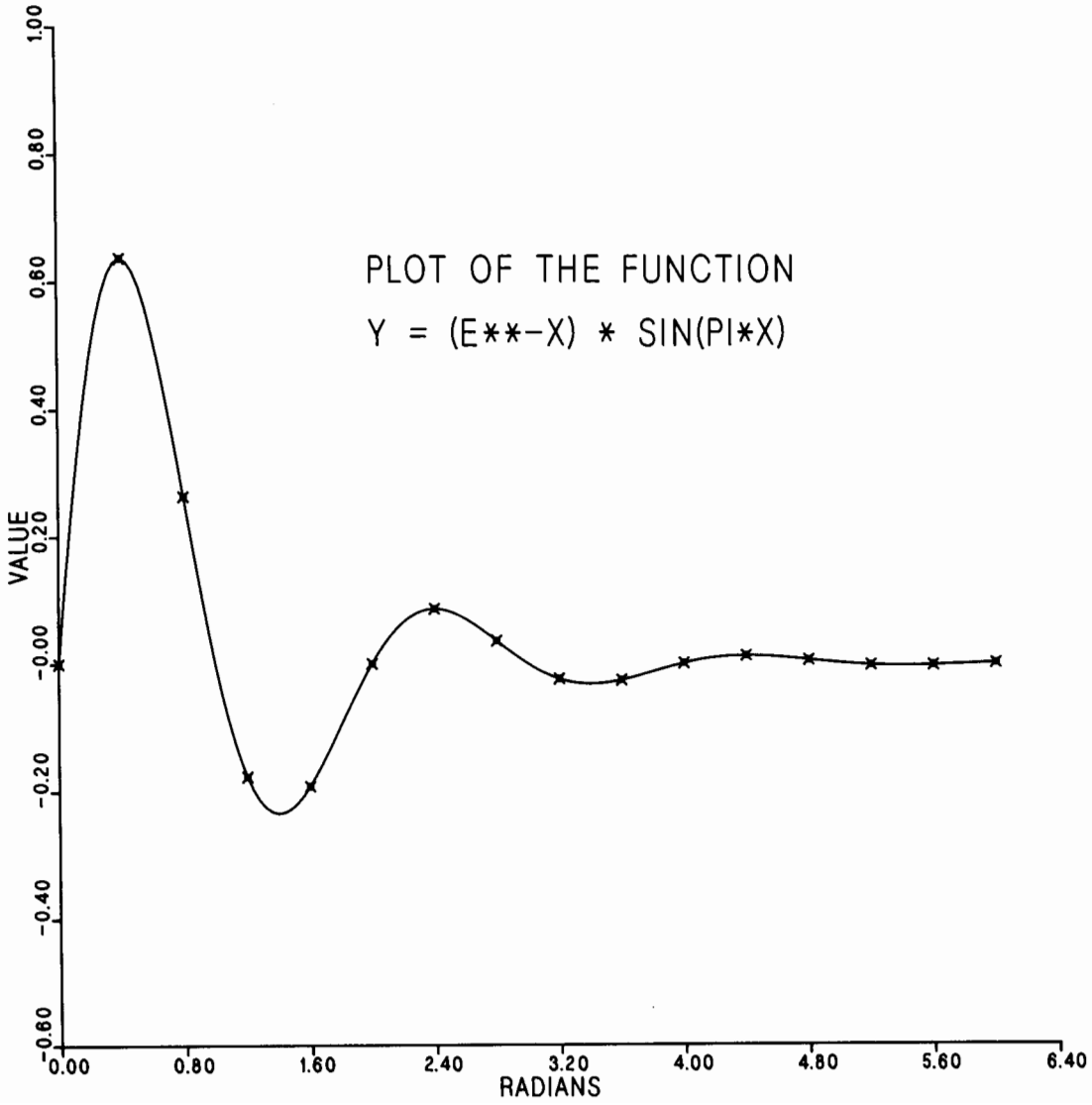
Some systems may require that named COMMON blocks be declared in a separate BLOCK DATA subprogram. If initialization of the individual variables is also required to occur in the BLOCK DATA subprogram, each variable may be initialized to "0.0" (real variables) or to "0" (integer variables).

Appendix B
Sample Program

The following sample program shows the use of subroutines SCALE, AXIS, LINE, and SYMBOL to generate a graph.

```
2            PROGRAM SAMPLE
3       C
4       C  THIS PROGRAM PRODUCES A PLOT USING SCALE, AXIS, LINE AND SYMBOL.
5       C  OUTPUT IS DIRECTED TO FORTRAN UNIT NUMBER 20.
        C
6            DIMENSION L1(4),L2(3),L3(10),L4(12)
7            DIMENSION X(153),Y(153)
8            DATA L1 /2HRA, 2HDI, 2HAN, 2HS /
9            DATA L2 /2HVA, 2HLU, 2HE /
10           DATA L3 /2HPL, 2HOT, 2H 0, 2HF , 2HTH, 2HE , 2HFU, 2HNC,
            +         2HTI, 2HON/
11           DATA L4 /2HY , 2H= , 2H(E, 2H**, 2H-X, 2H) , 2H* , 2HSI,
12          +         2HN(, 2HPI, 2H*X, 2H) /
14           PI = 3.14159265
15           E  = 2.71828183
16      C
17      C  CALCULATE THE DATA POINTS
18      C
19           DO 10 I = 1,151
22               XARG = 0.04 * FLOAT(I-1)
23               X(I) = XARG
24               Y(I) = (E**(-XARG)) * SIN(PI * XARG)
25       10  CONTINUE
26      C
27      C  INITIALIZE THE PLOTTER SOFTWARE SYSTEM
28      C
29           CALL HPINIT(0, 0, 0, 0, 20)
31      C
32      C  SCALE THE DATA
33      C
34           CALL SCALE(X, 8.0, 151, 1)
35           CALL SCALE(Y, 8.0, 151, 1)
36      C
37      C  DRAW THE X AND Y AXES
38      C
39           CALL AXIS(0.5, 0.5, L1, -7, 8.0, 0.0, X(152), X(153))
40           CALL AXIS(0.5, 0.5, L2, 5, 8.0, 90.0, Y(152), Y(153))
41      C
42      C  PLOT THE DATA POINTS WITH CONNECTING LINES
43      C
44           CALL NEWPEN(2)
45           CALL PLOT(0.5, 0.5, -3)
46           CALL LINE(X, Y, 151, 1, 10, 10)
```

```
47          C
48          C  PRODUCE THE TITLE
49          C
50                CALL NEWPEN(3)
51                CALL SYMBOL(2.5, 6.0, 0.2, L3, 0.0, 20)
52                CALL SYMBOL(2.5, 5.5, 0.2, L4, 0.0, 23)
53          C
54          C  PUT THE PEN AWAY AND TERMINATE THE PLOT
55          C
56                CALL NEWPEN(0)
60                CALL PLOT(0.0, 0.0, 999)
61                STOP
62                END
```



B

Appendix C
Subroutine Cross-Reference Table

AXIS

Called by: None
Calls: PLOT, SYMBOL, NUMBER, ZZPACK

BIN

Called by: BUFF
Calls: ZZGET

BOUT

Called by: BUFF
Calls: ZZGET, ZZPUT

BUFF

Called by: WHERE, SYMBOL, PLOTS, PLOT, NEWPEN, HPINIT, FPINIT
Calls: BIN, BOUT

DPINIT

Called by: None
Calls: HPINIT

FACTOR

Called by: None
Calls: None

FPINIT

Called by: None
Calls: HPINIT, PLOT, BUFF, ZZGET

HPINIT

Called by: PLOTS, DPINIT, FPINIT
Calls: BUFF, ZZINIT

LINE

Called by: None
Calls: PLOT, SYMBOL

NEWPEN

Called by: None
Calls: BUFF

NUMBER

Called by: AXIS
Calls: SYMBOL, ZZPACK

OFFSET

Called by: None
Calls: None

PLOT

Called by: SYMBOL, LINE, AXIS, FPINIT
Calls: BUFF

PLOTS

Called by: None
Calls: BUFF, HPINIT

SCALE

Called by: None
Calls: None

SYMBOL

Called by: NUMBER, LINE, AXIS
Calls: ZZEXTR, ZZPACK, ZZLOW, PLOT, WHERE, BUFF

WHERE

Called by: SYMBOL
Calls: BUFF

XXORG

Called by: HPINIT
Calls: None

ZZGET

Called by: BOUT, BIN, FPINIT
Calls: ZZEXTR, ZZPACK

ZZHOST

Called by: ZZINIT
Calls: None

ZZINIT

Called by: HPINIT
Calls: ZZHOST

ZZLOW

Called by: SYMBOL
Calls: None

ZZPACK

Called by: ZZGET, SYMBOL, NUMBER, AXIS, ZZPUT
Calls: None

ZZPUT

Called by: BOUT
Calls: ZZPACK

ZZEXTR

Called by: ZZGET, SYMBOL
Calls: None

C

Appendix D
Plotter Character Sets

This appendix briefly describes the character fonts and lists the character sets available for each plotter.

A font is a letter style. Three fonts are used by the plotters:

Fixed-space vector font — Characters occupy an equal horizontal space and are always drawn using a fixed number of vectors.

Variable-space arc font — Characters are proportionately spaced; the amount of horizontal space occupied by each character varies with the character. The characters also have a contour smoothness which is programmable.

Fixed-spaced arc font — All characters occupy an equal horizontal space and contain the same programmable contour smoothness as an arc-drawn character.

The three fonts can be used in any of 20 possible character sets. A drafting character set, 99, is available. And a user-defined character set, −1, is available for downloadable characters on the HP 7550A and the HP 7586B. The HP 7580B and HP 7585B models with a serial prefix number of 2402 have the −1 character set.

In the following table, each of the first three columns represents a font. The character sets on the same line have the same characters in the same positions (the same mapping from bit pattern to displayed symbol).

Fixed-Space Vector Font	Variable-Space Arc Font	Fixed-Space Arc Font	Character Set	ISO Number
−1	—	—	Downloadable	—
0	10	20	ANSI ASCII	006
1	11	21	HP 9825 HPL Set	—
2	12	22	French/German	—
3	13	23	Scandinavian	—
4	14	24	Spanish/Latin American	—
5	15	25	Special Symbols	—
6	16	26	JIS ASCII	014
7	17	27	Roman Extensions	—
8	18	28	Katakana	013
9	19	29	ISO IRV (International Reference Version)	002

Fixed-Space Vector Font	Variable-Space Arc Font	Fixed-Space Arc Font	Character Set	ISO Number
30	40	50	ISO Swedish	010
31	41	51	ISO Swedish for Names	011
32	42	52	ISO Norwegian, Version 1	060
33	43	53	ISO German	021
34	44	54	ISO French	025
35	45	55	ISO British	004
36	46	56	ISO Italian	015
37	47	57	ISO Spanish	017
38	48	58	ISO Portuguese	016
39	49	59	ISO Norwegian, Version 2	061
99	—	—	Drafting	—

The following table lists the available character sets for each plotter.

HP 7220A/S	0-4
HP 7220C/T	0-4
HP 7470A	0-4
HP 7475A	0-4, 6-9, 30-39
HP 7550A	−1, 0-19, 30-49
HP 7580A	0-5, 10-15
HP 7580B	0-5, 10-15
HP 7580B*	−1, 0-59, 99
HP 7585A	0-5, 10-15
HP 7585B	0-5, 10-15
HP 7585B*	−1, 0-59, 99
HP 7586B	−1, 0-59, 99
HP 9872B/S	0-4
HP 9872C/T	0-4

*Plotters with serial prefix number 2402

Subject Index

h

l

m

n

o

p

r

s

w

17580B.00.00
PART NO. 17580-90004
MICROFICHE NO. 17580-90054

SEPTEMBER 1984
PRINTED IN U.S.A.