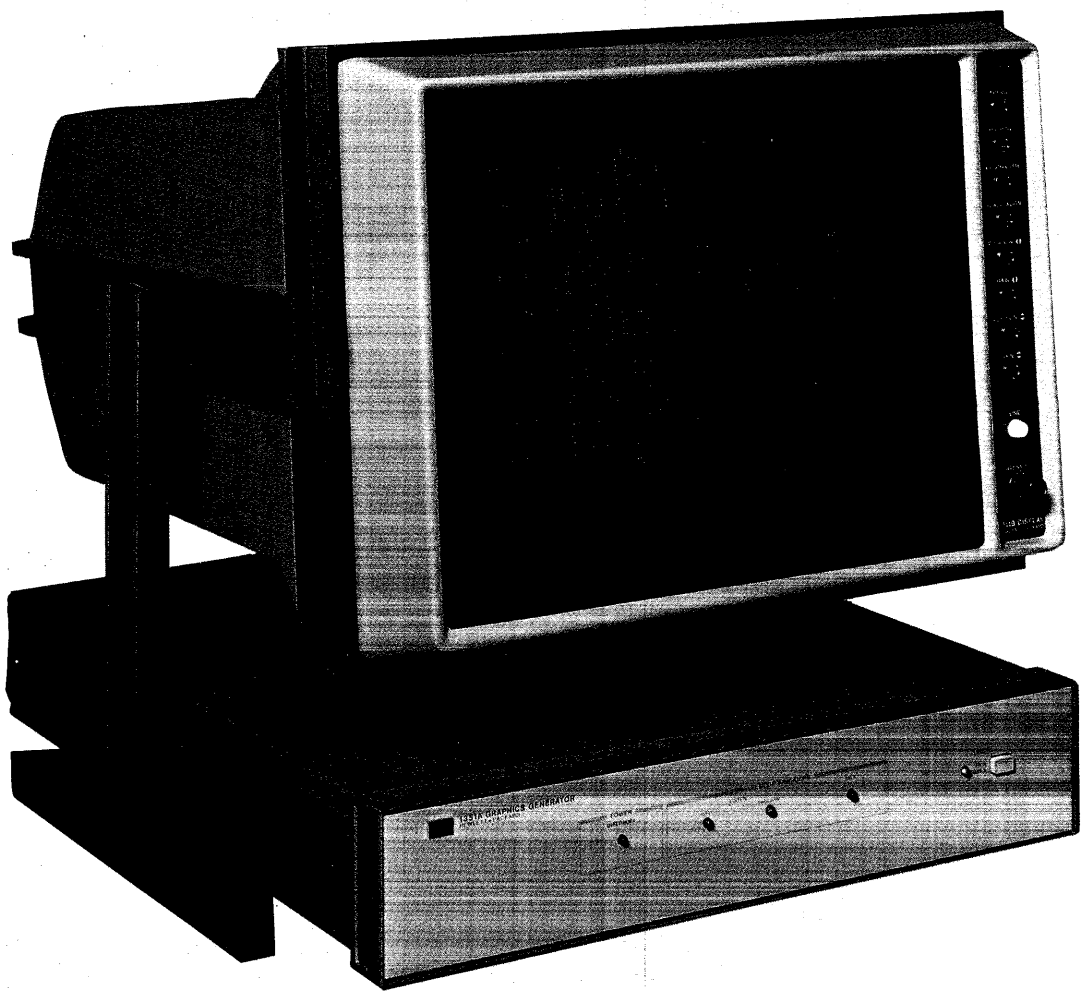
Operating and Programming Manual

1351A Graphics Generator



HEWLETT PACKARD

SAFETY SUMMARY

The following general safety precautions must be observed during all phases of operation, service, and repair of this instrument. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the instrument. Hewlett-Packard Company assumes no liability for the customer's failure to comply with these requirements.

GROUND THE INSTRUMENT.

To minimize shock hazard, the instrument chassis and cabinet must be connected to an electrical ground. The instrument is equipped with a three-conductor ac power cable. The power cable must either be plugged into an approved three-contact electrical outlet or used with a three-contact to two-contact adapter with the grounding wire (green) firmly connected to an electrical ground (safety ground) at the power outlet. The power jack and mating plug of the power cable meet International Electrotechnical Commission (IEC) safety standards.

DO NOT OPERATE IN AN EXPLOSIVE ATMOSPHERE.

Do not operate the instrument in the presence of flammable gases or fumes. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.

KEEP AWAY FROM LIVE CIRCUITS.

Operating personnel must not remove instrument covers. Component replacement and internal adjustments must be made by qualified maintenance personnel. Do not replace components with power cable connected. Under certain conditions, dangerous voltages may exist even with the power cable removed. To avoid injuries, always disconnect power and discharge circuits before touching them.

DO NOT SERVICE OR ADJUST ALONE.

Do not attempt internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

DO NOT SUBSTITUTE PARTS OR MODIFY INSTRUMENT.

Because of the danger of introducing additional hazards, do not install substitute parts or perform any unauthorized modification of the instrument. Return the instrument to a Hewlett-Packard Sales and Service Office for service and repair to ensure that safety features are maintained.

DANGEROUS PROCEDURE WARNINGS.

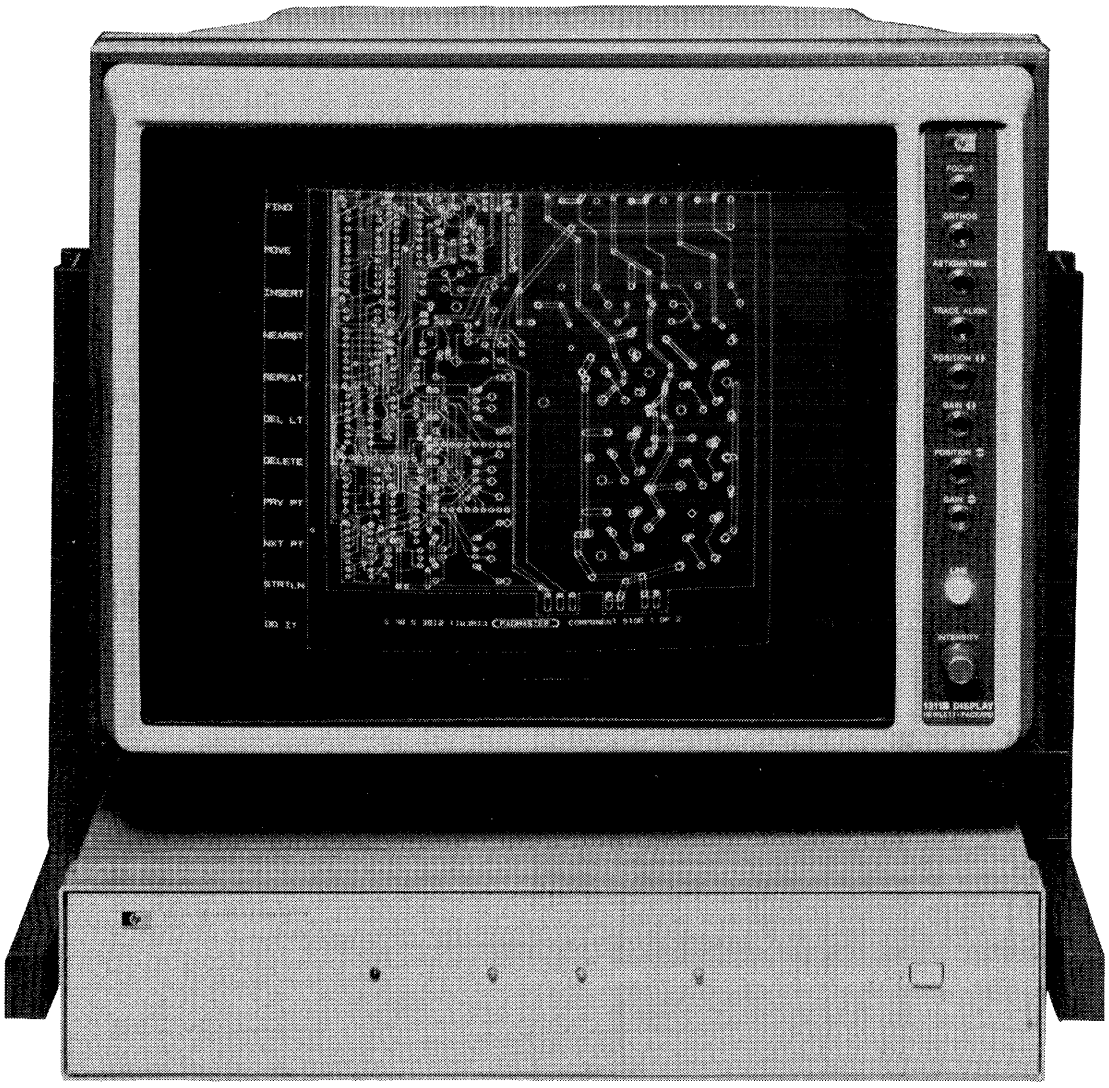
Warnings, such as the example below, precede potentially dangerous procedures throughout this manual. Instructions contained in the warnings must be followed.

WARNING

Dangerous voltages, capable of causing death, are present in this instrument. Use extreme caution when handling, testing, and adjusting.

Operating and Programming Manual

Model 1351A
Graphics Generator



MANUAL PART NO. 01351-90905
MICROFICHE PART NO. 01351-90805

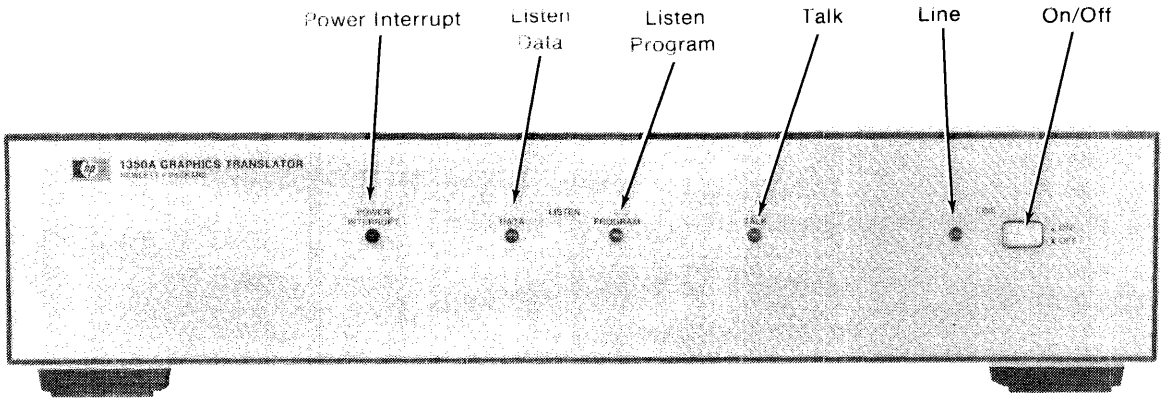
PRINTED: OCTOBER 1981

Table of Contents

Tables

Figures

Overview



Model 1351A Front-panel Controls and Connectors

Power Interrupt - Indicates voltage changes in AC power that could cause HP-IB bus information or information stored in 1351A memory to be altered.

Listen Data - Indicates data is being transferred from the controller to the 1351A.

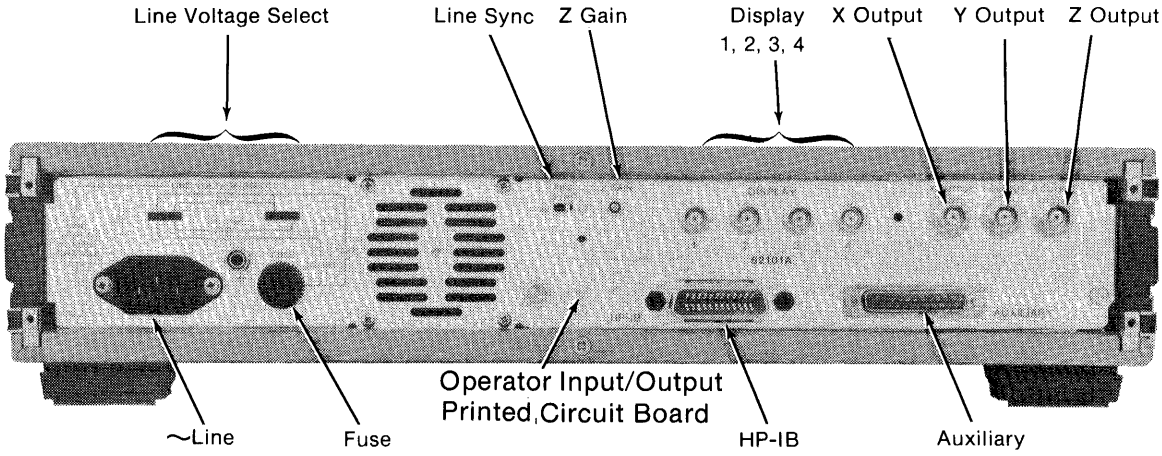
Listen Program - Indicates the 1351A has been addressed by the controller and the 1351A is "listening," ready to receive information from the controller.

Talk - Indicates the controller has given the 1351A a talk address and is waiting to receive data from the 1351A. This state should never occur, as the 1351A is not configured to transmit data to to the controlling computer (HP-IB).

Line - Indicates if AC power line is on or off.

On/Off - Turns AC power line on or off.

Overview



Model 1351A Rear-panel Controls and Connectors

Line Voltage Select - Selects desired AC line voltage.

Line Sync - Synchronizes 1351A to AC power line.

Z Gain - Adjusts output amplitude of Z axis signal.

Display 1, 2, 3, 4 - Provides TTL blanking to four separate displays.

X Output - X axis analog output signal for display input.

Y Output - Y axis analog output signal for display input.

Z Output - Z axis analog output signal for display input.

~Line - Connector for AC power cord.

Fuse - AC line protection.

HP-IB - Connector for Hewlett-Packard Interface Bus Cable (from controller).

Auxiliary - Connector that has TTL blanking outputs (Display 1, 2, 3, 4) as well as information to control the intensity of Hewlett-Packard displays via the display's binary Z-axis input.

Chapter 1
Introduction

General

The purpose of the Operating and Programming Manual is to acquaint the 1351A GRAPHICS GENERATOR user with the system features and capabilities. Throughout the discussions, it is assumed that the 1351A is being used in combination with one or more CRT displays. In order to achieve a thorough understanding of the total performance available, the manual discusses three principals and interrelated areas: programming, hardware, and specific examples. Each area compliments the others, and the material should be considered as a whole. This manual and the Operating and Service manuals for the 1351A and the appropriate CRT display provide a complete system documentation package.

The manual is organized into six chapters as follows:

- **Chapter 1, Introduction**
  Manual organization, attributes of the 1351A, and system description.
- **Chapter 2, Graphics Overview**
  Concepts for using 1351A based graphics.
- **Chapter 3, General Programming**
  Instructions, syntax, function formatting, and initialization.
- **Chapter 4, Hardware**
  Functional block diagram, memory organization and structure, and display considerations.
- **Chapter 5, Extras**
  Unique features of the 1351A for controlling multiple displays and display files.
- **Chapter 6, Examples**
  Four different examples, each given in HPL, BASIC, and FORTRAN language.

The material in this manual is sufficiently complete to help implement the 1351A on any computer or controller system. The only requirement is that the host system support HP-IB (IEEE 488), RS-232C, or 16-bit parallel with a hardware interface and an appropriate software device handler. In addition, the material in the manual is organized and indexed so as to serve as a quick reference guide.

Attributes

Since the 1351A Graphics Translator provides a collection of graphics features and capabilities not previously available in one product, it is worthwhile to list the attributes.

The 1351A, when combined with an HP high-resolution display, gives excellent directed-beam graphics. In addition, it is an inexpensive computer peripheral which provides general-purpose graphics:

- generates both lines (vectors) and text characters
- supports from 1-4 displays simultaneously and independently
- capable of high-speed, interactive display updating
- has available plug-in interface modules for parallel HP-IB (IEEE-488), serial RS-232C, or 16-bit parallel operation
- accepts high-level software commands.

The 1351A is also a cost-effective solution to a wide range of demanding graphics display application:

- reduces the host computer system overhead by providing external vector memory for updating and refreshing
- reliability is very high
- power requirements are low; nominally 150 watts for the 1351A and an HP large screen display.

Description

From a functional point of view, the 1351A Graphics Translator consists of:

1. a digital interface which receives ASCII coded instructions and data (HP-IB, RS-232C, or 16-bit parallel),

2. a graphics processor which controls the operations of the 1351A,

3. a digital memory which contains a description of the picture elements (in the case of a line, each of the locations of the end points), ASCII character and special control functions,

4. a vector generator which outputs the analog voltage to drive one of the HP directed beam X-Y-Z displays, and

5. four digital outputs which are used for controlling multiple displays.

Programming the 1351A is very similar to programming an intelligent plotter. There are 18 basic instructions which are expressed as a simple two character mnemonic. The instructions are centered around the concept that a graphics picture can be created by the positioning and status of a pen and by controlling files which contain pen strokes and/or character text. Table 3-1 in Chapter 3 contains a summary of the instructions.

Since the 1351A is an intelligent peripheral with its own instruction set, it can be implemented on any computer or controller system which can transmit an ASCII string of characters. This feature allows the 1351A to be used with any of the popular software languages (i.e., HPL, BASIC, FORTRAN, and PASCAL) without the need of special software device handlers. In addition, the 1351A can be easily integrated into existing user application programs or graphics software packages.

Chapter 2
Graphics Overview

General Features

The 1351A and a CRT display provide the same functions as a conventional high resolution plotter by presenting lines and alphanumeric characters. In addition, it has the extremely important feature of high speed selective memory updating. Any single memory location or one of the 64 data files can be as easily changed as all of the memory. This feature provides the capability of dynamic display interaction. By simply changing the end points of a line, it is automatically redrawn in the new position during the next display refresh cycle. It should be remembered that the lines on the display are truly continuous lines and not a series of dots.

Another feature of the 1351A is that all input and output operations are fully interleaved. From a programming point of view, the 1351A continues to up-date the CRT display while new instructions and data are being transmitted to it. This feature, plus the above mentioned features, supplies user written software with the capabilities of image rotation, translations, and three-dimensional rotations.

Concepts

In order to obtain the total benefits of a dynamic display system, the operation of the 1351A should be viewed as sketching a picture or filming a movie. Each picture and segment of the picture should be planned. In general, foreground features change more quickly and tend to be the center of attention. For this reason, it is often convenient to group like features in a single software/hardware file. This provides the convenience of updating a single file to change a feature or function. It also supplies a framework for order of priority when updating. For instance, the foreground may be updated hundreds of times for each time the background is changed.

Variation

After reviewing the manual, the user is encouraged to try several variations of a display theme. Consider, for example, a simple straight line. Fix one of the end points and consecutively vary the other end point as a mathematical function $y = f(x)$. The display might contain a family of lines fanning out or one line moving by itself.

As an extention of the moving line theme, the next example could be the use of a square or rectangle. In this case, plot a four-sided figure on the display and fix two opposite corners while moving the other two. The result appears to be that of stretching a rubber band or perhaps revolving a surface about an axis.

The last suggestion for a variation is meant to demonstrate the capabilities for three-dimensional rotation and the relationship to file structures. A sense of three-dimensional rotation of an object can be obtained by placing various views in separate files, then sequentially displaying each view separately.

The following chapters will provide the details for implementing these simple examples and assist the user with specific applications.

Chapter 3
General Programming

Introduction

This chapter discusses how to program the 1351A Graphics Translator from the viewpoint of what the 1351A must receive in order to perform its functions.

The first part of this chapter provides the formal programming rules.

The rest of this chapter is separated into the major 1351A functions of: (a) Initialization; (b) Drawing Vectors; (c) Displaying Text; and (d) File Operations.

Detailed explanations of 1351A commands are given under the functional areas where they most directly apply. Table 3-1 is a listing of 1351A functions and their command subsets.

Table 3-1. 1351A Functions and Command Subsets

FUNCTION	ASCII COMMANDS		FUNCTION	ASCII COMMANDS	
Initialization	EM	Erase Memory	File Operations	NF	Name File
	EN	Erase Names		SN	Stop Naming
	EX	Erase AuXiliary		FF	Find File
	BM	Blank Memory		EF	Erase File
	UM	Unblank Memory		BF	Blank File
				UF	Unblank File
Drawing Vectors	PA	Plot Absolute		FL	Find Location
	PE	Pen Enable			
			Multiple Displays	WX	Write AuXiliary
Displaying Text	CS	Character Size	(see Chapter 5)	SX	Stop AuXiliary
	TX	TeXt			

Formatting Rules

1. The 1351A receives all instructions in standard ASCII code; this is true for both the serial RS-232C interface and the parallel HP-IB interface. The 16-bit parallel interface operates in machine language. Detailed programming instructions for the 16-bit parallel interface are contained in the option 002 Operating and Service Manual.

2. All instruction commands are two letters (ASCII alpha characters).

3. The two-letter instruction commands may be either uppercase or lowercase. This manual will use uppercase characters.

4. All parameters are ASCII numeric characters (0 thru 9).

5. Each command consists of:

   a. The two-letter command code.

   b. A variable length parameter field (zero or more numerics, or a text character string).

   c. A delimiter following any parameters.

   d. A command terminator.

6. No spaces are allowed: (a) in front of a two-letter command; (b) between the two letters of a command; and (c) immediately prior to a terminator.

7. Following a Text (TX) command, the 1351A must receive an ETX (End of Text) character in order to exit text mode. After the ETX, the Text command must be terminated by a colon, or CR, or LF.

NOTE

The ASCII End of Text (ETX) character can be obtained on most terminals by pressing the Control (CNTL) and capital C keys at the same time.

Syntax

Table 3-2 gives a dictionary of parameter ranges for the command variables used by the 1351A.

< > Angle brackets denote a syntactical variable. The user must supply a value from the specified range of possibilities given for the item.

[ ] Square brackets denote required delimiters.

Table 3-2. Parameter Items and Ranges

ITEM	RANGE
<abs X>	0 . . 1020
<abs Y>	0 . . 1020
<enable>	0 . . 1
<size>	0 . . 7 with 8 . . 15 for special characters.
<location>	0 . . 8191
<file>	0 . . 63
<display>	0 . . 15
<string>	Any group of ASCII characters, excluding the ETX (End of Text) character (3 in base 10).
<terminator>	: or CR or LF

Command Terminators

Table 3-3 is a listing of 1351A command syntax. The commands are listed in alphabetical order.

A colon (:), Carriage Return (CR), or Line Feed (LF) character must be used to terminate each command. In table 3-3 a <terminator> is the ending condition of each command.

If more than one <terminator> is received, the 1351A responds as if only one <terminator> was received. For example, a colon at the end of a line may automatically be followed by CR/LF, as is often generated by interface drivers. Even though three <terminator> characters have been received, the 1351A only uses the first one.

NOTE

If parameter ranges are exceeded, or if formatting, syntax, and programming rules are not followed, then undesirable (and sometimes very confusing) displays can result.

Table 3-3. Command Syntax for 1351A ASCII Commands

COMMAND NAME	SYNTAX
Blank File	BF<file>[,]<terminator>
Blank Memory	BM<terminator>
Character Size	CS<size>[,]<terminator>
Erase File	EF<file>[,]<terminator>
Erase Memory	EM<terminator>
Erase Names	EN<terminator>
Erase AuXiliary	EX<terminator>
Find File	FF<file>[,]<terminator>
Find Location	FL<location>[,]<terminator>
Name File	NF<file>[,]<terminator>
Plot Absolute	PA<abs X>[,]<abs Y>[;]<terminator>
Pen Enable	PE<enable>[,]<terminator>
Stop Naming	SN<terminator>
Stop AuXiliary	SX<terminator>
TeXt	TX<string>[end text]<terminator>
Unblank File	UF<file>[,]<terminator>
Unblank Memory	UM<terminator>
Write AuXiliary	WX<display>[,]<terminator>

NOTE: Each comma (,) and semicolon (;) is a required item in the command string. The [end text] delimiter is the ASCII End of Text character (ETX = 3 in base 10).

Programming Rules

1. The 1351A follows the last Pen Enable (PE) command received (CRT beam on or off). PE can be overridden by Blank Memory (BM) or Blank File (BF) commands.

2. The Character Size (CS) command determines the size and orientation of text. When responding to a text command, the 1351A uses the last CS command received.

3. 1351A memory is made up of up to 8192 words. Each word may be either: (a) a vector endpoint coordinate pair (X,Y); or (b) an ASCII Text character.

4. 1351A memory can be partitioned into a maximum of 64 file areas. Each file can contain from one to 8192 words of memory. Total allocated file area must not exceed 1351A maximum memory size of 8192 words.

5. Stop Naming (SN) command should follow the last vector or character entered into a file.

6. A file does not exist until named and data entered via PA or TX commands. (Note: All words in 1351A memory default to file zero when an EN (Erase Names) command is received.)

7. Following a command that requires a parameter a data field exists until delimited by a comma (,). This data field is of variable length. In the data field, only the last four characters will be used by the 1351A. If less than four digits are sent, the 1351A inserts leading zeroes. If more than four characters are placed in the data field, leading spaces for example, the 1351A will ignore all except the last four characters preceeding the comma. Any characters in these last four positions will be treated as digits by the 1351A.

   Example:

   "NFXXX...XXDDDD,:" X=don't care; D=ASCII digit

NOTE

Data fields must contain integer values only in the last four positions preceeding the comma. If computed values (for example TAN X) are done in floating point, be careful that only integer results are sent to the 1351A (truncate). No decimal point is allowed.

---

8. Following a PA command, the X and Y entries both follow the above rule. The X data field is delimited by a comma (,). The Y data field is delimited by a semicolon (;). The Y data field (preceeding the semicolon) follows the same rules as do the data fields that are delimited by a comma.

NOTE

Remember:

X = 0 thru 1020
Y = 0 thru 1020

with NO sign or decimal point.

Required Initialization Sequence

Whenever the 1351A is powered on, or whenever a new program is to be sent to the 1351A, a specific initialization sequence is required. This sequence provides the proper starting point by "clearing out" the 1351A.

The following items must be sent to the 1351A in the order given below. This insures that the 1351A will initialize properly after power-on, or an interrupt.

[ETX][DC4]:EM:EN:EX:SN:SX:UM:

1. ETX — The ASCII End of Text character (3 in base 10) makes sure that the 1351A is taken out of a possible Text mode "wake up" condition. When in Text mode, the 1351A assumes that anything received is to be displayed as characters - including command terminators and control characters.

2. DC4 — The ASCII DC4 character (20 in base 10) turns the POWER INTERRUPT indicator off. This allows the 1351A to warn the operator if a power disruption occurs that may affect operation. POWER INTERRUPT is automatically turned on whenever the 1351A is powered on. Thus it must be turned off during initialization in order to indicate any subsequent power disruption.

NOTE

On most terminals, the ETX character can be obtained by pressing the Control [CNTL] and capital C keys at the same time. The DC4 character can be obtained by pressing [CNTL] and capital T keys at the same time.

---

3. : The colon is used as a <terminator> so that the 1351A will receive the next two characters as a two-letter command. ASCII CR or LF could replace the colon.

4. EM: The EM commands the 1351A to erase its memory. This "cleans out" all vector and pen enable values that are present due to the random power up of memory. The colon provides the <terminator> for the EM command.

5. EN: The EN commands the 1351A to erase all file names present due to random memory power up. Colon terminates EN command.

6. EX: Erases all multiple display blanking information from 1351A memory after random power up. Colon terminates EX command.

7. SN: Stop Naming insures that no files are named until the 1351A is specifically instructed to do so. See the File Operations section of this chapter. Colon terminates SN command.

8. SX: Stop Auxiliary insures that no data is assigned to multiple display blanking combination(s) until the 1351A is specifically instructed to do so. Multiple display blanking is covered in Chapter 5 of this manual. Colon terminates SX command.

9. UM: Unblank Memory takes the 1351A out of a possible "wake-up" condition where the memory could be in a non-viewable (blanked) state. In essence this command sets the entire memory to a viewable state (unblanked). Colon terminates UM command.

The following commands are discussed in detail under the Initialization function: (1) EM, Erase Memory; (2) EN, Erase Names; (3) EX, Erase Auxiliary; (4) BM, Blank Memory; and (5) UM, Unblank Memory.

Erase Memory Command

a. Syntax. EM<terminator>

b. Explanation.

Erase Memory erases all vector coordinate endpoints, text characters, and pen status (display beam on or off) from 1351A memory. This results in all 8192 words of memory containing vectors of zero length and pen status of "beam off." This command is normally used for initialization only.

Erase Memory also moves the Write Pointer to memory location zero. The Write Pointer is discussed in the WRITE POINTER section of this chapter.

NOTE

Each word of the 8192-word memory may contain either a vector endpoint or a text character. This is defined as the data content. File, Pen Enable, Memory Blanking, and Auxiliary information are defined as status information. Each of the data words are accompanied by this status information.

Erase Names Command

a. Syntax. EN<terminator>

b. Explanation.

Erase Names erases all file names from 1351A memory. This results in all words of memory being assigned to file zero. Data contained in the memory is not changed. Normally this instruction is used for initialization in order to clear out undesired file boundaries from memory.

Erase Auxiliary Command

a. Syntax. EX<terminator>

b. Explanation.

Erase Auxiliary removes all multiple display blanking information from 1351A memory. This results in all 8192 words of 1351A memory being assigned to the "unblank all rear-panel DISPLAY outputs (1-4)" combination. Data contained in the memory is not changed. Normally this insruction is used for initialization only.

Blank Memory Command

a. Syntax. BM<terminator>

b. Explanation.

Blank Memory prevents the 1351A from displaying any information. The Pen status (beam on or off), File unblanking, and multiple display unblanking are overridden. Data and status in the memory are are not changed.

This instruction is normally used when sending data to the 1351A in a batch mode. Memory is blanked (display inhibited), data loaded, then memory is unblanked so that the picture can be seen instantaneously.

NOTE

The hierarchy of blanking in the 1351A provides several levels for controlling the status of the display beam (on or off).

1. Blank Memory turns the beam completely off.
2. Blank File (BF) turns the beam off for all data in a particular file.
3. Pen Enable (PE) turns the beam on and off to draw pictures when not overridden by BM or BF.

Unblank Memory Command

a. Syntax. UM<terminator>

b. Explanation.

Unblank Memory is the complement of Blank Memory.

Unblank Memory allows the 1351A to display information according to Pen status, File blank/unblank, and multiple display instructions.

DRAWING VECTORS

This section tells how to use the Plot Absolute (PA) and Pen Enable (PE) commands in order to "draw" vectors on a display.

Display Area

The display area is defined by the limits of the X and Y values possible. The X axis range is from 0 to 1020. The Y axis range is from 0 to 1020.

The (0,0) point is at the bottom left-hand corner of the display. The (1020,1020) point is at the top right-hand corner. See figure 3-1.

All vectors are drawn in the display area according to the "vector endpoint" rule. This specifies that the X and Y values following a PA instructon determine the next beam location (endpoint) as an absolute value. For example, if X=1020 and Y=0 in a PA command (PA1020,0;:) the beam will move to the bottom right-hand corner of the display, no matter what its previous location.

After an EM (Erase Memory) command has been received during Initialization, all 8192 words of 1351A memory contain the coordinate pair (0,0).

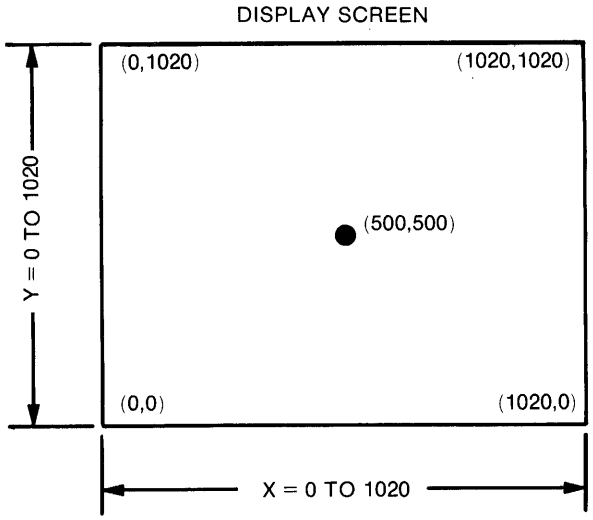


Figure 3-1. Vector Drawing Area as Defined by X and Y Values

Plot Absolute Command

a. Syntax.

PA<abs X>[,]<abs Y>[;]<terminator>

<abs X> = 0 to 1020
<abs Y> = 0 to 1020

b. Explanation.

The X and Y values determine the absolute location of the display beam (vector endpoint). Each X and Y coordinate pair occupies one location in 1351A memory. Thus the 1351A is capable of causing up to 8192 vectors to be drawn on the display.

The display beam moves sequentially to each new coordinate point from the previous coordinate point in the order that the coordinate points (X,Y) are received by the 1351A.

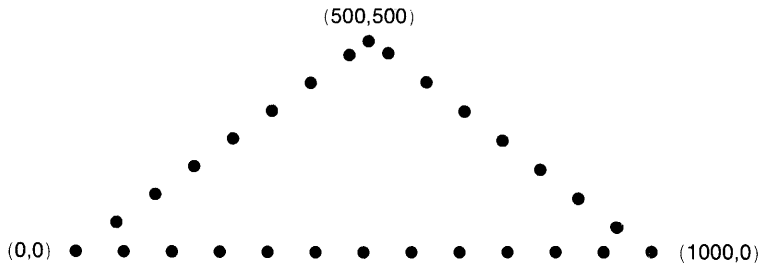
Several vector endpoints may be sent to the 1351A after a PA command by using the following syntax.

PA<absX>[,]<absY>[;]<absX>[,]<absY>[;] ... <absX>[,]<absY>[;]:

For example, let's assume that the beam position is at the (0,0) coordinates and the ASCII string shown below is sent to the 1351A.

PA500,500;1000,0;0,0;:

The beam will move in a pattern that describes an isosceles triangle on the bottom half of the display as shown below.



Pen Enable Command

a. Syntax.

PE<enable>[,]<terminator>

<enable>=0 for beam off
<enable>=1 for beam on

b. Explanation.

The PE parameter determines whether vector(s) are to be visible (display unblanked by PE1,:) or invisible (display blanked by PE0,:).

For a vector to be visible, the display must be set to "beam on" condition (unblanked).

The 1351A follows the last PE command received in order to determine whether a vector is to be visible or invisible on the display.

After an EM command has been received during Initialization, all 8192 words of 1351A memory are assigned Pen status of "beam off."

Even though a PE command specifies "beam on," it can be overridden by a Blank Memory (BM) or a Blank File (BF) command.

The blanking hierarchy is as follows:

1. Blank Memory turns the display beam completely off. Unblank Memory allows the picture to be shown according to file and pen unblanking.

2. Blank File overrides Pen Enable unblanking for all data in the file being blanked. Unblank File allows the data in the file (vectors and/or characters) to be shown according to Pen Enable unblanking.

3. Pen Enable turns the display beam on (unblank) and off (blank) in order to produce the picture.

Vector Drawing Example

In the following example assume that the 1351A has been Initialized. All 8192 words in memory contain the vector coordinates (0,0) so that the beam is positioned in the bottom left-hand corner of the display. Also, all words have been assigned the Pen status of "beam off." The 1351A memory has been unblanked by the UM command so that the beam can be turned on and off by PE commands.

If we wish to "draw" two lines on the display as shown in figure 3-2a, the 1351A must receive the following ASCII string.

PE0,:PA200,300;:PE1,:PA800,300;:PE0,:PA800,700;:PE1,: PA200,700;:

Let's break the "drawing" into the four vectors necessary to produce the display as shown in figure 3-2b.

NOTE

The vectors shown by " • " are those drawn with beam off.

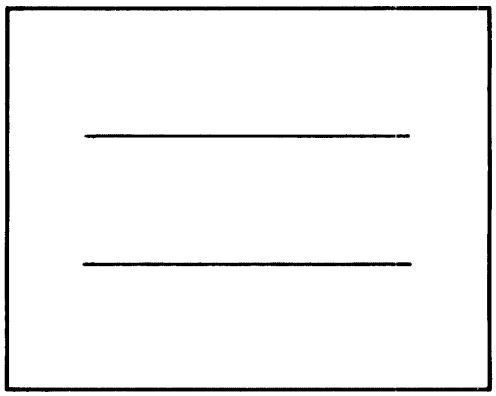


Figure 3-2a. Desired Example Display of Two Lines

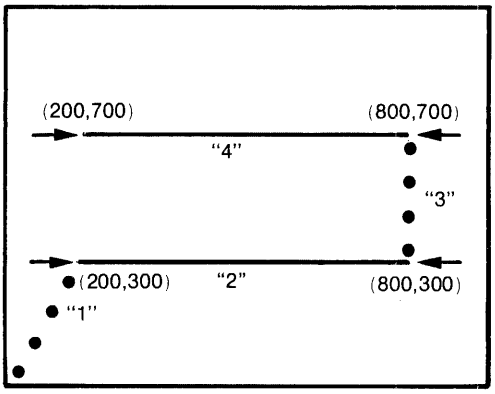


Figure 3-2b. Visible and Invisible Vectors Required to Draw the Picture Shown in Figure 3-2a

Vector "1" is produced by the "PE0,:PA200,300;:" portion of the command string. The beam is turned off and moved to the X-Y coordinates of (200,300).

Vector "2" is produced by the "PE1,:PA800,300;:" portion of the command string. The beam is turned on and moved to location (800,300).

Vector "3" is produced by the "PE0,:PA800,700;:" portion of the command string. The beam is turned off and moved to location (800,700).

Vector "4" is produced by the "PE1,:PA200,700;:" portion of the command string. The beam is turned on and moved to location (200,700).

It is important to remember that each movement of the beam produces a vector, whether or not it is to be visible in the drawing.

Drawing With Variables

In many applications it is desirable to "draw the picture" by using variables rather than fixed numbers.

When computations are used to produce the X and Y parameters for PA commands, make sure to use format statements that produce four-place fixed numbers (no decimal point).

Also, clipping should be performed on vector endpoints to insure that computed X and Y values are within their permissible ranges.
(Note: X = 0 to 1020; Y = 0 to 1020 with no sign or decimal.)

Displaying Text

This section explains how to use the Text (TX) and Character Size (CS) commands in order to display text.

For example, this feature allows the 1351A to label drawings or display program listings.

Each character to be displayed occupies one location in the 1351A 8192-word memory. This also includes space characters used in the text (32 in base 10 = space) and text mode control characters such as Carriage Return (CR) and Line Feed (LF).

Preconditioning the 1351A for Text Mode

Before text mode is entered the following sequence should be used. This sets the 1351A up so that the desired text will be properly displayed.

1. Turn the display beam off (PE0,:).

2. Move the display beam to the desired starting point for the text (PAX,Y;:). (Note: Remember that this vector uses one word of the 1351A memory.)

3. Set the size and rotation for text via a Character Size (CS) command. The last CS command received will control the size and rotation of the text to be displayed.

4. There is no need to use the PE1 command to display text. The TX command automatically turns on the display beam.

Character Size Command

a. Syntax.

CS<size>[,]<terminator>

			Character Area (in addressable points)		
<SIZE>	CHARACTER SIZE	ROTATION	WIDTH		HEIGHT
0	X 1	0 degrees	12	X	20
1	X 2	0 degrees	24	X	40
2	X 4	0 degrees	48	X	80
3	X 8	0 degrees	96	X	120
4	X 1	90 degrees CCW	12	X	20
5	X 2	90 degrees CCW	24	X	40
6	X 4	90 degrees CCW	48	X	80
7	X 8	90 degrees CCW	96	X	120

The following CS commands apply only when an optional alternate character ROM is installed in the 1351A.

			Character Area (in addressable points)		
<SIZE>	CHARACTER SIZE	ROTATION	WIDTH		HEIGHT
8	X 1	0 degrees	12	X	20
9	X 2	0 degrees	24	X	40
10	X 4	0 degrees	48	X	80
11	X 8	0 degrees	96	X	120
12	X 1	90 degrees CCW	12	X	20
13	X 2	90 degrees CCW	24	X	40
14	X 4	90 degrees CCW	48	X	80
15	X 8	90 degrees CCW	96	X	120

b. Explanation.

The CS command allows one of the four available sizes of text to be displayed. CS also allows text to be displayed either normally or else rotated 90 degrees counterclockwise (CCW).

When using the smallest size, 82 characters per line can be displayed. Since each character (both displayed and control, such as CR and LF) occupies one of the locations in 1351A memory, be careful in considering how many lines of text to display at one time.

If the smallest size is being used, for example, make sure to send a CR and an LF after each 82nd character to position the display beam for the next line of text. Remember to include both the CR and LF characters in the total when determining how many words of 1351A memory are being used.

Spacing between characters and lines is provided automatically by the 1351A to facilitate readability. This spacing is a function of the character size selected. The area for each character is equal to approximately 1.66 times the height and 1.5 times the width of the character. This "open area" is provided above and to the right of each character for spacing between characters and lines. Character area is diagrammed in figure 3-3 using a capital "T" character.

The automatic spacing causes each character to end at the starting point for the next character. This advancement of the beam causes the 1351A to function much like a typewriter.

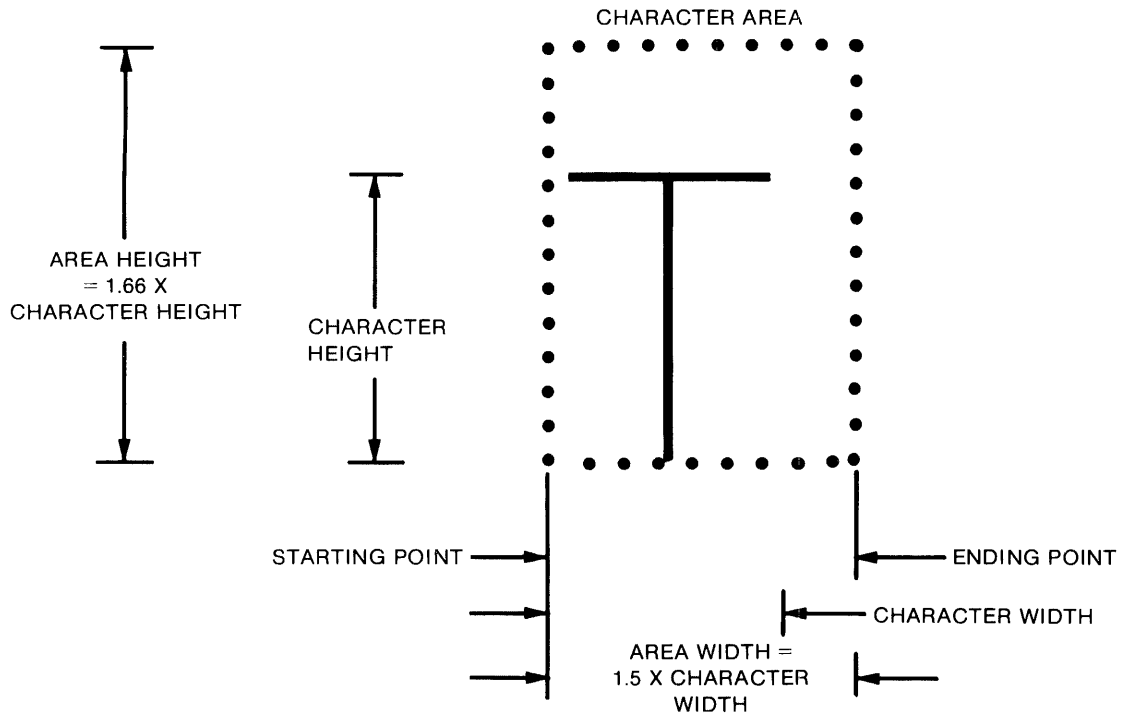


Figure 3-3. Automatic Spacing Provided by Character Area

For non-rotated text, the starting point for each character is the lower left-hand corner of the character area. For rotated text, the entire character is rotated 90 degrees CCW about this point. When displaying text near the edge of the allowable display area be careful that the display beam position will not cause part of the text to be unreadable.

The characters can all pile up on top of one another when the display beam is not properly positioned via either Plot Absolute (outside of text mode) or text mode control characters (when in *text mode*).

NOTE

Remember that the 1351A always uses the last CS command received when setting size and rotation of text to be displayed.

Text Command

a. Syntax.

TX<string>[end text]<terminator>

b. Explanation.

A Text (TX) command allows the 1351A to provide characters on the display without the programmer having to draw all the vectors necessary to produce each character.

The 1351A modified ASCII character set is listed in appendix E at the end of this manual.

The 1351A follows the last CS command received when displaying text (size and rotation).

Text mode is exited when the 1351A receives an ASCII ETX character (End of Text = 3 in base 10). After the ETX the 1351A must receive a command terminator (colon, CR, or LF) to end the Text command.

Everything between the TX command and the ETX is treated as text to be displayed (or text control characters such as CR or LF).

The TX command will automatically enable the pen (i.e., PE1) for the duration of the text mode. Therefore there is no need to enable the pen prior to text mode with a PE1 command.

Text Mode Control Characters

The text mode control characters listed in table 3-4 are used to position the display beam while the 1351A is in text mode.

Table 3-4. Text Mode Control Characters

DECIMAL VALUE	OCTAL VALUE	DEFINITION
8	010	Backspace (BS). Moves beam back one character area. Amount of movement is set by size of text.
9	011	Inverse Line Feed. Moves beam up one character area. Amount of movement is set by size of text.
10	012	Line Feed (LF). Moves beam down one character area. Amount of movement is set by size of text.
13	013	Carriage Return (CR). Moves beam to left-hand edge of display.

Text Mode Special Plotting Characters

Table 3-5 is a listing of special plotting characters that do not automatically advance the beam to the starting point or the next character. The beam is left at the center of the character.

These characters provide a means whereby lines on a graph can be differentiated to aid readability. (Note: See Example 4 in Chapter 6.)

Table 3-5. Text Mode Special Plotting Characters

DECIMAL VALUE	OCTAL VALUE	CHARACTER
11	013	Vertical Tick Mark (|)
12	014	Horizontal Tick Mark (—)
14	016	X marker symbol
15	017	Rectangle symbol
19	023	Pointer symbol (←)
30	036	Diamond symbol

A line can be drawn by either having the special characters connected (beam on between data points) or having them unconnected (beam off between data points).

More than one special character may be placed at each data point, thus increasing the possible number of line types.

The general method when using special plotting characters is:

1. Set the size and rotation for the special plotting characters with a Characer Size (CS) command.

2. Move the beam to the data point with a Plot Absolute (PA) command.

3. Mark the data point with the desired special character(s) via a Text (TX) command.

4. End text mode with an ETX character followed by a command terminator.

5. Repeat steps b through e for each data point on the line.

Text Display Example With Starting Point Calculation

The following ASCII string will cause the 1351A to display the text "1351A" on the center of the display. It is assumed that the 1351A has been Initialized.

PE0,:PA390,440;:CS2,:TX1351A ETX:

The "PE0,:" turns the display beam off. The "PA390,440;:" moves the display beam to the starting point for the text. "CS2,:" sets character size and rotation for the text. "TX1351A" commands text mode and specifies characters to be displayed (1351A).

How was the starting point determined? The center point of the display area is 510,510. The area for each character is 48 X 80 addressable points (CS2,:).

For the X-axis starting point:

5 characters (1351A) divided by two is 2.5 character areas; multiply 2.5 character areas by 48 points per character to get 120 addressable points; subtract 120 from 510 to get the X starting point of 390.

For the Y-axis starting point:

divide the 80 points per character by two for 40 addressable points; subtract 40 from 510 to get the Y starting point of 470.

Write Pointer

The 1351A Write Pointer specifies the word number of the next location in 1351A memory where a vector endpoint (or text character) will be placed. Each endpoint (or character) is accompanied by the current status information (Pen, File, etc.).

NOTE

The 1351A has a modulo 8192 word memory. The 8192 words are numbered 0 to 8191. If memory size is exceeded, then a wrap-around will occur. For example, location 8192 wraps around and will over-write the contents of memory word zero.

The 1351A will have only the first 2560 memory locations enabled upon power-up. When location 2560 is written to, the 1351A will open another page of 512 locations. The 1351A will continue to add additional pages of 512 locations each until the entire 8192 locations are enabled.

The Write Pointer advances sequentially through memory unless jumped to a different value by the commands:

(1) Erase Memory, EM
(2) Erase File, EF
(3) Find File, FF
(4) Find Location, FL

The Write Pointer is moved to location zero as a result of either an EM or an EF command.

The FF command moves the Write Pointer to the first word in the file specified.

The FL command moves the Write Pointer to any location in memory (regardless of whether files are being used or not).

Therefore the Write Pointer can function as a moveable cursor for placing information into 1351A memory. Since the Write Pointer will automatically advance one word at a time, the programmer is given the choice of either using or not using the mobility of the Write Pointer.

File Operations

This section explains how to create and manipulate files in the 1351A memory.

Each word in the 1351A memory may be either a vector endpoint (X,Y) or a text character.

A 1351A file is defined as a collection of memory words treated as a unit.

The 1351A memory may be sectioned into as many as 64 files. Each file can be made up of from one to 8192 words of memory. The total number of words in all files must not exceed 8192. Each file must contain at least one word.

All vectors and/or characters that have been grouped together as a file can be manipulated as one unit via 1351A file management commands.

Each file is individually addressable and can be blanked, unblanked, erased, or rewritten. This allows each group of data to be flashed on and off the display or rapidly updated.

File manipulation allows graphs to be quickly updated and compared, mechanical drawings to be "built up" or "stripteased," three-dimensional 'rotating' drawings to be created, or "animated" drawings to be produced.

The following commands are discussed in detail in this section: (a) NF, Name File; (b) SN, Stop Naming; (c) FF, Find File; (d) FL, Find Location; (e) BF, Blank File; (f) UF, Unblank File; and (g) EF, Erase File.

Creating Files

During Initialization the EN (Erase Names) command removes all file boundaries. All words in 1351A memory are assigned to file zero.

Each desired file can now be created by placing all vectors and/or characters for each file between the Name File (NF) and Stop Naming (SN) commands.

Although all words in a file do not necessarily have to be grouped together in order to function as a unit, we recommend that all data in a file be contiguous. If not grouped together sequentially in memory, then the exact location of each word in the file must be known in order to change or update the file.

Name File Command

a. Syntax.

NF<file>[,|<terminator>

<file>=0 to 63

b. Explanation.

Name File (NF) assigns each following vector endpoint (PAX,Y;:) or text character to the file specified by the <file> parameter.

Pen status (display beam on or off), and character size information (for text), automatically accompanies each vector and/or character placed in a file.

Each word in the file is written to 1351A memory starting at the current value of the 1351A memory Write Pointer.

After all desired words are placed in a file the naming process should be delimited (ended) via a Stop Naming (SN) command.

Stop Naming Command

a. Syntax.

SN<terminator>

b. Explanation.

The Stop Naming (SN) command tells the 1351A to quit assigning memory words to the last file named. This "closes" the current file.

It is possible for informtion to be lost if SN is not used properly.

Example:

The information in memory words 0003 and 0004 (part of file 1) and words 0005 thru 0007 (part of file 2) needs to be changed.

The Write Pointer is currently at word 0016 (in file 3; a "NF3,:" command has been sent and data is being assigned to file 3).

A "FL3,:" command is sent to the 1351A, moving the Write Pointer to word 0003.

A Stop Naming ("SN:") command was not sent before moving the Write Pointer.

Information in words 0003 thru 0007 is changed (with PA and/or TX commands).

Since file 3 was not "closed," words 0003 thru 0007 are now in file 3. Part of files 1 and 2 are now missing. See figure 3-4.

Considerable work would have to be done with FL, NF, and SN in order to restore the memory to proper order.

BEFORE		AFTER	
MEMORY WORD	FILE NAME	MEMORY WORD	FILE NAME
0	1	0	1
1	1	1	1
2	1	2	1
3	1	3	3
4	1	4	3
5	2	5	3
6	2	6	3
7	2	7	3
8	2	8	2
9	2	9	2
10	3	10	3
11	3	11	3
12	3	12	3
13	3	13	3
14	3	14	3
15	3	15	3
16			

Figure 3-4. Example of SN Command NOT Being Used Properly

Find Location Command

a. Syntax.

FL<location>[,]<terminator>

<location>=0 to 8191

b. Explanation.

Find Location (FL) moves the 1351A memory Write Pointer to the word number specified. The next word written into 1351A memory (vector endpoint or text character) is placed in the location specified by the FL parameter.

NOTE

The Write Pointer now returns to its normal mode whereby it automatically advances to the next memory location. Each vector endpoint (or text character) received will cause the Write Pointer to increment to the next higher memory word number (location). Example: If an FL1500,: command is used and a vector endpoint (or text character) is written into memory word 1500 (by a PA or TX command), then the Write Pointer value becomes 1501. The next endpoint (or character) received is automatically placed into location 1501 and the Write Pointer increments to 1501.

After power-on, the Write Pointer could be anywhere. If 1351A files are to be used it is good practice to first move the Write Pointer to memory word zero (FL0,:) before naming the first file. The Erase Memory command also positions the Write Pointer to word zero.

After data has been placed in a file, FL can be used to selectively change individual words in the file as long as the word number(s) of the item(s) to be changed are known. This can be accomplished fairly easily if the 1351A memory has been sectioned into fixed-length files.

Fixed Length Files

Before meaningful data is placed in 1351A memory, the memory can first be "sectioned" into files of known length and boundaries.

In order to mark fixed-length files in 1351A memory, use the following sequence.

1. Initialize the 1351A.
2. Move the Write Pointer to memory word zero with a Find Location command (FL0,:).
3. Set up a program loop for the desired number of files (FOR....NEXT or DO, for example).
4. Name each file with a Name File (NF) command on each pass through the above loop.
5. Set up a program loop that is internal to the above loop. Use this loop to assign the desired number of words to each file.
6. Send the 1351A a "dummy vector" on each pass through the internal loop. The "dummy vector" (PE0,:PA0,0;:) provides each word in the file with a required content so that the file truly exists and can be later addressed.
7. Send the 1351A a Stop Naming (SN:) command after the internal loop has completed all its iterations. This terminates the naming process before the outer loop increments to start naming the next file.

Placing a "dummy vector" in memory locations greater than 2059 will cause additional pages of memory to be enabled. If the specific application of the 1351A is sensitive to display flicker, do not place "dummy vectors" into unneeded memory pages. For more information on the operation of the 1351A memory, please refer to the Memory Map section of Chapter 4.

Example:

The following is a general example using FOR....NEXT loops.

a. Send the 1351A its Initialization sequence.

b. Send the 1351A the ASCII string "FL0,:".

c. FOR A=1 TO 3

d. Send the 1351A NF A followed by ASCII "," and ":" (or CR/LF).

e. FOR N=1 TO 500

f. Send the 1351A the ASCII string PE0,PA0,0;:

g. NEXT N

h. Send the 1351A the ASCII string SN:

i. NEXT A

j. END

The above example causes four files to be created in the 1351A memory. Files "1", "2", and "3" will each have 500 word allocated. Since the 1351A was Initialized at the start, the remainder of memory defaults to file "0".

File "1":	memory words	0-499.
File "2":	memory words	500-999.
File "3":	memory words	1000-1499.
File "0":	memory words	1500-2047.

The Find File (FF) command can now be used to place meaningful data in each file.

Find File Command

a. Syntax.

FF<file>[,]<terminator>

<file>=0 to 63

b. Explanation.

Find File (FF) moves the Write Pointer to the first memory word in the file specified. The file specified by the parameter must have been previously named by a NF....SN sequence and must have contents ("dummy vectors" or real data).

NOTE

If an FF command is executed for a non-existant file the 1351A never acknowledges tht the instruction was completed and the I/O will time out. See Chapter 4 for System Time Out Considerations.

The FF command may be used to place data in a file for the first time by positioning the Write Pointer so that the "dummy vectors" in a fixed-length file can be over-written by Plot Absolute (PA) and Text (TX) commands.

The FF command may also be used when the entire contents of an existing data file are to be changed.

Memory words remain sequential from 0 to 8191 regardless of the file to which they are assigned.

The FF command (and FL command) may thus be used both in the creation and updating of files. Updating files is one of the functions that falls under the File Manipulation category.

Manipulating Files

Previously created files (named and having contents) can be manipulated to enhance the pictures presented by the 1351A.

The commands used to manipulate 1351A files are: (a) BF, Blank File; (b) UF, Unblank File; (c) EF, Erase File; (d) FF, Find File; and (e) FL, Find Location.

All data contained in a file can be blanked (turned off) or unblanked (turned on) at one time. This selective display/no display feature allows the effect of motion to be simulated by using an animation technique. It also allows quick comparisons of different data to be presented for easier interpretation, or data of interest to be flashed on and off the display (Blank, wait, Unblank, wait loop).

Each file can have all its data contents erased and/or rewritten. Also, selective data within a file can be rewritten. This allows rapid updating of information to display new results.

Blank File Command

a. Syntax.

BF<file>[,]<terminator>

<file>=0 to 63

b. Explanation.

Blank File (BF) prevents any information contained in the file specified from being displayed. The file specified by the parameter must have already been named (NF....SN sequence) and have contents.

Blank File overrides all Pen Enable commands contained as status information for vectors and characters in the file specified. Data in the file specified is not changed.

In order to minimize the effect of a blanked file on the refresh rate all vectors in the blanked file will be plotted at their maximum rate or 6μs, whichever is less.

Unblank File Command

a. Syntax.

UF<file>[,]<terminator>

<file>=0 to 63

b. Explanation.

Unblank File (UF) allows vectors and/or characters contained in the file specified to be displayed according to the Pen Enable status (display beam on or off) of the vectors and/or characters.

The file specified by the parameter must have already been named (NF....SN sequence) and have contents.

After the Initialization sequence the 1351A is in the "unblank all files" configuration. When files are subsequently created they are unblanked until the 1351A is specifically told to blank a file by a BF command.

Erase File Command

a. Syntax.

EF<file>[,]<terminator>

<file>=0 to 63

b. Explanation.

Erase File (EF) causes all memory words in the file specified to be rewritten with "dummy vectors" (PE0,:PA0,0;:). The "dummy vectors" provide contents so that the file remains addressable. The number and location of words in the file remain unchanged.

EF moves the Write Pointer to memory word zero after the data in the file specified has been replaced by "dummy vectors." Therefore, the Write Pointer must be moved by either a Find File (FF) or a Find Location (FL) command before entering new vectors and/or characters. If not moved, the Write Pointer causes each new vector or character to be placed sequentially into memory starting at word zero. This will cause data in these locations to be over-written and thus lost.

The file specified by the parameter must have already been named (NF....SN sequence) and have contents.

Updating Files

Data in files may be changed by using the Find File (FF) and Find Loction (FL) commands.

If all data in a file is to be changed we recommend that the FF command be used. It is optional to use the EF command first, but provides good housekeeping to do so.

Example:

> Assume a fixed-length file of 500 words, with the first 300 words containing a picture and the last 200 words blank.
>
> If you want to redefine the file to a picture that only uses the first 200 words, and do not first erase the file, then the last 100 lines of the old picture will still be left in the file.

If memory word(s) other than the first word(s) in a file must be changed, use the FL command to move the Write Pointer to the specific word(s) to be changed. FL can be used to change a range of words within a file by moving the Write Pointer to the first word of the list to be changed, for example.

Data in a file is changed (after first positioning the Write Pointer) by sending the 1351A new vector endpoints (Plot Absolute command, PA) or new text characters (Text command, TX). Each vector endpoint (or character) should be preceeded by the appropriate Pen Enable (PE) command (display beam on or off). New text characters should also be preceeded by the appropriate Character Size (CS) command.

Chapter 4
Hardware

Introduction

This chapter will cover 1351A hardware as it pertains to the operator (programmer). A simple block diagram will be used to show basically how the 1351A works. Memory maps are then discussed to aid the operator understand the "whys" of programming considerations. The 1351A must refresh the display periodically so that the picture doesn't fade out. Refresh rates and their programming implications are given. Finally, 1351A Line Sync and Point Blanking switches are explained.

Block Diagram Description

Figure 4-1 is a simplified block diagram of the 1351A hardware.

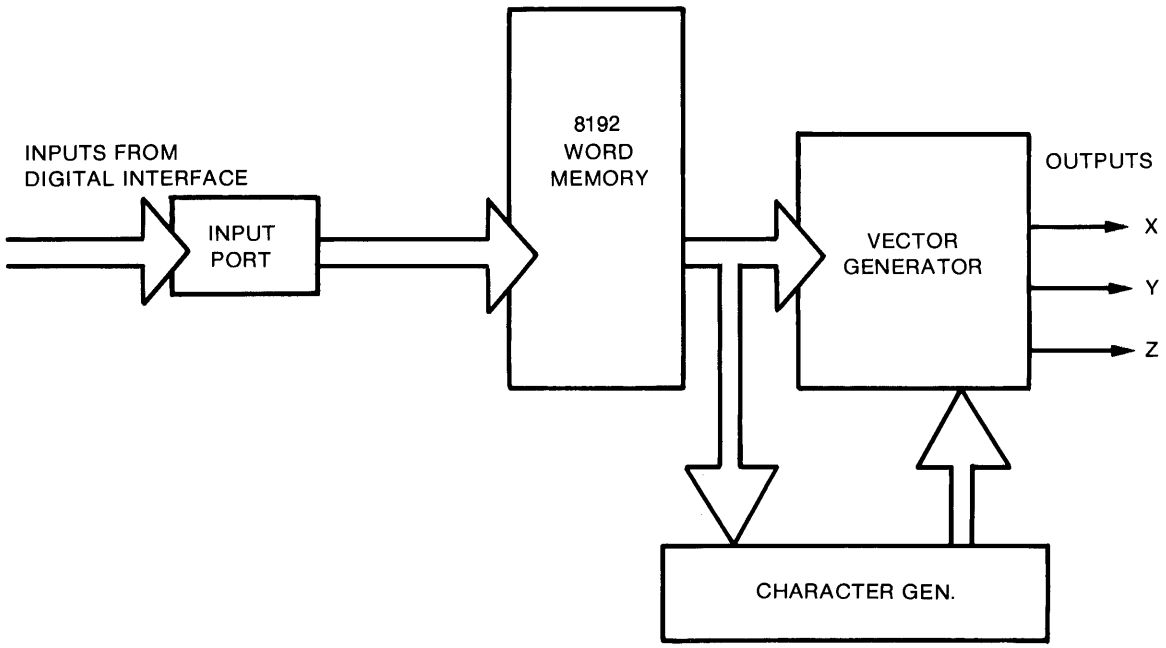


Figure 4-1. 1351A Simplified Block Diagram

Memory is the heart of the 1351A. Digital information from the memory produces the X, Y, and Z analog signals that "draw" pictures on the display.

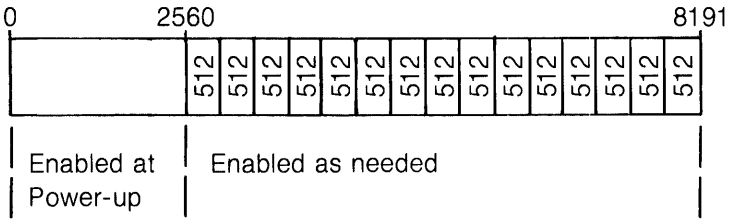
Information from the digital interface (programming commands) causes the desired picture-producing data to be written into the 1351A memory.

As each vector or character (plus status information) is written into a memory word, an internal device called the Write Pointer is automatically incremented one count. The Write Pointer value is the memory word number (0-8191) where the next vector or character that is received from the digital interface will be stored. The Write Pointer can be moved around by the programmer as discussed in Chapter 3.

The 1351A automatically outputs the enabled memory in repetitive cycles. As each word in turn is addressed, it outputs its contents. If the word contains vector endpoint coordinates (X,Y) then the data is sent directly from memory to the Vector Generator. If the word contains a text character then the Character Generator takes over the Vector Generator and produces all the vectors necessary to form the character.

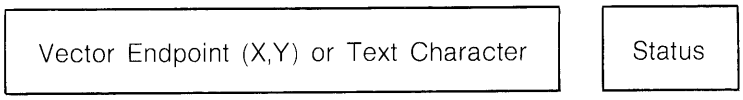
The continuous outputting of information is called "Reading" the memory. After the last enabled location has been "Read," location zero is addressed and the cycle begins again.

In order to minimize this cycle time the 1351A will have only the first 2560 (0 to 2559) locations enabled upon power-up. When the write pointer is directed to write into location 2560 an additional page of 512 locations will be enabled. This process will continue until the entire memory is enabled.



Memory Map

Each word of the memory is formatted as shown below.



The vector endpoint is made up of absolute X and Y values. It indicates the position where the beam will move next. The code for a text character causes the Character Generator to produce all vectors that form the character.

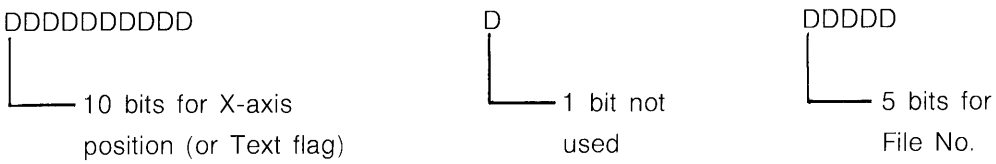
Status information that accompanies each of the words consists of:

1. Pen Enable (display beam on or off).
2. File name to which the word is assigned.
3. State of the file in which the word resides (blanked or unblanked).
4. Multiple display on/off combination.
5. If the word is a text character it is also accompanied by size and rotation information.

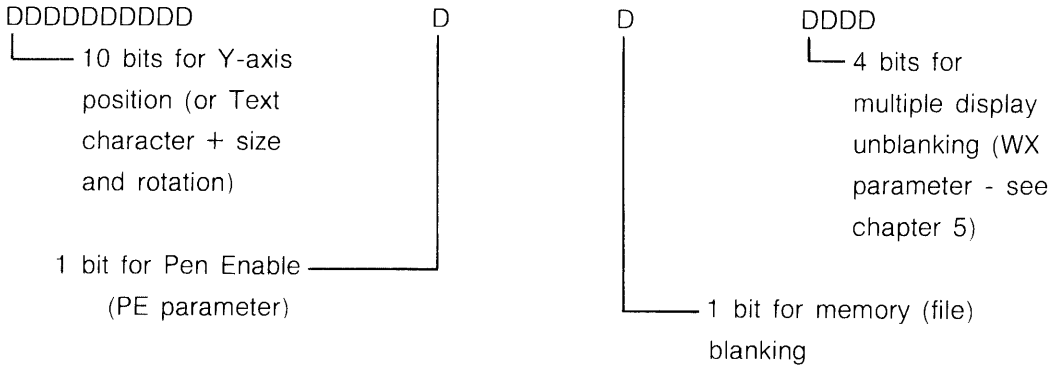
Memory Word Composition

Each word is 32 bits wide. The 32 bits are separated into two 16-bit bytes.

a. The first 16-bit byte contains:



b. The second 16-bit byte contains:



When a word is written into memory as a result of either a Plot Absolute (PA) or a Text (TX) command, it is automatically accompanied by the status information. All status commands should have been previously received by the 1351A in order to produce the desired results. See Chapter 3 of this manual for programming rules.

Examples.

1. Memory contents for a vector endpoint of 256,67 (PA256,67;:) that is in file 15 (NF15,:), which is to be visible (PE1,:), with memory and file unblanked (UM: of UF15,:), and only display "3" unblanked (WX11,:).

a.	First byte:	0100000000	0	01111	
b.	Second byte:	0001000011	1	1	1011

2. Memory contents for the text character "a" (TXa ETX:) that is in file 2 (NF2,:), which is to be visible (PE1,:), with memory and file unblanked (UM: or UF2,:), and all displays (1-4) unblanked (WX0,:). Also, the small "a" character is to be the smallest size available and rotated 90 degrees CCW (CS4,:).

NOTE

The Text flag is set internally by the 1351A when placed in the text mode by a TX command. The Text flag (a value of 1023) is placed in the X area of the first byte. The second byte contains size and rotation plus the ASCII code for the character in the Y area. The Text flag automatically preceeds each character until the Text mode is exited (ETX character) and the TX command then terminated by a colon (or CR or LF).

---

a.	First byte:	1111111111	0	00010	
b.	Second byte:	1001100001	1	1	0000

NOTE

Character size and rotation are placed in the three Most Significant Bits (MSBs) of the Y area. The ASCII code for the character is placed in the seven Least Significant Bits (LSBs). The "a" is 97 (decimal) which is 1100001 in binary. The Text flag (1023) is 1111111111 in binary.

Figure 4-2 below is a map of the first 2560-word page of memory sectioned into four files (1 thru 4) of 512 words per file. Each word of memory is designated by [Data + S] to indicate that it contains both Vector Endpoint (or Text Character) + Status Information.

MEMORY WORD (ADDRESS)	CONTENTS	FILE NO.
0	[Data + S]	
1	[Data + S]	
2	[Data + S]	
3	[Data + S]	
.	.	1
.	.	
509	[Data + S]	
510	[Data + S]	
511	[Data + S]	
512	[Data + S]	
513	[Data + S]	
514	[Data + S]	
.	.	2
.	.	
1021	[Data + S]	
1022	[Data + S]	
1023	[Data + S]	
1024	[Data + S]	
1025	[Data + S]	
1026	[Data + S]	
.	.	3
.	.	
1533	[Data + S]	
1534	[Data + S]	
1535	[Data + S]	
1536	[Data + S]	
1537	[Data + S]	
1538	[Data + S]	
.	.	4
.	.	
2045	[Data + S]	
2046	[Data + S]	
2047	[Data + S]	

Figure 4-2. 1351A Memory Map for Four Files (1 thru 4) of 512 Words Apiece

Status Register

Status commands are kept in a register so that they will be available whenever a vector endpoint (or text character) is written into memory.

Thus each status command establishes a condition (pen, file, character size, blanking, etc.) that remains in effect until changed by a subsequent status instruction.

Figure 4-3 is a programming diagram of the Status Register.

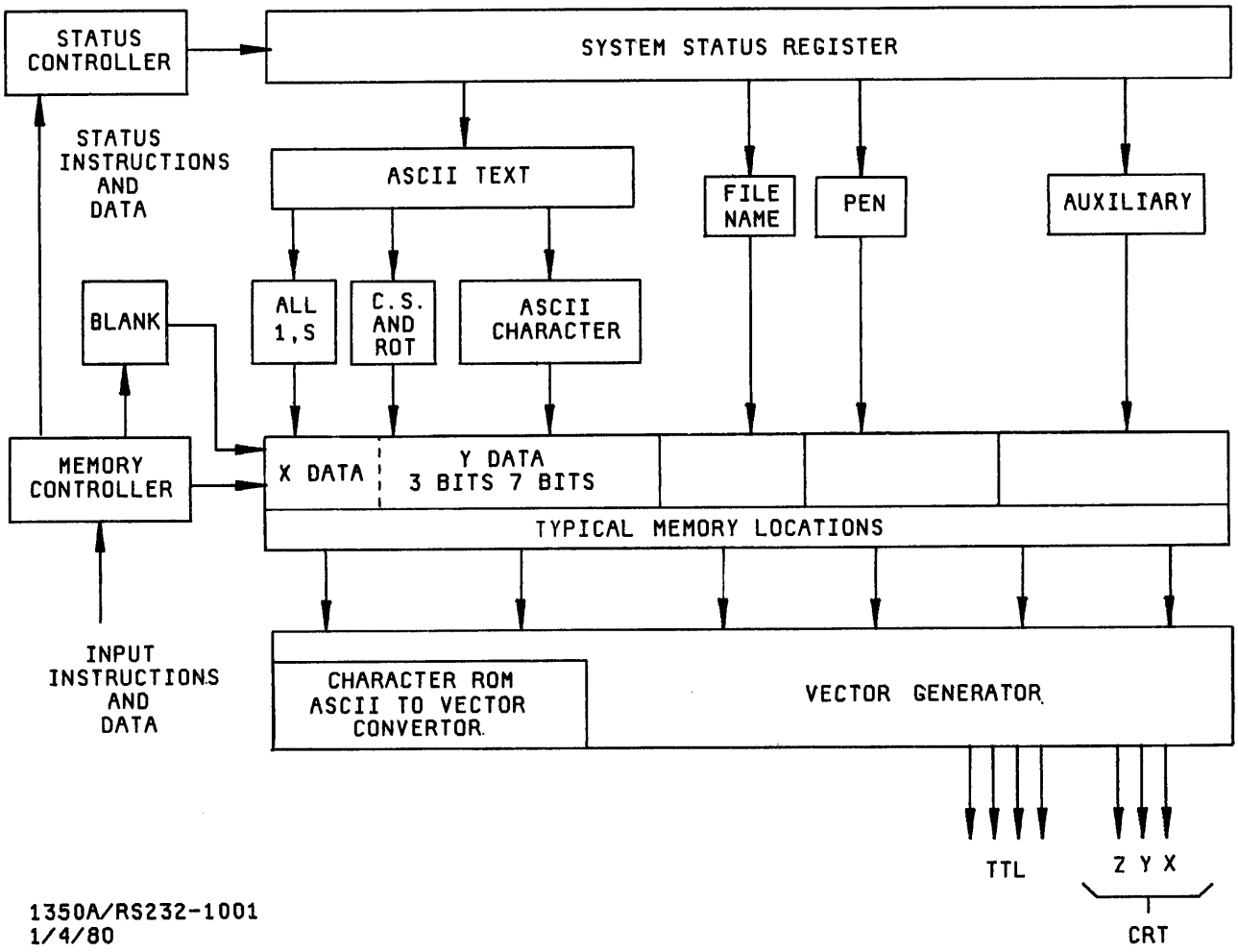


Figure 4-3. Programming Diagram of the Status Register

Refresh Rate

The 1351A produces a viewable picture by periodically "repainting" the picture on the display. This "repainting" is called refreshing the display. How often the 1351A refreshes the display is a function of how long it takes for the memory to output the data from all open memory locations. Each complete "Reading" of memory will "paint" the picture one time, and then the cycle repeats.

Each vector in the picture will require a time interval proportional to its length. The next memory word will not be addressed until the vector or character from the previous word has been completed. This causes the memory cycle time, and thus the refresh rate of the display, to be variable according to the picture being presented.

In order to minimize the time required to refresh the display, the 1351A plots blanked vectors at a much higher velocity than the unblanked vectors of equal length. Table 4-1 contains a comparison of these drawing times. All vectors in a file which is blanked will be plotted at their maximum blanked velocity as per Table 4-1. Also the 1351A will only cycle through enabled memory and not the memory which is not being used. Upon power-up, the 1351A will have the first 2560 memory locations enabled. Additional pages of 512 locations each will be enabled as the WRITE pointer is directed across the boundary of enabled memory into the first location of the next page of unenabled memory.

The absolute minimum time required for the 1351A to cycle through the first 2560-word page of memory ("Read" all locations) is approximately 3.85 milliseconds (ms). The recriprocal of 3.85 ms is 260 Hz (Hertz = cycles per second). This means that at its fastest, the 1351A can "repaint" (refresh) the display 260 times every second.

The memory cycle time can increase to the point where the light output of the display phosphor will expire before the picture gets repainted. This produces visable "flickering" of the picture. The use of many long vectors in a picture will slow the 1351A down. The slower the 1351A, the more the display "flickers." A typical phosphor display screen starts to flicker when refreshed at the rate of 30 to 50 times per second (Hertz).

Refresh Time Calculation

Typical drawing times for characters and vectors are listed in Table 4-1. To determine 1351A refresh time, total the drawing times for all characters and vectors to be presented along with the opened and unused memory locations. Each vector endpoint or chracter uses one memory location. Since each vector then takes 2 memory locations (start and end point), the following equation can be used to calculate the number of unused locations:

$$[2560 + N(512)] - [\text{number of non-continuous vectors} \times 2 + \text{number of characters}] = \text{unused locations to be refreshed}$$

where N = number of 512 location pages opened

Table 4-1. Typical Drawing Times for Vectors and Characters (variable speed off)

Average character	15 μs (microseconds)
Vectors:	
Blanked, 1/1020 to 1/32 screen diameter	1.5 μs
Blanked, 1/32 to 1/16 screen diameter	3.0 μs
Blanked, ≥16 screen diameter	6.0 μs
1/2 or greater screen diameter	48 μs
1/4 to 1/2 screen diameter	24 μs
1/8 to 1/4 screen diameter	12 μs
1/16 to 1/8 screen diameter	6 μs
1/32 to 1/16 screen diameter	3 μs
less than 1/32 screen diameter	1.5 μs

Example:

A CRT presentation contains: 50 characters; a graticule of 20 vertical and 20 horizontal lines; 30 tick marks on graticule lines; and 200 data points joined by short vectors.

Refresh time calculation:

50 characters × 15 μs (microseconds) = 750 μs
40 graticule vectors × 48 μs = 1,920 μs
(30 + 200) short vectors × 1.5 μs = 345 μs
1970 unused locations × 1.5 μs = 2,955 μs
Total = 5,970 μs

5,970 μs = 5.970 ms

NOTE

The above calculation is with variable speed off, and 2560 open memory locations.

Reducing Display Flicker

In some applications the 1351A can be made to refresh the display more often in order to reduce the flickering by presorting and grouping the data for minimum beam movement when drawing the picture. This reduces the time spent "traveling back and forth" and optimizes the display.

Don't forget that an invisible vector requires some drawing time, and thus has an effect on refresh rate. Therefore, you should try to minimize travel distance between visible vectors in order to decrease "flickering".

Line Sync

For systems having severe ground loop problems (or environments with strong magnetic fields) the 1351A may be synchronized to the frequency of the ac power line. This is done with the rear-panel LINE SYNC switch. Switch position nearest the fan is OFF.

Line sync should be used only when absolutely necessary. When line sync is used the display refresh rate may be reduced to the point where flicker becomes objectionable.

Configuring the 1351A to Listen Properly on the Digital Interface

Refer to the appropriate service documentation. For example, the main 1351A Operating and *Service* manual contains information on how to change the address when the 1351A is equipped with the HP-IB (Hewlett Packard Interface Bus) digital interface. When the 1351A is equipped with the RS-232C digital interface (Option 001) then the information is contained in the service document for Option 001.

For 16-bit parallel systems (option 002) the option 002 Operating and Service Manual contains both programming and service information.

When HP-IB is used, be careful not to exceed the cable length as specified in the standard (IEEE Std. 488-1978). Maximum HP-IB cable length is 2 meters times the number of devices on the bus, or 20 meters, whichever is LESS.

System Time Out Considerations

Many systems provide the capability of setting a limit to the time aliowed for an instruction to be completed. This allows the system to recover from errors or problems that cause the I/O bus to "hang up."

Since some 1351A commands take longer than others, we recommend that the minimum time out be 100 milliseconds. In practice, one or two seconds is not inappropriate.

Example:

If the 1351A is told to find a non-existant file (FF command) then it cannot respond that the command was completed. The I/O bus will "hang up." This will cause a time out to occur.

NOTE

A simple output buffering scheme may not check for time outs. If the I/O bus "hangs up," make sure that all files addressed by FF (Find File) commands do exist. A file only exists when it has been named and has contents.

Factory Switch Settings

The 1351A is shipped from the factory with the internal switches set as per table 4-2.

Have only qualified service personnel change these switch settings as the covers of the 1351A must be removed for this operation.

The 1351A Operating and Service Manual contains instructions for the setting of these switches.

Table 4-2. 1351A Factory Switch Settings

HP-IB Address — 18

I/O File/Aux Switch — S3 — FILE

Variable Speed Switch — S1 — ON

1	2	3	4	5
ON	OFF	OFF	OFF	N/A

Hardware Blinking Switch S2

Chapter 5
Extras

Introduction

This chapter explains the programming considerations necessary when using the ancillary functions of the 1351A. These functions are: (a) simultaneous presentation of separate pictures on up to four different displays; (b) picture blinking through hardware; and (c) programmable intensity.

Multiple Picture Displays

This section describes how to use the Write Auxiliary (WX) and Stop Auxiliary (SX) commands to display several different pictures at the same time.

In order to display four different pictures at once four X-Y displays are required.

To produce four separate pictures, the 1351A must:

1. Turn one display on and the other three off.
2. Output vector and/or character (picture) information to the active display.
3. Turn the next display on and the other three off.
4. Output picture information to the active display.
5. Repeat steps c and d for the next two displays.

This cycle of "turn a display on and send out a picture, turn the next display on and send out another picture" provides the effect of different pictures being presented simultaneously (see figure 5-1).

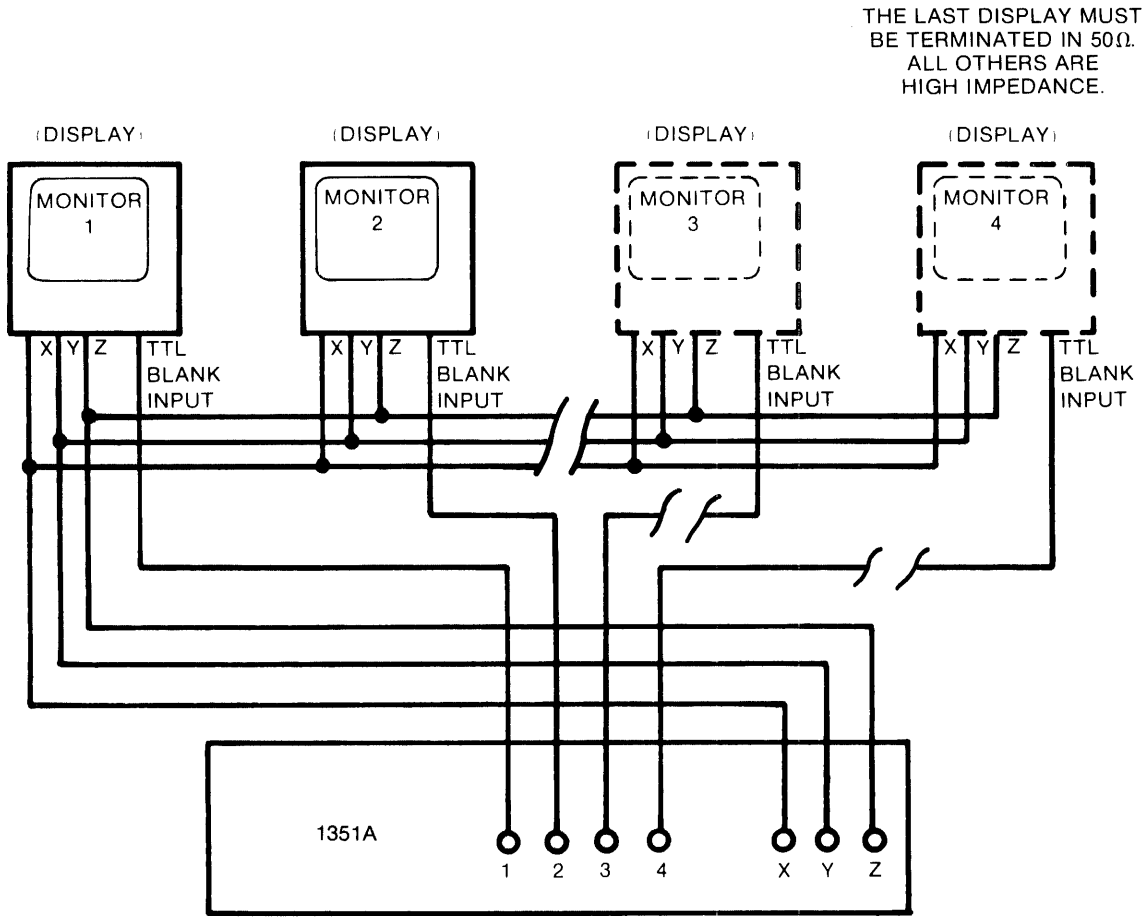


Figure 5-1. 1351A/Display Interconnections

Other combinations are possible, such as having the same picture on displays "1" and "2" while displays "3" and "4" each have their own pictures. First turn displays "1" and "2" on ("3" and "4" off) and send them a picture. Then turn on display "3" only and send its picture. Repeat for display "4".

With four displays there are 16 possible combinations available for determining which displays are on (and which are off) when the 1351A is sending out a picture (see table 5-1).

Multiple displays are turned on and off via the 1351A rear-panel "DISPLAY 1 thru 4" outputs. These outputs must be connected to the "TTL BLANKING" inputs (positive-going) on the displays. Refer to Section II in the Model 1351A Operating and Service manual and consult qualified service personnel when setting up a multiple display system.

Table 5-1. WX Display Blank/Unblank Combinations

WX PARAMETER <DISPLAY>	DISPLAY			
	1	2	3	4
0	U	U	U	U
1	B	U	U	U
2	U	B	U	U
3	B	B	U	U
4	U	U	B	U
5	B	U	B	U
6	U	B	B	U
7	B	B	B	U
8	U	U	U	B
9	B	U	U	B
10	U	B	U	B
11	B	B	U	B
12	U	U	B	B
13	B	U	B	B
14	U	B	B	B
15	B	B	B	B

U = Unblank (display on). B = Blank (display off).

The Write Auxiliary (WX) command determines the specific combination of which displays are on and which are off when the 1351A is sending out a picture. Each combination must be bracketed by a WX and an SX (Stop Auxiliary) set of commands.

Write Auxiliary Command

a. Syntax.

WX <display>[,]<terminator>

<display>=0 to 15

b. Explanation.

Write Auxiliary (WX) sets the combination of which displays are turned on (unblanked) and which are turned off (blanked) when the 1351A is outputting vector and/or character information.

Each specific display blank/unblank combination is set by the WX parameter. Table 5-1 is a listing of the blank/unblank combinations for all WX parameters.

The picture that is sent to a specific blank/unblank combination should be followed by a Stop Auxiliary (SX) command.

Stop Auxiliary Command

a. Syntax. SX <terminator>

b. Explanation.

Stop Auxiliary (SX) ends the assignment of vectors and/or characters (picture information) to a WX blank/unblank combination.

SX performs the same function for WX that SN (Stop Naming) does for NF (Name File). This WX....SX bracketing of pictures provides the proper control so that pictures are shown only on the desired display(s).

The recommended command sequence is: (a) WX, Write Auxiliary; (b) NF, Name File (or other file commands such as UF); (c) Picture Information (PE, CS, PA, TX); (d) SN, Stop Naming; and (e) SX, Stop Auxiliary.

Updating Displayed Information

We recommend that each picture be placed in a file (see Chapter 3).

When information shown on the display(s) needs to be changed, the Find File (FF) command may then be used and new vector/character commands entered into the picture file.

Changes in the picture content do not affect the WX blank/unblank combination.

Multiple Picture Example.

Let's assume that the picture for display "1" is in file one, the picture for display "2" is in file two, the picture for display "3" is in file nine, and the picture for display "4" is in file twenty-seven.

In order to produce the four pictures on separate displays the 1351A should receive the following sequence of ASCII commands.

a. "BF1,:BF2,:BF9,:BF27,:"

   This blanks the four files being used for the pictures.

b. "WX14,:UF1,:SX:"

   This turns display "1" on and sends it the picture in file one. The other three displays are blanked (turned off).

c. "WX13,:UF2,:SX:"

   This turns on display "2" only and sends it the picture in file two.

d. "WX11,:UF9,:SX:"

   This turns on display "3" only and sends it the picture in file nine.

e. "WX7,;UF27,;SX;"

   This turns on display "4" only and sends it the picture in file twenty-seven.

Hardware Picture Blinking

The 1351A can cause pictures to flash on and off the display(s) via internal switches (hardware). This will eliminate the need for software "blank file/wait/unblank file/wait" type of loops.

Hardware blinking will cause the picture to flash on and off the display approximately four times per second.

NOTE

When 1351A hardware blinking is used, display "4" will always be blanked. Therefore, blinking is only possible on displays "1", "2", and "3".

Blinking is actually determined by both hardware and software. The hardware portion is configured by switches inside the 1351A. The software portion relies on the parameter used with a Write Auxiliary (WX) command.

Blink Switches

In order to have displays "1", "2", or "3" blink, the corresponding Blinking Switch(es) must be turned on.

These switches are inside the 1351A cabinet and should be changed only by qualified service personnel. Refer service personnel to the Model 1351A Operating and Service manual, Section III.

The 1351A is shipped from the factory with all blink switches in the off position except for display 1.

Write Auxiliary (WX) Effect on Blinking

Blinking is only possible when WX parameters 8 thru 15 are used. Blinking is not possible with WX parameters 0 thru 7.

Table 5-2 is a listing of the blank/unblank-blink possibilities when using WX 8 thru 15. For a complete description of WX, see the Multiple Picture Displays section immediately preceeding this section.

Table 5-2. Display Blank/Unblank-Blink Combinations for WX 8-15

WX PARAMETER	DISPLAY			
	1	2	3	4
8	U*	U*	U*	B
9	B	U*	U*	B
10	U*	B	U*	B
11	B	B	U*	B
12	U*	U*	B	B
13	B	U*	B	B
14	U*	B	B	B
15	B	B	B	B

B = Blank (display off).

U* = Unblank (display on) and blink if switch is on.

NOTE

If the intensity switch S-3 on the I/O Board is set to "X" (Auxiliary) and 3 or 4 displays are used, the WX parameters will provide blank/unblank control. However, a change in brightness of displayed data will occur if the vraiable speed switch (S-1 on Display Board) is set "ON".

When using blinking, remember:

1. Blink switch(es) for the desired display(s) must be turned on.

2. Display "4" will always be blanked.

3. Display(s) to show a blinking picture must be unblanked (turned on) by a WX parameter.

Programmable Intensity

When used in conjunction with a Hewlett-Packard Graphics Display which has a binary Z-axis input, the 1351A can command eight brightness levels. These brightness levels can be used for differentiating information in the display. For example, new data might be very bright and older data dimmer with the graticule lines being very dim. The weight of lines in a drafting application might be indicated by brightness at which it is drawn.

This feature is restricted to single display applications because the TTL blanking signals are the signals that make up the intensity control code.

The Auxiliary Connector on the 1351A I/O is the source of this control code for the display. An interface cable, Hewlett-Packard part number 8120-2739 is available to make this interface quickly.

The intensity control is accomplished with the WX instruction. Blinking can be obtained if the hardware blinking switch S3 is set to the display 1 "ON" position.

Table 5-3 is a listing of the intensity level and blinking possibilities when using this mode.

Table 5-3. WX Programmable Intensity

WX INST	INTENSITY LEVEL	HDW BLINK
0	Brightest	No
1	2nd Brightest	No
2	3rd Brightest	No
3	4th Brightest	No
4	5th Brightest	No
5	6th Brightest	No
6	Dimmest	No
7	Blanked	No
8	Brightest	Yes
9	2nd Brightest	Yes
10	3rd Brightest	Yes
11	4th Brightest	Yes
12	5th Brightest	Yes
13	6th Brightest	Yes
14	Dimmest	Yes
15	Blanked	Yes

Chapter 6
Example Programs

Introduction

This chapter contains four different example programs for the 1351A. Each program is presented in three languages: (1) HPL, which is used by the Hewlett-Packard Model 9825A Calculator; (2) BASIC; and (3) FORTRAN.

The four examples are:

1. Initialization and drawing a simple picture.
2. Drawing a geometric design by using variables.
3. Printing all available sizes and rotations of text.
4. File usage.

Each example program is preceeded by a discussion of the intent of the program and the general procedure that will be used to produce the desired results. The actual programs have been commented to show how the general procedure is implemented in each language.

If you are using a different language, the comments and general procedures provide a base of information that aids the translation process. If problems are encountered, contact your nearest Hewlett-Packard office.

The BASIC and FORTRAN examples were done on a Hewlett-Packard System 1000 Computer. The HPL examples were done on an HP 9825A Calculator.

All program examples are for a 1351A equipped with the HP-IB interface. HP-IB (Hewlett-Packard Interface Bus) is Hewlett-Packard's implementation of IEEE Standard 488-1978, "Standard Digital Interface for Programmable Instrumentation."

For the HPL programs, the 9825A Calculator is an Option 002 (23K byte Read/Write memory) equipped with: 98034A HP-IB Interface; 98210A String-Advanced Programming ROM; 98213A General I/O - Extended I/O ROM.

For the BASIC programs, we used HP BASIC/1000D: HP92101-16001 REV. 1913 (781108).

NOTE

In BASIC, the ETX character (End of Text) is obtained by pressing the Control key (CNTL) and the capital C key at the same time. The DC4 character is obtained by pressing CNTL and capital T at the same time.

For the FORTRAN programs, we used FTN4 COMPILER: HP92060-16092 REV. 1913 (790206).

The System 1000 used for both BASIC and FORTRAN examples is equipped with RTE IV, Driver 37, and a 59310B I/O Card.

Each of the four examples is accompanied by a photo. The photo shows what the display should be, no matter which language is used in the example.

Example 1: Initialization and Drawing a Simple Picture

Intent.

This example shows how to Initialize the 1351A, draw a triangle, and print a title below the triangle. See figure 6-1.

General Procedure.

1. Initialize the 1351A with the following ASCII sequence.

   ETX End of Text character (decimal 3) removes the 1351A from possible Text mode.

   DC4 DC4 character (decimal 20) turns off 1351A POWER INTERRUPT light.

   EM The EM command clears memory of vectors, text, and pen values.

   EN EN command removes all file assignments from memory.

EX EX command removes all multiple display blanking assignments from memory.

SN SN command stops any possible file naming.

SX SX command stops any possible multiple display blanking assignment.

UM UM command places entire memory in an unblanked state.

Each of the commands must be followed by a valid <terminator>. The ASCII string is shown below using colon (:) terminators.

ETX DC4 :EM:EN:EX:SN:SX:UM:

2. Draw the triangle.

   a. Turn the beam off (lift the beam) with a PE0,: command.

   b. Move the beam to the starting point with a PAX1,Y1;: command. X1 = first X value; Y1 = first Y value.

   c. Turn the beam on (lower the beam) with a PE1,: command.

   d. Move the beam in the triangle pattern with a PAX2,Y2;X3,Y3;X1,Y1;: command.

   The ASCII string is:

   PE0,:PAX1,Y1;:PE1,:PAX2,Y2;X3,Y3;X1,Y1;:

3. Print the title.

   a. Turn the beam off with a PE0,: command.

   b. Move the beam to the starting point for the title with a PAX4,Y4;: command.

   c. Set the character size to "2" with a CS2,: command.

   d. Write the title with a TX1351A command.

e. Exit Text mode with an ETX character.

f. Terminate the Text command with a colon (:).

The ASCII string is:

PE0,:PAX4,Y4;:CS2,:TX1351A ETX:

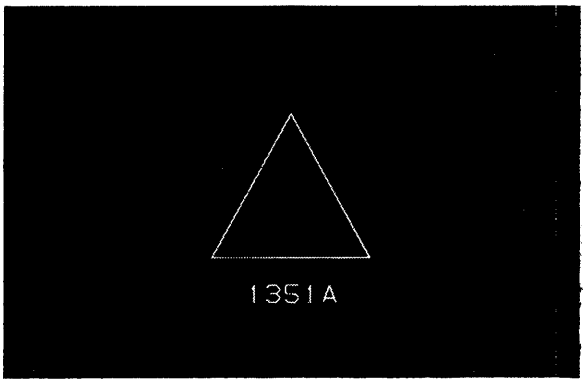


Figure 6-1. Example 1 Display

```
0: "                                                                    ":
1: "   - - - - - - - - - - - -   EXAMPLE 1   - - - - - - - - - - - -    ":
2: "                              HPL                                   ":
3: "                                                                    ":
4: "                INITIALIZATION AND SIMPLE PLOTTING                  ":
5: "                                                                    ":
6: " initialize the 1351A by putting out an ASCII 3 to terminate any text,":
7: "   an ASCII 20 to clear the power interrupt light":
8: "   a SN to stop the naming of files":
9: "   a SX to stop any auxiliary action":
10: "   a EM to erase all vectors, characters, and beam controls":
11: "   a EN to erase all files":
12: "   a EX to erase all auxiliary actions":
13: "   and a UM to unblank memory":
14: "   All these commands are ended with the : terminator":
15: "   note: all output is directed to 718, 7 is the I/O channel (HPIB)":
16: "         and 18 is the select code (1351A I/O card must be set to 18)":
17: wtb 718,3,20,":sn:sx:em:en:ex:um:"
18: " draw a triangle by lifting the beam, moving it to the starting ":
19: " position, lowering the beam, and drawing the triangle":
20: wrt 718,"pe0,:pa300,300;:pe1,:pa500,700;700,300;300,300;:"
21: " write the title by lifting the beam, moving it to the starting":
22: " position, lowering the beam, setting the character size, and":
23: " writing the title":
24: "   note: the 3 is an ETX which is used to exit TEXT mode":
25: wtb 718,"pe0,:pa390,175;:cs2,:tx1351A",3,":"
26: stp
*20855
```

```
.
10  REM
20  REM  - - - - - - - - - -   EXAMPLE 1   - - - - - - - - - -
30  REM                          BASIC
40  REM
50  REM          INITIALIZATION AND SIMPLE PLOTTING
60  REM
70  REM   SET THE ASCII ETX CODE (CONTROL C)
80  LET E$=""
90  REM   SET THE ASCII DC4 CODE (CONTROL T)
100  LET D$=""
110  REM   INITIALIZE THE 1351A BY PUTTING OUT AN ASCII 3 TO EXIT TEXT
120  REM     MODE, AN ASCII 20 TO CLEAR THE POWER INTERRUPT LIGHT
130  REM     A SN TO STOP NAMING OF FILES
140  REM     A SX TO STOP ANY AUXILIARY ACTION
150  REM     A EM TO ERASE ALL VECTORS, CHARACTERS, AND BEAM CONTROLS
160  REM     A EN TO ERASE ALL FILES
170  REM     A EX TO ERASE ALL AUXILIARY ACTIONS
180  REM     AND A UM TO UNBLANK MEMORY
190  REM     ALL THESE COMMANDS ARE ENDED WITH THE ; TERMINATOR
200  PRINT #18;E$,D$,":SN;SX;EM;EN;EX;UM;"
210  REM   DRAW A TRIANGLE BY LIFTING THE BEAM, MOVING IT TO THE STARTING
220  REM     POSITION, LOWERING THE BEAM, AND DRAWING THE TRIANGLE
230  PRINT #18;"PE0,;PA300,300;;PE1,;PA500,700;700,300;300,300;;"
240  REM   WRITE THE TITLE BY LIFTING THE BEAM, MOVING IT TO THE STARTING
250  REM     POSITION, LOWERING THE BEAM, SETTING THE CHARACTER SIZE, AND
260  REM     WRITING THE TITLE
270  PRINT #18;"PE0,;PA390,175;;CS2,;TX1351A",E$,";"
280  STOP
290  END
```

```
0001  FTN,L
0002  C
0003  C - - - - - - - - - - - - - -   EXAMPLE 1   - - - - - - - - - - - - - -
0004  C                               FORTRAN
0005  C
0006  C               INITIALIZATION AND SIMPLE PLOTTING
0007  C
0008        PROGRAM GMFE1
0009  C.....  SET THE LU OF THE 1351A
0010        LU1350=18
0011  C.....  SET THE ASCII ETX CODE IN THE UPPER 8 BITS
0012        IETX=3*256
0013  C.....  SET THE ASCII DC4 CODE IN THE UPPER 8 BITS
0014        IDC4=20*256
0015  C.....  INITIALIZE THE 1351A BY PUTTING OUT AN ASCII 3 TO EXIT TEXT MODE
0016  C.....    AN ASCII 20 TO CLEAR THE POWER INTERRUPT LIGHT
0017  C.....    A SN TO STOP THE NAMING OF FILES
0018  C.....    A SX TO STOP ANY AUXILIARY ACTION
0019  C.....    A EM TO ERASE ALL VECTORS, CHARACTERS, AND BEAM CONTROLS
0020  C.....    A EN TO ERASE ALL FILES
0021  C.....    A EX TO ERASE ALL AUXILIARY ACTION
0022  C.....    AND A UM TO UNBLANK MEMORY
0023  C.....    ALL THESE COMMANDS ARE TERMINATED WITH THE : TERMINATOR
0024        WRITE(LU1351,1) IETX,IDC4
0025      1 FORMAT(A1,A1,":SN:SX:EM:EN:EX:UM:")
0026  C.....  DRAW A TRIANGLE BY LIFTING THE BEAM, MOVING IT TO THE STARTING
0027  C.....    POSITION, LOWERING THE BEAM, AND DRAWING THE TRIANGLE
0028        WRITE(LU1351,2)
0029      2 FORMAT("PE0,:PA300,300;:PE1,:PA500,700;700,300;300,300;:")
0030  C.....  WRITE THE TITLE BY LIFTING THE BEAM, MOVING IT TO THE STARTING
0031  C.....    POSITION, LOWERING THE BEAM, SETTING THE CHARACTER SIZE, AND
0032  C.....    WRITING THE TITLE
0033        WRITE(LU1351,3) IETX
0034      3 FORMAT("PE0,:PA390,175;:CS2,:TX1351A",A1,":")
0035        STOP
0036        END
```

Example 2: Drawing a Geometric Design by Using Variables

Intent.

This example shows how to draw an involved pattern (figure 6-2) that requires the use of formatted outputs in order to plot variables.

General Procedure.

1. Allocate two arrays.

2. Initialize the control variables.

    a. Set the number of points to a prime number less than or equal to 31 (P=31).

    b. Set the radius to 500.

    c. Set THETA to (2 X PI)/P (i.e., radians). PI is 3.14159...

    d. Set the center to 511,511.

3. Compute the points on the outside of the circle.

4. Initialize the 1351A.

5. Move the beam to the starting point X[P],Y[P] by using the following format controls.

    a. Turn the beam off with a PE0,: command.

    b. Move the beam with a PA command. Do the following things to generate the proper command:

        * Convert X[P] to a 4 place integer (no decimal point).
        * Insert a comma.
        * Convert Y[P] to a 4 place integer (no decimal point).
        * Insert a semicolon (;) and a colon (:).

c. Turn the beam on with a PE1,: command.

The format to do step 5 is:

"PE0,:PA" conv-4-Int "," conv-4-Int ";:PE1,:"

where conv-4-Int means to convert to a 4 place integer and all the characters within the quote marks are required ASCII entries. Note: Conv-4-Int is f4.0 in HPL, 4D in BASIC, and I4 in FORTRAN.

6. Plot the design again using a format similiar to step 5, except it only uses the following items.

"PA" conv-4-Int "," conv-4-Int ";:"

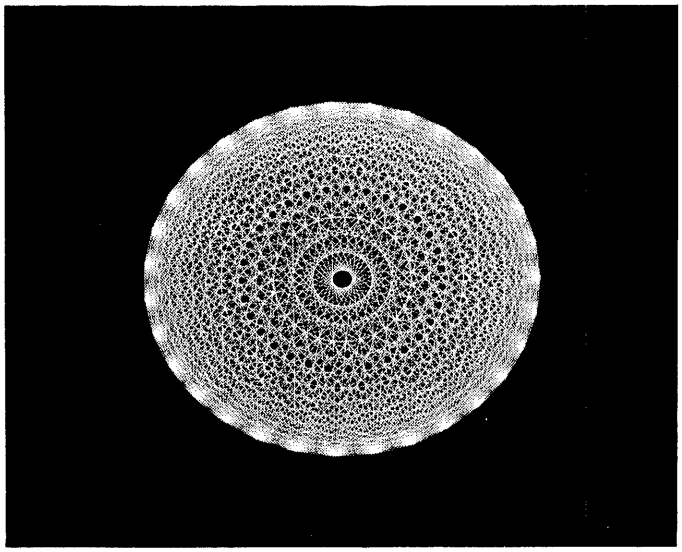


Figure 6-2. Example 2 Display

```
0: "                                                                    ":
1: "   - - - - - - - - - - - -   EXAMPLE 2   - - - - - - - - - - - -    ":
2: "                               HPL                                  ":
3: "                                                                    ":
4: "   DRAWS A GEOMETRIC DESIGN  (PLOTTING WITH FORMATTED VARIABLES)    ":
5: "                                                                    ":
6: dim X[31],Y[31]
7: " set up the plotting formats":
8: fmt 1,"pe0,:pa",f4.0,",",f4.0,";:pel,:"
9: fmt 2,"pa",f4.0,",",f4.0,";:"
10: " set the number of points - must be a prime number":31→P
11: " set the radius":500→R
12: " set theta in radians":rad;2*π/P→T
13: " set the center of the design":511→A→B
14: " compute the outside points of the design":
15: for I=1 to P;T*I→Z;R*cos(Z)+A→X[I];R*sin(Z)+B→Y[I];next I
16: " initialize the 1351A":
17: wtb 718,3,20,":sn:sx:em:en:ex:um:"
18: " move the beam to the starting position":
19: int(P/2)→N;wrt 718.1,X[P],Y[P]
20: " draw the geometric design":
21: for K=1 to N;K→L
22: for C=1 to P
23: wrt 718.2,X[L],Y[L];L+K→L
24: if L>P;L-P→L
25: next C
26: next K
27: stp
*8718
```

```
10  REM
20  REM  - - - - - - - - - - - - -   EXAMPLE 2   - - - - - - - - - - - - -
30  REM                               BASIC
40  REM
50  REM   DRAW A GEOMETRIC DESIGN  (PLOTTING WITH FORMATTED VARIABLES)
60  REM
70  DIM X[31],Y[31]
80  REM   SET THE ASCII ETX CODE (CONTROL C)
90  LET E$=""
100  REM   SET THE ASCII DC4 CODE (CONTROL T)
110  LET D$=""
120  REM   SET THE NUMBER OF POINTS, MUST BE A PRIME LESS THAN 32
130  LET P=31
140  REM   SET THE RADIUS
150  LET R=500
160  REM   SET THETA IN RADIANS
170  LET T=2*3.14159/P
180  REM   SET THE CENTER OF THE SCREEN
190  LET C=511
200  REM   COMPUTE THE OUTSIDE POINTS
210    FOR I=1 TO P
220    LET S=T*I
230    LET X[I]=R*COS(S)+C
240    LET Y[I]=R*SIN(S)+C
250    NEXT I
260  REM   INITIALIZE THE 1351A
270  PRINT #18;E$,D$,":SN:SX:EM:EN:EX:UM:"
280  REM   MOVE THE BEAM TO THE STARTING POSITION
290  PRINT #18 USING 300;X[P],Y[P]
300  IMAGE "PE0,:PA",4D,",",4D,";:PE1,:"
310  REM   DRAW THE GEOMETRIC DESIGN
320    FOR I=1 TO P/2
330    LET L=I
340      FOR J=1 TO P
350      PRINT #18 USING 360;X[L],Y[L]
360      IMAGE "PA",4D,",",4D,";:"
370      LET L=L+I
380      IF L>P LET L=L-P
390      NEXT J
400    NEXT I
410  STOP
420  END
```

```
0001  FTN,L
0002  C
0003  C - - - - - - - - - - - - - -   EXAMPLE 2   - - - - - - - - - - - - -
0004  C                               FORTRAN
0005  C
0006  C  DRAW A GEOMETRIC DESIGN  (PLOTTING WITH FORMATTED VARIABLES)
0007  C
0008        PROGRAM GMFE2
0009        DIMENSION IX(31),IY(31)
0010  C.....  SET THE LU OF THE 1351A
0011        LU1350=18
0012  C.....  SET THE ASCII ETX CODE IN THE UPPER 8 BITS
0013        IETX=3*256
0014  C.....  SET THE ASCII DC4 CODE IN THE UPPER 8 BITS
0015        IDC4=20*256
0016  C.....  SET THE NUMBER OF POINTS, MUST BE A PRIME LESS THAN 32
0017        IPTS=31
0018  C.....  SET THE RADIUS
0019        RAD=500.
0020  C.....  SET THETA IN RADIANS
0021        THETA=2*3.141592654/FLOAT(IPTS)
0022  C.....  SET THE CENTER OF THE DESIGN
0023        IXC=511
0024        IYC=511
0025  C.....  COMPUTE THE OUTSIDE POINTS
0026        DO 1 I=1,IPTS
0027          S=THETA*FLOAT(I)
0028          IX(I)=IFIX(RAD*COS(S))+IXC
0029          IY(I)=IFIX(RAD*SIN(S))+IYC
0030      1 CONTINUE
0031  C.....  INITIALIZE THE 1351A
0032        WRITE(LU1351,2) IETX,IDC4
0033      2 FORMAT(A1,A1,":SN:SX:EM:EN:EX:UM:")
0034  C.....  MOVE THE BEAM TO THE STARTING POSITION
0035        WRITE(LU1351,3) IX(IPTS),IY(IPTS)
0036      3 FORMAT("PE0,:PA",I4,",",I4,";:PE1,:")
0037  C.....  DRAW THE GEOMETRIC DESIGN
0038        DO 5 I=1,IPTS/2
0039          L=I
0040          DO 5 J=1,IPTS
0041            WRITE(LU1351,4) IX(L),IY(L)
0042      4     FORMAT("PA",I4,",",I4,";:")
0043            L=L+I
0044            IF(L.GT.IPTS)  L=L-IPTS
0045      5 CONTINUE
0046        STOP
0047        END
```

Example 3: Printing All Available Sizes and Rotations of Text

Intent.

This example shows how to generate text on the display (see figure 6-3).

General Procedure.

1. Initialize the 1351A.
2. Write text.
    a. Set the size and rotation for the characters with a CS<size>,: command.
    b. Turn the beam off with a PE0,: command.
    c. Move the beam to the starting point with a PAX,Y;: command.
    d. Print the message with a TX<string> command.
    e. Exit Text mode with an ETX (End of Text) character.
    f. Terminate TX command with a colon (:).

The ASCII string to do this is:

CS0,:PE0,:PAX,Y;:TXmessage ETX:

where CS0,: is used as an example of the CS<size>,: command.

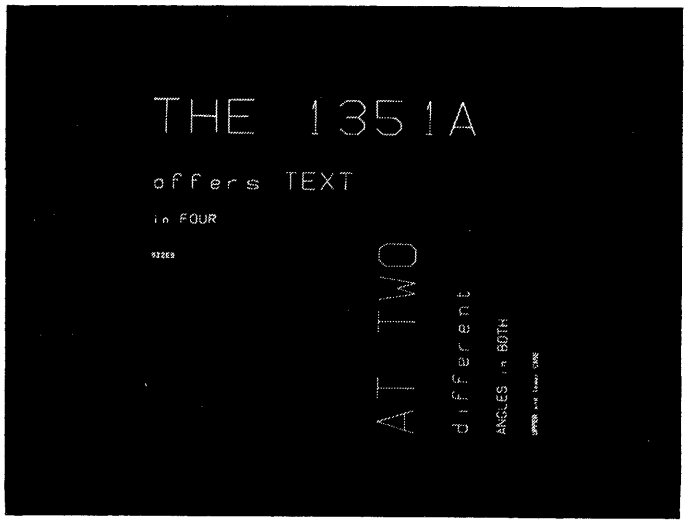


Figure 6-3. Example 3 Display

```
0: "                                                                    ":
1: "   - - - - - - - - - - - - -   EXAMPLE 3   - - - - - - - - - - - -    ":
2: "                                 HPL                                ":
3: "                                                                    ":
4: "                       SHOWS USES OF TEXT OUTPUT                    ":
5: "                                                                    ":
6: " initialize the 1351A":
7: wtb 718,3,20,":sn:sx:em:en:ex:um:"
8: " write the text - stating character size, beam up, position beam, ":
9: "   beam down, and message":
10: wtb 718,"cs3,:pe0,:pa0,860;:txTHE 1351A",3,":"
11: wtb 718,"cs2,:pe0,:pa0,700;:txoffers TEXT",3,":"
12: wtb 718,"cs1,:pe0,:pa0,600;:txin FOUR",3,":"
13: wtb 718,"cs0,:pe0,:pa0,500;:txSIZES",3,":"
14: wtb 718,"cs7,:pe0,:pa680,0;:txAT TWO",3,":"
15: wtb 718,"cs6,:pe0,:pa820,0;:txdifferent",3,":"
16: wtb 718,"cs5,:pe0,:pa920,0;:txANGLES in BOTH",3,":"
17: wtb 718,"cs4,:pe0,:pa1000,0;:txUPPER and lower CASE",3,":"
18: stp
*7995
```

```
10  REM
20  REM  - - - - - - - - - - - - -   EXAMPLE 3   - - - - - - - - - - - - -
30  REM                               BASIC
40  REM
50  REM                        SHOWS USES OF TEXT OUTPUT
60  REM
70  REM   SET THE ASCII ETX CODE (CONTROL C)
80  LET E$=""
90  REM   SET THE ASCII DC4 CODE (CONTROL T)
100  LET D$=""
110  REM   INITIALIZE THE 1351A
120  PRINT #18;E$,D$,":SN:SX:EM:EN:EX:UM:"
130  REM   WRITE TEXT BY STATING THE CHARACTER SIZE, BEAM UP, POSITION
140  REM     BEAM, BEAM DOWN, TEXT COMMAND WITH THE MESSAGE
150  PRINT #18;"CS3,:PE0,:PA0,860;:TXTHE 1351A",E$,":"
160  PRINT #18;"CS2,:PE0,:PA0,700;:TXoffers TEXT",E$,":"
170  PRINT #18;"CS1,:PE0,:PA0,600;:TXin FOUR",E$,":"
180  PRINT #18;"CS0,:PE0,:PA0,500;:TXSIZES",E$,":"
190  PRINT #18;"CS7,:PE0,:PA680,0;:TXAT TWO",E$,":"
200  PRINT #18;"CS6,:PE0,:PA820,0;:TXdifferent",E$,":"
210  PRINT #18;"CS5,:PE0,:PA920,0;:TXANGLES in BOTH",E$,":"
220  PRINT #18;"CS4,:PE0,:PA1000,0;:TXUPPER and lower CASE",E$,":"
230  STOP
240  END
```

```
0001  FTN,L
0002  C
0003  C - - - - - - - - - - - - -   EXAMPLE 3   - - - - - - - - - - - - -
0004  C                             FORTRAN
0005  C
0006  C                      SHOWS USES OF TEXT OUTPUT
0007  C
0008        PROGRAM GMFE3
0009  C.....  SET THE LU OF THE 1351A
0010        LU1350=18
0011  C.....  SET THE ASCII ETX CODE IN THE UPPER 8 BITS
0012        IETX=3*256
0013  C.....  SET THE ASCII DC4 CODE IN THE UPPER 8 BITS
0014        IDC4=20*256
0015  C.....  INITIALIZE THE 1351A
0016        WRITE(LU1351,1) IETX,IDC4
0017      1 FORMAT(A1,A1,":SN:SX:EM:EN:EX:UM:")
0018  C.....  WRITE TEXT BY STATING THE CHARACTER SIZE, BEAM UP, POSITION
0019  C.....    BEAM, BEAM DOWN, TEXT COMMAND WITH THE MESSAGE
0020        WRITE(LU1351,2) IETX
0021      2 FORMAT("CS3,:PE0,:PA0,860;:TXTHE 1351A",A1,":")
0022        WRITE(LU1351,3) IETX
0023      3 FORMAT("CS2,:PE0,:PA0,700;:TXoffers TEXT",A1,":")
0024        WRITE(LU1351,4) IETX
0025      4 FORMAT("CS1,:PE0,:PA0,600;:TXin FOUR",A1,":")
0026        WRITE(LU1351,5) IETX
0027      5 FORMAT("CS0,:PE0,:PA0,500;:TXSIZES",A1,":")
0028        WRITE(LU1351,6) IETX
0029      6 FORMAT("CS7,:PE0,:PA680,0;:TXAT TWO",A1,":")
0030        WRITE(LU1351,7) IETX
0031      7 FORMAT("CS6,:PE0,:PA820,0;:TXdifferent",A1,":")
0032        WRITE(LU1351,8) IETX
0033      8 FORMAT("CS5,:PE0,:PA920,0;:TXANGLES in BOTH",A1,":")
0034        WRITE(LU1351,9) IETX
0035      9 FORMAT("CS4,:PE0,:PA1000,0;:TXUPPER and lower CASE",A1,":")
0036        STOP
0037        END
```

Example 4: File Usage

Intent.

This example shows how to construct files (figure 6-4), blink a file, erase a file, and update a file (figure 6-5). In addition, this example also shows how to plot special symbols to aid in line differentation.

General Procedure.

1. Initialize the 1351A.

2. Draw the axes and print the title.

3. Place curve 1 into 1351A file 1 by naming the file with an NF1,: command. Also be sure that the beam is off with a PE0,: command.

4. Compute and scale the data point for plotting.

5. Plot the point by using the following format.

   "PA" conv-4-Int "," conv-4-Int ";:PE1,:"

NOTE

See example 3 for definitions used in this format.

---

6. Repeat steps 4 and 5 for all data points to complete the drawing of curve 1.

7. Stop placing data in file 1 with an SN: command.

8. Place curve 2 into 1351A file 2 with a NF2,: command. Also set the character size to 1 with a CS1,: command.

9. Draw curve 2 using a special plotting character. We will use the vertical tick (decimal 11) to designate each data point on the curve.

NOTE

Other special plotting characters include: horizontal tick (12); vertical tick (13); X marker (14); rectangle (15); and a diamond (30). These characters are unique because they do not advance the beam like normal text characters would, and thus are used for plotting. Since the beam is not advanced, these symbols can be combined at each data point to produce special symbols (for example, a vertical and horizontal tick at the same data point combine to form a + symbol). The special plotting characters are affected by the character size command in the same way as all other text characters.

a. Move the beam to the data point with a PA command.

b. Output the text character (vertical tick).

c. Use an ETX character and a colon (:) to end the text command.

The format is as follows:

"PA" conv-4-Int "," conv-4-Int ";:TX" 1AC ETX ":"

where 1AC means to output an ASCII character. In HPL this is 1b, in BASIC it's A, and in FORTRAN it's A1.

10. Repeat steps 8 and 9 for all data points in curve 2.

11. Stop placing data in file 2 with an SN: command.

12. Place curve 3 in 1351A file 3 with a NF3,: command.

13. Draw curve 3 using the special plotting characters of a vertical tick (11) and a horizontal tick (12) at each data point.

a. Turn the beam off with a PE0,: command.

b. Move the beam to the data point with a PA command.

c. Output the Text characters (vertical and horizontal ticks).

d. Use an ETX character and a colon (:) to end the text command.

The format is as follows:

"PE0,:PA" conv-4-Int "," conv-4-Int ":TX" 1AC 1AC ETX":"

14. Stop placing data in file 3 with an SN: command.

15. Blink file 3 the desired number of times.

    a. Blank file 3 with a BF3,: command.

    b. Wait some length of time.

    c. Unblank file 3 with a UF3,: command.

    d. Wait some length of time.

    e. Repeat.

16. Erase all vector, text, and beam control contents from file 2 with an EF2,: command.

17. Update the contents of file 1.

    a. Move the Write Pointer to the first word in 1351A file 1 with a FF1,: command.

    b. Update the file by outputting new curve information with the same method that it was first entered in steps 4-6.

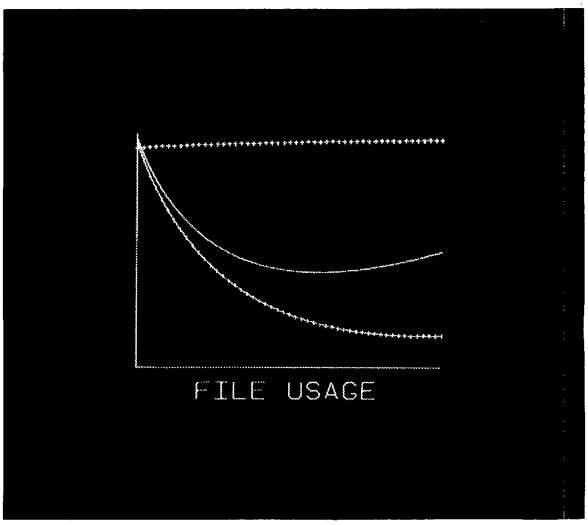


Figure 6-4. Example 4: All Three Files Displayed

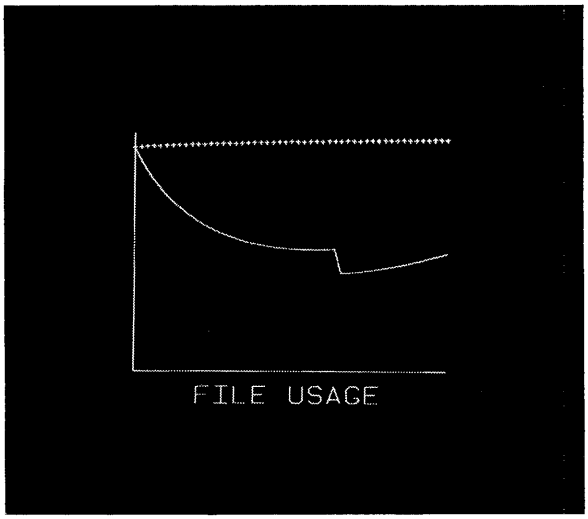


Figure 6-5. Example 5: File 2 Has Been Erased; File 1 is Being Updated

```
0: "                                                                    ":
1: "   - - - - - - - - - - - -   EXAMPLE 4   - - - - - - - - - - - -    ":
2: "                              HPL                                   ":
3: "                                                                    ":
4: "                          1351A FILE USAGE                          ":
5: "                                                                    ":
6: fmt 1,"pa",f4.0,",",f4.0,";:pel,:"
7: fmt 2,"pa",f4.0,",",f4.0,";:tx",lb,lb,":"
8: fmt 3,"pe0,:pa",f4.0,",",f4.0,";:tx",lb,lb,lb,":"
9: " initialize the 1351A":
10: wtb 718,3,20,":sn:sx:em:en:ex:um:"
11: " print the title":
12: wtb 718,"pe0,:pa315,40;:cs2,:txFILE USAGE",3,":"
13: " set the system to radians":rad
14: " draw the axes":
15: wrt 718,"pe0,:pa158,800;:pel,:pa158,130;950,130;:"
16: " compute and draw curve 1, putting it into 1351A file 1":
17: wrt 718,"nf1,:pe0,:"
18: for I=10 to 60
19: "   compute the data point":
20: I/10→A
21: -91n(A)+21n(2)A-2.5+50tan(A/360*2*π)→B
22: "   scale the data point for plotting":
23: A*158.7→X;800+B*66.6→Y
24: "   plot the point":wrt 718.1,X,Y
25: next I
26: " stop placing data in file 1":
27: " compute and draw curve 2, putting it into 1351A file 2":
28: wrt 718,"sn:nf2,:csl,:pe0,:"
29: for I=10 to 60
30: "   compute the data point":
31: I/10→A
32: -91n(A)-2+90tan(A/360*2*π)→B
33: "   scale the data point for plotting":
34: A*158.7→X;800+B*66.6→Y
35: "   plot the point as a vertical tick (11)":
36: wrt 718.2,X,Y,11,3
37: next I
38: " stop placing data in file 2":
39: " compute and draw curve 3, putting it into 1351A file 3":
40: wrt 718,"sn:nf3,:"
41: for I=10 to 60
42: "   compute the data point":
43: I/10→A
44: .21*ln(A)-.6→B
45: "   scale the data point for plotting":
46: A*158.7→X;800+B*66.6→Y
47: "   plot the point as a vert tick (11) and a horz tick (12)":
48: wrt 718.3,X,Y,11,12,3
49: next I
50: " stop naming file 3":wrt 718,"sn:"
```

```
51: " blink 1351A file 3 by a blank/wait/unblank/wait loop":
52: for I=1 to 10
53: wrt 718,"bf3,:";wait 200
54: wrt 718,"uf3,:";wait 200
55: next I
56: wait 2000
57: " erase 1351A file 2":wrt 718,"ef2,:"
58: wait 2000
59: " change file 1 to show a new curve":
60: "   find the starting position of 1351A file 1":
61: wrt 718,"ff1,:pe0,:"
62: "   compute the new curve":
63: for I=10 to 60
64: "   compute the data point":
65: I/10→A
66: -71n(A)+1n(2)A-2.5+65tan(A/360*2*π)→B
67: "   scale the data for plotting":
68: A*158.7→X;800+B*66.6→Y
69: " plot the new point":
70: wrt 718.1,X,Y
71: next I
72: stp
*14942
```

```
10  REM
20  REM  - - - - - - - - - - - - -   EXAMPLE 4   - - - - - - - - - - - - -
30  REM                                 BASIC
40  REM
50  REM   FILE USAGE
60  REM
70  REM   SET THE ASCII ETX CODE (CONTROL C)
80  LET E$=""
90  REM   SET THE ASCII DC4 CODE (CONTROL T)
100  LET D$=""
110  REM   SET THE ASCII CODE FOR A HORIZONTAL TICK (CONTROL L)
120  LET H$=""
130  REM   SET THE ASCII CODE FOR A VERTICAL TICK (CONTROL K)
140  LET V$=""
150  REM   INITIALIZE THE 1351A
160  PRINT #18;E$,D$,":SN:SX:EM:EN:EX:UM:"
170  REM   PRINT THE TITLE
180  PRINT #18;"PE0,:PA315,40;:CS2,:TXFILE USAGE",E$,":"
190  REM   DRAW THE AXES
200  PRINT #18;"PE0,:PA158,800;:PE1,:PA158,130;950,130;:"
210  REM   SET THE VALUE OF 2*PI
220  LET T=2*3.14159
230  REM   COMPUTE AND DRAW CURVE 1, PUTTING IT INTO 1351A FILE 1
240  PRINT #18;"NF1,:PE0,:"
250    FOR I=10 TO 60
260    REM   COMPUTE THE DATA POINT
270    LET A=I/10
280    LET B=-9*LN(A)+2*LN(2)*A-2.5+50*TAN(A/360*T)
290    REM   SCALE THE DATA POINT FOR PLOTTING
300    LET X=A*158.7
310    LET Y=800+B*66.6
320    REM   PLOT THE POINT
330    PRINT #18 USING 340;X,Y
340    IMAGE "PA",4D,",",4D,";:PE1,:"
350    NEXT I
360  REM   STOP PLACING DATA IN FILE 1
370  REM   COMPUTE AND DRAW CURVE 2, PUTTING IT INTO 1351A FILE 2
380  PRINT #18;"SN:NF2,:CS1,:PE0,:"
390    FOR I=10 TO 60
400    REM   COMPUTE THE DATA POINT
410    LET A=I/10
420    LET B=-9*LN(A)-2+90*TAN(A/360*T)
430    REM   SCALE THE DATA POINT FOR PLOTTING
440    LET X=A*158.7
450    LET Y=800+B*66.6
460    REM   PLOT THE POINT AS A VERTICAL TICK (11)
470    PRINT #18 USING 480;X,Y,V$,E$
480    IMAGE "PA",4D,",",4D,";:TX",A,A,":"
490    NEXT I
500  REM   STOP PLACING DATA IN FILE 2
510  REM   COMPUTE AND DRAW CURVE 3, PUTTING IT INTO 1351A FILE 3
520  PRINT #18;"SN:NF3,:"
530    FOR I=10 TO 60
540    REM   COMPUTE THE DATA POINT
550    LET A=I/10
560    LET B=.21*LN(A)-.6
570    REM   SCALE THE DATA POINT FOR PLOTTING
580    LET X=A*158.7
590    LET Y=800+B*66.6
600    REM   PLOT THE POINT AS A HORIZ TICK (12) AND A VERT TICK (11)
```

```
610    PRINT #18 USING 620;X,Y,H$,V$,E$
620    IMAGE "PE0,:PA",4D,",",4D,";:TX",A,A,A,":"
630    NEXT I
640  REM   STOP NAMING FILE 3
650  PRINT #18;"SN:"
660  REM  BLINK FILE 3 BY A BLANK WAIT UNBLANK WAIT LOOP
670    FOR I=1 TO 10
680    PRINT #18;"BF3,:"
690    WAIT (200)
700    PRINT #18;"UF3,:"
710    WAIT (200)
720    NEXT I
730  WAIT (2000)
740  REM   ERASE FILE 2
750  PRINT #18;"EF2,:"
760  WAIT (2000)
770  REM   CHANGE FILE 1 TO SHOW A NEW CURVE
780  REM     FIND THE STARTING POSITION OF FILE 1
790  PRINT #18;"FF1,:PE0,:"
800  REM   COMPUTE THE NEW CURVE
810    FOR I=10 TO 60
820    REM   COMPUTE THE DATA POINT
830    LET A=I/10
840    LET B=-7*LN(A)+LN(2)*A-2.5+65*TAN(A/360*T)
850    REM   SCALE THE DATA POINT FOR PLOTTING
860    LET X=A*158.7
870    LET Y=800+B*66.6
880    REM   PLOT THE NEW POINT
890    PRINT #18 USING 340;X,Y
900    NEXT I
910  STOP
920  END
```

```
0001  FTN,L
0002  C
0003  C - - - - - - - - - - - - -   EXAMPLE 4   - - - - - - - - - - - - -
0004  C                              FORTRAN
0005  C
0006  C                           1351A FILE USAGE
0007  C
0008        PROGRAM GMFE4
0009  C.....  SET THE LU OF THE 1351A
0010        LU1351=18
0011  C.....  SET THE ASCII ETX CODE IN THE UPPER 8 BITS
0012        IETX=3*256
0013  C.....  SET THE ASCII DC4 CODE IN THE UPPER 8 BITS
0014        IDC4=20*256
0015  C.....  SET THE ASCII CODE FOR A VERTICAL TICK (11)
0016        IVT=11*256
0017  C.....  SET THE ASCII CODE FOR A HORIZONTAL TICK (12)
0018        IHT=12*256
0019  C.....  INITIALIZE THE 1351A
0020        WRITE(LU1351,1) IETX,IDC4
0021      1 FORMAT(A1,A1,":SN:SX:EM:EN:EX:UM:")
0022  C.....  PRINT THE TITLE
0023        WRITE(LU1351,2) IETX
0024      2 FORMAT("PE0,:PA315,40;:CS2,:TXFILE USAGE",A1,":")
0025  C.....  DRAW THE AXIS
0026        WRITE(LU1351,3)
0027      3 FORMAT("PE0,:PA158,800;:PE1,:PA158,130;950,130;:")
0028  C.....  SET THE VALUE OF 2*PI
0029        TWOPI=2.*3.141592654
0030  C.....  COMPUTE AND DRAW CURVE 1, PUTTING IT INTO FILE 1
0031        WRITE(LU1351,4)
0032      4 FORMAT("NF1,:PE0,:")
0033        DO 6 I=10,60
0034  C.....    COMPUTE THE DATA POINT
0035          A=FLOAT(I)/10.
0036          B=-9.*ALOG(A)+2.*ALOG(2.)*A-2.5+50.*TAN(A/360.*TWOPI)
0037  C.....    SCALE THE DATA POINT FOR PLOTTING
0038          IX=IFIX(A*158.7)
0039          IY=IFIX(800.+B*66.6)
0040  C.....    PLOT THE POINT
0041          WRITE(LU1351,5) IX,IY
0042      5   FORMAT("PA",I4,",",I4,";:PE1,:")
0043      6 CONTINUE
0044  C.....  STOP PLACING DATA IN FILE 1
0045  C.....  COMPUTE AND DRAW CURVE 2, PUTTING IT INTO FILE 2
0046        WRITE(LU1351,7)
0047      7 FORMAT("SN:NF2,:CS1,:PE0,:")
0048        DO 9 I=10,60
0049  C.....    COMPUTE THE DATA POINT
0050          A=FLOAT(I)/10.
0051          B=-9.*ALOG(A)-2.+90.*TAN(A/360.*TWOPI)
0052  C.....    SCALE THE DATA POINT FOR PLOTTING
0053          IX=IFIX(A*158.7)
0054          IY=IFIX(800.+B*66.6)
0055  C.....    PLOT THE POINT AS A VERTICAL TICK (11)
```

```
0056          WRITE(LU1351,8) IX,IY,IVT,IETX
0057       8  FORMAT("PA",I4,",",I4,";:TX",A1,A1,":")
0058       9 CONTINUE
0059  C.....  STOP PLACING DATA IN FILE 2
0060  C.....  COMPUTE AND DRAW CURVE 3, PUTTING IT INTO FILE 3
0061          WRITE(LU1351,10)
0062      10 FORMAT("SN:NF3,:")
0063          DO 12 I=10,60
0064  C.....    COMPUTE THE DATA POINT
0065            A=FLOAT(I)/10.
0066            B=.21*ALOG(A)-.6
0067  C.....    SCALE THE DATA POINT FOR PLOTTING
0068            IX=IFIX(A*158.7)
0069            IY=IFIX(800.+B*66.6)
0070  C.....    PLOT THE POINT AS A HORZ TICK (12) AND A VERT TICK (11)
0071            WRITE(LU1351,11) IX,IY,IVT,IHT,IETX
0072      11    FORMAT("PE0,:PA",I4,",",I4,";:TX",A1,A1,A1,":")
0073      12 CONTINUE
0074  C.....  STOP NAMING FILE 3
0075          WRITE(LU1351,13)
0076      13 FORMAT("SN:")
0077  C.....  BLINK FILE 3 BY A BLANK/WAIT/UNBLANK/WAIT LOOP
0078          DO 16 I=1,10
0079            WRITE(LU1351,14)
0080      14    FORMAT("BF3,:")
0081            CALL EXEC(12,0,1,0,-20)
0082            WRITE(LU1351,15)
0083      15    FORMAT("UF3,:")
0084            CALL EXEC(12,0,1,0,-20)
0085      16 CONTINUE
0086          CALL EXEC(12,0,1,0,-200)
0087  C.....  ERASE FILE 2
0088          WRITE(LU1351,17)
0089      17 FORMAT("EF2,:")
0090          CALL EXEC(12,0,1,0,-200)
0091  C.....  CHANGE FILE 1 TO SHOW A NEW CURVE
0092  C.....    FIND THE STARTING POSITION OF FILE 1
0093          WRITE(LU1351,18)
0094      18 FORMAT("FF1,:PE0,:")
0095  C.....  COMPUTE THE NEW CURVE
0096          DO 19 I=10,60
0097  C.....    COMPUTE THE DATA POINT
0098            A=FLOAT(I)/10.
0099            B=-7.*ALOG(A)+ALOG(2.)*A-2.5+65.*TAN(A/360.*TWOPI)
0100  C.....    SCALE THE DATA POINT FOR PLOTTING
0101            IX=IFIX(A*158.7)
0102            IY=IFIX(800.+B*66.6)
0103  C.....    PLOT THE NEW POINT
0104            WRITE(LU1351,5) IX,IY
0105      19 CONTINUE
0106          STOP
0107          END
```

Appendix A

AC Power Requirements

The 1351A Graphics Translator requires a power souce of 100, 120, 220, or 240 volts AC +5% −10%, single phase, 48 Hz to 440 Hz. Average power dissipation at 60 Hz, 120 V without any options is approximately 80 watts.

> **CAUTION**
>
> INSTRUMENT DAMAGE MAY RESULT IF THE LINE VOLTAGE SELECTION SWITCHES ARE NOT SET CORRECTLY FOR THE INPUT POWER SOUCE BEING USED.

The instrument is normally set at the factory for the applicable line voltage of the shipment destination. However, this selection should be checked before applying power.

To operate the instrument from any other AC power source, proceed as follows:

1. Disconnect the 1351A power cord from the power source.

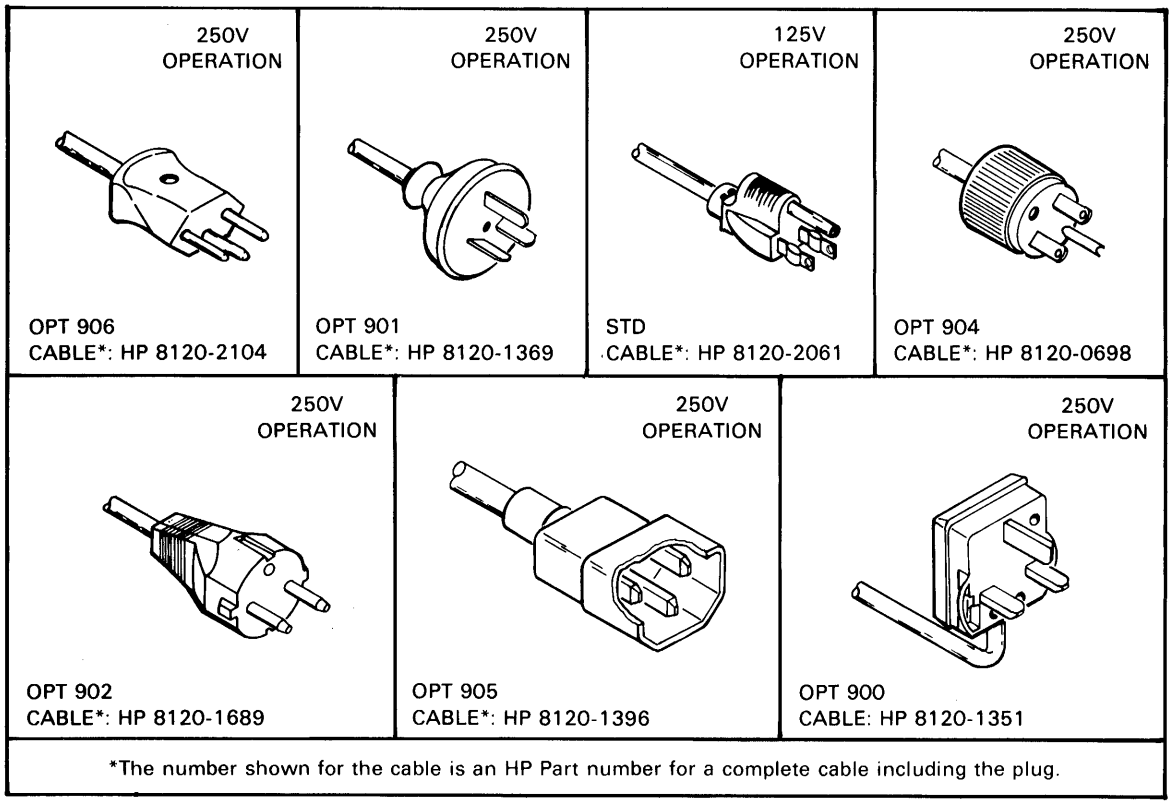
2. Using a blade-type screwdriver, position the rear-panel Line Voltage Select switches for the desired AC voltage input.

3. For 220 V - 240 V inputs, replace the rear-panel fuse F1 with the 800 mAT fuse (HP Part Number 2110-0020).

4. Connect the 1351A input power cord to the power source.

Appendix B
Power Cord

The Power Cord required depends on the AC input voltage and the country in which the instrument is to be used. The following figure illustrates standard power receptacle (wall outlet) configurations. The HP Part Number shown above each receptacle drawing specifies the power cord equipped with the appropriate mating plug for that receptacle. If the appropriate power cord is not included with your instrument, notify the nearest HP Sales/Service Office and a replacement cord will be provided.



HP Power Cord Part Numbers

Appendix C
Auxiliary Connector

The Auxiliary Connector is provided on the rear panel for connecting the 1351A to the Hewlett-Packard Model large screen display with binary Z-axis input.

The signals interfacing the 1351A and binary Z-axis may also be used for other applications. The applications are numerous, and beyond the scope of this manual.

The Hewlett-Packard Part Number for the connector that mates with the Auxiliary Connector is 1251-0063. The Manufacturer's Part Number is ITT DBM-25P. An assembled cable, Hewlett-Packard Part Number 8120-2739, is also available.

All signals, input or output, at the Auxiliary Connector are standard TTL levels.

Following is a description of all 25 pins.

Pin 1. No connection.

Pin 2. No connection.

Pin 3. No connection.

Pin 4. Same as BNC connector for DISPLAY 1. Unblanks Display 1 of 4 to display information. LSB of the binary Z-axis control code.

Pin 5. Same as BNC connector for DISPLAY 2. Unblanks Display 2 of 4 to display information. Second bit of the binary Z-axis control code.

Pin 6. When pin 6 is low, the 1351A is indicating to the device connected to the Auxiliary Connector that it should be in a remote mode, ready to accept information from the 1351A. Pin 6 is also used as a power up reset, holding off pin 9 and pin 13 operation during the reset period.

Pins 7,8. Pins 7 and 8 provide a 2-bit binary code indicting the file name output by the 1351A.

Binary File Name Code

FILE NAME	MSB PIN 7	LSB PIN 8
16-31	0	0
32-47	0	1
48-63	1	0
0-15	1	1

Pin 9. Pin 9 is a positive edge clock that occurs within 25 ns minimum after the binary data. The data is valid only at the clock edge.

Pin 10. Pin 10 is low while drawing vectors in the 1351A.

Pin 11. Pin 11 is +5 V supply with a 10 ohm resistor in series.

Pin 12. Pin 12 is +5 V supply with a 10 kohm resistor in series.

Pin 13. A low state from the device connected to the Auxiliary Connector will hold-off the 1351A from outputing vectors.

Pin 14,15,16. Internal 1351A grounds for signals.

Pin 17. Same as BNC connector for DISPLAY 3. Unblanks Display 3 of 4 to display information. MSB of the binary Z-axis control code.

Pin 18. Same as BNC connector for DISPLAY 4. Unblanks Display 4 of 4 to display information.

Pins 19-25. Internal 1351A grounds for +5 V supply and chassis ground.

Appendix D

1351A Modified ASCII Character Set

Decimal Value		Octal Value	Character
3		003	ETX
8		010	Backspace
9		011	Inverse Line feed
10		012	Line feed
11	*	013	Vertical Tick Mark (|)
12	*	014	Horizontal Tick Mark(—)
13		015	Carriage Return
14	*	016	X Marker Symbol
15	*	017	Rectangle Symbol ▭
19		023	Pointer (←)
20		024	DC4
30	*	036	Diamond Symbol ◇
32		040	space
33		041	!
34		042	"
35		043	#
36		044	$
37		045	%
38		046	&
39		047	'
40		050	(
41		051	)
42		052	*
43		053	+
44		054	,
45		055	-
46		056	.
47		057	/
48		060	0
49		061	1
50		062	2
51		063	3
52		064	4
53		065	5
54		066	6
55		067	7
56		070	8
57		071	9
58		072	:
59		073	;
60		074	<
61		075	=
62		076	>
63		077	?
64		100	@
65		101	A
66		102	B
67		103	C
68		104	D
69		105	E
70		106	F
71		107	G

NOTE: These are plotting characters that do not automatically advance the beam to the next character starting point. They are useful when drawing graphs. The beam is left at the center of the character.

Decimal Value	Octal Value	Character
72	110	H
73	111	I
74	112	J
75	113	K
76	114	L
77	115	M
78	116	N
79	117	O
80	120	P
81	121	Q
82	122	R
83	123	S
84	124	T
85	125	U
86	126	V
87	127	W
88	130	X
89	131	Y
90	132	Z
91	133	[
92	134	$\sqrt{}$
93	135	]
94	136	↑
95	137	←
96	140	\
97	141	a
98	142	b
99	143	c
100	144	d
101	145	e
102	146	f
103	147	g
104	150	h
105	151	i
106	152	j
107	153	k
108	154	l
109	155	m
110	156	n
111	157	o
112	160	p
113	161	q
114	162	r
115	163	s
116	164	t
117	165	u
118	166	v
119	167	w
120	170	x
121	171	y
122	172	z
123	173	π
124	174	|
125	175	→
126	176	Σ
127	177	├─

Subject Index

a

b

c

d

e

f

h

i

l

m

n

p

r

s

t

u

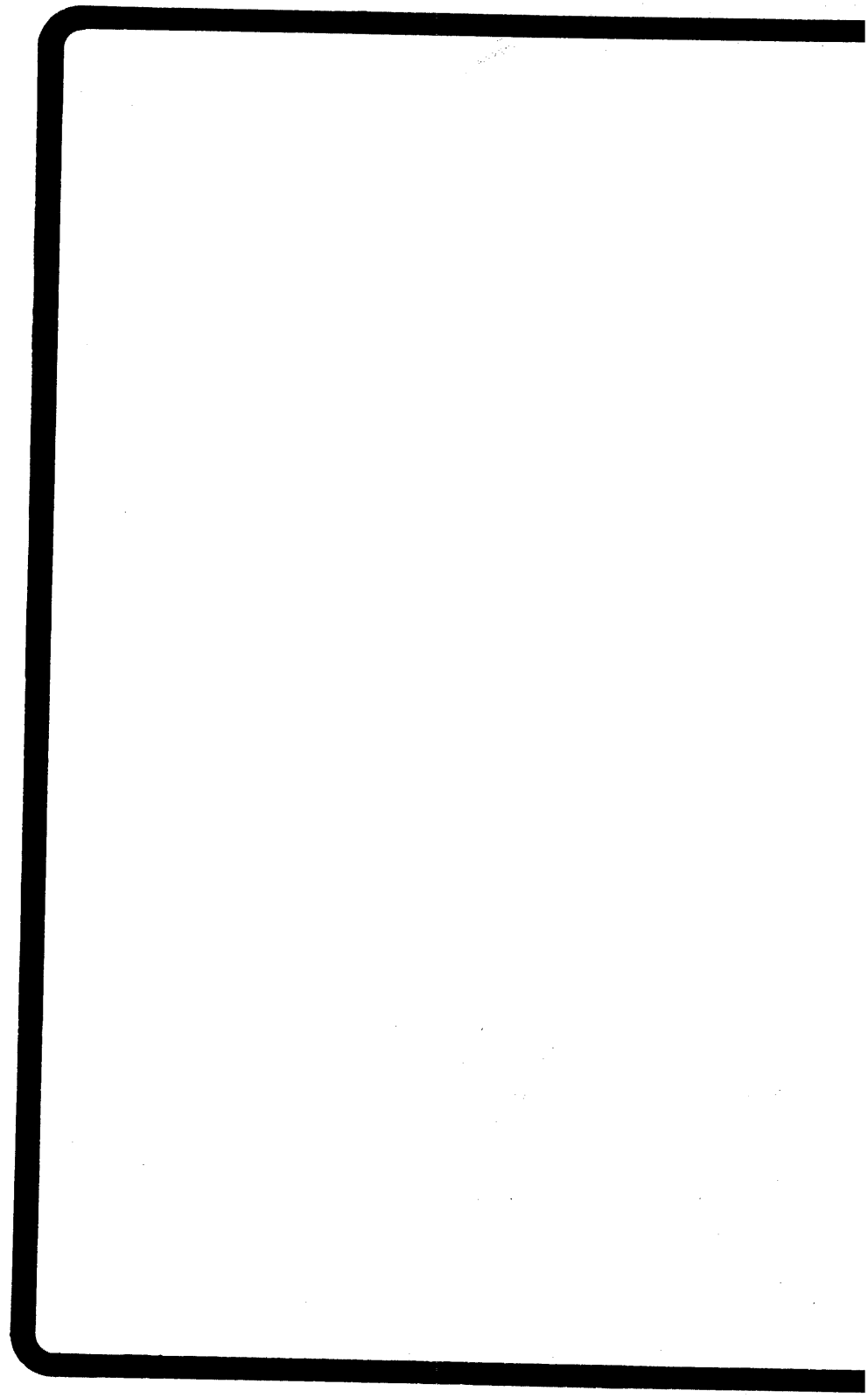
v

w

1351A Functions and Command Subsets

FUNCTION	ASCII COMMAND		PAGE
Initialization	EM	Erase Memory	14
	EN	Erase Names	15
	EX	Erase auXiliary	15
	BM	Blank Memory	15
	UM	Unblank Memory	16
Drawing Vectors (Plotting)	PA	Plot Absolute	18
	PE	Pen Enable	19
File Operations	NF	Name File	30
	SN	Stop Naming	31
	FF	Find File	35-36
	EF	Erase File	38
	BF	Blank File	37
	UF	Unblank File	37
	FL	Find Location	33
Multiple Displays	WX	Write auXiliary	55
	SX	Stop auXiliary	56

Note: See pages 9 and 10 for parameters and syntax.



HEWLETT PACKARD

Part Number 01351-90905

PRINTED IN U.S.A.