PFC/RR-84-14

DOE/ET/51013-132
UC20

GRAPHICS PACKAGE
VERSION 4

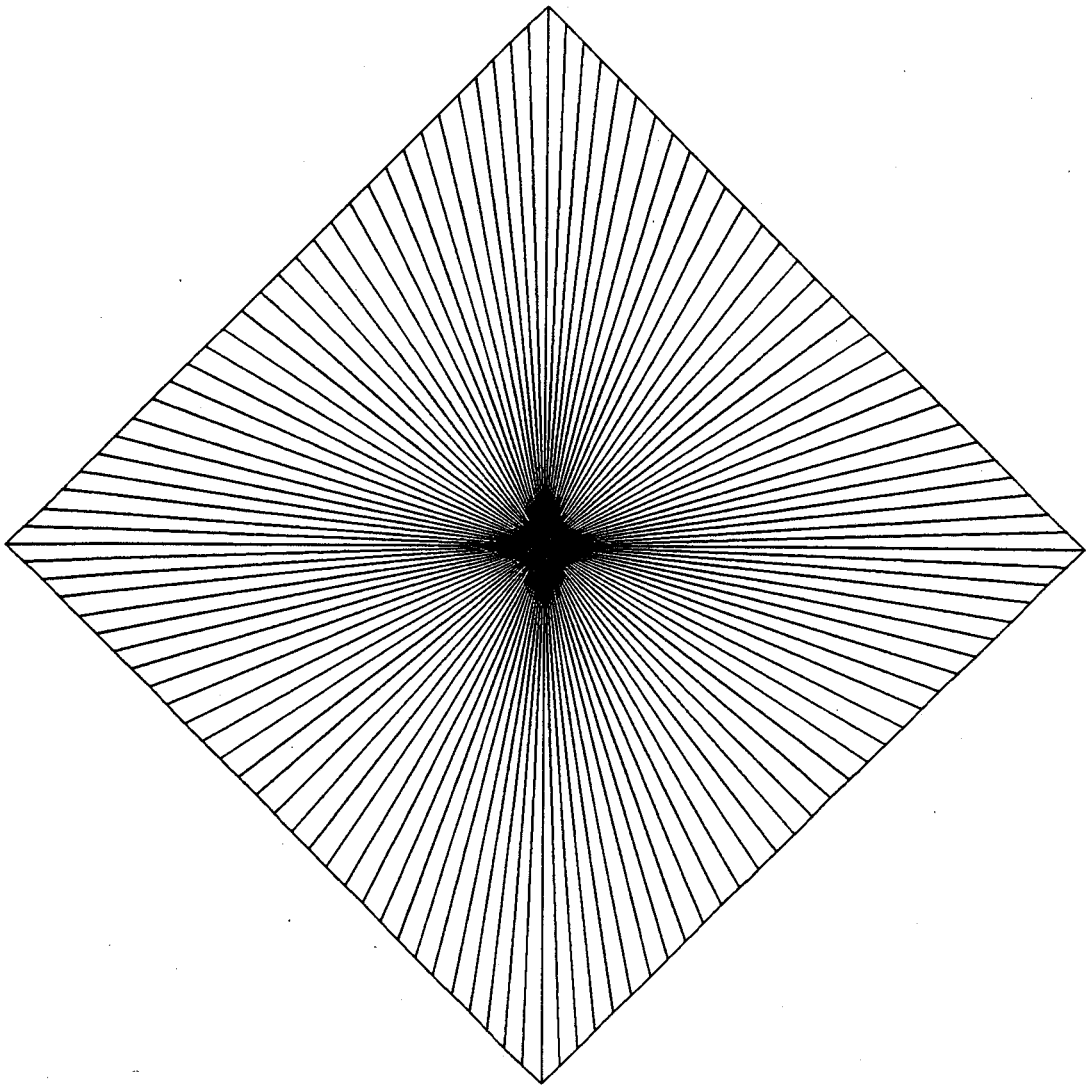
User Reference Manual

James D. Sullivan

Plasma Fusion Center
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

December 1984

Test Pattern



Graphics Package version 4.1

TABLE OF CONTENTS

CHAPTER 1

GENERAL INFORMATION

This manual provides the technical information that a FORTRAN programmer needs to write graphics computer programs. The purpose of this graphics package is to provide an internally consistent, easy-to-use set of graphic subprograms for making publication style plots and illustrations on a variety of display devices.

The descriptions in this chapter give an introduction to simple plotting. The intent has been to present an overview in the hope that, with an idea of what is going on, more general methods will be easier to learn.

1.1 APPLICATION PROGRAM STRUCTURE

Only a minimum knowledge of computer graphics is required to use this graphics package. Most of the parameters needed to define and produce an illustration, such as plotting boundaries, scaling type, etc., are initialized to default values although they may be changed by calling various subprograms.

In using the graphics package, certain operations should be done in a given order. The subroutine BEGPLT must be used to initiate the package and the subroutine ENDPLT to terminate. The subroutine NUPAGE should be used to define each new page. In general, the graph frame is generated by a call to the subroutine GRFRAM usually preceeded by one or two calls to some set-up subroutine (there are several). Plotting a curve in this frame is accomplished by calling some subroutine with actual data values, not device coordinates. A curve is considered either as an open series of discrete points (possibly connected) or as a finite array of points; in other words, a subroutine call may be generated for each point or for each array; occasionally an external function may be specified. Subprograms are provided to draw plane curves, projections of space curves, and projections and isograms (contour maps) of two dimensional surfaces; there is no direct provision for representing three dimensional objects in this version of the graphics package. A typical procedure for producing a plot of a plane curve is:

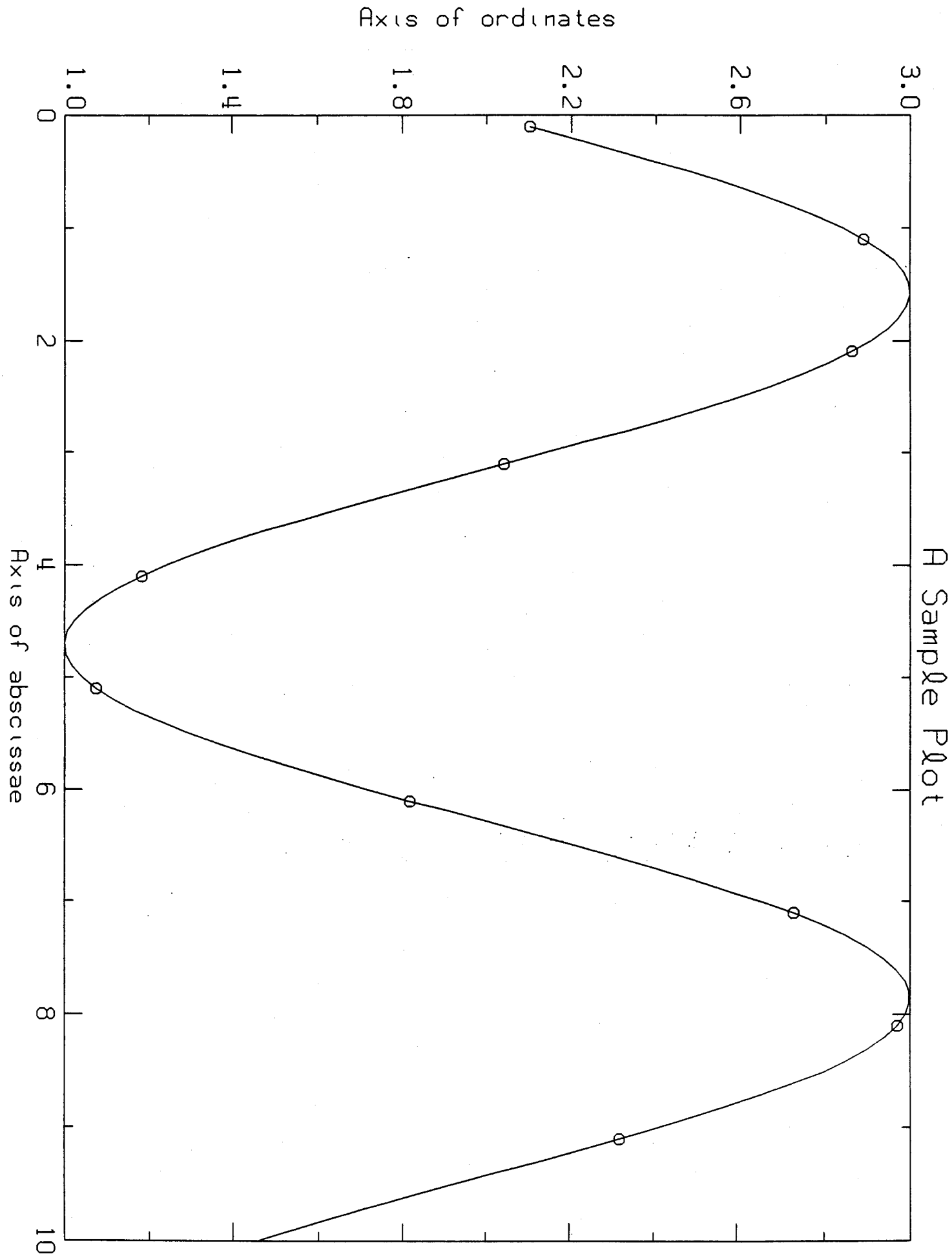
1)	Open the graphics package	CALL BEGPLT (arguments)
2)	Define a page	CALL NUPAGE (arguments)
3)	Set the domain and range	CALL BOUNDS (arguments)
4)	Draw the frame	CALL GRFRAM (arguments)
5)	Plot a curve of data	CALL LINRAY (arguments)
6)	Close the graphics package	CALL ENDPLT (arguments)

Usually, steps 1) and 6) are only done once a program no matter how many pages of graphics are made.

1.2 A SAMPLE PLOTTING PROGRAM

To illustrate this procedure, the sample program that produced the graph in Figure 1 follows:

```
C ------------------------------------------------------------------
C
C                  A SAMPLE PLOTTING PROGRAM
C
C
C                 GRAPHICS PACKAGE version 4.1
C
C
C                      VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP0
C
      CHARACTER*63 PORTID                    !computer port identification
      CHARACTER*8 DEVTYP                     !type of display device
      REAL X(100), Y(100)                    !reserve space for 100 data values
C INITIALIZE THE ARRAYS WITH SOME DATA
      DO 1 I=1,100
      X(I) = I/10.0
    1 Y(I) = 2.0 + SIN(X(I))
C ASSUME THIS PROGRAM IS BEING RUN INTERACTIVELY FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 0'
      TYPE 998
      ACCEPT 999, PORTID                     !accept the port identifier
      TYPE 997
      ACCEPT 999, DEVTYP                     !accept the device type
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C DEFINE A PAGE                              !define a new page defaulted to
      CALL NUPAGE                            !the maximum size for the device
C SET THE DOMAIN AND RANGE
C               !define the domain as [0,10] with scale flag, 0, linear axis
C               !define the range as [1,3] with scale flag, 0, linear axis
      CALL BOUNDS ( 0.0, 10.0, 0, 1.0, 3.0, 0)
C DRAW THE FRAME                        !the texts are character variables
      CALL GRFRAM ( 'Axis of abscissae', 'Axis of ordinates',
     1   'A Sample Plot')
C PLOT A CURVE OF DATA
      CALL LINRAY ( X, Y, 100)                !plot 100 (X,Y) pairs
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT                     !by default, pause and reset the terminal
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```


Axis of ordinates
1.0
1.4
1.8
2.2
2.6
3.0
0
2
4
6
8
10
Axis of abscissae
A Sample Plot

1.3 PLANNING A GRAPH

The idea of a page is basic to this graphics package; all output occurs on pages. A page is associated with an actual device at execution time; this gives device independence to the user's programs so that the device may be changed without any program modification. For hardcopy devices a page is best thought of as a piece of graph paper and for video terminals as the full screen.

A complete illustration is considered to be a page consisting of one or more regions with a graph frame enclosing one or more panels in each region. That is, an illustration is a set of regions; each region or graph is a mosaic of subgraphs or panels. A graph frame is an enclosed rectangular plotting space with linear, logarithmic, or time axes, with tic marks all around, and with the axes labeled and annotated; one or more curves may be drawn in each panel of a graph frame. It is generally helpful to sketch the desired illustration on paper before structuring the subprogram calls to generate it. The quick plotting subroutines (PICTUR, PICTEXT, and PICT3D) frequently give adequate illustrations without any layout effort.

1.4 GRAPH NOMENCLATURE

The nomenclature for describing an illustration is summarized in Figures 2 and 3 wherein curves of data have been omitted. Tic marks, which bound intervals of an axis, are classified as either major or minor; only major tic marks may have a tic label and be extended into a grid mesh. Remote exponents are added to the end of the appropriate axis label if needed. In regions whose height is less than the full height of the page, the title and comment line for the graph are suppressed. A region or page may consist of only text. On devices which make hard copy, a page stamp appears along the right hand border of the illustration; for those devices which have a roll of paper, a cutline is drawn to delineate the pages. Margins are specified from the lower left corner of a page with the graph frame (area in which curves may be plotted) measured from there; the borders are what is left. The program used to draw the graph nomenclature figures is listed in appendix B.

1.5 COORDINATE TRANSFORMATIONS

A graph is designed in linear object space (consider a piece of standard graph paper) but the data are normally defined in subject space (i.e., user coordinates in engineering units). Two or three dimensional subject space coordinates are converted to the corresponding linear space coordinates (DATCON) which are transformed into two dimensional object space coordinates (GLTRAN) which are finally transformed into actual device coordinates (PLOT). Various set-up routines define each transformation. Transformation amoung the coordinate systems is normally transparent to the user. Even when special map projections or coordinate systems are used only the appropriate set up subprograms need be called.

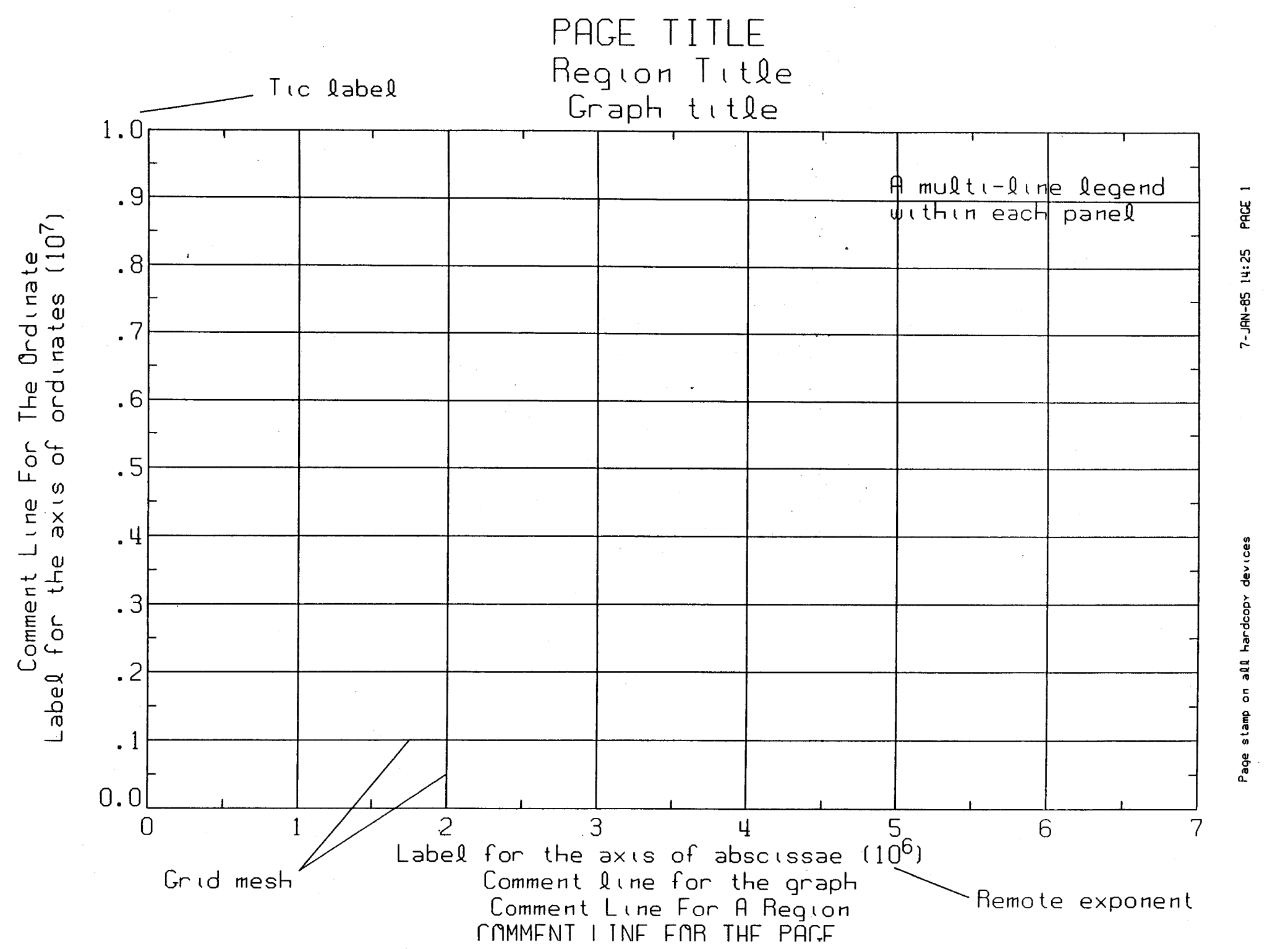
1.5.1 Missing Data

One real number is reserved to indicate a missing subject space coordinate; this number is called "fill". This value is generally ignored and the corresponding point is skipped; a gap will appear in any curve. On a VAX, the largest positive real number is used for fill (1.7014117E+38, 'FFFF7FFF'X, or -32769).

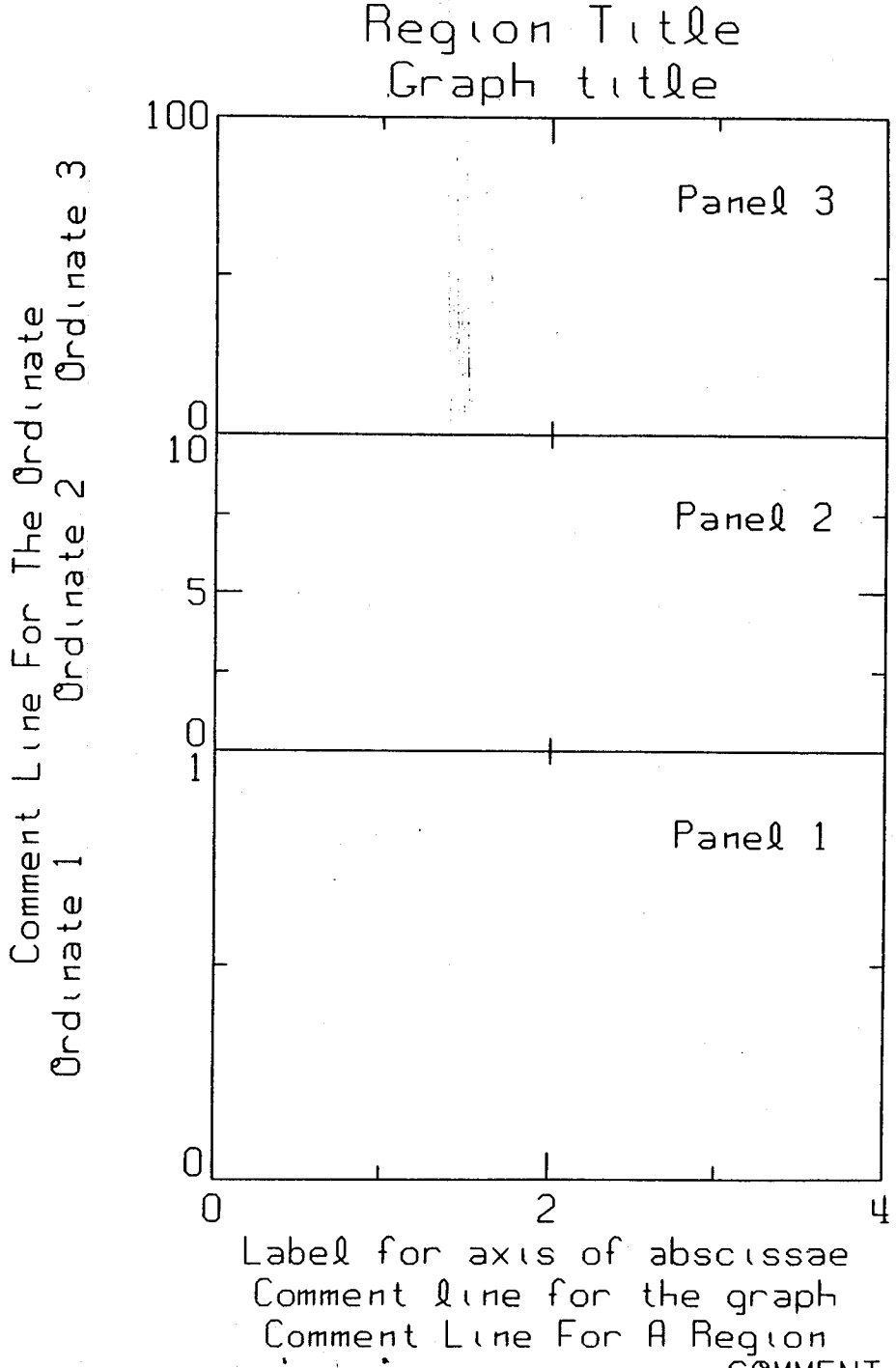
1.6 OPERATING CONSIDERATIONS

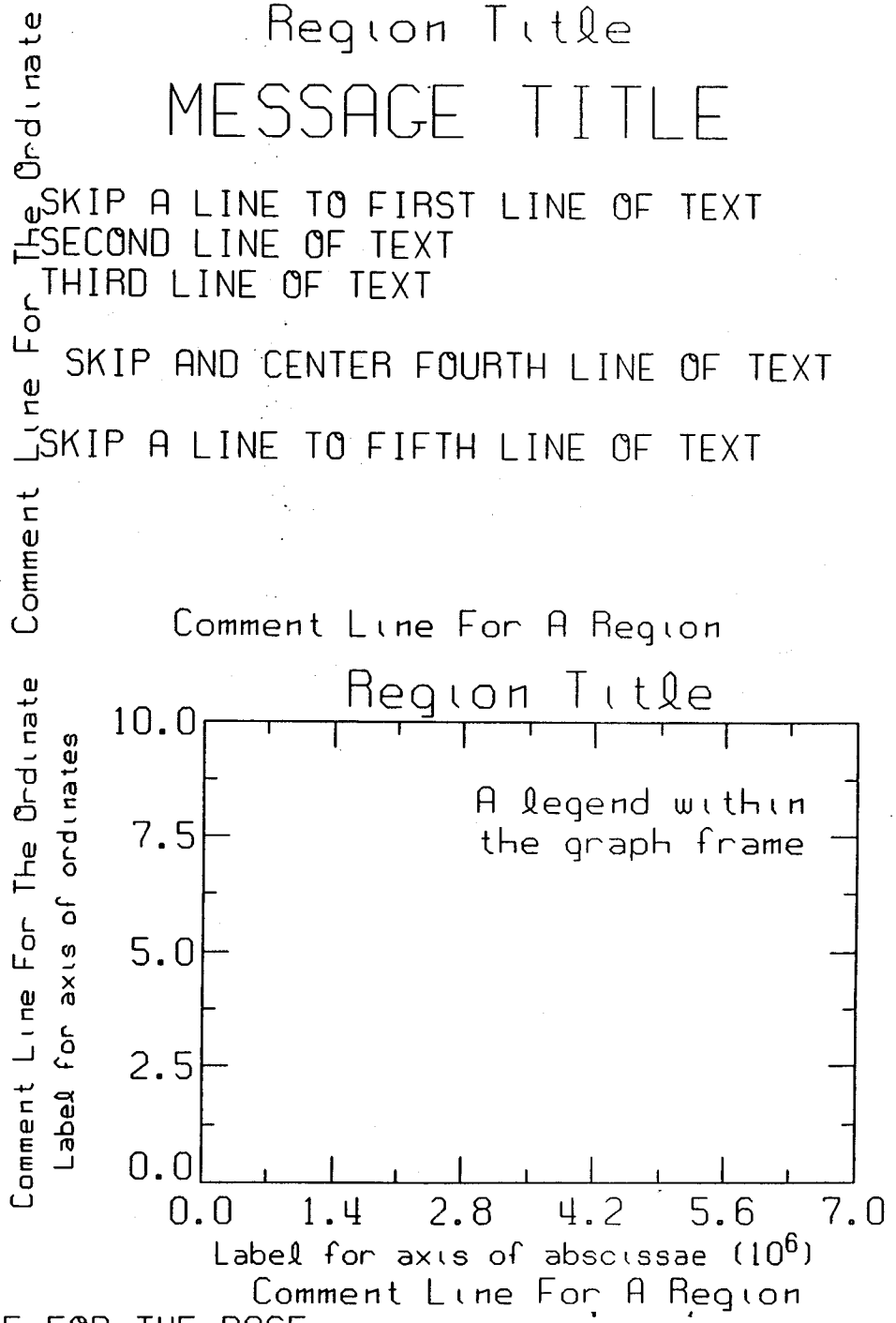
An understanding of some of the operating characteristics of each type of display device is desirable when planning an application program. For example, the quality of appearence of a finished plot on an electromechanical plotter depends on the combination of pen, paper, and ink used.

On video display terminals the axis of abscisae is horizontal and the axis of ordinates is vertical. In order to preserve this aspect ratio as far as possible, on devices which make hard copy a page is considered 11"x8.5" not 8.5"x11"; a reversal of normal conventions. When rotated the hardcopy page is 8.5"x11". On laser printers as prepunched paper is assumed, the margins are automatically adjusted and the page size


PAGE TITLE
Region Title
Graph title
Tic label
A multi-line legend
within each panel
Comment Line For The Ordinate
Label for the axis of ordinates (10^7)
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
0.0
0
1
2
3
4
5
6
7
Label for the axis of abscissae (10^6)
Grid mesh
Comment line for the graph
Comment Line For A Region
COMMENT LINE FOR THE PAGE
Remote exponent
7-JAN-85 14:25 PAGE 1
Page stamp on all hardcopy devices

PAGE TITLE ON A MULTI-REGION PAGE





7-JAN-85 14:25 PAGE 2

Page stamp on all hardcopy devices

COMMENT LINE FOR THE PAGE

correspondingly reduced. On those devices which have a long roll of paper other physical sizes may be specified for use (in NUPAGE).

1.7 ERROR PROCEDURE AND PROGRAMMING PRAXIS

Errors may occur in user programs. The following are a few recomendations which, if used, may reduce their occurence.

1) Take care to match the name of a subprogram with the correct number, position, and type of arguments. Except for character type variables, all variables or array elements are assumed 4 bytes in length (the FORTRAN default).

2) Be careful of the order of subscripts when using multi-dimensioned arrays for storing data values. This is especially important when processing only part of an array. In FORTRAN the first subscript varies most rapidly.

3) Never use a graphic package subprogram (or reserved) name for a user subprogram or labeled common block name. A list of all the names used in the graphics package is given in appendix C.

4) Limit the output to one plot of reasonable length in the debugging stage; perhaps just the frame at first.

5) Write out the data to be plotted in the order in which they will be plotted.

6) Be particularly careful when writing ASCII text. In general, the annotation routines use character variables for input arguments whereas all routines internal or basic to the package use byte variables. Because VAX/VMS FORTRAN handles these two variable types so differently errors can easily occur. However, quoted text is handled correctly for both cases.

CHAPTER 2

SUBPROGRAMS 1

This chapter contains descriptions of all of the graphics package subprograms intended for the casual user (the remaining subprograms are described in a later chapter). The subprograms available for use vary from the very general (i.e., drawing the entire graph) to the very basic (i.e., setting a single parameter or drawing only certain parts of the graph). Any of the subprograms may be combined to make illustrations in a manner which is most convenient.

FORTRAN naming conventions are used throughout and all variables are four bytes long. Character type variables are the only exception. In the argument lists of subprogram calls, variables in capital letters must be supplied while those in lower case may be omitted in place. Normally, the value of an omitted argument is unchanged from the value last established. If all remaining arguments may be omitted the argument list may be truncated after the last argument used. Square brakets indicate truncation points of argument lists; curly brakets indicate arguments intended for internal use only.

Variable and subprogram names are generally mnemonic. The following conventions apply to the names of all variables:

L	as prefix: LOGICAL VARIABLE
P	as affix: PAGE
RAY	as suffix: FORTRAN ARRAY
S	as suffix: SUBJECT SPACE (USER COORDINATES)
TXT	as affix: CHARACTER TYPE VARIABLE (may never be omitted)
VP	as prefix: VIRTUAL PAGE (REGION)
X	as affix: ABSCISSA
Y	as affix: ORDINATE

In the names of subprograms there is an additional convention:

SET	as prefix: SET PARAMETERS (INPUT ROUTINE)
	as suffix: PARAMETER SETTINGS (OUTPUT ROUTINE)

In the explanation of the arguments an I indicates a variable used for input, an O one used for output, and I/O one for both input and ouput; all variables are assumed to be input unless otherwise indicated. Angles are always in degrees.

The descriptions of the subprograms are arranged alphabetically within functional groups. In general subprograms may be called in any order within a grouping but those in one group should be called before those in succeeding groups.

2.1 OPEN/CLOSE

These subprograms must be called to produce any graphics output. The first graphics package subprogram called should be BEGPLT and the last ENDPLT.

2.1.1 BEGPLT

Purpose: To initiate plotting and to set the computer port and device type; also to close any previous BEGPLT call.

CALL BEGPLT (PORTID, DEVTYP, ier)

PORTID I CHARACTER VARIABLE TERMINAL PORT

DEVTYP	I	CHARACTER VARIABLE DEVICE TYPE
IER	O	ERROR FLAG 0 = success 1 = unknown device type 2 = unsupported device type

A call to ENDPLT with no pause and no reset is implicit. On error a null device type is assumed.

2.1.2 ENDPLT

Purpose: To terminate graphic output and close files.

CALL ENDPLT (lreset, inword)

LRESET	I	LOGICAL FLAG TO RESET VIDEO TERMINALS TO ASCII MODE (default .TRUE.)
INWORD	O	FOUR BYTE WORD FROM INTERACTIVE TERMINAL = 0 FOR NON TERMINAL DEVICE TYPES or if LRESET is .FALSE. or if the page has already been ended. = ' ' FOR <CR> WITHOUT ANY ENTRY

A call to ENDPAG with a pause if LRESET is .TRUE. is implicit.

2.1.3 GPINIT

Purpose: To facilitate opening the plot package during interactive sessions.

CALL GPINIT (MSGTXT)
MSGTXT MESSAGE DISPLAYED BEFORE PROMPTING FOR THE PORT ID AND DEVICE TYPE

BEGPLT is explicitly called by GPINIT after the port and device type are accepted.

2.2 CHANGING GLOBAL PARAMETERS

These subprograms are seldom needed as all the parameters are initialized to default values by BEGPLT but they may be called at any time to change the global characteristics of the graphics package.

2.2.1 DEFSYM

Purpose: To allow redefinition of characters in the symbol dictionary.

CALL DEFSYM (IFONT, IREF, BSTR)

IFONT	FONT NUMBER (1 or 2)
IREF	DECIMAL CODE OF CHARACTER (0 to 127)
BSTR	BYTE STRING OF VECTORS FOR DRAWING THE CHARACTER

All arguments are input.

2.2.1.1 Character Design - Each character is defined by a byte string of vectors on a grid which is scaled by the character height. One grid unit equals one seventh of the character height. A byte string :

'X0Y0X1Y1X2Y2...XnYn'

where XiYi denotes one byte pair of X and Y vectors; each Xi or Yi is one hex digit. Byte 0 gives the character offset in excess 8 notation; byte 1 gives the center of the character when used as a marker; a byte equal to F0 gives a pen up command; and a byte is the last byte if Xn = 'F' and Yn > '0'. The advance to the next character is Yn.

Regular characters (31< IREF < 127) are defined on a 4X7 grid centered in a 6X7 matrix, thus there are 2 units between adjacent characters; markers are special centered characters with 0<= IREF < 32 and are normally on a 4x4 grid; lines are spaced 11 units apart; sub/superscripts are 5.5 units high and shifted -3 and +4.5 units respectively.

For example, to define a right arrow as a new marker:

```
CALL DEFSYM ( 1, 22, 'AA2202423342314222F4')
```

2.2.2 SETBAS

Purpose: To set conversion base, machine rounding, and fill values.

```
CALL SETBAS ( base, round, fill)
```

BASE	NUMBER BASE (DEFAULT 10)
ROUND	MACHINE ROUNDING FACTOR (DEFAULT 1.000005)
FILL	VALUE FOR MISSING SUBJECT SPACE COORDINATE (DEFAULT 'FFFF7FFF'X)

This routine may be used to cause numbers to be output in octal or hexadecimal and to change the value reserved for missing data. Default values are reestablished on each call to SETBAS.

2.2.3 SETPAG

Purpose: To set some general page values.

```
CALL SETPAG ( chsc, npag, nclab, nxax, nyax)
```

CHSC	SCALE DIVISOR FOR CHARACTER HEIGHT (DEFAULT 56.0)
NPAG	NUMBER OF THE NEXT PAGE (DEFAULT 1)
NCLAB	NUMBER OF CHARACTERS IN A TIC LABEL (DEFAULT 5)
NXAX	NUMBER OF X AXES, 1 or 2 (DEFAULT 1)
NYAX	NUMBER OF Y AXES, 1 or 2 (DEFAULT 1)

This subprogram sets the character height scale for subsequent NUPAGE calls. Default values are reestablished on each call to SETPAG.

2.2.4 SETSCAL

Purpose: To change the break points between linear and logarithmic axes for the automatic axis setting routines.

```
CALL SETSCAL ( fzero, rlog, nints)
```

FZERO	BREAK VALUE BETWEEN FIXED ZERO AND NON ZERO LINEA BOUNDS (DEFAULT 8)
RLOG	BREAK VALUE BETWEEN LINEAR AND LOGARITHMIC AXES (DEFAULT 3000)
NINTS	EXTREME NUMBER OF MAJOR INTERVALS (DEFAULT 10)

Default values are reestablished on each call to SETSCAL.

2.2.5 SETUSR

Purpose: To redefine the page stamp text.

```
CALL SETUSR ( USRTXT, ilen)
```

USRTXT	NEW TEXT CHARACTER VARIABLE STRING FOR PAGE STAMP <31 CHARACTERS PRESERVES THE INSTITUTION IDENTIFIER <60 CHARACTERS PRESERVES THE DATE, TIME AND PAGE NUMBER
ILEN	LENGTH OF PAGE STAMP (MAXIMUM 84; DEFAULT 84)

The institution identifier, date, time and page number are reestablished on a call to SETUSR with a short string.

2.3 QUICK PLOTTING

Each of these three subprograms may be used to make a complete illustration; a call to NUPAGE is implicit in each.

2.3.1 PICTEXT

Purpose: To write a page of text; to make textual viewgraphs.

```
CALL PICTEXT ( TXTRAY, NLINES, chfrac)
```

```
 TXTRAY  ARRAY OF CHARACTER VARIABLE STRINGS
 NLINES  NUMBER OF CHARACTER VARIABLE STRINGS
 CHFRAC  ARRAY OF FRACTIONAL CHARACTER HEIGHTS (of default value)
         IF NEGATIVE CENTER CHARACTER STRING ON LINE
```

All arguments are input only. The lines are numbered top down; the character size if CHFRAC is omitted is twice the page default size.

2.3.2 PICTUR

Purpose: To provide a quick plotting routine for up to 5 curves.

```
 CALL PICTUR ( XTXT, YTXT, TXTL, XR1, YR1, N1[, inteq1[, XR2, YR2,
               N2[, inteq2[, ... [, XR5, YR5, N5, inteq5]]]]]]]])
```

```
 XTXT    CHARACTER VARIABLE ABSCISSA TITLE
 YTXT    CHARACTER VARIABLE ORDINATE TITLE
 TXTL    CHARACTER VARIABLE GRAPH TITLE
 XR1     ARRAY OF ABSCISSA VALUES FOR FIRST CURVE
 YR2     ARRAY OF ORDINATE VALUES FOR FIRST CURVE
 N1      NUMBER OF POINTS IN THE FIRST CURVE
 INTEQ1  INTEGER EQUIVALENT FOR MARKER OF THE FIRST CURVE (default 1)
 XR2     ARRAY OF ABSCISSA VALUES FOR SECOND CURVE
 .
 .
 .
 INTEQ5  INTEGER EQUIVALENT FOR MARKER OF THE FIFTH CURVE (default 5)
```

All arguments are input only; the axes are autoscaled.

2.3.3 PICT3D

Purpose: To represent a two dimensional surface as either a projection or an isogram.

```
 CALL PICT3D ( XTXT, YTXT, XTXT, TXTL, XR, YR, FR, NX, ny,
               theta, phi, mode)
```

```
 XTXT    CHARACTER VARIABLE X-AXIS TITLE
 YTXT    CHARACTER VARIABLE Y-AXIS TITLE
 ZTXT    CHARACTER VARIABLE Z-AXIS TITLE
 TXTL    CHARACTER VARIABLE GRAPH TITLE
 XR      ARRAY OF VALUES FOR X-AXIS
 YR      ARRAY OF VALUES FOR Y-AXIS
 FR      TWO DIMENSIONAL ARRAY OF VALUES FOR Z-AXIS ( F(x,y))
 NX      NUMBER OF POINTS IN THE X-AXIS ARRAY and the row dimension
         of FR
 NY      NUMBER OF POINTS IN THE Y-AXIS ARRAY (default NX)
 THETA   POLAR VIEWING ANGLE IN DEGREES (default 40.0)
         if 0.0 an isogram is drawn.
 PHI     AZIMUTHAL VIEWING ANGLE IN DEGREES (default -45.0)
```

MODE	FLAG WORD CONTROLLING HIDDEN LINE REMOVAL (default 3) =1 plot all points =3 plot only visible points (remove hidden lines)

All arguments are input only; the axes are autoscaled.

2.4 PAGE DEFINITION

The coordinates defined here are the object space system; it is these coordinates that the basic package uses.

2.4.1 ENDPAG

Purpose: To terminate a page— draw cutlines, write a page stamp, flush the plot buffers, and pause.

CALL ENDPAG (lpause, inword)

LPAUSE	I	LOGICAL PAUSE FLAG (default .TRUE.) there is no pause for hardcopy or null devices if there is a pause then the prompt appears Type <CR> to continue
INWORD	O	FOUR BYTE WORD FROM INTERACTIVE TERMINAL = 0 FOR NON TERMINAL DEVICE TYPES or if LPAUSE is .FALSE. or if the page has already been ended. = ' ' FOR <CR> WITHOUT ANY ENTRY

This subprogram is seldom used as it is implicitly called by NUPAGE and ENDPLT. There is no action if the page is already ended.

2.4.2 NUPAGE

Purpose: To start a new page after finishing the old.

CALL NUPAGE (pw, ph, gl, gh, plm, pbm, lrot, ch, tl)

PW	PAGE WIDTH
PH	PAGE HEIGHT
GL	GRAPH LENGTH
GH	GRAPH HEIGHT
PLM	PAGE LEFT MARGIN
PBM	PAGE BOTTOM MARGIN
LROT	LOGICAL ROTATION FLAG (DEFAULT .FALSE.)
CH	CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE) (DEFAULT ESTABLISHED IN SETPAG)
TL	TIC LENGTH (DEFAULT CH)

All arguments are input to this subprogram. A call to NUPAGE without PW and PH sets these to the device range. In any case PW and PH define page coordinates. The margins are by default based on the CH to allow enough space for axis and tic labels. The graph frame size is set last. Subject space is only definable within this graph frame. Pen number 1 is reselected and the actual margins reset. A call to ENDPAG is implicit.

2.5 PAGE ANNOTATION

Space is not allocated for these text lines unless needed.

2.5.1 PAGLYN

Purpose: Draw a line of text along the bottom of a page.

CALL PAGLYN (PCLTXT, cfact)

PCLTXT	CHARACTER VARIABLE PAGE COMMENT LINE
CFACT	CHARACTER HEIGHT MULTIPLIER (default 1.0)

2.5.2 PAGTITL

Purpose: Draw a page title

CALL PAGTITL (PTXTL, cfact)

PTXTL	CHARACTER VARIABLE PAGE TITLE
CFACT	CHARACTER HEIGHT MULTIPLIER (default 1.5)

2.6 REGION DEFINITION

A page automatically has one region defined equal to the full page. Thus every subprogram which works within a region may be used without calling any routine in this section. The routines in this section are used to define multiple regions on a page. Only SETREG is normally used.

2.6.1 REGION

Purpose: To define a virtual page in a region.

CALL REGION (vpw,vph,gl,gh,vplm,vpbm,lrot,ch,tl)

VPW	VIRTUAL PAGE WIDTH
VPH	VIRTUAL PAGE HEIGHT
GL	GRAPH LENGTH
GH	GRAPH HEIGHT
VPLM	VIRTUAL PAGE LEFT MARGIN
VPBM	VIRTUAL PAGE BOTTOM MARGIN
LROT	LOGICAL ROTATION FLAG
CH	CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
TL	TIC LENGTH

All arguments are input. By default the virtual page is a reduced replica of the full page so that REGION is seldom called. SETREG should be called first.

2.6.2 SETREG

Purpose: To define a region of a page.

CALL SETREG [(mxr,myr,nxr,nyr{,scrxa,scrya,scrxb,scryb})]

MXR	ABSCISSA SEGMENT NUMBER (if a real number then fraction of window for start)
MYR	ORDINATE SEGMENT NUMBER (if a real number then fraction of window for start)
NXR	NUMBER OF ABSCISSA SEGMENTS (if a real number then fraction of window for stop)
NYR	NUMBER OF ORDINATE SEGMENTS (if a real number then fraction of window for stop)
SCRXA	SCREEN (DEVICE) ABSCISSA START (window change)
SCRYA	SCREEN (DEVICE) ORDINATE START (window change)
SCRXB	SCREEN (DEVICE) ABSCISSA STOP (window change)
SCRYB	SCREEN (DEVICE) ORDINATE STOP (window change)

All arguments are input. No arguments resets to full page. The last four arguments are

for internal use only. Note that two distinct methods are available for defining regions of a page. Either by a rational fraction or by explicit real extrema. The following program fragments illustrate this:

```
C UPPER LEFT QUADRANT
      CALL SETREG ( 1, 2, 2, 2)

C MIDDLE (HALF) OF PAGE
      CALL SETREG ( 0.25, 0.25, 0.75, 0.75)
```

2.7 REGION ANNOTATION

Additional annotation beyond that available for the page. Space is not allocated until needed for these text lines.

2.7.1 ORDLYN

Purpose: To draw a line of text along the left edge of a region.

```
CALL ORDLYN ( ORDTXT, cfact)

 ORDTXT  CHARACTER VARIABLE COMMENT LINE FOR LEFT EDGE
 CFACT   CHARACTER HEIGHT MULTIPLIER (default 1.0)
```

2.7.2 REGLYN

Purpose: To draw a line of text along the bottom of a region.

```
CALL REGLYN ( RCLTXT, cfact)

 RCLTXT  CHARACTER VARIABLE REGION COMMENT LINE
 CFACT   CHARACTER HEIGHT MULTIPLIER (default 1.0)
```

2.7.3 REGTITL

Purpose: Draw a region title

```
CALL REGTITL ( RTXTL, cfact)

 RTXTL   CHARACTER VARIABLE REGION TITLE
 CFACT   CHARACTER HEIGHT MULTIPLIER (default 1.4)
```

2.8 REGION PLOTTING

These subprograms are similar to the quick plotting routines described above. The chief difference is that they work within a region. However as a region is always defined by NUPAGE these may be used as needed.

2.8.1 REGPIX

Purpose: To provide quick plotting in regions for up to 5 curves

```
 CALL REGPIX ( XTXT, YTXT, TXTL, XR1, YR1, N1[, inteq1[, XR2, YR2,
               N2[, inteq2[, ... [, XR5, YR5, N5, inteq5]]]]]]])

 XTXT    CHARACTER VARIABLE ABSCISSA TITLE
 YTXT    CHARACTER VARIABLE ORDINATE TITLE
 TXTL    CHARACTER VARIABLE GRAPH TITLE
 XR1     ARRAY OF ABSCISSA VALUES FOR FIRST CURVE
 YR2     ARRAY OF ORDINATE VALUES FOR FIRST CURVE
```

N1 NUMBER OF POINTS IN THE FIRST CURVE
INTEQ1 INTEGER EQUIVALENT FOR FIRST CURVE MARKER (default 1)
XR2 ARRAY OF ABSCISSA VALUES FOR SECOND CURVE
.
.
.
INTEQ5 INTEGER EQUIVALENT FOR FIFTH CURVE MARKER (default 5)

All arguments are input.

2.8.2 REGTEXT

Purpose: To write a region of text.

CALL REGTEXT (TXTRAY, NLINES, chfrac)

TXTRAY ARRAY OF CHARACTER VARIABLE STRINGS
NLINES NUMBER OF CHARACTER STRINGS
CHFRAC ARRAY OF FRACTIONAL CHARACTER HEIGHTS (of default value)
IF NEGATIVE CENTER CHARACTER STRING ON LINE

All arguments are input. Lines are numbered top down.

2.8.3 REG3D

Purpose: To represent a two dimensional surface as either a projection or an isogram.
CALL REG3D (XTXT, YTXT, XTXT, TXTL, XR, YR, FR, NX, ny,
theta, phi, mode)

XTXT CHARACTER VARIABLE X-AXIS TITLE
YTXT CHARACTER VARIABLE Y-AXIS TITLE
ZTXT CHARACTER VARIABLE Z-AXIS TITLE
TXTL CHARACTER VARIABLE GRAPH TITLE
XR ARRAY OF VALUES FOR X-AXIS
YR ARRAY OF VALUES FOR Y-AXIS
FR TWO DIMENSIONAL ARRAY OF VALUES FOR Z-AXIS (F(x,y))
NX NUMBER OF POINTS IN THE X-AXIS ARRAY and the row dimension of FR
NY NUMBER OF POINTS IN THE Y-AXIS ARRAY (default NX)
THETA POLAR VIEWING ANGLE IN DEGREES (default 40.0)
if 0.0 an isogram is drawn.
PHI AZIMUTHAL VIEWING ANGLE IN DEGREES (default -45.0)
MODE FLAG WORD CONTROLLING HIDDEN LINE REMOVAL (default 3)
=1 plot all points
=3 plot only visible points (remove hidden lines)

All arguments are input only; the axes are autoscaled.

2.9 SETTING OF GRAPH PARAMETERS

Normally either BOUNDS, SETXAX and SETYAX, or AUTXST and AUTYST are called with SETGRD called whenever a grid mesh is desired. The automatic axis setting routines may be changed by SETSCAL q.v. For projections, SET3D must be called before the other set-up routines in order to establish axis lengths; a z-axis length, GZ, not equal to zero signals a 3D transformation. The same parameter may be set in a number of different subprograms; consequently an omitted argument is unchanged from the last value established.

2.9.1 AUTXST

Purpose: To automatically set x-axis parameters for an array.

CALL AUTXST (XRAY, NPTS, rxal, ninc)

XRAY	ARRAY OF ABSCISSA DATA VALUES
NPTS	NUMBER OF VALUES
RXAL	RELATIVE AXIS LENGTH TO GRAPH PANEL LENGTH (unless an integer 1< NPANH <17 NUMBER OF PANELS)
NINC	INCREMENT (DEFAULT 1)

All of the arguments in SETXAX are computed.

2.9.2 AUTYST

Purpose: To automatically set y-axis parameters for an array.

CALL AUTYST (YRAY, NPTS, ryal, ninc)

YRAY	ARRAY OF ORDINATE DATA VALUES
NPTS	NUMBER OF VALUES
RYAL	RELATIVE AXIS LENGTH TO GRAPH HEIGHT (unless an integer 1< NPANV <17 NUMBER OF PANELS)
NINC	INCREMENT (DEFAULT 1)

All of the arguments in SETYAX are computed.

2.9.3 AUTZST

Purpose: To automatically set z-axis parameters for an array.

CALL AUTZST (ZRAY, NPTS, rzal, ninc)

ZRAY	ARRAY OF ORDINATE DATA VALUES
NPTS	NUMBER OF VALUES
RZAL	RELATIVE Z-AXIS LENGTH (not used)
NINC	INCREMENT (DEFAULT 1)

All of the arguments in SETZAX are computed.

2.9.4 BOUNDS

Purpose: Set data bounds both linear and logarithmic.

CALL BOUNDS (XA,XB,IX,YA,YB,IY[,ZA,ZB,IZ])

XA	ABSCISSA LOWER BOUND
XB	ABSCISSA UPPER BOUND
IX	AXIS FLAG (0,linear; 1,logarithmic)
YA	ORDINATE LOWER BOUND
YB	ORDINATE UPPER BOUND
IY	AXIS FLAG (0,linear; 1,logarithmic)
ZA	HEIGHT LOWER BOUND
ZB	HEIGHT UPPER BOUND
IZ	AXIS FLAG (0,linear; 1,logarithmic)

2.9.5 GRFSYZ

Purpose: To define a graph frame or panel.

CALL GRFSYZ (gpl,gph,nx,ny,ch,tl,dtic)

GPL	GRAPH PANEL LENGTH
GPH	GRAPH PANEL HEIGHT
NX	NUMBER OF ABSCISSA INTERVALS
NY	NUMBER OF ORDINATE INTERVALS
CH	CHARACTER HEIGHT (IF ABSCENT USE PAGE VALUE;

```
        IF NEGATIVE -CHFRACT OF PAGE VALUE)
 TL     TIC LENGTH (default CH)
 DTIC   FRACTION OF TIC ON PLUS SIDE OF AXIS (default 1.0)
```

The x-axis and y-axis parameters are automatically set using device limits. This subprogram is occasionally used with BOUNDS in which case GRFSYZ is normally called first. The z-axis length is set to zero and other parameters reset.

2.9.6 SET3D

Purpose: To define an isometric map or an orthogonal projection.

CALL SET3D (iproj, theta, phi, mode, rx, ry, rz, jpn)

```
 IPROJ   TRANSFORMATION FLAG
         =1  ORTHOGONAL PROJECTION
         =2  ISOMETRIC MAP
 THETA   POLAR ANGLE OR ISOMETRIC ETA (angle from x-axis to horizontal)
 PHI     AZIMUTH OR ISOMETRIC ZETA (angle from y-axis to horizontal)
 MODE    FLAG WORD CONTROLLING HIDDEN LINE REMOVAL (default 3)
         =1  PLOT ALL POINTS
         =3  PLOT ONLY VISIBLE POINTS
 RX      RELATIVE X-AXIS LENGTH (LINEAR SPACE)
 RY      RELATIVE Y-AXIS LENGTH (LINEAR SPACE)
 RZ      RELATIVE Z-AXIS LENGTH (LINEAR SPACE)
 JPN     PEN NUMBER FOR DRAWING BEHIND THE PROJECTION PLANE
```

The default transformation is an on orthogonal projection from (40,315). A call to SETGRD is implicit with a full ruling for the orthogonal projections and with a ruling parallel to the x-axis for the isometric map.

2.9.7 SETGRD

Purpose: To cause a grid mesh to be drawn; to draw a ruling on a surface

CALL SETGRD (lparlx, lparly, inp, inc3d)

```
 LPARLX  GRID LINES PARALLEL TO X-AXIS (default .FALSE.)
 LPARLY  GRID LINES PARALLEL TO Y-AXIS (default .FALSE.)
 INP     NEW PEN NUMBER (default 1)
 INC3D   INCREMENT BETWEEN ACCENTED RULINGS (default 10)
```

After a grid mesh is drawn the flags are reset.

2.9.8 SETXAX

Purpose: To define x-axis parameters

CALL SETXAX (xa,xb,ix,gx,nx,lx,jx,ixt,iyt,xft,dxlt)

```
 XA      ABSCISSA LOWER BOUND
 XB      ABSCISSA UPPER BOUND
 IX      AXIS FLAG ( 0,linear; 1,logarithmic)
 GX      LENGTH OF THE X AXIS (DEFAULT GPL)
 NX      NUMBER OF minor INTERVALS ALONG ABSCISSA
 LX      NUMBER OF major LABELED INTERVALS
 JX      NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
 IXT     TOP AXIS DRAWING FLAG
 IXB     BOTTOM AXIS DRAWING FLAG
         AXIS FLAG: 1  DRAW AXIS (default)
                    0  DO NOT DRAW AXIS
                   -1  OMIT LABEL ON Y-AXIS
                   11  OMIT SCALE
 XFT     ABSCISSA VALUE OF FIRST TIC (DEFAULT XA)
 DXLT    DELTA VALUE BETWEEN MAJOR TICS (DEFAULT 0.0)
```

Parameters which are omitted are computed except for the first 4.

2.9.9 SETYAX

Purpose: To define y-axis parameters

CALL SETYAX (ya,yb,iy,gy,ny,ly,jy,iyl,iyr,yft,dylt)

YA	ORDINATE LOWER BOUND
YB	ORDINATE UPPER BOUND
IY	AXIS FLAG (0,linear; 1,logarithmic)
GY	LENGTH OF THE Y-AXIS (DEFAULT GPH; IF NEGATIVE FRACTION OF GH)
NY	NUMBER OF minor INTERVALS ALONG ORDINATE
LY	NUMBER OF major LABELED INTERVALS
JY	NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
IYL	LEFT AXIS DRAWING FLAG
IYR	RIGHT AXIS DRAWING FLAG
	AXIS FLAG: 1 DRAW AXIS (default)
	0 DO NOT DRAW AXIS
	-1 OMIT LABEL ON X-AXIS
	11 OMIT SCALE
YFT	ORDINATE VALUE OF FIRST TIC (DEFAULT YA)
DYLT	DELTA VALUE BETWEEN MAJOR TICS (DEFAULT 0.0)

Parameters which are omitted are computed except for the first 4.

2.9.10 SETZAX

Purpose: To define z-axis parameters

CALL SETZAX (za,zb,iz,gz,nz,lz,jz,izl,izr,zft,dzlt)

ZA	ORDINATE LOWER BOUND
ZB	ORDINATE UPPER BOUND
IZ	AXIS FLAG (0,linear; 1,logarithmic)
GZ	LENGTH OF THE Z-AXIS (DEFAULT 0.0)
NZ	NUMBER OF minor INTERVALS ALONG ORDINATE
LZ	NUMBER OF major LABELED INTERVALS
JZ	NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
IZL	LEFT AXIS DRAWING FLAG
IZR	RIGHT AXIS DRAWING FLAG
	AXIS FLAG: 1 DRAW AXIS (default)
	0 DO NOT DRAW AXIS
ZFT	ORDINATE VALUE OF FIRST TIC (DEFAULT ZA)
DZLT	DELTA VALUE BETWEEN MAJOR TICS (DEFAULT 0.0)

Parameters which are omitted are computed except for the first 4.

2.10 FRAME DRAWING

There is only one frame per region; the frame may have several panels.

2.10.1 FRAM3D

Purpose: To draw a complete graph frame for 3D graphs.

CALL FRAM3D (XTXT, YTXT, ZTXT, TXTL)

XTXT	CHARACTER VARIABLE ABSCISSA TITLE
YTXT	CHARACTER VARIABLE ORDINATE TITLE
ZTXT	CHARACTER VARIABLE HEIGHT TITLE
TXTL	CHARACTER VARIABLE GRAPH TITLE

2.10.2 GRFRAM

Purpose: To draw a complete graph frame

CALL GRFRAM (XTXT, YTXT, TXTL)

XTXT	CHARACTER VARIABLE ABSCISSA TITLE
YTXT	CHARACTER VARIABLE ORDINATE TITLE
TXTL	CHARACTER VARIABLE GRAPH TITLE

2.10.3 MOSAIC

Purpose: To draw complete panel frames

CALL MOSAIC (XTXT,YTXT,TXTL,rgph)

XTXT	CHARACTER VARIABLE ABSCISSA TITLE
YTXT	CHARACTER VARIABLE ORDINATE TITLE
TXTL	CHARACTER VARIABLE GRAPH TITLE
RGPH	RELATIVE GRAPH PANEL HEIGHT

The panels are stacked bottom up. The abscissa title will only be written on the first (bottom) panel; the graph title on the top panel.

2.11 ADDITIONAL GRAPH ANNOTATION

These annotation routines are intended for use with GRFRAM and MOSAIC.

2.11.1 MRKTXT

Purpose: To write a marker key in a panel

CALL MRKTXT (INTEQ, MTXT, xfr, yfr, chfr)

INTEQ	INTEGER EQUIVALENT OF MARKER
MTXT	CHARACTER VARIABLE LINE OF TEXT
XFR	FRACTION OF GPL FOR START OF SYMBOL STRING
YFR	FRACTION OF GPH FOR START OF SYMBOL STRING
CHFR	CHARACTER HEIGHT FRACTION OF DEFAULT HEIGHT

Consecutive calls within a single panel will give multiple lines one below another left justified. The key text immediately follows the marker; a leading space, ' ...', may be useful or even ' = ...' when writing a key.

2.11.2 PANTXT

Purpose: To write addtional annotations within a panel

CALL PANTXT (PTXT, xfr, yfr, chfr)

PTXT	CHARACTER VARIABLE LINE OF TEXT
XFR	FRACTION OF GPL FOR START OF SYMBOL STRING
YFR	FRACTION OF GPH FOR START OF SYMBOL STRING
CHFR	CHARACTER HEIGHT FRACTION OF DEFAULT HEIGHT

Consecutive calls within a single panel will give multiple lines one below another left justified.

2.11.3 TXTLYN

Purpose: To write an additional line of text on a graph, a comment line.

```
CALL TXTLYN ( COMTXT)

 COMTXT  CHARACTER VARIABLE LINE OF TEXT
```

Only one comment line per graph frame. Space for this line is allocated for all regions with the full page height.

2.12 CURVE DRAWING 1

These subprograms all use subject space coordinates. Some plot arrays of points others single points; SYMRAY and PLTDAT are perhaps the most general.

2.12.1 ARYPLT

Purpose: To plot a parameterized array of data points {X(t),Y(t)} with optional labeling of the curve.

```
CALL ARYPLT ( XRAY,YRAY,NPTS[,TRAY,dtc,nsym,dtn,ch,ndig])

 XRAY    ARRAY OF ABSCISSA VALUES
 YRAY    ARRAY OF ORDINATE VALUES
 NPTS    NUMBER OF POINTS IN THE ARRAYS
 TRAY    CONTROL ARRAY FOR CURVE TIC MARKS
 DTC     DELTA FOR TIC MARKS (DEFAULT = 1.0)
         ASSUME TIC AT 0.0
 INTEQ   INTEGER EQUIVALENT FOR DESIRED SYMBOL if 0<=INTEQ<=13
         else tic normal to curve. (DEFAULT=TIC NORMAL TO CURVE)
 DTN     DELTA FOR TIC MARK ANNOTATION
         MUST BE A MULTIPLE OF DTC (DEFAULT=0.0 FOR NO ANNOTATION)
 CH      CHARACTER HEIGHT (DEFAULT PAGE VALUE; <0 -CHFACT)
 NDIG    NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT (default=-1)
```

All arguments are input. The 3-argument call gives a plot without tics and annotation.

2.12.2 HISTRA

Purpose: To plot a histogram

```
CALL HISTRA ( XRAY1,XRAY2,YRAY1,YRAY2,NPTS[,REFL,IHST])

 XRAY1   ARRAY OF ABSCISSA VALUES
 XRAY2   ARRAY OF ABSCISSA VALUES ( possibly XRAY1 shifted)
 YRAY1   ARRAY OF ORDINATE VALUES
 YRAY2   ARRAY OF ORDINATE VALUES ( possibly YRAY1 shifted)
 NPTS    NUMBER OF HISTOGRAM INTERVALS
 REFL    REFERENCE VALUE FOR DRAWING COMPLETE HISTOGRAM
 IHST    FLAG FOR ABSCISSA(=2)/ORDINATE(=1) REFERENCE
```

Line segments are plotted joining the points {XRAY1(I),YRAY1(I)} and {XRAY2(I),YRAY2(I)} for I=1,NPTS. A complete histogram has a line from the reference level to the value for every point. Note that CALL HISTRA (XRAY,XRAY,YRAY,YRAY,NPTS) is equivalent to CALL SYMRAY (XRAY,YRAY,,,, NPTS).

2.12.3 LINRAY

Purpose: To produce a line or symbol curve from pairs of data values. The data points may be represented by centered symbols with/without connecting lines between points.

```
CALL LINRAY ( XRAY,YRAY,NPTS,INC,LINTYP,INTEQ,ch,theta)

 XRAY     ARRAY OF ABSCISSA VALUES
 YRAY     ARRAY OF ORDINATE VALUES
 NPTS     NUMBER OF POINTS IN THE ARRAYS
 INC      INDEX INCREMENT FOR THE ARRAYS
 LINTYP   INTEGER CONTROL PARAMETER FOR TYPE OF LINE TO BE DRAWN
          The magnitude of LINTYP determines the frequency of
          plotted symbols; that is, if LINTYP = 4, a symbol is plotted
          at every fourth data point although every point is plotted.
          If LINTYP is positive then connecting lines are drawn; if
          negative then no connecting lines are drawn and only the
          symbols are plotted.
 INTEQ    INTEGER EQUIVALENT FOR DESIRED SYMBOL
 CH       CHARACTER HEIGHT (if abscent page value; if negative fraction
          of page value)
 THETA    ANGLE IN DEGREES (default 0.0)

Note CH and THETA are in page coordinates.
```

2.12.4 NUMDAT

```
Purpose:  NUMBER with subject space coordinates

CALL NUMDAT ( XS, YS, CH, FPN, THETA, NDIG, ingraf)

 XS       I   SUBJECT SPACE ABSCISSA VALUE
 YS       I   SUBJECT SPACE ORDINATE VALUE
 CH       I   CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
 FPN      I   FLOATING POINT NUMBER
 THETA    I   ANGLE IN DEGREES
 NDIG     I   NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
              (-1 suppresses the decimal point)
 INGRAF   O   POINT IN GRAPH FLAG
              +1  IN GRAPH
               0  OFF GRAPH
              -1  CONVERSION FAILURE

Note CH and THETA are in page coordinates.
```

2.12.5 PLSIGDAT

```
Purpose:  To plot data points with standard deviations

CALL PLSIGDAT ( XRAY, sigxra, YRAY, sigyra, NPTS, inteq, inc)

 XRAY     ARRAY OF ABSCISSA VALUES
 SIGXRA   ARRAY OF STANDARD DEVIATIONS OF XRAY (default 0.0)
 YRAY     ARRAY OF ORDINATE VALUES
 SIGYRA   ARRAY OF STANDARD DEVIATIONS OF YRAY (default 0.0)
|NPTS|    NUMBER OF DATA POINTS
          >0  POINTS CONNECTED
          <0  POINTS UNCONNECTED
 INTEQ    INTEGER EQUIVALENCE FOR MARKER AT DATA POINTS (DEFAULT 1)
 INC      INCREMENT (DEFAULT 1)
```

2.12.6 PLTDAT

```
Purpose:  To draw line segments in subject space.

CALL PLTDAT [( XS,YS[,IPEN[,INGRAF])]

 XS       I   SUBJECT SPACE ABSCISSA VALUE
 YS       I   SUBJECT SPACE ORDINATE VALUE
 IPEN     I   DRAW/MOVE FLAG
              +3  MOVE
```

```
            +2  DRAW
            +1  CLIP
             0  ONLY IF FULL SEGMENT IN RANGE
            -1  FOLD
            -2  DISCRETE SYMBOLS DRAW
            -3  DISCRETE SYMBOLS MOVE
           -11  RESET FOLDING (NO OUTPUT)
INGRAF  O   POINT IN GRAPH FLAG
            +1  IN GRAPH
             0  OFF GRAPH
            -1  CONVERSION FAILURE
```

The full range of options in PLTDAT are selected in SETPLT. The 2 argument call draws a curve with the current point connected to the previous point according to the value of ICURV. A call with no arguments resets PLTDAT and may be used to start a new curve.

2.12.6.1 SETPLT - Purpose: To change PLTDAT options particularly for 2 argument calls.

CALL SETPLT (icurv,nrmin,nrmax,inteq,theta,ch)

```
ICURV   INTEGER FLAG FOR CURVE TYPE (DEFAULT -1)
        +3  MOVE
        +2  DRAW
        +1  CLIP
         0  ONLY IF FULL SEGMENT IN RANGE
        -1  FOLD
        -2  DISCRETE SYMBOLS DRAW
        -3  DISCRETE SYMBOLS MOVE
NRMIN   MINIMUM FOLDING RANGE (DEFAULT -1)
NRMAX   MAXIMUM FOLDING RANGE (DEFAULT +1)
INTEQ   INTEGER EQUIVALENT FOR DESIRED SYMBOL
        (DEFAULT 6 IF ICURV = -1 ELSE 0)
THETA   ANGLE IN DEGREES
        (DEFAULT 22.5 IF ICURV = -1 ELSE 0.0)
CH      CHARACTER HEIGHT (DEFAULT PAGE VALUE IF NEGATIVE
        -CHFRACT OF PAGE VALUE)
```

Defaults are reset on each call.

2.12.7 PLTFAZ

Purpose: To plot a phase angle with wrap around

CALL PLTFAZ (XRAY, PHASE, NPTS)

```
XRAY    ARRAY OF ABSCISSA VALUES
PHASE   ARRAY OF ORDINATE VALUES
NPTS    NUMBER OF POINTS IN THE ARRAYS
```

The ordinate scale is assumed to be one full cycle whether the phase is in degrees or radians. For example, either -180 to 180 or -pi to pi or 0 to 2*pi and so forth. The shortest line (angle) is used to connect consecutive points.

2.12.8 SYMDAT

Purpose: SYMBOL with subject space coordinates

```
CALL SYMDAT ( XS, YS, CH, IBCD, THETA, NBCD, ingraf)     !or
CALL SYMDAT ( XS, YS, CH, INTEQ, THETA, -ICODE, ingraf)

XS      I   SUBJECT SPACE ABSCISSA VALUE
YS      I   SUBJECT SPACE ORDINATE VALUE
CH      I   CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
IBCD    I   BYTE TEXT STRING
THETA   I   ANGLE IN DEGREES
NBCD    I   NUMBER OF CHARACTERS IN IBCD
```

INGRAF	O	POINT IN GRAPH FLAG +1 IN GRAPH 0 OFF GRAPH −1 CONVERSION FAILURE
INTEQ	I	INTEGER EQUIVALENT FOR SPECIAL CENTERED SYMBOL
ICODE	I	DRAW/MOVE FLAG (2 DRAW)

Note CH and THETA are in page coordinates.

2.12.9 SYMRAY

Purpose: To provide a symbol routine for arrays in subject space

CALL SYMRAY (XRAY,YRAY,ch,inteq,theta,NPTS,icurv,nrmin,nrmax,inc)

XRAY	ARRAY OF ABSCISSA VALUES
YRAY	ARRAY OF ORDINATE VALUES
CH	CHARACTER HEIGHT (if abscent page value; if negative fraction of page value)
INTEQ	INTEGER EQUIVALENT FOR DESIRED SYMBOL (default 0)
THETA	ANGLE IN DEGREES (default 0.0)
NPTS	NUMBER OF POINTS IN THE ARRAYS
ICURV	INTEGER FLAG FOR CURVE TYPE (default 1)
NRMIN	MINIMUM FOLDING RANGE (default −1)
NRMAX	MAXIMUM FOLDING RANGE (default 1)
INC	INDEX INCREMENT FOR THE ARRAYS (default 1)

Note CH and THETA are in page coordinates.

2.13 SPACE CURVES AND SURFACE REPRESENTATION

All of these routines use subject space coordinates.

2.13.1 CONMAP

Purpose: To locate and to plot specified isopleths of a function of two independent variables; to draw isograms.

CALL CONMAP (XRAY,YRAY,FRAY,MX,my,cray,mc,fmax,mf,kx,ky,incx,incy,npnray)

XRAY	ARRAY OF ABSCISSA VALUES
YRAY	ARRAY OF ORDINATE VALUES
FRAY	TWO DIMENSIONAL ARRAY OF FUNCTION VALUES
MX	DIMENSION OF XRAY
MY	DIMENSION OF YRAY (default MX)
CRAY	ARRAY OF ISOPLETHS IN ASCENDING ORDER
MC	DIMENSION OF CRAY (default 0) MC > 0 CRAY CONTAINS ACTUAL VALUES OF THE ISOPLETHS MC < 0 CRAY CONTAINS FRACTIONAL VALUES FOR THE ISOPLETHS RELATIVE TO THE MAXIMUM WHICH IS COMPUTED MC = 0 TEN ISOPLETHS ARE INTERNALLY GENERATED WITH A SPACING OF CRAY(1) IF NON ZERO
FMAX	MAXIMUM VALUE OF FRAY (OUTPUT)
MF	ROW DIMENSION OF FRAY (default MX)
KX	START INDEX FOR XRAY (default 1)
KY	START INDEX FOR YRAY (default 1)
INCX	INDEX INCREMENT FOR XRAY (default 1)
INCY	INDEX INCREMENT FOR YRAY (default 1)
NPNRAY	ARRAY OF LENGTH |MC| GIVING NEW PEN NUMBERS FOR EACH ISOPLETH

If both CRAY and MC are not given then 10 equispaced isopleths are plotted; if CRAY is not given then |MC| equispaced isopleths are plotted; if MC is not given then isopleths spaced CRAY(1) apart are plotted.

2.13.1.1 SETCON - Purpose: To define an external function input to CONMAP

CALL SETCON (lext, functn)

LEXT	LOGICAL VARIABLE TRUE IF AN EXTERNAL FUNCTION IS USED (DEFAULT .FALSE.)
FUNCTN	NAME OF A FUNCTION OF TWO VARIABLES

If an external function is used then FRAY must be a scratch array of length MY.

2.13.2 FNCON

Purpose: To locate and to plot specified isopleths of a function of two independent variables; to draw isograms and to label each isopleth.

CALL FNCON (XRAY,YRAY,FRAY,MX,my,rmin,rmax,rinc,lbl1,lbl2,idsrn)

XRAY	ARRAY OF ABSCISSA VALUES
YRAY	ARRAY OF ORDINATE VALUES
FRAY	TWO DIMENSIONAL ARRAY OF FUNCTION VALUES
MX	DIMENSION OF XRAY AND ROW DIMENSION OF FRAY
MY	DIMENSION OF YRAY (default MX)
RMIN	MINIMUM ISOPLETH
RMAX	MAXIMUM ISOPLETH
RINC	INCREMENT BETWEEN ISOPLETHS
LBL1	NUMBER OF CHARACTERS TO BE DISPLAYED IN THE LABELS (DEFAULT 6)
LBL2	NUMBER OF CHARACTER POSITIONS TO LEFT OF DECIMAL POINT (DEFAULT 4)
IDSRN	DATA SET REFERENCE NUMBER FOR INFORMATIONAL MESSAGES (DEFAULT 0 NO OUTPUT)

Fill is not handled by FNCON. No external function may be specified and the isopleths are always linearly spaced. If RMIN=RMAX=0 then RINC levels are generated (if RINC=0 then 10 levels). Note that the graph comment line (TXTLYN) is used to display a remote exponent for the isopleth labels; therefore TXTLYN should not be used with FNCON.

2.13.3 PLTXYZ

Purpose: To plot a point in 3-D subject space

CALL PLTXYZ (xs, ys, zs, IPEN, ingraf)

XS	I	SUBJECT SPACE ABSCISSA VALUE (DEFAULT XA)
YS	I	SUBJECT SPACE ORDINATE VALUE (DEFAULT YA)
ZS	I	SUBJECT SPACE HEIGHT VALUE (DEFAULT ZA)
IPEN	I	PEN STATUS FLAG IPEN = 2 PEN DOWN (DRAW) IPEN = 3 PEN UP (MOVE)
INGRAF	O	POINT IN GRAPH FLAG +1 IN GRAPH 0 OFF GRAPH -1 CONVERSION FAILURE

By omitting a coordinate a plane curve may be drawn in the corresponding coordinate plane.

2.13.4 SPCURV

Purpose: To draw a space curve defined by arrays

CALL SPCURV (XR, YR, ZR, NPTS, jpn)

XR	ARRAY OF X VALUES
YR	ARRAY OF Y VALUES
ZR	ARRAY OF Z VALUES
NPTS	NUMBER OF POINTS

JPN PEN NUMBER FOR FAR POINTS

2.13.5 THREED

Purpose: To draw an isometric map or orthogonal projection of a two dimensional surface.

CALL THREED (XRAY, YRAY, fray, MX, my, mrow, kpts, lpts)

XRAY	ARRAY OF SUBJECT SPACE VALUES FOR X-AXIS
YRAY	ARRAY OF SUBJECT SPACE VALUES FOR Y-AXIS
FRAY	TWO DIMENSIONAL ARRAY OF SUBJECT SPACE VALUES OF THE SURFACE (not needed if an external function is specified)
MX	DIMENSION OF XRAY
MY	DIMENSION OF YRAY (DEAFULT MX)
MROW	ROW DIMENSION OF FRAY (DEFAULT MX)
KPTS	NUMBER OF INTERPOLATION POINTS ALONG X (DEFAULT 0)
LPTS	NUMBER OF INTERPOLATION POINTS ALONG Y (DEFAULT 0)

The transformation is normally set in SET3D although SETGLT could be used. Virtual memory is used for scratch storage.

2.13.5.1 SETEXT — Purpose: To define an external function input to THREED

CALL SETEXT (lext, functn)

LEXT	LOGICAL VARIABLE TRUE IF AN EXTERNAL FUNCTION IS USED (default .FALSE.)
FUNCTN	NAME OF A FUNCTION OF TWO VARIABLES

2.14 CURVE DRAWING 2

These routines all use page coordinates.

2.14.1 PLERRBAR

Purpose: To plot a marker with error bars

CALL PLERRBAR (XMIN, XP, XMAX, YMIN, YP, YMAX, IP,
inteq, ch, ibar1, ibar2, ibar3, ibar4)

XMIN	MINIMUM ABSCISSA VALUE
XP	NOMINAL ABSCISSA VALUE
XMAX	MAXIMUM ABSCISSA VALUE
YMIN	MINIMUM ORDINATE VALUE
YP	NOMINAL ORDINATE VALUE
YMAX	MAXIMUM ORDINATE VALUE
IP	DRAW/MOVE FLAG (2/3)
INTEQ	INTEGER EQUIVALENT FOR MARKER (DEFAULT 1)
CH	CHARACTER HEIGHT (DEFAULT PAGE VALUE)
IBAR1	MINIMUM ABSCISSA MARKER (DEFAULT 13)
IBAR2	MAXIMUM ABSCISSA MARKER (DEFAULT 13)
IBAR3	MINIMUM ORDINATE MARKER ROTATED 90.0 (DEFAULT 13)
IBAR4	MAXIMUM ORDINATE MARKER ROTATED 90.0 (DEFAULT 13)

The error bars are excluded from a box about each marker.

2.14.2 PLTOBJ

Purpose: To draw line segments in object space

CALL PLTOBJ (XO, YO[, IPEN[, INGRAF])

XO	I	OBJECT SPACE ABSCISSA VALUE
YO	I	OBJECT SPACE ORDINATE VALUE
IPEN	I	DRAW/MOVE FLAG
		+3 MOVE
		+2 DRAW
		+1 CLIP
		0 ONLY IF FULL SEGMENT IN RANGE
		-1 FOLD
		-2 DISCRETE SYMBOLS DRAW
		-3 DISCRETE SYMBOLS MOVE
		-11 RESET FOLDING (NO OUTPUT)
INGRAF	O	POINT IN GRAPH FLAG
		+1 IN GRAPH
		0 OFF GRAPH
		-1 CONVERSION FAILURE

The full range of options in PLTOBJ are selected in SETPLT. The 2 argument call draws a curve with the current point connected to the previous point according to the value of ICURV. See PLTDAT for further discusion of use.

2.14.3 STRPLT

Purpose: To plot a multi-segmented curve and label each segment.

CALL STRPLT (VAL,NSEG,OPEN,NUMPTS,XPRAY,YPRAY,JORD,IORD,KCRS,KSEG,J1,J2)

VAL	REAL VALUE ASSOCIATED WITH CURVE
NSEG	NUMBER OF DISJOINT SEGMENTS IN CURVE (NSEG<=KSEG)
OPEN(KSEG)	LOGICAL*1 FLAG INDICATING WHETHER A SEGMENT IS OPEN OR NOT
NUMPTS(KSEG)	NUMBER OF POINTS IN EACH SEGMENT
XPRAY(KCRS)	ABSCISSA OF EVERY POINT IN THE CURVE
YPRAY(KCRS)	ORDINATE OF EVERY POINT IN THE CURVE
JORD(KCRS)	INTEGER*2 INDEX OF EACH POINT IN EACH SEGMENT, CLOSE PACKED
IORD(KCRS)	INTEGER*2 SCRATCH ARRAY (USED TO HOLD INDEX ARRAY FOR EACH SEGMENT)
KCRS	MAXIMUM NUMBER OF POINTS
KSEG	MAXIMUM NUMBER OF SEGMENTS
J1	NUMBER OF CHARACTERS TO BE DISPLAYED IN THE LABEL
J2	NUMBER OF CHARACTER POSTIONS TO THE LEFT OF THE DECIMAL POINT

Beware the non standard variable declarations. This routine is used by FNCON but may be used directly or serve as a model for curve labeling.

2.15 BASIC SUBPROGRAMS

These are the bottom level routines. They are sufficient to perform all elementary plotting operations such as drawing lines, writing annotations, selecting a pen, scaling a plot, etc. The PLOT, SYMBOL, and NUMBER sample program in appendix B demonstrates their use.

2.15.1 FACTOR

Purpose: To enlarge or reduce the size of the entire plot or any portion thereof

CALL FACTOR (factx, facty)

FACTX	THE RATIO OF THE DESIRED PLOT SIZE TO THE NORMAL PLOT SIZE. IN PARTICULAR A MULTIPIER FOR ALL ABSCISSA VALUES (DEFAULT 1.0)
FACTY	THE MULTIPIER FOR ALL ORDINATE VALUES (DEFAULT FACTX)

2.15.2 NEWPEN

Purpose: To change pens on electromechanical plotters.

```
CALL NEWPEN ( INP)
 INP     THE NUMBER OF THE NEXT PEN TO BE SELECTED
```

On electromechanical plotters, the old pen is raised and the new pen moved to the same physical location at which the old pen was positioned. For the ZETA8, the pen numbers normally follow the convention:

1	Black (default)	5	Brown
2	Blue	6	Yellow
3	Red	7	Violet
4	Green	8	Orange

On video terminals, INP selects line type according to the following table:

1	solid (default)	5	long-dash
2	dotted	6	dash/long-dash
3	dot/long-dash	7	long-dash/three-dot
4	dash	8	long-dash/three-dash

In addition to line type, the polarity of the pixels may be set on a separate call to NEWPEN according to the folowing table:

-2	Draw by complementing each dot
-1	Draw in black (turn dots off)
0	Draw in white (turn dots on) (default)

These codes have no effect except on video terminals. A trace may be erased by redrawing it in black.

2.15.3 NUMBER

Purpose: To draw the decimal equivalent of a floating point number

```
CALL NUMBER ( xp,yp,ch,FPN,theta,NDIG)

 XP      PAGE ABSCISSA OF LOWER LEFT CORNER
 YP      PAGE ORDINATE OF LOWER LEFT CORNER
 CH      CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
 FPN     FLOATING POINT NUMBER
 THETA   ANGLE IN DEGREES
 NDIG    NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
```

Fill plots out as the word FILL. Omitted arguments use the previous values from SYMBOL or NUMBER whichever was called last.

2.15.4 OLDPEN

Purpose: To return the number of the pen currently selected

```
CALL OLDPEN ( INP)
 INP     O   CURRENT PEN NUMBER
```

2.15.5 PLOT

Purpose: To draw lines

```
CALL PLOT ( XP, YP, IPEN, inword)

 XP      I   PAGE ABSCISSA
 YP      I   PAGE ORDINATE
 IPEN    I   PEN STATUS FLAG
               IPEN  <   0  REDEFINE VIRTUAL ORIGIN
```

```
              IPEN  =  -1  NO PEN MOVEMENT
             |IPEN| =   2  PEN DOWN (DRAW)
             |IPEN| =   3  PEN UP (MOVE)
              IPEN  = 998  TERMINATE BUT DO NOT RESET TERMINALS TO ASCII
              IPEN  = 999  TERMINATE AND RESET TERMINALS TO ASCII
              IPEN = 1000  FLUSH BUFFERS AND PAUSE
              IPEN = 1001  CLEAR TERMINAL SCREEN
              IPEN = 1002  FLUSH
              IPEN = 1003  FLUSH BUFFERS AND RESET TO ASCII
 INWORD  O   FOUR BYTE WORD FROM TERMINALS OR POINT PLOT FLAG (1,IN; 0,OUT)
```

2.15.6 PLTMSG

Purpose: To permit interactive messages during program execution.

```
CALL PLTMSG ( PROMPT, REPLY)

 PROMPT  I   CHARACTER VARIABLE WITH PROMPT MESSAGE
 REPLY   O   CHARACTER VARIABLE WITH REPLY FROM TERMINAL
```

The prompt will appear on the screen and after the reply has been entered both will be erased.

2.15.7 SNUMBR

Purpose: To output a number in scientific notation

```
CALL SNUMBR ( XP, YP, CH, FPN, THETA, NDIG)

 XP      PAGE ABSCISSA OF LOWER LEFT CORNER
 YP      PAGE ORDINATE OF LOWER LEFT CORNER
 CH      CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
 FPN     FLOATING POINT NUMBER
 THETA   ANGLE IN DEGREES
 NDIG    NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
```

2.15.8 SYMBOL

Purpose: To produce plot annotations at any angle and in any size.

SYMBOL has two modes of operation that are invoked by using two different calling formats: 1) the standard call, which is used for drawing titles, captions, and legends; and 2) the special call, which is used to draw special centered characters (markers), such as a box or a triangle, at a point.

The standard call is:

```
CALL SYMBOL ( xp, yp, ch, BSTR, theta, +NCHAR)

 XP      PAGE ABSCISSA FOR LOWER LEFT CORNER
 YP      PAGE ORDINATE FOR LOWER LEFT CORNER
 CH      CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
 BSTR    BYTE STRING OF CHARACTERS
 THETA   ANGLE IN DEGREES
 NCHAR   NUMBER OF CHARACTERS (0 <= NCHAR <= 512)
```

The special call is:

```
CALL SYMBOL ( xp, yp, ch, INTEQ, theta, -icode)

 XP      PAGE ABSCISSA FOR CENTER OF CHARACTER
 YP      PAGE ORDINATE FOR CENTER OF CHARACTER
 CH      CHARACTER HEIGHT (IF NEGATIVE -CHFRACT OF PAGE VALUE)
 INTEQ   INTEGER EQUIVALENT OF THE DESIRED SYMBOL (centered)
 THETA   ANGLE IN DEGREES
 ICODE   DRAW/MOVE FLAG ( 1,MOVE; 2,DRAW) (default 1)
```

Characters Available in SYMBOL routine [VAX/VMS]

	0	□	16		32	0	48	≅	64	P	80	'	96	ρ	112		128	□	144		160	0	176	ℑ	192	Π	208	'	224	π	240
○	1	○	17	!	33	1	49	A	65	Q	81	α	97	φ	113	○	129	○	145	!	161	1	177	A	193	Θ	209	α	225	θ	241
△	2	△	18	"	34	2	50	B	66	R	82	b	98	r	114	△	130	△	146	"	162	2	178	B	194	P	210	β	226	ρ	242
+	3	◇	19	#	35	3	51	C	67	S	83	c	99	s	115	+	131	◇	147	#	163	3	179	H	195	Σ	211	η	227	σ	243
×	4	□	20	$	36	4	52	D	68	T	84	d	100	t	116	×	132	□	148	$	164	4	180	Δ	196	T	212	δ	228	τ	244
◇	5	▽	21	%	37	5	53	E	69	U	85	e	101	u	117	◇	133	▽	149	%	165	5	181	E	197	ϒ	213	ϵ	229	υ	245
⇑	6		22	&	38	6	54	F	70	V	86	f	102	v	118	⇑	134		150	¢	166	6	182	Φ	198	V	214	ϕ	230	φ	246
⋈	7		23	'	39	7	55	G	71	W	87	g	103	w	119	⋈	135		151	'	167	7	183	Γ	199	Ω	215	γ	231	ω	247
≥	8		24	(	40	8	56	H	72	X	88	h	104	x	120	≥	136		152	(	168	8	184	X	200	ϒ	216	χ	232	ξ	248
∨	9		25	)	41	9	57	I	73	Y	89	ι	105	ψ	121	∨	137		153	)	169	9	185	I	201	Ψ	217	ι	233	ψ	249
⌑	10		26	*	42	:	58	J	74	Z	90	j	106	z	122	⌑	138		154	*	170	:	186	J	202	Z	218	j	234	ζ	250
✳	11		27	+	43	;	59	K	75	[	91	k	107	÷	123	✳	139		155	+	171	;	187	K	203	[	219	κ	235	÷	251
⧖	12		28	,	44	<	60	L	76	\	92	l	108	|	124	⧖	140		156	,	172	<	188	Λ	204	\	220	λ	236	|	252
|	13		29	−	45	=	61	M	77	]	93	m	109	}	125	|	141		157	−	173	=	189	M	205	]	221	μ	237	}	253
✡	14		30	.	46	>	62	N	78	⌒	94	n	110	°	126	✡	142		158	.	174	>	190	N	206	⌒	222	ν	238	°	254
_	15		31	/	47	?	63	O	79	_	95	o	111	√	127	−	143		159	/	175	?	191	O	207	_	223	o	239	√	255

Note: Integer Equivalence > 127 are in font 2
. INTEQ 0 thru 31 are special centered symbols

If XP and/or YP are omitted the annotation is continued from where the last annotation ended; XP and YP are treated independently. If CH or THETA is omitted the last value is used.

The table of symbols is shown in figure 4 along with the integer equivalent for each symbol; the program used to draw figure 4 is listed in appendix B.

The control codes are used for both in stream commands and to reference the special centered characters. In general the commands are parameter setting with all parameters reset at each subprogram call. As in-stream commands the control codes have the following functions:

Mnemonic	Inteq	Meaning
nul	0	Null
stx	2	Start of text
etx	3	End of text
bs	8	Backspace
lf	10	Line feed
luf	11	Line unfeed
cr	13	Carriage return
sup	17	Superscript mode
nor	18	Normal line mode
sub	19	Subscript mode
can	24	Cancel next character as a command
latin	28	Latin character font (font 1)
greek	29	Greek character font (font 2)

2.15.9 WHERE

Purpose: To return the current pen location

CALL WHERE (XP, YP, factx, facty)

XP	O	CURRENT PAGE ABSCISSA
YP	O	CURRENT PAGE ORDINATE
FACTX	O	ABSCISSA MULTIPIER FROM FACTOR
FACTY	O	ORDINATE MULTIPIER FROM FACTOR

2.16 UTILITY 1

These routines are not graphic subprograms consequently there are no restrictions on their use.

2.16.1 MINMAX

Purpose: To find the minimum and maximum values of an array.

CALL MINMAX (ARAY,NPTS,AMIN,AMAX,imin,imax,inc)

ARAY	I/O	ARRAY OF DATA VALUES (output if NPTS<0)
|NPTS|	I	NUMBER OF VALUES
AMIN	O	MINIMUM VALUE OF ARAY
AMAX	O	MAXIMUM VALUE OF ARAY
IMIN	O	INDEX OF FIRST OCCURENCE OF AMIN
IMAX	O	INDEX OF FIRST OCCURENCE OF AMAX
INC	I	INCREMENT (default 1)

Fill values are skipped. If NPTS < 0, then the values in ARAY are linearly shifted to have a zero minimum.

2.16.2 TRUNCRAY

Purpose: To find truncation points of an ascending array

CALL TRUNCRAY (ARRAY, NPTS, AMINL, AMAXL, IMIN, IMAX)

ARRAY	I	ARRAY OF VALUES (monotone increasing)
|NPTS|	I	DIMENSION OF ARRAY
AMINL	I	MINIMUM LIMIT
AMAXL	I	MAXIMUM LIMIT
IMIN	O	INDEX FOR MINIMUM LIMIT (default 1)
IMAX	O	INDEX FOR MAXIMUM LIMIT (default NPTS)

Fill values are skipped. If NPTS > 0, then ARAY(IMIN)<=AMIN and AMAX<=ARAY(IMAX); if NPTS < 0, then AMIN<=ARAY(IMIN) and ARAY(IMAX)<=AMAX.

CHAPTER 3

SAMPLE PROGRAMS

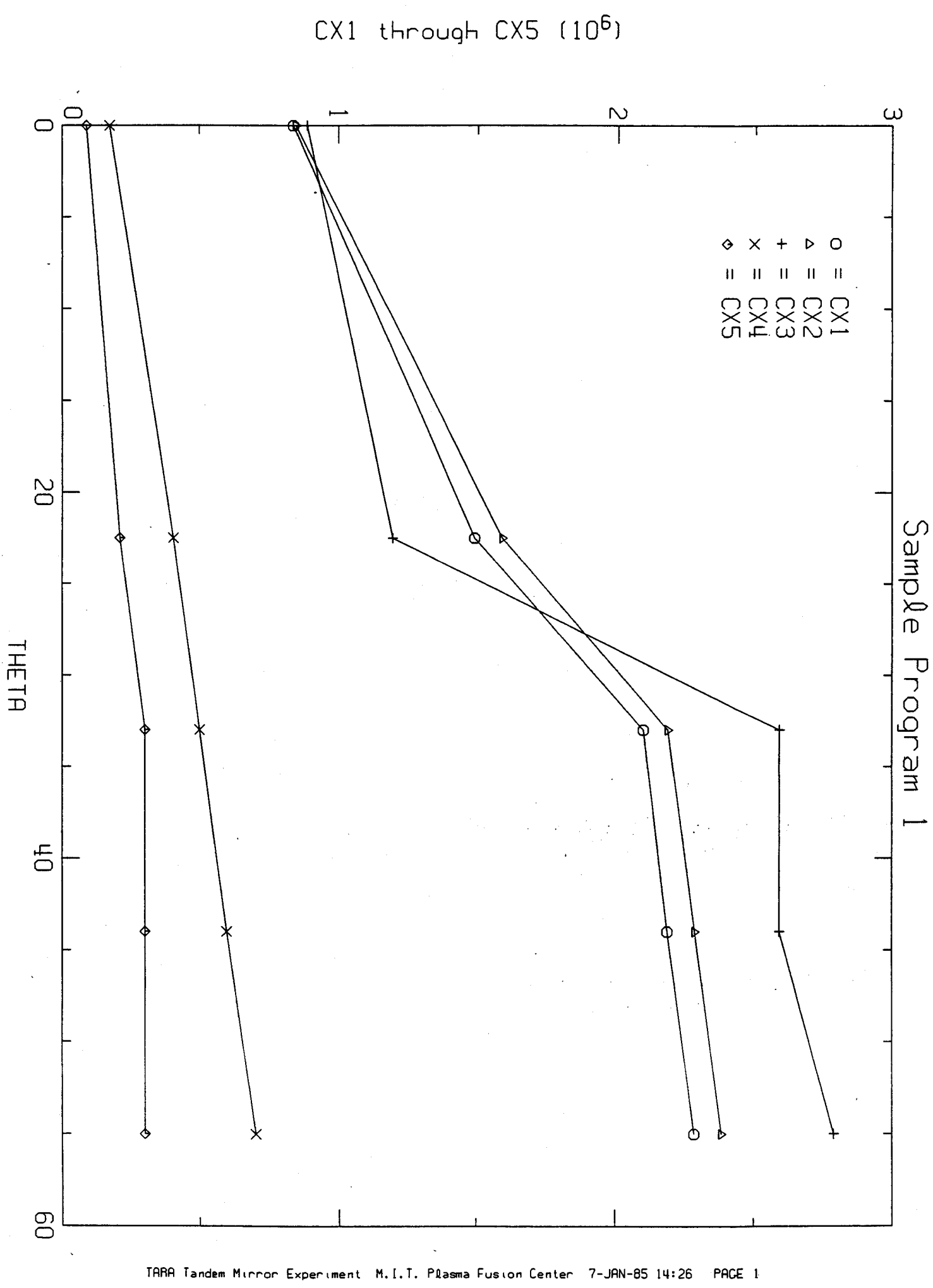
This chapter contains six sample FORTRAN programs and the plots which they produce to give the graphics package user an idea of what to expect. All the plots were made on a QMS Lasergrafix 1200 printer.

Program 1	Quick Plot of Several Curves
Program 2	Quick Text ViewGraph
Program 3	A Multi-panel Graph
Program 4	A Multi-region Illustration
Program 5	Isogram (contour map)
Program 6	An Orthogonal Projection of a Surface

3.1 QUICK PLOT: DATA

This example illustrates the use of PICTUR to draw five curves within one graph frame; each curve is identified by a different marker. The marker key is added by calling MRKTXT.

```
C ------------------------------------------------------------------
C
C                        QUICK PLOT: DATA
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP1
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      REAL CX1(10), CX2(10), CX3(10), CX4(10), CX5(10), THETA(10)
C INITIALIZE THE ARRAYS WITH SOME DATA
      DATA CX1/836800.,1490000.,2099999.,2190000.,2290000.,5*0.0/
      DATA CX2/846900.,1590000.,2190000.,2290000.,2390000.,5*0.0/
      DATA CX3/886900.,1190000.,2590000.,2590000.,2790000.,5*0.0/
      DATA CX4/175800.,402800.,500000.,600000.,700000.,5*0.0/
      DATA CX5/87880.,211100.,300000.,300000.,300000.,5*0.0/
      DATA THETA/0.0,22.5,33.0,44.0,55.0,5*0.0/
C NPTS    NUMBER OF DATA POINTS
      NPTS = 5
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 1'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C USE PICTUR TO DEFINE A PAGE, SET THE DOMAIN AND RANGE, DRAW THE FRAME,
C AND PLOT THE CURVES
      CALL PICTUR ( 'THETA', 'CX1 through CX5', 'Sample Program 1',
     1   THETA, CX1, NPTS, 1,
     2   THETA, CX2, NPTS, 2,
     3   THETA, CX3, NPTS, 3,
     4   THETA, CX4, NPTS, 4,
     5   THETA, CX5, NPTS, 5)
C USE MRKTXT TO WRITE A SYMBOL KEY IN UPPER LEFT CORNER
      CALL MRKTXT ( 1, ' = CX1', 0.1)
      CALL MRKTXT ( 2, ' = CX2', 0.1)
      CALL MRKTXT ( 3, ' = CX3', 0.1)
      CALL MRKTXT ( 4, ' = CX4', 0.1)
      CALL MRKTXT ( 5, ' = CX5', 0.1)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```


CX1 through CX5 (10^6)
Sample Program 1
THETA
0
20
40
60
0
1
2
3
o = CX1
▷ = CX2
+ = CX3
× = CX4
◇ = CX5

3.2 QUICK PLOT: TEXT

This example shows how to use character type variables and indirectly the versatility of the SYMBOL routine which ultimately writes all text.

```
C ------------------------------------------------------------------
C
C                        QUICK PLOT: TEXT
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                         VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP2
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      CHARACTER*132 CSTR(20)
      REAL CHFR(20)
C
C ------------------------------------------------------------------
C
C DEFINE SPECIAL CHARACTERS TO USE AS IN-STREAM COMMANDS
C IN CHARACTER STRINGS INTENDED FOR DISPLAY
C
C nul      NULL
C stx      START OF TEXT
C etx      END OF TEXT
C bs       BACKSPACE
C lf       LINE FEED
C luf      LINE UNFEED
C cr       CARRIAGE RETURN
C sup      SUPERSCRIPT MODE
C nor      NORMAL LINE MODE
C sub      SUBSCRIPT MODE
C can      CANCEL NEXT CHARACTER AS A COMMAND
C latin    LATIN CHARACTER FONT
C greek    GREEK CHARACTER FONT
C
      CHARACTER*(*) nul,stx,etx,bs,lf,luf,cr,sup,nor,sub,can,latin,greek
      PARAMETER (nul=CHAR(0),stx=CHAR(2),etx=CHAR(3),bs=CHAR(8),
     1   lf=CHAR(10),luf=CHAR(11),cr=CHAR(13),sup=CHAR(17),nor=CHAR(18),
     2   sub=CHAR(19),can=CHAR(24),latin=CHAR(28),greek=CHAR(29))
C ------------------------------------------------------------------
C INITIALIZE THE CHARACTER ARRAYS WITH SOME TEXT
C WITH AN INTERNAL WRITE AND A FORMAT STATEMENT
      FPN = 15.32
      WRITE ( CSTR, 900) FPN, FPN
  900 FORMAT ('Centered Title'//'Skip a line'/'Number = ',F6.2/
     1   'Number = ',1PE9.2)
C OR WITH DIRECT REPLACEMENT AND CONCANTENATION
      CSTR(9) = 'Another test line'
      CSTR(10) = 'A Greek letter lamda subscript 2:'//greek//
     1  ' L'//sub//'2'//nor//' I'//sub//'2'
      CSTR(12) = 'Marker 2 = '//can//CHAR(2)
C SET CHARACTER HEIGHT AND CENTERING
      DO 1 I=1,20
    1 CHFR(I) = 2.0
      CHFR(1) = -3.0
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 2'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
```

```
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C WRITE A PAGE OF TEXT
      CALL PICTEXT ( CSTR, 20, CHFR)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

Centered Title

Skip a line
Number = 15.32
Number = 1.53E+01

Another test line
A Greek letter lamda subscript 2: Λ_2 λ_2

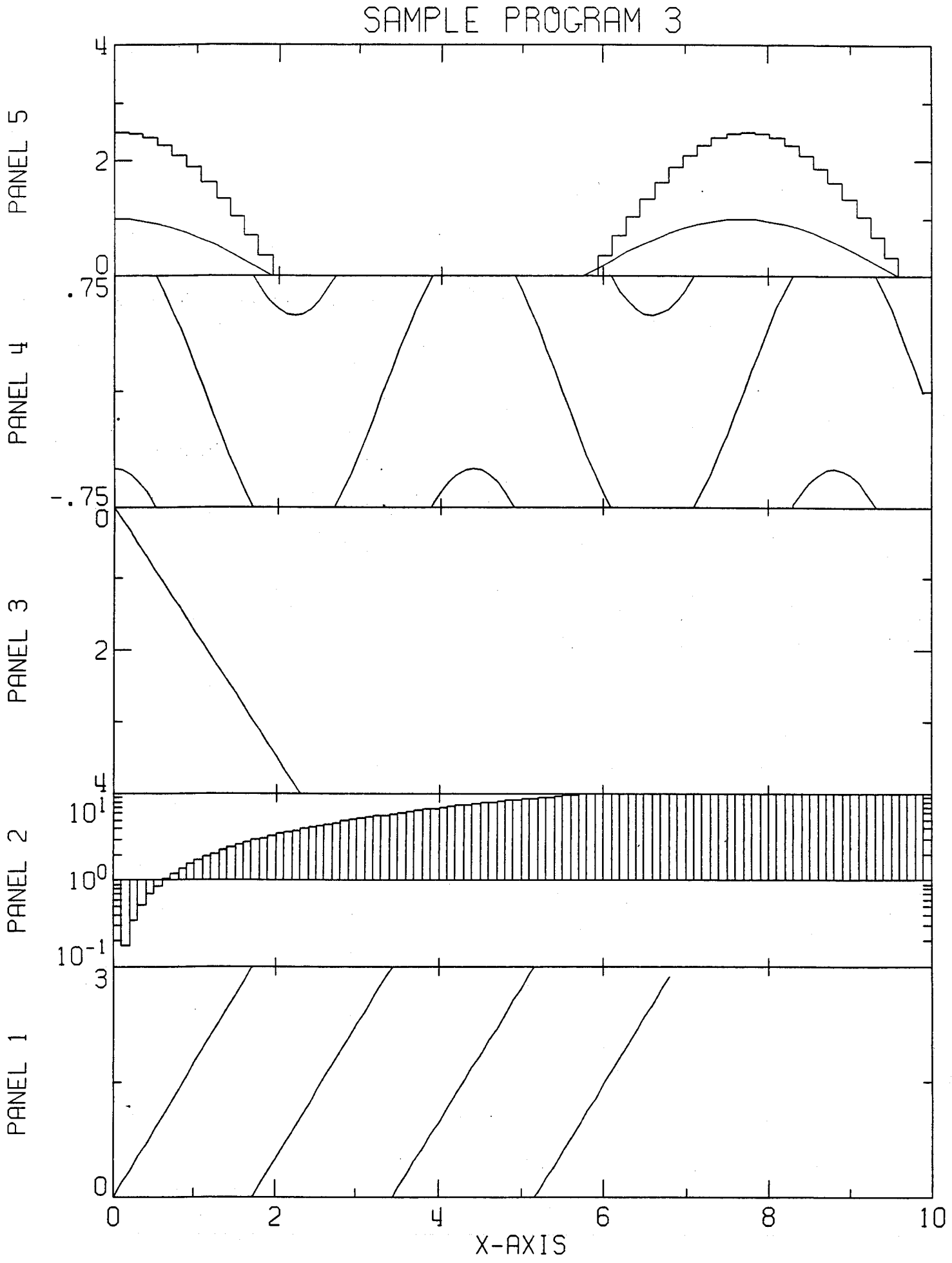
Marker 2 = △

3.3 MULTI-PANEL GRAPH

This example illustrates the use of MOSAIC to make a multipanel graph, the use of SYMRAY and HISTRA to draw curves, and the use of SETXAX and SETYAX to set domains and ranges.

```
C ------------------------------------------------------------
C
C                       MULTI-PANEL GRAPH
C
C
C                   GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------
C
      PROGRAM GPSP3
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      REAL X(100),Y(100),CS(100),R(100)
C COMPUTE VALUES FOR DATA ARRAYS
      DO 1 I=1,100
      X(I) = (I-1)/7.0
      Y(I) = (I-1)/10.0
      CS(I) = COS(X(I))
    1 R(I) = SQRT(X(I)*X(I)+Y(I)*Y(I))
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 3'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C A NEW PAGE WITH DEVICE DEPENDENT ASPECT RATIO
      CALL NUPAGE (,,,,,, .TRUE.)
      CALL SETXAX ( 0.0, 10.0, 0)
C FIRST PANEL; DRAW PANEL FRAME
      CALL SETYAX ( 0.0, 3.0, 0, -0.2)
      CALL MOSAIC ( 'X-AXIS', 'PANEL 1', 'SAMPLE PROGRAM 3')
C SOME CURVE WITH FOLDOVER; NOTE UP TO PLUS THREE FOLDINGS ALLOWED
      CALL SYMRAY ( Y, R, 0.14, 12, 22.5, 100, -1, -1, 3)
C DEFINE SECOND PANEL; DRAW A HISTOGRAM
      CALL SETYAX ( 0.1, 10.0, 2, -0.15)
      CALL SETGRD ( .TRUE., .FALSE.)
      CALL MOSAIC ( 'X-AXIS', 'PANEL 2', 'SAMPLE PROGRAM 3')
C HISTOGRAM WITH REFERENCE LINE
      CALL HISTRA ( Y, Y(2), R, R, 99, 1.0, 1)
C THIRD PANEL; DECREASING SCALE WITH INCREASING COORDINATE
      CALL SETYAX ( 4.0, 0.0, 0, -0.25)
      CALL MOSAIC ( 'X-AXIS', 'PANEL 3', 'SAMPLE PROGRAM 3')
      CALL SYMRAY ( Y, R,,, 100)
C FOURTH PANEL; (DEFAULT) FOLDOVER
      CALL SETYAX ( -0.75, 0.75, 0, -0.2)
      CALL MOSAIC ( 'X-AXIS', 'PANEL 4', 'SAMPLE PROGRAM 3')
      CALL SYMRAY ( Y, CS,,, 100, -1)
C FIFTH PANEL
      CALL SETYAX ( 0.0, 4.0, 0, -0.25)
C NOTE PANEL AS DEFINED IS TOO HIGH
C PANEL PLOTTED IS SHRUNK ACCORDINGLY (IN MOSAIC)
      CALL MOSAIC ( 'X-AXIS', 'PANEL 5', 'SAMPLE PROGRAM 3')
      CALL SYMRAY ( R, CS, 0.14, 10, 0.0, 100)
C CHANGE SCALE
      CALL SETYAX ( 0.0, 1.6, 0)
C ANOTHER CURVE IN SAME PANEL
      CALL HISTRA ( R, R(2), CS, CS, 99)
C SIXTH PANEL; DELETED AS OFF FRAME
```

```
      CALL SETYAX ( -1.0, 1.0, 0, -0.2)
C MOSAIC SHUTS OFF FURTHER PLOTTING AS GRAPH IS FULL
      CALL MOSAIC ( 'X-AXIS', 'PANEL 6', 'SAMPLE PROGRAM 3')
C DATA CONVERSION FAILS SO NOTHING PLOTTED
      CALL SYMRAY ( Y, CS,,,, 100)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```


SAMPLE PROGRAM 3
PANEL 5
4
2
0
PANEL 4
.75
0
-.75
PANEL 3
0
2
4
PANEL 2
10^1
10^0
10^{-1}
PANEL 1
3
0
0
2
4
6
8
10
X-AXIS

3.4 MULTI-REGION PAGE

This example illustrates the use of SETREG to define regions of a page and gives further examples of writing text. Data points with error bars are plotted using PLSIGDAT.

```
C ------------------------------------------------------------------
C
C                        MULTI-REGION PAGE
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP4
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      CHARACTER*40 CSTR(20)
      REAL X(100), Y(100), R(100)
      REAL XX(10), YY(10), XV(10), YV(10)
C COMPUTE VALUES FOR DATA ARRAYS
      DO 1 I=1,100
      X(I) = (I-1)/7.0
      Y(I) = (I-1)/10.0
    1 R(I) = SQRT(X(I)*X(I)+Y(I)*Y(I))
      DO 2 I=1,10
      XX(I) = I
      XV(I) = 0.5
      YY(I) = 2.0**I
    2 YV(I) = SQRT(YY(I))
C PREPARE SOME TEXT FOR OUTPUT
      CSTR(1) = 'A region may be used for text'
      CSTR(3) = 'Any annotation may be written'
      CSTR(4) = 'Including markers and greek'
      CSTR(6) = CHAR(24)//CHAR(2)//' '//CHAR(24)//CHAR(3)//
     1   ' '//CHAR(24)//CHAR(4)//' ...'
      CSTR(8) = CHAR(29)//' A, B, C, ..., W'
      CSTR(11) = 'Also sub/super-scripts etc.'
      CSTR(13) = 'He'//CHAR(17)//'4'//CHAR(18)//', C'//CHAR(19)//
     1   '2'//CHAR(18)//'H'//CHAR(19)//'2'//CHAR(17)//'+'//CHAR(18)//
     2   ', ...'
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 4'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C MULTI-REGION PAGE
      CALL NUPAGE
      CALL PAGTITL ( 'Sample Program 4 : Multi-Region Page')
C DEFINE FIRST REGION OF SCREEN ( lower left quadrant)
      CALL SETREG ( 1,1, 2,2)
      CALL SETGRD ( .TRUE., .TRUE.)
      CALL REGPIX ( 'X-AXIS', 'Y-AXIS', 'REGION-1', X, Y, 100)
C DEFINE A SECOND REGION OF SCREEN ( upper left quadrant)
      CALL SETREG ( 1,2)
      CALL REGTEXT ( CSTR, 20)
C DEFINE A THIRD REGION OF SCREEN ( right half)
      CALL SETREG ( 2,1, 2,1)
      CALL BOUNDS ( 0.0, 15.0, 0, 0.0, 10.0, 0)
C DEFINE A PANEL HALF THE GRAPH HEIGHT
      CALL MOSAIC ( 'X-AXIS', 'PANEL_1', ' ', 0.5)
      CALL SYMRAY ( X, Y,, 2,, 100, -3,,, 10)
```

```
C ANOTHER PANEL LIKE THE FIRST
      CALL AUTYST ( YY, 10, 2)
      CALL MOSAIC ( 'X-AXIS', 'PANEL_2', 'GRAPH HEADING')
      CALL PLSIGDAT ( XX, XV, YY, YV, -10)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

Sample Program 4 : Multi-Region Page

A region may be used for text

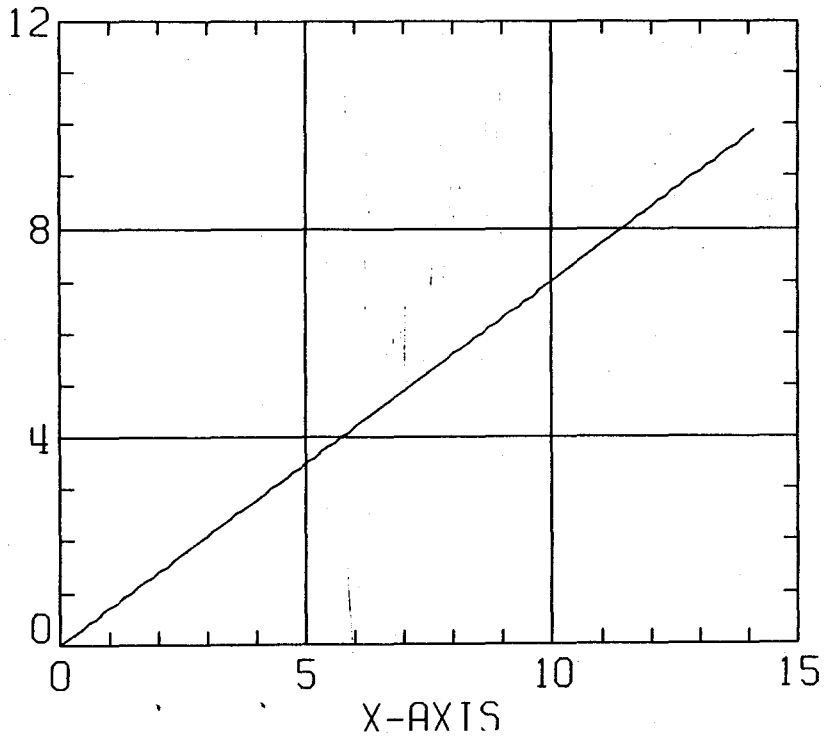
Any annotation may be written
Including markers and greek

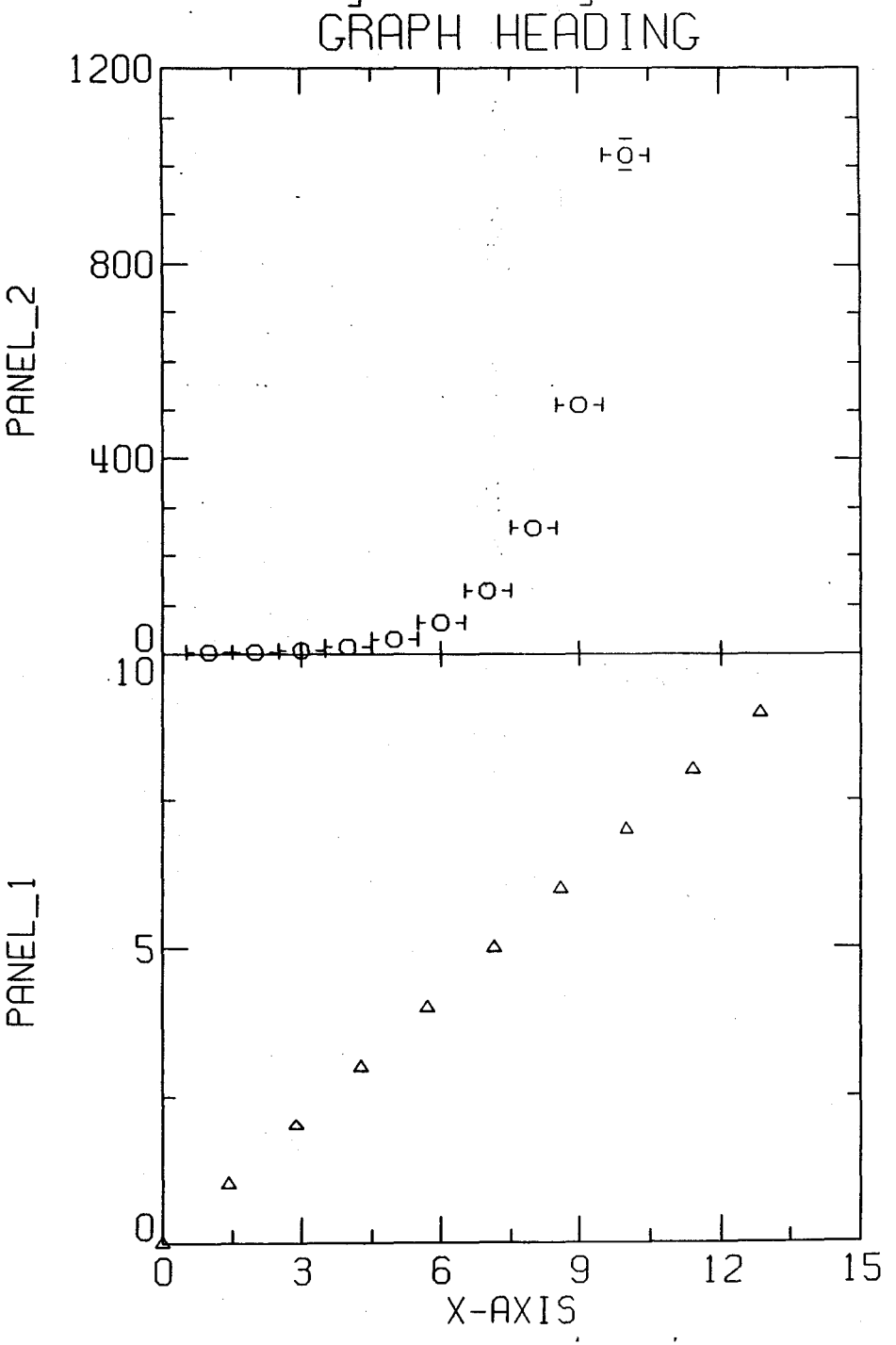
△ + × ...

A, B, H, ..., Ω

Also sub/super-scripts etc.

He^4, $C_2H_2^+$, ...

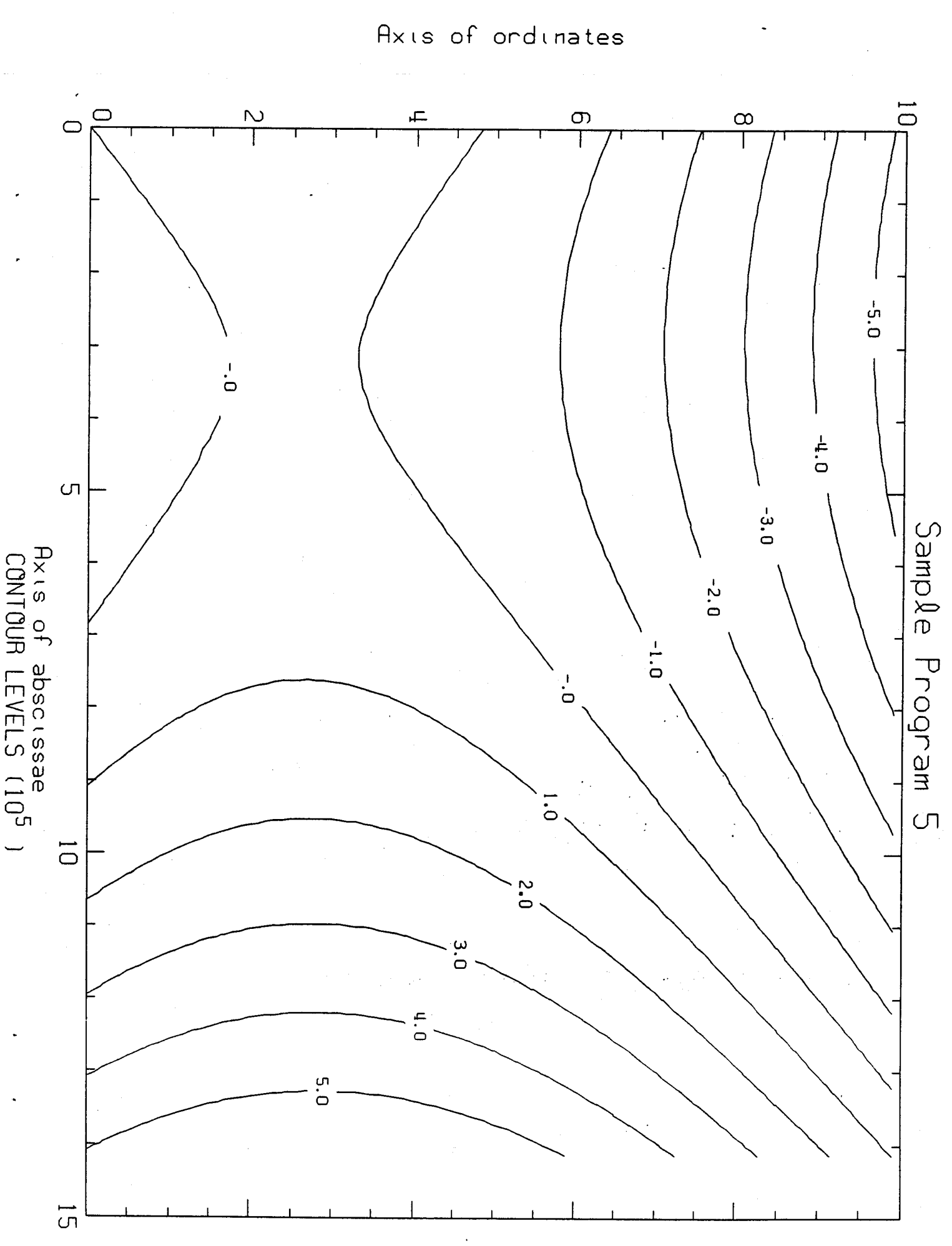




3.5 ISOGRAM

This example uses FNCON to draw an isogram with labeled isopleths; the same surface is represented in sections 3.6, 5.2, and 5.4.

```
C ------------------------------------------------------------------
C
C                        ISOGRAM MAP
C
C
C                   GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP5
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      REAL X(100),Y(100),RA(100,100)
C COMPUTE VALUES FOR THE DATA ARRAYS
      DO 1 I=1,100
      X(I) = (I-1)/7.0
      Y(I) = (I-1)/10.0
      DO 1 J=1,100
    1 RA(I,J) = ((I-25)**2-(J-25)**2+I*J/10)*100
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 5'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
      CALL NUPAGE
C AUTOMATIC AXIS SETTING FOR ARRAYS
      CALL AUTXST ( X, 100)
      CALL AUTYST ( Y, 100)
C MOSAIC MAY BE USED INSTEAD OF GRFRAM
      CALL MOSAIC ( 'Axis of abscissae', 'Axis of ordinates',
     1   'Sample Program 5')
C DRAW DEFAULT ISOPLETHS OF THE ARRAY RA AND LABEL THE VALUES
      CALL FNCON ( X, Y, RA, 100)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

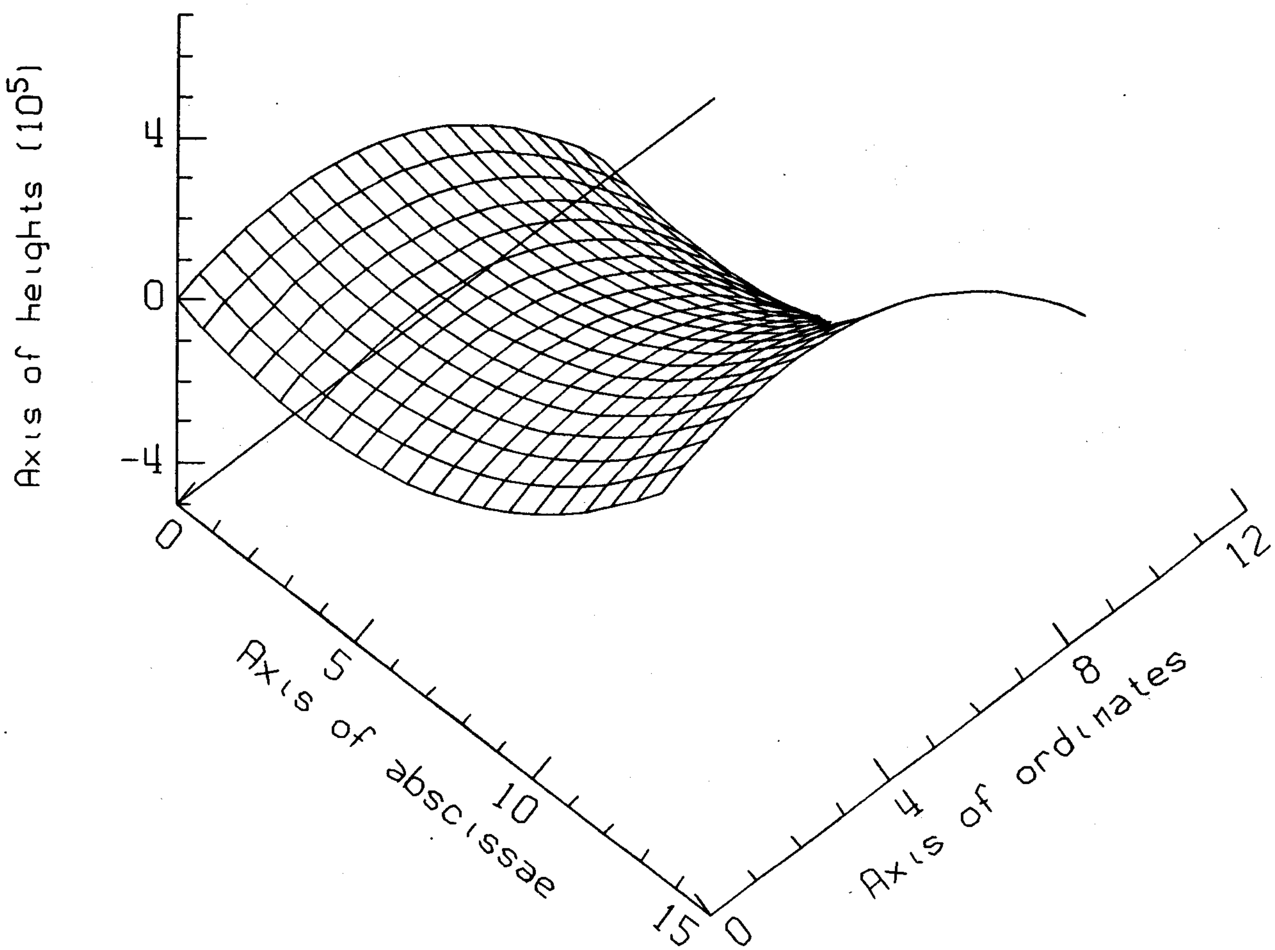

Axis of ordinates
0
2
4
6
8
10
-5.0
-4.0
-3.0
-2.0
-1.0
-.0
-.0
1.0
2.0
3.0
4.0
5.0
0
5
10
15
Axis of abscissae
CONTOUR LEVELS (10^5)
Sample Program 5

3.6 ORTHOGONAL PROJECTION OF A SURFACE

This example illustrates projecting a surface orthogonally; the same surface is represented in sections 3.5, 5.2, and 5.4.

```
C ------------------------------------------------------------------
C
C             AN ORTHOGONAL PROJECTION OF A SURFACE
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                      VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP6
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      PARAMETER ( npts=20)
      REAL X(npts), Y(npts), RA(npts,npts)
C COMPUTE VALUES FOR THE DATA ARRAYS
      DO 1 I=1,npts
      II = (I-1)*100/npts+1
      X(I) = (II-1)/7.0
      Y(I) = (II-1)/10.0
      DO 1 J=1,npts
      JJ = (J-1)*100/npts+1
    1 RA(I,J) = ((II-25)**2-(JJ-25)**2+II*JJ/10)*100
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample Program 6'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
      CALL NUPAGE
C SELECT A 3-D PROJECTION: ORTHOGONAL
      CALL SET3D
C AUTOMATIC AXIS SETTING FOR ARRAYS
      CALL AUTXST ( X, npts)
      CALL AUTYST ( Y, npts)
      CALL AUTZST ( RA, npts*npts)
C DRAW THE GRAPH FRAME
      CALL FRAM3D ( 'Axis of abscissae', 'Axis of ordinates',
     1   'Axis of heights', 'Sample Program 6')
C DRAW PROJECTION OF THE ARRAY RA
      CALL THREED ( X, Y, RA, npts)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

Sample Program 6



Orthogonal Projection: theta = 40.0, phi = 315.0

CHAPTER 4

SUBPROGRAMS 2

The conventions described in the introduction to chapter 2 apply in this chapter.

4.1 INTERACTIVE CONTROL

Interactive control of a program may be done in two ways. After each page is complete a 4 character word may be returned from ENDPAG (or ENDPLT); these routines were described in chapter 2. Graphic output may be interrupted for interactive control by using the routine PAGPAUS (or PLTMSG).

4.1.1 PAGPAUS

Purpose: To permit interactive control in mid-graph.

CALL PAGPAUS (PROMPT, REPLY)

PROMPT	I	CHARACTER VARIABLE WITH THE PROMPT MESSAGE
REPLY	O	CHARACTER VARIABLE WITH THE INTERACTIVE REPLY

Graphic output is interrupted and the prompt written on the screen. After the reply has been entered, both the prompt and the reply will be erased and plotting is continued from where it left off.

4.2 DATA CONVERSION

These are the coordinate transformation subprograms; their use is implicit in many of the subprograms described in chapter 2. The following program fragment shows how they might be used (assuming the domain and range are defined) to only connect consecutive points if both are within the graph frame while displaying every point within the graph frame.

```
C ASSUME IN A LOOP WITH INDEX I AND IPEN = 3 TO START
      CALL DATCON ( XRAY(I), YRAY(I), XL, YL, INGRAF)
      CALL GLTRAN ( XL, YL, XP, YP, INGRAF)
      IF ( INGRAF.EQ.1) THEN
        CALL PLOT ( XP, YP, IPEN)
        IF ( IPEN.EQ.3) CALL PLOT ( XP, YP, 2)
        IPEN = 2
      ELSE
        IPEN = 3
      END IF
```

Direct inversion of the transformations is not implemented; however, the transformations may be undone by using DATSET, GLTSET, and PJTSET to get the transformation definitions and then inverting.

4.2.1 DATCON

Purpose: To convert from subject space to linear space

```
CALL DATCON ( XS,YS,XL,YL,INGRAF) !or

CALL DATCON ( XS,YS,ZS,XL,YL,ZL,INGRAF)
```

XS,YS,ZS	I	SUBJECT SPACE COORDINATES
XL,YL,ZL	O	LINEAR SPACE COORDINATES
INGRAF	O	DATA CONVERSION FLAG (+1 denotes success; -1 failure)

An external conversion routine may be supplied see GENCON. Failure of the conversion will occur if a "fill" value is encountered or if on a logarithmic axis, the logarithim of a non positive number should be computed.

4.2.1.1 SETDAT - Purpose: To set conversion constants in DATCON.

```
CALL SETDAT ( XA,XB,IX,GX, YA,YB,IY,GY [,ZA,ZB,IZ,GZ])
```

XA	LOWER BOUND
XB	UPPER BOUND
IX	LINEAR/LOGARITHMIC FLAG (0=LINEAR; 1=LOGARITHMIC)
GX	AXIS LENGTH (LINEAR SPACE)

SIMILARLY (YA,YB,IY,GY) AND (ZA,ZB,IZ,GZ)

4.2.1.2 GENCON - Purpose: To define an external conversion routine for DATCON.

```
CALL GENCON ( lext, xyzcon)
```

LEXT	LOGICAL FLAG FOR EXTERNAL FUNCTION (default .FALSE.)
XYZCON	EXTERNAL SUBROUTINE WITH SAME CALLING STRING AS DATCON WITHOUT INGRAF (note that DATCON still converts after XYZCON)

4.2.2 GLTRAN

Purpose: To provide a general linear transform from linear to page coordinates

```
CALL GLTRAN ( XL,YL,XP,YP,INGRAF) !or

CALL GLTRAN ( XL,YL,ZL,XP,YP,INGRAF,zp) !or

CALL GLTRAN ( XP,YP,INGRAF)
```

XL,YL,ZL	I	LINEAR SPACE COORDINATES
XP,YP,ZP	O	PAGE COORDINATES (ZP IS USEFUL FOR TELLING ON WHICH SIDE OF THE PAGE A POINT LIES)
INGRAF	I/O	IN GRAPH FLAG (+1 denotes in panel; 0 off panel) (ON INPUT THIS FLAG SHOULD HAVE A VALUE OF 1 IT IS ASSUMED SET BY DATCON)

4.2.2.1 SETGLT - Purpose: To define the linear transformation

```
CALL SETGLT ( tran, voff) !or

{CALL SETGLT ( XBEG,YBEG,XEND,YEND)}
```

TRAN(9)	TRANSFORMATION MATRIX (DEFAULT IDENTITY)
VOFF(2)	OFFSET VECTOR (DEFAULT NULL VECTOR)

XBEG,YBEG PANEL BOUNDARIES IN OBJECT SPACE
XEND,YEND

The coordinate triplets are taken to be column vectors so that the matrix follows normal FORTRAN conventions; if considered to be 3x3, the first index is the row index and the second the column index, e.g.

```
XP = TRAN(1)*XL + TRAN(4)*YL + TRAN(7)*ZL + VOFF(1)
```

4.2.2.2 FOLDOR - Purpose: To compute foldover intercepts for y-folding

```
CALL FOLDOR ( INX, INY, Y0IN, XI, YI1, YI2)
```

INX	X fold range; 0 is original panel
INY	Y fold range
Y0IN	folded Y value in panel
YI1	Y range cross out
YI2	Y range cross in

All arguments are output from this routine which should be called after GLTRAN.

4.2.2.3 YCLIPT - Purpose: To compute clipping within panel.

```
CALL YCLIPT ( LGRAF,NGRAF,X1,Y1,X2,Y2,MGRAF)
```

LGRAF	I	LAST INGRAF FLAG
NGRAF	I	CURRENT INGRAF FLAG
X1,Y1	I/O	PAGE COORDINATES OF THE PREVIOUS POINT INTERCEPT POINT 1
X2,Y2	O	INTERCEPT POINT 2
MGRAF	O	NUMBER OF INTERCEPT POINTS

This routine should be called after GLTRAN which saves the page coordinates of the current point.

4.3 CURVE DRAWING 3

4.3.1 ELLIPS

Purpose: To draw an arc of an ellipse (circle)

```
CALL ELLIPS ( xc, yc, A, b, phi, phi0, phi1)
```

XC	ABSCISSA OF ELLIPSE CENTER IN SUBJECT SPACE (DEFAULT 0.0)
YC	ORDINATE OF ELLIPSE CENTER IN SUBJECT SPACE (DEFAULT 0.0)
A	SEMIMAJOR AXIS IN ABSCISSA UNITS
B	SEMIMINOR AXIS IN ORDINATE UNITS (DEFAULT A)
PHI	ANGLE OF ROTATION OF THE ELLIPSE (DEFAULT 0.0)
PHI0	ANGLE OF BEGINNING OF THE ARC (DEFAULT -180.0)
PHI1	ANGLE OF ENDING OF THE ARC (DEFAULT 180.0)

4.3.2 PLTGLT

Purpose: To provide PLOT in 3-D linear coordinates

```
CALL PLTGLT ( XL, YL, ZL, IPEN, ingraf, inp)
```

XL	I	LINEAR SPACE X-COORDINATE
YL	I	LINEAR SPACE Y-COORDINATE
ZL	I	LINEAR SPACE Z-COORDINATE
IPEN	I	DRAW/MOVE FLAG (2/3)
INGRAF	O	POINT IN GRAPH FLAG (1 POINT WITHIN)
INP	I	NUMBER OF THE PEN TO BE USED IF THE POINT IS BEHIND THE DISPLAY PLANE (DEFAULT CURRENT PEN NUMBER)

The pen is changed before and after the move.

4.3.3 SYMGLT

Purpose: To provide SYMBOL in 3-D linear coordinates

CALL SYMGLT (XL, YL, ZL, ch, IBCD, angle, NCHAR, ingraf, inp)

XL	I	LINEAR SPACE X-COORDINATE
YL	I	LINEAR SPACE Y-COORDINATE
ZL	I	LINEAR SPACE Z-COORDINATE
CH	I	CHARACTER HEIGHT IN PAGE COORDINATES
IBCD	I	BYTE STRING OF TEXT OR INTEGER EQUIVALENT FOR A MARKER
ANGLE	I	ANGLE IN DEGREES
NCHAR	I	LENGTH OF IBCD OR -ICODE
INGRAF	O	POINT IN GRAPH FLAG (1 POINT WITHIN)
INP	I	NUMBER OF THE PEN TO BE USED IF THE POINT IS BEHIND THE DISPLAY PLANE (DEFAULT CURRENT PEN NUMBER)

The pen is changed before and after the move.

4.4 EXTENSION OF BASIC ROUTINES

Variations on NUMBER, SNUMBR, and SYMBOL to allow the reference to be either the lower right corner (suffix RiT), the center (suffix CeN), or the end of the last string (suffix CoN). The ...CoN routines are not needed in version 4; the same action may be obtained directly with the basic routines. All of these routines use page coordinates.

4.4.1 LABSYM

Purpose: To center and to plot a text string on a fixed length line

CALL LABSYM (XP,YP,AXLEN,IBCD,THETA,NBCD,ch,chused)

XP,YP	I	PAGE COORDINATES
AXLEN	I	AXIS LENGTH
IBCD	I	BYTE TEXT STRING
THETA	I	ANGLE OF LINE
|NBCD|	I	NUMBER OF CHARACTERS TO CENTER IF NEGATIVE STRING RUN BACKWARDS AND UPSIDE DOWN
CH	I	CHARACTER HEIGHT (IF ABSCENT USES PAGE VALUE; IF NEGATIVE -CHFRACT OF PAGE VALUE)
CHUSED	O	CHARACTER HEIGHT ACTUALLY USED

The height of the characters will be diminished in order to squeeze all of the text string into the line.

4.4.2 NUMCEN

CALL NUMCEN (XP,YP,CH,FPN,THETA,NDIG)

4.4.3 NUMCON

CALL NUMCON (FPN,NDIG)

4.4.4 NUMRIT

```
CALL NUMRIT ( XP,YP,CH,FPN,THETA,NDIG)
```

4.4.5 SNUMCN

```
CALL SNUMCN ( FPN,NDIG)
```

4.4.6 SNUMRT

```
CALL SNUMRT ( XP,YP,CH,FPN,NDIG)
```

4.4.7 SYMCEN

```
CALL SYMCEN ( XP,YP,CH,BSTR,THETA,+NCHAR)
```

4.4.8 SYMCON

```
CALL SYMCON ( BSTR,+NCHAR)
```

4.4.9 SYMRIT

```
CALL SYMRIT ( XP,YP,CH,BSTR,THETA,+NCHAR)
```

4.5 UTILITY 2

These subprograms are not graphic subprograms consequently there is no restriction on their use.

4.5.1 FNCALN

Purpose: To call an external function with a variable number of arguments.

```
= FNCALN( %val(FTNADR),ARG1,arg2,arg3,arg4,arg5, ...,argf)
```

(VAX specific function)

4.5.2 NEWARG

Purpose: To determine the presence of an argument in the calling sequence.

```
= NEWARG( IARG)
```

IARG	I	INDEX OF ARGUMENT IN THE CALLING SEQUENCE
NEWARG	O	=1 ARGUMENT GIVEN =0 ARGUMENT OMITTED

(VAX specific function in MACRO)

4.5.3 NOPARI

Purpose: To compute non-parallel intercepts of two line segments in a plane.

CALL NOPARI (A0,B0,A1,B1,X0,Y0,X1,Y1,U,V,I)

A0,B0	I	FIRST LINE SEGMENT
A1,B1		
X0,Y0	I	SECOND LINE SEGMENT
X1,Y1		
U,V	O	COORDINATES OF INTERCEPT
I	O	INTERCEPT FLAG
		+1 INTERCEPT WITHIN BOTH SEGMENTS
		0 PARALLEL
		-1 INTERCEPT EXTERNAL TO AT LEAST ONE SEGMENT

4.5.4 NUMP

Purpose: To find the number of arguments in a calling sequence.

CALL NUMP (NARG) or

= NUMP(NARG)

NARG	O	NUMBER OF ARGUMENTS IN CALLING SEQUENCE
=NUMP	O	NUMBER OF ARGUMENTS IN CALLING SEQUENCE

(VAX specific subprogram in MACRO)

4.5.5 NUMSTR

Purpose: To convert a real number to a byte text string

CALL NUMSTR (FPN,NDIG,STR,ILEN)

FPN	I	REAL NUMBER
NDIG	I	NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
		(-1 SUPPRESSES DECIMAL POINT)
STR	O	BYTE STRING
ILEN	O	NUMBER OF BYTES

Fill is converted to the string "FILL".

4.5.6 SBCALN

Purpose: To call an external subroutine with a variable number of arguments

CALL SBCALN (%val(SUBRADR),arg1,arg2,arg3,arg4,arg5, ...,argf)

(VAX specific routine)

4.6 SCALING

4.6.1 AUSCAL

Purpose: To choose all the axis parameters given the extreme data values.

CALL AUSCAL (WMIN,WMAX,MAXINT,WA,WB,IW,NW,LW,WFT,DWLT)

WMIN	I	DATA MINIMUM
WMAX	I	DATA MAXIMUM
MAXINT	I	MAXIMUM NUMBER OF INTERVALS
WA	O	AXIS LOWER BOUND

WB	O	AXIS UPPER BOUND
IW	O	LINEAR/LOGARITHMIC AXIS FLAG
NW	O	NUMBER OF MINOR INTERVALS
LW	O	NUMBER OF MAJOR INTERVALS
WFT	O	VALUE AT FIRST MAJOR TIC
DWLT	O	DELTA VALUE BETWEEN MAJOR TICS

The break points between linear and logarithmic scales and linear with and without an extremum of zero are set in SETSCAL q.v.

4.6.2 GPSCAL

Purpose: To pick limit values for a linear scale.

CALL GPSCAL (AMIN, AMAX, MAJINT, FIRSTV, ENDV, minint)

AMIN	I	MINIMUM DATA VALUE
AMAX	I	MAXIMUM DATA VALUE
MAJINT	I	NUMBER OF MAJOR INTERVALS
FIRSTV	O	ADJUSTED FIRST VALUE
ENDV	O	ADJUSTED LAST VALUE
MININT	O	NUMBER OF MINOR INTERVALS

4.6.3 SCALDN

Purpose: To scale a number down

CALL SCALDN (W, INT, WW, NW, LW)

W	I	REAL NUMBER
INT	I	MAXIMUM NUMBER OF MAJOR INTERVALS
WW	O	SCALED NUMBER
NW	O	RECOMMENDED NUMBER OF MINOR INTERVALS
LW	O	RECOMMENDED NUMBER OF MAJOR INTERVALS

4.6.4 SCALUP

Purpose: To scale a number up to a convenient value

CALL SCALUP (W, INT, WW, NW, LW)

W	I	REAL NUMBER
INT	I	MAXIMUM NUMBER OF MAJOR INTERVALS
WW	O	SCALED NUMBER
NW	O	RECOMMENDED NUMBER OF MINOR INTERVALS
LW	O	RECOMMENDED NUMBER OF MAJOR INTERVALS

4.7 NUMBER FORMATING

These routines are used in axis labeling.

4.7.1 FPNORM

Purpose: To compute normalization of a real number

CALL FPNORM (FPN, AF, KEXP)

FPN	I	REAL NUMBER
AF	O	ABSOLUTE VALUE NUMBER [1.0,10.0)
KEXP	O	POWER OF 10

```
FPN = SIGN( AF * 10.0**KEXP, FPN)
```

4.7.2 NCHAR

Purpose: To compute the number of characters output by NUMBER

```
= NCHAR( FPN, NDIG)

 FPN      FLOATING POINT NUMBER TO BE OUTPUT
 NDIG     NUMBER OF DIGITS TO RIGHT OF DECIMAL POINT
```

4.7.3 NDIGIT

Purpose: To determine the number of digits in axis annotation

```
CALL NDIGIT ( LF,NFMAX,DF,FA,FB,NF,KF)

 LF       I   NUMBER OF LABELED INTERVALS
 NFMAX    I   MAXIMUM NUMBER OF CHARACTERS TO BE DISPLAYED
 DF       I/O DELTA BETWEEN LABELED INTERVALS
 FA       I/O AXIS START VALUE
 FB       I/O AXIS END VALUE
 NF       O   NUMBER OF DIGITS
 KF       O   REMOTE EXPONENT
```

4.7.4 NDIGT

Purpose: To compute number of digits till repeating or termination or until maximum number of digits exceeded

```
= NDIGT( FPN, NFMAX, LF)

 FPN      I   REAL NUMBER
 NFMAX    I   MAXIMUM NUMBER OF CHARACTERS TO BE DISPLAYED
 LF       I   NUMBER OF LABELED INTERVALS
```

4.8 FRAME DRAWING 2

All of these subprograms use page coordinates.

4.8.1 AXLINE

Purpose: To draw a line with tic marks

```
CALL AXLINE ( XP,YP,THETA,AXLEN,NWMIN,DWMAJ,DELW)

 XP       PAGE ABSCISSA FOR AXIS START
 YP       PAGE ORDINATE FOR AXIS START
 THETA    ANGLE OF ROTATION (DEGREES) ABOUT ( XP,YP)
 AXLEN    LENGTH OF LINE
 NXMIN    MINOR TIC FLAG
          = 0  LOGARITHMIC
          | |  LINEAR; NUMBER OF MINOR INTERVALS
          < 0  NO SEMI-MAJOR TIC MARKS
 DWMAJ    LENGTH OF A MAJOR INTERVAL
          > 0  LOG AXIS ASCENDING
          < 0  LOG AXIS DESCENDING
 DELW     OFFSET TO FIRST MAJOR TIC
```

4.8.2 BNDARY

Purpose: To draw a border about the current region.

CALL BNDARY

There are no arguments; the border surrounds the available plotting space for the graph and consequently if called after page and region annotation will exclude them.

4.8.3 GRIDX

Purpose: To draw grid lines parallel to the x-axis

CALL GRIDX (DELY, DYMAJ, GPL, GPH)

DELY	ORDINATE VALUE OF FIRST TIC
DYMAJ	ORDINATE INTERVAL BETWEEN MAJOR INTERVALS
GPL	GRAPH PANEL LENGTH
GPH	GRAPH PANEL HEIGHT

4.8.4 GRIDY

Purpose: To draw grid lines parallel to the y-axis

CALL GRIDY (DELX, DXMAJ, GPL, GPH)

DELX	ABSCISSA VALUE OF FIRST TIC
DXMAJ	ABSCISSA INTERVAL BETWEEN MAJOR INTERVALS
GPL	GRAPH PANEL LENGTH
GPH	GRAPH PANEL HEIGHT

4.9 PARAMETER SETTINGS

Values of miscellaneous graph parameters; all arguments are ouput. In general refer to the corresponding set-up routine for the argument descriptions.

4.9.1 BASSET

Purpose: To return number base, round, fill, etc.

CALL BASSET (base, round, fill, dr, pi)

BASE	NUMBER BASE
ROUND	MACHINE ROUNDING FACTOR
FILL	VALUE FOR MISSING SUBJECT SPACE COORDINATES
DR	CONVERSION FACTOR FROM DEGREES TO RADIANS
PI	ACOS(-1.0)

This routine insures that the entire graphics package uses the same constants.

4.9.2 DATSET

Purpose: To return conversion constants from DATCON.

CALL DATSET (xa,xb,ix,gx, ya,yb,iy,gy, za,zb,iz,gz)

This conversion is from user to linear space coordinates.

4.9.3 DVLSET

Purpose: To return device limits

CALL DVLSET (DVXL, DVYL[, DVXLM, DVYLM])

4.9.4 GLTSET

Purpose: To return GLTRAN settings

CALL GLTSET (T, VOFF) !or

CALL GLTSET (XBEG, YBEG, XEND, YEND)

```
 T(9)    TRANSFORMATION MATRIX FROM LINEAR SPACE TO OBJECT SPACE
 VOFF(2) OFFSET VECTOR IN OBJECT SPACE

 XBEG    .
 YBEG    .OBJECT SPACE WINDOW BOUNDARIES
 XEND    .
 YEND    .
```

This transformation is from linear space to page coordinates.

4.9.5 GRDSET

Purpose: To return grid flags

CALL GRDSET (lparlx, lparly, inp, inc3d)

4.9.6 GRFSET

Purpose: To return characteristics of top and bottom tic marks.

CALL GRFSET (CH,TL,TLB,TLT,DTIC,DTICB,DTICT)

4.9.7 GTITSET

Purpose: To control the drawing of the graph title and comment line

CALL GTITSET (LGTIT)

```
 LGTIT   LOGICAL FLAG CONTROLLING GRAPH TITLE AND COMMENT LINE
```

4.9.8 NXTSYM

Purpose: To return position of next character to be output

CALL NXTSYM (xp,yp,ch,theta,snth,csth)

4.9.9 P3DSET

Purpose: To return 3D parameters

CALL P3DSET (iproj,theta,phi,mode,rx,ry,rz,jpn,iq,sx,sy,sz)

```
 IQ      NUMBER OF THE OCTENT
 SX      LENGTH OF THE X-AXIS ON THE PAGE
 SY      LENGTH OF THE Y-AXIS ON THE PAGE
 SZ      LENGHT OF THE Z-AXIS ON THE PAGE
```

4.9.10 PAGSET

Purpose: To return constants from NUPAGE and SETPAG.

CALL PAGSET (pw, ph, gl, gh, plm, pbm, lrot, chp, tl, chsc, npag)

PW	PAGE WIDTH
PH	PAGE HEIGHT
GL	GRAPH LENGTH
GH	GRAPH HEIGHT
PLM	PAGE LEFT MARGIN
PBM	PAGE BOTTOM MARGIN
CHP	CHARACTER HEIGHT FOR PAGE
TL	TIC LENGTH
CHSC	CHARACTER HEIGHT SCALE
NPAG	PAGE NUMBER OF THE LAST PAGE

4.9.11 PANSET

Purpose: To return panel settings

CALL PANSET (gpl,gph,xop,yop,ch)

GPL	GRAPH PANEL LENGTH
GPH	GRAPH PANEL HEIGHT
XOP	ABSCISSA OF PANEL ORIGIN IN PAGE COORDINATES
YOP	ORDINATE OF PANEL ORIGIN IN PAGE COORDINATES
CH	CHARACTER HEIGHT IN PAGE COORDINATES

4.9.12 PJTSET

Purpose: To return the plot joint transformation

CALL PJTSET (TJ, VJ)

TJ(4)	JOINT TRANSFORMATION (SETAFF+SETROT+FACTOR)
VJ(2)	OFFSET OF JOINT TRANSFORMATION

The joint transformation goes from page to device coordinates.

4.9.13 PLOSET

Purpose: To return window definition

CALL PLOSET (XPMIN,YPMIN,XPMAX,YPMAX)

Minimum and maximum limits of plotting window

4.9.14 USRSET

CALL USRSET (CTLTXT)

CTLTXT CUTLINE TEXT CHARACTER STRING

4.9.15 XAXSET

CALL XAXSET (XA,XB,IX,GX,NX,LX,JX,IXT,IYT,XFT,DXLT)

4.9.16 YAXSET

```
CALL YAXSET ( YA,YB,IY,GY,NY,LY,JY,IYL,IYR,YFT,DYLT)
```

4.9.17 ZAXSET

```
CALL ZAXSET ( ZA,ZB,IZ,GZ,NZ,LZ)
```

4.10 SURFACE REPRESENTATION 2

These routines are used internally by various of the 3D subprograms.

4.10.1 HLRHVS

Purpose: To check for hidden lines

```
CALL HLRHVS ( H,V,LV,NR,IN,IX,IC,JN,JX,JC)
```

```
 H       I    2D ARRAY OF HORIZONTAL COORDINATES
 V       I    2D ARRAY OF VERTICAL COORDINATES
 LV      I/O  2D FLAG ARRAY
              =2   INTERPOLATED VALUE
              =1   IN RANGE VALUE
              =0   OUT OF RANGE VALUE
              =-1  CONVERSION FAILURE
              <-1  HIDDEN POINT
 NR      I    ROW DIMENSION
 IN      I    .
 IX      I    .DO LOOP RANGE FOR X-VARIABLE
 IC      I    .
 JN      I    -
 JX      I    -DO LOOP RANGE FOR Y-VARIABLE
 JC      I    -
```

The algorithm depends on no horizontal component to z-axis values.

4.10.2 PLTHVS

Purpose: To plot a ruled surface

```
CALL PLTHVS ( H,V,LV,NR,IN,IX,IC,JN,JX,JC)
```

```
 H       2D ARRAY OF HORIZONTAL COORDINATES
 V       2D ARRAY OF VERTICAL COORDINATES
 LV      2D FLAG ARRAY
         =2   INTERPOLATED VALUE
         =1   IN RANGE VALUE
         =0   OUT OF RANGE VALUE
         =-1  CONVERSION FAILURE
         <-1  HIDDEN POINT
 NR      ROW DIMENSION
 IN      .
 IX      .DO LOOP RANGE FOR X-VARIABLE
 IC      .
 JN      -
 JX      -DO LOOP RANGE FOR Y-VARIABLE
 JC      -
```

The ruling is controlled by SETGRD.

4.10.3 SUBJHV

Purpose: To convert from subject space to object space; to linearly interpolate extra values to the surface

```
CALL SUBJHV ( XR,YR,fr,MX,MY,MR,KPTS,LPTS,H,V,LV)
```

XR	I	SUBJECT SPACE ARRAY OF VALUES
YR	I	SUBJECT SPACE ARRAY OF VALUES
FR	I	2D SUBJECT SPACE ARRAY OF VALUES
MX	I	DIMENSION OF XRAY
MY	I	DIMENSION OF YRAY
MR	I	ROW DIMENSION OF FRAY
KPTS	I	NUMBER OF INTERPOLATION POINTS IN X
LPTS	I	NUMBER OF INTERPOLATION POINTS IN Y
H	O	2D ARRAY OF HORIZONTAL COORDINATES
V	O	2D ARRAY OF VERTICAL COORDINATES
LV	O	2D FLAG ARRAY
		=2 INTERPOLATED VALUE
		=1 IN RANGE VALUE
		=0 OUT OF RANGE VALUE
		=-1 CONVERSION FAILURE

An external function may be used if SETEXT is called.

4.11 SETTING OF GRAPH PARAMETERS 2

Of these routines only SETAFF is intended for regular use. All of these routines might logically be considered internal subprograms; their use should be avoided.

4.11.1 SETAFF

Purpose: To provide a general affine transformation

```
CALL SETAFF [( THAFF [,XOFAFF, YOFAFF, ax, ay])] or
CALL SETAFF ( A1, A2, A3, A4, XOFAFF, YOFAFF)
```

THAFF	ANGLE FOR TRANSFORMATION (DEFAULT 0.0)
XOFAFF	OFFSET VECTOR (DEFAULT [0.0,0.0])
YOFAFF	
AX	ABSCISSA SCALE FACTOR (DEFAULT 1.0)
AY	ORDINATE SCALE FACTOR (DEFAULT = AX)

A1,A2,A3,A4 ELEMENTS OF A 2x2 TRANSFORMATION MATRIX

This routine is used by some of the frame drawing subprograms. GRFRAM, and MOSAIC.

4.11.2 SETAXL

Purpose: To define tic mark characteristics in AXLINE

```
CALL SETAXL ( TL, DTIC)
```

TL	TIC LENGTH
DTIC	FRACTION OF TIC ON POSITIVE SIDE OF AXIS

4.11.3 SETCTL

Purpose: To set cutline parameters

```
CALL SETCTL ( lcutln, lpause, ladvan)
```

LCUTLN	CUT LINE FLAG (DEFAULT .FALSE.)
LPAUSE	PAUSE FLAG (DEFAULT .NOT.LCUTLN)

LADVAN PAGE ADVANCE FLAG (DEFAULT .FALSE.)

This subprogram sets parameters for subsequent NUPAGE calls and is implicitly called by BEGPLT.

4.11.4 SETDVL

Purpose: To set device limits

CALL SETDVL (DVXL, DVYL[, DVXLM, DVYLM])

DVXL	DEVICE LIMIT FOR "X" AXIS
DVYL	DEVICE LIMIT FOR "Y" AXIS
DVXLM	DEVICE MAXIMUM LIMIT FOR "X" AXIS (DEFAULT DVXL)
DVYLM	DEVICE MAXIMUM LIMIT FOR "Y" AXIS (DEFAULT DVYL)

The optional maximum device settings are used for example with electromechanical plotters with a roll of paper.

4.11.5 SETPAN

Purpose: To define a panel

CALL SETPAN [(XOP,YOP,GPL,GPH)]

XOP	ABSCISSA OF PANEL ORIGIN
YOP	ORDINATE OF PANEL ORIGIN
GPL	GRAPH PANEL LENGTH
GPH	GRAPH PANEL HEIGHT

No arguments resets to page values; this routine is called by MOSAIC.

4.12 INTERNAL SUBPROGRAMS

WARNING: THESE SUBPROGRAMS SHOULD NOT BE CALLED BY ANY USER!

4.12.1 Device Drivers

The different drivers provide a choice of output media. The following drivers are available:

QMS	QMS Lasergrafix 1200 printer
REG	for the VT240 and the VT241 which allow ReGIS graphics
TEK	for RetroGraphic modifications of VT100 like terminals using Tektronix 4010/4014 graphics
ZT8	NICOLET ZETA8 electromechanical plotter

Each driver has several entries:

```
_OPEN ( PORTID)
_LIMITS ( DVXL, DVYL)
_PLOT ( XD, YD, IP)
_LINE ( INP)
_FLUSH
_PAUSE [( INWORD)]
_CLEAR [( XD, YD)]
_RESET
_CLOSE [( XD, YD)]
_MSG ( PROMPT, REPLY)
```

A three character prefix distinguishes the different drivers; the _MSG entry is optional.

These drivers may serve as models for writing other drivers; to actually add a driver only BEGPLT need be modified in order to actually reference the new device type and its driver.

4.12.2 PLTEXT

Purpose: To reference external device drivers from PLOT.

CALL PLTEXT (PLT, NEWP, FLSH, PAUS, CLR, REST, CLOS, msg)

PLT	LINE DRAWING
NEWP	CHANGE PEN
FLSH	BUFFER FLUSHING
PAUS	PAUSE
CLR	CLEAR DISPLAY
REST	RESTORE TERMINALS TO ASCII MODE
CLOS	CLOSE PLOTTING
MSG	INTERACTIVE MESSAGE

External subroutines for a given device driver.

4.12.3 SETPLO

Purpose: To define window on the page

CALL SETPLO (XPMIN,YPMIN,XPMAX,YPMAX)

Minimum and maximum limits of plotting window

4.12.4 SETROT

Purpose: To provide a general transformation on the page

CALL SETROT [(THROT [,XOFROT, YOFROT, rx, ry])] or
CALL SETROT (R1, R2, R3, R4, XOFROT, YOFROT)

THROT	ANGLE FOR TRANSFORMATION (DEFAULT 0.0)
XOFROT	OFFSET VECTOR (DEFAULT [0.0,0.0])
YOFROT	
RX	ABSCISSA SCALE FACTOR (DEFAULT 1.0)
RY	ORDINATE SCALE FACTOR (DEFAULT = RX)

R1,R2,R3,R4 ELEMENTS OF A 2x2 TRANSFORMATION MATRIX

4.13 UNSUPPORTED ROUTINES

These subprograms are provided to give compatibility with CalComp basic software; their use is not recommended. All use page coordinates assumed to be in inches.

4.13.1 AXIS

CALL AXIS (XPAGE, YPAGE, IBCD, NCHAR, AXLEN, ANGLE, FIRSTV, DELTAV)

XPAGE	AXIS LINE'S STARTING POINT
YPAGE	
IBCD	BYTE ARRAY CONTAINING AXIS LABEL
NCHAR	NUMBER OF CHARACTERS IN IBCD (sign determines on which side of the axis tic marks and labeling will be placed
AXLEN	LENGTH OF THE AXIS IN PAGE COORINATES
ANGLE	ANGLE AT WHICH THE AXIS IS TO BE DRAWN (DEGREES)
FIRSTV	STARTING VALUE AT FIRST TIC MARK
DELTAV	INCREMENT OF DECREMENT BETWEEN SUCCESSIVE TIC MARKS

AXLINE is recommended instead of this unsupported CalComp routine.

4.13.2 LINE

CALL LINE (XARRAY, YARRAY, NPTS, INC, LINTYP, INTEQ)

XARRAY	ARRAY OF ABSCISSA VALUES
YARRAY	ARRAY OF ORDINATE VALUES
NPTS	NUMBER OF POINTS
INC	INDEX INCREMENT
LINTYP	INTEGER CONTROL PARAMETER FOR TYPE OF LINE TO BE DRAWN The magnitude of LINTYP determines the frequency of plotted symbols; that is, if LINTYP = 4, a symbol is plotted at every fourth point although every point is plotted. If LINTYP is positive then connecting lines are drawn; if negative then no connecting lines are drawn and only the symbols are plotted.
INTEQ	INTEGER EQUIVALENT FOR DESIRED SYMBOL (THIS IS THE CALCOMP EQUIVALENT)

LINRAY is recommended instead of this unsupported CalComp routine.

4.13.3 PLOTS

A call to PLOTS has no effect; it is a dummy entry. The graphics package must be opened with BEGPLT.

4.13.4 SCALE

CALL SCALE (ARRAY, AXLEN, NPTS, INC)

ARRAY	ARRAY OF DATA POINTS TO BE EXAMINED
AXLEN	LENGTH OF THE AXIS
NPTS	NUMBER OF DATA VALUES TO BE SCANNED IN THE ARRAY. THE ARRAY MUST BE DIMENSIONED AT LEAST TWO ELEMENTS MORE THAN THE NUMBER OF VALUES BEING SCANNED AS THE OUPUT IS STORED IN ARRAY(NPTS*|INC|+1) = FIRSTV ARRAY(NPTS*|INC|+|INC|+1) = DELTAV
INC	INCREMENT TO HANDLE MULTIDIMENSIONED ARRAYS NORMALLY 1

GPSCAL is recommended instead of this unsupported CalComp routine.

CHAPTER 5

MORE SAMPLE PROGRAMS

This chapter contains five more sample FORTRAN programs and the plots which they produce to demonstrate some additional features of the graphics package. All of the plots were made on a QMS Lasergrafix 1200 printer.

Program 7	Interactive Control
Program 8	External Data Function
Program 9	External Coordinate Conversion
Program 10	Surface Fitting of Irregularly Distributed Points
Program 11	A Terminal Movie

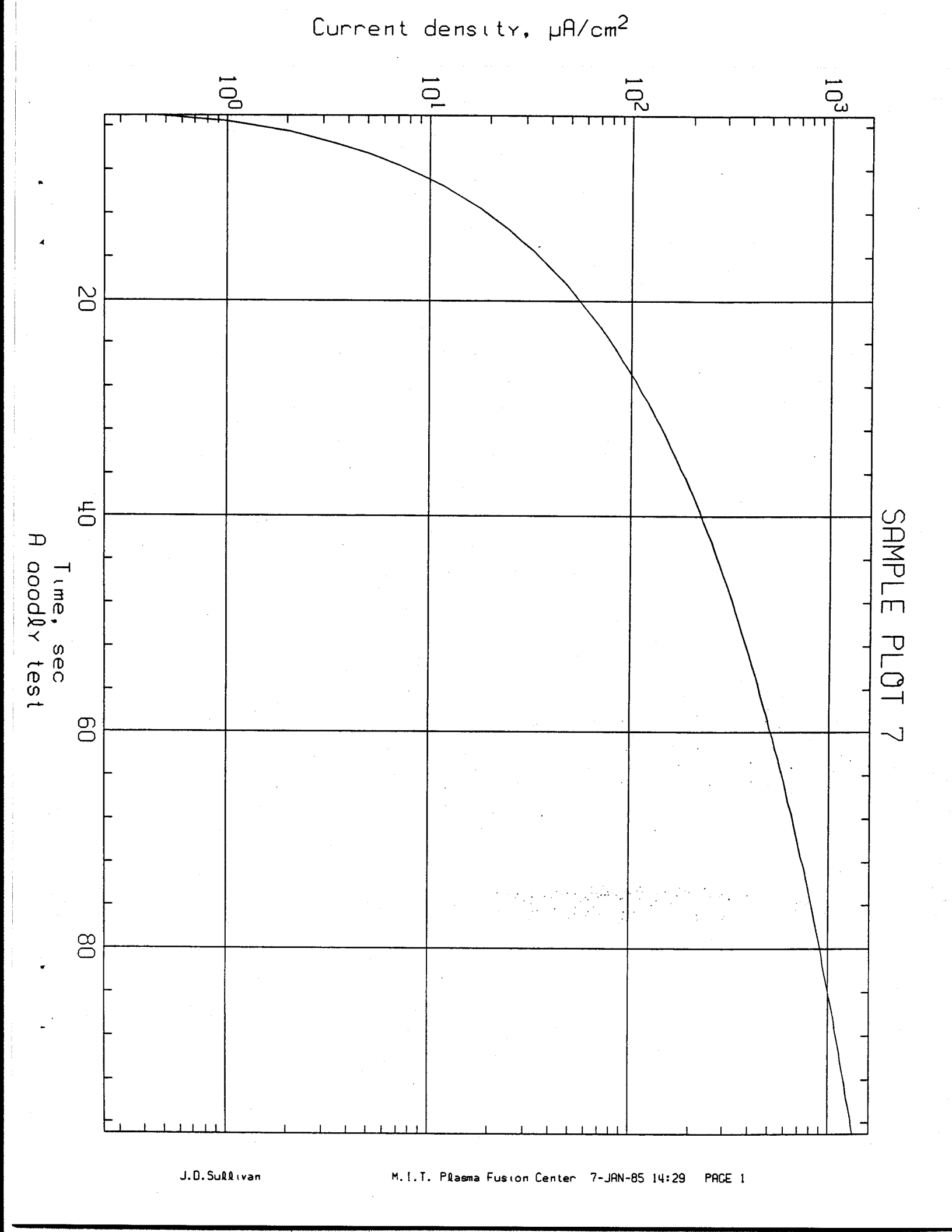
5.1 INTERACTIVE CONTROL

This program illustrates an interactive session with a graphics program. The two figures were produced with the following interactive responses:

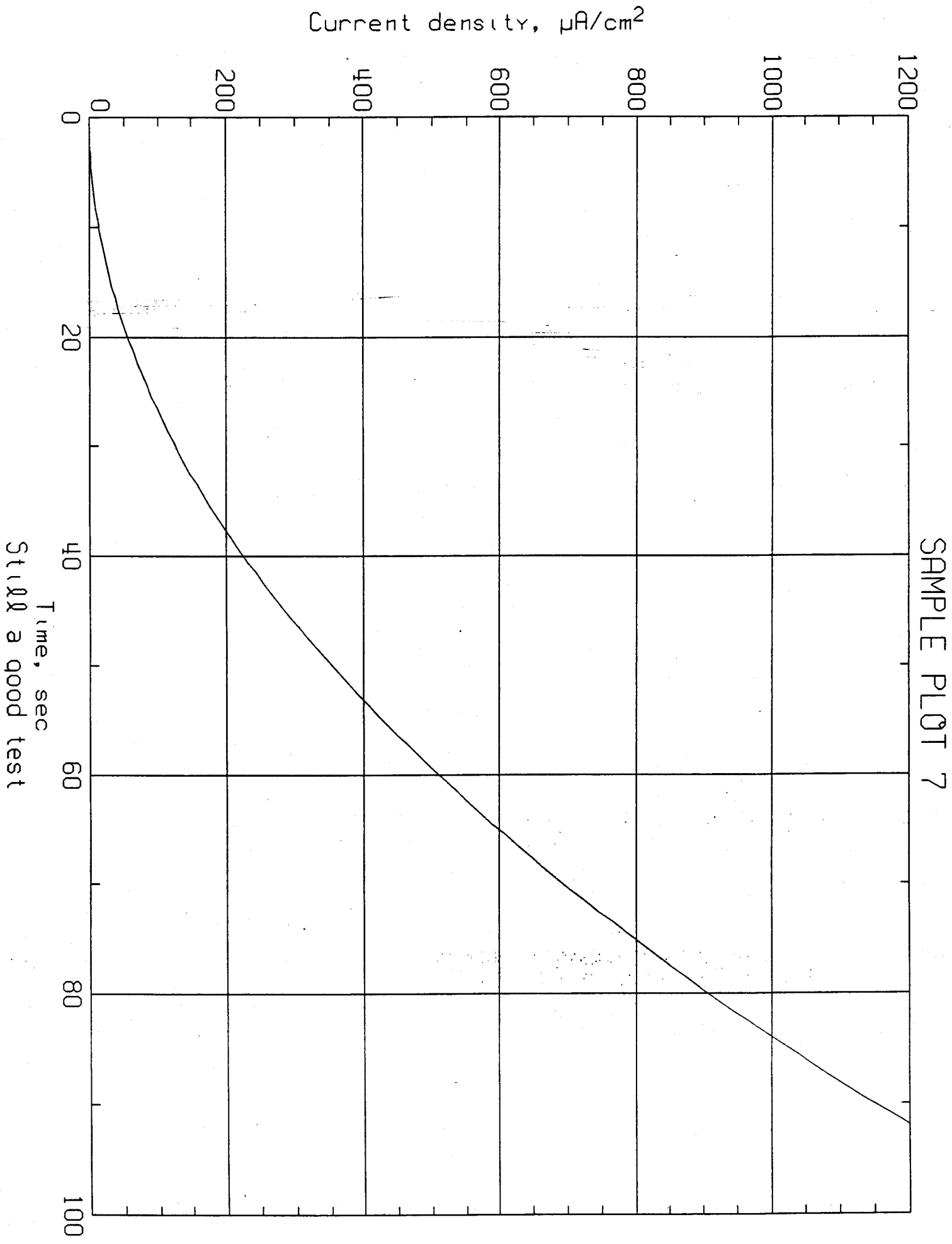
```
Enter PORT ID > QMS
Enter DEVICE TYPE > QMS
Enter your name: J.D.Sullivan
Enter a comment line for the graph: A goodly test
Enter new abscissa parameters: XA,XB,IX,NX,LX = 0.0,100.0,0,10,5
Enter new ordinate parameters: YA,YB,IY,NY,LY = 0.0,1200.0,0,24,6
Type <CR> to continue
Enter a comment line for the graph: Still a good test
Type <CR> to continue
```

After the first end of page prompt "Type <CR> to continue ", a response of either "EXIT" or "QUIT" will terminate this program and produce only one graph. A file QMS.DAT is produced which may be copied to the appropriate device (on the TARA system: $ COPY QMS.DAT LP4:). Were a video terminal selected then the interactive prompts would appear on the screen in the upper left of the screen with the graph still displayed.

```
C ------------------------------------------------------------------
C
C                       INTERACTIVE CONTROL
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP7
C
      CHARACTER*80 REPLY
      REAL X(100),Y(100)
C VAX/VMS RTL convention for STR$TRIM
      INTEGER*2 NLEN
C COMPUTE VALUES FOR THE DATA ARRAYS
      DO 1 I=1,100
      X(I) = I-0.5
    1 Y(I) = ABS(X(I)*(I-1)/7.0-0.5)
C OPEN THE GRAPHICS PACKAGE INTERACTIVELY
      CALL GPINIT ( 'Sample Program 7')
C GET THE USER'S NAME
      CALL PAGPAUS ( 'Enter your name: ', REPLY)
C USE A ROUTINE FROM THE VAX RUN TIME LIBRARY TO FIND LENGTH OF REPLY
      CALL STR$TRIM ( REPLY, REPLY, NLEN)
C SET THE USER IDENTIFIER
      CALL SETUSR ( REPLY(1:NLEN))
C DEFINE A PAGE
      CALL NUPAGE
C DEFINE AXES : SEMILOG BIASED AXES
      CALL SETXAX ( 3.0, 97.0, 0,, 5,,,, 0.0, 20.0)
      CALL SETYAX ( 0.25, 1600.0, 1)
C ADD A FULL GRID MESH
      CALL SETGRD ( .TRUE., .TRUE.)
C DRAW THE GRAPH FRAME
      CALL GRFRAM ( 'Time, sec',
     1   'Current density, '//CHAR(29)//'m'//CHAR(28)//
     2   'A/cm'//CHAR(17)//'2', 'SAMPLE PLOT 7')
C A CURVE
      CALL SYMRAY ( X, Y,,,, 100)
C GET A COMMENT LINE FROM THE USER
      CALL PAGPAUS ( 'Enter a comment line for the graph:', REPLY)
C OUTPUT COMMENT LINE
      CALL TXTLYN ( REPLY)
C GET NEW VALUES FOR THE ABSCISSA PARAMETERS
   10 CALL PAGPAUS ( 'Enter new abscissa parameters: XA,XB,IX,NX,LX = ',
```


Current density, μA/cm²
10⁰
10¹
10²
10³
20
40
60
80
Time, sec
A goodly test
SAMPLE PLOT 7

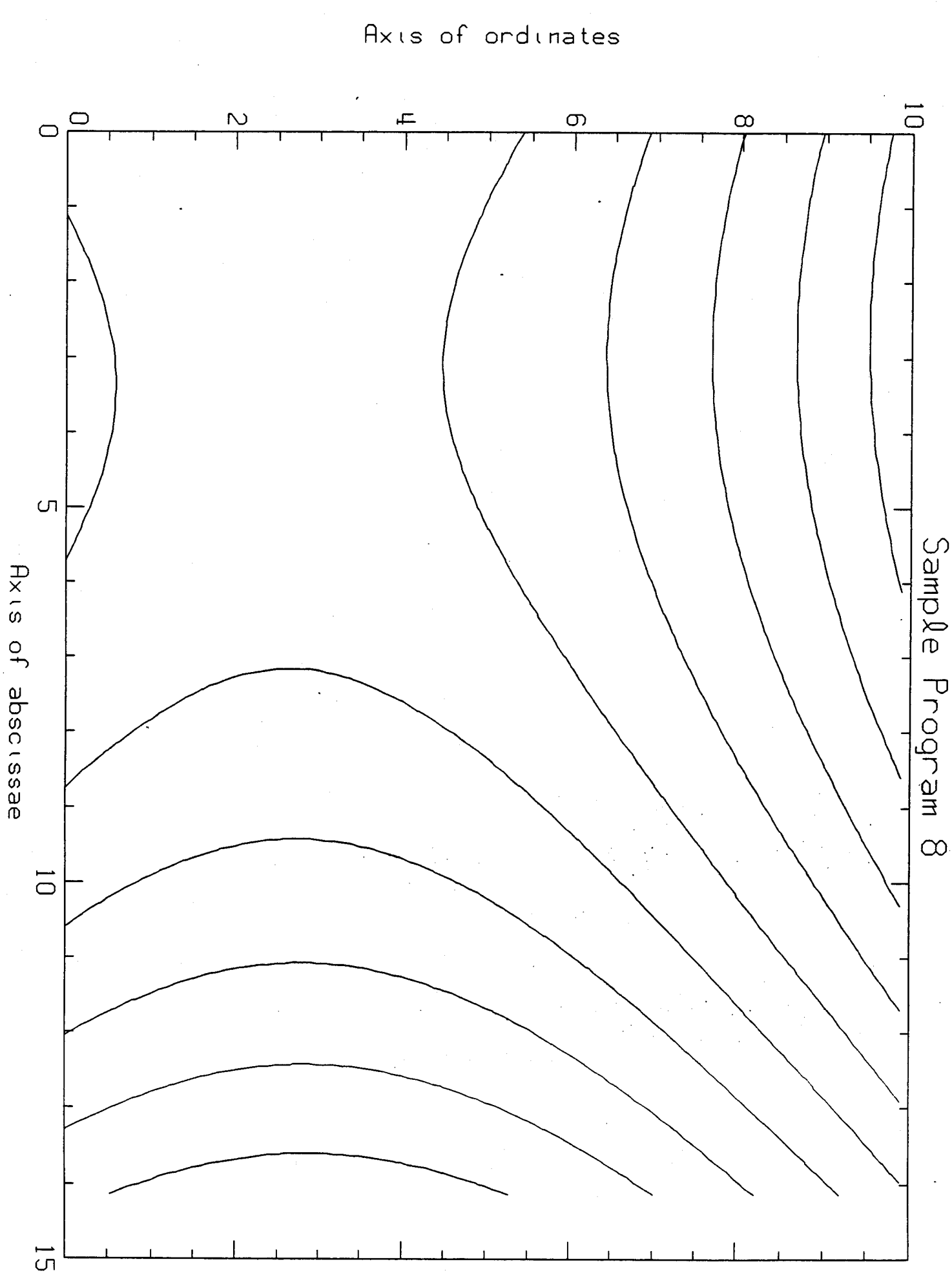
```
     1   REPLY)
C USE A FORMATTED INTERNAL READ TO DECODE THE INPUT
      CALL STR$TRIM ( REPLY, REPLY, NLEN)
      READ (REPLY(1:NLEN),900,ERR=10) XA,XB,IX,NX,LX
C GET NEW VALUES FOR THE ORDINATE PARAMETERS
   11 CALL PAGPAUS ( 'Enter new ordinate parameters: YA,YB,IY,NY,LY = ',
     1   REPLY)
C USE A FORMATTED INTERNAL READ TO DECODE THE INPUT
      CALL STR$TRIM ( REPLY, REPLY, NLEN)
      READ (REPLY(1:NLEN),900,ERR=10) YA,YB,IY,NY,LY
C END THIS PAGE.  IF THE RESPONSE IS EXIT OR QUIT, STOP.
      CALL ENDPAG ( .TRUE., INWORD)
      IF ( INWORD.EQ.'EXIT' .OR. INWORD.EQ.'QUIT') GO TO 100
C REDO THE GRAPH WITH NEW PARAMETERS
      CALL NUPAGE
      CALL SETXAX ( XA,XB,IX,,NX,LX)
      CALL SETYAX ( YA,YB,IY,,NY,LY)
      CALL SETGRD ( .TRUE., .TRUE.)
      CALL GRFRAM ( 'Time, sec',
     1   'Current density, '//CHAR(29)//'m'//CHAR(28)//
     2   'A/cm'//CHAR(17)//'2', 'SAMPLE PLOT 7')
      CALL SYMRAY ( X, Y,,,, 100)
      CALL PAGPAUS ( 'Enter a comment line for the graph:', REPLY)
      CALL TXTLYN ( REPLY)
C CLOSE THE GRAPHICS PACKAGE
  100 CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  900 FORMAT (2F,3I)
      END
```


Current density, μA/cm²
0
200
400
600
800
1000
1200
0
20
40
60
80
100
Time, sec
SAMPLE PLOT 7
Still a good test

5.2 EXTERNAL DATA FUNCTIONS

This program draws an isogram of a surface given by a function subprogram; the same surface is represented in sections 3.5, 3.6, and 5.4.

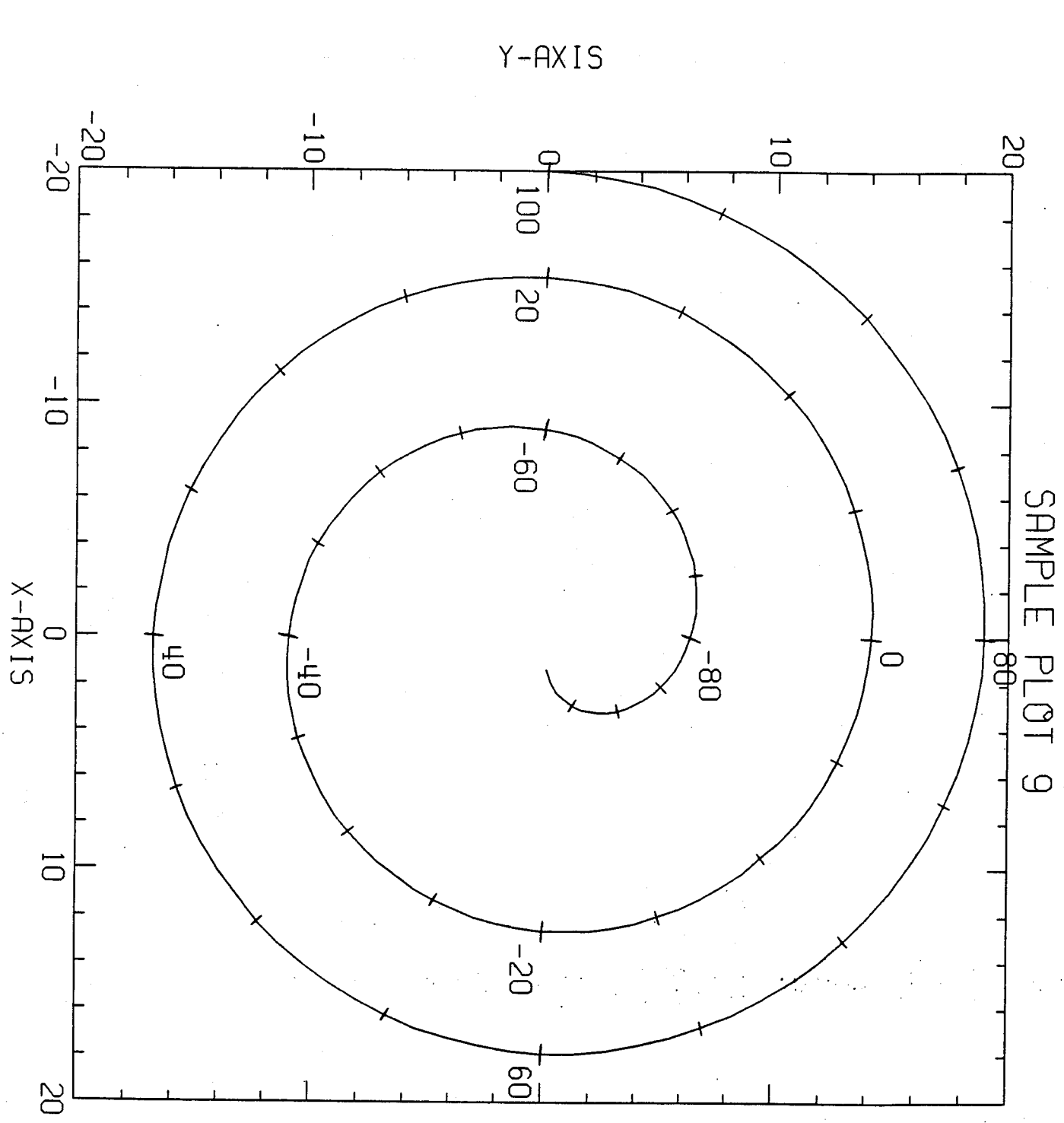
```
C ------------------------------------------------------------------
C
C                     EXTERNAL DATA FUNCTION
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP8
C
      EXTERNAL FNTN
      REAL X(100),Y(100),S(100)
C COMPUTE VALUES FOR THE DATA ARRAYS
      DO 1 I=1,100
      X(I) = (I-1)/7.0
    1 Y(I) = (I-1)/10.0
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      CALL GPINIT ( 'Sample Program 8')
      CALL NUPAGE
C AUTOMATIC AXIS SETTING FOR ARRAYS
      CALL AUTXST ( X, 100)
      CALL AUTYST ( Y, 100)
C MOSAIC MAY BE USED INSTEAD OF GRFRAM
      CALL MOSAIC ( 'Axis of abscissae', 'Axis of ordinates',
     1   'Sample Program 8')
C DRAW DEFAULT ISOPLETHS OF THE FUNCTION FNTN
      CALL SETCON ( .TRUE., FNTN)
      CALL CONMAP ( X, Y, S, 100)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
      END
C
C SAMPLE EXTERNAL FUNCTION
C
      FUNCTION FNTN ( X, Y)
      I = 7*X+1.01
      J = 10*Y+1.01
      FNTN = ((I-25)**2-(J-25)**2+I*J/10)*100
      RETURN
      END
```


Axis of ordinates
0
2
4
6
8
10
0
5
10
15
Axis of abscisse
Sample Program 8

5.3 EXTERNAL COORDINATE CONVERSION

This program uses a user supplied data conversion subprogram to plot a curve in polar coordinates. This method may also be used to make various map projections, e.g. Hammer-Aitoff equal area all-sky maps. Note that ARYPLT is used to draw a parameterized curve.

```
C ------------------------------------------------------------------
C
C                 EXTERNAL COORDINATE CONVERSION
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP9
C
      EXTERNAL POLAR
      REAL TH(200),R(200),T(200)
C COMPUTE VALUES FOR THE DATA ARRAYS
      PI = ACOS(-1.0)
      DO 1 I=1,200
      TH(I) = I*PI/40.0
      R(I) = SQRT(FLOAT(2*I))
    1 T(I) = I-100
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      CALL GPINIT ( 'Sample Program 9')
C DEFINE A PAGE WITH A SQUARE FRAME
      CALL NUPAGE ( 1.0, 1.0, 0.8, 0.8, 0.15, 0.1)
C DEFINE DOMAIN AND RANGE
      CALL SETXAX ( -20.0, 20.0, 0,, 20, 4)
      CALL SETYAX ( -20.0, 20.0, 0,, 20, 4)
C DRAW THE GRAPH FRAME
      CALL GRFRAM ( 'X-AXIS', 'Y-AXIS', 'SAMPLE PLOT 9')
C SET THE EXTERNAL CONVERSION
      CALL GENCON ( .TRUE., POLAR)
C DRAW THE CURVE
      CALL ARYPLT ( R, TH, 200, T, 5.0, -1, 20.0)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
      END
C
C SAMPLE EXTERNAL SUBROUTINE
C
      SUBROUTINE POLAR ( R, THRAD, X, Y)
      X = R*COS(THRAD)
      Y = R*SIN(THRAD)
      RETURN
      END
```



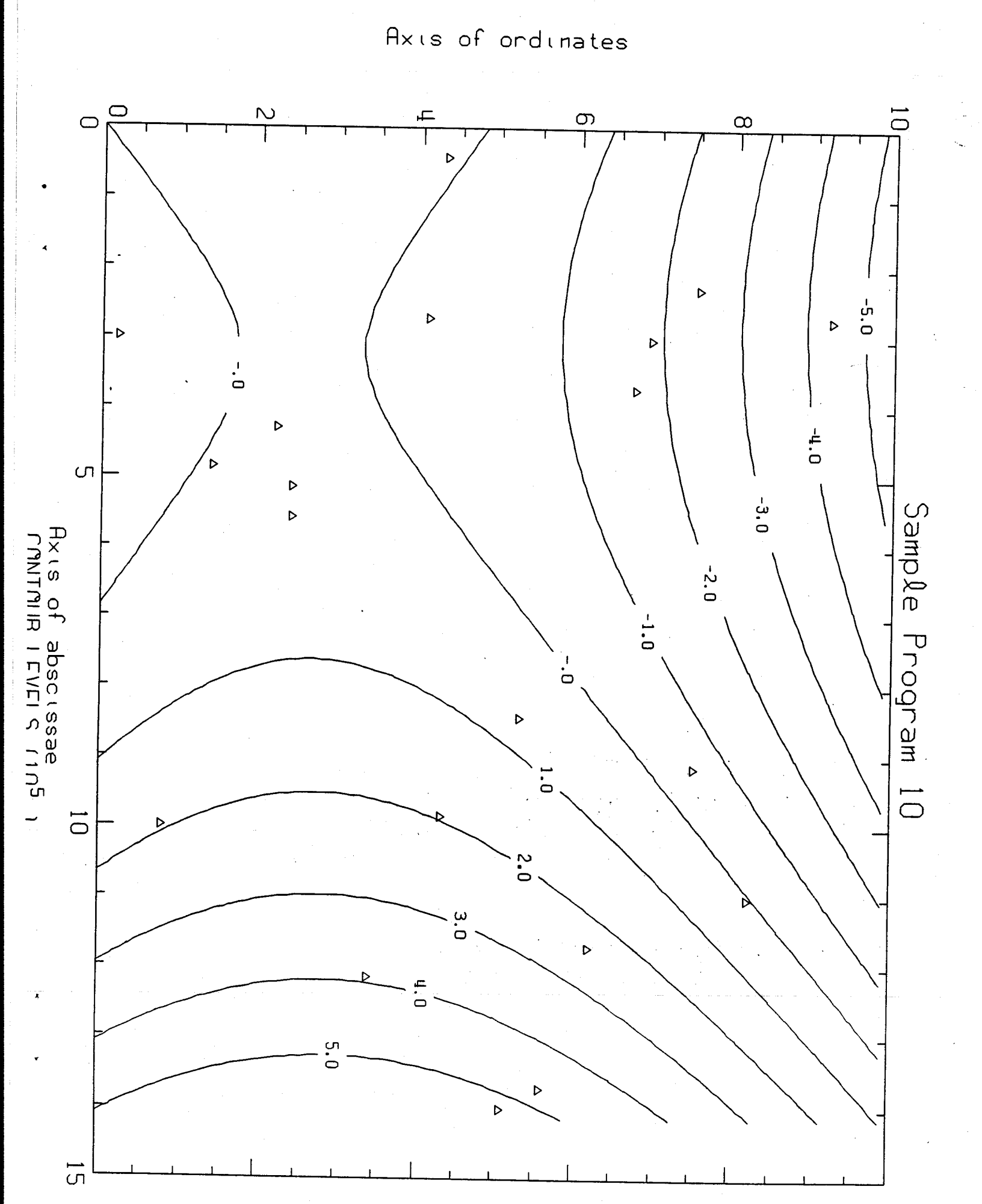
TARA Tandem Mirror Experiment M.I.T. Plasma Fusion Center 7-JAN-85 14:31 PAGE 1

5.4 SURFACE FITTING

This program uses a subprogram from the IMSL library to fit a surface with irregularly distributed data and draws an isogram of the fit to the surface as well as plotting the irregularly distributed data points. Although, this is the same surface as in sections 3.5, 3.6, and 5.2, only 20 points are used to define the surface instead of 10,000 (but 10,000 points are generated before plotting). Note the use of VAX/VMS virtual memory to reduce the fixed size of the program. Finally, remember that the points input to IQHSCV must be distinct.

```
C ------------------------------------------------------------------
C
C       SURFACE FITTING OF IRREGULARLY DISTRTIBUTED POINTS
C               (requires the IMSL library)
C
C
C               GRAPHICS PACKAGE version 4.1
C
C
C                   VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP10
C IMSL VARIABLES
      PARAMETER ( nxi=100)
      PARAMETER ( nyi=100)
      PARAMETER ( izi=100)
      REAL XI(nxi), YI(nyi)
      PARAMETER ( nd=20)
      REAL XD(nd), YD(nd), ZD(nd)
C SEEDS FOR RANDOM NUMBER GENERATOR
      DATA ISEED1/32656/, ISEED2/45217/
C COMPUTE VALUES FOR THE DATA ARRAYS
      DO 1 I=1,nxi
    1 XI(I) = (I-1)/7.0
      DO 2 J=1,nyi
    2 YI(J) = (J-1)/10.0
C SELECT nd NON-IDENTICAL ARBITRARY POINTS FROM THE SURFACE
      DO 3 I=1,nd
      K = nxi*RAN( ISEED1, ISEED2)
      K = MOD(K,nxi)+1
      L = nyi*RAN( ISEED1, ISEED2)
      L = MOD(L,nyi)+1
      XD(I) = XI(K)
      YD(I) = YI(L)
    3 ZD(I) = FNTN( XD(I), YD(I))
C USE VAX/VMS RUN TIME LIBRARY ROUTINES TO GET
C VIRTUAL STORAGE FOR EXTRA ARRAYS:  ZI, IWK, WK
      CALL LIB$GET_VM ( 4*nxi*nyi, IADRZI)
      CALL LIB$GET_VM ( 4*(31*nd+nxi*nyi), IADRIWK)
      CALL LIB$GET_VM ( 4*6*nd, IADRWK)
C USE IMSL ROUTINE IQHSCV TO FIT THE SURFACE IN ARRAY:  ZI
      CALL IQHSCV ( XD,YD,ZD,nd,XI,nxi,YI,nyi,%val(IADRZI),izi,
     1   %val(IADRIWK),%val(IADRWK),IER)
C RELEASE STORAGE FOR ARRAYS:  IWK, WK
      CALL LIB$FREE_VM ( 4*(31*nd+nxi*nyi), IADRIWK)
      CALL LIB$FREE_VM ( 4*6*nd, IADRWK)
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      CALL GPINIT ( 'Sample Program 10')
      CALL NUPAGE
C AUTOMATIC AXIS SETTING FOR ARRAYS
      CALL AUTXST ( XI, nxi)
      CALL AUTYST ( YI, nyi)
C MOSAIC MAY BE USED INSTEAD OF GRFRAM
      CALL MOSAIC ( 'Axis of abscissae', 'Axis of ordinates',
     1   'Sample Program 10')
C DRAW DEFAULT ISOPLETHS
      CALL FNCON ( XI, YI, %val(IADRZI), nxi, nyi)
C INDICATE THE nd POINTS FROM WHICH THE SURFACE WAS GENERATED
```

```
      CALL SYMRAY ( XD, YD,, 2,, nd, -3)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C RELEASE STORAGE FOR ARRAY:  ZI
      CALL LIB$FREE_VM ( 4*nxi*nyi, IADRZI)
C
      STOP
      END
C
C SAMPLE EXTERNAL FUNCTION
C
      FUNCTION FNTN ( X, Y)
      I = 7*X+1.01
      J = 10*Y+1.01
      FNTN = ((I-25)**2-(J-25)**2+I*J/10)*100
      RETURN
      END
```

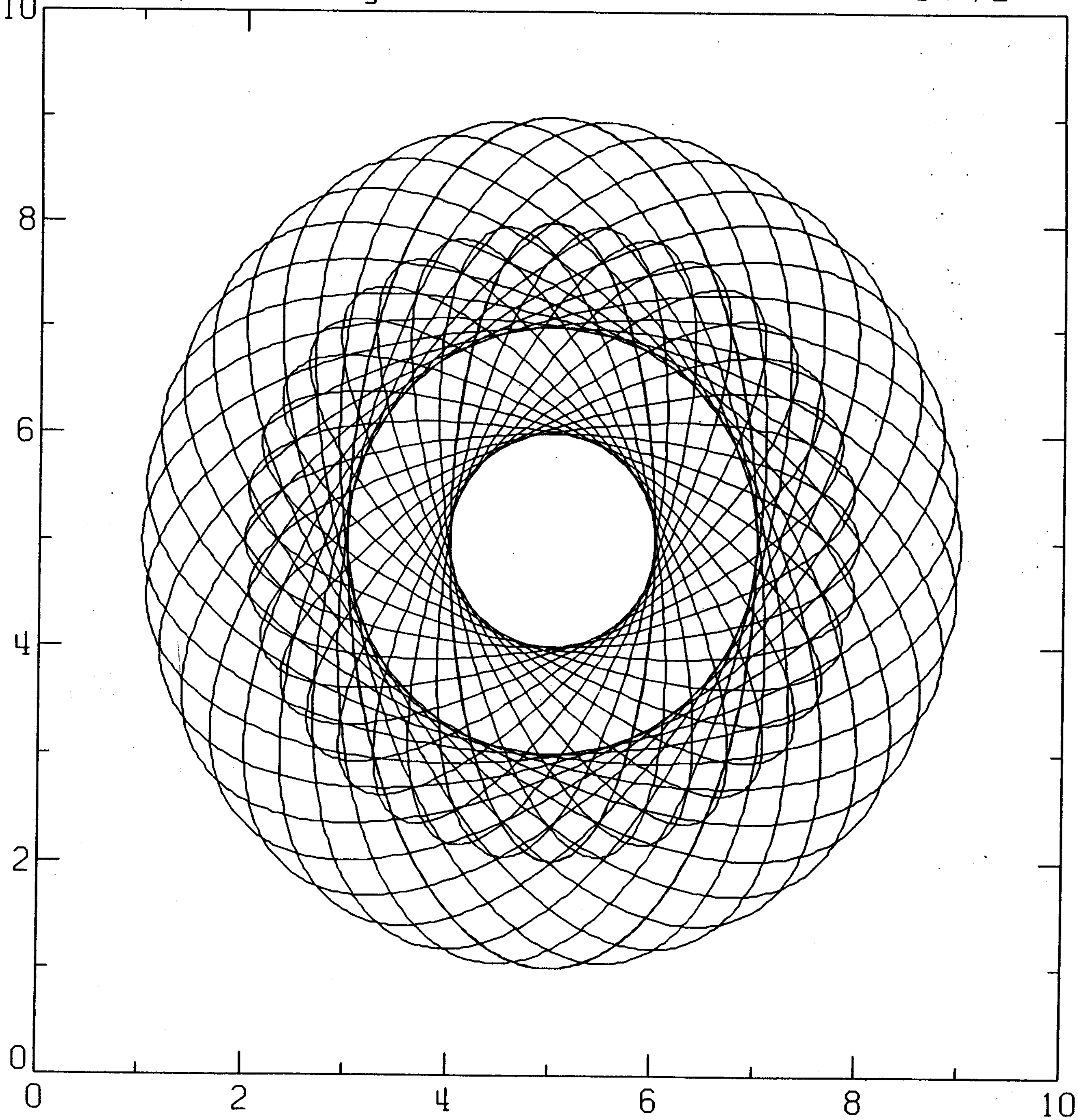

Axis of ordinates
Sample Program 10
Axis of abscissae
CONTOUR LEVELS (105)
0
2
4
6
8
10
0
5
10
15
-5.0
-4.0
-3.0
-2.0
-1.0
-.0
-.0
1.0
2.0
3.0
4.0
5.0

5.5 A TERMINAL MOVIE

This program makes a "movie" on a video screen by drawing and erasing consecutive curves. As the program output is intended to be viewed on a video display terminal the sample illustration is inaccurate and shows all of the curves overlayed. A good way to run this program is to make a file and then copy it to the terminal (see 5.1).

```
C ------------------------------------------------------------------
C
C                        A TERMINAL MOVIE
C
C
C                   GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSP11
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      CALL GPINIT ( 'Sample Program 11')
C DEFINE A SQUARE PAGE
      CALL NUPAGE ( 1.0, 1.0)
C SET THE DOMAIN AND RANGE
      CALL BOUNDS ( 0.0, 10.0, 0, 0.0, 10.0, 0)
C DRAW THE FRAME
      CALL GRFRAM ( ' ', ' ', 'Sample Program 11 : A Terminal Movie')
C A FAN OF ELLIPSES
      DO 10 PHI=0.0,180.0,10.0
   10 CALL ELLIPS ( 5.0,5.0,2.0,4.0,PHI)
C ERASE THE ELLIPSE FAN
      CALL NEWPEN ( -1)
      DO 11 PHI=0.0,180.0,10.0
   11 CALL ELLIPS ( 5.0,5.0,2.0,4.0,PHI)
C AN ELLIPSE MOVIE
      DO 12 PHI=0.0,180.0,10.0
      CALL NEWPEN ( 0)
      CALL ELLIPS ( 5.0,5.0,1.0,3.0,PHI)
      CALL NEWPEN ( -1)
   12 CALL ELLIPS ( 5.0,5.0,1.0,3.0,PHI)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
      END
```

Sample Program 11 : A Terminal Movie



TARA Tandem Mirror Experiment M.I.T. Plasma Fusion Center 7-JAN-85 14:32 PAGE 1

APPENDIX A

PRIMER FOR CALCOMP USERS

A.1 DISCUSSION

Although the package mimics CalComp and generally conforms to CalComp standards, several differences from standard CalComp should be noted. The PLOTS entry is null and all initialization is performed by BEGPLT q.v.; similarly plot termination is via ENDPLT. The routines AXIS, LINE, and SCALE are available but are not supported; OFFSET is unavailable. The SYMBOL routine has an expanded character and command set. The use of 999.0 as a special coordinate value is not supported. To use CalComp calls which are in inches the page must be defined appropriately e.g.

```
CALL NUPAGE ( 8.5, 11.0,,, 0.0, 0.0, .TRUE.)     or

CALL NUPAGE ( 11.0, 8.5,,, 0.0, 0.0)
```

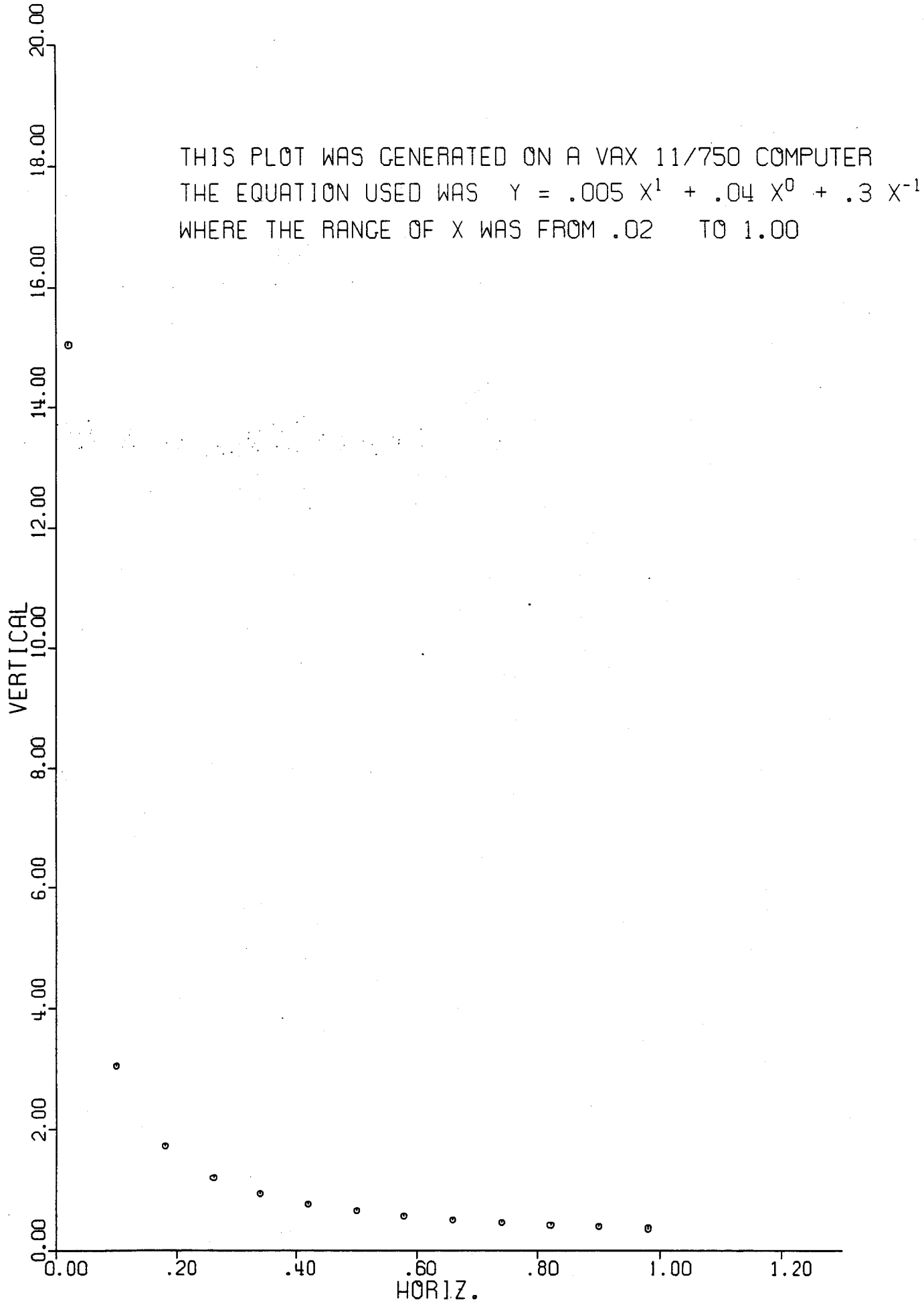
Because most video display devices are rectangular with the long axis horizontal, the graph orientation on hardcopy devices is rotated relative to normal conventions which have the long axis vertical. The drivers automatically take care of this to preserve the proportion as closely as possible. Also note that the page margins are set to zero; this is because PLOT only defines virtual origins on a page. That is PLOT is not actually a device driver as it is in CalComp systems.

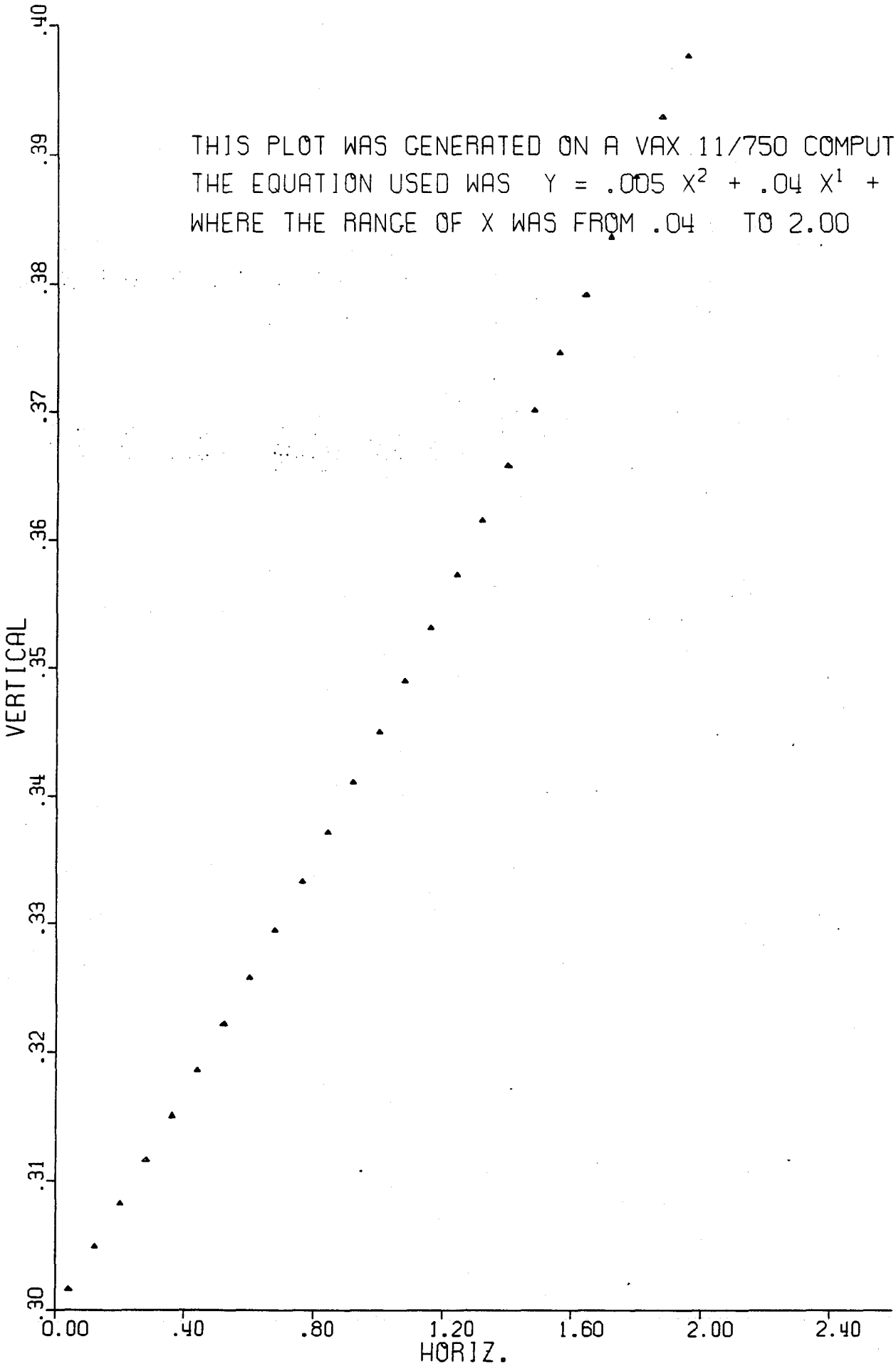
A.2 SAMPLE CALCOMP PROGRAM

This program demonstrates the use of PLOT, SYMBOL, and NUMBER as well as AXIS, SCALE, and LINE in making five graphs. Differences from CalComp praxis must occur in the initiation of each graph ; these differences are delimited between C* + * and C* - * comment statements in the program listing.

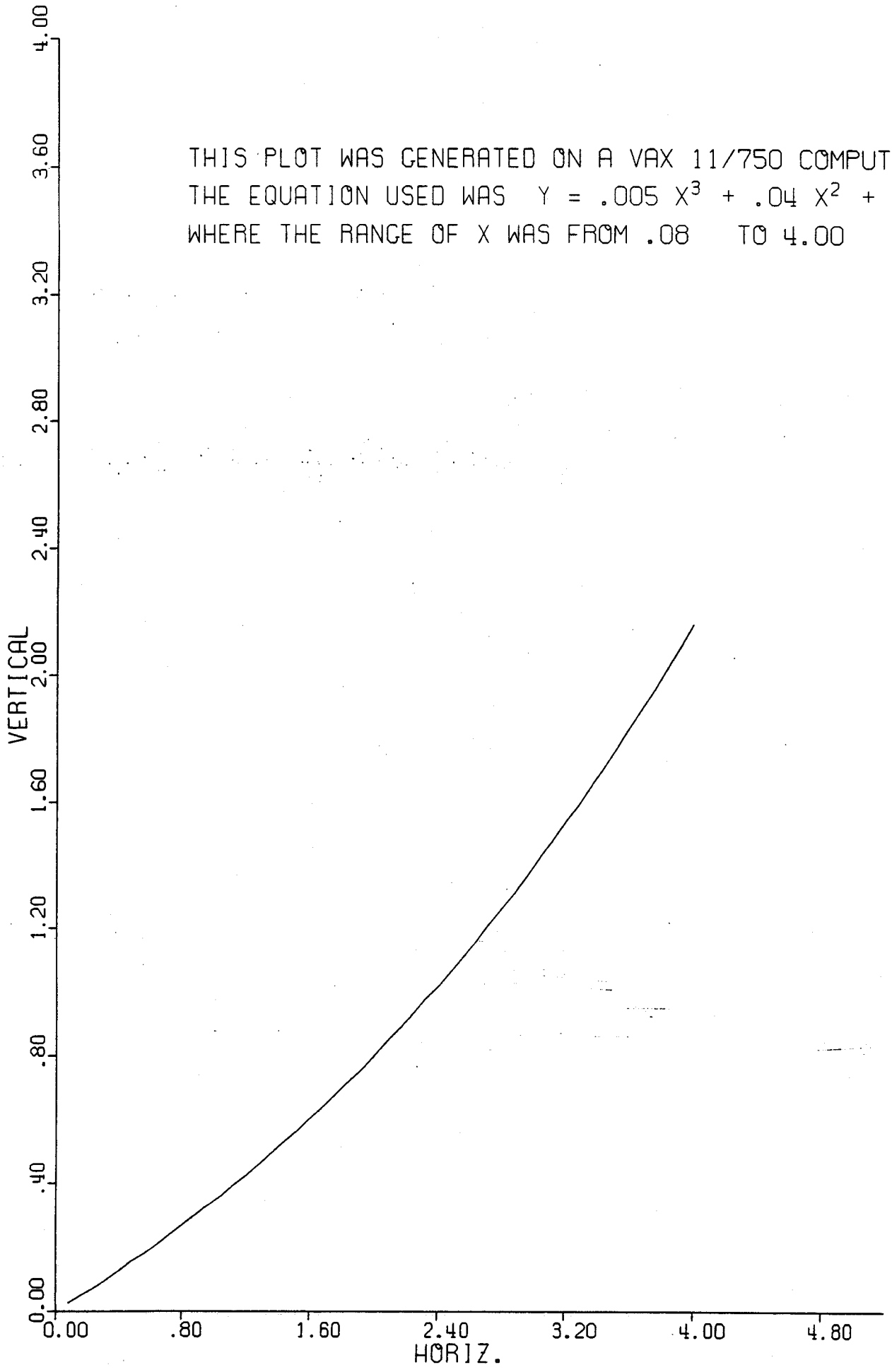
```
C ------------------------------------------------------------------
C
C                     SAMPLE CALCOMP PROGRAM
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPSCCP
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      DIMENSION X(52), Y(52)
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Sample CalComp Program'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
C* + *
```

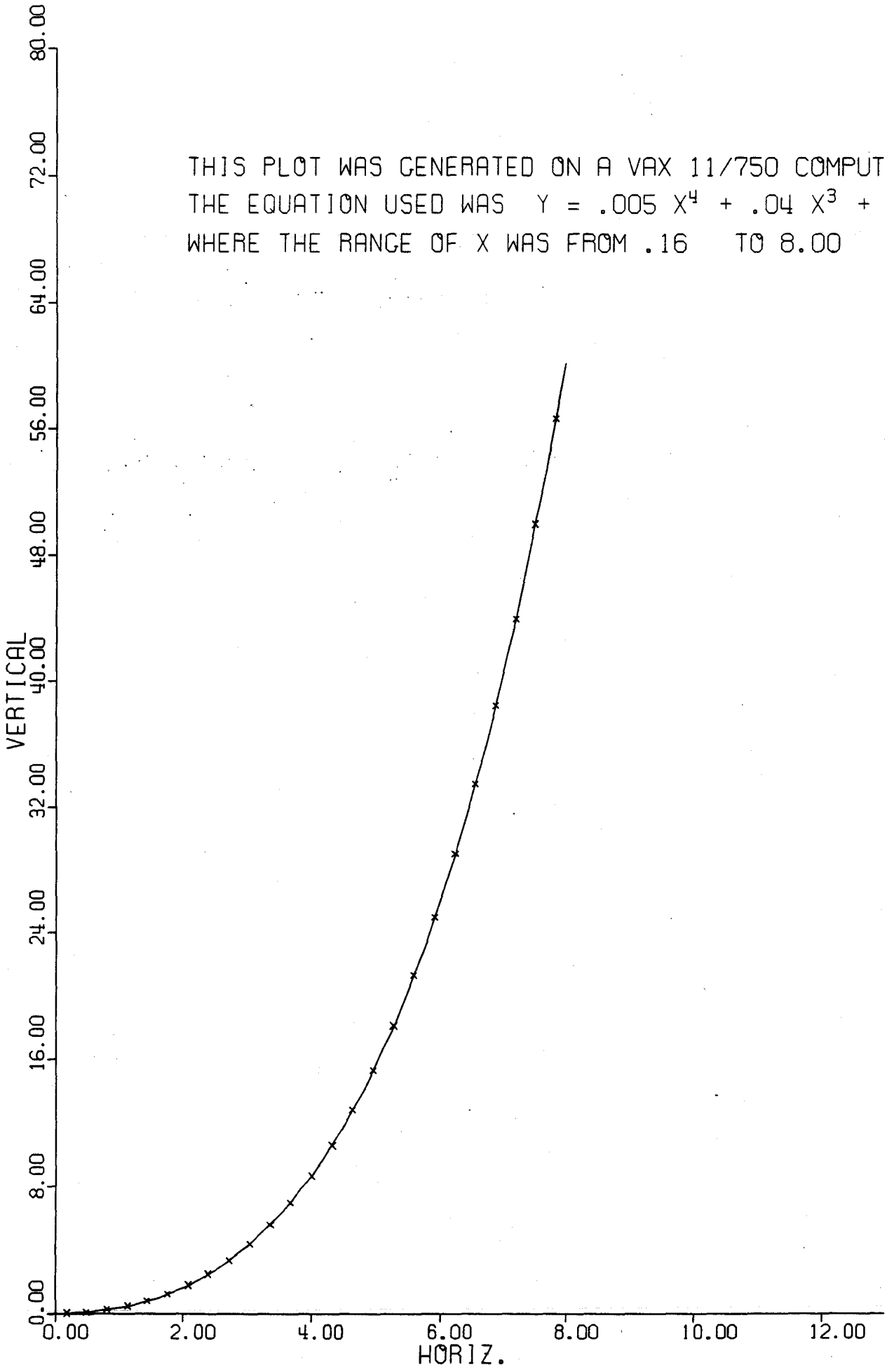
```
      CALL BEGPLT ( PORTID, DEVTYP)
C* - *
C GENERATE SEVERAL GRAPHS
      DLTX = 0.01
      DO 11 I=1,5
      DLTX = DLTX*2.0
C INITIALIZE THE ARRAYS WITH SOME DATA
      X(1) = DLTX
      DO 21 J=1,50
      Y(J) = 0.005*X(J)**I+0.04*X(J)**(I-1)+0.3*X(J)**(I-2)
   21 X(J+1) = X(J)+DLTX
C SPECIFIC CODE TO START A NEW PAGE (GRAPH)
C* + *
      CALL NUPAGE ( 8.5, 11.0,,, 0.0, 0.0, .TRUE.)
C* - *
C CALCOMP STANDARD CODE
      CALL PLOT (1.0,0.5,-3)
      CALL NUMBER (4.6,8.4,0.14,X(1),0.0,2)
      CALL NUMBER (5.68,8.4,.14,X(50),0.0,2)
      CALL SCALE (X,6.5,50,1)
      CALL SCALE (Y,10.0,50,1)
      CALL LINE (X,Y,50,1,(I-3)*2,I)
      CALL AXIS ( 0.0,0.0,'HORIZ.',-6,6.5,0.0,X(51),X(52))
      CALL AXIS ( 0.0,0.0,'VERTICAL',8,10.0,90.0,Y(51),Y(52))
      CALL SYMBOL (1.0,9.0,0.14,
     1'THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER',0.0,48)
      CALL SYMBOL (1.0,8.7,0.14,
     1'THE EQUATION USED WAS  Y = .005 X  + .04 X  + .3 X',0.0,50)
      CALL SYMBOL (1.0,8.4,0.14,
     1'WHERE THE RANGE OF X WAS FROM       TO            ',0.0,50)
      CALL NUMBER (4.96,8.79,0.10,FLOAT(I),0.0,-1)
      CALL NUMBER (6.04,8.79,0.10,FLOAT(I-1),0.0,-1)
   11 CALL NUMBER (7.00,8.79,0.10,FLOAT(I-2),0.0,-1)
C CLOSE THE GRAPHICS PACKAGE
C* + *
      CALL ENDPLT
C* - *
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

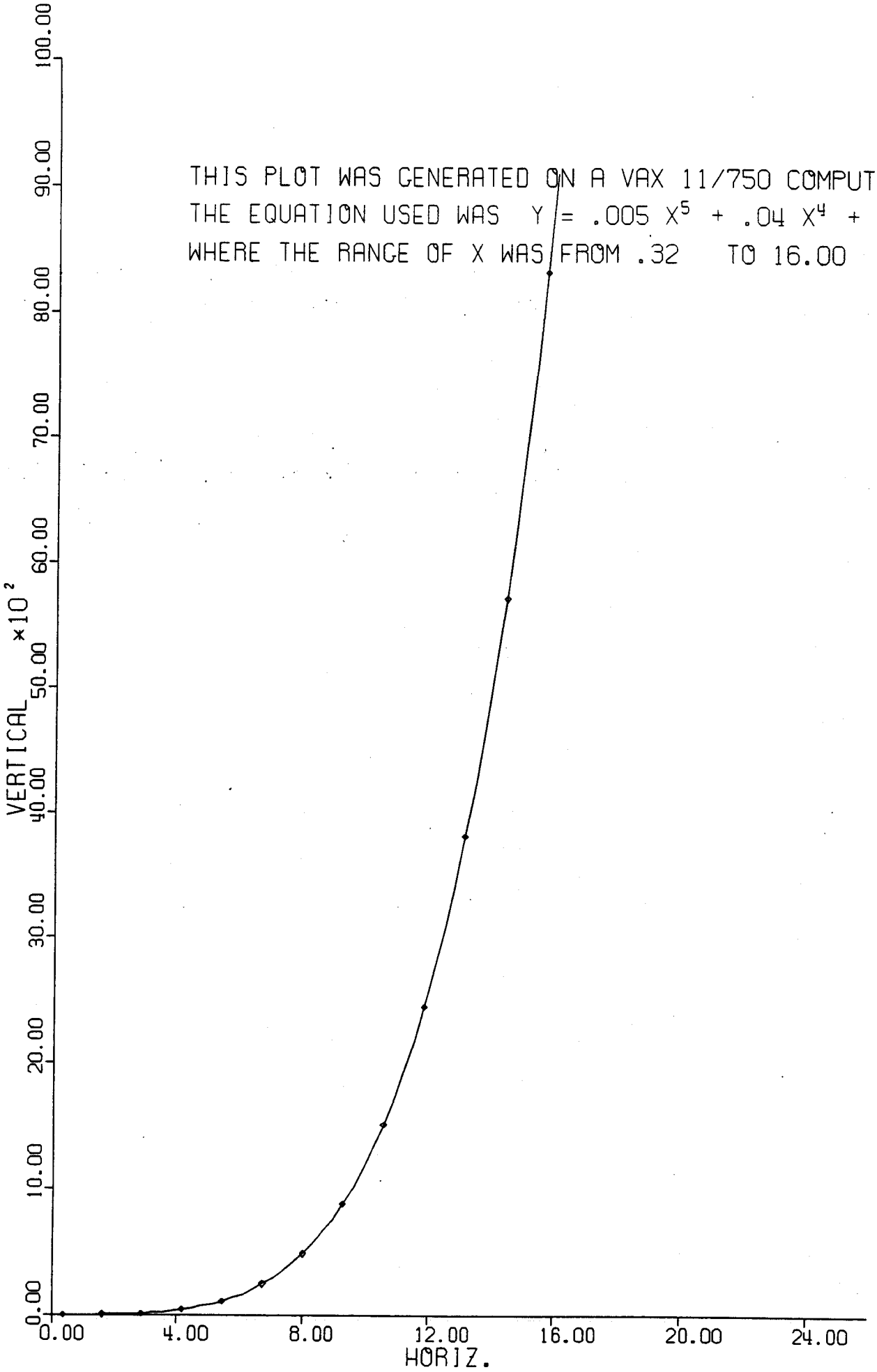

THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER
THE EQUATION USED WAS Y = .005 X^1 + .04 X^0 + .3 X^-1
WHERE THE RANGE OF X WAS FROM .02 TO 1.00
VERTICAL
20.00
18.00
16.00
14.00
12.00
10.00
8.00
6.00
4.00
2.00
0.00
0.00
.20
.40
.60
.80
1.00
1.20
HORIZ.


THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER
THE EQUATION USED WAS Y = .005 X^2 + .04 X^1 + .3 X^0
WHERE THE RANGE OF X WAS FROM .04 TO 2.00
VERTICAL
.40
.39
.38
.37
.36
.35
.34
.33
.32
.31
.30
0.00
.40
.80
1.20
1.60
2.00
2.40
HORIZ.

THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER
THE EQUATION USED WAS $Y = .005\ X^3 + .04\ X^2 + .3\ X^1$
WHERE THE RANGE OF X WAS FROM .08 TO 4.00




THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER
THE EQUATION USED WAS Y = .005 X^4 + .04 X^3 + .3 X^2
WHERE THE RANGE OF X WAS FROM .16 TO 8.00
VERTICAL
0.00
8.00
16.00
24.00
32.00
40.00
48.00
56.00
64.00
72.00
80.00
0.00
2.00
4.00
6.00
8.00
10.00
12.00
HORIZ.


THIS PLOT WAS GENERATED ON A VAX 11/750 COMPUTER
THE EQUATION USED WAS Y = .005 X^5 + .04 X^4 + .3 X^3
WHERE THE RANGE OF X WAS FROM .32 TO 16.00
VERTICAL ×10^2
HORIZ.
0.00
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
90.00
100.00
0.00
4.00
8.00
12.00
16.00
20.00
24.00

APPENDIX B

MISCELLANEOUS PROGRAMS

Amoung others, the programs used to produce different figures in the text are listed here if they are not listed elsewhere. The following programs are included:

GPGTP	Generate a Test Pattern
GPGNP	Graph Nomenclature
GPASCP	ASCII Symbol Character Table
GPPSN	Plot, Symbol, and Number
GP3CRV	A Space Curve and its Projections

B.1 GENERATE A TEST PATTERN

This program was used to produce the lead illustration; it uses PLOT to test the linearity of any display device.

```
C ------------------------------------------------------------------
C
C                    GENERATE A TEST PATTERN
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPGTP
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Generate a Test Pattern'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C DEFINE A PAGE
      CALL NUPAGE ( 8.5, 11.0,,, 0.0, 0.0, .TRUE.)
C DRAW A BORDER ABOUT THE PAGE
      CALL BNDARY
C A PAGE TITLE
      CALL PAGTITL ( 'Test Pattern', 2.0)
      CALL PAGLYN ( 'Graphics Package version 4.1')
C SHIFT THE ORIGIN TO THE CENTER OF THE PAGE
      CALL PLOT ( 4.25, 5.5, -3)
      X = 2.9
      Y = .1
      I = 0
 10   I = I+1
      GO TO (20,30,40,50,80),I
 20   DX = .1
      DY = -.1
      GO TO 60
 30   Y = 2.9
      DY = .1
      GO TO 60
 40   X = -2.9
      DX = -.1
      GO TO 60
 50   Y = -2.9
      DY = -.1
 60   DO 70 J = 1,15
      CALL PLOT ( X, Y, 2)
      CALL PLOT ( X+DX, Y+DY, 3)
      CALL PLOT ( X-DX, Y-DY, 2)
      CALL PLOT ( 0.0, 0.0,02)
      X = X-2.*DX
      Y = Y-2.*DY
 70   CONTINUE
      GO TO 10
C CLOSE THE GRAPHICS PACKAGE
 80   CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
```

```
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

B.2 GRAPH NOMENCLATURE

This program was used to produce figures 2 and 3 in chapter 1.

```
C ------------------------------------------------------------------
C
C                        GRAPH NOMENCLATURE
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                        VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPGNP
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
     -CHARACTER*40 STR(20)
      REAL CHFR(20)/20*1.0/
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'Graph Nomenclature'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C CHANGE CUTLINE TEXT
      CALL SETUSR ( 'Page stamp on all hardcopy devices')
C DEFINE A PAGE
      CALL NUPAGE
C PAGE ANNOTATION
      CALL PAGTITL ( 'PAGE TITLE')
      CALL PAGLYN ( 'COMMENT LINE FOR THE PAGE')
C REGION ANNOTATION
      CALL ORDLYN ( 'Comment Line For The Ordinate')
      CALL REGTITL ( 'Region Title')
      CALL REGLYN ( 'Comment Line For A Region')
C SET THE DOMAIN AND RANGE (large enough to force remote exponents)
      CALL SETXAX ( 0.0, 7.0E6, 0,, 14, 7)
      CALL SETYAX ( 0.0, 1.0E7, 0,, 20, 10)
C GRID IN PEN 3
      CALL SETGRD ( .TRUE., .TRUE., 3)
C DRAW THE FRAME
      CALL GRFRAM ( 'Label for the axis of abscissae',
     1   'Label for the axis of ordinates',
     2   'Graph title')
C SPECIAL ANNOTATION
      CALL TXTLYN ( 'Comment line for the graph')
      CALL PANTXT ( 'A multi-line legend')
      CALL PANTXT ( 'within each panel')
C NON STANDARD ANNOTATION
      CALL NEWPEN ( 2)
      CALL PANSET ( GPL, GPH,,, CH)
      CALL PLOT ( -CH/2.0, GPH+CH, 3)
      CALL PLOT ( 0.1*GPL, GPH+2.0*CH, 2)
      CALL NEWPEN ( 1)
      CALL SYMBOL ( 0.1*GPL, GPH+2.0*CH, CH, ' Tic label',
     1   0.0, 10)
      CALL SYMBOL ( 5.55*GPL/7.0, -40.0*CH/7.0, CH,
     1   'Remote exponent', 0.0, 15)
      CALL SYMRIT ( 0.14*GPL, -33.0*CH/7.0, CH, 'Grid mesh',
     1   0.0, 9)
      CALL NEWPEN ( 2)
      CALL NXTSYM ( XP, YP)
      CALL PLOT ( XP+CH/3.0, YP+CH, 3)
```

```
      CALL PLOT ( 0.25*GPL, 0.1*GPH, 2)
      CALL PLOT ( 2.0*GPL/7.0, 0.05*GPH, 3)
      CALL PLOT ( XP+CH/3.0, YP+CH, 2)
      CALL PLOT ( 5.0*GPL/7.0, -24.0*CH/7.0, 3)
      CALL PLOT ( 5.5*GPL/7.0, -36.0*CH/7.0, 2)
C
C DEFINE A PAGE
C
      CALL NUPAGE
C PAGE ANNOTATION
      CALL PAGTITL ( 'PAGE TITLE ON A MULTI-REGION PAGE')
      CALL PAGLYN ( 'COMMENT LINE FOR THE PAGE')
C DEFINE A REGION (LEFT HALF OF PAGE)
      CALL SETREG ( 1, 1, 2, 1)
C REGION ANNOTATION
      CALL ORDLYN ( 'Comment Line For The Ordinate')
      CALL REGTITL ( 'Region Title')
      CALL REGLYN ( 'Comment Line For A Region')
C SET THE DOMAIN AND RANGE
      CALL SETXAX ( 0.0, 4.0, 0,, 4, 2)
      CALL SETYAX ( 0.0, 1.0, 0,, 20, 10)
C DRAW THE FRAME (for several panels)
      CALL MOSAIC ( 'Label for axis of abscissae',
     1   'Ordinate 1', 'NOT WRITTEN 1', 0.4)
      CALL PANTXT ( 'Panel 1')
      CALL TXTLYN ( 'Comment line for the graph')
      CALL SETYAX ( 0.0, 10.0, 0, -0.3, 5, 2)
      CALL MOSAIC ( 'NOT WRITTEN 2', 'Ordinate 2', 'NOT WRITTEN 3')
      CALL PANTXT ( 'Panel 2')
      CALL SETYAX ( 0.0, 100.0, 0,, 10, 2)
      CALL MOSAIC ( 'NOT WRITTEN 4', 'Ordinate 3', 'Graph title', 0.3)
      CALL PANTXT ( 'Panel 3')
C DEFINE A REGION (UPPER RIGHT QUADRANT)
      CALL SETREG ( 2, 2, 2, 2)
C DEFINE A VIRTUAL PAGE WITHOUT MARGINS
      CALL REGION (,,,, 0.0, 0.0)
C REGION ANNOTATION
      CALL ORDLYN ( 'Comment Line For The Ordinate')
      CALL REGTITL ( 'Region Title')
      CALL REGLYN ( 'Comment Line For A Region')
C WRITE TEXT INTO REGION
C PREPARE TEXT ARRAY AND SIZE/CENTERING ARRAY
      STR(1) = 'MESSAGE TITLE'
      STR(3) = 'SKIP A LINE TO FIRST LINE OF TEXT'
      STR(4) = 'SECOND LINE OF TEXT'
      STR(5) = 'THIRD LINE OF TEXT'
      STR(7) = 'SKIP AND CENTER FOURTH LINE OF TEXT'
      STR(9) = 'SKIP A LINE TO FIFTH LINE OF TEXT'
      CHFR(1) = -2.0
      CHFR(7) = -1.0
      CALL REGTEXT ( STR, 20, CHFR)
C DEFINE A REGION (LOWER RIGHT QUADRANT)
      CALL SETREG ( 2, 1, 2, 2)
C REGION ANNOTATION
      CALL ORDLYN ( 'Comment Line For The Ordinate')
      CALL REGTITL ( 'Region Title')
      CALL REGLYN ( 'Comment Line For A Region')
C SET THE DOMAIN AND RANGE
      CALL BOUNDS ( 0.0, 7.0E6, 0, 0.0, 10.0, 0)
C DRAW THE FRAME
      CALL GRFRAM ( 'Label for axis of abscissae',
     1   'Label for axis of ordinates',
     2   'Graph title')
      CALL TXTLYN ( 'Comment line for the graph')
      CALL PANTXT ( 'A legend within')
      CALL PANTXT ( 'the graph frame')
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
```

```
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

B.3 ASCII SYMBOL CHARACTER TABLE

This program was used to produce the figure in chapter 2.

```
C ------------------------------------------------------------------
C
C                       ASCII CHARACTER SET
C
C
C                   GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPASCP
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'ASCII Character Set'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C DEFINE A PAGE
      CALL NUPAGE ( 11.0, 8.5,,, 0.0, 0.0)
C CALCOMP STANDARD CODE
      CALL SYMBOL ( 1.0, 8.10, .21,
     1   'Characters Available in SYMBOL routine [VAX/VMS]', 0.0, 48)
      K = 0
      XS = -0.4
      DO 1 I=1,16
      XS = XS + .65
      XN = XS + .25
      Y = 7.5
      DO 1 J=1,16
      RK = K
      CALL SYMBOL ( XS, Y, .2, K, 0., -1)
      CALL NUMBER ( XN, Y, .1, RK, 0., -1)
      Y = Y - .4
    1 K = K + 1
      CALL SYMBOL ( 1.0, .45, .14,
     1   'Note:  Integer Equivalence > 127 are in font 2', 0.0, 46)
      CALL SYMBOL ( 1.0, .2, .14,
     1   '       INTEQ 0 thru 31 are special centered symbols',
     2   0.0, 51)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
C
      STOP
C
C FORMAT STATEMENTS
C
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

B.4 PLOT, SYMBOL, AND NUMBER

This program demonstrates the use of the basic subprograms: PLOT, SYMBOL, and NUMBER. When running it note that the page orientation flag, a logical variable, is also input in addition to the customary computer port identification and device type; for the figure reproduced here this flag was .FALSE.

```
C ------------------------------------------------------------------
C
C                    PLOT, SYMBOL, and NUMBER
C
C
C                  GRAPHICS PACKAGE version 4.1
C
C
C                       VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GPPSN
C
      CHARACTER*63 PORTID
      CHARACTER*8 DEVTYP
      LOGICAL LROT
C ASSUME THIS PROGRAM IS BEING RUN FROM A GRAPHICS TERMINAL
C AND GET THE COMPUTER PORT AND DEVICE, TERMINAL, TYPE
      TYPE *, 'PLOT, SYMBOL, and NUMBER'
      TYPE 998
      ACCEPT 999, PORTID
      TYPE 997
      ACCEPT 999, DEVTYP
      TYPE 996
      ACCEPT 995, LROT
C
C OPEN THE GRAPHICS PACKAGE
      CALL BEGPLT ( PORTID, DEVTYP)
C PAGE COORDINATES DEFINED IN NUPAGE
C NOTE MARGINS ARE SET TO ZERO
      CALL NUPAGE (,,,, 0.0, 0.0, LROT)
C PLOT, SYMBOL, and NUMBER all use page coordinates
C SO GET THE PAGE SIZE
      CALL PAGSET ( PW, PH,,,,, CH)
C DRAW PAGE OUTLINE
      CALL PLOT ( 0.0, 0.0, 3)
      CALL PLOT ( PW, 0.0, 2)
      CALL PLOT ( PW, PH, 2)
      CALL PLOT ( 0.0, PH, 2)
      CALL PLOT ( 0.0, 0.0, 2)
C ANGULAR LETTER TEST
      THETA = 0.0
      H = 3.0*CH/4.0
      DO 140 I=1,8
      TH = THETA*0.01745329
      CALL SYMBOL ( PW/2.0+3*CH*COS(TH), PH/2.0+3*CH*SIN(TH), H,
     1    'A=', THETA, 2)
      CALL NUMBER (,,, THETA,, -1)
      CALL SYMBOL (,,, ',H=',, 3)
      FPN = H/CH
      CALL NUMBER (,,, FPN,, 2)
      H = H + CH/4.0
  140 THETA = THETA + 45.0
C CENTERED TITLING
      CALL LABSYM ( 0.0,PH-3*CH,PW,'PLOT, SYMBOL, and NUMBER',
     1    0.0,24,-2.0)
      CALL LABSYM ( 0.0,PH-10*CH,PW,'Angular Letter Test',0.0,19)
C LEGEND
      CALL SYMBOL ( PW/8.5, 4*CH, CH, 'A = ANGLE IN DEGREES', 0.0, 20)
      CALL SYMBOL ( PW/8.5, 2*CH, CH, 'H = RELATIVE CHARACTER HEIGHT',
     1    0.0, 29)
C CLOSE THE GRAPHICS PACKAGE
      CALL ENDPLT
```

```
C
      STOP
C
C FORMAT STATEMENTS
C
  995 FORMAT (L)
  996 FORMAT (' Enter logical rotation flag > ',$)
  997 FORMAT (' Enter DEVICE TYPE > ',$)
  998 FORMAT (' Enter PORT ID > ',$)
  999 FORMAT (A)
      END
```

PLOT, SYMBOL, and NUMBER

Angular Letter Test

A=0,H=.75
A=45,H=1.00
A=90,H=1.25
A=135,H=1.50
A=180,H=1.75
A=225,H=2.00
A=270,H=2.25
A=315,H=2.50

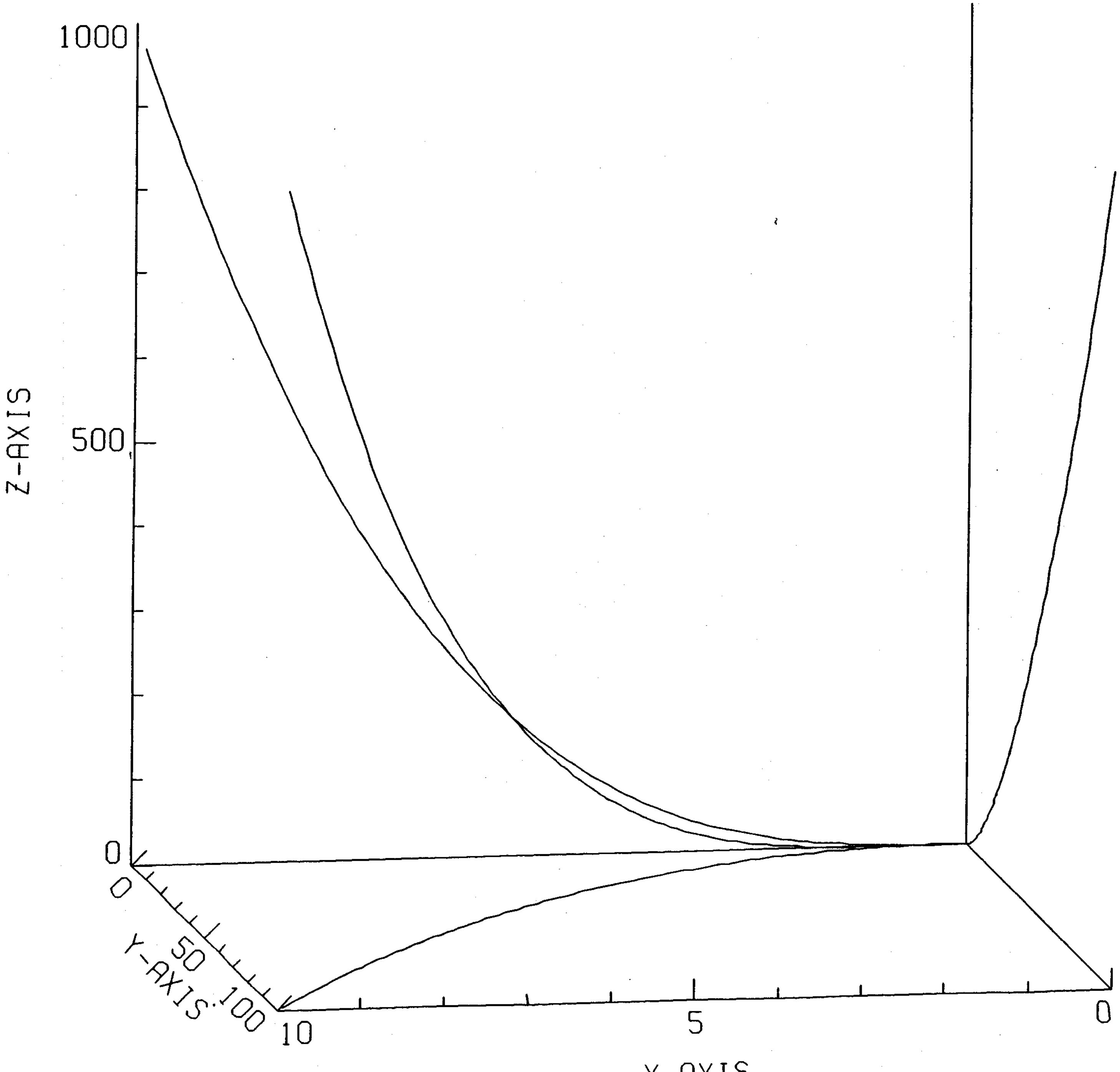
A = ANGLE IN DEGREES
H = RELATIVE CHARACTER HEIGHT

B.5 A SPACE CURVE

This program draws a space curve and its projections on the corrdinate planes.

```
C ------------------------------------------------------------------
C
C                          A SPACE CURVE
C
C
C                    GRAPHICS PACKAGE version 4.1
C
C
C                         VAX/VMS FORTRAN 77
C
C ------------------------------------------------------------------
C
      PROGRAM GP3CRV
      REAL CX(100), CY(100), CZ(100)
      CHARACTER*80 REPLY
C VAX/VMS RTL convention for STR$TRIM
      INTEGER*2 NLEN
C DEFINE THE ARRAYS
      DO 1 I=1,100
      C = (I-1)/10.0
      CX(I) = C
      CY(I) = C*C
    1 CZ(I) = C*C*C
C OPEN THE GRAPHICS PACKAGE
      CALL GPINIT ( 'A Space Curve')
   10 CALL PAGPAUS ( 'Enter THETA and PHI', REPLY)
      CALL STR$TRIM ( REPLY, REPLY, NLEN)
      IF ( NLEN.EQ.0) GO TO 50
      READ (REPLY(1:NLEN),99,ERR=10) TH, PH
      CALL NUPAGE
      CALL SET3D (, TH, PH)
      CALL SETXAX ( 0.0, 10.0, 0,, 10, 2)
      CALL SETYAX ( 0.0, 100.0, 0,, 10, 2)
      CALL SETZAX ( 0.0, 1000.0, 0,, 10, 2)
      CALL FRAM3D ( 'X-AXIS', 'Y-AXIS', 'Z-AXIS', 'A Space Curve')
      CALL SPCURV ( CX, CY, CZ, 100)
C PROJECTIONS ON THE REFERENCE PLANES
      CALL NEWPEN (3)
      IPEN = 3
      DO 21 I=1,100
      CALL PLTXYZ ( CX(I), CY(I),, IPEN)
   21 IPEN = 2
      CALL NEWPEN (7)
      IPEN = 3
      DO 22 I=1,100
      CALL PLTXYZ ( CX(I),, CZ(I), IPEN)
   22 IPEN = 2
      CALL NEWPEN (5)
      IPEN = 3
      DO 23 I=1,100
      CALL PLTXYZ (, CY(I), CZ(I), IPEN)
   23 IPEN = 2
      CALL NEWPEN (1)
      CALL ENDPAG ( .FALSE.)
      GO TO 10
C CLOSE THE GRAPHICS PACKAGE
   50 CALL ENDPLT
C
      STOP
   99 FORMAT (2F)
      END
```

A Space Curve



Orthogonal Projection: theta = 80.0 phi = 80.0

APPENDIX C

LIST OF RESERVED NAMES

C.1 MODULE NAMES

Labeled common block

Symbol	Defined By	Referenced By ...	
/SYMTAB/	SYMBLK	DEFSYM	SYMBOL

Cross reference by symbol

Symbol	Defined By	Referenced By ...			
ARYPLT	ARYPLT				
AUSCAL	AUSCAL	AUTXST	PICTUR		
AUTXST	AUTXST	PICT3D			
AUTYST	AUTXST	PICT3D			
AUTZST	AUTXST	PICT3D			
AXIS	AXIS				
AXLINE	AXLINE	FRAM3D	GRFRAM		
BASSET	NUMSTR	AUSCAL	BOUNDS	CONMAP	DATCON
		ELLIPS	FPNORM	FRAM3D	GRFRAM
		LABSYM	MINMAX	NCHAR	NDIGIT
		PLOT	SNUMBR	SYMBOL	TRUNCRAY
BEGPLT	BEGPLT	GPINIT			
BNDARY	BNDARY				
BOUNDS	BOUNDS				
CONMAP	CONMAP				
DATCON	DATCON	ARYPLT	ELLIPS	FNCON	FRAM3D
		GRFRAM	NUMDAT	PLSIGDAT	PLTDAT
		PLTXYZ	SPCURV	SUBJHV	
DATSET	DATCON	ARYPLT	PLTFAZ	PLTXYZ	
DEFSYM	DEFSYM				
DVLSET	SETREG	NUPAGE			
ELLIPS	ELLIPS				
ENDPAG	NUPAGE	BEGPLT			
ENDPLT	BEGPLT				
FACTOR	PLOT	BEGPLT			
FNCALN	FNCALN	CONMAP	SUBJHV		
FNCON	FNCON	PICT3D			
FOLDOR	GLTRAN	PLTDAT			
FPNORM	FPNORM	FNCON	NDIGIT	NDIGT	SCALUP
FRAM3D	FRAM3D	PICT3D			
GENCON	DATCON	NUPAGE			
GLTRAN	GLTRAN	ARYPLT	FNCON	FRAM3D	GRFRAM
		NUMDAT	PLSIGDAT	PLTDAT	PLTGLT
		PLTXYZ	SUBJHV		
GLTSET	GLTRAN	FRAM3D			
GPINIT	GPINIT				
GPSCAL	GPSCAL	AUSCAL			
GRDSET	BOUNDS	GRFRAM	PLTHVS		
GRFRAM	GRFRAM	PICT3D	PICTUR		
GRFSET	BOUNDS	FRAM3D	GRFRAM		
GRFSYZ	BOUNDS	NUPAGE			
GRIDX	GRIDX	GRFRAM			
GRIDY	GRIDX	GRFRAM			
GTITSET	NUPAGE	GRFRAM			
HISTRA	HISTRA				

HLRHVS	HLRHVS	THREED			
LABSYM	LABSYM	FRAM3D	GRFRAM	NUPAGE	PICTEXT
LINE	LINE				
LINRAY	LINRAY	PICTUR			
MINMAX	MINMAX	AUTXST	PICTUR		
MOSAIC	GRFRAM				
MRKTXT	GRFRAM				
NCHAR	NCHAR	FRAM3D	GRFRAM	NDIGIT	SNUMBR
NDIGIT	NDIGIT	FRAM3D	GRFRAM		
NDIGT	NDIGT	NDIGIT			
NEWARG	NEWARG	ARYPLT	AUSCAL	AUTXST	AXLINE
		BEGPLT	BOUNDS	CONMAP	DATCON
		ELLIPS	FNCON	GPSCAL	GRFRAM
		LABSYM	LINRAY	MINMAX	NUMDAT
		NUMSTR	NUPAGE	PICT3D	PICTUR
		PLERRBAR	PLOT	PLSIGDAT	PLTDAT
		PLTGLT	PLTXYZ	QMS_STORE	QMS_OPEN
		REG_OPEN	SETREG	SNUMBR	SPCURV
		SUBJHV	SYMBOL	SYMRAY	TEK_OPEN
		TEK_STORE	THREED	ZT8_LIMITS	
NEWPEN	PLOT	BEGPLT	CONMAP	GRFRAM	NUPAGE
		PLTGLT	PLTHVS		
NOPARI	NOPARI	GLTRAN			
NUMBER	NUMBER	ARYPLT	AXIS	FRAM3D	GRFRAM
		NUMDAT	SNUMBR	STRPLT	
NUMCEN	NUMBER	STRPLT			
NUMCON	NUMBER	SNUMBR			
NUMDAT	NUMDAT				
NUMP	NUMP	ARYPLT	BEGPLT	BOUNDS	DATCON
		ELLIPS	FNCALN	GLTRAN	HISTRA
		NUPAGE	PICTEXT	PICTUR	PLOT
		PLTDAT	SBCALN	SETREG	SNUMBR
		SYMRAY	ZT8_LIMITS		
NUMRIT	NUMBER	FRAM3D	GRFRAM	SNUMBR	
NUMSTR	NUMSTR	NUMBER	NUPAGE		
NUPAGE	NUPAGE	PICT3D	PICTEXT	PICTUR	
NXTSYM	SYMBOL	FNCON	FRAM3D	GRFRAM	SNUMBR
OLDPEN	PLOT	CONMAP	GRFRAM	PLTGLT	PLTHVS
ORDLYN	NUPAGE				
P3DSET	BOUNDS	AUTXST	FRAM3D	HLRHVS	
PAGLYN	NUPAGE				
PAGPAUS	NUPAGE				
PAGSET	NUPAGE	AUTXST	BNDARY	BOUNDS	GRFRAM
		PICTUR	SETREG		
PAGTITL	NUPAGE				
PANSET	NUPAGE	ARYPLT	AUTXST	BOUNDS	ELLIPS
		FRAM3D	GRFRAM	LABSYM	LINRAY
		PICTEXT	PICTUR	PLERRBAR	PLTDAT
		SNUMBR	STRPLT	SYMBOL	SYMRAY
PANTXT	GRFRAM				
PICT3D	PICT3D				
PICTEXT	PICTEXT				
PICTUR	PICTUR				
PJTSET	PLOT				
PLERRBAR	PLERRBAR	PLSIGDAT			
PLOSET	PLOT	NUPAGE			
PLOT	PLOT	ARYPLT	AXIS	AXLINE	BEGPLT
		BNDARY	FRAM3D	GRFRAM	GRIDX
		LINE	NUPAGE	PLERRBAR	PLTDAT
		PLTGLT	PLTHVS	PLTXYZ	STRPLT
		SYMBOL			
PLOTS	PLOT				
PLSIGDAT	PLSIGDAT				
PLTDAT	PLTDAT	ARYPLT	CONMAP	ELLIPS	HISTRA
		LINRAY	PLTFAZ	SYMRAY	
PLTEXT	PLOT	BEGPLT			
PLTFAZ	PLTFAZ				
PLTGLT	PLTGLT	FRAM3D	SPCURV		
PLTHVS	PLTHVS	THREED			
PLTMSG	PLOT	NUPAGE			
PLTOBJ	PLTDAT	ARYPLT			
PLTXYZ	PLTXYZ				

QMS_CLEAR	QMS_OPEN	BEGPLT			
QMS_CLOSE	QMS_OPEN	BEGPLT			
QMS_FLUSH	QMS_OPEN	BEGPLT			
QMS_LIMITS	QMS_OPEN	BEGPLT			
QMS_LINE	QMS_OPEN	BEGPLT			
QMS_OPEN	QMS_OPEN	BEGPLT			
QMS_PAUSE	QMS_OPEN	BEGPLT			
QMS_PLOT	QMS_OPEN	BEGPLT			
QMS_RESET	QMS_OPEN	BEGPLT			
QMS_STORE	QMS_STORE	QMS_OPEN	QMS_STORESTR		
QMS_STORESTR	QMS_STORESTR	QMS_OPEN			
REG3D	PICT3D				
REGION	NUPAGE	SETREG			
REGLYN	NUPAGE				
REGPIX	PICTUR				
REGTEXT	PICTEXT				
REGTITL	NUPAGE	PICT3D			
REG_CLEAR	REG_OPEN	BEGPLT			
REG_CLOSE	REG_OPEN	BEGPLT			
REG_FLUSH	REG_OPEN	BEGPLT			
REG_LIMITS	REG_OPEN	BEGPLT			
REG_LINE	REG_OPEN	BEGPLT			
REG_OPEN	REG_OPEN	BEGPLT			
REG_PAUSE	REG_OPEN	BEGPLT			
REG_PLOT	REG_OPEN	BEGPLT			
REG_RESET	REG_OPEN	BEGPLT			
SBCALN	SBCALN	DATCON	PLOT		
SCALDN	SCALUP	AUSCAL			
SCALE	SCALE				
SCALUP	SCALUP	AUSCAL			
SET3D	BOUNDS	PICT3D			
SETAFF	PLOT	AXLINE	BEGPLT	FRAM3D	NUPAGE
SETAXL	AXLINE	BOUNDS	FRAM3D	GRFRAM	
SETBAS	NUMSTR	BEGPLT			
SETCON	CONMAP				
SETCTL	NUPAGE	BEGPLT			
SETDAT	DATCON	BOUNDS	GRFRAM		
SETDVL	SETREG	BEGPLT			
SETEXT	SUBJHV				
SETGLT	GLTRAN	BOUNDS	GRFRAM	NUPAGE	
SETGRD	BOUNDS	GRFRAM			
SETPAG	NUPAGE	BEGPLT			
SETPAN	NUPAGE	GRFRAM			
SETPLO	PLOT	SETREG			
SETPLT	PLTDAT	AXLINE	PLTFAZ	SYMRAY	
SETREG	SETREG	NUPAGE			
SETROT	PLOT	BEGPLT	NUPAGE		
SETSCAL	AUSCAL	BEGPLT			
SETUSR	NUPAGE	BEGPLT			
SETXAX	BOUNDS	AUTXST	PICTUR		
SETYAX	BOUNDS	AUTXST	GRFRAM	PICTUR	
SETZAX	BOUNDS	AUTXST			
SNUMBR	SNUMBR	FNCON	FRAM3D	GRFRAM	
SNUMCN	SNUMBR				
SNUMRT	SNUMBR	FRAM3D	GRFRAM		
SPCURV	SPCURV				
STRPLT	STRPLT	FNCON			
SUBJHV	SUBJHV	THREED			
SYMBLK	SYMBLK	SYMBOL			
SYMBOL	SYMBOL	AXIS	FNCON	FRAM3D	GRFRAM
		LABSYM	LINE	NUMBER	NUMDAT
		NUPAGE	PICTEXT	PLERRBAR	PLTDAT
		PLTGLT	SNUMBR	STRPLT	
SYMCEN	SYMBOL	NUMBER			
SYMCON	SYMBOL	NUMBER			
SYMDAT	NUMDAT	ARYPLT	LINRAY		
SYMGLT	PLTGLT				
SYMRAY	SYMRAY				
SYMRIT	SYMBOL	NUMBER			
TEK_CLEAR	TEK_OPEN	BEGPLT			
TEK_CLOSE	TEK_OPEN	BEGPLT			
TEK_FLUSH	TEK_OPEN	BEGPLT			

TEK_GETTT	TEK_GETTT	TEK_OPEN			
TEK_LIMITS	TEK_OPEN	BEGPLT			
TEK_LINE	TEK_OPEN	BEGPLT			
TEK_MSG	TEK_OPEN	BEGPLT			
TEK_OPEN	TEK_OPEN	BEGPLT			
TEK_PAUSE	TEK_OPEN	BEGPLT			
TEK_PLOT	TEK_OPEN	BEGPLT			
TEK_RESET	TEK_OPEN	BEGPLT			
TEK_SETTT	TEK_SETTT	TEK_OPEN			
TEK_STORE	TEK_STORE	TEK_OPEN			
THREED	THREED	PICT3D			
TRUNCRAY	TRUNCRAY				
TXTLYN	GRFRAM	FNCON			
USRSET	NUPAGE				
WHERE	PLOT	ARYPLT	LINE		
XAXSET	BOUNDS	AUTXST	FRAM3D	GRFRAM	PICTUR
YAXSET	BOUNDS	AUTXST	FRAM3D	GRFRAM	PICTUR
YCLIPT	GLTRAN	PLTDAT			
ZAXSET	BOUNDS	AUTXST	FRAM3D		
ZT8_CLEAR	ZT8_LIMITS	BEGPLT			
ZT8_CLOSE	ZT8_LIMITS	BEGPLT			
ZT8_FLUSH	ZT8_LIMITS	BEGPLT			
ZT8_LIMITS	ZT8_LIMITS	BEGPLT			
ZT8_LINE	ZT8_LINE	BEGPLT			
ZT8_OPEN	ZT8_OPEN	BEGPLT			
ZT8_PAUSE	ZT8_LIMITS	BEGPLT			
ZT8_PLOT	ZT8_PLOT	BEGPLT			
ZT8_RESET	ZT8_LIMITS	BEGPLT			

APPENDIX D

USING THE GRAPHICS PACKAGE ON THE TARA VAX

D.1 GETTING STARTED

Version 4.1 of the graphics package is installed on the TARA VAX. A good way to get started is to print out a sample program and then to compile, link and run it. The source for each of the sample programs is in TARA$ROOT:[AMS.SAMPLES].

For example, to copy sample program 0 into your own (login) directory type:

```
$ COPY TARA$ROOT:[AMS.SAMPLES]GPSP0.FOR SYS$LOGIN
```

to get a listing of it type:

```
$ PRINT/HEAD GPSP0.FOR
```

to compile and link it type:

```
$ FORTRAN GPSP0
$ LINK GPSP0,AML$:GP4/LIB
```

The object library for linking with user programs is AML$:GP4.OLB. The linked program may be run and the plot output generated by:

```
$ RUN GPSP0
```

In response to the PORTID prompt

```
Enter PORTID >
```

type TT (your terminal); when the DEVICE TYPE is requested

```
Enter DEVICE TYPE >
```

type either TEK or REGIS depending on your terminal type. The illustration from sample program 0 should now appear on the screen of your terminal. If the display darkens hit the space bar once to refresh the screen. Type a <CR> to allow the program to continue and terminate normally.

After running the example you may wish to delete the object and executable files; to do this type:

```
$ DELETE GPSP0.OBJ;0,GPSP0.EXE;0
```

D.2 COMPUTER PORT IDENTIFICATION

The following computer ports or devices are defined as system logical names:

GT1	GRAPHICS TERMINAL 1 (DQ650M)
GT2	GRAPHICS TERMINAL 2 (DQ650M)
GT3	GRAPHICS TERMINAL 3 (DQ650M)
GT4	GRAPHICS TERMINAL 4 (DQ650M)
LP4	QMS Lasergrafix 1200 printer
TT	LOGIN TERMINAL

Also an explicit device may be used for the port identification e.g. TTB7: A null device may be specified as NL: A node name may be part of the computer port identification enabling output to be sent to a remote device.

D.2.1 Plot Files

If an unknown port identification is specified a file will be created; e.g. TTB7 would make a file TTB7.DAT. The file may later be copied to the correct device (any port) and the illustration produced. The filetype will be either .DAT or .TEK depending on the driver.

D.3 DEVICE TYPES

The following device types are available:

```
DQ640M   RETRO-GRAPHICS modification of a VT101
DQ650M   RETRO-GRAPHICS modification of a VT101
NULL     A null device type ( no output)
QMS      QMS Lasergrafix 1200 printer ( port LP4)
REGIS
TEK      For all the VT100 like terminals
VT240
VT241
VT640    RETRO-GRAPHICS modification of a VT100
ZETA8    Electromechanical multi-pen x-y plotter
```

with these types to be added:

```
ANADEX    printer
LA100     line printer
VERSATEC  printer
```

The terminal characteristics are automatically changed and and changed back for DQ640M, DQ650M, and VT640 terminals. The ZETA8 electromechanical plotter is located in NW17-186; it is in series before a VT640 graphics terminal and can only be used in tandem with it. There is also an ANADEX printer attached to the same terminal for copying the screen.

D.3.1 Absolute Dimensions

For the hard copy devices absolute dimensions may be useful on occassion.

To use calls which are in inches on the ZETA8 plotter,the page must be defined appropriately e.g.

```
CALL NUPAGE ( 8.5, 11.0,,,,, .TRUE.)     !or
CALL NUPAGE ( 11.0, 8.5)
```

To use calls which are in inches on the QMS printer, the page must be defined appropriately e.g.

```
CALL NUPAGE ( 7.794, 10.087,,,,, .TRUE.)     !or
CALL NUPAGE ( 10.087, 7.794)
```

To output centimeters instead of inches with the above definitions the scale factor should be changed e.g.

```
CALL FACTOR ( 1.0/2.54)
```

D.4 DEBUGGING

The VAX-11 Symbolic Debugger provides a way of examining the values of arrays and variable arguments used in the calling sequences; its use with FORTRAN is described in the FORTRAN USER'S GUIDE. However, there is not a debug version of the library. The comments in section 1.7 may facilitate program debugging.