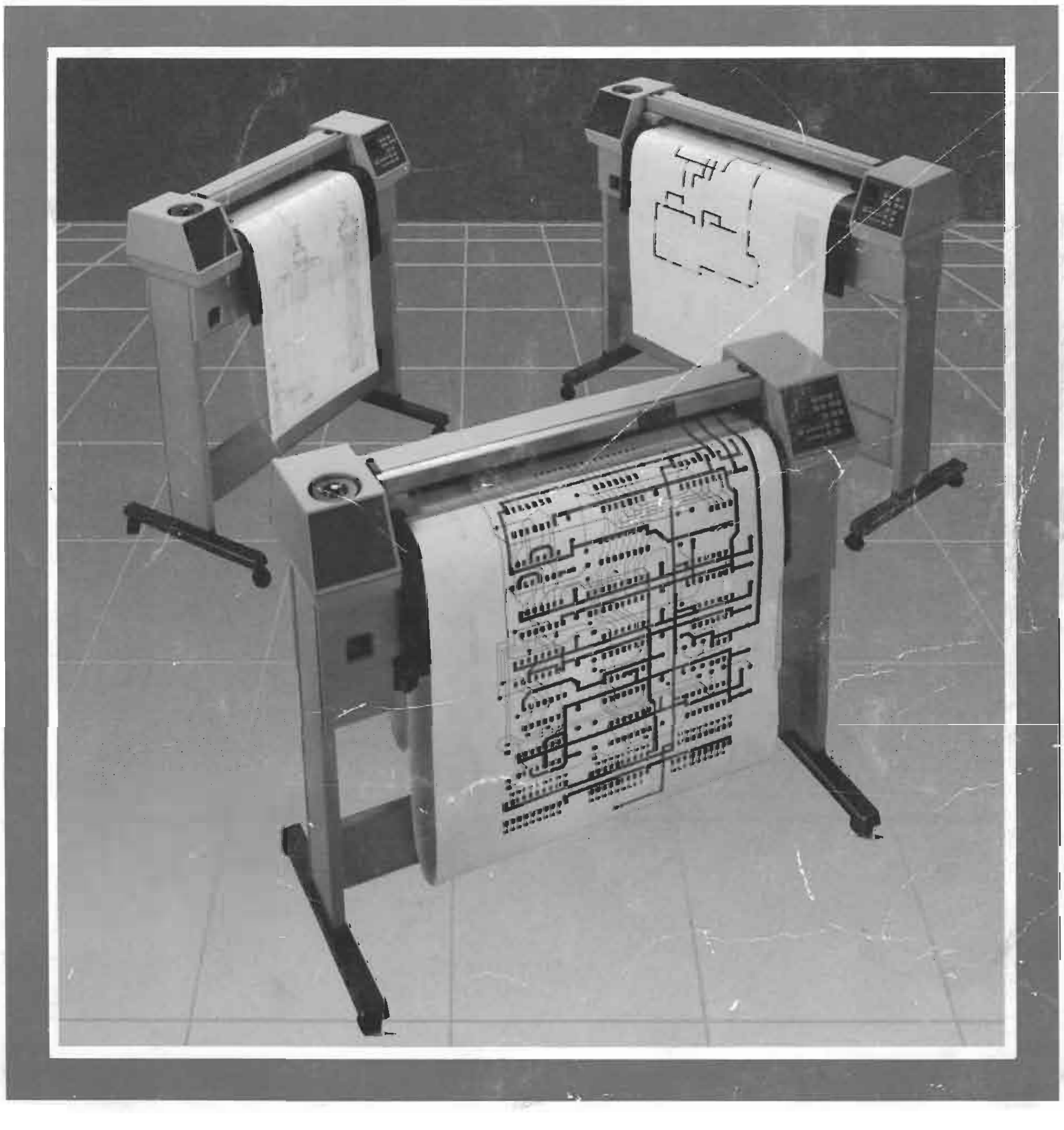
Hewlett-Packard
7580B, 7585B, and 7586B Drafting Plotters
Interfacing and Programming Manual

Using HP-GL Instructions

DESIGNED FOR HP-IB SYSTEMS

RS-232-C/CCITT V.24

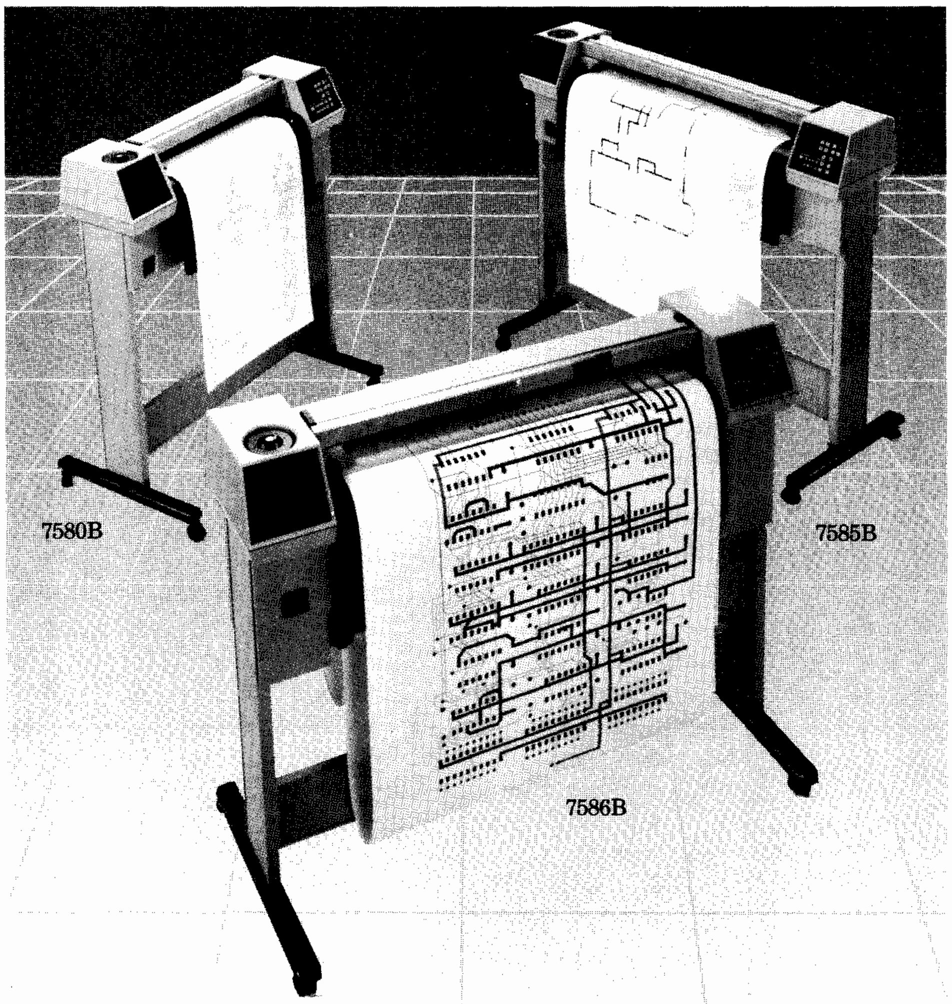


Hewlett-Packard
7580B, 7585B, and 7586B Drafting Plotters
Interfacing and Programming Manual

Using HP-GL Instructions

DESIGNED FOR HP-IB SYSTEMS

RS-232-C/CCITT V.24



NOTE: *This manual applies to HP 7580B and HP 7585B plotters with serial prefix number* ***2402*** or higher.

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

Manual Summary

Chapter 1
Owner's Information
Contains information concerning manual usage, accessories, inspection, power and line voltage options, and maintenance.

Chapter 2
Plotter Operation
Describes the physical and operating characteristics of the plotter, introduces Hewlett-Packard Graphics Language (HP-GL) and device control instructions, and defines the conditions governing HP-GL instruction examples.

Chapter 3
Scaling, Windowing, and Rotation
Explains the methods for locating the scaling points and the instructions which can be used to specify user unit scaling, plotting limits (windows), and rotation of the coordinate system.

Chapter 4
Pen Control and Plotting
Describes the instructions for pen control, absolute and relative vector plotting, and the graphics limits.

Chapter 5
Circles, Arcs, and Polygons
Describes arc and circle generation and how to define, edge, and fill a polygon.

Chapter 6
Plot Enhancement
Describes instructions for drawing tick marks and differentiating traces.

Chapter 7
Labeling Plots
Describes the instructions used in labeling. Included are instructions to vary direction, size, slant, and spacing of the characters, as well as instructions for character set selection and label positioning.

Chapter 8
Digitizing
Describes the instructions used to digitize with the plotter and demonstrates how to check for the presence of a digitized point.

Chapter 9
General Programming Instructions
Describes the instructions to obtain general output information.

Chapter 10
Putting the Instructions to Work
A step-by-step example illustrating the procedures to be followed to draw, label, and plot data using HP-GL instructions.

Manual Summary
(Continued)

Chapter 11
Plotting with Roll Media
Describes roll-feed operations, HP-GL page-advance instructions, and long axis plotting on the HP 7586.

Chapter 12
HP-IB Interfacing
Summarizes operation of the plotter with the Hewlett-Packard Interface Bus (HP-IB) and explains the methods for addressing and sending/receiving data over the interface bus.

Chapter 13
RS-232-C/CCITT V.24 Interfacing
Summarizes the methods for establishing a handshake protocol between the plotter and computer, and explains the device control instructions (escape sequences) that are used to set up and control the handshake protocol.

Appendix A
An HP-IB Overview
Provides an overview of the Hewlett-Packard Interface Bus (HP-IB)

Appendix B
Instruction Syntax
Included is a summary of both HP-GL and device control instructions.

Appendix C
Reference Material
Included is a summary of default conditions, error messages, scaling equations, ASCII codes, and character sets.

Table of Contents

Table of Contents
(Continued)

Table of Contents (Continued)

Table of Contents
(Continued)

Table of Contents (Continued)

Table of Contents
(Continued)

Chapter 12

HP-IB Interfacing (Continued)

Table of Contents (Continued)

Table of Contents
(Continued)

Chapter 1
Owner's Information

What You'll Learn in This Chapter

In this chapter you will learn what is covered in this manual and what other manuals you may need or find useful. In addition, this chapter contains a description of the HP 7580B, 7585B, and 7586B Drafting Plotters and general information to enable you to verify that your plotter has not been damaged in shipment and is compatible with the power available in your area.

Terms You Should Understand

HP-GL (Hewlett-Packard Graphics Language) — the graphics language instructions understood by these plotters and other HP graphics devices. The instruction's two-letter-mnemonic is suggestive of its role. For instance, PA is used to plot to absolute coordinates, SP is used to select a pen, and DR is used to establish the relative direction of labeling.

HP-IB (Hewlett-Packard Interface Bus) — HP's implementation of IEEE standard 488-1978 digital interface for programmable instrumentation, commonly found on HP desktop computers, and some larger computers.

RS-232-C/CCITT V.24 Interface — another popular standardized interface. It is commonly found on large computers and in environments where the plotter will be placed at a considerable distance from the computer.

How to Use HP 7580/7585/7586 Documentation

This manual contains operating, interfacing, and programming information for the HP 7580B, 7585B, and 7586B Drafting Plotters.* All plotters are equipped with dual interfacing capabilities that allow switch selection of either the HP-IB Interface Mode or RS-232-C Interface Mode. HP-GL is used in both interface modes to control the plotter's graphics capabilities. Unless specifically noted, all information in this manual pertains to both interface modes.

NOTE: All information in this manual for the RS-232-C interface applies equally to the CCITT V.24 interface. For purposes of simplicity, both are referred to as RS-232-C. ■

Documentation for this plotter has been designed to enable you to use the plotter to its full potential. All plotters are shipped with this manual, an Operator's Manual (Part No. 07580-90033), and a Reference Card (Part No.

**This manual applies to HP 7580B and 7585B plotters with serial prefix number 2402 or higher.*

07580-90035). The Operator's Manual contains all information you will need to select pen and media combinations and operate, but not program, the plotter. The Reference Card contains a summary of HP-GL and device control instructions, plotter default conditions, and a list of error numbers and their meanings. Depending on the options specified, you may also have received a Programming Guide for use with a specific HP computer. The plotters are supported on a number of HP computer systems using higher level programming support than is covered in this manual. In most cases, high level graphics support is available through graphics programming ROMs or software, each of which is supported with a comprehensive user's manual which will answer most of your questions related to programming. Contact your HP sales representative regarding high level graphics support available with your HP computer.

For First Encounters with the HP 7580, 7585, or 7586

If you have just received your plotter, read Chapter 1 before attempting to operate the plotter. After inspecting your plotter and accessories, refer to Chapter 12 for initial setup and addressing for the HP-IB Interface Mode or Chapter 13 for initial setup and a description of handshaking protocol for the RS-232-C Interface Mode. If you are only concerned with plotter operation, you may then read the Operator's Manual.

For First Encounters with HP-GL

If you have never programmed in HP-GL, after reading the appropriate interfacing chapter, read Chapters 2 through 9 in order. These chapters contain detailed operating information and a description of each instruction and its uses. Running the examples given with the instructions will help you learn. Next, read Chapter 10 to see how the instructions work together in a program.

For Experienced HP-GL Programmers

If you are an experienced HP-GL programmer, you may find the instruction summary in Appendix B of this manual or the Reference Card most helpful. Since there are differences in operation and syntax between this and other plotters, you should read Chapter 2 of this manual before programming. The HP 7580, 7585, and 7586 have capabilities not found in earlier plotters. Among these are the ability to mirror labels, using negative size and direction parameters, and mirror vectors, when user unit scaling is specified. To understand these differences, you need to read the sections on scaling (SC, Chapter 3), and setting label size and direction in Chapter 7. In Appendix B, page numbers for the complete description are listed with each instruction summary.

Understanding Manual Conventions and Syntax

Before reading any part of this manual, you should understand the meaning of type styles, symbols, and number representation used in text. A detailed explanation of syntax symbols is given in the section entitled HP-GL Syntax, Chapter 2, and Syntax for Device Control Instructions, Chapter 12. The following conventions also apply. Words typed in **SMALL BOLDFACE TYPE** are either buttons, switches, or words actually found on the plotter or computer. **BOLD CONDENSED** type is used to denote a single ASCII character which should be sent to the plotter. Numbers are typed using SI (International System of Units) standards; numbers with more than four digits are placed in groups of three, separated by a space instead of commas, counting both to the left and right of the decimal point (54 321.123 45).

A Brief Look at the Plotter

The plotter provides large format, multicolor, high-resolution vector plotting capability using a wide choice of drawing media and pen types. It will draw on paper, vellum, or double-matte polyester sheets with a choice of fiber-tip, roller-ball, or liquid-ink drafting pens.

The HP 7580 plotter easily accommodates English (ANSI) A through D and metric (ISO) A4 through A1 size media. In addition to the smaller sizes, the HP 7585 plotter easily accommodates English E and metric A0 size media. The HP 7586 plotter accommodates all of these sizes and several sizes of roll media.

The extensive multicolor graphics capability is provided by programmed or front-panel selection of up to eight pens from one of three manually interchangeable pen carousels. Each carousel is designed to hold pens of a specific type and is clearly marked to identify the pen type that it will cap properly. The carousel type is electronically sensed to establish default pen speed, force, and acceleration parameters.

Paper size is also electronically sensed to establish plot limits, and default conditions are automatically established for all other plotting parameters. These settings have been optimized to produce quality graphics in a wide range of plotting applications. However, users who want additional flexibility can override default parameters using the front-panel controls or HP-GL programming instructions.

There are 77 HP-GL programming instructions for pen control, absolute and relative vector plotting, arc, circle, and polygon generation, output of pen position and plotter status, and labeling. HP-GL instructions also control label direction, size, slant, and spacing of characters as well as character set choice and label positioning. In addition, the HP 7586 has five instructions for advancing paper and long-axis plotting. All HP-GL instructions are initially stored in the plotter's data buffer to improve throughput. Your computer is free to perform other tasks while the plotter is processing graphics data from the buffer.

Both interface modes also use device control instructions as well as graphics instructions. With the HP-IB Interface Mode, device control instructions (escape sequences) are implemented to establish Monitor Mode, obtain immediate output of available buffer space, and ascertain the status of the plotter. With the RS-232-C Interface Mode, additional device control instructions are implemented to establish output mode conditions, handshake protocol, and to obtain immediate output of plotter-related data. Device control instructions are executed immediately and are never stored in the buffer.

Some applications for which the plotter is well suited are:

- Drafting/CAD
- Surveying
  - Earthfill charts
  - Contour maps
  - Subdivision layouts
- Schematics
- Mechanical drawings
- Architectural drawings
- Business management presentations and reports

Initial Inspection and Accessories Inventory

The individual parts of your plotter were thoroughly inspected before the unit was shipped, and the instrument should be in good operating order. Carefully inspect the plotter and accessories for any physical damage sustained in transit. Notify the nearest HP Sales and Support Office and file a claim with the carrier if the unit is received in a damaged condition.

Refer to the table of Accessories Supplied and check to ensure that you have received all of the items that should accompany the plotter. If you have any difficulties with the plotter, if it is not operating properly, or if accessories are missing, contact the nearest HP Sales and Support Office.

Accessories Supplied

The following items are supplied with each plotter.

Item	Qty	Part Number
Interfacing and Programming Manual	1	07580-90034
Operator's Manual	1	07580-90033
Programmer's Reference Card	1	07580-90035
Drafting Supplies Catalog	1	
USA only		5957-4118(D)
International		5957-9419
Pen Carousels		
Fiber-tip carousel	1	07580-60035
Roller-ball pen carousel	1	07580-60082
Disposable drafting pen carousel	1	07580-60240
Power Cord (appropriate cord supplied, based on origin of sales order)	1	
Supplies Organizer	1	4040-1953
Pens		
Roller-ball		
Package of 4, 1 each, black, red, green, blue	1	5061-5037
Fiber-tip		
Package of 4, 1 each, black, red, green, blue, 0.7 mm line width	1	5060-6858
Package of 6, 1 each, burnt orange, lime green, gold, turquoise, violet, brown, 0.7 mm line width	1	5060-6895
Drafting (disposable)		
Package of 4, 1 each, black, red, green, blue	1	5061-7566
Package of 4, long body adapters and pen cap assemblies	1	5061-7578

Owner's Info

Accessories Supplied (Continued)

Item	Qty	Part Number
Grit Wheel Brush	1	8710-1386
Media Sampler*		
For 7580B		
A1 — Metric-size media	1	9280-0524
D — English-size media	1	9280-0523
For 7585B		
A0 — Metric-size media	1	9280-0586
E — English-size media	1	9280-0585
For 7586B		
A0 — Metric-size media	1	9820-0586
E — English-size media	1	9820-0585
914.4 mm (36 in.) roll paper	1	9820-0637
914.4 mm (36 in.) paper spool	1	9300-1069
Male-to-Male Interface Cable RS-232-C/CCITT V.24	1	17355M

*Metric or English media is supplied based on destination of plotter. Each sampler includes 50 sheets of chart paper and 5 sheets of vellum.

Accessories Available

The following table lists additional accessories that you can purchase using the appropriate part number. All roll media is 150 feet (46 metres) long.

An assortment of smaller paper, various pens and liquid-ink pen tips, and other supplies are also available from Hewlett-Packard. Please refer to the HP Drafting Supplies Catalog (5957-4118) or Computer Users Catalog (5953-2450) for a complete listing.

Item	Part Number
Service Manual	
For 7580B	07580-90022
For 7585B	07585-90002
For 7586B	07586-90000
Media	
Chart Paper	
D-size, 200 sheet package	9280-0527
E-size, 150 sheet package	9280-0587
A1-size, 200 sheet package	9280-0528
A0-size, 150 sheet package	9280-0590
609.6 mm (24 in.) roll	9280-0638
914.4 mm (36 in.) roll	9280-0637
Vellum	
D-size, 150 sheet package	9280-0529
E-size, 150 sheet package	9280-0593
A1-size, 150 sheet package	9280-0530
A0-size, 150 sheet package	9280-0594
609.6 mm (24 in.) roll	9280-0635
914.4 mm (36 in.) roll	9280-0636

Accessories Available (Continued)

Item	Part Number
Tracing bond	
D-size, 150 sheet package	9280-0600
E-size, 150 sheet package	9280-0598
A1-size, 150 sheet package	9280-0599
A0-size 150 sheet package	9280-0597
Polyester film double matte	
D-size, 50 sheet package	9280-0526
E-size, 50 sheet package	9280-0591
A1-size, 50 sheet package	9280-0525
A0-size, 50 sheet package	9280-0592
609.6 mm (24 in.) roll	9280-0634
914.4 mm (36 in.) roll	9280-0633
Paper spool	
609.6 mm (24 in. spool)	9300-1068
914.4 mm (36 in. spool)	9300-1069
Writing Accessories Chest, hardwood	07580-60163
Digitizing Sight (angled view)	07585-60191
Pen Caps for Carousels	
For fiber or roller-ball pens, available individually	4320-0361
For drafting pens, available individually	4320-0365
HP Industry Standard Plotting Package (HP-ISPP) Contact your HP Sales and Support Office	

Media and liquid-ink plotting pen tips may also be purchased from your local engineering supply store. Refer to the Operator's Manual for information on suitable pen tips and media.

Input Power Requirements

WARNING

To prevent operator injury or damage to the plotter, verify that the line voltage setting is correct **BEFORE** connecting the line power. Also ensure the power cord is connected to a line power outlet that is provided with a protective earth ground contact.

Power Options

The plotter can be configured to operate with any of the following power sources.

Line Voltage:	100 V ~ +5%, −10% 120 V ~ +5%, −10% 220 V ~ +5%, −10% 240 V ~ +5%, −10%
Line Frequency:	48 to 66 Hz, single phase
Maximum Line Current:	2.4 A @ 100 V 2.0 A @ 120 V 1.1 A @ 220 V 1.0 A @ 240 V
Power Consumption:	182 Watts maximum

Line Voltage Selection

The plotter is shipped from the factory with the line voltage set to the nominal voltage for the area specified as the shipment's destination. The voltage selected for the plotter is identified in the recessed window on the rear panel. The line voltage can be changed by qualified service personnel only. Line voltage selection procedures are contained in the plotter's Service Manual.

Fuse Protection

The plotter is factory equipped with a fuse appropriate to the factory-set line voltage. The fuse should be replaced by qualified service personnel only. Replacement procedures are contained in the plotter's Service Manual.

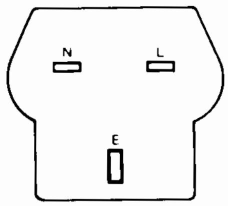
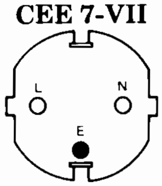
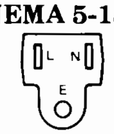
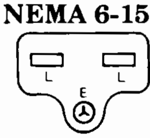
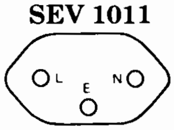
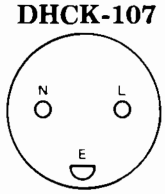
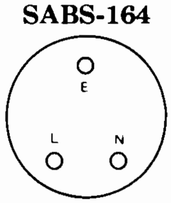
Grounding Requirements

To protect operating personnel, the plotter must be properly grounded. The plotter is equipped with a three-conductor power cable which, when connected to an appropriate power outlet, grounds the plotter. To preserve this protection feature, do not operate the plotter from a line power outlet that has no ground connection.

Power Cords

Power cords with different plugs are available for the plotter. The cord packaged with each plotter depends upon its destination. The power cords supplied by HP have a standard female plug which mates with the power-input socket in the plotter. The polarities of the male plugs shown in the accompanying chart are matched to the line power outlets used in the indicated areas. If the plotter has the wrong power cord for the area, please contact your HP Sales and Support Office.

Power Cord Configurations

Plug	Diagram	Description	Option No.
BS 1363A		HP Part Number 8120-1351; 250 V, 13 A, 1 ϕ plug rating. For use in United Kingdom, Cyprus, Nigeria, Zimbabwe, Singapore.	**900**
AS C112		HP Part Number 8120-1369; 250 V, 10 A, 1 ϕ plug rating. For use in Australia, New Zealand.	**901**
CEE 7-VII		HP Part Number 8120-1689; 250 V, 10/16 A, 1 ϕ plug rating. For use in East and West Europe, Egypt.	**902**
NEMA 5-15P		HP Part Number 8120-1751; 125 V, 15 A, 1 ϕ plug rating. For use in Canada, Japan, Mexico, Philippines, Taiwan, Saudi Arabia, UL approved in United States.	**903**
NEMA 6-15P		HP Part Number 8120-3996; 250 V, 15 A, 1 ϕ plug rating. For use in Canada, UL approved in United States.	**904**
SEV 1011		HP Part Number 8120-2104; 250 V, 10 A, 1 ϕ plug rating. For use in Switzerland.	**906**
DHCK-107		HP Part Number 8120-2956; 250 V, 10 A, 1 ϕ plug rating. For use in Denmark.	**912**
SABS-164		HP Part Number 8120-4211; 250 V, 10 A, 1 ϕ plug rating. For use in India, Republic of South Africa.	**917**

NOTE: All plugs are viewed from connector end.

L = Line or Active Conductor (also called "live" or "hot")
N = Neutral or Identified Conductor
E = Earth or Safety Ground

7580-A-209-2

Operator Maintenance

General Cleaning

WARNING
Disconnect the plotter from the power source prior to performing any maintenance. DO NOT allow water to run onto electrical components and circuits or through openings in the enclosure as this may create a shock hazard

Thorough cleaning should be performed periodically. Cleaning intervals are determined by the type of operation, local air contamination, and climatic conditions. Cleaning procedures should include the following:

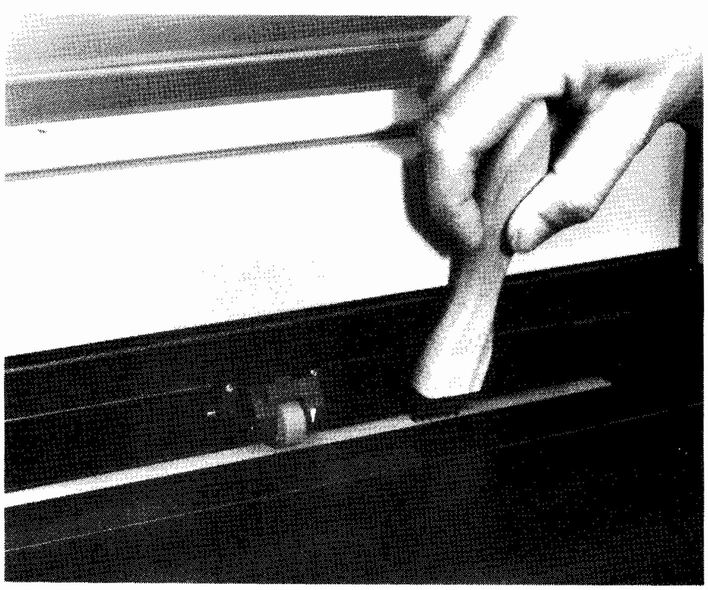
1. Blow away dust accumulation with compressed air if available.
2. Wipe accumulated paper dust from the rubber pinch wheels. Do not use the grit wheel brush to clean the pinch wheels.
3. Clean the outer surface of the plotter with a damp sponge or cloth. Use a mild cleaning solution if necessary. Wipe dry after cleaning.

NOTE: To prevent scratching, do not use abrasive cleansers on the plastic carriage cover or on the outer surface of the plotter. In addition, some "mild" detergents might cause the paint to blister because they contain chemicals that strip water-based paints. For this reason, it is recommended that you use a soft cloth dampened with a 50-50 solution of isopropyl alcohol and water. Then dry with a soft, lint-free cloth. ■

Grit Wheel Cleaning

Cleaning grit wheels is limited to removing dust from between the particles of grit to ensure that paper engagement is not impaired. Remove dust as follows:

1. Disconnect power from the plotter.
2. Raise the carriage cover to gain access to the grit wheels.



7580-A-71-1

CAUTION

Using any brush other than the one supplied with the plotter may damage the grit on the grit wheel.

3. Manually rotate the grit wheels and brush dust from grit surface using the brush supplied with the plotter.

Paper Sensor Cleaning

The front and rear paper sensors should be cleaned periodically and whenever the plotter fails to sense and drops paper. Use a dry cotton swab. It is necessary to bend the swab to clean the rear sensor.

1. Manually move the pen holder to the far right.
2. Raise the cover to gain access to the sensors.
3. Wipe the sensors using the cotton swab.

Care and Use of Pens

Line quality is directly related to pen condition and the pen and media combinations in use, as well as the pen velocity, acceleration, and pen force. Refer to the Operator's Manual, Part No. 07580-90033, for detailed information on pens and media, and optimal settings of velocity and force parameters for a specific pen type.

Notes


LINE
DSR
OUT OF LIMIT
RESET
CLEAR
CHART HOLD
CHART UNLOAD
P1
VIEW
REMOTE
AXIS ALIGN
P2
BY PASS
ROTATE
PEN SPEED

Chapter 2
Plotter Operation

What You'll Learn in This Chapter

In this chapter you will learn about the functions of each control and indicator, the plotter's operating states, and the Cartesian coordinate system. You will also learn how to load pens and paper, how to determine that the plotter is functional, and how to perform all front-panel operations. The plotter's HP-GL language, its syntax, and default conditions are described. An explanation of the conditions upon which the HP-GL examples in subsequent chapters are based is given at the end of this chapter.

HP-GL Instructions Covered

DF The Default Instruction
IN The Initialize Instruction
NR The Not Ready Instruction

Terms You Should Understand

Operating States — Not-Ready, View, and Remote are physical states that determine the plotter's response to any front-panel activity. They establish conditions for (1) loading paper, (2) viewing a complete or in-process graphics plot, and (3) processing HP-GL instructions from the internal buffer.

Configuration Switches — switches that pertain to both interface modes. They determine how close the pen is permitted to come to the physical edges of the paper and how the plotter interprets HP-GL instructions.

Interface Switches — switches that pertain to a specific interface mode. They establish the conditions under which communication between the plotter and computer will occur.

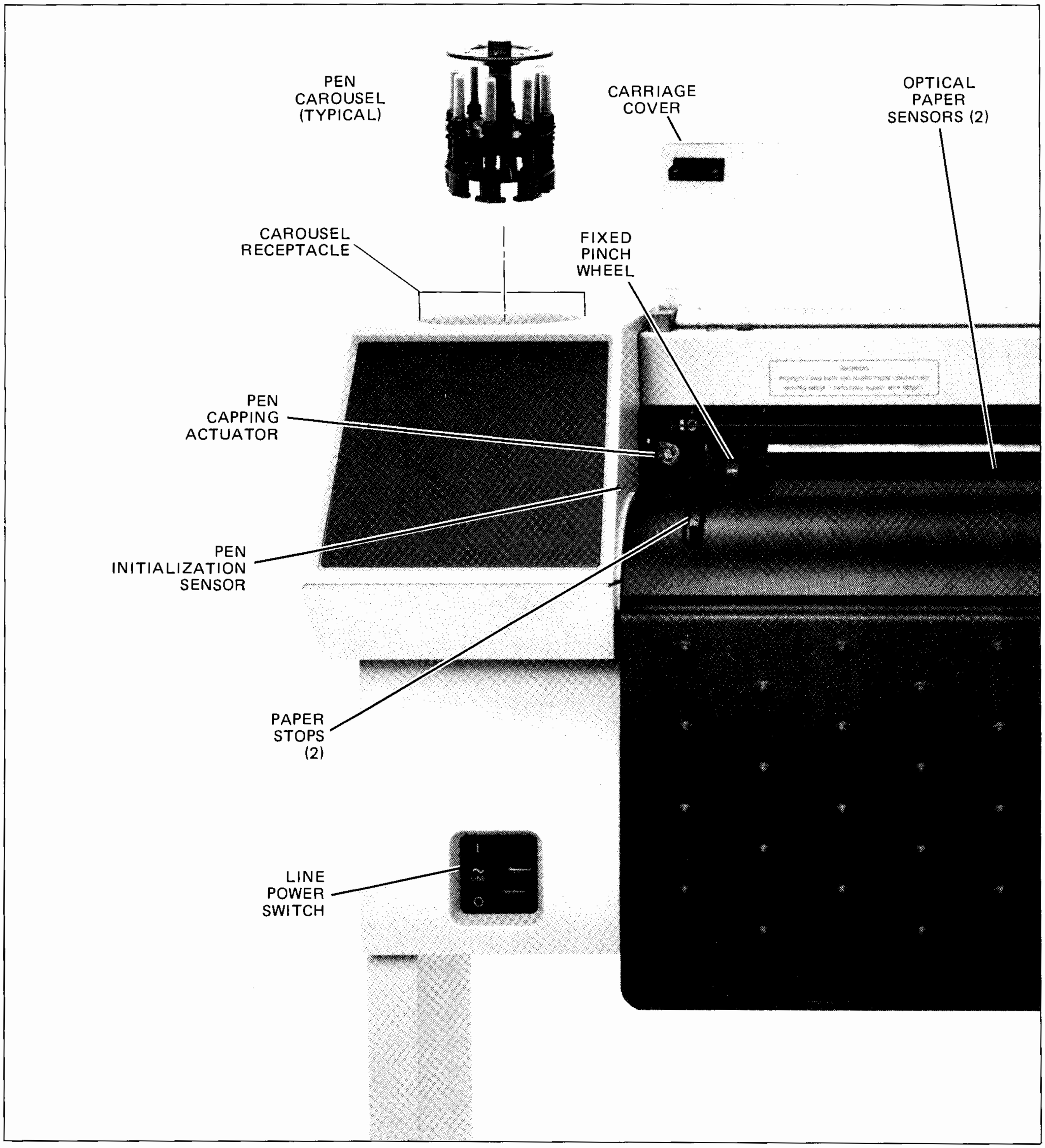
Major Feature Locations

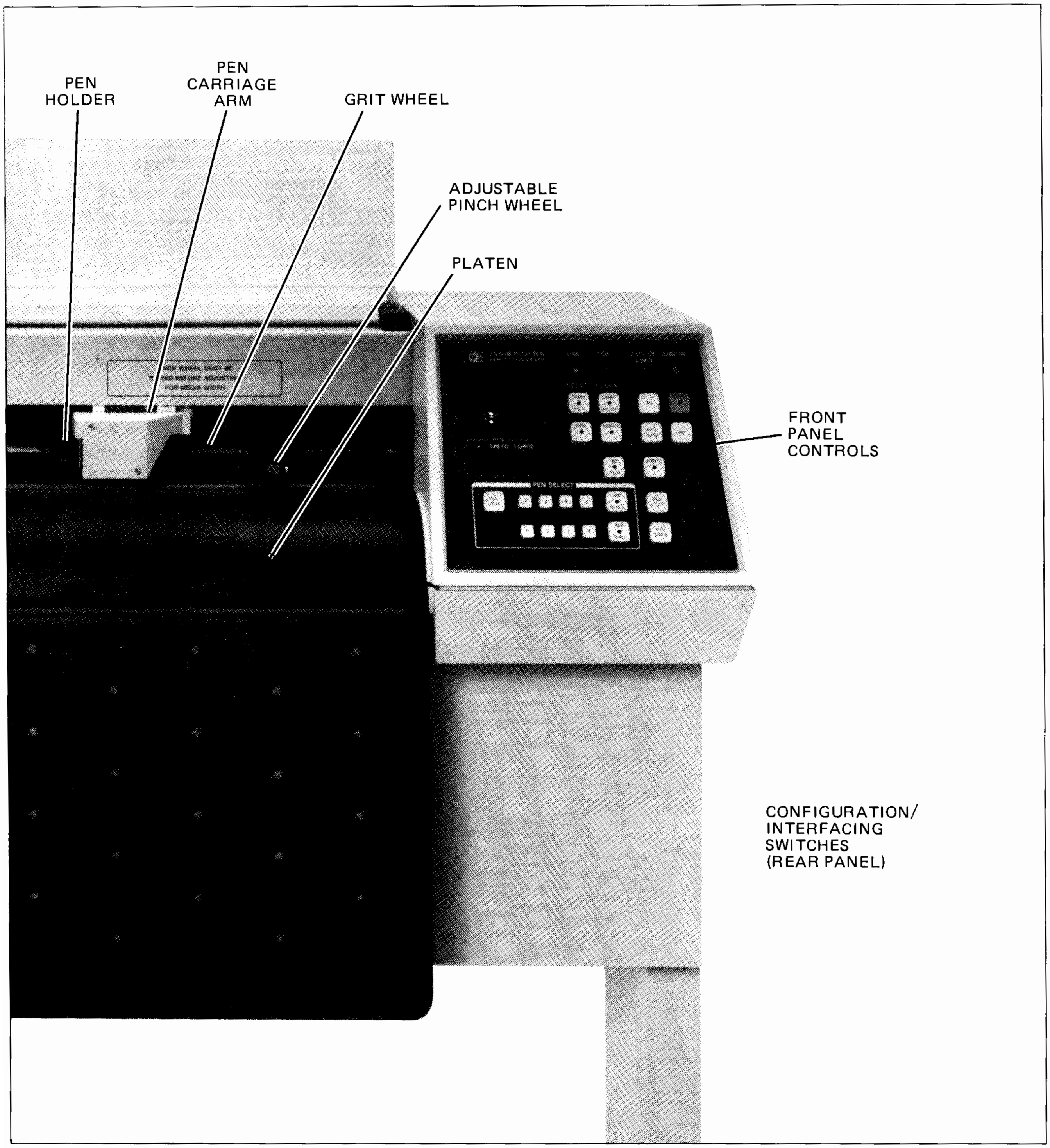
The following illustration shows the locations of the major operating features. The pen holder is shown near the right limit of the drawing range.

The HP 7580 platen is 24.5 inches wide and easily accommodates standard English (ANSI) A through D and metric (ISO) A4 through A1 size media. The HP 7585 and 7586 platen is 36.5 inches wide and can easily accommodate standard English A through E and metric A4 through A0 sizes. The HP 7586 accommodates roll media as well. The paper moves back and forth across the platen, driven by pinch wheels and grit-covered drive wheels. Different paper sizes are accommodated by manually positioning the adjustable pinch wheel at the right-hand paper edge. A vacuum fan holds the paper against the platen.

The rotating pen carousel is located on the left side of the plotter and holds up to eight pens which can be accessed by the pen holder.

The operating controls and indicators are grouped on the front panel at the right side of the plotter. Configuration/interfacing switches are located on the rear panel.




PEN HOLDER
PEN CARRIAGE ARM
GRIT WHEEL
ADJUSTABLE PINCH WHEEL
PLATEN
FRONT PANEL CONTROLS
CONFIGURATION/ INTERFACING SWITCHES (REAR PANEL)

Controls and Indicators and Their Functions

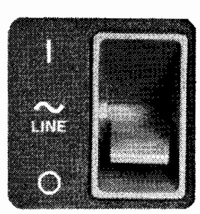
A description of front-panel controls and indicators, including their functions follows. Configuration switches that pertain to both HP-IB and RS-232-C Interface Modes are also described. Refer to Chapters 12 and 13, respectively, for descriptions and the functions of interface switches and indicators which pertain only to HP-IB or RS-232-C Interface Modes.

NOTE: Some operating controls and indicators are nonfunctional in one or more of the three operating states. For restrictions which apply to each control and indicator, refer to the State Diagram and Function Table, Chapter 2.

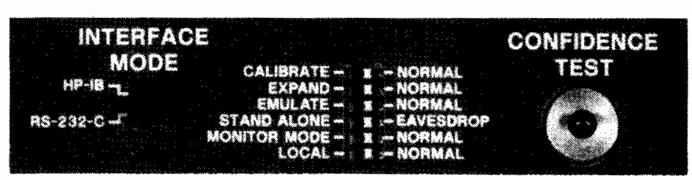
During programmed operation, front-panel actions are recognized when the plotter is ready to process a character. They are acted on one or two vectors later if vectors are currently in progress. For example, if **VIEW** is pressed while a program is running, the **VIEW** LED turns on at the completion of the current vector, but the plotter may not assume the view position until after one or two more vectors are completed. *This delayed response will be evident only if graphics data is stored in the buffer when the front-panel action is initiated.* ■

Controls and Indicators. *(HP 7586B is shown; HP 7580B and 7585B are similar.)*

FRONT PANEL



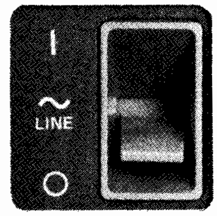
FRONT BASE



REAR PANEL

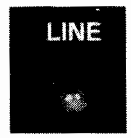
Front Base

~ **ON/OFF** — The LINE switch controls application of ac power to plotter. Power is off when switch is set to 0 and on when set to 1. When power is first applied, the plotter performs a power-up initialization sequence. Refer to Not-Ready State/Power-Up Initialization, Chapter 2.



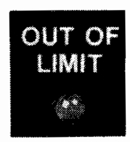
Front Panel

LINE — The LINE light emitting diode (LED) is lit when ac power is applied to the plotter.



DSR (HP 7580 and 7585 only) — The DSR LED pertains only to the RS-232-C Interface Mode. Refer to RS-232-C Controls and Indictors, Chapter 13, for the LED's function.

OUT OF LIMIT — The OUT OF LIMIT LED turns on when the plotter is requested to plot outside the window area or beyond the hard-clip limits. The LED blinks if the commanded position puts the plotter in the "lost" state. Refer to HP-GL instructions IW and PA, Chapters 3 and 4, respectively.



ERROR — The ERROR LED turns on when the plotter detects an I/O error, or an HP-GL error for which the error mask has been set. For a description of errors, refer to the HP-GL instructions IM and OE, Chapter 9, and to the device control instruction ESC . E, Chapter 13.

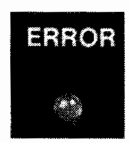
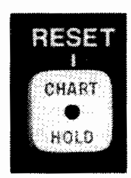
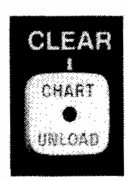


CHART HOLD — Pressing this pushbutton turns on its LED and causes the plotter to lower pinch wheels while remaining in the Not-Ready state. Refer to Operating States, Chapter 2.

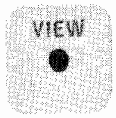


RESET — The RESET function is invoked if this pushbutton is pressed while the ENTER LED is blinking. The RESET function clears the buffer, cancels any pending output from the plotter, resets the parser, and establishes default conditions as defined for the HP-GL instruction IN. The position of the pinch wheels (up or down) does not change.

CHART UNLOAD — Pressing this pushbutton turns on its LED and places the plotter in the Not-Ready state with pinch wheels up to permit unloading of paper. This is the power-on state and otherwise can only be entered from the View state or the Not-Ready state with pinch wheels down. Refer to Operating States, Chapter 2.



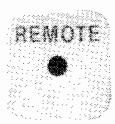
CLEAR — The CLEAR function is invoked if this pushbutton is pressed while the ENTER LED is blinking. The CLEAR function clears the buffer, cancels any pending output from the plotter, and resets the parser.



VIEW — Pressing this pushbutton when paper is loaded turns on its LED and causes the plotter to enter the View state. The paper is fully extended in this state to permit viewing of the entire plotting area. Refer to Operating States, Chapter 2.

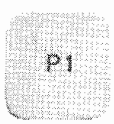
When the plotter is idle, the view function is performed immediately. When the plotter is plotting, the view function is not performed until the plotter completes the current vector. If the plotter is executing an area fill instruction when the **VIEW** button is pressed, the entire area fill is completed before the paper is extended to the view position.

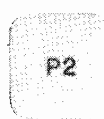
Entry into the View state from the Not-Ready state establishes the hard-clip limits by sensing the paper size, removes any previous rotation due to axis alignment, and defaults the Axis Align point. The initial entry into the View state from the Not-Ready state after power-on or a front-panel reset also sets P1 and P2 to their default locations.



REMOTE — Pressing this pushbutton turns on its LED and causes the plotter to enter the Remote state. This state can be entered from the Not-Ready state or the View state to initiate the processing of HP-GL instructions from the internal buffer. Refer to Operating States, Chapter 2.

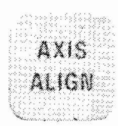
Entry into the Remote state directly from the Not-Ready state causes the plotter to establish the plotter unit coordinate system prior to processing any HP-GL instructions. Refer to The Plotter Coordinate System, Chapter 2.



and


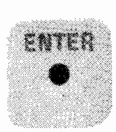
P1/P2 — Pressing **P1** or **P2** causes the plotter to raise the pen and move it to the corresponding physical point on the paper. If P1 or P2 lie off the paper, the pen will move toward that point, stop at the hard-clip boundary, and the **OUT OF LIMIT** LED will come on.

Pressing **P1** or **P2** while the **ENTER** LED is blinking establishes the current pen position as P1 or P2. Refer to The Plotter Coordinate System, Chapter 2.



AXIS ALIGN — Pressing **AXIS ALIGN** causes the plotter to raise the pen and move it to the Axis Align point on the paper.

This pushbutton is used in conjunction with **P1** and **ENTER** to align grids on the paper with the physical axes of the plotter. The Axis Align function rotates the coordinate system about the physical point P1 and has a maximum range of ±6 degrees. Refer to The Plotter Coordinate System, Chapter 2.



ENTER — A multi-purpose pushbutton with an LED which can be off, on, or blinking.

1. When this LED is on but not blinking, the plotter has received the HP-GL instruction DP and is in the digitize mode. If **ENTER** then is pressed, the actual X and Y pen coordinates and the pen status (up or down) are stored and will be sent to the computer upon receipt of the HP-GL instruction OD. Refer to Digitizing, Chapter 8.

2. When the LED is blinking, the plotter is waiting for a second pushbutton to be pressed. Pressing **ENTER** again turns off the LED but causes no other action.

There are several two-button sequences which result in action by the plotter. When a second pushbutton is pressed the LED in **ENTER** goes off.

a.–c. When **ENTER** is pressed followed by **P1**, **P2**, or **AXIS ALIGN**, the actual pen location at the time the second button is pressed becomes the new P1, P2, or Axis Align point.

d. Pressing **ENTER** followed by **RESET** resets the plotter to its default conditions and resets P1 and P2 and the Axis Align point.

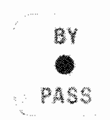
e. Pressing **ENTER** followed by **CLEAR** clears the plotter. See **CLEAR**.

f. Pressing **ENTER** followed by one of the **PEN SELECT** pushbuttons **1–8** causes the plotter to store the pen currently in the pen holder into the selected carousel location if possible. Refer to Pen and Pen Carousel Operation, Chapter 2.

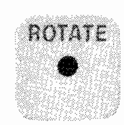
g. Pressing **ENTER** followed by **ROTATE** causes the positions of P1, P2, and the Axis Align point to be rotated counterclockwise 90 degrees about the origin and moved to the margins of the paper. Refer to The Plotter Coordinate System, Chapter 2.

h. (HP 7586 only) Pressing **ENTER** followed by **NEXT PAGE** advances the paper half the normal advance length.

3. **ENTER** is also used at the end of a multi-button sequence to set pen force and speed values from the front panel. The **ENTER** LED is off during these multiple-button sequences. Refer to Front-Panel Operations, Chapter 2.

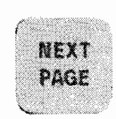


BYPASS — The **BYPASS** pushbutton and LED pertains only to the RS-232-C Interface Mode. Refer to RS-232-C Controls and Indicators, Chapter 13, for the pushbutton function.



ROTATE — Pressing this pushbutton turns on its LED and causes the coordinate system to rotate 90 degrees counterclockwise about the coordinate origin. Depending upon their positions, P1, P2, and the Axis Align point may be rotated outside the hard-clip limits. Pressing **ROTATE** again turns the LED off and cancels this 90-degree rotation.

If the **ENTER** LED is blinking when **ROTATE** is pressed, the positions of P1, P2, and the Axis Align point are rotated 90 degrees and assume their default settings at the margins of the paper. This may result in a change in the size of a user unit. Refer to The Plotter Coordinate System, Chapter 2.

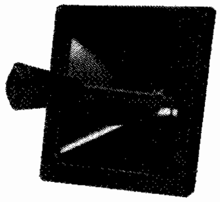


NEXT PAGE (HP 7586 only) — Pressing **NEXT PAGE** with the plotter in the View state or the Remote state advances roll media to the next physical page. The physical page length can be 17, 36, or 48 inches depending on the width of the paper loaded. Pressing **ENTER** and then **NEXT PAGE** advances the paper half the physical page length.

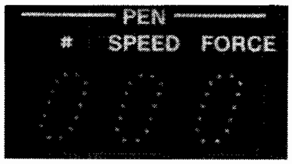
Pressing **NEXT PAGE** when the **CHART HOLD** or **CHART UNLOAD** LED is on will turn on the take-up motor for as long as the button is held down. Do this to wind any slack onto the takeup spool.

NOTE: The plotter will not execute a page advance if sheet media is loaded. ■

Refer to Using Roll Media, Chapter 11.



The joystick moves the pen in the direction of its inclination. In general, pen speed increases as more pressure is exerted on the joystick.



PEN display — The three-digit PEN display shows the current PEN # (zero through 8) and its corresponding SPEED and FORCE in the pen down state. Integers from 1 through 8 are displayed for both SPEED and FORCE values.

SPEED integers 1 through 6 represent 10 times their value in cm/s and integers 7 and 8 represent the maximum plotter velocity of 60 cm/s.

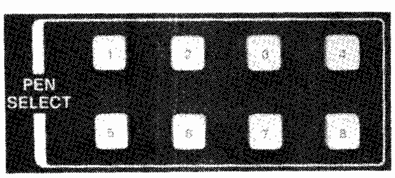
NOTE: Two-digit velocities set via the HP-GL instruction VS cannot be fully displayed on the single-digit SPEED display. Therefore, velocities set by the VS instruction are displayed only to the nearest tens digit of the actual velocity. ■

Integers displayed for FORCE are defined as follows:

1 = 10 grams	5 = 42 grams
2 = 18 grams	6 = 50 grams
3 = 26 grams	7 = 58 grams
4 = 34 grams	8 = 66 grams

The three PEN displays also show any self-test errors that are detected during the power-up initialization sequence or during plotter operation. Refer to The Error Display and The Plotter Self-Test, Chapter 2.

The three PEN displays are on, off, or blinking at various points in the multiple-button sequences for setting new pen speed and force values. For details, refer to Setting Pen Speed/Force, Chapter 2.



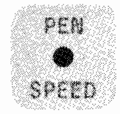
PEN SELECT — Pressing any PEN SELECT pushbutton causes the plotter to retrieve that pen from the carousel, if it is present. The plotter stores its current pen, if any, before it retrieves the newly selected pen. The old pen is stored into the carousel position from which it came, or into the lowest-numbered empty position if its original carousel position has since become unavailable.

If the ENTER LED is blinking when one of the PEN SELECT buttons is pressed, the plotter stores its current pen into the selected carousel position, if possible. If the selected carousel position is occupied, the plotter stores its current pen into the lowest-numbered empty position, if any. If none of the above actions is possible, the command is ignored. Refer to Pen and Pen Carousel Operation, Chapter 2.

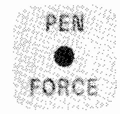
The PEN SELECT buttons are also used in the multiple-button sequences for setting new pen speed and force values. For details, refer to Setting Pen Speed/Force, Chapter 2.

ALL PENS — This pushbutton is pressed as part of the multiple-button sequence for setting new pen speed and force values on all pens. For details, refer to Setting Pen Speed/Force, Chapter 2.

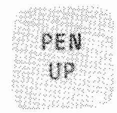
PEN SPEED — Pressing this pushbutton causes its LED to turn on and the **PEN #** display to blink. This initiates a multiple-button sequence for setting a new pen speed value for any one pen or for all pens. For details, refer to Setting Pen Speed, Chapter 2.



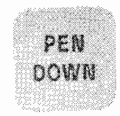
PEN FORCE — Pressing this pushbutton causes its LED to turn on and the **PEN #** display to blink. This initiates a multiple-button sequence for setting a new pen force value for any one pen or for all pens. For details, refer to Setting Pen Force, Chapter 2.



PEN UP/PEN DOWN — These pushbuttons raise or lower the pen. **PEN DOWN** can be used in conjunction with the joystick to draw lines, or to digitize a point.



and

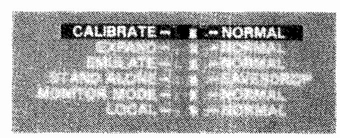


Rear Panel

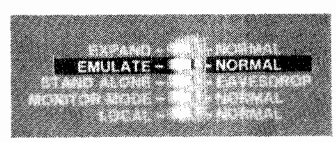
CONFIDENCE TEST (HP 7580/7585 only) — A momentary-contact pushbutton switch that initiates the confidence test. Refer to The Confidence Test, Chapter 2.

CONFIDENCE TEST (HP 7586 only) — Pressing the **CONFIDENCE TEST** button when the plotter is in the Not-Ready operating state with medium loaded and pinch wheels down causes the plotter to either draw a test plot or initiate accuracy calibration. The test plot is drawn if the **CALIBRATE/NORMAL** switch is set to **NORMAL.** Accuracy calibration is initiated if the switch is set to **CALIBRATE.** Refer to the Confidence Test, Chapter 2.

CALIBRATE/NORMAL (HP 7586 only) — This switch determines the function that is performed when the **CONFIDENCE TEST** button is pressed. The position of this switch is sensed only when **CHART HOLD** is pressed. Refer to The Confidence Test, Chapter 2, or Recalibration of the HP 7586 Plotter, Chapter 11.

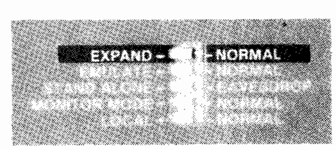


EMULATE/NORMAL — A two-position switch that is sensed each time paper is loaded. It is usually placed in the **NORMAL** position. In the **EMULATE** position, the interpretation of the following HP-GL instructions is changed slightly: DF, DR, IN, IW, OC, OD, OP, SI, SR, and UC. Refer to the descriptions of the above instructions for details.



NOTE: To take full advantage of its design features, the plotter should be used with this switch in the **NORMAL** position, except when backwards compatibility with software created for an HP 9872 plotter is needed. ■

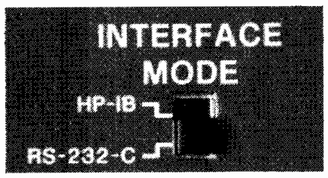
EXPAND/NORMAL (Sheet Media) — A two-position switch that determines how close the pen is permitted to come to the physical edges of the paper. In the recommended **NORMAL** position, the pen is permitted to come to within approximately 15 mm of three edges of the paper, and to within 39 mm of the rear edge that is clamped by both pinch wheels. In the **EXPAND** position, the plotting area is expanded to allow the pen to come to within 5 mm of the three edges and to within 29 mm of the fourth edge,



which may allow the pinch wheels to roll over plotted lines. The position of this switch is sensed each time paper is loaded. Refer to The Plotter Coordinate System, Chapter 2.

EXPAND/NORMAL (Roll Media) — This switch determines the margin widths for each physical page size (refer to the **NEXT PAGE** functional description). In the **NORMAL** position, all four margin widths are 15 mm. In the **EXPAND** position, all four margin widths are reduced to 5 mm. The position of this switch is sensed only when **CHART HOLD** is pressed.

NOTE: Expand mode is not recommended for use with roll media. The media can pass within a 3-mm band about the pinch wheel, possibly causing the pen to draw off the page. ■



INTERFACE MODE — A two-position switch that is sensed only at power-on. It selects either the HP-IB Interface Mode or the RS-232-C Interface Mode. Selecting either mode turns on its associated LED. Refer to Chapters 12 and 13, respectively, for a complete description of the HP-IB and RS-232-C Interface Modes.

Operating States

The plotter has three operating states: Not-Ready, View, and Remote. These operating states establish conditions for (1) loading paper, (2) viewing a complete or in-process graphics plot, and (3) processing HP-GL instructions from the internal buffer. The **CHART UNLOAD, CHART HOLD, VIEW,** and **REMOTE** pushbuttons are used to invoke changes in operating states. The View state can also be entered programmatically from the Remote state by executing the HP-GL instruction NR. The **CHART UNLOAD** LED indicates the plotter is in the Not-Ready state with the pinch wheels up. The **CHART HOLD** LED indicates the plotter is in the Not-Ready state with the pinch wheels down. The **VIEW** and **REMOTE** LEDs indicate the plotter is in the View or Remote state, respectively. The operating states are mutually exclusive, so only one of the above LEDs can be on at any time. Also, the buttons must be pressed in the correct sequence to invoke a desired state change. The permitted sequences are defined by the function table and state diagram on the following pages.

The function table describes the plotter's response to any front-panel activity in each operating state when single sheet media is loaded.

Function Table

	Front-Panel Functions	Not-Ready	View	Remote
1	Chart Hold	A1	NOP	NOP
2	Chart Unload	A2	B	NOP
3	Reset	C	C	C
4	Clear	K	K	K
5	Rotate (90°)/Unrotate	NOP	Yes	Yes
6	Set P1, Axis Align, P2	NOP	Yes	Yes
7	(Go To) P1, P2, or Axis Align	D	Yes	Yes
8	(Go To) View	E1	E2	E2
9	(Go To) Remote	F1	F2	NOP
10	Pen Up or Down	NOP	G	G
11	Position Pen (Joystick)	H	Yes	Yes
12	Enter	I	I	J
13	Select/Store Pen	Yes	Yes	Yes
14	Set Force or Speed	Yes	Yes	Yes
15	Initialize Carousel	Yes	Yes	Yes
16	Bypass	L	L	L
17	Page Advance	M	Yes	Yes
18	Half Page Advance	M	Yes	Yes

A1 — Lower the pinch wheels and find the Y-axis paper dimension. The pinch wheels will only stay down if paper is loaded and the white scribe mark is within the acceptable range over a grit wheel.

A2 — Raise the pinch wheels, thus undoing the previous chart hold operation.

B — Move to the view position, store the pen if possible, raise the pinch wheels, and enter the Not-Ready state.

C — Reset plotter and perform all functions listed under K, plus perform the equivalent of HP-GL instruction IN.

D — If the pinch wheels are down, find the X-axis paper edges, move to the indicated point, and go to View state. If pinch wheels are up, do nothing.

E1 — If the pinch wheels are down, find the X-axis paper edges, move to the view position, and enter the View state. If pinch wheels are up, do nothing.

E2 — Move to the view position and enter the View state if currently in the Remote state.

F1 — If the pinch wheels are down, find the X-axis paper edges, move to the view position, and enter the Remote state. If the pinch wheels are up, do nothing.

F2 — Move to the graphics position and enter the Remote state.

G — Change the monitored and physical pen state as if a pen up or pen down command had been sent with the HP-GL instruction PU or PD.

H — In this state, pen can only be moved horizontally, using joystick.

I — LED off, **ENTER** may be pressed as the first button of seven two-button sequences to set P1, P2, or Axis Align point; store a pen; rotate and default P1, P2, and Axis Align points; and, acting like a shift key with Chart Unload and Chart Load, to **CLEAR** or **RESET** the plotter. LED blinks until a second button is pressed. LED off, **ENTER** may also be pressed as last of a multi-button sequence to set pen speed or force for one or all pens.

J — Functions as in I, plus, when the plotter is in digitize mode and the LED is on steady, pressing the button prepares the X,Y pen position for output by OD command.

K — Clear the buffer, cancel pending I/O from plotter, and reset parser.

L — Functional only if RS-232-C Interface Mode is selected and plotter is operating in the eavesdrop environment. Toggles plotter between programmed-off and programmed-on in either normal or local operating mode.

M — Turn on take-up motor and wind slack while button is held down.

NOP — No Operation.

The state diagram graphically summarizes the three operating states, the button sequences that cause a state transition, and the actual pen, pinch wheel, and paper motion that occurs upon each transition.

Detailed explanations of the motion's end functions that occur upon entry into each operating state are given in the following paragraphs. These paragraphs track the flow sequence shown in the state diagram. Each paragraph assumes that the conditions described in the previous paragraph are established.

Not-Ready State/ Power-Up Initialization

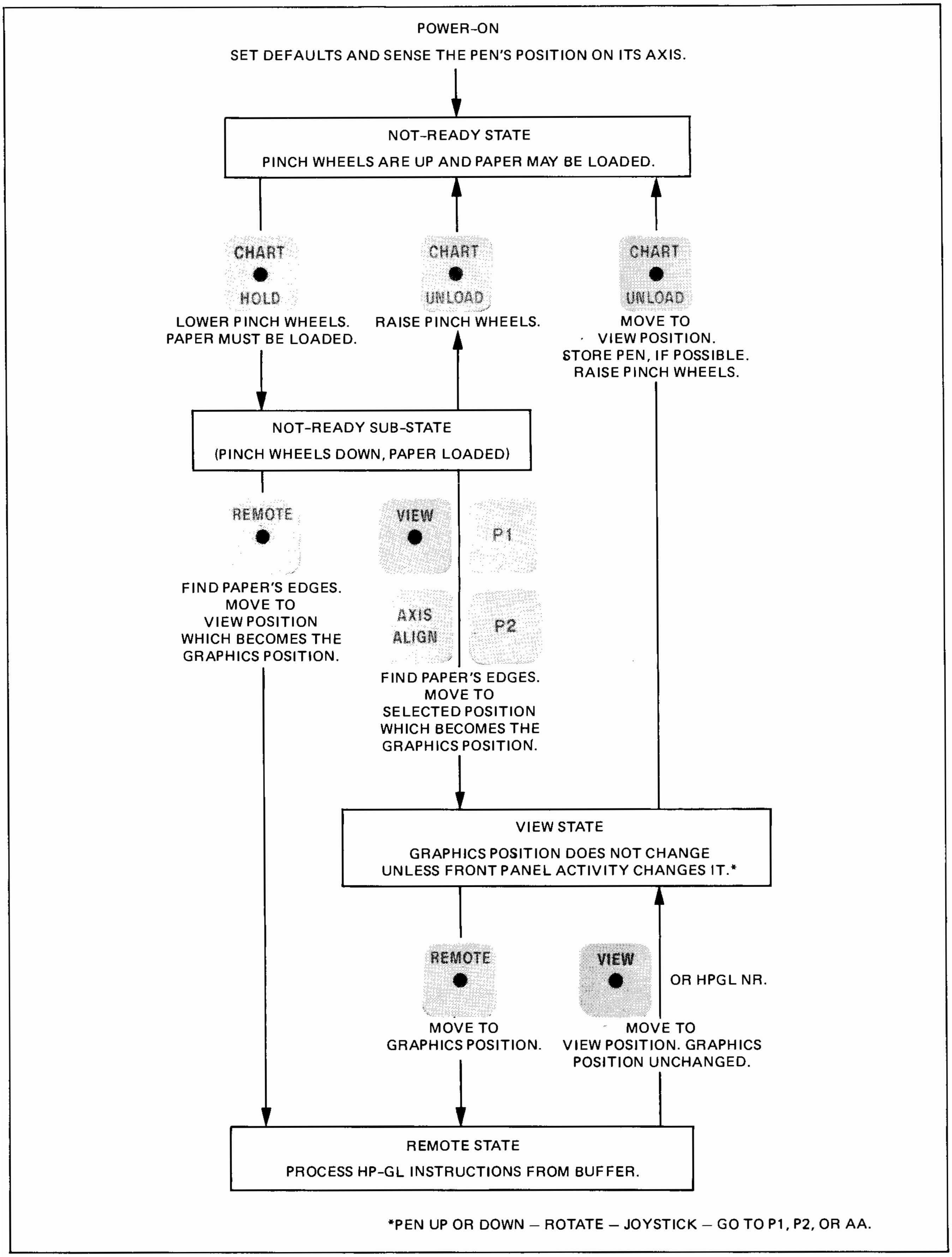
Upon power-up, the plotter performs the following initialization sequence and enters the Not-Ready state with the pinch wheels up.

1. When the **LINE** switch is set to **1** (on), all front-panel LEDs turn on momentarily as part of the plotter's automatic self-test. If, during this self-test, a malfunction is detected, the letter E and a two-digit error number are shown in the **PEN** displays, and the initialization sequence is stopped. Otherwise, the plotter proceeds in its initialization sequence. Refer to The Plotter Self-Test, Chapter 2.

2. The plotter sets default conditions as defined for the HP-GL instruction IN.

3. The pen holder moves approximately two inches to the right. Then the pen is moved to the pen initialization sensor at the left end of the pen carriage arm. This sequence establishes a Y-axis starting position and determines if a pen is in the pen holder. All LEDs except **LINE**, **DSR**, and **CHART UNLOAD** are turned off during this sequence.

   NOTE: The **BYPASS** LED will also remain on if RS-232-C Interface Mode and eavesdrop operating environment are selected at power-on. ■


POWER-ON
SET DEFAULTS AND SENSE THE PEN'S POSITION ON ITS AXIS.
NOT-READY STATE
PINCH WHEELS ARE UP AND PAPER MAY BE LOADED.
CHART HOLD
LOWER PINCH WHEELS. PAPER MUST BE LOADED.
CHART UNLOAD
RAISE PINCH WHEELS.
CHART UNLOAD
MOVE TO VIEW POSITION. STORE PEN, IF POSSIBLE. RAISE PINCH WHEELS.
NOT-READY SUB-STATE
(PINCH WHEELS DOWN, PAPER LOADED)
REMOTE
FIND PAPER'S EDGES. MOVE TO VIEW POSITION WHICH BECOMES THE GRAPHICS POSITION.
VIEW
P1
AXIS ALIGN
P2
FIND PAPER'S EDGES. MOVE TO SELECTED POSITION WHICH BECOMES THE GRAPHICS POSITION.
VIEW STATE
GRAPHICS POSITION DOES NOT CHANGE UNLESS FRONT PANEL ACTIVITY CHANGES IT.*
REMOTE
MOVE TO GRAPHICS POSITION.
VIEW
OR HPGL NR.
MOVE TO VIEW POSITION. GRAPHICS POSITION UNCHANGED.
REMOTE STATE
PROCESS HP-GL INSTRUCTIONS FROM BUFFER.
*PEN UP OR DOWN – ROTATE – JOYSTICK – GO TO P1, P2, OR AA.

4. The pen carousel is initialized if a carousel is presently loaded (refer to Carousel Initialization, Chapter 2). If a pen carousel is not loaded, or if the carousel's type cannot be determined by the plotter, default values for pen force, speed, and acceleration are invoked as though a drafting pen carousel were loaded.

   NOTE: After pen carousel initialization is completed, the **DSR** LED assumes its logical state: on if the Data Set Ready line is high; off if the Data Set Ready line is low. ■

The plotter is now locked in the Not-Ready state and cannot leave this state until paper, or other media, is loaded and **CHART HOLD** is pressed to initiate the following sequence (refer to Loading Paper, Chapter 2):

1. To load paper successfully, paper must be positioned against the front and rear paper-stops while the white arrow on the moveable right-hand pinch wheel is aligned with the right paper edge. When **CHART HOLD** is pressed, its LED comes on and the pen holder moves to a position immediately over the front optical paper sensor.
2. The pinch wheels are lowered and the paper-stops are retracted. If paper has not been loaded, the pinch wheels will lift and the paper-stops will return to their original position.
3. After the pinch wheels are lowered, the pen holder moves to the right until it locates the moveable right-hand pinch wheel. This sequence establishes the Y-axis paper dimension. On the HP 7585 and 7586, if the white arrow is not positioned within one of its acceptable ranges, the pinch wheels will lift and the paper-stops will return to their original position.
4. When the moveable pinch wheel is located, the pen holder moves to a position immediately over the front optical paper sensor.

The plotter can now enter either the View or Remote state and can subsequently be toggled between these two states using **VIEW** and **REMOTE.**

View State

When the **CHART HOLD** LED is on, the View state can be entered by pressing **VIEW** or by pressing **P1, P2,** or **AXIS ALIGN.** Pressing any of these buttons will cause the same motions to occur, except that the final pen location which establishes the new graphics position, will correspond to the button used. The graphics position is the position which would be returned by the HP-GL instruction OC and is the programmed pen position.

Entering the View state from the Not-Ready state with the pinch wheels down initiates finding the X-axis paper edges and establishes the plotter unit coordinate system (refer to The Plotter Coordinate System, Chapter 2). The following sequence describes the motions that occur when the View state is entered after a **CHART HOLD** is invoked.

1. Pressing **VIEW, P1, P2, or AXIS ALIGN** causes the **VIEW** LED to come on.
2. Next, the paper retracts until the front optical paper sensor finds the front paper edge. Then, the paper is extended until the rear optical paper sensor finds the rear paper edge.
3. Finally, the pen moves to the position which corresponds with the button used to initiate this sequence. This position becomes the graphics position.

Remote State

If the Remote state is entered from the View state, the REMOTE LED comes on and the plotter begins executing any HP-GL instructions that are stored in the buffer.

NOTE: Prior to entering the Remote state, HP-GL instructions can be received by the plotter. This data is stored in the buffer to be processed when the Remote state is entered.

Device control instructions are not stored in the buffer and are executed immediately upon receipt in any operating state. ■

If the Remote state is entered directly from the Not-Ready state, the plotter performs the motions to find the X-axis paper edges and establish the plotter unit coordinate system prior to executing any HP-GL instructions. This sequence establishes the view position as the graphics position.

After the Remote state is entered, the only means of unloading paper is to put the plotter into the View state and then into the Not-Ready state. The motions performed by this sequence are as follows:

1. When VIEW is pressed while in the Remote state, its LED comes on, the pen is moved to the view position, and the paper is fully extended.
2. When CHART UNLOAD is pressed, its LED comes on and the pen is stored in the pen carousel, if possible.
3. Next, the pinch wheels are raised and the paper-stops are moved back against the paper. The plotter is in the Not-Ready state.

NOTE: Executing the HP-GL instruction NR programmatically simulates pressing **VIEW**. NR forces the plotter out of the Remote state and is normally used at the end of a program to ensure that a subsequent program does not plot over a completed plot. It is impossible to programmatically return the plotter to the Remote state from the View state once an NR command has been sent. ■

The Plotter Coordinate System

The plotting area should be thought of as a Cartesian coordinate system. When paper has been successfully loaded and entry is made into the View or Remote state, the plotting area becomes a Cartesian coordinate system with plotter units as the units in both the X- and Y-axes. The hard-clip limits (see below) and the coordinate origin are established at this time. The orientation of the X- and Y-axes is established as shown in the diagram which follows, regardless of whether paper is longer in the X- or Y-direction.

Plotter Units

A plotter unit is 0.025 mm (approximately 0.001 in.) in length, and the coordinate origin (0, 0 plotter units) is located at the physical center of the hard-clip limits. The numeric range of plotter units is -2^{29} to $2^{29}-1$, and far greater than the limits of the largest paper which can be used with the plotter. When the plotter is requested to plot data which is within the above range but beyond the limits of the medium, it plots to the medium's edge, monitors the location of the off-scale data points, and resumes plotting when the data again represents points on the medium. The graphics position is defined to be this monitored position regardless of whether it is on or off the medium.

Hard-Clip Limits

Hard-clip limits determine the maximum limits of the pen's motion. The pen cannot plot beyond these limits; the pen can never write off the medium. The hard-clip limits are fixed distances in from the medium's edge and are determined by the position of the **EXPAND/NORMAL** switch at the time paper is loaded.

In the recommended **NORMAL** position, the hard-clip limits for sheet media are approximately 39 mm from the top paper edge and 15 mm from the other edges.

In the **EXPAND** position, the plotting area for sheet media is expanded to allow the pen to come to within 29 mm of the top paper edge and to within 5 mm of the other three edges. If the switch is set to **EXPAND**, the pinch wheels can roll over drawn lines, possibly smearing wet ink. Careful attention to the order and position of line drawing or software checks to avoid such paper motion should be used when the plotter is in the **EXPAND** position. (The **EXPAND** position is *not* recommended for use with roll media on the HP 7586.) In either setting, the paper must have 90-degree corners and be positioned squarely against the paper-stops to prevent the paper from moving out from under one of the pinch wheels.

The Scaling Points P1 and P2 and the Axis Align Point

The default positions of P1 and P2 are set 15 mm inside the hard-clip limits. The default position of the Axis Align point is the intersection of a line parallel to the X-axis that passes through P1 and a line parallel to the Y-axis that passes through P2.

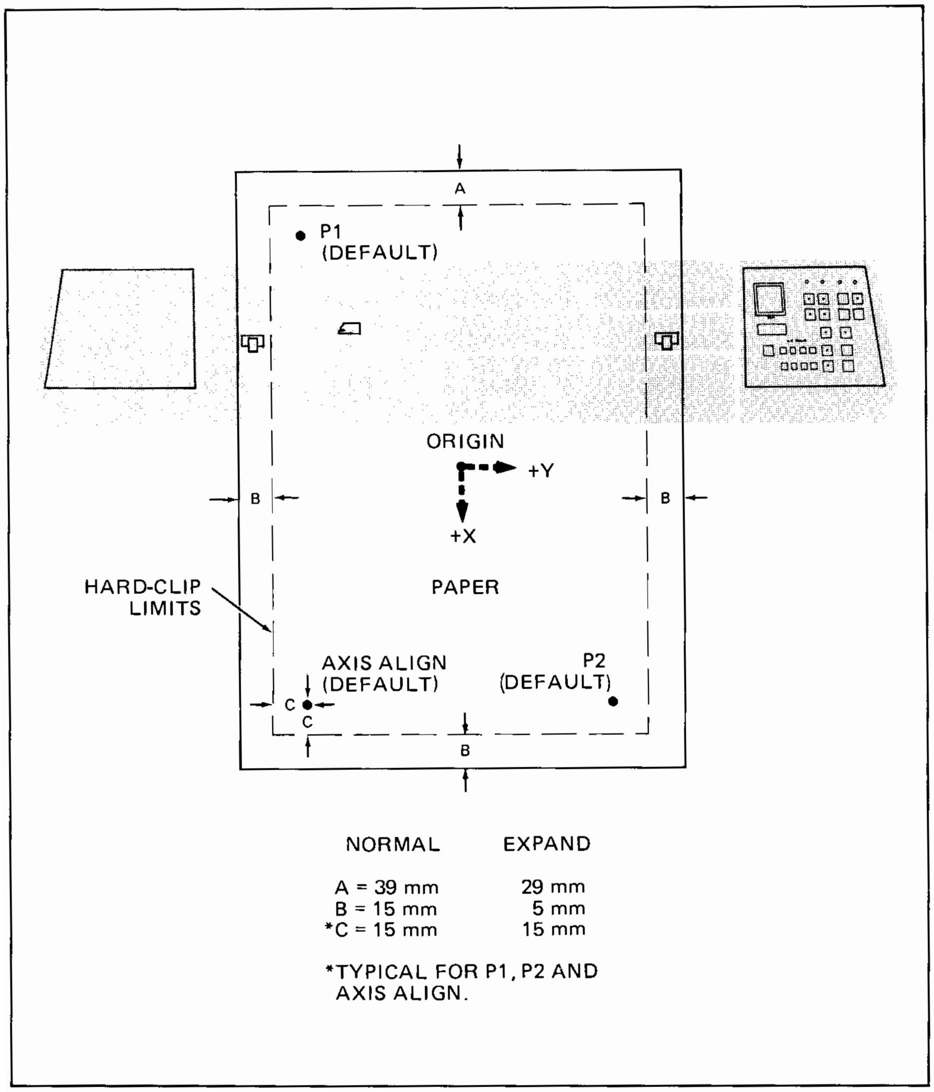
The positions of P1 and P2 are used in conjunction with the HP-GL instruction SC to establish a user-unit coordinate system. (Refer to Scaling, Chapter 3.) P1 and P2 may be changed manually from the front panel or programmatically with the HP-GL instruction IP. (Refer to Setting the Scaling Points, Chapter 3.) Both P2 and the Axis Align point can be moved without changing the position of P1. However, P1 acts as a "handle" for P2 and the Axis Align point in that when P1 moves, P2 and the Axis Align point move so that their distances from P1 do not change. In addition, when the Axis Align point is moved, P2 and the coordinate origin are rotated about P1 such that their distances from P1 and each other do not change. This is shown in the illustration in the following section.

P1, P2, and the Axis Align point can be returned to their default X,Y coordinate values by any of the following methods. These coordinate values are set when the paper limits are established.

NOTE: If an axis alignment has been set, only P1 will return to its default physical position on the paper. The hard-clip limits are still compressed and P2 and the Axis Align point will be rotated from their physical positions by an amount corresponding to the axis alignment. ■

- Execution of the HP-GL instruction IN.
- Execution of the HP-GL instruction IP without parameters.
- Pressing **ENTER RESET** (front-panel reset).
- Pressing **ENTER ROTATE.**

Plotter
Coordinate System
(Default Orientation)



Defaulting the Axis Align point does not cancel any currently existing axis alignment rotation, but merely moves the Axis Align point to the intersection of lines through P1 and P2 parallel to the X- and Y-axis.

The default X- and Y-coordinate values of P1 and P2 depend on the paper size loaded at the time they are established and any skew created by the paper not being positioned against both paper-stops when loaded. Approximate coordinate values of P1 and P2 with selected sheet paper sizes are given in the following tables.

Plotter Operation

Typical HP 7580 Default Coordinate Values for Scaling Points P1 and P2

Paper Size	Normal				Expand			
	$P1_X$	$P1_Y$	$P2_X$	$P2_Y$	$P1_X$	$P1_Y$	$P2_X$	$P2_Y$
A	− 2790	− 4500	2790	4500	− 3190	− 4900	3190	4900
B	− 7100	− 4500	7100	4500	− 7500	− 4900	7500	4900
C	− 9640	− 7530	9640	7530	−10040	− 7930	10040	7930
D	−15710	−10060	15710	10060	−16110	−10460	16110	10460
A4	− 2700	− 4830	2700	4830	− 3090	− 5230	3090	5230
A3	− 6780	− 4830	6780	4830	− 7180	− 5240	7180	5240
A2	−10280	− 7260	10280	7260	−10680	− 7660	10680	7660
A1	−15120	−10760	15120	10760	−15520	−11160	15520	11160
A and A4 sizes are loaded with the longer axis horizontal. All other sizes have long axis vertical.								

Typical HP 7585/7586 Default Coordinate Values for Scaling Points P1 and P2

Paper Size	Normal				Expand			
	$P1_X$	$P1_Y$	$P2_X$	$P2_Y$	$P1_X$	$P1_Y$	$P2_X$	$P2_Y$
A	− 2790	− 4500	2790	4500	− 3190	− 4900	3190	4900
B	− 7100	− 4500	7100	4500	− 7500	− 4900	7500	4900
C	− 7090	−10075	7090	10075	− 7485	−10480	7485	10480
D	−15710	−10060	15710	10060	−16110	−10460	16110	10460
E	−20840	−16180	20840	16180	−21250	−16580	21250	16580
A4	− 2700	− 4830	2700	4830	− 3090	− 5230	3090	5230
A3	− 6780	− 4830	6780	4830	− 7180	− 5240	7180	5240
A2	− 6875	−10785	6875	10785	− 7270	−11180	7270	11180
A1	−15120	−10760	15120	10760	−15520	−11160	15520	11160
AO	−22190	−15740	22190	15740	−22590	−16145	22590	16145
A, C, A4, and A2 sizes are loaded with the longer axis horizontal. All other sizes have long axis vertical.								

Rotating the Coordinate System

The plotter coordinate system can be rotated in two ways: the Axis Align function, and the Rotate function. Both rotations are effective whether plotting is being done in plotter units or in user units.

AXIS ALIGNMENT

Axis alignment is only used to align the coordinate system with the pre-printed axis or grid on the media, and can only by accomplished using front-panel controls. Information regarding the occurrence of this rotation is not available to send to the computer, and it has no effect on the output status byte. (Refer to The Output Status Instruction, OS, Chapter 9.) However, the Axis Align function does compress the hard-clip limits as shown in the following illustration, so the pen can never draw outside of the default hard-clip limits. Using axis alignment, the coordinate system can be rotated ±6 degrees about P1; normally only a few degrees are required for paper grid alignment. Refer to Axis Alignment with Preprinted Grids, Chapter 2, for instructions on setting the Axis Align point. Rotation due to axis alignment can be changed at any time while the plotter is in View or Remote, but it is nullified upon entry into the Not-Ready state (pinch wheels are raised so that paper can be unloaded).

90-DEGREE ROTATION OF AXES

The 90-degree rotate function causes the coordinate system to be rotated 90 degrees about the origin. This rotate function is independent of existing axis alignment. There are four front-panel actions which cause rotation. Each of these four actions can be duplicated programmatically using one or two HP-GL instructions.

First, pressing **ROTATE** when the LED is off turns the LED on and causes 90-degree counterclockwise rotation of the coordinate system. Note the new orientation of the X- and Y-axes as shown in Detail A of the following diagram. Second, pressing **ROTATE** when its LED is on turns the LED off and undoes the previous 90-degree rotation. The X- and Y-axis return to their original power-up directions.

These two front-panel operations are equivalent to the HP-GL instructions RO 90 ; and RO 0 ; respectively.

In both **ROTATE** cases, positions of P1, P2, and the Axis Align point are rotated with the coordinate systems. The numeric values of their coordinates do not change. (The coordinate values of points on the medium do change.)

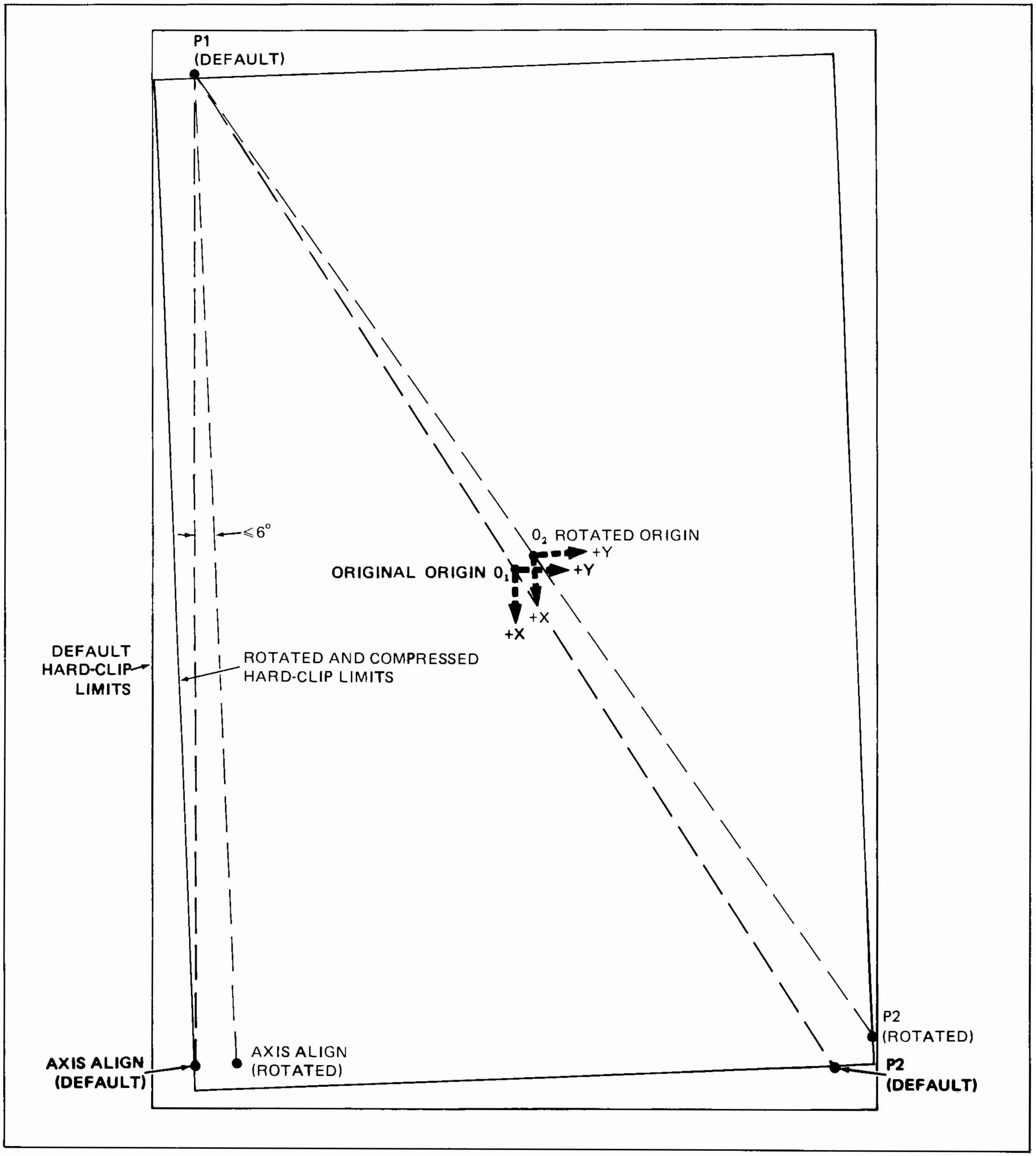
The third and fourth operations are two-button sequences. When **ENTER ROTATE** is pressed when the **ROTATE** LED is off, the coordinate system is rotated in the same manner as when **ROTATE** itself is pressed *and* P1 and P2 assume their default positions 15 mm from the hard-clip limits. (See diagram.) Likewise, pressing **ENTER ROTATE** when the **ROTATE** LED is on undoes the third sequence and moves P1, P2, and the Axis Align point to their default positions.

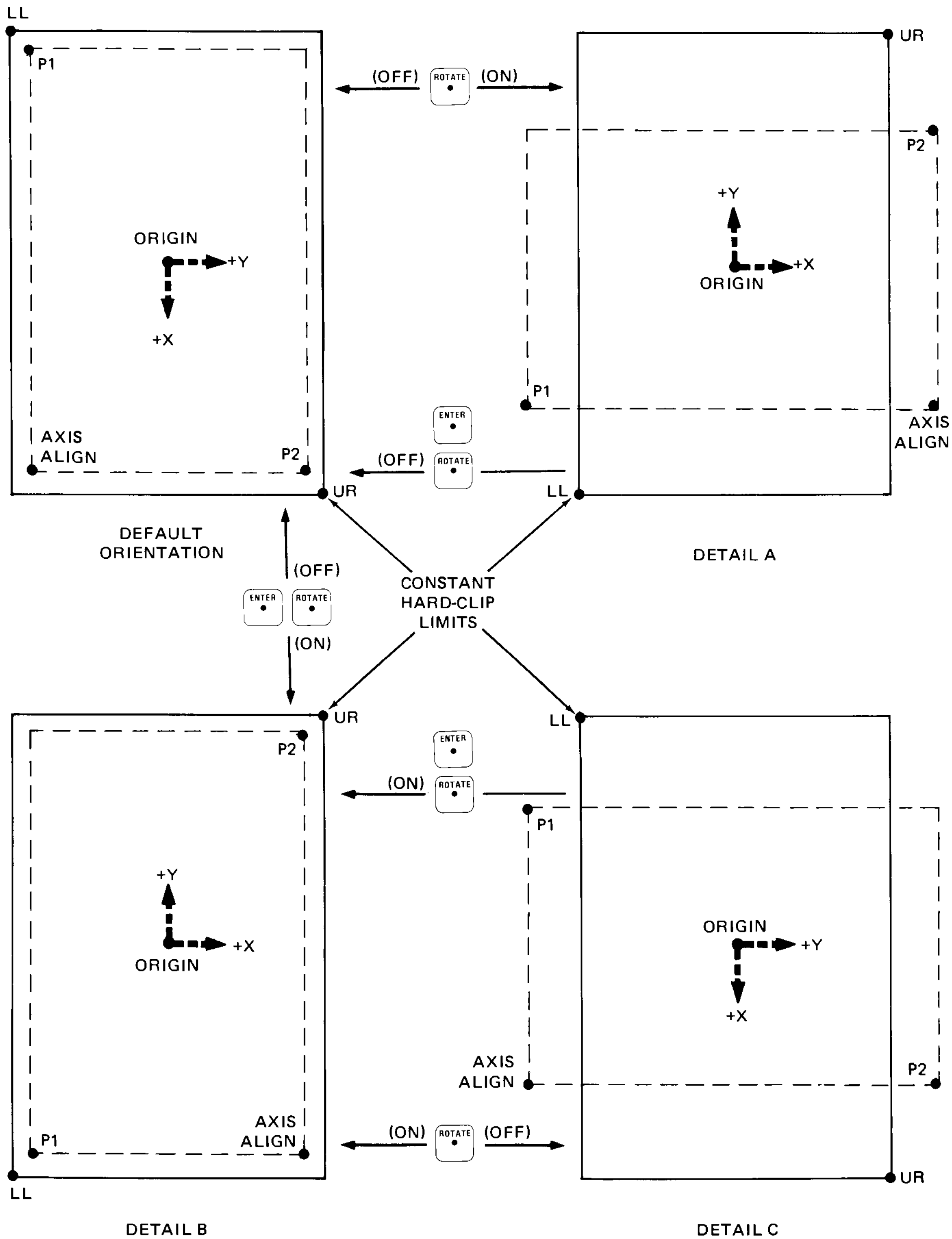
The third and fourth operations are equivalent to sending the HP-GL instructions RO 90 ; IP ; and RO 0 ; IP ; respectively. In both **ENTER ROTATE** cases, axis alignment is maintained, but the Axis Align point is defaulted (becomes the intersection of a line through P1 parallel to the preprinted grid's X-axis and a line through P2 parallel to the preprinted grid's Y-axis). This movement of P1 and P2 will affect plots drawn while an SC instruction with parameters is in effect, and will affect such things as tick length and relative character size which depend on the location of P1 and P2.

Assuming the user does not change P1, P2, and the Axis Align point using HP-GL instructions or front-panel controls, the following diagram shows the effect of pressing **ROTATE** and **ENTER ROTATE** on the coordinate system.

The physical size of the hard-clip limits is not affected by any of the above defined 90-degree rotations. However, the lower-left (LL) and upper-right (UR) orientation of the hard-clip limits is changed by 90 degrees with each rotation, as shown in Details A, B, and C. These orientations change the positions of LL and UR and therefore cause a corresponding change in their coordinate values. The current coordinate values of LL and UR can be obtained by executing the HP-GL instruction OH (refer to The Output Hard-Clip Limits Instruction, OH, Chapter 3). Only if the current paper size is such that the established hard-clip limits form a perfect square, will the coordinate values of LL and UR not change as a result of 90-degree rotation.

Effects of Axis Align Function on Hard-clip Limits





NOTE: (ON) AND (OFF) DEFINES STATE OF ROTATE LAMP AFTER ROTATE BUTTON IS PRESSED.

Pen and Pen Carousel Operation

Introduction

There are three types of pen carousels which can be manually inserted into the plotter. A separate pen carousel is supplied for fiber-tip pens, roller-ball pens, and drafting pens. Each carousel is clearly labeled on top as to the type of pen it houses, and is equipped with pen caps which are optimized for its particular pen type. All pens and the digitizing sight fit into all eight pen positions of each carousel. However, pens stored in a carousel of another type will not be effectively capped and will tend to dry out, unless the caps are changed as described under Customized Pen Loading, Chapter 2.

A carousel can be removed or inserted at any time while the plotter is in any operating state. Removing a carousel invokes no plotter action, but if a pen select or store is requested when no carousel is present, the request is ignored. The plotter senses the presence or absence of a carousel at all times, but only senses the type when it is initialized.

Carousel Initialization

NOTE: Removing the foil reflectors from the lower rim of the pen carousel will interfere with the initialization process. Pens cannot be selected or stored in a carousel from which the reflectors have been removed. If the long reflector or both reflectors are missing, the plotter behaves as if no carousel were present. If the short reflector is missing, the plotter acts as if a drafting pen carousel were loaded. ■

The carousel is initialized when it is inserted into the plotter or at power-on if a carousel is mounted at that time. The carousel initialization process senses and records the type of carousel and the occupied and vacant pen positions within the carousel. This information is used when pen select and store actions are requested. The presence of a pen in the pen holder is not sensed as a part of carousel initialization, but is sensed as part of the pen select or store sequence.

When carousel initialization is required, it becomes the top priority plotter operation. Initialization causes the carousel to rotate at least one complete revolution to determine the occupied and vacant pen positions. Rotation then continues until pen position 1 is presented to the pen holder. Initialization also establishes a set of default pen force, speed, and acceleration values for the carousel if it is of a different type than the previously loaded carousel. If a carousel is replaced with one of the same type, only the stall-occupancy map is updated. The pen parameters for force, speed, and acceleration stored in memory, whether they be default or specified values, are not changed. The default values established by each type carousel are shown in the following table. Any and all of these default values can be changed programmatically or, with the exception of acceleration, manually from the front panel. Default force, speed, and acceleration values apply to all pens in a carousel.

Pen Type	Carousel Type from OT Instruction	Force (grams) act.*	disp.*	Speed (cm/s) act.*	disp.*	Acceleration (g's) act.*
Fiber-tip	1	18	2	50	5	4
Roller-ball	2	50	6	60	6	4
Drafting	3	10	1	15	2	4

*act. = actual; disp. = displayed

Default Pen Values

Pen Number Determination

Upon power-up, the presence of both a pen in the pen holder and a mounted carousel is sensed. The current pen number, which is displayed on the front panel, is determined as follows:

1. If there is no pen in the pen holder, the current pen number is set to zero.
2. If there is a pen in the pen holder but no carousel is mounted, or if a carousel with no vacant pen positions is mounted, the current pen number is set to one.
3. If there is a pen in the pen holder and a carousel with one or more vacant pen positions is mounted, the current pen number is set to correspond to the lowest-numbered vacancy.

After this determination, the current pen number is affected only by programmatic or front-panel pen select and store requests. It is not affected by the front-panel reset function.

Pen Select and Store Algorithm

When a pen select or store is requested either programmatically or from the front panel, the algorithm first detects whether or not a carousel is mounted. If no carousel is mounted, the request is ignored. If a carousel is mounted, the pen map data is accurate and is therefore used in making the decisions of the algorithm. The pen select and store decisions are as follows:

PEN STORAGE

1. If a pen store request is received and there are no vacant positions in the carousel, the request is ignored.
2. If a pen store request is received and the requested position is occupied, the pen is stored in the lowest-numbered vacancy.
3. Otherwise, it is stored in the specified stall.

PEN SELECTION

1. If the requested position is empty, the pen select request is ignored.
2. When the pen select request is received, the plotter stores its current pen, if any, before it retrieves the new selected pen from the requested position. The current pen is stored in accordance with the pen storage rules stated above.

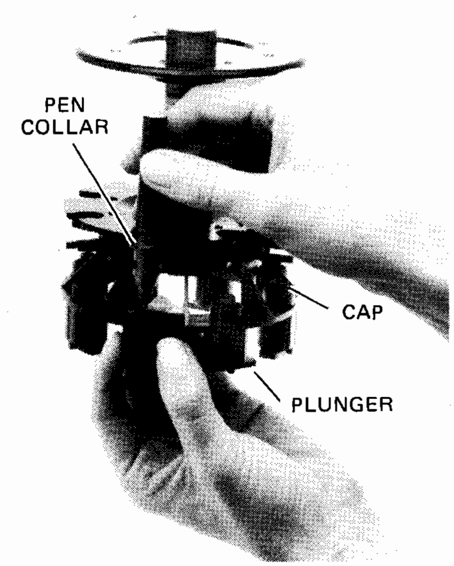
Loading Pens and Carousels

There are two methods of loading pens in the carousels. The first is to use pens of the same type as the carousel; the second is to customize a carousel so that it contains and properly caps a variety of pen types. If the second method is used, default pen speed and force values should be changed to correspond with each pen installed in the customized carousel. Each method is described separately in the following paragraphs.

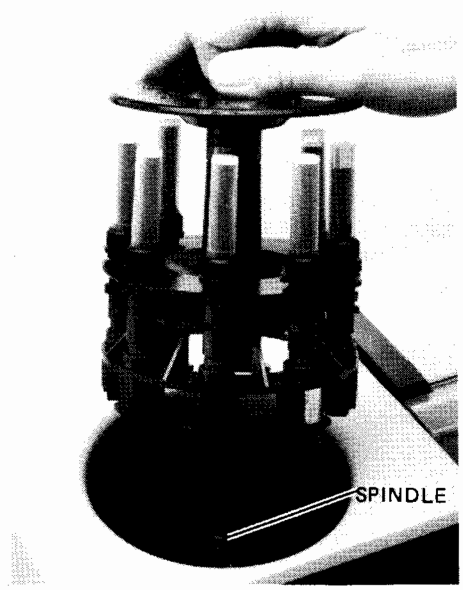
STANDARD PEN LOADING

Load pens of the same type into standardly equipped carousels as follows:

1. Determine the pen type and pen colors to be loaded into each pen position of the carousel.
2. Select the carousel that corresponds to the desired pen type.
3. With the thumb and forefinger of one hand, pull down a plunger, bracing your other fingers against the lower rim of the carousel. Lowering the plunger first in this way helps to prevent damage to the pen's tip.
4. With the other hand, position the pen collar just below the rounded notch and slide the pen straight into the pen holding jaws.
5. Gently release the plunger so that the cap fits over the tip of the pen. Pull the plunger down again and release it to assure that the cap seals tightly without binding.



7580-A-234-1A



7580-A-235-1A

6. Repeat above procedure to load up to eight pens in the carousel.
7. Insert carousel into the plotter and rotate slightly until it dro position on the spindle. No force is required.
8. If power is on, the plotter senses the presence of the carou immediately initializes the carousel. Refer to Carousel Initia Chapter 2.

CUSTOMIZED PEN LOADING

Instructions for customizing a carousel so that different pen types are properly capped when loaded in a single carousel are as follows:

NOTE: To prevent damage to pens, customize the carousel that defaults to the pen force required by the most delicate pen type to be loaded. Using the drafting pen carousel is recommended. ■

1. Determine the pen types to be loaded into each pen position of the carousel. Roller-ball and fiber-tip pens use the same pen caps. If mixing pens on these two types only, proceed to step 6.
2. Select the carousel to be customized and pull the rubber pen cap out of each pen capping plunger that is to be loaded with pens of another type.
3. Remove the required rubber pen caps from the proper carousel.

   Additional pen caps may be ordered through your HP Sales and Support Office. Store the caps in another carousel or some safe place.
4. Press each of these rubber pen caps into the pen capping plungers until they are firmly seated.
5. Load pens and carousel as described under Standard Pen Loading, steps 3. through 8.
6. Set proper pen speed and force values for each pen. Refer to Setting Pen Speed/Force, Chapter 2.

The carousel may also be loaded with pens of the same type and color to extend the effective writing distance beyond the limits of one pen. Refer to The Select Pen Group Instruction, SG, and The Group Pen Instruction, GP, in Chapter 4.

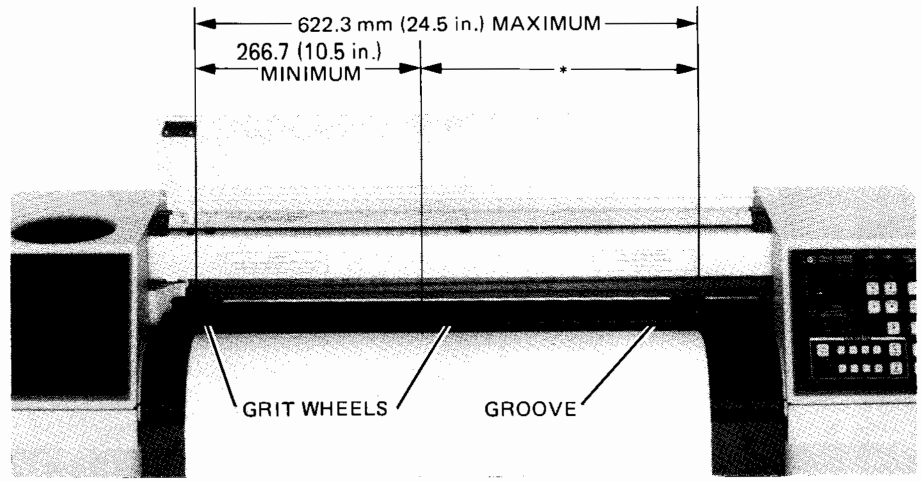
Loading Sheet Paper

NOTE: For loading roll media onto the HP 7586 plotter, see Chapter 11. ■

Before you load a sheet of plotting paper, be sure to select a size that your plotter can accommodate. The HP 7580 plotter accepts both ANSI standard sizes A, B, C, and D, and ISO standard sizes A4, A3, A2, and A1. The HP 7585 and 7586 plotters accept ANSI standard size E and ISO standard size A0 in addition to the above sizes.

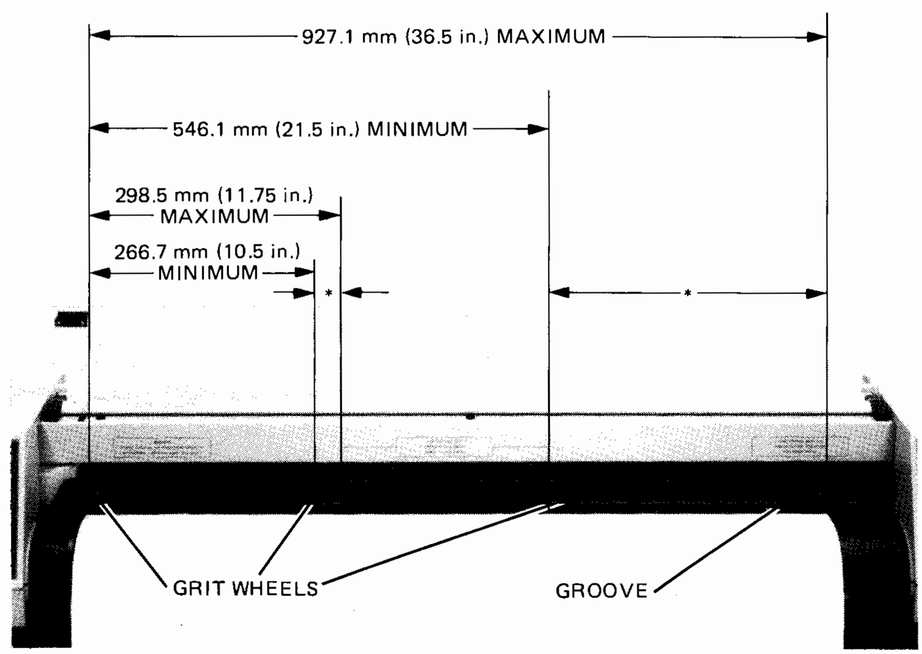
All plotters will accept a slightly larger range of paper sizes than the standard sizes listed above. The most important factor in selecting the paper size is that the right edge of the paper must extend at least 12.7 mm (0.5 in.) over a grit wheel when the left edge of the paper is against the paper stops. The following illustrations show the acceptable paper dimensions. The side of the paper that rolls back and forth across the platen must be at least 203.2 mm (8 in.) long and no longer than 1231.9 mm (48.5 in.).

HP 7580 Acceptable Sheet Paper Dimensions



*Scribe mark on adjustable pinch wheel must align with sheet edge and be within indicated range

HP 7585/7586 Acceptable Sheet Paper Dimensions



*Scribe mark on adjustable pinch wheel must align with sheet edge and be within one of the indicated ranges.

After selecting your paper size, determine whether to load your paper with the long side (length) or the short side (width) along the platen. Remember that the right edge of the paper must extend at least 12.7 mm (0.5 in.) over a grit wheel. This sometimes restricts the way that you can load your paper. The following table shows whether the length or the width may be used for loading standard sized paper.

Acceptable Sheet Paper Loading Directions

Standard Paper Size	HP 7580		HP 7585/7586	
	Width Along Platen	Length Along Platen	Width Along Platen	Length Along Platen
A (8-1/2×11 in.) A4 (210×297 mm)	NO	YES	NO	YES
B (11×17 in.) A3 (297×420 mm)	YES	YES	YES	NO
C (17×22 in.) A2 (420×594 mm)	YES	YES	NO	YES
D (22×34 in.) A1 (594×841 mm)	YES	NO	YES	YES
E (34×44 in.) A0 (841×1189 mm)	—	—	YES	NO

All paper sizes are loaded as follows:

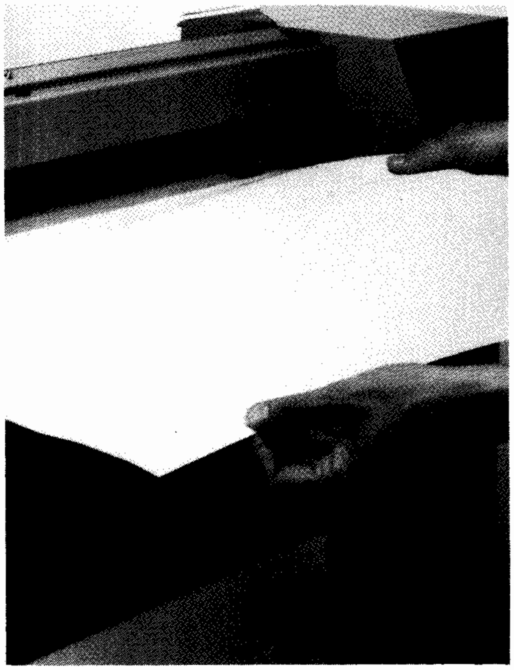
WARNING

To prevent injury, be sure that hair, fingers, and clothing do not become entangled in paper moving mechanism.

1. Set the **LINE** switch to **ON** to place the plotter in the Not-Ready state with the pinch wheels up.
2. Raise the carriage cover.
3. Slide the paper under the pinch wheels until it is approximately centered on the platen as shown in the following illustration.
4. Position the paper so that its left edge is flush against both the front and back paper-stops and hold in this position.
5. Manually position the right pinch wheel to align its scribe mark with the right paper edge.
6. Close the carriage cover and press **CHART HOLD** to lower the pinch wheels. On the HP 7585 and 7586, the pinch wheels will only stay down if the scribe mark is within one of the acceptable ranges over a grit wheel. The plotter will not operate with the carriage cover raised.

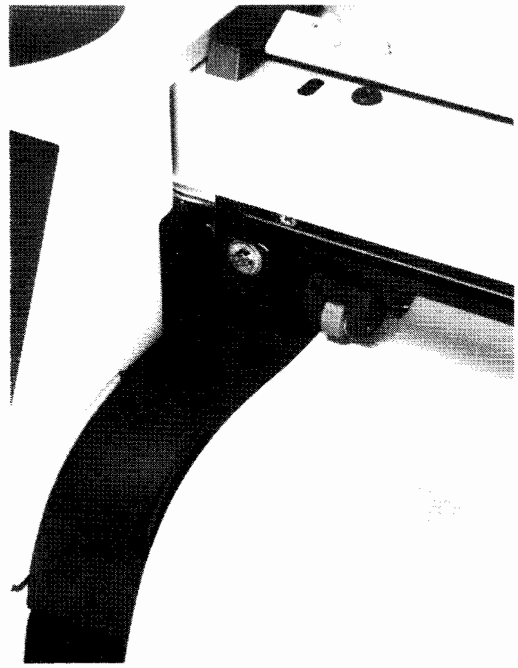
   The plotter can now enter either the View or Remote state and can subsequently be toggled between these two states using **VIEW** and **REMOTE**.

NOTE: Paper cannot be unloaded while in the Remote state. The only means of unloading paper is to first enter the View state and then press **CHART UNLOAD** to raise the pinch wheels. ■



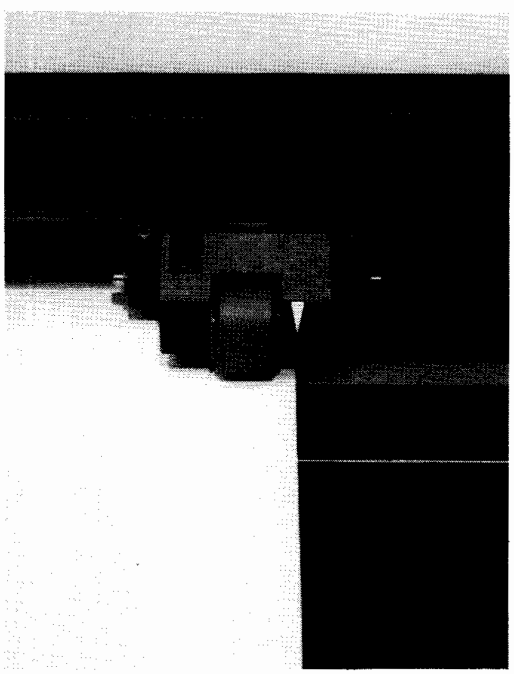
7580-A-101-1

INSERT PAPER
(STEP 3)



7580-A-196-1

POSITION PAPER
(STEP 4)



7580-A-79-1

POSITION PINCH WHEEL
(STEP 5)

7580-A-229-1

LOWER PINCH WHEELS
(STEP 6)

Front-Panel Operations

The procedures given in the following paragraphs define the front-panel button sequences that are used to perform alignment of media with preprinted grids and to set pen speeds and forces. Axis alignment can only be done from the front panel. Each of the other operations can also be executed programmatically.

Axis Alignment with Preprinted Grids

The sole purpose of the Axis Align function is to align grids and axes on preprinted forms with the plotter's coordinate system. (Refer to The Plotter Coordinate System, Chapter 2.) This function can only be invoked in the View or Remote states using the following procedure:

1. Use the digitizing sight for accurate alignment and to avoid pen marks on the media. You may load the sight directly into the pen holder by inserting the pen collar into the groove on the front part of the pen holder.
2. Press **P1** followed by **PEN DOWN**.
3. Using the joystick, position the dot in the digitizing sight directly over one end of a paper grid line which is in the direction of the X-axis.
4. Press **ENTER** followed by **P1** to store the new location of P1.
5. Press **AXIS ALIGN**.
6. Using the joystick, position the digitizing sight directly over the other end of the previously selected paper grid line.
7. Press **ENTER** and **AXIS ALIGN** to store the new location of the Axis Align point.

   NOTE: The plotter will reject the new Axis Align point if the resultant Axis Alignment rotation would exceed ±6 degrees. In this case, the **ENTER** LED would continue to blink because the **AXIS ALIGN** pushbutton has effectively not been pressed. ■
8. Finally, press **P1** to be sure that the pen tracks the paper grid line as the pen moves to P1.
9. Set P2 in a similar manner at the upper-right corner of the grid if scaling of the grid is required in the application.

Setting Pen Speed

The following procedure is used in any operating state to assign different speeds to one or more pens or to set the same speed for all pens. Once set, these speeds remain in effect until new values are entered from the front panel, a VS instruction with parameters is executed, or default values are established by any of the following methods:

- Execution of HP-GL instruction IN,
- Execution of HP-GL instruction VS,
- Pressing **ENTER RESET** (front-panel reset),
- Inserting a different type carousel,
- Power-up initialization.

To set pen speed:

1. Press **PEN SPEED.** Note that its LED comes on, the **PEN FORCE** display is blanked, the the **PEN #** display starts blinking to indicate that the next required action is to either select all pens or one particular pen.

2. Press **ALL PENS** or one of the eight **PEN SELECT** buttons. If **ALL PENS** is used, the speed being entered will apply to all pens. Pressing one of the **PEN SELECT** buttons indicates the speed is to apply only to that particular pen. An uppercase A for all pens or the selected pen number will appear in the **PEN #** display. The **PEN SPEED** display now starts blinking to indicate that the next required action is to select the desired speed.

3. Press one **PEN SELECT** button to select the desired pen speed. Pressing **PEN SELECT** buttons **1** through **6** selects a speed in cm/s equal to 10 times the button number. Pressing **PEN SELECT** buttons **7** or **8** selects the maximum pen speed of 60 cm/s. The selected speed integer (1 through 6) is displayed in the **PEN SPEED** display; however, the pen speed change is not invoked until **ENTER** is pressed.

4. Press **ENTER.**

5. Repeat the above procedure for each pen requiring a different pen speed.

NOTE: Prior to pressing **ENTER,** any front-panel action other than those described in the above procedure will cause the speed setting sequence to be canceled. ■

Setting Pen Force

The following procedure is used in any operating state to assign different forces to one or more pens or to set the same force for all pens. Once set, these forces remain in effect until new values are entered from the front panel, an FS instruction with parameters is executed, or default values are established by any of the following methods:

- Execution of HP-GL instruction IN,
- Execution of HP-GL instruction FS without parameters,
- Pressing **ENTER RESET** (front-panel reset),
- Inserting a different type carousel,
- Power-up initialization.

To set pen force:

1. Press **PEN FORCE.** Note that its LED comes on, the **PEN SPEED** display is blanked, and the **PEN #** display starts blinking to indicate that the next required action is to either select all pens or one particular pen.

2. Press **ALL PENS** or one of the eight **PEN SELECT** buttons. If **ALL PENS** is used, the force being entered will apply to all pens. Pressing one of the **PEN SELECT** buttons indicates the force is to apply only to that particular pen. An uppercase A for all pens or the selected pen number will appear in the **PEN #** display. The **PEN FORCE** display now starts blinking to indicate the the next required action is to select the desired force.

3. Press one **PEN SELECT** button to select the desired pen force. Pressing **PEN SELECT** buttons **1** through **8** selects a force as follows:

1 = 10 grams	5 = 42 grams
2 = 18 grams	6 = 50 grams
3 = 26 grams	7 = 58 grams
4 = 34 grams	8 = 66 grams

   The selected force integer is immediately displayed in the **PEN FORCE** display; however, the pen force change is not invoked until **ENTER** is pressed.

4. Press **ENTER.**

5. Repeat the above procedure for each pen requiring a different pen force.

NOTE: Prior to pressing **ENTER,** any front-panel action other than those described in the above procedure will cause the force setting sequence to be cancelled. ■

Opening the Carriage Cover

The transparent carriage cover can be opened at any time, even while the plotter is in operation. The opening and closing of this cover also opens and closes two switches. When the cover is opened, the plotter finishes its current vector and pending output, if any, and then enters an operator-suspended state. Bit 6 of the extended status word is set to 1, and all plotter activity except loading the buffer is prohibited. For details of the extended status word, refer to the Output Extended Status Instruction, ESC . O, Chapters 12 and 13.

When the carriage cover is again closed, bit 6 of the extended status word is set to 0 and plotter activity resumes as though it had never been suspended.

The Error Display and The Plotter Self-Test

Upon power-up, the plotter performs an automatic self-test to determine, as far as possible, that it is operating correctly. If a malfunction is detected, the letter E and a two-digit error number are shown in the **PEN** displays. Self-testing is also performed and errors are reported during programmed operation and during the confidence test. A table of error numbers, their definitions, and recommended action is given in Appendix C.

The Confidence Test

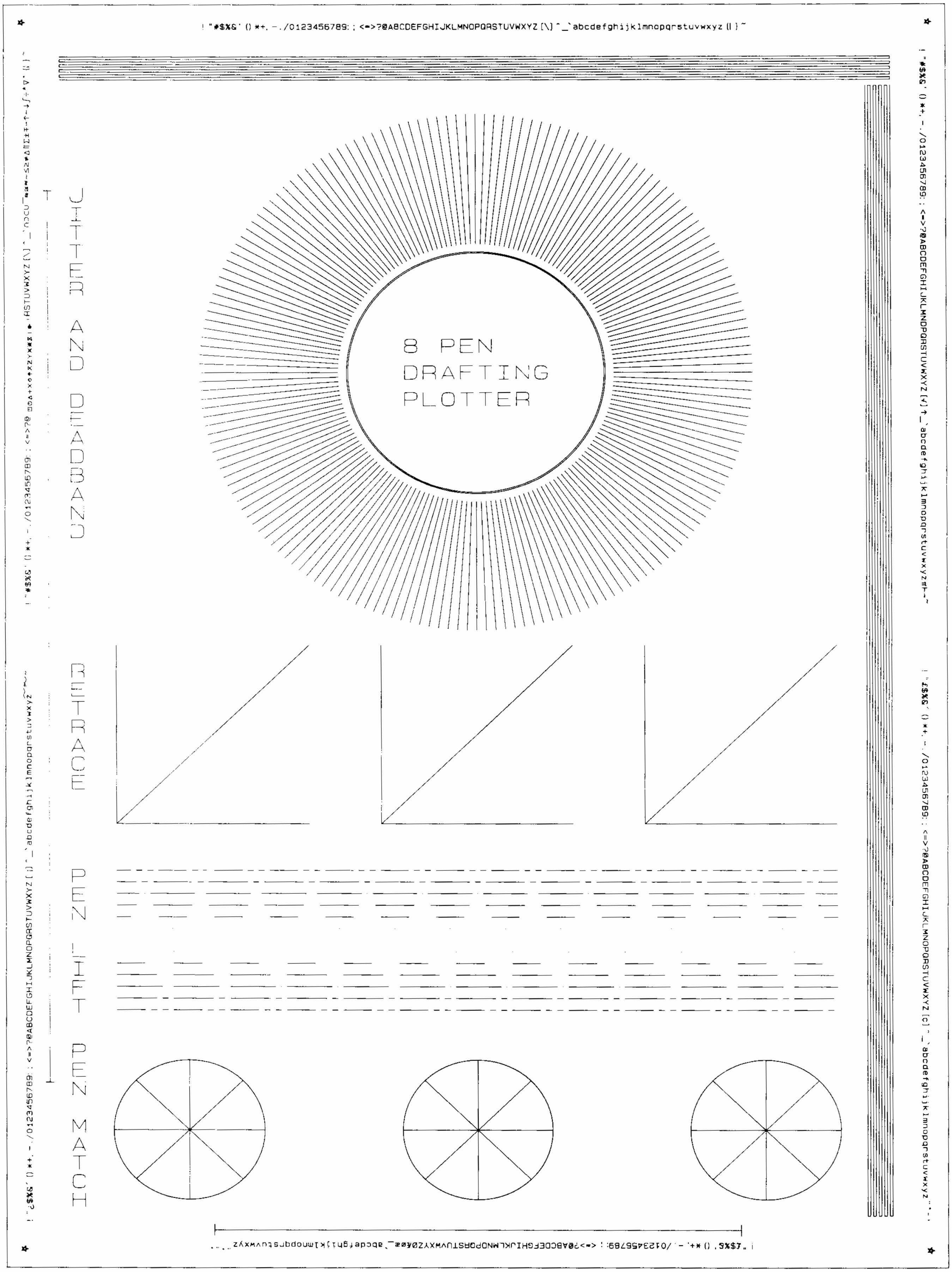
The confidence test draws a test plot which is automatically scaled to fit on the sheet paper loaded at the time the **CONFIDENCE TEST** switch is pressed. The plot is designed to test the mechanical features of the plotter and can be examined to ascertain that the writing mechanism is properly adjusted. If the line quality is poor, check the pens and medium. If new pens and media do not give a satisfactory plot, call your HP Sales and Support Office.

HP 7580 and 7585

The confidence test plot is drawn with the pen speed, force, and acceleration currently in effect. The test will run even if no pen is loaded. To run the test:

1. Turn the plotter on, load paper, and install pens of different colors in stalls 1 through 8 of the appropriate carousel.
2. Press the **CONFIDENCE TEST** switch on the rear panel. The plotter finds the paper edges and draws the plot shown in the next illustration.
3. If an error occurs during the test, the error number is displayed in the **PEN** displays and the test stops.
4. When the plot is completed, the plotter moves to the view position and returns to the Not-Ready state unless **PEN SELECT** button **7** or **8** was the last button pressed during the plot. If **PEN SELECT** button **8** was the last button pressed, the plot will be drawn once and then repeated again and again without the pen. If **7** was the last button pressed, one more confidence test plot will be drawn.

A reduction of the confidence test plot run on D-size paper follows. A horizontal and a vertical line with tick marks at the endpoints extends to the hard-clip limits and no tick marks are drawn when the plot is run on paper smaller than A1 or D size.


8 PEN
DRAFTING
PLOTTER
JITTER AND DEADBAND
RETRACE
PEN LIFT
PEN MATCH

HP 7586

The HP 7586 will run two different confidence tests, depending on the operating mode.

1. *Single-sheet mode or roll-feed mode without automatic take-up* (front take-up spool not installed).

   a. Turn the plotter on, and install pens of different colors in stalls 1 through 8 of the appropriate carousel. Be sure that stall 8 contains a black pen.

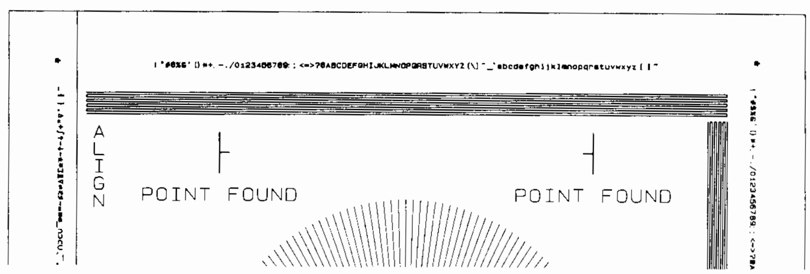
   b. Be sure the **CALIBRATE/NORMAL** switch is set to **NORMAL.** Press the **CONFIDENCE TEST** switch on the rear panel.

   c. If an error occurs during the test, the error number is displayed in the **PEN** displays and the test stops.

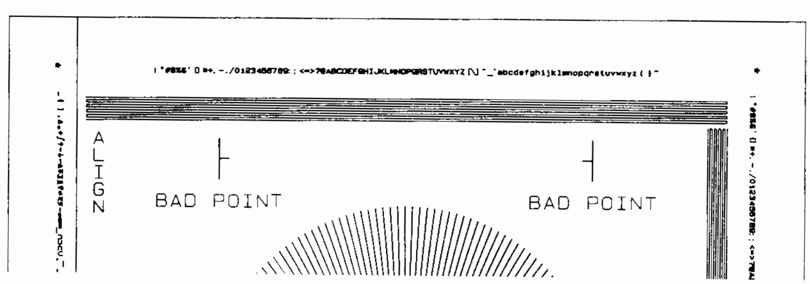
   d. When the plot is completed, the plotter returns to the Not-Ready state unless **PEN SELECT** button **7** or **8** was the last button pressed during the plot. If **PEN SELECT** button **8** was the last button pressed, the plot will be repeated again and again. If **7** was the last button pressed, the plotter will draw one more test.

The test will appear as in the previous illustration, but will include an additional digitizing function. Because successful digitization requires registration of black marks by an optical sensor, you must place a *black* pen in stall 8 for the confidence test to work properly. The confidence test will produce one of the three following results.

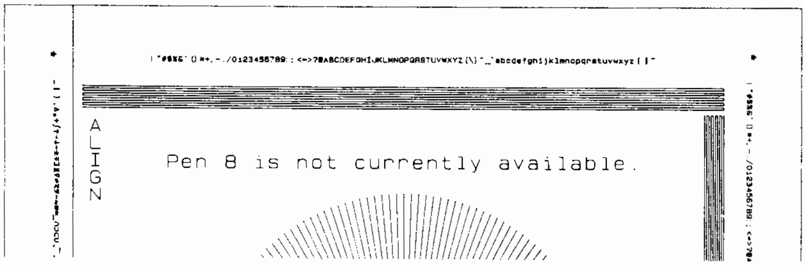
- Digitization functioning properly.



- Bad point. Digitization not functioning properly. Use of any color pen other than black in pen stall 8 will also give this result.



- Pen stall 8 empty or no carousel installed.



2. *Roll-feed mode with automatic take-up* (front take-up spool installed).

   This confidence test draws a long axis plot in two adjacent frames in addition to performing all of the functions in the previous confidence test. It also tests the polygon fill capabilities and the three character set fonts. *It will run correctly only on ANSI size E or ISO size A0 paper.* To run this test:

   a. Turn the plotter on, and install pens of different colors in stalls 1 through 8 of the appropriate carousel. Be sure that stall 8 contains a black pen.

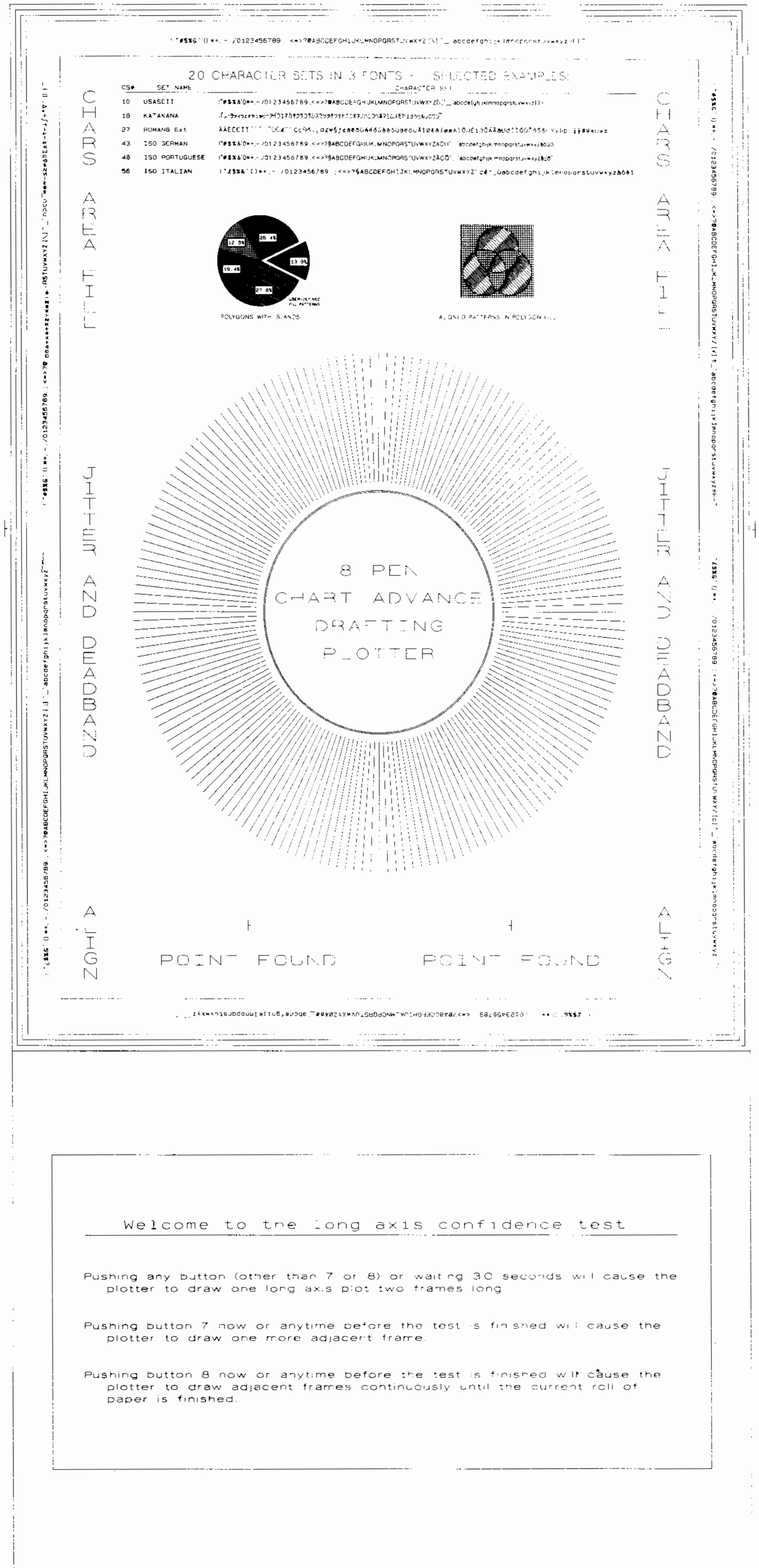
   b. Be sure the **CALIBRATE/NORMAL** switch is set to **NORMAL.** Press the **CONFIDENCE TEST** switch on the rear panel.

   c. If an error occurs during the test, the error number is displayed in the **PEN** displays and the test stops.

   d. When the plot is completed, the plotter returns to the Not-Ready state unless **PEN SELECT** button **7** or **8** was the last button pressed during the plot.

      Pressing button **7** anytime before the test is completed causes the plotter to draw one more adjacent frame.

      Pressing button **8** anytime before the test is completed causes the plotter to draw adjacent frames continuously until the current roll of paper is finished.



20 CHARACTER SETS IN 3 FONTS - SELECTED EXAMPLES
CHARS AREA FILL
JITTER AND DEADBAND
ALIGN
8 PEN
CHART ADVANCE
DRAFTING
PLOTTER
POINT FOUND
POINT FOUND
Welcome to the long axis confidence test
Pushing any button (other than 7 or 8) or waiting 30 seconds will cause the plotter to draw one long axis plot two frames long
Pushing button 7 now or anytime before the test is finished will cause the plotter to draw one more adjacent frame.
Pushing button 8 now or anytime before the test is finished will cause the plotter to draw adjacent frames continuously until the current roll of paper is finished.

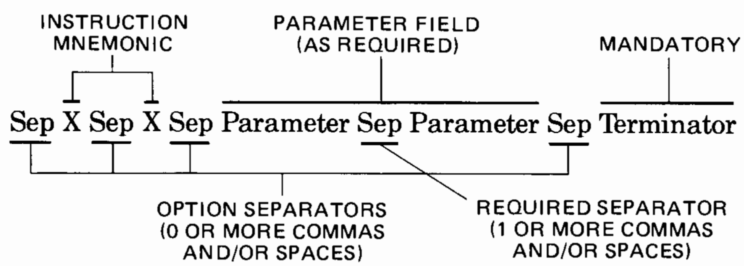
HP-GL and Device Control Instructions

Both the HP-IB and RS-232-C Interface Modes of the plotter use the same HP-GL instruction set. The set consists of 77 HP-GL instructions which activate the plotter and additional instructions which cause no operation but are included for compatibility with other HP plotters. In addition, the HP 7586 has five paper-advance instructions. Device control instructions are also used in both interface modes. These instructions are used to establish monitor mode and to obtain immediate output of available buffer space and the status of specific device conditions. Additional device control instructions are required by the RS-232-C Interface Mode and are used to establish plotter output conditions and handshake protocol, and to control conditions and obtain outputs which are pertinent only to the RS-232-C interface environment.

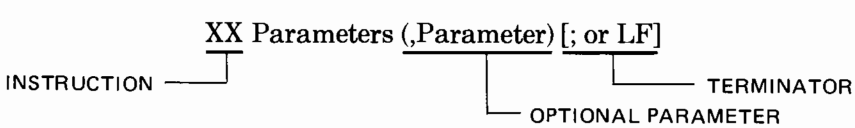
All HP-GL instructions enter the plotter's internal buffer and are executed in first-in/first-out sequence. Device control instructions do not enter the buffer, but instead are executed immediately upon receipt. Refer to Chapters 12 and 13 for the syntax and explanations of device control instructions.

HP-GL Syntax

Each HP-GL instruction begins with a two-letter mnemonic, which may be upper- or lowercase. If parameters are required following the mnemonic, they must be separated from each other by at least one comma or space. In the HP-IB Interface Mode, HP-GL instructions are terminated by a semicolon, line feed, or the first letter of the next mnemonic. In the RS-232-C Interface Mode, HP-GL instructions are terminated only by a semicolon or the first letter of the next mnemonic. In either interface mode, optional separators may be inserted as shown below:



NOTE: The syntax implemented on the HP 7580, 7585, and 7586 is extremely flexible and differs from that used on other plotters such as the HP 9872. Therefore, any software written for the HP 7580, 7585, and 7586 which takes advantage of their less rigorous syntax will not be able to drive earlier devices. If software is to be written to drive other HP-GL plotters and other graphics devices, the more rigorous syntax of the HP 9872 plotter should be used.



The 9872 syntax totally ignores spaces. It does not allow commas between the characters of the mnemonic, only one comma must separate parameters, and only a semicolon or LF may be used as the terminator. In addition, parameters requiring integer format may not contain a decimal point or decimal fraction. ■

Carriage return characters will be ignored, except as label characters or an output response terminator in an RS-232-C environment. Some instructions have optional parameters which, when omitted, assume a default value.

The label instruction, LB, and buffer label instruction, BL, are special cases: each must be terminated with the label terminator character. This character defaults to the ASCII end-of-text character, ETX (decimal equivalent 3), but may be changed from its default value using the define terminator instruction, DT.

The parameter fields must be specified in the format defined by the syntax of each respective HP-GL instruction. The format can be of three types:

1. Integer Format: The parameter must be an integer between $-2^{26}+1$ and $2^{26}-1$ after rounding ($2^{26} = 67\,108\,864$). Rounding is automatically performed if the parameter must be an integer. If the first digit after the decimal point is $\geqslant 5$, positive parameters are incremented by 1 and negative parameters are decremented by 1. If no sign is specified, the parameter is assumed to be positive; e.g., 1008.5 is rounded to 1009 while −1008.6 is rounded to −1009.
2. Decimal Format: A number where the integer portion is between $-2^{26}+1$ and $2^{26}-1$, and the optional decimal fraction has a maximum of 10 significant digits. The decimal point may be omitted when no decimal fraction is specified. If no sign is specified, the parameter is assumed to be positive.
3. Label Fields: Any sequence of characters. Refer to The Label Instruction, LB, (Chapter 7) for a complete description.

The syntax shown under the description for each HP-GL instruction uses the following notations:

MNemonic	— For readability, the mnemonic is shown uppercase and separated from the parameters and/or terminator.
required parameter	— All typeset items are required parameters.
( )	— All items in parentheses are optional.
c . . . c	— Any number of labeling characters.
(, . . .)	— Any number of the specified parameter.
term	— Required terminator. A semicolon (;) or the next mnemonic are valid terminators. For HP-IB mode, a line feed (LF) is also a valid terminator.

[TERM]	— The output terminator. For the HP-IB mode, all output responses include a carriage return and line feed [CR LF] as a terminator. For the RS-232-C mode, all output responses include a terminator as defined by The Set Output Mode Instruction, ESC . M. The default output terminator is a carriage return [CR].

The output terminator, denoted by [TERM] is sent from the plotter to the computer at the end of a response to an output instruction. This differs from the HP-GL terminator, denoted by *term,* which indicates the end of an HP-GL instruction sent to the plotter from the computer. You will learn more about output instructions and output terminators in Chapters 9, 12, and 13.

NOTE ON HP-IB OUTPUTS: After an output instruction is completely processed, the data requested is available when the plotter is instructed to talk. However, if another output instruction is processed before the previously requested data has been read from the plotter, the earlier output is no longer available; i.e., the only data available whenever the plotter is instructed to talk is that from the most recently processed output instruction. This note applies to both the buffered HP-GL and the nonbuffered device control output instructions. ■

NOTE ON RS-232-C OUTPUTS: Data requested by both buffered HP-GL and nonbuffered device control output instructions is output in accordance with the conditions specified by the ESC . M and/or ESC . N instructions. To avoid problems, it is recommended that requested data be read from the plotter prior to requesting additional data. An error 10 condition is set if another output request is received while the original request is executing. In this case, the original output instruction will continue normally while the new output instruction is ignored. ■

The following table summarizes the HP-GL instruction set.

HP-GL Instruction Set

Instruction	Definition
VECTOR GROUP	
AA X,Y,arc angle (, chord tolerance)	Arc absolute X,Y (i); arc angle, chord tolerance (d)
AR X,Y,arc angle (, chord tolerance)	Arc relative X,Y (i); arc angle, chord tolerance (d)
CI radius (, chord tolerance)	Circle radius (i); chord tolerance (d)
PA X,Y(,X,Y(, . . .))	Plot Absolute (i)
PD (X,Y, . . .)	Pen down (i)
PR X,Y(,X,Y(, ...))	Plot relative (i)
PU (X,Y, . . .)	Pen up (i)
(d) = decimal format (i) = integer format	

Instruction		Definition
CHARACTER GROUP		
BL	c . . . c	Buffer label (c)
CA	n	Designate alternate set n (i)
CC	chord angle	Character chord angle (d)
CM	switch mode (,fallback mode)	Character select mode n (i)
CP	spaces, lines	Character plot (d)
CS	m	Designate standard set m (i)
DI	run, rise	Absolute direction (d)
DL	(pen)X,Y,pen(, . . .)	Downloadable character (i)
*DR	run, rise	Relative direction (d)
DS	slot,set	Designate character set (d)
DT	t	Define label terminator (c)
ES	width(,height)	Extra width between characters and height between lines (d)
IV	slot (, left)	Invoke character slot (i)
LB	c . . . c	Label ASCII string (c)
LO	n	Label origin n (i)
OL		Output label length (d return)
PB		Print label buffer
SA		Select alternate character set
*SI	width, height	Absolute character size (d)
SL	tan θ	Absolute character slant from vertical (d)
*SR	width, height	Relative character size (d)
SS		Select standard character set
*UC	(pen,)X,Y,pen(, . . .)	User-defined character (i)
PEN/LINE TYPE GROUP		
AS	a(,n)	Select acceleration a for pen n (i)
FS	f(,n)	Select tip force f for pen n (i)
GP	(group(,pen number (,number of pens (,length))))	Define pen group (i)
LT	t(,l)	Designate line type t (i) and length l (d)
SG	n	Select pen group n (i)
SM	c	Symbol mode (c)
SP	n	Select pen n (i)
VS	v(,n)	Select velocity v for pen n (i)

* = Parameter range, conditions, or return affected by setting of **EMULATE/NORMAL** switch.

(c) = ASCII character

(d) = decimal format

(i) = integer format

Plotter Operation

Instruction	Definition
ADVANCE PAPER GROUP (HP 7586 only)	
AF	Advance one page length
AH	Advance one half-page length
EC (n)	Enable cutter (i)
FR	Advance frame
PG (n)	Advance one page length (i)
POLYGON GROUP	
EA X,Y	Edge rectangle absolute (i)
EP	Edge polygon
ER X,Y	Edge rectangle relative (i)
EW radius, start angle, sweep angle (,chord tolerance)	Edge wedge (i, d, d, d)
FP	Fill polygon
FT type (,spacing (,angle))	Fill type (i, i, d)
PM n	Polygon mode n (i)
PT n	Select pen thickness n (d)
RA X,Y	Fill rectangle absolute (i)
RR X,Y	Fill rectangle relative (i)
WG radius, start angle, sweep angle (, chord tolerance)	Fill wedge (i, d, d, d)
UF gap_1, gap_2 (, . . .)	User-defined fill type (i)
DIGITIZE GROUP	
DC	Digitize clear
DP	Digitize point
*OD	Output digitized point and pen status (i return)
AXES GROUP	
TL tp(,tn)	Tick length (d)
XT	X-axis tick
YT	Y-axis tick
CONFIGURATION AND STATUS GROUP	
AP n	Automatic pen operations (i)
CT n	Set chord tolerance mode (i)
*DF	Set default values
IM e(,s(,p))	Input e, s, and p masks (i)
*IN	Initialize

* = Parameter range, conditions, or return affected by setting of **EMULATE/NORMAL** switch.

(d) = decimal format

(i) = integer format

Instruction	Definition
CONFIGURATION AND STATUS GROUP (Continued)	
IP $P1_X,P1_Y(,P2_X,P2_Y)$	Input P1 and P2 (i)
*IW X_1,Y_1,X_2,Y_2	Input window (i)
NR	Enter View state
OA	Output actual position and pen status (i return)
*OC	Output commanded position and pen status (i return)
OE	Output error (i return)
OF	Output factors (i return)
OH	Output hard-clip limits (i return)
OI	Output 5 character identification (c return)
OO	Output options (i return)
*OP	Output P1 and P2 (i return)
OS	Output status (i return)
OT	Output carousel type and pen map (i return)
*OW	Output window (i return)
RO n	Rotate coordinate system (i)
SC $X_{min},X_{max},Y_{min},Y_{max}$	User unit scaling (i)

* = Parameter range, conditions, or return affected by setting of **EMULATE/NORMAL** switch.

(c) = ASCII character

(i) = integer format

The Default Instruction, DF

USES: The DF instruction sets certain plotter functions to predefined default conditions. Use this instruction to return certain HP-GL instructions to a known state while maintaining conditions that may have been set from the front panel. This assures that unwanted graphics parameters such as character size, slant, or scaling are not inherited from another program but that operator set conditions remain unchanged. The DF instruction is equivalent to pressing **ENTER** and then **CHART UNLOAD** on the plotter's front panel.

SYNTAX: *DF term*

PARAMETERS: None.

EXPLANATION: No parameters are used. However, the terminator must be included to complete the instruction. The following plotter functions are *not* affected by a DF instruction:

- Locations of P1 and P2
- Current pen and its position
- Current pen group and its definition

- Pen speed, force, and acceleration
- 90-degree rotation or axis alignment
- EC state (HP 7586 only)

Default conditions are affected by the position of the **EMULATE/NORMAL** switch. When the switch is set to **EMULATE**, a DF command is equivalent to executing all of the instructions shown under the **EMULATE** heading in the following table. If the switch is set to **NORMAL**, the functions identified with an asterisk are replaced by the functions and instructions under the **NORMAL** switch heading at the bottom of the table. In addition, the following conditions are established when the switch is in either position.

- The carriage return point is updated to the current pen position.
- PD and PU instructions with parameters are defaulted to be forms of the PA instruction (refer to The Plot Absolute Instruction, PA, Chapter 4).

Default Conditions

Function	Equivalent Instructions	Condition
	EMULATE	
Pen control	AP ;	Automatic as follows: • Lift or store a motionless pen after 10 seconds for drafting pens or after 65 seconds for fiber-tip and roller-ball pens. • Select only when required to draw.
Label buffer	BL **ETX**	Cleared
Alternate set	CA 0 ;	Character set 0
Character chord	CC ;	Set arc font chord angle to 5°
Character selection mode	CM ;	HP seven-bit mode
Standard set	CS 0 ;	Character set 0
Chord tolerance	CT ;	Set to 5 degrees for AA, AR, and CI
Digitize clear	DC ;	Clear DP instruction and turn **ENTER** LED off
*Relative direction	DR 1 , 0 ;	Horizontal characters
Label terminator	DT **ETX**	ETX (decimal equivalent 3)
Extra space	ES 0 , 0 ;	No extra space between characters
Fill type	FT ;	Type 1, solid bidirectional fill
Fill spacing	FT ;	1% of the diagonal distance from P1 to P2
Fill angle	FT ;	0°

*Replaced when switch is set to **NORMAL.**

Plotter Operation

Default Conditions (Continued)

Function	Equivalent Instructions	Condition
	EMULATE (Continued)	
Mask value	IM 223 , 0 , 0 ;	Recognizes all defined errors
Input window	IW ;	Set to hard-clip limits
Label origin	LO 1 ;	Standard labeling starting at current position
Line pattern length	LT ;	4% of the diagonal distance from P1 to P2
Line type	LT ;	Solid line
Plotting mode	PA ;	Absolute
Polygon buffer	PM 0 ; PM 2 ;	Polygon buffer cleared
Pen thickness	PT ;	0.3 mm
Scaling	SC ;	User unit scaling off
Character slant	SL 0 ;	0°
Symbol mode	SM ;	Off
*Relative size	SR ;	Character width = 0.285 cm Character height = 0.375 cm
Select set	SS ;	Select standard character set
Tick length	TL .5 , .5 ;	tp and tn = 0.5% of $\|P2_X - P1_X\|$ for Y-tick and 0.5% of $\|P2_Y - P1_Y\|$ for X-tick.
User-defined fill type	UF ;	Solid bidirectional fill
	NORMAL	
Absolute direction	DI 1 , 0 ;	Horizontal characters
Absolute size	SI ;	Character width = 0.285 cm Character height = 0.375 cm.

*Replaced when switch is set to **NORMAL**.

NOTE: These are absolute direction and size specifications. Subsequent changes in the location of scaling points P1 and P2 do not affect labels drawn with default parameters when the plotter is in **NORMAL** mode. ■

The Initialize Instruction, IN

USES: The IN instruction returns the plotter to the initial power-up conditions by program control. This instruction has no effect on handshake protocol in any RS-232-C environment but does clear any existing

I/O error condition. The instruction returns the plotter to a known state but cancels all conditions, except axis alignment, that may have been set from the front panel.

SYNTAX: *IN term*

PARAMETERS: None.

EXPLANATION: An IN instruction is equivalent to switching the plotter off and then on again, except that the plotter remains in the Remote state, the paper limits are not reestablished, and axis alignment is not changed. The initialize instruction sets the plotter to the same conditions as the default instruction, DF, and sets these additional conditions.

- Raises the pen graphically and physically.
- Cancels 90-degree rotation.
- Sets default pen speed, force, and acceleration values in accordance with the carousel installed in the plotter.
- Sets bit position 3 of the status word to 1.
- Clears any HP-GL (graphics) error condition.
- Clears lost mode (refer to The Plot Absolute Instruction, PA, Chapter 4).
- Clears memory in downloadable character buffer.
- Clears pen group definition.
- Returns P1, P2, and the Axis Align point to the X,Y coordinate values that were set when the paper limits were established. Remember that any existing axis alignment is maintained.

NOTE: If an axis alignment has been set, only P1 will return to its default physical position on the paper. The hard-clip limits are still compressed and P2 and the Axis Align point will be rotated from their default physical positions by an amount corresponding to the axis alignment. ■

The Not Ready Instruction, NR

USES: The NR instruction programmatically simulates pressing the front-panel **VIEW** button. The instruction is normally used at the end of a plotting program to ensure that the next program does not start to draw over the plot just completed.

SYNTAX: *NR term*

PARAMETERS: None.

EXPLANATION: After an NR instruction is executed, the front-panel **REMOTE** button must be pressed before subsequent plotting can occur. Normally, new paper is loaded.

Explanation of HP-GL Examples

All program listings are printed in dot matrix characters. Lines in program listings are either BASIC statements or strings of HP-GL instructions. Only the HP-GL instructions should be sent to the plotter. The output statements necessary to do this are not included in the listings. BASIC statements such as FOR, NEXT, or assignment statements should not be sent to the plotter.

Some strings of HP-GL instructions may require controller-dependent format statements in order to be accepted by the plotter. Lines requiring format statements and BASIC statements not sent to the plotter are noted with a ★ or ■ in each program listing.

All examples are intended to be run on ANSI D or ISO A1 size paper. Positioning instructions are included in each program listing so that a number of examples will fit on the same sheet of paper. In general, each example is drawn within an area 5000 plotter units square. The initialize instruction, IN, or the default instruction, DF, is included at the beginning of each program listing to assure no unwanted parameters are in effect from a prior plot.

Unless otherwise stated, all examples assume that:

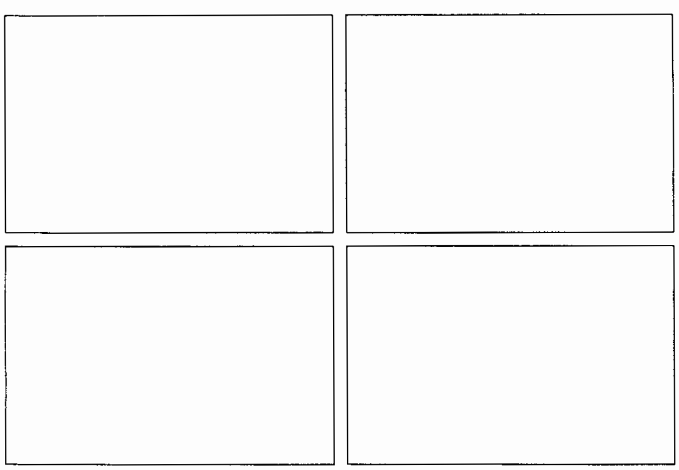
1. The **EMULATE/NORMAL** switch is set to **NORMAL.**
2. The **EXPAND/NORMAL** switch is set to **NORMAL.**
3. A pen carousel, with pens, is inserted.
4. Paper is loaded and the plotter is in the Remote state.

Advanced Programming Tips

Many software packages read P1 and P2 and use these points to define the maximum plotting area. You may want to obtain the largest plot possible on the plotter. This is the area of the hard-clip limits, as determined by paper size and the setting of the **EXPAND/NORMAL** switch, not the area established by the default settings of P1 and P2. The first three lines of the following listing will read the hard-clip limits and set P1 and P2 to these points, so that the largest area possible is used for plotting. In order to change the plotting area, this HP-GL routine should precede the PLOTTER IS statement when programming on HP desktop computers in BASIC.

Sometimes you want more than one plot on a page. The rest of the instructions define X and Y as the center of the plotting area, set windows, and outline four separate areas. A small space has been left between each area by adding or subtracting a constant value S from X- and Y-coordinates in the center of the total area. This program could be modified to divide the plotting area into thirds or into areas of any other size.

```
"IN;OH;"
!INSERT LINE TO READ COORDINATES INTO X1,Y1,X2,Y2
"IP";X1;Y1;X2;Y2
X=X1+(X2-X1)/2
Y=Y1+(Y2-Y1)/2
S=100
"IW";X1;Y1;X-S;Y-S;"SP1;PA";X1;Y1
"PD";X-S;Y1;X-S;Y-S;X1;Y-S;X1;Y1;"PU"
"IW";X+S;Y1;X2;Y-S;"SP2;PU";X+S;Y1
"PD";X2;Y1;X2;Y-S;X+S;Y-S;X+S;Y1;"PU"
"IW";X+S;Y+S;X2;Y2;"SP1;PA";X+S;Y+S
"PD";X2;Y+S;X2;Y2;X+S;Y2;X+S;Y+S;"PU"
"IW";X1;Y+S;X-S;Y2;"PU;SP2;PA";X1;Y+S
"PD";X-S;Y+S;X-S;Y2;X1;Y2;X1;Y+S;"SPO"
```




SCALE 1/2"=1'-0"
CEILING HEIGHT
R19 MINIMUM
MINIMUM
1,476 SQ.

Chapter 3
Scaling, Windowing and Rotation

What You'll Learn in This Chapter

In this chapter you will learn how to scale the plotting area into user units appropriate for your data, and how to set or read the current scaling points. After reading this chapter, you will also be able to restrict plotting to only a portion of the plotting area, read the current hard-clip limits, and rotate plots.

HP-GL Instructions Covered

IP	The Input P1 and P2 Instruction
OP	The Output P1 and P2 Instruction
SC	The Scale Instruction
OH	The Output Hard-Clip Limits Instruction
IW	The Input Window Instruction
OW	The Output Window Instruction
RO	The Rotate Coordinate System Instruction

Terms You Should Understand

Scaling — dividing the plotting area into units convenient for your application. Units need not be the same physical size in both axes, nor do there need to be an equal number of units in the X- and Y-axes.

Scaling Points — the points on the plotting surface moved to when the front-panel buttons **P1** and **P2** are pressed. These points are assigned the user-unit values specified by the parameters of the scaling instruction SC.

Window — that part of the plotting area in which plotting of points, lines, and labels can occur. At power-on, the window is set to the hard-clip limits of the plotter. Nothing can be drawn outside the current window.

Clipping — restricting plotting to a portion of the plotting area by establishing a window of a certain size.

Setting the Scaling Points

The size of a user unit is determined by the physical dimensions of the rectangular area whose opposite corners are P1 and P2, and by the parameters of the HP-GL instruction SC. The default location of P1 is the lower-left corner of the plot; the plotter unit coordinates of this point are negative X-and negative Y-values. The default location of P2 is the upper-right corner and its coordinates are positive in both X and Y.

Scaling points P1 and P2 can be set manually or programmatically. When manually relocating either or both the scaling points, always set P1 first, since P2 moves when P1 is moved.

To set P1: Use the joystick to position the pen at the location chosen for P1. Press **ENTER** and **P1** to store the new location of P1.

To set P2: Use the joystick to position the pen at the location chosen for P2. Press **ENTER** and **P2** to store the new location of P2.

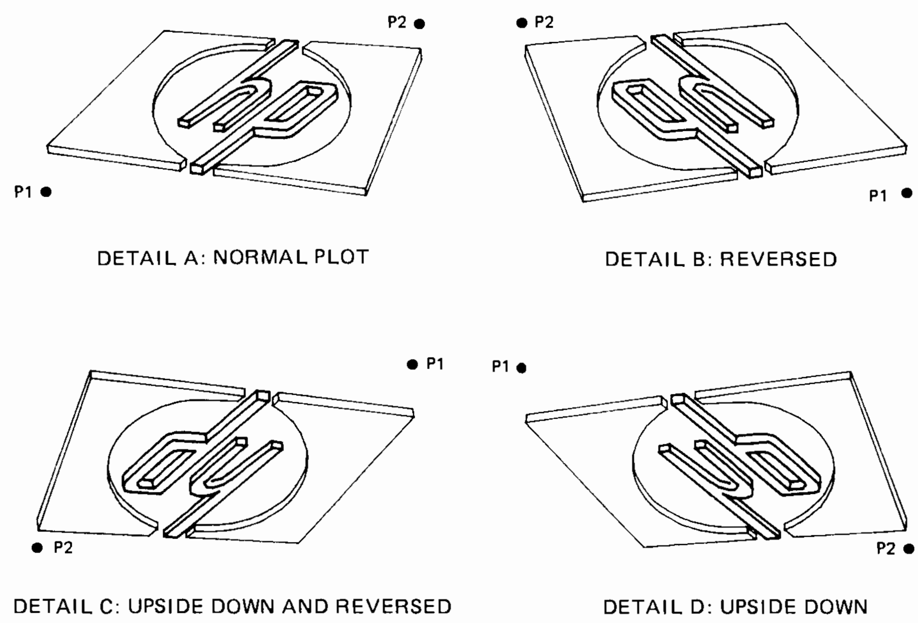
To set the scaling points programmatically, refer to The Input P1 and P2 Instruction, IP, in this chapter.

Unusual orientations of P1 and P2 (when P1 is the lower-right, upper-right, or upper-left corner and P2 is the opposite corner) can be used to produce mirror images of plots when the **EMULATE/NORMAL** switch is set to **NORMAL**. The conditions that govern mirror images of vectors and/or labels are as follows:

1. Vectors are mirrored only if user-unit scaling is specified using the SC instruction and P1 is not the lower-left corner.
2. Label characters are independent of P1 and P2 only if character direction and size is specified by a DI and SI instruction.

The following illustration shows the three normal types of mirror images. Specifying combinations of DI, DR, SI, and SR can produce many unusual combinations of mirrored label characters. For additional information on mirror images, refer to the SC instruction in this chapter and to the DI, DR, SI, and SR instructions in Chapter 7.

Mirror Images



The Input P1 and P2 Instruction, IP

USES: The IP instruction relocates the scaling points P1 and P2 through program control. The IP instruction is often used to ensure that a plot is always the same size, especially when the user and programmer are not the same person. It establishes program control of plot size and label direction. You can also use this instruction to move the scaling points P1 and P2 from their default or current locations; to give mirror images of vectors and labels; to change the size of a user unit, thus reducing or enlarging an image; to change the size or direction of labels when relative character size or direction is in effect; and to set P1 and P2 back to their default coordinates.

SYNTAX: *IP* $P1_X, P1_Y(, P2_X, P2_Y)$ *term*
or
IP *term*

PARAMETERS: All parameters must be in integer format and, for sheet media, lie within the range of $\pm(2^{26} - 1)$. Default values depend on paper size. Refer to the Default Coordinate Values for Scaling Points P1 and P2 tables in Chapter 2.

EXPLANATION: The coordinates of P1 and P2 are specified in the order shown above and are interpreted as plotter units.

Upon receipt of a valid IP instruction, bit position 1 of the status byte is set to 1. Refer to The Output Status Instruction, OS, in Chapter 9. An IP instruction with no parameters (IP ;) will set P1, P2, and the Axis Align point to the default positions established by the current paper size. Any existing axis align rotation is not affected (refer to Axis Alignment, Chapter 2).

If the coordinates of P2 are not specified, then P2 tracks P1 and its coordinates change so that the X- and Y-distances between P1 and P2 do not change. The Axis Align point always tracks P1, and its coordinates change when P1 moves.

If either coordinate of P2 is set equal to the corresponding coordinate of P1, then that coordinate of P2 is incremented by one plotter unit.

Refer to The Scale Instruction, SC, in Chapter 3 for examples of relocating P1 and P2 through program control. P1 and P2 may be relocated using front-panel controls rather than the IP instruction. An explanation of this method is included in the prior section of this chapter.

Typical default settings for P1 and P2 with various sizes of paper are listed under The Plotter Coordinate System, Chapter 2.

The Output P1 and P2 Instruction, OP

USES: The OP instruction makes the current coordinates of the scaling points P1 and P2 (in plotter units) available for output. Use the instruction to determine the position of P1 and P2 in plotter units. This information can be used with The Input Window Instruction, IW, to set the window to P1 and P2 under program control, to compute the number of plotter units per user unit when scaling is on, or to determine the numeric coordinates of P1 and P2 when they have been set manually.

SYNTAX: *OP* *term*

PARAMETERS: None.

EXPLANATION: After an OP instruction is received, the plotter will output the coordinates in plotter units as four integers in ASCII in the following form:

$P1_X$, $P1_Y$, $P2_X$, $P2_Y$ [TERM]

The range of the integers is determined by the setting of the EMULATE/NORMAL switch as follows:

EMULATE = −32 768 to +32 767

NORMAL = $-2^{26} + 1$ to $2^{26} - 1$ except that P2 tracks P1 and may be outside the range.

Upon completion of output, bit position 1 of the status word is cleared (set to 0).

The Scale Instruction, SC

USES: The SC instruction establishes a user-unit coordinate system by mapping values onto the scaling points P1 and P2. Use this instruction to plot in user units convenient to your application. Scaling can be anisotropic (unequal in X and Y) or isotropic (equal in X and Y). In addition, you can create mirror images of the plot and enlarge or reduce the plot size by changing the positions of P1 and P2.

SYNTAX: *SC* X_{min}, X_{max}, Y_{min}, Y_{max} *term*
or
SC *term*

PARAMETERS: All parameters must be in integer format and, for sheet media, lie within the range of $\pm(2^{26} - 1)$.

EXPLANATION: The parameters X_{min} and Y_{min} define the user-unit coordinates of P1, and the parameters X_{max} and Y_{max} define the user-unit coordinates of P2. Scaling points P1 and P2 retain the assigned user-unit coordinate values until scaling is turned off or another SC instruction redefines their user-unit coordinate values. Therefore, the size of a user unit will change when any change is made in the relative position and distance between P1 and P2. Specifying $X_{max} = X_{min}$ or $Y_{max} = Y_{min}$, or an invalid number of parameters will set an error condition (error number 3 or 2, respectively).

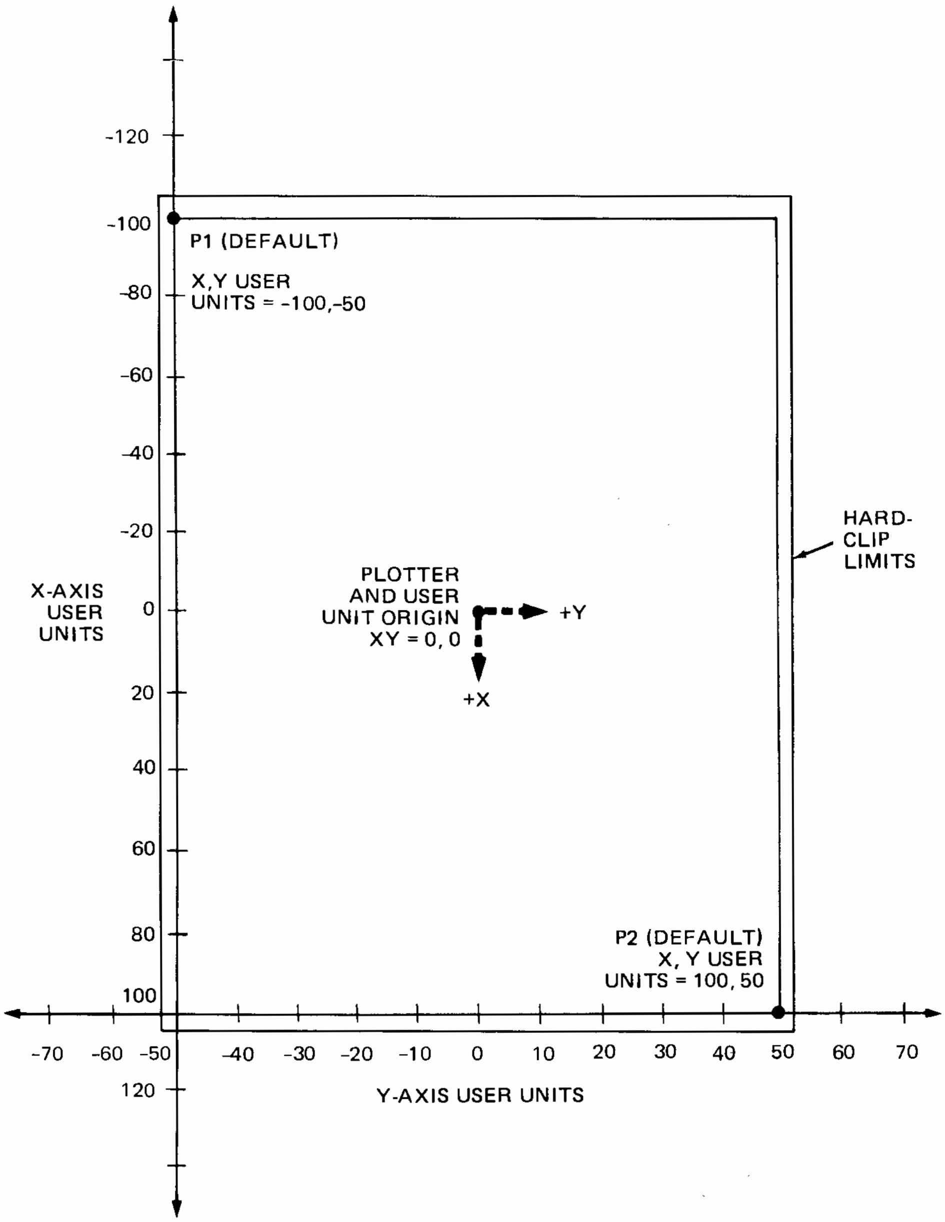
Execution of an SC instruction with no parameters (SC;) turns off scaled mode and all parameters of subsequent graphics instructions are interpreted as plotter units.

The user-unit coordinate system that is mapped onto the plotter-unit coordinate system by the SC instruction is not limited to the rectangle defined by P1 and P2; it extends over the entire range of the plotter unit coordinate system. (Refer to The Plotter Coordinate System, Chapter 2.)

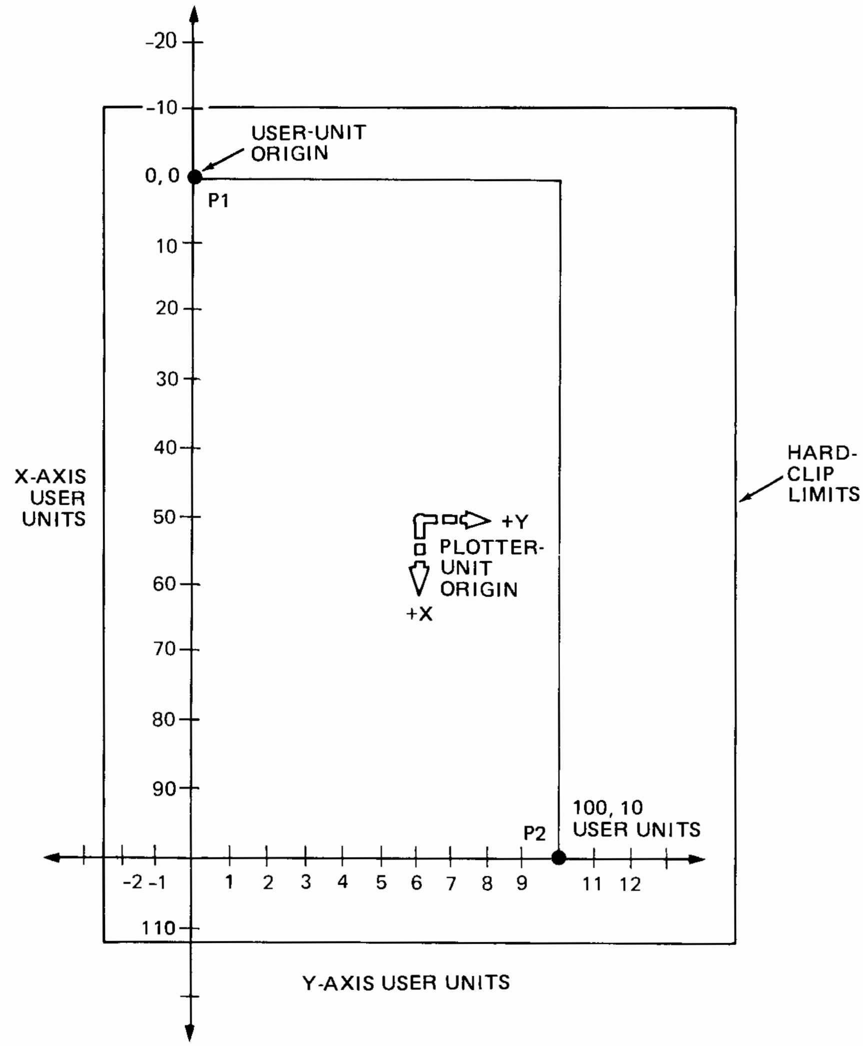
P1 and P2 may be any two opposite corners of a rectangle and may be located on or off the surface of the medium. While the default settings of P1 and P2, shown in the first illustration, lie 15 mm inside the hard-clip limits, it is possible to set P1 and P2 anywhere within the limits of the medium using front-panel controls, and anywhere within the range $-2^{26} + 1$ to $2^{26} - 1$ using The Input P1 and P2 Instruction, IP. The locations of P1 and P2 do not change when new paper is loaded unless a front-panel reset, HP-GL instruction IN, or another IP instruction is executed; so, when changing to a smaller piece of paper, P1 or P2 may lie outside the hard-clip limits. No matter where P1 and P2 are, the parameters of the SC instruction are always mapped onto the current P1 and P2 location.

In the following illustration, P1 and P2 assume default settings slightly inside the hard-clip limits. Both the plotter-unit and user-unit origins are at the center of the rectangular area defined by P1 and P2.

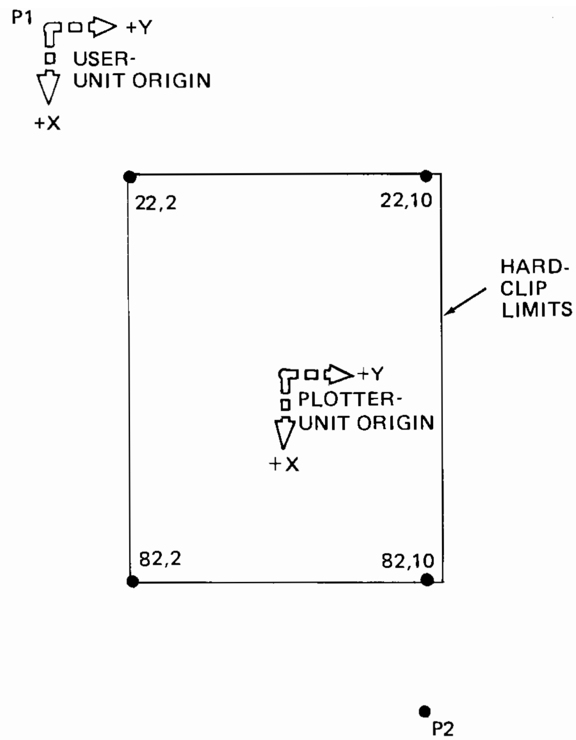
The HP-GL instructions IN ; SC −100 , 100 , −50 , 50 ; would give the following scaling on any size paper. Subsequent illustrations assume that D-size paper is loaded.



In the next illustration, P1 and P2 have been set from the front panel and placed well inside the hard-clip limits. The instruction SC 0 , 100 , 0 , 10 places the user unit origin at P1.



The third illustration shows what would happen if we loaded smaller paper and did not reset P1 and P2 from the position shown in the previous diagram. The user-unit coordinates nearest the corners of the paper are shown. A plot which filled the paper on the previous page would be clipped so only part of it fits on the paper shown here.



Once an SC instruction has been executed, all parameters of graphics instructions AA, AR, CI, EA, ER, EW, FT, PA, PR, RA, RR, and WG and the response to an OC instruction are interpreted as user units. The parameters of the IW instruction and the response to an OD or OW instruction are also interpreted as user units if the **EMULATE/NORMAL** switch is set to **NORMAL.**

One factor to consider when determining scaling parameters, besides the range of values of the variables, is the required smoothness of the plot. The plotter can only move to integer locations on the plotting surface. The smallest addressable move on the plotting surface is one plotter or user unit in the X- or Y-direction. This is the resolution of the plot. Resolution is limited in that 0.025 mm (0.001 in.) is the smallest move the plotter can make. This is true even when finer resolution is specified by the parameters of the SC instruction and the positions of P1 and P2. When resolution finer than 0.025 mm is specified, the plotter uses conventional rounding and positions the pen as close as possible to the specified coordinates in increments of 0.025 mm.

When the range of the scale parameters is small, resolution is coarse and may be insufficient to produce a satisfactory plot. While scaling could establish a user unit larger than the medium, the plotting area is usually divided into several hundred or several thousand units. The limitation on coarse resolution is determined by the minimum number of user units that will allow creation of a satisfactory plot.

A second factor to consider is whether the plots represent geometric or actual shapes. If so, isotropic scaling (equal in X and Y) must be established. The means to assure sufficient resolution and set isotropic scaling are demonstrated in the following examples.

The examples are based on the equations for a circle in polar coordinate form.

$$X = R\cos T \qquad \text{and} \qquad Y = R\sin T$$

where R is the radius of the circle and T is the angle in radians. Since there are π radians in 180 degrees, T must vary from 0 to 2π to draw a circle.

Example of Insufficient Resolution

This example shows that an indiscernible plot may result if user units are too large. More often the plot will be recognizable but the curves will not be sufficiently smooth. The above equations are used to draw a circle in the center of the plotting area. The resolution of the plot as established by the SC instruction is as follows: X-axis resolution = 5 mm since there are 5000 plotter units in the area divided into 25 user units in X.

$$5000 \times \frac{0.025\ \text{mm}}{25} = 5\ \text{mm}$$

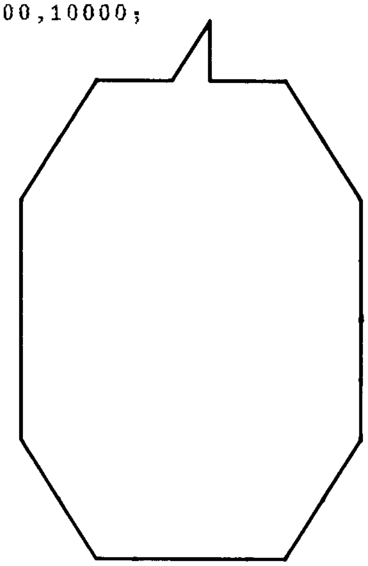
For the Y-axis resolution,

$$5000 \times \frac{0.025\ \text{mm}}{16} = 7.8\ \text{mm}$$

This is 200 times coarser than plotter-unit resolution.

Send:

```
  LINE 1.   IN;SP1;IP-15000,5000,-10000,10000;
  LINE 2.   SC0,25,0,16;
■ LINE 3.   FOR T=0 TO 2*π STEP π/20
■ LINE 4.   X=4.5*COS(T)+12.5
■ LINE 5.   Y=4.5*SIN(T)+8
★ LINE 6.   PAX,Y;PD;
■ LINE 7.   NEXT T
  LINE 8.   PU;
```



Indiscernible Circle

Line 1 initializes the plotter; selects pen number 1; positions P1 and P2

Line 2 scales the X-axis into 25 user units and the Y-axis into 16 user units

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Lines 3, 7 define a loop in which lines 4, 5, and 6 are executed 40 times as T ranges from 0 to 2π radians, in steps of $\pi/20$ radians

Line 4 computes the X-coordinate of a point on a circle with a radius of 4.5 user units, centered horizontally in the scaled area at 12.5

Line 5 computes the Y-coordinate of a point on the circle and centers the circle vertically at 8

Line 6 plots the X- and Y-coordinates computed in lines 4 and 5 after rounding them to integers

Line 8 raises the pen

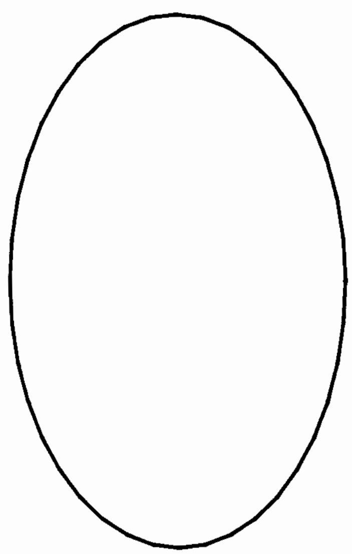
Example of Elliptical Distortion

This example solves the resolution problem by using a multiplier of 1000 in both the scaling parameters and the equations for X and Y. The multiplier is also used in the offset values to center the plot.

Notice that the plot is an ellipse, rather than the expected circle. This is a result of the unequal scaling in X and Y established by the parameters of the SC instruction.

Send:

```
  LINE 1.   IN;SP1;IP-10000,5000,-5000,10000;
  LINE 2.   SC0,25000,0,16000;
■ LINE 3.   FOR T=0 TO 2*π STEP π/20
■ LINE 4.   X=4.5*1000*COS(T)+12500
■ LINE 5.   Y=4.5*1000*SIN(T)+8000
★ LINE 6.   PAX,Y;PD;
■ LINE 7.   NEXT T
  LINE 8.   PU;
```



Distorted Circle

Examples of Equal Scaling

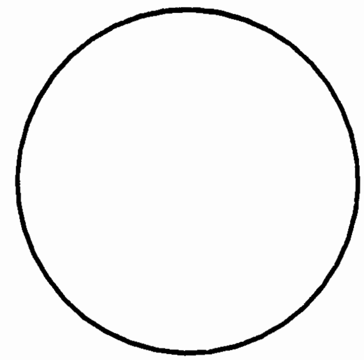
When P1 and P2 define a square area as they do here, the way to compensate for the distortion shown in the prior example is to set an equal number of user units in both X and Y. The plot will be a perfect circle.

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Send:

```
  LINE 1.   IN;SP1;IP-5000,5000,0,10000;
  LINE 2.   SC0,25000,0,25000;
■ LINE 3.   FOR T=0 TO 2*π STEP π/20
■ LINE 4.   X=4.5*1000*COS(T)+12500
■ LINE 5.   Y=4.5*1000*SIN(T)+12500
★ LINE 6.   PAX,Y;PD;
■ LINE 7.   NEXT T
  LINE 8.   PU;
```



Perfect Circle

When the area defined by P1 and P2 is not square and the scaling parameters do not establish isotropic scaling, a geometrically accurate plot can be created by using multipliers in the equations for X and Y or by adjusting the parameters of the SC instruction. This is frequently done in graphics programs which may run on a variety of graphic devices or paper sizes. The constant would be calculated elsewhere in the graphics program, once the device to be used for that plot was determined. The multiplier would then appear in lines 4 or 5 as a variable instead of the constant 1000 that appears in line 5 of the second program and the SC instruction in line 2 of the second program would depend on that constant.

The same circle will be drawn by both the following programs in the same area as the ellipse drawn earlier. Note the area defined by P1 and P2 is not square. The first program adjusts the SC parameters to match the IP instruction and one scaled unit equals one plotter unit. One scaled unit could equal any number of plotter units. The second program adjusts the multiplier in the equation for Y to correspond to the ratio $(P1_X - P2_X)/(P1_Y - P2_Y)$.

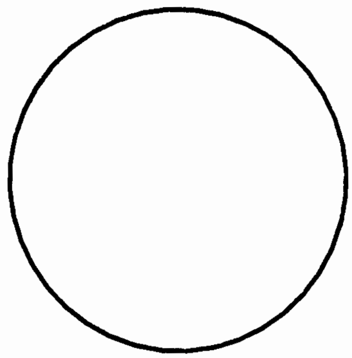
Send:

```
  LINE 1.   IN;SP6;IP-10000,6875,-5000,8125;
  LINE 2.   SC0,5000,0,1250;
■ LINE 3.   FOR T=0 TO 2*PI STEP PI/20
■ LINE 4.   X=4.5*200*COS(T)+2500
■ LINE 5.   Y=4.5*200*SIN(T)+625
★ LINE 6.   PAX,Y;PD;
■ LINE 7.   NEXT T
  LINE 8.   PU;
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

```
  LINE 1.   IN;SP6;;IP-10000,6875,-5000,8125;
  LINE 2.   SC0,5000,0,5000;
■ LINE 3.   FOR T=0 TO 2*PI STEP PI/20
■ LINE 4.   X=4.5*200*COS(T)+2500
■ LINE 5.   Y=4.5*800*SIN(T)+2500
★ LINE 6.   PAX,Y;PD;
■ LINE 7.   NEXT T
  LINE 8.   PU;
```



Scaling can also be used to expand or contract (rubber) a graphics plot. To do this, the locations of P1 and P2 are changed.

Example of a Contracted Plot

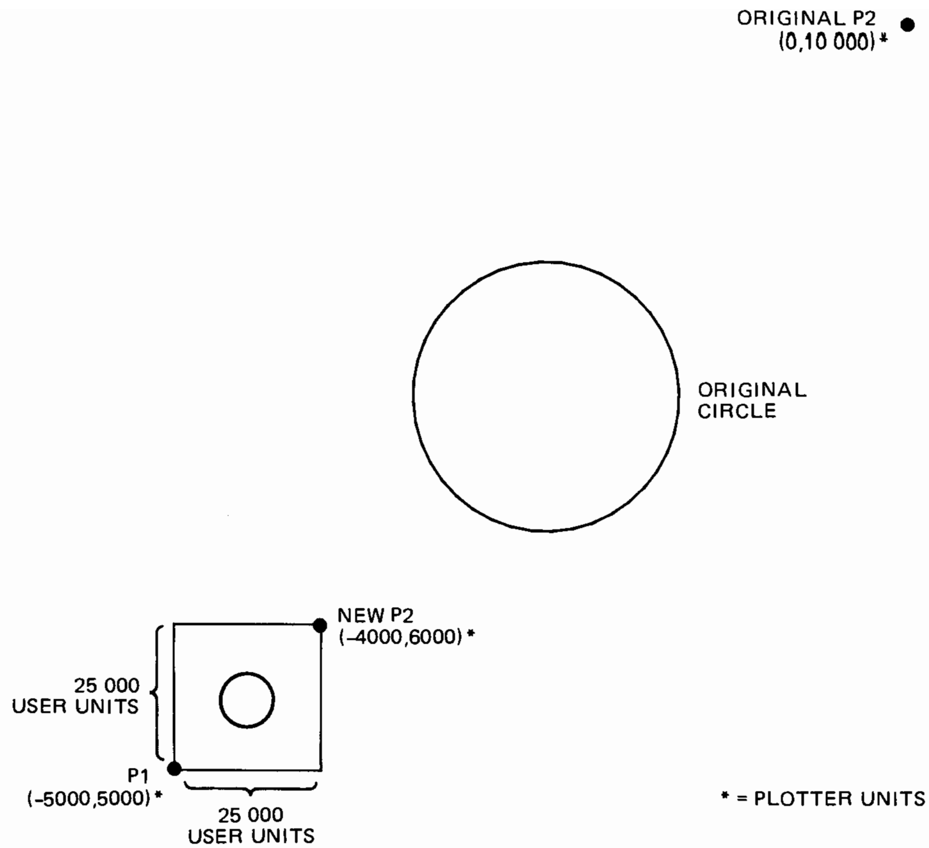
The following listing retraces the perfect circle plotted in example 3, then relocates P2 closer to P1 and generates a second circle. Notice that the second circle is much smaller, but is centered within the rectangle defined by P1 and the new P2 location.

Send:

```
  LINE 1.   IN;SP1;IP-5000,5000,0,10000;
  LINE 2.   SC0,25000,0,25000;
■ LINE 3.   GOSUB 7
  LINE 4.   IP-5000,5000,-4000,6000;
■ LINE 5.   GOSUB 7
■ LINE 6.   END
■ LINE 7.   FOR T=0 TO 2*π STEP π/20
■ LINE 8.   X=4.5*1000*COS(T)+12500
■ LINE 9.   Y=4.5*1000*SIN(T)+12500
★ LINE 10.  PAX,Y;PD;
■ LINE 11.  NEXT T
  LINE 12.  PU;
■ LINE 13.  RETURN
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.



A third phenomena which can be accomplished using scaling is plotting of mirror images.

Example of Mirror Images

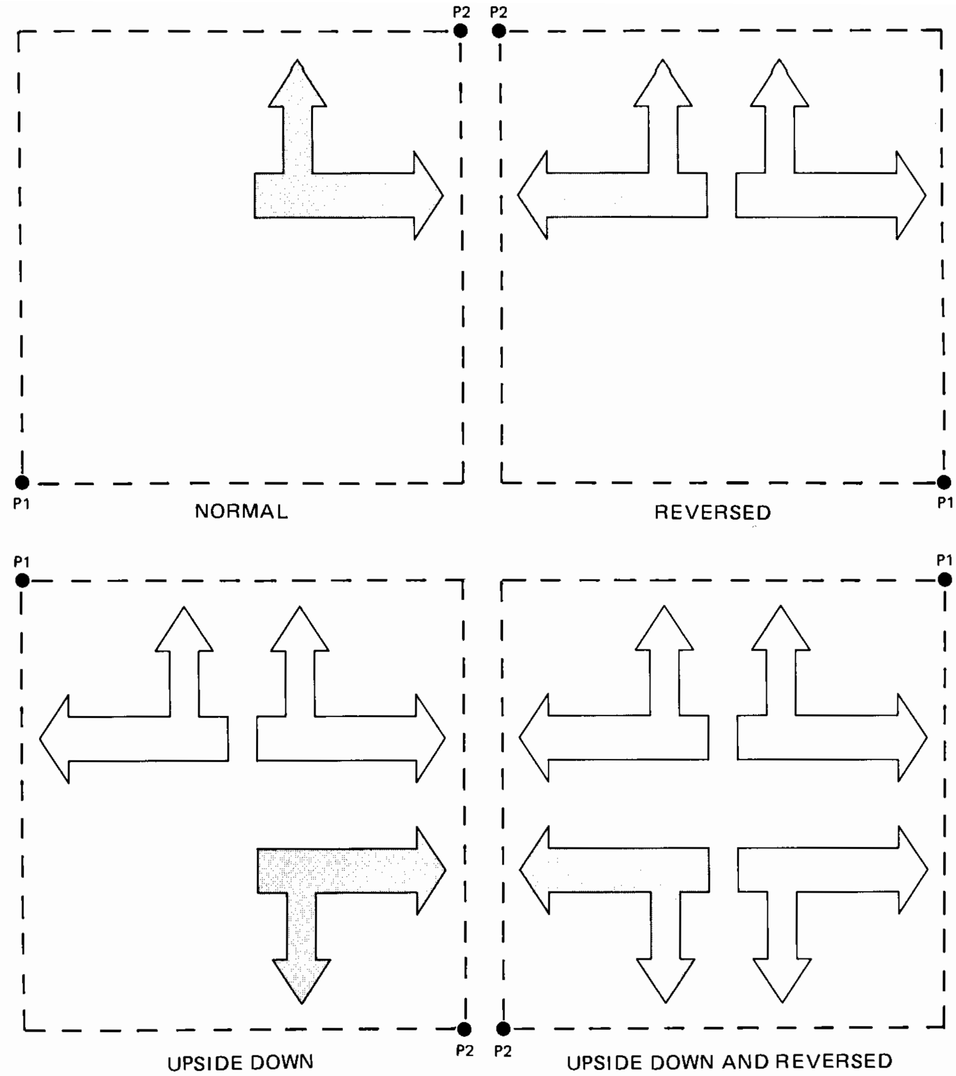
Mirror images of normal vector plots can be produced by changing the locations of scaling points P1 and P2 to one of their unusual orientations after the SC instruction has been executed (refer to Setting the Scaling Points, Chapter 3). The following listing illustrates how mirror images of vectors are produced. The resultant plot is shown in four stages to track the sequence of the program listing.

Send:

```
  LINE 1.   IN;SP1;IP0,5000,5000,10000;
  LINE 2.   SC-15,15,-10,10;
■ LINE 3.   GOSUB 11
  LINE 4.   IP5000,5000,0,10000;
■ LINE 5.   GOSUB 11
  LINE 6.   IP0,10000,5000,5000;
■ LINE 7.   GOSUB 11
  LINE 8.   IP5000,10000,0,5000;
■ LINE 9.   GOSUB 11
■ LINE 10.  END
  LINE 11.  PA1,2,PD,1,4,3,4,3,7,2,7,4,9,6,7,5,7,5,4;
  LINE 12.  PA12,4,12,5,14,3,12,1,12,2,1,2,PU;
■ LINE 13.  RETURN
```

Line 1	initializes the plotter, selects pen number 1, and positions P1 and P2 in their normal orientation
Line 2	scales the plot into user units
Lines 3, 11	produce a normal plot of the arrows
Line 4	relocates P2 above and to the left of P1
Lines 5, 11	draw a second set of arrows which are a reversed mirror image of the normal plot
Line 6	relocates P2 below and to the right of P1
Lines 7, 11	draw a third set of arrows which are an upside down mirror image of the normal plot
Line 8	relocates P2 below and to the left of P1
Lines 9, 11	draw a fourth set of arrows which are an upside down and reversed mirror image of the normal plot

■ BASIC statement. Do not send to the plotter.



The Output Hard-Clip Limits Instruction, OH

USES: The OH instruction outputs the lower-left (LL) and upper-right (UR) coordinates of the current hard-clip limits. Use the instruction to determine the plotter-unit dimensions of the area in which plotting can occur.

SYNTAX: *OH term*

PARAMETERS: None.

EXPLANATION: After an OH instruction is received, the plotter will output the LL and UR coordinates in plotter units as four integers in ASCII in the following form:

X_{LL}, Y_{LL}, X_{UR}, Y_{UR} [TERM]

The plotter outputs a negative sign for negative coordinates; leading zeros and positive signs are suppressed.

The hard-clip limits determine the maximum available plotting area and are initially established when the paper edges are sensed upon entry into either the View or Remote operating state. Refer to The Plotter Coordinate System, Chapter 2. The Input Window Instruction, IW, which follows can further limit the plotting area by establishing soft-clip limits anywhere within the hard-clip limits. Use of an IW instruction (soft clipping) does not affect the output from the OH instruction.

Other than changing the paper size, only an axis alignment operation can change the physical size of the hard-clip area. However, the 90-degree rotation function will change the LL and UR coordinate values.

The Input Window Instruction, IW

USES: The IW instruction restricts programmed pen motion to a specific rectangular area. This area is called the window. Use the instruction to establish a soft-clip area, i.e., restrict plotting to a certain area of the paper. The instruction is especially useful when your data should fall in a certain range but your scaling is larger (perhaps you have left room for labels) and you don't want lines outside the normal data area. It is also useful when hatching (shading) rectangular areas.

SYNTAX: *IW* X_1, Y_1, X_2, Y_2 *term*
or
IW *term*

PARAMETERS: All parameters must be in integer format and, for sheet media, lie within the range of $\pm(2^{26} - 1)$.

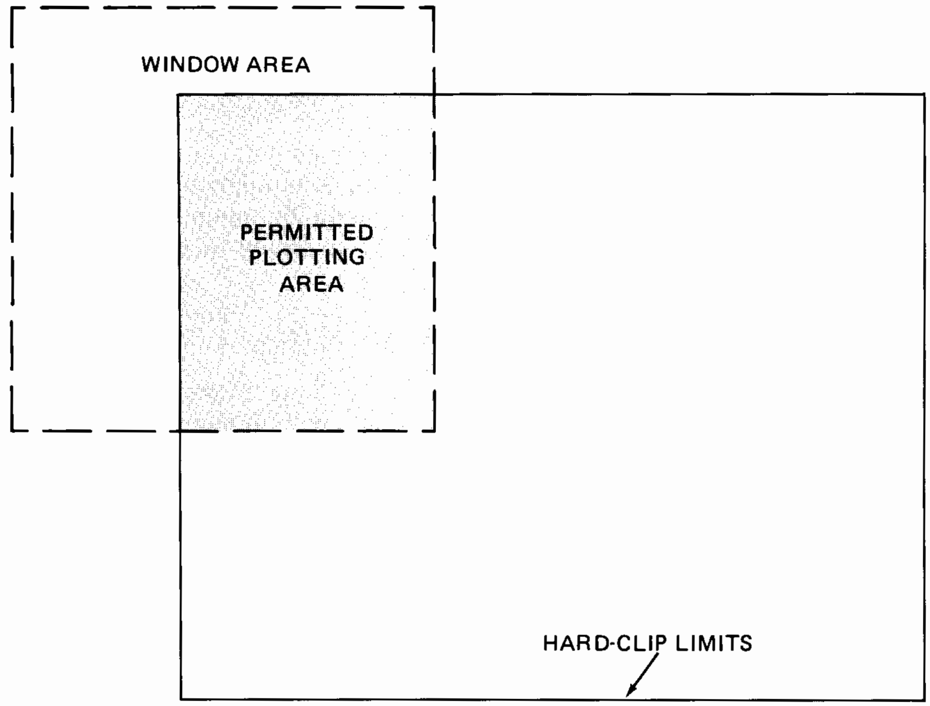
EXPLANATION: Specifying the wrong number of parameters or $X_1 = X_2$ or $Y_1 = Y_2$ will set an error condition (errors 2 and 3, respectively). The four parameters specify the X- and Y-coordinates of any opposite corners of the window area in units of the currently defined coordinate system. The position of the **EMULATE/NORMAL** switch and the status of user-unit scaling determines whether the integer parameters are interpreted as plotter units or user units as follows:

EMULATE — User-unit scaling does not affect the IW instruction and the parameters are interpreted as plotter units.

NORMAL — If user-unit scaling *is not* active at the time the window is defined, the parameters are interpreted as plotter units.

If user-unit scaling *is* active at the time the window is defined, the parameters are interpreted as current user units. When a window is defined in current user units, the window will compress or expand (rubber) in direct proportion to subsequent changes in the scaling points P1 and P2. Turning off scaling does not eliminate or change the window. Clipping will still occur if an attempt is made to plot outside the physical area previously defined in user units. The plotter now monitors that same area by expressing the same boundaries as plotter units.

Any specified window (plotter units or user units) is ANDed with the current hard-clip limits and restricts programmed pen motion (labeling and vectors) to that area which is common to both the window area and the hard-clip limits as shown in the following illustration.



At power-on or upon execution of a DF or IN instruction, or IW instruction without parameters (IW ;), the window is automatically set at the current hard-clip limits.

The window can be entirely outside the hard-clip limits if:

- The IW instruction defines it as such.
- The current window is defined in user units, and a subsequent change in the position of P1 and/or P2 causes it to move so it no longer intersects the hard-clip limits.
- Default values are not reestablished and paper is loaded which is so much smaller in one or both axes that a previously defined window no longer intersects the newly established hard-clip limits.

If a window is completely outside of the hard-clip limits, the **OUT OF LIMIT** light will be on and no programmed pen motion can occur. For additional details concerning programmed pen motion with respect to the window area, refer to The Plot Absolute Instruction, PA, in Chapter 4.

NOTE: Pen motions invoked from the front panel are not restricted by a window. However, the **OUT OF LIMIT** light will be on if the pen is moved outside of the window area. ■

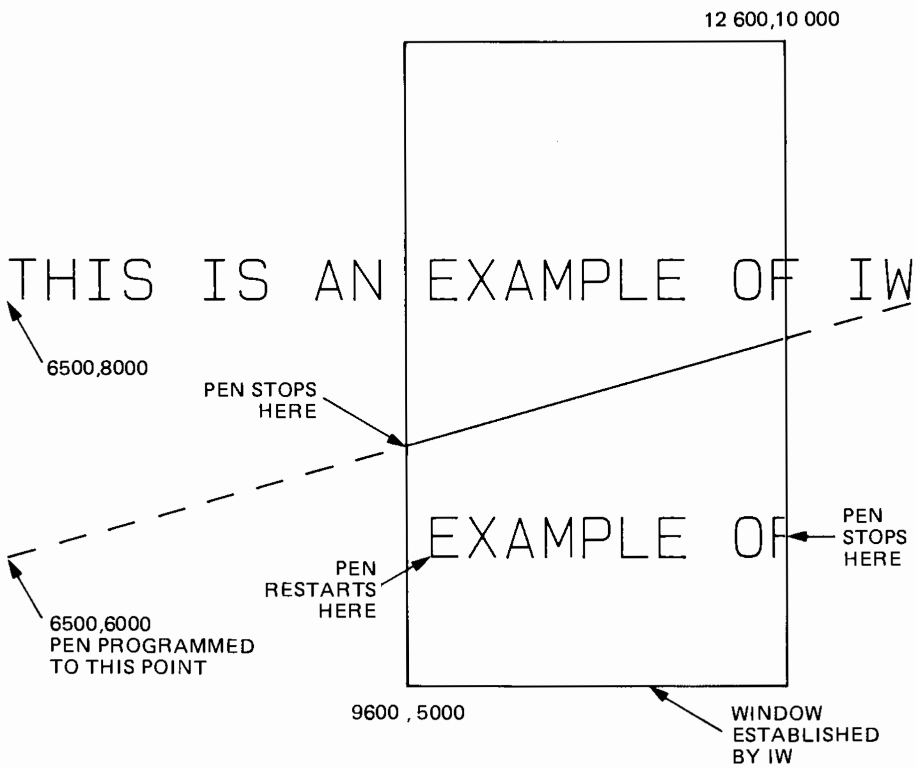
The window is set to the hard-clip limits (the equivalent of IW ;) if the window is defined in user units and the coordinates of either corner are out of the range -2^{29} to $2^{29} - 1$ when expressed in plotter units. For a discussion of converting from plotter units to user units, see Appendix C.

Example — Effects of IW on Labels and Vectors

The following example demonstrates the effects on labels and vectors when you specify a window.

Send:

```
LINE 1.   IN;SP1;
LINE 2.   PA6500,8000;SI.5,.75;LBTHIS IS AN EXAMPLE OF IW␃
LINE 3.   IW9600,5000,12600,10000;
LINE 4.   PD;PA6500,6000;LBTHIS IS AN EXAMPLE OF IW␃
```



Line 1 initializes the plotter and selects pen number 1

Line 2 moves the pen to the position where the label will begin, sets the character size, and sends the label string

Line 3 establishes a window to limit the plotting area

Line 4 moves the pen down and to the left margin to repeat the label string. Note that the vector and label string are clipped and are only plotted within the window area

Scaling & Windows

The input window instruction can be used to specify a limited portion of a plot or an enlarged limited portion of the plot if scaling is on. The following example plots the function sine x from -4π to 4π.

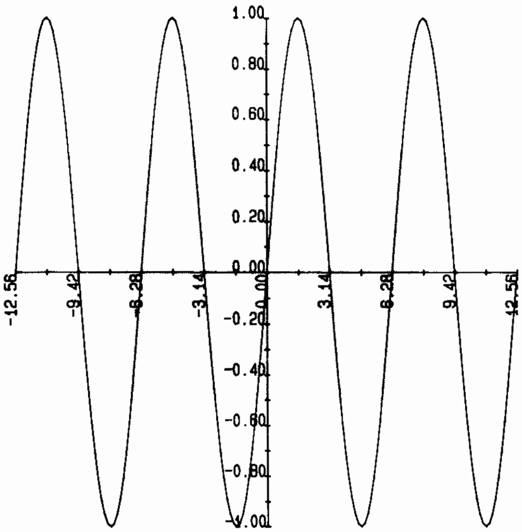
The subroutine to draw the labeled axes and the sine function is called three times; first without clipping, then with the window defined in line 7, and finally clipped to the same user unit window defined on the new P1, P2 rectangle specified in line 9. Reduced versions of the plots resulting from each call are shown here. You may wish to delay study of the subroutine portion of the program (line 12 on) until after you have read through Chapter 8 in this manual.

Send:

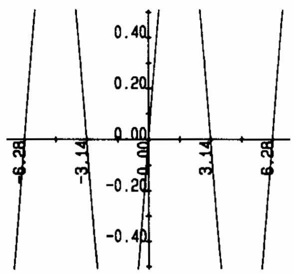
```
  LINE 1.   IN;SP1;
  LINE 2.   SR1.1,2.2;
  LINE 3.   IP-15000,0,-10000,5000;
  LINE 4.   SC0,6500,0,6500;
■ LINE 5.   GOSUB 12
  LINE 6.   IP-10000,0,-5000,5000;
  LINE 7.   IW1425,1625,5075,4875;
■ LINE 8.   GOSUB 12
  LINE 9.   IP-7500,-2500,2500,7500;
■ LINE 10.  GOSUB 12
■ LINE 11.  END
  LINE 12.  PA50,3250;PD;XT;
■ LINE 13.  FOR Xinterval=1 TO 16
  LINE 14.  PR400,0;XT;
■ LINE 15.  NEXT Xinterval
  LINE 16.  PU;PA50,3250;DI0,1;CP-6,0;
■ LINE 17.  FOR Xlabel=-12.56 TO 12.56 STEP 3.14
★ LINE 18.  LBXlabel%CP-6,0;PU;PR800,0;
■ LINE 19.  NEXT Xlabel
  LINE 20.  PA3250,50;PD;YT;
■ LINE 21.  FOR Yinterval=1 TO 20
  LINE 21.  PR0,320;YT;
■ LINE 22.  NEXT Yinterval
  LINE 23.  PU;PA3250,50;DI;CP-6,0;
■ LINE 24.  FOR Ylabel=-1 TO 1 STEP .2
★ LINE 25.  LBYlabel%CP-6,0;PU;PR0,640;
■ LINE 26.  NEXT Ylabel
■ LINE 27.  FOR X=-4*PI TO 4*PI+.1 STEP PI/20
■ LINE 28.  Xcoord=6400/25.13*X+3250
■ LINE 29.  Ycoord=3200*SIN(X)+3250
★ LINE 30.  PAXcoord,Ycoord;PD;
■ LINE 31.  NEXT X
  LINE 32.  PU;PA6500,6500;
■ LINE 33.  RETURN
```

■ BASIC statement. Do not send to the plotter.

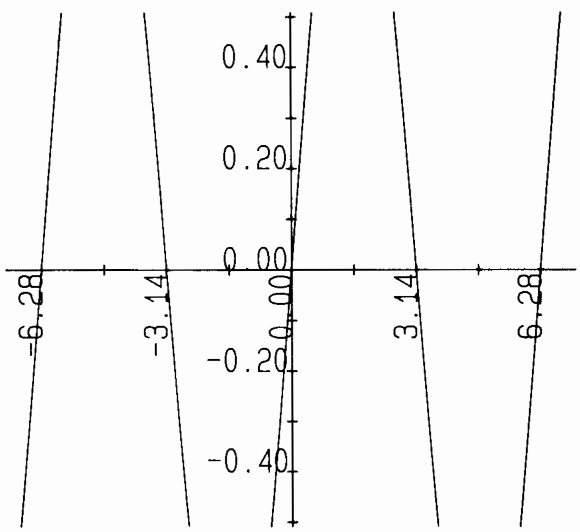
★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.



ORIGINAL CALL



CALL WITH WINDOW



CALL WITH WINDOW RUBBERED ONTO NEW AREA

The Output Window Instruction, OW

USES: The OW instruction obtains the X- and Y-coordinates of the lower-left and upper-right corners of the window area in which plotting can currently occur. Use the instruction to determine the intersection of the plotting window and the physical hard-clip limits.

SYNTAX: *OW term*

PARAMETERS: None.

EXPLANATION: After an OW instruction is received, the plotter will output the coordinates of opposite corners of the plotting area. In **NORMAL** mode, the output is in current units as four integers in ASCII in the following form:

X_{LL}, Y_{LL}, X_{UR}, Y_{UR} [TERM]

In EMULATE mode, output is in plotter units. If an IW instruction has been executed, the plotter will return the intersection of the parameters of the IW instruction and the hard-clip limit window.

The range of integers is limited to the range of the currently selected paper size.

The Rotate Coordinate System Instruction, RO

USES: The RO instruction programmatically rotates the plotter-unit and user-unit coordinate systems 90 degrees about the plotter-unit coordinate origin. This instruction allows plots to be oriented vertically or horizontally, regardless of whether the paper is loaded with the short or long dimension along the platen.

SYNTAX: *RO* n *term*
or
RO *term*

PARAMETERS:

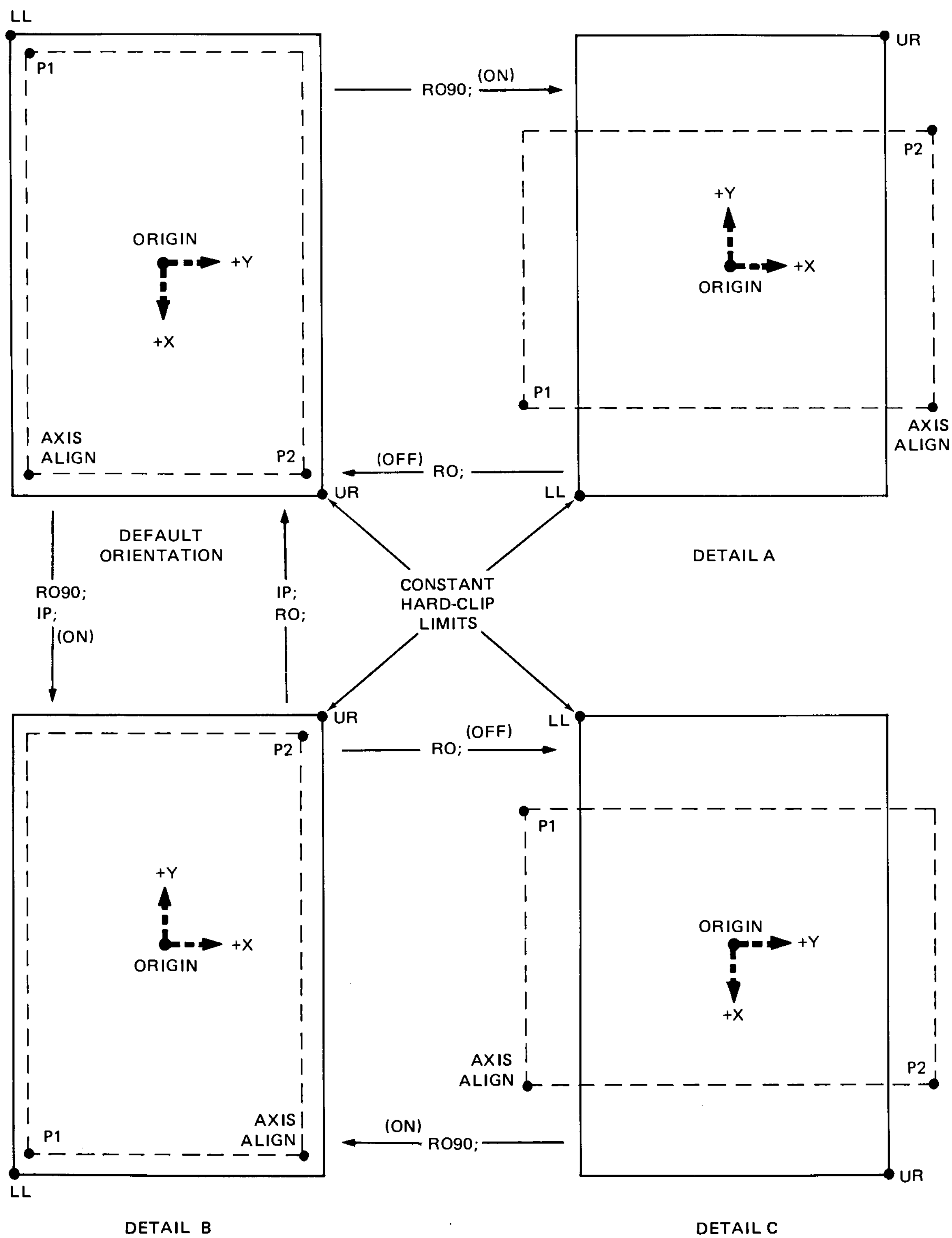
Parameter	Format	Range	Default
n	integer	0 or 90	0

EXPLANATION: The parameter n may take on only the values 0 or 90. The instruction RO 0 ; is the same as RO ; and turns off rotation. The instruction RO 90 ; rotates the current coordinate system 90 degrees about the plotter-unit coordinate origin. Rotations are not cumulative, and the rotate function can only be toggled on or off. Specifying a parameter other than 0 or 90 will set an error condition and cause the instruction to be ignored.

The following table shows the plotter's response to respective forms of the RO instruction and front-panel operation.

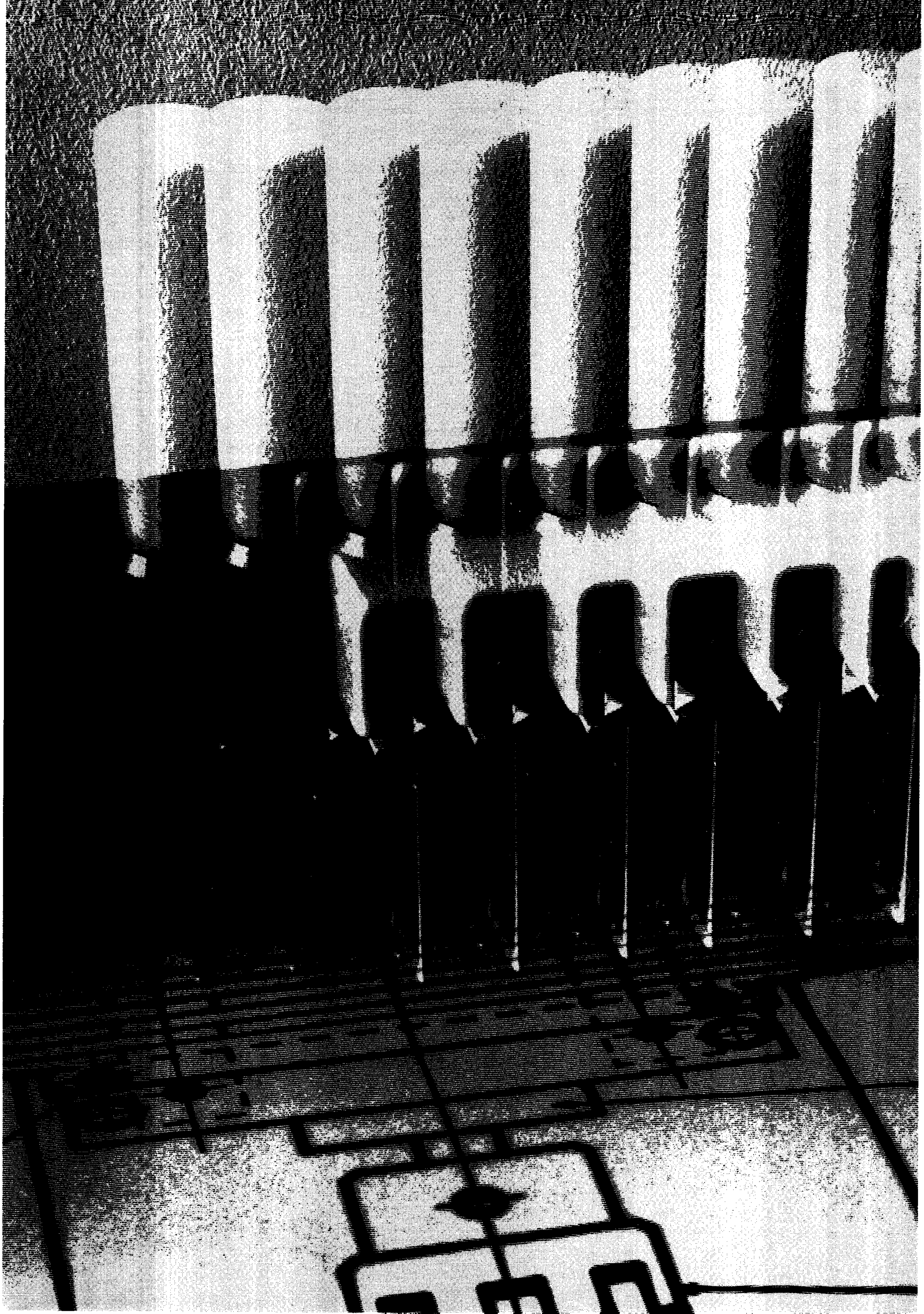
Instruction	Front-panel Buttons	Plotter Response
RO90;	ROTATE	Rotates the X- and Y-axes 90 degrees counterclockwise.
RO90;IP;	ENTER followed by ROTATE	Rotates the axes counterclockwise and relocates P1 and P2 inside the paper's hard-clip limits.
RO; or RO0;	ENTER followed by CHART HOLD	Cancels rotation.

Rotation can be turned on programmatically and turned off via the front-panel ROTATE button or vice versa. The following illustration is similar to the one used to describe the operation of the front-panel ROTATE and differs only in that it shows equivalent program instructions instead of front-panel controls. Refer to The Plotter Coordinate System, Chapter 2, for the detailed explanation of 90-degree rotation.



NOTE: (ON) AND (OFF) DEFINES THE STATE OF ROTATE LAMP AFTER INDICATED INSTRUCTION(S).

Scaling & Windows



Chapter 4
Pen Control and Plotting

What You'll Learn in This Chapter

In this chapter you will learn all aspects of pen control and plotting movements. You will learn how to select pens and control the automatic pen pickup and store operations, how to set pen acceleration, velocity, and force, and how to map pens to a specified group. You will learn how to plot absolute X,Y coordinates and how to plot with coordinates relative to the current pen position. Finally, you will learn how to send variables as parameters of plot instructions.

HP-GL Instructions Covered

SP	The Pen Select Instruction
PU	The Pen Up Instruction
PD	The Pen Down Instruction
PA	The Plot Absolute Instruction
PR	The Plot Relative Instruction
AP	The Automatic Pen Pickup Instruction
AS	The Acceleration Select Instruction
FS	The Force Select Instruction
VS	The Velocity Select Instruction
GP	The Group Pen Instruction
SG	The Select Pen Group Instruction

Terms You Should Understand

Absolute Plotting — plotting to the points whose location is specified with relation to the origin (0,0). A PA instruction with parameters always moves to the same point on the plotting surface, regardless of the pen's location before the move.

Relative Plotting — plotting to a point whose location is specified relative to the current pen position. The point moved to then becomes the effective origin for the next parameter of a plot relative instruction.

Plotter Unit Equivalent — the X,Y coordinates of a point, given in user units, if the location was expressed in plotter units.

Current Units — plotter units (if scaling is off) or user units (if scaling is on). The Scale Instruction, SC, determines whether scaling is on or off. For discussions of plotter units, user units, and scaling, refer to Chapters 2 and 3. If a plotting instruction can be interpreted as either plotter or user units, it will be described in the Parameters section as being specified in "current units."

The Pen Select Instruction, SP

USES: The SP instruction exchanges pens between the carousel and the pen holder. Use this instruction to load a pen into the pen holder so that drawing will occur. You can also use it to select pens of different colors or widths during a plotting program. Then, use SP at the end of a program to return the pen to the carousel.

SYNTAX: *SP* pen number *term*
or
SP *term*

PARAMETERS:

Parameter	Format	Range	Default
pen number	integer	0–8	0

EXPLANATION: The pen number parameter corresponds to the pen numbers marked on the carousels. The pen number is interpreted as listed below.

1. Pen Number = 1–8. Pen numbers 1 through 8 cause the appropriate pen to be placed in the pen holder. If there is currently a pen in the holder, this pen is stored before the new pen is selected. When selecting a pen, the SP instruction is ignored in these cases: there is no pen in the selected stall, a pen number larger than 8 is selected, or there is no carousel in the plotter. After selecting a new pen, the pen holder returns to the current graphics position.
2. Pen Number = 0 or No Parameter. Pen number 0 (or no parameter) returns the pen currently in the pen holder to the stall from which it came, if possible. If the stall is occupied, the pen is placed in the lowest-numbered vacant stall. The SP instruction is ignored if there are no vacant stalls, or if there is no carousel in the plotter. After storing a pen, the pen holder remains by the carousel, but the current graphics position does not change.

NOTE: The **PEN** displays on the front panel show the pen number, speed, and force for the pen currently in the pen holder. The display is not updated upon receipt of a pen select instruction until the pen is actually selected. When bit 2 of the pen control word is set (refer to the AP instruction in this chapter), the pen is not retrieved until needed to draw, and the **PEN** display information is not updated until the new pen is retrieved. ■

If you specify a pen parameter less than 0, error 3 (bad parameter) is generated and the instruction is ignored. If you specify a pen parameter greater than 8, the instruction is ignored without generating an error.

The Pen Instructions, PU and PD

USES: The PU and PD instructions raise and lower the pen. Use them with parameters to draw or move to the points specified by the parameters.

SYNTAX: *PU term*
or
PU X-coordinate, Y-coordinate (, . . .) *term*

and

PD term
or
PD X-coordinate, Y-coordinate (, . . .) *term*

PARAMETERS:

Parameter	Format	Range	Default
X- and Y-coordinates	integer	$\pm(2^{26}-1)$ current units	none

EXPLANATION: The PU and PD instructions are interpreted as listed in the following paragraphs.

1. X- and Y-coordinates. When parameters are included, PU and PD are interpreted as forms of one of the plot instructions, PA and PR, depending on which (PA or PR) was the most recent plot instruction. (If no plot instruction has been previously executed, PA is assumed.)

   For example, instead of drawing a line by executing both a PD (without parameters) and a PA instruction (with parameters), you can issue just a PD instruction with the absolute coordinates for the line. Replacing PR with PD (with parameters) is similar, except that the parameters are interpreted to be relative coordinates.

   For complete information on how the parameters are interpreted, refer to The Plot Absolute Instruction, PA, and The Plot Relative Instruction, PR, located next in this chapter.

2. No Parameters. The PU instruction without parameters raises the pen without moving the pen to a new location. Similarly, the PD instruction without parameters lowers the pen without moving the pen to a new location. However, the pen is lowered only if it is within the window and is not in the up portion of a dashed line.

The Plot Absolute Instruction, PA

USES: The PA instruction establishes absolute plotting mode and moves the pen to the specified point(s). You can use this instruction with PU to move to a point with the pen up, or with PD to plot to a point with the pen down.

SYNTAX: *PA* X-coordinate, Y-coordinate (, . . .) *term*
or
PA term

PARAMETERS:

Parameter	Format	Range	Default
X- and Y-coordinates	integer	$\pm(2^{26}-1)$ current units	none

EXPLANATION: The PA instruction is interpreted as listed in the following paragraphs.

1. X- and Y-coordinates. The X- and Y-coordinates specify the absolute position in the plotting area to which the pen will move. Both coordinates must be specified as a pair; however, you may specify as many coordinate pairs as you wish. When you include more than one coordinate pair, the pen moves to each point in the order given, using the current up/down status. The up/down status is based on the most recent PU or PD instruction. This concept is discussed in the next section, titled Plotting with PA, PU, and PD.

   The coordinates are interpreted as current units: user units if scaling is on or plotter units if scaling is off.

2. No Parameters. The PA instruction without parameters establishes absolute plotting mode as long as PD or PU forms of the PA instruction are subsequently executed. Otherwise, the PA instruction without parameters is ignored.

The following table summarizes the possible error conditions for the PA instruction.

Condition	Error	Plotter Response
parameters out-of-range	3	instruction ignored
odd number of coordinates	2	each coordinate pair is executed; the extra coordinate is ignored
lost mode	none	conforms to the conditions described under Relationship of Plotting Instructions and Graphics Limits in this chapter

Plotting with PA, PU, and PD

As mentioned previously, the PA instruction moves the pen to the specified point(s) using the current pen up/down status. At power-on, after a front-panel reset, and after an IN or DF instruction, the pen is up. Therefore, the following string of instructions will cause the pen to move to the point 2000,2000 without drawing a line.

IN; PA 2000,2000;

Now, in order to draw from that point to another point, you must put the pen down, as follows. A line is drawn from 2000,2000 to 5000,5000.

IN;PA2000,2000;PD5000,5000;

Note that the PD form of PA was used. This is equivalent to issuing these separate instructions:

IN;PA 2000,2000;PD;PA 5000,5000

You can also insert PD or PU within a list of X,Y coordinate pairs. When you do this, commas or spaces are required between numeric parameters, but are optional before and after the two-letter mnemonics. You must include the required terminator after the last entry. In the following examples, the comma is used to indicate the optional separators and the semicolon is used to indicate the required terminator.

PA , PD , X_1 , Y_1 , PU , X_2 , Y_2 , PD , X_3 , Y_3 , ;

OPTIONAL

PU , X_1 , Y_1 , PD , X_2 , Y_2 , X_3 , Y_3 , PU , ;

OPTIONAL

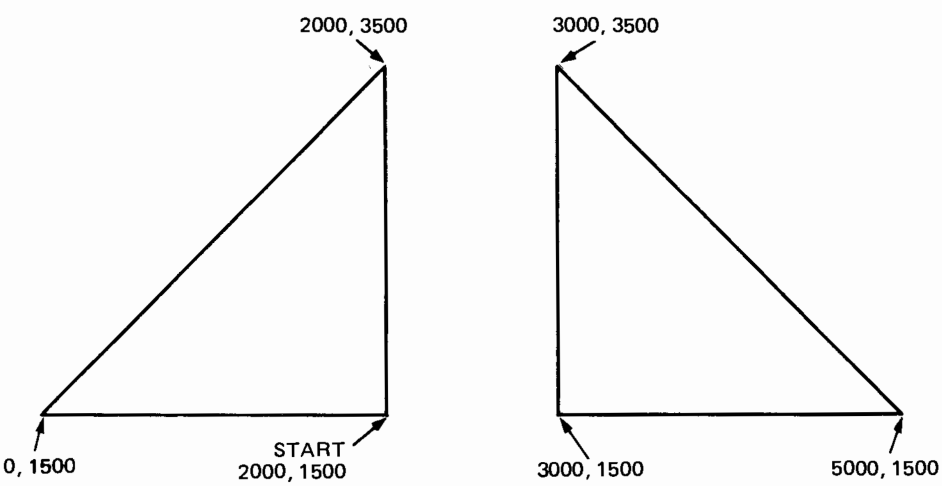
PD , X_1 , Y_1 , X_2 , Y_2 , X_3 , Y_3 , ;

OPTIONAL

The following string of HP-GL instructions causes the plotter to draw two triangles and then move to the point 5000,5000 with the pen up.

Send:

```
                            OPTIONAL
LINE 1.   IN;SP1;
LINE 2.   PA2000,1500,PD,0,1500,2000,3500,2000,1500;
          PU3000,1500;
LINE 3.   PAPD5000,1500,3000,3500,3000,1500;
          PU,5000,5000,;
                            OPTIONAL
```



The Plot Relative Instruction, PR

USES: The PR instruction establishes relative plotting mode and moves the pen to the specified point(s), relative to the current pen position. You can use this instruction with PU to move to a point with the pen up, or with PD to plot to a point with the pen down.

SYNTAX: *PR* X-increment, Y-increment (, . . .) *term*
or
PR *term*

PARAMETERS:

Parameter	Format	Range	Default
X- and Y-increments	integer	$\pm(2^{26}-1)$ current units	none

EXPLANATION: The PR instruction is interpreted as listed in the following paragraphs.

1. X- and Y-increments. The X- and Y-increments specify the position in the plotting area to which the pen will move, relative to the current pen position. The X-increment specifies the number of units for the pen to move in the direction of the X-axis; the Y-increment specifies the number of units for the pen to move in the direction of the Y-axis. The signs of the increments determine the relative direction in which the pen moves; a positive value moves the pen in a positive direction, and a negative value moves the pen in a negative direction.

   Both increments must be specified as a pair; however, you may specify as many increment pairs as you wish. When you include more than one increment pair, the pen moves to each point in the order given, using the current up/down status. The up/down status is based on the most recent PU or PD instruction. This concept is discussed in the next section, titled Plotting with PR, PU, and PD.

   The increment parameters are interpreted as current units: user units if scaling is on or plotter units if scaling is off.

2. No Parameters. The PR instruction without parameters establishes relative plotting mode as long as PD or PU forms of the PR instruction are subsequently executed. Otherwise, the PR instruction without parameters is ignored.

The following table summarizes the possible error conditions for the PR instruction.

Condition	Error	Plotter Response
parameters out-of-range	3	instruction ignored
odd number of increments	2	each increment pair is executed; the extra increment is ignored
lost mode	none	conforms to the conditions described under Relationship of Plotting Instructions and Graphics Limits in this chapter

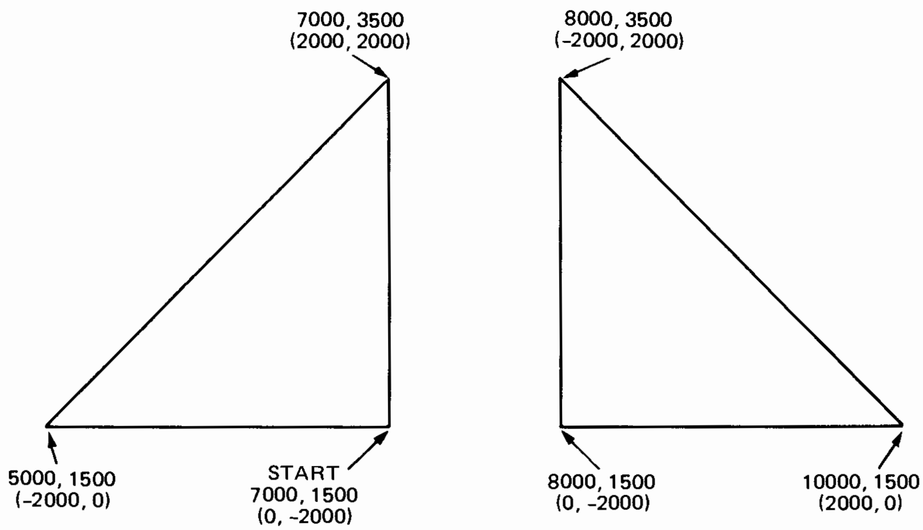
Plotting with PR, PU, and PD

The PR instruction operates with PU and PD in exactly the same manner as described in this section under The Plot Absolute Instruction, PA. The only difference is that PR moves are relative, whereas PA moves are absolute.

The following string of HP-GL instructions draws triangles that are identical to the ones previously drawn with the PA instruction. These triangles are drawn adjacently to the ones previously drawn. The numbers in parentheses are the X,Y increments of the PR instructions. The numbers without parentheses are the plotter unit coordinates of the vertices.

Send:

```
LINE 1.  IN;SP1;
LINE 2.  PA7000,1500;PRPD,-2000,0,2000,2000,0,-2000;
         PAPU8000,1500;
LINE 3.  PRPD2000,0,-2000,2000,0,-2000;PAPU10000,5000;
```



Relationship of Plotting Instructions and Graphics Limits

For most plotting situations, the pen will probably draw as you expect it to. However, depending on what your graphics limits (windows) are and the X,Y coordinates you specify, the pen might not always move and plot. This section describes the situations that affect the movement of the pen. It applies directly to these plotting instructions: PA, PR, RA, RR, EA, ER, CI, AA, AR, WG, EW, EP, FP, CP, LB, PB, UC, XT, and YT.

Types of X,Y Coordinates

The pen motion caused by plotting instructions depends on the type of X,Y coordinates specified. There are three types of X,Y coordinates:

1. Inside window area
2. Outside window area and in-range
3. Out-of-range

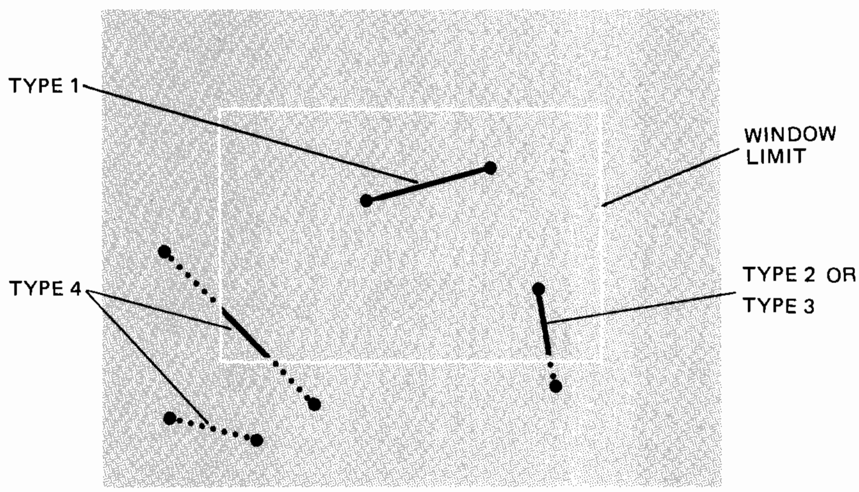
Out-of-range coordinates are defined as either X or Y, or both, being less than $-2^{26}+1$ or greater than $2^{26}-1$. Specifying out-of-range coordinates always sets error 3 (bad parameter) and results in the plotting instruction being ignored.

The other two types of coordinates (inside window and outside window) are discussed in the next section. Remember that plotting occurs only within the currently defined window. At power-on, the currently defined window is the hard-clip limits. However, you can define smaller windows using The Input Window Instruction, IW. Refer to Chapter 3.

Plotting Inside and Outside of Windows

There are, in general, four types of plotting vectors that can be developed from a given point to some new point, as follows:

	Last Point		New Point
1.	Inside window area	to	inside window area
2.	Inside window area	to	outside window area
3.	Outside window area	to	inside window area
4.	Outside window area	to	outside window area



In type 1, the pen moves from the last point to the new point with the pen up or down as programmed.

In type 2, the pen moves from the last point toward the new point and stops where the vector between these points intersects the window limit. The pen up/down state is as programmed until the intersection point is reached. Then, the pen lifts and the OUT OF LIMIT light comes on. The pen remains in this position until a plotting instruction is received that brings the pen back inside the window (refer to type 3).

In type 3, the pen movement depends on whether the pen is programmed to be up or down. If the pen is programmed to be up, it moves directly to the new point. If the pen is programmed to be down, it moves, up, to the point where the vector between the last point and the new point intersects the window limit. When the pen reaches this point, the OUT OF LIMIT light goes out and the pen assumes its programmed down position. The pen then continues plotting to the new point.

Plotting

In type 4, no pen movement occurs unless any of the vector between the last and new points intersects the window area. When no pen movement occurs, the **OUT OF LIMIT** light remains on, but the X- and Y-coordinates of the current pen position are updated.

If part of the vector is in the window area, the pen moves, up, to the first intersection point with the window. The **OUT OF LIMIT** light goes out and the pen then moves as programmed (up or down) to the second intersection point, where it lifts and stops moving. Thus, this situation is a combination of type 3, followed by type 1, followed by type 2. If none of the vectors lie within the window area, the X- and Y-coordinates of the current pen position are updated even though the pen does not move.

Lost Mode

There is a condition known as lost mode, which will never be encountered under normal plotting situations. However, you should be aware of it. Lost mode is established only when scaling is on and the plotter-unit equivalents of the specified user-unit coordinates exceed the range of $\pm(2^{29}-1)$. (Refer to Appendix C for a method of converting from user units to plotter units.) Lost mode is impossible when scaling is off, because when the plotter-unit coordinates exceed $\pm(2^{26}-1)$, error 3 (bad parameter) is generated and the plotting instruction is ignored.

NOTE: Remember that relative moves (as with the PR, AR, RR, and ER instructions) are cumulative. Therefore, lost mode is entered when the cumulative moves exceed the range specified above. ■

When the plotter is in the lost mode, the AA, AR, CI, CP, LB, PB, PR, UC, XT, YT, RA, RR, EA, ER, WG, EW, FP, and EP instructions are all ignored until lost mode is cleared. If a PD or PU instruction is received while in lost mode, the programmatic pen status is updated, but the pen remains physically up until lost mode is cleared.

Any of the following will clear lost mode:

- executing a PA instruction with in-range parameters
- executing an IN or RO instruction
- pressing the following buttons on the front panel:
  RESET, P1, P2, ENTER P1, ENTER P2, and **AXIS ALIGN.**
- any joystick motion
- transition to Not-Ready state (with the NR instruction or the front-panel **VIEW** button)

NOTE: If PA coordinates are in-range, but outside of the window, lost mode is cleared, but the pen does not move. ■

Plotting with Variables

In many plotting applications it is necessary to plot using variables rather than fixed numbers to define the X- and Y-coordinate values. The values of all HP-GL instruction parameters have the same restrictions (integers or decimals in a valid range) when sent as variables as when sent as literals (fixed number). The terminators and delimiters of HP-GL instructions must be sent to the plotter too. The method of defining output format and variable precision varies from computer to computer. Refer to your computer manual for the appropriate format statement that may be needed in your program.

The following example illustrates the use of variables in plotting. Quotation marks are used by many computers to delimit the literal characters that are to be sent. Note the comma in line 6 which is part of the HP-GL statement and is sent to the plotter. Here it is sent as a literal in quotes. In this plotter, a space may be substituted for the literal comma shown in quotes. If your computer automatically includes spaces between the variables X and Y, these spaces will delimit the coordinate values and a literal comma or space will not be required. The variables X and Y are automatically rounded to integers by the plotter. Unless you are writing software to be compatible with plotters such as the HP 9872, it is not necessary to add a statement to assure they are sent as integers by your computer.

```
  LINE 1.   "IN;SP1;IP-5000,5000,0,10000;"
  LINE 2.   "SC0,25000,0,25000;"
■ LINE 3.   FOR T=0 TO 2*π STEP π/20
■ LINE 4.   X=4.5*1000*COS(T)+12500
■ LINE 5.   Y=4.5*1000*SIN(T)+12500
★ LINE 6.   "PA",X,",",Y,";PD;"
■ LINE 7.   NEXT T
  LINE 8.   "PU;"
```

This statement causes the plotter to move to the X, Y coordinate defined by the X- and Y-variables and lower the pen. The HP-GL mnemonics, delimiters and terminators are sent as literals in quotes on some computers. The delimiter "," is included here to delineate the variables X and Y.

The Automatic Pen Pickup Instruction, AP

USES: The AP instruction activates selected automatic pen operations. Use the instruction to disable the default automatic pen operations when the digitizing sight is in use and to enable one or more of the operations when plotting is resumed.

SYNTAX: *AP* n *term*
or
AP *term*

PARAMETERS:

Parameter	Format	Range	Default
n	integer	*	7

*see table on following page

EXPLANATION: The parameter n is the decimal equivalent of a 16-bit binary integer. The three least signicant bits are used to form the pen control word and to turn on or off the automatic pen operations.

The instructions AP; and AP7; turn on all automatic pen operations and are equivalent to setting all three bits of the pen control word to 7.

The automatic pen operations and the bits which enable them are defined in the following table.

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Set Bit	Decimal Value	Automatic Pen Operation
Bit 0	1	Lift pen if it has been down without motion for the allotted time.
Bit 1	2	Put pen away if it has been without motion for the allotted time. If unable to put pen away, then lift pen (if down).
Bit 2	4	Do not lift pen selected by an SP instruction until the new pen is required to draw; otherwise, retrieve pen immediately.
Bits 0, 1, 2	7	Enable all three of the above operations.

NOTE: Any pen select operation invoked from the front panel is always executed upon recognition regardless of the state of the pen control word. ■

If a fiber-tip (type 1) or a roller-ball (type 2) carousel is installed, the pen can remain motionless for 65 seconds before automatic lift or storage occurs.

If a drafting pen (type 3) carousel is installed or if no carousel is installed, the pen can remain motionless for 10 seconds before automatic lift or storage occurs.

The plotter has no way to detect a pen that is inserted directly into the pen holder until a subsequent pen change activity occurs to update the existing pen map. Therefore, automatic pen operations will not occur until the inserted pen is detected and the pen map is updated. For pen mapping information, refer to Pen and Pen Carousel Operation, Chapter 2, and to the GP and SG instructions in this chapter.

The Acceleration Select Instruction, AS

USES: The AS instruction specifies pen acceleration through program control. The default 4 g acceleration is suitable for all recommended pen and media combinations. Slowing acceleration may improve line qualiity if heavier than recommended media must be used.

SYNTAX: *AS* pen acceleration (,pen number) *term*
or
AS *term*

PARAMETERS:

Parameter	Format	Range	Default
pen acceleration	integer	1-4	4
pen number	integer	1-8	all pens

Plotting

EXPLANATION: Use of the AS instruction without parameters (AS ;) sets acceleration to 4 for all pens, regardless of the carousel type. Following is an explanation of the parameters.

1. Pen acceleration. This parameter is interpreted as g's and applies to all pens unless the optional pen number parameter is specified.

2. Pen number. The optional pen number parameter applies the acceleration to the specified pen only. When this parameter is not used, the acceleration applies to all pens.

If you insert a different type of carousel, acceleration defaults to 4 for all pens. Refer to Pen and Pen Carousel Operation, Chapter 2.

The following table summarizes the error conditions for the AS instruction.

Condition	Error	Plotter Response
pen acceleration parameter > 4	none	parameter interpreted as 4
pen number parameter > 8	none	instruction ignored
more than 2 parameters	2	extra parameter ignored
negative or zero parameter	3	instruction ignored

The Force Select Instruction, FS

USES: The FS instruction sets the pen force. Use the instruction to optimize pen life and line quality for each pen and media combination.

SYNTAX: *FS* pen force (,pen number) *term*
or
FS *term*

PARAMETERS:

Parameter	Format	Range	Default
pen force	integer	1-8	*18, 50, 10
pen number	integer	1-8	all pens

*for fiber-tip, roller-ball, and drafting pens, respectively

EXPLANATION: The following paragraphs list further details of each parameter.

1. Pen force. The pen force parameter specifies the force of the pen tip on the paper in increments of 8 grams as follows:

1 = 10 grams	5 = 42 grams
2 = 18 grams	6 = 50 grams
3 = 26 grams	7 = 58 grams
4 = 34 grams	8 = 66 grams

2. Pen number. The optional pen number parameter specifies the pen to which the force applies. When this parameter is omitted, the force applies to all pens.

Plotting

The instruction without parameters (FS ;) sets pen force for all pens to the default value of the current carousel type: fiber tip (18 grams); roller ball (50 grams); drafting (10 grams).

If you insert a different type of carousel, the force value for all pens defaults to the value determined by the new carousel type. Refer to Pen and Pen Carousel Operation, Chapter 2.

The following table summarizes the error conditions for the FS instruction.

Condition	Error	Plotter Response
pen force parameter > 8	none	parameter interpreted as 8
pen number parameter > 8	none	instruction ignored
more than 2 parameters	2	extra parameter ignored
negative or zero parameter	3	instruction ignored

The Velocity Select Instruction, VS

USES: The VS instruction specifies pen speed. Use the instruction to optimize pen life and line quality for each pen and media combination. You can create a slightly thicker line on any media by slowing the pen speed.

SYNTAX: *VS* pen velocity (,pen number) *term*
or
VS *term*

PARAMETERS:

Parameter	Format	Range	Default
pen velocity	integer	1-60	50, 60, 15*
pen number	integer	1-8	all pens

*for fiber-tip, roller-ball, and drafting pens, respectively

EXPLANATION: The following paragraphs list further details of each parameter.

1. Pen velocity. The pen velocity parameter specifies pen speed in centimetres per second (cm/s). The selected velocity applies only when the pen is down. Pen movement with the pen up is executed at 60 cm/s.
2. Pen number. The optional pen number parameter applies the velocity to the specified pen only. When this parameter is omitted, the velocity applies to all pens.

The instruction without parameters (VS ;) sets the pen speed for all pens to the default value of the current carousel type: fiber tip (50 cm/s); roller ball (60 cm/s); drafting (15 cm/s).

If you insert a different type of carousel, the pen velocity for all pens defaults to the value of the new carousel type. Refer to Pen and Pen Carousel Operations, Chapter 2.

NOTE: Two-digit velocities cannot be fully displayed by the single-digit front-panel **PEN SPEED** display. Therefore, two-digit velocities are displayed only to the nearest tens digit of the actual velocity. ■

The following table summarizes the error conditions for the VS instruction.

Condition	Error	Plotter Response
pen velocity parameter > 60	none	parameter interpreted as 60
pen number parameter > 8	none	instruction ignored
more than 2 parameters	2	extra parameter ignored
negative or zero parameter	3	instruction ignored

The Group Pen Instruction, GP

USES: The GP instruction specifies which pens belong to a designated group. Use this instruction with pens of the same color and width to extend the effective writing distance beyond the limits of one pen.

SYNTAX: *GP* (group number (,pen number (,number of pens (,length)))) *term*

PARAMETERS:

Parameter	Format	Range	Default
group number	integer	1-8	all groups
pen number	integer	1-8	specified group number
number of pens	integer	1-8*	1
length	integer	1-50 000	100

*pen number + number of pens must be ≤ 9

EXPLANATION: The parameters used in the GP instruction are explained below.

1. Group number. The first parameter designates the group to be defined and is used in conjunction with the SG instruction.

2. Pen number. The second parameter is the number of the *first* pen in the group. When a pen number is omitted, a group will consist of one pen whose number is the same as the group number.

3. Number of pens. The third parameter is the total number of pens in the group. The group begins with the pen specified in parameter two and increases sequentially by one until the total number equals the third parameter.

   The pen sequence will stop with pen 8, even if the number of pens in the sequence is less than the number specified in parameter three. (See the examples below.)

   If the number of pens in a group is omitted, the group will default to one pen.

4. Length. The fourth parameter is the length in metres at which the pens will switch. Numbers larger than 50 000 will be clipped to 50 000. When omitted, the parameter defaults to 100.

At power-up, or when the plotter is initialized (IN ;), or when the GP instruction is issued without parameters (GP ;), *all* groups will consist only of the corresponding pen. For example, GP ; is equivalent to GP 1 , 1 ; GP 2 , 2 ; GP 3 , 3 ; etc.

Specifying one parameter causes *only* the group specified to point to one pen. For example, GP 1 ; is equivalent to GP 1 , 1 , 1 ;.

Specifying two parameters maps a group to the pen specified. For example GP 2 , 3 ; means that group 2 consists only of pen 3.

Below are some additional examples.

GP 2 , 7 , 4 ;	Group 2 will contain pens 7 and 8 and will switch every 100 metres (default).
GP 9 , 2 , 3 ;	NOP'd.
GP 1 , 9 , 3 ;	NOP'd. An SG 1 ; instruction would also be NOP'd.
GP 2 , 2 , 2 , 5000 ;	Group 2 will contain pens 2 and 3 and and will switch every 5000 metres.
GP 1 , 1 , 1 , 300 ;	Group 1 will contain only pen 1, so the switch distance is irrelevant.

None of the above examples will cause any errors. However, if any of the parameters of the GP instruction are less than one, the instruction is ignored and the group being defined will not change.

The Select Pen Group Instruction, SG

USES: The SG instruction enables the plotter to alternate between predesignated pens within a group. Use this instruction with the GP instruction to extend the effective writing distance beyond the limits of one pen.

SYNTAX: *SG* group number *term*

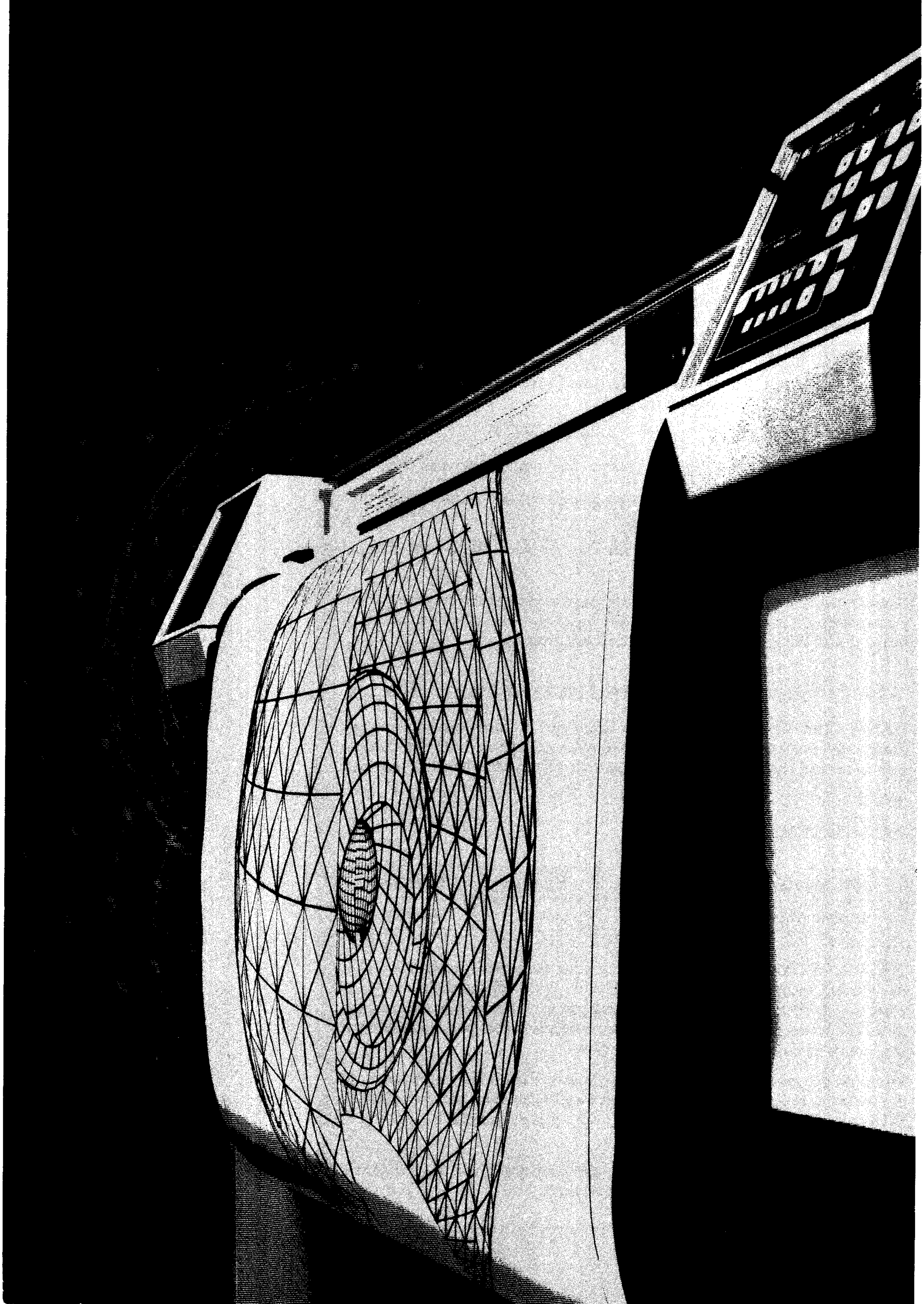
PARAMETERS:

Parameter	Format	Range	Default
group number	integer	0 to 8	0 (no pen)

EXPLANATION: The parameter applies to a pen group previously designated with the GP instruction. The SG instruction selects the pen or pens which belong to this corresponding group. When the group contains more than one pen, the plotter will sequentially select pens according to the length parameter of the GP instruction.

When no groups have been previously defined with a GP instruction, the SG instruction is equivalent to a Select Pen Instruction, SP. Accordingly, SG 0 ; and SG ; are equivalent to SP 0 ;: any pen in the pen holder will be placed in the carousel.

If an out-of-range parameter is used, the instruction will be ignored. A parameter less than 0 will also generate error 3.



Chapter 5
Circles, Arcs, and Polygons

What You'll Learn in This Chapter

In this chapter you will learn about defining, outlining, and filling polygons. Some polygons can be created with specific instructions; for example, the circle instruction, the wedge instructions, and the rectangle instructions. In addition, you can define your own polygons using the polygon mode instruction.

Besides creating polygons, you will also learn how to outline them and how to fill them with shading patterns. These shading patterns include patterns offered by the plotter and patterns that you can design.

Finally, you will learn about the polygon buffer and how to determine the proper memory allocation in this buffer for your polygons.

HP-GL Instructions Covered

CT	The Chord Tolerance Instruction
CI	The Circle Instruction
AA	The Arc Absolute Instruction
AR	The Arc Relative Instruction
FT	The Fill Type Instruction
UF	The User-Defined Fill Type Instruction
PT	The Pen Thickness Instruction
WG	The Fill Wedge Instruction
EW	The Edge Wedge Instruction
RA	The Fill Rectangle Absolute Instruction
RR	The Fill Rectangle Relative Instruction
EA	The Edge Rectangle Absolute Instruction
ER	The Edge Rectangle Relative Instruction
PM	The Polygon Mode Instruction
EP	The Edge Polygon Instruction
FP	The Fill Polygon Instruction

Terms You Should Understand

Absolute Coordinates — coordinates specified relative to the origin (0,0). The current position of the pen has no effect on the destination of the pen.

Relative Coordinates — coordinates specified relative to the current pen position; in other words, coordinates specified as increments. The destination of the pen thus depends on the current position of the pen.

Arc — a portion of the circumference of a circle.

Wedge — the combination of an arc and the lines connecting the arc endpoints to the circle center. Wedge is equivalent to the geometric term "sector."

Chord — a straight line joining two points on an arc or the circumference of a circle.

Chord Tolerance — the allowable deviation from a perfectly smooth circle or arc. A perfectly smooth circle or arc is composed of an infinite number of chords. The human eye cannot discern the individual chords; in fact, when circles are small, even as few as 72 chords per circle produces an apparently smooth circle. The chord tolerance determines the number of chords, and thus the smoothness, of a circle or arc. The chord tolerance can be defined in the CT instruction to be an angle in degrees, or a deviation distance in current units.

Polygon — any shape constructed of a number of vertices in such a way that lines connecting the vertices meet to form a closed area. Simple shapes such as circles, rectangles, and wedges are polygons. More complex shapes such as block letters can also be polygons. In this chapter, polygon most often refers to the shapes that are defined in polygon mode.

Subpolygon — the name applied to a polygon when it is being defined along with other polygons while in polygon mode. For example, the block letter O is actually defined in polygon mode as two subpolygons: the outside circle and the inside circle.

Polygon Mode — a mode established by the PM instruction for defining polygons. In this mode, certain HP-GL instructions are not executed so that they can be used to define the vertices of the polygon.

Polygon Buffer — a portion of the plotter's "memory" that is used to store the vertices of the polygon that is currently being defined. The polygon can be one defined by the RA, RR, WG, or PM instructions.

Vertex — a coordinate point that defines part of the shape of a polygon. Each corner of a rectangle, for example, is a vertex.

Edge — the outline of a polygon.

Fill — the shading pattern used in a polygon.

Cross-hatch — a type of fill, or shading pattern, where a set of parallel lines is drawn at a 90-degree angle over another set of parallel lines.

The Chord Tolerance Instruction, CT

USES: The CT instruction establishes whether the chord tolerance parameter of all subsequent AA, AR, CI, EW, and WG instructions is interpreted as degrees or as a deviation distance. Since these instructions assume degrees mode, you only need to execute CT if you want to change to deviation distance mode, or if you have previously specified deviation distance mode but want to return to degrees mode.

The degrees mode and deviation distance mode govern the smoothness of arcs and circles. In degrees mode, using the same chord tolerance parameter causes arcs and circles of any radius to be drawn with the same number of chords. As the radius increases, therefore, the arc or circle becomes less smooth. In deviation mode, using the same chord tolerance parameter causes arcs and circles of any radius to be drawn with the same smoothness, but with different numbers of chords. In this case, large arcs and circles will require more time to draw because more chords are used.

SYNTAX: *CT* n *term*
or
CT *term*

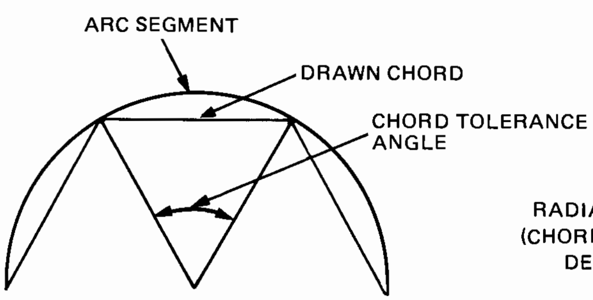
PARAMETERS:

Parameter	Format	Range	Default
n	integer	0 or 1	0

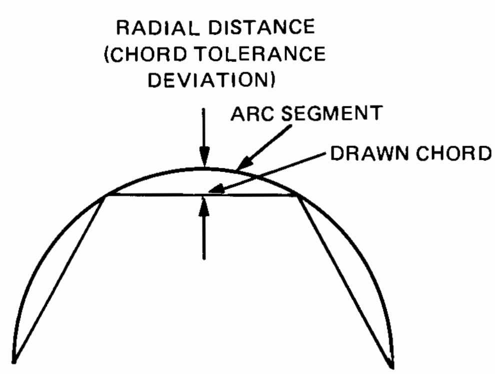
EXPLANATION: A CT instruction without parameters (CT ;) is equivalent to a CT instruction with the parameter 0. The two possible parameter values are listed next.

1. n = 0. A parameter of 0 or a CT instruction without parameters sets degrees mode. When degrees mode is set, the chord tolerance parameter is subsequently interpreted as degrees and sets the maximum angle subtended by a chord that is drawn to represent an arc segment. The actual angle subtended may be slightly smaller in order to divide the circle or arc into equal segments so that all chords are the same length.
2. n = 1. A parameter of 1 sets deviation mode. When deviation mode is set, the chord tolerance parameter is subsequently interpreted as current units and sets the maximum radial distance permitted between the chord drawn and the arc segment it represents. The actual radial distance may be slightly less since the plotter will divide the arc or circle to create chords of equal length.

The two modes are illustrated below.



DEGREES MODE



DEVIATION MODE

Circles & Polygons

The tolerance mode (degrees or deviation) remains in effect until another valid CT instruction is executed or the plotter is initialized or set to default conditions.

A parameter of other than 0 or 1 will set error condition 3. The instruction will be ignored and the chord tolerance parameter will not change.

The Circle Instruction, CI

USES: The CI instruction draws a circle of a specified radius and chord tolerance. All computations are internal to the plotter to reduce computer overhead.

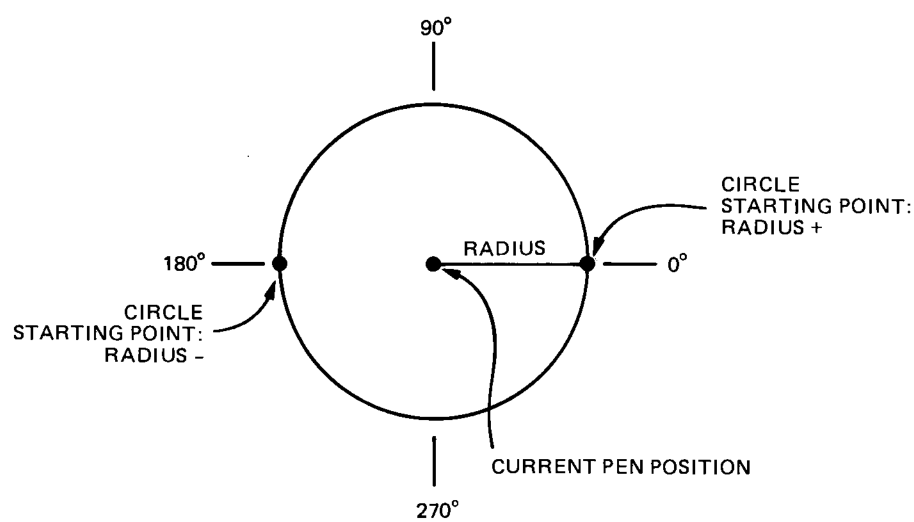
SYNTAX: *CI* radius (,chord tolerance) *term*

PARAMETERS:

Parameter	Format	Range	Default
radius	integer	$\pm(2^{26}-1)$ current units	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

EXPLANATION: The radius and chord tolerance parameters are interpreted as described in the following paragraphs and illustrations.

1. Radius. The radius determines the size of the circle. Its sign defines the starting point of the circle: a circle with a positive radius starts at the 0-degree point; a circle with a negative radius starts at the 180-degree point. The current pen position is the center of the circle. If scaling is off, the radius is in plotter units. If scaling is on, the radius is in user units. If user units are not the same size in the X- and Y-directions, ellipses will be drawn. Refer to The Scale Instruction, SC, in Chapter 3.

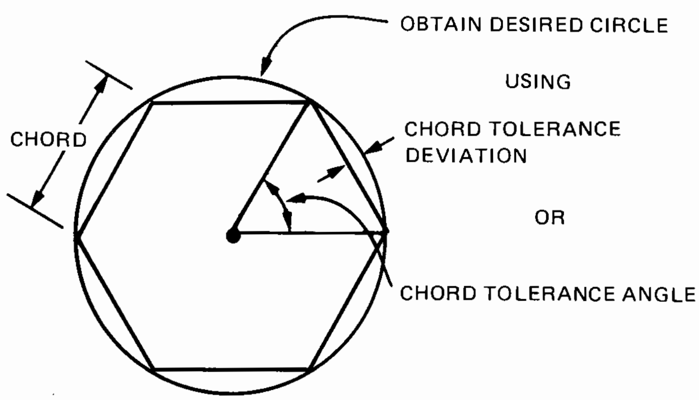


Circles & Polygons

2. Chord Tolerance. A circle is actually made up of a series of straight line segments, or chords, as shown below. Increasing the number of chords (by decreasing the chord tolerance parameter) increases the smoothness of the circle. However, this also increases the length of time required to draw the circle.

   The chord tolerance parameter is interpreted as either degrees or a deviation distance, depending on whether degrees mode or deviation mode has been set (by the CT instruction or default conditions). Refer to The Chord Tolerance Instruction, CT, in this chapter. If degrees mode is set, the angle is interpreted as modulo 360.

   The sign of this parameter is ignored. The default chord tolerance is 5 degrees, which causes the circle to be made up of 72 chords. The first example after the error conditions table illustrates the effects of different chord tolerance angles.



The CI instruction includes an automatic pen down feature. When a CI instruction is received, the pen lifts (if it was down), moves from the center of the circle to the starting point on the circumference, lowers the pen, draws the circle, then returns, pen up, to the center of the circle. After drawing the circle, the pen assumes the pen state (up or down) that was in effect prior to the CI instruction. To avoid drawing lines to the center of the circle, move to and away from the circle's center with the pen up.

Each chord of the circle is drawn using the currently defined line type. Refer to The Line Type Instruction, LT, in Chapter 6. Since each chord is small, patterns adjusted to chord endpoints are hard to distinguish. Therefore, use positive parameters in the LT instruction when drawing circles.

The following table summarizes the error conditions for the CI instruction.

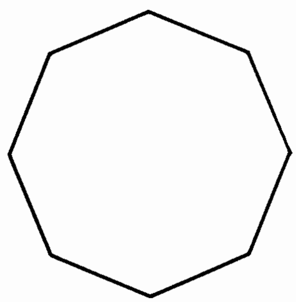
Condition	Error	Plotter Response
no parameters	2	instruction ignored
more than 2 parameters	2	executed with first 2 parameters
parameter out-of-range	3	instruction ignored
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, in Chapter 4.

Example — Effects of Chord Tolerance on Circle Smoothness

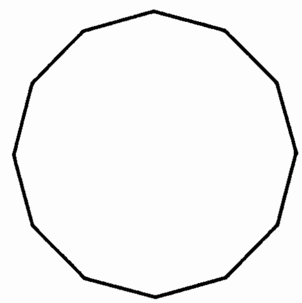
The following instructions, when sent to the plotter using your computer's output statements, draw circles with the same radius, but with different chord tolerance angles. Note that the circle becomes smoother as the chord tolerance parameter decreases.

Send:

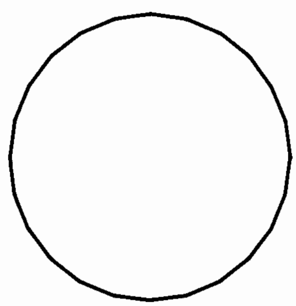
```
LINE 1.   IN;SP1;IP10000,0,15000,5000;
LINE 2.   SC-100,100,-100,100;
LINE 3.   PA-50,50;CI30,45;
LINE 4.   PA50,50;CI30,30;
LINE 5.   PA-50,-50;CI30,15;
LINE 6.   PA50,-50;CI30,5;
```



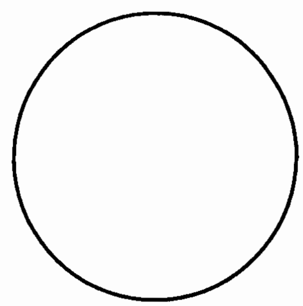
45 DEGREE CHORD TOLERANCE



30 DEGREE CHORD TOLERANCE



15 DEGREE CHORD TOLERANCE



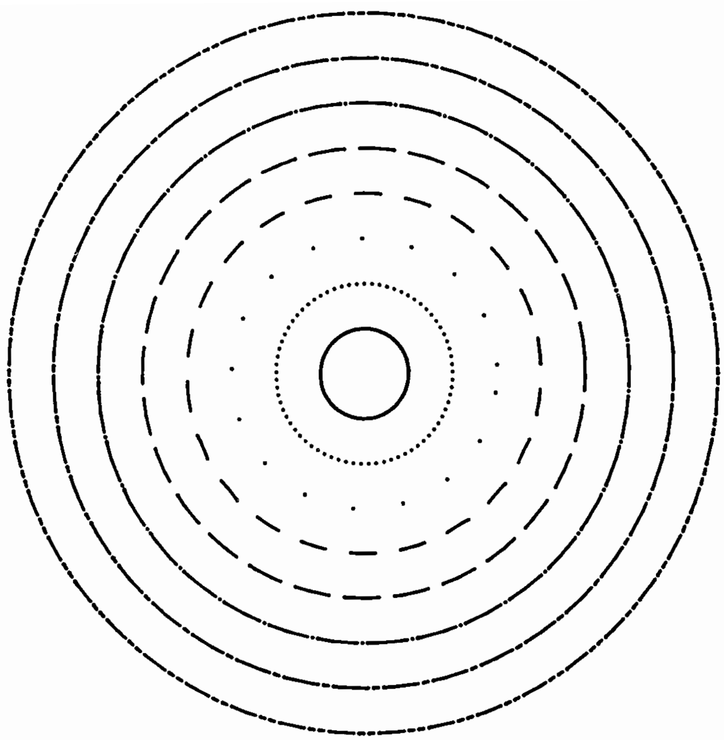
5 DEGREE CHORD TOLERANCE

Example — Drawing Circles with Different Radii and Line Types

The following example draws circles with different line types, radii, and starting points. It includes the SC instruction to establish user units. Note that the user units are equal in the X- and Y-directions.

Send:

```
LINE 1.   IN;SP1;IP-15000,-5000,-10000,0;
LINE 2.   SC-100,100,-100,100;
LINE 3.   PA0,0;LT;CI10,5;LT0;CI-20,5;LT1;CI30,5;
LINE 4.   LT2;CI-40,5;LT3;CI50,5;LT4;CI-60,5;LT5;
          CI70,5;LT6;CI80,5;
```



The Arc Instructions, AA and AR

USES: The AA and AR instructions draw an arc based on the present pen position and the specified center point. Use the arc absolute instruction, AA, to specify the center point in absolute coordinates. Use the arc relative instruction, AR, to specify the center point in relative coordinates. The AA and AR instructions are identical except for the type of coordinates used. For in-depth discussions of absolute and relative plotting, refer to The Plot Absolute Instruction, PA, and The Plot Relative Instruction, PR, in Chapter 4.

SYNTAX: *AA* X-coordinate, Y-coordinate, arc angle (,chord tolerance) *term*
or
AA *term*

AR X-increment, Y-increment, arc angle (,chord tolerance) *term*
or
AR *term*

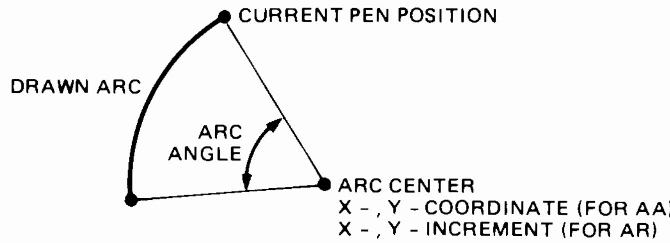
PARAMETERS:

Parameter	Format	Range	Default
X- and Y-coordinates; or X- and Y-increments	integer	$\pm(2^{26}-1)$ current units	none
arc angle	decimal	$\pm(2^{26}-1)$ degrees	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

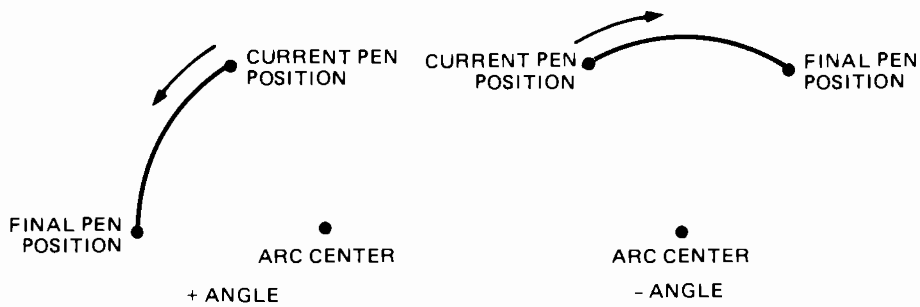
Circles & Polygons

EXPLANATION: The parameters are interpreted as described in the following paragraphs and illustrations.

1. X,Y coordinates (for AA) or X,Y increments (for AR). These parameters specify the center of the arc in plotter units if scaling is off, or in user units if scaling is on. The center is that point from which the radii would be drawn to the endpoints of the arc. However, the AA and AR instructions only draw the arcs, not the radii. Refer to the following illustrations.



2. Arc Angle. The arc angle is the angle through which the arc is drawn. A positive angle draws counterclockwise (CCW) from the current pen position, and a negative angle draws clockwise (CW), as shown below.



3. Chord Tolerance. The chord tolerance parameter is interpreted as either degrees or as a deviation distance, depending on whether degrees mode or deviation mode has been set (by the CT instruction or default conditions). Refer to The Chord Tolerance Instruction, CT, in this chapter.

   The sign of this parameter is ignored. The default chord tolerance is 5 degrees. As discussed under the CI and CT instructions, increasing the number of chords (by decreasing the chord tolerance parameter) generates a smoother arc.

Arcs are drawn starting at the current pen position using the current pen state (up or down) and line type. If no pen state has been commanded since initialization, pen up is assumed. As with the CI instruction, you will find that positive parameters in the LT instruction produce the expected line patterns. After the arc has been drawn, the pen position will remain at the end of the arc, rather than returning to the beginning.

The arc center can be located on or off the plotting surface. The plotter then draws the arc according to the definitions given for plotting under the PA instruction in Chapter 4. Remember that AR moves add to the present position. If cumulative moves result in a pen position that exceeds the range of -2^{29} to $2^{29}-1$, the plotter enters the lost mode.

The following table summarizes the error conditions for the AA and AR instructions.

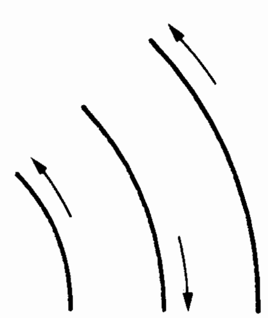
Condition	Error	Plotter Response
no parameters	2	instruction ignored
more than 4 parameters	2	executed with first 2 parameters
parameter out-of-range	3	instruction ignored
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, and The Plot Relative Instruction, PR, in Chapter 4

Example — Using the AA Instruction

The following instructions illustrate the effects of changing the distance to the center point while keeping the same arc angle. They also show the effect of changing the sign of the angle. Note that the pen remains down at the end of each arc, so you must lift the pen with the PU instruction in order to move to the beginning of the next arc without drawing a line.

Send:

```
LINE 1.   IN;SP1;IP-10000,-5000,-5000,0;
LINE 2.   PA-6000,-4000;
LINE 3.   PD;AA-8000,-4000,45;
LINE 4.   PU-6940,-2940;PD;AA-8000,-4000,-45;
LINE 5.   PU-7000,-4000;PD;AA-8000,-4000,45;
```

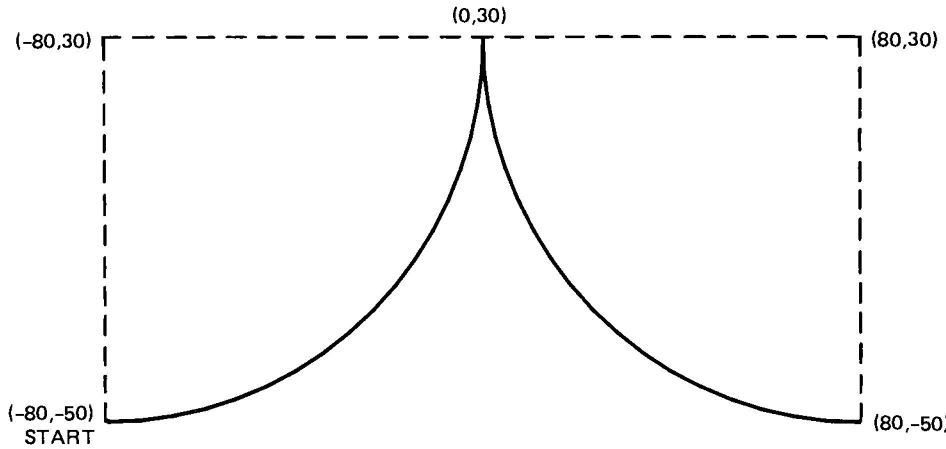


Example — Using the AR Instruction

The following instructions draw an arc centered around 0,80 plotter units relative to the starting pen position, followed by an arc centered around 80,0 plotter units relative to the new starting pen position. Note that the PD instruction is required to draw the arcs.

Send:

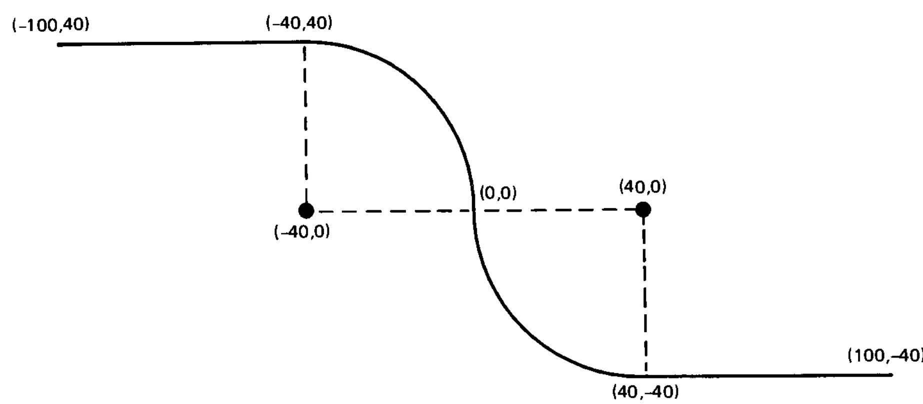
```
LINE 1.   IN;SP1;IP0,-5000,5000,0;
LINE 2.   SC-100,100,-100,100;
LINE 3.   PA-80,-50;PD;AR0,80,90;AR80,0,90;PU;
```



The next instructions show the effects of changing both the relative centers of the arcs and the signs of the angles. The first arc is drawn clockwise around the center (positive angle), whereas the second arc is drawn counterclockwise (negative angle).

Send:

```
LINE 1.   IN;SP1;IP5000,-5000,10000,0;
LINE 2.   SC-100,100,-100,100;
LINE 3.   PA-100,40;PD;PR60,0;AR0,-40,-90;
          AR40,0,90;PR60,0;PU;
```



Circles & Polygons

The Fill Type Instruction, FT

USES: The FT instruction selects the type of area fill for use with an FP, RA, RR, or WG instruction. You can use this instruction to enhance pie charts, bar charts, and other charts with solid fill, parallel lines, cross-hatching, or a fill type of your own design.

SYNTAX: *FT* type (,spacing (,angle)) *term*
or
FT *term*

PARAMETERS:

Parameter	Format	Range	Default
fill type	integer	1-6	1
spacing	decimal	0-($2^{26}-1$) current units	1% of the distance between $P2_X$ and $P1_X$
angle	decimal	$\pm(2^{26}-1)$ degrees, modulo 360	0 degrees

EXPLANATION: The FT instruction can have three parameters: type, spacing, and angle. All are optional. If you omit parameters and this is the first FT instruction in your program, the plotter uses the default parameter values. If this is not the first FT instruction, and you omit any parameters, the plotter uses the corresponding parameter values from the previous FT instruction. If you omit all parameters (FT ;), the plotter uses the default values.

The following paragraphs list further details of each parameter.

1. Type. The six parameters and corresponding fill types are:

   1 solid bidirectional filling (lines with spacing derived from the PT instruction)

   2 solid unidirectional filling (lines with spacing derived from the PT instruction)

   3 parallel lines (bidirectional for solid line types, unidirectional for dashed line types)

   4 cross-hatch (bidirectional for solid line types, unidirectional for dashed line types)

   5 user-defined bidirectional (as defined by the UF instruction)

   6 user-defined unidirectional (as defined by the UF instruction)

   All fill types are drawn using the current pen. Fill types 3-6 use the current line type, whereas solid-fill types 1 and 2 always generate a solid line. In this case, any line type specified by the LT instruction is ignored.

   To use parameters 5 and 6, you must first define a fill style using the UF instruction. If you specify parameter 5 or 6 but do not define a fill style with UF, the FT instruction will be executed with the default UF (solid fill). The angle and the direction of fill will remain the same as specified in the FT instruction.

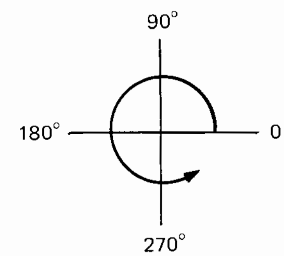
2. Spacing. Spacing is the distance between parallel lines in the fill area for fill types 3 and 4. Units for spacing are interpreted as plotter units if scaling is off or user units (as measured along the X-axis) if scaling is on.

The spacing parameter is ignored for solid-fill types 1 and 2; in these cases spacing is determined by the PT instruction. For fill types 5 and 6, the spacing parameter is used as a pattern repeat length.

The spacing parameter must be a positive number. If you specify zero (0) for this parameter, the plotter uses default spacing (solid).

If scaling is on and spacing on the X- and Y-axes is not equal, the spacing parameter is interpreted in units measured on the X-axis.

3. Angle. The angle is referenced counterclockwise from the positive direction of the X-axis (0 and 180 are horizontal; 90 and 270 are vertical). The angle applies to all fill types. For cross-hatching, the first set of lines is drawn at the specified angle. The cross-hatched lines are then drawn at that angle plus 90 degrees.



NOTE: If the plotter's unit system is rotated, the fill angle will rotate accordingly. ■

The following table summarizes the error conditions for the FT instruction.

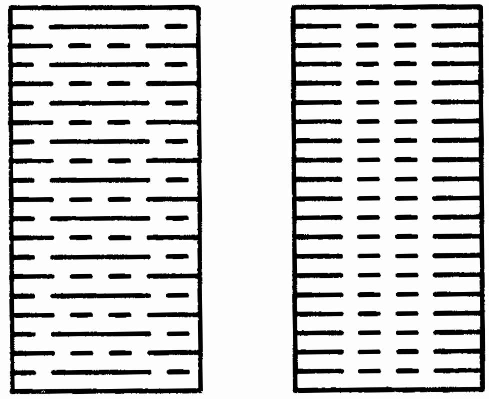
Condition	Error	Plotter Response
no parameters	none	executes default parameters
more than 3 parameters	2	executes first 3 parameters; ignores the rest
parameter out-of-range	3	ignores instruction; uses parameters of previous FT instruction

Example — Effects of Line Type Patterns

The following example shows the effects of positive and negative line types in filling a rectangular area. The positive line type produces an alternating pattern that can often be quite attractive. Negative line types may work well for regular symmetrical polygons (such as rectangles), but will have unusual results for polygons with islands. For more examples of area fill, refer to The Fill Rectangle Instructions, RA and RR, The Fill Wedge Instruction, WG, The Polygon Mode Instruction, PM, and The Fill Polygon Instruction, FP.

Send:

```
LINE 1.   IN;SP1;IP10000,-5000,15000,0;
LINE 2.   PA11000,-4000;
LINE 3.   LT6;FT3,100,0;RA12000,-2000;
LINE 4.   LT;EA12000,-2000;
LINE 5.   PA12500,-4000;
LINE 6.   LT-6;RA13500,-2000;
LINE 7.   LT;EA13500,-2000;
```



Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 sets the starting pen position

Line 3 selects line type 6 (fixed pattern); selects fill type (parallel lines) with lines spaced every 100 plotter units at an angle of 0 degrees; defines and fills a rectangle

Line 4 selects solid line type; edges the rectangle

Line 5 sets the starting pen position for the next rectangle

Line 6 selects line type −6 (adjusting pattern); defines and fills a rectangle using the same fill pattern specified in line 3

Line 7 selects solid line type; edges the rectangle

The User-Defined Fill Type Instruction, UF

USES: The UF instruction enables you to define a fill pattern composed of "gaps" between parallel lines. This becomes the fill pattern used by parameters 5 and 6 in the FT instruction. Use this fill pattern with the RA, RR, WG, and FP instructions to generate such patterns as a semi-log fill pattern, or to generate a candy-stripe effect.

SYNTAX: *UF* gap_1 (,gap_2, . . . gap_{20}) *term*
or
UF *term*

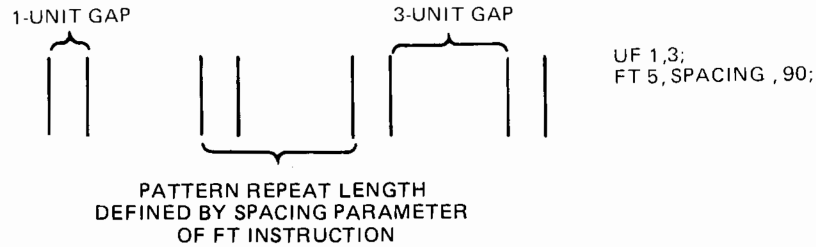
PARAMETERS:

Parameter	Type	Range	Default
gap	decimal	0-$(2^{26}-1)$	none

EXPLANATION: The gap parameters for the UF instruction are unitless numbers that represent the distance to the next fill line. In order to access the fill style described by the UF instruction, you must also execute the FT instruction. You can use the parameters in the FT instruction to influence the appearance of your fill style. This is because the plotter normalizes the sum of the gaps specified in the UF instruction to equal the spacing parameter of the FT instruction. The spacing parameter thus

Circles & Polygons

becomes a pattern repeat length for the fill pattern you have defined. This concept is shown below and further illustrated in the example at the end of this explanation section.



The pattern repeat length always begins at the plotter-unit origin (0,0), regardless of where the shape you wish to fill begins. Thus, depending on where your shape begins in the plotting area, you might notice that your fill pattern does not always begin with the same gap parameter.

NOTE: On the HP 7586, the FR instruction maintains registration of fill patterns across frame boundaries. This is done by referencing the fill pattern with the plotter-unit origin in the first frame drawn, which will have a different plotter-unit value (but the same current-unit value if scaling is on) after the FR is executed. ■

You can specify up to 20 parameters in one UF instruction. Each parameter must be positive. You may include parameters equal to zero (0), but the sum of the parameters must be greater than zero. A UF instruction without parameters (UF ;) produces solid fill.

Lines are drawn using the current pen and line type. Because of the way gaps are defined, positive parameters in the LT instruction do not always produce the alternating fill pattern described under the FT instruction. Therefore, you may find that negative line-type parameters produce more satisfactory results and solid lines look even better with user-defined fill type. Lines are either bidirectional or unidirectional, depending on whether parameter 5 or 6 was specified in the FT instruction.

The following table summarizes the error conditions for the UF instruction.

Condition	Error	Plotter Response
no parameters	none	executes with solid fill
more than 20 parameters	2	executes first 20 parameters; ignores the rest
parameter out-of-range or sum of gap parameters $\leqslant 0$	3	ignores instruction; executes previous UF instruction

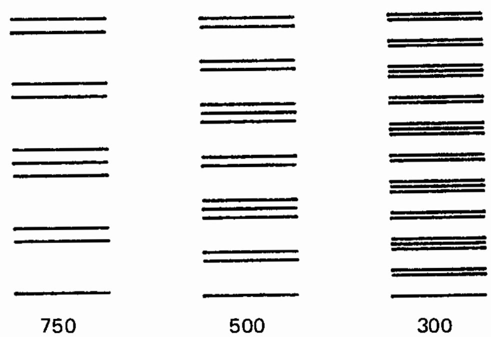
Examples — Creating Special Effects

The following example shows the effects of executing the same UF instruction, but decreasing the spacing parameter in the FT instruction to produce shorter pattern repeat lengths.

Circles & Polygons

Send:

```
LINE 1.   IN;SP1;IP-15000,-10000,-10000,-5000;
LINE 2.   UF1,4,1,4,1;
LINE 3.   FT5,750,0;
LINE 4.   PA-14000,-9000;RA-13500,-7500;
LINE 5.   FT5,500,0;
LINE 6.   PA-13000,-9000;RA-12500,-7500;
LINE 7.   FT5,300,0;
LINE 8.   PA-12000,-9000;RA-11500,-7500;
```



Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 defines the fill pattern gaps

Line 3 selects fill type 5 (unidirectional) with a spacing parameter of 750 plotter units at an angle of 0 degrees

Line 4 sets the starting pen position; defines and fills a rectangle

Line 5 changes the fill-type spacing parameter to 500 plotter units

Line 6 sets the starting pen position for the next rectangle; defines and fills the rectangle

Line 7 changes the fill-type spacing parameter to 300 plotter units

Line 8 sets the starting pen position for the next rectangle; defines and fills the rectangle

If the pattern repeat length is small, so that spaces between clusters of lines are much greater than the spaces between lines in a cluster, the appearance of different line widths can be created.

Send:

```
LINE 1.   IN;SP1;IP-15000,-10000,-10000,-5000;
LINE 2.   PA-11000,-9000;
LINE 3.   UF1,5,1,1,5,1,1,1,5,1,1,1,1,5;
LINE 4.   FT5,300,0;
LINE 5.   RA-10500,-7500;
```

Line 3 defines the fill pattern gaps

Line 4 selects fill type 5 (bidirectional) with a spacing parameter of 300 plotter units at an angle of 0 degrees

Line 5 defines and fills a rectangle

Or, you might want to create a semi-log pattern, as illustrated below.

Send:

```
LINE 1.   IN;SP1;IP-15000,-10000,-10000,-5000;
LINE 2.   PA-14000,-7000;
LINE 3.   UF9,8,7,6,5,4,3,2,1;
LINE 4.   FT5,500,0;RA-13500,-5500;
```

The Pen Thickness Instruction, PT

USES: The PT instruction determines the optimum spacing between the lines drawn in a solid fill for rectangles, wedges, and polygons. Use this instruction with the FT, RA, RR, WG, and FP instructions to produce a good solid fill.

SYNTAX: *PT* pen thickness *term*
or
PT *term*

PARAMETERS:

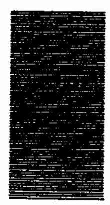
Parameter	Type	Range	Default
pen thickness	decimal	0.1-5.0 millimetres	.3

EXPLANATION: The pen thickness parameter represents the physical pen-tip width in millimetres. Following is a list of typical pen thicknesses. Specify the number that corresponds to the currently selected pen.

Pen Type	Thickness
Felt-tip or transparency (normal)	.30
Felt-tip or transparency (wide)	.70
Roller-ball	.30
Drafting 4 × 0 (not commonly used)	.18
Drafting 3 × 0 (not commonly used)	.25
Drafting 0	.35
Drafting 1	.50
Drafting 2	.70
Drafting 4	1.00

The current pen thickness determines (but is not identical to) the spacing of lines needed to produce a solid fill. The thicker the pen, the wider the gap allowed between lines. If your solid fill has gaps showing between the lines, adjust the pen thickness downward. If a felt-tip pen is getting "fat" through wear, or if improved throughput is desired at the expense of less dense fill, adjust the pen thickness upward.

PT.7;FT1 drawn with 0.7-mm wide pen. This example produces the optimum solid fill for a 0.7-mm wide pen.



PT.7;FT1 drawn with 0.3-mm wide pen. This example has a less dense fill.

The PT instruction pertains only to the currently selected pen. It remains in effect until:

- a new pen is selected using an SP instruction or the front-panel controls
- a new PT instruction is executed
- the plotter is initialized upon power-up, front-panel reset, or through the DF and IN instructions

The following table summarizes the error conditions for the PT instruction.

Condition	Error	Plotter Response
more than 1 parameter	2	executes first parameter
parameter out-of-range	3	executes default parameter

The Fill Wedge Instruction, WG

USES: The WG instruction fills any arc sector of a circle. Use this instruction with the FT, UF, and PT instructions to produce individual wedges that can be combined to create a pie chart.

SYNTAX: *WG* radius, start angle, sweep angle (, chord tolerance) *term*

PARAMETERS:

Parameter	Type	Range	Default
radius	integer	$\pm(2^{26}-1)$ current units	none
start angle	decimal	$\pm(2^{26}-1)$ degrees, truncated at ± 360	none
sweep angle	decimal	$\pm(2^{26}-1)$	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

EXPLANATION: The first three parameters are required since they have no default values. The next paragraphs list details of each parameter, followed by illustrations of how the parameters are interpreted. Further explanation of the WG instruction follows these illustrations.

1. Radius. The radius is the distance from the current pen position to the start of the wedge's arc. The radius is in plotter units if scaling off and user units if scaling is on. If user units are not the same size in the X- and Y-directions, elliptical wedges will be drawn. The sign of the radius defines the zero-degree reference point for the start angle and sweep angle.

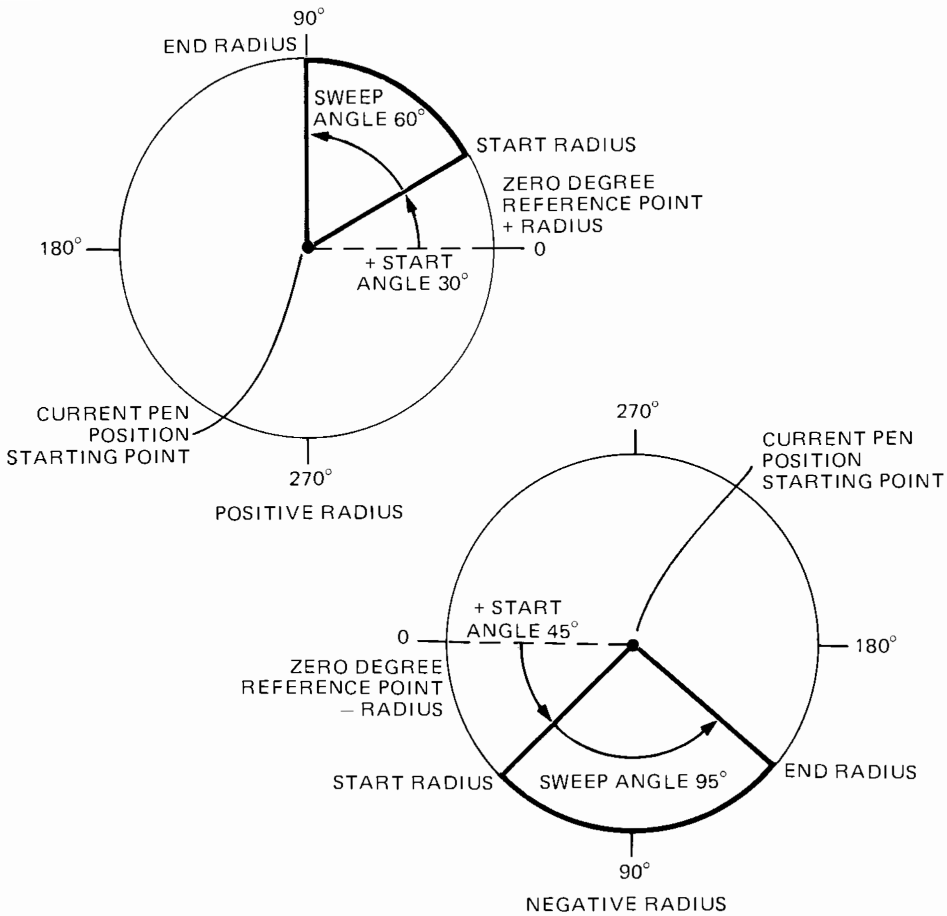
2. Start Angle. The start angle defines where the radius is first drawn. A positive start angle positions the radius counterclockwise from the zero-degree reference point; a negative start angle positions the radius clockwise from the zero-degree reference point.

3. Sweep Angle. The sweep angle defines the number of degrees through which the arc sector is drawn from the start angle. A positive sweep angle draws the arc sector counterclockwise; a negative sweep angle draws the arc sector clockwise.

4. Chord Tolerance. The chord tolerance parameter is interpreted as either degrees of deviation distance, depending on whether degrees mode or deviation mode has been set (by the CT instruction or default conditions). Refer to The Chord Tolerance Instruction, CT, in this chapter.

   The sign of this parameter is ignored. The default chord tolerance is 5 degrees. As discussed under the CI and CT instructions, increasing the number of chords (by decreasing the chord tolerance parameter) generates a smoother arc.

   NOTE: The WG instruction clears the polygon buffer and then uses it for the wedge definition before filling the wedge. ■



The WG instruction defines and shades an arc wedge using the current pen and fill type (and the current line type if the fill is not solid). The reference point for the wedge is the current pen position, which should be thought of as the center of the circle that would be drawn if the wedge were 360 degrees. Upon completion of the wedge, the pen returns to this position and the pen state is restored.

The following table summarizes the error conditions for the WG instruction.

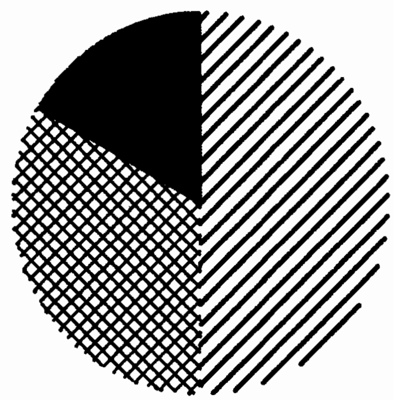
Condition	Error	Plotter Response
no parameters	none	instruction ignored
fewer than 3 parameters	2	instruction ignored
more than 4 parameters	2	executed with first 4 parameters
parameter out-of-range	3	instruction ignored
polygon buffer overflow	7	no filling is done; the points that fit in the buffer are drawn with subsequent FP instructions
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, in Chapter 4

Example — Defining and Filling Wedges Using the WG Instruction

The following example illustrates the use of the WG instruction in creating a pie chart.

Send:

```
LINE 1.   IN;SP1;IP-10000,-10000,-5000,-5000;
LINE 2.   PA-7500,-7500;
LINE 3.   FT3,75,45;
LINE 4.   WG-1000,90,180;
LINE 5.   SP3;FT1;
LINE 6.   WG-1000,270,60;
LINE 7.   SP4;FT4,60,45;
LINE 8.   WG-1000,330,120;
```



Line 1 initializes the plotter; selects pen 2; positions P1 and P2

Line 2 sets the starting pen position (center of the pie)

Line 3 selects fill type 3 (parallel lines) with lines spaced every 75 plotter units at an angle of 45 degrees

Line 4 defines and fills a wedge based on the zero-degree reference point for a negative radius of 1000 plotter units, starting at 90 degrees, sweeping counterclockwise through 180 degrees

Line 5 selects pen 3; selects fill type 1 (solid, bidirectional)

Line 6 defines and fills a wedge with the same zero-degree reference point and radius, starting at 270 degrees, sweeping counterclockwise through 60 degrees

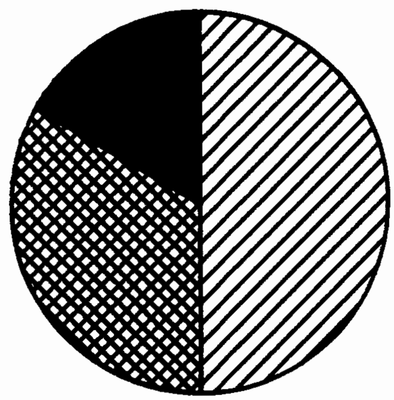
Line 7 selects pen 4; selects fill type 4 (cross-hatch) with lines spaced every 60 plotter units at an angle of 45 degrees

Line 8 defines and fills a wedge with the same zero-degree reference point and radius, starting at 330 degrees, sweeping counterclockwise through 120 degrees

Example — Edging the Wedges Using the EW or EP Instructions

You can edge the wedges with either The Edge Wedge Instruction, EW, or the Edge Polygon Instruction, EP. You might find the EP instruction to be more effcient since it does not require you to type the parameters defining the wedge. This is because EP draws the edge of the wedge that the WG instruction defines and stores in the polygon buffer. Therefore, the EP instruction must follow the WG instruction. The EW instruction, on the other hand, may be executed either before or after the WG instruction. (However, the usual graphics programming practice is to edge after filling.)

To edge the wedges with the EP instruction, refer to method 1, adding the lines to the program in the previous example. Then, replace these lines with those given in method 2 to see how the EW instruction works. For more detailed information, refer to the EW and EP instructions in this chapter.



1. To edge the wedges in the previous program using the EP instruction, simply add EP ; following each WG instruction.
2. To edge the wedges in the previous program using the EW instruction, simply add these lines following each respective WG instruction. (If you have already added the EP instruction, delete it before adding the following lines.)

```
EW-1000,90,180;
EW-1000,270,60;
EW-1000,330,120;
```

The Edge Wedge Instruction, EW

USES: The EW instruction edges any arc sector of a circle. Use this instruction to outline individual wedges that can be combined to create a pie chart.

SYNTAX: *EW* radius, start angle, sweep angle (, chord tolerance) *term*

PARAMETERS:

Parameter	Type	Range	Default
radius	integer	$\pm(2^{26}-1)$ current units	none
start angle	decimal	$\pm(2^{26}-1)$ degrees, modulo 360	none
sweep angle	decimal	$\pm(2^{26}-1)$ degrees, truncated at ± 360	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

Circles & Polygons

EXPLANATION: The first three parameters are required since they have no default values. The next paragraphs list details of each parameter, followed by illustrations of how the parameters are interpreted. Further explanation of the EW instruction follows these illustrations. Note that the parameters of this instruction are interpreted in exactly the same way as the WG instruction. The only difference is that the wedge is outlined with EW, and filled with WG. Also, EW does not use the polygon buffer.

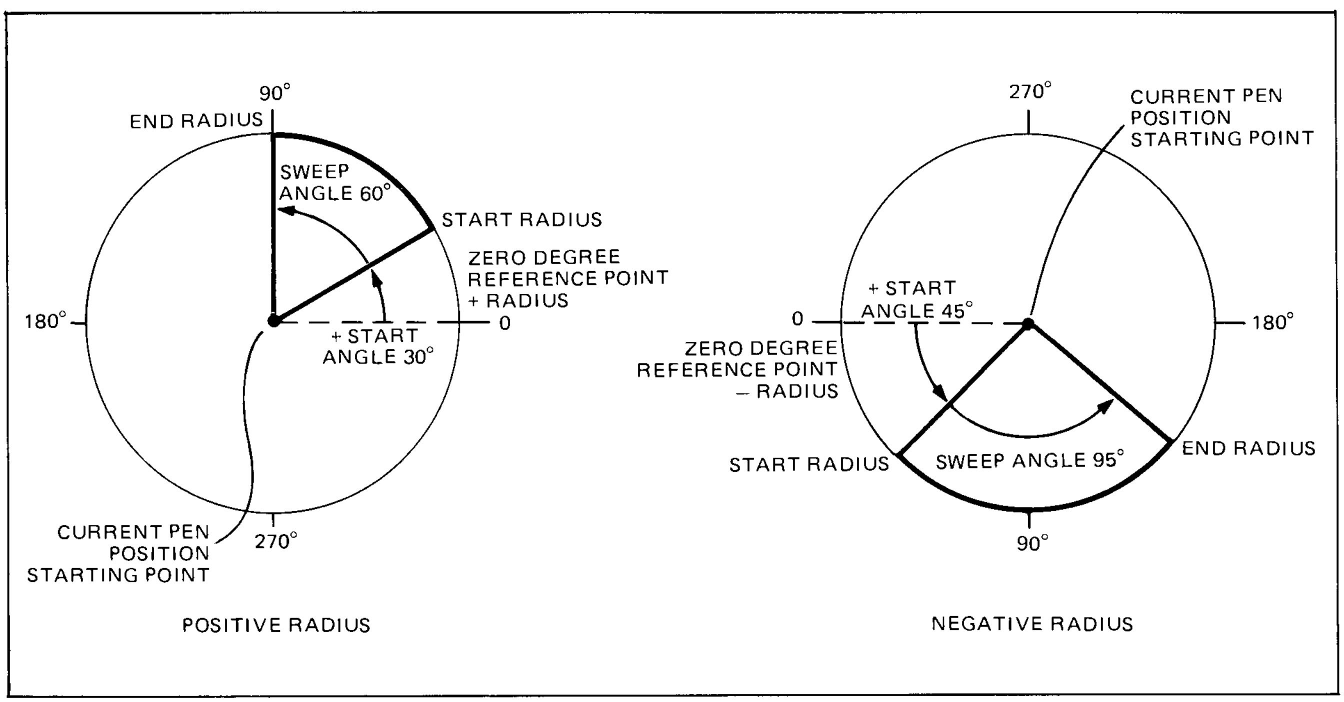
1. Radius. The radius is the distance from the current pen position to the start of the wedge's arc. The radius is in plotter units if scaling is off and user units if scaling is on. If user units are not the same size in the X- and Y-directions, elliptical wedges will be drawn. The sign of the radius defines the zero-degree reference point for the start angle and sweep angle.

2. Start Angle. The start angle defines where the radius is first drawn. A positive start angle positions the radius counterclockwise from the zero-degree reference point; a negative start angle positions the radius clockwise from the zero-degree reference point.

3. Sweep Angle. The sweep angle defines the number of degrees through which the arc segment is drawn from the start angle. A positive sweep angle draws the arc segment counterclockwise; a negative sweep angle draws the arc segment clockwise.

4. Chord Tolerance. The chord tolerance parameter is interpreted as either degrees or deviation distance, depending on whether degrees mode or deviation mode has been set (by the CT instruction or default conditions). Refer to The Chord Tolerance Instruction, CT, in this chapter.

   The sign of this parameter is ignored. The default chord tolerance is 5 degrees. As discussed under the CI and CT instructions, increasing the number of chords (by decreasing the chord tolerance parameter) generates a smoother arc.



The EW instruction defines and edges an arc wedge using the current pen type and line type. The reference point for the wedge is the current pen position, which should be thought of as the center of the circle that would be drawn if the wedge were 360 degrees. Upon completion of the wedge, the pen returns to this position and the pen state is restored. The EW instruction does not use the polygon buffer.

NOTE: If the sweep angle is ±360 degrees (or truncated at that), the wedge becomes a circle and the lines to the center point are not drawn. ■

The following table summarizes the error conditions for the EW instruction.

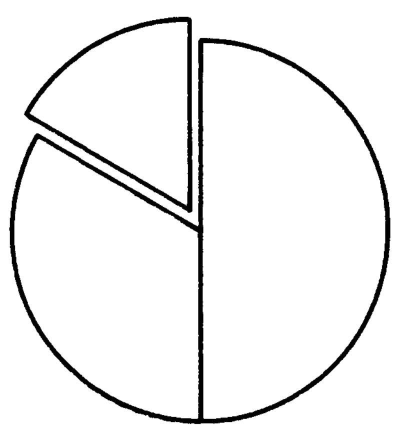
Condition	Error	Plotter Response
no parameters	none	instruction ignored
fewer than 3 parameters	2	instruction ignored
more than 4 parameters	2	executed with first 4 parameters
parameter out-of-range	3	instruction ignored
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, in Chapter 4

Example — Creating a Pie Chart with an Offset Wedge

The following example creates the same pie chart as the example for the WG instruction, except that one wedge is offset and the wedges are not filled.

Send:

```
LINE 1.  IN;SP2;IP-5000,-10000,0,-5000;
LINE 2.  PA-2500,-7500; .
LINE 3.  EW-1000,90,180;
LINE 4.  SP3;PR-60,110;
LINE 5.  EW-1000,270,60;
LINE 6.  SP4;PA-2500,-7500;
LINE 7.  EW-1000,330,120;
```



Circles & Polygons

Line 1 initializes the plotter; selects pen 2; positions P1 and P2

Line 2 sets the starting pen position (center of the pie)

Line 3 defines and edges a wedge based on the zero-degree reference point for a negative radius of 1000 plotter units, starting at 90 degrees, sweeping counterclockwise through 180 degrees

Line 4 selects pen 3; sets a new pen position, which changes the pie's center point in order to offset the wedge

Line 5 defines and edges a wedge with the same zero-degree reference point and radius, starting at 270 degrees, sweeping counterclockwise through 60 degrees

Line 6 selects pen 4; sets the pen position back to the original pie center

Line 7 defines and edges a wedge with the same zero-degree reference point and radius, starting at 330 degrees, sweeping counterclockwise through 120 degrees

The Fill Rectangle Instructions, RA and RR

USES: The RA and RR instructions define and fill a rectangle. Use either of these instructions with the FT and PT instructions to fill rectangles for bar charts, logos, and other plots. Choose the fill rectangle absolute instruction, RA, when you want to define a rectangle using absolute coordinates; choose the fill rectangle relative instruction, RR, when you want to define the rectangle using relative coordinates. The RA and RR instructions are identical except for the type of coordinates used. For in-depth discussions of absolute and relative plotting, refer to The Plot Absolute Instruction, PA, and The Plot Relative Instruction, PR, in Chapter 4.

SYNTAX: *RA* X-coordinate, Y-coordinate *term*

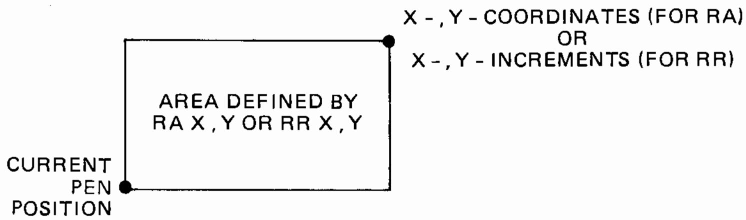
RR X-increment, Y-increment *term*

PARAMETERS:

Parameter	Type	Range	Default
X- and Y-coordinates or X- and Y-increments	integer	$\pm(2^{26}-1)$ current units	none

EXPLANATION: Parameters are in plotter units if scaling is off and in user units if scaling is on.

The current pen position is the starting point of the area to be filled. The X,Y coordinates (or increments) specify the opposite diagonal corner of the rectangle, as shown in the following illustration.



The plotter fills the designated rectangle using the current pen and fill type (and the current line type if the fill is not solid). Upon completion of the rectangle, the plotter returns to the starting point and restores the pen status.

NOTE: The RA and RR instructions clear the polygon buffer and then use it for the rectangle definition before filling the rectangle. Enough memory to fit 5 points per rectangle must be allocated in the polygon buffer for these instructions to work. The default memory is sufficient; however, if you wish to decrease the polygon buffer size, refer to The Polygon Buffer at the end of this chapter. ■

The following table summarizes the error conditions for the RA and RR instructions.

Condition	Error	Plotter Response
no parameters	none	instruction ignored
1 parameter	2	instruction ignored
more than 2 parameters	2	executed with first 2 parameters
parameter out-of-range	3	instruction ignored
polygon buffer overflow	7	no filling is done; the points that fit in the buffer can be drawn with subsequent EP instructions
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, in Chapter 4

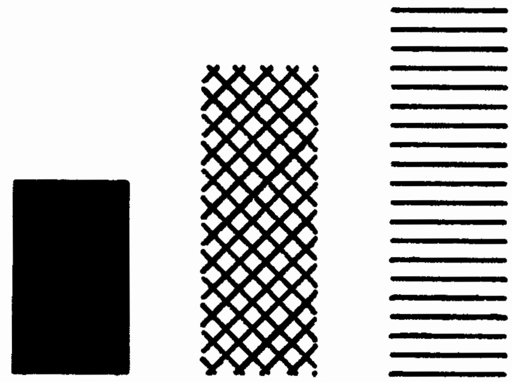
Example — Defining and Filling Rectangles Using RA or RR

The following example demonstrates the use of the RA and RR instructions with the FT instruction. The full program listing uses the RA instruction. To see how the RR instruction works, simply replace the RA instructions in the program on the left with the RR instructions shown to the right. Notice the differences between the absolute coordinates in the RA instructions and the relative coordinates in the RR instructions. Whichever instructions you use, the plot produced will be the same.

```
LINE 1.   IN;SP1;IP0,-10000,,5000,5000;
LINE 2.   PA1000,-9000;
LINE 3.   FT1;RA1600,-8000;                        RR 600,1000;
LINE 4.   PA2000,-9000;
LINE 5.   FT4,100,45;RA2600,-7400;                 RR 600,1600;
LINE 6.   PA3000,-9000;FT3,100,0;
LINE 7.   FT3,100,0;RA3600,-7000;                  RR 600,2000;
```

Program using RA instruction

Replace the shaded RA instructions with these RR instructions to see how RR works



Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 sets the starting pen position

Line 3 selects fill type 1 (solid bidirectional); defines and fills the rectangle

Line 4 sets the starting pen position for the next rectangle

Line 5 selects fill type 4 (cross-hatch) with lines spaced every 100 plotter units at an angle of 45 degrees; defines and fills the rectangle

Line 6 sets the starting pen position for the next rectangle

Line 7 selects fill type 3 (parallel lines) with lines spaced every 100 plotter units at an angle of 0 degrees; defines and fills the rectangle

Example — Edging Rectangles Using EA, ER, or EP

You have three choices of instructions for edging rectangles:

EA The Edge Rectangle Absolute Instruction (for edging a rectangle using absolute coordinates)

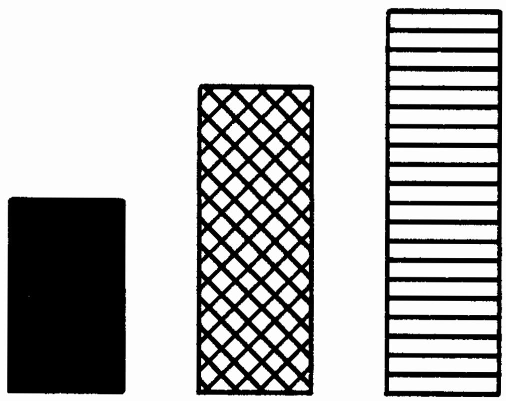
ER The Edge Rectangle Relative Instruction (for edging a rectangle using relative coordinates)

EP The Edge Polygon Instruction (for edging a rectangle without using coordinates)

You might find the EP instruction to be more efficient than the EA and ER instructions, since EP does not require you to type the X,Y coordinates. This is because EP draws the edge of the rectangle that is stored in the buffer. Therefore, the EP instruction must follow the RA or RR instruction.

The EA and ER instructions, on the other hand, may be executed either before or after the RA and RR instructions. (However, the usual graphics programming practice is to edge after filling.)

To edge the rectangles with the EP instruction, refer to method 1, adding the lines to the program in the previous example. Then, replace these lines with those given in method 2 to see how the EA instruction works, or with those given in method 3 to see how the ER instruction works. Each method produces the same plot, as shown. For more detailed information, refer to EP, EA, and ER instructions in this chapter.



1. To use the EP instruction to edge the rectangles in the previous program, simply add EP; following each RA instruction. These lines can be added to both the program that uses the RA instruction, and the modified program that uses the RR instruction.

2. To use the EA instruction to edge the rectangles, simply add these lines following the respective RA instruction to the program that uses the RA instruction. (If you have already added the EP instruction in method 1, delete it before adding the following lines.)

```
EA 1600,2000
EA 2600,2600
EA 3600,3000
```

3. To use the ER instruction to edge the rectangles, simply add these lines following the respective RR instruction to the modified program that uses the RR instruction. (If you have already added the EP instruction in method 1, delete it before adding the following lines.)

```
ER 600,1000
ER 600,1600
ER 600,2000
```

The Edge Rectangle Instructions, EA and ER

USES: The EA and ER instructions draw the outline of a rectangle. Use these instructions with the RA and RR instructions to add an edge to a filled rectangle. Or, use these instructions to draw other charts that use rectangles; for example, flow charts, organization charts, and scheduling charts. Choose The Edge Rectangle Absolute Instruction, EA, when you want to outline a rectangle using absolute coordinates; choose The Edge Rectangle Relative Instruction, RR, when you want to outline a rectangle using relative coordinates. The EA and ER instructions are identical except for the type of coordinates used. For in-depth discussions of absolute and relative plotting, refer to The Plot Absolute Instruction, PA, and the Plot Relative Instruction, PR, in Chapter 4.

SYNTAX: *EA* X-coordinate, Y-coordinate *term*

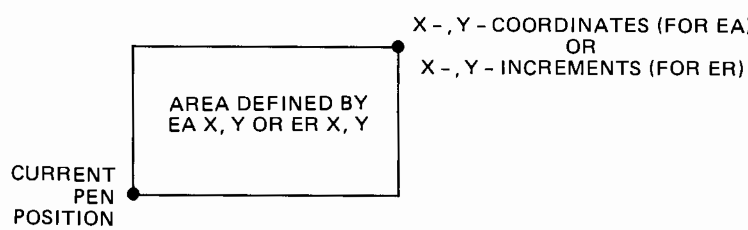
ER X-increment, Y-increment *term*

PARAMETERS:

Parameter	Type	Range	Default
X- and Y-coordinates or X- and Y-increments	integer	$\pm(2^{26}-1)$ current units	none

EXPLANATION: Note that the EA and ER instructions are interpreted in exactly the same way as the RA and RR instructions. The only difference is that EA and ER outline rectangles, whereas RA and RR fill rectangles. Also, it is not necessary to have any space allocated in the polygon buffer to use these instructions.

Parameters are in plotter units if scaling is off and in user units if scaling is on. The current pen position is the starting point of the area to be edged. The X,Y coordinates (or increments) specify the opposite diagonal corner of the rectangle, as shown in the following illustration.



The plotter edges the designated rectangle using the current pen and a solid line. Upon completion of the rectangle, the plotter returns to the starting point and restores the pen status.

The following table summarizes the error conditions for the EA and ER instructions.

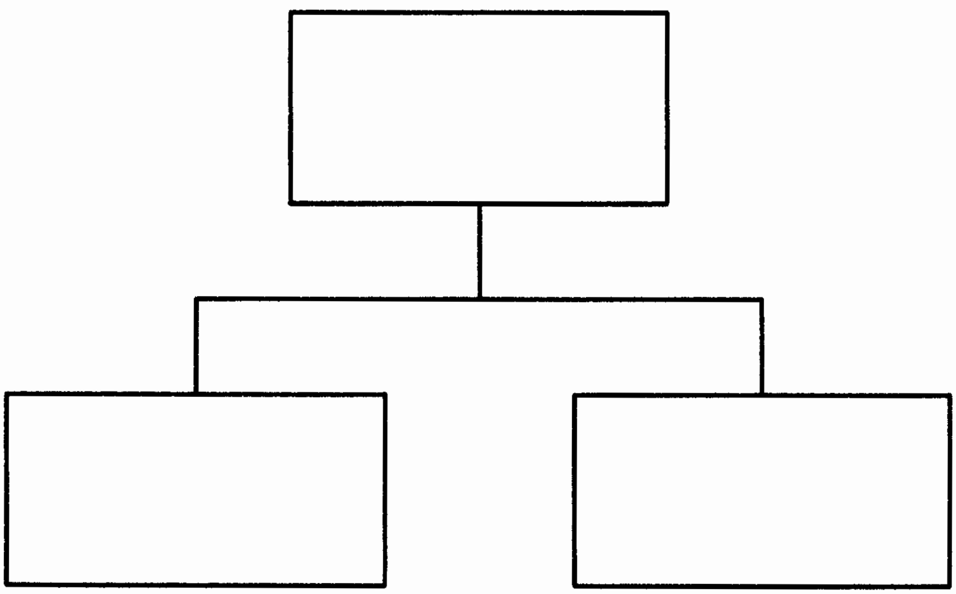
Circles & Polygons

Condition	Error	Plotter Response
no parameters	none	instruction ignored
1 parameter	2	instruction ignored
more than 2 parameters	2	executed with first 2 parameters
parameter out-of-range	3	instruction ignored
lost mode	none	conforms to the conditions described under The Plot Absolute Instruction, PA, in Chapter 4

Example — Outlining Rectangles Using EA or ER

For examples of outlining filled rectangles, refer to The Fill Rectangle Instructions, RA and RR, located in this chapter.

You can also outline rectangles for use in flow charts, organization charts, and scheduling charts. The following programs illustrate how you can use the EA and ER instructions to set up the outline of an organization chart (reduced in the following example). Both programs produce the same plot. However, the first program (method 1) draws the rectangles and connecting lines using absolute coordinates, whereas the second program (method 2) draws the rectangles and connecting lines using relative coordinates.



1. This program uses absolute coordinates for all pen movements to produce the previous plot. An explanation for both methods 1 and 2 follows the listing in method 2.

```
LINE 1.  IN;SP1;IP5000,-10000,12000,-5000;
LINE 2.  PA8000,-7000;
LINE 3.  EA10000,-6000;
LINE 4.  PA9000,-7000;
LINE 5.  PD9000,-7500;
LINE 6.  PU10500,-7500;
LINE 7.  PD7500,-7500,7500,-8000;
LINE 8.  PU8500,-8000;
LINE 9.  EA6500,-9000;
LINE 10. PU10500,-7500;
LINE 11. PD10500,-8000;
LINE 12. PU11500,-8000;
LINE 13. EA9500,-9000;
```

2. This program uses absolute coordinates to establish only the initial pen position. All subsequent pen movements use relative coordinates.

```
LINE 1.  IN;SP1;IP5000,-10000,12000,-5000;
LINE 2.  PA8000,-7000;
LINE 3.  ER2000,1000;
LINE 4.  PR1000,0;
LINE 5.  PD0,-500;
LINE 6.  PU1500,0;
LINE 7.  PD-3000,0,0,-500;
LINE 8.  PU1000,0;
LINE 9.  ER-2000,-1000;
LINE 10. PU2000,500;
LINE 11. PD0,-500;
LINE 12. PU1000,0;
LINE 13. ER-2000,-1000;
```

The following explanation applies to both of the previous programs. Notice the differences between the absolute and relative coordinates in the PD, PU, EA, and ER instructions.

Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 sets the starting pen position using absolute coordinates (in both methods)

Line 3 defines and outlines the top rectangle (in absolute coordinates for method 1, and relative coordinates for method 2)

Line 4 moves to the starting pen position for the first vertical connecting line (in method 1, this line establishes absolute coordinates for all subsequent PU and PD instructions; in method 2, this line establishes relative coordinates for all subsequent PU and PD instructions)

Line 5 draws the first vertical connecting line

Line 6 moves to the starting pen position for the horizontal connecting line

Line 7 draws the horizontal connecting line, followed by the left vertical connecting line

Line 8 moves to the starting pen position for the left rectangle

Line 9 defines and outlines the left rectangle

Line 10 moves to the starting pen position for the right vertical connecting line

Line 11 draws the right vertical connecting line

Line 12 moves to the starting pen position for the right rectangle

Line 13 defines and outlines the right rectangle

The Polygon Mode Instruction, PM

USES: The PM instruction places the plotter in polygon definition mode. In this mode, you can construct polygons using other HP-GL instructions. A polygon is simply a shape consisting of a number of vertices. Thus, you can use this instruction to design shapes such as block letters and logos. The PM instruction only defines the polygon. In order to draw the polygon, you must fill it with the FP instruction and/or edge it with the EP instruction.

SYNTAX: *PM* n *term*

PARAMETERS:

Parameter	Format	Range	Default
n	integer	0, 1, or 2	none

EXPLANATION: The PM instruction accepts only three parameters, as follows.

0 Clears the polygon buffer and enters polygon mode

1 Closes current subpolygon

2 Closes current subpolygon and exits polygon mode

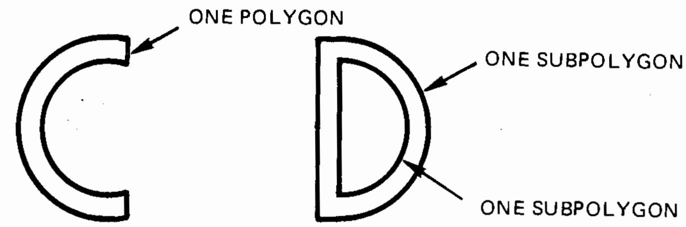
In addition, the plotter recognizes only the following HP-GL instructions while it is in polygon mode. (All other instructions are ignored.) The first column lists those instructions which are not executed but are stored in the polygon buffer, where they can be accessed by the EP and FP instructions to draw the polygon. The second column lists the instructions which are executed by the plotter.

Stored in Polygon Buffer		Executed	
PM	Polygon mode instruction	IN	Initialization instruction (exits polygon mode)
PA/PR	Plotting instructions	*	All output instructions
PU/PD	Pen instructions		
AA/AR	Arc instructions		
CI	Circle instruction		
CT	Chord Tolerance instruction		

*OA, OC, OD, OE, OF, OH, OI, OL, OO, OP, OS, OT, OW

Circles & Polygons

While in polygon mode, you can define either one polygon, or a series of subpolygons. For example, the block letter C is one complete polygon. However, the block letter D consists of two subpolygons: the outline and the "hole."



To define one polygon (e.g., the letter C): move to the starting location and execute PM 0 to enter polygon mode. Then use the appropriate HP-GL instructions to define the shape of the C. Finally, execute PM 2 to exit polygon mode.

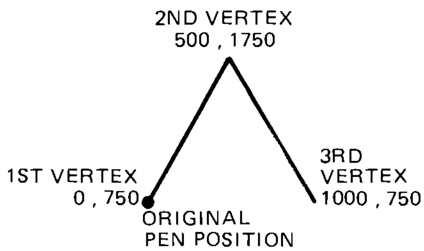
To define a series of subpolygons (e.g., the letter D): move to the starting location and execute PM 0 to enter polygon mode. Then use the appropriate HP-GL instructions to define the outside shape of the D, and end this subpolygon with PM 1. Now use the appropriate HP-GL instructions to define the inside shape of the D. Finally, execute PM 2 to exit polygon mode.

The current pen position *at the time of* PM 0 is the first point of polygon, thus the first point stored in the buffer. The first point *after* a PM 1 is the initial vertex of the next polygon or subpolygon. When edging with the EP instruction, the plotter always moves to the first point after a PM 1 with the pen up, regardless of the pen's current up/down state.

Each subsequent point defines a vertex of the polygon. The vertices can be defined with the pen up or down. However, if you intend to outline the polygon with the EP instruction, note that EP only draws those points that are defined with the pen down. The FP instruction, on the other hand, fills all vertices, regardless of the pen up/down state. (The line to the first point after a PM 1 is not a fill boundary.)

It is good programming practice to close the polygon before executing PM 1 or PM 2. However, if you have not closed the polygon, PM 1 and PM 2 do force the closure. In this case, executing PM 1 or PM 2 adds a vertex to close the polygon. The original pen position and status (up/down) are restored after PM 2 is executed.

```
PU0,750;PM0;
PD500,1750,1000,750;
```



Open polygon

```
PU0,750;PM0;
PD500,1750,1000,750,0,750;PM2;
```

Closed polygon

NOTE: The PM instruction uses the polygon buffer. Enough memory must be allocated in the buffer for this instruction to work. The default buffer size is usually sufficient for a polygon with about 218 points (vertices). For full details on determining the proper memory allocation, refer to The Polygon Buffer at the end of this chapter. ■

The following table summarizes the error conditions for the PM instruction.

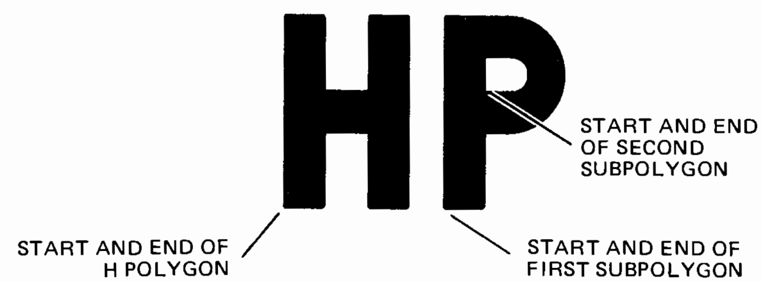
Condition	Error	Plotter Response
parameter other than 0, 1, or 2	3	instruction ignored
PM 1 or PM 2 executed before PM 0	none	PM 1 or PM 2 instruction ignored (therefore, any HP-GL instructions included to define the polygon are executed instead of being placed in the buffer, and subsequent FP and EP instructions operate on any previously entered polygon)
use of any HP-GL instruction other than those listed above while in polygon mode	1	illegal instruction ignored
too many points; buffer overflow	7	points that caused the overflow are ignored
polygon mode not exited (PM 2 not executed after PM 0)	none	all instructions after PM 0 ignored except for those listed previously

Example — Creating Block Letters in Polygon Mode

The following example demonstrates how you can define the letter H as a polygon, and the letter P as two subpolygons. To design polygons, it is often helpful to draw them first on grid paper, in order to help you determine the proper coordinates. (The letters in this program are defined using relative coordinates starting at the lower-left corner of each letter and moving up and clockwise around the letter.) Remember, you cannot see the polygons until you fill or edge them. Refer to The Fill Polygon Instruction, FP, and The Edge Polygon Instruction, EP, for details.

Send:

```
LINE 1.   IN;SP1;IP-15000,5000,-10000,10000;
LINE 2.   PA-12000,6000;
LINE 3.   PM0;PR;PD0,0,0,1000,200,0,0,-400;
LINE 4.   PD250,0,0,400,200,0,0,-1000,-200,0;
LINE 5.   PD0,400,-250,0,0,-400,-200,0;
LINE 6.   PU;PM2;
LINE 7.   FP;EP;PU850,0;
LINE 8.   PM0;PD0,0,0,1000,325,0"
LINE 9.   AR0,-300,-180;
LINE 10.  PD-125,0,0,-400,-200,0;PM1;
LINE 11.  PU200,600;PD0,200,150,0;
LINE 12.  AR0,-100,-180;
LINE 13.  PD-150,0;PU;PM2;
LINE 14.  FP;EP;SP0;
```



Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 sets the initial pen position

Line 3 clears the polygon buffer and enters polygon mode; specifies relative coordinates; lowers the pen and defines the first four vertices of the letter H, beginning with the lower-left corner

Line 4 defines the next five vertices of the H

Line 5 defines the last four vertices of the H (including the closing vertex)

Line 6 lifts the pen (as a precaution against unwanted lines being drawn); exits polygon mode

Line 7 fills the H using default values (solid, bidirectional fill); outlines the H; moves with the pen up to the starting position for the outside of the letter P

Line 8 clears the polygon buffer and enters polygon mode; lowers the pen and defines the first three vertices of the outline of the P, beginning with the lower-left corner

Line 9 defines the outside rounded portion of the P, still using relative coordinates, and a counterclockwise angle of 180 degrees

Line 10 defines the last three vertices of the outline of the P (including the closing vertex); closes this subpolygon

Line 11 moves with the pen up to the starting position for the inside of the P; defines the first three vertices of the inside of the P

Line 12 defines the inside rounded portion of the P

Line 13 defines the final (closing) vertex of the inside of the P; lifts the pen; exits polygon mode

Line 14 fills the P; outlines the P; returns the pen to the carousel

Example — Using the Circle Instruction, CI, in Polygon Mode

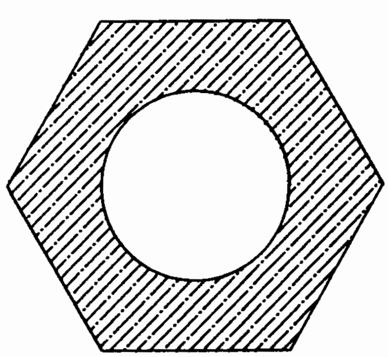
A circle is interpreted slightly differently from the other HP-GL instructions that can be used in polygon mode. The difference is that a circle is always considered to be a complete subpolygon. That is, when the CI instruction is used in polygon mode, the plotter treats it as if it were preceded and followed by PM1. The first coordinate points entered after a CI instruction thus become the first vertex of the next subpolygon. If a circle is to be the first element of a polygon, be sure to move the pen to the desired center point before executing PM0.

As noted previously, PM1 closes the current subpolygon if it is open. This usually adds another vertex to the subpolygon and changes the pen position. Thus, if you do not close the current subpolygon before executing the CI instruction, the pen position could change. Since the CI instruction is based on the current pen position, this means that the location of the circle could change. You should also note that there is no need to specify PD, as the CI instruction includes an automatic pen down feature. Remember also that when the circle is complete, the pen returns to the last position before the CI instruction was executed.

The following example demonstrates the use of the CI instruction in defining the two subpolygons needed to draw a hexagonal nut. Notice the alternating alignment of the dash patterns. The alignment is maintained even as the pattern crosses the center island.

Send:

```
LINE 1.   IN;SP1;IP-10000,5000,-5000,10000;
LINE 2.   PA-7500;7500;
LINE 3.   PM0;CI1000,60;CI500;PM2;
LINE 4.   LT4;FT3,50,45;FP;LT;EP;
```



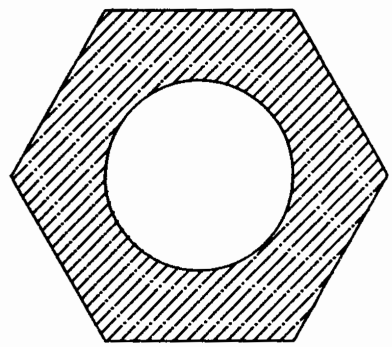
Line 1 initializes the plotter; selects pen 1; positions P1 and P2

Line 2 sets the initial pen position

Line 3 clears the polygon buffer; and enters polygon mode; specifie a circle with a radius of 1000 plotter units and chord tolerance o 60 degrees as the first subpolygon; specifies a circle with a radiu: of 500 plotter units and default chord tolerance of 5 degrees as th second subpolygon; exits polygon mode

Line 4 selects line type 4 (dashed lines); defines fill type 3 (parallel lines with lines spaced every 50 plotter units at an angle of 45 degrees fills the polygon; selects the default line type in order to edge witl a solid line; edges the polygon

Below is an example of using a negative line type. Substitution of LT-4 for LT4; in program line 4 produces this result. Notice the irregular effec of negative line patterns when drawing polygons with islands.



The Edge Polygon Instruction, EP

USES: The EP instruction outlines a defined polygon. Use this instruction to edge polygons that have been defined with the PM, RA, RR, and WG instructions.

SYNTAX: *EP term*

PARAMETERS: None.

EXPLANATION: The EP instruction outlines the polygon that has been previously defined in the polygon buffer. Outlining is done with the current pen and line type. Valid polygons include those defined by the PM, RA, RR, and WG instructions. While the EP instruction does access the data in the polygon buffer, note that EP does not clear the buffer or change the data in any way.

Only vertices that were defined with the pen down are edged. These are edged with the current pen. Upon completion of the polygon, the original pen position and status (up/down) are restored.

The following table summarizes the possible error conditions for the EP instruction.

Condition	Error	Plotter Response
previous PM, RA, RR, or WG instruction overflowed the buffer	none	EP instruction edges only those points that remain in the buffer
no polygon previously defined	none	instruction ignored

Circles & Polygons

For examples using the EP instruction, refer to the RA, RR, WG, and PM instructions in this chapter.

The Fill Polygon Instruction, FP

USES: The FP instruction fills a defined polygon with the fill type specified in the FT instruction. Use the FP instruction to fill polygons that have been defined with the PM instruction.

SYNTAX: *FP term*

PARAMETERS: None.

EXPLANATION: The FP instruction fills the polygon that has been previously defined in the polygon buffer. Valid polygons include those defined by the PM, RA, RR, and WG instructions. (However, since RA, RR, and WG each include the fill feature, you will probably only use FP to fill polygons defined by PM.) While the FP instruction does access the data in the polygon buffer, note that FP does not clear the buffer or change the data in any way.

The polygon is filled with the current pen, the current fill type, and the current line type. If subpolygons are defined, the FP instruction fills alternating areas with what is known as an "odd/even" algorithm. (To see how this works, refer to the examples at the end of this section.) Upon completion of the polygon, the original pen position and status (up/down) are restored.

The following table summarizes the possible error conditions for the FP instruction.

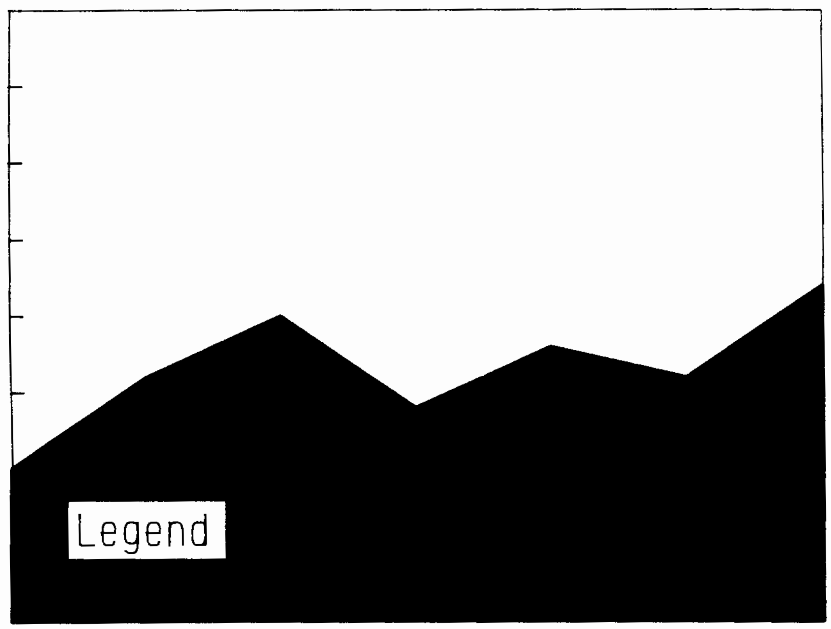
Condition	Error	Plotter Response
previous PM, RA, RR, or WG instruction overflowed the buffer	none	instruction ignored
no polygon previously defined	none	instruction ignored

Example — Creating a Surface Chart

The following example shows how to use the PM and FP instructions to draw and fill a curve in a surface chart, leaving a blank area for the legend. (This program also uses tick and labeling instructions that are described in Chapters 6 and 7.)

Send:

```
  LINE 1.  IN;IP-5000,5000,4000,12000;
  LINE 2.  SC-20,80,-10,60;
  LINE 3.  SP2;PA0,0;
  LINE 4.  PM0;PD0,0,0,10,10,16;
  LINE 5.  PD20,20,30,14,40,18,50,16;
  LINE 6.  PD60,22,60,0,0,0;PM1;
  LINE 7.  PU4,4;PD4,8,16,8,16,4,4,4;PU;PM2;
  LINE 8.  FP;EP;
  LINE 9.  SP1;SI.25,.5;
  LINE 10. PU5,5;LBLegend ;
  LINE 11. PA0,0;ER60,40;
  LINE 12. TL1.5,0;
■ LINE 13. FOR X=0 TO 60 STEP 10
★ LINE 14. PA";X,"0;XT;
■ LINE 15. NEXT X
  LINE 16. TL1,0;
■ LINE 17. FOR Y=0 TO 40 STEP 5
★ LINE 18. PA0,",Y,"YT;
■ LINE 19. NEXT Y
```



Line 1 initializes the plotter; positions P1 and P2

Line 2 scales the plotting area to user units (subsequent plotting instructions will now be interpreted as user units)

Line 3 selects pen 2; sets the initial pen position

Line 4 clears the polygon buffer and enters polygon mode; lowers the pen and defines the first three vertices of the outside of the surface curve, beginning at the user-unit origin

Line 5 defines the next four vertices

Line 6 defines the last three vertices (including the closing vertex); closes this subpolygon

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Line 7 moves with the pen up to the starting position for the rectangular "legend" area (this is also the first vertex); lowers the pen and defines the four vertices (including the closing vertex); lifts the pen; exits polygon mode

Line 8 fills the surface curve using default fill conditions (solid, bidirectional); edges the surface curve and legend area

Line 9 selects pen 1; defines the character size

Line 10 moves with the pen up to the starting position for the label; draws the label ("Legend")

Line 11 moves to the user-unit origin; defines and edges a rectangle around the chart

Line 12 defines the size of the X-axis tick marks

Line 13 BASIC statement that starts the loop for drawing the X-axis ticks at intervals of 10 user units

Line 14 draws to the tick location; draws the tick

Line 15 BASIC statement that ends the loop

Line 16 defines the size of the Y-axis tick marks

Line 17 BASIC statement that starts the loop for drawing the Y-axis ticks at intervals of 5 user units

Line 18 draws to the tick location; draws the tick

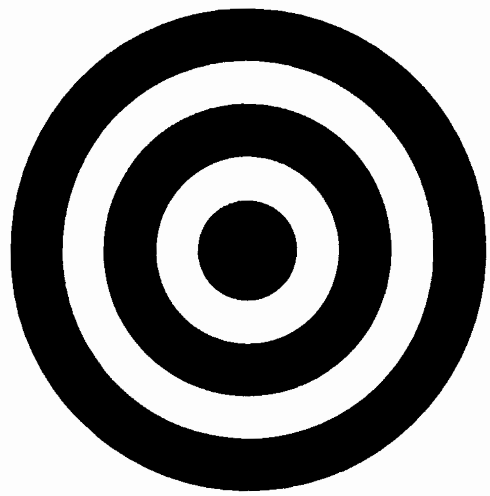
Line 19 BASIC statement that ends the loop

Example — Filling Alternating Subpolygons

The following example shows the effects of defining and filling several subpolygons within one PM0 instruction. In addition, it shows how to use the ESC . T instruction to increase the polygon buffer size in order to accommodate all of the subpolygons. (For further details, refer to The Polygon Buffer at the end of this chapter, and The Configure Memory Instruction, ESC . T in Chapter 13.)

Send:

```
LINE 1.   ESC.T 1000;6000;0:
LINE 2.   IN;SP1;IP5000,5000,10000,10000;
LINE 3.   PA7500,7500;
LINE 4.   PM0;CI250;CI500;CI750;CI1000;CI1250;PM2;
LINE 5.   FP;EP;SP0
```



Line 1 allocates 6000 bytes to the polygon buffer

Line 2 initializes the plotter; selects pen 1; positions P1 and P2

Line 3 sets the initial pen position

Line 4 clears the polygon buffer and enters polygon mode; defines five subpolygons consisting of circles with increasing radii; exits polygon mode

Line 5 fills alternating subpolygons using default fill conditions (solid, bidirectional); edges the subpolygons; puts pen away

The Polygon Buffer

When a polygon is created with the PM, RA, RR, or WG instructions, the points defining the polygon are stored in the polygon buffer. When you want to fill or edge a polygon, the points are accessed from this buffer by the FP, EP, RA, RR, and WG instructions.

The size of the polygon buffer is determined by the second parameter in the ESC . T instruction. This section describes how to determine what the size of the buffer should be. In many cases, you can leave the buffer at the default size of 3072 bytes. However, if you are defining a large polygon, you must increase the buffer to avoid error 7, buffer overflow. Also, if you are not defining polygons and need extra bytes for one of the other buffers (such as the downloadable character set buffer), you might want to decrease the polygon buffer size. For specifics on using the ESC . T instruction to change the buffer sizes, refer to The Configure Memory Instruction, ESC . T, in Chapter 12 or 13.

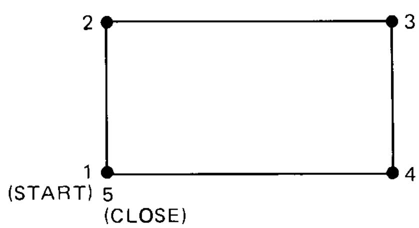
Determining the Size of the Polygon Buffer

To determine how much space to set in the buffer, you need to convert the number of points in your polygon into bytes. The following formula is simple and will suit most situations.

$$\# \text{ of bytes} = \# \text{ of points} \times 14$$

However, this formula is approximate. If you need a more accurate determination, read the next section. The default size of the polygon buffer — 3072 bytes — will accommodate approximately *218* points. The buffer accepts only even-numbered allocations. If you allocate an odd number of bytes, the plotter will automatically round up to the next even integer.

Each vector that you define with a coordinate pair is a point (also called a vertex). Thus, a rectangle would consist of 5 points. A rectangle has 4 corners, but the starting and closing positions count as 2 points, as shown below.

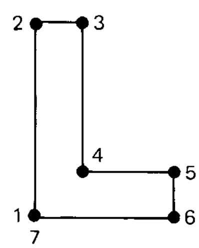



```
PU0,0;PM0;PD0,100,200,100,
200,0,0,0;PM2
```

OR

```
PA0,0;RA200,100
```

The following shape has 7 points.




```
PU0,0;PM0;PD0,1000,200,1000,
200,200,600,200,600,0,0,0;PM2
```

The number of points in an arc or circle is equal to the number of chords in the arc, plus one. You can use this formula:

$$\# \text{ of points} = \frac{\text{arc size (degrees)}}{\text{chord angle (degrees)}} + 1$$

Thus, a full circle with the default chord angle of 5 degrees consists of 73 points ($360°/5° + 1 = 73$). A 45-degree arc with an angle of 3° consists of 16 points ($45°/3° + 1 = 16$).

If the chord angle does not divide evenly into the arc, the remainder is rounded up to the next integer before adding one: $(45°/2° + 1) = (23 + 1) = 24$.

This calculation changes if the chord tolerance is in deviation distance mode rather than degrees mode. For more information, see the CT instruction in this chapter.

How Polygons Are Stored

Of course, determining the size of a polygon is more complicated than implied in the previous paragraphs. In most situations, the methods described will provide the best approximation. However, if you want to know the details of how bytes are allocated, read this section.

The total number of bytes used by a polygon depends on the following:

1. the number of points (vertices) used to define the polygon
2. the number of subpolygons
3. the number of changes in pen status (PU/PD) entered within the PM instruction

At the start of a new polygon or subpolygon, the "boundary pen position" is always considered to be UP. Whenever the pen is put down, a one-byte flag will be put into the buffer. Whenever the pen is subsequently picked up, a one-byte flag will be put into the buffer.

The PU or PD instructions do not by themselves cause a byte to enter the buffer; the change in pen status is recorded in the buffer only when an actual point is entered.

In the following example, the first three pen instructions are not recorded in the buffer.

```
PD;PU;PD;PD0,100,200,100,200,0,0,0
```

Also note that implied pen changes, such as PD with a CI instruction, are stored in the buffer.

Here are the steps to determine the number of bytes required for a given polygon.

1. Make the following allocations:
   - 1 byte for each new series of points

     A new series of points is started by each of these conditions: a pen state change (PU/PD), a subpolygon closure (PM 1), or whenever a series consists of more than 256 points.
   - 1 byte that indicates how many points are in a series
   - 8 bytes for each coordinate pair in a series
   - 1 byte for each polygon closure (PM 1 or PM 2)
   - 1 byte for each change in pen status (PU/PD)
2. Multiply this total by 1.5. This allows for bytes that are needed for overhead. If the resulting number is a fraction, round up to the next integer.
3. If this total is an odd number, round up by one to the next even integer.

For instance, if T = the total number of bytes allocated in step 1, the final total would be

$(T \times 1.5) +$ rounding up to the next even integer, if necessary.

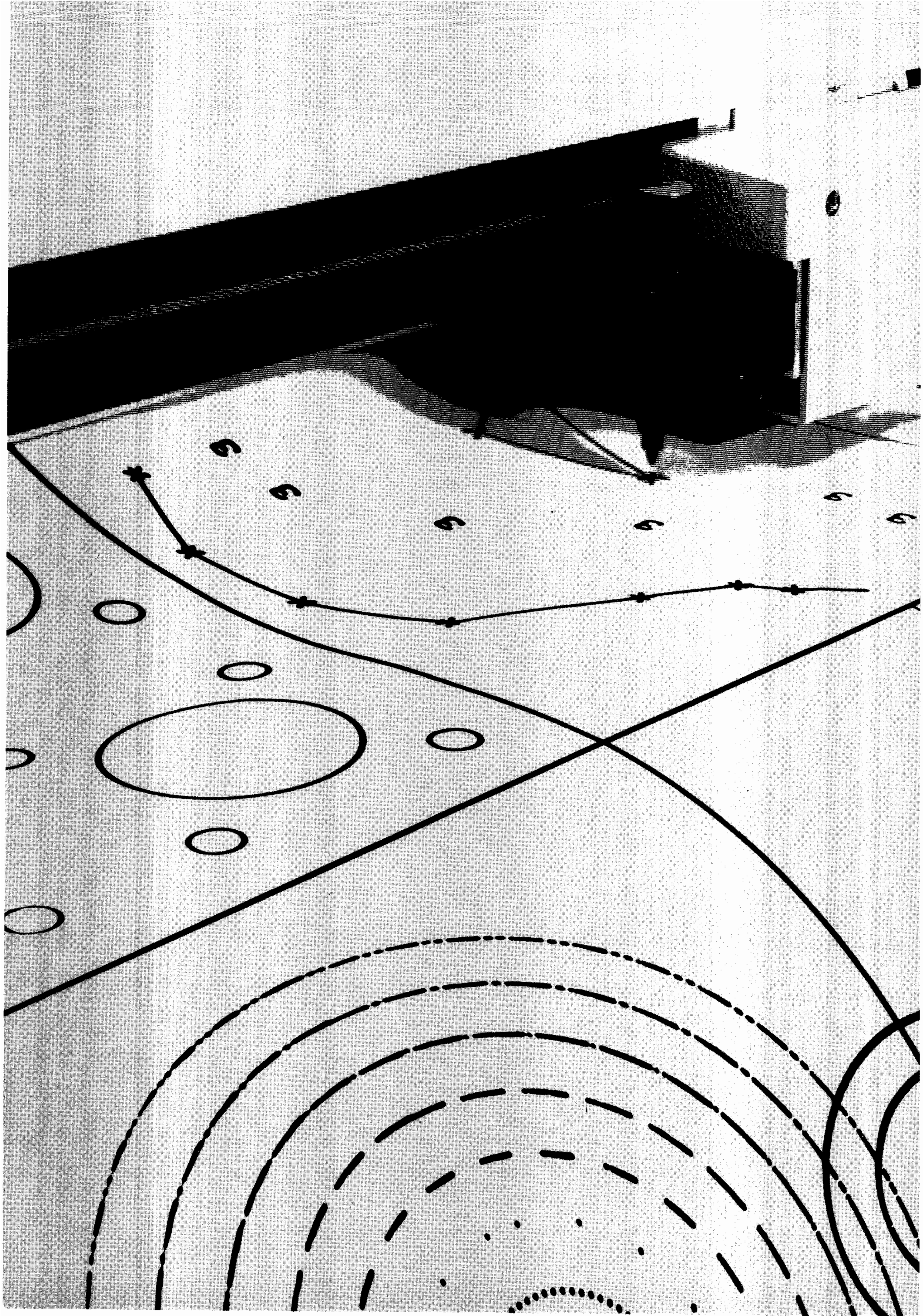
NOTE: A CI instruction in polygon mode is treated as though preceded and followed by a PM 1; these two PM 1 flags are stored in the buffer, plus one byte for the implied pen change. ■

Because a PM 1 represents a PU (see number 3 above), PM 1 will cause a PD to be put into the buffer after it if the pen was down before the PM 1 and stays down for the next point after the PM 1.

Here's an example. If scaling is off, the instructions "PA 0,0;RA 200,200" cause 70 bytes to be placed in the buffer, as follows:

Start first series of points	1 byte
One point in series (byte count)	1 byte
Point	8 bytes
Pen down	1 byte
Start second series of points	1 byte
Four points in series (byte count)	1 byte
Point	8 bytes
Point	8 bytes
Point	8 bytes
Point	8 bytes
Close polygon	1 byte
Total	46 bytes
46×1.5 =	69 bytes
69 rounded up to the next even integer =	70 bytes

Circles & Polygons



Chapter **6**

Plot Enhancement

What You'll Learn in This Chapter

Now that you know how to draw lines, let's look at some ways to enhance your plots. In this chapter you will learn how to draw tick marks on axes or use them to create grids, how to draw a symbol or character of your choice at each data point, and how to draw with dashed or dotted line patterns. Using these enhancements makes your data easier to interpret.

HP-GL Instructions Covered

XT The X-Tick Instruction
YT The Y-Tick Instruction
TL The Tick Length Instruction
SM The Symbol Mode Instruction
LT The Line Type Instruction

The Tick Instructions, XT and YT

USES: The XT instruction draws a vertical X-tick at the current pen location. The YT instruction draws a horizontal Y-tick at the current pen location. Use these instructions to draw tick marks on axes, draw grid lines by making the tick length 100%, or draw horizontal or vertical lines either centered on or ending at the current pen position.

SYNTAX: *XT term*
or
YT term

PARAMETERS: None.

EXPLANATION: These instructions include an automatic pen down feature. After the tick is drawn, the pen assumes the pen state (up or down) that was in effect prior to the tick instruction.

The length of the tick is specified by The Tick Length Instruction, TL. If no tick length has been specified, the default length is 0.5% of $|P2_X - P1_X|$ for YT or 0.5% of $|P2_Y - P1_Y|$ for XT.

Example — Using the XT and YT Instructions

The following example scales the plotting area into 1200 user units in X, draws a horizontal line 1200 user-units long, and places ticks at the endpoints and at the points 400,0 and 800,0.

Send:

```
LINE 1.   IN;SP1;IP10000,-5000,15000,0;
LINE 2.   SC0,1200,-10,10;
LINE 3.   PA0,0;PD;XT;PR400,0;XT;
LINE 4.   PR400,0;XT;PR400,0;XT;PU;
```

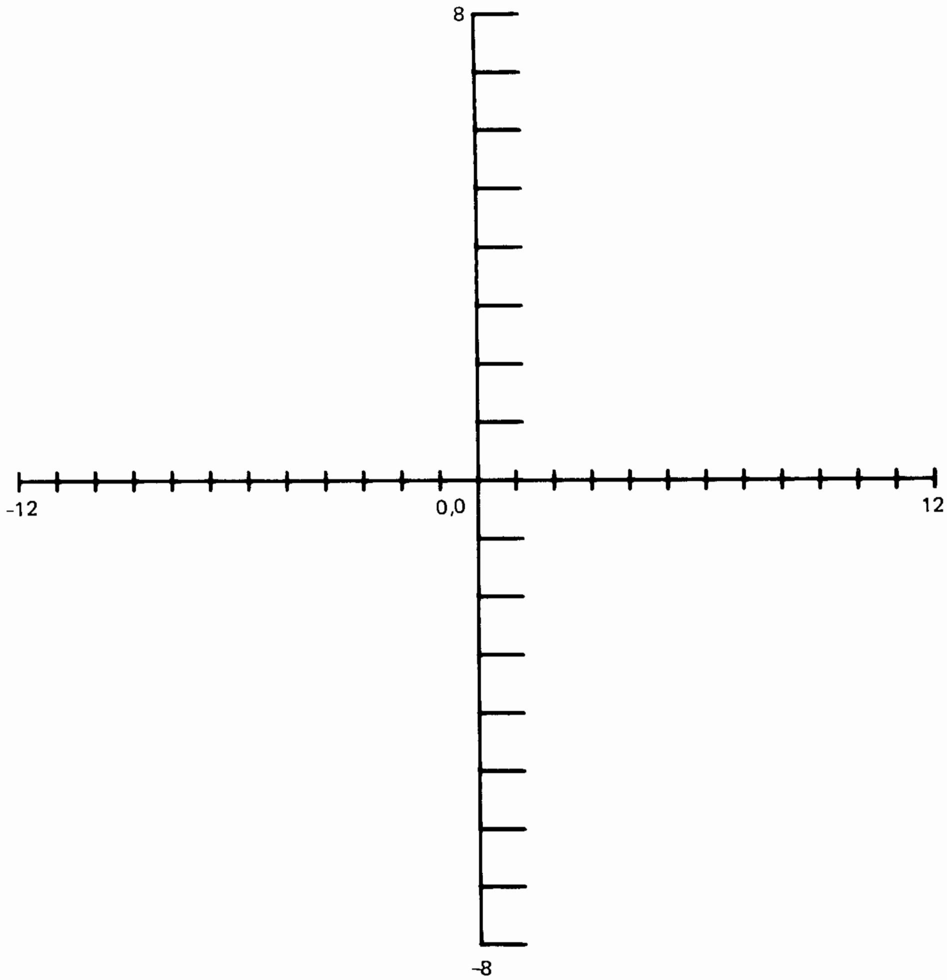
The following example demonstrates using XT and YT to place ticks along the X-axis from −12 to 12 and along the Y-axis from −8 to 8. A tick length of 1 percent is specified for both the positive and negative portions of the ticks on the X-axis. Only the positive portion of the Y-ticks are drawn at a length of 5 percent. Sometimes tick marks of different lengths are drawn on the same axis, the larger tick marks denoting major axis divisions. Refer to The Tick Length Instruction, TL, which follows.

Send:

```
  LINE 1.   IN;SP1;IP-15000,-10000,-10000,-5000;
  LINE 2.   SC-12,12,-8,8;TL1,1;
  LINE 3.   PA-12,0;
■ LINE 4.   FOR X=-12 TO 12
★ LINE 5.   PAPDX,0;XT;
■ LINE 6.   NEXT X
  LINE 7.   PAPU0,-8;TL5;
■ LINE 8.   FOR Y=-8 TO 8
★ LINE 9.   PAPD0,Y;YT;PU;
■ LINE 10.  NEXT Y
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.



The Tick Length Instruction, TL

USES: The TL instruction specifies the length of the tick marks on the X- and Y-axes. Use the instruction to set both the positive and negative portions of tick marks, or to draw a grid.

SYNTAX *TL* tp (, tn) *term*
or
TL *term*

PARAMETERS:

Parameter	Format	Range	Default
tp and tn	decimal	0 to $2^{26}-1$	0.5% of $\lvert P2_X - P1_X \rvert$ and $\lvert P2_Y - P1_Y \rvert$

EXPLANATION: A TL instruction with no parameters (TL ;) will set default values. A TL instruction with only one parameter specifies the length of tp; tn will be zero. The parameters are interpreted as described in the following paragraphs and illustrations.

1. Tp. The up and right tick length, tp, determines the length of the upward portion of the tick marks drawn along the X-axis and the right-side portion of the Y-axis.

2. Tn. The down and left tick length, tn, determines the length of the downward portions of the tick marks drawn down along the X-axis and the left side portion of the tick marks drawn along the Y-axis.

Both parameters are specified as a percentage of the horizontal and vertical distances between P1 and P2. The values are a percentage of the vertical scale length, $|P2_Y - P1_Y|$, when used with the XT instruction. With the YT instruction, the values are a percentage of the horizontal scale length, $|P2_X - P1_X|$. Note that the actual tick length is a function established by P1 and P2. Therefore, unless the area defined by P1 and P2 is square, the tick lengths on the X- and Y-axes will differ even if the same tick length percentage value is specified for both XT and YT.

The default value for both tp and tn is 0.5% of the horizontal and vertical distances between P1 and P2. A TL instruction remains in effect until another TL instruction is executed or the plotter is initialized or set to default conditions.

Use the instruction with only one parameter to suppress the negative portion of the tick mark, or with a first parameter, tp, of zero to suppress the positive portion of the tick. Setting the first parameter to 100 enables you to draw grids easily using the XT and YT instructions.

Example — Drawing a Grid

The following example draws a 6 × 4 grid within the area bounded by points P1 = −15 000, −10 000 and P2 = 15 000, 10 000. The grid is the result of specifying 100% tick length for all ticks except at X = 10 000, where 75% is specified. Note that only the up and right tick length parameter, tp, is specified. Refer to the previous section for further examples of the TL instruction.

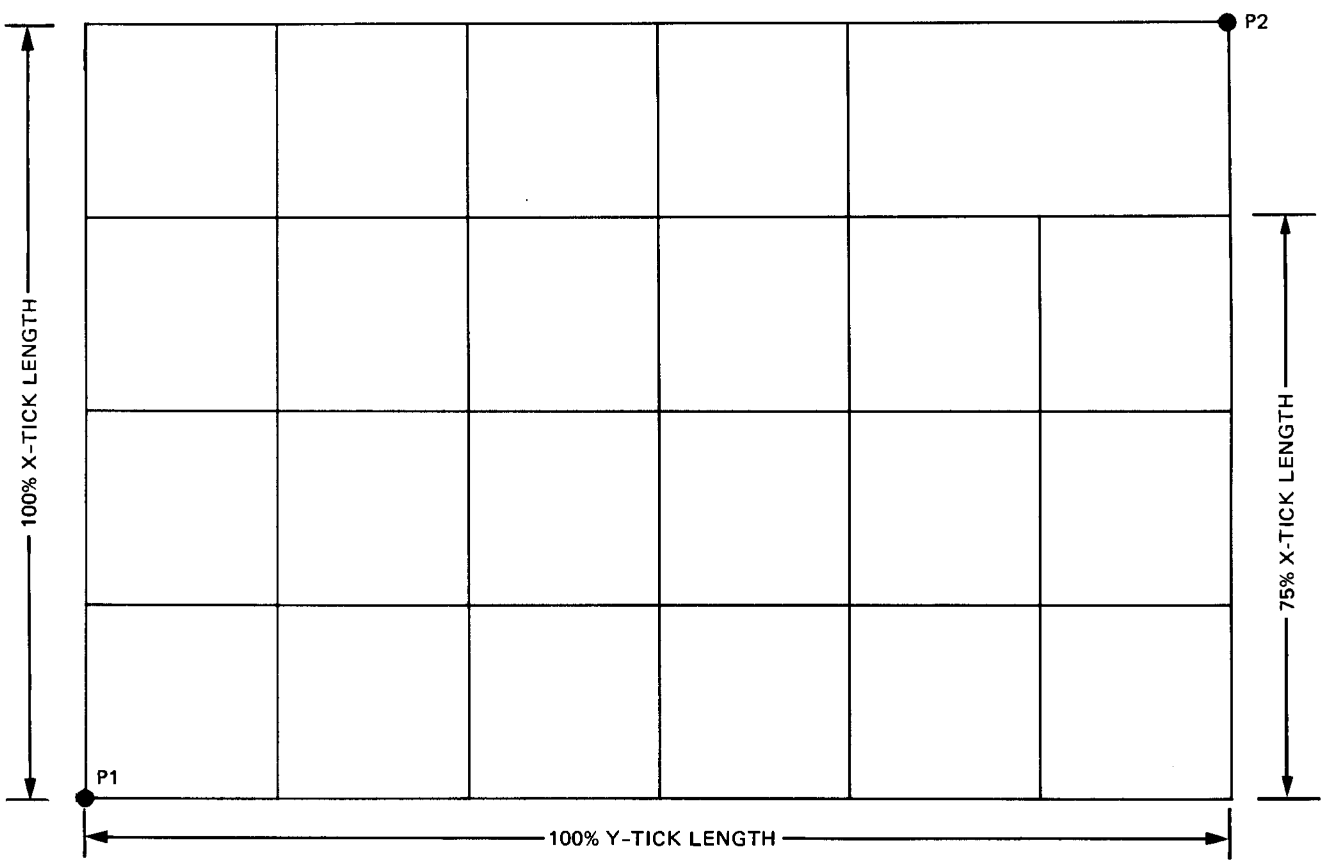
Send:

```
  LINE 1.   IN;SP1;
  LINE 2.   IP-15000,-10000,15000,10000;
  LINE 3.   PA-15000,-10000;TL100;
■ LINE 4.   FOR X=-15000 TO 15000 STEP 5000
  LINE 5.   IF X=10000 THEN PRINT "TL75;"
★ LINE 6.   PA,X,-10000;XT;TL100;
■ LINE 7.   NEXT X
■ LINE 8.   FOR Y=-10000 TO 10000 STEP 5000
★ LINE 9.   PA-15000,Y;YT;
■ LINE 10.  NEXT Y
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Tick Length Example



The Symbol Mode Instruction, SM

USES: The SM instruction is used with the PA and PR instructions to draw a symbol at the end of each vector. Use symbol mode plotting to draw a specified character at each data point in order to differentiate data contained in scattergrams, geometric drawings, or multiple-line graphs.

SYNTAX: *SM* character *term*
or
SM *term*

PARAMETERS: The parameter is a single character which must be one of the printing characters of the character set chosen.

EXPLANATION: An SM instruction can specify any printing character (decimal values 33-126) but the semicolon. The semicolon (decimal value 59) is a terminator and is used only to cancel symbol mode (SM ;).

After an SM instruction has been executed, subsequent PA and PR instructions function as described in Chapter 4, except that the symbol mode character is drawn at the end of each vector and is centered on the plotted point. The character is independent of the current pen state (up or down).

The character is drawn according to the character set selected as the SM instruction is executed. If you have defined downloadable characters in set −1, you can use these in symbol mode plotting. (Refer to The Downloadable Character Instruction, Chapter 7). Once selected, the character is independent of subsequent character set changes. An SM instruction remains in effect until another valid SM instruction is executed or the plotter is initialized or set to default conditions. The size (SI and SR), slant (SL), and direction (DI and DR) instructions affect the character drawn.

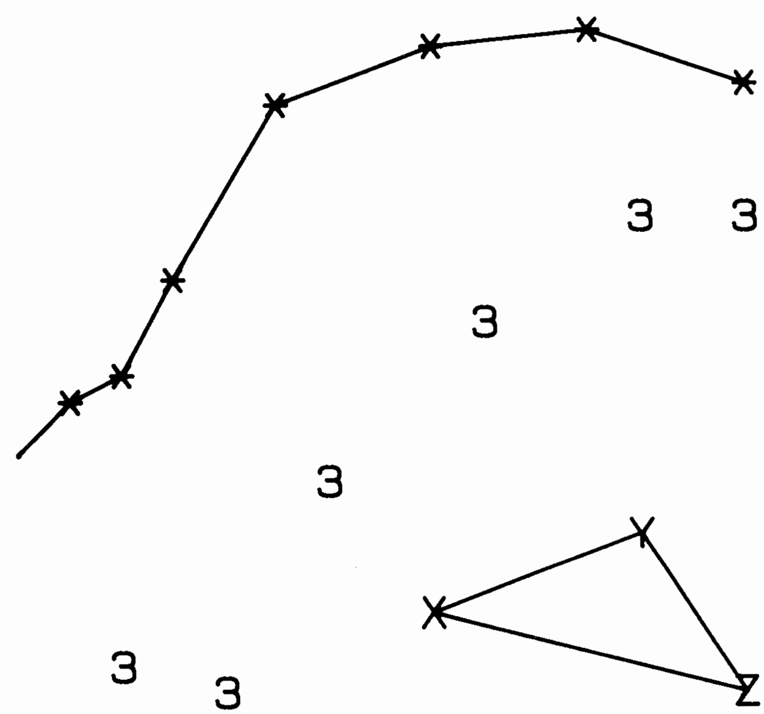
The following table summarizes the error conditions for the SM instruction.

Condition	Error	Plotter Response
specification of a space or any control character but a carriage return	3	no symbol plotted
specification of a carriage return (SM ↵)	none	cancels symbol mode

Example — Symbol Mode Plotting

The following example shows symbol mode plotting with the pen up and the pen down as might be used in line graphs, geometric drawings, and scattergrams.

```
LINE 1.   IN;SP1;IP-10000,-10000,-5000,-5000;
LINE 2.   SC0,180,0,180;PA10,90,PD;
LINE 3.   SM*;PA20,100,30,105,40,123,60,156,90,167;
          PA120,170,150,160;PU;
LINE 4.   SM;PA10,30;SM3;PA30,50,50,45,70,85,100,115;
          PA130,135,150,135;PU;
LINE 5.   SM;PA90,60;PD;SMY;PA130,75;SMZ;PA150,45;
          SMX;PA90,60;PU;
```



The Line Type Instruction, LT

USES: The LT instruction specifies the type of line that will be used with AA, AR, CI, EA, EP, ER, EW, FP, FT, PA, PR, RA, RR, WG, and UF instructions. Use the instruction to draw dashed or dotted lines in fixed or adaptive pattern lengths. This facilitates trace differentiation on multiple-line graphs and enables emphasis or deemphasis of plotted lines or grids. One line type causes only dots to be plotted at each data point.

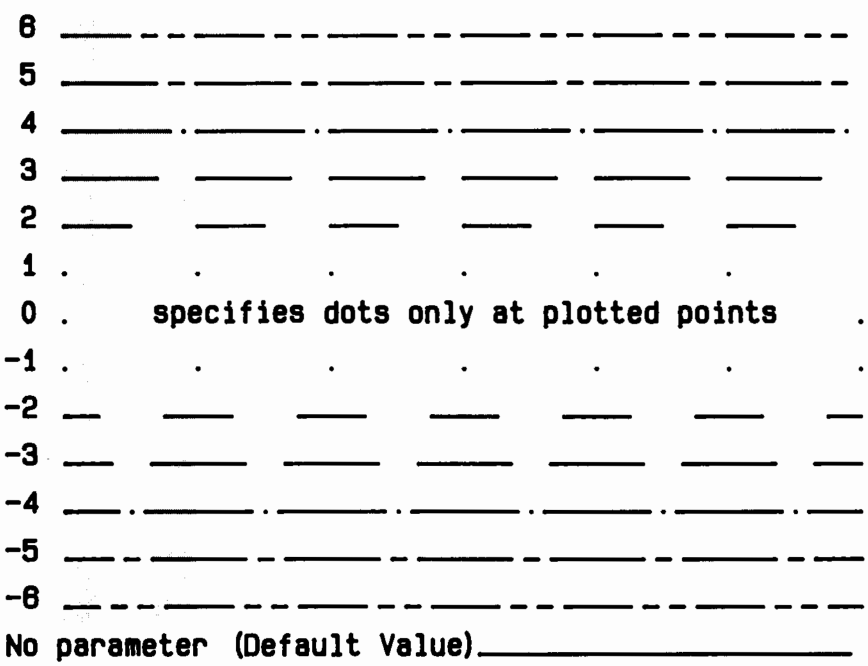
SYNTAX: *LT* pattern number (, pattern length) *term*
or
LT *term*

PARAMETERS:

Parameter	Format	Range	Default
pattern number	integer	−6 to +6	no parameter (solid line)
pattern length	decimal	0.0000000001 to $2^{26}-1$ (percentage)	4% of the distance between P1 and P2

EXPLANATION: An LT instruction with no parameters (LT ;) will default line type to a solid line. The pattern number and pattern length parameters are described in the following paragraphs and illustrations.

1. Pattern number. Shown below are the pattern numbers and the line patterns they each generate.



The shaded portion of each of the line patterns above is one complete segment of the pattern.

Specification of an out-of-range parameter generates error number 3. The pattern will be drawn with the last valid line type.

2. Pattern length. The optional pattern length parameter specifies the length of one complete pattern and is expressed as a percentage of the diagonal distance between the scaling points P1 and P2. If a pattern length parameter is not specified, the last value received is used. If no pattern length has ever been specified, a length of 4% is used.

After an LT instruction is executed, subsequent vectors drawn with AA, AR, CI, EA, EP, ER, EW, FP, FT, PA, PR, RA, RR, UF, and WG instructions will be drawn in the specified line type. Once selected, the pattern remains in effect until another valid LT instruction is executed or the plotter is initialized or set to default conditions.

NOTE: If a vector ends in the pen up portion of the pattern, a pen down instruction (PD) will not physically put the pen down until the next vector instruction is executed. The pen up instruction (PU) clears the carry-over portion of a pattern segment. ■

Patterns resulting from positive pattern numbers are drawn to the specified pattern length. Any portion of a pattern which is not used to draw the specified vector will be carried over into the next vector.

Patterns resulting from negative pattern numbers automatically adjust the specified pattern length to insure that the commanded vector contains one or more complete pattern segments.

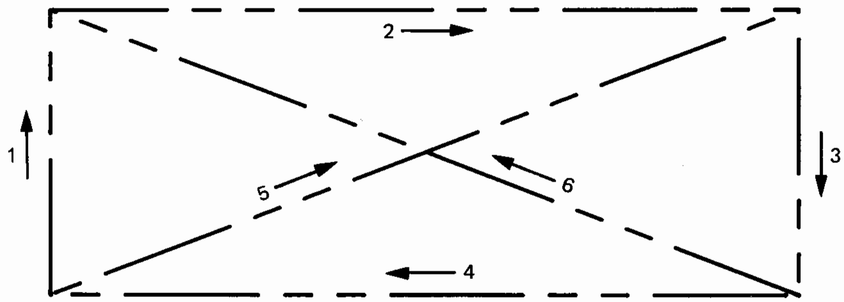
NOTE: When a negative pattern number is in effect during arc or circle generation, pattern length is adjusted to chord length. Because chord length is usually short, patterns cannot be easily distinguished. Use of positive pattern numbers is recommended when drawing arcs and circles. ■

Example — Fixed and Adaptive Line Types

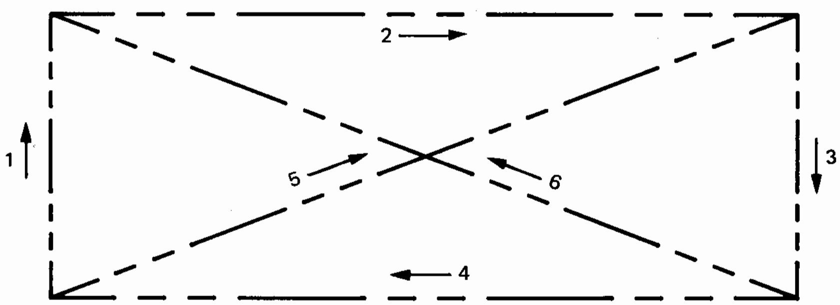
The following example demonstrates the difference between positive and negative pattern numbers. The sample program plots two rectangular boxes, one with line type 6 and one with line type −6. The pattern length in both cases is 20. Notice that each vector in the adaptive pattern length rectangle contains complete pattern segments; whereas, in the fixed pattern length rectangle, the unused portions of the pattern segments are carried over into the next vector. The lines are numbered in the order in which they were drawn.

Send:

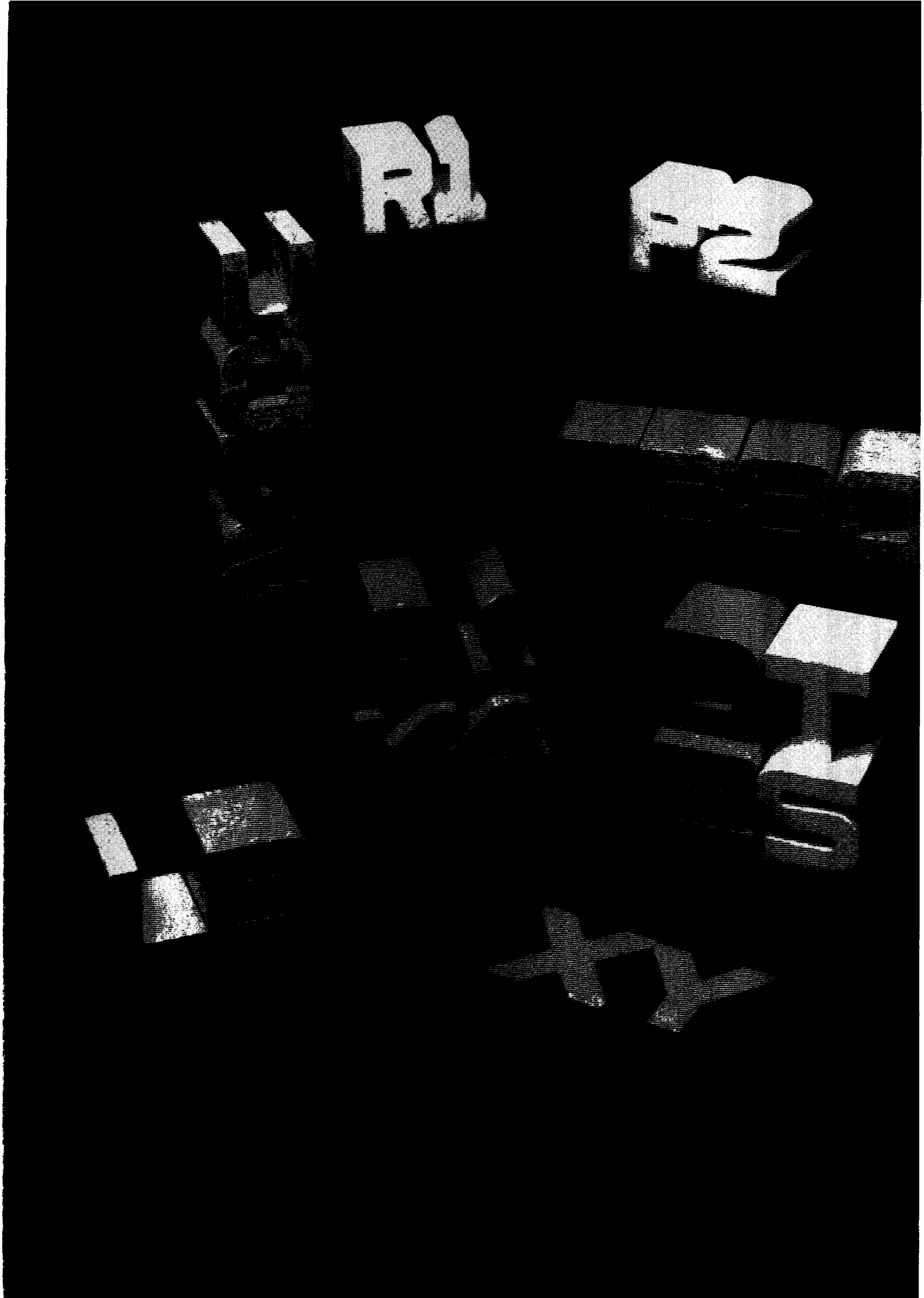
```
LINE 1.   IN;SP1;IP-5000,-10000,0,-5000;
LINE 2.   SC0,10,0,10;
LINE 3.   LT-6,20;PA1,1,PD,1,4,9,4,9,1,1,1,9,4,PU,9,1,
          PD,1,4;
LINE 4.   LO16;PAPU5,1;LBADAPTIVE PATTERN LENGTH␃
LINE 5.   LT6,20;PA1,6,PD,1,9,9,9,9,6,1,6,9,9,PU,9,6,
          PD,1,9;
LINE 6.   PAPU5,6;LBFIXED PATTERN LENGTH␃
LINE 7.   PA0,0;
```



FIXED PATTERN LENGTH



ADAPTIVE PATTERN LENGTH



Chapter 7
Labeling Plots

What You'll Learn in This Chapter

In this chapter you will learn about character sets and labels used to create effective annotated graphics. You will learn how to designate and select character sets, how to use the label instruction with both constant and variable parameters, and how to set the size, slant, and direction of labels. Character spacing, label positioning, buffered labels and characters, alternative character set modes, defining your own characters, and moving the pen any number of character spaces and/or lines will also be discussed.

HP-GL Instructions Covered

CS The Designate Standard Character Set Instruction
CA The Designate Alternate Character Set Instruction
SS The Select Standard Set Instruction
SA The Select Alternate Set Instruction
LB The Label Instruction
DT The Define Terminator Instruction
ES The Extra Space Instruction
DI The Absolute Direction Instruction
DR The Relative Direction Instruction
CP The Character Plot Instruction
SI The Absolute Character Size Instruction
SR The Relative Character Size Instruction
SL The Character Slant Instruction
LO The Label Origin Instruction
BL The Buffered Label String Instruction
OL The Output Label Length Instruction
PB The Print Buffered Label Instruction
CC The Character Chord Angle Instruction
UC The User-Defined Character Instruction
DL The Downloadable Character Instruction
CM The Character Selection Mode Instruction
DS The Designate Character Set into Slot Instruction
IV The Invoke Character Slot Instruction

Terms You Should Understand

Label Terminator — the final character in every label string. It takes the plotter out of label mode so that characters are no longer drawn but are again interpreted as HP-GL instructions and parameters. Its default value is the ASCII character ETX (decimal equivalent 3) but it may be redefined using the DT instruction.

Plotter Character Sets

A total of 62 character sets are available on the plotter. The plotter has 20 character sets available on each of three fonts. In addition to these 60 sets, the plotter contains set 99, the Drafting set, and set −1, the Downloadable set. Downloadable characters are characters you define yourself and store using a program. This set is explained in more detail later at the end of this chapter.

Note that "font" simply denotes a style of lettering, which is described below.

Fixed-space vector font:	Characters all 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. Refer to Character Spacing and the Character Chord Angle Instruction, CC, in this chapter.
Fixed-space arc font:	Characters all occupy an equal horizontal space and contain the same programmable contour smoothness of an arc-drawn charactr.

In the table on the next page, each of the 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).

In each type of font, a line feed causes vertical movement equal to twice the height of the letter "A."

All sets except Special Symbols (sets 5, 15, and 25), Roman Extensions (sets 7, 17, and 27), and Katakana (sets 8, 18, and 28) draw identical upper- and lowercase letters and numbers. With these exceptions, sets differ only in the additional characters which are needed in a certain language, for example German in set 2.

Character sets 0, 10, and 20 from the three respective fonts are shown on the next page. The difference in the total space required to plot the sets emphasizes the difference between fixed-space fonts and variable-space fonts. Compare also the variation in the smoothness of the characters between the arc fonts and the vector font. All of the character sets are shown in Appendix C.

Fixed-Space Vector Font	Variable-Space Arc Font	Fixed-Space Arc Font	Character Set	ISO Registration Number
0	10	20	ANSI ASCII	006
1	11	21	9825 Character 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
30	40	50	ISO Swedish	010
31	41	51	ISO Swedish For Names	011
32	42	52	ISO Norway, Version 1	060
33	43	53	ISO German	021
34	44	54	ISO French	025
35	45	55	ISO United Kingdom	004
36	46	56	ISO Italian	015
37	47	57	ISO Spanish	017
38	48	58	ISO Portuguese	016
39	49	59	ISO Norway, Version 2	061
99	—	—	Drafting	—

CHARACTER SET 0

```
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
```

CHARACTER SET 10

```
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
```

CHARACTER SET 20

```
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_'
abcdefghijklmnopqrstuvwxyz{|}~
```

The Drafting Set

Set 99, the Drafting set, is designed to provide reliable character recognition in situations where photo reduction may cause image degradation and loss of resolution. The characters are drawn in a way to avoid confusion between lines and figures such as

B 8 S

V U

5 6 S

The set also contains symbols used in drafting, such as ∞ or Δ. Refer to Appendix C to see the entire set. The following table describes some characters in more detail.

-B-	Bottom wider than top.
-6-	Large body, stem curved and open.
-8-	Lower part larger than upper, full and round to avoid blur.
-9-	Large body, stem curved but open.

Character Set Modes

The plotter can generate character sets in any of four different modes. Refer to Choosing Other Character Selection Modes in this chapter to determine if you wish to use a mode other than the plotter's default mode.

Regardless of mode, the plotter can label using either a standard character set or an alternate character set. Under default conditions, the plotter automatically sets both the standard set and the alternate set to ANSI ASCII character set 0.

The Designate Standard Character Set Instruction, CS

USES: The CS instruction designates one of the character sets as the standard character set. Use the instruction to change the standard character set to one with characters appropriate for your application. It is especially useful when labels are in a language other than English.

SYNTAX: *CS* set *term*
or
CS *term*

PARAMETERS:

Parameter	Format	Range	Default
character set	integer	−1, 0–59, 99	0

EXPLANATION: The character set designated by the CS instruction is used for all labeling operations when the standard set is selected by the SS instruction, by the control character shift-in (decimal 15) in a label string, or by IV 0,0. Character set 0 is automatically designated as the standard character set whenever the plotter is initialized.

CS instructions executed while the standard set is selected will immediately change the character set used for labeling.

CS instructions executed while other slots are selected will not change the set used for labeling until the standard set is selected with the SS or IV instruction or the shift-in character.

A CS instruction with no parameters (CS ;) defaults to set 0.

The following table summarizes the error conditions for the CS instruction.

Condition	Error	Plotter Response
invalid parameter	3	ignores instruction
designate unsupported character set	5	ignores instruction
extra parameters	2	uses first parameter

Labeling Plots

The Designate Alternate Character Set Instruction, CA

USES: The CA instruction designates one of the character sets as the alternate character set. Use the instruction to provide an additional character set that you can easily access in a program, especially when a single label contains characters found in two different sets.

SYNTAX: *CA* set *term*
or
CA *term*

PARAMETERS:

Parameter	Format	Range	Default
character set	integer	−1, 0-59, 99	0

EXPLANATION: The character set designated by the CA instruction is used for all labeling operations when the alternate set is selected by the SA instruction, by the control character shift-out (decimal 14) in a label string, or by IV 1,0. Character set 0 is automatically designated as the alternate character set whenever the plotter is initialized.

CA instructions executed while the alternate set is selected will immediately change the character set used for labeling.

CA instructions executed while a slot other than G1 is selected will not change the set used for labeling until (if ever) the alternate set is selected with the SA or IV instruction or the shift-out character.

A CA instruction with no parameters (CA ;) defaults to set 0.

The following table summarizes the error conditions for the CA instruction.

Condition	Error	Plotter Response
invalid parameter	3	ignores instruction
designate unsupported character set	5	ignores instruction
extra parameters	2	uses first parameter

The Select Standard Set Instruction, SS

USES: The SS instruction selects the standard set designated by the CS or DS 0 instruction as the character set to be used for all labeling. Use the instruction to shift from any other character set slot to the currently designated standard character set so characters in another set can be accessed. Using the control character shift-in inside a label string or using IV 0 , 0 is equivalent to executing this instruction.

SYNTAX: *SS term*

PARAMETERS: None. The terminator must be included to complete the instruction.

EXPLANATION: An SS instruction with parameters will set an error condition (error 2), but the standard set is invoked.

The ANSI ASCII character set (set 0) is automatically selected when the plotter is first turned on or initialized. The standard set can be selected within a label instruction by sending the ASCII control character for shift-in (decimal 15).

The Select Alternate Set Instruction, SA

USES: The SA instruction selects the alternate set designated by the CA or DS 1 instruction as the character set to be used for all labeling. Use the instruction to shift from any other character set slot to the currently designated alternate character set to access characters in a second set. Using the control character shift-out inside a label string or using IV 1 , 0 is equivalent to executing this instruction.

SYNTAX: *SA term*

PARAMETERS: None. The terminator must be included to complete the instruction.

EXPLANATION: An SA instruction with parameters will set an error condition (error 2), but the alternate set is invoked.

This instruction should be executed prior to executing a labeling statement whenever the alternate character set is to be used. The alternate set can be selected within a label instruction by sending the ASCII control character for shift-out (decimal 14). Shift-in and shift-out are particularly useful when a line of text must be composed with symbols from two character sets.

The following instructions label using two different character sets where the underline is drawn with and without a backspace. The shift-out character is used to change from the standard to the alternate set 4.

```
'SP2;CS0;CA4;SS;LBS_E_T_0_%S_E_T_4_%"
```

```
S_E_T_0_SET4
```

The Label Instruction, LB

USES: The LB instruction letters text, expressions, or string variables using the currently defined character set. Use the instruction to annotate drawings or create text-only flip charts.

SYNTAX: *LB* c . . . c terminator
where the terminator is as defined by the DT instruction
or
LB terminator

PARAMETERS: The parameters (c . . . c) consist of any number of printing or control-code characters.

EXPLANATION: All printing characters following the LB instruction are drawn using the currently defined character set.

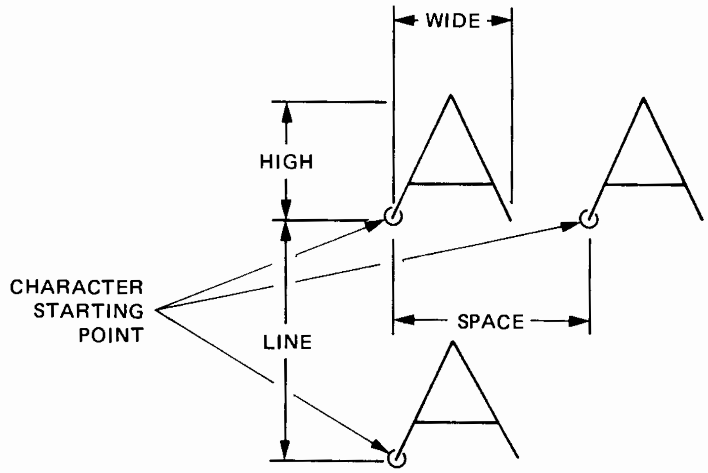
The character set used is specified by the instructions CA, CS, and DS and selected by the instructions SS, SA, or IV or the ASCII control characters shift-out or shift-in (decimal equivalent 14 and 15, respectively). If not specified, the default character set (set 0) is used.

The direction, size, and slant of the characters and the spacing between characters assume default values if not previously specified by the DI, DR, SI, SR, SL, or ES instructions. The label begins at the current pen position unless its placement has been set using an LO (label origin) command.

The label mode can be terminated only by sending a label terminator at the end of the character string or by any of the operations that clear the plotter. Refer to The Define Terminator Instruction, DT, following this discussion.

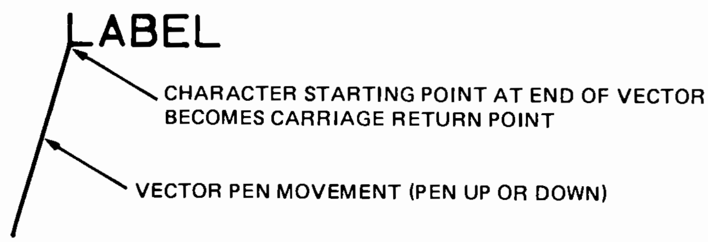
Before executing the LB instruction, the pen should be moved to the location where labeling is to begin using one of the plot instructions (PA, PR, AA, AR, or CP) or by using one of the front-panel controls. This establishes the carriage return point and the lower-left corner of the first character (unless modified by the LO instruction).

The illustration below shows vertical and horizontal spacing for fixed-font sets. For a further explanation of character spacing, refer to Character Spacing in this chapter.



Control characters, such as carriage return, line feed, or inverse line feed (VT) can be included in the label character string. These characters are not labeled, but cause the specified function to be executed. For example, a carriage return character causes the pen to return to a defined carriage return point prior to labeling subsequent characters in the label character string. The carriage return point is established as follows:

Vector Group Instructions — The endpoint of the last vector drawn with a PA, PR, AA, or AR instruction or the center of the circle drawn with a CI instruction becomes the carriage return point.



Direction Instructions — The current pen position when a DI or DR instruction is executed becomes the carriage return point. Refer to the DI instruction example in this chapter.

Update Instruction — The current pen position when a DF, IN, or RO instruction or a front-panel **ENTER RESET**, **ENTER CLEAR**, **ROTATE**, or **ENTER ROTATE**, or an interface clear function is executed becomes the carriage return point.

Front-Panel Controls — The point moved to using a front-panel control becomes the carriage return point.

NOTE: Interface clear functions are HP-IB device clear and interface clear and RS-232-C ESC . K. ■

Characters in the label string are placed in a 150-character buffer. The setting of the label origin determines when labels or segments of labels are written from the buffer. When label origins 1 through 3 or 11 through 13 are in effect, the last label may be repeated by executing the PB instruction. When other label origins are in effect, only the part of the label which follows an embedded carriage return (if any) will be repeated by the PB instruction. Executing another LB or BL instruction completely overwrites any previous contents of the label buffer. The label buffer is cleared by executing a front-panel reset, a DF or IN instruction, or an LB or BL instruction with only the label terminator.

The LB instruction is a NOP (no operation) if executed when the plotter is in lost mode (refer to The Plot Absolute Instruction, PA, Chapter 4). Lost mode may also be entered during execution of a label instruction. Depending on the current label origin parameter, part of the label may or may not be drawn. In addition, position overflow will occur if the length of the character string's base line exceeds the character generator's numeric range of 2^{32} primitive grid units. (Refer to The User-Defined Character Instruction, UC, in this chapter.) The position-overflow condition sets an error 6. (Refer to The Output Error Instruction, OE, Chapter 9.) It is almost certain that lost mode will be entered before this position-overflow condition occurs. The position-overflow condition, but not the error, is cleared by a UC or a CP instruction without parameters; by any of the actions which reset the carriage return point; by the joystick; or by a carriage return in the label character string, assuming the plotter is not in lost mode.

The following example illustrates the use of the LB instruction and shows that the contents of the label buffer can be drawn at a different location. The example is drawn using default condition.

Send:

```
LINE 1.   IN;SP1;IP0,-10000,5000,-5000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA300,4000;LBLabel strings are buffered.␍␊␃
LINE 4.   LBLast string is repeatable.␃
LINE 5.   PA300,3000;PB;
LINE 6.   PA300,2500;PB;
```

```
Label strings are buffered.
Last string is repeatable.


Last string is repeatable.

Last string is repeatable.
```

The Define Terminator Instruction, DT

USES: The DT instruction specifies the character to be used as the label terminator. Use the instruction to change the label terminator from its default value if ETX (decimal equivalent 3) cannot be used by your computer.

SYNTAX: *DT* t *term*
or
DT *term*

PARAMETER: The parameter t is the ASCII character designated as the terminator.

EXPLANATION: The label mode initiated by LB or BL instructions can only be terminated by sending a label terminator at the end of the label character string or by the interface clear functions listed earlier under the LB instruction. The ASCII character ETX (decimal equivalent 3) is the default label terminator. Issuing the instruction without parameters restores the default terminator.

With a few exceptions, ASCII control characters can be used as a label terminator and will not print when invoked, although the function normally performed by the character will be performed (i.e., CR will terminate but will also cause a carriage return).

ASCII characters NULL, ESC, LF, and ; (decimal equivalents 0, 27, 10, and 59, respectively) cannot be used as label terminators. The latter two restore the terminator to default (ETX).

ASCII characters (decimal equivalents 33 through 58 and 60 through 126) can also be used as the terminator, but the character will be printed at the end of the label character string.

The following examples of text in a label instruction demonstrate the use of the label terminator.

Send:

```
LINE 1.   IP:SP1;IP5000,-10000,10000,-5000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA0,4500;LBDefault control character ETX ␍␊␃
LINE 4.   LBterminates by performing end-␍␊␃
LINE 5.   LBof-text function.␃
LINE 6.   PA0,3250;DT#;LBPrinting characters terminate,␍␊#
LINE 7.   LBbut are also printed.#
LINE 8.   PA0,2250;DT␍;LBControl characters terminate␊␍
LINE 9.   LBand perform their function.␍
```

Default control character ETX terminates by performing end-of-text function.

Printing characters terminate, #but are also printed.#

Control characters terminate and perform their function.

Lines 3, 4, 5 write labels with the default label terminator ETX (decimal equivalent 3) which will not print at the end of a label string.

Lines 6, 7 define the ASCII character # (decimal equivalent 35) and use it as the label terminator. Notice that the # symbol is printed as the first and last character of the second text line. The first # symbol terminates the label string specified in line 6, but is printed at the start of the next text line since it follows the line feed and carriage return control characters. The second # symbol terminates the label string specified in line 7.

Lines 8, 9 define the ASCII control character CR (decimal equivalent 13) and use it as the label terminator. Notice that the ASCII control character LF (decimal equivalent 10) is used in conjunction with the CR terminator to shift the carriage return point down one line. The character CR terminates the label string in both lines and causes the return to the carriage return point.

Character Spacing

Character spacing and line spacing are functions of character size as set by the SI and SR instructions and any extra spacing set by an extra space, ES, instruction. The diagrams which follow show fixed-space font and variable-space font characters in relation to a nominal character plot cell (NCP). This cell is twice as high and 1-1/2 times the width of an uppercase "A." This cell can be adjusted for more or less spacing as specified in an ES instruction. The adjusted cell is called a character plot cell (CP) and is used to determine character and line spacing and as the spaces and lines units in a CP instruction.

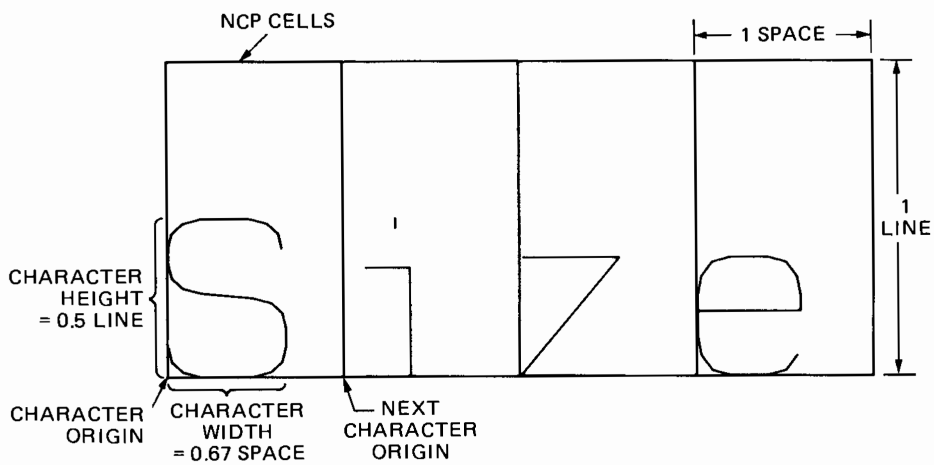
All characters are drawn on a primitive grid which is mapped onto the nominal character plot cell (NCP). All vector endpoints in a character fall at points of intersection on this grid. The grid rubbers with changes in the character size. Refer to The User-Defined Character Instruction, UC, in this chapter for more information.

When labeling is being done with a fixed-space font set, the distance between lines and the distance from one character origin to the next is determined solely by the size of the CP cell. When labeling is being done with a variable-space font set, the distance between lines is determined by the height of a CP cell. However, the width of characters and the space between characters varies. Therefore, the distance varies between character origins depending on the character combinations used. In both fonts, a space parameter of 1 in a CP instruction will always move the pen the width of one CP cell (1-1/2 times the width of an uppercase "A" as set by an SI or SR instruction, plus or minus any amount specified by an ES instruction).

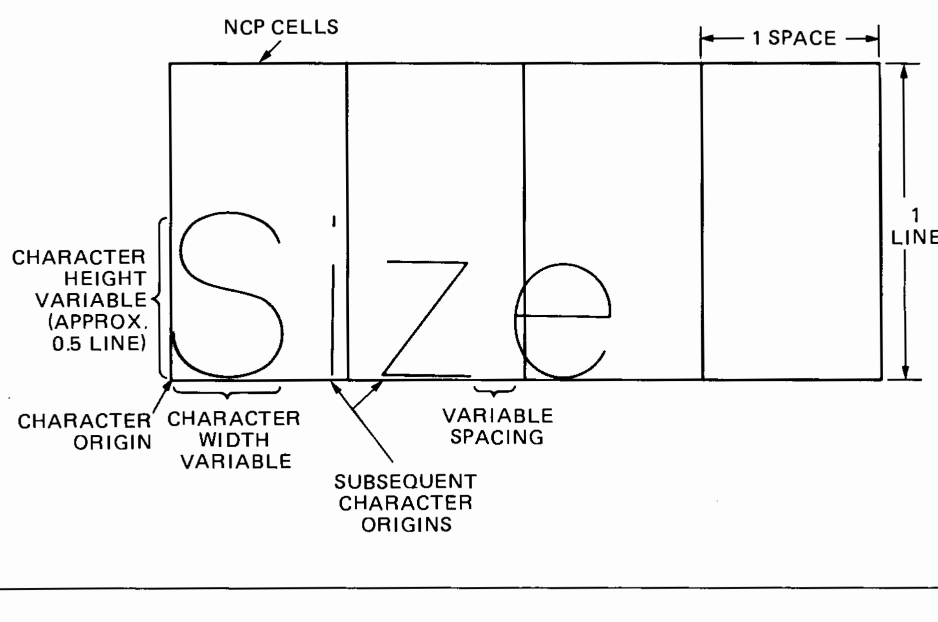
In the illustration which follows, the smoothness of the variable-space arc font set is a result of the default character chord angle of 5 degrees. The smoothness varies depending on the character chord angle. The smoothness of the fixed-space vector font set never changes.

There are several ways to position a label. The complete label string can be easily positioned using the label origin instruction, LO. This instruction automatically positions the label in relation to the current pen position and is sufficient for most applications. The BL and OL instructions (buffer label and output label length), taken together, return the space required for any given label. This information can then be considered when positioning the pen at the label starting point.

Fixed-Space Font Character Spacing



Variable-Space Arc Font Character Spacing



The Extra Space Instruction, ES

USES: The ES instruction adjusts the spacing between characters and lines without affecting character size.

SYNTAX: *ES* spaces (, lines) *term*
or
ES *term*

PARAMETERS:

Parameter	Format	Range	Default
spaces	decimal	$\pm(2^{26}-1)$	0
lines	decimal	$\pm(2^{26}-1)$	0

EXPLANATION: The space parameter and optional line parameter are in terms of the width and height of the NCP cell. Each represents the amount added or subtracted to the normal spacing. Positive numbers cause the label characters to be drawn farther apart; negative numbers cause the label characters to be drawn closer together, or even overlapping.

An ES instruction with no parameters (ES ;) is equivalent to ES 0 , 0 ; and will default spacing between characters and lines to the dimensions of the NCP cell, as set by the most recent SI or SR instruction. Refer to the previous section, Character Spacing.

The following illustration shows fixed-space font and variable-space font characters with increased and decreased spacing.

Character Set 0	Character Set 10
ES; CAUSES THIS SPACING.	ES; CAUSES THIS SPACING.
ES-.1,-.25; CAUSES THIS SPACING.	ES-.1,-.25; CAUSES THIS SPACING.
ES.2,.25; CAUSES THIS SPACING.	ES.2,.25; CAUSES THIS SPACING.

Labeling With Variables

In some applications, it is desirable to label the plot using variables rather than literals to define the label string. Many different conventions are used in different computer languages and computers to define variable length and the character field format in which these variables will be printed. To avoid unexpected placement of the labels defined by variables, refer to your computer manual for a definition of the conventions used to define the output character field.

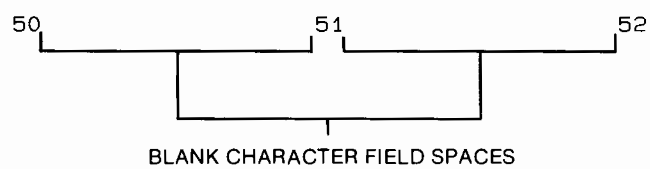
Labeling Plots

Quotation marks are used by many computers to define the literal characters that are to be sent, but variables are not included within quotation marks. The comma is used by some computers as a delimiter between variables to cause the label string to be right-justified in a specific character field width. The unused character positions in this field are normally sent as leading blank spaces to establish fixed spacing between label strings. For close spacing of the label strings, the blank spaces can normally be suppressed by substituting a semicolon as a delimiter between variables.

The following example illustrates use of the comma to establish fixed spacing when using variables for labeling. When the value of X is 50, the labels shown are produced by the given HP-GL instructions. The second statement causes the plotter to label the value of X, X + 1, and X + 2. Blank spaces between the printed integers normally include space for the sign which may not be printed depending on your computer. The number of blank character-field spaces may vary with different computers.

Send:

```
■LINE 1. X=50
 LINE 2. "LB",X,X+1,X+2,"␃"
```



The following example illustrates the closer spacing achieved in BASIC when semicolons separate variables in labeling instructions. The semicolons between the variables cause suppression of blank spaces. The space between the printed integers varies with different computers, but normally includes the sign space.

Send:

```
■LINE 1. X=50
 LINE 2. "LB";X;X+1;X+2;"␃"
```

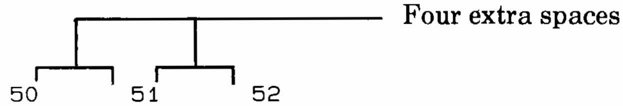
50 51 52

Any spaces required to fit into the context of the item being labeled must normally be sent enclosed in quotes. The following example labels the same variables as above, but with four extra spaces between each of the integers. Note that four spaces enclosed in quotes are sent between each variable, but the semicolon suppresses unwanted blank spaces.

■ BASIC statement. Do not send to the plotter.

Send:

```
■LINE 1. X=50
 LINE 2. "LB";X;"    ";X+1;"    ";X+2;"␃"
```



The Absolute Direction Instruction, DI

USES: The DI instruction specifies the direction in which characters are drawn. Use the instruction to change the direction of labeling to a new absolute direction (absolute meaning independent of P1,P2 settings). It is especially useful for labeling survey boundaries and machine drawing dimensions.

SYNTAX: *DI* run, rise *term*
or
DI *term*

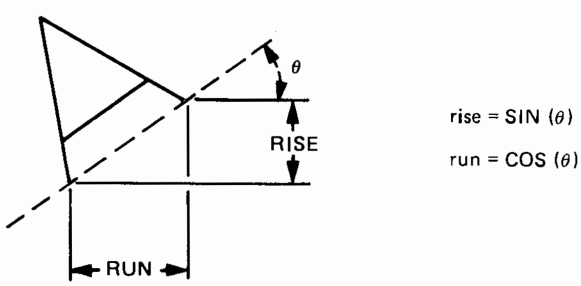
PARAMETERS:

Parameter	Format	Range	Default
run	decimal	$\pm(2^{26}-1)$	1
rise	decimal	$\pm(2^{26}-1)$	0

EXPLANATION: Run and rise specify the direction according to the relationship:

$$\theta = \tan^{-1}\left(\frac{\text{rise}}{\text{run}}\right)$$

where:



At least one parameter must be effectively nonzero (|parameter| ⩾ 0.0000000001).

A DI instruction with no parameters (DI ;) will default to the values DI 1 , 0 (horizontal).

■ BASIC statement. Do not send to the plotter.

A DI instruction with a rise parameter of zero will produce horizontal labeling.

A DI instruction with a run parameter of zero will produce vertical labeling.

A change in the orientation of P1 and P2 will not affect the direction of labeling.

A DI instruction remains in effect until another DI or DR instruction is executed or the plotter is initialized or set to default conditions.

A DI instruction updates the carriage return point to the current pen position.

The following table summarizes the error conditions for DI instruction.

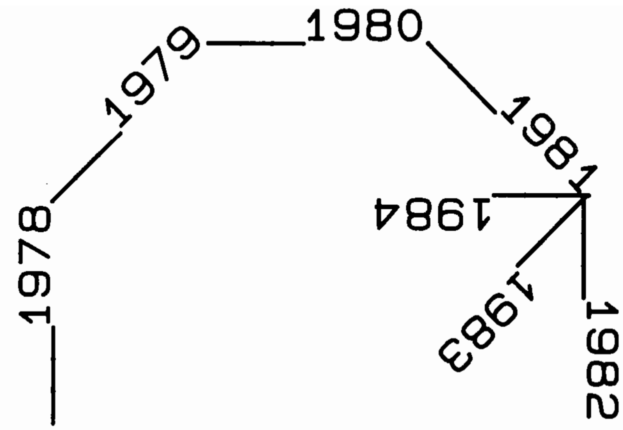
Condition	Error	Plotter Response
both parameters are zero	3	ignores instruction
more than two parameters	2	uses first two parameters
only one parameter	2	ignores instruction

When the angle, θ, necessary to establish the desired label direction is known, the instruction DI cos θ, sin θ can be used to establish label direction.

The following example labels the years, 1978 through 1984, in a circular pattern starting with vertical labeling. The direction in which each year is labeled is changed by 45 degrees. The last three years include a carriage return prior to terminating the label mode. Note that the carriage return point is the same for each of the last three years and is established by the DI instruction. Refer to The Label Instruction, LB, in this chapter.

Send:

```
LINE 1.   IN;SP1;IP10000,-10000,15000,-5000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA1050,1450;
LINE 4.   DI0,1;LB___1978␃
LINE 5.   DI1,1;LB___1979␃
LINE 6.   DI1,0;LB___1980␃
LINE 7.   DI1,-1;LB___1981␃
LINE 8.   DI0,-1;LB___1982␍␃
LINE 9.   DI-1,-1;LB___1983␍␃
LINE 10.  DI-1,0;LB___1984␍␃
LINE 11.  NR;
```



The Relative Direction Instruction, DR

USES: The DR instruction specifies the direction in which characters are to be drawn relative to the scaling points P1 and P2. Use the instruction to change the direction of lettering from its default direction, horizontal, to a direction which is relative to P1,P2 settings. It is useful when creating drawings which will be plotted in several sizes and you want labels to have the same relationship to the data on all plots.

SYNTAX: *DR* run, rise *term*
or
DR *term*

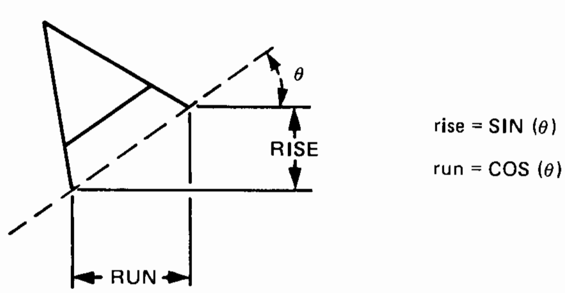
PARAMETERS:

Parameter	Format	Range	Default
run	decimal	$\pm(2^{26}-1)$	1
rise	decimal	$\pm(2^{26}-1)$	0

EXPLANATION: Run and rise specify the label direction according to the relationship:

$$\theta = \tan^{-1}\left(\frac{\text{rise}}{\text{run}}\right)$$

where:

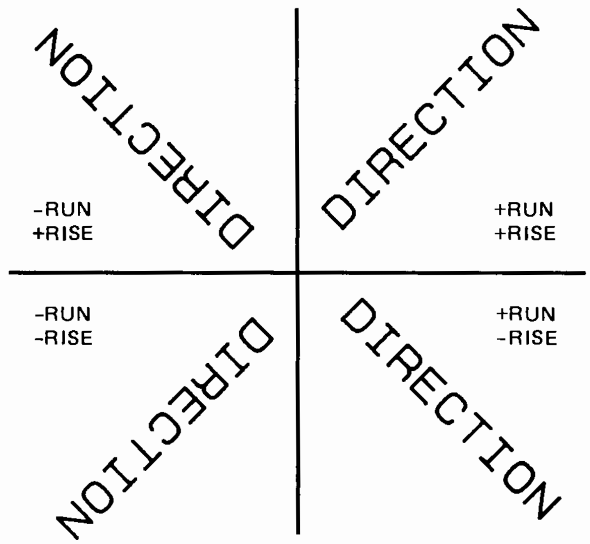


Run and rise specify a percentage of the absolute distance between P1 and P2 or a percentage of the algebraic distance between P1 and P2. The interpretation of the distance between P1 and P2 is determined by the setting of the **EMULATE/NORMAL** switch as follows:

EMULATE — Run = percentage of $|P2_X - P1_X|$
Rise = percentage of $|P2_Y - P1_Y|$

NORMAL — Run = percentage of $(P2_X - P1_X)$
Rise = percentage of $(P2_Y - P1_Y)$

If you imagine the current pen position to be the origin, the sign of the parameters determines in which quadrant the lettering will be. In the following example, rise and run assume all combinations of ±1 with default P1 and P2.



A change in P1 or P2 will affect the direction of the lettering. Refer to the section Parameter Interaction in Labeling Instructions in this chapter.

A DR instruction remains in effect until another DR or DI instruction is executed or the plotter is initialized or set to default conditions.

A DR instruction with no parameters (DR ;) will default to the values DR 1 , 0 (horizontal).

The following table summarizes the error conditions for the DR instruction.

Condition	Error	Plotter Response
both parameters are zero	3	ignores instruction
more than two parameters	2	uses first two parameters
only one parameter	2	ignores instruction

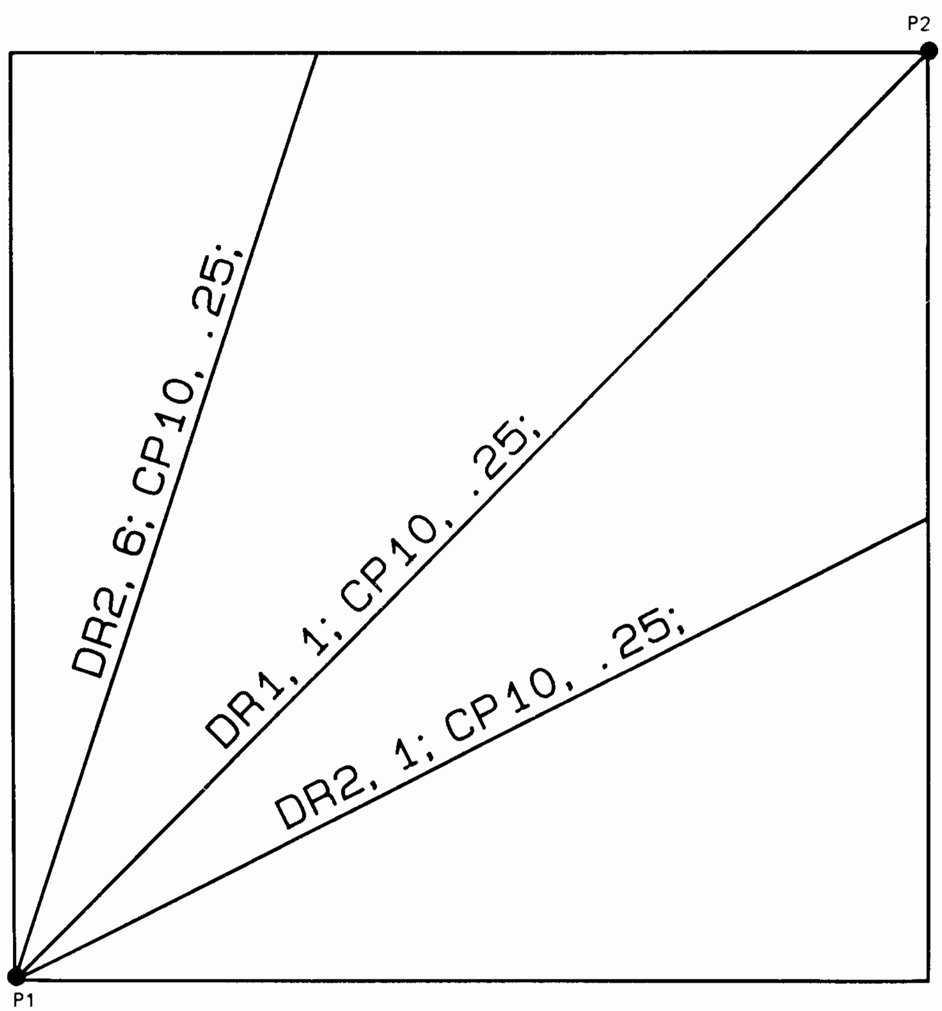
The following description may help you visualize the direction of labeling using the DR instruction with various parameters. Think of directional lines as being parallel to a line starting at the lower-left scaling point (usually P1) and intersecting the opposite side or the top edge of the plotting area established by the current P1 and P2.

Form a fraction in lowest terms ≤ 1 using the run and rise parameters. If run = rise, the fraction will equal 1 and the directional line will go from P1 to P2. If run > rise, the directional line will intersect the side of the plotting area that fraction of the way up toward P2. If rise > run, the directional line will intersect the top of the plotting area that fraction of the way across toward P2. Remember, since lettering starts at the current pen position, labels will be parallel to these lines, not necessarily along them.

The accompanying program illustrates the DR instruction with a given P1 and P2.

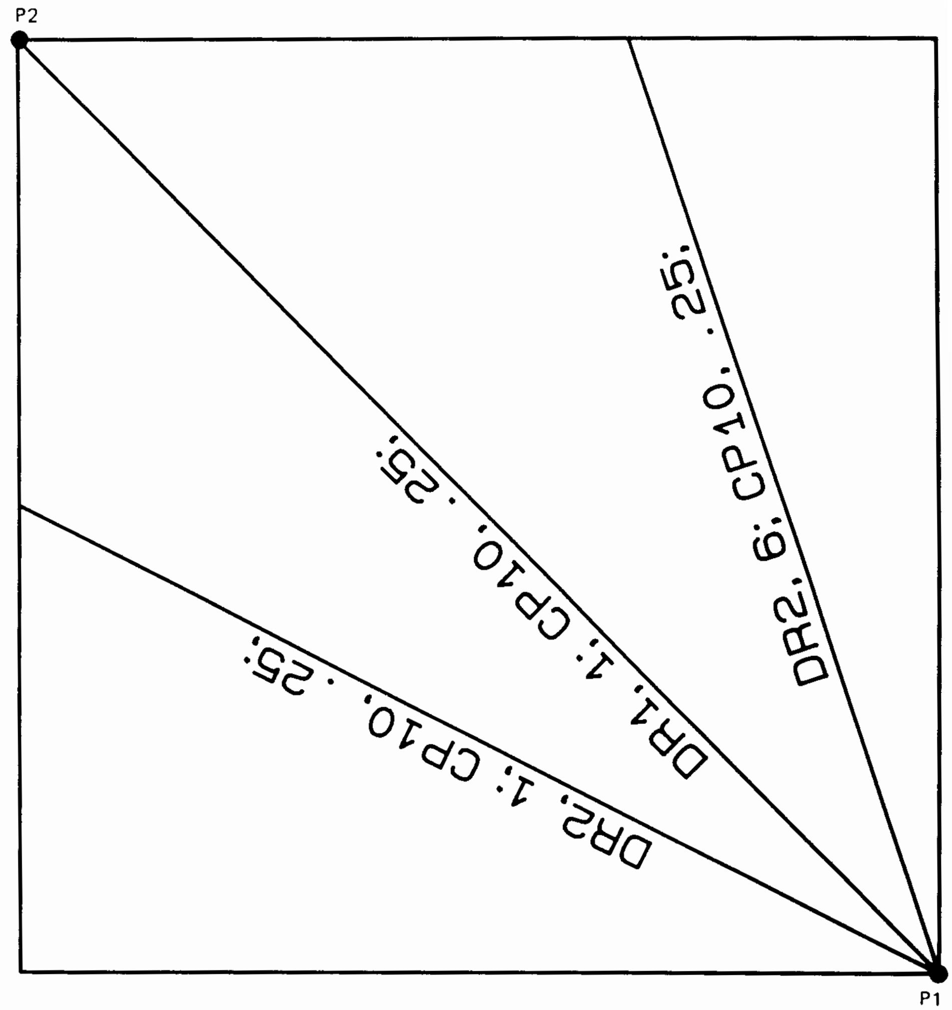
Send:

```
LINE 1.   IN;SP1;IP-15000,5000,-10000,10000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA0,0;PRPD0,5000,5000,0,0,-5000,-5000,0;
LINE 4.   PA5000,5000,PU,0,0;DR1,1;CP10,.25;LBDR1,1;
          CP10,.25;␍
LINE 5.   PA0,0,PD1667,5000,PU;
LINE 6.   DR2,6;PA0,0;CP10,.25;LBDR2,6;CP10,.25;␍
LINE 7.   PA0,0,PD,5000,2500,PU;
LINE 8.   DR2,1;PA0,0;CP10,.25;LBDR2,1;CP10,.25;␍
```



Changing the position of P1 and P2 will cause the plot and the label's direction to change as shown below.

```
LINE 1.  IN;SP1;IP-10000,5000,-15000, 10000;
```



If the programs are run in **EMULATE** mode, the lettering will be too small to be legible, and program 2 will produce the same output as program 1. This is because DR uses the $|P2_X - P1_X|$ when in **EMULATE** mode and the IN instruction causes an SR instruction without parameters to be executed, hence small lettering with the IP parameters used here.

The Character Plot Instruction, CP

USES: The CP instruction moves the pen the specified number of CP cells. Use the instruction to move the pen any number of character spaces or lines from a point on the plotting surface. Also use it to align with a left-hand margin, or to center or right-justify a label. Thus, you can move the label slightly above or below a line, insert spaces or lines in text, or center labels.

SYNTAX: *CP* spaces, lines *term*
or
CP *term*

PARAMETERS:

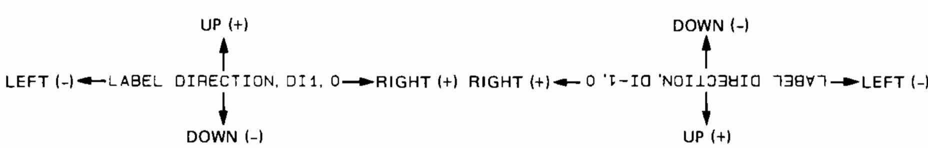
Parameter	Format	Range	Default
spaces	decimal	$\pm(2^{26}-1)$	none
lines	decimal	$\pm(2^{26}-1)$	none

EXPLANATION: The parameters are interpreted as described in the following paragraphs and illustrations.

1. Spaces. The space parameter is in terms of the width of one CP cell. Positive values specify the number of spaces the pen will move to the right; negative values specify the number of spaces the pen will move to the left.

2. Lines. The line parameter is in terms of the height of one CP cell. Positive values specify the number of lines the pen will move up; negative values specify the number of lines the pen will move down.

Note that right, left, up, and down are relative to the label direction as shown below:



A CP instruction without parameters (CP ;) performs a carriage return and line feed operation, moving one line down and returning to the carriage return point. Refer to The Label Instruction, LB, in this chapter.

A CP instruction which has a nonzero lines parameter shifts the carriage return point up or down by the amount specified.

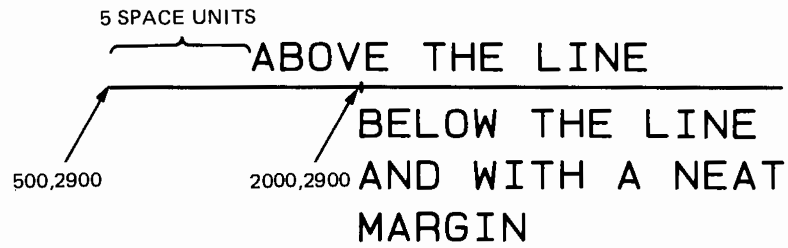
CP instructions are executed with the pen in its instructed state (up or down) and all moves are with respect to the current character origin. If an LO instruction is in effect, a pen-up move is made to the label origin before the CP instruction is executed. Following execution of the CP instruction, a pen-up move is made to regain the Y-axis corresponding to the carriage return point.

The instruction CP ; is useful to produce lettering with alignment along a left-hand margin, and a CP instruction with small decimal parameters enables lettering along a line but not on top of it. This is illustrated in the following example. Refer to The Label Origin Instruction, LO, later in this chapter for another method of positioning labels.

Send:

```
LINE 1.   IN;SP1;IP-10000,5000,-5000,10000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA500,2900,PD;PR4000,0,PU,PR-4000,0;
LINE 4.   CP5,.35;LBABOVE THE LINE␃
LINE 5.   PA2000,2900;XT;CP0,-.95;LBBELOW THE LINE␃
LINE 6.   CP;LBAND WITH A NEAT␃CP;LBMARGIN␃
```

The CP instruction in line 4 moves the label 5 space units (CP cells) to the right and slightly above the line. The CP instruction in line 5 moves the label slightly below the line and the two CP instructions in line 6 cause a carriage return to the left margin established by the plot instruction and line feeds to move each label down one line.



The Absolute Character Size Instruction, SI

USES: The SI instruction specifies the size of characters and symbols in centimetres. Use the instruction to establish absolute character sizing in centimetres so that size is not dependent on the settings of P1 and P2.

SYNTAX: *SI* width, height *term*
or
SI *term*

PARAMETERS:

Parameter	Format	Range		Default
		EMULATE	NORMAL	
width	decimal	>0 to $2^{26}-1$	$\pm(2^{26}-1)$*	0.285
height	decimal	>0 to $2^{26}-1$	$\pm(2^{26}-1)$*	0.375

*excluding zero and values approaching zero

EXPLANATION: The range of the width and height parameters varies according to the setting of the **EMULATE/NORMAL** switch indicated in the table above.

An SI instruction without parameters (SI ;) causes the plotter to use default values.

Character size set by an SI instruction is not affected by changes in P1 or P2. An SI instruction remains in effect until another SI or SR instruction is executed or the plotter is initialized or set to default conditions.

The following table summarizes the error conditions for SI instruction.

Condition	Error	Plotter Response
either parameter is zero	3	ignores instruction
more than two parameters	2	uses first two parameters
negative parameter in **EMULATE** setting	3	ignores instruction
only one parameter	2	ignores instruction

When the **EMULATE/NORMAL** switch is set to **NORMAL,** negative SI parameters will produce mirror images of labels as shown below.

INSTRUCTION	RESULTING LABEL
`SI.55,.75;LB HP␃`	

A negative SI width parameter will mirror labels in the right-to-left direction.

INSTRUCTION	RESULTING LABEL
`SI-.55,.75;LB HP␃`	

A negative SI height parameter will mirror labels in the top-to-bottom direction.

INSTRUCTION	RESULTING LABEL
`SI.55,-.75;LB HP␃`	

Two negative SI parameters will mirror the label in both directions and it will appear to be rotated 180 degrees.

INSTRUCTION	RESULTING LABEL
`SI-.55,-.75;LB HP␃`	

When the **EMULATE/NORMAL** switch is set to **EMULATE,** negative SI parameters are not allowed, so mirror imaging does not occur. For further information on the effects of negative parameters, refer to the section Parameter Interaction in Labeling Instructions in this chapter.

The Relative Character Size Instruction, SR

USES: The SR instruction specifies the size of characters and symbols as a percentage of the distance between scaling points P1 and P2. Use the instruction to define character size relatively so that if the P1, P2 distance changes, the character size will adjust to occupy the same "relative" amount of space.

SYNTAX: *SR* width, height *term*
or
SR *term*

PARAMETERS:

Parameter	Format	Range		Default
		EMULATE	NORMAL	
width	decimal	>0 to $2^{26} - 1$	$\pm(2^{26} - 1)$*	0.285
height	decimal	>0 to $2^{26} - 1$	$\pm(2^{26} - 1)$*	0.375

*excluding zero and values approaching zero

EXPLANATION: The range of the width and height parameters varies according to the setting of the **EMULATE/NORMAL** switch indicated in the table above.

The parameters specify width and height as a percentage of

- the absolute distance between the X- and Y-coordinates of P1 and P2 in the **EMULATE** mode, or
- the algebraic distance between the X- and Y-coordinates of P1 and P2 in the **NORMAL** mode.

EMULATE — Character width = $(\text{width}/100 \times |P2_X - P1_X|$
Character height = $(\text{height}/100 \times |P2_Y - P1_Y|$

NORMAL — Character width = $(\text{width}/100 \times (P2_X - P1_X)$
Character height = $(\text{height}/100 \times (P2_Y - P1_Y)$

An SR instruction without parameters (SR ;) causes the plotter to use default values. These are the same default values as for the SI instruction. This size will rubber with future changes in the location of P1 and P2.

An SR instruction remains in effect until another SI or SR instruction is executed or the plotter is initialized or set to default conditions.

The following table summarizes the error conditions for SR instruction.

Condition	Error	Plotter Respnse
either parameter is zero	3	ignores instruction
more than two parameters	2	uses first two parameters
negative parameter in **EMULATE** setting	3	ignores instruction
only one parameter	2	ignores instruction

The following example demonstrates that changing the locations of P1 and P2 changes character size set by an SR instruction. Lines 1 and 2 scale a 5000 plotter unit square into 5000 user units. Line 3 defines the relative character size and labels the characters 7580A. Line 4 changes the location of P1 and P2 to form a square with sides of 2500 plotter units. Since the width and height of the square are reduced by half, the characters labeled by the PB instruction in line 5 are halved accordingly.

Send:

```
LINE 1.  IN;SP1;IP0,5000,5000,10000;
LINE 2.  SC0,5000,0,5000;PA125,3000;
LINE 3.  SR13.5,18;LB7580␃
LINE 4.  IP0,5000,2500,7500;PA125,2350;
LINE 5.  PB;
```

7580

7580

When the plotter is in NORMAL mode, changing either the relative locations of P1 and P2 or using negative parameters will cause mirroring of labels. Refer to The Absolute Character Size Instruction, SI, and Parameter Interaction in Labeling Instructions for more information on mirroring.

Parameter Interaction in Labeling Instructions

There are four factors which interact and affect the direction and mirroring of labels; the label direction as specified by DI or DR instructions, the sign of the parameters for the size instructions SI or SR, the relative positions of P1 and P2, and the setting of the EMULATE/NORMAL switch. Especially when the plotter is in NORMAL mode, these interactions are complex. This section considers the four possible combinations of DI, DR, SI, and SR and illustrates the effects of various parameters, settings of P1 and P2, and the position of the EMULATE/NORMAL switch on labels. The labels used in the illustrations are the instructions which cause the direction, size, and mirroring of that label. All descriptions are in terms of the standard X,Y coordinate system shown in Chapter 2 under The Plotter Coordinate System. An arrow is shown for each label; this arrow is the baseline along which labeling occurs and shows the left-to-right direction that is the standard direction of a label without mirroring.

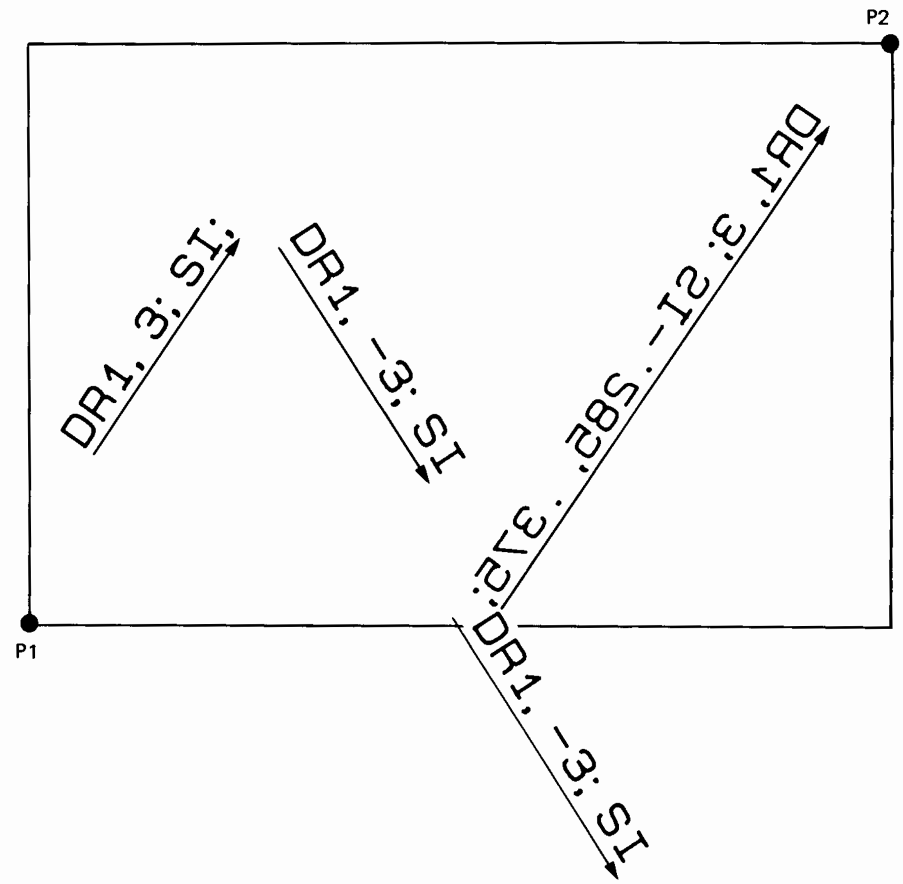
Use of DI and SI

When DI and SI instructions are used together, the DI instruction establishes the label's direction. When the plotter is in NORMAL mode, this line serves as the axis along and about which labels (written with negative SI parameters) are mirrored.

When the plotter is in EMULATE mode, an SI instruction may not contain negative parameters. Labels cannot be mirrored. Label direction can only change when a new direction instruction is executed. Refer to the sections The Absolute Direction Instruction, DI, and The Absolute Size Instruction, SI, for a description of parameters.

Use of DR and SI

When DR and SI instructions are used together, the label size is determined by the SI instruction and does not change with changes in the settings of P1 and P2. However, when the plotter is in **NORMAL** mode, negative parameters of an SI instruction will cause mirroring of labels along or about the baseline. This is shown in the following illustration. Any points P1 and P2 which establish a rectangle twice as wide as it is high will produce this label direction with the parameters given. Note the baselines are the same in each side of the figure. The left side of the illustration would be produced by the plotter in both **NORMAL** and **EMULATE** modes. The right side can only be drawn in **NORMAL** mode.

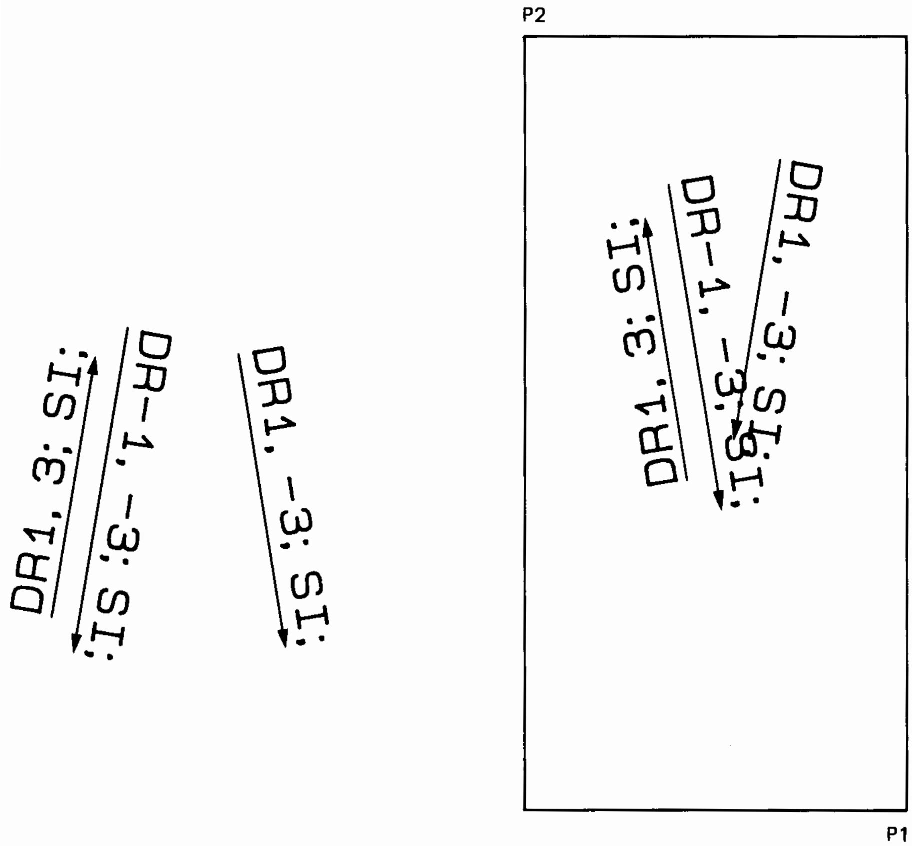


EITHER MODE NORMAL MODE ONLY

There are also differences in label direction caused by the setting of the **EMULATE/NORMAL** switch. When the switch is set to **NORMAL,** the run and rise parameters are multiplied by the algebraic differences ($P2_X - P1_X$) and ($P2_Y - P1_Y$), respectively. The signs of the parameters and the algebraic differences are important. The resulting parameters, when applied to the standard coordinate system, determine the label baseline.

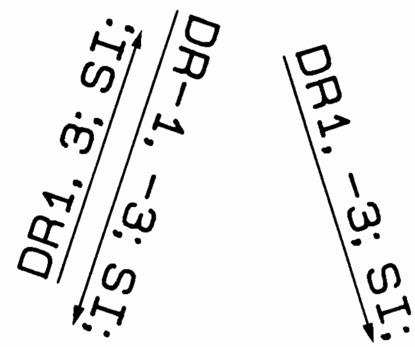
When the plotter is in **EMULATE** mode, the plotter uses only the absolute value of the differences $|P2_X - P1_X|$ and $|P2_Y - P1_Y|$. Changes in P1 and P2 which alter the distance between them will change the angle of the label but only the parameters of the DR instruction affect the quadrant in which the baseline falls.

Compare the output shown below when the switch is set in **NORMAL** and **EMULATE** mode. Any setting of P1 and P2 which establishes a rectangle twice as high as it is wide, and P1 is the lower-right corner and P2 the upper-left corner, will produce this output.



EMULATE MODE NORMAL MODE

When P1 and P2 define an area (a square in this illustration) in the normal orientation, i.e., P1 is the lower-left corner and P2 is the upper-right corner, the output is the same for both settings of the **EMULATE/NORMAL** switch.



EMULATE MODE
AND
NORMAL MODE

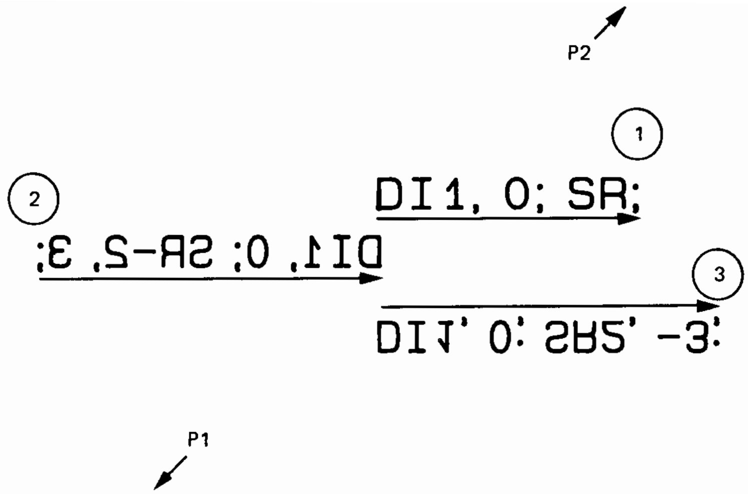
Use of DI and SR

When the DI instruction is used with SR, only the DI instruction affects the directional baseline of labels. This holds true in both **NORMAL** and **EMULATE** modes. Changes in the settings of P1 and P2 do not affect the direction of the baseline.

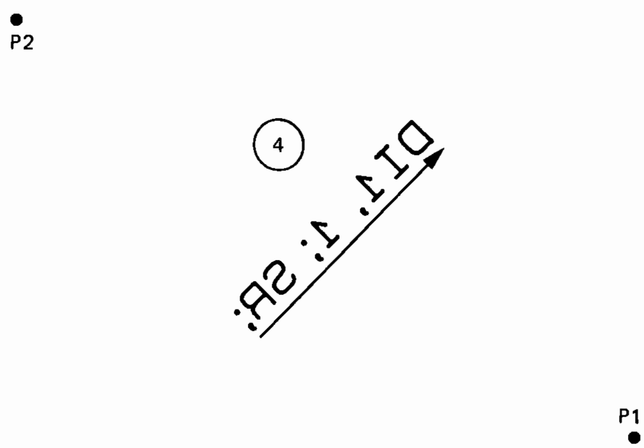
The SR instruction sets the size of the labels and mirroring, if any. When the plotter is in **EMULATE** mode, negative SI parameters are not allowed. Since the plotter uses the absolute value of the differences, $|P2_X - P1_X|$ and $|P2_Y - P1_Y|$, changes in the relationship of P1 to P2 do not affect the label orientation. No mirroring can be done. Refer to The Relative Size Instruction, SR, in this chapter for information on, and examples of how, the parameters and the setting of P1 and P2 affect the size of labels.

When the plotter is in **NORMAL** mode, either a negative SR width or height parameter with a positive difference $(P2_X - P1_X)$ or $(P2_Y - P1_Y)$, respectively, or a positive SR parameter with a negative difference will cause the label to be mirrored right to left or up and down, respectively. If respective parameters and differences are both positive or both negative, no mirroring will occur.

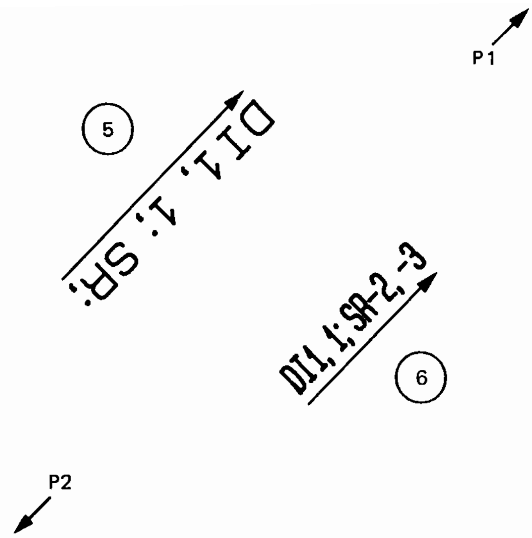
In the illustrations which follow, labels 1, 2, and 3 are written with the **EMULATE/NORMAL** switch set to **NORMAL** and P1 and P2 in standard orientation in a 5000-plotter-unit square. Label direction is horizontal, left to right. Label 2 has a negative width parameter so it is mirrored right to left. Label 3 has a negative height parameter and is mirrored top to bottom.



In label 4, P1 and P2 have been shifted so P1 is the lower-right corner and P2 the upper-left corner. Since SR ; is interpreted as positive parameters and ($P2_X - P1_X$) is negative, the label is mirrored right to left along the label baseline of 45 degrees established by the DI 1 , 1 instruction.



For labels 5 and 6, P1 and P2 are again corners of a 5000 plotter unit square but both ($P2_X - P1_X$) and ($P2_Y - P1_Y$) are negative. Label 5, with positive SR parameters, is mirrored both left to right and top to bottom. Label 6, with negative SR parameters and negative distances, is not mirrored.



Use of DR and SR

When these two instructions are used together, interactions are most complex. Using only standard settings of P1 and P2, where P1 is the lower-left corner and P2 is the upper-right corner, will make it easier to visualize the direction and mirroring of labels.

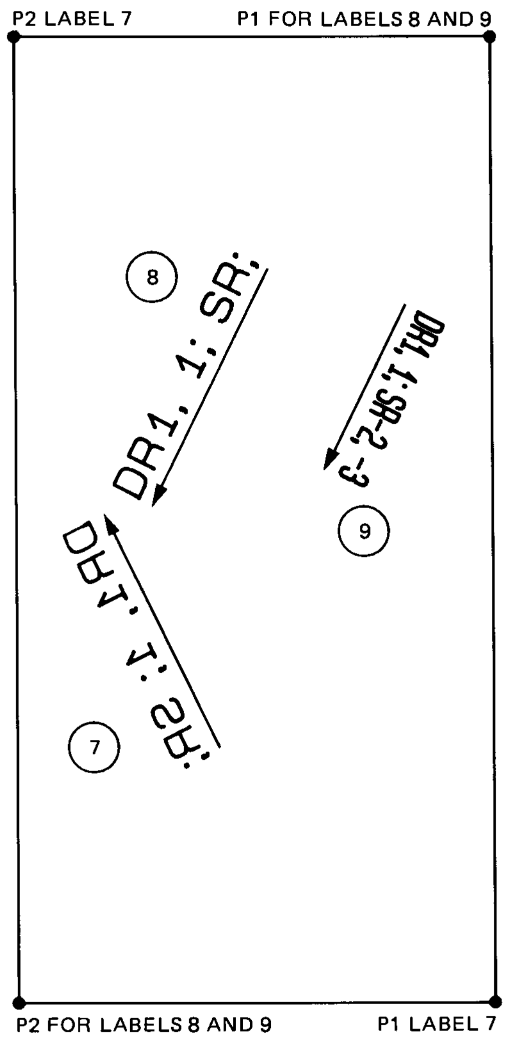
When the plotter is in EMULATE mode, size and direction both vary as the absolute X- and Y-distances between P1 and P2. Unless parameter signs are changed, the label direction will not change quadrants. Negative parameters are not allowed and changes in the relative positions of P1 and P2 will not cause mirroring.

When the switch is set to NORMAL, DR parameters interact with the algebraic differences ($P2_X - P1_X$) and ($P2_Y - P1_Y$) to establish label direction, and SR parameters interact with these differences to create mirroring. Signs of both the parameters and the distances are important. A negative sign in either the parameter or the distance will affect both DR and SR.

Having both parameter and distance either positive or negative will cause standard direction or no mirroring.

The following program prints labels 7, 8, and 9.

```
LINE 1.   IP12500,-5000,10000,0;
LINE 2.   DR1,1;SR;LBDR1,1;SR;␃␃␃
LINE 3.   IP12500,0,10000,-5000;
LINE 4.   DR1,1;SR;LBDR1,1;SR;␃␃␃␃
LINE 5.   DR1,1;SR-2,-3;LBDR1,1;SR-2,-3␃
```



In label 7, the label baseline is in the second quadrant, not the first, because $P2_X - P1_X$ is negative and the DR run parameter is positive. Likewise, the label is mirrored left to right because that distance is negative while the parameter is positive.

In labels 8 and 9, the label direction baseline is in the third quadrant because both $P2_X - P1_X$ and $P2_Y - P1_Y$ are negative. Label 8 is mirrored in both directions. (Rotate the manual so the arrow points to 45 degrees to see this more clearly.)

In label 9, the label is not mirrored because both parameters and distances are negative. (Again this may be easier to see if you rotate the manual.)

The Character Slant Instruction, SL

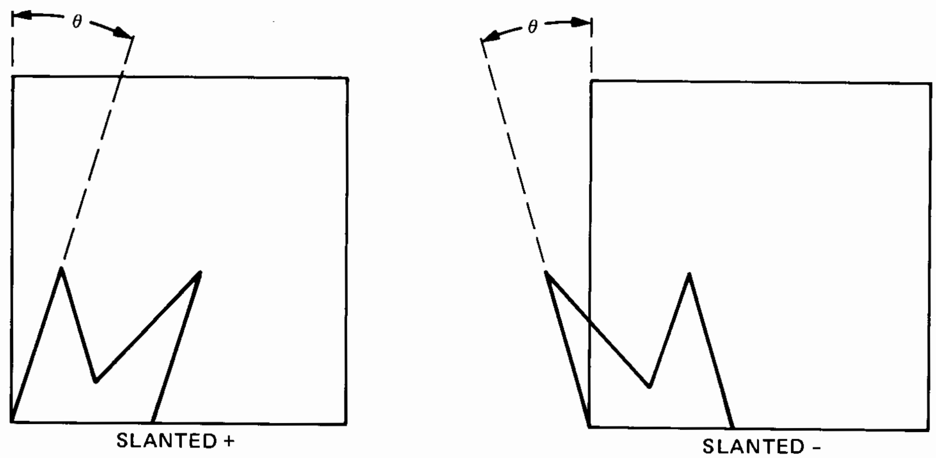
USES: The SL instruction specifies the slant with which characters are drawn. Use the instruction to create slanted text, particularly for emphasis, or to reestablish upright labeling after an SL instruction with parameters has been in effect.

SYNTAX: *SL* tan θ *term*
or
SL *term*

PARAMETERS: The parameter is in decimal format and is the tangent of the angle θ.

EXPLANATION: An SL instruction with no parameters (SL ;) will default to the same values as SL 0 (no slant).

The parameter is the tangent of the angle θ from vertical as follows:



The useful parameter range is ± 0.05 to ± 2 when using default absolute character size and up to ± 3.5 for large letters.

A change in scaling points P1 and P2 will not affect the angle θ.

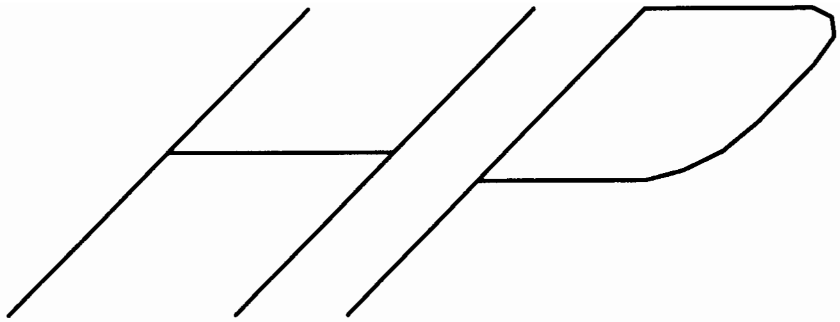
An SL instruction with more than one parameter will set error 2. However, the first parameter will be used to set the character slant.

An SL instruction remains in effect until another valid SL instruction is executed or the plotter is reinitialized or set to default conditions.

The following example letters HP at a slant of +45 degrees.

Send:

```
LINE 1.   IN;SP1;IP5000,5000,10000,10000;
LINE 2.   SC0,5000,0,5000;PA350,1750;
LINE 3.   SL1;SI3,4;LBHP␃
```



The Label Origin Instruction, LO

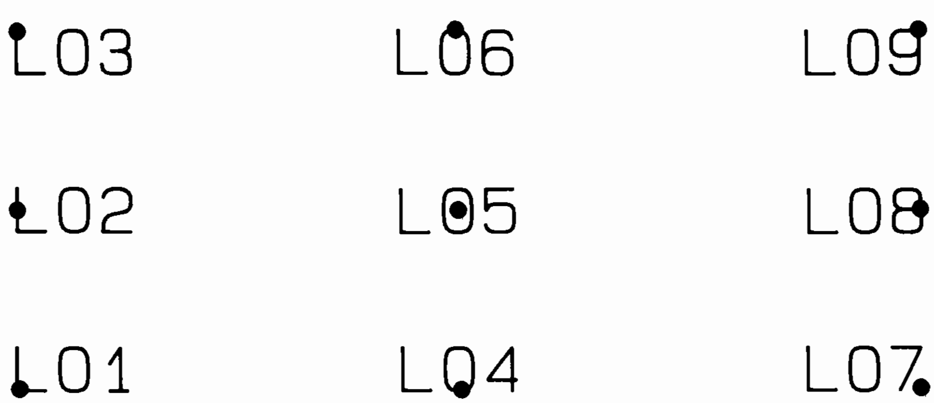
USES: The LO instruction positions labels relative to the current pen position. Use the instruction to center labels or justify them to the left or right of the current pen position. Positioning can be above or below the current pen position and can also be offset by an amount equal to 1/2 the character's width and height.

SYNTAX: *LO* position number *term*
or
LO *term*

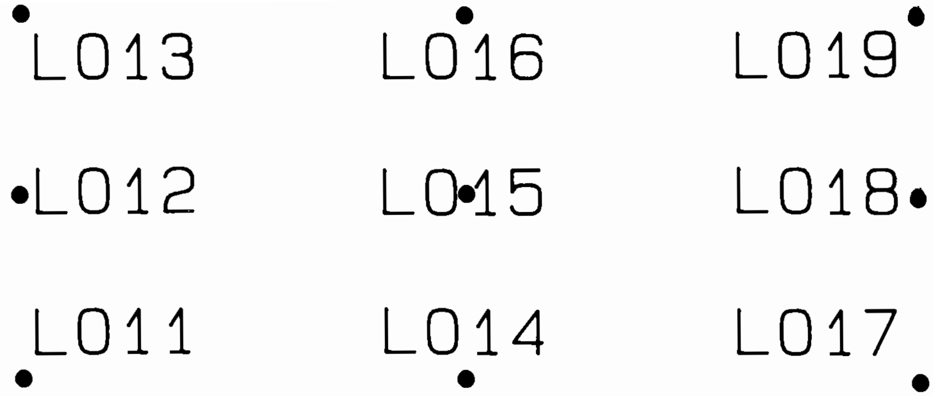
PARAMETERS:

Parameter	Format	Range	Default
position number	integer	1 to 9 or 11 to 19	1

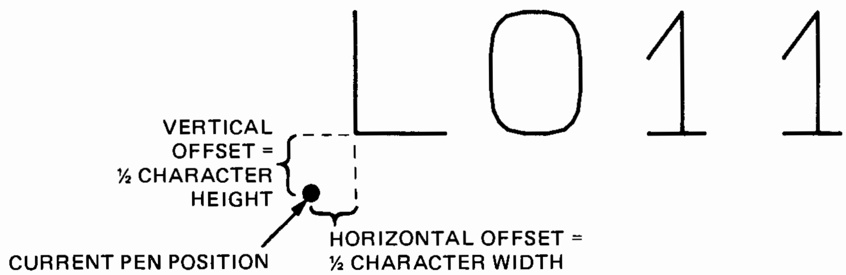
EXPLANATION: An LO instruction with no parameters (LO ;) sets the default label origin, LO 1. The position number determines the position of the label with respect to the current pen position. The illustration below graphically summarizes this relationship by showing the instructions LO 1 through LO 9 in the labeled position which they produce. Each dot represents the current pen position.



The positions resulting from the instructions LO 11 through LO 19 are the same as shown for the instructions LO 1 through LO 9, except that the labels are offset from the current pen position as shown on the next page.



The amount of offset is equal to 1/2 of the character's width and 1/2 of the character's height as specified by the most recent SI or SR instruction. The offset is shown below.



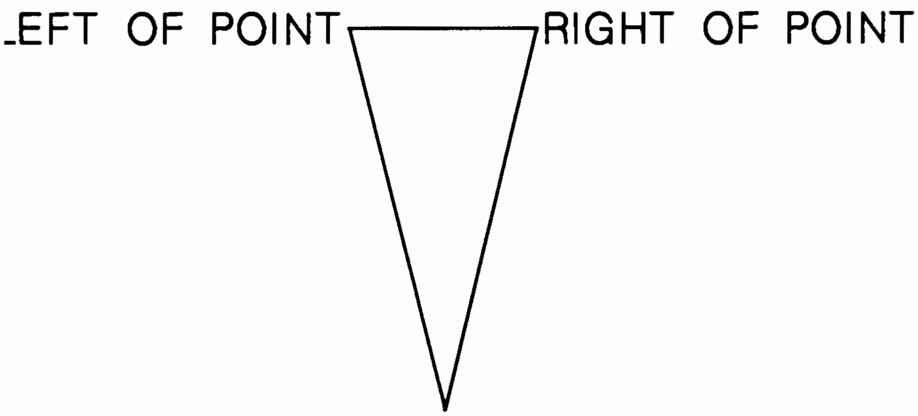
An LO instruction remains in effect until another LO instruction is received or the plotter is initialized or set to default conditions.

If position numbers 1 through 3, or 11 through 13, are specified, positioning calculations are performed on individual characters and the characters are drawn as they are received and stored in the label buffer. However, if position numbers 4 through 9, or 14 through 19, are specified, positioning calculations are not performed until the length of the label string is established by receipt of either a carriage return or the label terminator. In these cases, the characters are stored in the label buffer and the label string is not drawn until a carriage return or label terminator is received. The capacity of the label buffer limits line length for these centered or right-justified label applications to 150 characters, including control characters and the carriage return or label terminator.

The pen position is updated after each character is drawn, and the pen automatically moves to the next character origin in anticipation of additional characters. If the pen position is to be returned to its location prior to the label, a carriage return character must be sent following the label string and before the label terminator. The following example demonstrates the use of the carriage return with an LO instruction.

Send:

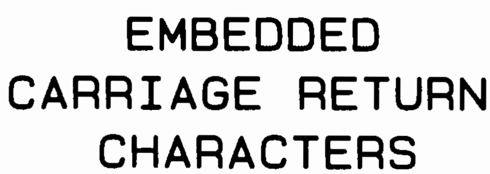
```
LINE 1.   IN;SP1;IP10000,5000,15000,10000;
LINE 2.   SC0,5000,0,5000;
LINE 3.   PA2000,1500;PRPD-500,2000;LO18;LBLEFT OF POINT␍␃
LINE 4.   PR1000,0;LO12;LBRIGHT OF POINT␍␃
LINE 5.   PR-500,-2000,PU;
```



When carriage return characters are embedded in an LB label string, each portion of the label preceding, between, and following a carriage return character is positioned according to the label origin. This is shown in the following illustration.

Send:

```
LINE 1.   IN;SP1;IP-15000,0,-10000,5000;
LINE 2.   SC0,5000,0,5000;PA2500,2600;LO14;
LINE 3.   LBEMBEDDED␍␊CARRIAGE RETURN␍␊CHARACTERS␃
```



The Buffered Label String Instruction, BL

USES: The BL instruction stores a label string in the label buffer. Use the instruction to determine the space requirements of a label string prior to drawing it. You can then draw the string any number of times at the desired locations.

SYNTAX: *BL* c . . . c terminator
where the terminator is as defined by the DT instruction
or
BL terminator

PARAMETERS: The parameters (c . . . c) can consist of up to 150 characters.

EXPLANATION: All characters following the BL instruction are stored in the label buffer. The capacity of the label buffer limits the length of the label string to 150 characters, including control characters and the label terminator. You can terminate the label string only by sending a label terminator at the end of the label string. Refer to The Define Terminator Instruction, DT, in this chapter.

The label buffer is cleared by executing a front-panel **RESET**, a DF or IN instruction, or an LB or BL instruction with only the label terminator.

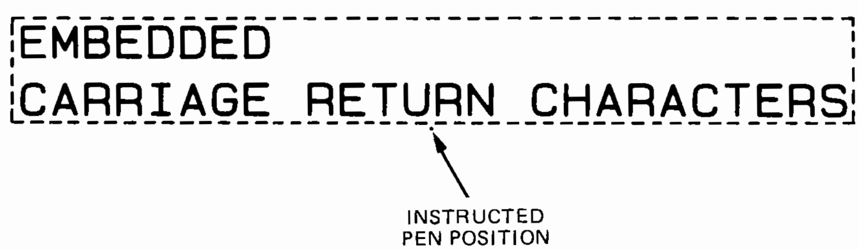
The BL instruction does not label. Once stored in the buffer, the string is labeled by executing a PB instruction. You can use the BL instruction in conjunction with the OL instruction to obtain information about the label string prior to drawing it. The buffered label string can then be drawn in the appropriate places, any number of times, using the PB instruction. The label is drawn in the currently selected character set in accordance with the ES, DI or DR, LO, SI or SR, and SL parameters which are in effect at the time of each subsequent PB instruction.

Executing another BL or LB instruction completely overwrites any previous contents of the label buffer.

Buffered labels, like labels drawn with the LB instruction, are subject to positioning and control characters. However, when embedded carriage return characters are used, an LO instruction will position a buffered label differently than a label drawn with an LB instruction. For a buffered label, each line is left-justified and the entire label is positioned (using the label origin in effect) as though it were enclosed in a rectangular box. The box is the width of the longest text line. The following example demonstrates use of the LO instruction to position a buffered label. Refer to the embedded carriage return characters example given under the LO instruction where each line of the label is centered.

Send:

```
LINE 1.   IN;SP1;IP-10000,0,-5000,5000;
LINE 2.   SC0,5000,0,5000;PA2500,2300;LO14;
LINE 3.   BLEMBEDDED␍␊CARRIAGE RETURN CHARACTERS␃
LINE 4.   PB;
```



The Output Label Length Instruction, OL

USES: The OL instruction outputs information on the contents of the label buffer. The instruction is normally used in conjunction with the BL instruction to determine the space requirements of the buffered label string prior to drawing it.

SYNTAX: *OL term*

PARAMETERS: No parameters are used. However, the terminator must be included to complete the instruction.

EXPLANATION: After an OL instruction is received, the plotter will output information concerning the buffered label as three numbers in ASCII, separated by commas and followed by a terminator. The output syntax is:

length, characters, line feeds [TERM]

The length field specifies the length of the longest line in the buffered label in terms of the space dimension of the CP cell. If a label is in the buffer, the length field is always output with four decimal places.

The characters field contains an integer, the number of printing characters and spaces in the longest line of the buffered label. A backspace counts as −1: a character with automatic backspace (e.g., ~ in set 1) counts as zero.

The line feeds field contains an integer, the net number of line feeds that will occur when the buffered label is drawn. An inverse line feed character (VT) counts as −1 and a line feed character (LF) counts as +1. If the buffered label contains the same number of VT and LF characters, then zero is returned.

The OL instruction is normally used in conjunction with the BL instruction but will always return information on the current contents of the buffer, whether the buffer was loaded by an LB or BL instruction. If no label is in the buffer, three zeros are returned.

The Print Buffered Label Instruction, PB

USES: The PB instruction prints the contents of the label buffer. Use the instruction to draw the contents of the label buffer any number of times. Positioning, spacing, direction, slant, and size may be changed each time the label string is drawn.

SYNTAX: *PB term*

PARAMETERS: No parameters are used. However, the terminator must be included to complete the instruction.

EXPLANATION: The contents of the label buffer are drawn using the currently selected character set in accordance with the ES, DI or DR, LO, SI or SR, and SL parameters in effect at the time of the PB instruction. The buffer contents remain after printing and can be repeated any number of times. You can change direction, slant, and size of labels between repeated printings of a label.

The label buffer is limited to 150 characters and contains the most recent string of label characters received in a BL or LB instruction. Executing a front-panel RESET, a DF or IN instruction, or an LB or BL instruction with only the label terminator will clear the label buffer.

An example which uses the PB instruction is found under The Buffered Label String Instruction, BL, earlier in this chapter.

The Character Chord Angle Instruction, CC

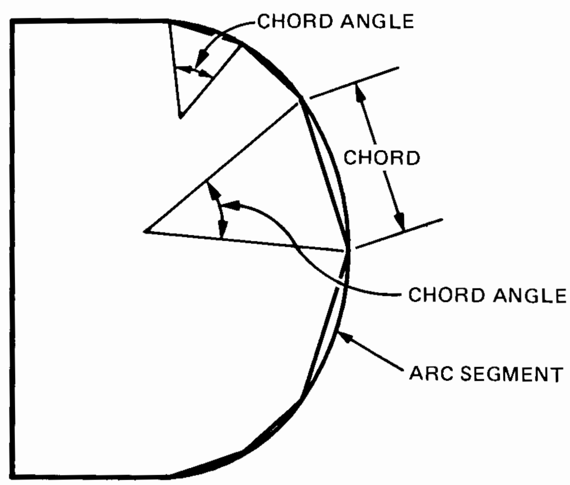
USES: The CC instruction sets the angle which determines the smoothness of characters drawn when you select one of the arc-font character sets. Use the instruction to change arc-font character smoothness from its default value. This allows using fewer chords to construct characters, and thus reduces the time required to draw rough or preliminary plots.

SYNTAX: *CC* chord angle *term*
or
CC term

PARAMETERS:

Parameter	Format	Range	Default
chord angle	decimal	$\pm(2^{26}-1)$	5 degrees

EXPLANATION: The character chord angle is the maximum angle subtended by the chord used to approximate any arc segment in a character from an arc-font character set. Characters in arc-font sets are composed of arc segments with various radii combined with straight lines. A single character may have several different radii. The following "D" was drawn with a chord angle of 45 degrees which is marked on two different arc segments.

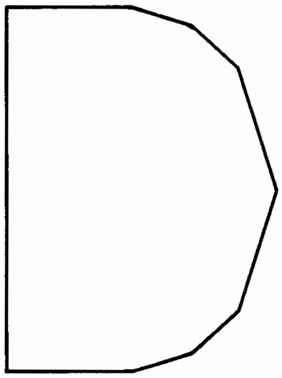


The chord angle is in decimal format, $-2^{26}+1$ to $2^{26}-1$. The sign of the parameter is ignored. A chord angle too large to result in satisfactory character quality is overridden by a ceiling chord angle of 45 degrees. A chord angle of zero is overridden by a very small chord angle. A CC instruction with no parameters (CC ;) will default to a chord angle of 5 degrees, the angle in effect at power-on and after a DF or IN instruction or front-panel **RESET** has been executed.

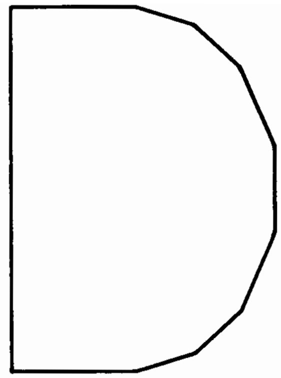
Increasing the chord angle decreases the number of chords used to construct the character. Decreasing the chord angle increases the number of chords and produces smoother arcs. However, this also increases the time required to draw the characters. The effect of different chord angles is shown in the following example.

Send:

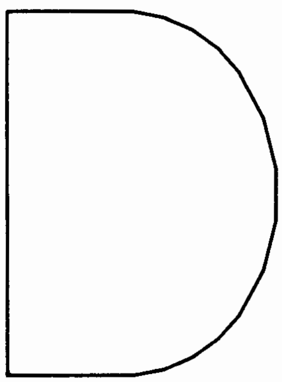
```
LINE 1.   IN;SP1;IP10000,0,15000,5000;
LINE 2.   SC-100,100,-100,100;CS10;SI3.6,4.8;
LINE 3.   LO4;PA-50,20;CC45;LBD␃;PA-50,10;
LINE 4.   PA50,20;CC30;LBD␃
LINE 5.   PA-50,-80;CC15;LBD␃
LINE 6.   PA50,-80;CC5;LBD␃
```



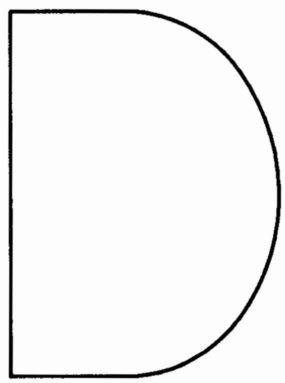
45 DEGREE CHORD ANGLE



30 DEGREE CHORD ANGLE



15 DEGREE CHORD ANGLE



5 DEGREE CHORD ANGLE

The User-Defined Character Instruction, UC

USES: The UC instruction enables you to draw characters of your own design. Use this instruction to create symbols not included in the plotter's character sets or to draw logos.

SYNTAX: *UC* (pen control,) X-increment, Y-increment (, . . .)
(,pen control)(, . . .) *term*
or
UC *term*

PARAMETERS: A detailed explanation of the parameters follows.

EXPLANATION: The characters are drawn in accordance with the instruction's parameters on the primitive grid that is superimposed on the NCP cell. The parameter ranges and the resolution of the primitive grid are determined by the setting of the **EMULATE/NORMAL** switch as described below:

	EMULATE
1. X,Y increments	— Integers between ±98 interpreted as primitive grid units.
2. Pen control parameters	— Integers ⩾+99 interpreted as pen down Integers ⩽−99 interpreted as pen up Integers > 32 767 or < −32 768 are not recognized.

EMULATE
(Continued)

3. Primitive grid resolution — 1 NCP space = 6 grid units
   1 NCP line = 16 grid units
   Since an NCP cell is 1-1/2 times character width and twice character height, a character drawn using UC should fit in a 4 × 8 grid when matching characters drawn in a given size with LB or BL and PB instructions.

NORMAL

1. X,Y increments — Integers between ±9998, interpreted as primitive grid units.
2. Pen control parameters — Integers +9999 to $2^{26} - 1$ interpreted as pen down.
   Integers −9999 to $-2^{26} + 1$ interpreted as pen up.
3. Primitive grid resolution
   Fixed-space font set selected — 1 NCP space = 48 grid units
   1 NCP line = 64 grid units
   Characters in a 32 × 32 grid will be same size as characters drawn with LB or BL and PB instructions.

   Variable-space font set selected — 1 NCP space = 42 grid units
   1 NCP line = 72 grid units
   Characters in a 28 × 36 grid will be same size as characters drawn with LB or BL and PB instructions.

A user-defined character is drawn in the following manner:

1. Each X,Y increment is drawn using the pen up/down status of the most recent pen control parameter. Upon entry into a UC instruction, the plotter sets the pen status up and the pen at the point 0,0 on the primitive grid. The first pen control parameter is optional unless the first move is to be with the pen down.

2. The pen moves to the point defined by each X,Y increment pair in order. The X-increment specifies the number of primitive grid units that the pen will move horizontally from the current pen position. A positive increment causes the pen to move to the right, and a negative increment cause it to move to the left.

   The Y-increment specifies the number of primitive grid units that the pen will move vertically from the current pen position. A positive increment moves the pen up and a negative increment moves the pen down. All references to the right, left, up, and down are relative to the current label direction. UC characters are mirrored in the same way as labeled characters.

3. The pen control parameter is internal to the UC instruction. Since this internal pen status is initially set to up by the plotter, nothing will be drawn by a UC instruction which does not have at least one pen down parameter. Once a pen down parameter is specified, the pen remains down for the following X,Y increment moves until a pen up parameter is specified or the UC instruction is completed. Upon termination of the UC instruction, the pen is raised and moves to the next character

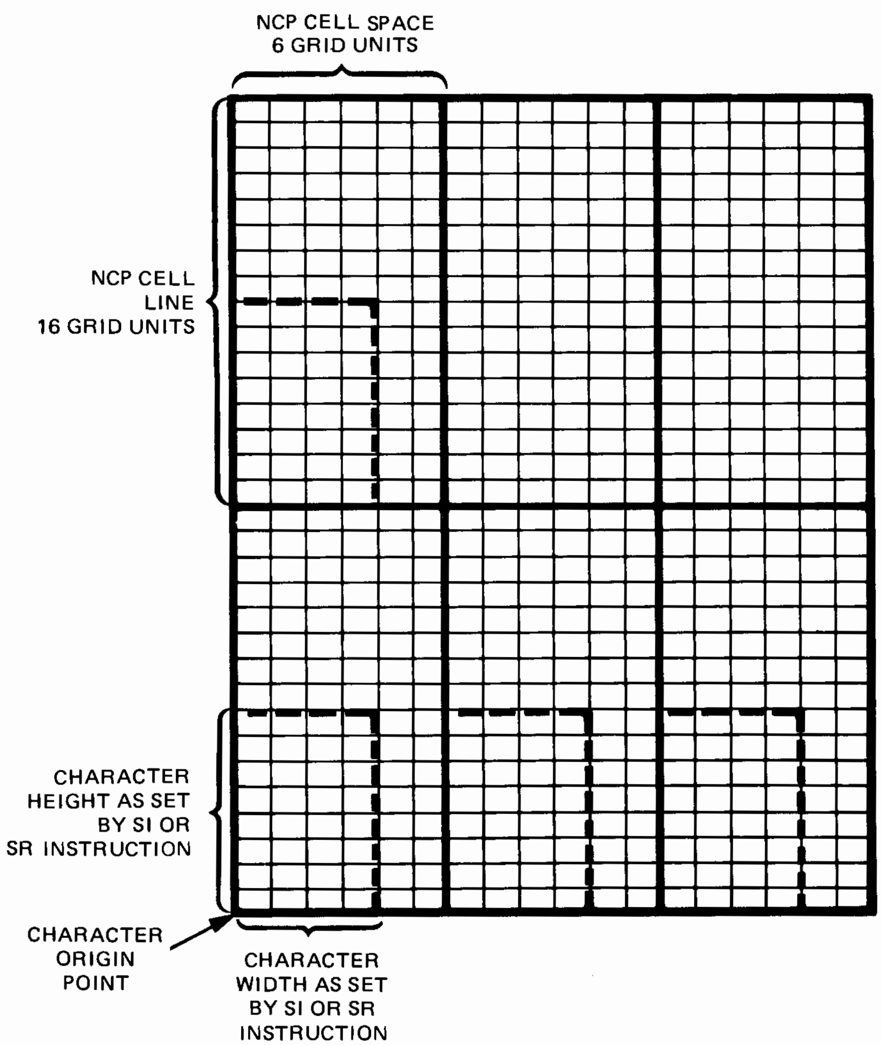
origin determined by the CP cell, i.e., including extra space(s) set by an ES instruction. The pen then assumes the status (up or down) of the most recent PU or PD instruction.

A UC instruction with no parameters (UC ;) causes the pen to return to the carriage return point.

User defined characters are drawn in accordance with the ES, DI or DR, SI or SR, and SL parameters which are in effect at the time the UC instruction is executed. The user defined character is also subject to the vertical positioning component of the current LO instruction, but not the horizontal positioning component.

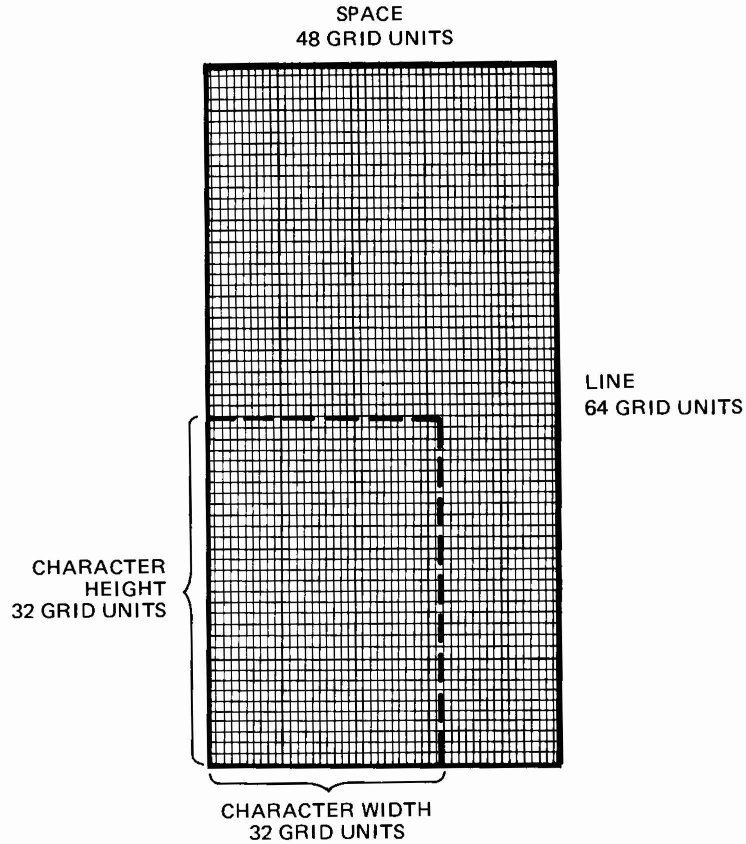
Since the primitive grids are different for **EMULATE** mode and for **NORMAL** mode in fixed- and variable-font, UC characters will be drawn in different sizes and with different proportions, depending on the plotter's mode and the character font selected. However, the use of the UC instruction is similar for each primitive grid resolution that is superimposed on the NCP cell. The three primitive grid resolutions are shown below. The size of the NCP cell is independent of the character font selected and the position of the **EMULATE/NORMAL** switch. Its size will always be 1-1/2 times the width and twice the height of an uppercase "A" as set by the most recent SI or SR instruction.

EMULATE Mode

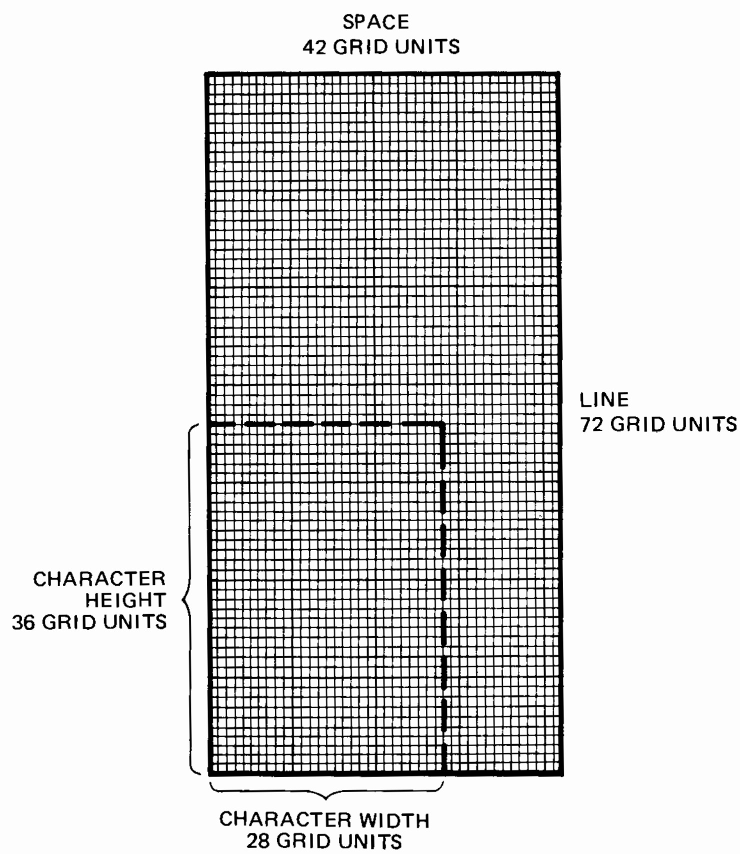


Labeling Plots

NCP Cell, NORMAL Mode, Fixed-Space Font Selected



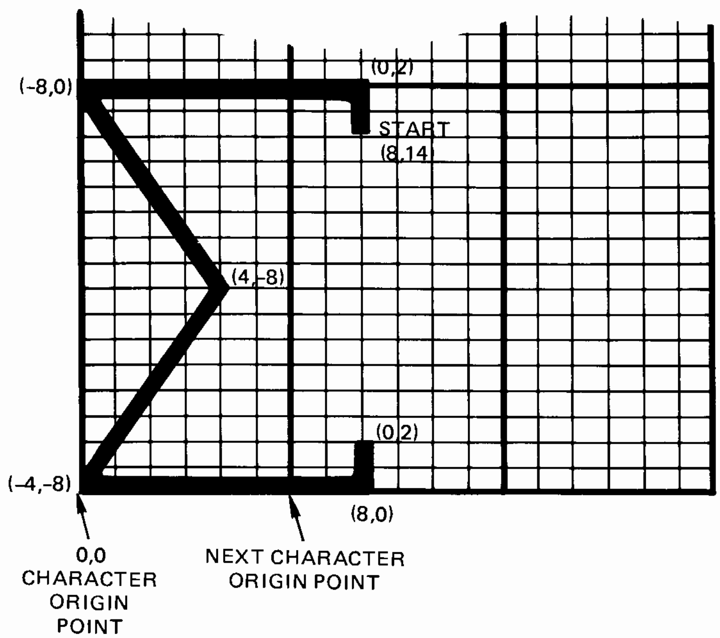
NCP Cell, NORMAL Mode, Variable-Space Font Selected



As shown in the next illustration, user-defined characters are not limited to a single NCP cell. If the user-defined character is larger than one NCP cell, a PA, PR, or CP instruction should be used to move the pen beyond the limits of the user-defined character. Otherwise, the next character will be superimposed on the character just drawn. The parameters of a CP instruction used for this purpose are measured from the pen position at the completion of the UC instruction (one CP space to the right of the user defined character's origin point).

The illustration below shows a Σ created by the following UC instruction when the plotter is in **EMULATE** mode. The numbers in parentheses are X,Y increments of the UC instruction.

```
UC8,14,99,0,2,-8,0,4,-8,-4,-8,8,0,0,2;
```



A Σ of the same size will be drawn by the following UC instructions when the plotter is in **NORMAL** mode and the indicated fonts are selected.

Fixed-Space Font (Character Sets 0-9, 20–39, 50–59)

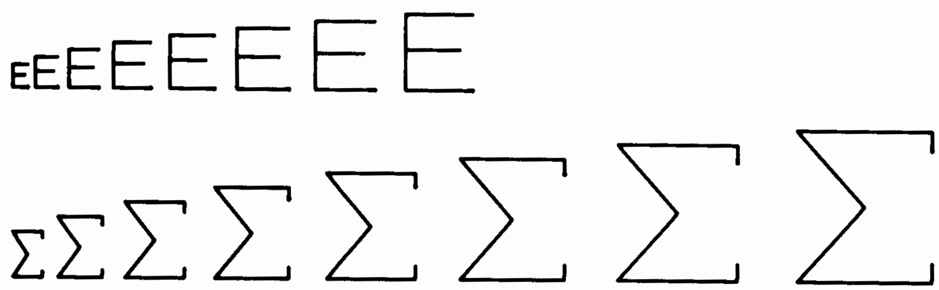
```
UC64,56,9999,0,8,-64,0,32,-32,-32,-32,64,0,0,8;
```

Variable-Space Font (Character Sets 10-19, 40–49)

```
UC56,63,9999,0,9,-56,0,28,-36,-28,-36,56,0,0,9;
```

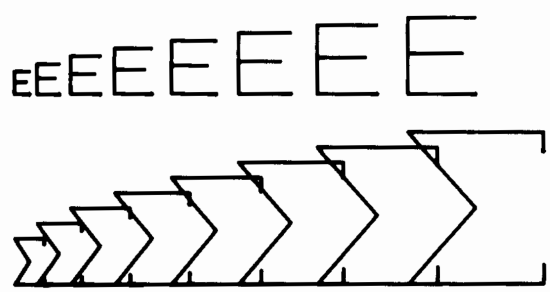
The following example generates a Σ symbol. The capital E is lettered by the LB instruction and is included to show that the SI instruction has the same effect on both user defined characters and normal characters. The CP instruction in line 6 creates spacing between Σ characters. It is only necessary when the user defined character does not fit in one NCP cell.

```
 LINE 1.  IN;SP1;IP-5000,0,0,5000;
 LINE 2.  SC0,5000,0,5000;PA100,2000;
■LINE 3.  FOR A=.2 TO .9 STEP .1
★LINE 4.  SI A,A*.1;
 LINE 5.  UC64,56,9999,0,8,-64,0,32,-32,-32,-32,64,0,0,8;
 LINE 6.  CP1,0%
■LINE 7.  NEXT A
 LINE 8.  PA100,3000;
■LINE 9.  FOR B=.2 TO .9 STEP .1
★LINE 10. SI B,B*.1;
 LINE 11. LBE%
■LINE 12. NEXT B
```



Now send the above program with line 6 deleted. Notice that when the CP instruction is omitted, the character origin points for each symbol and its corresponding E are at the same horizontal location. Now the Σ symbols are partially superimposed because Σ is designed in two NCP cells and is twice as large as a standard uppercase character, but the pen position on completion of a UC instruction is always only one NCP cell to the right of the character origin point.

Result:



The size of user-defined characters can be changed not only by the SI or SR instructions, but also by dividing or multiplying the X,Y increments of the UC instruction. Change line 5 of the previous example by dividing each of the X,Y increment values by two. Now each symbol is only half as large as before, and is the same size as the E character.

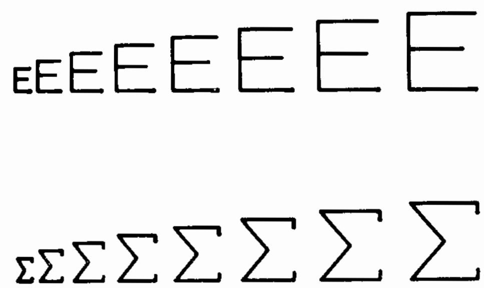
Send program with line 5 revised to be:

```
LINE 5.  UC32,28,9999,0,4,-32,0,16,-16,-16,-16,32,0,0,4;
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

Result:



NOTE: Use of the instruction PR 0 , 0 is recommended in EMULATE mode between SI and UC instructions when exact alignment with labeled characters is necessary. Roundoff error may otherwise cause slight misalignment which will be more noticeable in EMULATE mode where the primitive grid used by the UC instruction has less resolution. ■

The Downloadable Character Instruction, DL

USES: The DL instruction enables you to design characters and save them in a buffer for convenient repeated use in a program. Use the instruction whenever you want to create characters or symbols not included in the plotter's character sets.

SYNTAX: *DL* character number (,pen control), X-coordinate,
Y-coordinate(, . . .) (,pen control)(, . . .) *term*
or
DL character number *term*
or
DL *term*

PARAMETERS:

Parameter	Format	Range	Default
character number	integer	33-126	none
pen control	integer	−128	none
X,Y coordinates	integer	−127-127	none

EXPLANATION: Use of the instruction with no parameters (DL ;) clears the entire downloadable character buffer. Use of the instruction with the first parameter only (character number), clears that character only from the buffer. Following are further details of each parameter.

1. Character number. The first parameter specifies the decimal value of the character being defined. Only printing character numbers from 33-126 can identify downloadable characters. If the character number has been previously defined, the new definition overwrites the old one.
2. Pen control (−128). This parameter indicates that the next X,Y coordinate pair defines a *move* with the pen up. It is optional when it

immediately follows the character number, since the first X,Y coordinate pair always defines a move with the pen up. Except for these two cases, all other parameter pairs define a *draw* with the pen down.

3. X,Y coordinates. Numeric coordinate pairs after the character number constitute a series of vectors in *absolute* units.

Remember that the DL instruction uses absolute coordinates whereas the UC instruction uses relative coordinates. Also, with the UC instruction, you can program the first pen movement with the pen down; with the DL instruction, the first move is always with the pen up.

The following table summarizes error conditions for the DL instruction.

Condition	Error	Plotter Response
out-of-range parameter	3	instruction ignored; if previously defined, character retains old definition
> 255 parameters given (including pen control, excluding character number) in character definition	2	instruction ignored; if previously defined, character retains old definition
X is given without corresponding Y in character definition	2	instruction ignored; if previously defined, character retains old definition
buffer overflow (no space available in downloadable buffer)	7	character definition lost and character remains or becomes undefined; if previously defined, old definition also lost

How to Define a Downloadable Character

Define the character in absolute units on a 32 × 32 unit grid. The origin (0, 0) is in the lower-left corner. This is the same grid that is used for the fixed-vector character sets resident in the plotter. The area occupied by a 32 × 32 unit grid approximates the size of an uppercase "A." The downloaded character may extend outside this grid to ±127 units on each axis.

NOTE: Unlike the UC instruction, the grid used for downloadable characters is independent of both the currently selected character set and the setting of the **EMULATE/NORMAL** switch. ■

A downloadable character is drawn in the following manner.

1. Allocate space in the downloadable character table with the ESC . T instruction (see below).

2. Assign a character number (decimal value) to the downloadable character.

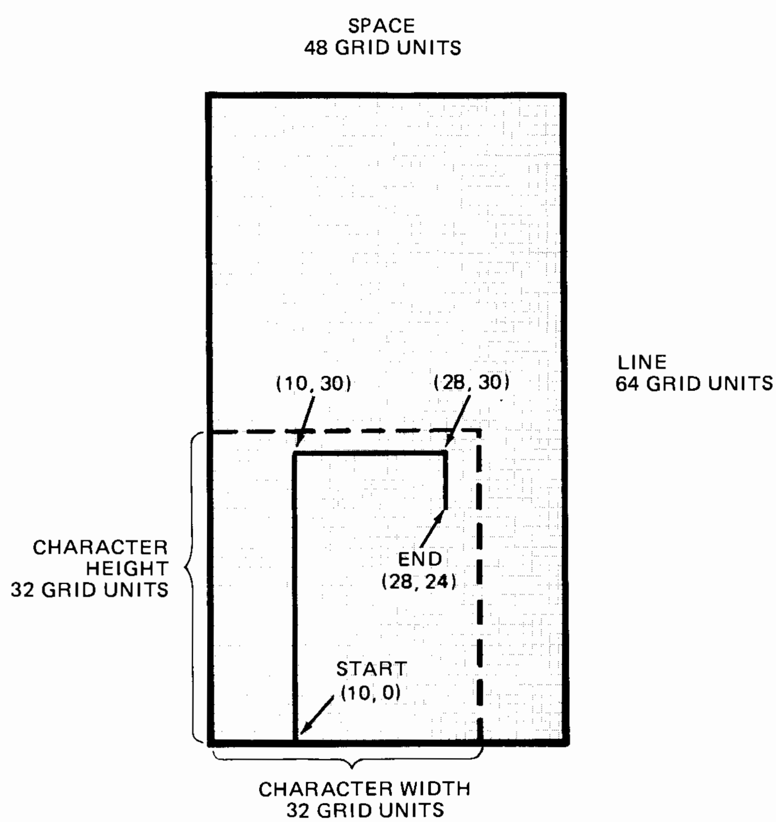
3. Move the pen to the starting point with the first X,Y coordinate pair. The pen control parameter (−128), which indicates the first pair is a move with the pen up, is optional.

4. Specify the vectors of the character with a sequence of X,Y coordinate pairs indicating the location on the grid from the origin. The X-coordinate specifies the number of grid units that the pen will move horizontally from the origin. The Y-coordinate specifies the number of grid units that the pen will move vertically from the origin.

NOTE: Remember that character set −1 must be designated and selected in the same manner as any other character set before downloadable characters can be used. ■

The diagram below illustrates how to create a gamma on a grid. This character is used in the label string in the programming example on the next page.

Cell for Creating Downloadable Characters



Any character defined by the DL instruction becomes a part of character set −1. Once defined, downloadable characters can be used in the LB and BL instructions when set −1 is selected. Characters may be downloaded in any order. The same character can be redefined as needed. When a character is redefined or undefined, the plotter reclaims the space and performs full data compaction.

The downloadable character buffer is cleared upon power up and upon receipt of an IN ;, DL ;, or ESC . T : instruction. All characters in set −1 become undefined. Response to these characters in a label string follows the current fallback mode as defined in the CM instruction. A DF instruction can be used to initialize other various settings without changing the downloadable buffer.

Allocation of Memory for Downloadable Characters

Space must be allocated in the downloadable character buffer with the ESC . T instruction before a character is defined with the DL instruction. Each character defined requires

one byte for each pen flag
one byte for each X-coordinate
one byte for each Y-coordinate
one additional byte for the byte count

The minimum amount of space required for downloadable characters is 444 bytes for overhead, plus whatever you need according to the above formula to define the character. For the gamma defined in the following program, you would need to allocate a minimum of 453 bytes: 444 for overhead, 8 for the coordinates, and 1 for the byte count. Refer to the ESC . T instruction for a complete explanation of memory allocation.

Use of Downloadable Characters in a Label Instruction

Downloadable characters are treated as a fixed-font set. After the character is printed with an LB or PB instruction, the character origin will advance 1.5 character widths (48 grid units) unless modified by the ES instruction. Refer to Character Spacing in this chapter for elaboration of label spacing. All regular text attributes — size, slant, direction, and label origin — apply to downloadable characters. They may also be combined with the plotter's resident character sets in label strings.

Use of Downloadable Characters with Symbol Mode

Define the downloadable character with the DL instruction *before* designating the character in The Symbol Mode Instruction, SM. Otherwise, the plotter generates error 3 and no symbol is drawn. Error 3 is also generated if symbol mode is enabled and the character becomes undefined after another SM instruction is received. If a previous definition exists for the character, the plotter will use the current definition instead of the old one. No error is generated in this case.

Example — Defining a Downloadable Character

The following example shows how to define a downloadable character and use it in a label string.

Send:

```
LINE 1.   CHR$(27)&".T1024;0;1000:"
LINE 2.   IN;CA-1;SP1;IP10000,0,15000,5000;
LINE 3.   DL 65,10,0,10,30,28,30,28,24;
LINE 4.   PA10500,2500;
LINE 5.   LB The symbol for gamma is %A%;SP0;
```

The symbol for gamma is Γ

Line 1 allocates 1000 bytes in downloadable character buffer

Line 2 initializes plotter; designates the downloadable set (−1) as the alternate character set; positions P1 and P2

Line 3 designates decimal equivalent 65 as downloadable character; defines character

Line 4 sets initial pen position

Line 5 labels "The symbol for gamma is," shifts out to alternate set, prints decimal equivalent of A in set −1, terminates label mode with $^{E}{}_{X}$; returns pen to carousel

Choosing Other Character Selection Modes

The default mode on all HP plotters is the HP 7-bit compatibility mode. In this mode you can access any character set using the CS, SS, CA, and SA instructions, which are presented earlier in this chapter. You can also use the shift-in and shift-out control characters in a label string to access the designated standard and alternate sets.

The plotter also offers three other character modes: the HP 8-bit mode, the ISO 7-bit mode, and the ISO 8-bit mode. One underlying architecture supports all of the modes. The HP 8-bit mode is a tool for converting European software to HP Systems software and will be of use primarily if you are using new HP systems software in a foreign language environment. The ISO modes permit more flexibility to European users who frequently want to access more character sets, yet conserve buffer space.

The following sections discuss background information common to the character modes: the in-use code table, fallback mode, and control characters. The remaining sections then present each character selection mode and the three instructions that must be executed to implement the alternative modes. These instructions are:

- The Character Selection Mode Instruction, CM — used to establish one of the modes.
- The Designate Character Set into Slot Instruction, DS — used to designate character sets (similar to the CS and CA instructions).
- The Invoke Character Slot Instruction, IV — used to select the set used for labeling (similar to the SS and SA instructions).

The specific details of the syntax of the CM, DS, and IV instructions are presented following the discussions of the individual character selection modes.

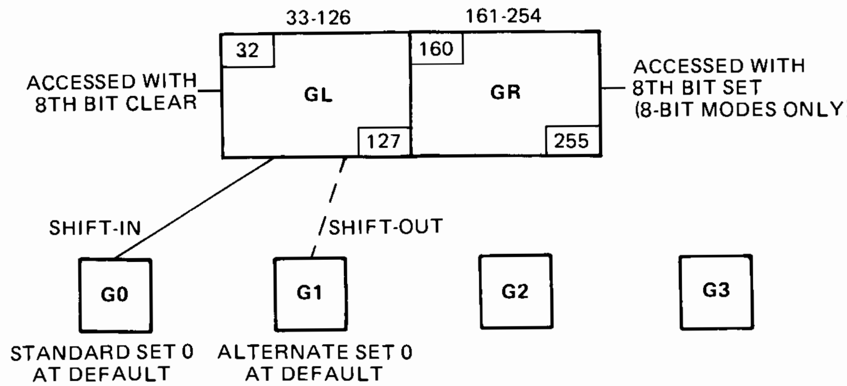
If the default HP 7-bit mode is adequate for your labeling needs, you do not need to read this section, nor the explanations of the CM, DS, and IV instructions that follow.

NOTE: To properly implement either the HP 8-bit mode or the ISO 8-bit mode, make sure the **PARITY** switch on the plotter's rear panel is set to **OFF** and that the computer system is operating with no parity. If your computer system is operating with parity, you must use a 7-bit mode. ■

The In-Use Code Table

On the next page is a model of the in-use code table which provides the underlying architecture for selection of the character sets. This table is used to map each character in a label string to the picture to be displayed. Each side, GL or GR, can hold a character set such as ANSI ASCII or ISO German.

The In-Use Code Table



GL and GR contain the currently active character sets and are accessed with the eighth bit off or on, respectively. GR can only be accessed by an 8-bit mode (the eighth bit is always set). GL and GR can be viewed as pointers to one of four "slots," G0-G3. These slots in turn point to the actual character sets, such as ANSI ASCII.

Each mode has character set 0 as both the default standard set in G0 and default alternate set in G1. Other default conditions are described below with the respective mode. Full implementation of this table in the ISO modes requires the following two-step process.

1. Designation of a character set into a slot with the DS instruction.
2. Invocation of a slot into GL or GR with the IV instruction.

A 7-bit mode can access a 128-character set which is contained in GL. In actuality, GL accesses characters 33–126, since the first 32 characters of any set, regardless of mode, are nonprinting control characters.

An 8-bit mode can access 256 characters. Control characters for an 8-bit mode are 128–160. In the 8-bit mode, GL accesses characters 33–126 and GR accesses characters 161–254.

Fallback Mode

The fallback mode pertains only to printed characters (decimal 33–126 and, for 8-bit modes, decimal 161-254) which are undefined. This mode determines the action taken when the plotter receives undefined printed characters in a label string. Examples of undefined characters are those areas which are blank in the Katakana and Roman Extension sets (see Plotter ASCII Code Definitions in Appendix C). The downloadable character set also consists initially of undefined characters. Unrecognized control characters are always NOP'd.

The plotter's default response is to ignore all undefined characters (NOP). When the fallback mode parameter is set to one, the plotter will draw a box (□) as a fallback character in place of an undefined character if the character is in the printing range. Some graphics standards mandate that character codes in the range of 33–126 (sometimes 127) which do not have graphic representation be executed with a special printing character. The fallback mode supports this. This fallback mode is also useful for debugging when the disappearance of characters from a label string may be confusing.

Control Characters

Control characters recognized by the plotter in the LB instruction in all CM modes are backspace, line feed, carriage return, shift-in, and shift out. In addition, VT (decimal 11) is implemented as an inverse line feed and HT (decimal 9) is implemented as a half-backspace.

Single-shift characters 142 and 143 are recognized in the ISO 8-bit mode. Unrecognized control characters are NOP'd regardless of the fallback mode, which pertains only to printed characters.

NOTE: The full range of invocations described in ISO 2022.2 which use two-character escape sequences are not operative in this version of the ISO modes. Since the escape character (decimal 27) is trapped by the escape sequence parser, which implements the immediately executed ESC . instructions, any following character not part of a plotter escape sequence would appear as part of the printed label. ■

HP 7-Bit Compatibility Mode

The HP 7-bit compatibility mode is the power-up mode currently implemented on all HP plotters. To use the HP 7-bit mode, you do not need to designate sets or invoke slots with the DS and IV instructions. Instead, you can simply access any of plotter's character sets with the CS, SS, CA, and SA instructions. Refer to these instructions, presented earlier in this chapter, for syntax details and examples of using different character sets in this mode.

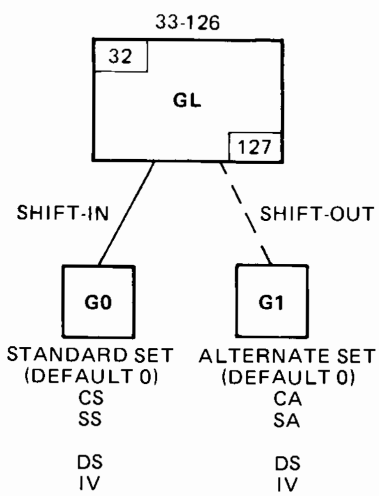
The ASCII control characters, shift-out and shift-in, are used to change character sets within a single label or line of text.

In this mode, the eighth bit is always cleared and ignored. All character sets are singlets composed of 128 characters; printed characters range from 33–126. Thus all character sets always reside in GL.

All invocations other than shift-in, shift-out, IV0,x, and IV1,x are NOP'd. In this mode, G2 and G3 cannot be designated with the DS instruction nor invoked with the IV instruction.

The in-use table for the HP 7-bit compatibility mode follows.

In-Use Code Table for HP 7-Bit Mode

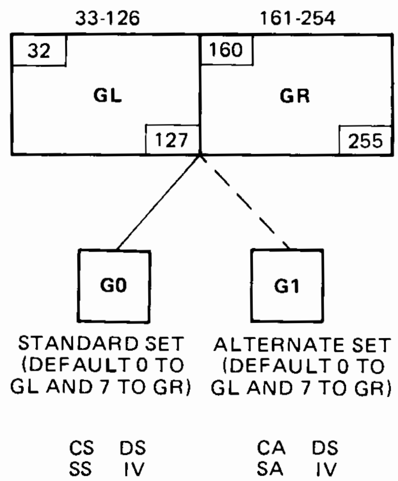


HP 8-Bit Mode

In the HP 8-bit mode, the plotter treats all sets as 256-character sets, or doublets. The setting of the eighth bit enables the plotter to access twice as many characters as it can in the 7-bit mode. Characters 0–127 reside in GL; characters 128–255 reside in GR. Only slots G0 and G1 are used in this mode.

When a singlet set is selected, its right half is considered to be composed of undefined characters and is treated according to the current fallback mode. The in-use code table for the HP 8-bit mode follows.

In-Use Code Table for HP 8-Bit Mode



The only full doublet sets are Roman8 and Katakana8. Roman8 contains ANSI ASCII and Roman Extensions. Katakana8 contains JIS ASCII and Katakana. These names (Roman8 and Katakana8) do not appear in the table below, but are mentioned here for reference purposes.

When ANSI ASCII is selected in any of the three fonts, characters 33–126 are from the ANSI ASCII set, and characters 161–254 are automatically lifted from Roman Extensions. When JIS ASCII is selected, characters 33–126 are from the JIS ASCII, and characters 161–254 are automatically lifted from Katakana. These paired relationships, shown in the table below, hold true regardless of whether ANSI ASCII or JIS ASCII is selected as an alternate or standard set.

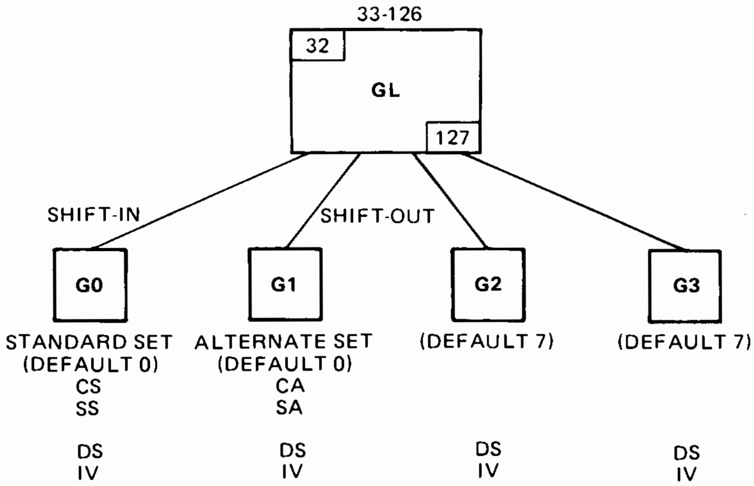
Set Selected	GL Characters 33-126	GR Characters 161-254
ANSI ASCII (0, 10, 20)	ANSI ASCII	Roman Extensions (7, 17, or 27)
JIS ASCII (6, 16, 26)	JIS ASCII	Katakana (8, 18, or 28)

The ASCII control characters, shift-in and shift-out, are used to change character sets within a single label or line of text.

All invocations other than shift-in, shift-out, IV 0,x and IV 1,x are NOP'd. In this mode, G2 and G3 cannot be designated with the DS instruction nor invoked with the IV instruction.

ISO 7-Bit Mode

In the ISO 7-bit mode, the eighth bit is always cleared and ignored by the plotter. When the plotter is initialized or default conditions are established, character set 0 resides in G0 and G1 and character set 7 resides in G2 and G3. The in-use code table for the ISO 7-bit mode follows.



In-Use Code Table for ISO 7-Bit Mode

Character sets are assigned to each slot with The Designate Character Set Instruction, DS. Standard and alternate sets, which are always located in G0 and G1, can also be assigned with the CS and CA instructions, which are subsets of DS.

In the listing below, n represents the character set number.

- G0 is designated with DS 0, n or CS n
- G1 is designated with DS 1, n or CA n
- G2 is designated with DS 2, n
- G3 is designated with DS 3, n

Slots are then invoked into GL in one of the following ways:

- The Invoke Character Slot Instruction, IV
- The Select Standard Set Instruction, SS
- The Select Alternate Set Instruction, SA
- The control codes, shift-in and shift-out, in a label string

All invocations with the IV instruction are locking. The invoked set remains resident in GL until it is overridden with a new IV instruction or default conditions are reestablished.

In the IS0 7-bit mode, any character set programmatically invoked into GR will automatically be invoked into GL. (For example, IV 3, 1; which normally invokes slot 3 into GR will instead invoke slot 3 into GL.)

Example — Using the DS and IV Instructions in the ISO 7-Bit Mode

Send:

```
LINE 1.  IN;SP1;IP5000,0,10000,5000;
LINE 2.  PA5100,2500;
LINE 3.  CM2;DS1,2;DS2,4;
LINE 4.  LB English  ␎Fran\ais␃
LINE 5.  IV2;LB  Espan|ol␃
```

English Français Español

Line 1 initializes plotter; selects pen 1; positions P1 and P2

Line 2 sets the initial pen position

Line 3 establishes the ISO 7-bit mode; designates character set 2 (French) into slot G1; designates character set 4 (Spanish) into slot G2

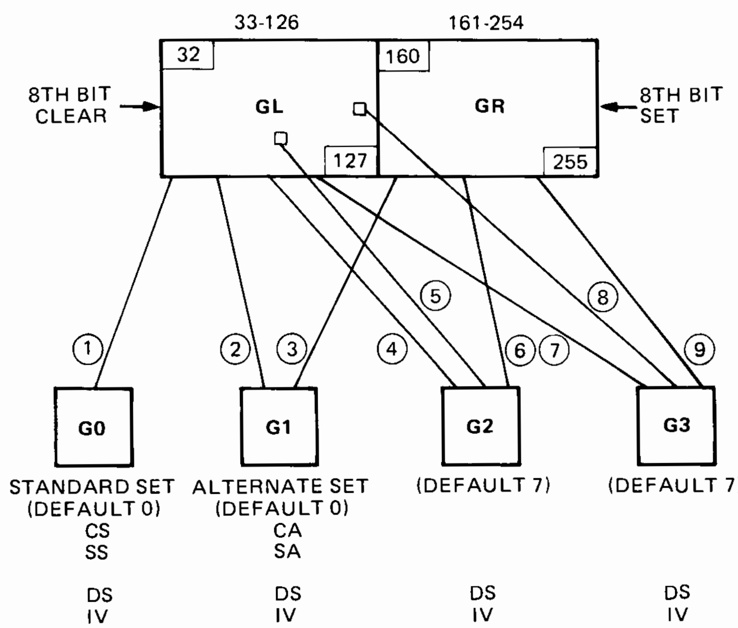
Line 4 prints "English" in the default character set; uses the shift-out control character to invoke slot G1 (containing the alternate set) and prints "Francais"

Line 5 invokes slot G2 into GL and prints "Espanol"

ISO 8-Bit Mode

The ISO 8-bit mode enables you to access two character sets at once without having to use invocations and designations. You can also access four character sets in any label string with the same economy. When the plotter is initialized or default conditions are established, character set 0 resides in G0 and G1 and character set 7 resides in G2 and G3. The in-use code table for the ISO 8-bit mode follows. The numbered items refer to possible invocations.

In-Use Code Table for ISO 8-Bit Mode



1. use G0 for GL (SS instruction or shift-in)
2. use G1 for GL (SA instruction or shift-out)
3. use G1 for GR
4. use G2 for GL
5. use G2 for next character only (single shift G2)
6. use G2 for GR
7. use G3 for GL
8. use G3 for next character only (single shift G3)
9. use G3 for GR

Use the DS instruction to designate character sets for each slot. You can also use the CS and CA instructions, which are subsets of DS, to designate standard and alternate sets for G0 and G1. Listed below are alternatives for designating character sets; n represents the character set number.

- Designate G0 with DS 0, n or CS n
- Designate G1 with DS 1, n or CA n
- Designate G2 with DS 2, n
- Designate G3 with DS 3, n

Slots are then invoked into GL and GR in one of the following ways.

- The Invoke Character Slot Instruction, IV
- The Select Standard Set Instruction, SS
- The Select Alternate Set Instruction, SA
- the control codes, shift-in and shift-out, in a label string
- the single-shift control codes (see below) in a label string

Invocations can be locking or single. All invocations with the IV instruction are locking: the invoked set remains resident in GL or GR until it is overridden with a new instruction or default conditions are reestablished.

A single invocation is in effect for one character only. The set is invoked, one character is printed, and the previous set is automatically reestablished. Control characters SS2 (decimal 142) and SS3 (decimal 143) enable a single shift, as indicated in the following table.

Single-Shift Control Character	Plotter Response
SS2 (142)	Invokes slot G2 for next character only
SS3 (143)	Invokes slot G3 for next character only

NOTE: In order to access the single-shift control codes or a character in GR, you must use a computer function such as CHR$ (or the equivalent for your computer). To determine the decimal value to use in the CHR$ function, simply add 128 to the decimal value of the desired character. You can find the decimal values of characters in Appendix C. Thus, to access an A in GR, you would use the CHR$ (193) function (128 + 65). ■

Example — Embedding a Single-Shift in a Label String

The following example shows how to embed a single-shift character in a label string. Four character sets are used in this example: The ANSI ASCII English set (0), the French/German set (2), the Special Symbols set (5), and the Drafting set (99).

Send:

```
 LINE 1.   IN;IP0,0,5000,5000;CM3;
 LINE 2.   SP1;PA100,4000;DS1,2;DS2,5;DS3,99
 LINE 3.   LB trigonometric formula:␎␍␊
           formule trigonome´trique:␃
★LINE 4.   CP;LB sec(";CHR$(142);"n-";
           CHR$(143);"\)=;CHR$(142);" psec";CHR$(143);"\
```

trigonometric formula:
formule trigonométrique:
sec (Π−ϕ) = ±secϕ

Line 1 initializes plotter; positions P1 and P2

Line 2 selects pen 1; sets initial pen position; designates character set 2 into slot 1; designates character set 5 into slot 2; designates character set 99 into slot 3

Line 3 enters label mode and labels "trigonometric formula" with default character set 0, then performs a shift-out (decimal 14) and carriage return (decimal 13) to label "formule trigonometrique" with character set 2

Line 4 performs carriage return and line feed with CP ;

uses decimal 142 for a single shift to pick up character Π (decimal 110) in character set 5

uses decimal 143 for a single shift to pick up character ϕ (decimal 92) in 99

uses decimal 142 for a single shift to pick up character ± (decimal 112) in set 5

uses decimal 143 for a single shift to pick up character ϕ in set 99

The Character Selection Mode Instruction, CM

USES: The CM instruction specifies the mode of character set selection and usage. Use this instruction to access one of four different character set modes: the HP 7-bit mode, the HP 8-bit mode, the ISO 7-bit mode, or the ISO 8-bit mode.

SYNTAX: *CM* switch mode (,fallback mode) *term*
or
CM *term*

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

PARAMETERS:

Parameter	Format	Range	Default
switch mode	integer	0–3	0
fallback mode	integer	0–1	0

EXPLANATION: A CM instruction with no parameters (CM ;) will default to the HP 7-bit mode and fallback mode 0. When only the first parameter is given, the second defaults to 0. The following paragraphs list further details of each parameter.

1. Switch Mode. The switch mode chooses which character selection mode will be used by all subsequent labeling instructions. The meanings of the four parameters follow. For more details, refer to the separate descriptions of each mode earlier in this chapter.

   0 HP 7-bit compatibility mode
   1 HP 8-bit mode
   2 ISO 2022.2 7-bit mode
   3 ISO 2022.2 8-bit mode

   NOTE: To properly implement either the HP 8-bit mode or the ISO 8-bit mode, make sure the PARITY switch on the plotter's rear panel is set to OFF and that the computer system is operating with no parity. If your computer system is operating with parity, you must use a 7-bit mode. ■

2. Fallback Mode. The fallback mode determines what the plotter does when it encounters undefined printing characters. The meanings of the two parameters follow. For more details, refer to Fallback Mode earlier in this chapter.

   0 Undefined characters are ignored
   1 A box (□) is drawn in place of undefined characters

Execution of any CM instruction clears the label buffer. Any execution of a CM instruction that changes from the current mode to a different mode sets default designations and invocations for the new mode. The CM instruction remains in effect until another CM instruction is executed, or the plotter is initialized or set to default conditions.

The CM instruction determines the implementation of The Designate Character Set into Slot Instruction, DS, and The Invoke Character Slot Instruction, IV. All labeling instructions (LB, BL, PB) operate according to the mode established by the CM instruction.

For examples of the CM instruction, refer to separate descriptions of the character selection modes presented earlier in this chapter.

The Designate Character Set into Slot Instruction, DS

USES: The DS instruction designates up to four character sets to be immediately available for labeling instructions. Use this instruction primarily with the ISO modes when accessing multiple character sets.

SYNTAX: *DS* slot, set *term*
or
DS *term*

PARAMETERS:

Parameter	Format	Range	Default
slot	integer	HP modes: 0–1 ISO modes: 0–3	0
set	integer	−1, 0–59, 99	0

EXPLANATION: Use of the instruction without parameters (DS ;) establishes default conditions. Use of the instruction with only one parameter sets error number 2. The following paragraphs list further details of each parameter.

1. Slot. The slot parameter represents the part of the in-use code table where character sets can be designated. Once it is designated, you can invoke any slot into GL or GR for use in a labeling instruction. For more details, refer to the separate descriptions of the character selection modes presented earlier in this chapter. The slot parameters follow.

   0 Slot G0
   1 Slot G1
   2 Slot G2
   3 Slot G3

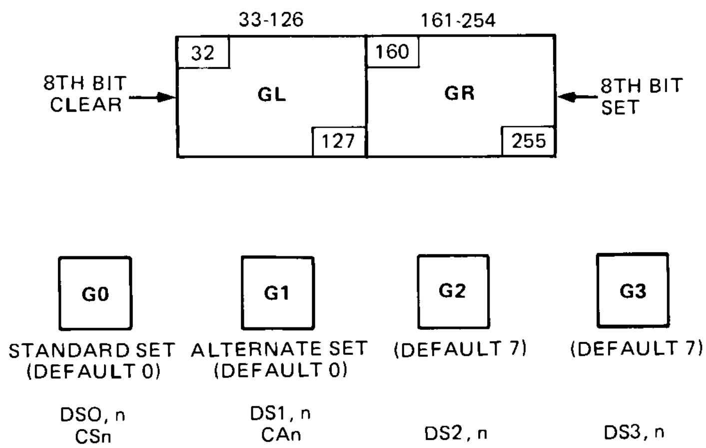
2. Set. The set parameter is the number of the character set to be designated into each slot. The character set numbers are listed under Plotter Character Sets at the beginning of this chapter, and in Appendix C.

The DS instruction designates any of the plotter's character sets into slots G0–G3. The slots are then invoked into the in-use code table with the IV, SS, or SA instructions, or control codes in a label string.

The Character Selection Mode Instruction, CM, determines the implementation of this instruction. It is implemented fully in the ISO modes only. Refer to the separate descriptions of each character set selection mode earlier in this chapter for examples and more details of how DS is implemented.

When the plotter is initialized or default conditions are established, slots G0 and G1 each contain character set 0; slots G2 and G3 each contain character set 7. The relationship of the slots to the in-use code table follows; n represents the character set number.

Slot Designations with DS Instruction



In order to designate character sets into more than one slot, you must execute the DS instruction once for each slot. For example, to place character set 2 in slot G0, set 3 into slot G1, set 5 into slot G2, and set 8 into slot G3, you would send the following instruction sequence.

DS0,2;DS1,3;DS2,5;DS3,8;

The designation for a particular slot remains in effect until another DS (or CS or CA) instruction is executed for that slot, or the plotter is initialized or set to default conditions.

NOTE: The CS and CA instructions are subsets of the DS instruction. Any designation into slot G0 (DS0,n) is the same as designating a standard set with the CS instruction. Accordingly, any designation into slot G1 (DS1,n) is the same as designating an alternate set with the CA instruction. However, if you are using either ISO mode, you must use DS to designate character sets into G2 or G3. Remember that in the HP modes, slots G2 and G3 are ignored. ■

The following table summarizes the possible error conditions for the DS instruction.

Condition	Error	Plotter Response
one parameter	2	ignores instruction
more than 2 parameters	2	executes first 2 parameters
slot parameter out-of-range	3	ignores instruction
set parameter out-of-range	5	ignores instruction

The Invoke Character Slot Instruction, IV

USES: The IV instruction invokes a character set slot into either the left or right half of the in-use code table (GL or GR, respectively). Use this instruction primarily with the ISO modes for accessing multiple character sets.

The IV instruction differs from the SS and SA instructions in that you can invoke the G2 and G3 slots with the IV instruction. You can also invoke slots into GR with the IV instruction. If you are only interested in invoking the G0 and G1 slots in GL (you are using the HP modes, for example), you can also use the SS and SA instructions, which are subsets of the IV instruction.

SYNTAX: *IV* slot (,left) *term*
or
IV *term*

PARAMETERS:

Parameter	Format	Range	Default
slot	integer	HP modes: 0–1 ISO modes: 0–3	0
left	integer	$\pm(2^{26}-1)$	0

EXPLANATION: An IV instruction without parameters (IV ;) establishes default conditions: for all modes G0 is invoked into GL; for the ISO 8-bit mode, G1 is invoked into GR. The following paragraphs list further details of each parameter.

1. Slot. The slot parameter is the number of the slot to be invoked into the right or left half of the in-use code table. For more details, refer to The In-Use Code Table and the separate descriptions of the character selection modes earlier in this chapter. The values for the slot parameters follow.

   0 Slot G0
   1 Slot G1
   2 Slot G2
   3 Slot G3

2. Left. This parameter designates which side of the in-use code table will receive the character set, as follows.

0 or no parameter	slot invoked into GL
any integer within the range, except 0	slot invoked into GR

The implementation of the IV instruction is determined by The Character Selection Mode Instruction, CM. Refer to separate descriptions of each character selection mode earlier in this chapter for more details of how IV is implemented.

All invocations with the IV instruction are locking. The invoked set remains in GL or GR until overwritten with a new instruction (IV, SS, or SA), or the plotter is initialized or set to default conditions.

In the ISO 7-bit mode, all invocations are automatically put into GL. The second parameter is ignored.

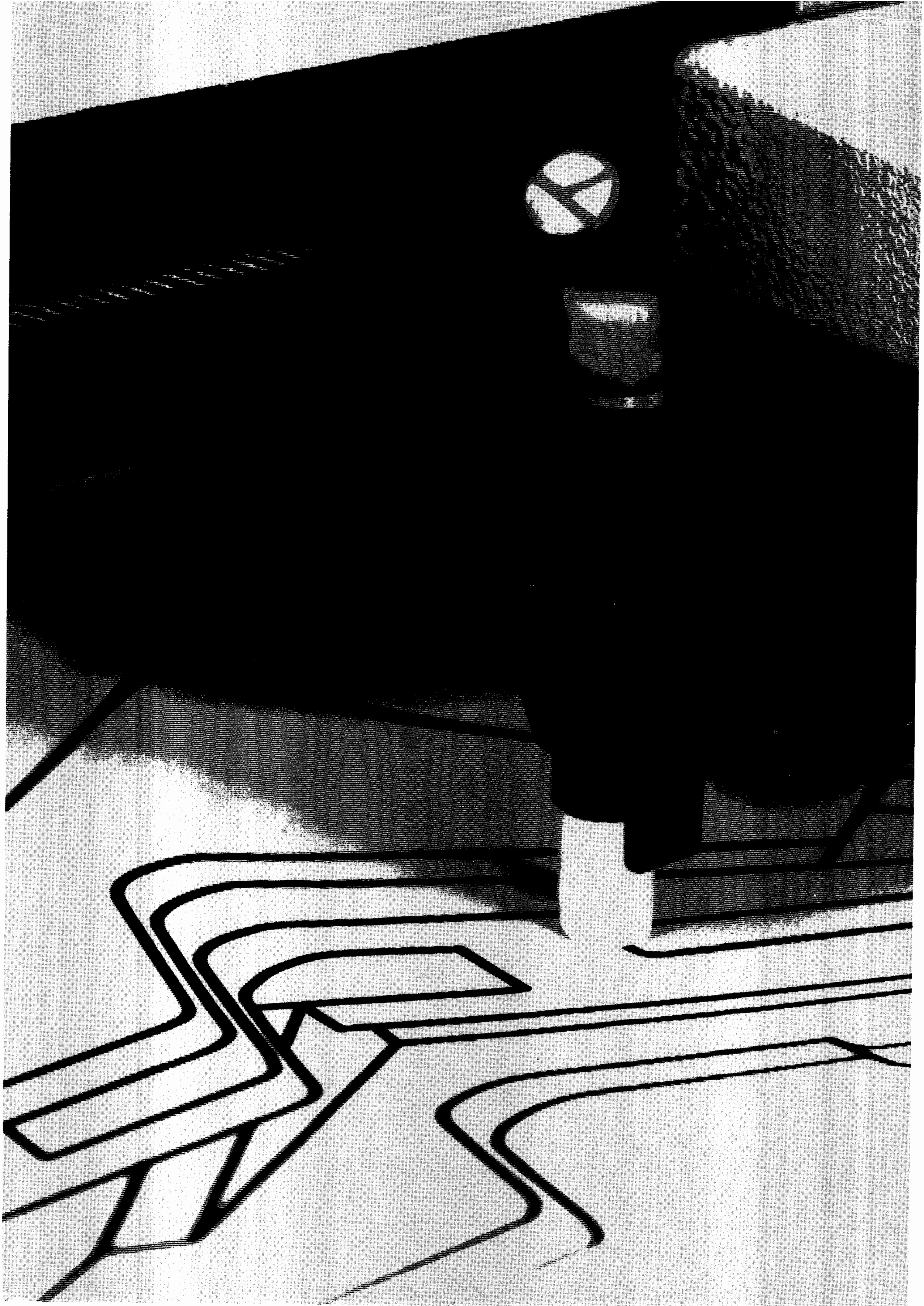
In the ISO 8-bit mode, the instruction is implemented fully and executed as specified.

In the HP modes, G2 and G3 cannot be invoked explicitly. The second parameter is ignored because GL and GR are linked in this mode.

NOTE: The instruction IV 0,0; or IV,0; is the same as SS; or a shift-in character (decimal 15) in the label string. All invoke the standard set in slot G0 into GL. Similarly, IV 1,0; or IV 1; is the same as SA or a shift-out character (decimal 14) in the label string. All invoke the alternate set in slot G1 into GL. ■

The following table summarizes the possible error conditions for the IV instruction.

Condition	Error	Plotter Response
one parameter	none	invokes specified slot into GL
more than 2 parameters	2	executes first 2 parameters
parameter out-of-range	3	ignores instruction



Chapter 8
Digitizing

What You'll Learn in This Chapter

The plotter can be used as a digitizer as well as a plotter. Digitizing consists of moving the pen or digitizing sight to a point on the plotting surface, entering the point, and sending the coordinates of that point to the computer. This chapter describes the three instructions used in digitizing, and contains a discussion of the steps required by a computer program in which digitizing is done as well as some sample programs. Included in the discussion are three different methods of assuring that a point has been entered. The method used depends on the application and the interface (HP-IB or RS-232-C).

HP-GL Instructions Covered

DP The Digitize Point Instruction
DC The Digitize Clear Instruction
OD The Output Digitized Point and Pen Status Instruction

Digitizing

Preparing the Plotter for Use as a Digitizer

The digitizing sight supplied with the plotter should be used for digitizing. It may be loaded in a carousel stall from which the boot has been removed, or placed directly into the pen holder on the pen arm and used with no carousel loaded. The sight is used in the pen down position.

Digitizing cannot be done when an HP-IB plotter is in listen-only mode because the plotter cannot send the coordinates of a point to the computer.

When the plotter is in digitize mode (the LED in **ENTER** is on), automatic pen storage and pen lift are disabled. Even when the plotter is not in digitize mode, if a digitizing sight has been manually loaded into the pen holder after a pen has been stored using front-panel controls or SP 0, the digitizing sight will not be stored or lifted automatically. An AP 4 or AP 0 instruction will disable automatic pen storage and pen lift, and may be included in programs which digitize to assure that the digitizing sight will not be stored or raised without user action. However, the AP 4 or AP 0 instruction disables storage for pens as well as the digitizing sight.

The Digitize Point Instruction, DP

USES: The DP instruction digitizes points on the plotter. Use this instruction to input data for a graphics program or obtain the coordinates of a point or points on the plot.

SYNTAX: *DP term*

PARAMETERS: None.

EXPLANATION: When the DP instruction is sent, the ENTER LED comes on, indicating that a point can be entered. Automatic pen storage and pen lift, as set by default or an AP instruction, are disabled. When ENTER is pressed, the X- and Y-coordinates of that point and pen up/down status are stored for retrieval by the OD instruction. Pressing ENTER also turns off the ENTER LED and sets bit position 2 of the status byte, indicating a digitized point is available for output. After ENTER has been pressed, automatic pen storage and pen lift are again controlled by the AP instruction or default values.

The Digitize Clear Instruction, DC

USES: The DC instruction terminates the digitize mode. Use this instruction to terminate digitize mode without entering a point. If you are using an interrupt routine in a digitizing program to branch to some other plotting function, you could use DC to clear digitize mode immediately after branching.

SYNTAX: *DC term*

PARAMETERS: None.

EXPLANATION: This instruction terminates digitize mode without entering a point. The ENTER LED is turned off (if it was on).

The Output Digitized Point and Pen Status Instruction, OD

USES: The OD instruction outputs the X- and Y-coordinates and pen status (up or down) associated with the last digitized point. Use this instruction after DP and ENTER in all digitizing applications to return the coordinates of the digitized point to the computer.

SYNTAX: *OD term*

PARAMETERS: None.

EXPLANATION: The timing of the output depends on the plotter's interface (HP-IB or RS-232-C). Refer to the introduction to Chapter 9 and Chapter 12 (HP-IB) or Chapter 13 (RS-232-C) for more information.

The pen position and status are output to the computer as integers in ASCII in the form:

X, Y, P [TERM]

where X is the X-coordinate of the digitized point
Y is the Y-coordinate of the digitized point
P is the pen status (0 = pen up, 1 = pen down) when the point was entered.

The range of the X,Y values and the units in which they are output are determined by the setting of the **EMULATE/NORMAL** switch as follows:

EMULATE

Range — X- and Y-coordinates within the current hard-clip limits
Units — X- and Y-coordinates in plotter units

NORMAL

Range — X- and Y-coordinates within the current hard-clip limits (expressed as plotter or user units)
Units — X- and Y-coordinates in plotter units when scaling is off, and in user units when scaling is on

The plotter outputs a negative sign for negative numbers; positive signs are suppressed.

Upon execution of the OD instruction by the plotter, bit position 2 of the status byte is cleared.

Digitizing With the Plotter

When using the plotter as a digitizer, it is important to ascertain that a point has been entered before an attempt is made to retrieve that point using the OD instruction. There are three methods for doing this.

The first method, which might be called the manual method, is easiest to understand. It is not efficient in applications where many points will be entered, or in an RS-232-C environment where the mainframe is not adjacent to the plotter or where human intervention in program execution is not possible. The steps in this method are as follows:

1. In a program, send a DP instruction to the plotter, causing the **ENTER** LED to come on indicating the plotter is waiting for a point to be entered. Follow the DP instruction immediately with a statement that will cause the program to pause until instructed to continue. The BASIC statement, PAUSE, should be used on HP desktop computers.

2. Move the digitizing sight (pen) to the point to be entered, either programmatically or using the joystick.

3. Press **ENTER** on the plotter's front panel. Now resume running of the program. This is done on HP desktop computers by pressing the key marked **CONTINUE** or **CONT**.

4. The program step following the pause will now be executed. The next steps of the program, in order, should be an OD instruction to the plotter, a read statement by the computer to read the X- and Y-coordinate and the pen status, and then steps to process the digitized data in the appropriate manner.

Using this method there is no need to monitor the status byte because the program does not proceed to the OD instruction until the user enters a point and causes the program to resume.

A short program to digitize a single point and display the coordinates and pen status is given below. The program is in BASIC for the HP 9835/9845.

```
10 PRINTER IS 7,5
20 PRINT"DP;"
30 PAUSE
40 PRINT "OD;"
50 ENTER 7,5;X,Y,P
60 DISP X,Y,P
70 END
```

The second method can be used with either interface and is the only method of checking based on software that can be done in an RS-232-C environment. This method monitors bit position 2, the third least significant bit, of the plotter's status byte which is set when a digitized point is available. Refer to The Output Status Instruction, OS, Chapter 9, for more information.

Monitoring bit position 2 can be done in a variety of ways depending on the instructions available on the computer being used. If there are instructions to check bits directly, the third least significant bit (lsb) should be checked for the occurrence of a 1. If no bit operations are available, the status byte can be operated on arithmetically to check for the availability of a digitized point. Executing successive divisions of a number by two and checking for an odd or even integer answer is a common way of monitoring bits without converting the number to binary form. The following sequences of BASIC instructions will check the proper bit of the status byte. The first sequence checks bit 2 directly, using the BIT function available in HP desktop computers. The second sequence uses the division method where the MOD function is available. The third sequence uses only the division method. Insert a suitable BASIC read statement, where indicated, to read the status byte into a variable called Status.

```
1    PRINT "OS;"
2                          ! STORE STATUS BYTE IN Status
3    IF BIT(Status,2) THEN 5  ! BRANCHES IF BIT 2 IS 1
4    GOTO 1           ! NO POINT AVAILABLE;KEEP CHECKING
5    PRINT "OD;"         ! SEND OD SINCE POINT AVAILABLE
```

```
10   PRINT "OS;"
20                         ! STORE STATUS BYTE IN Status
30   Status=INT(Status/2)        ! SHIFTS BITS RIGHT ONE
40                               ! POSITION
50   Status=INT(Status/2)        ! SHIFTS BITS RIGHT
60                               ! ANOTHER POSITION
70   Status=Status MOD 2       ! USES BUILT IN FUNCTION;
80                             ! RESULT 0 IF LSB NOT 1
90   IF Status=0 THEN 10         ! NO POINT AVAILABLE;
100                              ! KEEP CHECKING
110 PRINT "OD;"        ! SEND OD SINCE POINT AVAILABLE
```

```
200 PRINT "OS;"
210                        ! STORE STATUS BYTE IN Status
220 Status=INT(Status/4)       ! COMBINES LINES 30 & 50
230                            ! IN 2nd PROGRAM
240 IF Status=INT(Status/2)*2 THEN 200 !EVEN INTEGER;
250                                      ! LSB NOT 1;
260                                     ! KEEP CHECKING
270 PRINT "OD;"        ! SEND OD SINCE POINT AVAILABLE
```

In many applications, a large number of points need to be digitized. When the computer is used to monitor bit position 2, the points may or may not be processed immediately. In most applications, memory would be allocated for the total number of points to be digitized. A loop could be established to process the total number of points, calling the subroutine each time to check that a point had been entered. A complete BASIC program for the HP 9835/9845 follows. This program prints out the 500 points after they all have been entered.

```
10  PRINTER IS 7,5
20  OPTION BASE 1
30  INTEGER X(500),Y(500),P(500)
40  FOR C=1 TO 500
50    PRINT "DP;"
60    DISP "ENTER POINT ";C
70    GOSUB 160
80    PRINT "OD;"
90    ENTER 7,5;X(C),Y(C),P(C)
100 NEXT C
110 PRINTER IS 0
120 FOR C=1 TO 500
130   PRINT X(C),Y(C),P(C)
140 NEXT C
150 STOP
160 ! CHECK SUBROUTINE
170   PRINT "OS;"
180   ENTER 7,5;Status
190   IF BIT(Status,2)=0 THEN 170
200   RETURN
210 END
```

A third method can be used by advanced programmers thoroughly familiar with the HP-IB interface, polling techniques, and interrupts. It should only be used when the computer can perform useful tasks while waiting for the digitized point to be entered. This method involves setting a value of 4 in the S-mask of the IM instruction, e.g., IM 223 , 4 , 0 ; to cause the plotter to generate an RQS (service request) when a digitized point is available. With an interrupt routine enabled for service requests, the computer can send a DP instruction to initiate digitizing, and then proceed with some other task until the digitized point is entered. When the point is available, the computer is interrupted by the RQS, and program execution branches to the routine to process the digitized data. This routine could simply send an OD instruction and read the digitized point, or it could perform bit checking of the plotter status byte if multiple S-mask values have been specified to generate the RQS. The status byte can be obtained by serial polling or simply by sending an OS instruction. Because interrupts and polling are highly machine-dependent and beyond the scope of this manual, no examples will be given.


0 PRINTER IS 705
0 PRINT "OP;"
0 ENTER 705 ; A,B,C,D
0 PRINT "OE;"
0 ENTER 705 ; E
0 PRINT "OI;"
0 ENTER 705 ; A$
0 DISP A;B;C;D
0 DISP E;A$
0 END
-20848 -16171 20848
0

Chapter 9 General Programming Instructions

What You'll Learn in This Chapter

In this chapter you will learn about the conditions under which an error indicator, service request message, and parallel poll response will occur. You will also learn how to obtain information from the plotter concerning its current pen location, error types, factors, identification, options, status, and the currently installed carousel type. The timing of output depends on your interface (HP-IB or RS-232-C). Before using the output instructions, you should read the notes below and the appropriate interfacing chapter in this manual.

HP-GL Instructions Covered

IM	The Input Mask Instruction
OA	The Output Actual Position and Pen Status Instruction
OC	The Output Commanded Position and Pen Status Instruction
OE	The Output Error Instruction
OF	The Output Factors Instruction
OI	The Output Identification Instruction
OO	The Output Options Instruction
OS	The Output Status Instruction
OT	The Output Carousel Type Instruction

Notes for an HP-IB User

When the plotter is in the HP-IB Interface Mode, the terminator for an output statement, denoted [TERM], is a carriage return followed by a line feed.

The output instructions in this chapter should not be used when the plotter is in listen-only mode because the plotter in listen-only mode cannot output anything. Output instructions will be ignored by the plotter so the computer will get no response to its read statement, and, typically, the program will halt.

A plotter in the HP-IB Interface Mode will respond only when the computer sends a read command (the plotter is instructed to talk). Therefore, a read statement should directly follow any output instruction. When a second output instruction is received before the data from the first instruction has been read, the new data overwrites the old data and the old data is lost. Note also that the plotter sets the EOI line high concurrently with output of the line feed terminator. This EOI signal is interpreted by some computers to mean that the read statement has been satisfied and program execution can resume. Refer to Chapter 12 for more information.

Notes for an RS-232-C User

In the RS-232-C Interface Mode, the terminator for an output statement, denoted [TERM], is a carriage return, unless the terminator is modified by an ESC . M instruction. As soon as an output instruction has been parsed by the plotter, output occurs according to the handshake protocol established by the ESC . M and ESC . N instructions. Use of plotter turnaround delays and intercharacter delays should be specified as necessary to assure that output will not be lost because the computer is not prepared to receive it. The information necessary to assure this should be contained in the documentation for your computer. Refer to Chapter 13 of this manual for more information.

The Input Mask Instruction, IM

USES: The IM instruction controls the conditions under which HP-GL error status is reported, the conditions that can cause an HP-IB service request message, and the conditions that can cause a positive response to an HP-IB parallel poll. Use this instruction with either the HP-IB or RS-232-C Interface Mode to change the conditions under which HP-GL error status is reported. In the HP-IB Interface Mode, use the instruction to enable the plotter to send a service request message when specified bits of the status byte are set, and/or enable a positive response to a parallel poll under the conditions specified.

SYNTAX: *IM* E-mask value (, S-mask value (, P-mask value)) *term*
or
IM *term*

PARAMETERS:

Parameter	Format	Range	Default
E-mask value	integer	*	223
S-mask value	integer	*	0
P-mask value	integer	*	0

*Refer to the table on the next page for each parameter.

EXPLANATION: The plotter, when set to default values or initialized, automatically sets the E-mask to 223 (error numbers 1, 2, 3, 4, 5, 7, 8), the S-mask to 0 (none of the status-byte bits can send the service request message), and the P-mask to 0 (none of the status-byte bits can cause a parallel poll response of logical 1).

An IM instruction with no parameters (IM ;) will automatically set the values to 223 , 0 , 0.

In the RS-232-C Interface Mode, the S- and P-masks are of no use and are ignored if present.

1. E-mask. The E-mask value specified is the sum of any combination of the bit values shown below. When an HP-GL error occurs, the bit in the E-mask corresponding to the error number as shown below is tested to determine if the error bit (bit 5) of the status byte is to be set and the front-panel **ERROR** LED is to be turned on. Refer to The Output Status Instruction, OS, later in this chapter. If a bit is not set, there is no way to ever determine if that error occurred.

E-Mask Bit Value	Bit	Error Number	Meaning
1	0	1	Instruction not recognized
2	1	2	Wrong number of parameters
4	2	3	Bad parameters received
8	3	4	Not used
16	4	5	Unknown character set
32	5	6	Position overflow
64	6	7	Not used
128	7	8	Not used

An E-mask value 60 (4 + 8 + 16 + 32) will specify that errors 3 through 6 will set the error bit in the status byte and turn on the ERROR LED whenever they occur. Errors 1 and 2, however, will not set the error bit or turn on the ERROR LED if they occur since they are not included in the E-mask value. Errors 4, 7, and 8 never occur. Therefore, specifying these errors in E-mask has no effect.

2. S-mask. The S-mask value specified is the sum of any of the bit values shown below. When a bit of the status byte changes value, the status byte is ANDed with the S-mask in a bit-by-bit fashion to determine if bit 6 of the status byte is to be set and the service request message sent. The status of bit 6 changes as plotter conditions change, and is cleared or set as required.

S-Mask Bit Value	Status Bit Number	Meaning
1	0	Pen down
2	1	P1 or P2 changed
4	2	Digitized point available
8	3	Initialized
16	4	Ready for data (buffer empty)
32	5	Error
64	6	Not used
128	7	Not used

For example, an S-mask value of 16 specifies that the Ready for Data bit (bit 4) of the status byte will send the service request message. The other bits (bits 0 through 3, bits 5 through 7) will not send the service request message.

3. P-mask. The P-mask value specifies which of the status-byte conditions will result in a logical 1 response to an HP-IB parallel poll.

P-Mask Bit Value	Status Bit Number	Meaning
1	0	Pen down
2	1	P1 or P2 changed
4	2	Digitized point available
8	3	Initialized
16	4	Ready for data
32	5	Error
64	6	Not used
128	7	Not used

For example, a P-mask value of 48 specifies that only bits 4 and 5 (16 + 32) of the status byte can cause the plotter to respond to a parallel poll with a logical 1 on the appropriate data line.

The Output Actual Position and Pen Status Instruction, OA

USES: The OA instruction outputs the X- and Y-coordinates and pen status (up or down) associated with the actual pen position. Use this instruction to determine the pen's current position in plotter units. You might use that information to position a label or figure, or determine the parameters of some desired window.

SYNTAX: *OA term*

PARAMETERS: None.

EXPLANATION: The OA instruction can be used with scaling on or off (use OD when in digitize mode).

The pen position and status are output to the computer as integers in ASCII in the form:

X, Y, P [TERM]

where X is always the X-coordinate in plotter units
Y is always the Y-coordinate in plotter units
P is the pen status (0 = pen up, 1 = pen down).

The ranges of the X- and Y-coordinates are the current hard-clip limits.

The plotter outputs a negative sign for negative numbers; positive signs are suppressed.

The Output Commanded Position and Pen Status Instruction, OC

USES: The OC instruction outputs the X- and Y-coordinates and pen status (up or down) associated with the last valid pen position instruction. Use this instruction to determine the pen's last valid commanded position in plotter units or user units depending whether scaling is off or on. You

Gen. Prog. Inst.

might use this information to position a label or figure, or determine the parameters of an instruction which moved the pen to the limits of some window.

SYNTAX: *OC term*

PARAMETERS: None.

EXPLANATION: The OC instruction can be used with scaling on or off (use OD when in digitize mode). This instruction is especially useful when the pen is physically at the plotting limits and the pen position does not coincide with the instructed position, or when output in user units is desired.

The pen position and status will be output to the computer as integers in ASCII in the form:

X, Y, P [TERM]

where X is the X-coordinate in plotter units or user units
Y is the Y-coordinate in plotter units or user units
P is the pen status (0 = pen up, 1 = pen down).

When scaling is off, X- and Y-coordinates are in plotter units. When scaling is on, X- and Y-coordinates are in user units. The plotter outputs a negative sign for negative numbers; positive signs are suppressed. The range of both X- and Y-coordinates is determined by the setting of the EMULATE/NORMAL switch as follows:

EMULATE = −32 768 to +32 767

NORMAL = -2^{29} to $2^{29} - 1$

The Output Error Instruction, OE

USES: The OE instruction outputs the number corresponding to the first HP-GL error (if any). Use this instruction to determine the type of the first error. It is useful when debugging programs or to determine if all data or instructions were accepted by the plotter.

SYNTAX: *OE term*

PARAMETERS: None.

EXPLANATION: When an OE instruction is received, the plotter converts the last HP-GL error to a positive decimal integer in ASCII which is output in the form:

error number [TERM]

The error numbers are defined on the following page.

HP-GL Errors

Error Number	Meaning
0	No error
1	Instruction not recognized
2	Wrong number of parameters
3	Out-of-range parameters, or illegal character
4	Not used
5	Unknown character set
6	Position overflow
7	Buffer overflow for polygons or downloadable characters
8	Page-advance instruction (AF, AH, FR, or PG) executed when sheet paper loaded (HP 7586 only)

After execution of an OE instruction, bit position 5 of the status byte is cleared (if set), and the ERROR LED (if lit) is turned off (unless there is an RS-232-C error which has not been cleared by an ESC . E instruction).

The Output Factors Instruction, OF

USES: The OF instruction outputs the number of plotter units per millimetre in each axis. This instruction enables the plotter to be used with software which must know the size of a plotter unit.

SYNTAX: *OF term*

PARAMETERS: None.

EXPLANATION: The plotter will always output the following:

40 , 40 [TERM]

These factors indicate that there are 40 plotter units per millimetre in the X-axis and in the Y-axis (0.025 mm/plotter unit).

The Output Identification Instruction, OI

USES: The OI instruction outputs a plotter identifier. This instruction is especially useful in a remote operating environment to determine which model plotter is on-line.

SYNTAX: *OI term*

PARAMETERS: None.

EXPLANATION: The plotter will always output the following appropriate character string:

7580B [TERM]
or
7585B [TERM]
or
7586B [TERM]

The Output Options Instruction, OO

USES: The OO instruction outputs eight option parameters. This instruction is especially useful in a remote operating environment to determine which options are available in the plotter that is on-line.

SYNTAX: *OO term*

PARAMETERS: None.

EXPLANATION: The plotter will always output the following eight ASCII integers separated by commas:

```
C,1, 0,0,1,1,0,1  [TERM]
                  └─ Indicates plotter has user-definable RAM.
             └─Indicates plotter has polygon fill instructions.
           └─Indicates plotter has arc and circle instructions.
  └─Indicates plotter has pen select capability.
└─ Indicates plotter has clean paper/paper advance status.
```

C can take on a value from 0 to 3 depending on the following settings of the paper check bit and the advance option. The paper check bit is set to "1" whenever the plotter has drawn a mark on a piece of paper; otherwise it is set to "0". The advance option is available only on the HP 7586; the plotter must be loaded with roll media for the OO instruction to indicate the advance option is available.

Paper Check Bit		HP 7586 Advance Option		C Value
0 (paper clean)	+	0 (sheet media loaded)	=	0
0 (paper clean)	+	1 (roll media loaded)	=	1
1 (paper marked)	+	0 (sheet media loaded)	=	2
1 (paper marked)	+	1 (roll media loaded)	=	3

On the HP 7580/7585 plotter, C can only have a value of 0 or 2.

On the HP 7586 plotter, C can have values 0 through 3.

The Output Status Instruction, OS

USES: The OS instruction outputs the decimal equivalent of the status byte. Use this instruction in debugging operations and in digitizing applications.

SYNTAX: *OS term*

PARAMETERS: None.

EXPLANATION: Upon receipt of the OS instruction, the internal eight-bit status byte is converted to an integer between 0 and 255. Output is in ASCII in the form:

status [TERM]

The status byte bits are defined on the next page.

Bit Value	Bit Position	Meaning
1	0	Pen down
2	1	Changed or attempted change of P1 or P2; cleared by OP
4	2	Digitized point available; cleared by OD
8	3	Initialized; cleared by OS
16	4	Ready for data (buffer empty)
32	5	Error; cleared by OE
64	6	Request service (RQS)
128	7	Not used (always 0)

Upon power-up, the status is 26, the sum of 2 (P1 and/or P2 newly established), 8 (initialized), and 16 (ready for data).

Upon execution of an OS instruction by the plotter, bit position 3 is cleared and the status is digital 16 (ready for data).

The Output Carousel Type Instruction, OT

USES: The OT instruction outputs the current carousel type and stall occupancy information. This instruction is especially useful in a remote operating environment to determine the carousel type and occupied pen stalls.

SYNTAX: *OT term*

PARAMETERS: None.

EXPLANATION: The current carousel type and its pen map are output as two decimal integers in ASCII in the form:

type, map [TERM]

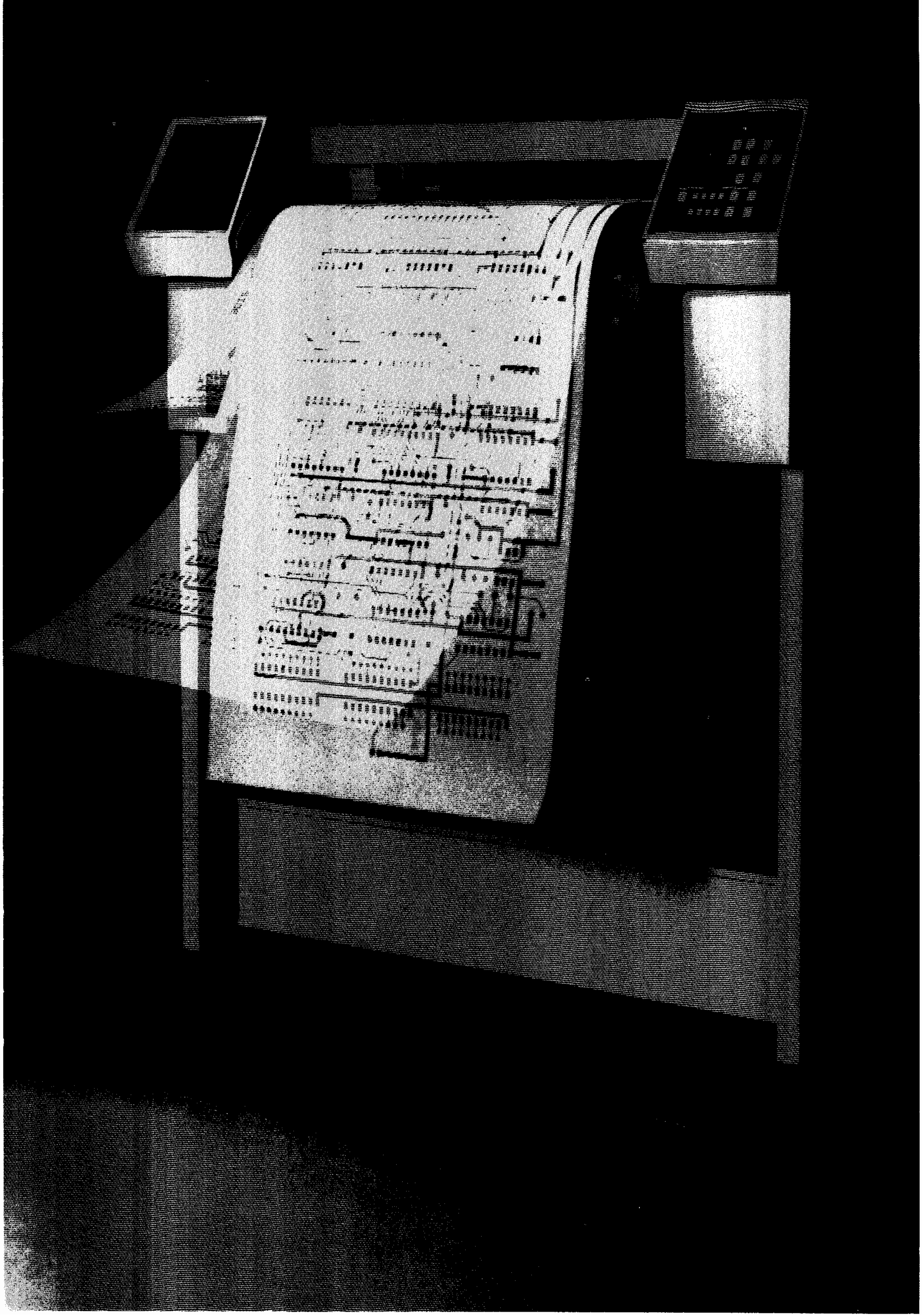
The type field can contain the values −1 through 3 which have the following meaning:

Type Value	Meaning
−1	Carousel of unknown type is installed
0	Carousel is not installed
1	Fiber-tip carousel is installed
2	Roller-ball carousel is installed
3	Drafting pen carousel is installed

Gen. Prog. Inst.

The map value is defined as the sum (0 through 255) of any combination of the following bit values. For example, a map value of 15 (1 + 2 + 4 + 8) indicates pens are installed only in stalls 1, 2, 3, and 4 of the carousel.

Bit Value If Occupied	Pen Stall Number
1	1
2	2
4	3
8	4
16	5
32	6
64	7
128	8

A map value of 0 means either all stalls are empty or no carousel is installed.

Chapter 10
Putting the Instructions to Work

What You'll Learn in This Chapter

In this chapter you'll learn how to put instructions together to develop and label plots. In previous examples, programs have been kept to a less advanced level in order to clearly demonstrate the use of one or two instructions. This example is designed to show how to integrate many instructions into a complete program, how data might be handled, and how subroutines might be used to program a task which would be common to many plots and used in several programs.

This program draws a portion of a subdivision map showing a typical cul-de-sac and the lots on it. Data for this plot came from output of a civil engineering program. Only the back lines have been labeled. The concepts of data plotting and labeling demonstrated here are applicable in drafting applications of all kinds.

For readability, a variety of allowable separators and terminators have been used. In an application where it is important to minimize the number of characters sent over the interface, the spaces between instructions and the semicolons preceding the next mnemonic could and should be omitted.

Problem

Scale, draw, and label a section of a subdivision map according to scale. Label the back lines and centerline of the street. Mark the north direction and draw the scale at the bottom of the plot.

Solution

Positioning and Scaling

The area which can be used for plotting is determined by the hard-clip limits, which are automatically established when paper is loaded and the paper size is sensed. This plot will fit on A2 or C size paper loaded with the long side horizontal, or on A1 and A0 or D and E size paper. With A2 or C size paper, the scaling points P1 and P2 will be off the paper, but all lines will be inside the hard-clip limits.

The data in this program came directly from a civil engineer's computations. The relationship between scaled user units and feet would be found in that program. However, by examining the portion of this program which draws the scale (lines 26 to 33), we can see that 10.16 user units represent 1 foot. The number of plotter units in each user unit, and hence the number of plotter units that represent 1 foot, is determined by the setting of the scaling points. The drawing could be enlarged or reduced according to scale by changing P1 and P2. However, in an application like this that represents physical measurements, it is important to have isotropic scaling (equal in X and Y). With the setting of P1 and P2 given here, one plotter unit in X or Y is 0.55 097 . . . user units.

Set the plotter to default conditions, enter the scaling points, and scale the area using the following instructions:

```
LINE 1.  DF; IP-7837,-10142,7837,10142; SC0,8636,0,11176;
```

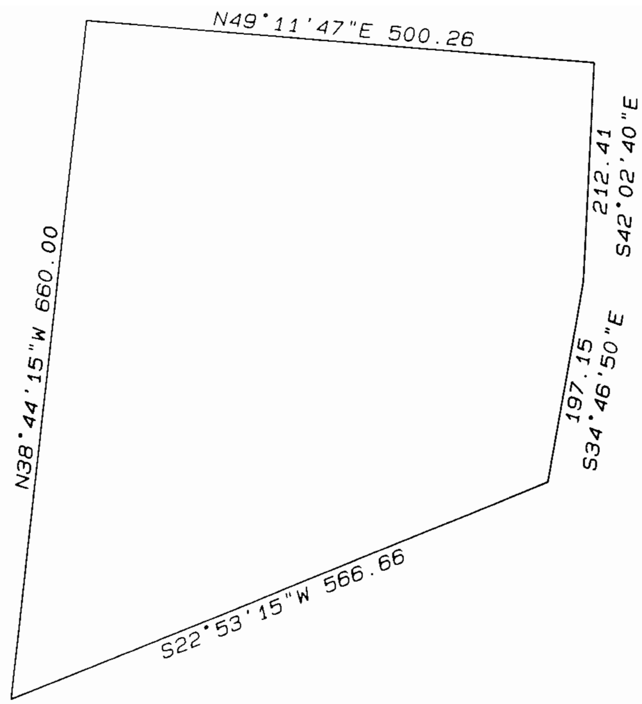
Draw and Label the Subdivision Boundaries

There are two tasks to accomplish. The first is line drawing and the second is labeling. Lines are drawn using PA instructions. Notice how PU and PD are included between X,Y parameter pairs to control the pen status and either draw a line or move to a point. Labels are drawn in an appropriate size, slant, and direction and with a label origin which centers the label at the midpoint of the line, or above or below the line, depending whether the line is at the top, bottom, right, or left of the plot. Lines 2 through 12 are a series of one-line HP-GL instructions to draw the line, followed by one or two lines of instructions to position and draw the labels. The direction parameters used here are a result of calculations interior to the civil engineering program. The slant is approximately 10 degrees. Size is set absolutely by the SI instruction. There are three size instructions in this program to establish the size of large and small labels and to establish the size of the north arrow.

Refer to the following note when sending lines 4, 6, 8, 10, 12, and 21.

NOTE: The use of a language- and computer-dependent function, such as CHR$, may be necessary in a label statement to send ASCII characters not contained on the keyboard or to draw quotation marks. (Quotation marks in a BASIC statement denote the end of a literal string, and hence cannot be sent as ” in a label string.) ■

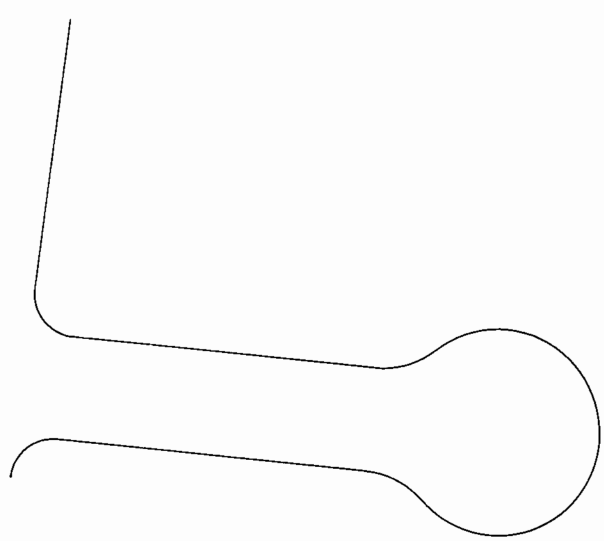
```
LINE 2.   SP1; CS3; PA1016,1016 PD1747,7682 PU1382,4349;
LINE 3.   SI.44,.58; SL.176; DI10.9,99.4; LO14;
LINE 4.   LBN38°44′15"W 660.00␃
LINE 5.   PA1747,7682 PD6816,7310 PU4282,7496;
LINE 6.   DI99.8,-7.3; LBN49°11′47"E 500.26␃
LINE 7.   PA6816,7310 PD6705,5154 PU6761,6232; DI5.15,99.9;CP0,-1;
          LO16;
LINE 8.   LBS42°02′40"E␍␃LO14; LB212.41␃
LINE 9.   PA6705,5154 PD6350,3183 PU6527,4169;DI17.7,98.4;CP0,-1;
          LO16
LINE 10.  LBS34°46′50"E␍␃LO14; LB197.15␃
LINE 11.  PU6350,3183 PD1016,1016 PU3683,2100;
LINE 12.  DI92.6,37.6; LO16; LBS22°53′15"W 566.66␃
```



Draw the Street Boundaries

The street boundaries are made up of arcs and line segments. In lines 15 through 17, four arcs are drawn with the same center because the data from which our plot is drawn defines arcs and lines from each lot boundary. There are actually six arc segments which form the end of the cul-de-sac. Counting the arcs at the beginning of the street, there are eight arcs in all drawn with the arc absolute instruction. The last part of the street is drawn by the subroutine, Moves, because it consists of three pen down moves which are most easily plotted using the subroutine.

```
LINE 13. PU2052,7659 PD1878,6078;AA2131,6050,92.066,5;
LINE 14. PD2112,5797,2872,5741,3910,5665;
LINE 15. AA3948,6172,43.342,5;AA4653,5305,-72.38,5;
LINE 16. AA4653,5305,-60.96,5;AA4653,5305,-60.96,5;
LINE 17. AA4653,5305,-72.38,5;AA3829,4551,43.342,5;
LINE 18. PD3866,5057,2027,5192; AA2009,4939,87.93,5;
```



Draw the Lots

This part of the program has been coded as a subroutine, Moves, to show how data representing many lines might be accessed from stored data. Refer to the DATA statements at the end of the subroutine; you will notice each X,Y point has been stored with an extra parameter P with value 0 or 1, denoting that the move is to be made with the pen up (0) or down (1). Using a computed GOTO statement, the proper mnemonic is inserted and sent to the plotter with the plot instruction and the X,Y coordinates. The P value of 2 in the last DATA statement signals the end of points and causes a return to the main program. Refer to the completed plot at the end of the chapter to see the lot lines.

```
■LINE     CALL Moves


■LINE     SUB Moves
■LINE     READ P
■LINE     ON P+1 GOTO LINE 39, LINE 41, LINE 51
 LINE     PU
■LINE     GOTO LINE 42
 LINE     PD
■LINE     READ X,Y
★LINE     PA;X;Y
■LINE     GOTO LINE 37
■LINE     DATA 1,1756,4966,1,1557,3149,1,1337,1146,0,3648,2085
■LINE     DATA 1,3714,2990,1,3788,4003,1,3866,5057,0,1667,4158
■LINE     DATA 1,3788,4003,0,1557,3149,1,3714,2990,0,5749,2939
■LINE     DATA 1,4909,4752,0,5261,5260,1,6705,5154,0,4987,5815
■LINE     DATA 1,6006,7369,0,3910,5665,1,4046,7513,0,2872,5741
■LINE     DATA 1,3074,7584,0,1512,5535,2
■LINE     SUBEND
```

Draw and Label the Street Centerline and Width

The centerline is drawn in a dashed line type. Use of a negative parameter in the LT instruction causes complete pattern segments to be used to draw the line. The LT ; instruction returns the line type to a solid line upon completion of the centerline. Note the label size is reduced for the three following labels, the direction is changed for each label, and the slant remains unchanged. Refer to the completed plot to see the centerline and labels.

```
LINE 20. LT-6,7.2; PD4653,5305; LT; PU3083,5420;
LINE 21. SI.26,.35 DI99.7,-7.3 LO14 LBN49°11'47"E␍␃
LINE 22. LO16 LB310.00␃ PU1900,7670; DI17.3,99.7; LO12
         LB30.02␃
LINE 23. PU1176,1081; DI-37.643,92.6; LO18 LB34.10␃
```

Draw the North Arrow

This symbol is a natural use of the UC instruction. Once designed in the horizontal position, it can be enlarged or reduced, using SI or SR instructions, and oriented with a DI instruction. This symbol might be included in the library of subroutines maintained by the civil engineer for inclusion in other programs. It is important that no carriage return or line feed

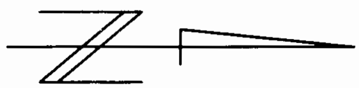
Using Instructions

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

characters be included in the middle of the UC instruction if it is sent by two print lines. If the whole instruction cannot be sent as one line by your computer, any end-of-line characters sent should be suppressed.

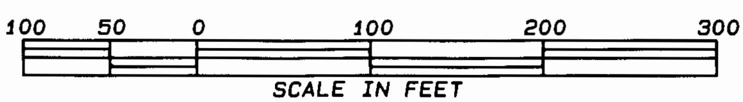
```
 LINE 24. SR 7.6125,5.8; DI 1,1; SL; LO; PU 6096,8636;
■LINE 25. CALL Northarrow

■LINE 52. SUB Northarrow
▲LINE 53. UC 9999,98,0,-49,5,0,-10,-9999,-11,-5,
▲LINE 54. 9999,-29,0,24,20,-9999,-24,0,9999,29,0,-24,-20;
■LINE 55. SUBEND
```



Draw and Label the Scale

The PR instruction is used here to draw lines of equal length and equal height which fit into each 50 or 100 foot section of the scale. Note the use of the NR ; instruction at the end of the program. The pen is stored by this instruction and the plotter placed in View state. The whole plot is visible, and the front-panel **REMOTE** button must be pressed before another plot can be drawn.

```
 LINE 26. DI;SL.176;PU7620,711 PD3556,711 PU3556,610;
 LINE 27. PD7620,610 PU7620,508 PD3556,508 PU SI.26,.35;
 LINE 28. PD3556,711; DI; LO14 LB100␃ PU4064,508 PD4064,711;
 LINE 29. LB50␃ PU4572,508 PD4572,711; LB0␃ PU5588,508
          PD5588,711;
 LINE 30. LB100␃ PU6604,508 PD6604,711; LB200␃ PU7620,508;
 LINE 31. PD7620,711; LB300␃ PU3556,660 PD;
 LINE 32. PR508,0,0,-102,508,0,0,102,1016,0,0,-102,1016,0;
 LINE 33. PR0,102,1016,0,0,-102;
 LINE 34. PUPA5588,508 LO16 LBSCALE IN FEET␃NR;
■LINE 35. END
```



Listing

The complete listing and the results, reduced to 50% of actual size, are shown on the next pages.

Using Instructions

■ BASIC statement. Do not send to the plotter.
▲ Suppress any end-of-line characters such as CR, LF when sending these lines.

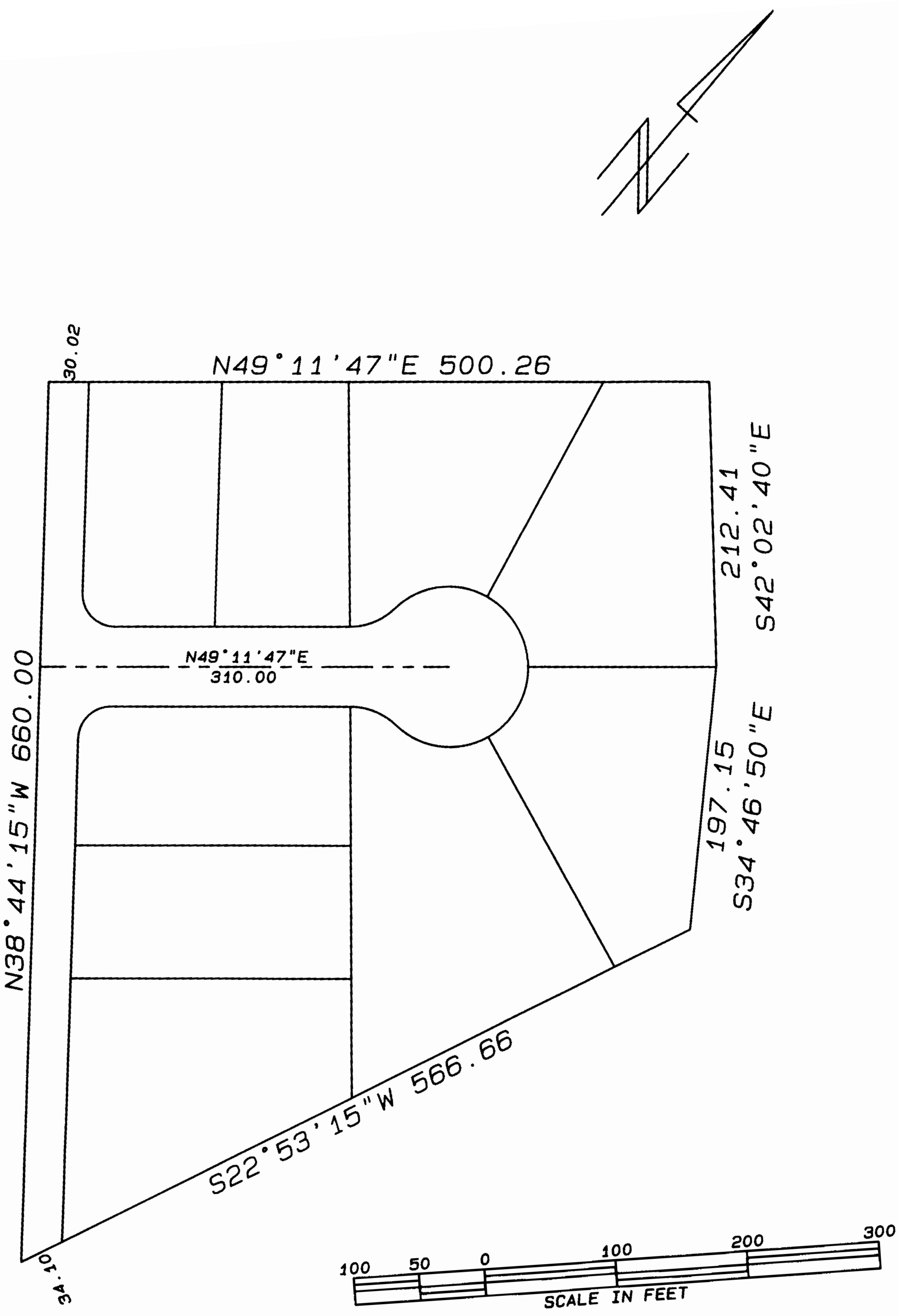
```
 LINE 1.  DF; IP-7837,-10142,7837,10142; SC0,8636,0,11176;
 LINE 2.  SP1; CS3; PA1016,1016 PD1747,7682 PU1382,4349;
 LINE 3.  SI.44,.58; SL.176; DI10.9,99.4; LO14;
 LINE 4.  LBN38°44'15"W 660.00↖
 LINE 5.  PA1747,7682 PD6816,7310 PU4282,7496;
 LINE 6.  DI99.8,-7.3; LBN49°11'47"E 500.26↖
 LINE 7.  PA6816,7310 PD6705,5154 PU6761,6232; DI5.15,99.9;CP0,-1;
          LO16;
 LINE 8.  LBS42°02'40"E↖↖LO14; LB212.41↖
 LINE 9.  PA6705,5154 PD6350,3183 PU6527,4169;DI17.7,98.4;CP0,-1;
          LO16
 LINE 10. LBS34°46'50"E↖↖LO14; LB197.15↖
 LINE 11. PU6350,3183 PD1016,1016 PU3683,2100;
 LINE 12. DI92.6,37.6; LO16; LBS22°53'15"W 566.66↖
 LINE 13. PU2052,7659 PD1878,6078;AA2131,6050,92.066,5;
 LINE 14. PD2112,5797,2872,5741,3910,5665;
 LINE 15. AA3948,6172,43.342,5;AA4653,5305,-72.38,5;
 LINE 16. AA4653,5305,-60.96,5;AA4653,5305,-60.96,5;
 LINE 17. AA4653,5305,-72.38,5;AA3829,4551,43.342,5;
 LINE 18. PD3866,5057,2027,5192; AA2009,4939,87.93,5;
■LINE 19. CALL Moves
 LINE 20. LT-6,7.2; PD4653,5305; LT; PU3083,5420;
 LINE 21. SI.26,.35 DI99.7,-7.3 LO14 LBN49°11'47"E↖↖
 LINE 22. LO16 LB310.00↖PU1900,7670; DI17.3,99.7; LO12
          LB30.02↖
 LINE 23. PU1176,1081; DI-37.643,92.6; LO18 LB34.10↖
 LINE 24. SR 7.6125,5.6; DI 1,1; SL; LO; PU 6096,8636;
■LINE 25. CALL Northarrow
 LINE 26. DI;SL.176;PU7620,711 PD3556,711 PU3556,610;
 LINE 27. PD7620,610 PU7620,508 PD3556,508 PU SI.26,.35;
 LINE 28. PD3556,711; DI; LO14 LB100↖PU4064,508 PD4064,711;
 LINE 29. LB50↖PU4572,508 PD4572,711; LB0↖PU5588,508
          PD5588,711;
 LINE 30. LB100↖PU6604,508 PD6604,711; LB200↖PU7620,508;
 LINE 31. PD7620,711; LB300↖PU3556,660 PD;
 LINE 32. PR508,0,0,-102,508,0,0,102,1016,0,0,-102,1016,0;
 LINE 33. PR0,102,1016,0,0,-102;
 LINE 34. PUPA5588,508 LO16 LBSCALE IN FEET↖NR;
■LINE 35. END
■LINE 36. SUB Moves
■LINE 37. READ P
■LINE 38. ON P+1 GOTO LINE 39, LINE 41, LINE 51
 LINE 39. PU
■LINE 40. GOTO LINE 42
 LINE 41. PD
■LINE 42. READ X,Y
★LINE 43. PA;X;Y
■LINE 44. GOTO LINE 37
■LINE 45. DATA 1,1756,4966,1,1557,3149,1,1337,1146,0,3648,2085
■LINE 46. DATA 1,3714,2990,1,3788,4003,1,3866,5057,0,1667,4158
■LINE 47. DATA 1,3788,4003,0,1557,3149,1,3714,2990,0,5749,2939
■LINE 48. DATA 1,4909,4752,0,5261,5260,1,6705,5154,0,4987,5815
■LINE 49. DATA 1,6006,7369,0,3910,5665,1,4046,7513,0,2872,5741
■LINE 50. DATA 1,3074,7584,0,1512,5535,2
■LINE 51. SUBEND
■LINE 52. SUB Northarrow
▲LINE 53. UC 9999,98,0,-49,5,0,-10,-9999,-11,-5,
▲LINE 54. 9999,-29,0,24,20,-9999,-24,0,9999,29,0,-24,-20;
■LINE 55. SUBEND
```

■ BASIC statement. Do not send to the plotter.

★ A controller-dependent format statement may be required for this statement to be accepted by the plotter.

▲ Suppress any end-of-line characters such as CR, LF when sending these lines.

Using Instructions


N49°11'47"E 500.26
30.02
212.41
S42°02'40"E
N49°11'47"E
310.00
N38°44'15"W 660.00
197.15
S34°46'50"E
S22°53'15"W 566.66
34.10
100
50
0
100
200
300
SCALE IN FEET

Chapter 11
Plotting with Roll Media

What You'll Learn in This Chapter

In this chapter, which pertains only to the HP 7586, you will learn how to load and unload roll media, how to use the plotter's chart advance capabilities, and how to create a long axis plot. In addition, you will learn how to recalibrate the plotter if necessary.

HP-GL Instructions Covered

AF (PG) The Advance Full Page Instruction
AH The Advance Half Page Instruction
FR The Advance Frame Instruction
EC The Enable Cut Line Instruction

Terms You Should Understand

Page — On roll media, the physical plotting area determined by the media's width and the respective automatic advance length. P1 and P2 retain their current values on each new page, just as though a new sheet of paper were loaded into the plotter.

Frame — On roll media, one segment of a long axis plot. Frames are aligned with no margins in between. Each frame contains a new coordinate system relative to the original one.

Using Roll Media

The HP 7586 plotter can accommodate roll media in the same two width ranges shown for sheet media in Chapter 2. However, the HP 7586 is primarily designed to use roll media in either 11-, 24-, or 36-inch widths. These three widths allow generating plots on a physical page size that approximates either ANSI A, B, C, D, or E or ISO A4, A3, A2, A1, or A0 standard sheet sizes. The plotter determines the length of a physical "page" on roll media according to the position of the adjustable pinch wheel. Positioning the adjustable pinch wheel within one of the following width ranges and advancing either a full page or a half page will produce the indicated equivalent sheet size. Refer to the following table and illustration.

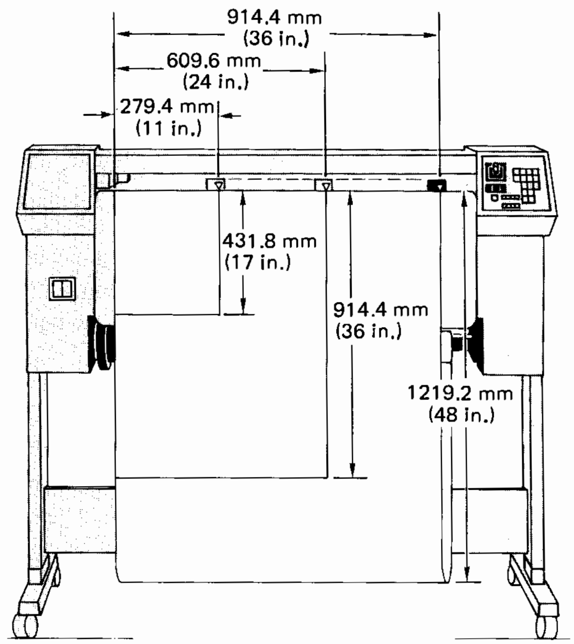
HP 7586 Acceptable Roll Media Dimensions

Width Range	Full Page Advance Length*	Half Page Advance Length**	Equivalent Sheet Size
266.7–298.5 mm (10.5–11.75 in.)	—	215.9 mm (8.5 in.)	A4 (A)
	431.8 mm (17 in.)	—	A3 (B)
546.7–718.8 mm (21.5–28.3 in.)	—	457.2 mm (18 in.)	A2 (C)
	914.4 mm (36 in.)	—	A1 (D)
721.4–927.1 mm (28.4–36.5 in.)	—	609.6 mm (24 in.)	A1 (D)
	1219.2 mm (48 in.)	—	A0 (E)

*Advanced with the AF, PG, and FR instructions. Also advanced by pressing **NEXT PAGE** when the plotter is in the Remote or View state.

Advanced with the AH instruction. Also advanced by pressing **ENTER and then **NEXT PAGE** when the plotter is in the Remote or View state.

Page-Advance Lengths for Roll Media. *The medium advances 17, 36, or 48 inches, depending on the location of the adjustable pinch wheel.*



The HP 7586 has two roll-media operation modes:

- continuous feed with automatic take-up on the front spool as each plot is completed.
- continuous feed with no front spool installed. Each plot must be removed as it is completed.

The plotter determines its operating mode during the medium-sensing sequence that is initiated when you press one of the following front-panel buttons while the **CHART HOLD** indicator lamp is on: **REMOTE, VIEW, P1, P2,** or **AXIS ALIGN.** During this sequence, the plotter checks for the physical presence of the front and rear spools and searches for leading and trailing ends of the medium. The take-up feature is automatically disabled if the front spool is not installed.

NOTE: Two identical spools and an empty cardboard core are supplied with the HP 7586 plotter. You can use either spool for the rear supply spool or the front take-up spool. All HP roll media has a core that is identical to the empty core supplied with the plotter. You can also use these cores on the front take-up spool. ■

Loading Roll Media

The following procedure describes how to load roll media with automatic take-up. If you intend to cut off each plot as it is completed, disregard the steps for installing the front take-up spool.

WARNING

To prevent injury, keep hair, fingers, and clothing from becoming entangled in the paper moving mechanism.

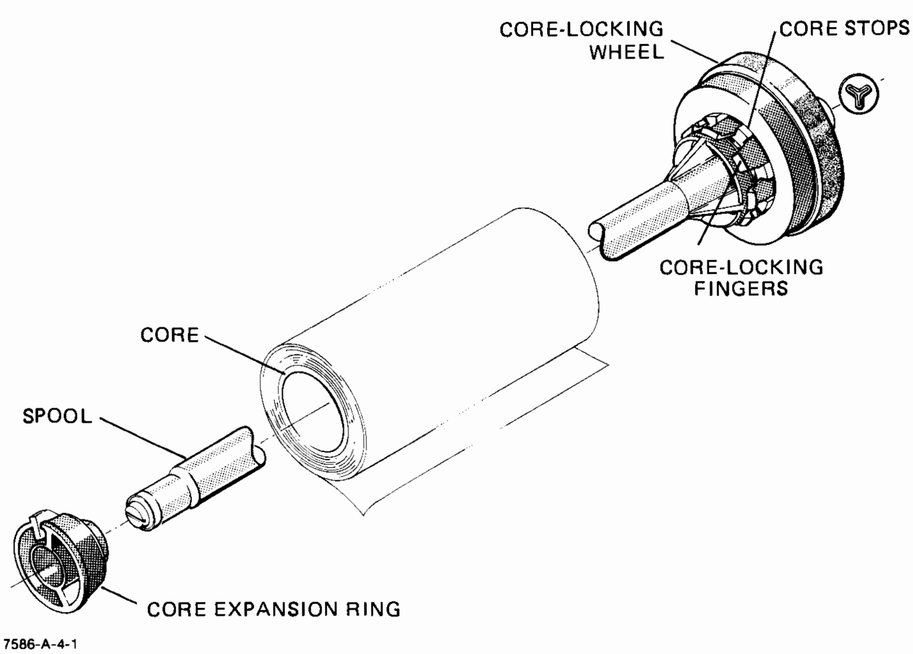
1. Mount a roll of medium on one spool and an empty core on the second spool. Be sure that the medium pulls off of the roll as shown in the following illustration. The procedure for both operations is as follows:

   a. Loosen the core-locking wheel to retract the core-locking fingers.

   b. Slide the spool into the core until the core stops are flush against the end of the core.

   c. Compress the core expansion ring and slide it into the other end of the core.

   d. Tighten the core-locking wheel to expand the core-locking fingers against the inside of the core. *Be sure that the core remains flush against the core stops.*



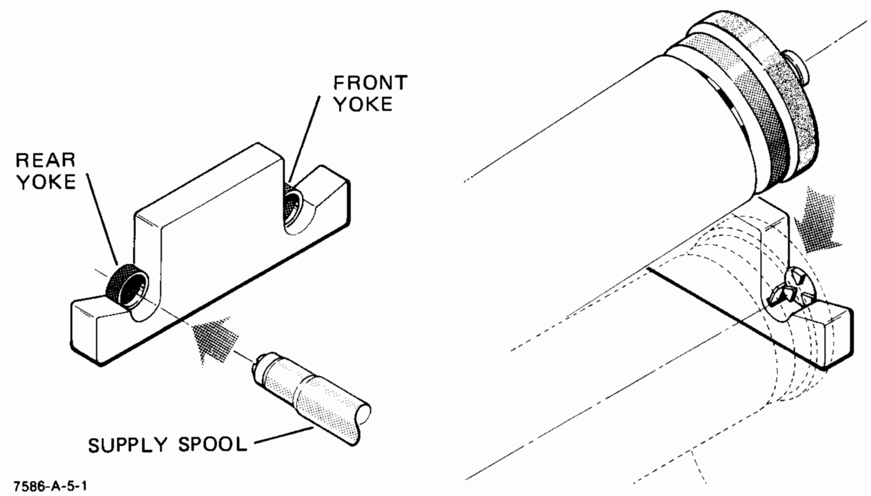
Loading Roll Medium on the Supply Spool

2. Install the supply spool in the rear yoke and the take-up spool in the front yoke. Refer to the following illustration for the proper orientation of each spool. The procedure for installing either spool is as follows:

    a. With a push-in motion, spring-load the slotted end of the supply roll into the left yoke at the rear of the plotter.

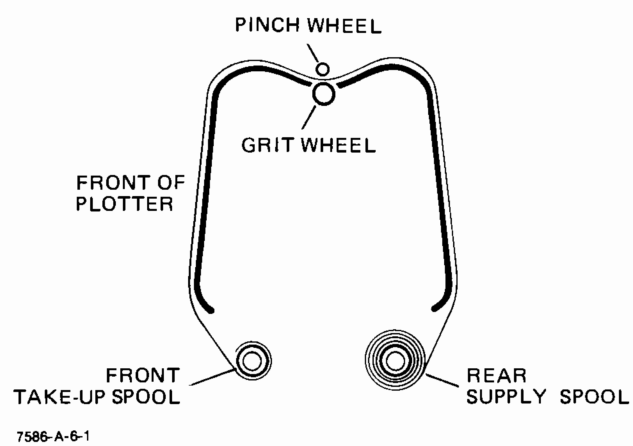
    b. Then snap the opposite end into place in the right yoke. Rotate the spool as necessary to make sure that the raised "Y" or cloverleaf is properly aligned and engaged with the shaft.

Installing Spools in the Front and Rear Yokes



3. Raise the carriage cover and thread the medium from the supply spool between the grit wheels and the pinch wheels. Refer to the following illustrations.

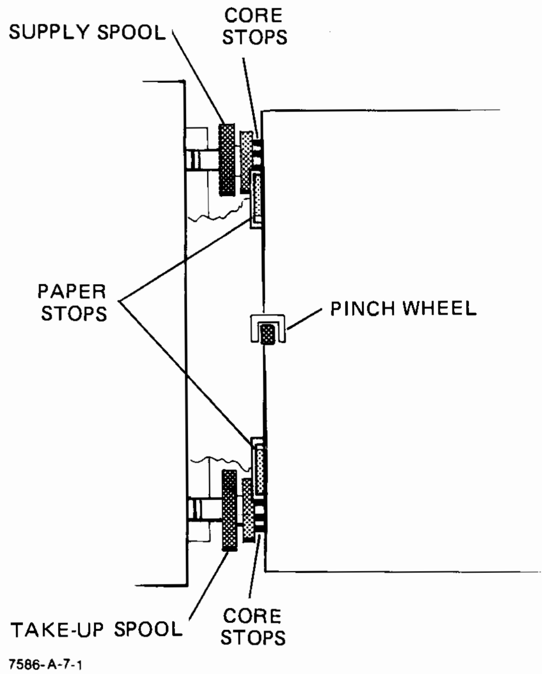
Threading Roll Medium



The following conditions must be achieved simultaneously to ensure successful roll medium operation with automatic take-up.

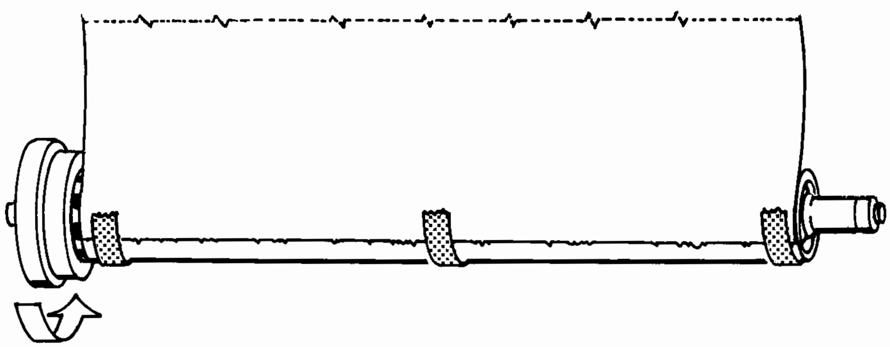
- All layers of the medium on the supply spool must be aligned to form a square roll edge against the core stops of the supply spool.
- The left edge of the medium must be positioned against both platen paper stops.

- The end of the medium must be taped to the take-up core so that the left edge of the medium is positioned against the core stops on the take-up spool, as indicated in the following illustration.



Roll Medium Alignment. *The medium edge must be flush against the core stops and the paper stops.*

4. Attach a piece of masking tape to the center and to each end of the medium. Pull the medium down to the take-up spool and tape it securely to the empty core. *Be sure the medium spreads evenly along the core so that it will not buckle.* Manually rotate the take-up spool two or three complete turns to be certain that the medium will spool evenly. Refer to the following illustration.



Attachment of Medium to the Take-up Core

5. Manually slide the adjustable pinch wheel to align the white scribe mark with the right edge of the medium.

6. Close the carriage cover, turn on the power, and press **CHART HOLD** to lower the pinch wheels and initiate the medium-sensing sequence. The pinch wheels will remain down only if the adjustable pinch wheel is within one of the ranges previously indicated in the table for HP 7586 Acceptable Sheet Media Dimensions. Note also that the HP 7586 is inoperative with the carriage cover raised.

7. Press **REMOTE** to complete the medium-sensing sequence. Pressing **REMOTE** will cause a two-foot leader to be pulled off of the supply spool and wound tightly onto the take-up spool (if attached). The proper physical page length is then pulled off of the supply spool. To determine its operating mode, the plotter searches for both a front and rear end of the medium. The following chart is a simple truth table that indicates the operating mode that is enabled when the plotter detects the indicated conditions.

HP 7586 Operating Mode Truth Table

Spools Detected		Medium Edge Detected		
Front	**Rear**	**Front**	**Rear**	**Operating Mode Enabled**
Yes	Yes	Yes	Yes	Single sheet
Yes	Yes	No	Yes	Error: Out of paper
Yes	Yes	No	No	Continuous feed, take-up enabled
No	Yes	Yes	No	Continuous feed, take-up disabled
No	Yes	No	No	Continuous feed, take-up disabled
No	No	Yes	Yes	Single sheet

Individual plots can be cut off as they are completed even if the medium was initially attached to the take-up spool. However, if the medium must be reattached to the take-up spool, you must first perform step 2 of the following unloading sequence and then retape the medium to the front spool. It should be noted that four feet of medium are wasted each time it is reattached to the take-up spool.

Unloading Roll Media

1. Advance the medium, using the **NEXT PAGE** button, and cut off the most recent plot.
2. Press **VIEW** and then **CHART UNLOAD** to raise the pinch wheels and release the medium.

   NOTE: When the **CHART UNLOAD** or **CHART HOLD** indicator lamp is on, the **NEXT PAGE** button can be held down to wind the loose medium onto the take-up spool. The take-up spool motor will turn off automatically if the **NEXT PAGE** button is held down longer than 10 seconds. ■

3. Push the take-up spool to the right and lift it from the yoke assembly.
4. Loosen the core-locking wheel and remove the roll of completed plots from the take-up spool.

HP-GL Instructions for Use with Roll Media

The HP 7586 plotter has four HP-GL instructions for use with roll media only. These instructions are ignored on the HP 7580 and 7585. When used with sheet media on the HP 7586, all except the EC instruction will generate error 8.

The Enable Cut Line Instruction, EC

USES: The EC instruction draws a cut line between plots after an AF, PG, or AH instruction. Use the instruction to indicate where to cut the paper.

SYNTAX: *EC* (n) *term*

PARAMETERS: The parameter n can be any number and is optional.

EXPLANATION: Use of this instruction without parameters (EC ;) instructs the plotter to draw a cut line. The cut line will be drawn after each AF, PG, or AH instruction until the instruction is disabled.

Use of the instruction with a parameter disables the drawing function. This overrides any previously issued EC ; instruction.

The Advance Full Page Instruction, AF

USES: The AF instruction advances the paper one full page length. Use this instruction to advance the roll for the next plot.

SYNTAX: *AF* *term*
or
PG *term*
or
PG n *term*

PARAMETERS: No parameters are used with AF. Any parameter used with PG is ignored.

EXPLANATION: The AF instruction advances the paper one full page length and establishes the origin at the center of the new physical page. The page advance length is determined when the paper's width is sensed by the plotter in **CHART HOLD.** Paper which is 11, 24, or 36 inches wide will have a respective page-advance length of 17, 36, or 48 inches (see Using Roll Media in this chapter).

The current value of P1 and P2 and the current plot rotation remain in effect.

The AF instruction performs differently according to whether the automatic take-up is enabled or disabled. Following is a description of the respective functions.

1. *Automatic take-up enabled* (front take-up spool attached):
   a. If the cut line function has been enabled with the EC instruction, the plotter will draw a line along the border between the plots.
   b. The finished plot will be wound onto the take-up roll as the next plotting area is pulled off the rear supply roll. Two complete forward-backward passes are made (between the X-direction hard-clip limits) to develop grit impression. The paper is now in the view position and the plotter is ready to plot.

NOTE: This sequence is also performed after pressing **NEXT PAGE** (or pressing **ENTER** and **NEXT PAGE** for a half-page advance) on the plotter's front panel when the plotter is in the View or Remote operating state. ■

2. *Automatic take-up disabled* (front take-up spool not attached):

   a. If the cut line function has been enabled with the EC instruction, the plotter will draw a line along the border between the plots.

   b. In any case, this border now lines up with the crease in the plotter's front skirt (the cut position) to facilitate cutting.

   c. The **REMOTE** indicator lamp flashes, signaling you to cut off the plot. All other front-panel indicator lamps will be off, and all front-panel buttons except **REMOTE** and **CHART UNLOAD** are inactivated. (The joy-stick remains operative in the X-direction in case it is necessary to improve the alignment between the cut position and the crease in the front skirt.)

   d. Press **REMOTE** to restore the front-panel indicator lamps and complete the page-advance sequence. The paper is now in the view position and the plotter is ready to plot.

   or

   Press **CHART UNLOAD** to restore the front-panel indicator lamps, abort the page-advance sequence, lift the pinch wheels, and place the plotter in the Not-Ready state.

   NOTE: This sequence is also performed after pressing **NEXT PAGE** (or pressing **ENTER** and **NEXT PAGE** for a half-page advance) on the plotter's front panel. The only difference is that if the front-panel functions are executed in the View state instead of the Remote state, the **VIEW** indicator lamp will flash during the waiting period instead of the **REMOTE** indicator lamp, and the **VIEW** button is pushed to complete the page-advance sequence. ■

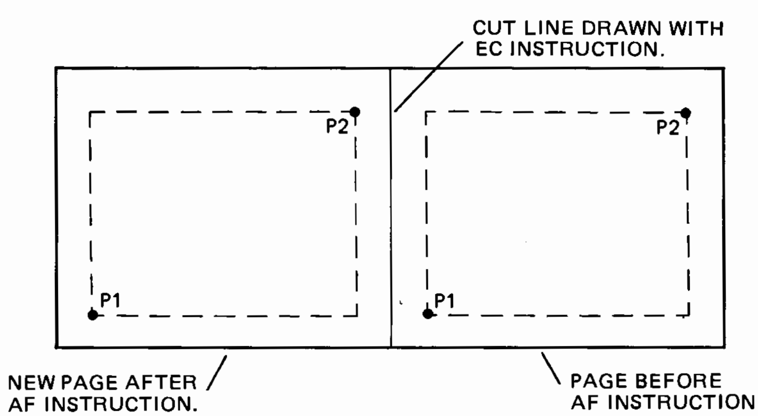
Execution of this instruction sets the bits of the ESC . O instruction to the proper state.

Execution of this instruction with sheet media loaded on the HP 7586 generates error 8.

NOTE: If you desire a new reading of certain rear-panel switches after advancing the page, the plotter must be taken out of the Remote state and put into the Not-Ready state (either the **CHART HOLD** or **CHART UNLOAD** indicator lamp will be on). The **EXPAND/NORMAL** switch and *all other switches in the top block* are read only when the plotter is in the Not-Ready state. ■

The following diagram shows the location of P1 and P2 after an AF instruction.

P1, P2 Location Following an AF Instruction. *P1 and P2 retain the same location on the new page.*



The Advance Half Page Instruction, AH

USES: The AH instruction advances the paper one half-page length.

SYNTAX: *AH term*

PARAMETERS: None.

EXPLANATION: The AH instruction advances the paper one half-page length. Paper which is 11, 24, or 36 inches wide will have a respective half-page advance length of 8.5, 18, or 24 inches (see Using Roll Media in this chapter).

Otherwise, this instruction performs identically to the Advance Full Page Instruction, AF. Refer to the AF instruction for a description of plotter performance with the automatic paper take-up either enabled or disabled.

Execution of this instruction with sheet media loaded on an HP 7586 generates error 8.

The Advance Frame Instruction, FR

USES: The FR instruction advances the paper to the next plot frame and calculates a relative coordinate system for that frame. Use the instruction to perform multi-frame long axis plotting.

SYNTAX: *FR term*

PARAMETERS: None.

EXPLANATION: The FR instruction performs a long axis frame advance and alignment. The plotter draws registration marks on the present frame, digitizes the marks, advances the paper, and digitizes the registration marks again. In this way it calculates a new coordinate system and aligns that system to the old one.

The values of P1 and P2 change by the advance length of the frame in order to keep the same relative location on the paper. The length advanced with an FR instruction is shorter than that of an AF instruction; the margins between the plotting areas are deleted so that the frames share a common edge.

The FR instruction will align a full-size plot area (the area advanced with a PG or AF instruction), to the present plot area, which can be either full or half size. When executed, FR clears the polygon buffer.

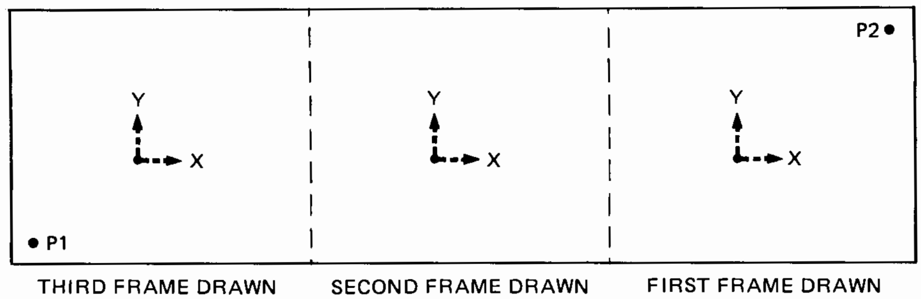
Execution of this instruction with sheet media loaded on an HP 7586 generates error 8.

The following section, Long Axis Plotting, presents the use of the FR instruction in more detail.

An Introduction to Long Axis Plotting

On the HP 7586, you can use the Advance Frame Instruction, FR, to align adjacently drawn frames to form the equivalent of a long axis plot. The diagram on the next page shows a typical location of P1 and P2 in a long axis plot.

Frame Sequence of a Long Axis Plot (Nonrotated)



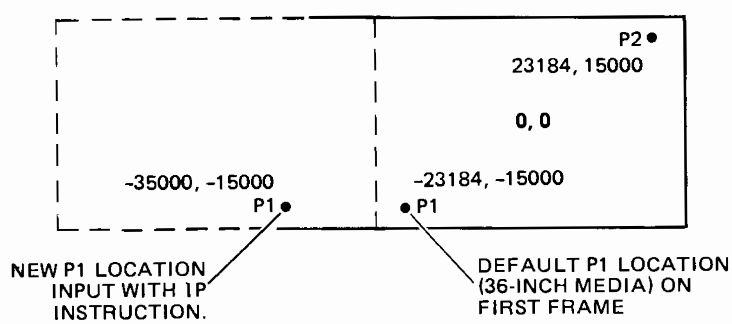
Use the IP instruction to change the X-coordinate of P1 to a large negative value. The frame advance normally occurs in the negative X-direction, as pictured above. Therefore the right-most frame is the first frame drawn. When the plot is rotated, frame advance will occur in the positive Y-direction and the bottom-most frame is the first frame drawn.

The plotter treats each frame as a separate window and plots only the data which falls within that frame. During the initial program sequence, the plotter draws only the plot data in the rightmost frame. After an execution of the FR instruction, the paper is advanced, all data is sent again, and the next frame is drawn.

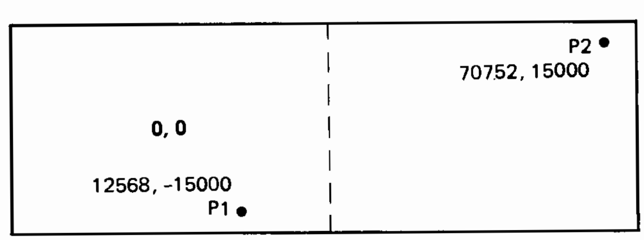
It is suggested that you use scaling for long axis plotting. Scaling is performed either in the plotter with the SC instruction, or in the controller, which uses P1, P2 values obtained from an OP instruction. If you define user units in the plotter with an SC instruction, it is not necessary to redefine coordinates after an execution of an FR instruction. When scaling is performed in the controller, you must use an OP instruction after each FR instruction in order to obtain the P1, P2 location with respect to the new coordinate origin.

With each frame advance, the plotter-unit origin relocates to the center of the current frame. The plotter adjusts P1 and P2 values to reflect this change. The *physical* location of P1 and P2 are retained, but the *logical* location relative to the current origin changes. You can see this in the following illustration.

Input P1, P2 Coordinates. *The plotter-unit origin is on the first frame drawn.*



New Coordinate System Established After FR. *The plotter-unit origin relocates on the next frame.*



The P1, P2 values are input as IP–35 000, –15 000, 23 184, 15 000. (The next section explains how to calculate a P1 X-coordinate and how to determine the number of frames your plot will require.)

After execution of FR, the logical values of P1 and P2 on the new frame become 12 568, –15 000, 70 752, 15 000. Note that the span of the X-axis between P1 and P2 is 58 184 plotter units in each case. The plotter performs these computations internally so that, when scaling is in effect, the currently scaled window is mapped onto the current frame.

The plotter computes these values by drawing registration marks (directly on the X-boundary and 7.5 mm within the Y-boundary) to indicate the hard-clip limits of each frame. After a frame advance, the plotter digitizes the marks in order to properly align the next frame. These marks are drawn by a black pen which must be installed in pen stall 8. If pen stall 8 is empty, no registration marks are drawn, and the proper frame alignment cannot be guaranteed.

The plotting area available after an FR instruction is the same as after a PG instruction, but the actual advance length of an FR instruction is shorter. The margins between plotting areas are deleted so that the frames share a common edge.

For optimum results, plot within the default P1, P2 location. This insures a minimal 30-mm margin between the paper's edge and plotting area. Failure to maintain a 30-mm margin may cause the following:

- Plotting lines and registration marks are too close together for proper digitization of the marks, resulting in possible inaccuracy.
- The pinch wheels may track over the plotting area. Inherent rotation in the plotter's coordinate axis relative to the paper's edges can cause subsequent frames to wander off the paper's surface. The longer the plot, the greater the margin that you must have.

X-Axis Hard-Clip Limits

The plotter determines the advance length of a physical page according to the width of the paper loaded. The plotter also uses the width of the paper to automatically determine the discreet (in that they are not visibly apparent on roll media) X-axis hard-clip limits for one frame. Knowing these limits makes P1 and P2 predictable; you can calculate the length of a frame and thus determine the number of frames to set up in your plot. This feature is especially useful in a spooled environment when you cannot use an OP instruction to interrogate the plotter for hard-clip limit values.

The hard-clip limits for the Y-axis will vary slightly according to the exact location of the pinch wheel. The X-axis hard-clip limits are fixed at default. The following table lists the default settings for one frame, based on paper width. P1 and P2 remain 15 mm (600 plotter units) inside of the hard-clip limits.

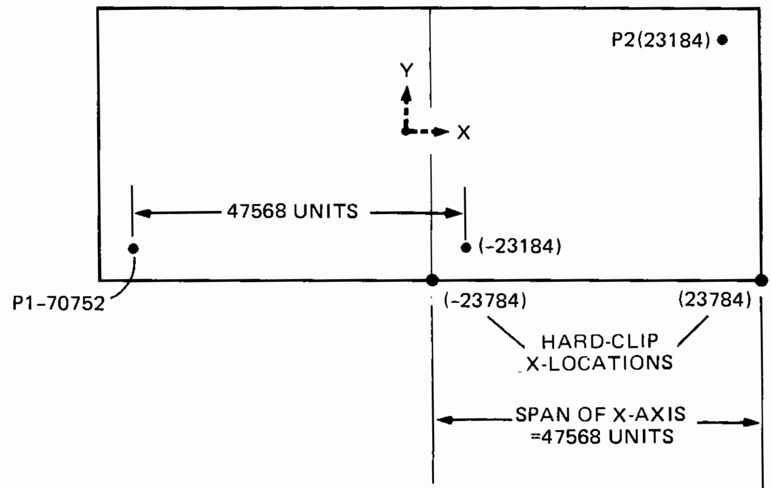
Roll Media Width	Default Hard-Clip X-Coordinates		Default P1, P2 X-Coordinates	
	X_{min}	X_{max}	X_{min}	X_{max}
914.4 mm/36 in.	–23 784	23 784	–23 184	23 184
609.6 mm/24 in.	–17 688	17 688	–17 088	17 088
279.4 mm/11 in.	– 8036	8036	– 7436	7436

Locating a P1 X-Coordinate on a Long Axis Plot

One simple formula for calculation of a P1 location is to multiply the number of frames added to the original frame times the difference in the X-axis hard-clip limits for one frame. (This gives you the hard-clip span of the X-axis.) Then subtract this amount from the default P1 location for the first frame. You can use the preceding table to determine the default P1 location.

Following are calculations for a two-frame plot on E-size paper. Only the X-coordinate locations are shown in the diagram.

X-Coordinate Locations on a Long Axis Plot



$P1 = \text{default P1 on first frame} - [\#\ \text{additional frames} \times (X_{max} - X_{min})]$

$P1 = -23184 - [1 \times (23784 - (-23784))]$

$P1 = -23184 - [1 \times (47568)]$

$P1 = -23184 - [47568]$

$P1 = -70752$

You can see how this formula applies in the following program.

Example — Creating a Long Axis Plot

On the next page is a simple example of a three-view mechanical drawing plotted on E-size paper. This program uses a subroutine to resend the block of data after the frame advance.

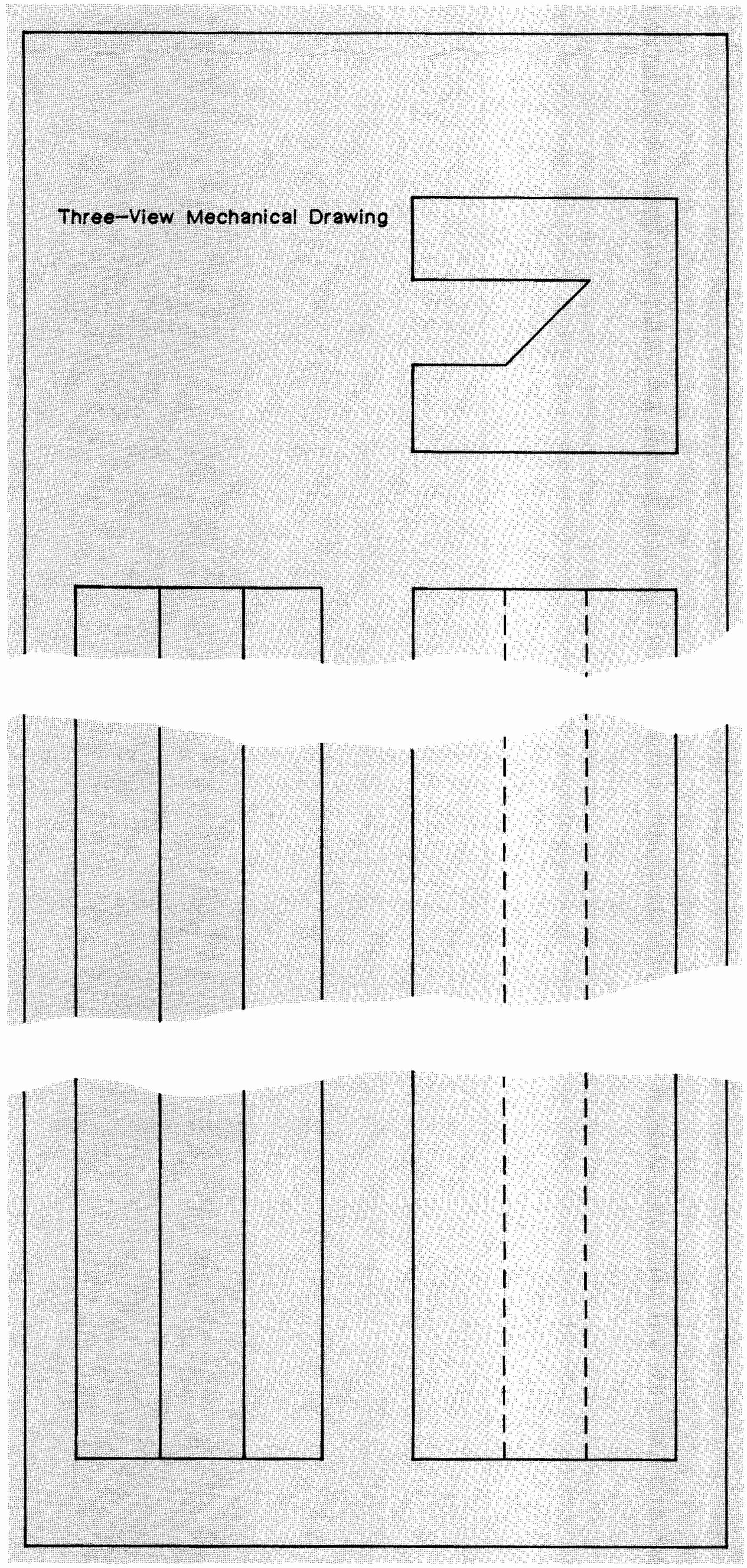
Send:

```
 LINE 1.    IN;IP-70752,-16899,23184,16899;
 LINE 2.    SC0,2789,0,1000;LT;LO13;DI0,1;
            SI0.855,1.125;

■LINE 3.    GOSUB 7
 LINE 4.    FR;
■LINE 5.    GOSUB 7
■LINE 6.    END

 LINE 7.    SP1;PU;PA0,0;PD2780,0,2780,1000,0,1000,0,0;
 LINE 8.    SP2;PU2502,50;PD914,50,914,408,2502,408,
            2502,50;
 LINE 9.    PU2502,169;PD914,169;PU914,289;PD2502,289;
 LINE 10.   PU2502,592;PD914,592,914,950,2502,950,
            2502,592;
 LINE 11.   PU2502,711;LT3,1;PD914,711;PU914,831;
            PD2502,831;LT;
 LINE 12.   PU636,592;PD517,592,517,711,397,831,397,
            592,278,592;
 LINE 13.   PD278,950,636,950,636,592;
 LINE 14.   PU278,50;LBThree-View Mechanical Drawing␃;
■LINE 15.   RETURN
```

Line 1	initializes the plotter; positions P1 and P2. The coordinates for the X-value of P1 are the same as in the preceding section
Line 2	scales the axes into user units; selects the default line type; selects offset label origin 13; selects vertical labeling; selects size of labeling character in centimetres
Line 3	calls block of data for first frame
Lines 7-15	send the entire block of data for frame one. The plotter draws the part of the block located in the first frame
Line 4	draws registration marks on the current frame, digitizes the marks, advances the paper, and digitizes the marks again to set up a new coordinate system
Line 5	calls block of data for the second frame
Lines 7-15	send the entire block of data again to draw frame two

■ BASIC statement. Do not send to plotter.

Reproduction of a Long Axis Plot. *This is a simulated drawing of the plot created by the example on the preceding page.*



Media Absorbency

On the HP 7586 plotter, liquid ink on vellum or polyester film will not dry fast enough to allow immediate spooling on the take-up roll after a plot is finished. In order to have sufficient drying time, the plotter must be programmed with a delay before a page-advance instruction. The time required will vary according to relative temperature and humidity. At 50% relative humidity, 5 to 10 minutes drying time is sufficient with HP-recommended media and ink.

Stabilization of Chart Paper, Vellum, and Tracing Bond

All non-plastic media (chart paper, vellum, and tracing bond) is unstable, and will expand or contract when exposed to changes in relative humidity. Any change in relative humidity (R.H.) from 0 to 100% can result in a change of up to ±1% in media size.

To ensure maximum plotting accuracy and repeatability, sheet and roll paper should be exposed to the air and permitted to stablize before plotting.

Inner layers of roll paper will not stabilize until fully exposed to the air. Therefore, roll paper cannot stabilize properly if it is unwound from the supply roll and plotted on immediatly. It will expand or contract during plotting, resulting in an inaccurate plot.

To allow stabilization of roll paper, the following programming practice is recommended before each single-frame plot. This technique does *not* work with the FR instruction on long-axis plots.

1. Advance the page with a PG or AF instruction. This unwinds the paper from the supply roll and exposes it to the air as it hangs in queue in the service loop.
2. Program a five-minute wait so that the paper can stablilize.
3. Program two forward-backward passes of the paper to develop grit track to improve registration.
4. Send the data for the plot.

The first three steps are illustrated in the following programming example.

Send:

```
LINE 1.     PG;
LINE 2.     WAIT 300
LINE 3.     VS45;SP0;PD
LINE 4.     PA Xmax,0;PA Xmin,0
LINE 5.     PA Xmax,0;PA Xmin,0
LINE 6.     VS;
```

The WAIT instruction must conform to the programming language of the computer.

A pen velocity of 45 is used because it is considered to be the best general velocity for developing grit tracts.

Recalibration of the HP 7586 Plotter

The HP 7586 plotter contains an EAROM (electrically alterable read-only-memory), where certain plotter-specific numbers are stored. These numbers are "calibration constants" which summarize the physical characteristics of the plotter, media, temperature, and humidity when the plotter was calibrated at the factory. The constants are actually adjustment or scale values, and do not represent specific measurements such as temperature. An internal firmware program of the machine adjusts actual drawing characteristics of the plotter to compensate for the various curvatures and non-linearities introduced by the plotter and its environments.

The EAROM and an optical sensor, capable of detecting black ink lines on either paper or polyester film, make recalibration possible. The sensor is located in the pen carriage.

When to Consider Recalibration

Recalibration is not necessary unless the plotter is found to be measurably out of tolerance. This determination requires an accurate distance measuring device, and specific factory-generated calibration drawings. Recalibration of the plotter takes about one half-hour, and should not be required on a regular basis.

Recalibration may be desirable when the following conditions occur.

1. *The physical hardware has changed (the pinch rollers, for example).*
2. *The plotter has been subjected to physical stress.* If the machine has been physically stressed, it is possible that the resultant drawings are not as accurate as before the stress. Under these conditions, it may be desirable to recalibrate.
3. *The plotter's new physical environment differs considerably from the calibration environment at the factory.* The original calibration was in a specific drawing environment. Since your current conditions may differ considerably, you may wish to repeat the calibration procedure with your own media to ensure the most accurate operation.

Overview of the Calibration Procedure

The plotter makes two separate drawings which are illustrated on a following page. After the first set of patterns is drawn, sheet one is rotated 180 degrees and reloaded into the machine. Location of the crosses determines one type of distortion inherent in the physical mechanism.

The second sheet has a similar pattern of crosses. After sheet two is drawn, it is cut along the bottom to make it square. Sheet two is reloaded into the machine after being rotated by 90 degrees to the left, or counter-clockwise. The plotter measures a different set of distortions on this sheet by locating the crosses with the optical sensor.

Calibration with Different Media

The plotter was calibrated at the factory using E-size polyester film with a black drafting pen located in pen stall 8. Recalibration using paper and a black fiber-tip pen is possible, but will result in different calibration constants stored in the EAROM. Whether or not you detect differences when you calibrate the plotter with paper or with polyester film depends upon which medium you use to produce your drawings, and also upon the particular measuring instruments external to the machine.

Calibration of the plotter using smaller than E-size (A0) media results in greater variation in the values obtained and is not recommended. If you calibrate the plotter with other than E-size polyester film and a black drafting pen, it may not perform to specifications.

How to Do the Calibration Procedure

The plotter's front panel will display codes which enable you to monitor the recalibration process. A table containing the description of each code appears at the end of this section. In the following instructions, the code displayed during each phase of the procedure appears in brackets [ ].

1. Turn the machine on. Load a black-ink 0.35-mm drafting pen into pen stall 8. Make sure that the pen is operable and not clogged. Recommendations:

   Let the machine run long enough to come to approximate thermal equilibrium, perhaps 20 to 30 minutes.

   Lay two sheets of E-size polyester film flat upon a nearby table in order to "soak" it in the environment of the plotter for 20 to 30 minutes. Turn it once. This will stabilize its temperature and water absorption characteristics. Media dimensions are sensitive to humidity, temperature and handling.

2. Carefully line up the center of the media with the paper stops on the left side of the plotter. Press the CHART HOLD button on the front panel.

3. Set the rear-panel CALIBRATE/NORMAL switch to the CALIBRATE position and press the CONFIDENCE TEST button on the rear panel.

   The plotter will enter the calibrate mode [C00]. It proceeds to digitize the "primary distance standard" marks inset into the platen [C01]. The actual physical distance between these marks must have been previously stored in the EAROM. Failure to locate these marks will cause the calibrate mode to abort, and the plotter will return to the Not-Ready power-up state.

   Pen 8 will draw a specialized pattern of 21 carefully placed crosses on sheet one [C12]. If the plotter fails to find pen 8 [C88], it will return to the Not-Ready power-up state.

4. When sheet one is completed, the **CHART UNLOAD** light will flash [C03]. Press **CHART UNLOAD** and remove sheet one.

5. Rotate sheet one by 180 degrees and reload it into the plotter [C07]. *Do not turn the sheet over;* it should remain with the drawn side up. Press the flashing **CHART HOLD** button.

   The plotter now locates the crosses on sheet one with its optical sensor [C02].

   If the plotter fails to locate any of the 21 crosses, the lights on front-panel buttons **PEN SPEED** and **PEN FORCE** flash alternately [C10]. In this case:

   Press either **PEN SPEED** or **PEN UP** to redraw sheet one

   or

   Press either **PEN FORCE** or **PEN DOWN** to return to the power-up state without changing the EAROM.

6. After the crosses are all located on sheet one, the **CHART UNLOAD** light will flash [C03].

7. Unload sheet one and set it aside. Load a fresh, blank, E-size sheet into the plotter and press the flashing **CHART HOLD** button [C05].

   The plotter now draws a second set of 12 calibration crosses on sheet two [C11]. If pen 8 is not present [C88], the procedure ends, and the plotter returns to the power-up state.

   When sheet two is finished, the plotter draws a CUT LINE along the bottom of the sheet.

8. Cut the media along the cut line and press **CHART UNLOAD** to remove sheet two [C03]. Make sure that the cut is outside of the line drawn (see illustration). Do not cut into the large square sheet which remains.

9. Rotate sheet two by 90 degrees *counterclockwise*. Reload it into the plotter and press the flashing **CHART HOLD** button [C06].

   The plotter now locates the crosses on sheet two [C13]. Upon successful completion of sheet two, the **PEN SPEED** and **PEN FORCE** lights will flash [99].

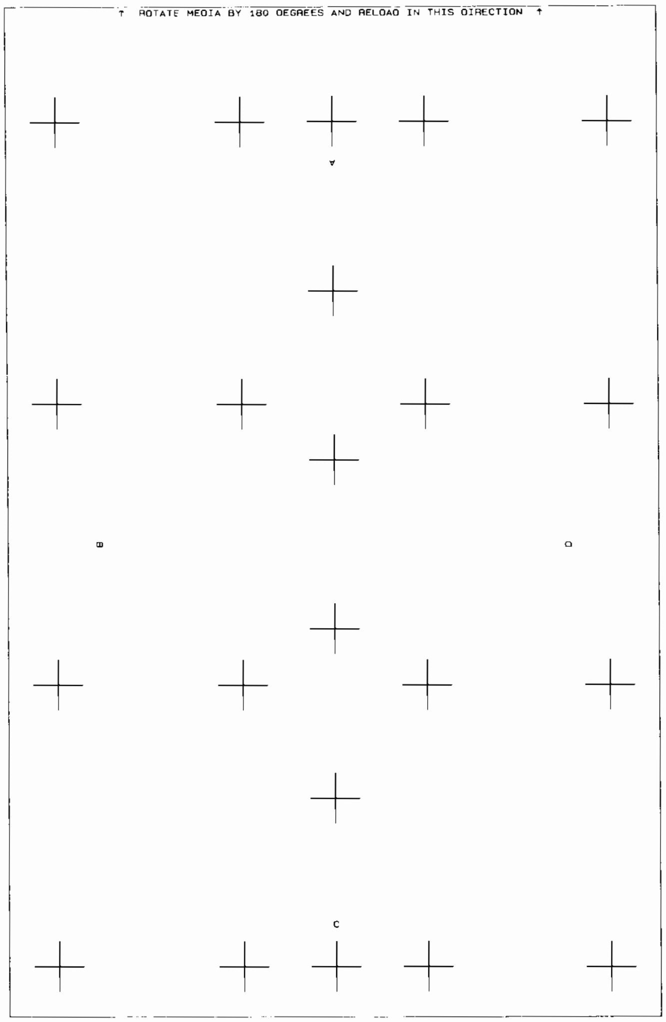
10. Press **PEN SPEED** or **PEN UP** to store new constants in the EAROM. The plotter will write data into the EAROM [C14] and display AEC, Acceptable End of Calibration.

    or

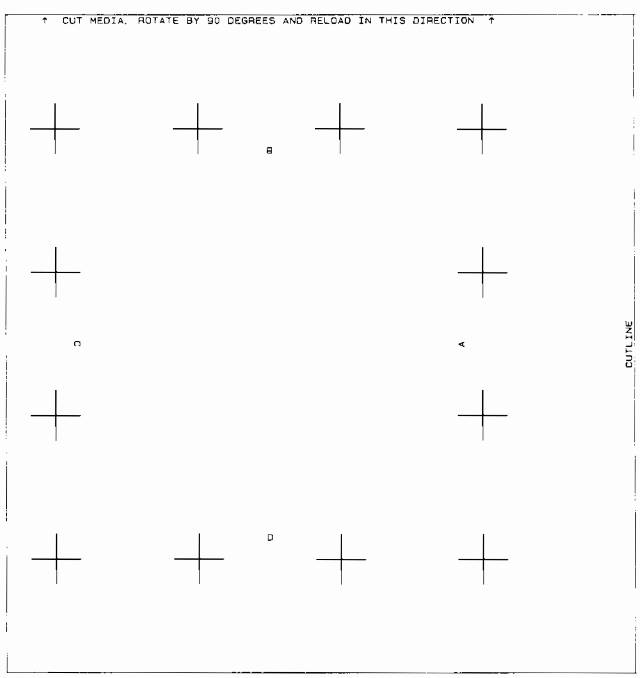
    Press **PEN FORCE** or **PEN DOWN** to end the procedure without storing new constants.

    The plotter puts the pen away and goes to the power-up, Not-Ready state regardless of how the procedure ends.

NOTE: If the calibration procedure aborts for any reason, the plotter will retain and use its previous calibration constants. Only when AEC appears on the display has any change taken place in the EAROM. ■



Sheet One After Rotation. *The sheet has been rotated 180° and is ready to be reloaded into the plotter.*



Sheet Two After Rotation. *The sheet has been cut, rotated 90° counterclockwise, and is ready to be reloaded into the plotter.*

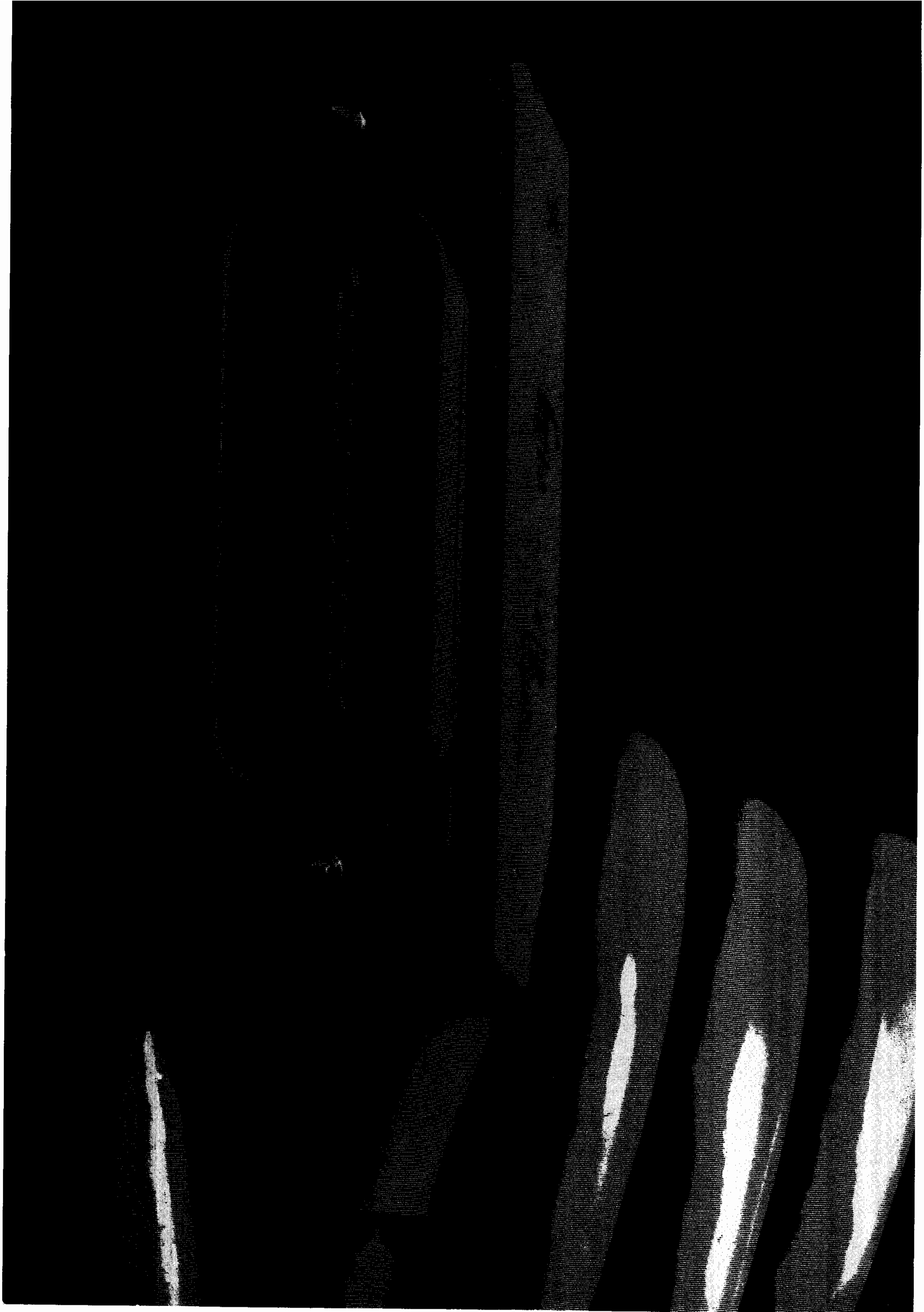
Calibration Procedure Front-Panel Display Codes

The following tables define the front-panel display codes for the calibration procedure. The first table lists codes which prompt an action; the second table lists codes which indicate the specific procedure taking place. Codes are listed in numerical order, not in the sequential order in which they may appear in the procedure.

Code	Operation Action
C03	Press CHART UNLOAD button and remove sheet.
C04	Error. Start over. Load blank sheet one into plotter and press CHART HOLD.
C05	Load blank sheet two into plotter and press CHART HOLD.
C06	Trim bottom of sheet two. Rotate it by 90 degrees counter-clockwise. Reload sheet into plotter and press CHART HOLD.
C07	Rotate sheet one by 180 degrees lengthwise. Reload it into plotter and press CHART HOLD.
C10	Internal error. This may indicate a memory failure. To end program, push PEN DOWN or PEN FORCE. To repeat sheet, press PEN UP or PEN SPEED.
C99	No obvious errors. To save new constants in the EAROM, press PEN UP or PEN SPEED. To keep original constants, press PEN DOWN or PEN FORCE.

Code	Procedure
C00	Initializing the calibration program. Finding sheet size.
C01	Searching the platen for primary distance standard marks.
C02	Trying to locate the crosses on media sheet one.
C11	Drawing crosses on sheet two.
C12	Drawing crosses on sheet one.
C13	Trying to locate the crosses on sheet two.
C14	Writing the calculated constants into EAROM. Wait a few seconds.
C18	Program ended. Can't correctly digitize the platen marks.
C21	Error. Can't locate one of the crosses.
C88	Can't find pen in pen stall 8. Program ended.
BAD	The calibration procedure has ended *without changing* the internal constants of the plotter.
AEC	Acceptable End of Calibration. The calculated constants have been saved into the EAROM.

Notes



Chapter 12
HP-IB Interfacing

What You'll Learn in This Chapter

This chapter defines the HP-IB functions implemented in the plotter and describes the switch selectable operating modes: addressable mode, listen-only mode, and monitor mode. Also included are the purpose and syntax of HP-IB device control instructions, the plotter's reaction to bus clear commands, serial and parallel polling, addressing the plotter as a talker or listener, and examples of sending and receiving data using a variety of computers.

This chapter assumes the user has a working knowledge of the HP-IB; however, if you wish to refresh your memory on HP-IB structure, refer to Appendix A of this manual entitled, An HP-IB Overview.

HP-IB Implementation on the Plotter

The HP-IB conforms to ANSI/IEEE 488-1978 specifications, and direct interconnection of the HP-IB is via the HP-IB PORT connector on the rear panel.

The HP-IB functions implemented in the plotter are as follows:

1. Source Handshake (SH1)
2. Acceptor Handshake (AH1)
3. Talker (T6)
4. Listener (L3)
5. Service Request (SR1)
6. No Remote Local (RL0)
7. Parallel Poll (PP0 if listen only; PP2 if addr < 8; PP1 otherwise)
8. Device Clear (DC1)
9. No Device Trigger (DT0)
10. No Controller (C0)

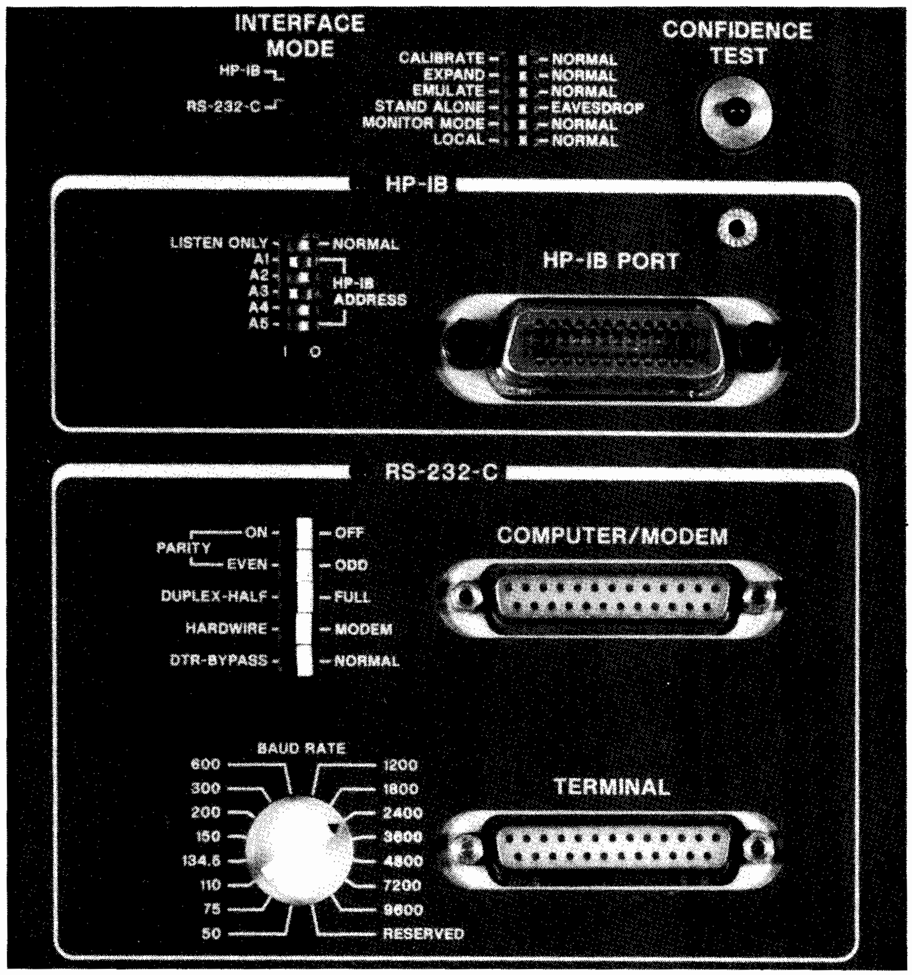
Interface Switches and Controls

Rear-panel switches and controls that pertain to plotter operation in the HP-IB Interface Mode are shown unshaded in the following illustration.

NOTE: The positions of the **CALIBRATE/NORMAL** (HP 7586 only), **EXPAND/NORMAL, EMULATE/NORMAL,** and **MONITOR MODE/NORMAL** switches are sensed each time the front-panel **CHART UNLOAD** and **CHART HOLD** functions are invoked. The **CONFIDENCE TEST** switch is sensed only when the **CHART HOLD** LED is on. The positions of all other switches are sensed only at power-on.

The **CALIBRATE/NORMAL, EXPAND/NORMAL, EMULATE/NORMAL,** and **CONFIDENCE TEST** switches are not related to interface functions and are explained in Chapter 2 of this manual. ■

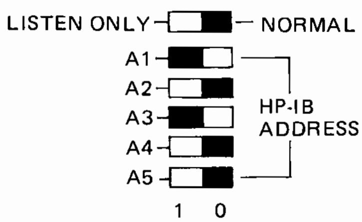
HP-IB Interface Switches and Controls



The HP-IB Interface Mode is selected when the **INTERFACE MODE** switch is set to **HP-IB.** This allows the plotter to function in either of two modes, addressable mode and listen-only mode. In addressable mode, the plotter can function as a talker or as a listener depending on the instructions it receives from the computer. In listen-only mode, it can only listen and it hears all activity on the bus. Monitor mode can be selected in either addressable mode or listen-only mode, and allows all HP-GL instructions to be displayed on a terminal as they are parsed from the plotter's buffer.

Addressing the Plotter

The **HP-IB ADDRESS** switches establish the plotter's address when the **LISTEN ONLY/NORMAL** switch is set to **NORMAL.** Each HP-IB interface can have as many as 15 devices connected to it, set to different specific address codes. The plotter can be set to any one of 31 HP-IB addresses, ranging from 0 through 30. Each address can be selected by setting the **HP-IB ADDRESS** switches to the appropriate bit positions for the particular address value desired. The address selected establishes the plotter's device address. The plotter is set to an address of 05 at the factory. This corresponds to a listen character of % and a talk character of E. Check the following figure for the factory-set **HP-IB ADDRESS** switch positions.



When using the plotter with an HP desktop computer, do not use address 21 because it is reserved for the desktop computer's address. Address 31 should not be used with any computer because it is the universal unlisten command. When not using an HP desktop computer, be sure the computer and plotter do not have the same address. The following table lists the switch positions for each address value.

HP-IB Interface

Address Switch Positions	Address Characters		Address Switch Settings					Address Codes	
	Listen	Talk	A5	A4	A3	A2	A1	Decimal	Octal
	SP	@	0	0	0	0	0	0	0
	!	A	0	0	0	0	1	1	1
	“	B	0	0	0	1	0	2	2
	#	C	0	0	0	1	1	3	3
	$	D	0	0	1	0	0	4	4
PRESET	%	E	0	0	1	0	1	5	5
	&	F	0	0	1	1	0	6	6
	’	G	0	0	1	1	1	7	7
	(	H	0	1	0	0	0	8	10
	)	I	0	1	0	0	1	9	11
	*	J	0	1	0	1	0	10	12
	+	K	0	1	0	1	1	11	13
	,	L	0	1	1	0	0	12	14
	–	M	0	1	1	0	1	13	15
	.	N	0	1	1	1	0	14	16
	/	O	0	1	1	1	1	15	17
	0	P	1	0	0	0	0	16	20
	1	Q	1	0	0	0	1	17	21
	2	R	1	0	0	1	0	18	22
	3	S	1	0	0	1	1	19	23
	4	T	1	0	1	0	0	20	24
RESERVED FOR HP DESKTOP COMPUTER ADDRESS	5	U	1	0	1	0	1	21	25
	6	V	1	0	1	1	0	22	26
	7	W	1	0	1	1	1	23	27
	8	X	1	1	0	0	0	24	30
	9	Y	1	1	0	0	1	25	31
	:	Z	1	1	0	1	0	26	32
	;	[	1	1	0	1	1	27	33
	<	\	1	1	1	0	0	28	34
	=	]	1	1	1	0	1	29	35
	>	^	1	1	1	1	0	30	36
RESERVED FOR UNIVERSAL UNLISTEN COMMAND	?	_	1	1	1	1	1	31	37

Listen-Only Mode

The plotter is in listen-only mode when the **LISTEN ONLY/NORMAL** switch on the rear panel is in the **LISTEN ONLY** position at power-on. In this state, the plotter does not have an address, but listens to all data transmitted on the bus. The plotter cannot be placed in a talker-active state and will not respond to a serial or parallel poll. Listen-only is useful in a system that has no controller but instead has a dedicated talker, such as a magnetic tape drive, or other mass storage unit, transmitting information to the plotter. Also, when more than one plotter is set to the same address, all but one can be placed in listen-only mode. In this case, one plotter responds to output requests but all plotters receive and react to plot commands.

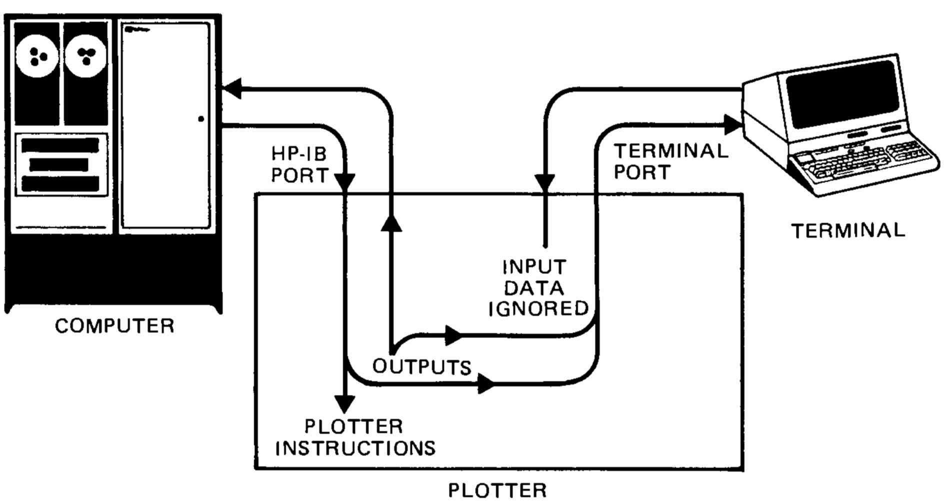
Monitor Mode

Monitor mode is primarily intended to be used as a "debugging" aid for program development. This mode can be enabled in either addressable mode or listen-only mode, and allows all HP-GL instructions that are parsed from the plotter's buffer to be retransmitted to a terminal, via the **TERMINAL** connector. Operation in this mode requires setting the **BAUD RATE** switch to match the baud setting on the terminal.

Monitor mode is enabled when **CHART HOLD** is invoked if the **MONITOR MODE/NORMAL** switch is set to **MONITOR MODE.** If the switch is set to **NORMAL,** the mode can be enabled after **CHART HOLD** is invoked using the Set Plotter Configuration Instruction, ESC . @, or the Set Monitor Mode Instruction, ESC . Q. Monitor mode cannot be disabled using the ESC . @ or ESC . Q instructions if the mode was manually enabled. If monitor mode is enabled in addressable mode, plotter output responses to HP-GL instructions are sent to both the computer and terminal. All data sent to the plotter from the terminal, including Break, is ignored. The following diagram shows how the plotter processes HP-GL instructions while monitor mode is enabled in addressable mode. If monitor mode is enabled in listen-only mode, output responses are not possible.

NOTE: Device control instructions and any resultant plotter output responses are not sent to the terminal while in monitor mode. ■

Data Flow in Monitor Mode



HP-IB Device Control Instructions

In the HP-IB Interface Mode, the plotter responds to 12 device control instructions. These instructions are three-character escape sequences beginning with the ASCII escape character, followed by "." and one of the characters @, A, B, E, J, K, L, O, Q, R, S, or T. The plotter recognizes but ignores nine additional device control instructions with the third character being H, I, M, N, P,), (, Y, or Z. Unlike HP-GL instructions, device control instructions do not enter the plotter's buffer and are acted on immediately when received. Output responses to these instructions are terminated by the same characters, carriage return (CR) and line feed (LF), used to terminate HP-GL output responses. Note also that the plotter sets the EOI line high concurrently with output of the line feed terminator. This EOI signal is interpreted by some computers to mean that transmission of the output response is completed and program execution can resume. For a description of the syntax conventions used to describe device control instructions, refer to Chapter 13, under Syntax for Device Control Instructions.

HP-IB Interface

The Set Plotter Configuration Instruction, ESC . @

USES: The ESC . @ instruction sets an effective logical I/O buffer size and sets parameters necessary for monitor mode. Use the instruction to enlarge the buffer and to enable or disable monitor mode when the **MONITOR MODE/NORMAL** switch is set to **NORMAL**.

SYNTAX: **ESC** . @ [(<DEC>) ; (<DEC>)] :

DEFAULT: **ESC** . @ : Sets logical I/O buffer equal to physical buffer size set by ESC . T instruction. Default physical buffer size is 1024 bytes.

EXPLANATION: Refer to Monitor Mode in this chapter. A description of the ESC . @ instruction parameters follows:

<DEC> The first parameter specifies the *logical* buffer size. At default, both the physical and logical buffer size are the same. The *physical* buffer size is definable with the ESC . T instruction. The logical size should always be less than or equal to the physical buffer size. If you specify a logical size greater than the physical size specified in the ESC . T instruction, the plotter automatically uses the physical size.

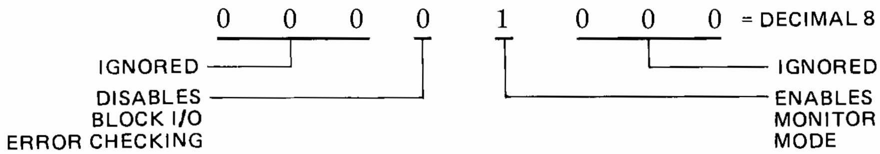
<DEC> The second parameter controls monitor mode. Decimal equivalents in the range of 0 to 127 may be specified. However, only bits 3 and 4 (out of bits 0 through 6) are recognized.

For bit 3, logic 0 disables monitor mode; logic 1 enables monitor mode.

For bit 4, logic 0 disables block I/O error checking; logic 1 enables block I/O error checking. This function exists primarily for I/O compatibility with the RS-232-C interface.

EXAMPLE: ESC . @ ; 8 :

The semicolon defaults the first parameter. The decimal value 8 specifies the corresponding binary value to set the logic state of bits 3 and 4 as follows:



The Output Identification Instruction, ESC . A

USES: The ESC . A instruction outputs the plotter's model number. Use the instruction to interrogate the plotter regarding its model number. The plotter's output response is available immediately, regardless of the current operating state (Not-Ready, View, or Remote). In contrast, the HP-GL instruction, OI, is buffered and will not be parsed from the data buffer until the plotter is placed in the Remote operating state.

SYNTAX: ESC . A

EXPLANATION: No parameters are used.

RESPONSE: The plotter will output the appropriate model number: 7580B, 7585B, or 7586B. The response is terminated by a carriage return, CR, and line feed, LF.

The Output Buffer Space Instruction, ESC . B

USES: The ESC . B instruction outputs the plotter's currently available I/O buffer space. Use this instruction to interrogate the plotter regarding available buffer space as established at default or by the ESC . @ instruction.

SYNTAX: ESC . B

EXPLANATION: No parameters are used. In contrast to HP-GL output instructions which enter the data buffer and are processed on a first-in, first-out basis, this output is available immediately. Failure to read it immediately could cause the wrong information to be read after executing a later HP-GL output instruction.

RESPONSE:

<DEC> The plotter's response is a decimal number that represents the number of bytes of space currently available in the I/O buffer for storing graphic instructions sent by the computer. The range is the range of the logical buffer size, or 0 to 1024 if the default size is used. The response is terminated by a carriage return, CR, and a line feed, LF.

The response can be as large as 18432 if both the ESC . T and ESC . @ instructions have established a larger buffer than default size.

HP-IB Interface

The Output Extended Error Instruction, ESC . E

USES: The ESC . E instruction outputs a number which defines any error related to device control instructions and turns off the front-panel **ERROR** light. Use the instruction to define what type of device control instruction error has occurred, if any.

SYNTAX: **ESC** . E

EXPLANATION: No parameters are used. The ESC . E instruction always outputs the I/O error and clears any error condition.

RESPONSE:

<DEC> The plotter's response is a decimal number, either 0 or in the range of 11-14, followed by the output terminator. The meaning of the response is as defined in the following table.

Error No.	Meaning
0	No error has occurred.
11	Invalid byte received after first two characters, **ESC** ., in a device control instruction.
12	Invalid byte received while parsing a device control instruction. The parameter containing the invalid byte and all following parameters are defaulted.
13	Parameter out of range.
14	Too many parameters received. Additional parameters beyond the proper number are ignored; parsing of the instruction ends when a colon (normal exit) or the first byte of another instruction is received (abnormal exit). **NOTE:** The receipt of something other than another parameter, a semicolon, or a colon will result in error 12 overwriting error 14. ■

The response is terminated by a carriage return, CR, and line feed, LF.

The Abort Device Control Instruction, ESC . J

USES: The ESC . J instruction aborts any device control instruction that may be partially decoded or executed.

SYNTAX: **ESC** . J

EXPLANATION: Unspecified parameters of partial instructions are defaulted. All pending or partially transmitted output requests, from either HP-GL or device control instructions are immediately terminated, including plotter outputs.

The Abort Graphic Instruction, ESC . K

USES: The ESC . K instruction aborts any partially decoded HP-GL instruction and discards instructions in the buffer. Use the instruction as part of an initialization sequence when starting a new program or to terminate plotting of HP-GL instructions in the buffer.

SYNTAX: **ESC** . K

EXPLANATION: This instruction aborts any partially decoded HP-GL instruction but permits the instruction being executed to finish. All pending graphic instructions in the buffer are discarded, leaving the buffer empty. The currently executing vector is allowed to be completed, but all other scheduled vectors are aborted. This condition is particularly evident if ESC . K is executed while the plotter is in the process of completing polygon, label, arc, circle, or line type instructions which contain multiple vector moves, or if an HP-GL VS instruction has specified a slow velocity. There is no response from the plotter.

The Output Buffer Size Instruction, ESC . L

USES: The ESC . L instruction outputs the logical size, in bytes, of the plotter's I/O buffer. Use the instruction to obtain information on the buffer size.

SYNTAX: **ESC** . L

EXPLANATION: No parameters are used.

RESPONSE:

<DEC> The instruction causes the plotter to output, in ASCII, a decimal number equal to the number of bytes of buffer space established by the ESC . @ instruction. If no logical size has been specified, the response is the default buffer size, 1024 bytes. The response is not transmitted until the buffer is empty.

The response can be as large as 18432 if both the ESC . T and ESC . @ instructions have established a larger buffer size than default size.

The response is terminated by a carriage return, CR, and a line feed, LF.

The Output Extended Status Instruction, ESC . O

USES: The ESC . O instruction outputs the plotter's extended status. Use this instruction to interrogate the plotter regarding the data buffer and the current operating state. Unlike the HP-GL output status instruction, OS, the ESC . O instruction does not enter the data buffer but is executed immediately. Failure to read it immediately could cause the wrong information to be read after executing a later HP-GL output instruction.

SYNTAX: **ESC** . O

EXPLANATION: No parameters are used.

HP-IB Interface

RESPONSE:

<DEC> The plotter's response is the decimal equivalent of a 16-bi extended status word. The response is terminated by a carriage return, CR, and line feed, LF. The status wor bits are as defined in the following table.

Extended Status Word Bits

Bit Position	Logic State	Decimal Value	Meaning
*0	0	0	Single sheet loaded.
	1	1	Roll paper loaded.
1	0	0	"Clean" paper loaded. (Set after paper is sensed.)
	1	2	Current page is not clean. Set under following conditions: a. No paper loaded or advanced since power-up or execution of ESC . R instruction. b. Pen has touched paper since last paper load or advance. c. No paper sensed before or after attempted paper advance (end of roll).
*2	0	0	No paper advance (no AF, AH, FR, or PG instruction executed) since last ESC . O instruction. The ESC . O instruction resets this bit to 0.
	1	4	Paper advance instruction (AF, AH, FR, or PG) executed since last ESC . O instruction.
3	0	0	Buffer is not empty.
	1	8	Buffer is empty and ready for data.
4 5	0 0	0 0	Remote state. Processing HP-GL instructions.
4 5	1 0	16 0	View state. Paper loaded but graphics suspended.
4 5	0 1	0 32	Not ready. Paper not loaded, graphics suspended.
6	0	0	Cover is closed.
	1	64	Cover is open.
7	0	0	Plotter in **NORMAL** mode.
	1	128	Plotter in **EMULATE** mode.
8	0	0	**EXPAND** mode is not on.
	1	256	**EXPAND** mode is on.
9–15	†	†	Not used.

*Bit recognized by the HP 7586 plotter only.

†Undefined.

Possible outputs (exclusive of bits 0–2) and their meanings are described in the following table.

Extended Status Word Decimal Code Meanings

Decimal Value* Output to the Computer	Meaning						
	Buffer Is Not Empty	Buffer Is Empty	Remote State. Processing HP-GL Instructions	View State. Paper Is Loaded. Graphics Suspended	Not-Ready State. Paper Not Loaded. Graphics Suspended	Pen Carriage Cover Closed	Pen Carriage Cover Open
0	X		X			X	
8		X	X			X	
16	X			X		X	
24		X		X		X	
32	X				X	X	
40		X			X	X	
64	X		X				X
72		X	X				X
80	X			X			X
88		X		X			X
96	X		X		X		X
104		X			X		X

*Add 128 when plotter is in **EMULATE** mode.
Add 256 when plotter is in **EXPAND** mode.

NOTE: *(Applies to the HP 7586 plotter only.)* A response of 5 (bits 0 and 2 set to 1) indicates paper advance has been completed. If the response is 1 (bit 0 set to 1 and bit 2 set to 0), execute a second ESC . O so that sufficient time will elapse for the paper advance instruction to be completed. ■

The Set Monitor Mode Instruction, ESC . Q

USES: The ESC . Q instruction enables or disables monitor mode when the **MONITOR MODE/NORMAL** switch is set to **NORMAL.**

SYNTAX: **ESC** . Q <DEC> :

DEFAULT: **ESC** . Q : Disables monitor mode.

EXPLANATION: The ESC . Q instruction enables the monitor mode. It accomplishes the same as setting bit 3 of the ESC . @ instruction. A description of the single parameter for this instruction follows:

<DEC>

Parameter	Result
0	Disables monitor mode (default)
1	Activates monitor mode
2	Activates monitor mode

A parameter greater than 2 sets I/O error 13 and establishes default conditions.

The Reset Instruction, ESC . R

USES: The ESC . R instruction establishes a power-up I/O reset. You can use this instruction in an initialization sequence when starting a new program.

SYNTAX: ESC . R

EXPLANATION: No parameters are used.

ESC . R is a convenient way to establish the same reset conditions as sending the following instruction sequence:

ESC . J   ESC . K   ESC . T :

This instruction flushes all buffers, resets the parser, and defaults user-defined RAM allocations (physical I/O buffer size, polygon buffer size, and the downloadable character buffer size).

The Output Memory Size Instruction, ESC . S

USES: The ESC . S instruction outputs the total memory size of user-definable RAM, or the memory space available in any of its three buffers: the I/O buffer, the polygon buffer, and the downloadable character buffer. Use the instruction to determine how much memory is currently allocated to each buffer.

SYNTAX: ESC . S (<DEC>) :

DEFAULT: ESC . S : Outputs total RAM space available, 18432 bytes (same as ESC . S 0 :).

EXPLANATION: A colon must be used following the last parameter.

A description of the instruction's parameters follows:

<DEC> The parameter range is 0 to 3.

0 requests total RAM memory available (always 18432 bytes)

1 requests current input buffer size

2 requests current polygon buffer size

3 requests current downloadable character buffer size

RESPONSE:

<DEC> The plotter outputs the total amount of bytes in user-definable memory or the memory available in one of the three buffers, according to the parameter used. The response is terminated by a carriage return, CR, and a line feed, LF.

The Configure Memory Instruction, ESC . T

USES: The ESC . T instruction allocates memory in user-definable RAM, which consists of three separate buffers: the physical I/O buffer, the polygon buffer, and the downloadable character buffer. Use the instruction to enlarge the I/O buffer and the polygon buffer as needed, or to allocate memory to the downloadable character buffer.

SYNTAX: ESC . T [(<DEC>) ; (<DEC>) ; (<DEC>)] :

DEFAULT: ESC . T : Sets the I/O buffer (physical size) to 1024. Sets the polygon buffer to 3072 bytes and the downloadable character buffer to 0 bytes. Discards all data.

EXPLANATION: Below is an explanation of each parameter, followed by a table which summarizes the options with the ESC . T instruction.

Use of this instruction without parameters (ESC . T :) sets default values and clears all three areas. Any data is discarded.

<DEC> The first parameter is a decimal value for the size of the *physical* I/O buffer. See the ESC . @ instruction for an explanation of the difference between *physical* and *logical* buffer size.

NOTE: For practical purposes, HP-IB only uses the logical buffer. If you use ESC . T to increase the physical I/O buffer size from a previously defined value, you must also send the ESC . @ instruction to make the larger buffer space available on HP-IB. ESC . B, ESC . L, and parallel and serial polls all give information about the logical buffer, so it is especially important to reset the buffer size using ESC . @ after enlarging the physical size with ESC . T.

If you set the physical buffer size to *less than* the current logical buffer size established by ESC . @, the logical size is automatically set to the value of the new physical size. ■

<DEC> The second parameter is the decimal value for the size of the polygon buffer.

<DEC> The third parameter is the decimal value for the size of the downloadable character buffer. Since the default size of this table is zero, you must allocate space to this table before using the DL instruction.

HP-IB Interface

The following table presents the range of memory allocation for each parameter.

Parameter	Minimum	Maximum	Default
Physical I/O buffer	2*	18 432***	1024
Polygon buffer	2**	18 430***	3072
Downloadable character buffer	444**	18 430***	0

*Specifying zero allocates 2 bytes.

**Specifying zero allocates no bytes. Specifying a value between 1 and the minimum allocates the minimum.

***Specifying greater than the maximum allocates the maximum.

NOTE: In all three buffers, the plotter will automatically increment odd input values to make them even. ■

The total number of available bytes is 18 432. If the sum of the three allocations exceeds 18 432, default values are used and error 13 is set.

Bus Commands

Reaction to Bus Commands DCL, SDC, and IFC

The computer can set all devices on the HP-IB system to a predefined or initialized state by sending the device clear command DCL. The computer can also set selected devices to a predefined or initialized state by sending a selected device clear command SDC along with the addresses of the devices. The basic difference is that devices will obey SDC only if they are addressed to listen, whereas DCL clears all devices on the bus. The interface clear command IFC is used by the computer to override all bus operations and return the bus to a known quiescent state.

Upon receipt of either DCL, SDC, or IFC commands, the plotter stops any vector currently in process, cancels any vectors requested but not yet begun, clears the buffer, resets the I/O to begin accepting a new instruction, and disables any current output.

The device clear and interface clear commands *do not* reset parameters in the plotter to their default values. They are not the same as the HP-GL DF or IN instructions.

Serial and Parallel Polling

Polling is the process used by the computer to determine which device on the HP-IB bus has initiated a require service message. The conditions which will cause the require service message to be sent to the computer are defined by The Input Mask Instruction, IM. Refer to Chapter 9.

THE SERIAL POLL

A serial poll enables the computer to learn the status or condition of devices on the bus. It is commonly used by the computer to determine who is requesting service.

The serial poll is so named because the computer polls devices one at a time rather than all at once. The plotter will respond to a serial poll by sending the status byte as described under The Output Status Instruction, OS, Chapter 9. The S-mask parameter of The Input Mask Instruction, IM, is used to specify which status byte conditions will send the service request message. Unless the user changes the S-mask value from the default setting of 0, the plotter will never send a service request. Bit position 6 of the status byte will be set to 1 (if the S-mask value is not 0) when any of the conditions designated by the S-mask are true. Bit position 6 will be set to 0 after a serial poll has occurred or the condition causing the initial request no longer exists.

A computer must issue special commands to initiate and terminate a serial poll. During a serial poll, a device must be instructed to talk and the computer to listen. Therefore, a serial poll cannot be executed when a plotter is in listen-only mode.

THE PARALLEL POLL

Parallel polling can only be done to plotters with an address 0 through 7 or to plotters with an address > 7 which have been remotely configured to respond to a parallel poll. The plotter will respond positively to a parallel poll only if the conditions specified in the P-mask are satisfied and parallel poll response is enabled. The P-mask parameter of the input mask instruction, IM, is used to specify which status byte conditions will result in a logical 1 response to a parallel poll. The response to a parallel poll is limited to setting the appropriate data line to a logical 1. The line used is determined by the plotter's address value as shown in the table below:

Plotter Address	Parallel Poll Bit Position	HP-IB Data Line Number	
0	7	8	
1	6	7	
2	5	6	
3	4	5	
4	3	4	
5	2	3	← PLOTTER PRESET ADDRESS
6	1	2	
7	0	1	

Unless remotely configured to do so, plotters with address settings from 8 through 30 will not respond to a parallel poll.

To execute a parallel poll, the controller sets the ATN and EOI lines to 1. The controller reads the eight data lines, and determines from these lines which instrument on the bus is requesting service. The computer then sends the parallel poll disable command. Not all computers have parallel poll capability.

It is important to remember that the plotter will not send a logical 1 unless the P-mask bit value has been changed from the default value of 0 and some condition included in the new P-mask value is true. The plotter does not respond to a parallel poll in listen-only mode.

Addressing the Plotter As a Talker or Listener

In order to communicate effectively with the plotter, it is important that you completely understand the addressing protocol of your computer. Therefore, you may wish to review this aspect of your computer before proceeding.

Computers with No High Level I/O Statements

On low level computers, addressing devices on the HP-IB bus is accomplished using mnemonics, such as CMD, which serve as the "bus command."

When bus commands are necesary, a typical addressing sequence is

<Unlisten Command> <Talk Address> <Listen Addresses>

This sequence is made up of three major parts which serve the following purposes.

1. The unlisten command is the universal bus command with a character code of "?". It unaddresses all listeners. After the unlisten command is transmitted, no active listeners remain on the bus.
2. The talk address designates the device that is to talk. A new talk address automatically unaddresses the previous talker.
3. The listen addresses designate one or more devices that are to listen. A listen address adds the designated device as listener along with other addressed listeners.

This basic addressing sequence simply states who is to talk to whom. The unlisten command ("?") plays a vital role in this sequence. It is important that a device receive only the data that is intended for it.

When a new talk address is transmitted in the addressing sequence, the previous talker is unaddressed. Therefore, only the new talker can send data on the bus and there is no need to routinely use an untalk command in the same manner as the unlisten command.

Computers with High Level I/O Statements

In more powerful computers, higher level input/output (I/O) statements are used to specify addresses on the HP-IB bus. In these cases, the addressing protocol (unlisten, talk, listen) is a function of the computer's internal operating system and need not be of concern to the user.

Sending and Receiving Data

Computer-to-Plotter

Transmitting data from a computer to the plotter is typically accomplished using I/O statements such as WRITE, PRINT, PRINT#, or OUTPUT. The following examples of sending program data to the plotter from various computers are only intended to illustrate the necessity for understanding the I/O statement protocol implemented by your computer. Each of these examples will cause the plotter to label the identity of the computer sending data, beginning at the X,Y coordinates 1000, 2000. The examples involve sending both character string and numeric data as variables, and constants or literals.

HP 9825 Example

```
0: fxd;dim A$[13]
1: " SENDING DATA"→A$
2: 2000→Y
3: 9825→B
4: wrt 705,"SP1;PA1000,",Y
5: wtb 705,"LBHP",str[B],A$,3
6: end
```

A terminator is sent by the HP 9825 at the end of a wrt statement.

Result: HP 9825 SENDING DATA

HP 9835/9845 Example

```
10   PRINTER IS 7,5
20   A$=" SENDING DATA"
30   B=9835
40   C=9845
50   Y=2000
60   PRINT "SP1;PA1000,";Y
70   PRINT USING "K";"LBHP ",B,"/",C,A$,CHR$(3)
80   END
```

A terminator is sent by the HP 9835/9845 at the end of a PRINT statement.

Result: HP 9835/9845 SENDING DATA

HP 2647 Example

```
10   ASSIGN "H#5" TO #1
20   DIM A$[13]
30   A$="SENDING DATA"
40   B=2647
50   Y=2000
60   PRINT #1;"SP1;PA1000,",Y
70   PRINT #1;"LBHP",B,A$,CHR$(3)
80   END
```

A terminator is sent by the HP 2647 at the end of PRINT #1 statements.

Result: HP 2647 SENDING DATA

HP Series 200 (HP 9816/9826/9836) Example (Using 2.0 BASIC)

```
10  PRINTER IS 705
20  A$=" SENDING DATA"
30  Y=2000
40  PRINT "SP1;PA1000,";Y
50  PRINT USING "K";"LBHP SERIES 200",A$,CHR$(3)
60  END
```

A terminator is sent by the HP 9816/9826/9836 at the end of a PRINT statement.

Result: HP SERIES 200 SENDING DATA

HP-85 Example

```
10 PRINTER IS 705
20 A$="SENDING DATA"
30 B=85
40 Y=2000
50 PRINT "SP1;PA1000,",Y
60 PRINT "LBHP";B;A$;"Ñ"
70 END
```

A terminator is sent by the HP-85 following PRINT statements.

Result: HP 85 SENDING DATA

HP 1000 Example

This example uses FORTRAN IV and assumes that the system has been configured so that the logic unit number 13 sends data to an HP-IB address of 1. Therefore, the plotter address switch on the back panel has been set to 1. The program name, VBP1, may be changed to any other legal program name and line numbers as given are only necessary for the format statements. The field width specification of 4 is adequate to plot to the location 1000,2000; you may wish to use a field width of 5 or 6 to allow for pen positions up to the hard-clip limits of the plotter. A field larger than six characters would only be necessary for scaled data greater than 999 999 or less than −99 999.

```
FTN4,L
      PROGRAM VBP1
10    IY=2000
20    IB=1000
30    WRITE [13,101] IY
40    WRITE [I3,102] IB
101   FORMAT ["SP1;PA1000,",14]
102   FORMAT ["LB HP ",14," SENDING DATA␃"
      END
```

A terminator is sent by the HP 1000 following WRITE statements.

Result: HP 1000 SENDING DATA

Plotter-to-Computer

Outputting data from the plotter to the computer is typically accomplished using I/O statements such as READ, INPUT, or ENTER. The following examples of obtaining output data from the plotter using various computers are only intended to illustrate the necessity for understanding the I/O statement protocol implemented on your computer. Each of these examples instructs the pen to move to plotter coordinates X = 1000, Y = 1000 and then output the current pen position and the plotter identifier string to the computer. For simplicity, each example uses 7580B as the plotter identifier.

HP 9825 Example

```
0: dim A$[5];fxd0
1: wrt 705,"PA1000,1000;OC"
2: red 705,A,B,C
3: wrt 705,"OI"
4: red 705,A$
5: dsp A,B,C,A$
6: end
```

Displayed current pen position and identification.

1000 1000 0 7580B

HP 9835/9845 Example

```
10    PRINTER IS 7,5
20    PRINT "PA1000,1000;OC"
30    ENTER 705;A,B,C
40    PRINT "OI"
50    ENTER 705;A$
60    DISP A,B,C,A$
70    END
```

Displayed current pen position and identification.

1000 1000 0 7580B

HP 2647 Example

```
10 ASSIGN "H#5" TO #1
20 PRINT #1;"PA1000,1000;OC;"
30 READ #1;A,B,C
40 PRINT #1;"OI"
50 READ #1;A$
60 PRINT A,B,C,A$
70 END
```

Displayed current pen position and identification.

1000 1000 0 7580B

HP-IB Interface

HP Series 200 (HP 9816/9826/9836) Example (Using 2.0 BASIC)

```
10  PRINTER IS 705
20  PRINT "SP1;PA1000,1000;OC"
30  ENTER 705;A,B,C
40  PRINT "OI"
50  ENTER 705;A$
60  DISP A,B,C,A$
70  END
```

Displayed current pen position and identification.

1000	1000	0	7580B

HP-85 Example

```
10   PRINTER IS 705
20   PRINT "PA1000,1000;OC"
30   ENTER 705;A,B,C
40   PRINT "OI"
50   ENTER 705;A$
60   DISP A,B,C,A$
70   END
```

Displayed current pen position and identification.

1000	1000
0	7580B

HP 1000 Example

This example uses FORTRAN IV and assumes that the system has been configured so that the logic unit number 13 sends data to an HP-IB address of 1. Therefore, the plotter address switch on the back panel has been set to 1. The program name, VBP2, may be changed to any other legal program name and the line numbers as given are only necessary for the format statements. A field width specification of 5 would be necessary to output pen positions where either the X- or Y-position was greater than 9999 or less than −999. Data is read in from the plotter using free field format so that the commas sent by the plotter between output parameters are interpreted as separators.

```
FTN4,L
      PROGRAM VBP2
      DIMENSION A[3]
10    WRITE [13,101]
20    READ [13,*] IX,IY,IP
30    WRITE [13,103]
40    READ [13,104] A
45    WRITE [1,102] IX,IY,IP
50    WRITE [1,104] A
101   FORMAT ["PA1000,1000;OC"]
102   FORMAT [3I5]
103   FORMAT ["OI"]
104   FORMAT [3A2]
      END
```

Displayed current pen position and identification.

```
 1000 1000     0
7580B
```

HP-IB Interface

Chapter 13

RS-232-C/CCITT V.24 Interfacing

Computer Museum

What You'll Learn in This Chapter

This chapter pertains only to the RS-232-C Interface Mode. It describes control functions and how to connect the plotter, terminal, and computer in a modem or hardwire environment. It also describes connector pin allocations, stop bits, baud rates, and transmission errors. A tutorial description of the four handshaking methods is also included. The last part of the chapter is devoted to the device control instructions. The syntax of device control instructions is given, followed by a detailed explanation of each instruction.

NOTE: All information in this chapter applies equally to RS-232-C and CCITT V.24 interfaces. For purposes of simplicity, both are referred to as RS-232-C. ■

Terms You Should Understand

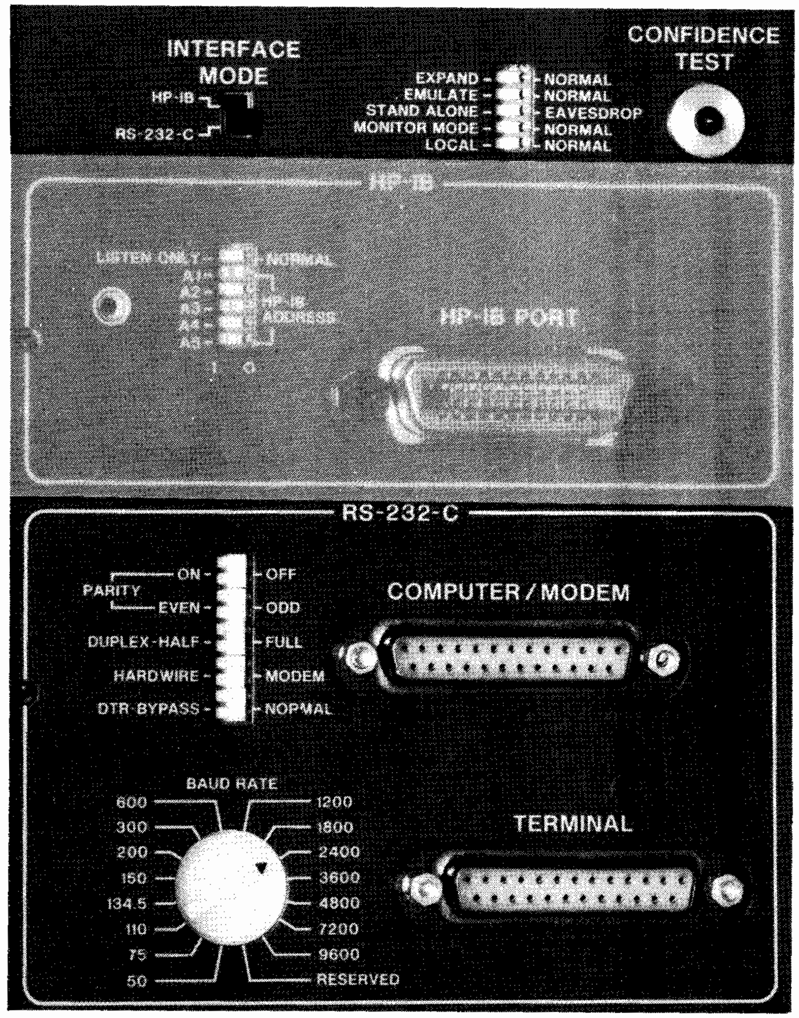
Programmed-on — A plotter-on instruction, ESC . (or ESC . Y, has been received. The plotter will accept data and interpret it as either HP-GL or device control instructions.

Programmed-off — A plotter-off instruction, ESC .) or ESC . Z, has been received. The plotter ignores all data, except a plotter-on instruction.

RS-232-C Controls and Indicators

Rear-panel controls and indicators that pertain to plotter operation in the RS-232-C Interface Mode are shown unshaded in the following illustration.

RS-232-C Interface Switches and Indicators

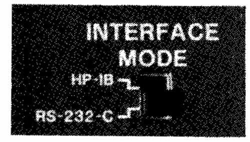


The functions of rear-panel and front-panel controls and indicators that pertain only to the RS-232-C Interface Mode are as follows:

NOTE: The positions of the **CALIBRATE/NORMAL, EXPAND/NORMAL, EMULATE/NORMAL, MONITOR MODE/NORMAL,** and **LOCAL/NORMAL** switches are sensed each time the front-panel **CHART UNLOAD** and **CHART HOLD** functions are invoked. The position of the **DTR** switch is sensed at all times. The **CONFIDENCE TEST** switch is sensed only when the **CHART HOLD** LED is on. The positions of all other switches are sensed only at power-on.

The **CALIBRATE/NORMAL, EXPAND/NORMAL, EMULATE/NORMAL,** and **CONFIDENCE TEST** switches are not related to interface functions and are explained in Chapter 2 of this manual. ■

Rear Panel



INTERFACE MODE — A two-position switch that is sensed only at power-on. In the RS-232-C position, the **RS-232-C** LED is on steady and indicates the plotter is operating in the RS-232-C Interface Mode.

RS-232-C — An LED that is on steady only when the **INTERFACE MODE** switch is set to **RS-232-C** at power-on.

STAND ALONE/EAVESDROP — A two-position switch that determines whether the plotter is programmed-on or programmed-off at power-on. It is used in conjunction with the **LOCAL/NORMAL** and **MONITOR MODE/NORMAL** switches to configure the plotter for operation in various system environments. Refer to Plotter Environments, in this chapter.

In **STAND ALONE,** the plotter is programmed-on at power-on, and cannot be programmed-off. The primary purpose for this switch position is to provide compatibility with HP 7580 and HP 7585 plotters that are equipped with the Option 001 (RS-232-C) interface. The plotter is compatible when it is connected to a computer via the **COMPUTER/MODEM** connector, and the **LOCAL/NORMAL** and **MONITOR MODE/NORMAL** switches are set to **NORMAL.** The plotter can also be installed in the other defined environments, but only programmed-on operation capabilities can be used.

NOTE: When three-wire data communication is desired, bit 1 of the second parameter of the ESC . @ instruction must be set to 1 to ensure full compatibility with plotters having the Option 001 (RS-232-C) interface. ■

In **EAVESDROP,** the plotter is programmed-off at power-on, and must be programmed-on before it will respond to any other instructions. The plotter can be installed in any of the defined plotter environments, and both programmed-on and programmed-off operating capabilities can be used.

MONITOR MODE/NORMAL — A two-position switch that enables monitor mode 1 when in the **MONITOR MODE** position. Monitor mode 1 and 2 can also be programmatically enabled using the Set Plotter Configuration Instruction, ESC . @, or the Set Monitor Mode Instruction, ESC . Q.

NOTE: If monitor mode 1 is manually enabled, it cannot be disabled using the ESC . @ or ESC . Q instruction. However, both instructions can be used to toggle between monitor mode 1 and monitor mode 2. Refer to Monitor Mode in this chapter. ■

LOCAL/NORMAL — A two-position switch that selects either the normal or local operating mode. Refer to Plotter Environments, in this chapter.

PARITY — The parity **ON/OFF** and **EVEN/ODD** switches are effective when the plotter is programmed-on in either the normal or local operating mode. These switches set parity checking and generation electronics for use with odd or even parity as established by system requirements. If parity is not used, the **ON/OFF** switch is set to **OFF,** and the setting of the **EVEN/ODD** switch is irrelevant.

DUPLEX — The **DUPLEX** switch is effective only when the plotter is set to local, programmed-off mode or monitor mode 2.

During local, programmed-off mode operation the switch function is as follows:

1. When set to **FULL**, the plotter returns (echoes) data received from the terminal back to the terminal.
2. When set to **HALF**, the echo is suppressed.

During monitor mode 2 operation the switch function is as follows:

1. When set to **FULL**, and the computer is working in an echo-plex environment, all plotter output responses are echoed from the computer, through the plotter, to the terminal.
2. When set to **HALF**, all plotter output responses are sent to both the computer and the terminal. All data received from the computer is also sent to the terminal.

HARDWIRE/MODEM — A two-position switch that determines the power-on bit states of the 2nd parameter in the ESC . @ instruction. After power-on, the switch position can be overridden programmatically using the ESC . @ instruction.

In **MODEM**, the bits are set to disable hardwire handshake and to enable full duplex data communications protocol.

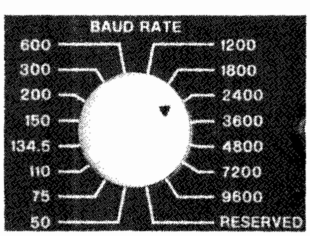
In **HARDWIRE**, the bits are set to enable hardwire handshake and to permit data communication in an environment where the only lines used are Transmitted Data (BA), Received Data (BB), and ground.

NOTE: Data communication requires only three wires; however, the Data Terminal Ready (CD) line must also be connected if hardwire handshake is to be used. ■

DTR — A two-position switch that is sensed at all times. It determines whether the plotter or terminal has control of the **COMPUTER/MODEM** DTR line state.

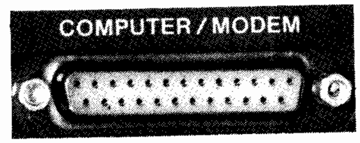
In **NORMAL**, the plotter holds the DTR line high, unless hardwire handshake is enabled. Refer to Hardwire Handshake, in this chapter.

In **BYPASS**, the DTR line is controlled by the terminal. This position is provided to prevent certain types of modems, such as the Bell 113, from inadvertently logging the user off the computer if plotter power is cycled on or off. Hardwire handshake is functionally disabled in this position because the DTR line is held high by the terminal.

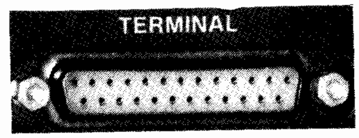


BAUD RATE — A 16-position rotary switch that establishes the baud rate at which data is received and transmitted by the plotter. This switch should be set at the same baud rate selected on the computer and/or terminal.

The **COMPUTER/MODEM** connector is used to interconnect the plotter with a modem or host computer. It is a standard 25-pin, female-type, RS-232-C connector.



The **TERMINAL** connector is identical to the **COMPUTER/MODEM** connector and is used to interconnect the plotter with a terminal.



Front Panel

DSR (HP 7580 and 7585 only) — The **DSR** LED indicates the state of the modem or host computer when the plotter is connected into a computing system. The LED is on steady when the Data Set Ready (CC) line from the modem or host computer is high.

BYPASS — A pushbutton switch with an LED that is functional only if the plotter is in the RS-232-C Interface Mode. The operating characteristics of the **BYPASS** pushbutton are determined by the operating environment selected by the **STAND ALONE/EAVESDROP** switch.

In the stand alone environment, the **BYPASS** LED is set to off at power-on to indicate the plotter is programmed-on. The plotter cannot be programmed-off, and pressing the **BYPASS** pushbutton has no effect.

In the eavesdrop environment, the **BYPASS** LED is set to on at power-on to indicate the plotter is programmed-off in either the normal or local operating mode. Repeatedly pressing the **BYPASS** pushbutton will toggle the plotter between programmed-off (LED on) and programmed-on (LED off). Refer to Plotter Environments, in this chapter.

Setting Up the Plotter: a Checklist

The following steps outline the decisions that must be made to correctly interface the plotter with a given system environment.

1. Determine which installation and operating environment matches your system. Refer to Plotter Environments, in this chapter.

2. Check that you have the required cables and connect the plotter as shown for the environment chosen in step 1. Use the cable supplied with the plotter when connecting to the **COMPUTER/MODEM** connector. Use the cable supplied with your terminal when connecting to the **TERMINAL** connector.

3. Determine if parity checking is used on your system and set the **PARITY** switches accordingly.

4. Determine the baud rate at which your computer sends data and set the **BAUD RATE** switch accordingly.

5. Determine which handshake your system uses. The four basic kinds of handshakes are described in the section entitled Handshaking. Note which device control instructions are used to establish that handshake. Refer to the manuals for your computer to determine which parameters must be set and to what values. The instructions you will use to set up the required handshake are explained in the last part of this chapter.

Plotter Environments

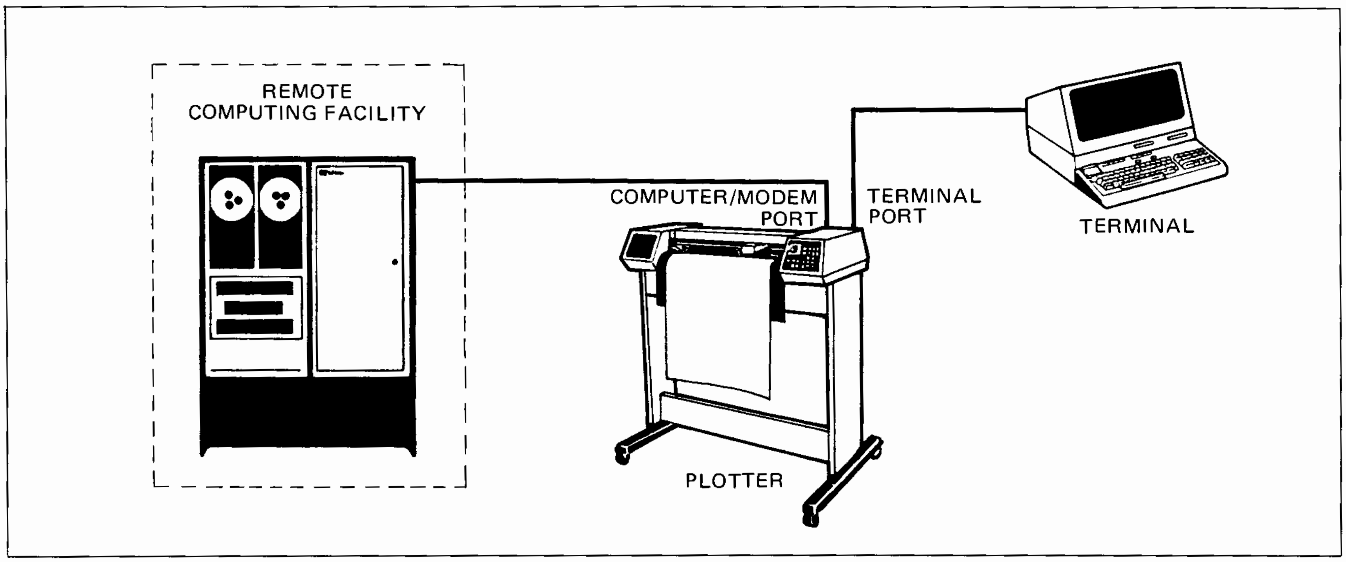
There are three basic ways to position the plotter in a computer system. These positions are referred to as the eavesdrop environment when the plotter is connected in series between a computer and terminal; the stand alone environment when the plotter is connected only to a computer; and the local environment when the plotter is connected only to a terminal. Installation and the operating modes which can be accessed in each environment are described in the following pages; you need only read the section which applies to your system.

Eavesdrop Environment

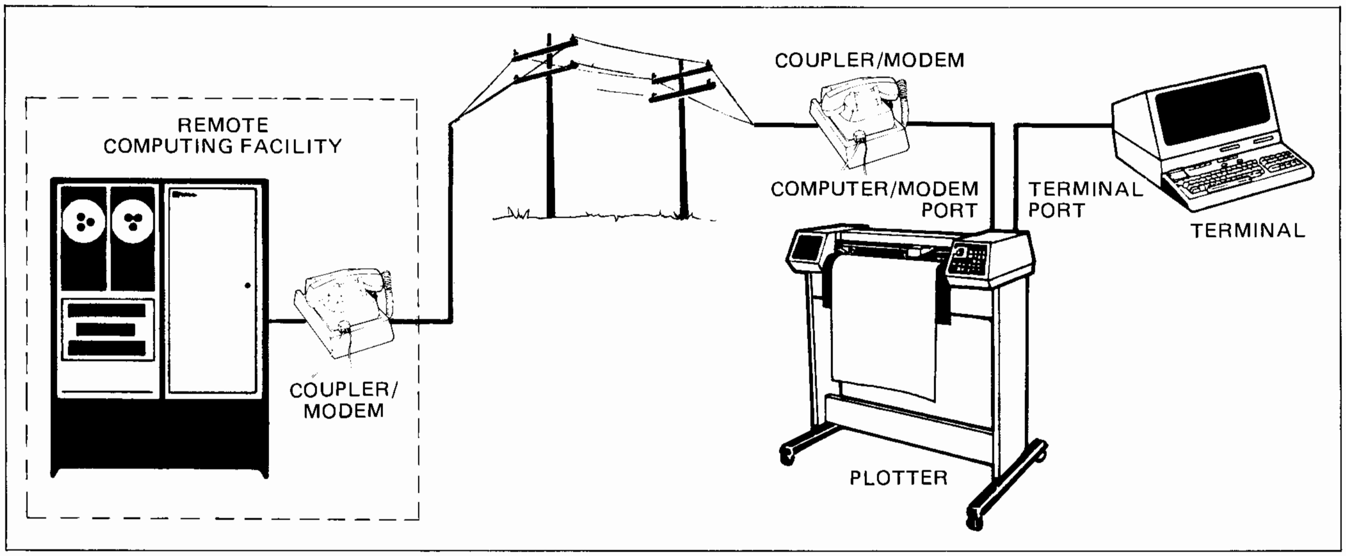
EAVESDROP INSTALLATION

In the eavesdrop environment, the plotter is connected in series between the computer and terminal. Interconnection may be directly through RS-232-C cabling or via a modem for communication over telephone lines. Diagrams of both interconnection methods are as follows:

Plotter Interconnection with a Terminal and Remote Facility Using RS-232-C/CCITT V.24 Cabling



RS-232-C Interface



EAVESDROP OPERATION

Eavesdrop operation is permitted with the **STAND ALONE/EAVESDROP** switch set to either position. The primary difference is that in **EAVESDROP** the plotter is programmed-off at power-on, and in **STAND ALONE** it is programmed-on and cannot be programmed-off. While operating in the eavesdrop environment, data cannot be routed between the computer and terminal with the plotter power turned off. After the plotter is powered-on, data routing is a function of whether the plotter is programmed-on or programmed-off in either the normal or local operating mode. Monitor mode is available only when the plotter is programmed-on in the normal mode. Data routing and operation details for all six modes are decribed in the following paragraphs.

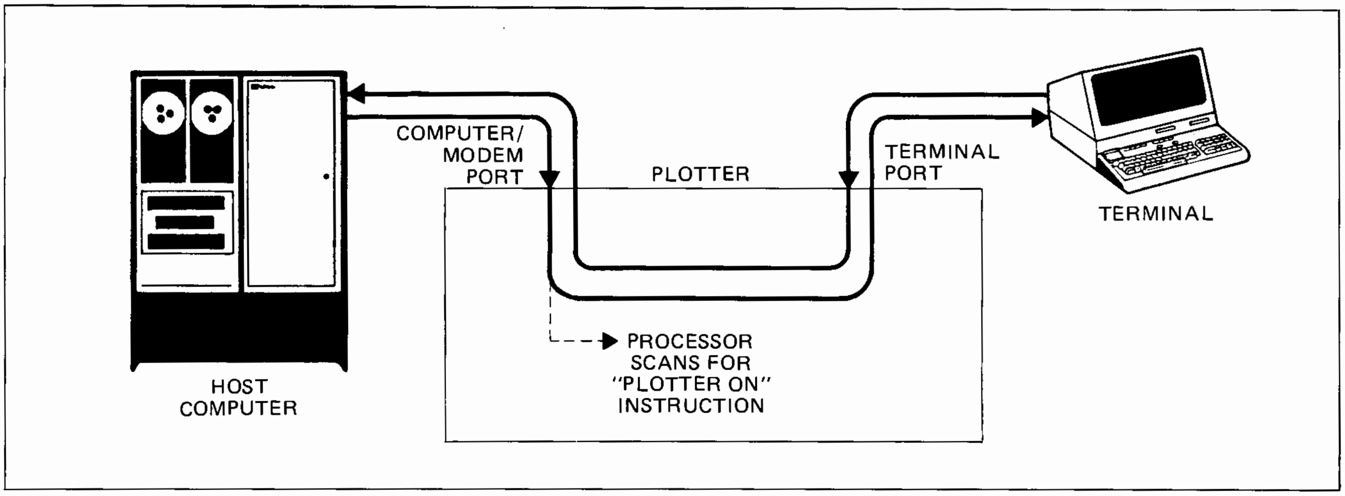
Power-Off Mode

When the plotter is turned off, data communication through the plotter is not possible. However, the **DTR** switch can be set to **BYPASS** to prevent inadvertently logging the user off the computer if plotter power is cycled off and on. If this protection feature is used, hardwire handshake cannot be used because the DTR line is held high by the terminal.

Normal, Programmed-Off Mode

The plotter can be in this mode only if the **STAND ALONE/EAVESDROP** switch is set to **EAVESDROP** and the **LOCAL/NORMAL** switch is set to **NORMAL.** In this mode, the plotter's processor passes data between the computer and the terminal as shown in the following diagram. The plotter will respond only to a plotter on instruction, ESC . (or ESC . Y, from the computer or the front-panel **BYPASS** pushbutton.

Plotter in Normal, Programmed-Off Mode



RS-232-C Interface

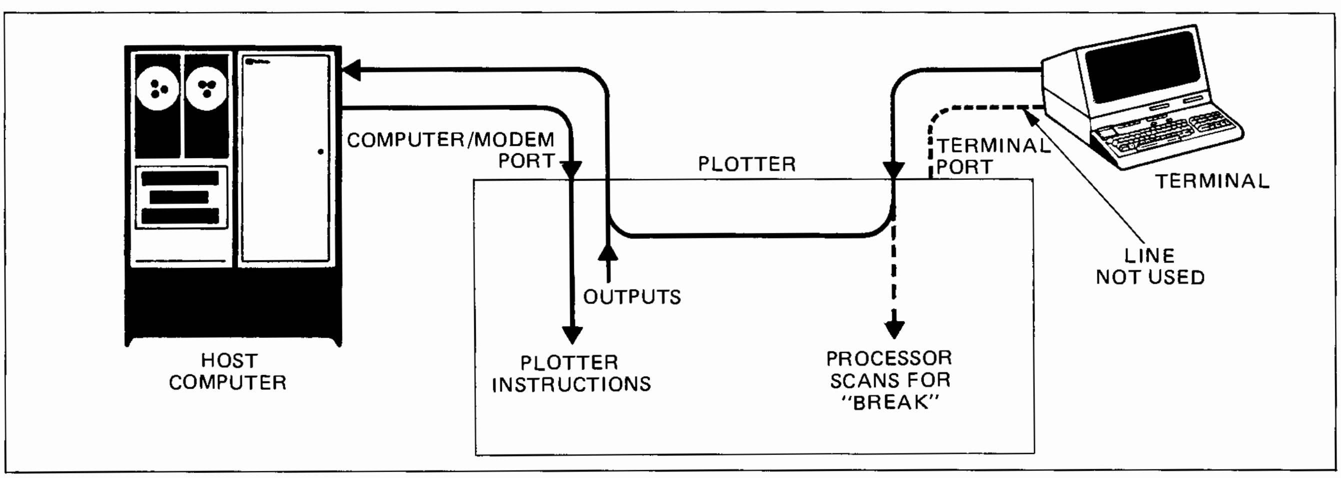
Normal, Programmed-On Mode

In this mode, the plotter operates in response to instructions received from the computer as shown in the following diagram. When the plotter instructions request output, it is provided as shown. The communication channel from the terminal to the computer, through the plotter, is maintained to provide operator entry into the computer.

The plotter's processor monitors the channel from the terminal to the computer for a terminal-generated break signal. If the **STAND ALONE/EAVESDROP** switch is set to **EAVESDROP,** the plotter interprets this break signal as an instruction to flush the graphics buffer and return the plotter to the normal, programmed-off mode. A break signal of equal duration is retransmitted to the computer. If **STAND ALONE** is selected, the break signal is ignored.

It should be noted that simultaneous transmission of plotter outputs and terminal-generated data will result in garbled data transmission to the computer.

Plotter in Normal, Programmed-On Mode



Monitor Mode

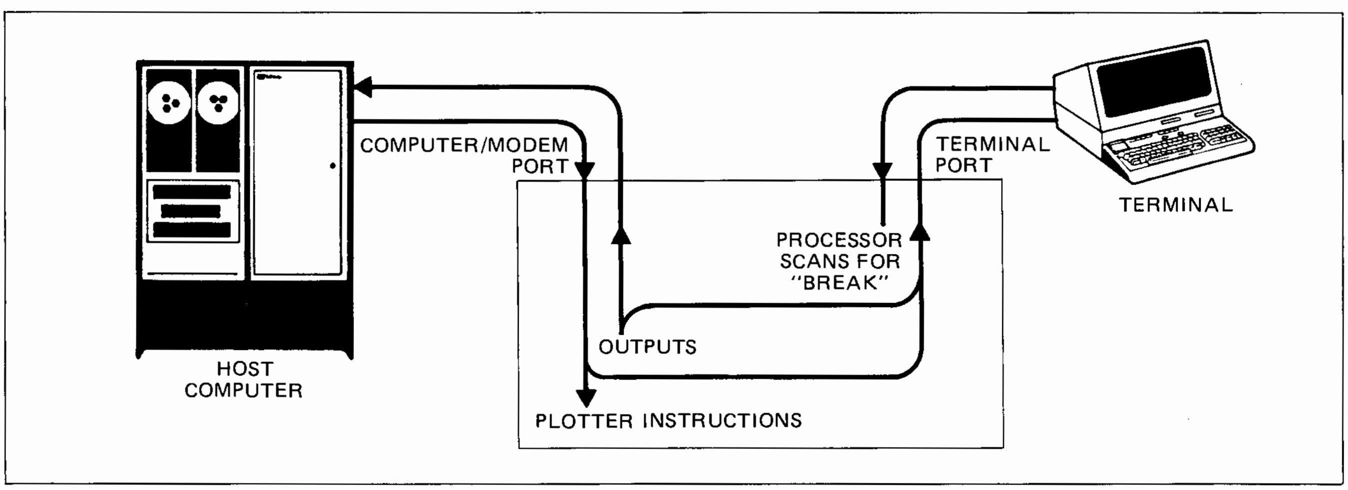
Two mutually exclusive monitor modes may be enabled when the plotter is in the normal, programmed-on mode. These monitor modes are primarily intended to be used as a "debugging" aid for program development and are available with the **STAND ALONE/EAVESDROP** switch set to either position.

Monitor mode 1 causes HP-GL instructions to be retransmitted to a terminal, via the **TERMINAL** connector, as they are parsed from the plotter's buffer. If the **MONITOR MODE/NORMAL** switch is set to **MONITOR MODE,** monitor mode 1 is enabled when **CHART HOLD** is invoked. If the switch is set to **NORMAL** when **CHART HOLD** is invoked, monitor mode 1 or monitor mode 2 can be enabled using either the Set Plotter Configuration Instruction, ESC . @, or the Set Monitor Mode Instruction, ESC . Q. If monitor mode 1 is manually enabled, it cannot be disabled using either instruction. However, both instructions can be used to toggle between monitor mode 1 and monitor mode 2. Monitor mode 2 causes both HP-GL and device control instructions to be retransmitted to the terminal as they are received by the plotter.

When monitor mode 2 is enabled, routing of data is a function of the **DUPLEX** switch as shown in the following diagrams. If the **DUPLEX** switch is set to **FULL,** all plotter outputs are sent to the computer, and if the computer is working in an echo-plex environment, each of the plotter outputs is then echoed from the computer, through the plotter, to the terminal. If the **DUPLEX** switch is set to **HALF,** all plotter outputs are sent to both the computer and terminal. When monitor mode 1 is enabled, the **DUPLEX** switch position has no effect on how data is routed. However, data routing in monitor mode 1 is the same as shown for monitor mode 2 with the **DUPLEX** switch set to **HALF.**

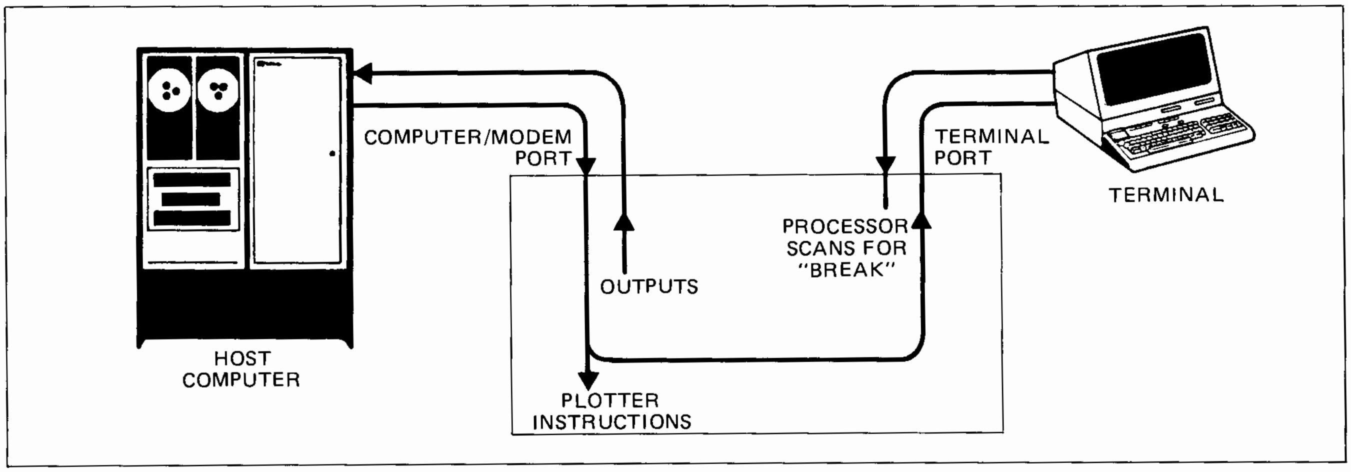
While in monitor mode 1 or 2, the communications channel between the terminal and computer is disconnected to assure that the terminal will not take part in any handshake process between the plotter and the computer. With the exception of the break signal, all terminal-generated data is ignored by the plotter. If the **STAND ALONE/EAVESDROP** switch is set to **EAVESDROP,** the plotter interprets a terminal-generated break signal as an instruction to flush the graphics buffer and return the plotter to the normal, programmed-off mode. If **STAND ALONE** is selected, the break signal is ignored. Refer to The Set Plotter Configuration Instruction, ESC . @, for additional information.

Plotter in Monitor Mode 1 or Half-Duplex Monitor Mode 2



RS-232-C Interface

Plotter in Full-Duplex Monitor Mode 2

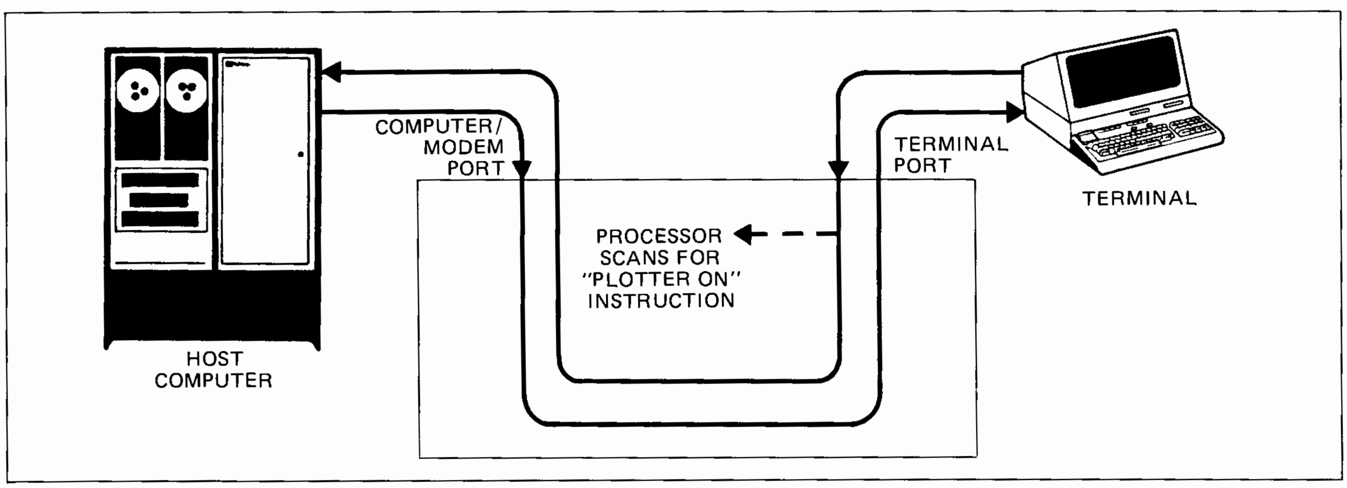


Local, Programmed-Off Mode

The plotter can be in this mode only if the **STAND ALONE/EAVESDROP** switch is set to **EAVESDROP** and the **LOCAL/NORMAL** switch is set to **LOCAL.** This mode permits communication between a terminal and computer, but the plotter will respond only to a plotter on instruction, ESC . (or ESC . Y, from the terminal or from the front-panel **BYPASS** pushbutton. The **DUPLEX** switch is not functional in this mode and terminal-generated data is displayed on the terminal only if the data is echoed from a host computer. This mode is normally used where the host computer must service many terminal modes and it is desirable to down-load to a smart-terminal to reduce computer-connect time. Data routing in this mode is shown in the following diagram.

RS-232-C Interface

Plotter in Local, Programmed-Off Mode



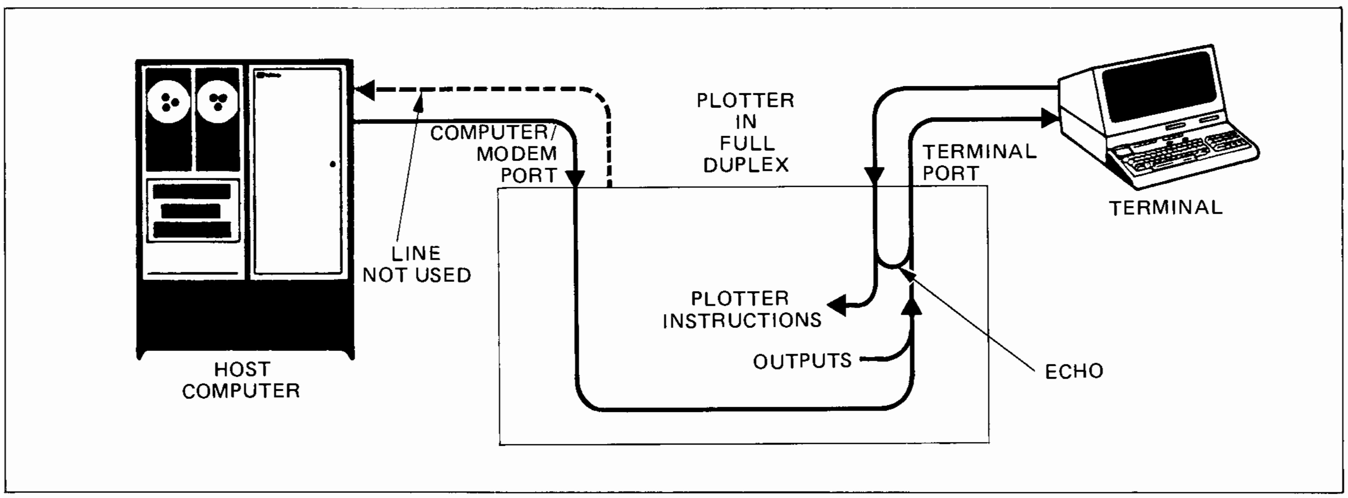
Local, Programmed-On Mode

In this mode, the plotter operates in response to instructions received from the terminal as shown in the following diagrams. When the plotter instructions request output, the output is sent only to the terminal as shown. The plotter **DUPLEX** switch is operational in this mode. In **FULL** duplex the plotter instructions are echoed back to the terminal, and in **HALF** duplex the echo is suppressed.

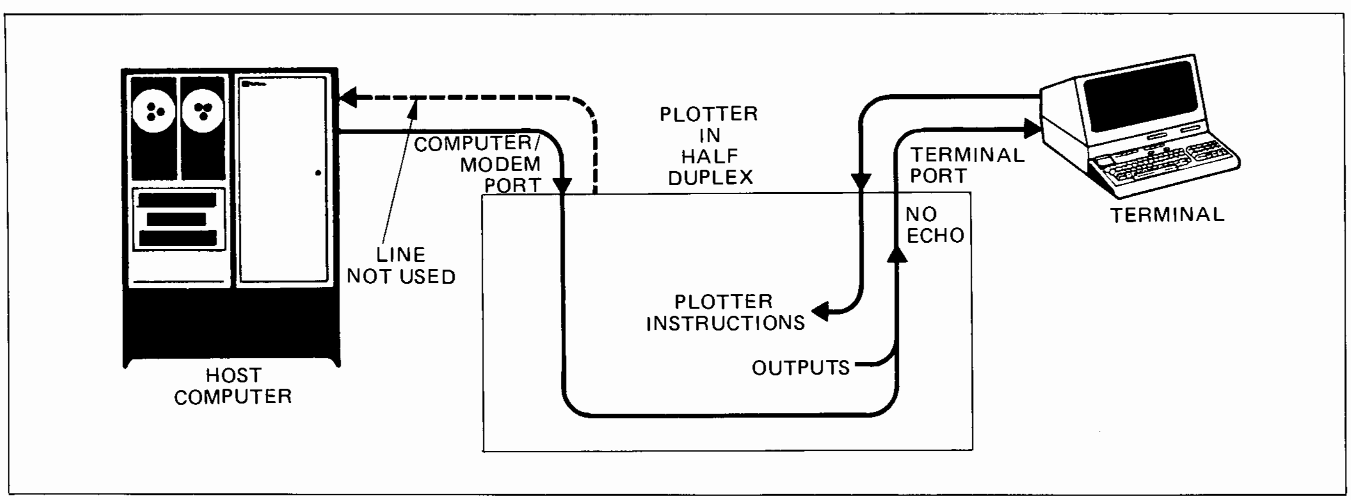
If the **STAND ALONE/EAVESDROP** switch is set to **EAVESDROP,** the plotter interprets a terminal-generated break signal as an instruction to flush the graphics buffer and return the plotter to the local, programmed-off mode. If **STAND ALONE** is selected, the break signal is ignored.

It should be noted that any instruction from the computer is passed through the plotter to the terminal, and may be confused with echoed terminal instructions and/or plotter outputs.

Plotter in Local, Programmed-On Mode (Full Duplex)



Plotter in Local, Programmed-On Mode (Half Duplex)



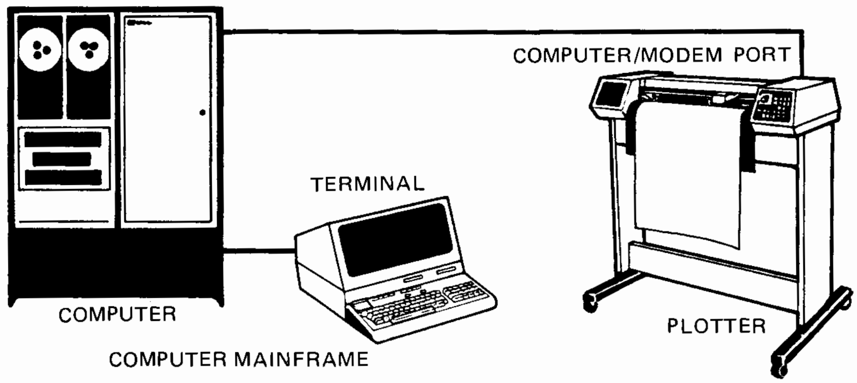
RS-232-C Interface

Stand Alone Environment

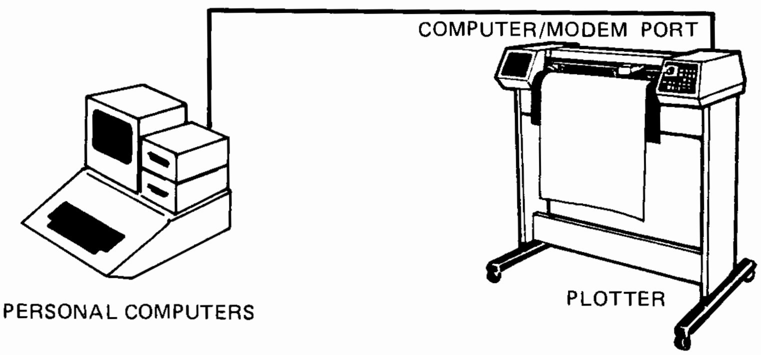
STAND ALONE INSTALLATION

In the stand alone environment, the plotter is connected only to a computer and is usually adjacent to it. Entry to the computer is by a keyboard or terminal through a separate port, rather than through the plotter. Diagrams of this type of system for both large computer mainframes and personal computers are as follows:

Plotter Interconnection with a Computer Mainframe or Personal Computer



OR



STAND ALONE OPERATION

Operation in the stand alone environment is limited to the normal, programmed-on and normal, programmed-off operating modes described under the eavesdrop environment. The only difference is that a terminal is not connected. The **STAND ALONE/EAVESDROP** switch is normally set to **STAND ALONE** to assure compatibility with HP 7580 and HP 7585 plotters that are equipped with the Option 001 (RS-232-C) interface.

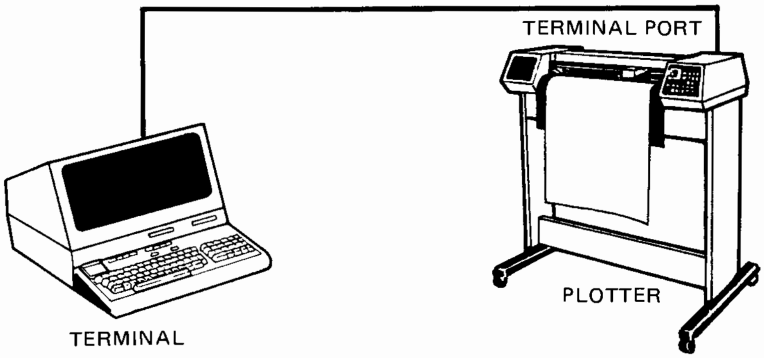
NOTE: When three-wire data communication is desired, bit 1 of the second parameter of the ESC . @ instruction must be set to 1 to ensure full compatibility with plotters having the Option 001 (RS-232-C) interface. ■

Local Environment

LOCAL INSTALLATION

In the local environment, the plotter is connected only to a terminal and is usually adjacent to it. A diagram of this type system is as follows:

Plotter Interconnection with Only a Terminal



LOCAL OPERATION

Operation in the local environment is limited to the local operating modes described under the eavesdrop environment. The only difference is that a computer is not connected.

Interface Wiring Allocations

The plotter interfaces to the RS-232-C communications lines through two standard 25-pin female connectors. The plotter is compatible with EIA RS-232-C and CCITT V.24 full duplex protocol and is also capable of data communications in a three-wire (transmit, receive, ground) configuration. Specifications for DTE (Data Terminal Equipment) are met when the male-to-male interface cable (HP P/N 17355A), supplied with the plotter, is attached. With this cable in place, any piece of DCE (Data Computer Equipment) that meets the standard will directly interface to the cable attached to the plotter. Note that only seven pins are connected to the plotter's internal circuitry. All remaining pins are hardwired directly between the **COMPUTER/MODEM** and **TERMINAL** connectors.

Connector pin allocations for the three-wire configuration are identified and described in the following table.

Minimum Interface Connector Pin Allocations

Pin No.	RS-232-C	CCITT V.24	Function/Signal Level
COMPUTER/MODEM Connector			
2	BA (TDATA)	103	Data line from plotter High = SPACE = "0" = +12 V Low = MARK = "1" = −12 V
3	BB (RDATA)	104	Data line to plotter High = SPACE = "0" = +3 V to +25 V Low = MARK = "1" = −3 V to −25 V
7	AB (SGND)	102	Signal ground (Return line)

Minimum Interface Connector Pin Allocations *(Continued)*

Pin No.	RS-232-C	CCITT V.24	Function/Signal Level
		TERMINAL Connector	
2	BA (TDATA)	103	Data line to plotter High = SPACE = "0" = +3 V to +25 V Low = MARK = "1" = −3 V to −25 V
3	BB (RDATA)	104	Data line from plotter High = SPACE = "0" = +12 V Low = MARK = "1" = −12 V
7	AB (SGND)	102	Signal ground (Return line)

In addition to the minimum requirements for communication, six more lines are internally connected at the **COMPUTER/MODEM** and **TERMINAL** connectors as shown in the following table. These lines are required to implement full duplex communication and hardwire handshake mode.

Pin 23, the Data Signal Rate Selector, chooses the higher baud rate if two are available.

Additional Connector Pin Allocations

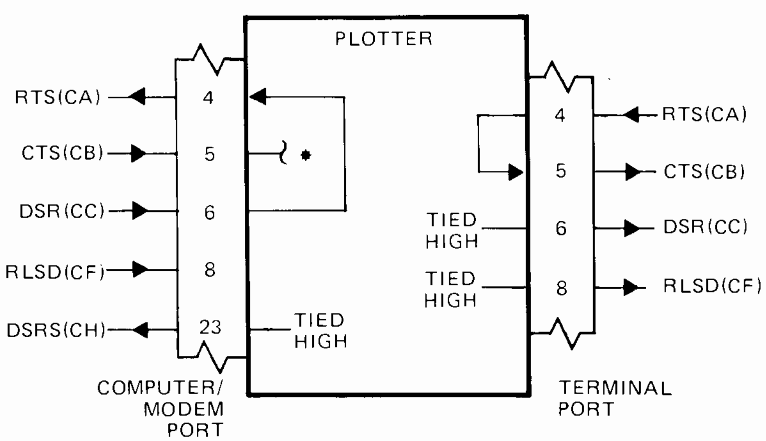
Pin No.	RS-232-C	CCITT V.24	Function/Signal Level
		COMPUTER/MODEM Connector	
4	CA	105	Request To Send from the plotter High = ON = +12 V Low = OFF = −12 V
5	CB	106	Clear To Send to the plotter High = ON = +3 V to +25 V Low = OFF = −3 V to −25 V
6	CC	107	Data Set Ready to the plotter High = ON = +3 V to +25 V Low = OFF = −3 V to −25 V
20	CD	108.2	Data Terminal Ready to computer/modem High = ON = +12 V Low = OFF = −12 V
23	CH	111	Data Signal Rate Selector Tied High = ON = +12 V

Pin No.	RS-232-C	CCITT V.24	Function/Signal Level
		TERMINAL Connector	
4	CA	105	Request To Send to the plotter High = ON = +3 V to +25 V Low = OFF = −3 V to −25 V
5	CB	106	Clear To Send to the terminal High = ON = +12 V Low = OFF = −12 V
6	CC	107	Data Set Ready from the plotter High = ON = +12 V Low = OFF = −12 V
8	CF	109	Received Line Signal Detector from plotter Tied High = ON = +12 V
20	CD	108.2	Data Terminal Ready to computer/modem High = ON = +3 V to +25 V Low = OFF = −3 V to −25 V

Additional Connector Pin Allocations *(Continued)*

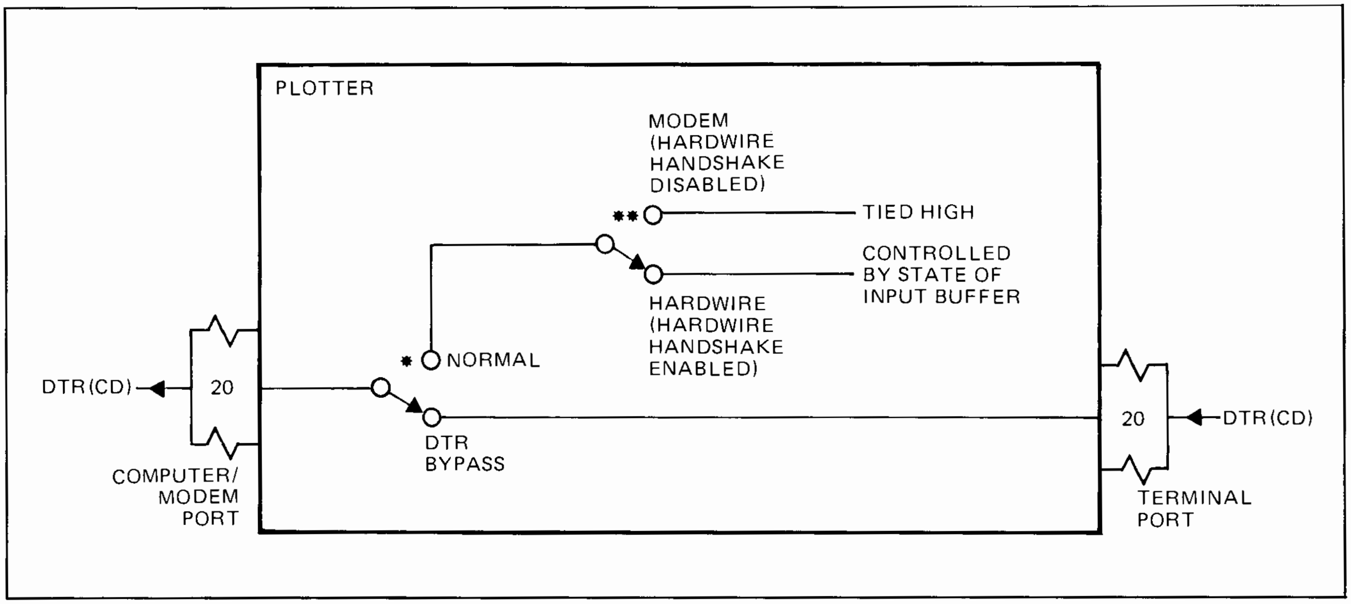
The following diagram shows signal line routing in the power-on modes and a functional representation of how the computer DTR signal line is controlled.

Signal Line Routing in All Power-On Modes



*If full duplex communication is enabled, data can only be transmitted to the computer when both DSR and CTS are set high by the computer or modem. Full duplex communication is enabled at power-on when the **HARDWIRE/MODEM** switch is set to **MODEM**, or it can be enabled after power-on using the ESC . @ instruction.

Functional Representation of DTR Signal Line Control



*__DTR BYPASS__ position is used only in a modem environment and precludes the use of hardwire handshake mode.

**The switch position at power-on determines the bit states of the second parameter in the ESC . @ instruction. After power-on, the switch position may be overridden using the ESC . @ instruction.

Stop Bit Jumper Connections

When shipped from the factory, the plotter is configured to automatically verify or generate two stop bits at the 150 and lower baud rates and one stop bit for all baud rates greater than 150. If two stop bits are required at all baud rates, a jumper can be installed on the I/O board located inside the plotter. Information on how to access this I/O board and install the jumper can be found in the HP 7580B Drafting Plotter Service Manual, Part Number 07580-90022, the HP 7585B Drafting Plotter Service Manual, Part Number 07585-90002, or the HP 7586B Drafting Plotter Service Manual, Part Number 07586-90000.

Transmission Errors

Transmission errors occur when communication between the computer and plotter is incomplete or does not conform to what is expected or required by either party.

Transmission errors include:

- Framing error — The plotter does not detect a valid stop bit at the end of every character.
- Parity error — The plotter does not detect the expected parity (odd or even). This error is inhibited if the plotter **PARITY** switch is set to **OFF**.
- Overrun error — A character writes over another character before it is placed in the buffer or processed.
- Buffer overflow error — The plotter receives more bytes of data than it has space for in the buffer.

When the plotter detects a framing, parity, or overrun error, it turns on the front-panel **ERROR** light and sets error code 15. This error code generally indicates that the communication incompatibility is hardware related (incorrect stop bit jumper installation, wrong parity selection, incompatible or incorrectly set baud rates, etc.).

When the plotter detects a buffer overflow, it turns on the front-panel **ERROR** light and sets code 16. The last HP-GL data that caused the overflow will be lost. Error code 16 generally indicates an improperly established handshake protocol.

The **ERROR** light remains on until one of these conditions occurs:

- The plotter processes an HP-GL Initialization Instruction, IN.
- The user interrogates the plotter via an Output Extended Error command, ESC . E, and the plotter responds with the appropriate error code.
- The user turns the plotter off.
- The user invokes a front-panel **RESET.**

A complete list of RS-232-C error codes is included with the discussion of the ESC . E instruction.

NOTE: A buffer overflow condition may also cause an HP-GL error to occur. In this case, the user must also instruct the plotter to report the HP-GL error, via an HP-GL OE instruction, in order to clear the **ERROR** light. See Chapter 9 for an explanation of The Output Error Instruction, OE. ■

Handshaking

The plotter uses an input buffer to synchronize the processing of data with the rate at which it is received. The presence of an input buffer requires that the computer and the plotter transfer information to one another in such a way that data will not be lost or misinterpreted. This is the purpose of handshaking.

The plotter is capable of using any one of four handshaking methods to prevent buffer overflow and the resulting loss of data. The computer system's capabilities and requirements dictate which handshake method is appropriate.

- Hardwire Handshake — uses a physical wire, pin 20 of the RS-232-C cable to control handshaking. It can be used if the computer system can or does monitor pin 20 (DTR).
- Xon-Xoff Handshake — is managed by the peripheral device. It can be used if the computer system follows an Xon-Xoff protocol (control characters are transmitted from the peripheral to the computer).
- Enquire/Acknowledge Handshake — is managed by the computer system and interface. This handshake is often used in Hewlett-Packard systems and is so named because the ASCII characters ENQ and ACK may be used to control the handshake.
- Software Checking Handshake — is managed by the applications programmer. It can be used on almost any computer system, but it must be used if the system cannot implement any of the other three handshaking methods.

Once the handshake method is selected, the plotter can be programmatically instructed to match the computer system requirements, implement the chosen handshake method, and function properly within the system-dependent communication environment. This is done by specifying certain variables in device control commands which are issued to the plotter at the beginning of each computer session or graphics program. The variables which may be specified through the four handshake methods available to the plotter are:

- Output Trigger Character — The output trigger character, when used, is the last character output by the computer when making a request of a graphics peripheral. Defining this character in a command tells the plotter, "Don't respond to my request until you receive this trigger character." This character is often a DC1 (decimal equivalent 17) or some other nonprinting ASCII character such as LF or CR or, when using some implementations of BASIC, the ? (decimal equivalent 63), which does print.
- Turnaround Delay — The turnaround delay is the length of time the plotter will wait after receiving a computer request before it responds. The purpose of this time delay is to delay the plotter's transmission of requested data until the computer is ready to receive and process it. Systems may require either a turnaround delay or a trigger character, or both.
- Output Initiator Character — The output initiator character is a one-character initiator that is sent by the plotter at the beginning of a string. The output initiator tells the computer, "This starts my transmission." Some computers which require an output initiator expect the start-of-text character, STX, (decimal equivalent 2), as the plotter's output initiator.
- Output Terminator — The output terminator is a one- or two-character terminator that the computer requires the plotter to send at the end of each response to a data request. The output terminator tells the computer, "This completes my transmission." Often, computers expect the carriage return character CR (decimal equivalent 13) as the plotter's output terminator.
- Echo Terminate Character — Echoing is commonly found in full-duplex systems. Use of the echo terminate parameter in a device control command tells the plotter that the computer will echo all responses and that this echoed data should be ignored (the plotter's data buffer should be closed) until an echo terminate character is received. When the plotter receives the echo terminate character, it reopens the data buffer to receive graphics data from the computer. Computers often use the line feed character LF (decimal equivalent 10) as the echo terminator. If the computer does not echo the peripheral's response, this variable must be a null character (decimal equivalent 0) or must be omitted.
- Intercharacter Delay — Some computers cannot process data as fast as the plotter can transmit it due to limited buffering in the I/O port. This can be compensated for by delaying each transmission from the plotter a period of time as specified by the intercharacter delay variable. This intercharacter delay is added to a turnaround delay (if one has been specified) before the first character is sent by the plotter, and is also inserted before each subsequent character in a string being sent to the computer.

- Enquiry Character — In some systems the computer sends an enquiry character to ask the plotter if it has room for a block of data, thereby initiating the handshake process. If Xon-Xoff handshake mode is to be established, a NULL character (decimal equivalent 0) must be specified as the enquiry character. If enquire/acknowledge is to be established, an ENQ character (decimal equivalent 5) or any other ASCII character besides the NULL is used.
- Immediate Response String — Certain system environments require an immediate response from the plotter acknowledging the enquiry from the computer. Systems of this type include a computer that transmits data to the plotter after a certain time interval but before receiving a go ahead signal from the plotter. If the plotter's buffer is full and the computer sends more data, the buffer will overflow. The immediate response string prevents this inadvertent transmission of data before the plotter is ready. It is transmitted by the plotter immediately after receipt of an enquiry character and tells the computer, "Wait, I am here and checking my buffer space." The computer will wait indefinitely until it receives an acknowledgment string. Computers frequently require a DC3 character (decimal equivalent 19) for the immediate response.
- Acknowledgment String — The acknowledgment string specifies the character or characters that the plotter will send to the computer when the plotter's input buffer has room for another block of data. Computers frequently require that the ACK character (decimal equivalent 6) be used for the acknowledgment string.
- Maximum Buffer Size — This variable establishes the effective usable buffer size in the plotter.
- Data Block Size — This is the maximum size of each data block the computer will transmit to the plotter.
- Data Terminal Ready (CD) Line Control — This variable sets the configuration of the plotter's Data Terminal Ready control line (pin 20) to enable or disable the hardwire handshake mode. Pin 20 is held on (+12 V) if hardwire handshake is disabled.
- Xoff Threshold Level — In the Xon-Xoff handshake mode, this defines how many empty bytes remain in the buffer when the plotter sends the Xoff trigger character to the computer, telling it to stop sending data. To maintain maximum throughput, the user should specify the smallest possible value that ensures the buffer does not overflow.
- Xoff Trigger Character — This specifies the character string the plotter will use to signal the computer to temporarily stop sending data while the plotter processes what it has already received. The DC3 character (decimal equivalent 19) is generally used for the Xoff trigger.
- Xon Trigger Character — This specifies the character string the plotter will use to signal the computer that there is sufficient space in the buffer to resume sending data. The DC1 character (decimal equivalent 17) is generally used for the Xon trigger.

The following discussion of the four handshake methods includes the pertinent variables and identifies the commands which will establish their values.

Software Checking

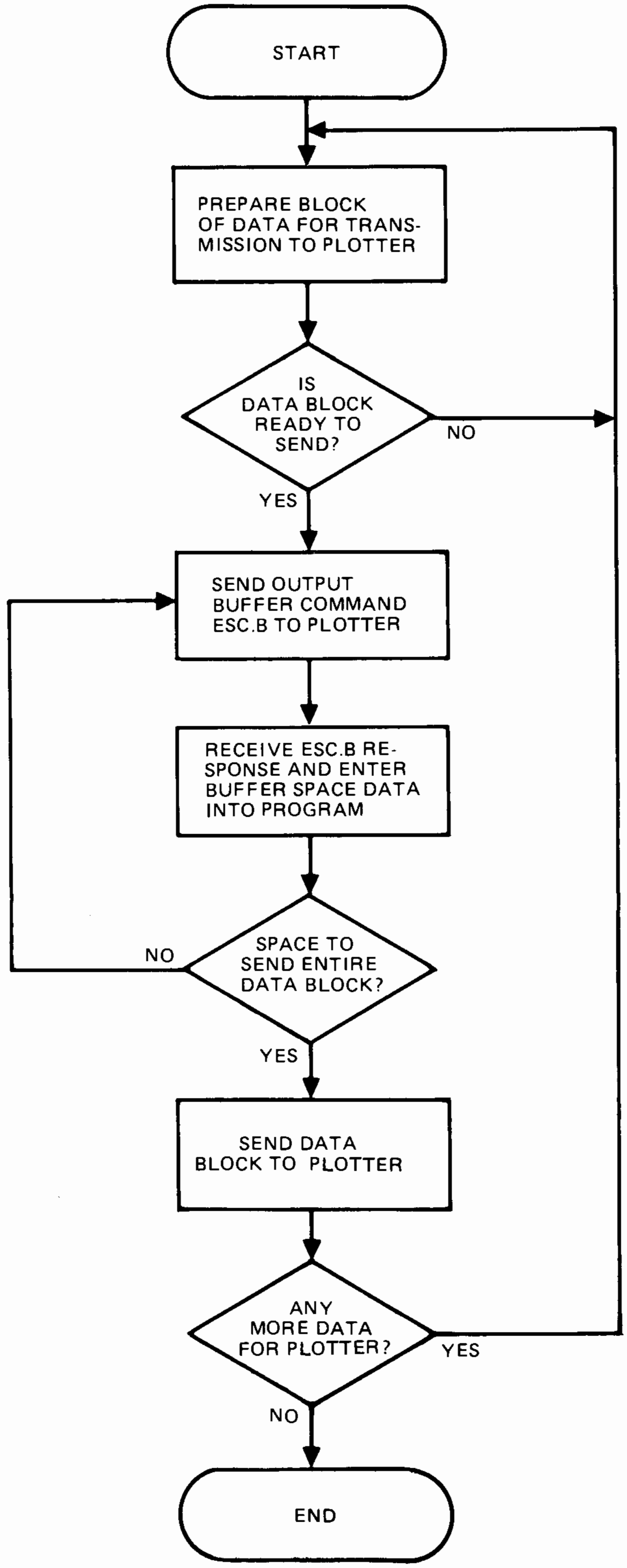
Software checking is a nonautomatic handshake method in which the user's program repeatedly asks the plotter how many characters of empty space remain in the buffer. When the plotter response is bigger than the next block of data, the program will transmit the data block to the plotter. This method is inefficient in time-share environments.

The advantage of software checking is that it is independent of hardware and operating system abilities required to implement other handshake modes; therefore, it usually makes software transportable between computer systems. The limitation of this method of handshaking is that it uses up computer time.

To match the requirements of the computer system, these variables may be specified for the software checking handshake mode by using the appropriate instruction:

- Maximum buffer size (ESC . @ and ESC . T instructions)
- Turnaround delay (ESC . M instruction)
- Output trigger character (ESC . M instruction)
- Echo terminate character (ESC . M instruction)
- Output initiator character (ESC . M instruction)
- Output terminator (ESC . M instruction)
- Intercharacter delay (ESC . N instruction)

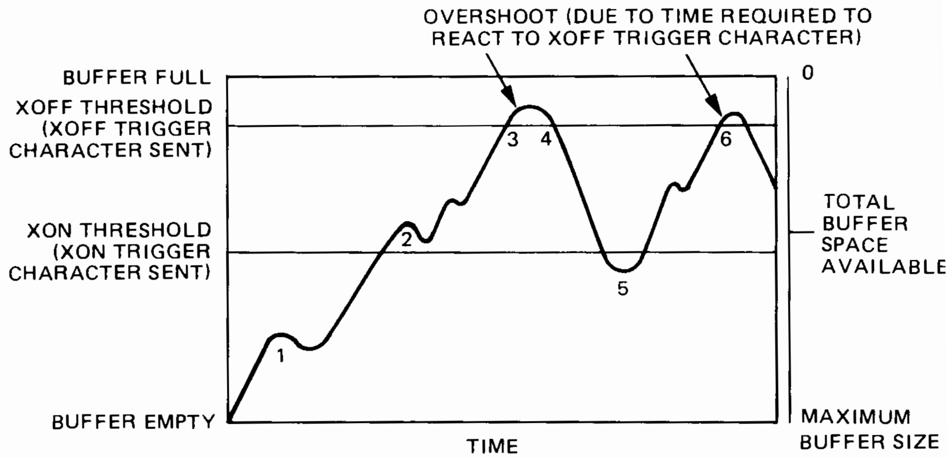
The following flow diagram illustrates the functional elements of a typical software checking handshake mode within a user's program.


START
PREPARE BLOCK OF DATA FOR TRANSMISSION TO PLOTTER
IS DATA BLOCK READY TO SEND?
NO
YES
SEND OUTPUT BUFFER COMMAND ESC.B TO PLOTTER
RECEIVE ESC.B RESPONSE AND ENTER BUFFER SPACE DATA INTO PROGRAM
SPACE TO SEND ENTIRE DATA BLOCK?
NO
YES
SEND DATA BLOCK TO PLOTTER
ANY MORE DATA FOR PLOTTER?
YES
NO
END

Xon-Xoff Handshake

With the Xon-Xoff handshake method, the plotter controls the data exchange sequence by telling the computer when it has room in its buffer for data and when to shut off the flow. The plotter uses buffer threshold indicators (an Xon trigger character and an Xoff trigger character) to prevent buffer overflow.

Xon-Xoff Threshold Levels



As data is sent to the plotter by the computer, it is stored in the buffer and simultaneously acted on by the plotter. The preceding figure is representative of the way the Xon-Xoff handshake works; the numbers represent the following:

1. Data enters the buffer faster than it can be acted on by the plotter, and the buffer starts to fill.
2. The plotter begins processing the input data faster than the computer sends it, and the buffer starts to empty.
3. The data enters the buffer at a faster rate than the plotter can process. The amount of data stored in the buffer reaches the Xoff threshold level, at which point the plotter sends the Xoff trigger character stopping the flow of data from the computer.
4. Due to a finite delay between the time the plotter sends the Xoff trigger character and the time it takes the computer to react, a slight overshoot may occur. For this reason, the Xoff threshold level should always be specified at least as large as the data block size to allow room for this overshoot.
5. When the amount of stored data drops to the Xon threshold level, the plotter sends the Xon trigger character to signal the computer to resume sending data. The Xon threshold level is automatically set at 50% of the total buffer size.
6. Data is again stored in the buffer until all data is transferred or until the Xoff threshold level is exceeded again.

The following conditions can be specified for the Xon-Xoff handshake mode to match the requirements of the computer system, by using the appropriate instruction:

- Xon trigger character (ESC . I instruction)
- Xoff trigger character (ESC . N instruction)

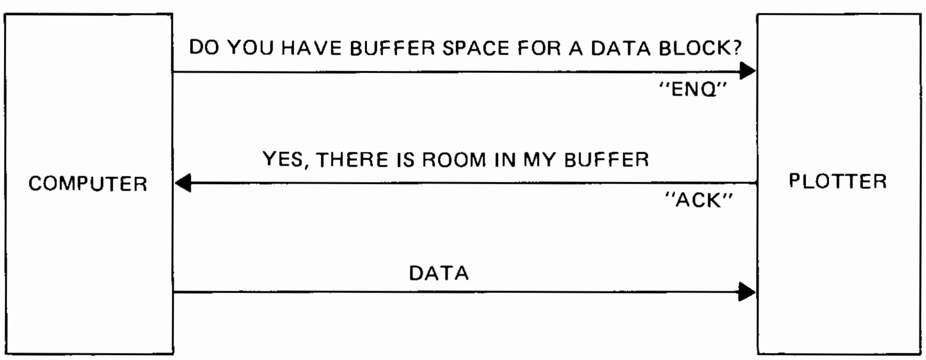
- Xoff threshold level (ESC . I instruction)
- Intercharacter delay (ESC . N instruction)
- Maximum buffer size (ESC . @ and ESC . T instructions)

Enquire/Acknowledge Handshake

With the enquire/acknowledge handshake, the computer initiates the data exchange process by querying the plotter about the availability of buffer space. The format of the exchange is dependent upon the requirements of the computer. The following conditions can be specified for the enquire/acknowledge handshake mode by using the appropriate instruction:

- Turnaround delay (ESC . M instruction)
- Output trigger character (ESC . M instruction)
- Echo terminate character (ESC . M instruction)
- Output initiator character (ESC . M instruction)
- Output terminator (ESC . M instruction)
- Intercharacter delay (ESC . N instruction)
- Immediate response string (ESC . N instruction)
- Data block size (ESC . I or ESC . H instruction)
- Enquiry character (ESC . I or ESC . H instruction)
- Acknowledgment string (ESC . I or ESC . H instruction)
- Maximum buffer size (ESC . @ and ESC . T instructions)

In its simplest form, the data exchange looks like this:



ENQ/ACK Handshake Protocol Example 1

In a more complex form, the communication might look like the following example, where the two instructions **ESC** . M 250 ; 17 ; 10 ; 13 : and **ESC** . H 100 ; 5 ; 6 : have been sent to specify the variables as:

Turnaround delay = 250 ms

Output trigger character = DC1 (ASCII value 17)

Echo terminate character = LF (ASCII value 10)

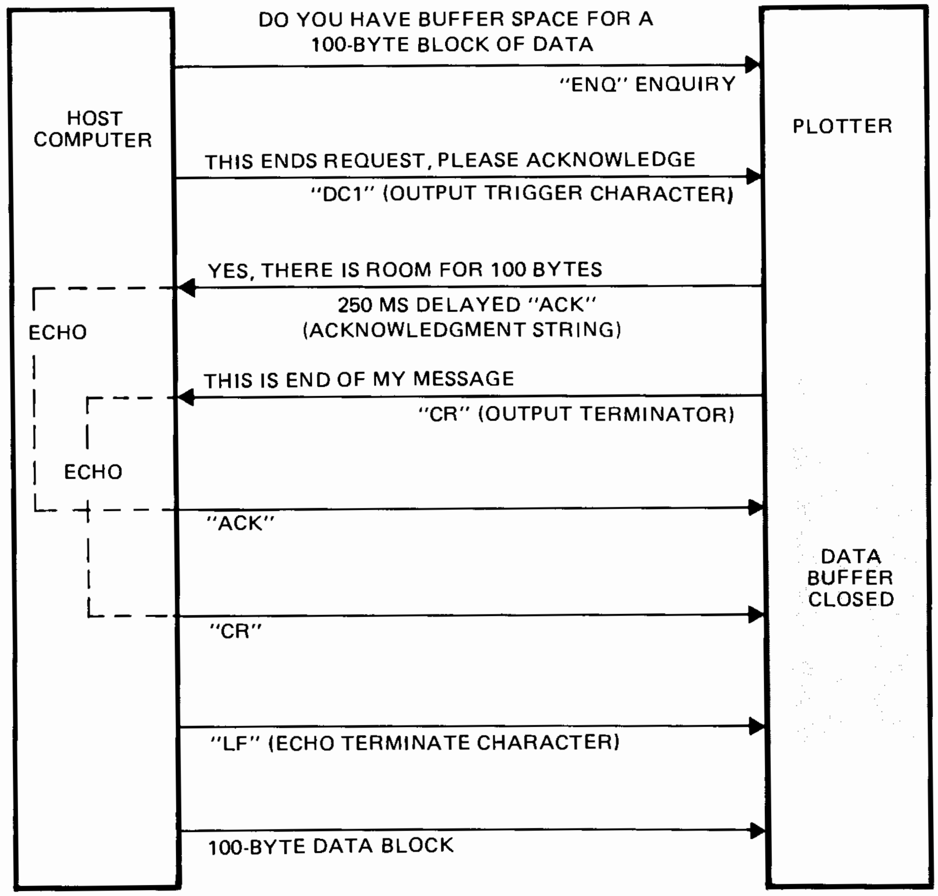
Output terminator = CR (ASCII value 13)

Data block size = 100 bytes

Enquiry character = ENQ (ASCII value 5)

Acknowledgment string = ACK (ASCII value 6)

ENQ/ACK Handshake Protocol Example 2



Hardwire Handshake

As the name implies, the hardwire handshake takes place in the hardware rather than the firmware or software. The plotter controls the data exchange sequences by setting the electrical voltage on pin 20 of the connector (CD line) to the computer to signal the computer when to send another block of data. The Data Terminal Ready (DTR or CD) is set high at power-on and is not set low until available buffer space is less than the data block size specified by either ESC . H or ESC . I (default is 80 bytes). DTR remains low until the buffer is half empty or, if the data block size exceeds half the buffer's capacity, until available buffer space is one byte greater than the data block size. By monitoring this line, the computer knows when it can or cannot safely transmit another block of data.

The following conditions may be specified for the hardwire handshake mode by using the appropriate instruction:

- Maximum buffer size (ESC . @ and ESC . T instructions)
- Data Terminal Ready (CD) line control (ESC . @ instruction)
- Data block size (ESC . I or ESC . H instruction)

Automatic Disconnect Modes

Two modes are available for automatic disconnection at the end of an RS-232-C/CCITT V.24 session conducted over phone lines. These modes achieve disconnection with no one present to manually hang up the phone.

For either disconnect mode to work, the **DTR-BYPASS/NORMAL** switch on the rear panel must be set to **NORMAL.** Neither disconnect mode is enabled during the normal power-up state.* An explanation of each mode is contained in the following paragraphs.

1. *The switched/datex-line disconnect mode.* Depress the pen **5** button during power-up to activate this mode.

   The CTS and DSR (CB and CC) lines control the DTR line and plotter outputs. As long as the CTS and DSR lines are high, the DTR line is high. When either the CTS or DSR lines goes low, the DTR goes low, causing a disconnect. Plotter output will not be sent to the modem, but will be buffered until both CTS and DSR are high. Overflowing this buffer causes I/O error 17. Input from the modem to the plotter is ignored until both lines are high.

2. *The leased-line disconnect mode.* Depress the pen **6** button during power-up to activate this mode.

   The CTS, DSR, and DCD (CB, CC, and CF) lines control the DTR line and plotter output. conditions are the same as for the switched/datex disconnect mode. If any of the three lines goes low, the DTR line goes low, causing a disconnect. Plotter output will not be sent to the modem, but will be buffered until all three lines are high. Overflowing this buffer causes I/O error 17. Input from the modem to the plotter is ignored until all lines are high.

The only way to leave either mode is to cycle the power. Bits 0 and 1 of the second parameter of the ESC . @ instruction (and therefore the setting of the **HARDWIRE/MODEM** switch) are ignored until leaving these modes.

RS-232-C Interface

RS-232-C Device Control Instructions

Device control instructions establish the handshake protocol to be used by the plotter. All communications conform to the protocol established by these instructions. The instructions serve two purposes: to control the format by which data is transferred between the computer and the plotter (input/output operations), and to give the computer the ability to query and to receive information from the plotter.

Each instruction's name gives an immediate clue to its purpose: if "output" is the first word in the name of the instruction, the computer wants a response from the plotter. Otherwise, the instruction concerns the I/O functions. The word "set" in the title indicates the command establishes conditions under which subsequent I/O is to occur.

The plotter acts on device control instructions immediately upon receipt. It does not store them in the data buffer.

*The DTR line (CD) of the modem connector is controlled by the setting of bit 0 of the second parameter of the ESC . @ instruction. Transmission of plotter output instructions is controlled by the setting of bit 1. The power-up values of these bits are set by the **HARDWIRE/MODEM** rear-panel switch.

Syntax for Device Control Instructions

Device control instructions are three-character escape code sequences comprised of "ESC" and "." followed by one of the characters @, A, B, E, H, I, J, K, L, M, N, O, P, Q, R, S, T,), (, Y, or Z.

These syntax conventions are used with the instructions discussed in this chapter:

[ ]	Brackets indicate that all parameters enclosed are optional.
( )	Parentheses indicate that each individual parameter is optional.
;	The semicolon follows and delimits parameters. If a semicolon appears without a parameter, the parameter is defaulted.
:	The colon terminates any instruction which may have parameters and can occur after any valid number of parameter entries. Any parameter that is not specified is defaulted.
<DEC>	This symbol specifies a decimal value parameter. For example, the characters 10 would represent the decimal value ten; the characters 13 would represent the decimal value thirteen.
<ASC>	This symbol specifies the decimal equivalent for an ASCII character (see the ASCII Character Equivalents table in Appendix C). In this case, the characters 10 would represent the ASCII line feed character, LF, and 13 would represent the ASCII carriage return character, CR.
. . .	Specifies any number of optional parameters. Each parameter must be followed by a delimiter (;) or the terminator (:).
[TERM]	Unless changed by an ESC . M instruction, all RS-232-C output responses include a CR as a terminator.
Default Values; Omitting Parameters	Any parameter may be omitted or, if the parameter is required, it can be set to its default value by omitting the parameter and entering only the semicolon delimiter. All parameters may be omitted and therefore set to default values by entering only the colon terminator after the instruction.
ESC	Denotes the single ASCII character, Escape, which in most computers is accessed by striking a single key on the keyboard.

NOTE: There is no delimiter (semicolon) between the three-character command sequence, e.g., **ESC** . M, and the first parameter. ■

RS-232-C Interface

The Plotter-On Instruction, ESC . (or ESC . Y

USES: The ESC . (or ESC . Y instruction enables the plotter to accept data and interpret it as HP-GL or device control instructions. Use the instruction to establish programmed-on operation in either the normal or local mode.

SYNTAX: ESC . (

or

ESC . Y

EXPLANATION: Depending on the selected operating mode, the plotter scans for a plotter on instruction from either the computer or the terminal. After receipt of this instruction from the selected source, the plotter is programmed-on and will interpret all subsequent data from the selected source as plotter instructions. If the plotter is already programmed-on, it will ignore this instruction. Refer to Plotter Environments in this chapter.

The Plotter-Off Instruction, ESC .) or ESC . Z

USES: The ESC .) or ESC . Z instruction disables the plotter so that it will only accept a plotter-on instruction. Use the instruction to establish programmed-off operation in either the normal or local mode.

SYNTAX: ESC .)

or

ESC . Z

EXPLANATION: After a plotter off instruction is received, the plotter is programmed-off. Any HP-GL instructions remaining in the buffer are executed, but no additional instructions will be accepted until a plotter on instruction is received. If the plotter is already programmed-off, it will ignore this instruction.

If **EAVESDROP** is selected, a break signal from the terminal will have the same effect as a plotter-off instruction. If **STAND ALONE** is selected, the plotter cannot be programmed-off.

The Set Plotter Configuration Instruction, ESC . @

USES: The ESC . @ instruction sets an effective logical I/O buffer size and controls hardwire handshake, communications protocol, two mutually exclusive monitor modes, and block I/O error checking. Use the instruction to set a smaller than the physical size I/O buffer, to enable monitor mode and block I/O checking, and to select either full-duplex communications protocol or three-wire communications capability.

SYNTAX: ESC . @ [(<DEC>) ; (<DEC>)] :

DEFAULT: ESC . @ : Sets logical I/O buffer size to 1024 bytes; selects monitor mode 1; disables monitor mode and block I/O checking.

NOTE: Defaulting the second parameter does not affect the current logic state of bits 0 and 1. ■

EXPLANATION: The first parameter of this instruction sets a logical buffer size to provide added assurance that data will not overflow the buffer. The actual physical buffer size is not changed, but the plotter's response to an output request for buffer size or buffer space is restricted to a value equal to or less than the physical buffer size.

At default, the physical and logical buffer size are equal. The *physical* buffer size, definable with the ESC . T instruction, is the actual number of bytes in the buffer. Exceeding the physical size causes buffer overflow and byte loss. The *logical* buffer size set by ESC . @ affects only handshaking thresholds; exceeding the logical buffer size causes a buffer overflow error but no byte loss unless the physical size is exceeded. Therefore, the logical size should always be less than or equal to the physical buffer size. If you specify a logical size greater than the physical size specified in the ESC . T instruction, the plotter automatically uses the physical size.

A description of the instruction's parameters follows.

<DEC> The first parameter specifies an effective logical buffer size. Parameter range is 0 to 32 767. A parameter equal to or greater than default size is interpreted as default size unless a larger physical buffer size has been established by the ESC.T instruction. In this case, a parameter equal to or greater than 18 432 is interpreted as 18 432.

<DEC> The second parameter is specified as a decimal value in the range of 0 to 127. It controls the state of bits 0 through 4, which establish hardwire handshake, communications protocol, two mutually exclusive monitor modes, and block I/O checking.

The power-on state of the first two bits is determined by the position of the **HARDWIRE/MODEM** switch.

If **MODEM** is selected, bits 0 and 1 are set to "0." (Bits 0 and 1 are ignored if either automatic disconnect mode is used.) If **HARDWIRE** is selected, bits 0 and 1 are set to "1."

Bits 2 and 3 always default to "0." Subsequent states for these bits can only be used by executing this instruction.

Setting bit 4 to "1" enables block I/O error checking and marks the beginning of a data block. The block is terminated with an ESC . E instruction. Refer to the ESC . E instruction for a full description of block error checking.

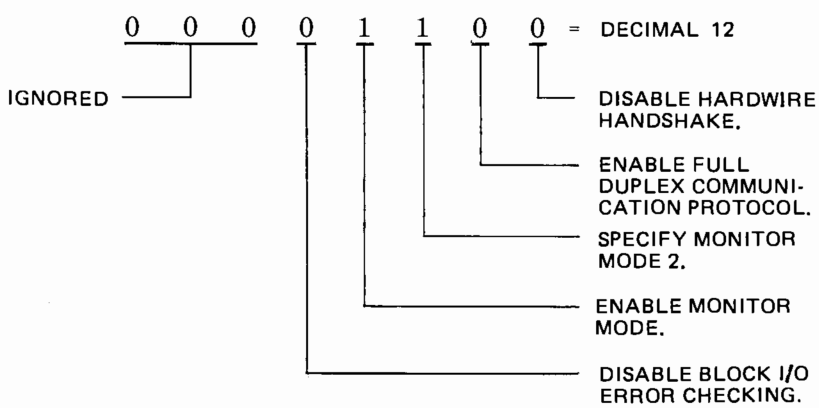
Only bits 0 through 4 are significant, as defined in the following table.

Bit No.	Logic State	Description
0	0	Disable hardwire handshake mode (set and hold pin 20 of COMPUTER/MODEM connector high).
	1	Enable hardwire handshake mode.*
1	0	Enable full duplex data communication.
	1	Enable three-wire data communication.
2	0	Specify monitor mode 1 (only HP-GL instructions are displayed as they are parsed from the buffer). Refer to Monitor Mode, in this chapter.
	1	Specify monitor mode 2 (all bytes are displayed as they are received by the plotter).
3	0	Disable monitor mode.
	1	Enable the monitor mode specified by bit 2.
4	0	Disable block I/O error checking.
	1	Enable block I/O error checking.

*When hardwire handshake is enabled, the plotter uses the Data Terminal Ready (DTR) line in the same manner that buffer threshold indicators are used in the Xon-Xoff handshake. DTR is set high at power-on and is not set low until available buffer space is less than the data block size specified by either ESC . H or ESC . I (default is 80 bytes). DTR remains low until the buffer is half empty or, if the data block size exceeds half the buffer's capacity, until available buffer space is one byte greater than the data block size.

EXAMPLE: ESC . @ 5000 ; 12 :

The 5000 specifies a logical buffer size of 5000 bytes. The decimal value 12 specifies the corresponding binary value to set the logic state of bits 0 through 3 as follows:



The Output Identification Instruction, ESC . A

USES: The ESC . A instruction outputs the plotter's model number. Use the instruction to interrogate the plotter regarding its model number. The plotter's output response is available immediately, regardless of the current operating state (Not-Ready, View, or Remote). In contrast, the HP-GL instruction, OI, is buffered and will not be parsed from the data buffer until the plotter is placed in the Remote operating state.

SYNTAX: ESC . A

EXPLANATION: No parameters are used.

RESPONSE:

<DEC>	The plotter outputs 7580B, 7585B, or 7586B, as appropriate, in a character string.
[TERM]	The terminator defaults to carriage return, CR, or is as set by ESC . M.

The Output Buffer Space Instruction, ESC . B

USES: The ESC . B instruction outputs the plotter's logical buffer space as established at default or with the ESC . @ instruction. You can use this instruction in a software checking handshake to interrogate the plotter regarding available I/O buffer space.

SYNTAX: **ESC** . B

EXPLANATION: No parameters are used.

RESPONSE:

<DEC>	The plotter's response is a decimal number in the range 0 to 1024 if the default buffer size is used. It represents the number of bytes of space currently available in the I/O buffer for storing graphic instructions sent by the computer. On all plotters, the response can be as large as 18432 if both the ESC . T and ESC . @ instructions have established a larger buffer than default size.
[TERM]	The terminator defaults to carriage return, CR, as is set by ESC . M.

The Output Extended Error Instruction, ESC . E

USES: The ESC . E instruction outputs a number defining any RS-232-C related I/O error and turns off the front-panel **ERROR** light. In addition, but only if the block I/O checking mode has been activated, ESC . E enables you to retransmit a block of data if a transmission error has occurred.

SYNTAX: **ESC** . E

EXPLANATION: No parameters are used. The ESC . E instruction always outputs the I/O error and clears any error condition.

RESPONSE:

<DEC>	The plotter's response is a decimal number, either 0 or in the range 10–18, followed by the output terminator. The meaning of the response is as defined in the following table.

Error No.	Meaning
0	No I/O error has occurred.
10	Output instruction received while another output instruction is executing. The original instruction will continue normally; the one in error will be ignored.

RS-232-C Interface

Error No.	Meaning
11	Invalid byte received after first two characters, ESC ., in a device control instruction.
12	Invalid byte received while parsing a device control instruction. The parameter containing the invalid byte and all following parameters are defaulted.
13	Parameter out of range.
14	Too many parameters received. Additional parameters beyond the proper number are ignored; parsing of the instruction ends when a colon (normal exit) or the first byte of another instruction is received (abnormal exit). **NOTE**: The receipt of something other than another parameter, a semicolon, or a colon will result in error 12 overwriting error 14. ■
15	A framing error, parity error, or overrun error has been detected.
16	The input buffer has overflowed. As a result, one or more bytes of data have been lost, and therefore, an HP-GL error will probably occur.
17	Baud rate mismatch or, full-duplex data communication is selected and conditions for data transmission are not met, i.e., cabling is configured for three-wire communications.
18	I/O error of indeterminate cause.

[TERM] The terminator defaults to carriage return, CR, unless it is set by an ESC . M.

Block I/O Error Checking

In addition to checking I/O errors, the ESC . E instruction is used to start and end a data block for block I/O checking. If the block I/O error checking has been activated (bit 4 of the second parameter of the ESC . @ instruction is set to 1), the ESC . E instruction will terminate a data block. If there are no transmission errors (response to ESC . E is 0), the data is executed normally. If response to ESC . E indicates a transmission error, the data block is discarded. You can then retransmit the entire block of data and prevent errors in the plot.

Although device control instructions bypass the buffer, the current data block is rejected if a device control instruction contains an I/O transmission error.

A new data block begins after an ESC . E instruction is received.

The following diagram illustrates block I/O error checking.

Block Checking

Computer	Plotter	Comments
ESC . @ ; 16 : ⟶		Activate block I/O checking
Data block A ⟶		Send a block of data Assume a byte gets garbled (bad parity).
ESC . E ⟶		Any I/O errors?
	⟵ 15 [TERM]	Parity, framing, or overrun error At this point, the plotter discards the block.
Data block A ⟶ ESC . E ⟶		Retransmit the block Any I/O errors?
	⟵ 0 [TERM]	No errors Plotter executes block.
Data block B ⟶		Send a block of data Assume a handshake byte gets lost, and buffer overflows.
ESC . E ⟶		Any I/O errors?
	⟵ 16 [TERM]	Buffer overflow Plotter discards block.
Data block B ⟶ ESC . E ⟶		Retransmit the block Any I/O errors?
	⟵ 0 [TERM]	No errors Plotter executes block.

The Set Handshake Mode 1 Instruction, ESC . H

USES: The ESC . H instruction may be used with the enquire/acknowledge handshake to establish the plotter's interpretation of the handshake protocol. Use the instruction to configure the plotter for enquire/acknowledge handshake when the computer requires that the plotter's output is sent in accordance with the parameters set in the ESC . M and ESC . N instructions.

SYNTAX: **ESC** . H [(<DEC>) ; (<ASC>) ; (<ASC>(; . . . <ASC>))] :

DEFAULT: **ESC** . H: The enquire/acknowledge handshake is disabled. Data block size is 80 bytes, and there is no enquiry character or acknowledgment string. If however, the computer is configured to send an ENQ anytime it is ready to send data to the plotter, the plotter will automatically respond with ACK when it receives ENQ. This "dummy handshake" is not dependent upon available buffer space and does not protect against buffer overflow.

EXPLANATION: The syntax for both ESC . H and ESC . I is the same, but the two instructions are mutually exclusive. Depending on the requirements of the computer system, either instruction can be used for the enquire/acknowledge handshake, but only ESC . I can be used for the Xon-Xoff handshake.

A description of the parameters, as interpreted for the enquire/acknowledge handshake, is given next. Refer to ESC . I for a description of the parameters as interpreted for the Xon-Xoff handshake.

For Enquire/Acknowledge Handshake

<DEC> This first parameter specifies the **data block size.** Parameter range is 0 to 32 767. Default block size set when the parameter is omitted is 80 bytes.

<ASC> This parameter sets the **enquiry character**. The parameter may be the decimal equivalent of any ASCII character in the range 0 to 126. If the parameter is omitted, it assumes the default value 0 (NULL character) disabling enquire/acknowledge handshake. Any value other than 0 enables enquire/acknowledge handshake. However, the value 5 (enquiry character, ENQ) is generally used.

<ASC>...<ASC> This is a list of 1 to 10 parameters, separated by semicolons, which specify the **acknowledgment string**. Decimal equivalents of ASCII characters 0 to 127 are valid. The value 0 is not transmitted and will terminate the string. The value 6 (acknowledgment character, ACK) is generally used. If the parameter is omitted, it assumes its default value and no characters are sent.

The ESC . M and ESC . N parameters that affect output responses of handshake mode 1, handshake mode 2, and all output instructions are shown in the following table. Choose the mode and use handshake mode 1 (ESC . H) or handshake mode 2 (ESC . I) depending on the requirements of the computer operating system.

Parameter Effectivity in Plotter Responses

ESC . M/ESC . N Parameters	Handshake Mode 1	Handshake Mode 2		Output Instructions
	Enq/Ack	Enq/Ack	Xon-Xoff	
Turnaround Delay	YES	YES	NO	YES
Output Trigger Character	YES	NO	NO	YES
Echo Terminator	YES	NO	NO	YES
Output Terminator	YES	NO	NO	YES
Output Initiator*	NO	NO	NO	YES
Intercharacter Delay	YES	YES	YES	YES
Immediate Response String	YES	YES	NO	NO

*If an output initiator is required on enquiry character responses, it should be specified as the first character of the acknowledgment string and/or the immediate response string, depending on the system.

EXAMPLE:

ESC . H 132 ; 19 ; 20 ; 7 : will set the block size to 132 bytes, the ASCII character DC3 as the enquiry character, and the two characters, DC4 and Bell, as the acknowledgment string. Since ESC . H sets handshake mode 1, the currently defined output terminator, output trigger character, and echo terminator as well as both turnaround delay, intercharacter delay, and immediate response string, are used when the response string, DC4 Bell, is sent.

The Set Handshake Mode 2 Instruction, ESC . I

USES: The ESC . I instruction may be used with the enquire/acknowledge or Xon-Xoff handshake to establish the plotter's interpretation of the handshake protocol. This instruction establishes the data block size, the enquiry character, and the acknowledgment string for the enquire/acknowledge handshake when the computer does not expect the parameters set by ESC . M to be included in the response to the enquiry character. It sets the Xoff threshold level and the Xon trigger character for Xon-Xoff handshake.

SYNTAX: **ESC** . I [(<DEC>) ; (<ASC>) ; (<ASC>(; . . . <ASC>))] :

DEFAULT: **ESC** . I : Neither Xon-Xoff nor enquire/acknowledge handshake is enabled. Block size is 80 bytes, and there is no enquiry character or acknowledgment string. If, however, the computer is configured to send an ENQ anytime it is ready to send data to the plotter, the plotter will automatically respond with ACK when it receives ENQ. This "dummy handshake" is not dependent upon available buffer space and does not protect against buffer overflow.

EXPLANATION: The syntax for both ESC . I and ESC . H is the same, but the two instructions are mutually exclusive. Depending on the requirements of the computer system, either instruction can be used for the enquire/acknowledge handshake, but only ESC . I can be used for the Xon-Xoff handshake. With handshake mode 2, some of the parameters set by the ESC . M instruction have no effect on when the plotter outputs the acknowledgment string or the Xon and Xoff triggers. Refer to the table under the ESC . H instruction to see which parameters are used. This information will help you choose whether to use ESC . I or ESC . H.

A description of the parameters, as interpreted for the Xon-Xoff handshake, is given next. Refer to ESC . H for a description of the parameters as interpreted for the enquire/acknowledge handshake.

For Xon-Xoff Handshake

<DEC> This first parameter sets the **Xoff threshold level** by specifying the number of empty bytes remaining in the buffer when the Xoff character is to be sent. If the Xoff parameter is specified to be greater than 512 (half the buffer size), the Xon threshold level will be reset (from its automatic setting of half the buffer size) so that the Xon character will be sent when one byte more than the Xoff level is available.

<ASC> To specify the Xon-Xoff handshake mode, this parameter should be omitted by entering only the semicolon or the value 0 followed by the semicolon. To enable Xon-Xoff handshake, the next parameter, which specifies an Xon trigger character(s), must also be included.

<ASC> . . . <ASC> This is a list of from 1 to 10 parameters, separated by semicolons, which specify the **Xon trigger character(s)**. Decimal equivalents of ASCII characters 0 to 127 are valid. The value 0 is not transmitted and will terminate the string.

EXAMPLES:

For Enquire/Acknowledge Handshake

ESC . I ; 5 ; 6 : will set the block size to its default value of 80 bytes, the ASCII character ENQ as the enquiry character, and the single ASCII character ACK as the acknowledgment string. Only the turnaround delay, intercharacter delay, and immediate response string, if any, are used when sending the acknowledgment string. No output initiator will precede it, even if one is defined, and no output terminator will follow it.

For Xon-Xoff Handshake

ESC . I 81 ; ; 17 : will set the Xoff threshold level to 81 (the Xoff character will be sent when 81 empty bytes remain in the plotter's buffer) and set the Xon trigger character to DC1. The second parameter is defaulted as required for this handshake. The Xoff trigger character must be set using the ESC . N instruction.

The Abort Device Control Instruction, ESC . J

USES: The ESC . J instruction aborts any device control instruction that may be partially decoded or executed. Use this instruction in an initialization sequence when you first access the plotter.

SYNTAX: **ESC** . J

EXPLANATION: This instruction aborts any device control instruction that may be partially decoded or executed. Unspecified parameters of partial instructions are defaulted. All pending or partially transmitted output requests, from either HP-GL or device control instructions, are immediately terminated, including handshake outputs. Intermediate output operations, such as turnaround delay and echo suppression, are aborted, and buffer input is enabled. Only the specified execution of an output operation is aborted. The handshake and output mode parameters remain as specified. There is no response from the plotter.

The Abort Graphic Instruction, ESC . K

USES: The ESC . K instruction aborts any partially decoded HP-GL instruction and discards instructions in the buffer. Use the instruction as part of an initialization sequence when starting a new program or to terminate plotting of HP-GL instructions in the buffer.

SYNTAX: **ESC** . K

EXPLANATION: This instruction aborts any partially decoded HP-GL instruction but permits the instruction being executed to finish. All pending graphic instructions in the buffer are discarded, leaving the buffer empty. The currently executing vector is allowed to be completed, but all other scheduled vectors are aborted. This condition is particularly evident if ESC . K is executed while the plotter is in the process of completing label, arc, circle, polygon, or line type instructions which contain multiple vector moves, or if an HP-GL VS instruction has specified a slow velocity. There is no response from the plotter.

The Output Buffer Size Instruction, ESC . L

USES: The ESC . L instruction outputs the logical size, in bytes, of the plotter's buffer. Use the instruction to obtain information on the size of the plotter's buffer. This information might be used to determine parameters of instructions which set up handshaking.

SYNTAX: **ESC** . L

EXPLANATION: No parameters are used. The instruction causes the plotter to output, in ASCII, a decimal number equal to the number of bytes of buffer space established by the ESC . @ instruction. If the parameter has not been specified, the response is 1024 bytes, the default buffer size. The response is not transmitted by the plotter until the buffer is empty.

RESPONSE:

<DEC>	1024 bytes or the number of bytes currently allocated to buffer use. The response can be as large a 18432 bytes if both the ESC . T and ESC . @ instructions have established a larger buffer than default.
[TERM]	Defaults to carriage return, CR, or is as set by ESC . M.

The Set Output Mode Instruction, ESC . M

USES: The ESC . M instruction establishes parameters for the plotter's communication format. Use the instruction to establish a turnaround delay, an output trigger character, an echo terminate character, and an output initiator character. Also use it to change the output terminator from its default value, carriage return.

SYNTAX: **ESC** . M [(<DEC>) ; (<ASC>) ; (<ASC>) ; (<ASC> (;
(<ASC>)) ; (<ASC>)] :

DEFAULT: **ESC** . M : Sets the carriage return character (decimal equivalent 13) as the output terminator. It also specifies that there is no turnaround delay and no output trigger, echo terminate, or output initiator character.

EXPLANATION: A colon must be used following the last parameter (if any). Use of the instruction without parameters is equivalent to ESC . M : (see DEFAULT).

A description of the instruction's parameters follows:

<DEC>	The first parameter is optional. If present, it is the **turnaround delay**. The parameter range is 0 to 32767 milliseconds. If parameters follow, the semicolon must be included even if this decimal parameter is omitted.

RS-232-C Interface

<ASC> The second parameter is also optional and, if omitted, assumes its default value of 0 (no trigger character). If included, it specifies a single character which becomes the **output trigger character**. The parameter may be the decimal equivalent of any ASCII character in the range 0 to 126. If parameters follow, the semicolon must always be included, even when this parameter is omitted.

<ASC> The third parameter is optional and, if omitted, assumes its default value 0 (no echo terminate character). If included, it specifies a single character which becomes the **echo terminate character**. The parameter may be the decimal equivalent of any ASCII character in the range 0 to 126. If parameters follow, the semicolon must always be included, even when this parameter is omitted.

<ASC> . . . <ASC> The fourth parameter is optional and defaults to 13, the decimal equivalent of the single ASCII character, carriage return.

If included, the parameter may be the decimal equivalent(s) of one or two ASCII characters in the range 0 to 127. This becomes the **output terminator**. The value 0 is not transmitted, and will terminate the string. If a parameter follows, the semicolon must always be included, even when this parameter is omitted. If the fifth parameter is specified, this fourth parameter must consist of two characters, or the second character must be specified as null using the semicolon.

<ASC> The fifth parameter is optional and, if omitted, assumes its default value 0 (no output initiator character). If included, it is the decimal equivalent of a single character which becomes the **output initiator** character. The parameter may be the decimal equivalent of any ASCII character in the range 0 to 127. The parameter is followed by a colon.

The flowchart on the next page depicts plotter output.

The Set Extended Output and Handshake Mode Instruction, ESC . N

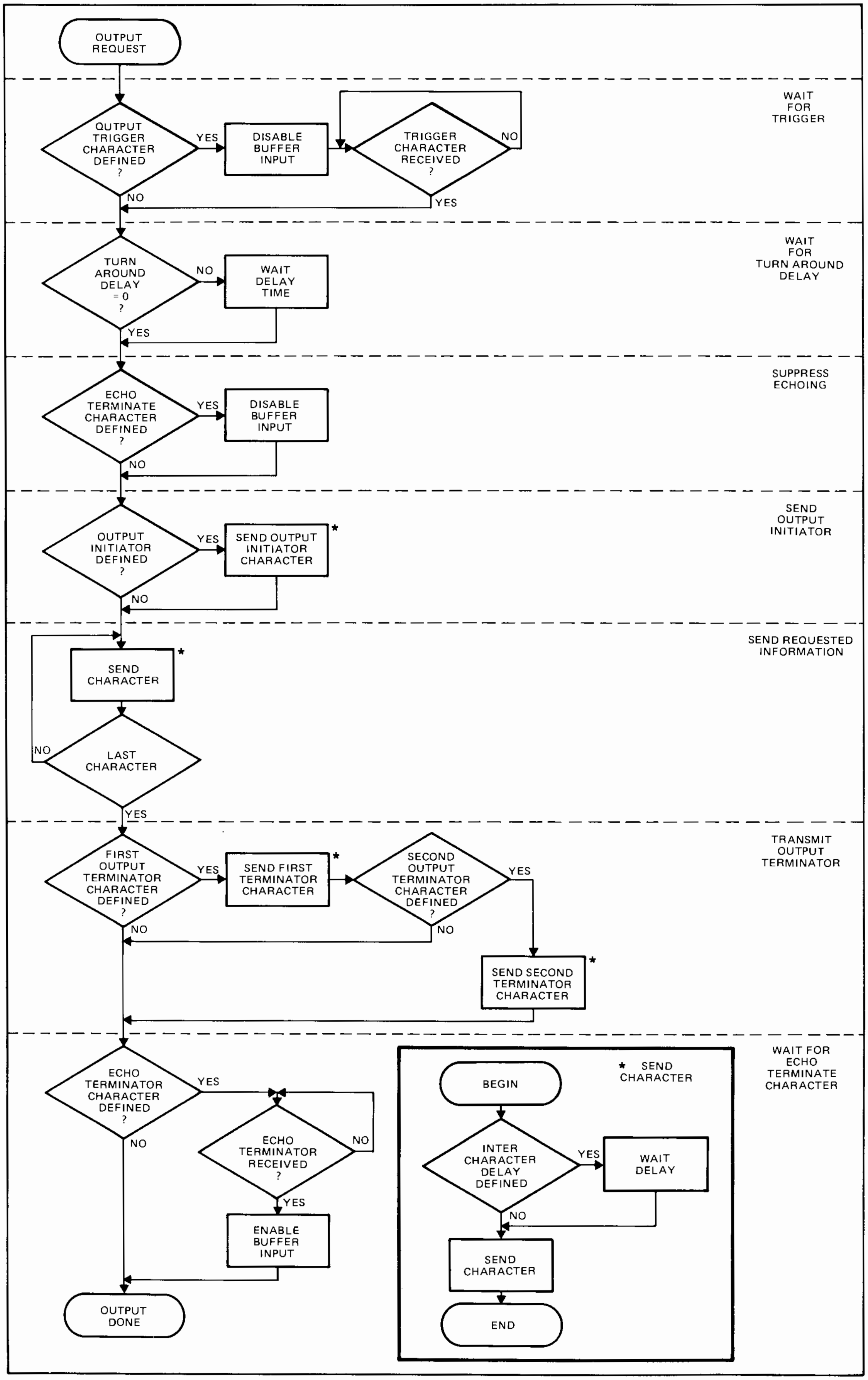
USES: The ESC . N instruction establishes parameters for the plotter's communication format. Use the instruction to specify an intercharacter delay in all handshake modes, the immediate response string for enquire/acknowledge or the Xoff trigger character(s) for the Xon-Xoff handshake.

SYNTAX: **ESC** . N [(<DEC>) ; (<ASC> (; . . . <ASC>))] :

DEFAULT: **ESC** . N : No intercharacter delay and no Xoff trigger character or immediate response string.

EXPLANATION: A colon must be used following the last parameter. Use of the instruction without parameters is equivalent to ESC . N : (see DEFAULT).

Output Request Flowchart



A description of the instruction's parameters follows:

<DEC> The first parameter is optional. If present, it is the **intercharacter delay**. The parameter range is 0 to 32 767 milliseconds. If parameters follow, the semicolon must be included, even if this decimal parameter is omitted.

<ASC> . . . <ASC> This parameter is optional. If present, it is a list of the decimal equivalents of 1 to 10 ASCII characters in the range 0 to 127. For Xon-Xoff handshake mode, it specifies the **Xoff trigger character(s)**. For enquire/acknowledge handshake mode, it specifies the **immediate response string**. Semicolons must separate each parameter in the list.

The Output Extended Status Instruction, ESC . O

USES: The ESC . O instruction outputs the plotter's extended status. Use the instruction to determine, from a remote location, if the plotter is ready to plot.

SYNTAX: **ESC** . O

EXPLANATION: No parameters are used. Unlike the HP-GL instruction, OS, the ESC . O instruction does not enter the buffer but is executed immediately, subject to any turnaround or intercharacter delays specified by ESC . M and ESC . N.

RESPONSE:

<DEC> The response is the decimal equivalent of a 16-bit immediate status word, followed by the output terminator. The extended status word bits are as defined in the following table.

[TERM] The output terminator defaults to carriage return unless it is set by ESC . M.

Extended Status Word Bits

Bit Position	Logic State	Decimal Value	Meaning
*0	0	0	Single sheet loaded.
	1	1	Roll paper loaded.
1	0	0	"Clean" paper loaded. (Set after paper is sensed.)
	1	2	Current page is not clean. Set under following conditions: a. No paper loaded or advanced since power-up or execution of ESC . R instruction. b. Pen has touched paper since last paper load or advance. c. No paper sensed before or after attempted paper advance (end of roll).
*2	0	0	No paper advance (no AF, AH, FR, or PG instruction executed) since last ESC . O instruction. The ESC . O instruction resets this bit to 0.
	1	4	Paper advance instruction (AF, AH, FR, or PG) executed since last ESC . O instruction.
3	0	0	Buffer is not empty.
	1	8	Buffer is empty and ready for data.
4 5	0 0	0 0	Remote state. Processing HP-GL instructions.
4 5	1 0	16 0	View state. Paper loaded but graphics suspended.
4 5	0 1	0 32	Not ready. Paper not loaded, graphics suspended.
6	0	0	Cover is closed.
	1	64	Cover is open.
7	0	0	Plotter in **NORMAL** mode.
	1	128	Plotter in **EMULATE** mode.
8	0	0	**EXPAND** mode is not on.
	1	256	**EXPAND** mode is on.
9–15	†	†	Not used.

*Bit recognized by the HP 7586 plotter only.

†Undefined.

RS-232-C Interface

Possible outputs (exclusive of bits 0–2) and their meanings are described in the following table.

Immediate Status Word Decimal Code Meanings

Decimal Value* Output to the Computer	Meaning						
	Buffer Is Not Empty	Buffer Is Empty	Remote State. Processing HP-GL Instructions	View State. Paper Is Loaded. Graphics Suspended	Not-Ready State. Paper Not Loaded. Graphics Suspended	Pen Carriage Cover Closed	Pen Carriage Cover Open
0	X		X			X	
8		X	X			X	
16	X			X		X	
24		X		X		X	
32	X				X	X	
40		X			X	X	
64	X		X				X
72		X	X				X
80	X			X			X
88		X		X			X
96	X		X		X		X
104		X			X		X

*Add 128 when plotter is in **EMULATE** mode.
Add 256 when plotter is in **EXPAND** mode.

NOTE: *(Applies to the HP 7586 plotter only.)* A response of 5 (bits 0 and 2 set to 1) indicates paper advance has been completed. If the response is 1 (bit 0 set to 1 and bit 2 set to 0), execute a second ESC . O so that sufficient time will elapse for the paper advance instruction to be completed. ■

The Set Handshake Mode Instruction, ESC . P

USES: The ESC . P instruction simplifies specification of the handshaking method the plotter uses when "talking" to the computer. Use this instruction to select among four standard handshakes.

SYNTAX: **ESC** . P <DEC> :

DEFAULT: **ESC** . P : No handshake is set (parameter 0).

EXPLANATION: The ESC . P instruction selects among the four standard handshakes. (Use ESC . @, H, I, M, and N to select a nonstandard handshake.) This instruction contains no additional functions; it simplifies selection of the handshaking functions indicated in the following table.

A description of the instruction's parameter follows:

<DEC> The parameter specifies a handshaking method using decimal values between 0 and 3.

Parameter	Handshake Type	Equivalent Instruction Sequence
0	No handshake	ESC . I :* ESC . M : ESC . N : ESC . @ :**
1	Xon/Xoff	ESC . I 80 ; ; 17 : ESC . M 50 ; ; 10 ; 13 : ESC . N 10 ; 19 : ESC . @ ; 2 :
2	ENQ/ACK	ESC . I 80 ; 5 ; 6 : ESC . M ; 17 ; 10 ; 13 : ESC . N : ESC . @ ; 2 :
3	Hardwire	ESC . I :* ESC . M : ESC . N : ESC . @ ; 3 :

*Selects a "dummy handshake," as at power-up.
**Disables block I/O error checking, as at power-up.

A parameter greater than 3 sets error 13 and establishes default conditions.

RS-232-C Interface

The Set Monitor Mode Instruction, ESC . Q

USES: The ESC . Q instruction enables or disables monitor mode 1 or 2 when the **MONITOR MODE/NORMAL** switch is set to **NORMAL.**

SYNTAX: ESC . Q <DEC> :

DEFAULT: ESC . Q : Disables the selected monitor mode; leaves selection unchanged.

EXPLANATION: This instruction enables monitor mode 1 (parse) or monitor mode 2 (reception). It accomplishes the same as setting bits 2 and 3 of the ESC . @ instruction. A description of the single parameter for this instruction follows.

<DEC>

Parameter	Meaning
0	Monitor mode disabled; selection unchanged.
1	Activates monitor mode 1 (parse).
2	Activates monitor mode 2 (reception)

A parameter greater than 2 sets I/O error 13 and the plotter defaults to 0.

The Reset Instruction, ESC . R

USES: The ESC . R instruction establishes a power-up I/O reset. You can use this instruction in an initialization sequence when starting a new program.

SYNTAX: ESC . R

EXPLANATION: No parameters are used.

ESC . R is a convenient way to establish the same reset conditions as sending the following instruction sequence:

ESC . J **ESC** . K **ESC** . P : **ESC** . T :

This instruction flushes all buffers, resets the parser, and defaults user-defined RAM allocations (physical I/O buffer size, polygon buffer size, and the downloadable character buffer size).

The Output Memory Size Instruction, ESC . S

USES: The ESC . S instruction outputs the total memory size of user-definable RAM, or the memory space available in any of its three buffers: the I/O buffer, the polygon buffer, and the downloadable character buffer. Use the instruction to determine how much memory is currently allocated to each buffer.

SYNTAX: **ESC** . S (<DEC>) :

DEFAULT: **ESC** . S : Outputs total RAM space available (same as ESC . S 0 :).

EXPLANATION: A colon must be used following the last parameter.

A description of the instruction's parameters follows.

<DEC> The parameter range is 0 to 3.

- 0 requests total RAM memory available
- 1 requests current I/O buffer size
- 2 requests current polygon buffer size
- 3 requests current downloadable character buffer size

RESPONSE:

<DEC> The plotter outputs the total amount of bytes in user-definable memory or the memory available in one of the three buffers, according to the parameter used.

[TERM] The terminator defaults to carriage return, CR, or is as set by ESC . M.

The Configure Memory Instruction, ESC . T

USES: The ESC . T instruction allocates memory in user-definable RAM, which consists of three separate buffers: the *physical* I/O buffer, the polygon buffer, and the downloadable character buffer. Use the instruction to enlarge the I/O buffer and the polygon buffer as needed, or to set up the downloadable character buffer.

SYNTAX: **ESC** . T [(<DEC>) ; (<DEC>) ; (<DEC>)] :

DEFAULT: **ESC** . T : Sets the I/O buffer (physical size) to 1024 bytes. Sets the polygon buffer to 3072 bytes and the downloadable character buffer to 0 bytes. Discards all data.

EXPLANATION: Below is an explanation of each parameter. Use of this instruction without parameters (ESC . T :) clears all three areas. Any data is discarded.

<DEC> The first parameter is a decimal value for the size of the *physical* I/O buffer. See the ESC . @ instruction for an explanation of the difference between *physical* and *logical* buffer size.

NOTE: If you set the physical buffer size to less than the current logical buffer size established by ESC . @, the logical size is automatically set to the value of the new physical size. ■

<DEC> The second parameter is the decimal value for the size of the polygon buffer.

<DEC> The third parameter is the decimal value for the size of the downloadable character buffer. Since the default size of this table is zero, you must allocate space to this table before using the DL instruction.

The following table summarizes the options with the ESC . T instruction.

Parameter	Minimum	Maximum	Default
Physical I/O buffer	2*	18 432***	1024
Polygon buffer	2**	18 430***	3072
Downloadable character buffer	444**	18 430***	0

*Specifying zero allocates 2 bytes.

**Specifying zero allocates no bytes. Specifying a value between 1 and the minimum allocates the minimum.

***Specifying greater than the maximum allocates the maximum.

NOTE: In all three buffers, the plotter will automatically increment odd input values to make them even. ■

The total number of available bytes is 18 432. If the sum of the three allocations exceeds 18 432, default values are used and error 13 is set.

RS-232-C Interface

Appendix A
An HP-IB Overview

The HP Interface Bus (HP-IB) provides an interconnecting channel for data transfer between devices on the HP-IB.

The following list defines the terms and concepts used to describe HP-IB (bus) system operations.

HP-IB System Terms

1. **Addressing** — The characters sent by a controlling device specifying which device sends information on the bus and which device(s) receives the information.
2. **Byte** — A unit of information consisting of 8 binary digits (bits).
3. **Device** — Any unit that is compatible with the ANSI/IEEE 488-1978 Standard.
4. **Device Dependent** — A response to information sent on the HP-IB that is characteristic of an individual device's design, and may vary from device to device.
5. **Operator** — The person that operates either the system or any device in the system.
6. **Polling** — The process typically used by a controller to locate a device that needs to interact with the controller. There are two types of polling:
   - **Serial Poll** — This method obtains one byte of operational information about an individual device in the system. The process must be repeated for each device from which information is desired.
   - **Parallel Poll** — This method obtains information about a group of devices simultaneously.

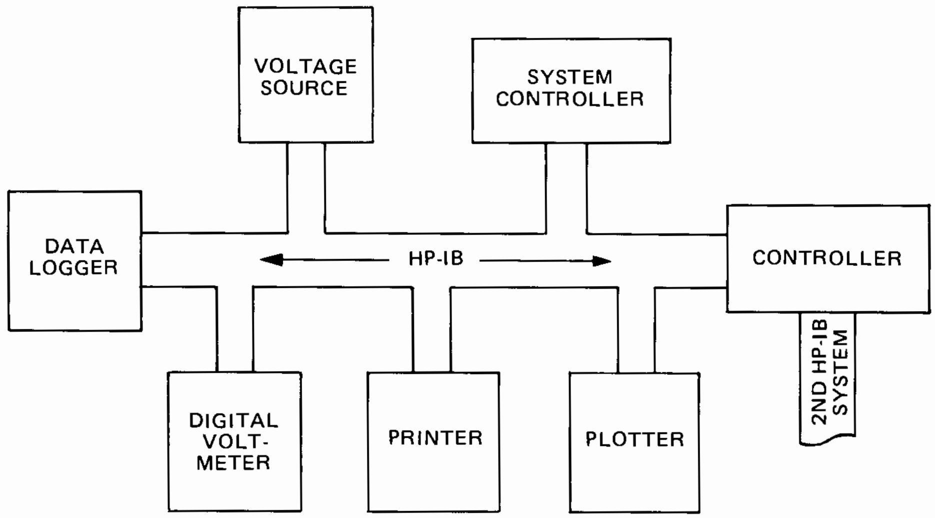
Interface Bus Concepts

Devices which communicate along the interface bus can be classified into three basic categories.

1. **Talkers** — Devices which send information on the bus when they have been addressed.
2. **Listeners** — Devices which receive information sent on the bus when they have been addressed.
3. **Controllers** — Devices that can specify the talker and listeners for an information transfer. Controllers can be categorized as one of two types:

- **Active Controller** — The current controlling device on the bus. Only one device can be the active controller at any time.
- **System Controller** — The only controller that can take priority control of the bus if it is not the current active controller. Although each bus system can have only one system controller, the system can have any number of devices capable of being the active controller.

A typical HP-IB system is shown below.



Message Concepts

Devices which communicate along the interface bus are transferring quantities of information. The transfer of information can be from one device to another device, or from one device to more than one device. These quantities of information can easily be thought of as "messages."

In turn, the messages can be classified into 12 types. The list below gives the 12 message types for the HP-IB.

1. **The Data Message.** This is the actual information which is sent from one talker to one or more listeners along the interface bus.
2. **The Trigger Message.** This message causes the listening device(s) to perform a device-dependent action when addressed.
3. **The Clear Message.** This message causes either the listening device(s) or all of the devices on the bus to return to their predefined device-dependent states.
4. **The Remote Message.** This message causes listening devices to switch from local front-panel control to remote program control when addressed to listen.
5. **The Local Message.** This message clears the Remote Message from the listening device(s) and returns the device(s) to local front-panel control.
6. **The Local Lockout Message.** This message prevents a device operator from manually inhibiting remote program control.

7. **The Clear Lockout/Local Message.** This message causes all devices on the bus to be removed from Local Lockout and revert to Local. This message also clears the Remote Message for all devices on the bus.
8. **The Require Service Message.** A device can send this message at any time to signify that the device needs some type of interaction with the controller. This message is cleared by sending the device's Status Byte Message if the device no longer requires service.
9. **The Status Byte Message.** A byte that represents the status of a single device on the bus. Bit 6 indicates whether the device sent a Require Service Message, and the remaining bits indicate operational conditions defined by the device. This byte is sent from a talking device in response to a serial poll operation performed by a controller.
10. **The Status Bit Message.** A byte that represents the operational conditions of a group of devices on the bus. Each device responds on a particular bit of the byte thus identifying a device-dependent condition. This bit is typically sent by devices in response to a parallel poll operation.

    The Status Bit Message can also be used by a controller to specify the particular bit and logic level that a device will respond with when a parallel poll operation is performed. Thus more than one device can respond on the same bit.
11. **The Pass Control Message.** This transfers the bus management responsibilities from the active controller to another controller.
12. **The Abort Message.** The system controller sends this message to unconditionally assume control of the bus from the active controller. This message terminates all bus communications (but does not implement a Clear Message).

These messages represent the full implementation of all HP-IB system capabilities. Each device in a system may be designed to use only the messages that are applicable to its purpose in the system. It is important for you to be aware of the HP-IB functions implemented on each device in your HP-IB system to ensure the operational compatibility of the system.

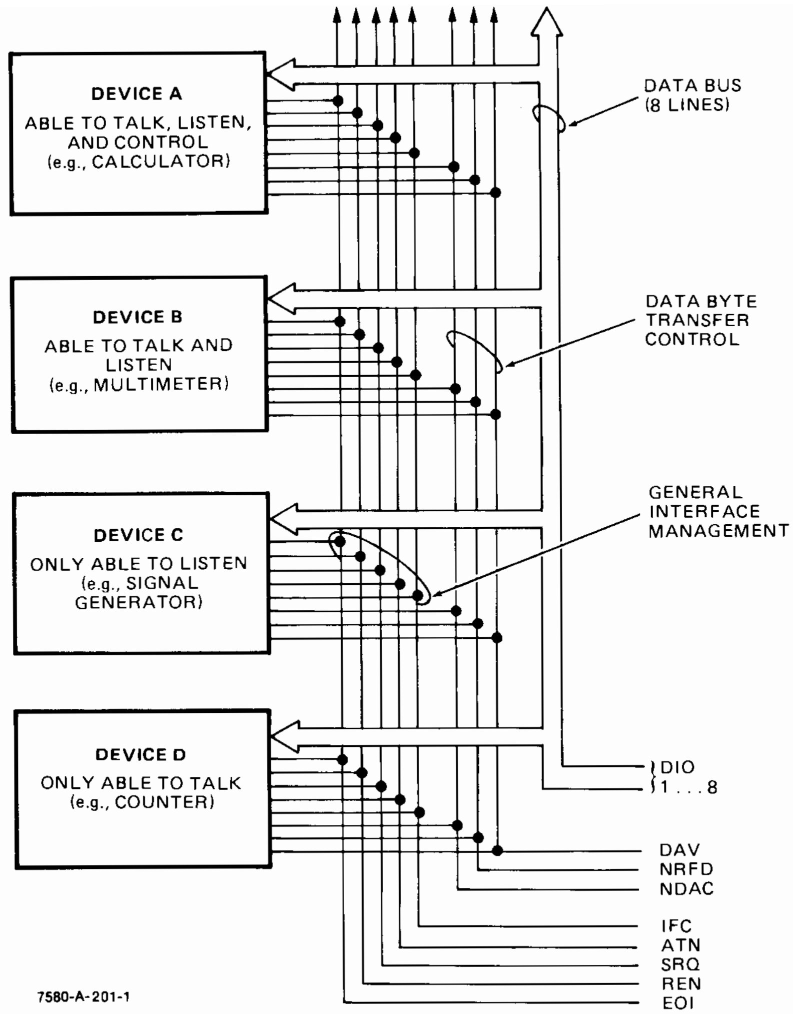
The HP Interface Bus

HP-IB Lines and Operations

The HP Interface Bus transfers data and commands between the components of an instrumentation system on 16 signal lines. The interface functions for each system component are performed within the component so only passive cabling is needed to connect the systems. The cables connect all instruments, controllers, and other components of the system in parallel to the signal lines.

The eight Data I/O lines (DIO1 through DIO8) are reserved for the transfer of data and other messages in a byte-serial, bit-parallel manner. Data and message transfer is asynchronous, coordinated by the three handshake lines: Data Valid (DAV), Not Ready For Data (NRFD), and Not Data Accepted (NDAC). The other five lines are for management of bus activity. See the following figure.

HP-IB Signal Lines



Devices connected to the bus may be talkers, listeners, or controllers. The controller dictates the roll of each of the other devices by setting the ATN (attention) line true and sending talk or listen addresses on the data lines. Addresses are set into each device at the time of system configuration either by switches built into the device or by jumpers on a PC board. While the ATN line is true, all devices must listen to the data lines. When the ATN line is false, only devices that have been addressed will actively send or receive data. All others ignore the data lines.

Several listeners can be active simultaneously but only one talker can be active at a time. Whenever a talk address is put on the data lines (while ATN is true), all other talkers will be automatically unaddressed.

Information is transmitted on the data lines under sequential control of the three handshake lines (DAV, NRFD, and NDAC). No step in the sequence can be initiated until the previous step is completed. Information transfer can proceed as fast as devices can respond, but no faster than allowed by the slowest device presently addressed as active. This permits several devices to receive the same message byte concurrently.

The ATN line is one of the five bus management lines. When ATN is true, addresses and universal commands are transmitted on only seven of the data lines using the ASCII code. When ATN is false, any code of 8 bits or less understood by both talker and listener(s) may be used.

The IFC (interface clear) line places the interface system in a known quiescent state.

The REN (remote enable) line is used with the Remote, Local, and Clear Lockout/Set Local Messages to select either local or remote control of each device.

Any active device can set the SRQ (service request) line true via the Require Service Message. This indicates to the controller that some device on the bus wants attention, such as a counter that has just completed a time-interval measurement and wants to transmit the reading to a printer.

The EOI (end or identify) line is used by a device to indicate the end of a multiple-byte transfer sequence. When a controller sets both the ATN and EOI lines true, each device capable of a parallel poll indicates its current status on the DIO line assigned to it.

In the interest of cost-effectiveness, it is not necessary for every device to be capable of responding to all the lines. Each can be designed to respond only to those lines that are pertinent to its function on the bus.

The operation of the interface is generally controlled by one device equipped to act as controller. The interface transmits a group of commands to direct the other instruments on the bus in carrying out their functions of talking and listening.

The controller has two ways of sending interface messages. Multi-line messages, which cannot exist concurrently with other multi-line messages, are sent over the eight data lines and the three handshake lines. Uni-line messages are transferred over the five individual lines of the management bus.

The commands serve several different purposes:

- Addresses or talk and listen commands select the instruments that will transmit and accept data. They are all multi-line messages.
- Universal commands cause every instrument equipped to do so to perform a specific interface operation. They include multi-line messages and three uni-line commands: interface clear (IFC), remote enable (REN), and attention (ATN).
- Addressed commands (also referred to as primary commands) are similar to universal commands, except that they affect only those devices that are addressed and are all multi-line commands. An instrument responds to an addressed command, however, only after an address has already told it to be talker or listener.
- Secondary commands are multi-line messages that are always used in series with an address, universal command, or addressed command to form a longer version of each. Thus they extend the code space when necessary.

To address an instrument, the controller uses seven of the eight data-bus lines. This allows instruments using the ASCII 7-bit code to act as controllers. As shown in the following table, five bits are available for addresses, and a total of 31 allowable addresses are available in one byte. If all secondary commands are used to extend this into a two-byte addressing capability, 961 addresses become available (31 allowable addresses in the second byte for each of the 31 allowable in the first byte).

Command and Address Codes

Code Form								Meaning
X	0	0	A_5	A_4	A_3	A_2	A_1	Universal Commands
X	0	1	A_5	A_4	A_3	A_2	A_1	Listen Addresses
				except				
X	0	1	1	1	1	1	1	Unlisten Command
X	1	0	A_5	A_4	A_3	A_2	A_1	Talk Address
				except				
X	1	0	1	1	1	1	1	Untalk Command
X	1	1	A_5	A_4	A_3	A_2	A_1	Secondary Commands
				except				
X	1	1	1	1	1	1	1	Ignored

Code used when attention (ATN) is true (low).
X = don't care.

Interface Functions

Interface functions provide the physical capability to communicate via HP-IB. These functions are defined in the ANSI/IEEE 488-1978 Standard. This standard, which is the designer's guide to the bus, defines each interface function in terms of state diagrams that express all possible interactions.

Bus capability is grouped under 10 interface functions, for example: Talker, Listener, Controller, Remote/Local. The following table lists the functions, including two special cases of Controller.

HP-IB Interface Functions

Mnemonic	Interface Function Name
SH	Source Handshake
AH	Acceptor Handshake
T	Talker (or TE = Extended Talker)*
L	Listener (or LE = Extended Listener)*
SR	Service Request
RL	Remote Local
PP	Parallel Poll
DC	Device Clear
DT	Device Trigger
C	Any Controller
C_N	A Specific Controller (for example: C_A, C_B...)
C_S	The System Controller

*Extended Talkers and Listeners use a two-byte address. Otherwise, they are the same as Talker and Listener.

HP-IB Overview

Bus Messages

Since interface functions are the physical agency through which bus messages are implemented, each device must implement one or more functions to enable it to send or receive a given bus message.

The following table lists the functions required to implement each bus message. Each device's operating manual lists the functions implemented by that device. Some devices, such as the 98034A Interface, list the functions implemented directly on the device.

Functions Used by Each Bus Message

Bus Message	Functions Required sender function → receiver function(s) (support functions)
Data	T → L* (SH, AH)
Trigger	C → DT* (L, SH, AH)
Clear	C → DC* (L, SH, AH)
Remote	C_S → RL* (SH, AH)
Local	C → RL* (L, SH, AH)
Local Lockout	C → RL* (SH, AH)
Clear Lockout/Set Local	C_S → RL*
Require Service	SR* → C
Status Byte	T → L* (SH, AH)
Status Bit	PP* → C
Pass Control	C_A → C_B (T, SH, AH)
Abort	C_S → T, L*C

*Since more than one device can receive (or send) this message simultaneously, each device must have the function indicated by an *.

HP-IB Overview

Appendix B
Instruction Syntax

HP-GL Syntax

This section lists the formal syntax for each plotter instruction in alphabetical order of the instruction's two-letter mnemonic.

Each instruction is listed with its purpose, syntax, parameter or response type, and range. If no parameter range is given, the range is $-2^{26}+1$ to $2^{26}-1$. Refer to the indicated pages for details. The semicolon is included as the terminator for all instructions except the label instructions. A line feed character or the next mnemonic can also be used as the instruction terminator.

[TERM] means the terminator sent by the plotter at the end of output to a terminal. [TERM], on the respective interfaces, is as follows:

HP-IB	CRLF
RS-232-C	CR or as defined by the ESC . M instruction

AA
The Arc Absolute Instruction
Page 5-7

AA X-coordinate, Y-coordinate, arc angle (, chord tolerance) ;

Purpose: Draws arc of specified number of degrees with specified tolerance, center at X,Y coordinate, using current pen status (up or down).

Parameters: X- and Y-coordinates — integers in plotter units unless scaling in effect; then in user units.

Arc angle — decimal, negative value specifies clockwise arc, positive value specifies counterclockwise arc.

Chord tolerance — decimal, in degrees or deviation distance, as set by CT instruction. Default 5°.

AF
The Advance Full Page Instruction
(HP 7586 only)
Page 11-7

AF ; or *PG* ; or *PG* n ;

Purpose: Programmatically advances roll paper one full-page length and establishes the plotter-unit origin at the center of the new page.

AH
The Advance Half Page Instruction
(HP 7586 only)
Page 11-9

AH ;

Purpose: Programmatically advances roll paper one half-page length.

AP
The Automatic Pen Pickup Instruction
Page 4-10

AP n ;

Purpose: Controls automatic pen lift and storage.

Parameters: Integer ⩽ 32 767, only 3 least significant bits of binary equivalent used.

Bit 0 set causes pen lift after 10 seconds for pen from drafting carousel, after 65 seconds for others.

Bit 1 causes pen storage after same time interval.

Bit 2 set delays pen retrieval until pen needed to draw.

Default 7, all three bits set.

AR
The Arc Relative Instruction
Page 5-7

AR X-increment, Y-increment, arc angle (, chord tolerance) ;

Purpose: Draws arc of specified number of degrees with specified tolerance, center relative to current pen position, using current pen status (up or down).

Parameters: X- and Y-increments — integers in plotter units unless scaling in effect; then in user units.

Arc angle — decimal, negative value specifies clockwise arc, positive value specifies counterclockwise arc.

Chord tolerance — decimal, in degrees or deviation distance, as set by CT instruction. Default 5°.

AS
The Acceleration Select Instruction
Page 4-11

AS pen acceleration (, pen number) ;

Purpose: Sets pen acceleration for one or all pens.

Parameters: Pen acceleration — integer 1 to 4. Values > 4 set to 4, negative values are errors and instruction ignored. Default 4.

Pen number — integer: if omitted, acceleration applies to all pens; if 1 through 8, specifies pen to which acceleration applies; if ⩽ 0, error; and if > 8, instruction ignored.

BL c . . . c terminator

BL
The Buffered Label String Instruction
Page 7-35

Purpose: Store characters in label buffer to be sized by OL and/or printed by PB.

Parameters: c . . . c — $\leqslant$ 150 ASCII characters including control characters and label terminator.

Terminator — label terminator defined by DT. Default ETX, decimal 3.

CA n ;

CA
The Designate Alternate Character Set Instruction
Page 7-5

Purpose: Designates the alternate character set.

Parameter: Integers 0–9, 30–39, and 99 specify a fixed-space vector set.
Integers 10–19 and 40–49 specify a variable-space arc set.
Integers 20–29 and 50–59 specify a fixed-space arc set.

CC chord angle ;

CC
The Character Chord Angle Instruction
Page 7-37

Purpose: Specifies maximum arc used when drawing in variable-arc sets 10 through 19.

Parameter: Angle in degrees, values > 45° set to 45°. Default 5°.

CI radius (, chord tolerance) ;

CI
The Circle Instruction
Page 5-4

Purpose: Draws a circle of specified radius centered at current pen position.

Parameters: Radius — integer, in plotter units if scaling not in effect; otherwise in user units. Starting point at 0° with positive parameter; 180° with negative parameter.

Chord tolerance — in degrees or deviation distance, as set by CT instruction. Default 5°.

CM switch mode (,fallback mode) ;

CM
The Character Selection Mode Instruction
Page 7-56

Purpose: Specifies mode of character set selection and usage.

Parameters: Switch mode — integer 0 to 3, specifies character selection mode.

0 HP 7-bit compatibility mode (default)
1 HP 8-bit mode
2 ISO 7-bit mode
3 ISO 8-bit mode

Fallback mode — integer, 0 or 1. Value of 0 (default) ignores undefined characters; value of 1 draws symbol for undefined characters.

Omitting parameters establishes default conditions.

CP
The Character Plot Instruction
Page 7-20

CP spaces, lines ;

Purpose: Move the pen the number of spaces and lines specified.

Parameters: Spaces — decimal, number of CP spaces, positive value moves pen in current label direction, negative value moves pen in opposite direction.

Lines — decimal, number of CP lines, positive value moves pen up, negative value moves pen down in relation to current label direction.

Omitting parameters causes carriage return, line feed.

CS
The Designate Standard Character Set Instruction
Page 7-4

CS m ;

Purpose: Designates the standard character set.

Parameter: Integers 0–9, 30–39, and 99 specify a fixed-space vector set.
Integers 10–19 and 40–49 specify a variable-space arc set.
Integers 20–29 and 50–59 specify a fixed-space arc set.

Default set 0.

CT
The Chord Tolerance Instruction
Page 5-2

CT n ;

Purpose: Determines whether chord tolerance parameter of CI, AA, and AR instructions is interpreted as degrees or as a deviation distance in plotter units (if scaling is not in effect) or user units.

Parameter: Integer, 0 or 1. Value of 0 or omitted parameter sets degree mode; value of 1 sets deviation mode.

DC
The Digitize Clear Instruction
Page 8-2

DC ;

Purpose: Clears digitize mode without entering a point from the front panel.

DF
The Default Instruction
Page 2-42

DF ;

Purpose: Returns plotter to default conditions. See table in Appendix C.

DI
The Absolute Direction Instruction
Page 7-15

SI Run, Rise ;

Purpose: Sets the direction of labels.

Parameters: Run, rise — decimal values, unitless. At least one must be nonzero, i.e., $|\text{parameter}| \geqslant 1.0^{-10}$.

Omitting parameters causes horizontal labels and is same as DI 1 , 0.

Syntax Summary

DL character number (,pen control) X-increment, Y-increment (. . . ,) (, pen control)(. . . ,) ;

DL
The Downloadable Character Instruction
Page 7-45

Purpose: Designs characters and stores them in a buffer for repeated use.

Parameters: Character number — decimal value (33-126) of any printing character.

Pen control — −128.

X,Y increments — absolute coordinates ranging from −127 to 127; drawn on a 32 × 32 unit grid.

Omitting parameters clears the downloadable character buffer.

DP ;

DP
The Digitize Point Instruction
Page 8-1

Purpose: Places plotter in digitize mode with **ENTER** LED on waiting for point to be entered from front panel.

DR run, rise ;

DR
The Relative Direction Instruction
Page 7-17

Purpose: Sets the direction of labels relative to P1 and P2.

Parameters: Decimals, **EMULATE** mode — run is % of $|P2_X - P1_X|$, rise is % of $|P2_Y - P1_Y|$.

NORMAL mode — run is % of $(P2_X - P1_X)$, rise is % of $(P2_Y - P1_Y)$.

Omitting parameters causes horizontal labels as does DR 1 , 0.

DS slot, set ;

DS
The Designate Character Set into Slot Instruction
Page 7-57

Purpose: Used with ISO modes to designate up to four character sets to be immediately available for plotting.

Parameters: Slot — integers 0 through 3 designates slots G0 through G3, respectively, to receive the character set.

Set — integers −1, 0 through 59, and 99 represent character set number assigned to slot.

DT t ;

DT
The Define Terminator Instruction
Page 7-10

Purpose: Defines the label terminator used in LB and BL commands.

Parameter: ASCII character 1 to 126. Line feed or semicolon (decimal 10 and 59) restore default terminator ETX (decimal 3).

EA
The Edge Rectangle Absolute Instruction
Page 5-27

EA X-coordinate, Y-coordinate ;

Purpose: Outlines a rectangle using absolute coordinates.

Parameters: Integers in current units representing the point diagonally opposite the current pen position.

EC
The Enable Cut Line Instruction
(HP 7586 only)
Page 11-7

EC ; or *EC* n ;

Purpose: Draws a cut line between "pages" on roll paper to indicate where to cut the paper. Used with AF, AH, and PG instructions.

Parameter: Without parameters, draws cut line after every AF, AH, and PG instruction. With a parameter (any number), the drawing function after these instructions is disabled.

EP
The Edge Polygon Instruction
Page 5-36

EP ;

Purpose: Outlines a polygon that has been previously defined in the polygon buffer with the PM, RA, RR, or WG instruction.

ER
The Edge Rectangle Relative Instruction
Page 5-27

ER X-increment, Y-increment ;

Purpose: Outlines a rectangle using relative coordinates.

Parameters: Integers in current units representing the point diagonally opposite the current pen position.

ES
The Extra Space Instruction
Page 7-13

ES spaces (, lines) ;

Purpose: Determines space to be added or deleted between characters and lines of labels.

Parameters: Decimals, positive or negative, in terms of space and line dimensions of the NCP cell. 1 space = 1-1/2 × character width as set by SI or SR. 1 line = 2× height of character as set by SI or SR. Omitting parameters causes no extra spacing.

EW
The Edge Wedge Instruction
Page 5-21

EW radius, start angle, sweep angle (, chord tolerance) ;

Purpose: Defines and edges any sector of a circle.

Parameters:

Parameter	Type	Range	Default
radius	integer	$\pm(2^{26}-1)$ current units	none
start angle	decimal	$\pm(2^{26}-1)$ degrees, modulo 360	none
sweep angle	decimal	$\pm(2^{26}-1)$ degrees, truncated at ± 360	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

Radius — in current units. The sign of the radius defines the zero-degree reference point for the start angle and sweep angle.

Start angle — a positive start angle positions the radius counterclockwise from 0 degrees; a negative start angle positions the radius clockwise from 0 degrees.

Sweep angle — a positive sweep angle draws the arc segment counterclockwise; a negative sweep angle draws the arc segment clockwise.

EW
(Continued)

FP ;

Purpose: Fills a polygon that has been previously defined in the polygon buffer with the PM, RA, RR, or WG instruction.

FP
The Fill Polygon Instruction
Page 5-37

FR ;

Purpose: Performs a long-axis frame advance and alignment and calculates a relative coordinate system for that frame.

FR
The Advance Frame Instruction
(HP 7586 only)
Page 11-9

FS pen force (, pen number) ;

Purpose: Sets pen force for one or all pens.

Parameters: Pen force — integer 1 to 8, specifies force of pen tip on paper in grams according to the formula:

$$\text{force in grams} = 10 + (\text{parameter} - 1)8$$

Default value when parameter not used determined by carousel installed: 18 for fiber-tip, 50 for roller-ball, and 10 for drafting pen carousel.

Pen Number — integer: if omitted, force applies to all pens; if 1 through 8, specifies pen to which force applies; if $\leqslant 0$, error; and if > 8, instruction ignored.

FS
The Force Select Instruction
Page 4-12

FT
The Fill Type Instruction
Page 5-10

FT (type (,spacing (, angle))) ;

Purpose: Selects type of area fill for use with FP, RA, RR, or WG instruction.

Parameters:

Parameter	Format	Range	Default
fill type	integer	1-6	1
spacing	decimal	0-($2^{26}-1$) current units	1% of the distance between $P2_X$ and $P1_X$
angle	decimal	$\pm(2^{26}-1)$ degrees, modulo 360	0 degrees

GP
The Group Pen Instruction
Page 4-14

GP (group number (, pen number (, number of pens (, length)))) ;

Purpose: Assigns pens of the same type/color to a group in order to extend the effective writing distance beyond the life of one pen.

Parameters:

Parameter	Format	Range	Default
group number	integer	1 to 8	all groups
pen number	integer	1 to 8	specified group number
number of pens	integer	1 to 8*	1
length	integer	1 to 50 000	100 metres

*pen number + number of pens must be < 9

IM
The Input Mask Instruction
Page 9-2

IM E-mask value (, S-mask value (, P-mask value)) ;

Purpose: Sets masks to specify which errors will cause the ERROR LED to come on and bit 5 of the status byte to be set, and to specify what conditions will generate an RQS or cause a positive response to a parallel poll.

Parameters: Integers 0 through 255. If parameters omitted, masks are set to 223, 0, 0, the default values.

IN
Instruction
Page 2-44

IN ;

Purpose: Sets the plotter to default conditions plus raises the pen, cancels any 90° rotation, sets P1 and P2 and the Axis Align point to their default coordinate values, clears downloadable character buffer, sets pen speed, force, and acceleration to the default values for the carousel loaded, sets bit position 3 of the status byte to 1, and clears HP-GL errors and lost mode.

Syntax Summary

IP $P1_X, P1_Y$ (, $P2_X, P2_Y$) ;

Purpose: Sets scaling points.

Parameters: Integers in plotter units. Omitting all parameters sets P1 and P2 to default coordinate values inside hard-clip limits. Omitting last two parameters causes P2 to move same relative distance as P1.

IP
The Input P1 and P2 Instruction
Page 3-2

IV (slot, (left)) ;

Purpose: Invokes a character set slot into either the right or left half of the in-use code table. Primarily used with ISO modes of character selection.

Parameters: Slot — integers, 0 through 3, invoke the slot of the same number. Default slot 0.

Left — integers 0 or 1. Value of 0 or omitted parameter invokes slot into left half; value of 1 invokes slot into right half.

IV
The Invoke Character Slot Instruction
Page 7-59

IW X_1, Y_1, X_2, Y_2 ;

Purpose: Sets window inside which plotting can occur.

Parameters: X- and Y-coordinates of opposite corners of the window.

EMULATE mode, parameters always in plotter units.

NORMAL mode, parameters in plotter units unless scaling is in effect, then in user units.

Omitting parameters sets window to current hard-clip limits, the default value.

IW
The Input Window Instruction
Page 3-15

LB c . . . c terminator

Purpose: Draws the character string using the currently selected character set.

Parameters: c . . . c — String limited to 150 ASCII characters including control characters and label terminator.

Terminator — label terminator defined by DT. Default ETX, decimal 3.

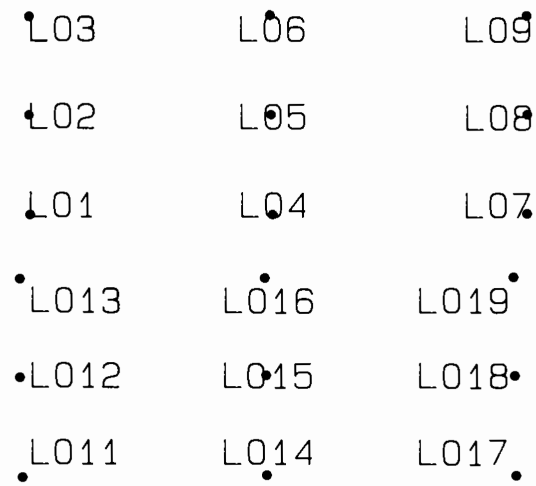
LB
The Label Instruction
Page 7-7

LO
The Label Origin Instruction
Page 7-33

LO position number ;

Purpose: Positions labels relative to current pen position.

Parameter: Integer, 1 through 9 or 11 through 19. Values 11 through 19 offset label 1/2 character width and/or height from current pen position.



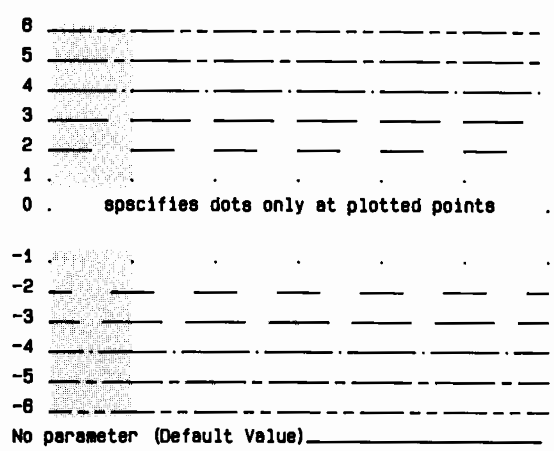
LT
The Line Type Instruction
Page 6-6

LT pattern number (, pattern length) ;

Purpose: Sets the line type used in drawing lines, arcs, and circles.

Parameters: Pattern number — integer between ±6. Negative parameters cause lines composed of complete pattern. Omitting parameter causes solid line.

Pattern length — decimal, a percentage of diagonal distance between P1 and P2 in the range 0.0000000001 to $2^{26} - 1$. Default is 4%.



NR ;

NR
The Not Ready Instruction
Page 2-45

Purpose: Programmatically simulates pressing **VIEW**.

OA ;

OA
The Output Actual Position and Pen Status Instruction
Page 9-4

Purpose: Used to output the pen's current physical position.

Response: X,Y,P [TERM] — integers, in ASCII.

X,Y — in plotter units within current hard-clip limits.

P — 0, pen up, or 1, pen down.

OC ;

OC
The Output Commanded Position and Pen Status Instruction
Page 9-4

Purpose: Used to output the pen position and status associated with last valid pen position instruction.

Response: X,Y,P [TERM] — integers, IN ASCII.

X,Y — In **EMULATE** mode, −32 768 to 32 767.

In **NORMAL** mode, -2^{29} to $2^{29} - 1$.

P — 0, pen up, or 1, pen down.

Plotter units unless scaling in effect; then in user units.

OD ;

OD
The Output Digitized Point and Pen Status Instruction
Page 8-2

Purpose: Used to output the physical pen position and status for the last digitized point.

Response: X,Y,P [TERM] — integers, in ASCII.

X,Y — In **EMULATE** mode represent plotter units.

In **NORMAL** mode represent plotter units, unless scaling in effect; then user units.

P — 0, pen up, or 1, pen down.

OE ;

OE
The Output Error Instruction
Page 9-5

Purpose: Used to output the last HP-GL error.

Response: Error number [TERM] — a positive ASCII integer, 0 through 3 or 5 through 8.

OF
The Output Factors Instruction
Page 9-6

OF ;

Purpose: Used to output the number of plotter units per millimetre in the X- and Y-axes.

Response: 40 , 40 [TERM] — integers, in ASCII.

OH
The Output Hard-Clip Limits Instruction
Page 3-14

OH ;

Purpose: Used to output the hard-clip limits at the time the instruction was received.

Response: X_{LL}, Y_{LL}, X_{UR}, Y_{UR} [TERM] — ASCII integers representing plotter units.

OI
The Output Identification Instruction
Page 9-6

OI ;

Purpose: Used to output the plotter's identification.

Response: 7580B [TERM] — ASCII string,
or
7585B [TERM] — ASCII string,
or
7586B [TERM] — ASCII string.

OL
The Output Label Length Instruction
Page 7-36

OL ;

Purpose: Used to output information on the label contained in the buffer.

Response: Length, characters, line feeds [TERM] — in ASCII.

Length — longest line in the buffered label as decimal number with four places to the right of the decimal. In terms of the space dimension of CP cell.

Characters — integer, the number of printing characters and spaces in the longest line of the buffered label.

Line feeds — integer, net number of line feeds.

OO
The Output Options Instruction
Page 9-7

OO ;

Purpose: Used to output features implemented on the plotter.

Response: C , 1 , 0 , 0 , 1 , 1 , 0 , 1 [TERM] — ASCII integers.

OP ;

OP
The Output P1 and P2 Instruction
Page 3-3

Purpose: Used to output the plotter unit coordinates of the scaling points P1 and P2.

Response: $P1_X$, $P1_Y$, $P2_X$, $P2_Y$ [TERM] — four integers in ASCII.

Range — In EMULATE mode, −32 768 to 32 767.
In NORMAL mode, $-2^{26} + 1$ to $2^{26} - 1$ for P1. P2 tracks P1 and may be outside this range.

OS ;

OS
The Output Status Instruction
Page 9-7

Purpose: Used to output the plotter's status.

Response: Status [TERM] — integer in ASCII in the range 0 to 255. Power-on status, 26.

OT ;

OT
The Output Carousel Type Instruction
Page 9-8

Purpose: Used to output information on the type of carousel loaded and the stalls occupied.

Response: Type, map [TERM] — integers in ASCII.

Type — 1 to 3: 1, fiber-tip; 2, roller-ball; 3, drafting pen.

Map — 0 to 255.

OW ;

OW
The Output Window Instruction
Page 3-11

Purpose: Used to output the X- and Y-coordinates of the window area in which plotting can occur.

Response: X_{LL}, Y_{LL}, X_{UR}, Y_{UR} [TERM] — integers, in ASCII.

NORMAL mode — X,Y in current units.

EMULATE mode — X,Y in plotter units.

PA X_1 coordinate, Y_1 coordinate (, X_2 coordinate, Y_2 coordinate, . . . , . . . , X_n coordinate, Y_n coordinate) ;

PA
The Plot Absolute Instruction
Page 4-3

Purpose: Plots to the X,Y coordinates in the order listed using the current pen up/down status.

Parameters: Pairs of integers representing plotter units if scaling not in effect, otherwise user units.

PB
The Print Buffered Label Instruction
Page 7-37

PB ;

Purpose: Prints contents of label buffer.

PD
The Pen Down Instruction
Page 4-2

PD ;
and
PD X,Y (, . . .) ;

Purpose: Programmatically lowers the pen.

PG
The Advance Full Page Instruction
(HP 7586 only)
Page 11-7

PG ; or *PG* n; or *AF* ;

Purpose: Programmatically advances roll paper one full-page length and establishes the plotter-unit origin at the center of the new page.

Parameter: The optional parameter for PG is any integer and is ignored.

PM
The Polygon Mode Instruction
Page 5-31

PM n ;

Purpose: Places the plotter in polygon definition mode in order to define shapes with other HP-GL instructions.

Parameters: Integers, 0 through 2:

0 clears polygon buffer and enters polygon mode
1 closes current subpolygon
2 closes current subpolygon and exits polygon mode

PR
The Plot Relative Instruction
Page 4-6

PR X_1 increment, Y_1 increment (, X_2 increment, Y_2 increment, ... , X_n increment, Y_n increment) ;

Purpose: Plots, in order, to the points indicated by the X,Y increments, relative to the previous pen position.

Parameters: Pairs of integers representing plotter units if scaling is not in effect, otherwise user units.

PT
The Pen Thickness Instruction
Page 5-16

PT (pen thickness) ;

Purpose: Determines the spacing between the lines drawn is a solid fill.

Parameter: Decimal between 0.1 mm and 5.0 mm. If parameter omitted, defaults to 0.3-mm size.

Syntax Summary

PU ;
and
PU X,Y (, . . .) ;

Purpose: Programmatically raises the pen.

PU
The Pen Up Instruction
Page 4-2

RA X-coordinate, Y-coordinate ;

Purpose: Defines and fills a rectangle using absolute coordinates.

Parameters: Integers in current units representing the point diagonally opposite the current pen position.

RA
The Fill Rectangle Absolute Instruction
Page 5-24

RO n

Purpose: Rotates the coordinate system 90° about the plotter unit coordinate origin.

Parameters: 0 or 90: 0 or omitting parameters turns off rotation; 90 rotates coordinate system 90°.

RO
The Rotate Coordinate System Instruction
Page 3-20

RR X-increment, Y-increment ;

Purpose: Defines and fills a rectangle using relative coordinates.

Parameters: Integers in current units representing the point diagonally opposite the current pen position.

RR
The Fill Rectangle Absolute Instruction
Page 5-24

SA ;

Purpose: Selects the alternate character set designated by the CA instruction as the character set to be used for subsequent labeling.

SA
The Select Alternate Set Instruction
Page 7-6

SC X_{min}, X_{max}, Y_{min}, Y_{max} ;

Purpose: Scales the plotting area into user units.

Parameters: Integers.

SC
The Scale Instruction
Page 3-4

SG group number ;

Purpose: Enables the plotter to continuously select among predesignated pens within a group in order to extend the effective writing distance beyond the limits of one pen.

Parameter: Integer, 0 through 8, selects a group previously defined with the GP instruction. If parameter is omitted, plotter defaults to 0 and instruction performs like SP 0 ;.

SG
The Select Pen Group Instruction
Page 4-15

Syntax Summary

SI The Absolute Character Size Instruction

Page 7-22

SI width, height ;

Purpose: Sets character width and height in centimetres for labels

Parameters: Width, height — decimals representing centimetres
in **EMULATE** mode, > 0 to $2^{26} - 1$
in **NORMAL** mode, $-2^{26} + 1$ to $2^{26} - 1$ excluding zero and numbers represented as zero.

Omitting parameters produces default size characters, 0.285 cm wide by 0.375 cm high.

SL The Character Slant Instruction

Page 7-32

SL tan θ ;

Purpose: Establishes the slant for labeled characters.

Parameters: Decimal, interpreted as the tangent of the angle from vertical.

Omitting parameters establishes no slant, the same as the default or SL 0.

SM The Symbol Mode Instruction

Page 6-5

SM character ;

Purpose: Causes specified symbol to be drawn at each plotted point

Parameter: Any printing character ASCII 33–126 excluding semicolon (ASCII 59). SM ; or SM↵ cancels symbol mode.

SP The Pen Select Instruction

Page 4-2

SP pen number ;

Purpose: Selects or stores a pen.

Parameter: Integers 0 through 8. Omitting parameters or a parameter of 0 stores the pen.

SR The Relative Character Size Instruction

Page 7-23

SR width, height ;

Purpose: Sets the character width and height in centimetres for labels

Parameters: Decimals representing a percentage of the distance between P1 and P2 as follows:

Width — in **EMULATE** mode, a percent of $|P2_X - P1_X|$
NORMAL mode, a percent of $(P2_X - P1_X)$

Height — in **EMULATE** mode, a percent of $|P2_Y - P1_Y|$
NORMAL mode, a percent of $(P2_Y - P1_Y)$

Omitting parameters produces characters 0.285 wide by 0.375 cm high which changes with subsequent changes in P1 and P2.

Syntax Summary

SS ;

SS
The Select Standard Set Instruction
Page 7-6

Purpose: Selects the standard character set designated by the CS instruction as the character set used for subsequent labeling.

TL tp (, tn) ;

TL
The Tick Length Instruction
Page 6-3

Purpose: Establishes the length of ticks drawn with the instruction XT and YT.

Parameters: Decimals, 0 to $2^{26} - 1$.

Tp — percentage of $|P2_Y - P1_Y|$ for XT or $|P2_X - P1_X|$ for YT. Denotes portion above the X-axis or to the right of the Y-axis.

Tn — same as tp except denotes portion below the X-axis and to the left of the Y-axis.

Omitting parameters causes tick length tp and tn 0.5% of $|P2_Y - P1_Y|$ or $|P2_X - P1_X|$, the same as the default value.

UC (pen control ,) X-increment, Y-increment (, . . .) (, pen control) (, . . .) ;

UC
The User-Defined Character Instruction
Page 7-39

Purpose: Draws characters or symbols defined by the user.

Parameters: Pen control — EMULATE mode $\geqslant +99$ pen down or $\leqslant -99$ pen up.

NORMAL mode $\geqslant +9999$ pen down $\leqslant -9999$ pen up.

X-increment, Y-increment in primitive grid units.

EMULATE mode range, ±98 grid units.

NORMAL mode range, ±9998 grid units.

Omitting parameters causes carriage return, line feed.

UF gap_1 (, gap_2 , . . . gap_{20}) ;

UF
The User-Defined Fill Type Instruction
Page 5-13

Purpose: Defines a fill pattern composed of gaps within parallel lines. Used to fill polygons and rectangles.

Parameters: Decimals, representing the distance to the next fill line. The spacing parameter of the FT instruction is the repeat length of the pattern defined with the UF instruction.

VS
The Velocity Select Instruction
Page 4-13

VS pen velocity (, pen number) ;

Purpose: Sets the velocity for one or all pens.

Parameters: Integers.

Pen velocity — 1 through 60 interpreted as cm/s. Defaults to 50 cm/s for pens from fiber-tip carousel, 60 cm/s for pens from roller-ball carousel, and 15 cm/s for pens from drafting pen carousel.

Pen number, 1 through 8, indicates pen to which velocity applies. If omitted, velocity applies to all pens.

WG
The Fill Wedge Instruction
Page 5-18

WG radius, start angle, sweep angle (, chord tolerance) ;

Purpose: Defines and fills any sector of a circle.

Parameters:

Parameter	Type	Range	Default
radius	integer	$\pm(2^{26}-1)$ current units	none
start angle	decimal	$\pm(2^{26}-1)$ degrees, modulo 360	none
sweep angle	decimal	$\pm(2^{26}-1)$ degrees, truncated at ± 360	none
chord tolerance	decimal	$\pm(2^{26}-1)$ current mode	5 degrees

Radius — in current units. The sign of the radius defines the zero-degree reference point for the start angle and sweep angle.

Start angle — a positive start angle positions the radius counterclockwise from 0 degrees; a negative start angle positions the radius clockwise from 0 degrees.

Sweep angle — a positive sweep angle draws the arc segment counterclockwise; a negative sweep angle draws the arc segment clockwise.

XT
The X-Tick Instruction
Page 6-1

XT ;

Purpose: Draws a vertical tick mark of the length specified by the TL instruction at the current pen position.

YT
The Y-Tick Instruction
Page 6-1

YT ;

Purpose: Draws a horizontal tick mark of the length specified by the TL instruction at the current pen position.

Device Control Instruction Summary

This section lists the formal syntax for HP-IB and RS-232-C device control instructions in alphabetical order of the escape sequence. Refer to the indicated page for details.

HP-IB

Set Plotter Configuration

Page 12-6

ESC . @ [(<DEC>) ; (<DEC>)] :

Purpose: Sets logical buffer size; enables or disables monitor mode 1.

Parameters: <DEC> — Specifies logical buffer size.

<DEC> — Monitor mode 1, decimal value 0–127. For bit 3: logic 0 disables monitor mode, logic 1 enables monitor mode. For bit 4: logic 0 disables block I/O error checking; logic 1 enables block I/O error checking.

Output Identification

Page 12-7

ESC . A

Purpose: Outputs the plotter model number.

Response: <DEC> [TERM] — 7580B, 7585B, or 7586B.

Output Buffer Space

Page 12-7

ESC . B

Purpose: Outputs the number of byte spaces currently available for data in the I/O buffer.

Response: <DEC> [TERM] — 0 to 1024, if default size is used. Otherwise, 0–18 432.

Output Extended Error

Page 12-7

ESC . E

Purpose: Outputs a decimal code to identify the type of device control instruction error that occurred.

Response: <DEC> [TERM] — 0, no error, or 11–14.

Abort Device Control

Page 12-8

ESC . J

Purpose: Aborts any partially decoded or executed device control instructions including outputs.

Abort Graphic Instruction

Page 12-9

ESC . K

Purpose: Aborts any partially decoded HP-GL instruction but permits instruction being executed to finish.

Output Buffer Size
Page 12-9

ESC . L

Purpose: Outputs the buffer size as specified by the ESC . @ or default 1024 bytes.

Response: 1024 bytes or as set by ESC . @. Not output until the buffer is empty.

Output Extended Status
Page 12-9

ESC . O

Purpose: Outputs the decimal equivalent value of a 16-bit immediate status word.

Response: <DEC> [TERM] — A positive value less than 488.

Set Monitor Mode
Page 12-11

ESC . Q <DEC> :

Purpose: Enables or disables monitor mode 1.

Parameter: <DEC> — 0 disables monitor mode
1 or 2 enables monitor mode

Reset
Page 12-12

ESC . R

Purpose: Establishes a power-up I/O reset. No parameters are used.

Output Memory Size
Page 12-12

ESC . S <DEC> :

Purpose: Outputs total memory size of user-definable RAM or space available in any of its three buffers.

Parameter: <DEC> — 0 requests total RAM available
1 requests current I/O buffer size
2 requests current polygon buffer size
3 requests current downloadable character buffer size

Response: <DEC> [TERM] — 0 to 18 432.

Configure Memory
Page 12-13

ESC . T [(<DEC>) ; (<DEC>) ; (<DEC>)] :

Purpose: Allocates memory in any of three separate buffers which make up the user-definable RAM.

Parameters: <DEC> — I/O buffer size: 1 to 18 432.

<DEC> — Polygon buffer size: 2 to 18 430.

<DEC> — Downloadable character table buffer size: 444 to 18 430.

ESC . (or **ESC** . Y

Plotter On
Page 13-27

Purpose: Places the plotter in the programmed-on mode.

ESC .) or **ESC** . Z

Plotter Off
Page 13-27

Purpose: Places the plotter in the programmed-off mode.

ESC . @ [(<DEC>) ; (<DEC>)] :

Set Plotter Configuration
Page 13-27

Purpose: Sets an effective logical buffer size, enables or disables hardwire handshake, monitor mode and block I/O checking, and selects either full duplex communication protocol or 3-wire communication capability.

Parameters: <DEC> — Maximum I/O buffer size, 0–18 432 bytes.

<DEC> — Hardwire handshake, communication protocol, block I/O error checking, and monitor mode. Decimal value in the range of 0–127.

ESC . A

Output Identification
Page 13-29

Purpose: Outputs the plotter model number.

Response: <DEC> [TERM] — 7580B, 7585B, or 7586B.

ESC . B

Output Buffer Space
Page 13-30

Purpose: Outputs the number of byte spaces currently available for data in the I/O buffer.

Response: <DEC> [TERM] — 0 to 1024, if default size is used. Otherwise, 0–18 432.

ESC . E

Output Extended Error
Page 13-30

Purpose: Outputs a decimal code to identify the type of RS-232-C related error that occurred. Also terminates a data block when block I/O error checking is enabled.

Response: <DEC> [TERM] — 0, no error, or 10–18.

Set Handshake Mode 1
Page 13-32

ESC . H [(<DEC>) ; (<ASC>) ; (<ASC> (; . . .<ASC>))] :

Purpose: Establishes parameters for handshake mode 1.

Parameters: <DEC> — Data block size.

<ASC> — Enquiry character.

<ASC> ... <ASC> — Acknowledgment string of 1 to 10 characters.

Dependent upon parameters of Set Output Mode, ESC . M.

Set Handshake Mode 2
Page 13-34

ESC . I [(<DEC>) ; (<ASC>) ; (<ASC> (; . . . <ASC>))] :

Purpose: Establishes parameters for handshake mode 2.

Parameters: <DEC> — Data block size or Xoff threshold level.

<ASC> — Enquiry character or omitted for Xon-Xoff.

<ASC> . . . <ASC> — Acknowledgment string of 1 to 10 characters or Xon trigger characters.

Independent of Set Output Mode, ESC . M.

Abort Device Control
Page 13-35

ESC . J

Purpose: Aborts any partially decoded or executed device control instructions including outputs.

Abort Graphic Instruction
Page 13-35

ESC . K

Purpose: Aborts any partially decoded HP-GL instruction but permits instruction being executed to finish.

Output Buffer Size
Page 13-36

ESC . L

Purpose: Outputs the buffer size as specified by the ESC . @ or default 1024 bytes.

Response: 1024 bytes or as set by ESC . @. Not output until the buffer is empty.

sc . M [(<DEC>) ; (<ASC>) ; (<ASC>) ; (<ASC>(; (<ASC>)) ;
(<ASC>)] :

Set Output Mode
Page 13-36

Purpose: Sets parameters for output.

Parameters: <DEC> — Turnaround delay, 0–32 767 milliseconds.

<ASC> — Output trigger character, ASCII 0–126.

<ASC> — Echo terminator character, ASCII 0–126.

<ASC>. . . <ASC> — 1 or 2 output terminators. ASCII 0–127, 0 terminates string.

<ASC> — Output initiator character, ASCII 0–127.

---

sc . N [(<DEC>) ; (<ASC> (; . . . <ASC>))] :

Set Extended Output and Handshake Mode
Page 13-37

Purpose: Establishes extended parameters for any output instruction.

Parameters: <DEC> — Delay between output characters, 0–32 767 milliseconds.

<ASC>. . . <ASC> — Immediate response string of 1 to 10 characters. ASCII 0–127, 0 terminates string; or Xoff trigger characters.

---

sc . O

Output Extended Status
Page 13-39

Purpose: Outputs the decimal equivalent value of a 16-bit immediate status word.

Response: <DEC> [TERM] — a value less than 488.

---

sc . P <DEC> :

Set Handshake Mode
Page 13-41

Purpose: Used to simplify the specification of one of the four standard handshaking methods.

Parameter: <DEC> — 0 No handshake
1 Xon/Xoff
2 ENQ/ACK
3 Hardwire

---

sc . Q <DEC> :

Set Monitor Mode
Page 13-42

Purpose: Enables or disables monitor mode.

Parameter: <DEC> — 0 disables monitor mode
1 enables monitor mode 1 (parse)
2 enables monitor mode 2 (reception)

Syntax Summary

Reset
Page 13-42

ESC . R

Purpose: Establishes a power-up I/O reset, or the same reset conditions as sending the following instruction sequence:

ESC . J **ESC** . K **ESC** . P : **ESC** . T :

No parameters are used.

Output Memory Size
Page 13-43

ESC . S <DEC> :

Purpose: Outputs total memory size of user-definable RAM, or space available in any of its three buffers.

Parameter: <DEC> — 0 requests total RAM available
1 requests current I/O buffer size
2 requests current polygon buffer size
3 requests current downloadable character buffer size

Response: <DEC> [TERM] — 0 to 18 432.

Configure Memory
Page 13-43

ESC . T [(<DEC>) ; (<DEC>) ; (<DEC>)] :

Purpose: Allocates memory in any of three separate buffers which make up the user-definable RAM.

Parameters: <DEC> — I/O buffer size: 1 to 18 432.

<DEC> — Polygon buffer size: 2 to 18 430.

<DEC> — Downloadable character table buffer size: 444 to 18 430.

Syntax Summary

Appendix C
Reference Material

Plotter Default Conditions

Function	Equivalent Instructions	Condition
		EMULATE
Pen control	AP ;	Lift or store a motionless pen after 10 seconds for drafting pens or after 65 seconds for fiber-tip or roller-ball pens. Select only when required to draw.
Label buffer	BL **ETX**	Cleared
Alternate set	CA 0 ;	Character set 0
Character chord	CC ;	Set arc font chord angle to 5°
Character selection mode	CM ;	HP seven-bit mode
Standard set	CS 0 ;	Character set 0
Chord tolerance	CT ;	Set to 5° for AA, AR and CI
Digitize clear	DC ;	Clear DP instruction and turn **ENTER** LED off
*Relative Direction	DR 1 , 0 ;	Horizontal characters
Label terminator	DT **ETX**	ETX (decimal equivalent 3)
Extra space	ES 0 , 0 ;	No extra space between characters
Fill type	FT ;	Type 1, solid bidirectional fill
Fill spacing	FT ;	1% of the X-axis distance from P1 to P2
Fill angle	FT ;	0 degrees
Mask value	IM 233 , 0 , 0 ;	Recognizes all defined errors
Input window	IW ;	Set to hard-clip limits
Label origin	LO 1 ;	Standard labeling starting at current position
Line pattern length	LT ;	4% of the diagonal distance from P1 to P2
Line type	LT ;	Solid line
Plotting mode	PA ;	Absolute
Polygon buffer	PM 0 ; PM 2 ;	Polygon buffer cleared
Pen thickness	PT ;	0.3 mm
Scaling	SC ;	User-unit scaling off
Character slant	SL 0 ;	0°
Symbol mode	SM ;	off
*Relative size	SR ;	Character width = 0.285 cm; Character height = 0.375 cm

Replaced when switch set to **NORMAL.**

Function	Equivalent Instructions	Condition
	EMULATE (Continued)	
Select set	SS ;	Select standard character set
Tick length	TL . 5 , . 5 ;	tp and tn = 0.5% of $\|P2_X - P1_X\|$ for Y-tick and 0.5% of $\|P2_Y - P1_Y\|$ for X-tick
User-defined fill type	UF ;	Solid bidirectional fill
	NORMAL	
Absolute direction	DI 1 , 0 ;	Horizontal characters
Absolute size	SI ;	Character width = 0.285 cm Character height = 0.375 cm

In addition, a DF instruction updates the carriage return point to the current pen position and resets the PA mode so that subsequent use of the mnemonics PD and PU are interpreted to be forms of the PA instruction.

An IN instruction or a front-panel **RESET** function sets the plotter to the same conditions as a DF instruction and:

- Raises the pen graphically and physically.
- Cancels any 90° rotation
- Returns P1, P2, and the Axis Align point to the X,Y coordinate values that were set when the paper limits were established. Any existing axis alignment is maintained.
- Sets default pen speed, force, and acceleration values in accordance with the currently installed carousel.
- Bit position 3 of the status byte is set to 1
- Lost mode is cleared
- Any HP-GL error condition is cleared (RS-232-C error conditions are maintained).
- Clears downloadable character buffer.

Self-Test Error Numbers

The table of self-test error numbers is given below. Unless otherwise noted, upon receipt of an error, repeat the procedure which caused the error. If the error recurs, call your HP Sales and Support Office.

CAUTION

If an error in the range E32 through E37 is displayed on the plotter, turn the plotter off immediately. Then proceed as instructed below and then try again.

Self-Test Error Number Definitions

Reference Material

Error Number	Definition	User Action
	Microprocessor Errors	
E01	Memory error	
E02	ROM checksum error	
E03	Arithmetic overflow in line clipper	
E04	Floating point underflow	Check data
E05	Floating point overflow	Check data
E06	EAROM checksum failure	
E07–E20	Not used	
	Hardware Errors	
E21	Pen initialization sensor has failed	
E22	Front optical paper sensor has failed or paper has dropped out of plotter	Clean sensors and check position of pinch wheels
E23	Rear optical paper sensor has failed or paper has dropped out of plotter	Clean sensors and check position of pinch wheels
E24	Pinch wheel motor rotation not confirmed	
E25	Excessive velocity in X-axis	
E26	Excessive velocity in Y-axis	
E27	X-axis motor overload	Remove any paper blocking movement
E28	Y-axis motor overload	Remove any paper blocking movement
E29	Not used	
E30	Primary safety switch broken	Call HP Sales and Support Office
E31	Reed switch broken or pinch wheel in impossible place	
*E32	Attempt to select pen with pen already in pen holder	**

*These hardware errors will typically occur if a pen is manually installed or removed from the pen holder by the operator while the carousel is loaded. Therefore, the pen map in memory does not match the physical condition of the plotter. Reinsert the pen carousel.

*Turn off the plotter. Check to be sure no pens or pen caps are loose in the carousel area. Check that the carousel has the proper caps for the pens installed. Do not interfere with carousel initialization.

Self-Test Error Number Definitions (Continued)

Error Number	Definition	User Action
	Microprocessor Errors	
E33	Attempt to select pen when pen is not in carousel	**
E34	Attempt to select pen is not successful	**
*E35	Attempt to store pen in an occupied carousel stall	**
*E36	Attempt to store a pen when there is no pen in the pen holder	**
E37	Attempt to store pen is not successful	
E38	Electronic failure or error of indeterminate cause	
E39-E99	Not used	

*These hardware errors will typically occur if a pen is manually installed or removed from the pen holder by the operator while the carousel is loaded. Therefore, the pen map in memory does not match the physical condition of the plotter. Reinsert the pen carousel

**Turn off the plotter. Check to be sure no pens or pen caps are loose in the carousel area. Check that the carousel has the proper caps for the pens installed. Do not interfere with carousel initialization.

HP-GL Error Messages

These error numbers are returned by executing an OE instruction.

HP-GL Error Codes

Error Number	Meaning
0	No HP-GL error for which the mask is set has occurred.
1	Instruction not recognized The plotter has received an illegal character sequence.
2	Wrong number of parameters Too many or too few parameters have been sent with an instruction.
3	Bad parameter The parameters sent to the plotter with an instruction are out-of-range for that instruction or include an illegal character.
4	Not used
5	Unknown character set A character set out of the range −1 through 59 or 99 has been designated (CS, CA, or DS instruction).
6	Position overflow Numeric overflow in plotter's character generator.
7	Buffer overflow for polygons or downloadable characters.
8	(HP 7586 only) AF, AH, FR, or PG instruction executed with sheet paper loaded.

HP-IB and RS-232-C Error Messages

These error numbers are returned by executing an ESC . E instruction.

I/O Error Codes

Error Number	Meaning
0	A zero indicates there was no I/O error.
*10	Output instruction received while another output instruction is executing. The original output instruction will continue normally while the one in error will be ignored.
11	Invalid byte received following the first two characters (ESC .) in a device control instruction.
12	Invalid byte received while parsing a device control instruction. Parameters are defaulted from the parameter where the invalid byte was received to the end of the instruction.
13	Parameter out of range.
14	Too many parameters received. Additional parameters beyond the proper number are ignored, and the parsing of the instruction ends when a colon (normal exit) or the first byte of another instruction is received (abnormal exit). **NOTE:** The receipt of something other than another parameter, a semicolon, or a colon will result in an error type 12 overwriting error type 14. ■
*15	A framing error, parity error, or overrun error has been detected.
*16	The input buffer memory has overflowed. As a result of the overflow, one or more bytes of data have been lost, and therefore, an HP-GL error will probably also occur.
*17	Baud rate mismatch or, full duplex data communication is selected and conditions for data transmission are not met, i.e., cabling is configured for three-wire data communications.
*18	I/O error of indeterminate cause.

*Applicable only to the RS-232-C environment.

No Operation Instructions

In order to maintain software compatibility with other HP plotters, the HP 7580, 7585, and 7586 recognize several HP-GL instructions and device control instructions as no operation (NOP) instructions. Device control instructions which are NOP'd on the HP-IB interface are functional on the RS-232-C interface.

In addition, page-advance instructions operative on the HP 7586 are NOP'd on the HP 7580 and 7585.

HP-GL Instructions (NOP'd on HP 7580, 7585, and 7586)

BF (Buffer Plot)
IC (Input Character)
OB (Output Character Box Dimensions)*
OK (Output Key)
RP (Replot)
WD (Write to Display)
VA (Adaptive Velocity)
VN (Normal Velocity)

HP-GL Instructions (NOP'd on HP 7580 and 7585 only)

AF (Advance Full Page)
AH (Advance Half Page)
EC (Enable Cut Line)
FR (Advance Frame)
PG (Advance Full Page)

Device Control Instructions (NOP'd on HP-IB)

ESC . (or ESC . Y
ESC .) or ESC . Z
ESC . H
ESC . I
ESC . M
ESC . N
ESC . P

Converting from User Units to Plotter Units

The plotter movements are in terms of plotter units where one plotter unit = 0.025 mm. It may be convenient for you to write programs where plotter movements are in some units other than plotter units. These "user units" can be converted into plotter units by the computer using the following equations:

$$X\ \text{scaled} = \left[\frac{P2_X - P1_X}{U2_X - U1_X}\right] A_X + P1_X - U1_X \left[\frac{P2_X - P1_X}{U2_X - U1_X}\right]$$

$$Y\ \text{scaled} = \left[\frac{P2_Y - P1_Y}{U2_Y - U1_Y}\right] A_Y + P1_Y - U1_Y \left[\frac{P2_Y - P1_Y}{U2_Y - U1_Y}\right]$$

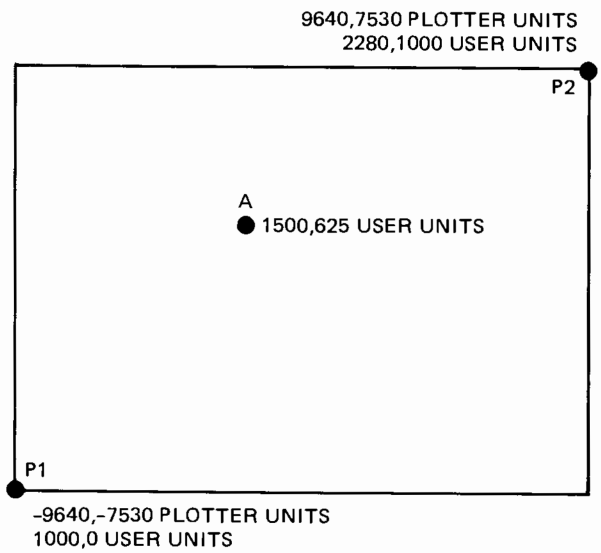
where: A_X is the X-coordinate of the desired point in user units,
A_Y is the Y-coordinate of the desired point in user units,
$P1_X$ is the X-coordinate of P1 in plotter units,
$P1_Y$ is the Y-coordinate of P1 in plotter units,
$P2_X$ is the X-coordinate of P2 in plotter units,
$P2_Y$ is the Y-coordinate of P2 in plotter units,
$U1_X$ is the X-coordinate of P1 in user units,
$U1_Y$ is the Y-coordinate of P1 in user units,
$U2_X$ is the X-coordinate of P2 in user units, and
$U2_Y$ is the Y-coordinate of P2 in user units.

*The OB instruction is not functional, but will return all zeros so that software will not hang waiting for a reply.

The following example is included to demonstrate the use of scaling equations.

PROBLEM

A C-sized sheet is scaled into 1000 to 2280 user units in X and 0 to 1000 user units in Y. The ranges of these numbers are in the same proportion as the X- and Y-distances between P1 and P2 on the paper, measured in plotter units. Therefore, there are equal increments between user units on each axis. When P1 and P2 are not at the default positions at the paper's edge, the coordinates of P1 and P2 may be obtained using the OP command. Find the plotter unit coordinates of the point 1500, 625.



SOLUTION

The plotter unit coordinates of P1 and P2 are obtained from the table in Chapter 2. The variables take on the following values:

A_X = 1500 user units
A_Y = 625 user units
$P1_X$ = −9640 plotter units
$P1_Y$ = −7530 plotter units
$P2_X$ = 9640 plotter units
$P2_Y$ = 7530 plotter units
$U1_X$ = 1000 user units
$U1_Y$ = 0 user units
$U2_X$ = 2280 user units
$U2_Y$ = 1000 user units

Solving for X and Y.

$$X = \left[\frac{P2_X - P1_X}{U2_X - U1_X}\right] A_X + P1_X - U1_X \left[\frac{P2_X - P1_X}{U2_X - U1_X}\right]$$

$$= \left[\frac{9640 - (-9640)}{2280 - 1000}\right] 1500 + (-9640) - 1000 \left[\frac{9640 - (-9640)}{2280 - 1000}\right]$$

$$= \left[\frac{19\,280}{1280}\right] 1500 - 9640 - 1000 \left[\frac{19\,280}{1280}\right]$$

$$= 22\,593.75 - 9640 - 15\,062.5$$

$= -2108.75$, which rounds to -2109 plotter units.

$$Y = \left[\frac{P2_Y - P1_Y}{U2_Y - U1_Y}\right] A_Y + P1_Y - U1_Y \left[\frac{P2_Y - P1_Y}{U2_Y - U1_Y}\right]$$

$$= \left[\frac{7530 - (-7530)}{1000 - 0}\right] 625 + (-7530) - 0 \left[\frac{7530 - (-7530)}{1000 - 0}\right]$$

$$= \left[\frac{15\,060}{1000}\right] 625 - 7530 - 0$$

$= 9412.5 - 7530 = 1882.5$, which rounds to 1883 plotter units

Therefore, the plotter-unit coordinates of the desired points are $-2109, 1883$.

Quick Reference ASCII/Decimal Table

Certain ASCII decimal equivalents are used frequently when developing control code instructions for the RS-232-C Interface Mode. The following quick reference table provides the decimal equivalents for these ASCII characters.

ASCII Characters	Decimal Equivalents
ESC . (	27,46,40
ESC .)	27,46,41
ESC . @	27,46,64
ESC . A	27,46,65
ESC . B	27,46,66
ESC . E	27,46,69
ESC . H	27,46,72
ESC . I	27,46,73
ESC . J	27,46,74
ESC . K	27,46,75
ESC . L	27,46,76
ESC . M	27,46,77
ESC . N	27,46,78
ESC . O	27,46,79
ESC . P	27,46,80
ESC . Q	27,46,81
ESC . R	27,46,82
ESC . S	27,46,83
ESC . T	27,46,84
ESC . Y	27,46,88
ESC . Z	27,46,89
ETX	3
LF	10
CR	13
:	58
;	59

ASCII Character Codes

Binary is often used as a code to represent not only numbers, but also alphanumeric characters such as "A" or "," or "?" or "x" or "2." One of the most common binary codes used is ASCII.[1] ASCII is an eight-bit code, containing seven data bits and one parity bit. The plotter uses ASCII for most I/O operations. The parity bit is ignored by the plotter. For example:

Character	ASCII Binary Code	ASCII Decimal Code
A	01000001	65
B	01000010	66
?	00111111	63

[1]American Standard Code for Information Exchange.

Following is a complete list of ASCII characters and their octal and decimal representations.

ASCII Character Code Table

ASCII Char.	Binary	Octal	Dec	ASCII Char.	Binary	Octal	Dec	ASCII Char.	Binary	Octal	Dec	ASCII Char.	Binary	Octal	Dec
NULL	00000000	000	0	space	00100000	040	32	@	01000000	100	64	`	01100000	140	96
SOH	00000001	001	1	!	00100001	041	33	A	01000001	101	65	a	01100001	141	97
STX	00000010	002	2	"	00100010	042	34	B	01000010	102	66	b	01100010	142	98
ETX	00000011	003	3	#	00100011	043	35	C	01000011	103	67	c	01100011	143	99
EOT	00000100	004	4	$	00100100	044	36	D	01000100	104	68	d	01100100	144	100
ENQ	00000101	005	5	%	00100101	045	37	E	01000101	105	69	e	01100101	145	101
ACK	00000110	006	6	&	00100110	046	38	F	01000110	106	70	f	01100110	146	102
BELL	00000111	007	7	'	00100111	047	39	G	01000111	107	71	g	01100111	147	103
BS	00001000	010	8	(	00101000	050	40	H	01001000	110	72	h	01101000	150	104
HT	00001001	011	9	)	00101001	051	41	I	01001001	111	73	i	01101001	151	105
LF	00001010	012	10	*	00101010	052	42	J	01001010	112	74	j	01101010	152	106
VTAB	00001011	013	11	+	00101011	053	43	K	01001011	113	75	k	01101011	153	107
FF	00001100	014	12	,	00101100	054	44	L	01001100	114	76	l	01101100	154	108
CR	00001101	015	13	–	00101101	055	45	M	01001101	115	77	m	01101101	155	109
SO	00001110	016	14	.	00101110	056	46	N	01001110	116	78	n	01101110	156	110
SI	00001111	017	15	/	00101111	057	47	O	01001111	117	79	o	01101111	157	111
DLE	00010000	020	16	Ø	00110000	060	48	P	01010000	120	80	p	01110000	160	112
DC$_1$	00010001	021	17	1	00110001	061	49	Q	01010001	121	81	q	01110001	161	113
DC$_2$	00010010	022	18	2	00110010	062	50	R	01010010	122	82	r	01110010	162	114
DC$_3$	00010011	023	19	3	00110011	063	51	S	01010011	123	83	s	01110011	163	115
DC$_4$	00010100	024	20	4	00110100	064	52	T	01010100	124	84	t	01110100	164	116
NAK	00010101	025	21	5	00110101	065	53	U	01010101	125	85	u	01110101	165	117
SYNC	00010110	026	22	6	00110110	066	54	V	01010110	126	86	v	01110110	166	118
ETB	00010111	027	23	7	00110111	067	55	W	01010111	127	87	w	01110111	167	119
CAN	00011000	030	24	8	00111000	070	56	X	01011000	130	88	x	01111000	170	120
EM	00011001	031	25	9	00111001	071	57	Y	01011001	131	89	y	01111001	171	121
SUB	00011010	032	26	:	00111010	072	58	Z	01011010	132	90	z	01111010	172	122
ESC	00011011	033	27	;	00111011	073	59	[	01011011	133	91	{	01111011	173	123
FS	00011100	034	28	<	00111100	074	60	\	01011100	134	92	|	01111100	174	124
GS	00011101	035	29	=	00111101	075	61	]	01011101	135	93	}	01111101	175	125
RS	00011110	036	30	>	00111110	076	62	^	01011110	136	94	~	01111110	176	126
US	00011111	037	31	?	00111111	077	63	_	01011111	137	95	DEL	01111111	177	127

Plotter ASCII Code Definitions in Character Sets

The plotter's reaction to nonprinting ASCII control characters (decimal equivalents 0 through 32 and 127, 142, and 143) and ASCII printing characters (decimal equivalents 33 through 126) are shown on the following pages for each character set. Each of the shaded symbols is automatically backspaced one character before it is drawn. Therefore, when an accented letter is required, the letter should be entered first, followed by the backspaced character.

The special centered symbols in character sets 5, 15, and 25 (ASCII decimal values 65 through 79) are designed for use in symbol mode with the SM instruction. When used in a label command, spacing will be irregular and may produce undesirable results.

A complete list of ASCII characters, their decimal representation, and the characters drawn by the plotter in each character set follow.

Plotter Character Sets

Fixed-Space Vector Font	Variable-Space Arc Font	Fixed-Space Arc Font	Character Set Name	ISO Registration Number
0	10	20	ANSI ASCII	006
1	11	21	9825 Character 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
30	40	50	ISO Swedish	010
31	41	51	ISO Swedish For Names	011
32	42	52	ISO Norway, Version 1	060
33	43	53	ISO German	021
34	44	54	ISO French	025
35	45	55	ISO United Kingdom	004
36	46	56	ISO Italian	015
37	47	57	ISO Spanish	017
38	48	58	ISO Portuguese	016
39	49	59	ISO Norway, Version 2	061
99	—	—	Drafting Set	—

Plotter ASCII Code Definitions in Character Sets

Decimal Value	ASCII Character	All Sets
0	NULL	No Operation (NOP)
1	SOH	NOP
2	STX	NOP
3	ETX	End Label Instruction
4	ETO	NOP
5	ENQ	NOP
6	ACK	NOP
7	BEL	NOP
8	BS	Backspace
9	HT	½ Backspace
10	LF	Line Feed
11	VT	Inverse Line Feed
12	FF	NOP
13	CR	Carriage Return
14	SO	Locking Shift — G1
15	SI	Locking Shift — G0
16	DLE	NOP
17	DC1	NOP
18	DC2	NOP
19	DC3	NOP
20	DC4	NOP
21	NAK	NOP
22	SYN	NOP
23	ETB	NOP
24	CAN	NOP
25	EM	NOP
26	SUB	NOP
27	ESC	NOP
28	FS	NOP
29	GS	NOP
30	RS	NOP
31	US	NOP
32	SP	Space
127	DEL	NOP
142	SS2	Single Shift — G2
143	SS3	Single Shift — G3

DECIMAL VALUE	SET 0	1	2	3	4	5	6	7	8	9	30	31	32	33	34	35	36	37	38	39	99
33	!	!	!	!	!	!	!	À	。	!	!	!	!	!	!	!	!	!	!	!	!
34	"	"	"	"	"	"	"	Â	「	"	"	"	"	"	"	"	"	"	"	"	″
35	#	#	£	£	¿	#	#	È	」	#	#	#	#	#	£	£	£	£	#	#	¢
36	$	$	$	$	$	$	$	Ê	、	¤	¤	¤	$	$	$	$	$	$	$	$	$
37	%	%	%	%	%	%	%	Ë	・	%	%	%	%	%	%	%	%	%	%	%	%
38	&	&	&	&	&	&	&	Î	ヲ	&	&	&	&	&	&	&	&	&	&	&	&
39	'	'	´	'	´	'	'	Ï	ァ	'	'	'	'	'	'	'	'	'	'	'	'
40	(	(	(	(	(	(	(	´	ィ	(	(	(	(	(	(	(	(	(	(	(	(
41	)	)	)	)	)	)	)	`	ゥ	)	)	)	)	)	)	)	)	)	)	)	)
42	*	*	*	*	*	*	*	ˆ	ェ	*	*	*	*	*	*	*	*	*	*	*	*
43	+	+	+	+	+	+	+	¨	ォ	+	+	+	+	+	+	+	+	+	+	+	+
44	,	,	,	,	,	,	,	˜	ャ	,	,	,	,	,	,	,	,	,	,	,	,
45	-	-	-	-	-	-	-	Ù	ュ	-	-	-	-	-	-	-	-	-	-	-	-
46	.	.	.	.	.	.	.	Û	ョ	.	.	.	.	.	.	.	.	.	.	.	.
47	/	/	/	/	/	/	/	£	ッ	/	/	/	/	/	/	/	/	/	/	/	/
48	0	0	0	0	0	0	0	¯	ー	0	0	0	0	0	0	0	0	0	0	0	0
49	1	1	1	1	1	1	1		ア	1	1	1	1	1	1	1	1	1	1	1	1
50	2	2	2	2	2	2	2		イ	2	2	2	2	2	2	2	2	2	2	2	2
51	3	3	3	3	3	3	3	°	ウ	3	3	3	3	3	3	3	3	3	3	3	3
52	4	4	4	4	4	4	4	Ç	エ	4	4	4	4	4	4	4	4	4	4	4	4
53	5	5	5	5	5	5	5	ç	オ	5	5	5	5	5	5	5	5	5	5	5	5
54	6	6	6	6	6	6	6	Ñ	カ	6	6	6	6	6	6	6	6	6	6	6	6
55	7	7	7	7	7	7	7	ñ	キ	7	7	7	7	7	7	7	7	7	7	7	7
56	8	8	8	8	8	8	8	¡	ク	8	8	8	8	8	8	8	8	8	8	8	8
57	9	9	9	9	9	9	9	¿	ケ	9	9	9	9	9	9	9	9	9	9	9	9
58	:	:	:	:	:	:	:	¤	コ	:	:	:	:	:	:	:	:	:	:	:	:
59	;	;	;	;	;	;	;	£	サ	;	;	;	;	;	;	;	;	;	;	;	;
60	<	<	<	<	<	<	<	¥	シ	<	<	<	<	<	<	<	<	<	<	<	<
61	=	=	=	=	=	=	=	§	ス	=	=	=	=	=	=	=	=	=	=	=	=
62	>	>	>	>	>	>	>	ƒ	セ	>	>	>	>	>	>	>	>	>	>	>	>
63	?	?	?	?	?	?	?	¢	ソ	?	?	?	?	?	?	?	?	?	?	?	?
64	@	@	@	@	@	@	@	â	タ	@	@	É	@	§	à	@	§	§	§	§	@

ASCII Code Definitions in Character Sets — Fixed-Space Vector Font (Continued)

DECIMAL VALUE	SET 0	1	2	3	4	5	6	7	8	9	30	31	32	33	34	35	36	37	38	39	99
65	A	A	A	A	A	⊡	A	ê	チ	A	A	A	A	A	A	A	A	A	A	A	A
66	B	B	B	B	B	⦶	B	ô	ツ	B	B	B	B	B	B	B	B	B	B	B	B
67	C	C	C	C	C	△	C	û	テ	C	C	C	C	C	C	C	C	C	C	C	C
68	D	D	D	D	D	+	D	á	ト	D	D	D	D	D	D	D	D	D	D	D	D
69	E	E	E	E	E	×	E	é	ナ	E	E	E	E	E	E	E	E	E	E	E	E
70	F	F	F	F	F	◇	F	ó	ニ	F	F	F	F	F	F	F	F	F	F	F	F
71	G	G	G	G	G	⇡	G	ú	ヌ	G	G	G	G	G	G	G	G	G	G	G	G
72	H	H	H	H	H	⌧	H	à	ネ	H	H	H	H	H	H	H	H	H	H	H	H
73	I	I	I	I	I	Ƶ	I	è	ノ	I	I	I	I	I	I	I	I	I	I	I	I
74	J	J	J	J	J	Y	J	ò	ハ	J	J	J	J	J	J	J	J	J	J	J	J
75	K	K	K	K	K	⌘	K	ù	ヒ	K	K	K	K	K	K	K	K	K	K	K	K
76	L	L	L	L	L	✳	L	ä	フ	L	L	L	L	L	L	L	L	L	L	L	L
77	M	M	M	M	M	⧗	M	ë	ヘ	M	M	M	M	M	M	M	M	M	M	M	M
78	N	N	N	N	N	ǀ	N	ö	ホ	N	N	N	N	N	N	N	N	N	N	N	N
79	O	O	O	O	O	✡	O	ü	マ	O	O	O	O	O	O	O	O	O	O	O	O
80	P	P	P	P	P	_	P	Å	ミ	P	P	P	P	P	P	P	P	P	P	P	P
81	Q	Q	Q	Q	Q	ˌ	Q	î	ム	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
82	R	R	R	R	R	R	R	Ø	メ	R	R	R	R	R	R	R	R	R	R	R	R
83	S	S	S	S	S	S	S	Æ	モ	S	S	S	S	S	S	S	S	S	S	S	S
84	T	T	T	T	T	T	T	å	ヤ	T	T	T	T	T	T	T	T	T	T	T	T
85	U	U	U	U	U	U	U	í	ユ	U	U	U	U	U	U	U	U	U	U	U	U
86	V	V	V	V	V	V	V	ø	ヨ	V	V	V	V	V	V	V	V	V	V	V	V
87	W	W	W	W	W	W	W	æ	ラ	W	W	W	W	W	W	W	W	W	W	W	W
88	X	X	X	X	X	X	X	Ä	リ	X	X	X	X	X	X	X	X	X	X	X	X
89	Y	Y	Y	Y	Y	Y	Y	ì	ル	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
90	Z	Z	Z	Z	Z	Z	Z	Ö	レ	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
91	[	[	[	Ø	[	[	[	Ü	ロ	[	Ä	Ä	Æ	Ä	°	[	°	¡	Ã	Ã	[
92	\	√	ç	Æ	¡	\	¥	É	ワ	\	Ö	Ö	Ø	Ö	ç	\	ç	Ñ	Ç	Ç	ø
93	]	]	]	ø	]	]	]	ï	ン	]	Å	Å	Å	Ü	§	]	é	¿	Õ	Õ	]
94	^	↑	^	æ	^	^	^	ß	"	^	^	Ü	^	^	^	^	^	^	^	^	⊥
95	_	_	_	_	_	_	_	Ô	°	_	_	_	_	_	_	_	_	_	_	_	_
96	`	`	`	`	`	`	`	Á		`	`	é	`	`	`	`	ù	`	`	`	`

DECIMAL VALUE	SET 0	1	2	3	4	5	6	7	8	9	30	31	32	33	34	35	36	37	38	39	99
97	a	a	a	a	a	∩	a	Ã		a	a	a	a	a	a	a	a	a	a	a	a
98	b	b	b	b	b	⊃	b	ã		b	b	b	b	b	b	b	b	b	b	b	b
99	c	c	c	c	c	⊂	c	Đ		c	c	c	c	c	c	c	c	c	c	c	c
100	d	d	d	d	d	∪	d	đ		d	d	d	d	d	d	d	d	d	d	d	d
101	e	e	e	e	e	¯	e	Í		e	e	e	e	e	e	e	e	e	e	e	e
102	f	f	f	f	f	≡	f	Ì		f	f	f	f	f	f	f	f	f	f	f	f
103	g	g	g	g	g	≅	g	Ó		g	g	g	g	g	g	g	g	g	g	g	g
104	h	h	h	h	h	≈	h	Ò		h	h	h	h	h	h	h	h	h	h	h	h
105	i	i	i	i	i	~	i	Õ		i	i	i	i	i	i	i	i	i	i	i	i
106	j	j	j	j	j	≤	j	õ		j	j	j	j	j	j	j	j	j	j	j	j
107	k	k	k	k	k	≥	k	Š		k	k	k	k	k	k	k	k	k	k	k	k
108	l	l	l	l	l	≠	l	š		l	l	l	l	l	l	l	l	l	l	l	l
109	m	m	m	m	m	Δ	m	Ú		m	m	m	m	m	m	m	m	m	m	m	m
110	n	n	n	n	n	Π	n	Ÿ		n	n	n	n	n	n	n	n	n	n	n	n
111	o	o	o	o	o	Σ	o	ÿ		o	o	o	o	o	o	o	o	o	o	o	o
112	p	p	p	p	p	±	p	Þ		p	p	p	p	p	p	p	p	p	p	p	p
113	q	q	q	q	q	∓	q	þ		q	q	q	q	q	q	q	q	q	q	q	q
114	r	r	r	r	r	→	r			r	r	r	r	r	r	r	r	r	r	r	r
115	s	s	s	s	s	↑	s			s	s	s	s	s	s	s	s	s	s	s	s
116	t	t	t	t	t	←	t			t	t	t	t	t	t	t	t	t	t	t	t
117	u	u	u	u	u	↓	u			u	u	u	u	u	u	u	u	u	u	u	u
118	v	v	v	v	v	∫	v	–		v	v	v	v	v	v	v	v	v	v	v	v
119	w	w	w	w	w	÷	w	¼		w	w	w	w	w	w	w	w	w	w	w	w
120	x	x	x	x	x	*	x	½		x	x	x	x	x	x	x	x	x	x	x	x
121	y	y	y	y	y	∇	y	ª		y	y	y	y	y	y	y	y	y	y	y	y
122	z	z	z	z	z	˚	z	º		z	z	z	z	z	z	z	z	z	z	z	z
123	{	π	¨	¨	~	{	{	«		{	ä	ä	æ	ä	é	{	à	˚	ã	ã	µ
124	|	⊢	˙	˙	~	|	|	□		|	ö	ö	ø	ö	ù	|	ò	ñ	ç	ç	°
125	}	→	¨	¨	~	}	}	»		}	å	å	å	ü	è	}	è	ç	õ	õ	∞
126	~	~	'	˙	~	~	~	±		¯	¯	ü	¯	ß	¨	¯	ì	~	˚	˚	~

ASCII Code Definitions in Character Sets — Variable-Space Arc Font

DECIMAL VALUE	SET 10	11	12	13	14	15	16	17	18	19	40	41	42	43	44	45	46	47	48	49
33	!	!	!	!	!	!	!	À	｡	!	!	!	!	!	!	!	!	!	!	!
34	"	"	"	"	"	"	"	Â	｢	"	"	"	"	"	"	"	"	"	"	"
35	#	#	£	£	¿	#	#	È	｣	#	#	#	#	#	£	£	£	£	#	§
36	$	$	$	$	$	$	$	Ê	､	¤	¤	¤	$	$	$	$	$	$	$	$
37	%	%	%	%	%	%	%	Ë	･	%	%	%	%	%	%	%	%	%	%	%
38	&	&	&	&	&	&	&	Î	ｦ	&	&	&	&	&	&	&	&	&	&	&
39	'	'	'	'	'	'	'	Ï	ｧ	'	'	'	'	'	'	'	'	'	'	'
40	(	(	(	(	(	(	(	´	ｨ	(	(	(	(	(	(	(	(	(	(	(
41	)	)	)	)	)	)	)	`	ｩ	)	)	)	)	)	)	)	)	)	)	)
42	*	*	*	*	*	*	*	^	ｪ	*	*	*	*	*	*	*	*	*	*	*
43	+	+	+	+	+	+	+	¨	ｫ	+	+	+	+	+	+	+	+	+	+	+
44	,	,	,	,	,	,	,	~	ｬ	,	,	,	,	,	,	,	,	,	,	,
45	-	-	-	-	-	-	-	Ù	ｭ	-	-	-	-	-	-	-	-	-	-	-
46	.	.	.	.	.	.	.	Û	ｮ	.	.	.	.	.	.	.	.	.	.	.
47	/	/	/	/	/	/	/	£	ｯ	/	/	/	/	/	/	/	/	/	/	/
48	0	0	0	0	0	0	0	¯	ｰ	0	0	0	0	0	0	0	0	0	0	0
49	1	1	1	1	1	1	1		ｱ	1	1	1	1	1	1	1	1	1	1	1
50	2	2	2	2	2	2	2		ｲ	2	2	2	2	2	2	2	2	2	2	2
51	3	3	3	3	3	3	3	·	ｳ	3	3	3	3	3	3	3	3	3	3	3
52	4	4	4	4	4	4	4	Ç	ｴ	4	4	4	4	4	4	4	4	4	4	4
53	5	5	5	5	5	5	5	ç	ｵ	5	5	5	5	5	5	5	5	5	5	5
54	6	6	6	6	6	6	6	Ñ	ｶ	6	6	6	6	6	6	6	6	6	6	6
55	7	7	7	7	7	7	7	ñ	ｷ	7	7	7	7	7	7	7	7	7	7	7
56	8	8	8	8	8	8	8	¡	ｸ	8	8	8	8	8	8	8	8	8	8	8
57	9	9	9	9	9	9	9	¿	ｹ	9	9	9	9	9	9	9	9	9	9	9
58	:	:	:	:	:	:	:	¤	ｺ	:	:	:	:	:	:	:	:	:	:	:
59	;	;	;	;	;	;	;	£	ｻ	;	;	;	;	;	;	;	;	;	;	;
60	<	<	<	<	<	<	<	¥	ｼ	<	<	<	<	<	<	<	<	<	<	<
61	=	=	=	=	=	=	=	§	ｽ	=	=	=	=	=	=	=	=	=	=	=
62	>	>	>	>	>	>	>	ƒ	ｾ	>	>	>	>	>	>	>	>	>	>	>
63	?	?	?	?	?	?	?	¢	ｿ	?	?	?	?	?	?	?	?	?	?	?
64	@	@	@	@	@	@	@	â	ﾀ	@	@	É	@	§	à	@	§	§	§	@

DECIMAL VALUE	SET																			
	10	11	12	13	14	15	16	17	18	19	40	41	42	43	44	45	46	47	48	49
65	A	A	A	A	A	⊡	A	ê	チ	A	A	A	A	A	A	A	A	A	A	A
66	B	B	B	B	B	⦶	B	ô	ツ	B	B	B	B	B	B	B	B	B	B	B
67	C	C	C	C	C	△	C	û	テ	C	C	C	C	C	C	C	C	C	C	C
68	D	D	D	D	D	+	D	á	ト	D	D	D	D	D	D	D	D	D	D	D
69	E	E	E	E	E	×	E	é	ナ	E	E	E	E	E	E	E	E	E	E	E
70	F	F	F	F	F	◇	F	ó	ニ	F	F	F	F	F	F	F	F	F	F	F
71	G	G	G	G	G	⇑	G	ú	ヌ	G	G	G	G	G	G	G	G	G	G	G
72	H	H	H	H	H	⧗	H	à	ネ	H	H	H	H	H	H	H	H	H	H	H
73	I	I	I	I	I	Z	I	è	ノ	I	I	I	I	I	I	I	I	I	I	I
74	J	J	J	J	J	Y	J	ò	ハ	J	J	J	J	J	J	J	J	J	J	J
75	K	K	K	K	K	⌑	K	ù	ヒ	K	K	K	K	K	K	K	K	K	K	K
76	L	L	L	L	L	✱	L	ä	フ	L	L	L	L	L	L	L	L	L	L	L
77	M	M	M	M	M	⧖	M	ë	ヘ	M	M	M	M	M	M	M	M	M	M	M
78	N	N	N	N	N	|	N	ö	ホ	N	N	N	N	N	N	N	N	N	N	N
79	O	O	O	O	O	✡	O	ü	マ	O	O	O	O	O	O	O	O	O	O	O
80	P	P	P	P	P	_	P	Å	ミ	P	P	P	P	P	P	P	P	P	P	P
81	Q	Q	Q	Q	Q	ı	Q	î	ム	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
82	R	R	R	R	R	R	R	Ø	メ	R	R	R	R	R	R	R	R	R	R	R
83	S	S	S	S	S	S	S	Æ	モ	S	S	S	S	S	S	S	S	S	S	S
84	T	T	T	T	T	T	T	å	ヤ	T	T	T	T	T	T	T	T	T	T	T
85	U	U	U	U	U	U	U	í	ユ	U	U	U	U	U	U	U	U	U	U	U
86	V	V	V	V	V	V	V	ø	ヨ	V	V	V	V	V	V	V	V	V	V	V
87	W	W	W	W	W	W	W	æ	ラ	W	W	W	W	W	W	W	W	W	W	W
88	X	X	X	X	X	X	X	Ä	リ	X	X	X	X	X	X	X	X	X	X	X
89	Y	Y	Y	Y	Y	Y	Y	ì	ル	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
90	Z	Z	Z	Z	Z	Z	Z	Ö	レ	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
91	[	[	[	Ø	[	[	[	Ü	ロ	[	Ä	Ä	Æ	Ä	·	[	·	¡	Ã	Æ
92	\	√	ç	Æ	¡	\	¥	É	ワ	\	Ö	Ö	Ø	Ö	ç	\	ç	Ñ	Ç	Ø
93	]	]	]	ø	]	]	]	ï	ン	]	Å	Å	Å	Ü	§	]	é	¿	Õ	Å
94	^	↑	^	æ	^	^	^	ß	゛	^	^	Ü	^	^	^	^	^	^	^	^
95	_	_	_	_	_	_	_	Ô	°	_	_	_	_	_	_	_	_	_	_	_
96	`	`	`	`	`	`	`	Á		`	`	é	`	`	`	`	ù	`	`	`

ASCII Code Definitions in Character Sets — Variable-Space Arc Font *(Continued)*

DECIMAL VALUE	SET 10	11	12	13	14	15	16	17	18	19	40	41	42	43	44	45	46	47	48	49
97	a	a	a	a	a	∩	a	Ã		a	a	a	a	a	a	a	a	a	a	a
98	b	b	b	b	b	⊃	b	ã		b	b	b	b	b	b	b	b	b	b	b
99	c	c	c	c	c	⊂	c	Đ		c	c	c	c	c	c	c	c	c	c	c
100	d	d	d	d	d	∪	d	đ		d	d	d	d	d	d	d	d	d	d	d
101	e	e	e	e	e	‾	e	Í		e	e	e	e	e	e	e	e	e	e	e
102	f	f	f	f	f	≡	f	Ì		f	f	f	f	f	f	f	f	f	f	f
103	g	g	g	g	g	≅	g	Ó		g	g	g	g	g	g	g	g	g	g	g
104	h	h	h	h	h	≈	h	Ò		h	h	h	h	h	h	h	h	h	h	h
105	i	i	i	i	i	~	i	Õ		i	i	i	i	i	i	i	i	i	i	i
106	j	j	j	j	j	≤	j	õ		j	j	j	j	j	j	j	j	j	j	j
107	k	k	k	k	k	≥	k	Š		k	k	k	k	k	k	k	k	k	k	k
108	l	l	l	l	l	≠	l	š		l	l	l	l	l	l	l	l	l	l	l
109	m	m	m	m	m	Δ	m	Ú		m	m	m	m	m	m	m	m	m	m	m
110	n	n	n	n	n	Π	n	Ÿ		n	n	n	n	n	n	n	n	n	n	n
111	o	o	o	o	o	Σ	o	ÿ		o	o	o	o	o	o	o	o	o	o	o
112	p	p	p	p	p	±	p	Þ		p	p	p	p	p	p	p	p	p	p	p
113	q	q	q	q	q	∓	q	þ		q	q	q	q	q	q	q	q	q	q	q
114	r	r	r	r	r	→	r			r	r	r	r	r	r	r	r	r	r	r
115	s	s	s	s	s	↑	s			s	s	s	s	s	s	s	s	s	s	s
116	t	t	t	t	t	←	t			t	t	t	t	t	t	t	t	t	t	t
117	u	u	u	u	u	↓	u			u	u	u	u	u	u	u	u	u	u	u
118	v	v	v	v	v	∫	v	—		v	v	v	v	v	v	v	v	v	v	v
119	w	w	w	w	w	÷	w	¼		w	w	w	w	w	w	w	w	w	w	w
120	x	x	x	x	x	*	x	½		x	x	x	x	x	x	x	x	x	x	x
121	y	y	y	y	y	∇	y	ª		y	y	y	y	y	y	y	y	y	y	y
122	z	z	z	z	z	˙	z	º		z	z	z	z	z	z	z	z	z	z	z
123	{	π	¨	¨	~	{	{	«		{	ä	ä	æ	ä	é	{	à	˙	ã	æ
124	|	⊢	˙	˙	~	|	|	□		|	ö	ö	ø	ö	ù	|	ò	ñ	ç	ø
125	}	→	¨	¨	~	}	}	»		}	å	å	å	ü	è	}	è	ç	õ	å
126	~	~	'	˙	~	˙~	~	±		—	—	ü	—	ß	¨	—	ì	~	˙	~

DECIMAL VALUE	SET 20	21	22	23	24	25	26	27	28	29	50	51	52	53	54	55	56	57	58	59
33	!	!	!	!	!	!	!	À	｡	!	!	!	!	!	!	!	!	!	!	!
34	"	"	"	"	"	"	"	Â	｢	"	"	"	"	"	"	"	"	"	"	"
35	#	#	£	£	¿	#	#	È	｣	#	#	#	#	#	£	£	£	£	#	§
36	$	$	$	$	$	$	$	Ê	､	¤	¤	¤	$	$	$	$	$	$	$	$
37	%	%	%	%	%	%	%	Ë	･	%	%	%	%	%	%	%	%	%	%	%
38	&	&	&	&	&	&	&	Î	ｦ	&	&	&	&	&	&	&	&	&	&	&
39	'	'	´	'	´	'	'	Ï	ｧ	'	'	'	'	'	'	'	'	'	'	'
40	(	(	(	(	(	(	(	´	ｨ	(	(	(	(	(	(	(	(	(	(	(
41	)	)	)	)	)	)	)	`	ｩ	)	)	)	)	)	)	)	)	)	)	)
42	*	*	*	*	*	*	*	ˆ	ｪ	*	*	*	*	*	*	*	*	*	*	*
43	+	+	+	+	+	+	+	¨	ｫ	+	+	+	+	+	+	+	+	+	+	+
44	,	,	,	,	,	,	,	˜	ｬ	,	,	,	,		,	,	,	,	,	,
45	–	–	–	–	–	–	–	Ù	ｭ	–	–	–	–	–	–	–	–	–	–	–
46	.	.	.	.	.	.	.	Û	ｮ	.	.	.	.	.	.	.	.	.	.	.
47	/	/	/	/	/	/	/	£	ｯ	/	/	/	/	/	/	/	/	/	/	/
48	0	0	0	0	0	0	0	¯	ｰ	0	0	0	0	0	0	0	0	0	0	0
49	1	1	1	1	1	1	1		ｱ	1	1	1	1	1	1	1	1	1	1	1
50	2	2	2	2	2	2	2		ｲ	2	2	2	2	2	2	2	2	2	2	2
51	3	3	3	3	3	3	3	·	ｳ	3	3	3	3	3	3	3	3	3	3	3
52	4	4	4	4	4	4	4	Ç	ｴ	4	4	4	4	4	4	4	4	4	4	4
53	5	5	5	5	5	5	5	ç	ｵ	5	5	5	5	5	5	5	5	5	5	5
54	6	6	6	6	6	6	6	Ñ	ｶ	6	6	6	6	6	6	6	6	6	6	6
55	7	7	7	7	7	7	7	ñ	ｷ	7	7	7	7	7	7	7	7	7	7	7
56	8	8	8	8	8	8	8	¡	ｸ	8	8	8	8	8	8	8	8	8	8	8
57	9	9	9	9	9	9	9	¿	ｹ	9	9	9	9	9	9	9	9	9	9	9
58	:	:	:	:	:	:	:	¤	ｺ	:	:	:	:	:	:	:	:	:	:	:
59	;	;	;	;	;	;	;	£	ｻ	;	;	;	;	;	;	;	;	;	;	;
60	<	<	<	<	<	<	<	¥	ｼ	<	<	<	<	<	<	<	<	<	<	<
61	=	=	=	=	=	=	=	§	ｽ	=	=	=	=	=	=	=	=	=	=	=
62	>	>	>	>	>	>	>	ƒ	ｾ	>	>	>	>	>	>	>	>	>	>	>
63	?	?	?	?	?	?	?	¢	ｿ	?	?	?	?	?	?	?	?	?	?	?
64	@	@	@	@	@	@	@	â	ﾀ	@	@	É	@	§	à	@	§	§	§	@

DECIMAL VALUE	SET 20	21	22	23	24	25	26	27	28	29	50	51	52	53	54	55	56	57	58	59
65	A	A	A	A	A	⊡	A	ê	チ	A	A	A	A	A	A	A	A	A	A	A
66	B	B	B	B	B	⏀	B	ô	ツ	B	B	B	B	B	B	B	B	B	B	B
67	C	C	C	C	C	△	C	û	テ	C	C	C	C	C	C	C	C	C	C	C
68	D	D	D	D	D	+	D	á	ト	D	D	D	D	D	D	D	D	D	D	D
69	E	E	E	E	E	×	E	é	ナ	E	E	E	E	E	E	E	E	E	E	E
70	F	F	F	F	F	◇	F	ó	ニ	F	F	F	F	F	F	F	F	F	F	F
71	G	G	G	G	G	↑	G	ú	ヌ	G	G	G	G	G	G	G	G	G	G	G
72	H	H	H	H	H	⧖	H	à	ネ	H	H	H	H	H	H	H	H	H	H	H
73	I	I	I	I	I	Z	I	è	ノ	I	I	I	I	I	I	I	I	I	I	I
74	J	J	J	J	J	Y	J	ò	ハ	J	J	J	J	J	J	J	J	J	J	J
75	K	K	K	K	K	⌘	K	ù	ヒ	K	K	K	K	K	K	K	K	K	K	K
76	L	L	L	L	L	✱	L	ä	フ	L	L	L	L	L	L	L	L	L	L	L
77	M	M	M	M	M	⧗	M	ë	ヘ	M	M	M	M	M	M	M	M	M	M	M
78	N	N	N	N	N	|	N	ö	ホ	N	N	N	N	N	N	N	N	N	N	N
79	O	O	O	O	O	✡	O	ü	マ	O	O	O	O	O	O	O	O	O	O	O
80	P	P	P	P	P	_	P	Å	ミ	P	P	P	P	P	P	P	P	P	P	P
81	Q	Q	Q	Q	Q	|	Q	î	ム	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
82	R	R	R	R	R	R	R	Ø	メ	R	R	R	R	R	R	R	R	R	R	R
83	S	S	S	S	S	S	S	Æ	モ	S	S	S	S	S	S	S	S	S	S	S
84	T	T	T	T	T	T	T	å	ヤ	T	T	T	T	T	T	T	T	T	T	T
85	U	U	U	U	U	U	U	í	ユ	U	U	U	U	U	U	U	U	U	U	U
86	V	V	V	V	V	V	V	ø	ヨ	V	V	V	V	V	V	V	V	V	V	V
87	W	W	W	W	W	W	W	æ	ラ	W	W	W	W	W	W	W	W	W	W	W
88	X	X	X	X	X	X	X	Ä	リ	X	X	X	X	X	X	X	X	X	X	X
89	Y	Y	Y	Y	Y	Y	Y	ì	ル	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
90	Z	Z	Z	Z	Z	Z	Z	Ö	レ	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
91	[	[	[	Ø	[	[	[	Ü	ロ	[	Ä	Ä	Æ	Ä	°	[	°	¡	Ã	Æ
92	\	√	ç	Æ	¡	\	¥	É	ワ	\	Ö	Ö	Ø	Ö	ç	\	ç	Ñ	Ç	Ø
93	]	]	]	ø	]	]	]	ï	ン	]	Å	Å	Å	Ü	§	]	é	¿	Õ	Å
94	^	↑	^	æ	^	^	^	ß	"	^	^	Ü	^	^	^	^	^	^	^	^
95	_	_	_	_	_	_	_	Ô	°	_	_	_	_	_	_	_	_	_	_	_
96	'	`	`	'	'	'	'	Á		'	'	é	'	'	'	'	ù	'	'	'

DECIMAL VALUE	SET 20	21	22	23	24	25	26	27	28	29	50	51	52	53	54	55	56	57	58	59
97	a	a	a	a	a	∩	a	Ã		a	a	a	a	a	a	a	a	a	a	a
98	b	b	b	b	b	⊃	b	ã		b	b	b	b	b	b	b	b	b	b	b
99	c	c	c	c	c	⊂	c	Đ		c	c	c	c	c	c	c	c	c	c	c
100	d	d	d	d	d	∪	d	đ		d	d	d	d	d	d	d	d	d	d	d
101	e	e	e	e	e	¯	e	Í		e	e	e	e	e	e	e	e	e	e	e
102	f	f	f	f	f	≡	f	Ì		f	f	f	f	f	f	f	f	f	f	f
103	g	g	g	g	g	≅	g	Ó		g	g	g	g	g	g	g	g	g	g	g
104	h	h	h	h	h	≈	h	Ò		h	h	h	h	h	h	h	h	h	h	h
105	i	i	i	i	i	~	i	Õ		i	i	i	i	i	i	i	i	i	i	i
106	j	j	j	j	j	≤	j	õ		j	j	j	j	j	j	j	j	j	j	j
107	k	k	k	k	k	≥	k	Š		k	k	k	k	k	k	k	k	k	k	k
108	l	l	l	l	l	≠	l	š		l	l	l	l	l	l	l	l	l	l	l
109	m	m	m	m	m	Δ	m	Ú		m	m	m	m	m	m	m	m	m	m	m
110	n	n	n	n	n	Π	n	Ÿ		n	n	n	n	n	n	n	n	n	n	n
111	o	o	o	o	o	Σ	o	ÿ		o	o	o	o	o	o	o	o	o	o	o
112	p	p	p	p	p	±	p	Þ		p	p	p	p	p	p	p	p	p	p	p
113	q	q	q	q	q	∓	q	þ		q	q	q	q	q	q	q	q	q	q	q
114	r	r	r	r	r	→	r			r	r	r	r	r	r	r	r	r	r	r
115	s	s	s	s	s	↑	s			s	s	s	s	s	s	s	s	s	s	s
116	t	t	t	t	t	←	t			t	t	t	t	t	t	t	t	t	t	t
117	u	u	u	u	u	↓	u			u	u	u	u	u	u	u	u	u	u	u
118	v	v	v	v	v	∫	v	–		v	v	v	v	v	v	v	v	v	v	v
119	w	w	w	w	w	÷	w	¼		w	w	w	w	w	w	w	w	w	w	w
120	x	x	x	x	x	*	x	½		x	x	x	x	x	x	x	x	x	x	x
121	y	y	y	y	y	∇	y	ª		y	y	y	y	y	y	y	y	y	y	y
122	z	z	z	z	z	•	z	º		z	z	z	z	z	z	z	z	z	z	z
123	{	π	¨	¨	~	{	{	«		{	ä	ä	æ	ä	é	{	à	˚	ã	æ
124	|	⊢	˙	˙	~	|	|	□		|	ö	ö	ø	ö	ù	|	ò	ñ	ç	ø
125	}	→	¨	¨	~	}	}	»		}	å	å	å	ü	è	}	è	ç	õ	å
126	~	~	'	˙ ˙	~	~	~	±		¯	¯	ü	¯	ß	¨	¯	ì	~	˙	~

Subject Index

a

b

c

Subject Index (Continued)

c (Continued)

d

d (Continued)

Subject Index (Continued)

e

Subject Index (Continued)

e (Continued)

f

Subject Index
(Continued)

Subject Index
(Continued)

j

l

m

n

n (Continued)

Subject Index
(Continued)

o

Subject Index
(Continued)

O (Continued)

P

p (Continued)

r

s

Subject Index
(Continued)

S (Continued)

t

u

Subject Index
(Continued)

V

W

X

Y

Owner's Info
Plotter Operation
Scaling & Windows
Plotting
Circles & Polygons
Plot Enhancement
Labeling Plots
Digitizing
Gen. Prog. Inst.
Using Instructions
Roll-Feed Plotting
HP-IB Interface
RS-232-C Interface
HP-IB Overview
Syntax Summary
Reference Material

HEWLETT
PACKARD

PART NO. 07580-90034
MICROFICHE NO. 07580-90084

NOVEMBER 1985
PRINTED IN U.S.A.