TMX - 69905

NSSDC

74-07

A Computer Graphics Display and Data Compression Technique

May 1974

NATIONAL SPACE SCIENCE DATA CENTER
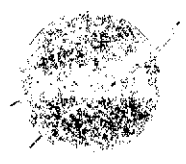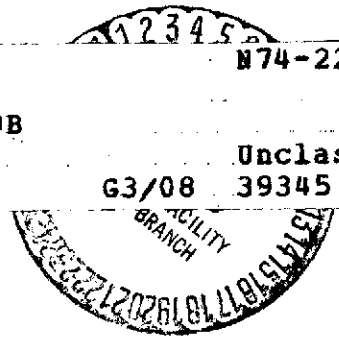
NATIONAL AERONAUTICS AND SPACE FLIGHT CENTER, GREENBELT, MD.

A Computer Graphics Display
and Data Compression Technique

By

Michael J. Teague
and
Howard G. Meyer
PMI Facilities Management Corporation

Technical Editor

Leon Levenson

May 1974

CONTENTS

LIST OF ILLUSTRATIONS

1. INTRODUCTION

The computer program discussed in this document, CRP, is intended for the graphical presentation of a general dependent variable X that is a function of two independent variables, U and V. The required input to the program is the variation of the dependent variable with one of the independent variables for various fixed values of the other. The computer program is named CRP, and the output is provided by the SD 4060 plotter. Program CRP is maintained in the GSFC program library and is suitable for operation on the IBM 360 series machines. Versions suitable for operation on UNIVAC 1108 and IBM 7094 machines may be obtained from the National Space Science Data Center.

The dependent variable $X = f(U,V)$ is displayed as the two orthogonal families, $X = f_1(U)$ for various $V = V_i$ and $X = f_2(V)$ for various $U = U_j$. The plot is a type of nomograph and is referred to as a carpet plot. An example of the output is shown in Figure 1 where the surface magnetic field declination given by the IGRF 1965 field model is shown as a function of geodetic latitude and longitude for the Northern Hemisphere. The display of a variable, $X = f(U,V)$, is a classical problem. Among the usual solutions are tabular or graphical displays of $X = f_1(U)$ for various values of V and contour plots of $U = f_3(V)$ for various values of X. These techniques suffer either from being too cumbersome to use and difficult to interpolate or else from requiring many pages of display. The carpet plot technique described in this document offers a method whereby data may be displayed in a concise fashion that is directly amenable to fast and accurate interpolation without the requirement of a calculator or slide rule. The accuracy with which interpolation may be performed is very much dependent upon what data are to be displayed and whether the dependent variable is plotted in a linear or logarithmic fashion. In many cases an interpolation accuracy of 2-3 percent may be obtained, and with care and experience even more accurate interpolation may be performed in some cases.

Program CRP is an extremely flexible program that offers the user a wide variety of options. The dependent variable may be presented in either

a linear or a logarithmic manner. Automatic centering of the plot is pro-
vided in the ordinate direction, and the abscissa is scaled automatically
for a logarithmic plot. Because of the wide variety of plots that may be
generated by CRP, flexible labeling and addressing of the orthogonal families
are included. The grid and the plot may be output on successive frames
of the SD 4060 output in order that color carpet plots may be generated.
This capability is most useful when complex plots with fine grids are gen-
erated. Program CRP has been extensively used at the National Space Science
Data Center, and samples of the applications have been presented by Teague
and Vette (1972 and 1974).

In Section 2 of this document, a description of the carpet plot tech-
nique is given; and in Section 3, the coordinate system used in the program
is presented. Section 4 discusses various aspects of the program logic
and gives detailed documentation of the data card format and NAMELIST input
parameters. A reference summary of these parameters is presented in Appendix
A. Section 5 is intended to provide the user with assistance in generating
usable carpet plots. Guidelines for making initial estimates of the impor-
tant input parameters are given, and methods for improving existing plots
are discussed. The JCL required with Program CRP is discussed in Section
6, and the program output is discussed in Section 7.

2. THE CARPET PLOT DISPLAY

As noted in the Introduction, the carpet plot display is a form of
nomograph (Adams, 1964). An arbitrary dependent variable X is presented
as a function of two independent variables U and V. In theory, an arbitrary
number of independent variables may be displayed as $X = f(U_1, U_2, U_3 \ldots U_k)$.
A practical limit of $k = 4$ exists, and even for $k>2$ the plots become physi-
cally very large. The dependent variable is plotted as a continuous function
of an independent variable at a variety of specified values of the other, i.e.,

$$X = f(U, V_i) = f_1(U) \text{ for various } V = V_i$$

The orthogonal set is plotted in the same manner, i.e.,

$$X = f(U_j, V) = f_2(V) \text{ for various } U = U_j$$

Throughout this document, the word 'family' is used to refer to the
set of curves that is $X = f_1(U)$ for various $V = V_i$ or the set that is
$X = f_2(V)$ for various $U = U_j$. A single curve of a family is referred to
as a 'member.' Note that the word 'orthogonal' is used with the strict
mathematical meaning of independence and does not mean that the two families
are plotted perpendicular to each other. Conventionally, the dependent
variable X is plotted as abscissa, and the independent variables as ordinates.
It is evident from Figure 1 that there is no fixed ordinate axis. The
origins of the ordinate axis are sliding although the ordinate scales
remain fixed. These scales are indicated by bars on the plot shown in
Figure 1. In Figure 2, an element of a general carpet plot matrix is shown.
The two members of one orthogonal family $X = f_1(U)$ for V_i and V_{i+1} and the
two members of the other family $X = f_2(V)$ for U_j and U_{j+1} define the four
matrix points (i,j), $(i+1,j)$, $(i,j+1)$, and $(i+1,j+1)$. In this report, the
convention is adopted that for an arbitrary matrix point (i,j) the first
index refers to the independent variable V being held constant at V_i and
the second index to U remaining constant at U_j. Successive points in the

3

matrix are related by the ordinate scales ΔU and ΔV and by the differences between the values of the independent variables at the successive points. In Figure 2, moving the direction of the arrow from point (i,j) to $(i,j+1)$, the vertical coordinate of the latter point $Y_{i,j+1}$ is

$$Y_{i,j+1} = Y_{i,j} + \frac{|U_{j+1}-U_j|}{\Delta U}$$

and the coordinate of the point $(i+1,j+1)$ is

$$Y_{i+1,j+1} = Y_{i,j} + \frac{|U_{j+1}-U_j|}{\Delta U} + \frac{|V_{i+1}-V_i|}{\Delta V}$$

In this fashion, the entire matrix is built up. The horizontal coordinate of each point is determined by evaluation of

$$X_{ij} = f(U_j, V_i)$$

Carpet plots are directly amenable to interpolation. In Figure 2, the value of the dependent variable is required at some arbitrary point $P \equiv (x,y)$. The intermediate matrix points (i,y), (x,j), $(i+1,y)$, and $(x,j+1)$ may be located with respect to the nearest matrix points by equations of the form

$$Y_{x,j+1} = Y_{i,j+1} + \frac{|V_x - V_i|}{\Delta V}$$

In practice, the locations of these points are determined by simple observation using the horizontal grid lines (Figure 1). Other intermediate matrix points may be determined on other members such as U_{j+2} and V_{i-1} if the bounding members for point P are very curved and the members $U = U_y$ and $V = V_i$ can be estimated to obtain the intercept P. In practice this process can be performed quickly, without the necessity for calculation, and the dependent variable $X_{x,y}$ can be found to an accuracy better than that given by linear interpolation. As noted in the Introduction, an accuracy of 2-3 percent may be expected for this interpolation, dependent upon the data being plotted.

3. THE CARPET PLOT COORDINATE SYSTEM

Program CRP displays on an SD 4060 plotter a parameter X that is a function of two independent variables, U and V, in the form of the two orthogonal families

$$X = f_1(U) \text{ for various } V = V_i$$

and

$$X = f_2(V) \text{ for various } U = U_j$$

For any given point (U,V) X must be single valued. Input to the program may be either one or both orthogonal families. In the former case, the other family is determined automatically, and this is the normal mode of operation of the program. The basic coordinate system used is the SD 4060 subjective coordinate system, X_s, Y_s. (The reader is referred to an SD 4060 manual for a complete description of the subjective coordinate system.) The maximum vertical extent of the plot defining the horizontal bounding grid lines is $Y_s = Y_{max}$ as shown in Figure 3. The value of Y_{max} is chosen arbitrarily to be 100. The horizontal subjective units for X_s are taken as those of the input dependent variable X. The origin of subject space in absolute SD 4060 rasters is the point (500,200), and the bounding grid lines in Figure 3 are defined by this point and the point (3600,2500). In the raster coordinate system, the maximum extent of the SD 4060 oscilloscope screen is defined by the point (4095,3071).

The vertical subjective coordinate Y_s is related to the real engineering units of U and V by the number of horizontal grid lines N and the vertical scale parameters ΔU and ΔV. That is, in subject space, the distance between successive horizontal grid lines is $Y_{max}/N-1$ and corresponds in real space to changes in the independent variables of ΔU and ΔV. In Figure 3, the curves $X = f_1(U)$ for $V = V_1$ and $X = f_2(V)$ for $U = U_1$ define the plot matrix point (1,1) having the minimum value of Y_s equal to Y_0. Then the subject coordinate Y_s for $X = f_1(Y_s)$ for $V = V_1$ are

$$Y_s = Y_0 + \frac{(U - U_1)}{\Delta U} \frac{Y_{max}}{N-1}$$

5

and the coordinates for the ith line $V = V_i$ are

$$Y_s = Y_o + \frac{(U - U_1)Y_{max}}{\Delta U} \frac{}{N-1} + \frac{(V_i - V_1)Y_{max}}{\Delta V} \frac{}{N-1} \qquad (1)$$

Similarly for the other orthogonal family, the jth line for $U = U_j$ is given by the expression

$$Y_s = Y_o + \frac{(V - V_1)Y_{max}}{\Delta V} \frac{}{N-1} + \frac{(U_j - U_1)Y_{max}}{\Delta U} \qquad (2)$$

For ease of interpolation, it is very desirable that the families of plot curves intersect at horizontal grid lines, such as the point (i,j) in Figure 3. If the point (1,1) is positioned p horizontal grid line increments from the line $Y_s = 0$, then

$$Y_o = \frac{pY_{max}}{N-1}$$

A general parameter ψ_{ij} is introduced where

$$\psi_{ij} = \frac{V_i}{\Delta V} + \frac{U_j}{\Delta U} \qquad (3)$$

and equation (1) becomes

$$Y_s = \frac{Y_{max}}{N-1} \left[p + \frac{U}{\Delta U} + \frac{V_i}{\Delta V} - \psi_{11} \right] = f(U) \text{ for } V = V_i \qquad (4)$$

and equation (2) becomes

$$Y_s = \frac{Y_{max}}{N-1} \left[p + \frac{U_j}{\Delta U} + \frac{V}{\Delta V} - \psi_{11} \right] f(V) \text{ for } U = U_j \qquad (5)$$

The condition that the arbitrary matrix point (i,j) lies on a horizontal grid line is, from equation (4) or (5), that

$$p + \psi_{ij} - \psi_{11} = I_{ij} \qquad (6)$$

6

where I is an arbitrary integer and p has already been given as an integer quantity. Specifically for the matrix point (n,m) having the maximum Y_s value

$$Y_s = \frac{Y_{max}}{N-1} \quad (p + \psi_{nm} - \psi_{11})$$

the condition is that

$$p + \psi_{nm} - \psi_{11} = I_{nm} \tag{7}$$

The parameter p is input to the program and the maximum and minimum matrix points (n,m) and (1,1) are positioned respectively distance Y_o below the line $Y_s = Y_{max}$ and above the line $Y_s = 0$. Then, the number of horizontal grid lines drawn for the plot is

$$N = p + I_{nm} + 1 \tag{8}$$

and the condition that any arbitrary point (i,j) lies on one of these grid lines is given by equation (6). If the matrix point (n,m) does not lie on a horizontal grid line, the left-hand side of equation (6) is truncated to obtain I_{nm}, i.e.,

$$I_{nm} = IFIX(p + \psi_{nm} - \psi_{11}) \tag{9}$$

where IFIX is the FORTRAN Fix function.

7

4. PROGRAM INPUT AND LOGIC

Program CRP consists of 12 subroutines, one BLOCK DATA subprogram and one FUNCTION subprogram. These are: MAIN, BLOCK DATA, READ, SUPER, INITIAL, GRIDPT, SCALE, LINT, REVERS, FINDF, INTERP, CRVFIT, SPLN2, and SPLIN(Y) the FUNCTION subprogram. A detailed description of the functions performed by these routines is not given, but reference will be made to them as the program input and logic are discussed. A list of the program CRP is given in Appendix B for the IBM 360 series machines.

There are many input variables for Program CRP, and their functions may be categorized as follows:

1 Carpet plot control parameters
2 Interpolation parameters
3 Grid parameters
4 Labeling and Titling parameters.

As far as possible these parameters are discussed separately, although in some cases a parameter may fit into two categories. Most of the input parameters are entered via the NAMELIST HEAD. Default values are given for each parameter, and in many cases a given parameter need not be input by the user. In addition, for cases in which several similar carpet plots are generated, the input parameters can be specified for the first plot only.

The data deck setup is shown in Figure 4. The initial card specifies the number of plots to be generated. The variable name is NPLOT, and the input format is I3. NPLOT groups of data follow this card. Each group contains a NAMELIST HEAD and a set of member data cards containing the data to be plotted.

4.1 The Input Data Cards

Under control of the NAMELIST variable RTYPE (Section 4.2.1), subroutine READ reads in one or both of the orthogonal families of curves.

9

If only one family is input, the other is determined by interpolation (Section 4.2.2). Program CRP assumes that the first family read in is the family $X = f(U)$ for various $V = V_i$. If both families are entered, the curves $X = f(V)$ for various $U = U_j$ follow. Each set (or pair of sets if both families are read in) of member data cards is preceded by a NAMELIST HEAD as indicated by Figure 4.

The deck setup for the member data cards is shown in Figure 5 and a detailed summary of the card formats is given in Table 1. From Figure 5 it is clear that each member of each orthogonal family is preceded by a header card that contains the number of data cards for the member and the value of the constant independent variable. Each orthogonal family is terminated by a blank card.

The format of the header card information is shown in Table 1. The first 67 columns are not used by Program CRP and are available for comments. Columns 68-69 contain the right justified integer that specifies the number of data cards, N_s, following for the member. Columns 75-80 contain, in F6.m format, the value of the independent variable that is constant for the members.

The format for the data cards is shown in Table 1. One of the N data cards is shown. Four pairs of independent-dependent variables are input on each card with format 4(F8.5,1X,E10.3). The last data card may contain fewer than four pairs, and if this is the case, the fields may be left blank and will be automatically rejected by Program CRP.

There are two restrictions on the data cards resulting from dimension statements. First, a maximum of 50 pairs of points per member is allowed with the result that $N_s \leq 12$. Second, a maximum of 20 and a minimum of two members per orthogonal family are allowed. Each member of an orthogonal family must be input in ascending order of the constant independent variable, i.e., in Figure 5, for the first orthogonal family, V =

V_1, V_2 ..., V_i must be in ascending order. However, the order in which
these members are plotted may be specified by the user (Section 4.2.1).
The varying independent variable on the data cards must be monotonic.

4.2 The NAMELIST Input

Forty-six input variables are entered through the NAMELIST HEAD. With
the exception of the variable IPR, which is discussed in Section 7, these
variables are discussed in this section under the four headings listed at
the beginning of Section 4 viz, (1) Carpet plot control parameters, (2)
Interpolation parameters, (3) Grid parameters, and (4) Labeling and
Titling parameters. A summary of the NAMELIST variables and their de-
fault values is given in Appendix A.

4.2.1 Carpet Plot Control Parameters

Eight variables are in the category of carpet plot control
parameters. These are RTYPE, PO, DELTAU, DELTAV, UPLOT, NUPLOT, VPLOT
and NVPLOT.

RTYPE is an integer variable that determines whether one or
both orthogonal families are input to Program CRP. For RTYPE = 0, the
family X = f_1 (U) for various V_i is input. For RTYPE = 1, this family
is followed by the family X= f_2 (V) for various U_j as shown in Figure 5.
For the latter case, the parameters UMAX, UMIN, VMAX, and VMIN (see
Section 4.2.3) must be specified. These are not necessary for RTYPE = 0.
The default value for RTYPE is 0.

The variable PO is the parameter p discussed in Section 3.
PO controls the vertical positioning of the carpet plot. PO is equal
to the number of increments of horizontal grid-line spacing that the
minimum carpet plot matrix point is positioned above the line $Y_s = 0$,
and the maximum matrix point is positioned below the line $Y_s = Y_{max}$.
For instance, in Figure 1 PO has a value of six. PO should be entered as

11

a whole number. The default value is four. As discussed in Section 3, the carpet plot matrix is generated from the minimum matrix point that is fixed on the p+1 th horizontal grid line. The maximum matrix point may not lie on a horizontal grid line, and equation (9) is used to determine the variable I_{nm}. As a result, the maximum matrix point will be positioned below $Y_s = Y_{max}$ a distance PO minus a fraction of a horizontal grid-line spacing.

The two variables DELTAU and DELTAV are the parameters ΔU and ΔV discussed in Sections 2 and 3. These are the most important parameters in the carpet plot generation and some discussion of their use is given in Section 5. DELTAU and DELTAV not only determine the vertical scale for the plot but also influence the number of horizontal grid lines that are drawn. As noted, the first orthogonal family input is the family $X = f_1(U)$ for various V_i. The scale parameter for this family, DELTAU, must be positive. A restriction on the input is that the V_i's must be monotonically increasing (Section 4.1). If DELTAV is a positive quantity, the curves for constant V_i will be plotted in the order in which they are input with the lowest value of V_i appearing at the bottom of the plot. (See Figure 1 for an example.) If DELTAV is a negative quantity, the family is plotted in the reverse order, with the highest value of V_i appearing at the bottom of the plot. This is a useful facility in generating carpet plots and is discussed further in Section 5. If both families of curves are input, DELTAU and DELTAV should both be positive. No default values are included for DELTAU and DELTAV.

The arrays UPLOT and VPLOT define the curves that are plotted on the carpet plot. NUPLOT and NVPLOT are the lengths of the arrays UPLOT and VPLOT, respectively. The maximum length is 20. The array UPLOT contains the values of U_j for which the family $X = f_2(V)$ is plotted. Similarly VPLOT defines the values of V_i for which the family

$X = f_1(U)$ is plotted. If the single family $X = f_1(U)$ for various V_i is input, the array VPLOT may be composed of any or all of the values of V_i for which $X = f_1(U)$ was input. That is, curves may be input that are not to be plotted. This is useful when the interpolated family $X = f_2(V)$ for various values of V_j has sharp changes of gradient. The input curves that are not plotted will provide grid points for the interpolation (Section 4.2.2). If both families of curves are input, UPLOT and VPLOT must be the same, respectively, as the U_j and V_i values for which $X = f_2(V)$ and $X = f_1(U)$ were input.

There is a restriction on the UPLOT and VPLOT arrays that relates to the input data. Whether the family $X = f_2(V)$ for various U_j is determined by interpolation or is read in, each line of constant UPLOT(I) must be continuous. For instance in the case of interpolation, if ten lines of $X = f_1(U)$ are input, and lines 1 through 5 and 7 through 10 satisfy UPLOT = $U_j \varepsilon U$, and line 6 does not, program termination occurs. However, if lines 3 through 8 satisfy UPLOT = $U_j \varepsilon U$, and lines 1, 2, 9, and 10 do not, the line UPLOT = U_j is continuous, and a correct plot will be obtained. If both families are input, Program CRP is not able to plot breaks in the input curves as it simply connects adjacent points in the input arrays.

Note also that if the orthogonal family $X = f_2(V)$ for various U_j is obtained by interpolation, for each value of UPLOT there must be at least two grid points given by the input family $X = f_1(U)$ for various V_i (see Section 4.2.2).

4.2.2 Interpolation Parameters

A single parameter, CHOICE, is in this category. This parameter is applicable to the case RTYPE = 0 only (Section 4.2.1), and determines which of two interpolation schemes will be used to determine the orthogonal family $X = f_2(V)$ for the array U_j = UPLOT. CHOICE may be

entered as either 'CRVFIT' or 'SPLINE'. The default value for CHOICE
is the former. CHOICE = 'CRVFIT' selects the interpolation routines
presented by Akima (1969), and CHOICE = 'SPLINE' selects the routines
given by Thompson (1970). Both interpolation schemes have the charac-
teristic that, given a set of grid points, a curve that is aesthetically
pleasing is forced through these points. Note that the schemes are not
similar to least squares fits as the resultant curve is constrained to
pass through the grid points. Two different schemes have been included
in CRP as the authors have encountered problems with spurious oscil-
lations in separate applications. However, no cases have been encountered
in which both schemes have been unsuccessful. No prior statement can be
made concerning the suitability of either scheme, and this must be de-
termined by observation.

For application on smaller computers for which limited core
capabilities exist, either a dummy subroutine may be substituted for
CRVFIT for the case CHOICE = 'SPLINE'; or for the case CHOICE = 'CRVFIT',
a dummy subroutine and function may be substituted for SPLN2 and SPLN(Y),
respectively.

For interpolated lines for which only two matrix points exist,
the above schemes are not used. In this case either linear or logarith-
mic interpolation between the matrix points is performed, depending upon
the value of ABSCA. In addition, when only three matrix points exist,
and CHOICE = 'SPLINE', a parabolic fit is used to determine the inter-
polated member without entering the spline-fit routines.

Substituting the input array V_i associated with input family
$X = f_1(U)$ into equation (5), Program CRP determines an array of the subject
coordinates for the matrix points of the orthogonal family $X = f_2(V)$
for each value contained in the array UPLOT. These arrays of matrix
points are passed to the curve-fitting interpolation routines to estimate

14

the complete orthogonal family. Each member of this orthogonal family
is restricted to an array of maximum length 50. The curve-fitting
routines estimate N_p points between the matrix points where

$$N_p = IFIX \quad \frac{50 - N_V}{N_V - 1} \tag{10}$$

where N_V is the length of the input array V_i and $N_V \geq$ NVPLOT.

4.2.3 Grid Parameters

Ten parameters are in the category of grid parameters: UMAX,
UMIN, VMAX, VMIN, KOLOR, ABSCA, XO, XF, DELTX, and SUBDIV.

The first four of these variables are required only for the
case RTYPE = 1, i.e., when both orthogonal families are input. UMAX
and UMIN are, respectively, the maximum and minimum values of U input
with either orthogonal family. VMAX and VMIN are similarly defined
for the independent variable V.

The integer parameter KOLOR allows the user to split the
generated carpet plot into two sucessive frames of SD 4060 output. For
KOLOR = 1, the carpet plot is displayed on a single frame and for
KOLOR = 2, the plot is split into two frames. The default value for
KOLOR is 1.

The purpose of this facility is to allow the user to gener-
ate a finished carpet plot in which the grid and the plot are printed
in different fashions. An example of a finished product is shown in
Figure 6 in which the grid has been printed in red and the plot and
labeling in black. The authors have also generated carpet plots in
which the grid has been printed using a dot matrix film, and the carpet
matrix has been printed using regular film. This facility is most use-
ful when complex plots are generated with many horizontal and vertical

15

grid lines, and confusion between grid lines and the members of the orthogonal families may result. For KOLOR = 2, Program CRP generates a frame of SD 4060 output containing the grid only, followed by a second frame containing the plot, labeling, titling, and scale markers. Since these two frames must be registered in the course of the enlarging and printing processes, corner registration marks are added to the two frames for KOLOR = 2. If this facility is to be used, the user is cautioned that obtaining correct registration of the two frames requires considerable care in both the enlargement and printing processes.

The variable ABSCA determines the abscissa scale for the dependent variable X. If ABSCA = 0, this scale is linear; if ABSCA = 1, it is logarithmic (base 10). For the former case, the NAMELIST variables XO, XF, DELTX, SUBDIV, and XFMT must be specified. These variables are not required for ABSCA = 1. The default value for ABSCA is 1.

The variables XO, XF, DELTX, and SUBDIV (see Section 4.2.4 for XFMT) are required only for plots with linear abscissa scales (ABSCA = 0). These parameters are determined automatically for logarithmic plots. XO and XF are, respectively, the minimum and maximum abscissa values required for the plot. Their units are those of the dependent variable X. No default values are given for these parameters. DELTX is the abscissa increment (in units of the dependent variable) at which the abscissa is to be labeled. Vertical grid lines and tick marks are also drawn at these increments. No default is given for DELTX. SUBDIV is the number of subdivisions of the increment DELTX; i.e, SUBDIV-1 vertical grid lines will be drawn in any incremental abscissa distance DELTX. The default value for SUBDIV is one, hence no vertical grid lines are drawn between labeled points. Usually the parameter DELTX will be an integer multiple of the abscissa distance XF-XO.

4.2.4 Labeling and Titling Parameters

There are 26 parameters in the category of Labeling and

Titling parameters. They are TITLE1, N1, TITLE2, N2, TITLE3, N3, XTITLE, NX, LABEL1, NLAB1, XPOS1, YPOS1, THETA1, LABEL2, NLAB2, XPOS2, YPOS2, THETA2, ULABEL, VLABEL, XTICK, YTICK, DCHAR, NCHAR, CHFMT, and XFMT.

The first six of these parameters provide a titling capability for the top of the carpet plot. The parameters TITLEi may contain arbitrary alphanumeric information of maximum length 60 characters. The parameters Ni contain the number of characters in the corresponding TITLEi array. TITLE1, 2, and 3 are, respectively, the first, second, and third lines of the title at the head of the plot. Each line of the title is automatically centered with respect to the grid. The default values for the TITLEi parameters are blanks and 1 for the N parameters.

The two parameters XTITLE and NX define the abscissa title drawn below the numeric labeling of the abscissa scale. XTITLE contains the alphanumeric title with a maximum length of 60 characters, and NX is the number of characters in XTITLE. XTITLE is automatically centered with respect to the grid. The defaults are blanks for XTITLE and 1 for NX.

The parameters LABELi, NLABi, XPOSi, YPOSi, and THETAi relate to the alphanumeric labeling of the two orthogonal families of curves. Since the position of these labels is arbitrary, either set of parameters can refer to either family. The alphanumeric title information is entered into LABELi and has a maximum length of 36 characters. NLABi are the number of characters in the LABELi arrays.

The defaults for LABELi and NLABi are, respectively, blanks and 1. The positions of the first characters of LABELi are defined by the parameters XPOSi and YPOSi. XPOSi are the abscissa coordinates measured from the left-hand grid margin and are entered in inches. The range of XPOSi is $-1.25 \leq XPOSi/(inches) \leq 8.0$. The defaults for XPOSi

17

are zeroes. YPOSi are the ordinate coordinates measured from the bottom grid margin and are entered in inches. The range of YPOSi is $-0.4 \leq$ YPOSi/(inches) ≤ 5.9. The defaults for YPOSi are zeroes. Values for XPOSi and YPOSi are given assuming that the SD 4060 hardcopy output measures 11 inches by 14 inches (grid size 8 inches by 5.9 inches). Occasionally, because of differing enlargements in the xerox process, the plot size may differ from the 11 inches by 14 inches nominal, and the values of XPOSi and YPOSi must be scaled accordingly. The variables THETAi define the orientations of the alphanumeric information contained in LABELi. THETAi are entered in degrees and define the angle measured from the horizontal in a counterclockwise direction. The ranges for THETAi are $0 \leq$ THETAi/(degrees) ≤ 90, and the default values are zeroes.

The variables ULABEL and VLABEL relate to the numeric labeling of the members of each orthogonal family. These members may be labeled at one or both ends of each curve. VLABEL relates to the first of the two families input; i.e., $X = f_1(U)$ for various V_i, and therefore determines the positions of the numeric V_i labels. For VLABEL = 0, the members of this family are labeled at the end of the curve with the greatest value of U. For VLABEL = 1, the members are labeled at the other end of the curve, viz, the end corresponding to the minimum value of U. For VLABEL = 2, both ends are labeled. The numeric labels are automatically offset from the ends of each member. The variable ULABEL is similarly defined for the family $X = f_2(V)$ for constant U_j that is either input following the first family or is determined by interpolation. However, in order to distinguish between the numeric labels for the two families, the position of these labels with respect to the end of each curve is determined by the two parameters, XTICK and YTICK, described in the next paragraph. The default values for ULABEL and VLABEL are zeroes.

The parameters XTICK and YTICK are associated with the variable ULABEL discussed in the previous paragraph. The numeric labels for the members of the family $X = f_2(V)$ for constant U_j are offset from the appropriate end of the curve (defined by ULABEL) by a tick mark the length and orientation of which are defined by XTICK and YTICK. An example of this may be seen in Figure 1. XTICK is the length of this tick mark projected in the abscissa direction and is measured in rasters. The raster coordinate system is the lowest level coordinate system available on the SD 4060. The abscissa axis is approximately 3100 rasters long, and a representative value for XTICK is 100, which number is used for the default value of XTICK. YTICK is similarly defined for the ordinate direction. The ordinate height of the grid is 2300 rasters, and a default value of 100 is included for YTICK. The numeric label for the family $X = f_2(V)$ is automatically set a small distance from the end of the tick mark. For all values of ULABEL, a positive value of XTICK results in a positive abscissa increment for the tick mark. In the ordinate direction, however, a positive YTICK results in a positive increment for ULABEL = 0 and a negative increment for ULABEL = 1. For ULABEL = 2, a positive YTICK results in a positive increment at one end and a negative increment at the other. Tick marks may be supressed by setting XTICK = YTICK = 0. If a single orthogonal family is read in, there is no option concerning which family is labeled with tick marks. However, if both families are input, the second input family is labeled with tick marks, and hence, the order of the input families may be revised to alter the labeling.

The three variables DCHAR, NCHAR, and CHFMT define the labeling of the vertical scale markers on the plot. The actual scales are defined by the two variables DELTAU and DELTAV (Section 4.2.1). The scale markers are two I-beam symbols and can be seen in the bottom left-hand corner of the carpet plot example shown in Figure 1. The horizontal positioning of these markers from the left-hand grid margin is one cycle for logarithmic abscissa and DELTX for linear abscissa. The vertical

positioning is one horizontal grid-line increment from the lowest grid margin with a single increment gap between the two I beams. DCHAR is a two-dimensional array containing alphanumeric titles for the scale markers. DCHAR(1,1) and DCHAR (2,1) contain the label for the DELTAV scale, and DCHAR(1,2) and DCHAR(2,2) contain those for the DELTAU scale. Both titles have a maximum length of eight characters. The DCHAR array defaults to blanks. NCHAR is the larger of the two numbers that are the number of characters in the DELTAU and DELTAV labels. The shorter label is filled with sufficient blanks to increase its length to that of the longer label. NCHAR defaults to 1. CHFMT is the F format for the values of DELTAU and DELTAV written with the scale markers and labels. The form of CHFMT is W.D where W is the field width and D is the number of decimal places. W is restricted to a maximum of 6, and CHFMT defaults to 4.2.

The final variable XFMT is required for linear abscissa only (ABSCA = 0) and specifies the format for the numeric labeling of the abscissa scale. The form is W.D where W is the field width and D is the number of decimal places. For positive W the format is F, and for negative W it is the E format.

5. CARPET PLOT GENERATION

Despite the large number of variables associated with Program CRP, generation of a usable carpet plot is not a difficult matter. There are three aspects of the finished plot: (1) the matrix of family members, (2) the interpolation of the input family to obtain the orthogonal family if both are not input and, (3) labeling and titling. It is strongly recommended that the first item be treated separately from the remaining two items. Further, if interpolation problems occur, these should be solved before final labeling and titling are added to the plot.

Obtaining a matrix of family members arranged in such a manner that interpolation can be performed is the most difficult part of carpet plot generation. Throughout this section the term "usable carpet plot" is used. A usable carpet plot is one on which interpolation between members can be performed satisfactorily. A necessary condition for this is that members of a given family must not cross each other and that members of opposing families must cross only once. In addition, the closer each member is inclined at 45° to the horizontal and 90° to each member of the orthogonal family, the more easily interpolation on the matrix may be performed. It should be stated that there are variables $X = f(U,V)$ that are not suitable for carpet plot display, but many of these can be adequately plotted by restricting the range of U and V on a given plot and mapping the entire field with several carpet plots. The functions that should be treated in this fashion are those for which either X exhibits several maxima and minima or $\partial X/\partial U$ or $\partial X/\partial V$ changes very rapidly in relation to the range of U and V covered.

The first computer run should be intended to test only the matrix of family members. All of the available data should be read in, but only a small number of members should be plotted. Initially, only the carpet plot control parameters (Section 4.2.1), some of the grid parameters (Section 4.2.3), and a small number of labeling and titling parameters

21

(Section 4.2.4) should be specified. If the required plot is logarithmic, only the following NAMELIST parameters should be specified: RTYPE, DELTAU, DELTAV, UPLOT, NUPLOT, VPLOT, NVPLOT, UMAX, UMIN, VMAX, VMIN, TITLE1, N1, TITLEX, NX, DCHAR, NCHAR, and CHFMT. If the plot is linear, the following additional parameters should be specified: ABSCA, XO, XF, DELTX, and SUBDIV. If only a single family is input, the parameters RTYPE, UMAX, UMIN, VMAX, and VMIN may be removed from this list. All of the other NAMELIST parameters discussed in Section 4.2 have their default values initially.

The choice of the six parameters DELTAU, DELTAV, UPLOT, NUPLOT, VPLOT, and NVPLOT is important for the initial program run. As noted in the previous paragraph, all of the available data should be input. However, the parameters UPLOT, NUPLOT, VPLOT, and NVPLOT should be used to plot only a small number (six or seven) of the salient members of the input family or families. The VPLOT array should include the first and last members of the input family $X = f_1(U)$ for various V_i. If the orthogonal family is read in, UPLOT should contain the first and last members of this family. If the orthogonal family is obtained by interpolation, UPLOT should contain values close to the minimum and maximum values of U input with the first family $X = f_1(U)$.

Initially DELTAV should be a positive quantity. DELTAU and DELTAV not only determine the relative positions of the various family members, but also influence the number of horizontal grid lines drawn. For ease of interpolation on the finished carpet plot matrix, approximately 40-50 horizontal grid lines should be drawn. The actual number drawn is output on the line printer (Section 7), and the suitability of this number may be assessed without the necessity of submitting the SD 4060 plot tape. From equations (3), (7), and (9), for the initial computer run, the number of horizontal grid lines drawn is, in FORTRAN notation,

$$N = 5 + IFIX(4 + \frac{U_m - U_1}{DELTAU} + \frac{V_n - V_1}{DELTAV}) \qquad (11)$$

where the subscripts 1 and n refer to the minimum and maximum matrix points, respectively. For RTYPE = 1, U_m, U_1, V_n, and V_1 are input via NAMELIST. For RTYPE = 0, assuming that the VPLOT and UPLOT arrays are monotonically increasing, $V_1 \equiv$ VPLOT(1)
and $V_n \equiv$ VPLOT(NVPLOT)
for positive DELTAV. For the purposes of estimating the number of horizontal grid lines drawn, it may be assumed that U_m is the greatest number in the UPLOT array that is less than or equal to the maximum U value input for the member $X = f_1(U)$ corresponding to V_i = VPLOT(NVPLOT). For example, in Figure 6, VPLOT(NVPLOT) = 18000 n.m., and the maximum UPLOT value contained in the Energy array for V_i = 18000 n.m. is E = 4 MeV, and thus $U_m = 4.0$. In many cases, however, such as that shown in Figure 1, U_m = UPLOT(NUPLOT), as each input member contains values for the entire range of U. The variable U_1 may be taken to be the smallest number in the UPLOT array that is greater than or equal to the minimum U value input for the $X = f_1(U)$ corresponding to V_i = VPLOT(1). U_1 will usually reduce to U_1 = UPLOT(1) as in Figures 1 and 6. U_1 and U_m should be obtained using these definitions by inspection of the input data deck.

Equation (11) is clearly not sufficient to estimate initial DELTAU and DELTAV values even if the points (1,1) and (n,m) (Figure 3) and the required number of horizontal grid lines are known before the initial plot is made. However, first estimates of DELTAU and DELTAV may be obtained by assuming that

$$\frac{U_m - U_1}{\text{DELTAU}} = \frac{V_n - V_1}{\text{DELTAV}} \qquad (12)$$

Assuming that approximately 45 horizontal grid lines are required, equations (11) and (12) become

$$\text{DELTAU} \simeq | U_m - U_1 | / 18 \qquad (13)$$

and $$\text{DELTAV} \simeq | \text{VPLOT(NVPLOT)} - \text{VPLOT(1)} | / 18 \qquad (14)$$

Applying equations (13) and (14) to the data plotted in Figure 6, initial estimates of DELTAU $\equiv \Delta E$ = 0.222 MeV and DELTAV $\equiv \Delta ALT$ = 777.77 n.m. are obtained. It should be reiterated that these estimates are obtained before any plots are made. Equations (13) and (14) fix the maximum and minimum matrix points on horizontal grid lines. The conditions that any other matrix point (i,j) will lie on a horizontal grid line, which is desirable for ease of interpolation, are that U_j - U_1 and V_i - V_1 must be integer multiples of DELTAU and DELTAV, respectively. It is clear that this is not likely to be the case for ΔE = 0.222 MeV or ΔALT = 777.77 n.m., and in general the initial estimates of DELTAU and DELTAV should be 'rounded' to more reasonable values. Therefore, in the above case, ΔE is rounded to 0.25 MeV and ΔALT to 750 n.m., and the results of the initial computer run are shown in Figure 7. It can be seen that sufficient labeling has been added to easily identify the family members and scale parameters. A second example is shown in Figure 8. In this case, initial estimates of the scale parameters were obtained from equations (13) and (14) as DELTAU $\equiv \Delta B$ = 0.01333 and DELTAV $\equiv \Delta L$ = 0.0666. These were rounded to ΔB = 0.015 and ΔL = 0.06. Note that in the case of the data shown in Figure 8, it was assumed that U_1 = UPLOT(1) = 0.04. As can be seen from Figure 8, the minimum matrix point does not, in fact, occur for $U \equiv B$ = 0.04. Nevertheless, a usable carpet matrix resulted from the use of equations (13) and (14), and this is the initial objective.

From the two examples shown in Figures 7 and 8, it is evident that usable carpet plots can be obtained for the input data. In some instances this will not be the case, and the various family members may cross over each other in some regions of the matrix. This does not necessarily mean that the data cannot be carpet plotted. For instance, the data plotted in Figure 8 are replotted in Figure 9 with different values of the scale parameters, and many of the members cross over each other. Attempts at carpet plotting should not be abandoned merely

because members cross initially. Two courses of action are possible. First, the data may be split into subsets each of which may be successfully plotted on separate frames. Second, the scale parameters may be varied in order to generate a usable matrix (as in Figures 8 and 9).

Varying the scale parameters should be performed with care since they affect not only the ordinate distance for any member but also the number of horizontal grid lines drawn. The change in the number of horizontal grid lines drawn is, from equations (11) and (13),

$$\delta N \simeq -18 \left[\frac{\delta(DELTAU)}{DELTAU} + \frac{\delta(DELTAV)}{DELTAV} \right] \qquad (15)$$

where the δ symbol indicates small changes in the independent variable following. Further, the change in the vertical extent of a family member expressed as the number of horizontal line increments that it covers, N_1, is

$$\frac{\delta N_1}{N_1} \simeq -\frac{\delta(DELTAU)}{DELTAU} \text{ or } -\frac{\delta(DELTAV)}{DELTAV} \qquad (16)$$

for the families $X = f_1(U)$ and $X = f_2(V)$, respectively. Expressed in terms of the SD4060 coordinate system, Y_s, the change in the vertical extent of a given member is

$$\frac{\delta Y_s}{Y_s} \simeq -\left[\frac{\delta(DELTAU)}{DELTAU} + \frac{\delta N}{N-1} \right] \text{ or } -\left[\frac{\delta(DELTAV)}{DELTAV} + \frac{\delta N}{N-1} \right] \qquad (17)$$

where δN is given by equation (15). For positive DELTAU and DELTAV, equation (15) indicates that an increase in either scale parameter results in a reduction of the number of horizontal grid lines drawn. Equation (16) shows that the vertical extent of a given line in terms of grid-line increments decreases for a positive change in the scale parameters. However, equation (17) for the subject coordinate distance

(which is the important variable in the context of modifying the carpet matrix) is ambiguous as far as the sign of δY_s is concerned, since the two terms have opposing signs. In general, however, the first term is dominant, and for a positive change in a scale parameter, δY_s is negative. For instance, if DELTAU is increased by 10 percent (reducing the number of grid lines drawn by one or two depending upon rounding), the vertical extent of a given member is reduced by 10 percent in terms of grid-line increments but only 5 percent in terms of subject coordinates.

In Figure 9, the input family is that labeled 'L Values', i.e., $X = f(B)$ for various L. The orthogonal family, labeled 'B Values', is obtained by interpolation. The members of the input family cross over each other primarily because the vertical extent of the higher L-value members is too large. From equations (15) and (17), assuming that approximately 45 horizontal grid lines are drawn, $\delta Y_s / Y_s \simeq -0.5$ $\delta(\text{DELTAU})/(\text{DELTAU})$. Thus, in Figure 9, in order to reduce the vertical extent of the L-value members by half, the value of DELTAU ($= \Delta B$) should be approximately doubled. It should be noted that equations (15) through (17) are only approximate, are applicable for small changes in the scale parameters, and assume that the maximum and minimum grid points do not change. The above expression for $\delta Y_s / Y_s$ is therefore approximate and should be used to determine trends only. From equation (15), if DELTAU is doubled and DELTAV is held constant, approximately 18 fewer horizontal grid lines will be drawn. As a result, DELTAV should be reduced to compensate for this. On the bases of the carpet matrix shown in Figure 9 and the trends indicated by equations (15) and (17), new values of the scale parameters may be estimated as DELTAU = ΔB = 0.02 and DELTAV = ΔL = 0.05. The carpet matrix for these scale parameters is shown in Figure 10, and it can be seen that this matrix is acceptable.

Having obtained a usable matrix, a plot should be made that contains all the required members in the matrix. At this point, any interpolation

problems should be solved. The two available interpolation schemes, SPLINE and CRVFIT, are contrasted in Figures 11 through 13. In each of these figures the L-value family is input. In Figures 11 and 12 the only members input are those that are plotted. It may be seen that the interpolated family obtained with the CRVFIT routine is acceptable whereas that obtained with the SPLINE routine has several unacceptable variations in the members, particularly for B = 0.04. In general, this type of oscillation is likely to occur when a large change of slope occurs over a small region of the varying independent variable. If interpolation problems are not solved simply by changing the routine used, additional input members must be added in order to damp the oscillation. For instance, in Figure 12 an input curve for L = 3.2 may be added. It is not necessary to plot this input curve, and thus values of the dependent variable may be input only for those values of the independent variable that are contained in the UPLOT array. That is, in Figure 12, the L = 3.2 member requires X values for B = 0.05, 0.1, 0.2, 0.4, and 0.5. The values of X may be estimated from the existing plot (Figure 12). The results of this procedure are shown in Figure 13, in which it can be seen that the oscillation has been damped. The results, however, are still clearly inferior to those obtained with CRVFIT routine, and in this case this routine should be used.

The final phase of the carpet plot generation, the addition of titling and final labeling, is a straightforward application of the NAMELIST parameters given in Section 4.2.4. As an example, the final version of the test plot shown in Figure 7 may be seen in Figure 6. Additional family members have been plotted, and the member numeric labels for the input family have been moved from one end to the other to avoid overwriting some members. In addition, comprehensive titling has been added.

6. JCL SETUP

Program CRP is available from the GSFC Program Library in source (FORTRAN IV) and object deck (FORTRAN IV H, OPT = 2 compilation) form suitable for operation on the IBM 360 series machines. Versions of the program for operation on IBM 7094 and UNIVAC 1108 machines are also available from the National Space Science Data Center. Because of core limitations with this computer, the IBM 7094 version is over-layed but offers the same capabilities as the IBM 360 version.

On the 360/75 machine, compilation of all subroutines requires less than 1.0 minute cpu time and 1.5 minutes I/O time. Execution time to generate 10 carpet plots with single family input is approximately 1.0 minute cpu time and 1.0 minute I/O time. Less execution time is required if both families are input. The program requires less than 170K for the EXEC LINKGO step.

The JCL required for Program CRP is relatively straightforward and is shown in Figures 14 and 15 for source and object decks, respectively. The output tape for the SD 4060 plotter is written in 7-track binary at 556 bpi and may either be user supplied or TEMPSV. The data deck setup has been discussed in Section 4 and is shown in Figures 4 and 5.

7. PROGRAM OUTPUT AND ERROR MESSAGES

7.1 Program Output

A binary, 7-track, 556 bpi SD 4060 plot tape is output by Program CRP. The number of frames on this tape is output by the SD 4060 package on the line printer. For the case KOLOR = 1, this number should be 2 plus the number of plots input on the first data card. For KOLOR = 2, the number should be 2 plus twice the number of plots. For ease of publication, it is suggested that 35-mm positive film output be requested from the SD 4060.

Line printer output is provided to enable the user to assess the success of the program run before submittal of the SD 4060 plot tape. The plotted members of each orthogonal family are output, and a sample list is shown in Figure 16. The first family output corresponds to the first family read in to the program ($X = f(U)$ for various V_i). The second family output corresponds either to the second input family or to the interpolated family, depending upon RTYPE. Note that only the family members to be plotted are output on the line printer. The values of the constant independent variable associated with each member input are listed before the plotted members (values of 999.0 in this list should be disregarded as these are dummy values). In Figure 16, the alphanumeric header for each member and the alphanumeric header for the columns of the varying independent variable are taken from the DCHAR array input via the NAMELIST. Pairs of dependent and independent variables are given on each line-printer line, together with the SD 4060 subject coordinate of each point (labeled Y).

Three checks may be performed on these tables before the plot tape is submitted. First, the constant independent variable heading each member may be compared to the UPLOT and VPLOT arrays to ensure that the intended lines are plotted. Second, the values of Y may be scanned to

31

make sure that all values of Y are in the range 0 to 100 (except the 999.0 default values), and that the variation of Y for a given member is significant. Values of Y outside the range 0 to 100 may result from 'overhang' of the input curves. That is, the plot is automatically scaled in a vertical direction by using the maximum and minimum grid points. The input curves may cover a wider range of independent variables than those encompassed by these grid points, with the result that values of $Y_s > Y(n,m)$ and $Y_s < Y(1,1)$ (see Figure 3) may result. In extreme cases, $Y_s > Y_{max}$ or $Y < 0$. This problem may be rectified by increasing the value of PO to allow more space between the bounding horizontal grid lines and the bounding grid points in the carpet matrix. If the variation of Y for a given member is small (< 10), the result will be an almost horizontal line in the plot, making interpolation on the matrix an inaccurate process. This may be rectified by making the appropriate vertical scale more sensitive, that is, by reducing either DELTAU or DELTAV. This should be done with caution, however, since all of the members of each family will be affected by this change, and the overall effect may not be to improve the plot. The third check that may be performed is to examine grid points in both family lists. The subject coordinate Y should be identical for a given grid point in both lists.

In addition to the plotted families, the number of horizontal grid lines is output. As noted in Section 5, this should be approximately 45 to facilitate interpolation on the carpet matrix. This number may be changed by varying DELTAU and DELTAV as discussed in Section 5. For the case of linear abscissae (ABSCA = 0.), the number of vertical grid lines is output.

Under control of the NAMELIST variable IPR (Appendix A), the entire NAMELIST may be written on the line printer. For NUPLOT or NVPLOT < 20, the UPLOT or VPLOT arrays are filled with dummy values of 999.0, and

these should be disregarded. Further, when the default values for the alphanumeric NAMELIST variables are used, numeric representations of these defaults are output on the line printer. These, too, should be disregarded.

7.2 Error Messages

A number of SD 4060 error codes may be generated by misuse of Program CRP. These are output by the IGS routines associated with the SD 4060 plot package. The reader is referred to the SD 4060 manual for a complete list of these. However, the two most likely SD 4060 error codes are 32 and 33 and refer, respectively, to bad abscissae and bad ordinate values. Bad abscissa coordinates are not likely to occur for a logarithmic carpet plot for which the abscissa scaling is performed automatically. However, if XO and XF are specified incorrectly for a linear carpet plot code 32, errors will result. Code 33 errors may result from the 'overhang' of the input curves that has been discussed in Section 7.1. These may be corrected by increasing PO or by extending the range of the parameter UPLOT.

A number of error codes are output by Program CRP and these are listed in Appendix C. Section C.1 shows four error messages associated with NAMELIST input errors, the last of which is a warning statement only. Program termination is associated with the first three error messages. The first message indicates that an incorrect value of RTYPE or ABSCA has been entered, and the second indicates that either DELTAU or DELTAV has been entered with the unacceptable value of zero. The third message indicates a problem with one or more of the five listed variables that are required for a linear carpet plot only. This error message will result either if these variables have not been specified or if XF $\leq$ XO. The fourth message is a warning indicating that any one of the three arrays (TITLE1, 2, and 3) has not been specified.

The error messages associated with the interpolation procedure for RTYPE = 0 are shown in Section C.2. The two return codes associated with the first error message indicate problems with the input data. The return code of 1 generally indicates that an input curve is not monotonic with respect to the varying independent variable, e.g., U of $X = f(U)$ for various $V = V_i$. The parameter INT is the number of increments of the interpolated curve between grid points and is $N_p + 1$ in equation (10). The maximum value of INT is M = 48 since the maximum length of the interpolated array is 50, and a minimum of two curves must be input. The return code of 2 generally indicates either that VPLOT is not monotonic or that VPLOT contains a value that was not input. IGP is the array of grid points stored for each value of UPLOT. Error message numbers two and three result if the UPLOT array contains one or more values that are not included in any of the members of the family $X = f(U)$ for various $V = V_i$. This will not normally terminate the run (unless the entire UPLOT array is incorrect) as the program will automatically eliminate these values. The corrected UPLOT array and its length are given by error message number three.

APPENDIX A. SUMMARY OF NAMELIST INPUT PARAMETERS

With the exception of the first parameter, IPR, the NAMELIST input parameters are defined in order in the four categories of (1) Carpet Plot control parameters, (2) Interpolation parameters, (3) Grid parameters, and (4) Labeling and Titling parameters. Each parameter is numbered. An alphabetical list of the parameters is given in Table 2 and this may be used as a convenient cross reference for the list given in this Appendix. Section references are given for a more detailed description of a given parameter.

No.	Name	Type	Default	Section	Description
1	IPR	I*4	None		Print Option Flag. 0 for NAMELIST variables not printed and 1 for NAMELIST variables printed.
2	RTYPE	I*4	0	4.2.1	Input Data Flag. 0 for a single-family input and 1 for both families input
3	PO	R*4	4.	4.2.1	Determines vertical positioning of carpet matrix. Number of horizontal grid increments below minimum grid point and above maximum grid point. Should be whole number.
4	DELTAU	R*4	0.	4.2.1	Increment in U (the independent variable not held constant for the members of the first family) corresponding to one increment between successive horizontal grid lines
5	DELTAV	R*4	0.	4.2.1	Increment in V corresponding to one increment between successive horizontal grid lines

No.	Name	Type	Default	Section	Description
6	VPLOT	R*4	None	4.2.1	The array of Vs for which $X = f(u)$ are plotted. VPLOT $\varepsilon\ V_i$, where V_i are the values of the independent variable held constant for the members of the first family input.
7	NVPLOT	I*4	0	4.2.1	Length of the VPLOT array. Maximum 20
8	UPLOT	R*4	None	4.2.1	The array of Vs for which $x = f(V)$ are plotted. UPLOT $\varepsilon\ U_i$ for RTYPE = 1.
9	NUPLOT	I*4	0	4.2.1	Length of the UPLOT array. Maximum 20
10	CHOICE	R*8	'CRVFIT'	4.2.2	Interpolation Method Flag. Applicable for RTYPE = 0 only. 'CRVFIT' for subroutine CRVFIT and 'SPLINE' for subroutine SPLN2 and function SPLIN(Y)
11	UMAX	R*4	0	4.2.3	Maximum grid U value for input data. Omit for RTYPE = 0.
12	UMIN	R*4	0	4.2.3	Minimum grid U value for input data. Omit for RTYPE = 0.
13	VMAX	R*4	0	4.2.3	Maximum grid V value for input data. Omit for RTYPE = 0.
14	VMIN	R*4	0	4.2.3	Minimum grid V value for input data. Omit for RTYPE = 0.
15	KOLOR	I*4	1	4.2.3	Flag for number of SD4060 frames output. 1 for single-frame output. 2 for double-frame output
16	ABSCA	R*4	1	4.2.3	Flag for type of abscissa scale. 0 for linear and 1 for logarithmic scale

No.	Name	Type	Default	Section	Description
17	XO	R*4	None	4.2.3	Minimum abscissa value in input data units. Omit for ABSCA = 1.
18	XF	R*4	None	4.2.3	Maximum abscissa value in input data units. Omit for ABSCA = 1.
19	DELTX	R*4	None	4.2.3	Increment in input data units for which vertical grid lines, tick marks, and labels are given. Omit for ABSCA = 1.
20	SUBDIV	R*4	1	4.2.3	Number of increments into which DELTX is subdivided by drawing vertical grid lines. Omit ABSCA = 1.
21	TITLE1	R*4	Blanks	4.2.4	First line of title in page header
22	N1	I*4	1	4.2.4	Number of characters in TITLE1. Maximum 60
23	TITLE2	R*4	Blanks	4.2.4	Second line of title in page header.
24	N2	I*4	1	4.2.4	Number of characters in TITLE2. Maximum 60
25	TITLE3	R*4	Blanks	4.2.4	Third line of title in page header
26	N3	I*4	1	4.2.4	Number of characters in TITLE3. Maximum 60
27	XTITLE	R*4	Blanks	4.2.4	Abscissa title. Appears below abscissa scale.
28	NX	I*4	1	4.2.4	Number of characters in XTITLE. Maximum 60
29	LABEL1	R*4	Blanks	4.2.4	Alphanumeric label for one family of curves
30	NLAB1	I*4	1	4.2.4	Number of characters in LABEL1. Maximum 36
31	XPOS1	R*4	0.	4.2.4	Horizontal position of first character of LABEL1 in inches. Measured from left-hand bounding grid line. $-1.25 \leq XPOS1 \leq 8.0$.

No.	Name	Type	Default	Section	Description
32	YPOS1	R*4	0.	4.2.4	Vertical position of first character of LABEL1 in inches. Measured from lower bounding grid line. $-0.4 \leq$ YPOS1 ≤ 5.9.
33	THETA1	R*4	0.	4.2.4	Orientation of LABEL1 in degrees. Measured counterclockwise from the horizontal. $0. \leq$ THETA1 ≤ 90.
34	LABEL2	R*4	Blanks	4.2.4	Alphanumeric label for other family of curves
35	NLAB2	I*4	1	4.2.4	Number of characters is LABEL2. Maximum 36
36	XPOS2	R*4	0.	4.2.4	As XPOS1 bar for LABEL2
37	YPOS2	R*4	0.	4.2.4	As YPOS1 bar for LABEL2
38	THETA2	R*4	0.	4.2.4	As THETA1 bar for LABEL2
39	ULABEL	I*4	0	4.2.4	Flag for numerical labels for lines of constant U_j. 0 for end of line with maximum V value. 1 for end of line with minimum V value. 2 for both
40	VLABEL	I*4	0	4.2.4	Flag for numerical labels for lines of constant V_i. 0 for end of line with maximum V value. 1 for end of line with minimum U value. 2 for both
41	XTICK	R*4	100.	4.2.4	Horizontal distance in rasters for tick mark used with numeric labels for the family $X = f(V)$ for various U_j. Horizontal distance positive for positive XTICK.

No.	Name	Type	Default	Section	Description
42	YTICK	R*4	100.	4.2.4	Vertical distance in rasters for tick mark used with numeric labels for the family X = f(V) for various U_j. Vertical distance positive for positive YTICK and ULABEL = 0. Vertical distance negative for positive YTICK and ULABEL = 1.
43	DCHAR	R*4	Blanks	4.2.4	Array containing alphanumeric labels for the scale markers. DCHAR(1,1) and DCHAR(2,1) contain label for DELTAV. DCHAR(1,2) and DCHAR(2,2) contain label for DELTAU. Both labels have a maximum length of eight characters.
44	NCHAR	I*4	1	4.2.4	The larger of the two numbers of characters in the DELTAV and DELTAU labels
45	CHFMT	R*4	4.2	4.2.4	Print format for numeric values of DELTAU and DELTAV. The form is W.D, where W is the field width (W $\leq$ 6) and D is the number of decimal places.
46	XFMT	R*4	2.0	4.2.4	Print format for numeric labeling of abscissa scale. The form is W.D, where W is the field width and D is the number of decimal places. For positive W the format is F, and for negative W the format is E.

```
DIMENSION Y1(50,20),Y2(50,20)
DIMENSION XP(50),YP(50),XV(50),YV(50)
DIMENSION NPTS(20),LGP(20)
DIMENSION XBAR(6),YBAR(6)
DIMENSION XHL(20),YHL(20),XVL(20),YVL(20),PVX(2),PVY(2)
DIMENSION XVLB(20),YVLB(20),XHR(20),YHR(20)
DIMENSION CHAR4(2,2)
REAL      L,LP
REAL*8    CHOICE
REAL*8    CHAR1, CHAR2
LOGICAL*1   EQUAL,BLANK,LAB1,LAB2
COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20),T(20),
1           L(50,20),LP(20)
INTEGER RTYP,ELABEL
COMMON /INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
COMMON /SCAL / CHOICE,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN,
1BMIN,YY,KOLOR,NLP,NBP
COMMON /TITL/ N1,TITLE1(15),N2,TITLE2(15),N3,TITLE3(15),NX,
1            XTITLE(15),NHT,HORIZ(9),NVT,VERT(9),XHZ,YHZ,THETA1,
2            XVT,YVT,THETA2,DCHAR(2,2),NCHAR,CHFMT,BLABEL,LLABEL,
3            DELTA,DELTA1,SUBDIV
COMMON /PLOT/ A(200)
COMMON /PSI/  PSI1(50,20),PSI2(50,20)
COMMON /LAB8/ LAB1(8),LAB2(8)
EQUIVALENCE (PSI2(1,1),Y2(1,1)),(PSI1(1,1),Y1(1,1))
EQUIVALENCE (CHAR1,LAB1(1)),(CHAR2,LAB2(1))
EQUIVALENCE (CHAR4(1,1),CHAR1),(CHAR4(1,2),CHAR2)
DATA EQUAL/1H=/,BLANK/1H /
INDEX(I) = (I-1)/IINT + 1
NCURVE=20
IDIM = 50
READ (5,1007) KARP
CALL MODESG(A,0)

START NEW PLOT

1 CALL INITAL
  CALL READ
  IINT = IDIM/NL
  IF (RTYP.LT.2) GO TO 10
  NHL = NLP
  NVL = NBP
  GO TO 40
10 IF (RTYP.EQ.1) GO TO 20
  NIP = NBP
  NFP = NLP
  CALL FINDF(BP,B,F1,F2,T)
  CALL LINT(L,T,F2)
  CALL GRIDPT(LP,L,IGP)
  NHL = NFP
  NVL = NIP
  GO TO 40
20 NIP=NLP
  NFP=NBP
  CALL FINDF(LP,L,F2,F1,T)
  CALL LINT(B,T,F1)
  CALL GRIDPT(BP,B,IGP)
  NHL = NIP
  NVL = NFP
40 K = 1
  DO 100 I=1,NHL
  CONST = LP(I)/DELL
```

41

```
      IF (RTYP.GT.0) GO TO 45
      N=INDEX(IGP(I))
      K = N
      GO TO 50
   45 N = I
      IF (RTYP .EQ.2) K=N
   50 DO 100 J=1,IDIM
      P = B(J,K)
      IF (P.EQ.999..OR.F1(J,N).EQ.999.) GO TO 100
      PSI1(J,N) = CONST + P/DELB
  100 CONTINUE
      K = 1
      DO 105 I=1,NVL
      CONST = BP(I)/DELB
      IF (RTYP.NE.1) N=I
      IF (RTYP.EQ.1) N=INDEX(IGP(I))
      IF (RTYP.EQ.1) K=N
      IF (RTYP.EQ.2) K=I
      DO 105 J=1,IDIM
      P = L(J,K)
      IF (P.EQ.999..OR.F2(J,N).EQ.999.) GO TO 105
      PSI2(J,N) = CONST + P/DELL
  105 CONTINUE
      IF (RTYP.LT.2) GO TO 110
      PMX = RLMAX/DELL  + BMAX/DELB
      PMN = RLMIN/DELL + BMIN/DELB
      RNUM= INT(2.*PO + PMX - PMN)
      PHI = PO - RLMIN/DELL - BMIN/DELB
      GO TO 115
  110 IF (RTYP.EQ.0) CALL SCALE(PSI2    ,RNUM,PHI)
      IF (RTYP.EQ.1) CALL SCALE(PSI1,    RNUM,PHI)
  115 CONTINUE
      IRNUM=RNUM+1.
      WRITE(6,1017) IRNUM
      IF (ABSCA .EQ. 1.) GO TO 116
      NOLINE=(((XF-XO)/DELTX)*SUBDIV+.0001)+1
      WRITE(6,1003) NOLINE
  116 CONTINUE
      DIVN = YY/RNUM
      DO 120 IL=1,NHL
      IF (RTYP.GT.0) LN= IL
      IF (RTYP.EQ.0) LN=INDEX(IGP(IL))
      DO 120 IB=1,IDIM
      P1 = PSI1(IB,LN)
      IF (P1.NE.999.) Y1(IB,LN) = DIVN*(PHI+P1)
  120 CONTINUE
      DO 135 IB=1,NVL
      IF (RTYP.EQ.1) LN=INDEX(IGP(IB))
      IF (RTYP.NE.1) LN = IB
      DO 135 IL=1,IDIM
      P2 = PSI2(IL,LN)
      IF (P2.NE.999.) Y2(IL,LN) = DIVN*(PHI+P2)
  135 CONTINUE
      IF (RTYP.EQ.2) GO TO 129
      IERR = 1
      IERR = 0
      I=1
      INVRT1=0
      INVRT2=0
      IF (RTYP .EQ. 1) GO TO 1370
 1360 IF (Y2(I,1) .NE. 999.) GO TO 1361
      I=I+1
      GO TO 1360
 1361 IF (Y2(I+1,1) .GT. Y2(I,1)) GO TO 1380
```

```
      CALL REVERS(F2,Y2,LGP)
      INVRT1=1
      DO 1362 J=1,20
         IGP(J)=LGP(J)
1362     CONTINUE
      GO TO 1380
1370 IF (Y1(I,1) .NE. 999.) GO TO 1371
      I=I+1
      GO TO 1370
1371 IF (Y1(I+1,1) .GT. Y1(I,1)) GO TO 1380
      CALL REVERS(F1,Y1,LGP)
      INVRT2=1
      DO 1372 J=1,20
         IGP(J)=LGP(J)
1372     CONTINUE
1380 CONTINUE
      IF (RTYP.EQ.0) CALL INTERP(Y2,F2,IGP,IINT,50,20,IERR)
      IF (RTYP .EQ. 1) CALL INTERP(Y1,F1,IGP,IINT,50,20,IERR)
      IF (INVRT1 .EQ. 0) GO TO 1390
      CALL REVERS(F2,Y2,LGP)
      DO 1385 J=1,20
         IGP(J)=LGP(J)
1385 CONTINUE
      GO TO 1399
1390 IF (INVRT2 .EQ. 0) GO TO 1399
      CALL REVERS(F1,Y1,LGP)
      DO 1395 J=1,20
         IGP(J)=LGP(J)
1395     CONTINUE
1399 CONTINUE
137 IF (IERR.EQ.0) GO TO 129
      WRITE (6,2001) IERR
129 CHAR4(1,1)=DCHAR(1,1)
      CHAR4(2,1)=DCHAR(2,1)
      CHAR4(1,2)=DCHAR(1,2)
      CHAR4(2,2)=DCHAR(2,2)
      DO 130 I=1,8
      IF (LAB1(I).NE.EQUAL) GO TO 130
      LAB1(I) = BLANK
      GO TO 132
130 CONTINUE
132 DO 133 I=1,8
      IF (LAB2(I).NE.EQUAL) GO TO 133
      LAB2(I) = BLANK
      GO TO 140
133 CONTINUE
140 WRITE(6,1000) CHAR1
      IN = 1
      DO 150 I=1,NHL
      WRITE(6,1001) CHAR1,LP(I),CHAR2,CHAR2
      IF (RTYP.EQ.0) IND=INDEX(IGP(I))
      IF (RTYP.GT.0) IND = I
      IF (RTYP.NE.1) IN = IND
      DO 150 J=1,IDIM,2
      K = J+1
      IF (F1(J,IND).NE.999..OR.F1(K,IND).NE.999.)
     *WRITE(6,1002) (F1(N,IND),Y1(N,IND),B(N,IN ),N=J,K)
150 CONTINUE
      WRITE(6,1000) CHAR2
      IN = 1
      DO 200 I=1,NVL
      WRITE(6,1001) CHAR2,HP(I),CHAR1,CHAR1
      IF (RTYP.NE.1) IND = I
      IF (RTYP.EQ.1) IND=INDEX(IGP(I))
      IF (RTYP.NE.0) IN = IND
```

```
      DO 200 J=1,IDIM,2
      K = J+1
      IF (F2(J,IND).NE.999..OR.F2(K,IND).NE.999.)
     *WRITE(6,1002) (F2(N,IND),Y2(N,IND),L(N,IN),N=J,K)
200 CONTINUE

**   BEGIN PLOTTING

      IPASS = 0
220 CALL RSETMG(A)
      IPASS = IPASS + 1
      IF (IPASS.EQ.2) CALL PAGEG(A,0,1,1)
      CALL SETSMG(A,19,4095.)
      CALL SETSMG(A,20,3071.)
      RMIN = 500.
      RMAX = 3600.
      YMIN = 200.
      YMAX =2500.
      XINCH=8.0625
      YINCH=5.90
      XRPERI=(RMAX-RMIN)/XINCH
      YRPERI=(YMAX-YMIN)/YINCH
      XVT1=XVT*XRPERI+RMIN
      XHZ1=XHZ*XRPERI+RMIN
      YVT1=YVT*YRPERI+YMIN
      YHZ1=YHZ*YRPERI+YMIN
      IF (KOLOR.EQ.2) CALL SUPER(RMIN,RMAX,YMIN,YMAX)
      CALL OBJCTG(A,RMIN,YMIN,RMAX,YMAX)
      IF (IPASS.EQ.2) GO TO 300
      XMAX = F1(1,1)
      XMIN = XMAX
      DO 250 I=1,NHL
      IF (RTYP.EQ.0) IND=INDEX(IGP(I))
      IF (RTYP.GT.0) IND = I
      NPTS(I) = 0
      DO  240 J=1,IDIM
      X = F1(J,IND)
      IF (X.EQ.999.) GO TO 240
      IF (X.LT.XMIN) XMIN = X
      IF (X.GT.XMAX) XMAX = X
      NPTS(I) = NPTS(I) + 1
240 CONTINUE
250 CONTINUE
      DO 260 J=1,NCURVE
          DO 255 I=1,IDIM
              IF (F2(I,J) .EQ. 999.) GO TO 255
              IF (F2(I,J) .GT. XMAX) XMAX=F2(I,J)
              IF (F2(I,J) .LT. XMIN) XMIN=F2(I,J)
255           CONTINUE
260       CONTINUE
300 IF (ABSCA.EQ.1.) GO TO 400
      CALL SUBJEG(A,X0,0.,XF,YY)
      CALL SETSMG(A,23,0.)
      IF (IPASS .NE. 1) GO TO 350
      DIVSN=DELTX/SUEDIV
      CALL GRIDG(A,DIVSN,DIVN,0,0)
350 IF (KOLOR.EQ.2.AND.IPASS.EQ.1) GO TO 220
      XSIZE=-30.
      XPLACE=-1.0
      CALL SETSMG(A,102,XSIZE)
      CALL LABELG(A,0,DELTX,0,XFMT)
      CFAC = (RMAX-RMIN)/(XF-X0)
      GO TO 455
400 CONTINUE
```

```
      IF (IPASS.EQ.2) GO TO 430
      IF (XMIN.LT.1.) XMIN=1.
      I1=ALOG10(XMIN)
      I2=ALOG10(XMAX)+1
      XO =10.**I1
      XF =10.**I2
      CFAC = (RMAX-RMIN)/(ALOG10(XF)-ALOG10(XO))
430   CALL SUBJEG(A,XO,0.,XF,YY)
      CALL SETSMG(A,23,1.)
      CALL SETSMG(A,100,3.)
      D = 2.
      C = DIVN
      IF (IPASS.EQ.1) CALL GRIDG(A,D,C,1,0)
      IF (KOLOR.EQ.2.AND.IPASS.EQ.1) GO TO 220
      YLAB =-YY/60.
      NC2=2
      I10=10
      DO 450 IX=I1,I2
      NC = 1
      IF (IX.GT.9) NC = 2
      IF (IX.GT.99) NC = 3
      XLAB = 10.**IX
      IF (IX.EQ.I2) XLAB = XLAB - NC*50.*9.*10.**(I2-1)/CFAC
      X10LAB=XLAB-(125./CFAC)*XLAB
      Y10LAB=YLAB-1.0
      CALL NUMBRG(A,X10LAB,Y10LAB,NC2,I10)
450   CALL NUMBRG(A,XLAB,YLAB,NC,IX)
455   CONTINUE
      IF (KOLOR.EQ.1) GO TO 456
      CALL SETSMG(A,113,RMIN)
      CALL SETSMG(A,114,YMIN)
      CALL SETSMG(A,115,RMAX)
      CALL SETSMG(A,116,YMAX)
456   CALL SETSMG(A,45,1.5)
      XPLACE=-2.5
      IF (ABSCA .EQ. 0.) XPLACE=-1.5
      CALL SETSMG(A,104,XPLACE)
      CALL TITLEG(A,NX,XTITLE,0,DUMMY,0,DUMMY)
      XPLACE = 2.0
      CALL SETSMG(A,104,XPLACE)
      CALL TITLEG(A,0,DUMMY,0,DUMMY,N3,TITLE3)
      XPLACE = XPLACE + 1.5
      CALL SETSMG(A,104,XPLACE)
      CALL TITLEG(A,0,DUMMY,0,DUMMY,N2,TITLE2)
      XPLACE = XPLACE + 1.5
      CALL SETSMG(A,104,XPLACE)
      CALL TITLEG(A,0,DUMMY,0,DUMMY,N1,TITLE1)
      CALL SETSMG(A,30,2.0)

      PLOT FIRST FAMILY

      BACK = RMIN - 200.
      YFAC = (YMAX-YMIN)/YY
      DELX = 1.5*DELTX
      IF (ABSCA .EQ. 1.) ALOGXO=ALOG10(XO)
      DO 500 LINE=1,NHL
      IF (RTYP.EQ.0) IND=INDEX(ICP(LINE))
      IF (RTYP.GT.0) IND = LINE
      LIM = NPTS(LINE)
      DO 460 NP=1,IDIM
      X = F1(NP,IND)
      Y = Y1(NP,IND)
      IF (X.EQ.999..OR.Y.EQ.999.) GO TO 460
      XP(1) = X
```

```
      YP(1) = Y
      NPT = 1
      I1  = NP + 1
      GO TO 470
460 CONTINUE
470 LIM = LIM + I1 -2
      DO 475 NP=I1,LIM
      NPT = NPT + 1
      XP(NPT) = F1(NP,IND)
      YP(NPT) = Y1(NP,IND)
      IF ( XP(NPT).GE.XO.AND.YP(NPT).NE.999.) GO TO 475
      IF ( (XP(NPT-1) - XP(NPT) ).GT.DELX) GO TO 474
      YP(NPT) = YP(NPT-1) + (XO - XP(NPT-1)) / (XP(NPT) - XP(NPT-1)) *
    1 (YP(NPT) - YP(NPT-1))
      XP(NPT) = XO
      GO TO 475
474 NPT = NPT-1
      GO TO 476
475 CONTINUE
476 CALL LINESG(A,NPT,XP,YP)
480 XRIGHT=XP(1)
      YRIGHT=YP(1)
      XLEFT=XP(NPT)
      YLEFT=YP(NPT)
      IF (XRIGHT .GT. XLEFT) GO TO 486
      XRIGHT=XP(NPT)
      YRIGHT=YP(NPT)
      XLEFT=XP(1)
      YLEFT=YP(1)
486 IF (ABSCA.EQ.1.) GO TO 490
      XHL(LINE) = BACK + (XLEFT-XO)*CFAC
      XHR(LINE) = RMIN + (XRIGHT-XO)*CFAC + 50.
      GO TO 495
490 XHL(LINE) = BACK + (ALOG10(XLEFT)-ALOGXO)*CFAC
      XHR(LINE) = RMIN + (ALOG10(XRIGHT)-ALOGXO)*CFAC + 10.
495 YHR(LINE) = YMIN + YRIGHT*YFAC
500 YHL(LINE) = YMIN + YLEFT*YFAC

      PLOT SECOND FAMILY

      DO 600 LINV=1,NVL
      DO 550 NP=1,IDIM
      IF (RTYP.NE.1) IND = LINV
      IF (RTYP.EQ.1) IND=INDEX(IGP(LINV))
      X = F2(NP,IND)
      Y = Y2(NP,IND)
      IF (X.EQ.999..OR.Y.EQ.999..OR.X.LT.XO.OR.X.GT.XF) GO TO 550
      KT = NP+1
      XV(1) = X
      YV(1) = Y
      NPT = 1
      GO TO 520
550 CONTINUE
520 DO 525 NPLT=KT,IDIM
      X = F2(NPLT,IND)
      Y = Y2(NPLT,IND)
      IF (X.EQ.999..OR.Y.EQ.999..OR.X.LT.XO.OR.X.GT.XF) GO TO 530
      NPT = NPT + 1
      XV(NPT) = X
      YV(NPT) = Y
525 CONTINUE
530 CALL LINESG(A,NPT,XV,YV)
540 CONTINUE
```

46

```
      IF (ABSCA.EQ.1.) GO TO 542
      XVL(LINV) = RMIN + (XV(NPT)-X0)*CFAC
      XVLB(LINV) = RMIN + (XV(1)-X0)*CFAC
      GO TO 545
  542 XVL(LINV) = RMIN + (ALOG10(XV(NPT)) - ALOGX0)*CFAC
      XVLB(LINV) = RMIN + (ALOG10(XV(1)) - ALOGX0)*CFAC
  545 YVL(LINV) = YMIN + YV(NPT)*YFAC
  600 YVLB(LINV) = YMIN + YV(1)*YFAC

      LABEL AND TITLE

      CALL SETSMG(A,23,0.)
      CALL OBJCTG(A,0.,YMIN,4095.,YMAX)
      CALL SUBJEG(A,0.,YMIN,4095.,YMAX)
      CALL SETSMG(A,14,1.)
      IF (ABSCA .EQ. 0.) XBAR(3) = RMIN + CFAC*DELTX
      IF (ABSCA.EQ.1.) XEAR(3) = RMIN + ALOG10(2.)*CFAC
      XBAR(1) = XBAR(3) - 25.
      XBAR(2) = XBAR(3) + 25.
      XBAR(4) = XBAR(3)
      XBAR(5) = XBAR(1)
      XBAR(6) = XBAR(2)
      CALL SETSMG(A,30,2.0)
      UNIT = DIVN*YFAC
      DO 675 J=1,2
      AD = 1.
      IF (J.EQ.2) AD = 3.
      DO 650 M=1,6
      IF (M.LE.3) YBAR(M) = AD*UNIT+ YMIN
  650 IF (M.GT.3) YBAR(M) = (AD+1.)*UNIT + YMIN
      CALL LINESG(A,6,XBAR,YBAR)
      YCOR = YBAR(1) + DIVN/2.*YFAC
      XL = XBAR(2) + 40.
      CALL SETSMG(A,52,2.)
      CALL LEGNDG(A,XL,YCOR,1,1HV)
      CALL SETSMG(A,52,0.)
      CALL TEXTG(A,NCHAR,DCHAR(1,J))
      CALL GETSMG(A,17,XLAST)
      CALL GETSMG(A,18,YLAST)
      XX = XLAST + 50. + 47.*NCHAR
      DL=ABS(DELL)
      DB=ABS(DELB)
      IF (J.EQ.1) CALL NUMBRG(A,XX,YLAST,CHFMT,DL)
      IF (J.EQ.2) CALL NUMBRG(A,XX,YLAST,CHFMT,DB)
  675 CONTINUE
      CALL SETSMG(A,45,1.0)
      DO 700 I=1,NHL
      P=ABS(LP(I))
      IF (P .LT. .01) FMT=6.3
      IF (P .GE. .01) FMT=5.2
      IF (P .GE. 10.) FMT=5.1
      IF (P .GE. 100.) FMT=6.1
      IF (P .GE. 1000.) FMT=6.0
      IF (P .GE. 10000.) FMT=7.0
      IF (P .EQ. 0.0) FMT=4.1
      IF (LLABEL.NE.1) CALL NUMBRG(A,XHL(I),YHL(I),FMT,LP(I))
  700 IF (LLABEL.NE.0) CALL NUMBRG(A,XHR(I),YHR(I),FMT,LP(I))
      CALL SETSMG(A,46,THETA1)
      CALL LEGNDG(A,XHZ1,YHZ1,NHT,HORIZ)
      CALL SETSMG(A,46,0.)
      DO 800 I=1,NVL
      V=ABS(BP(I))
      IF (V .LT. .01) FMT=6.3
      IF (V .GE. .01) FMT=5.2
```

```
      IF (V .GE. 10.) FMT=5.1
      IF (V .GE. 100.) FMT=6.1
      IF (V .GE. 1000.) FMT=6.0
      IF (V .GE. 10000.) FMT=7.0
      IF (V .EQ. 0.0) FMT=4.1
      IF (BLABEL.EQ.1) GO TO 750
      PVX(1) = XVL(I)
      PVX(2) = XVL(I) + DELTA
      PVY(1) = YVL(I)
      PVY(2) = YVL(I) + DELTA1
      CALL LINESG(A,2,PVX,PVY)
      DX=40.
      DY=30.
      X=PVX(2)-SIGN(DX,DELTA)
      Y=PVY(2)+SIGN(DY,DELTA1)
      CALL NUMBRG(A,X,Y,FMT,BP(I))
      IF (BLABEL.EQ.0) GO TO 800
  750 PVX(1) = XVLB(I)
      PVX(2) = PVX(1) + DELTA
      PVY(1) = YVLB(I)
      PVY(2) = PVY(1) -DELTA1
      CALL LINESG(A,2,PVX,PVY)
      DX=40.
      DY=30.
      X=PVX(2)-SIGN(DX,DELTA)
      Y=PVY(2)-SIGN(DY,DELTA1)
      CALL NUMBRG(A,X,Y,FMT,BP(I))
  800 CONTINUE
      CALL SETSMG(A,46,THETA2)
      CALL LEGNDG(A,XVT1,YVT1,NVT,VERT)
      CALL SETSMG(A,46,0.)
      CALL PAGEG(A,0,1,1)
      KARP = KARP - 1
      IF (KARP.GT.0) GO TO 1
      CALL EXITG(A)
      STOP
 1000 FORMAT(1H1,55X,18HLINES OF CONSTANT ,A8  /)
 1001 FORMAT(58X,A8 ,1H=,F10.3/13X,7HDEP VAR,9X,1HY,11X,A8 ,40X,7HDEP VA
     *R,10X,1HY,10X,A8  /)
 1002 FORMAT(10X,1PE12.5,4X,0PF6.1,5X,1PE12.5,37X,1PE12.5,4X,0PF6.1,5X,
     + 1PE12.5)
 1003 FORMAT(1H0,3X,31HNC. OF VERT GRID LINES CRAWN = ,I3)
 1007 FORMAT(I2)
 1017 FORMAT(1H0,3X,32HNC. OF HORIZ GRID LINES DRAWN = ,I3)
 1099 FORMAT(5X,I4,(8F12.2))
 2001 FORMAT(1H1,///////,36X,
     160H***** AN ERROR CONDITION HAS BEEN DISCOVERED IN INTERP *****,
     2////,49X,26HTHE ERROR RETURN CODE IS  ,I1,7H, WHERE,//,
     344X,40H1=   INT IS LESS THAN 1 CR GREATER THAN M,/,
     444X,44H     (THIS MEANS INT CAN NOT BE A VALID INDEX,/,
     544X,27H     INCREMENT IN XIN,YIN)..//,
     644X,52H2=   AN INDEX IN IGP(N) IS OUTSIDE THE INDEX RANGE OF,/,
     744X,50H     THE FIRST THROUGH LAST DATA POINTS OF CURVE N.)
      END
      BLOCKDATA

      INITIALIZE NAMELIST(SEE ALSO INIT)

      INTEGER RTYP,ELABEL
      COMMON /INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      COMMON /SCAL / CHOICE,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN,
     1BMIN,YY,KOLOR,NLP,NBP
      REAL*8  CHOICE
      REAL L,LP
```

48

```
      COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20),T(20),
     1             L(50,20),LP(20)
      COMMON /TITL/ N1,TITLE1(15),N2,TITLE2(15),N3,TITLE3(15),NX,
     1             XTITLE(15),NHT,HORIZ(9),NVT,VERT(9),XHZ,YHZ,THETA1,
     2             XVT,YVT,THETA2,DCHAR(2,2),NCHAR,CHFMT,ELABEL,LLABEL,
     3             DELTA,DELTA1,SUBDIV
      DATA CHOICE/'CRVFIT  '/,SUBDIV/1./
      DATA DELL,DELB,PO,NL,RTYP,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN
     *     /0. , 0., 4.,0, 0,   1.    ,0.,0.,1.   ,2.0 ,0.    ,0., 0./
      DATA  BMIN,YY,KOLOR /0.,100.,1/,ELABEL,LLABEL/2*0/
      DATA N1,N2,N3,XHZ,YHZ,THETA1,XVT,YVT,THETA2 /3*1,6*0./
      DATA NX,XTITLE/1,15*1H /
      DATA NHT,HORIZ/1,9*1H /
      DATA NVT,VERT /1,9*1H /
      DATA DCHAR/4*1H /,NCHAR/1/,CHFMT/4.2/
      DATA TITLE1,TITLE2,TITLE3 /15*1H  ,15*1H ,15*1H  /
      DATA LP,BP /40*999./
      DATA DELTA/100./,DELTA1/100./
      END
      SUBROUTINE READ
      DIMENSION A(4),C(4),IER(3),UPLOT(20),VPLOT(20)
      DIMENSION LABEL1(9),LABEL2(9)
      REAL L,LP
      COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20), T(20),
     *             L(50,20),LP(20)
      INTEGER RTYP,ELABEL,RTYPE,ULABEL,VLABEL
      COMMON /INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      COMMON /SCAL / CHOICE,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN,
     1BMIN,YY,KOLOR,NLP,NBP
      REAL*8   CHOICE
      COMMON /TITL/ N1,TITLE1(15),N2,TITLE2(15),N3,TITLE3(15),NX,
     1             XTITLE(15),NHT,HORIZ(9),NVT,VERT(9),XHZ,YHZ,THETA1,
     2             XVT,YVT,THETA2,DCHAR(2,2),NCHAR,CHFMT,ELABEL,LLABEL,
     3             DELTA,DELTA1,SUBDIV
      DATA IER/4HRTYP,4H BL ,4HABSC/
      NAMELIST /HEAD/ RTYPE,UMAX,UMIN,VMAX,VMIN,KOLOR,IPR,CHOICE,PO,
     2             DELTAU,DELTAV,DCHAR,NCHAR,CHFMT,UPLOT,NUPLOT,
     3             VPLOT,NVPLOT,ABSCA,XO,XF,DELTX,SUBDIV,XFMT,TITLE1,
     4             N1,TITLE2,N2,TITLE3,N3,XTITLE,NX,LABEL1,NLAB1,
     5             XPOS1,YPOS1,THETA1,LABEL2,NLAB2,XPOS2,YPOS2,
     6             THETA2,ULABEL,VLABEL,XTICK,YTICK
      EQUIVALENCE (RTYPE,RTYP),(DELTAU,DELB),(DELTAV,DELL),
     2           (VPLOT(1),LP(1)),(UPLOT(1),BP(1)),(VMAX,RLMAX),
     3           (UMAX,BMAX),(VMIN,RLMIN),(UMIN,BMIN),(NLAB1,NHT),
     4(LABEL1(1),HORIZ(1)),(NLAB2,NVT),(LABEL2(1),VERT(1)),(XPOS1,XHZ),
     5           (YPOS1,YHZ),(XPOS2,XVT),(YPOS2,YVT),(NVPLOT,NLP),
     6           (NUPLOT,NBP),(ULABEL,BLABEL),(VLABEL,LLABEL),
     7           (XTICK,DELTA),(YTICK,DELTA1)
      READ (5,HEAD)
      IF (RTYP .EQ. 1) RTYP=2
      L1 = NLP + 1
      L2 = NBP + 1
      IF(L1.GT.20)GO TO 91
      DO 1 I=L1,20
    1 LP(I) = 999.
   91 CONTINUE
      IF(L2.GT.20)GO TO 92
      DO 2 I=L2,20
    2 BP(I) = 999.
   92 CONTINUE
      IF (RTYP.LT.0.OR.RTYP.GT.2) GO TO 50
      IF (ABSCA.LT.0..OR.ABSCA.GT.1.) GO TO 52
      IF (ABSCA.EQ.0..AND.(XO.EQ.XF) ) GO TO 62
      IF (N1.EQ.0.OR.N2.EQ.0.OR.N3.EQ.0) WRITE(6, 700)
```

```
      IF ((XHZ .LT. -1.25) .OR. (XHZ .GT. 8.0625)) WRITE(6,700)
      IF ((XVT .LT. -1.25) .OR. (XVT .GT. 8.0625)) WRITE(6,700)
      IF ((YHZ .LT. -.4) .OR. (YHZ .GT. 5.90)) WRITE(6,700)
      IF ((YVT .LT. -.4) .OR. (YVT .GT. 5.90)) WRITE(6,7C0)
      IF (KOLOR.LT.1. OR.KOLOR.GT.2) KOLOR = 1
      IF (BLABEL.LT.0.OR.BLABEL.GT.2) BLABEL = 0
      IF (LLABEL.LT.0.OR.LLABEL.GT.2) LLABEL = 0
      I = 0
      NL = 0
      ML = 0
      IF (RTYP.EQ.1) GO TO 31

C     READ FIRST FAMILY

  5 I = I + 1
      READ (5,100) KARDS,T(I)
      IF (KARDS.EQ.0) GO TO 30
      NL = NL + 1
      DO 10 N=1,KARDS
      IFIRST = 4*(N-1)+1
      LAST = IFIRST + 3
      READ (5,200) (B(K,I),F1(K,I),K=IFIRST,LAST)
 10 CONTINUE
      DO 15 M=IFIRST,LAST
      IF (B(M,I).NE.0..AND.F1(M,I).NE.0.) GO TO 15
      B(M,I) = 999.
      F1(M,I) = 999.
 15 CONTINUE
      GO TO 5
 30 IF (RTYP.EQ.0) GO TO 99

C     READ SECOND FAMILY IF NEEDED

 31 I = 0
 32 I = I + 1
      READ (5,100) KARDS,T(I)
      IF (KARDS.EQ.0) GO TO 99
      ML = ML + 1
      DO 35 N=1,KARDS
      IFIRST = 4*(N-1)+1
      LAST  = IFIRST+3
 35 READ (5,200) (L(K,I),F2(K,I),K=IFIRST,LAST)
      DO 40 M=IFIRST,LAST
      IF (L(M,I).NE.0..AND.F2(M,I).NE.0.) GO TC 40
      L(M,I) = 999.
      F2(M,I)= 999.
 40 CONTINUE
      GO TO 32
 50 N=1
      GO TO 60
 52 N=3
 60 WRITE(6,400) IER(N)
      GO TO 999
 62 WRITE(6,600)
999 STOP
 99 CONTINUE
      IF (ML.GT.NL) NL=ML
      IF(I.NE.21) T(I)=999.
      IF(RTYP.EQ.2) RETURN
      IF (IPR.EQ.1) CALL WRIT (IPR)
      WRITE(6,801)
      IF(IPR.NE.1) WRITE(6,900)
      WRITE(6,800) T
```

```
      RETURN
100 FORMAT(67X,I2,5X,F6.0)
100 FORMAT(67X,I2,6X,F5.2)
200 FORMAT(4(F8.5,1X,E10.3),4X)
200 FORMAT(4(F6.5,E8.3,1X))
201 FORMAT(2(F6.5,E8.3,16X),20X)
300 FORMAT(I2)
400 FORMAT(1H1,31H**********   VALUE OF VARIABLE ,A4,52H NOT IN CORREC
   1T RANGE.   RUN TERMINATES.   ********** // 1X
   288H**********             VARIABLE           CORRECT RANGE
   3                      **********/1X,
   488H**********             RTYP                0 - 2
   5                      **********/1X,
   688H**********             NL                  2 - 20
   7                      **********/1X,
   888H**********             ABSCA               0 OR 1
   9                      **********/1X,10H**********68X,10H********** )
500 FORMAT(1H1,88H**********   VARIABLES DELL AND DELB MUST BOTH BE NO
   1N-ZERO.  RUN TERMINATES.   ********** )
600 FORMAT(1H1,99H**********   IF LINEAR SCALING DESIRED, ALL FOUR VAR
   1IABLES XO,XF,DELTX,XFMT MUST BE READ.   *******   /1X,93H**********
   2*           CHECK TO MAKE SURE XO IS NOT EQUAL TO XF.
   3         ******* )
700 FORMAT(1H1,102H**********   WARNING - GRAPH TITLES HAVE NOT BEEN S
   1PECIFIED.  BLANKS WILL BE SUBSTITUTED.   ********** ///)
900 FORMAT(1H1)
800 FORMAT(1H0,10(1X,F10.3,1X,),/,1X,10(1X,F10.3,1X))
801 FORMAT(1H0,3X,36HTHE-FOLLOWING-LINES-HAVE-BEEN-INPUT.)
      END
      SUBROUTINE WRIT (IPR)
      DIMENSION A(4),C(4),IER(3),UPLOT(20),VPLOT(20)
      DIMENSION LABEL1(9),LABEL2(9)
      REAL L,LP
      COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20), T(20),
     *            L(50,20),LP(20)
      INTEGER RTYP,ELABEL,RTYPE,ULABEL,VLABEL
      COMMON /INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      COMMON /SCAL / CHOICE,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN,
     1BMIN,YY,KOLOR,NLP,NBP
      REAL*8   CHOICE
      COMMON /TITL/ N1,TITLE1(15),N2,TITLE2(15),N3,TITLE3(15),NX,
     1            XTITLE(15),NHT,HORIZ(9),NVT,VERT(9),XHZ,YHZ,THETA1,
     2            XVT,YVT,THETA2,DCHAR(2,2),NCHAR,CHFMT,ELABEL,LLABEL,
     3            DELTA,DELTA1,SUBDIV
      EQUIVALENCE (RTYPE,RTYP),(DELTAU,DELB),(DELTAV,DELL),
     2            (VPLOT(1),LP(1)),(UPLOT(1),BP(1)),(VMAX,RLMAX),
     3            (UMAX,BMAX),(VMIN,RLMIN),(UMIN,BMIN),(NLAB1,NHT),
     4(LABEL1(1),HORIZ(1)),(NLAB2,NVT),(LABEL2(1),VERT(1)),(XPOS1,XHZ),
     5            (YPOS1,YHZ),(XPOS2,XVT),(YPOS2,YVT),(NVPLOT,NLP),
     6            (NUPLOT,NBP),(ULABEL,BLABEL),(VLABEL,LLABEL),
     7            (XTICK,DELTA),(YTICK,DELTA1)
      WRITE(6,1) IPR
      WRITE(6,2)
      WRITE(6,3)RTYPE
      WRITE(6,4) PO
      WRITE(6,5) DELTAU
      WRITE(6,6) DELTAV
      WRITE(6,7) (VPLOT(I),I=1,NVPLOT)
      WRITE(6,8) NVPLOT
      WRITE(6,9) (UPLOT(I),I=1,NUPLOT)
      WRITE(6,10) NUPLOT
      WRITE(6,11)
      WRITE(6,12) CHOICE
      WRITE(6,13)
```

```
      WRITE(6,14) UMAX
      WRITE(6,15) UMIN
      WRITE(6,16) VMAX
      WRITE(6,17) VMIN
      WRITE(6,18) KOLOR
      WRITE(6,19) ABSCA
      WRITE(6,20) X0
      WRITE(6,21) XF
      WRITE(6,22) DELTX
      WRITE(6,23) SUBDIV
      WRITE(6,24)
      WRITE(6,25) (TITLE1(I),I=1,15)
      WRITE(6,26) N1
      WRITE(6,27) (TITLE2(I),I=1,15)
      WRITE(6,28) N2
      WRITE(6,29) (TITLE3(I),I=1,15)
      WRITE(6,30) N3
      WRITE(6,31) (XTITLE(I),I=1,15)
      WRITE(6,52) NX
      WRITE(6,32) (LABEL1(I),I=1,9)
      WRITE(6,33) NLAB1
      WRITE(6,34) XPOS1
      WRITE(6,35) YPOS1
      WRITE(6,36) THETA1
      WRITE(6,37) (LABEL2(I),I=1,9)
      WRITE(6,38) NLAB2
      WRITE(6,39) XPOS2
      WRITE(6,40) YPOS2
      WRITE(6,41) THETA2
      WRITE(6,42) ULABEL
      WRITE(6,43) VLABEL
      WRITE(6,44) XTICK
      WRITE(6,45) YTICK
      WRITE(6,46) ((DCHAR(I,J),I=1,2),J=1,2)
      WRITE(6,47) NCHAR
      WRITE(6,48) CHFMT
      WRITE(6,49) XFMT
      RETURN
    1 FORMAT(1H1,5X,6HIPR = ,I2)
    2 FORMAT(1H0,30X,33HCARPET  PLOT  CONTROL  PARAMETERS)
    3 FORMAT(1H0,5X,8HRTYPE = ,I1)
    4 FORMAT(1H ,5X,5HPO = ,F4.1)
    5 FORMAT(1H ,5X,9HDELATU = ,F8.5)
    6 FORMAT(1H ,5X,9HDELTAV = ,F8.5)
    7 FORMAT(1H ,5X,8HVPLOT = ,/,5(1PE10.3,3X))
    8 FORMAT(1H ,5X,9HNVPLOT = ,I2)
    9 FORMAT(1H ,5X,8HUPLOT = ,/,5(1PE10.3,3X))
   10 FORMAT(1H ,5X,9HNUPLOT = ,I2)
   11 FORMAT(1H0,30X,25HINTERPOLATION  PARAMETERS)
   12 FORMAT(1H0,5X,9HCHCICE = ,A8)
   13 FORMAT(1H0,30X,16HGRID  PARAMETERS)
   14 FORMAT(1H0,5X,7HUMAX = ,F4.1)
   15 FORMAT(1H ,5X,7HUMIN = ,F4.1)
   16 FORMAT(1H ,5X,7HVMAX = ,F4.1)
   17 FORMAT(1H ,5X,7HVMIN = ,F4.1)
   18 FORMAT(1H ,5X,8HKOLOR = ,I1)
   19 FORMAT(1H ,5X,8HABSCA = ,F4.1)
   20 FORMAT(1H ,5X,5HX0 = ,1PE12.5)
   21 FORMAT(1H ,5X,5HXF = ,1PE12.5)
   22 FORMAT(1H ,5X,8HDELTX = ,1PE12.5)
   23 FORMAT(1H ,5X,9HSUBDIV = ,1PE12.5)
   24 FORMAT(1H0,30X,34HLABELING  AND  TITLING  PARAMETERS)
   25 FORMAT(1H0,5X,9HTITLE1 = ,15A4)
   26 FORMAT(1H ,5X,5HN1 = ,I2)
```

```
27 FORMAT(1H ,5X,5HTITLE2 = ,15A4)
28 FORMAT(1H ,5X,5HN2 = ,I2)
29 FORMAT(1H ,5X,5HTITLE3 = ,15A4)
30 FORMAT(1H ,5X,5HN3 = ,I2)
31 FORMAT(1H ,5X,5HXTITLE = ,15A4)
52 FORMAT(1H ,5X,5HNX = ,I2)
32 FORMAT(1H ,5X,5HLABEL1 = ,9A4)
33 FORMAT(1H ,5X,6HNLAB1 = ,I2)
34 FORMAT(1H ,5X,6HXPOS1 = ,F8.5)
35 FORMAT(1H ,5X,6HYPOS1 = ,F8.5)
36 FORMAT(1H ,5X,9HTHETA1 = ,F5.2)
37 FORMAT(1H ,5X,7HLAB2 = ,9A4)
38 FORMAT(1H ,5X,6HNLAB2 = ,I2)
39 FORMAT(1H ,5X,6HXPOS2 = ,F8.5)
40 FORMAT(1H ,5X,6HYPOS2 = ,F8.5)
41 FORMAT(1H ,5X,9HTHETA2 = ,F8.5)
42 FORMAT(1H ,5X,5HULABEL = ,I1)
43 FORMAT(1H ,5X,9HVLABEL = ,I1)
44 FORMAT(1H ,5X,6HXTICK = ,F6.2)
45 FORMAT(1H ,5X,6HYTICK = ,F6.2)
46 FORMAT(1H ,5X,6HDCHAR = ,2A4)
47 FORMAT(1H ,5X,6HNCHAR = ,I1)
48 FORMAT(1H ,5X,6HCHFMT = ,F8.5)
49 FORMAT(1H ,5X,7HXFMT = ,F8.5)
   END
   SUBROUTINE SUPER(XMIN,XMAX,YMIN,YMAX)
   COMMON/PLOT/A(200)
   DIMENSION PVX(2),PVY(2)

   ADDS REGISTRATION MARKS
   XMIN,XMAX,YMIN AND YMAX ARE RASTER COCOS OF GRID.

   X1=XMIN-350.
   X2=XMAX+350.
   Y1=YMIN-100.
   Y2=YMAX+100.
   CALL OBJCTG(A,X1,Y1,X2,Y2)
   CALL SETSMG(A,14,1.)
   ITIME=0
   CALL SETSMG(A,30,0.5)
10 ITIME=ITIME+1
   GO TO (1,2,3,4),ITIME
1  XC=250.
   YC=YMIN
   GO TO 20
2  XC=250.
   YC=YMAX
   GO TO 20
3  XC=3850.
   YC=YMAX
   GO TO 20
4  XC=3850.
   YC=YMIN
20 PVX(1)=XC-75.0
   PVX(2)=XC+75.0
   PVY(1)=YC
   PVY(2)=YC
   CALL LINESG(A,2,PVX,PVY)
   PVX(1)=XC
   PVX(2)=XC
   PVY(1)=YC-75.0
   PVY(2)=YC+75.0
   CALL LINESG(A,2,PVX,PVY)
   CALL SETSMG(A,83,5.0)
```

```
      CALL CIRARG(A,XC,YC,37.5,0.,360.)
      IF(ITIME.LT.4) GO TO 10
      CALL SETSMG(A,30,1.)
      RETURN
      END
      SUBROUTINE INITAL

      INITIALIZE NAMELIST

      REAL L,LP
      COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20),T(20),
     1             L(50,20),LP(20)
      COMMON /PSI/  PSI1(50,20),PSI2(50,20)
      COMMON /LAB8/ LAB1(8),LAB2(8)
      DATA IBLANK/1H /
      IDIM = 50
      DO 50 I=1,8
      LAB1(I) = IBLANK
      LAB2(I) = IBLANK
   50 CONTINUE
      DO 100 I=1,20
      IGP(I) = 0
      DO 100 J=1,IDIM
      PSI1(J,I) = 999.
      PSI2(J,I) = 999.
      B(J,I) = 999.
      F1(J,I) = 999.
      F2(J,I) = 999.
  100 L(J,I) = 999.
      RETURN
      END
      SUBROUTINE GRIDPT(RP,A,IGP)
      DIMENSION RP(20),A(50,20),IGP(20)
      INTEGER RTYP
      COMMON /INIT/ CELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      DO 40 I=1,NL
      P = RP(I)
      IF (P.NE.999.) GO TO 20
      GO TO 99
   20 DO 30 J=1,IDIM
      IF (A(J,1).NE.P) GO TO 30
      IGP(I) = J
      GO TO 40
   30 CONTINUE
   40 CONTINUE
   99 RETURN
      END
      SUBROUTINE SCALE(PSI,     RNUM,PHI)
      DIMENSION PSI(50,20)
      REAL L,LP
      COMMON /VAL/ BP(20),B(50,20),F1(50,20),F2(50,20),IGP(20),T(20),
     1             L(50,20),LP(20)
      INTEGER RTYP
      COMMON /INIT/ CELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP

      SCALES CARPET PLOT IN VERT DIRECTION.

      DO 110 I=1,NIP
      P = PSI(1,I)
      IF (P.EQ.999.) GO TO 110
      PMN = P
      PMX = PMN
      IMN = I
      KMN = 1
```

```
          GO TO 111
  110 CONTINUE
  111 DO 115 I=1,NIP
      DO 115 J=1,NFP
      K = IGP(J)
      PP = PSI(K,I)
      IF (PP.EQ.999.) GO TO 115
      IF (PP.GE.PMN) GO TO 114
      PMN =PP
      KMN = J
      IMN = I
      GO TO 115
  114 IF (PP.GT.PMX) PMX=PP
  115 CONTINUE
      RNUM = INT(2.*PO+PMX-PMN+.01)
      IF (RTYP.EQ.0) GO TO 120
      ITEMP = IMN
      IMN   = KMN
      KMN   = ITEMP
  120 AMINL = LP(KMN)
      BMIN  = BP(IMN)
      PHI = PO - AMINL/DELL - BMIN/DELB
      RETURN
      END
      SUBROUTINE LINT(A,T,F)
      DIMENSION A(50,20),T(20),F(50,20)
      INTEGER RTYP
      COMMON /INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      INT = IDIM/NL
      IL = NL - 1
      DO 20 I=1,IL
      K1 = INT*(I-1)+1
      K2 = K1 + INT - 1
      K = K1 + 1
      KN = K2 + 1
      A(K1,1) = T(I)
      DELA = T(I+1) - T(I)
      DEL=DELA/INT
      DO 20 IT=1,NIP
      F1 = F(K1,IT)
      F2 = F(KN,IT)
      IF ( (F1.EQ.999..OR.F2.EQ.999.).AND.IT.NE.1) GO TO 20
      FAC = (F2-F1)/DELA
      DO 10   N=K,K2
      IF (IT.EQ.1) A(N,1) = A(N-1,1) + DEL
      IF (F1.EQ.999..OR.F2.EQ.999.) GO TO 10
      F(N,IT) = F1 + (A(N,1) - A(K1,1))*FAC
  10 CONTINUE
  20 CONTINUE
      IND = INT*IL + 1
      A(IND,1) = T(NL)
      RETURN
      END
      SUBROUTINE REVERS(X,Y,LGP)
      DIMENSION X(50,20),Y(50,20),XT(50),YT(50),LGP(20)
      REAL L,LP
      COMMON /VAL/ EP(20),B(50,20),F1(50,20),F2(50,20),IGP(20),T(20),
     1            L(50,20),LP(20)
      COMMON / INIT/ DELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      DO 25 I=1,NFP
      J = NFP - I + 1
  25 LGP(I) = IDIM - IGP(J) + 1
      DO 100 I=1,NIP
```

55

```
      DO 50 J=1,IDIM
      K=IDIM-J+1
      XT(J) = X(K,I)
   50 YT(J) = Y(K,I)
      DO 100 N=1,IDIM
      X(N,I) = XT(N)
  100 Y(N,I) = YT(N)
      RETURN
      END
      SUBROUTINE FINDF(FP,A,AY,CY,T)
      REAL L,LP,LL
      INTEGER RTYP
      COMMON /INIT/ CELL,DELB,PO,NL,NFP,NIP,IDIM,RTYP
      DIMENSION FP(20),A(50,20),AY(50,20),CY(50,20),T(20)
      DIMENSION LABEL(2),MIS(20)
      DATA LABEL/4H X2=,4H X1=/
      IF (RTYP.EQ.1) GO TO 5
      ITP = LABEL(2)
      IRD = LABEL(1)
      LTYP= 2
      GO TO 6
    5 ITP = LABEL(1)
      IRD = LABEL(2)
      LTYP= 1
    6 DO 7 I=1,NIP
    7 MIS(I)= 0
      LIM = IDIM - 1
      INT = IDIM/NL
      DO 50 IL=1,NL
      IP = INT*(IL-1)+1
      DO 20 I=1,NIP
      BFIX = FP(I)
      DO 15 J=1,LIM
      P1 = A(J,IL)
      IF (P1.EQ.999.) GO TO 14
      IN2 = J+1
      P2 = A (IN2,IL)
      IF (P2.EQ.999.) GO TO 14
      X1 = AY(J,IL)
      IF (P1.NE.BFIX) GO TO 10
      CY(IP,I) = X1
      GO TO 20
    9 CY(IP,I) = AY(IN2,IL)
      GO TO 20
   10 CONTINUE
      IF (P2.LT.BFIX) GO TO 15
      IF (P1.LT.BFIX.AND.P2.GT.BFIX) GO TO 16
      IF (P2.EQ.BFIX) GO TO 9
   15 CONTINUE
   14 CONTINUE
      MIS(I) = MIS(I) + 1
      IF (P2.LT.BFIX) GO TO 50
      GO TO 20
   16 CONTINUE
      X2 = AY(IN2,IL)
      CY(IP,I) = X1 + (BFIX-P1)*(X2-X1)/(P2-P1)
   20 CONTINUE
   50 CONTINUE
      IER = 0
      I1  = 1
   55 DO 60 I=I1,NIP
      IN  = I + IER
      IF (MIS(IN).LT.NL) GO TO 60
```

```
      WRITE(6,1001) ITP,FP(I),LTYP
      IER = IER + 1
      J   = I + 1
      GO TO 70
   60 CONTINUE
      GO TO 999
   70 DO 80 N=J,NIP
   80 FP(N-1)=FP(N)
      FP(NIP)= 999.
      NIP = NIP - 1
      II  = J-1
      IF (II.LE.NIP) GO TO 55
      WRITE(6,1002)  NIP, (FP(II),II=1,NIP)
  999 RETURN
 1001 FORMAT(1H1,9HTHE VALUE,A4,F6.2,40H DOES NOT FALL ON ANY LINE OF CO
     1NSTANT X,I1,1H.)
 1002 FORMAT(5X,43HTHE NUMBER OF INTERPOLATED LINES CHANGED TO,I3,30H.TH
     1E NEW ARRAY IS AS FOLLOWS -/5X,20F6.2)
      END
      SUBROUTINE INTERP(XIN,YIN,LGP,INT,M,N,IERR)

      SPLINE OR CRVFIT INTERPLLATION DETERMINED BY CHOICE

      COMMON /SCAL / CHOICE,ABSCA,XO,XF,DELTX,XFMT,RLMAX,BMAX,RLMIN,
     1BMIN,YY,KOLOR,NLP,NBP
      REAL*8  CHOICE
      REAL*8 SPLINE
      DATA SPLINE/8HSPLINE  /
      DIMENSION XIN(M,N),YIN(M,N),LGP(1)
      COMMON X(50),U(50),S(50),DEL(50),Q1,QN,NFTS
      IDEBUG=0
      IF (IERR .EQ. 0) GO TO 150
      IDEBUG=1
      WRITE(6,100)
  100 FORMAT(1H1,//,49X,35H***** INTERP HAS BEEN ENTERED *****,//)
      WRITE(6,110) M,N,INT
  110 FORMAT(9X,6H  M = ,I6,9X,6H  N = ,I6,11X,6HINT = ,I6,///)
      WRITE(6,115) (LGP(I),I=1,N)
  115 FORMAT(40X,
     154HTHE INDEX OF THE FIRST PLOTTED POINT FOR EACH CURVE IS,//,
     24(40X,5(I2,8X),/),///)
      DO 140 J=1,N
      WRITE(6,120) J
  120 FORMAT(1X,//,48X,12HCURVE NO. = ,I6,5X,13HXIN,YIN PAIRS,//)
      WRITE(6,130) (XIN(I,J),YIN(I,J),I=1,M)
  130 FORMAT(1X,4(F10.2,3X,E10.2,8X),/)
  140 CONTINUE
  150 CONTINUE
      NCURVE=0
      IERR=0
      Q1=0.0
      QN=0.0
      Q1STRT=0.0
      QNSTRT=0.0
      IF ((INT .GT. 0) .AND. (INT .LE. M)) GO TO 1500
      IERR=1
      GO TO 9999
 1500 CONTINUE
 2000 NCURVE=NCURVE+1
      IF (NCURVE .GT. N) GO TO 9999
      DO 2100 IFIRST=1,M
         IF (XIN(IFIRST,NCURVE) .NE. 999.) GO TO 3000
 2100    CONTINUE
      GO TO 9999
```

```
3000 NPTS=1
     X(1)=XIN(IFIRST,NCURVE)
     U(1)=YIN(IFIRST,NCURVE)
     DO 3200 I=IFIRST,M
         IF (XIN(I,NCURVE) .EQ. 999.) GO TO 3600
         ILAST=I
3200     CONTINUE
3600 DO 3620 I=1,N
         IF (LGP(I) .LE. 0) GO TO 3650
         IF (LGP(I) .GE. IFIRST) GO TO 3640
3620     CONTINUE
3640 IF (LGP(I) .LE. ILAST) GO TO 3700
3650 IERR=2
     GO TO 9999
3700 INDEX=LGP(I)
     IF (INDEX .EQ. IFIRST) GO TO 4000
3800 IF ((INDEX-INT) .LE. IFIRST) GO TO 4000
     INDEX=INDEX-INT
     GO TO 3800
4000 IF (INDEX .GE. ILAST) GO TO 4200
     IF (INDEX .EQ. IFIRST) NPTS=NPTS-1
     NPTS=NPTS+1
     X(NPTS)=XIN(INDEX,NCURVE)
     U(NPTS)=YIN(INDEX,NCURVE)
     INDEX=INDEX+INT
     GO TO 4000
4200 NPTS=NPTS+1
     X(NPTS)=XIN(ILAST,NCURVE)
     U(NPTS)=YIN(ILAST,NCURVE)
     IF (CHOICE .NE. SPLINE) GO TO 6000
     IF (ABSCA.EQ.0.) GO TO 4400
     DO 4300 I=1,NPTS
         U(I)=ALOG10(U(I))
4300     CONTINUE
4400 IF (NPTS.LE.3) GO TO 5400
     Q1=(U(2)-U(1))/(X(2)-X(1))
     QN=(U(NPTS)-U(NPTS-1))/(X(NPTS)-X(NPTS-1))
     Q1STRT=Q1
     QNSTRT=QN
     CALL SPLN2
     NUMPT=1
5000 DO 5200 I=IFIRST,ILAST
         IF (XIN(I,NCURVE) .NE. X(NUMPT)) GO TO 5100
         NUMPT=NUMPT+1
         GO TO 5200
5100     YIN(I,NCURVE)=SPLIN(XIN(I,NCURVE))
         IF (ABSCA .EQ. 1.) YIN(I,NCURVE)=10.**(YIN(I,NCURVE))
5200     CONTINUE
5300 CONTINUE
     GO TO 5800
5400 IF ( NPTS .EQ. 2) GO TO 5700
     A =( (X(1)-X(2))*(U(2)-U(3)) - (X(2)-X(3))*(U(1)-U(2)) ) /
    1 ( (X(2)**2-X(3)**2)*(X(1)-X(2)) - (X(1)**2-X(2)**2)*(X(2)-X(3)) )
     B = ( U(1) - U(2) - A*( X(1)**2. - X(2)**2. ))/( X(1) - X(2) )
     C = U(3) - A*X(3)**2. - B*X(3)
     MLAST = ILAST - INT
     DO 5600 I=IFIRST,MLAST,INT
     I1 = I + 1
     I2 = I + INT - 1
     DO 5600 ND=I1,I2
     YIN(ND,NCURVE) = A*XIN(ND,NCURVE)**2. + B*XIN(ND,NCURVE) + C
5600 IF(ABSCA.EQ.1.) YIN(ND,NCURVE) = 10.**YIN(ND,NCURVE)
     GO TO 5800
5700 CONTINUE
```

```
      ND1 = IFIRST + 1
      ND2 = ILAST - 1
      DO 5750 I=ND1,ND2
      YIN(I,NCURVE) = U(1) + (XIN(I,NCURVE)-X(1)) /(X(2)-X(1))*
     1 (U(2) - U(1))
 5750 IF (ABSCA.EQ.1.) YIN(I,NCURVE) = 10.**YIN(I,NCURVE)
 5800 IF ( IDEBUG .EQ. 0) GO TO 2000
      WRITE(6,200)
  200 FORMAT(1H1,//,48X,37H***** INTERMEDIATE INTERP DEBUG *****,//)
      WRITE(6,210) NCURVE,NPTS,IFIRST,ILAST
  210 FORMAT(9X,8HNCURVE =,I6,5X,6HNPTS =,I6,5X,8HIFIRST =,I6,5X,
     1           7HILAST =,I6,//)
      IF (NPTS.LT.2) WRITE(6,220)
  220 FORMAT(1X,///,
     145X,44H********************************************,/,
     245X,44H*   THIS CURVE HAS FEWER THAN  TWO   POINTS  *,/,
     345X,44H*        NO INTERPOLATION HAS BEEN DONE      *,/,
     445X,44H********************************************,///)
      WRITE(6,230) Q1STRT,QNSTRT,Q1,QN
  230 FORMAT(//,57X,11HFIRST POINT,10X,10HLAST POINT,//,
     146X,6HSTART ,6X,F9.4,11X,F9.4,//,
     246X,6HFINISH,6X,F9.4,11X,F9.4,//)
      WRITE(6,120) NCURVE
      WRITE(6,130) (XIN(I,NCURVE),YIN(I,NCURVE),I=1,M)
      GO TO 2000
 6000 CONTINUE
      IF (ABSCA .EQ. 0.) GO TO 6200
      DO 6100 I=1,NPTS
         U(I)=ALOG10(U(I))
 6100    CONTINUE
 6200 CONTINUE
      MODE=1
      I1=NPTS
      I2=INT
      CALL CRVFIT(MODE,I1,X,U,I2,I3,XIN(IFIRST,NCURVE),YIN(IFIRST,NCURVE
     *))
      IF (ABSCA .EQ. 0.) GO TO 6500
      DO 6400 I=IFIRST,ILAST
         YIN(I,NCURVE)=10.**(YIN(I,NCURVE))
 6400    CONTINUE
 6500 CONTINUE
      GO TO 5800
 9999 IF (IDEBUG .EQ. 0) GO TO 7000
      WRITE(6,300)
  300 FORMAT(1H1,//,50X,34H***** INTERP IS BEING EXITED *****,//)
      WRITE(6,310) IERR,NCURVE,IFIRST,ILAST,NPTS
  310 FORMAT(9X,6HIERR =,I6,5X,8HNCURVE =,I6,5X,8HIFIRST =,I6,5X,
     1           7HILAST =,I6,5X,6HNPTS =,I6,//)
      WRITE(6,110) M,N,INT
      DO 340 J=1,N
      WRITE(6,120) J
      WRITE(6,130) (XIN(I,J),YIN(I,J),I=1,M)
  340 CONTINUE
  400 FORMAT(5X,2HA=,F6.2,3H B=,F6.2,3H C=,F6.2)
 7000 RETURN
      END
      SUBROUTINE CRVFIT(MODE,LO,X,Y,MO,NO,U,V)

      INTERPOLATION BY CRVFIT

      DIMENSION      X(50),Y(50),U(50),V(50)
      DIMENSION      AO(2),BO(2)
      EQUIVALENCE    (A,AO(1)),(B,AO(2)),(BO(1)),(C,BO(2)),
     1               (PO,X2),(QO,Y2),(DX,A2),(DY,B2)
      EQUIVALENCE    (FLM,TS,Z),(JP,JS),
```

```
      1               (DU,DA,D,X1),(DV,DB,R,Y1),
      2               (S2,S20,A1),(S3,S03,B1),
      3               (P1,S12),(P2,C12),(P3,R12),
      4               (Q1,S13),(Q2,C13),(Q3,R13)
       SCR(SIJ,CIJ)=ABS(SIJ)-ABS(CIJ)*1.0E-8
       MO=MODE
       L=L0
       LPS=L0
       M=M0
       KP1=L*M+1
       IP=L+1
       DO 10  JP=1,L
          KP1=KP1-M
          IP=IP-1
          U(KP1)=X(IP)
          V(KP1)=Y(IP)
   10     CONTINUE
       KP2=1
       KP3=1
       DO 20   I=2,L
          KP2=KP2+M
          IF(U(KP2).EQ.U(KP3).AND.V(KP2).EQ.V(KP3))  GO TO 20
          KP3=KP3+M
          U(KP3)=U(KP2)
          V(KP3)=V(KP2)
   20     CONTINUE
       L=KP3/M+1
       N=KP3
       IF(N.EQ.1)  GO TO 890
       IF(MO.NE.2) GO TO 50
   30 DO 40   KP4=1,N,M
          TS=U(KP4)
          U(KP4)=V(KP4)
          V(KP4)=TS
   40     CONTINUE
   50 MM1=M-1
       DZ=1.0/M
       IF(L.EQ.2)  GO TO 100
       LM1=L-1
       GO TO  200
  100 DU=(U(N)-U(1))*DZ
       DV=(V(N)-V(1))*DZ
       DO 110 KS=1,MM1
          U(KS+1)=U(KS)+DU
          V(KS+1)=V(KS)+DV
  110     CONTINUE
       GO TO  800
  200 X3=U(1)
       Y3=V(1)
       X4=U(M+1)
       Y4=V(M+1)
       K5=1+M+M
       X5=U(K5)
       Y5=V(K5)
       A3=X4-X3
       B3=Y4-Y3
       A4=X5-X4
       B4=Y5-Y4
       S34=A3*B4-A4*B3
       C34=A3*A4+B3*B4
       R34=SCR(S34,C34)
       IF(MO.LE.3)  GC TO 400
       K1=N-M-M
       X1=U(K1)
```

```
        Y1=V(K1)
        K2=K1+M
        X2=U(K2)
        Y2=V(K2)
        A1=X2-X1
        B2=Y2-Y1
        A2=X3-X2
        B2=Y3-Y2
230     S12=A1*B2-A2*B1
        C12=A1*A2+B1*B2
        R12=SCR(S12,C12)
        S13=A1*B3-A3*B1
        C13=A1*A3+B1*B3
        R13=SCR(S13,C13)
        S23=A2*B3-A3*B2
        C23=A2*A3+B2*B3
        R23=SCR(S23,C23)
        S24=A2*B4-A4*B2
        C24=A2*A4+B2*B4
        R24=SCR(S24,C24)
        ASSIGN 240 TO LBL
        GO TO  500
240     DO 290   I=2,L
        K5=K5+M
        X2=X3
        Y2=Y3
        X3=X4
        Y3=Y4
        X4=X5
        Y4=Y5
        A2=A3
        B2=B3
        A3=A4
        B3=B4
        S12=S23
        R12=R23
        S13=S24
        R13=R24
        S23=S34
        C23=C34
        R23=R34
        COS2=SGN*COS3
        SIN2=SGN*SIN3
        IF(I.LT.LM1)   GO TO 270
        IF(MD.EQ.4)    GO TO 260
        IF(I.EQ.LM1)   GO TO 450
        A4=A5
        B4=B5
        GO TO 280
260     IF(K5.GT.N)   K5=1+M
270     X5=U(K5)
        Y5=V(K5)
        A4=X5-X4
        B4=Y5-Y4
280     S24=A2*B4-A4*B2
        C24=A2*A4+B2*B4
        R24=SCR(S24,C24)
        S34=A3*B4-A4*B3
        C34=A3*A4+B3*B4
        R34=SCR(S34,C34)
        ASSIGN 600 TO LBL
        GO TO 500
290     CONTINUE
        GO TO 800
```

```
400 IF(R34.LE.0.0)   GO TO 430
    IF(MD.EQ.3)       GO TO 420
410 A1=A3
    A2=A3
    DB=2.*A3*S34/(A4*(A3+A4))
    B2=B3-DB
    B1=B2-DB
    GO TO 230
420 DA=A4-A3
    DB=B4-B3
    A2=A3-DA
    B2=B3-DB
    A1=A2-DA
    B1=B2-DB
    GO TO 230
430 A1=A3
    A2=A3
    B1=B3
    B2=B3
    GO TO 230
450 IF(R23.LE.0.0)   GO TO 480
    IF(MD.EQ.3)       GO TO 470
460 A5=A3
    A4=A3
    DB=2.*A3*S23/(A2*(A2+A2))
    B4=B3+DB
    B5=B4+DB
    GO TO 280
470 DA=A3-A2
    DB=B3-B2
    A4=A3+DA
    B4=B3+DB
    A5=A4+DA
    B5=B4+DB
    GO TO 280
480 A5=A3
    A4=A3
    B5=B3
    B4=B3
    GO TO 280
500 SGN=1.0
    IF(R23.LE.0.0)                           GO TO 550
    IF((R12.LE.0.0).AND.(R34.LE.0.0))        GO TO 580
    IF((R13.LE.0.0).AND.(R24.LE.0.0))        GO TO 580
    IF((R12.LE.0.0).OR. (R24.LE.0.0))        GO TO 560
    IF((R13.LE.0.0).OR. (R34.LE.0.0))        GO TO 570
    S2=S12*S24
    S3=S13*S34
    PATCH=S2*S3
    IF(PATCH.LT.0.0)  S3=-S3
    A=S2*A3*A3-S3*A2*A2
    B=S2*A3*B3-S3*A2*B2
    C=S2*B3*B3-S3*E2*B2
    D=S23*SQRT(S2*S3)
    PATCH=B*D
    IF(PATCH.LT.0.0)  D=-D
    B=B+D
    S20=A2*B0(1)-AC(1)*B2
    S03=A0(1)*B3-A3*B0(1)
    PATCH=S20*S03
    IF(PATCH.LE.0.0)  GO TO 510
    COS3=A0(1)
    SIN3=B0(1)
    GO TO 520
```

```
510 S20=A2*B0(2)-A0(2)*B2
    COS3=A0(2)
    SIN3=B0(2)
520 PATCH=S20*S23
    IF(PATCH.GT.0.0)   GO TO 590
    COS3=-COS3
    SIN3=-SIN3
    GO TO 590
550 IF(C23.LT.0.0)      SGN=-1.0
560 COS3=A2
    SIN3=B2
    GO TO 590
570 COS3=A3
    SIN3=B3
    GO TO 590
580 COS3=A2+A3
    SIN3=B2+B3
590 IF(MD.LE.2)  GO TO LBL.(240,600)
    R=SQRT(COS3*COS3+SIN3*SIN3)
    COS3=COS3/R
    SIN3=SIN3/R
    GO TO LBL. (240,600)
600 KS0=(I-2)*M+1
    Z=0.0
    IF(MD.GT.2)  GO TO 660
610 KS1=KS0
    P1=DX
    Q1=P1*SIN2/COS2
    Q2=3.*DY-2.*Q1-P1*SIN3/COS3
    Q3=DY-Q1-Q2
    DO 620  JS=1,MM1
       KS1=KS1+1
       Z=Z+DZ
       U(KS1)=P0+Z*P1
620    V(KS1)=Q0+Z*(Q1+Z*(Q2+Z*Q3))
    GO TO 290
660 KS2=KS0
    R=SQRT(DX*DX+DY*DY)
    P1=R*COS2
    P2=3.*DX-R*(2.*COS2+COS3)
    P3=DX-P1-P2
    Q1=R*SIN2
    Q2=3.*DY-R*(2.*SIN2+SIN3)
    Q3=DY-Q1-Q2
    DO 670  JS=1,MM1
       KS2=KS2+1
       Z=Z+DZ
       U(KS2)=P0+Z*(P1+Z*(P2+Z*P3))
670    V(KS2)=Q0+Z*(Q1+Z*(Q2+Z*Q3))
    GO TO 290
800 IF(MD.NE.2)  GO TO 890
    DO 810  KR=1,N
       TS=U(KR)
       U(KR)=V(KR)
       V(KR)=TS
810    CONTINUE
890 L0=LPS
    N0=N
    RETURN
    END
    SUBROUTINE SPLN2

    INTERPOLATION EY SPLINE FIT

    DIMENSION A(50),V(50)
```

```
      COMMON X(50),U(50),S(50),DEL(50),Q1,QN,NPTS
      N=NPTS
      IF(N-3) 5,5,1
1     IF(N-50) 6,6,5
6     DEL(2)=X(2)-X(1)
      V(1)=6.0*(((U(2)-U(1))/DEL(2))-Q1 )
      N1=N-1
      DO 2 I=2,N1
      DEL(I+1)=X(I+1)-X(I)
2     V(I)=((U(I-1)/DEL(I))-U(I)*((1.0/DEL(I))+(1.0/DEL(I+1)))
   C      +(U(I+1)/DEL(I+1)))*6.0
      V(N)=(QN+(U(N1)-U(N))/DEL(N) )*6.0
      A(1)=2.0*DEL(2)
      A(2)=1.5*DEL(2)+2.0*DEL(3)
      V(2)=V(2)-0.5*V(1)
      DO 3 I=3,N1
      C=DEL(I)/A(I-1)
      A(I)=2.0*(DEL(I)+DEL(I+1))-C*DEL(I)
      V(I)=V(I)-C*V(I-1)
3     CONTINUE
      C=DEL(N)/A(N1)
      A(N)=2.0*DEL(N)-C*DEL(N)
      V(N)=V(N)-C*V(N1)
      S(N)=V(N)/A(N)
      DO 4 J=1,N1
      I=N-J
4     S(I)=(V(I)-DEL(I+1)*S(I+1))/A(I)
5     RETURN
      END
      FUNCTION SPLIN(Y)
      COMMON X(50),U(50),S(50),DEL(50),Q1,QN,NPTS
      IF(Y-X(1)) 1,1,2
1     SPLIN=U(1)-Q1*(X(1)-Y)
      GO TO 5
2     DO 4 K=2,NPTS
      IF(Y-X(K)) 3,3,4
3     CONTINUE
      K1=K-1
      FF1=S(K1)*(X(K)-Y)**3
      FF2=S(K)*(Y-X(K1))**3
      FF3=1.0/(6.0*DEL(K))
      F1=(FF1+FF2)*FF3
      F2= (Y-X(K1))*((U(K)/DEL(K))-(S(K)*DEL(K))/6.0)
      F3= (X(K)-Y)*((U(K1)/DEL(K))-(S(K1)*DEL(K))/6.0)
      SPLIN=F1+F2+F3
      GO TO 5
4     CONTINUE
      SPLIN=U(NPTS)+QN*(Y-X(NPTS))
5     RETURN
      END
```

APPENDIX C. CRP ERROR MESSAGES

C.1 Namelist Input Errors

1. Value of variable xxxx not in correct range. Run terminates.

Variable	Correct Range
RTYP	0,1
ABSCA	0,1

2. Variables DELTAU and DELTAV must both be nonzero. Run terminates.

3. If linear scaling is desired, all five variables (XO, XF, DELTX, XFMT, and SUBDIV) must be read. Check to make sure XO is not equal to XF. Run terminates.

4. Warning - graph titles have not been supplied. Blanks will be substituted.

C.2 Interpolation Errors

1. An error condition has been discovered in INTERP. The error return code is X, where

   1 = INT less than 1 or greater than M

   2 = An index in IGP(I) outside index range of the first through last data points of curve N.

2. The value X1(or X2) xxx.xx does not fall on any line of constant X2 (or X1).

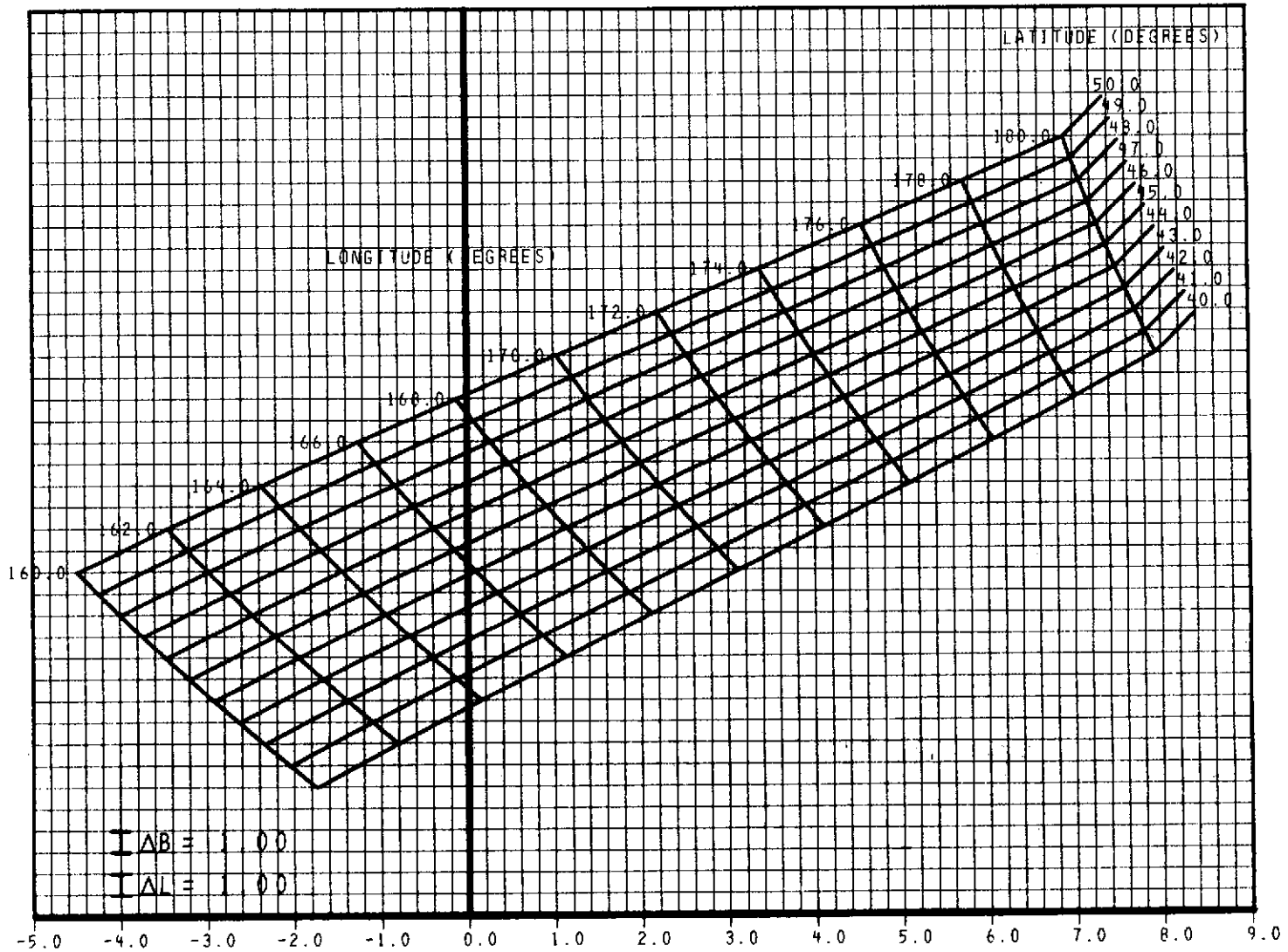3. The number of interpolated lines changed to NNN. The new array is as follows:

REFERENCES

Adams, D. P., Nomography. Archon Books, Manden, Connecticut, 1964.

Akima, H., "A Model of Smooth Curve Fitting," ESSA Technical Report, ERL 101-ITS73, January 1969.

Teague, M. J., and J. I. Vette, "The Use of the Inner Zone Electron Model AE-5 and Associated Computer Programs," NSSDC 72-11, November 1972.

Teague, M. J., and J. I. Vette, "A Model of the Trapped Electron Population at Solar Minimum," NSSDC 74-03, April 1974.

Thompson, R. F., "Spline Interpolation on a Digital Computer," NASA X-692-70-261, July 1970.

Preceding page blank

DECLINATION(EAST POSITIVE) AT SURFACE OF   FIG.1
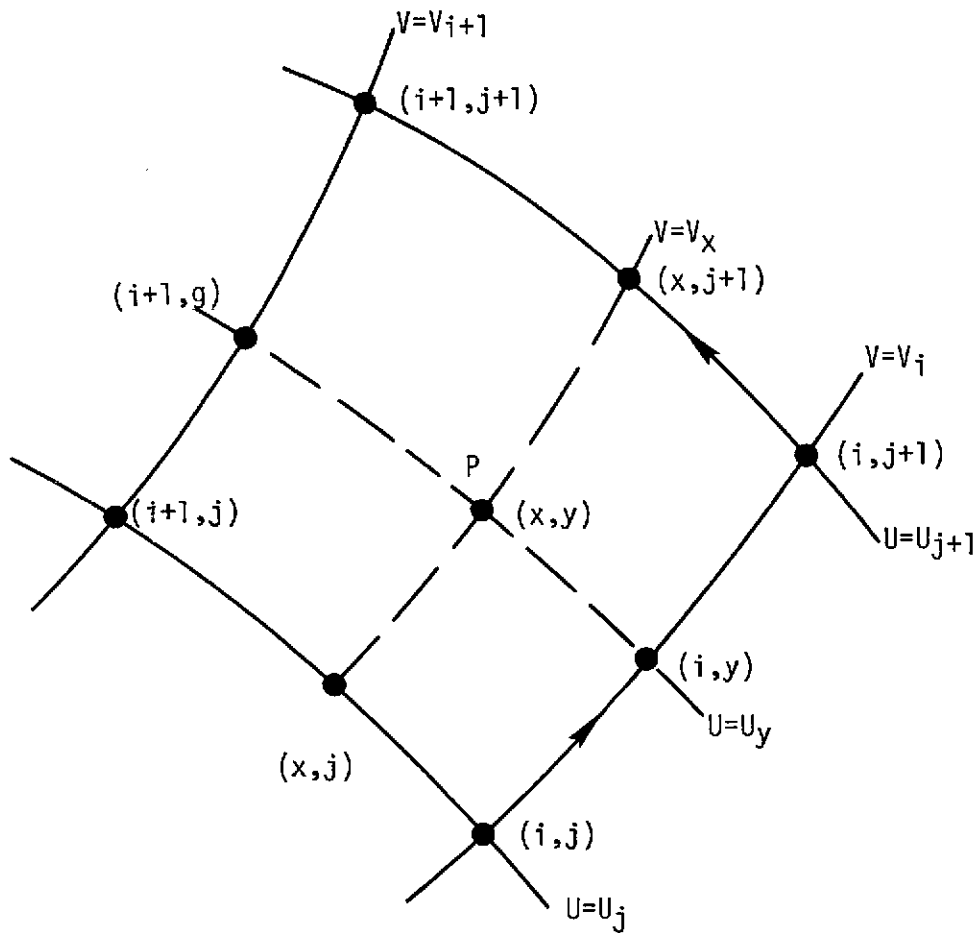EARTH VERSUS GEODETIC LATITUDE AND LONGITUDE
IGRF 1965 MODEL, EPOCH 1970

Figure 2. The Carpet Plot Grid
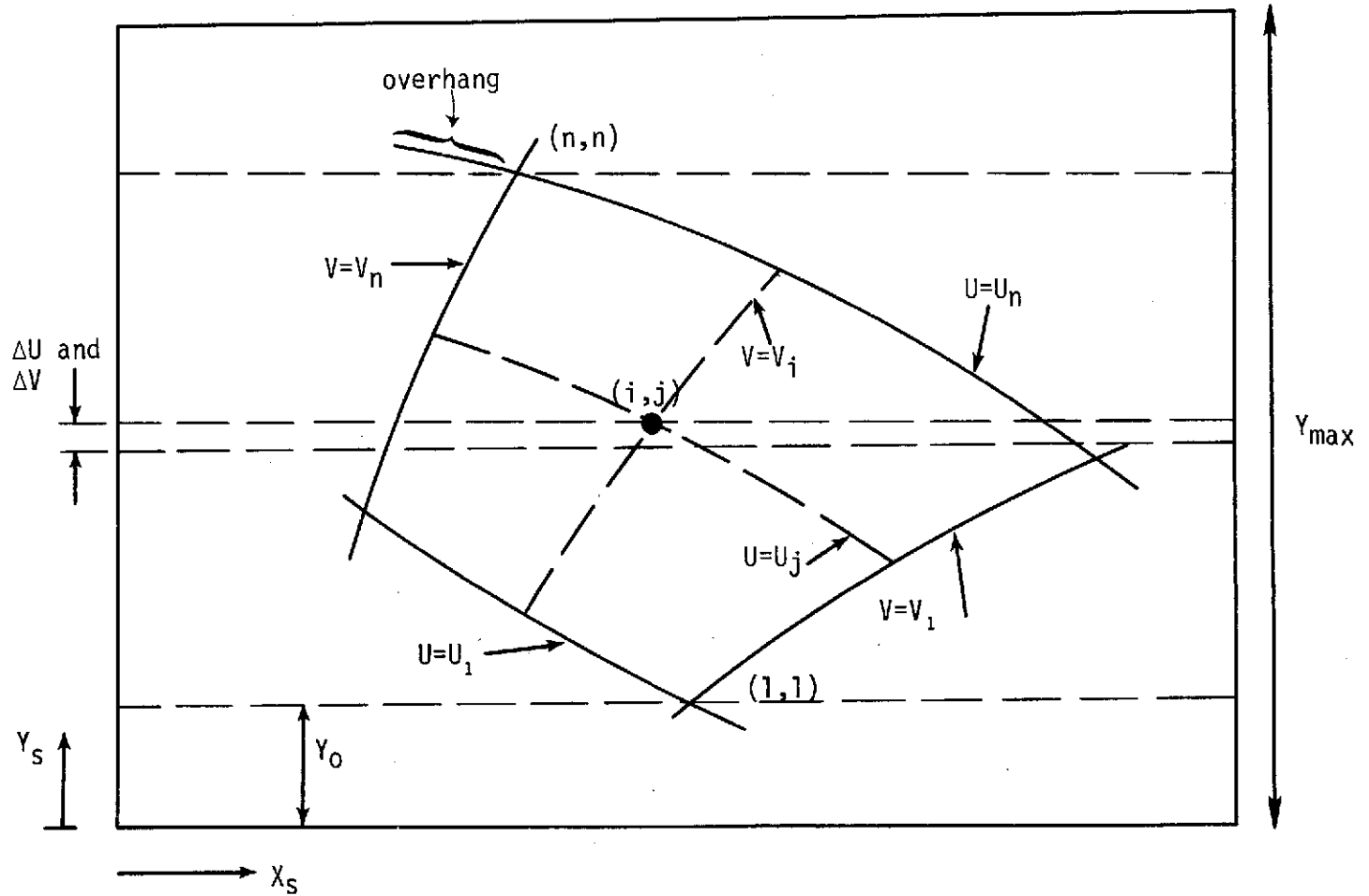
Figure 3. The Carpet Plot Coordinate System

etc.

Member Data Cards

$END

$HEAD

2nd Carpet Plot

Member Data Cards
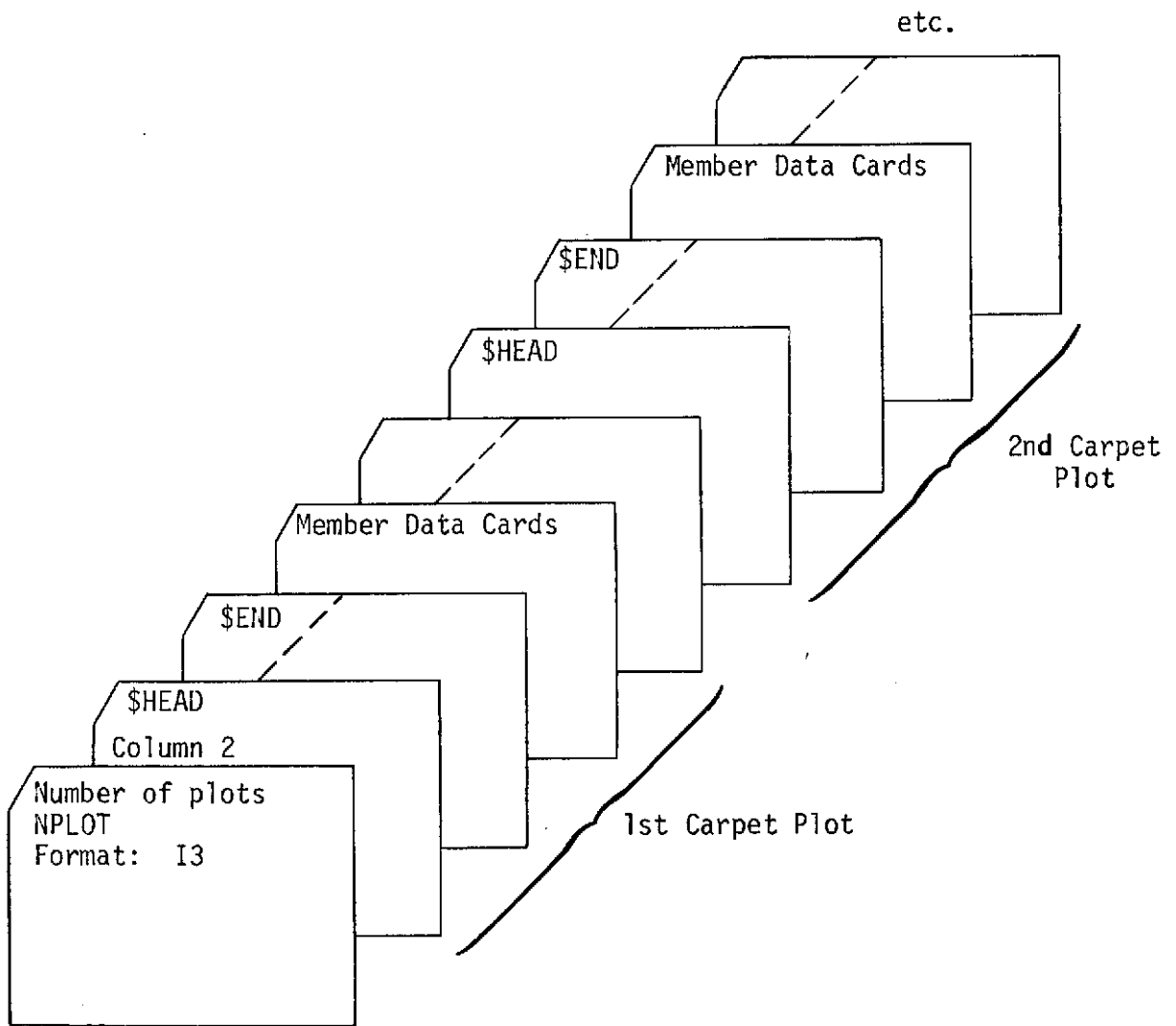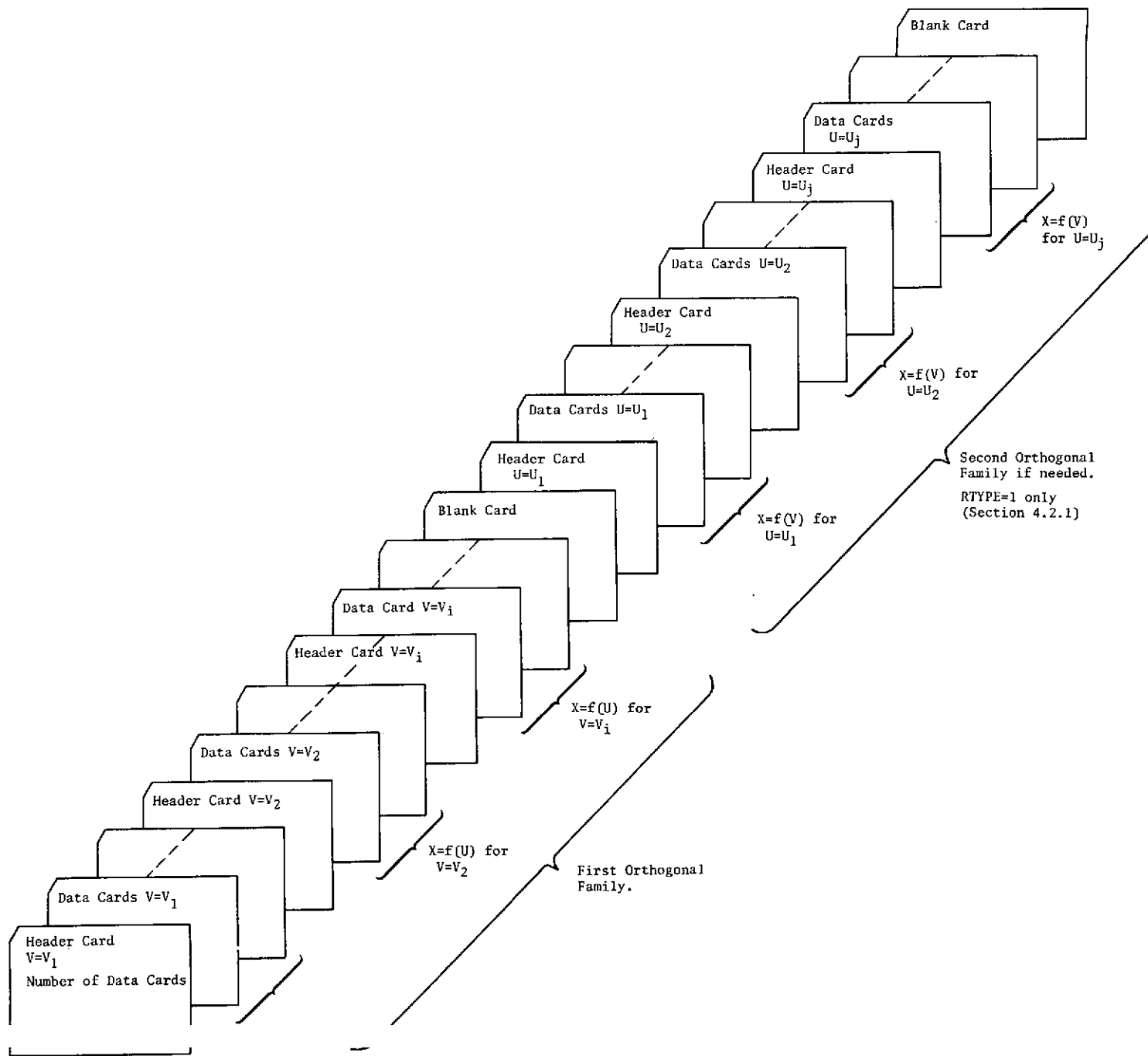
$END

$HEAD

Column 2

Number of plots
NPLOT
Format: I3

1st Carpet Plot

Figure 4. Data Deck Setup

Blank Card

Data Cards
U=U_j

Header Card
U=U_j

X=f(V)
for U=U_j

Data Cards U=U_2

Header Card
U=U_2

X=f(V) for
U=U_2

Data Cards U=U_1

Header Card
U=U_1

Blank Card

X=f(V) for
U=U_1

Second Orthogonal
Family if needed.

RTYPE=1 only
(Section 4.2.1)

Data Card V=V_i

Header Card V=V_i

X=f(U) for
V=V_i

Data Cards V=V_2

Header Card V=V_2

X=f(U) for
V=V_2

First Orthogonal
Family.

Data Cards V=V_1

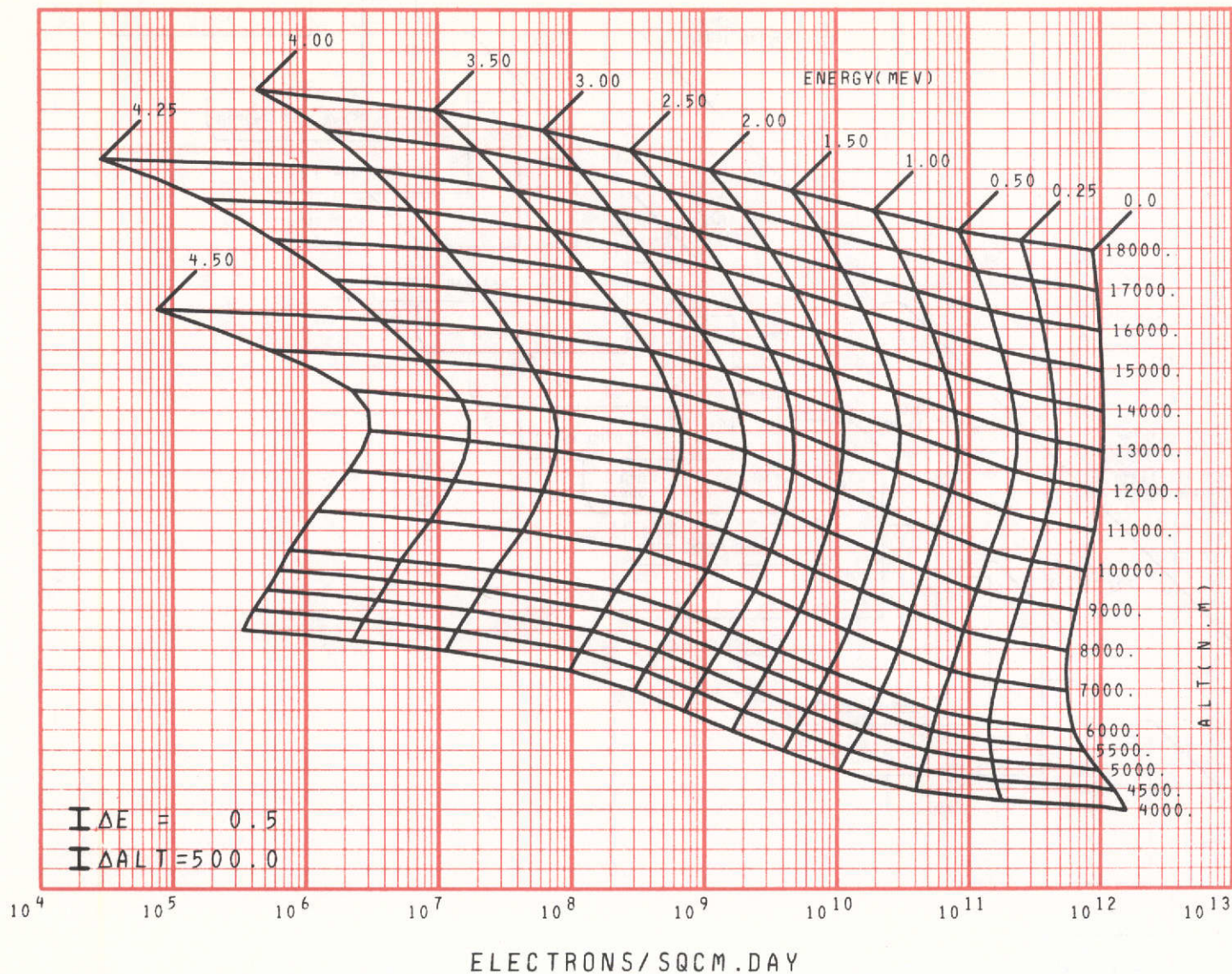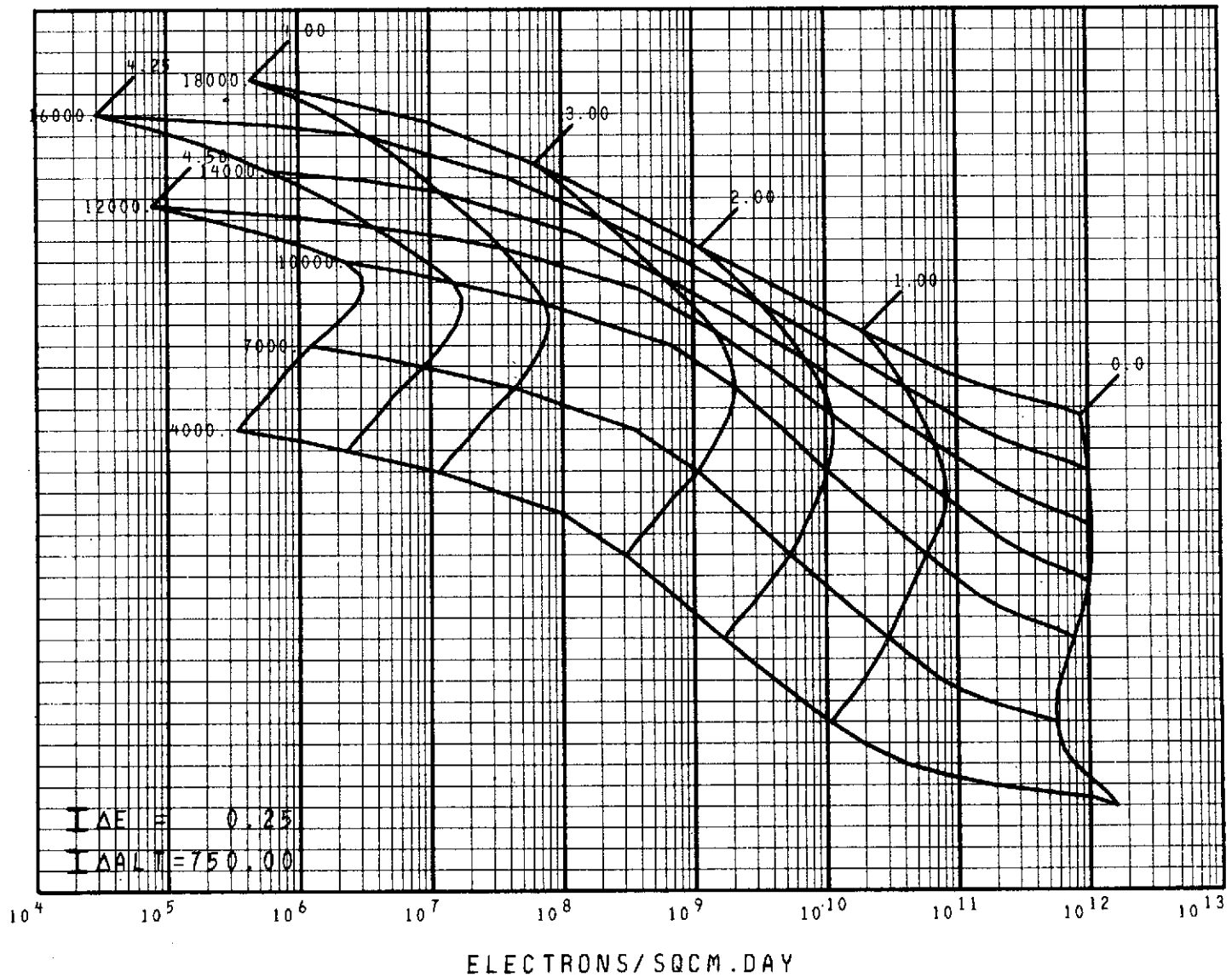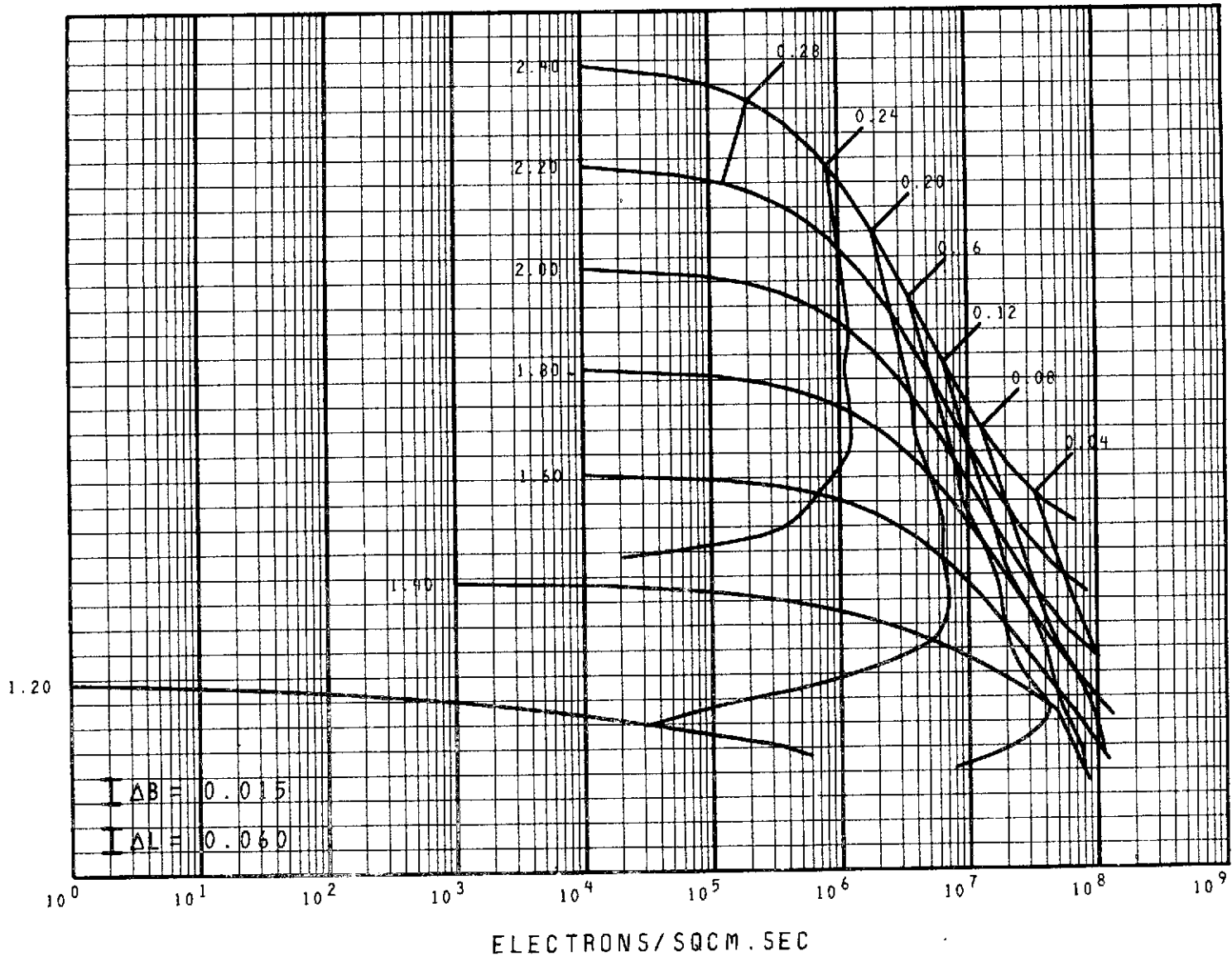Header Card
V=V_1
Number of Data Cards

73

ORBITAL INTEGRATION MAP EPOCH 1975   FIG.6
CIRCULAR ORBIT 60 DEG INCLINATION
SOLAR MINIMUM AE4 AND AE5 MODELS

ELECTRONS/SQCM.DAY

TEST OF CARPET MATRIX FOR 60 DEG.   FIGURE 7.

ELECTRONS/SQCM.DAY

TEST OF CARPET MATRIX FOR 40 KEV.　　　FIGURE 8.

ELECTRONS/SQCM.SEC

76

TEST OF CARPET MATRIX FOR 40 KEV.   FIGURE 9.

ELECTRONS/SQCM.SEC

77

TEST OF CARPET MATRIX FOR 40 KEV.   FIGURE 10.

ELECTRONS/SQCM.SEC

78

TEST OF CARPET MATRIX FOR 100 KEV.CRVFIT   FIGURE 11.

ELECTRONS/SQCM.SEC

79

TEST OF CARPET MATRIX FOR 100 KEV.SPLINE   FIGURE 12.

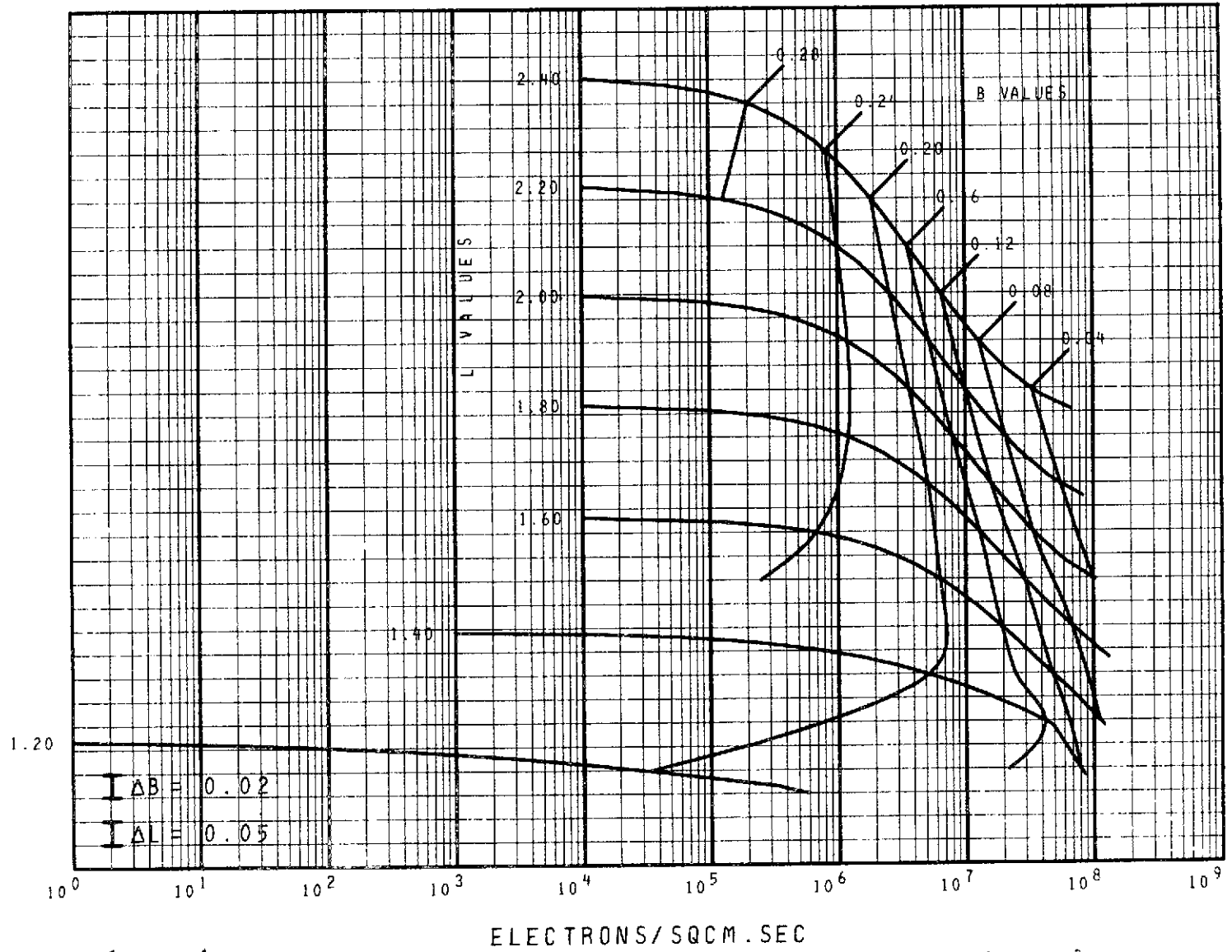ELECTRONS/SQCM.SEC

TEST OF CARPET MATRIX FOR 100 KEV.SPLINE   FIGURE 13.
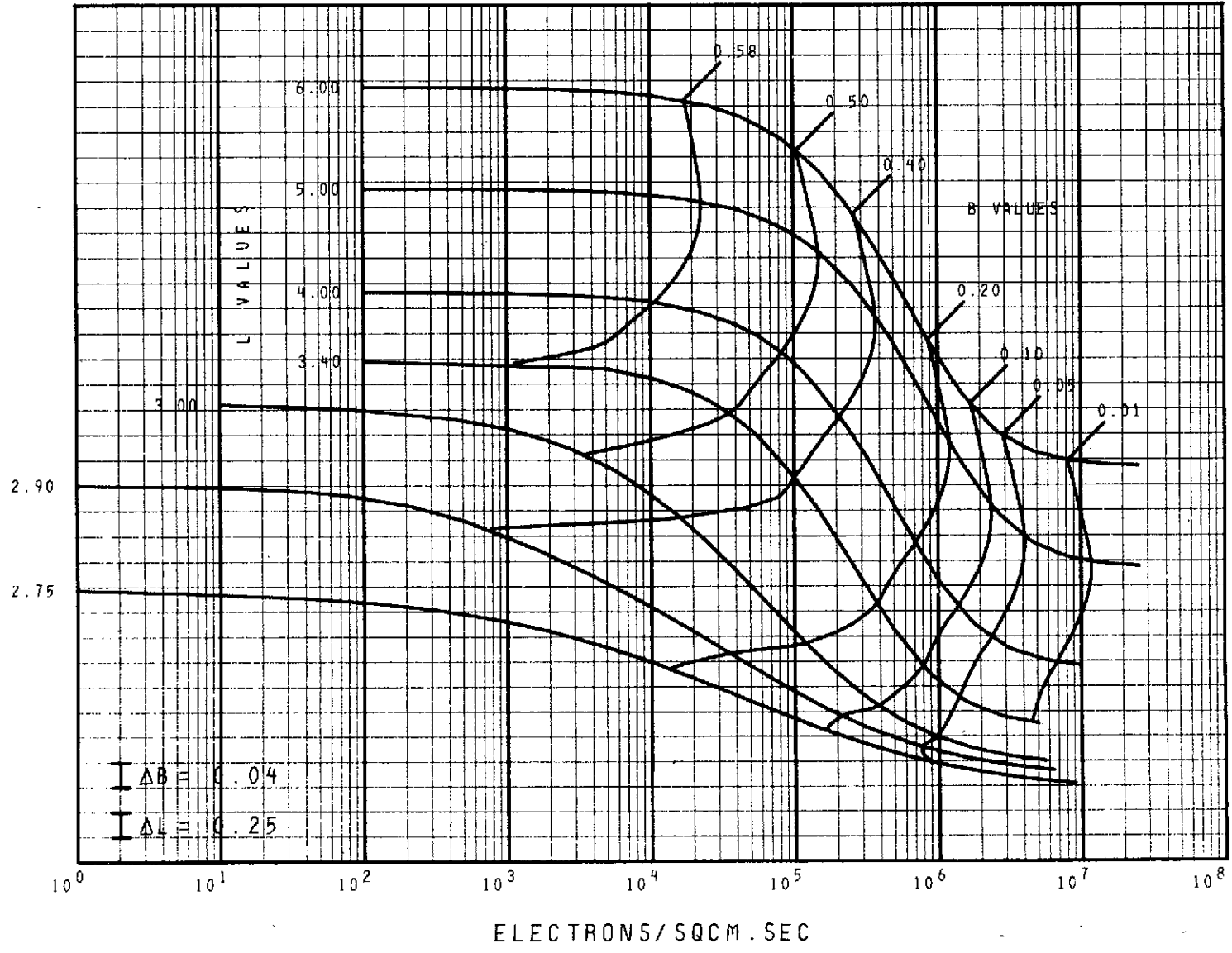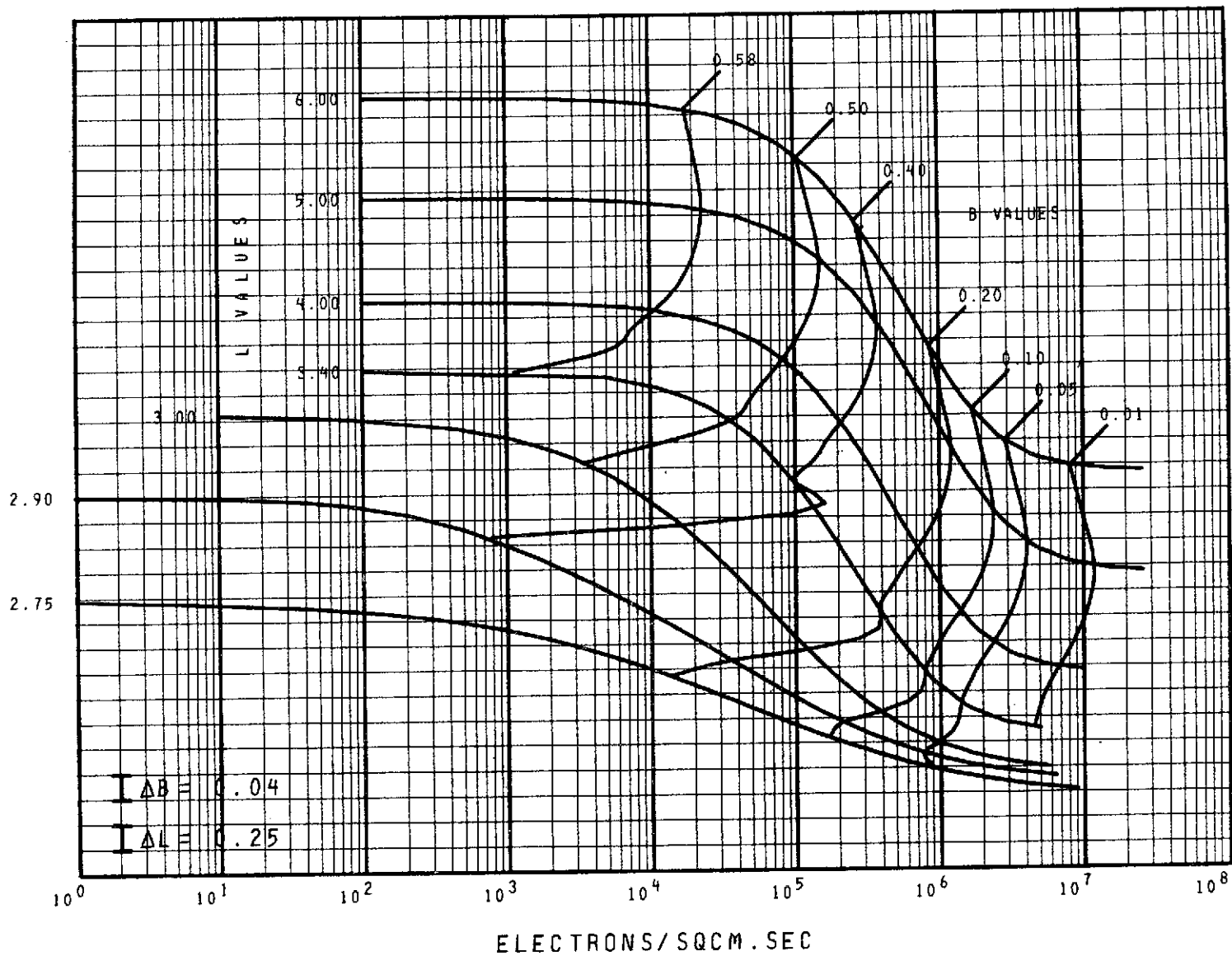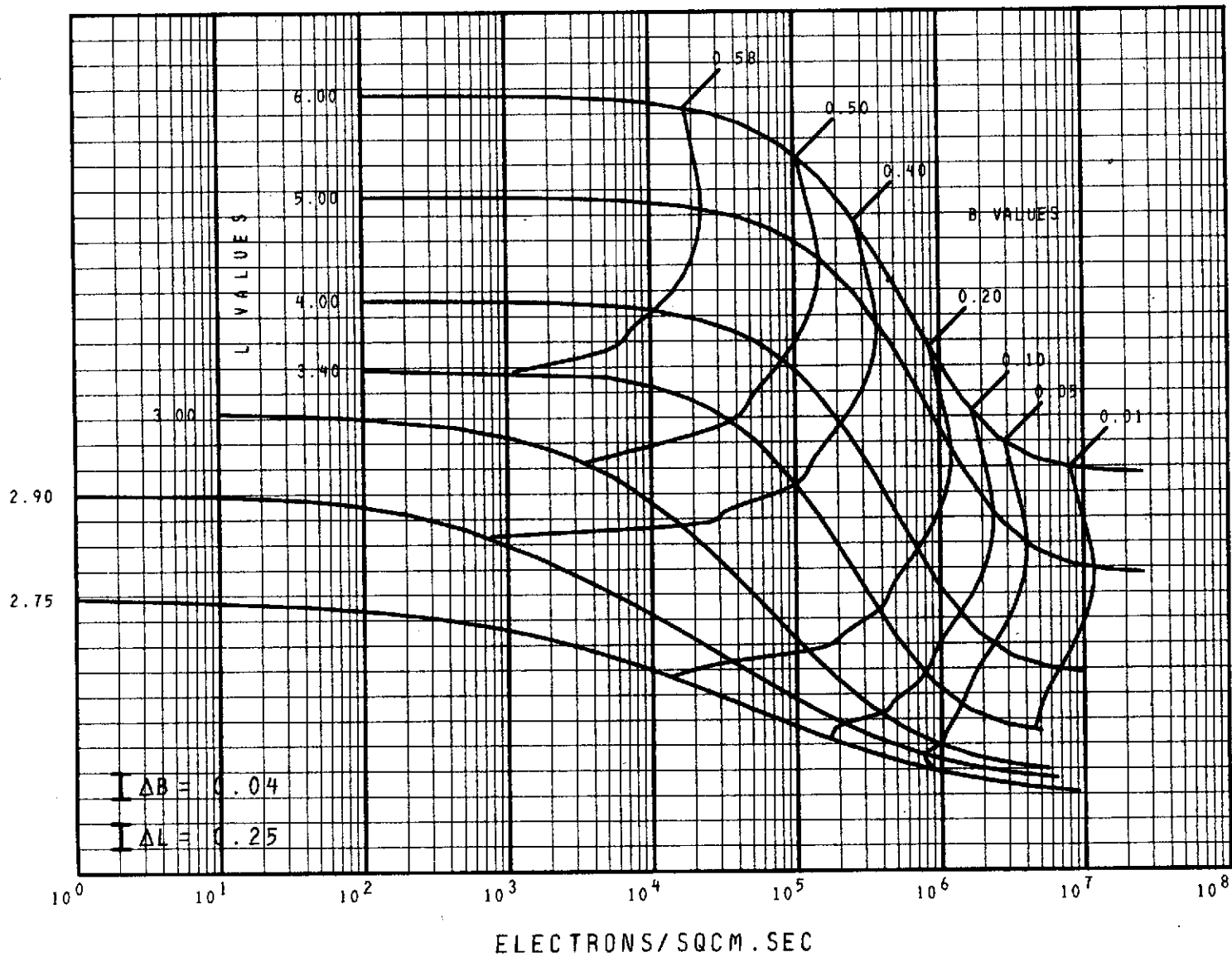

ELECTRONS/SQCM.SEC

```
// Job Card
// EXEC FORTRANH
//SOURCE.SYSIN DD *
                Source Decks
// EXEC LINKGO,REGION=170K
//LINK.SYSLIB DD DSNAME=SYS2.SD4060,DISP=SHR
//GO.SC4060ZZ DD DSN=job....id,UNIT=7TRACK,
//            LABEL=(1,BLP),DISP=(NEW,KEEP),
//            DCB=(DEN=1,TRTCH=C,RECFM=F,BLKSIZE=1024),
//            VOL=SER=xxxx†
//GO.SYSUDUMP DD SYSOUT=D
//GO.DATA5 DD *
                Data Deck
/*
//
```

† Tape Number.

Figure 14.   JCL Required with Source Decks

```
// Job Card
// EXEC LINKGO,REGION=170K
//LINK.SYSLIB DD DSNAME=SYS2.SD4060,DISP=SHR
//LINK.SYSLIN DD *
               Object Decks
//GO.SC4060ZZ DD DSN=job...id,UNIT=7TRACK,
//             LABEL=(1,BLP), DISP=(NEW,KEEP),
//             DCB=(DEN=1,TRTCH=C,RECFM=F,BLKSIZE=1024),
//             VOL=SER=xxxx†
//GO.SYSUDUMP DD SYSOUT=D
//GO.DATA5 DD *
               Data Deck
/*
//
```

† Tape Number

Figure 15.   JCL Required with Object Decks

LINES OF CONSTANT LONG

LONG = 0.0

DEP VAR	Y	LAT	DEP VAR	Y	LAT
3.23000E-01	10.0	1.00000E 01	3.25000E-01	10.2	1.10000E 01
3.27000E-01	10.4	1.20000E 01	3.30000E-01	10.6	1.30000E 01
3.33000E-01	10.8	1.40000E 01	3.36000E-01	11.0	1.50000E 01
3.39000E-01	11.2	1.60000E 01	3.43000E-01	11.4	1.70000E 01
3.47000E-01	11.6	1.80000E 01	3.50000E-01	11.8	1.90000E 01
3.54000E-01	12.0	2.00000E 01	3.58000E-01	12.2	2.10000E 01
3.63000E-01	12.4	2.20000E 01	3.67000E-01	12.6	2.30000E 01
3.71000E-01	12.8	2.40000E 01	3.76000E-01	13.0	2.50000E 01
3.80000E-01	13.2	2.60000E 01	3.85000E-01	13.4	2.70000E 01
3.89000E-01	13.6	2.80000E 01	3.94000E-01	13.8	2.90000E 01
3.98000E-01	14.0	3.00000E 01	4.03000E-01	14.2	3.10000E 01
4.07000E-01	14.4	3.20000E 01	4.12000E-01	14.6	3.30000E 01
4.16000E-01	14.8	3.40000E 01	4.20000E-01	15.0	3.50000E 01
4.25000E-01	15.2	3.60000E 01	4.29000E-01	15.4	3.70000E 01
4.33000E-01	15.6	3.80000E 01	4.37000E-01	15.8	3.90000E 01
4.41000E-01	16.0	4.00000E 01	4.44000E-01	16.2	4.10000E 01
4.48000E-01	16.4	4.20000E 01	4.52000E-01	16.6	4.30000E 01
4.55000E-01	16.8	4.40000E 01	4.58000E-01	17.0	4.50000E 01
4.62000E-01	17.2	4.60000E 01	4.65000E-01	17.4	4.70000E 01
4.68000E-01	17.6	4.80000E 01	4.71000E-01	17.8	4.90000E 01
4.74000E-01	18.0	5.00000E 01	9.99000E 02	999.0	9.99000E 02

LONG = 40.000

DEP VAR	Y	LAT	DEP VAR	Y	LAT
3.62000E-01	18.0	1.00000E 01	3.65000E-01	18.2	1.10000E 01
3.67000E-01	18.4	1.20000E 01	3.70000E-01	18.6	1.30000E 01
3.73000E-01	18.8	1.40000E 01	3.76000E-01	19.0	1.50000E 01
3.80000E-01	19.2	1.60000E 01	3.83000E-01	19.4	1.70000E 01
3.87000E-01	19.6	1.80000E 01	3.91000E-01	19.8	1.90000E 01
3.95000E-01	20.0	2.00000E 01	3.99000E-01	20.2	2.10000E 01
4.03000E-01	20.4	2.20000E 01	4.07000E-01	20.6	2.30000E 01
4.12000E-01	20.8	2.40000E 01	4.16000E-01	21.0	2.50000E 01
4.20000E-01	21.2	2.60000E 01	4.24000E-01	21.4	2.70000E 01
4.28000E-01	21.6	2.80000E 01	4.33000E-01	21.8	2.90000E 01
4.37000E-01	22.0	3.00000E 01	4.41000E-01	22.2	3.10000E 01
4.45000E-01	22.4	3.20000E 01	4.49000E-01	22.6	3.30000E 01
4.53000E-01	22.8	3.40000E 01	4.56000E-01	23.0	3.50000E 01
4.60000E-01	23.2	3.60000E 01	4.64000E-01	23.4	3.70000E 01
4.67000E-01	23.6	3.80000E 01	4.70000E-01	23.8	3.90000E 01
4.74000E-01	24.0	4.00000E 01	4.77000E-01	24.2	4.10000E 01
4.80000E-01	24.4	4.20000E 01	4.83000E-01	24.6	4.30000E 01
4.86000E-01	24.8	4.40000E 01	4.89000E-01	25.0	4.50000E 01
4.92000E-01	25.2	4.60000E 01	4.95000E-01	25.4	4.70000E 01
4.98000E-01	25.6	4.80000E 01	5.00000E-01	25.8	4.90000E 01
5.03000E-01	26.0	5.00000E 01	9.99000E 02	999.0	9.99000E 02

LONG = 80.000

DEP VAR	Y	LAT	DEP VAR	Y	LAT
4.05000E-01	26.0	1.00000E 01	4.07000E-01	26.2	1.10000E 01
4.09000E-01	26.4	1.20000E 01	4.11000E-01	26.6	1.30000E 01
4.14000E-01	26.8	1.40000E 01	4.17000E-01	27.0	1.50000E 01
4.20000E-01	27.2	1.60000E 01	4.24000E-01	27.4	1.70000E 01
4.28000E-01	27.6	1.80000E 01	4.32000E-01	27.8	1.90000E 01
4.36000E-01	28.0	2.00000E 01	4.40000E-01	28.2	2.10000E 01
4.45000E-01	28.4	2.20000E 01	4.50000E-01	28.6	2.30000E 01
4.55000E-01	28.8	2.40000E 01	4.60000E-01	29.0	2.50000E 01
4.66000E-01	29.2	2.60000E 01	4.71000E-01	29.4	2.70000E 01

Figure 16. Sample Line Printer Output

Table 1. Format of Header and Data Cards

Card	Column	Format	Name	Description
Header	1-67	67X		Available for Comments
	68-69	I2	N	Number of Data Cards to follow
	70-74	5X		Blanks
	75-80	F6.m		Value of constant independent variable for following member
Data	1-8	F8.5		Independent variable $\Big\}$ Pair[†] $4(n-1)+1$
	9	1X		Blank
	10-19	E10.3		Dependent variable
	20-27	F8.5		Independent variable $\Big\}$ Pair $4(n-1)+2$
	28	1X		Blank
	29-38	E10.3		Dependent variable
	39-46	F8.5		Independent variable $\Big\}$ Pair $4(n-1)+3$
	47	1X		Blank
	48-57	E10.3		Dependent variable
	58-65	F8.5		Independent variable $\Big\}$ Pair $4(n-1)+4$
	66	1X		Blank
	67-76	E10.3		Dependent variable
	77-80	4X		Blanks

[†] n is the number of the data card $1 \leq n \leq N$

Table 2.   Alphabetic List of NAMELIST Parameters
Described in Appendix A

Parameter No.	Parameter	Parameter No.	Parameter
16	ABSCA	33	THETA1
45	CHFMT	38	THETA2
10	CHOICE	21	TITLE1
43	DCHAR	23	TITLE2
4	DELTAU	25	TITLE3
5	DELTAV	39	ULABEL
19	DELTX	11	UMAX
1	IPR	12	UMIN
15	KOLOR	8	UPLOT
29	LABEL1	40	VLABEL
34	LABEL2	13	VMAX
44	NCHAR	14	VMIN
30	NLAB1	6	VPLOT
35	NLAB2	18	XF
9	NUPLOT	46	XFMT
7	NVPLOT	17	XO
28	NX	31	XPOS1
22	N1	36	XPOS2
24	N2	41	XTICK
26	N3	27	XTITLE
3	PO	32	YPOS1
2	RTYPE	37	YPOS2
20	SUBDIV	42	YTICK