1983/38. Hardcopy plotting on the Geological Survey mini-computer

R.G. Richardson

Abstract

The 36 inch COMPLOT plotter is accessible to users through a set of subroutines within the FORTRAN run-time library. Plot commands are normally put onto disc and transferred to the plotter at a later time.

INTRODUCTION

By calling the plotter subroutines in the appropriate order, a user creates a file of plotter commands on disc (logical unit 10) and those commands are interpreted and sent to the plotter by the system program SPLPLT.

The plotter has felt-tip, ball-point, and ink pens available but only the felt-tip and ink pens are suitable for producing dots. The plotter X-axis is along the length of the roll of paper, the Y-axis is across the width of the paper, and the origin is at the bottom right-hand corner of the paper when viewed from the front. The maximum allowed total length of a plot is 150 inches.

USE OF THE SOFTWARE

The user-called plotting routines send data to logical unit 10. This unit may be assigned to NULL: for testing or to a 256 character indexed disc file. If no file is given, either by assigning a file when the user program is loaded or by a call of PLTOUT, the plotter package will use the filename TEMP.PLT.

Files are transferred from the disc to the plotter by use of the SPLPLT command which has the syntax:

SPLPLT (FNAME) (,FACT)
where ( ) indicates an optional argument
FNAME is the name of the disc file used by the users program and defaults to TEMP.PLT
FACT is the scaling factor to be applied to the plot over and above any scaling initially applied. The default is 1.0.

Users should ensure that unwanted plotfiles are removed from disc as soon as possible. All integer parameters of plot calls should be INTEGER*4.

FUNCTIONS OF THE SUBROUTINES

PLTOUT - may be used to set the output plotfile name
INITAL - sets up certain constants and must only be called once per plot. If more than one plot is required use RSTR(1)
PLOT - drives the pen from its present position to a new position with the pen raised or lowered. It is also used to redefine the present position of the pen.
SCALE - sets and stores scaling information for AXIS and LINE routines.

AXIS - draws an axis with "Tick" marks at one inch intervals and an identification label. Numerals representing the magnitude of the plotted data are drawn at each "Tick" mark.

LINE - drives the pen through an array of points for point-to-point plotting.

SYMBOL - positions the pen and forms the characters to be plotted. Available characters are:

0-9	'	@	?
A-Z	(	+	/
!	)	;	[
"	*	<	\
$	:	,	]
%	=	>	
&	-	.	

NUMBER - positions the pen and forms the numerals that define the magnitude of an internal floating point number.

PWRITE - combines the functions of SYMBOL and NUMBER.

MARKER - draws an event marker at the present pen position.

FACTOR - scales all subsequent plotting subroutine calls.

CIRCLE - causes a circle to be drawn.

DASHL - drives the pen from its present position to a new position by joining the two with a dashed line.

WHERE - used to set a relative datum and interrogate the current scale factor from the user program. Most calls of WHERE are made prior to one or more commands relative to the pen position returned by WHERE and this is accomplished by calling MOVREL to make all following co-ordinates relative to the last call of WHERE. Absolute movement is resumed by a call of MOVABS.

MOVREL - makes co-ordinates relative to the last call of WHERE.

MOVABS - revokes the action of MOVREL.

PENUP - raises the pen.

PENDN - lowers the pen.

RSTR - dumps the plot buffer at the end of a plot. There must be only one call of either RSTR(0) or RSTR(2) in a program run.

SUBROUTINE DESCRIPTIONS

PLTOUT - may be used to set the output plotfile name to other than the default (TEMP.PLT). It must be called before the call of INITAL.

- format PLTOUT (FNAME)
  where FNAME is a variable containing the filename. FNAME may be read in the calling program or a string constant may be used.

INITAL - this routine is used to initialise the plotting routines for the creation of one plot or a group of plots by a given program. The INITAL routine need only be called once and must be called before any other call to the following plotting routines is given.

- a call to INITAL drives the pen down the Y-axis to the right hand side of the paper. This point (X,Y) is defined as (0.,0.).
- format CALL INITAL (IU,IS,IW,0,0,0)

where IU is the plotter logical unit number
IS is the plotter resolution in steps/inch
IW is the plotter width in inches

- the call to INITAL must be as follows:
  CALL INITAL (9,200,36,0,0,0)
  as our system requires the plotter to be on logical unit 9.

PLOT - this routine is most commonly used as it gives direct control of pen movement. PLOT is normally used to drive the pen from its present location to a new position, with the pen either up or down. This new position is specified in X and Y Cartesian co-ordinates, with the co-ordinate values specified in inches. In addition PLOT performs several auxiliary functions.

- format CALL PLOT (X,Y,IPEN)
  where X,Y are the floating point co-ordinates to which the pen is to be moved or the co-ordinates of the present pen position, depending on the magnitude of IPEN. X is the independent variable and Y is the dependent variable.
  IPEN is a signed integer which controls pen status (up or down) and definition of the origin.
  IPEN=0 - there is no plotter movement. The present position of the pen is redefined as X,Y.
  IPEN=1 - plotter moves to (X,Y) with the present condition of the pen (up or down).
  IPEN=2 - pen is dropped and plotter moves to (X,Y).
  IPEN=3 - pen is raised and the plotter moves to (X,Y).
  IPEN=-1, -2, or -3 the action of the pen and plotter is the same as for positive values, but after (X,Y) is reached, X and Y are redefined as (0.,0.)

SCALE - a call to SCALE allows determination of a convenient scale based on the plot dimensions. If the data to be plotted does not have a zero base scale or cannot conveniently be arranged to fit in a zero based scale, this subroutine will reset the scaling as necessary.

- format CALL SCALE (ARRAY,N,S,AMIN,DELTA)
  where ARRAY is a single subscripted array containing the data points to be scanned to determine scale values AMIN and DELTA, such that the first "N" "ARRAY" points can be plotted within a scale length S.
  N indicates the number of points in "ARRAY" to be scaled.
  S denotes the length (in inches) of the axis against which the data is to be plotted (S is a real number).
  AMIN is the smallest value to be plotted on the axis.
  DELTA is the scale increment. For a given scale, DELTA takes on one of the following values in order to produce a convenient scale:

DELTA=K*(10**m) where K=1,2,4,8,10, or 20; and m is the order of magnitude of (AMAX-AMIN)/S. Hence for the first "N" "ARRAY" points,
ARRAY(I)=(ARRAY(I)-AMIN)/DELTA where I=1,N.

AXIS - this subroutine may be used to generate the axes for a graph. It will draw an axis with a linear scale, place tick marks on the axis at one-inch intervals, and label the axis with numbers and letters 0.14 inches high. Any labelling is centred on the particular axis designated. There are normally two calls of this routine per graph, one for the X-axis and one for the Y-axis.

- format CALL AXIS(X,Y,LBL,NC,S,THETA,AMIN,DELTA,NN).
  where X,Y are the co-ordinates (in inches) of the starting point of the axis. The axis lines should be drawn at least half an inch from the side of the plot page. Normally the X and Y axes would be joined at the origin but other starting points may be used.
  LBL is the identification label for the axis. This label is written parallel to and centred on the axis line with letters 0.14 inches high. This label may be a single variable, a single subscripted array, or a Hollerith literal.
  NC specifies the number of characters in LBL which are to be plotted.
  If NC is positive, the scale, tick marks, and label are drawn on the counterclockwise side of the axis (normally used for the Y-axis).
  If NC is negative, the scale, tick marks, and label are drawn on the clockwise side of the axis (normally used for the X-axis).
  S specifies the length of the axis in inches.
  THETA specifies the angle, in degrees from the X-axis, at which the axis is to be drawn.
  THETA=0.0 implies a horizontal axis, reading left to right.
  THETA=90.0 implies a vertical axis, reading bottom to top.
  THETA may be in the range 0.0 to 359.0 degrees.
  AMIN is the minimum value represented on the axis. AMIN is usually determined by a call to SCALE but any value may be supplied by the programmer.
  DELTA is the scale increment value per inch on the axis scale. DELTA is usually calculated by a call to SCALE, but can also be directly supplied by the programmer. The tick marks are labelled at one inch intervals with the following values:

  AMIN,AMIN+DELTA,AMIN+(2*DELTA), etc.

NN determines the number of digits to be drawn after the decimal point on the numeric labelling. If NN is positive, NN digits are drawn after the decimal point.
If NN=0 output ends with the decimal point
NN=1 the decimal point is suppressed.

LINE - the LINE subroutine is used for point-to-point plotting of graphs, functions, charts, etc. The values of X and Y are stored in single subscripted arrays available to the subroutine. The subroutine also has the capability of drawing any one of the fourteen special markers at point intervals specified in the call.

- format CALL LINE (X,Y,N,KODE,ISPACE)
where X is the name of the array containing the X-co-ordinate pairs through which the line is to be drawn.
Y is the name of the array containing the Y-co-ordinate pairs through which the line is to be drawn.
N is the number of data points contained in the X and Y arrays just mentioned. The number of points in each array must be the same.
KODE is the integer equivalent of the marker to be drawn. If the sign of KODE is negative, no line will be drawn connecting the data points, and only the specified symbol will appear at the data points. If the sign of KODE is positive the marker points are joined.
If KODE=0 no marker is drawn.
If KODE=+-1 to +-14 markers as for subroutine MARKER are drawn.
ISPACE specifies the interval for choosing co-ordinate pairs from the X,Y arrays for line plotting and the interval at which the marker selected by KODE is drawn. For example, the symbols will be drawn at 1,1+(1*ISPACE),1+(2*ISPACE), etc. These symbols are drawn centred about the data points.

SYMBOL - this routine is used for writing alphanumerics. It is capable of drawing these characters at any angle and any height within the range of the available plotting surface. The standard symbols are defined earlier.

- format CALL SYMBOL (X,Y,HT,IHOL,THETA,N)
where X,Y specifies the co-ordinates (in inches) of the lower left hand corner of the character string to be drawn. The pen moves to this point in an up position.
HT is the height (in inches) of the characters to be plotted. For best quality, height should be an integer multiple of .035 inches. A "reference area" is defined as the region within which the character is drawn. This reference area is an imaginary 6 x 7 grid. The character to be drawn is centred laterally in this area. The height of the character is a full seven increments, while the width is four increments. Thus

if it is necessary to calculate the area required for each character, it may be done as follows:

H = height of character =HT
W = width of character =(4/7)*HT
S = width of reference area =(6/7)*HT

IHOL is the text to be used for plot annotation. this text is presented left justified in a single subscripted array or an integer variable or as a Hollerith literal.

THETA specifies the angle in degrees measured counterclockwise from the X-axis at which the lettering is to be drawn. For example, for THETA=0.0 the characters are drawn in the +X direction and for THETA=90.0 the characters are drawn in the +Y direction. THETA must be in the range 0.0 to 359.0 degrees.

N specifies the number of characters to be drawn.

NUMBER - this subroutine is used to draw the numerals which represent the value of any real number or variable. NUMBER reformats a floating point number as its fixed point equivalent, and draws the resulting alphanumeric characters. It establishes the starting location and angle, then draws the specified number at an assigned height.

- format CALL NUMBER(X,Y,HT,FLT,THETA,N)
  where X,Y and THETA have the same functions as described in subroutine SYMBOL.
  HT determines the size of the number, as described in subroutine SYMBOL. The sign of HT determines whether the output of the characters begins or ends at (X,Y). If HT is positive, the output begins at (X,Y), and if negative the output ends at (X,Y).
  FLT is the floating point number to be drawn.
  N designates the number of digits after the decimal point when N is positive.
  If N=0 output ends with the decimal point.
  If N=1 the decimal point is suppressed.

PWRITE - this routine combines the functions of SYMBOL and NUMBER and allows the writing of strings of text without having to count the number of characters or perform positioning to align output from calls of SYMBOL and NUMBER.

- format CALL PWRITE (X,Y,HT,THETA,BUFF)
  where X,Y,HT,THETA are as used in symbol and BUFF is a 132 character buffer.
- use of PWRITE
  Text is passed to PWRITE by encoding the data to be written into BUFF using a normal ENCODE statement before the call to PWRITE. The write statement must finish with an &.
  e.g.
```
INUM=6
VAL=67.809
ENCODE (BUFF,100) INUM,VAL
100 FORMAT ('THE',I2,'TH. VALUE IS',F7.3,'&')
CALL PWRITE (X,Y,HT,45.,BUFF)
```

7/14

- N.B. only one line of text consisting of less than 132 characters should be passed to PWRITE.

MARKER - this routine is used to draw any one of fourteen special symbols. The symbol drawn is centred at the present location of the pen.
- format CALL MARKER(I)
  where I has a value from 1 to 14 (see Appendix 2, Part III).

FACTOR - this routine is used to adjust the scale or size of a plot. The default value for scaling is 1.0.
- format CALL FACTOR(FAC)
  where FAC is the factor to multiply all following commands by.
- N.B. care should be used when mixing different values of FAC in one plot.

CIRCLE - this subroutine causes a circle to be drawn.
- format CALL CIRCLE(XORG,YORG,RAD)
  where XORG,YORG are the co-ordinates of the centre of the circle.
  RAD is the radius in inches.

DASHL - this routine is used to drive the pen from its present location to a new location and join the two with a dashed line composed of pen up and pen down segments. The pen is down at the start and end of the line.
- format CALL DASHL (X,Y,PDLEN,PULEN)
  where X and Y are the floating point co-ordinates of the end of the line.
  PDLEN is the length of a pen down segment (in inches).
  PULEN is the length of a pen up segment (in inches).

WHERE - this routine returns the current pen position co-ordinates (in inches) and the scaling factor currently being used in the three arguments.
- format CALL WHERE (X,Y,FACT)
  where X,Y is the present pen position co-ordinate
  FACT is the scaling factor currently being used.

MOVREL - causes following pen positions to be relative to the last call of WHERE.
- format CALL MOVREL.

MOVABS - causes pen positions to resume being specified in absolute co-ordinates.
- format CALL MOVABS.

PENUP - this routine causes the pen to be lifted.
- format CALL PENUP.

PENDN - this routine causes the pen to be lowered.
- format CALL PENDN.

GRAIN - this subroutine changes the effective resolution of the plotter to be IG times the normal step size (for our plotter 200 steps/inch).
- format CALL GRAIN(IG)
  where IG is the reduction in resolution required.
- N.B. this routine will rarely be used.

RSTR - used to output the plot buffer at the end of plotting.
- format CALL RSTR(IARG).
  where if IARG=0 the buffer is dumped leaving the pen where it is.
  IARG=1 the paper is moved to position the pen beyond the plot and the plotter awaits for more commands.

IARG=2 the paper is moved to position the pen beyond the plot and the plotter requires a call of INITAL before accepting more commands.

- N.B. IARG=2 is normally used.

THE SAMPLE PROGRAM

Program DEMPLT (Appendix 1) demonstrates the use of the plotter package. The main body of the program names the plotfile using a call of PLTOUT, initiallises the plotter routines, and produces the plot of Appendix 2 (Part I). The first subroutine, DEMOS produces the output shown in Appendix 2 (Part II). The final subroutine MARKS, produces the standard markers (Appendix 2, Part III) and a sample dashed line.

The plotfile was plotted by SPLPLT TEMPPLT.TMP, but this was found to be too large for copying, so a 20% reduction was produced by using SPLPLT TEMPPLT.TMP, 0.8.

[15 August 1983]

APPENDIX 1

Program DEMPLT

9/1A

```
$TITL  DEMO OF THE COMPLOT PLOTTER
C
C A SAMPLE FOR USERS
      DIMENSION X(101),Y(101),RTX(101),XSQ(101),LB1(6),
     . LB2(4),TEXT(33),LB3(3),LB4(1),IL(13),IN(3)
C SET UP ALL THE HEADINGS REQUIRED
      DATA IL/4HRMIN,4H=   ,4H DR=,4HYMIN,
     . 4H=   ,4H   D,4HY=   ,4HSMIN,4H=   ,
     . 4H DS=,4HXMIN,4H=   ,4H DX=/
      DATA IN/4HSAMP,4HLE P,4HLOT /
      DATA LB1/4H.01 ,4HX SQ,4HUARE,
     . 4H, MA,4HRKER,2H=+/
      DATA LB2/4HROOT,4H X, ,4HMARK,4HER=X/
      DATA LB3/4HX-60,4H,NO ,4HMARK/
      DATA LB4/2H+X/
      CALL PLTOUT('TEMPPLT.TMP')
C SET THE OUTPUT FILE NAME
C INITIALLISE THE PLOTTER - ALWAYS USE LOGICAL UNIT 9
      CALL INITAL(9,200,36,0,0,0)
C SET THE DESIRED SCALE FACTOR WE WANT 1:1
      CALL FACTOR(1.)
C GENERATE THE DATA
      DO 10 I=1,101
      X(I)=I-1
      Y(I)=X(I)-60.
      RTX(I)=SQRT(X(I))
   10 XSQ(I)=(X(I)**2)*.01
C GENERATED THE DATA
C
      CALL SCALE(X,101,5.,XMIN,DX)
      CALL SCALE(Y,101,8.,YMIN,DY)
      CALL SCALE(RTX,101,8.,RMIN,DR)
      CALL SCALE(XSQ,101,8.,SMIN,DS)
C SCALE THE ARRAYS
      CALL PLOT(-1.5,-1.5,0)
C CHANGE ORIGIN
      CALL AXIS(-1.,0.,LB1,22,8.,90.,SMIN,DS,-1)
      CALL AXIS(-1.,0.,LB2,-16,8.,90.,RMIN,DR,1)
      CALL AXIS(0.,0.,LB3,12,8.,90.,YMIN,DY,0)
      CALL AXIS(0.,0.,LB4,-2,5.,0.,XMIN,DX,-1)
C DRAW AND LABEL THE AXES
      CALL LINE(X,Y,101,0,20)
      CALL LINE(X,RTX,101,5,5)
      CALL LINE(X,XSQ,101,-4,10)
C NOW DRAW THE LINES.
C   THE FIRST IS A PLAIN LINE BETWEEN THE DATA POINTS
C   THE SECOND HAS A SYMBOL EVERY 5 DATA POINTS JOINED BY A LINE
C   THE THIRD IS SYMBOLS ONLY
      CALL SYMBOL(.5,2.,.14,IL(1),45.,12)
      CALL SYMBOL(1.25,1.25,.14,IL(4),45.,14)
      CALL SYMBOL(1.75,-1.0,.14,IL(11),0.,12)
      CALL SYMBOL(2.,8.,.21,IN,0.,11)
C PERFORM MORE LABELLING OF PLOTS
      CALL NUMBER(.95,2.45,.14,RMIN,45.,1)
      CALL NUMBER(1.55,3.05,.14,DR,45.,1)
      CALL NUMBER(1.7,1.7,.14,YMIN,45.,1)
      CALL NUMBER(2.5,2.5,.14,DY,45.,1)
      ENCODE(TEXT,100) SMIN,DS
```

```
  100 FORMAT('SMIN=',F4.1,' DS=',F4.1,'&')
      CALL PWRITE(1.65,.85,.14,45.,TEXT)
C USE PWRITE INSTEAD OF THE CALLS TO SYMBOL AND THEN NUMBER
      CALL NUMBER(2.4,-1.0,.14,XMIN,0.,1)
      CALL NUMBER(3.25,-1.,.14,DX,0.,1)
C INSERT THE NUMBERS IN THE LABELS
      CALL MARKS
C CALL THE DEMONSTRATION OF THE MARKER ROUTINE
      CALL DEMO1
C YET ANOTHER EXAMPLE OF GRAPHING
      CALL RSTR(2)
C FINISH PLOT
      STOP
      END
      SUBROUTINE DEMO1
C DRAW A GRAPH OF SIN(X)/X
      DIMENSION Y (101),X(102),KAB(6),MAB(6),NAB(18),LAB(1)
      INTEGER*2 KAB,MAB,NAB,LAB
C SET UP THE CAPTIONS
      DATA KAB/2HY ,2H= ,2HSI,2HN(,2HX),2H/X/
      DATA LAB/1HX/
      DATA MAB/2HSA,2HMP,2HLE,2H P,2HLO,2HT /
      DATA NAB/2HGR,2HAP,2HH ,2HOF,2H S,2HIN,2H(X,2H)/,
     &2HX ,2HFR,2HOM,2H .,2H01,2H T,2HO ,2H10,2H.0,2H1 /
      DELX=.1
      XMIN=.01
      XMAX=10.01
      NTOP=(XMAX-XMIN)/DELX+.5
      NTOP=NTOP+1
      X(1)=XMIN
C GENERATE THE VALUES TO BE PLOTTED
      DO 7040 I=1,NTOP
      Y(I)=SIN(X(I))/X(I)
      X(I+1)=X(I)+DELX
 7040 CONTINUE
C
C NOW MOVE THE PLOTTER TO THE RIGHT PLACE AND RESET THE ORIGIN
      CALL PLOT(0.,9.3,-3)
C GET THE MAXIMUM AND MARKER INTERVAL FOR THE AXES
      CALL SCALE(X,NTOP,6.,XMIN,DX)
      CALL SCALE(Y,NTOP,8.,YMIN,DY)
C DRAW THE Y AXIS
      CALL PLOT(-1.5,-1.,0)
      CALL PLOT(0.,0.,3)
      CALL PLOT(0.,8.,2)
      YY=0.
      YFLT=YMIN
C NOW LABEL THE Y AXIS
      DO 7110 I=1,9
      CALL NUMBER(-.2,YY,-.14,YFLT,0.,1)
      CALL PLOT(-.1,YY,3)
      CALL PLOT(0.0,YY,2)
      YY=YY+1
      YFLT=YFLT+DY
 7110 CONTINUE
C
C NOW THE CAPTION FOR THE Y AXIS
      CALL SYMBOL(-.75,3.25,.14,KAB,90.,12)
C NOW THE X AXIS - USE THE AXIS ROUTINE THIS TIME
      CALL AXIS(0.,2.,LAB,-1,6.,0.,XMIN,DX,1)
```

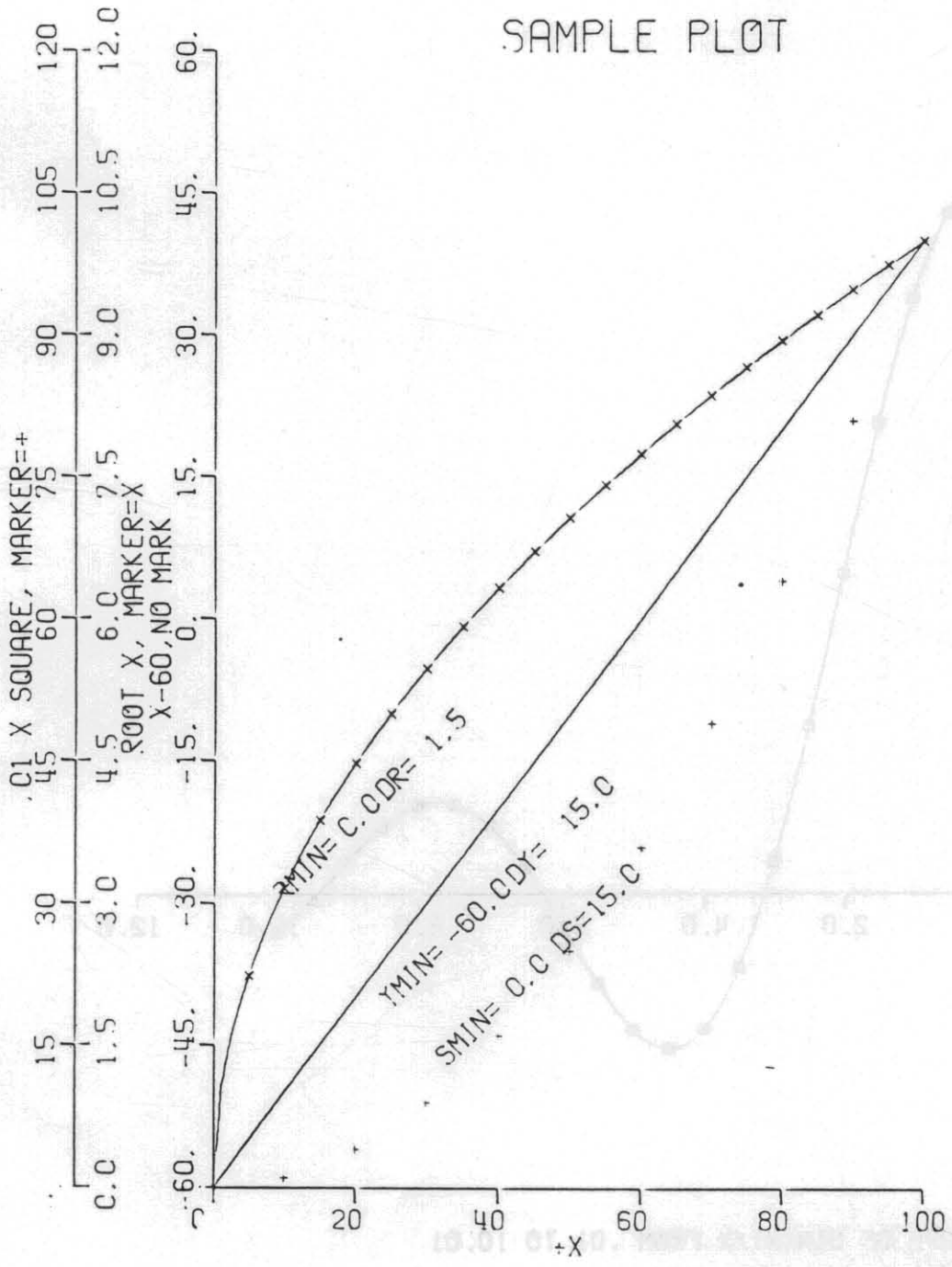
11/14

```
C NOW THE TITLE OF THE PLOT
      CALL SYMBOL(0.,8.5,.21,MAB,0.,12)
C DRAW THE LINE WHICH IS, AFTER ALL, THE POINT OF THE EXERCISE
C FIRST DRAW A SYMBOL EVERY 5 DATA POINTS THEN . . .
      CALL LINE(X,Y,NTOP,-1,5)
C DRAW A LINE THROUGH ALL THE DATA POINTS
      CALL LINE(X,Y,NTOP,0,1)
C AND ANOTHER CAPTION
      CALL SYMBOL(0.,-.5,.14,NAB,0.,35)
      RETURN
      END
      SUBROUTINE MARKS
C TO ILLUSTRATE THE USE OF MARKER
      REAL TITLE(4)
      DATA TITLE/4HCALL,4H MAR,4HKER(,4HI)  /
C THE TITLE OUTPUT ABOVE THE SYMBOLS
      CALL PLOT(0.,9.6,-3)
C MOVE TO 0.,9.  AND REDEFINE AS 0.,0.
      CALL SYMBOL(2.5,5.4,.14,TITLE,270.,14)
C WRITE THE TITLE STARTING AT 2.5,10
C NOW DRAW THE MARKERS
      DO 10 J=1,14
C ONLY 14 MARKERS
      YY=FLOAT(J)*0.6
C DRAW FROM RIGHT TO LEFT
      I=15-J
C SO NEED TO GET THE MARKERS IN THE REVERSE ORDER TO THE LOOP
      XK=I
C AND FLOAT THIS FOR POSITIONING CALLS
      CALL NUMBER(1.5,YY,.14,XK,270.,-1)
C WRITE THE NUMBER WITHOUT ANY DECIMAL POINT
      CALL PLOT(0.4,YY-0.07,3)
C PICK THE PEN UP AND MOVE TO THE PLACE TO DRAW THE MARKER
      CALL FACTOR(2.5)
C ENLARGE SYMBOLS BUT STILL KEEP A MULTIPLE OF THE BASIC SIZE
      CALL MARKER(I)
C DRAW THE MARKER
      CALL FACTOR(1.0)
   10 CONTINUE
C END OF MARKER DRAWING LOOP
      CALL PLOT(0.,0.,3)
C PEN UP AND MOVE TO BOTTOM TO START DASHED LINE
      CALL MARKER(5)
      CALL DASHL(0.,9.0,0.2,0.1)
C DRAW THE DASHED LINE
      CALL MARKER(5)
C AND MARK THE OTHER END OF THE LINE
      RETURN
      END
```

APPENDIX 2

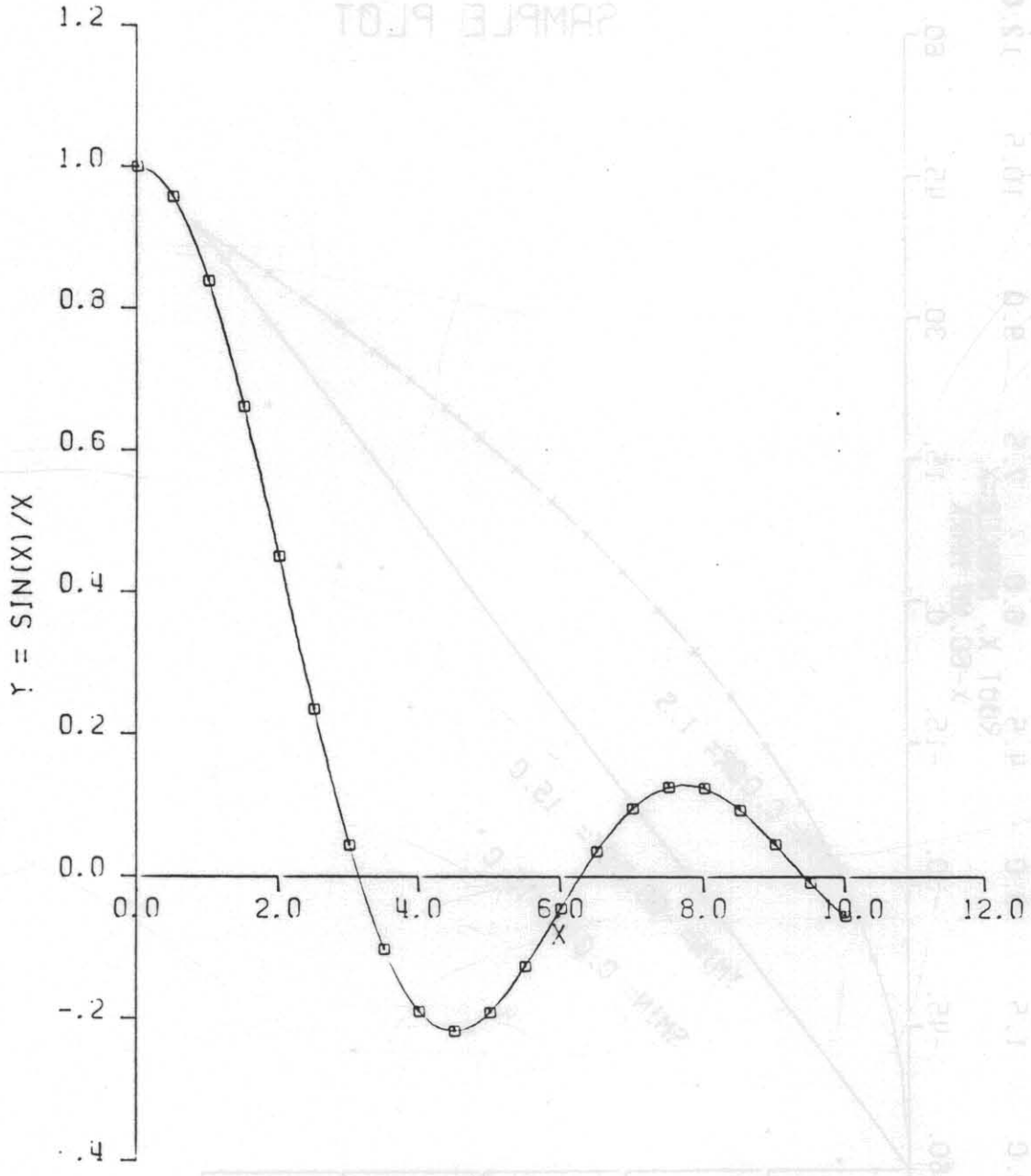
Sample plots

PART I



PART II

SAMPLE PLOT



GRAPH OF SIN(X)/X FROM .01 TO 10.01

5 cm

PART III