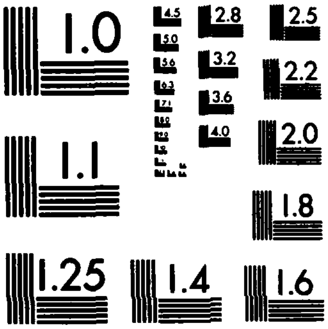
AD-A135 757   THE GRAPHIX LIBRARY - A COLLECTION OF INTERACTION   1/1
GRAPHICS ROUTINES FOR THE TEKTRONIX 4014(U) AIR FORCE
WRIGHT AERONAUTICAL LABS WRIGHT-PATTERSON AFB OH
UNCLASSIFIED   K D MACH OCT 83 AFWAL-TR-83-2071   F/G 9/2   NL

END
FILMED
1-84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

12

AFWAL-TR-83-2071

AD-A135 757

THE GRAPHIX LIBRARY - A Collection of Interaction Graphics Routines for the Tektronix 4014

Kervyn D. Mach

Components Branch
Turbine Engine Division

October 1983

Final Report for Period July 1980 - July 1982

Approved for public release; distribution unlimited

AERO PROPULSION LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Kervyn D Mack

KERVYN D. MACK
Aerospace Engineer

Charles E. Bentz

CHARLES E. BENTZ
Technical Area Manager

FOR THE COMMANDER

H I Bush

H. I. BUSCH
Director
Turbine Engine Division
Aero Propulsion Laboratory

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify___________, W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE *(When Date Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER
AFWAL-TR-83-2071

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*
The GRAPHIX Library
A Collection of Interactive Graphics Routines
for the Tektronix 4014

5. TYPE OF REPORT & PERIOD COVERED
Final Report
July 1980 - July 1982

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Kervyn D. Mach

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Aero Propulsion Laboratory (AFWAL/POTC)
Air Force Wright Aeronautical Laboratories
Wright-Patterson Air Force Base, Ohio 45433

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
P.E. 62203F, Proj 3066,
Task 306606, W.U. 30660602

11. CONTROLLING OFFICE NAME AND ADDRESS
AFWAL/POTC
WPAFB OH 45433

12. REPORT DATE
October 1983

13. NUMBER OF PAGES
14

14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*

15. SECURITY CLASS. *(of this report)*
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

The computer program (s) contained in this technical report is (are) theoretical and in no way reflect(s) any Air Force owned software programs.

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Computer Graphics
Tektronix 4014

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report describes GRAPHIX, a library of computer graphics routines for Tektronix 4014 terminals connected to CDC CYBER or Hewlett-Packard 1000 computers. Offering flexible output and ease of use with small memory overhead, GRAPHIX is an attractive alternative to PLOT-10 for many applications. An application guide is included.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

FOREWORD

This report describes the last activity conducted under Work Unit 30660602, Turbine Aeromechanical Analysis and constitutes the Final Report for that work unit. The work was conducted in the Turbine Engine Division of the Aero Propulsion Laboratory, Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1. INTRODUCTION

The GRAPHIX library is a collection of FORTRAN (and Pascal) callable subroutines for Tektronix 4014 terminals connected to CDC CYBER or Hewlett-Packard 1000 computers. GRAPHIX is somewhat simpler to apply than the Tektronix PLOT-10 library and requires less overhead. With GRAPHIX, it is possible to generate a simple X-Y plot with only one subroutine call.

GRAPHIX differs slightly from PLOT-10 in that it gathers all plot information into an internal buffer and transmits a single burst to the terminal rather than outputting a line at a time. Thus, nothing will seem to be happening for a few seconds after your program begins to execute. Then your plot will rapidly fill the screen.

The GRAPHIX routines were adapted from a set supplied with a large aerodynamic code. Consequently, some of the argument lists may seem peculiar. Nonetheless, the overall ease of use makes them attractive. In the following paragraphs, the basic routines will be described first, in the order in which they should be called. After that, the options will be described in alphabetical order.

2. APPLICATION GUIDE

2.1 SIMPLT - Do it all in one call. See Figure 3, Page 11.

CALL SIMPLT (X, Y, NXY, XTITLE, YTITLE, ILABEL, ISAME, IDEV)

X, Y : The arrays to be plotted, 100 max.
NXY : The number of points to be plotted from the X and Y arrays.
XTITLE,YTITLE : Axis labels. Up to eight characters each.

ILABEL = 1 to get an initial plot.
= 2 to add another curve on the previous grid.
ISAME = 1 to get the same scale on X and Y.
= 2 to scale X and Y independently.
IDEV = 2 always.*

* The original plan was to provide interfaces to other plotting devices, e.g. Calcomp, but that option was dropped. IDEV is still checked, however, and a value other than 2 suppresses the plot.

2.2 The basic do-it-yourself routines

PINIT initializes internal COMMON. It must be called only once, before any other graphics calls.

CALL PINIT (IDEV, IBAUD)

IDEV = 2 for Tektronix terminals. Any other value will suppress plotting.
IBAUD - 30, 120, or 960 depending on the transmission rate.

MNMX examines arrays to be plotted and finds their extremes.

```
CALL MNMX  (X, Y, NXY)
```

X, Y : The arrays to be plotted. There must be at least two elements in each, but there is no upper limit.
NXY : The number of elements in the X and Y arrays.

LIMITS computes scale factors from the extremes found by MNMX and adjusts them to get a neat grid.

```
CALL LIMITS  (ISAME)
    ISAME = 1 to get the same scale on X and Y.
          = 0 to scale X and Y independently.
```

PLON initializes the plot buffers and passes the scale factors computed by LIMITS to the routines which move the cursor. It must not be called before the scale factors are established and it must be called any time they are reset. It need not be recalled if you are drawing multiple plots with the same scale. PLON does not clear the screen.

```
CALL PLON  [No arguments]
```

CLEAR clears the screen. It may be called any time after PINIT.

```
CALL CLEAR  [No arguments]
```

LABEL draws the axes, draws grid lines, labels the grid increments, and titles the axes. Its use is optional. If you don't want axes, don't call it.

```
CALL LABEL  (XTTL, NCHX, YTTL, NCHY)
```

XTTL : X axis title, 60 characters (6 CDC or 30 H-P words) max.
NCHX : Number of characters in XTTL.
YTTL : Y axis title.
NCHY : Number of characters in YTTL.

LINE draws a line through your array of points and add symbols if you wish.

```
CALL LINE (X, Y, NPTS, LTP, ISYM)
```

X, Y : The arrays to be plotted.
There may be as few as one point each.
There is no upper limit.
NPTS : The number of points to be plotted from the X, Y arrays.
LTP = 0 No line. Use this to draw a single symbol at X, Y.
= 1 Solid line.
= 2 Dashed line on terminals with enhanced symbol at X, Y.

```
ISYM  = 0 No symbols
      = 1 Square
      = 2 Circle
      = 3 Triangle
      = 4 Plus sign
      = 5 X
        and so on, up to ISYM = 14.
```

HOME moves the cursor to the upper left corner of the screen. It may be called any time after PINIT.

```
CALL HOME  [No arguments]
```

PLOFF sets the terminal back to alpha mode and flushes the plot buffer. It must be called at the end of every plot.

```
CALL PLOFF  [No arguments]
```

Thus, the basic sequence for plotting a graph is:

```
CALL PINIT (2,120)
CALL MNMX (X, Y, NPTS)
CALL LIMITS (ISAME)
CALL PLON
CALL CLEAR
CALL LABEL (XTTL, NCHX, YTTL, NCHY)
CALL LINE (X, Y, NPTS, LTP, ISYM)
CALL PLOFF
```

This is what SIMPLT looks like inside.

After plotting a basic graph, you may add to it by calling LINE again with another pair of arrays, by adding legends and labels, or what have you. Just be sure to call PLOFF when you're done.

2.3 Options

AMODE sets the terminal to alpha mode.

```
CALL AMODE [No arguments]
```

ATKN performs parabolic or linear interpolation on given arrays.

```
YA = ATKN (X, Y, NP, NI, XA)

     X : Array of independent variable.
     Y : Array of dependent variable.
     NP : Length of X and Y arrays [= 2].
     NI : Not used.
     XA : Value of independent variable at which interpolation
          is performed.
```

BELL rings the bell on the terminal, but only after PINIT has been called.

CALL BELL [No arguments]

CHAPLT prints a line of text on the screen. PLON must have been called.

CALL CHAPLT (XREF, YREF, CSIZ, CROT, JUST, TITLE, NCHT)

```
XREF, YREF : Initial coordinates for positioning the line of
             text.  XREF and YREF are in user coordinates,
             i.e., the same units as X, Y in a call to LINE.
CSIZ = 0.1  Sets the smallest character size.
     = 0.12 Sets the next size.
     = 0.15 Sets the next to largest.
     = 0.2  Sets the largest size.
CROT = 0.0  Prints horizontally.
     = 450.0 Prints vertically, from the top down.
JUST is a two-digit integer specifying the location of the
     printed line relative to (XREF, YREF).  Each digit may
     be any of 1, 2, or 3.
     First digit = 1 Start printing XREF.
                 = 2 Center the line at XREF.
                 = 3 End the line at XREF.
     Second digit = 1 Bottom of line is at YREF.
                  = 2 Center of line is at YREF.
                  = 3 Top of line is at YREF.
     Thus, permissible values for JUST are 11, 12, 13,
     21, 22, 23, 31, 32, 33.
TITLE : The line to be printed, 60 characters (6 CDC or 30 H-P
      words) max.
NCHT : The number of characters in TITLE.
```

CURSOR, when called, displays the cross hairs on the screen. Move them to any location and strike any character and the carriage return. The cursor location (user coordinates) and the character struck will be returned to the calling program. Use after you have drawn a plot and called PLOFF.

CALL CURSOR (XC, YC, KC)

XC, YC : Cursor location in user coordinates.
KC : The character struck, in ASCII integer representation.

DATLIM is an alternative to MNMX and LIMITS, if you choose your extrema carefully.

CALL DATLIM (XMN, XMX, YMN, YMX)

where the arguments are the extrema of the X and Y arrays, supplied by you. Calling LIMITS afterward is a good idea, to get neat scales.

DRU draws a line from wherever the cursor happens to be to the location you specify. The terminal must be in graphics mode (see GMODE, below).

```
CALL DRU (UX, UY)
```

UX and UY are the user coordinates of a screen location.

GETLIM returns the current minimum and maximum values of X and Y.

```
CALL GETLIM (XMIN, XMAX, YMIN, YMAX)
```

GMODE sets the terminal to graphics mode. It is used primarily with DRU and MVU.

```
CALL GMODE [No arguments]
```

IBOUND allows you to change the inner graph boundary. For example, you could use only the right half of the screen. Call before LIMITS.

```
CALL IBOUND (XI1, XI2, YI1, YI2)
```

See Figure 1 for definitions. Defaults are:

XI1 = 1.0
XI2 = 15.0
YI1 = 0.0
YI2 = 10.0

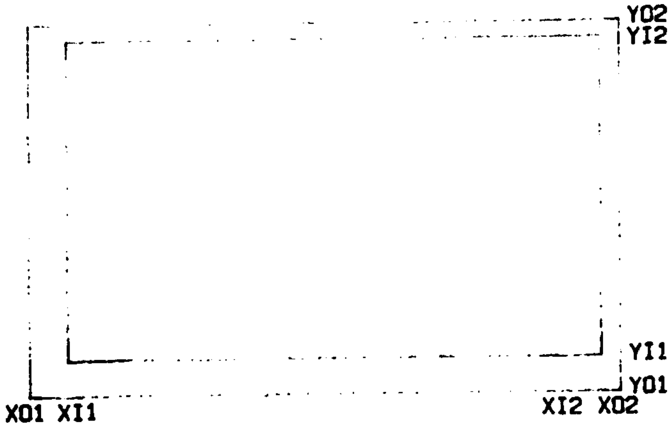


Figure 1. IBOUND Nomenclature

The defaults specify 14 vertical grid lines and ten horizontal grid lines (including axes). These fit nicely on the Tektronix screen.

INC lets you change the increment by which the X and Y arrays are plotted. Call before MNMX and LINE.

```
CALL INC (IX, IY)
```

IX, IY are the increments by which the X and Y arrays are plotted. Default is 1. To plot every other point, set to 2.

MVU moves the cursor to the location you specify. Precede the first call to MVU with a call to GMODE.

```
CALL MVU (UX, UY)
```

UX, UY : The coordinates of a screen location in user coordinates.

OBOUND allows you to change the outer plot boundary (See sketch under IBOUND). Call before LIMITS.

```
CALL OBOUND (X01, X02, Y01, Y02)
```

Default values are:

```
X01 = 0.0
X02 = 15.5
Y01 = -1.12
Y02 = 10.5
```

Think of the units of X01, XI1, etc. as representing grid divisions.

Multiple plots may be drawn on the same page by manipulating the plot boundaries with OBOUND and IBOUND. For example,

```
CALL OBOUND (0.0, 10.0, 0.0, 14.0)
CALL IBOUND (1.0, 10.0, 8.0, 14.0)
CALL PLON
```

Then call other routines as necessary to draw the first plot.

```
CALL IBOUND (1.0, 10.0, 1.0, 7.0)
CALL PLON
```

And call other routines to complete the second plot. See Figure 2, below, Figure 4, Page 13.

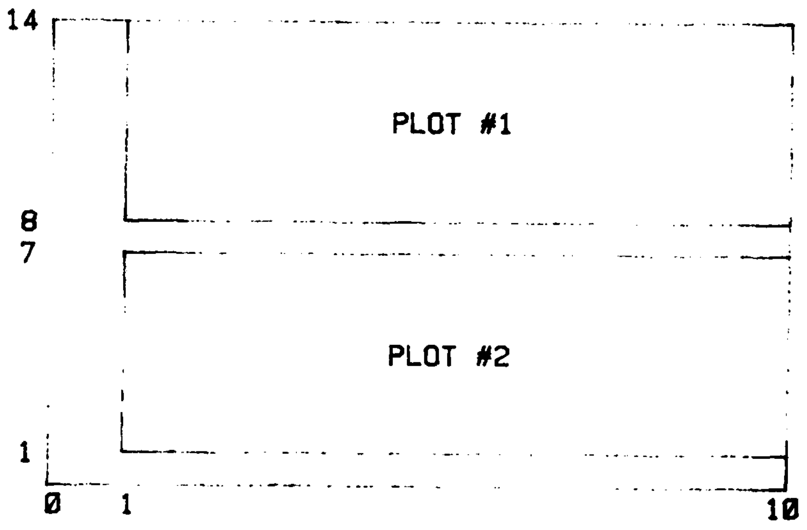


Figure 2. Multiple Plots with OBOUND and IBOUND

PAGE will make the program wait while you admire the picture, make a copy, or whatever. When you type in any character followed by a carriage return, it clears the screen and returns control to the calling program.

CALL PAGE (IPAUSE)

IPAUSE = 1 : Ring the bell, wait for input. When something is sent, clear the screen and return.
= 0 : Clear the screen and return.

PBELL rings the bell on the terminal, but does not require PINIT to be called first, i.e. PBELL may be called any time.

PCLEAR clears the screen. Like PBELL, it may be called any time.

CALL PCLEAR [No arguments]

PIE draws circles or pie charts with optional shading.

CALL PIE (XC, YC, RADIUS, PERC, SPAC, NPERC)

XC, YC : The center of the circle, in user coordinates.
RADIUS : The radius of the circle, in user coordinates.
PERC : An array of percentages for the pie chart. For a plain circle, there is only one element, whose value is 100.0.
SPAC : An array of spacing increments in user coordinates for shading the circle or pie chart. For a plain circle, there is only one element and its value is the same as RADIUS.
NPERC : The number of elements in PERC and SPAC. For a plain circle, NPERC = 1.

For a shaded circle, set SPAC to the interval you want between rings, e.g. 0.1, leave PERC at 100.0 and NPERC at 1. PIE will draw concentric circles with radius decreasing by SPAC until the smallest radius is less than or equal to SPAC. See Figure 5, Page 15.

SETSIZ lets you change the size of the printed characters.

CALL SETSIZ (ISIZE)

ISIZE = 1 Smallest

= 4 Largest

3. WHERE TO FIND IT

The GRAPHIX library is maintained on permanent file GRAPHIX, CY = 990, ID ID = P820317, SN = AFML, at the ASD Computer Center. A modified version is available for Hewlett-Packard 1000 computers. Source code (Fortran and Compass for CDC, Fortran and Pascal for H-P) is available from the author.

Following are a few examples illustrating the basic routines and some of the options in GRAPHIX. The Fortran program FIGRI (Page 9) and the Pascal program PARABOLA (Page 6) will both produce Figure 1. Program TWO (Page 12) shows how to manipulate the plot boundaries with OBOUND and IBOUND. Program THREE (Page 14) illustates the use of subroutine PIE.

```
FTN4,L
      PROGRAM  PARAB
      DIMENSION  X (26), Y (26)
        DO 1 I    = 1, 26
        X(I)      = 0.2 * FLOAT (I - 1)
        Y(I)      = 1.0 + (X(I) - 2.5) ** 2
    1   CONTINUE
C
C         H-P FORTRAN ALLOWS ONLY FIVE-LETTER MODULE NAMES
C
      CALL SMPLT (X, Y, 26, 8H X AXIS , 8H Y AXIS , 1, 1,
     $ 2)
      READ *, IGO
      END
```

```
 PROGRAM PARABOLA (INPUT, OUTPUT);
 TYPE PBL = ARRAY [1..26] OF REAL;
      AXIS_LABEL = PACKED ARRAY [1..8] OF CHAR;
      SHORTI = -16000..16000;
 VAR  X, Y : PBL;
      I : SHORTI;
      XLAB, YLAB : AXIS_LABEL;
      C : REAL;

 PROCEDURE SMPLT (X, Y : PBL; NPTS : SHORTI;
   XLBL, YLBL : AXIS_LABEL; ILBL, LNSCL, IDEV : SHORTI);
   EXTERNAL;

 BEGIN
   C := 0.0;
   FOR I := 1 TO 26 DO
     BEGIN
       X[I] := 0.2 * C;
       Y[I] := 1.0 + SQR(X[I] - 2.5);
       C := C + 1.0
     END;
(* PLOT *)
 SMPLT (X, Y, 26, ' X AXIS ', ' Y AXIS ', 1, 1, 2);
 READLN (I)
END.
```

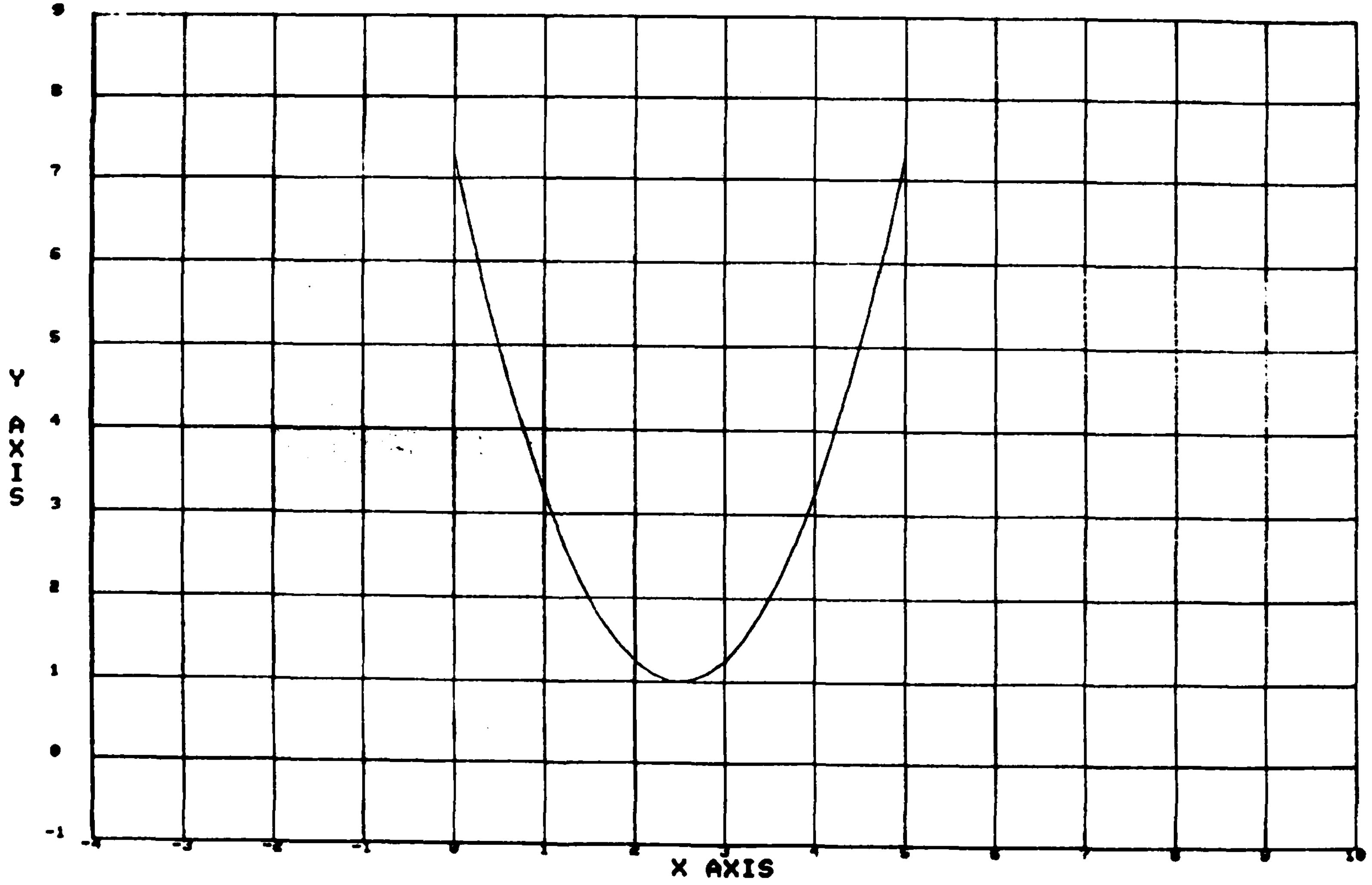


Figure 3. SIMPLT Output

```
      PROGRAM  TWO (INPUT, OUTPUT)
      DIMENSION   X (25), Y (25)
      A          = 1.0 / 12.0
      B          = A - 1.0
        DO 1 I     = 1, 25
        X(I)       = B + A * FLOAT(I)
        Y(I)       = 1.0 + X(I) ** 2
    1   CONTINUE
C
      CALL PINIT (2, 120)
      CALL MNMX (X, Y, 25)
      CALL OBOUND (0.0, 8.0, 0.0, 14.0)
      CALL IBOUND (1.0, 8.0, 8.0, 14.0)
      CALL LIMITS (1)
      CALL PLON
      CALL PAGE (1)
      CALL LABEL ("X AXIS", 6, "Y AXIS", 6)
      CALL LINE (X, Y, 25, 1, 1)
      CALL HOME
      CALL PLOFF
      CALL MNMX (Y, X, 25)
      CALL IBOUND (1.0, 6.0, 1.0, 7.0
      CALL LIMITS (1)
      CALL PLON
      CALL LABEL ("Y AXIS", 6, "X AXIS", 6)
      CALL LINE (Y, X, 25, 1, 1)
      CALL HOME
      CALL PLOFF
      CALL PAGE (1)
      END
```

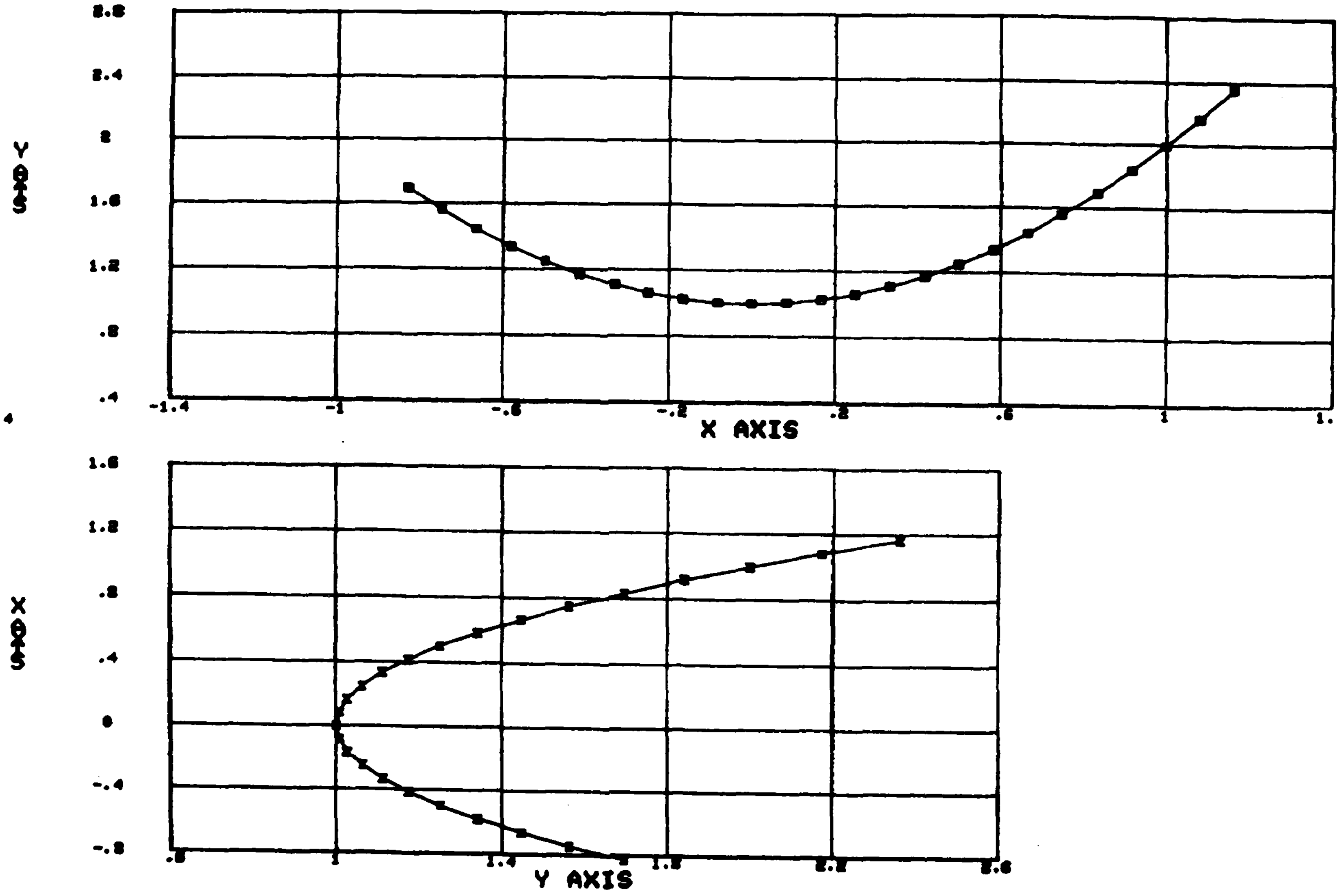


Figure 4. Using OBOUND and IBOUND

```
      PROGRAM  THREE (INPUT, OUTPUT)
      DIMENSION   D (4), PERC (3), SPAC (3)
      DATA     D / 0.0, 10.0, 0.0, 10.0 /
      DATA     PERC / 8.0, 15.0, 23.0 /
      DATA     SPAC / 3 * 2.0 /
C
      CALL PINIT (2, 120)
      CALL MNMX (D(1), D(3), 2)
      CALL LIMITS (1)
      CALL PLON
      CALL CLEAR
C
C        SIMPLE CIRCLE (R = 2.0, XC & RC ARE 2.5, 7.5)
C
      CALL PIE (2.5, 7.5, 2.0, 100.0, 2.0, 1)
C
C        CONCENTRIC CIRCLES (RMAX = 2.0, DR=0.4,
C        XC & YC ARE 7.5, 7.5)
C
      CALL PIE (7.5, 7.5, 2.0, 100.0, 0.4, 1)
C
C        PIE CHART WITHOUT SHADING
C        (R = 2.0, XC & YC ARE 2.5, 2.5)
C
      CALL PIE (2.5, 2.5, 2.0, PERC, SPAC, 3)
C
C        SAME PIE CHART WITH SHADING.
C        (XC & YC ARE 7.5, 2.5)
C
      SPAC(1)   = 0.4
      SPAC(2)   = 0.2
      SPAC(3)   = 0.1
C
      CALL PIE (7.5, 2.5, 2.0, PERC, SPAC, 3)
C
      CALL PLOFF
      CALL PAGE (1)
      END
```

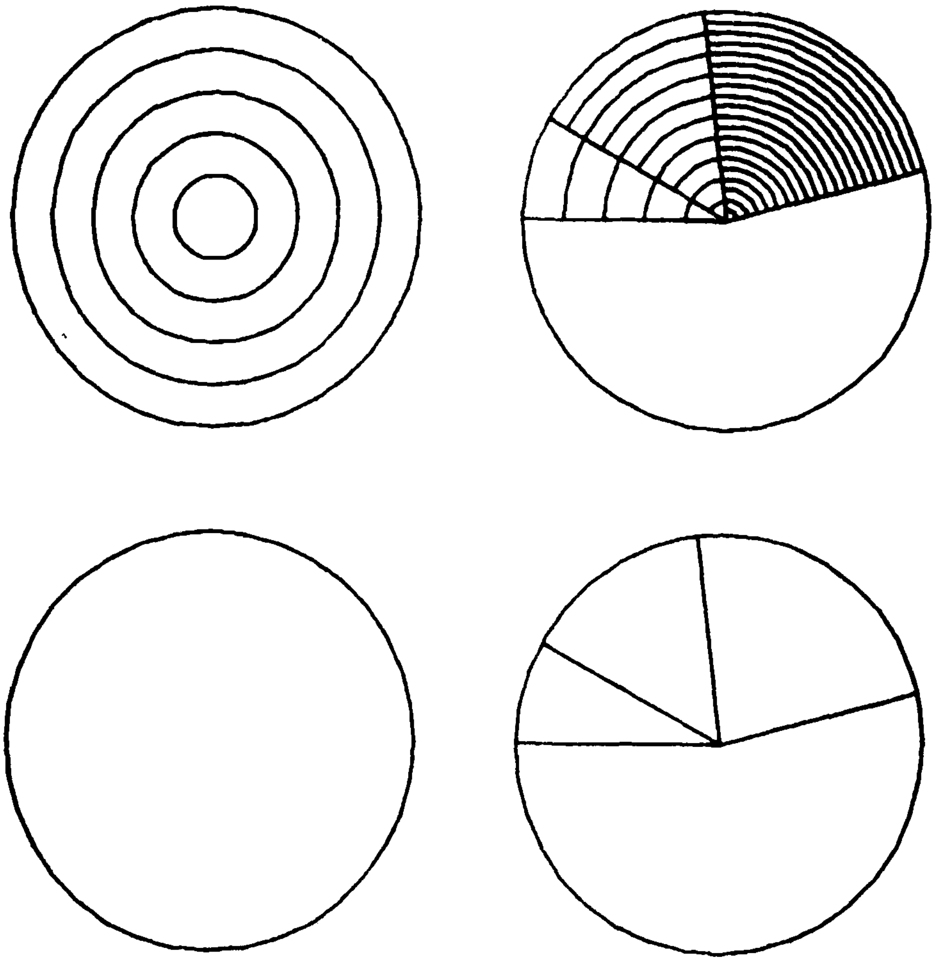


Figure 5. Using PIE