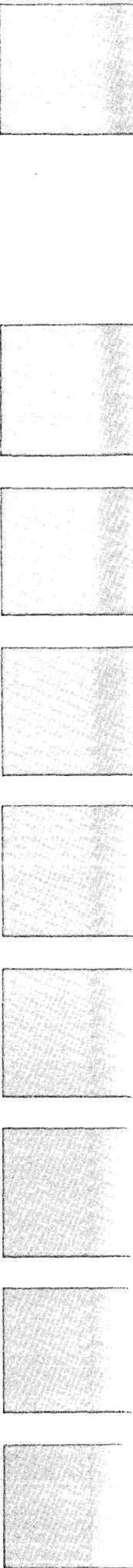


## Systems Reference Library

### IBM 1130/1800 Plotter Subroutines

This publication describes relocatable plotter subroutines for the IBM 1130 and 1800 systems. These subroutines can be used to draw and scale grid lines, to draw special point characters, to draw alphameric characters at various angles, and to plot curves, graphs, charts, and maps. The subroutines can be used in both assembler and FORTRAN language programs.



## PREFACE

This manual describes a group of relocatable plotter subroutines that aids the programmer in converting computed data into graphic form. These subroutines can be used with any IBM 1130 Computing System or 1800 Data Acquisition and Control System that includes an IBM 1627 Plotter, Model 1 or 2. These subroutines supplement the basic plotter subroutine included as a part of the 1130 and 1800 subroutine libraries and can be used with any of the following IBM programming systems:

- 1130 Card/Paper Tape
- 1130 Disk Monitor
- 1800 Card/Paper Tape

- 1800 Time-Sharing Executive (Phase II)

Core storage requirements are given in Appendix A, subroutine usage is listed in Appendix B, and two sample programs are shown in Appendix C.

Descriptions of related publications can be found in the following:

- IBM 1130 Computing System Bibliography  
(Form A26-5916)
- IBM 1800 Data Acquisition and Control System Bibliography (Form A26-5921)

Copies of this and other IBM publications can be obtained through IBM Branch Offices. A form has been provided at the back of this publication for reader's comments. If the form has been detached, comments may be directed to: IBM, Programming Publications Dept. 452, San Jose, Calif. 95114

## CONTENTS

INTRODUCTION .....	1	Assembler .....	5
General .....	1	Point .....	5
Plotter Characteristics .....	1	FORTRAN .....	6
Plotter Capabilities .....	1	Assembler .....	6
Subroutine Functions .....	2	Character .....	6
Input Format .....	2	FORTRAN .....	6
		Assembler .....	7
PLOTTER SUBROUTINES .....	3	Annotation .....	7
Scale .....	3	FORTRAN .....	7
FORTRAN .....	3	Assembler .....	8
Assembler .....	4		
Grid .....	4	APPENDIX A. CORE STORAGE REQUIREMENTS .....	9
FORTRAN .....	4		
Assembler .....	5	APPENDIX B. SUBROUTINE USAGE .....	10
Plot .....	5		
FORTRAN .....	5	APPENDIX C. SAMPLE PROGRAMS .....	11

GENERAL

When connected to an IBM 1130 Computing System or an IBM 1800 Data Acquisition and Control System, the IBM 1627 Plotter can be programmed to produce bar charts, organization charts, engineering drawings, maps, or special curves. This manual describes a set of subroutines, written in assembler language, used to control the plotter. The subroutines can also be called from a FORTRAN language program.

PLOTTER CHARACTERISTICS

The basic characteristics of the plotter are listed in Table 1.

X-axis movement is produced by rotating the chart paper on the drum under the pen. Rotating the drum down causes the pen to draw a line, effectively, in the up direction; this movement is the positive X-axis motion. Rotating the chart paper up produces the negative X-axis motion.

Y-axis movement is caused by moving the pen parallel to the drum axis. When looked at from the front of the plotter, the positive Y-axis motion is to the left; the negative, to the right.

Thus, the length of the X-axis is limited by the length of the roll of chart paper, and the length of the Y-axis is determined by the paper's width. Various combinations of paper and pen movement with the pen up or down are utilized to produce the desired drawings.

Table 1. IBM 1627 Plotter Characteristics

Characteristic	Model 1	Model 2
Chart Paper Width	12 inches	31 inches
Plotting Width	11 inches	29-1/2 inches
Chart Paper Length	120 feet	120 feet
Plotting Length	120 feet	120 feet
Incremental Step Size	1/100 inch	1/100 inch
Step Speed	Up to 18,000 steps/min.	Up to 12,000 steps/min.
Pen Status Change	600 Operations/min.	600 Operations/min.

PLOTTER CAPABILITIES

The plotter generates all lines by using a series of incremental straight line segments. The increment length is 0.01 inch, drawn in either a positive or a negative direction, parallel to either the X-axis or the Y-axis. Also, the paper and pen can be moved simultaneously to produce line increments at a 45° angle to either axis in either direction. This results in a diagonal line connecting opposite corners of an X-Y square. Combinations of increments at various angles can closely approximate any desired curve.

Preciseness of lines and characters depends on the size of the pen point and the scale selected by the programmer. Figure 1 illustrates the smallest line increments capable of being generated and a sample curve made by using these increments.

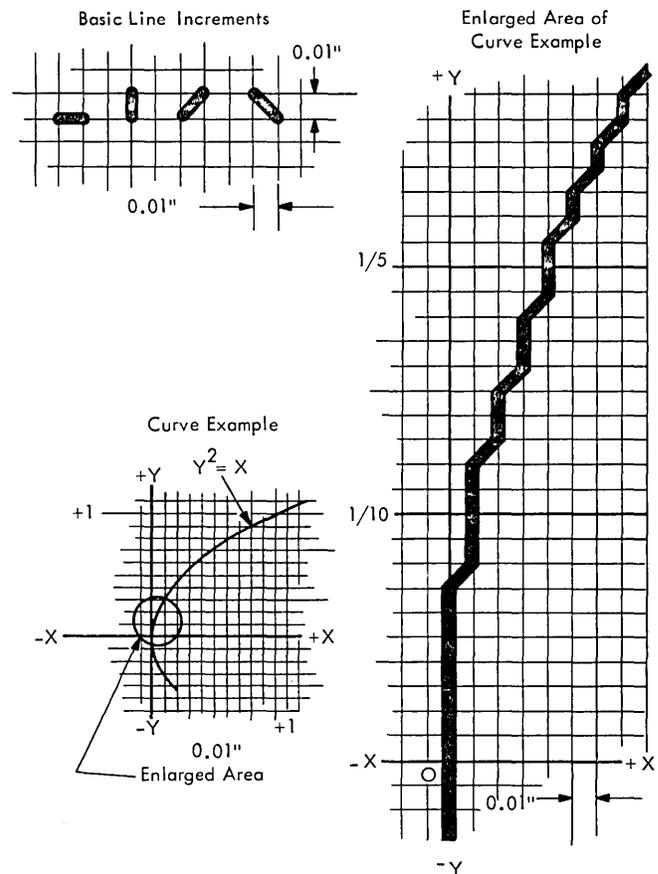


Figure 1. Plotter Line Generation

Graphs, curves, charts, etc., can be developed by programming the appropriate instructions to the plotter.

Because of the relative slowness of the plotter, compared with the computer, the plotter system has a buffering scheme which holds plotter instructions until they are executed. This leaves the computer free to do other jobs while the actual plotting is being completed.

## SUBROUTINE FUNCTIONS

There are six primary functions of the plotter subroutines described in this manual.

- **Scale:** Accepts and stores scaling information required by the grid and plot functions.
- **Grid:** Draws a line with scaled grid marks.
- **Plot:** Moves the pen from its present position to a new position. It can also raise or lower the pen either before or after the traverse movement.
- **Point:** Draws a special point character at the present location of the pen, if the pen is down.

The point characters available are +, ×, ∇, ◁, △, and ▷. All point characters are fixed in size.

- **Character:** Positions the pen for annotation and provides character size and angle information.
- **Annotation:** Forms the characters to be plotted from computer output data. Characters available are 0-9, A-Z, and ␣ - + . , \* / = ( ) \$ '. (The symbol ␣ represents a blank character.)

## INPUT FORMAT

The input data to the subroutines can be either in extended or standard precision format, but different subroutines are required for each precision, with the exception of the point subroutine. For example, to perform the plot function in standard precision, the FPLOTT subroutine is used; for extended precision, the EPLOTT subroutine is used. Standard precision uses two 16-bit words to form a constant or variable, while extended precision uses three 16-bit words for the same constant or variable.

SCALE

If the scale subroutine is not called, the plot subroutine assumes initial scale values of one inch per unit along both axes and establishes the origin (intersection of the X axis and the Y axis) at the present pen position. However, the scale subroutine must be called to define other scale factors and to establish the origin at other points. The scale subroutine can be called as often as required to redefine the scaling values and the origin position (see Figure 2).

Desired Scale: X Direction: 0.25 Inch Per User's Unit  
 Y Direction: 0.05 Inch Per User's Unit

Call Example (FORTRAN):

CALL SCALE (0.25, 0.05, -5., 30.)  
 X,Y Scale Factor Present Pen Location  
 (Inches Per User's Unit) (User's Units)

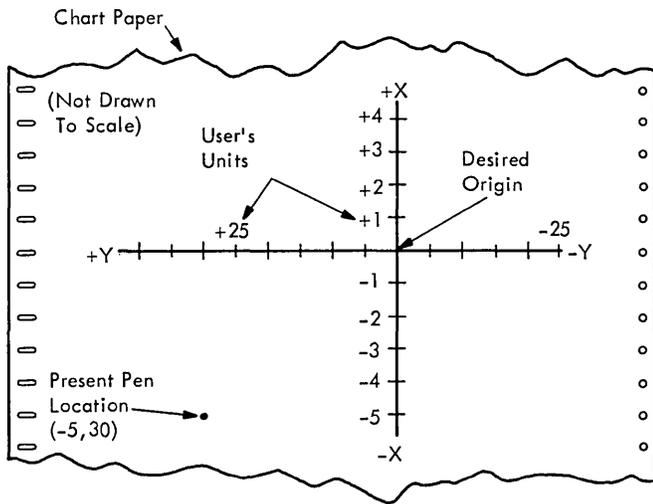


Figure 2. Scaling and Origin Location (Example)

NOTE: Each time the scale subroutine is called, the origin established is based on the physical location of the pen. Therefore, the pen must be moved to the position assumed by the subroutine before the subroutine is called. Also, the pen position cannot be moved more than 327.67 inches in either X direction from its physical location at the time the origin was last established.

Scale values are given in inches per unit of the using program. For example, to indicate a scale of 1/4 inch equal to 1 foot, the scale value would be 0.25. To indicate a scale of 1 inch equal to 10 years, the scale would be 0.1. Different scale values can be assigned to the X axis and the Y axis.

The pen is usually aligned by reticle adjustment to some point on the chart paper. The scale subroutine establishes the origin at any desired point relative to the physical location of the pen when the subroutine is called. Thereafter, or until the origin is moved, all measurements are calculated from this origin to prevent an accumulation of errors which would result from measuring from point to point with calculated values that have been rounded off or truncated.

The values inserted by this subroutine are positive or negative as measured perpendicularly from each desired axis to the present location of the pen.

FORTRAN

Standard precision:

CALL SCALF (X<sub>s</sub>, Y<sub>s</sub>, X<sub>o</sub>, Y<sub>o</sub>)

Extended precision:

CALL SCALE (X<sub>s</sub>, Y<sub>s</sub>, X<sub>o</sub>, Y<sub>o</sub>)

X<sub>s</sub> is a real constant or variable that defines the number of inches per user's unit to be used along the X axis.

$Y_S$  is a real constant or variable that defines the number of inches per user's unit to be used along the Y axis.

$X_O$  is a signed real constant or variable that specifies the X value of the present position of the pen relative to the desired origin, measured in the user's units.

$Y_O$  is a signed real constant or variable that specifies the Y value of the present position of the pen relative to the desired origin, measured in the user's units.

### Assembler

Standard precision:

CALL	SCALF
DC	A
DC	B
DC	C
DC	D

Extended precision:

CALL	SCALE
DC	A
DC	B
DC	C
DC	D

A, B, C, and D are the addresses of constants or variables whose values correspond to  $X_S$ ,  $Y_S$ ,  $X_O$ , and  $Y_O$ , respectively, as described in the FORTRAN calls.

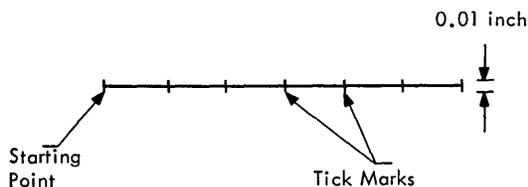
For the standard precision calling sequence, each constant or variable designated by A, B, C, and D must be two words. This is accomplished best by using the DEC statement. For extended precision, each constant or variable must be three words. This is accomplished best by using the XFLC statement.

### GRID

The grid subroutine plots a straight line parallel to either the X or the Y axis in a positive or a negative direction with tick marks at regularly spaced intervals. The tick marks are 0.10 inch long, with one half of the mark on each side of, and perpendicular to, the grid line. The programmer

must specify the starting point, the direction to be plotted, how far to go, and the distance between tick marks.

It is not required to know either the location of the pen or whether it is up or down when this subroutine is called. At the end of the subroutine, the pen is left in the up position.



### FORTRAN

Standard precision:

CALL FGRID (I, X, Y, L, N)

Extended precision:

CALL EGRID (I, X, Y, L, N)

I is an integer constant or variable that specifies the direction the grid line is to be generated, as follows:

- I = 0 specifies the +X direction
- I = 1 specifies the +Y direction
- I = 2 specifies the -X direction
- I = 3 specifies the -Y direction

Only the last two bits of I are used to determine the direction; therefore, use of larger numbers would repeat the instructions. For example, 24 or 4 plot the same as 0 (+X direction).

X is a signed real constant or variable that specifies the X value of the grid line starting point, measured in the user's units.

Y is a signed real constant or variable that specifies the Y value of the grid line starting point, measured in the user's units.

L is a real constant or variable that specifies the distance between tick marks, measured in the user's units.

N is an integer constant or variable that defines the length of the grid line. N is equal to the number of tick marks minus one, and must be less than 32768.

## Assembler

### Standard precision:

CALL	FGRID
DC	A
DC	B
DC	C
DC	D
DC	E

### Extended precision:

CALL	EGRID
DC	A
DC	B
DC	C
DC	D
DC	E

A, B, C, D, and E are the address of constants or variables corresponding to I, X, Y, L, and N, respectively, of the FORTRAN language calls.

The constants or variables represented by A and E should be one word each in both standard and extended precision calls. Each constant or variable represented by B, C, and D must be two words for standard precision and three words for extended precision.

## PLOT

This subroutine is called to move the pen from its present position to a new position. The pen can also be raised or lowered before or after the traverse movement, as a part of this subroutine.

NOTE: If the pen is up (or down), and a command is given to go up (or down), 100 milliseconds of plot time are wasted in performing the redundancy.

## FORTRAN

### Standard precision:

```
CALL      FPLOT (I, Xn, Yn)
```

### Extended precision:

```
CALL      EPLOT (I, Xn, Yn)
```

I is an integer expression controlling the pen as follows:

I = Zero	No change
I = Positive	Control pen before movement
I = Negative	Control pen after movement
I = Odd	Raise pen
I = Even	Lower pen

X<sub>n</sub> is a signed real constant or variable defining the X value of the new position, measured in the user's units.

Y<sub>n</sub> is a signed real constant or variable defining the Y value of the new position, measured in the user's units.

## Assembler

### Standard precision:

CALL	FPLOT
DC	A
DC	B
DC	C

### Extended precision:

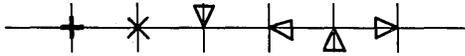
CALL	EPLOT
DC	A
DC	B
DC	C

A, B, and C are the addresses of constants or variables corresponding to I, X<sub>n</sub> and Y<sub>n</sub>, respectively, of the FORTRAN calls.

The constant or variable represented by A should be only one word in both standard and extended precision. Each constant or variable represented by B and C must be two words for standard precision and three words for extended precision.

## POINT

The point subroutine draws special point characters at the present position of the pen. The pen must be down when this subroutine is called. The point characters, and the points about which they are drawn, are illustrated below. All points are at the intersection of the horizontal and vertical line.



This subroutine assumes the pen is down and leaves the pen down when finished. Each point character is inscribed within a 0.10 inch square.

### FORTTRAN

The pen remains in the down position:

CALL POINT (I)

I is an integer expression that defines the character to be drawn as follows:

- I < 0 blank
- I = 0 +
- I = 1 x
- I = 2 ▽
- I = 3 ◁
- I = 4 △
- I = 5 ▷
- I ≥ 6 blank

### Assembler

The pen remains in the down position:

CALL POINT  
DC A

A is the address of a one word constant or variable whose contents correspond to the I of the FORTRAN call. The constant or variable represented by A must be only one word in length.

### CHARACTER

This subroutine is used to initialize the annotation subroutine by establishing the height and width of characters, the angle (relative to the X axis) they are drawn, and the starting location. Calling this subroutine also raises the pen (if down) and moves the pen to the specified starting location. The height and width parameters determine a rec-

tangle inside of which each character is drawn (see Figure 3). Once called, this subroutine does not have to be called again unless it is desired to change the parameters established by this subroutine.

### FORTTRAN

Standard precision:

CALL FCHAR (X<sub>n</sub>, Y<sub>n</sub>, X<sub>s</sub>, Y<sub>s</sub>, Theta)

Extended precision:

CALL ECHAR (X<sub>n</sub>, Y<sub>n</sub>, X<sub>s</sub>, Y<sub>s</sub>, Theta)

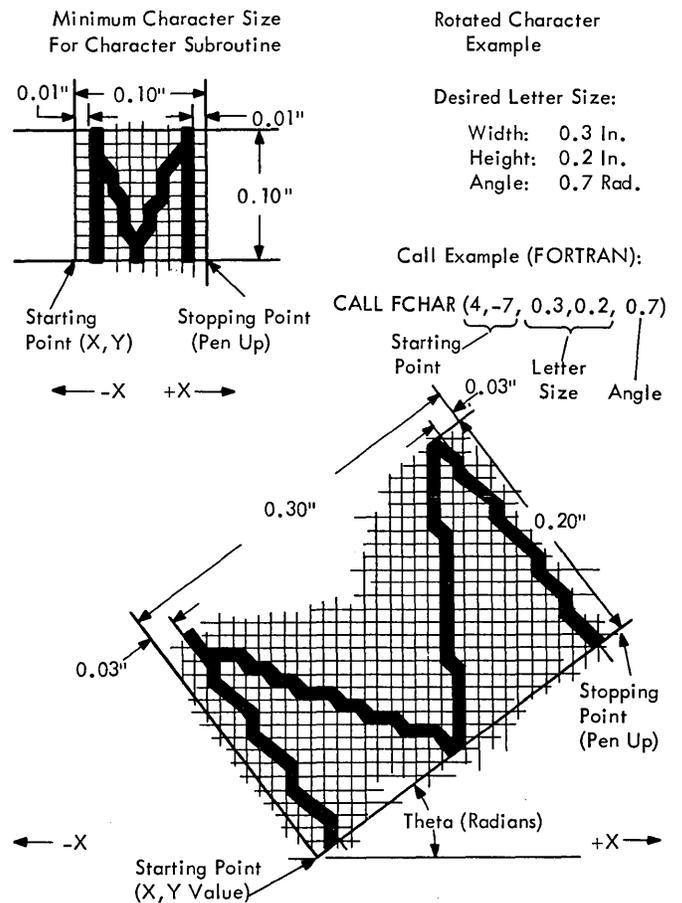


Figure 3. Character Plotting (Example)

$X_n$  is a signed real variable or constant defining the X value of the starting location.

$Y_n$  is a signed real variable or constant defining the Y value of the starting location.

$X_s$  is an unsigned real variable or constant defining the width of the character, expressed in inches. A value exceeding two decimal places will be rounded off to the nearest 0.01.

$Y_s$  is an unsigned real variable or constant defining the height of the character, expressed in inches. A value exceeding two decimal places will be rounded off to the nearest 0.01.

Theta is a signed real variable or constant defining the angle at which the character (or line of characters) is to be drawn, expressed in radians. Theta is measured by rotating a line parallel to the X axis about the starting location. Positive values produce counterclockwise rotation; negative values, the opposite.

Assembler

Standard precision:

CALL	FCHAR
DC	A
DC	B
DC	C
DC	D
DC	E

Extended precision:

CALL	ECHAR
DC	A
DC	B
DC	C
DC	D
DC	E

A, B, C, D, and E are the addresses of real constants or variables corresponding to the  $X_n$ ,  $Y_n$ ,  $X_s$ ,  $Y_s$ , and Theta, respectively, of the FORTRAN calling sequence. The constants or variables represented by A, B, C, D, and E must be two words in standard precision and three words in extended precision.

NOTE: When the plotter subroutines are first loaded,  $X_n$  and  $Y_n$  are set to 0.10 inch and Theta is set to zero. If the annotation subroutine is called without calling the character subroutine, the characters are plotted at this size and angle, starting at the present position of the pen. These values are replaced by those of the character subroutine when first called and, if needed, must be reinserted by using the character subroutine.

ANNOTATION

This subroutine forms the characters specified by computer output data to the parameters established by the character subroutine. These parameters determine a rectangle inside of which each character is drawn. The starting location is the lower left corner of the rectangle. In a continuous row of characters, the starting location is the lower left corner of the first character. When the last character is completed, this subroutine stops the pen in the up position over the lower left corner of the next character position in sequence. Repetitive lines are plotted end to end. The character set available is 0-9, A-Z, and  $\backslash - + . , * / = ( ) \$ ' .$

FORTRAN

Standard or extended precision:

WRITE (I, FORMAT) List

I is the unsigned integer constant or variable that specifies the logical unit number of the I/O unit (plotter) to be used for output data. This number can be changed in the 1800 system when establishing the FORTRAN I/O unit (IOU) table.

FORMAT is a statement number of the FORMAT statement describing the type of data conversion to be performed between the internal and external representation of each quantity in the list.

List is a list of variable names, separated by commas, which represent the output data.

Assembler

Standard precision:

LIBF	FCHRI
DC	(Non-Zero)
DC	A
DC	0 (Not Used)
.	.
.	.
.	.
A DC	U
DC	0 (Not Used)
EBC	V
EBC	W
.	.
.	.
.	.

Extended precision:

LIBF	ECHRI
DC	(Non-Zero)
DC	A

DC	0 (Not Used)
.	.
.	.
.	.
A DC	U
DC	0 (Not Used)
EBC	V
EBC	W
.	.
.	.
.	.

A is an arbitrarily assigned label.

U is an unsigned integer expressing the total number of words contained under the label A, including one for the unused DC statement (0).

V, W, etc., are the output data in EBCDIC. There are two 8-bit characters per word with a maximum of 18 words for any one EBC statement.

NOTE: If 18 words are used, the last character of the 18th word will be blank because of the restrictions of the EBC statement.

APPENDIX A. CORE STORAGE REQUIREMENTS

Subroutine Function	Subroutine Name	Core Size (Words)	Remarks
Annotation	FCHRI*	580	Standard precision
	ECHRI*	588	Extended precision
Character	FCHAR	52	Standard precision
	ECHAR	58	Extended precision
Grid	FGRID	102	Standard precision
	EGRID	110	Extended precision
Plot	FPLOT	48	Standard precision
	EPLOT	56	Extended precision
Point	POINT	108	Standard and ext. precision
Scale	SCALF	4	Standard precision
	SCALE	4	Extended precision
Plot Scaler	FRULE	114	Standard precision
	ERULE	130	Extended precision
Pen Mover	XYPLT	94	Standard and ext. precision
Interface	PLOTI	18	

\*In FORTRAN this function is performed using the WRITE statement.

APPENDIX B. SUBROUTINE USAGE

Subroutine		Other Subroutines Required	
Function	Name	Called by CALL	Called by LIBF
Annotation	FCHRX, FCHRI*, WCHRI*		FLOAT, FMPY, IFIX, FADD, FLDX, FINC, XYPLT, PLOTI, FSTOY, FLD
	ECHRX, ECHRI*, VCHRI*		FLOAT, EMPY, IFIX, EADD, ELDX, EINC, XYPLT, PLOTI, ESTOY, ELD
Character	FCHAR	FSIN, FCOS, FPLOT	FCHRX, FLD, FSTOX, FSTO
	ECHAR	ESIN, ECOS, EPLOT	ECHRX, ELD, ESTOX, ESTO
Grid	FGRID	FPLOT, POINT	FADD, FLD, FSTO, SNR
	EGRID	EPLOT, POINT	EADD, ELD, ESTO, SNR
Plot	FPLOT		FMOVE, XYPLT, PLOTI
	EPLOT		EMOVE, XYPLT, PLOTI, ELD, ESTO
Point	POINT		PLOTI
Scale	SCALF		FRULE
	SCALE		ERULE
Plot Scaler	FRULE, FMOVE, FINC		FLDX, FSUBX, FMPYX, FLD, FSTOX, FMPY, IFIX, FADD
	ERULE, EMOVE, EINC		ELDX, ESUBX, EMPYX, ELD, ESTOX, EMPY, IFIX, ESTO, EADD
Pen Mover	XYPLT		PLOTI
Interface	PLOTI		PLOTX

\*Separate entry points to the Annotation Routine

This appendix contains two sample plotter programs utilizing the IBM 1130 Monitor System.

Test Program 4 demonstrates the character-drawing capabilities of the plotter. Figure 4 shows a listing of this program, and Figure 5 shows the

resultant plotter drawing.

Test Program 5 demonstrates the curve-plotting capabilities of the plotter. Figure 6 shows a listing of this program, and Figure 7 shows the resultant plotter drawing.

```
**PLOTTER TEST 4
*NAME PLOO4
*LIST ALL
*ONE WORD INTEGERS
*IOCS(1132 PRINTER,PLOTTER,CARD)
```

PAGE 01

```
PLOTTER TEST 4
C TEST CHARACTER ROUTINE
  DIMENSION IA(47),IB(36),IC(10),ID(26)
  EQUIVALENCE (IA(1),IB(1),IC(1)),(IA(11),ID(1))
1  FORMAT(' PLOO4')
2  FORMAT(' THIS IS A TEST OF THE CHARACTER ROUTINE'/' TEST IS IN 5'
  1' PARTS'/' PART 1 - PLOT THE FOLLOWING CHARACTERS .2 IN SIZE')
3  FORMAT(1H 47A1)
4  FORMAT('/' PART 2 - PLOT CHARACTERS A-Z .27W BY .37H HORIZONTAL')
5  FORMAT(47A1)
6  FORMAT('/' PART 3 - PLOT CHARACTERS A-Z .25W BY .15H ON A 45')
7  FORMAT('/' PART 4 - PLOT CHARACTERS 0-9 .15W BY .25H VERTICAL')
8  FORMAT('/' PART 5 - PLOT CHARACTERS 0-9,A-Z .15 BY .15 ON A'
  1' 3 INCH CIRCLE')
  READ (2,5)IA
  WRITE (3,1)
  WRITE (3,2)
  PI = 3.14159
  WRITE (3,3)IA
  CALL FCHAR(0.,0.,.2,.2,0.)
  WRITE (7,5)IA
  WRITE (3,4)
  CALL FCHAR(0.,.3,.27,.37,0.)
  WRITE (7,5)ID
  WRITE (3,6)
  CALL FCHAR(.27,.75,.25,.15,PI/4.)
  WRITE (7,5)ID
  WRITE (3,7)
  CALL FCHAR(.25,1.2,.15,.25,PI/2.)
  WRITE (7,5)IC
  WRITE (3,8)
  T = 0.
  DEL = PI/18.
  DO 40 I = 1,36
  CALL FCHAR(2.+1.5*COS(T),5.75 +1.5*SIN(T),.15,.15,T)
  WRITE (7,5)IB(I)
40  T = T + DEL
  CALL FCHAR(1.,7.9,.1,.1,0.)
  WRITE (7,1)
  WRITE (3,1234)
  CALL EXIT
1234 FORMAT('/' END OF TEST'/'/)
C
C PLACE PLOTTER IN LOWER LEFT CORNER OF IMAGINARY
C 8 X 9 INCH RECTANGLE (8 = Y)
C
  END
```

PAGE 02

Figure 4. Plotter Test 4 Listing (Part 1 of 2)

PLOTTER TEST 4

PAGE 03

VARIABLE ALLOCATIONS

IA =002E IB =002E IC =002E ID =0024 PI =003A T =003C DEL =003E I =0044

STATEMENT ALLOCATIONS

1 =006F 2 =0074 3 =0083 4 =0088 5 =00D7 6 =00DA 7 =00F7 8 =0115 1234 =013A 40 =01F8

FEATURES SUPPORTED

ONE WORD INTEGERS  
IOCS

CALLED SUBPROGRAMS

FCHAR FCOS FSIN FADD FMPY FDIV FLD FSTO SRED SWRT SCOMP SFIO SIOAI SIOIX SUBSC  
CARDZ PRNTZ WCHR1

REAL CONSTANTS

.314159E 01=0046 .000000E 00=0048 .200000E 00=004A .300000E 00=004C .270000E 00=004E .370000E 00=0050  
.750000E 00=0052 .250000E 00=0054 .150000E 00=0056 .400000E 01=0058 .120000E 01=005A .200000E 01=005C  
.180000E 02=005E .150000E 01=0060 .575000E 01=0062 .100000E 01=0064 .790000E 01=0066 .100000E 00=0068

INTEGER CONSTANTS

2=006A 3=006B 7=006C 1=006D 36=006E

CORE REQUIREMENTS FOR PL004

COMMON 0 VARIABLES 70 PROGRAM 464

END OF COMPILATION

// XEQ

PL004

THIS IS A TEST OF THE CHARACTER ROUTINE  
TEST IS IN 5 PARTS

PART 1 - PLOT THE FOLLOWING CHARACTERS .2 IN SIZE  
0123456789ABCDEFGHIJKLMN0PQRSTUVWXYZ -+.\*/( )\$=,

PART 2 - PLOT CHARACTERS A-Z .27W BY .37H HORIZONTAL

PART 3 - PLOT CHARACTERS A-Z .25W BY .15H ON A 45

PART 4 - PLOT CHARACTERS 0-9 .15W BY .25H VERTICAL

PART 5 - PLOT CHARACTERS 0-9,A-Z .15 BY .15 ON A 3 INCH CIRCLE

END OF TEST

Figure 4. Plotter Test 4 Listing (Part 2 of 2)

FL004

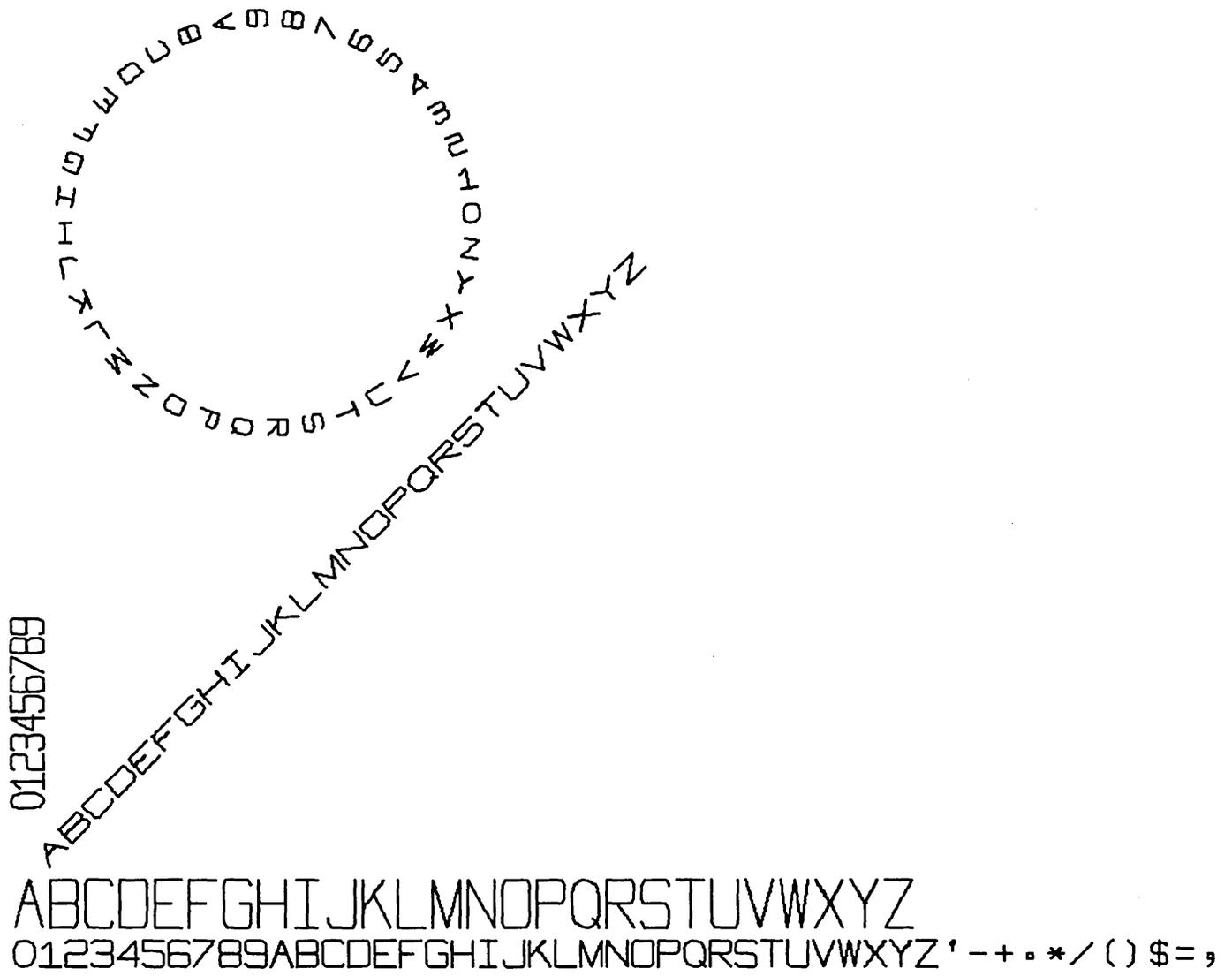


Figure 5. Plotter Test 4 Drawing

```

**PLOTTER TEST 5
*NAME PLO05
*LIST ALL
*ONE WORD INTEGERS
*IOCS(1132 PRINTER,PLOTTER)

```

PAGE 01

```

PLOTTER TEST 5
1  FORMAT(' PLO05')
3  FORMAT(' Y = SIN(2.*X) + COS(3.*X), X = -2.*PI, 2.*PI')
4  FORMAT(F7.4)
5  FORMAT(F7.3)
   WRITE(3,1)
   WRITE(3,3)
   PI = 3.14159
   SX = 14./4.*PI
   SY = 7./4.
   DY = .25
   PX = -2.*PI
   PY = -2.1
   CALL FPLLOT(2,0,0)
   CALL SCALF(SX,SY,PX,PY)
   CALL FGRID(0,0,0,PI/4,8)
   CALL FGRID(2,0,0,PI/4,8)
   CALL FGRID(1,0,0,DY,8)
   CALL FGRID(3,0,0,DY,8)
   T = 3.*PI/2.
   XD = .050/SX
   X = -2.*PI
   Y = -.05
   DEL = PI/4.
   EP = .001
   DO 10 I = 1,17
   IF(ABS(X) - EP)9,9,7
7  CALL FCHAR(X-XD,Y,.10,.10,T)
   WRITE(7,4)X
9  X = X + DEL
10 CONTINUE
   T = 0.
   Y = -2.
   X = -0.770/SX
   DEL = DY
   XD = .05/SY
   DO 20 I = 1,17
   IF(ABS(Y) - EP)19,11,11
11 CALL FCHAR(X,Y-XD,.10,.10,T)
   WRITE(7,5)Y
19 Y = Y + DEL
20 CONTINUE
   X = -2.*PI
   CALL FPLLOT(-2,X,SIN(2.*X)+COS(3.*X))
   DEL = PI/225.
   TEST = 2.*PI
30 Y = SIN(2.*X)+COS(3.*X)
   CALL FPLLOT(0,X,Y)
   IF(TEST-X)50,50,40
40 X = X + DEL
   GO TO 30
50 CALL FCHAR(-22.*.11/SX,-2.-.40/SY,.11,.16,0.)
   WRITE(7,3)
   CALL FPLLOT(-2,PX,PY)
   CALL FCHAR(-1.9*PI,-2.15,.1,.1,0.)
   WRITE(7,1)
   WRITE(3,1234)
   CALL EXIT
1234 FORMAT(//' END OF TEST'//)
C

```

PAGE 02

Figure 6. Plotter Test 5 Listing (Part 1 of 2)

```

PLOTTER TEST 5
C   PLACE PLOTTER IN LOWER LEFT CORNER OF IMAGINARY
C   7 X 14 INCH RECTANGLE (7 = Y)
C
END

```

PAGE 03

PLOTTER TEST 5

PAGE 04

```

VARIABLE ALLOCATIONS
PI  =0000 SX  =0002 SY  =0004 DY  =0006 PX  =0008 PY  =000A T   =000C XD  =000E X   =0010 Y   =0012
DEL =0014 EP  =0016 TEST =0018 I   =0024

```

```

STATEMENT ALLOCATIONS
1   =0055 3   =005A 4   =0073 5   =0075 1234 =0077 7   =0123 9   =0136 10  =013C 11  =016A 19  =017D
20  =0183 30  =01C0 40  =01E4 50  =01EC

```

```

FEATURES SUPPORTED
ONE WORD INTEGERS
IOCS

```

```

CALLED SUBPROGRAMS
FPLOT SCALF FGRID FABS FCHAR FSIN FCOS FADD FSUB FMPY FDIV FLD FSTO FSBR FDVR
SWRT SCOMP SF10 S10F SNR PRNTZ WCHR1

```

```

REAL CONSTANTS
.314159E 01=0026 .140000E 02=0028 .400000E 01=002A .700000E 01=002C .250000E 00=002E .210000E 01=0030
.000000E 00=0032 .300000E 01=0034 .200000E 01=0036 .500000E-01=0038 .100000E-02=003A .100000E 00=003C
.770000E 00=003E .225000E 03=0040 .220000E 02=0042 .110000E 00=0044 .400000E 00=0046 .160000E 00=0048
.190000E 01=004A .215000E 01=004C

```

```

INTEGER CONSTANTS
3=004E 2=004F 0=0050 8=0051 1=0052 17=0053 7=0054

```

```

CORE REQUIREMENTS FOR PLO05
COMMON 0 VARIABLES 38 PROGRAM 526

```

END OF COMPILATION

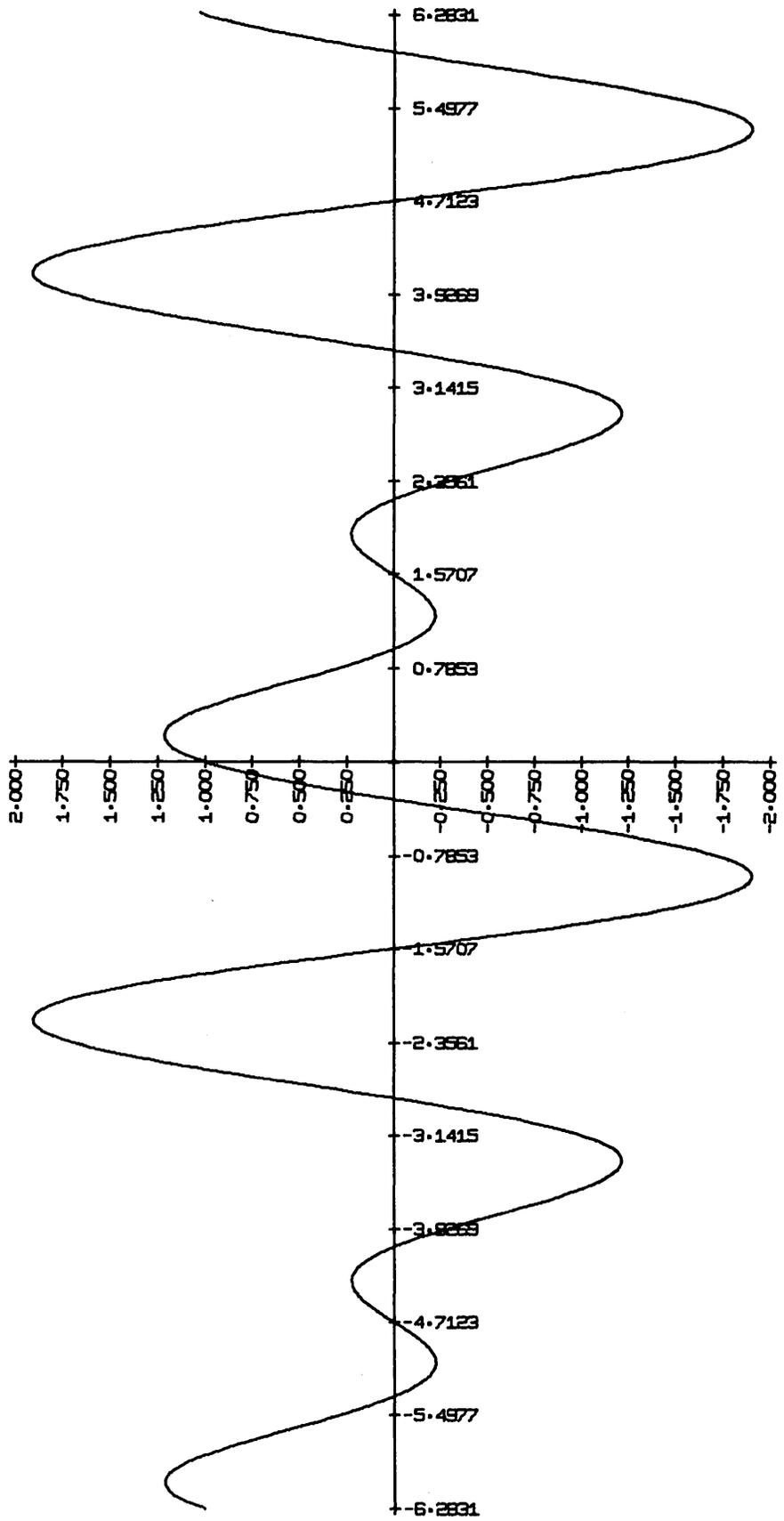
```

// XEQ
PLO05
Y = SIN(2.*X) + COS(3.*X), X = -2.*PI, 2.*PI

END OF TEST

```

Figure 6. Plotter Test 5 Listing (Part 2 of 2)



$Y = \sin(2X) + \cos(3X), X = -2\pi, 2\pi$

FL005

Figure 7. Plotter Test 5 Drawing

**READER'S COMMENT FORM**

IBM 1130/1800 Plotter Subroutines

Form C26-3755-0

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. Comments and suggestions become the property of IBM.

- |  | Yes   | No                       |
|--|---|--------------------------|
| • Does this publication meet your needs?                             | <input type="checkbox"/>                              | <input type="checkbox"/> |
| • Did you find the material:   |   |                          |
| Easy to read and understand?   | <input type="checkbox"/>                              | <input type="checkbox"/> |
| Organized for convenient use?  | <input type="checkbox"/>                              | <input type="checkbox"/> |
| Complete?  | <input type="checkbox"/>                              | <input type="checkbox"/> |
| Well illustrated?  | <input type="checkbox"/>                              | <input type="checkbox"/> |
| Written for your technical level?                                    | <input type="checkbox"/>                              | <input type="checkbox"/> |
| • What is your occupation? _____                                     |   |                          |
| • How do you use this publication?                                   |   |                          |
| As an introduction to the subject? <input type="checkbox"/>          | As an instructor in a class? <input type="checkbox"/> |                          |
| For advanced knowledge of the subject? <input type="checkbox"/>      | As a student in a class? <input type="checkbox"/>     |                          |
| For information about operating procedures? <input type="checkbox"/> | As a reference manual? <input type="checkbox"/>       |                          |

Other \_\_\_\_\_

- Please give specific page and line references with your comments when appropriate. If you wish a reply, be sure to include your name and address.

**COMMENTS**

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

fold

fold

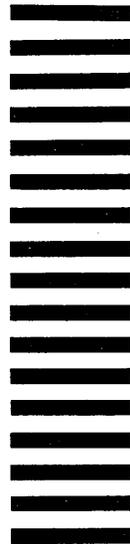
FIRST CLASS  
PERMIT NO. 2078  
SAN JOSE, CALIF.

**BUSINESS REPLY MAIL**  
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY . . .

IBM Corporation  
Monterey & Cottle Rds.  
San Jose, California  
95114

Attention: Programming Publications, Dept. 452



fold

fold

IBM 1130/1800 Printed in U.S.A. C26-3755-0



International Business Machines Corporation  
Data Processing Division  
112 East Post Road, White Plains, N. Y. 10601



International Business Machines Corporation  
Data Processing Division  
112 East Post Road, White Plains, New York