X-626-75-212
PREPRINT

NASA TM X- 70958

GRAY-SHADING FOR THE SD-4060 GRAPHICS DEVICE

C. GLOECKLER

AUGUST 1975

GSFC

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

GRAY-SHADING FOR THE SD-4060 GRAPHICS DEVICE

By

C. Gloeckler

August 15, 1975

NASA/Goddard Space Flight Center
Greenbelt, Maryland

TABLE OF CONTENTS

I. DESCRIPTION

GRAYS is a Fortran program which will generate gray shading for the SD-4060 graphics device. The program produces 10 shades of gray ranging from no shading at all to complete coverage of the film frame (see fig.1). The graphing capabilities are summarized and illustrated in figures 1 to 30. The figures displayed are representative of the microfilm output, but the distinction between various intensities is much clearer on the film, expecially at the more intense shading.

The general idea is that given an X-coordinate, X, a width, WIDTH, a Y-minumum, YMIN, and Y-maximum, YMAX, the area indicated below is shaded by one of the gray shades from 0 to 9.

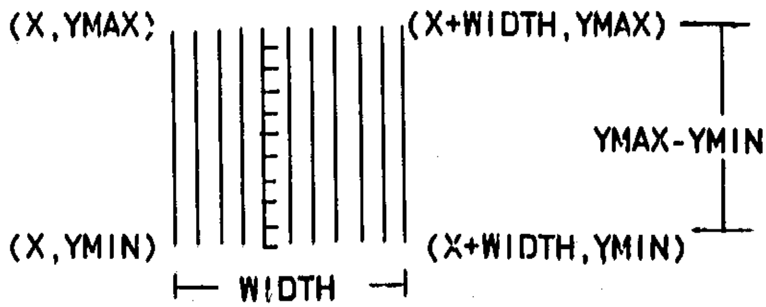

DIAGRAM 1

II. SHADING METHOD

The shading is accomplished by drawing one or more verticle lines from YMIN to YMAX in the interval determined by WIDTH. The number of lines, distance between lines, width of lines and concurrency of lines, are all varied to get the different shades. For some shades, short horizontal lines are added at equally spaced intervals from YMIN to YMAX as indicated above. The actual shading is based on a full working area of 4096 addressable points in the horizontal direction and 3072 in the vertical with a raster unit defined as the distance between any 2 addressable points in either the vertical or horizontal direction.

The test shading was done on a width of 15 raster units. The shading was then modified to give quality plots for widths varying from 11 to 21 rasters and acceptable shading from 8 to 10 rasters. The minimum allowable shading width is 8 rasters. To get shading for width greater than 21 rasters the shading has been generalized by taking the given raster width and plotting adjacent blocks each of equal raster width less than or equal to 21 until the given width is covered. The width of the block is the largest width ≤ 21 which will cover the given width in an integral number of blocks. Hence, to plot a given width of 22 rasters, two blocks of 11.0 rasters are plotted adjacent to each other while for a given width of 50 rasters, three blocks of 16.666 are plotted. Figures 29 and 30 illustrate this method by covering a width of 26 rasters with 2 blocks of 13 rasters.

III. GENERAL PROGRAM REQUIREMENTS

The initialization of the SD-4060 plot package is necessary as described in the Programmer's Reference Manual for the Integrated Graphics Software System[1]. The minimum initialization can be done by the following two statements:

Statement 1: DIMENSION AMODE(200)

Statement 2: CALL MODESG (AMODE, ITAPE)

Statement 1 reserves space for the 4060 MODE array which contains flags for various options to the 4060 plot package while Statement 2 actually does the initialization of the MODE array and opens the output data set, ITAPE.

The last call after all plotting is completed is to close the output data set and terminate the graphics routines.

Statement 3: CALL EXITG (AMODE)

It is assumed the user is familiar with the plot routines of the SD-4060 Software System but it is advisable to the new or occasional user to review the manual for the description of the various calls.

IV. GENERAL PROGRAM CONSTRAINTS

A. Definitions

Three optional statements in the SD-4060 system often included in programming for the 4060 plot package are listed below to define the variables discussed in part B of this section.

```
Statement 4:  CALL SETSMG (AMODE,19,A)

              Default A = 4095.

Statement 5:  CALL OBJCTG (AMODE,C,Y1,D,Y2)

              Default C = 0.
                     Y1 = 0.
                      D = 4095.
                     Y2 = 3071.

Statement 6:  CALL SUBJEG (AMODE,E,Y3,F,Y4)

              Default E = 0.
                     Y3 = 0.
                      F = 4095.
                     Y4 = 3071.
```

Statement 4 sets the normalization factor for the plot to 4095./A in the horizontal direction. Statement 5 defines the actual area of the scope to be used for this plot or portion of plot. Statement 6 redefines the limits of the area of the plot determined by Statement 5 to be in the user's coordinate system.

B. Constraints

The variable, WIDTH, is the width of the space to be shaded in <u>subject space</u> coordinates and should fulfill the following conditions:

1. WIDTH $\geq$ 8.*A/4095.*(F-E)/(D-C) where A,C,D,E,F are defined in statements 3,4 and 5 above.
2. The X-axis scale must be linear because of the method of shading.
3. The intensity values must be specified to range from 0 to 9, with 9 corresponding to the most intense shading. If any value of intensity is greater than 9, the most intense shading is used..
4. WIDTH = Δx

   where Δx is the difference between 2 adjacent X co-ordinates and should be constant for the arrays being plotted in any call to GRAYS. The program allows WIDTH to vary from call to call, but for each call to GRAYS the width used is the same for all points to be plotted. If Δx is constant for all points, and if WIDTH is set equal to Δx, and if there are no missing points, then there will be complete coverage in the x-direction. If WIDTH $<\Delta x$, there will be gaps between points and if WIDTH $>\Delta x$, there will be overlap shading. Hence care should be taken to specify WIDTH = Δx.

Shading which requires the use of varying widths poses the problem of having to compare intensities essentially using a different scale. That is, for different widths, different scales may be chosen from any of those in figures 1 to 28. For example, intensity 3 in figure 1 (21 raster width) is different for intensity 3 in figure 27 (8 raster width). The practice of using varying raster widths to shade on the same graph should be used with care. It is not recommended to change the width for each point.

To find the raster width, WIDE, of the area to be shaded given the width in subject space coordinates, WIDTH, use the following equation:

WIDE = WIDTH*4095./A*(D-C)/(F-E)

Then figures 1 to 28, which illustrate the shading for various raster widths, can be used to get an idea of the shading which will be selected.

As stated before, optimum shading is obtained for a raster width greater than 10, but acceptable shading can be obtained for raster widths of 8, 9 or 10.

V. MODES OF OPERATION

Depending on how the input arrays are dimensioned, GRAYS can operate in any of 3 modes.

A. Mode 1: Non-Dimensioned Variables

In this case GRAYS is called each time a point is to be plotted. There are no dimension statements necessary for GRAYS and the calling sequence is:

CALL GRAYS(YMIN,YMAX,WIDTH,X,INTS,1,AMODE,1,1)

All variables are in subject space units.

-YMIN and YMAX are the lower and upper bounds of the interval to be shaded.

-WIDTH is the width of the area to be shaded.

-X is the X-coordinate of the left side of the area to be shaded.

-INTS is the intensity value which determines the shade ($0 \le INTS \le 9$).

-AMODE is the SD-4060 mode array.

Care should be taken to have WIDTH the same in each call to GRAYS unless the user is certain it should vary (see Section IV).

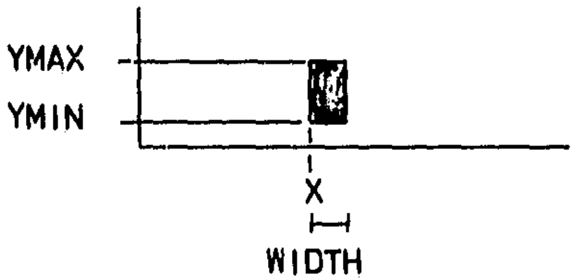

DIAGRAM 2:
One Call to GRAYS Plots 1 Point

B. Mode 2: Singly-Dimensioned Arrays

GRAYS will plot a series of points given the necessary coordinates YMIN(I), YMAX(I), X(I), INTS(I) for each point I. The calling sequence is:

```
DIMENSION YMIN(MR),YMAX(MR),X(MR),INTS(MR)
CALL GRAYS (YMIN,YMAX,WIDTH,X,INTS,1,AMODE,M,1)
```

All variables are in subject space coordinates.

-YMIN(I) and YMAX(I) are the lower and upper bounds of the area to be shaded for the I<u>th</u> point.

-WIDTH is the width of the area to be shaded.

-X(I) is the X-coordinate of the left side of the area to be shaded for the I<u>th</u> point.

-INTS(I) is the intensity value which determines the shading of the I<u>th</u> point. ($0 \le INTS(I) \le 9$)

-AMODE is the SD-4060 mode array.

-M is the number of points to shade in this call. ($0 \le M \le MR$)

In these definitions I=1,...,M

Thus it is possible to plot points anywhere in the subject space as is indicated by diagram 3.

Note that as a special case, if all X(I) are the same and YMIN and YMAX vary, a column will be plotted as in diagram 4.

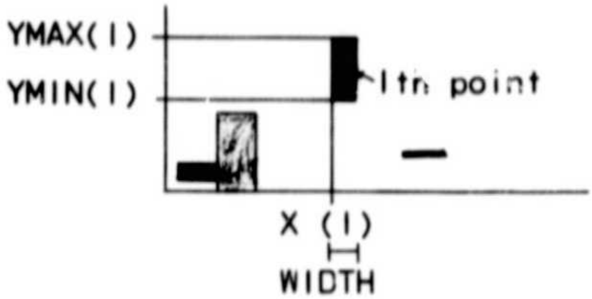

DIAGRAM 3:
One Call to GRAYS Plots M Points

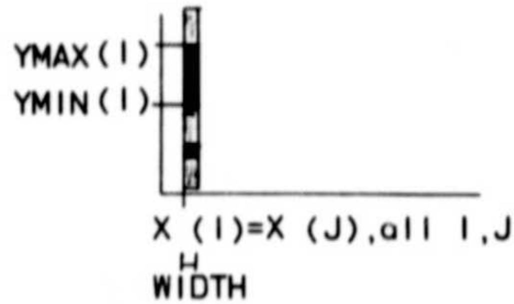

DIAGRAM 4:
One Call to GRAYS to Plot a Column

In order to plot a particular row or band, two methods could be used. Since to plot a band, all YMIN(I) are equal to the lower bound and all YMAX(I) are equal to the upper bound of the band, the same calling sequence described for this mode can be used where YMIN(I)=YMIN(J) and YMAX(I)=YMAX(J) for I and J = 1,...,M. But since all the YMIN(I) are equal and all the YMAX(I) are equal, it is not necessary to dimension YMAX or YMIN and then the calling sequence is

```
DIMENSION X(MR),INTS(MR)
CALL GRAYS (YMIN,YMAX,WIDTH,X,INTS,M,AMODE,1,MR)
```

The variables are as described previously for MODE 2. Diagram 5 illustrates the plotting of a single band.

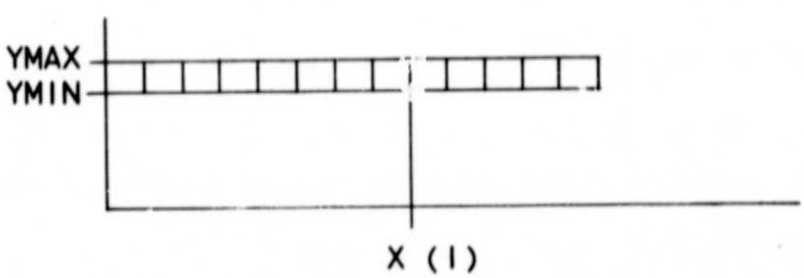

Diagram 5: One Call to GRAYS to Plot a Row

C. Mode 3: Doubly-Dimensioned Arrays

For efficiency in storing and plotting data, GRAYS is capable of plotting many bands across the frame in one call. The plotting proceeds from band 1 to band 2, etc. until all bands have been plotted. Within a band, the shading is done at the appropriate x-coordinate. The plotting of a band is illustrated below.

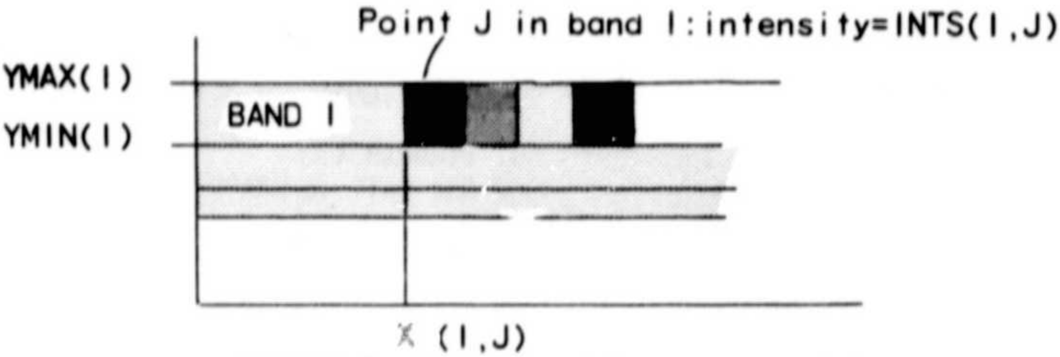

DIAGRAM 6: Plotting of Points in Band I

The calling sequence is

```
DIMENSION YMIN(MR),YMAX(MR),X(MR,MC),INTS(MR,MC),NPTS(MR)
CALL GRAYS(YMIN,YMAX,WIDTH,X,INTS,NPTS,AMODE,MR,MC)
```

-MR is the number of bands to be plotted.

-MC is the maximum number of points in each band.

-YMIN(I) is the lower bound of band I where I=1,MR.

-YMAX(I) is the upper bound of band I where I=1,MR.

-WIDTH is the width of the area to be shaded (in the direction of the abscissa).

-X(I,J) is the x-coordinate of the left side of the area to be shaded for the Jth point in band I. (The x-coordinate of the right side is X(I,J)+WIDTH).

-INTS(I,J) is the intensity of the Jth point in band I. (The range of permitted values is 0-9).

-NPTS(I) is the actual number of points to be plotted in band I (0≤NPTS(I)≤MC for all I).

VI. PROGRAM-RUN STATISTICS

A. Timing

IBM 360-91: The average time to plot one point is

CPU: 0.00094 seconds

IO: 0.00055 seconds

Hence in 1 minute of CPU and .6 minutes of IO approximately 63600 points can be plotted.

IBM 360-75: The average time to plot one point is

CPU: 0.003 seconds

IO: 0.0006 seconds

On this machine in 1 minute of CPU and .2 minutes of IO approximately 20,000 points can be plotted.

This is based upon plotting the 49000 points shown in figures 1 to 28. There is equal distribution of intensities and widths so the actual timing for a particular set of data may vary slightly according to the raster width used and average intensity to be plotted.

B. Size

Complied by the Fortran H compiler with OPT=2, GRAYS uses 4834 decimal bytes of core or approximately 4.8K.

VII. PUBLICATION PRINTS

Although 16 millimeter film is quite good for viewing plots on a microfilm reader, it does not produce acceptable results when an enlarged reproduction is made. Perhaps with sophisticated equipment which would lighten the dark areas and/or darken the light areas, good quality enlargements can be made. This has not been tried.

It is recommended then that 35 millimeter film be obtained of any plot to be reproduced. Figures 1-30 were output by the SD-4060 onto 35 mm film and then enlarged to 8"x10" fine grain positive contacts using a 50 mm lens at 2.6 seconds and an exposure of f/11. These were then contacted onto matte paper at an exposure of 3/10 sec. This method and settings may be used as a guide in obtaining enlarged prints.

VIII. EXAMPLES

A. Doubly Dimensional Arrays

Odd numbered plots in Figures 1-29: These plots were all produced using doubly-dimensioned arrays with 10 bands each containing 60 points plotted across the page. The Fortran code is outlined below for Figure 1:

```
      DIMENSION YMIN(30),YMAX(30),X(30,100),INTS(30,100),NPTS(30)
      DIMENSION Z(200)               (amode array)
      WIDTH = 21.
      CALL MODESG(Z,0)
      CALL SETSMG(Z,19,4095.)
      CALL SUBJEG(Z,0.,0.,4095.,3071.)
      CALL OBJCTG(Z,0.,0.,4095.,3071.)
C
C..SET YMIN(I) AND YMAX(I).  YMAX(I)-YMIN(I) = 99 RASTERS
C
     DO 100I = 1,10
     YMIN(I) = 100.*I-50.
100  YMAX(I) = YMIN(I)+99.
C
C..FILL X,INTS AND NPTS FOR 10 BANDS EACH
C..CONTAINING 60 POINTS
```

```
      DO 200 J = 1,10
      NPTS(J) = 60
      NPTS(J+10) = 0
      NPTS(J+20) = 0
      DO 200 I = 1,60
      X(J,I) = (I-1)*WIDTH
      INTS(J,I) = J-1
  200 CONTINUE
C
      CALL GRAYS(YMIN,YMAX,WIDTH,X,INTS,NPTS,Z,30,100)
      CALL EXITG(Z)
      STOP
      END
```

Note here that the dimensions were set up to handle 30 bands each containing a maximum of 100 points. Since only 10 bands were to be plotted NPTS(J) = 0, J>10. The dimensions of 30 and 100 were necessary in the call to GRAYS in order to access the correct values.

B. Singly Dimensional Arrays Used to Plot 3 Columns

Figure 31 is based on a shading of 18 rasters with a scale drawn at the right of the grid. The scale actually is 3 points wide (3 blocks of 18 rasters) covering 52 rasters in the horizontal direction and each shade is 99 rasters high. The following code was used:

```
      DIMENSION Z(200)
      DIMENSION VAL(20),X(10),IVAL(10)
      CALL MODESG(Z,0)
      CALL OBJCTG(Z,0.0,0.0,4095.,3071.)
      CALL SUBJEG(Z,0.0,0.0,4095.,3071.)
```

```
      OFF=0.0
C..Do 920 three times, once for each column.
      DO 920 J = 1,3
      DO 910 I = 1,10
      IVAL(I) = I-1
      X(I) = 4035.+OFF
      VAL(I) = 1030.+(I-1)*100.
      VAL(I+10) = VAL(I)+99.
  910 IF(J.EQ.3) CALL NUMBRG(Z,4020.,VAL(I)+40.,1,IVAL(I))
      CALL GRAYS(VAL(1),VAL(11),18.0,X,IVAL,1,Z,10,1)
  920 OFF=OFF+18.
```

C. Singly Dimensional Arrays to Plot Bands of Data

The data points plotted in figure 31 were plotted with the following code:

```
      DIMENSION Z(200)
      DIMENSION EMIN(28),EMAX(28),XR(20),IV(20),YMN(4)
      DATA YMN/130.0,855.,1580.,2305./
      CALL MODESG(Z,0)
C..REASSIGN OBJECT SPACE FOR EACH OF THE 4 PARTS OF THE PLOT
      DO 300 I = 1,4
      CALL OBJCTG(Z,200.,YMN(I),3900.,YMN(I)+665.)
      YMIN = ALOG10(.4)
      CALL SUBJEG(Z,0.0,YMIN,210.,3.)
      DO 250 IN = 1,28
  250 CALL GRAYS(EMIN(IN),EMAX(IN),1.0,XR,IV,Z,1,20)
  300 CONTINUE
```

IX. FIGURES

Figures 1-30 are for display purposes of shading. The widths indicated are raster units per point in the horizontal direction. The microfilm resolution between shades is much better than is illustrated, especially at the higher intensities. It is seen here that the height of each point makes a difference at the higher intensities on these enlargements. For instance, a difference between the two darkest shades is distinguishable on the even numbered plots where the height is equal to the width, more so than in the odd numbered plots where the height was fixed at 99 rasters. On the microfilm all intensities are easily distinguishable.

Figure 31 is a plot of actual space flight data obtained from the SSS-A (Explorer 45) satellite, using a raster width of 18.

X. PROGRAM LISTING OF SUBROUTINE GRAYS

```
      SUBROUTINE GRAYS(YMIN,YMAX,WIDTH,XCORD,INTS,NPTS,Z,M,N)
      DIMENSION YMIN(1),YMAX(1),XCORD(M,N),INTS(M,N),Z(1),NPTS(1),X(14)
      INTEGER*2 LINES(9,13)/
     1 0,0, 1, 0, 0,  7, -7,-11, 33,0,0, 1, 1, 1,  7, -7,+11, 33,
     2 0,0, 1, 3, 5,  9, -8,-12, 34,0,0, 1, 4, 5,  8, 13,+12, 34,
     3 0,1, 3, 4, 6, 10, -8, 21, 35,0,1, 3, 4, 6, 10, -8, 21, 35,
     4 0,2, 4, 6, 8, 12,+11, 25, 40,0,2, 4, 6, 8, 12,+11, 24, 40,
     5 0,2, 4, 6, 8, 12,+11, 24, 41,0,2, 4, 7, 9, 13,+12,+26, 43,
     6 1,3, 5, 8, 7, 16,-1',+20, 52,1,3, 5, 8, 7, 16,-14,-20, 54,
     7 1,3, 5, 8, 7, 16,-14,-21, 56/

      INTEGER*4 L(10)/10*0/
      DATA W,IN/0,0/
C.....................................................................
C...HOUSEKEEPING......................................................
C...THIS SECTION IS DONE INITIALLY AND EACH TIME THE WIDTH IS CHANGED
C
      IF(W.EQ.WIDTH) GO TO 300
      W=WIDTH
C
C..........COMPUTE WIDE IN RASTER UNITS

C
      WIDE=(Z(8)-Z(6))/(Z(4)-Z(2))*WIDTH*Z(10)
      WRITE(6,1) WIDTH,WIDE
    1 FORMAT(' GRAYS:  SUBJECT SPACE WIDTH=',F6.2,
     1  ' EQUIVALENT TO RASTER WIDTH=',F6.2)
C
C..........IF WIDE GT 21 RASTERS COMPUTE NUMBER OF BLOCKS (IPEAT) OF A
C..........SMALLER RASTER TO BE USED IN SHADING
C
      IPEAT=1

      IF(WIDE.LE.21.) GO TO 150
  140 IPEAT=IPEAT+1
      R=WIDE/IPEAT
      IF(R.GE.21.) GO TO 140
      WIDE=R
C
C..........INITIALIZE VARIABLES:
C..........FILL ARRAY L WITH THE APPROPRIATE VALUES STORED IN ARRAY
C
  150 DO 160 I=1,14

  160 X(I)=I*WIDE/15.
      WRITE(6,2) IPEAT,WIDE
    2 FORMAT('+',73X,'PLOTTED IN ',I3,' BLOCK(S) OF ',
     1  F6.2,' RASTERS EACH.')
      NRAST=WIDE+.5
```

```
          IF(NRAST.LT.9) NRAST=9
          NRAST=NRAST-8
          DO 200 I=2,10
          INE=LINES(I-1,NRAST)
      200 L(I)=IABS(INE)
C...................................................................
C..KEEP 4060 MODE ARRAY VALUES ......................................
C...THESE WILL BE CHANGED: THE 4060 MODE ARRAY WILL BE RESET UPON
C...RETURN
      300 A30=Z(30)

          A14=Z(14)
          A94=Z(94)
          A95=Z(95)
          CALL SETSMG(Z,95,1.0)
          CALL SETSMG(Z,94,1.0)
          LEV=1
C...................................................................
C...START LOOPING THROUGH ARRAYS TO BE SHADED:     ..................
C...     START WITH BAND 1 (LEV=1)
C...                   FIRST POINT IN THE BAND (I=1)

C
      400 IMAX=NPTS(LEV)
          IF(N.EQ.1) IMAX=1
          I=1
      500 IFIL=0
          XX=XCORD(LEV,I)
          NUM=MINO(INTS(LEV,I)+1,10)
          B=YMIN(LEV)
          T=YMAX(LEV)
C

C.........CHANGE COORDINATES FROM SUBJECT SPACE UNITS TO RASTER
C              UNITS

          CALL SETSMG(Z,14,0.)
          CALL SCALZZ(Z,XX,B,IX,IY)
          B=IY
          CALL SCALZZ(Z,XX,T,IX,IY)
          XX=IX
          T=IY
          CALL SETSMG(Z,14,1.)
          CALL SETSMG(Z,30,.5)

          XFIL=XX
C...................................................................
C...SELECT APPROPRIATE SHADING FOR GIVEN RASTER WIDTH AND INTEN ..
C
      550 IF(WIDE.GE.18.5)GO TO(1400,40,40,40,50,60,80,70,30,80),NUM
          IF(WIDE.GE.17.5)GO TO(1400,40,40,40,50,60,30,28,80,80),NUM
          IF(WIDE.GE.14.5)GO TO(1400,40,40,40,50,60,30,26,80,80),NUM
          IF(WIDE.GE.12.5)GO TO(1400,40,30,40,35,60,80,25,80,80),NUM
```

```
      IF(WIDE.GE.11.5)GO TO(1400,10,30,27,40,28,28,30,26,80),NUM
      IF(WIDE.GE.10.5)GO TO(1400,10,30,26,26,28,30,26,30,80),NUM
      IF(WIDE.GE.9.5) GO TO(1400,10,25,26,26,28,27,30,30,80),NUM
      IF(WIDE.GE.8.5) GO TO(1400,10,25,25,27,27,27,70,30,80),NUM
      GO TO (1400,10,25,26,28,28,28,70,70,80),NUM
C
C.......................................................................
C...SHADING IS DONE BY THE FOLLOWING CODE          ....................
C
   10 CALL SEGMTG(Z,1,XX+X(1),B,XX+X(1),T)
      XX=XX+X(8)-4.
      GO TO 30
   28 IN=IN-1
   27 IN=IN-1
   26 IN=IN-1
   25 IN=IN-1
   30 INE=(1.4*WIDE-7.0)/3.
      IF(14.5.GE.WIDE.AND.WIDE.GE.12.5) INE=INE-1
      NN=(T-B)/WIDE*(INE+IABS(IN))
      IN=0
      IF(NN.LT.2) CALL SEGMTG(Z,1,XX+X(7),B+4.,XX+X(7)+3.,B+4.)
      IF(NN.GE.2) CALL MLTPLG(Z,NN-2,XX+X(7),B+1.,XX+X(7)+3.,B+1.,
     1  XX+X(7),T-1.,XX+X(7)+3.,T-1.)
      IF(NUM.EQ.2) GO TO 1400
      IF(NUM.EQ.4.AND.WIDE.GE.12.5) GO TO 35
      IF(NUM.EQ.7.AND.WIDE.LT.10.5) GO TO 65
      IF(NUM.EQ.6.AND.WIDE.LT.10.5) GO TO 37
      IF(NUM.EQ.5.AND.WIDE.LT.10.5) GO TO 32
      IF(NUM.EQ.8 .AND. 12.5.GT.WIDE .AND. WIDE.GE.11.5) GO TO 40
      IF(NUM.GE.8) GO TO 70
      GO TO 40
   37 CALL MLTPLG(Z,0,XX+X(6),B,XX+X(6),T,XX+X(11),B,XX+X(11),T)
   32 CALL MLTPLG(Z,0,XX+X(4),B,XX+X(4),T,XX+WIDE,B,XX+WIDE,T)
      CALL MLTPLG(Z,L(NUM),XX+X(7),B,XX+X(7),T,XX+X(13),B,XX+X(13),T)
      GO TO 1400
   35 CALL SEGMTG(Z,1,XX+2.,B,XX+2.,T)
   40 XB=WIDE/(2.*L(NUM)+4.)
      XE=WIDE+XX-XB
      XB=XX+XB
      CALL MLTPLG(Z,L(NUM),XB,B,XB,T,XE,B,XE,T)
      GO TO 1400
   50 CALL MLTPLG(Z,L(NUM),XX+X(1),B,XX+X(1),T,XX+X(14),B,XX+X(14),T)
      GO TO 1400
   60 CALL SEGMTG(Z,1,XX+X(9),B,XX+X(9),T)
      IF(WIDE.GE.18.5) CALL SETSMG(Z,30,1.0)
   65 CALL MLTPLG(Z,L(NUM),XX+X(1),B,XX+X(1),T,XX+WIDE,B,XX+WIDE,T)
      GO TO 1400
```