

SH
11
.A2
S662
no.
87-25
c.2

SOUTHWEST FISHERIES CENTER

NATIONAL MARINE FISHERIES SERVICE

SOUTHWEST FISHERIES CENTER

P.O. BOX 271

LA JOLLA, CA 92038

DECEMBER 1987

ENVIRONMENTAL DATA MAPPING PROGRAM - EDMAP

By

L. E. Eber

ADMINISTRATIVE REPORT LJ-87-25



This Administrative Report is issued as an informal document to ensure prompt dissemination of preliminary results, interim reports and special studies. We recommend that it not be abstracted or cited.

ENVIRONMENTAL DATA MAPPING PROGRAM - EDMAP

L. E. Eber

Coastal Fisheries Resources Division

Southwest Fisheries Science Center

National Marine Fisheries Service

National Oceanic and Atmospheric Administration

P.O. Box 271

La Jolla, California 92038

ADMINISTRATIVE REPORT LJ-87-25

Revised July 1992

SH
11
.A2
5662
no, 87-25
c. 2

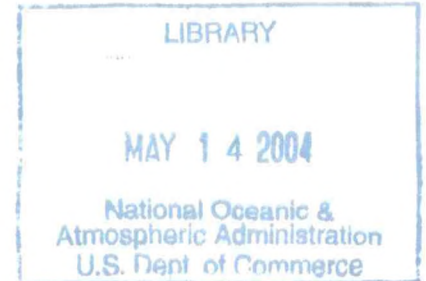


TABLE OF CONTENTS

INTRODUCTION.....	1
RUNEDMAP.....	2
PROGRAM PARAMETERS.....	7
Option Parameters.....	7
Specification Parameters.....	17
Filename Parameters.....	25
GRIDFIELD MODIFICATION WITH EDITFLD.....	26
CREATING OVERLAY PLOT FILES WITH MAKOVRLA.....	27
FITTING RANDOMLY DISTRIBUTED DATA TO A GRIDFIELD.....	33
TRACKING CONTOURS THROUGH A GRIDFIELD.....	36
APPENDIX A. Summary of option and specification parameters.....	38
APPENDIX B. Plot calls and parameters - CALLPSUB.....	42
APPENDIX C. Examples of plots and command files.....	45

INTRODUCTION

EDMAP is a self contained program for making contour plots of two-dimensional scalar fields, defined at the grid points, or nodes, of a rectangular grid. It can also make digital plots, showing the values of a field at the grid points or at the actual locations of the data points. It can display wind fields by plotting arrows at grid points. Special features, such as coastlines, scales and legends, can be superimposed on a plot.

To use EDMAP, the data to be plotted as contours or digital values must be submitted in a suitably formatted disk file. The user must provide EDMAP with information about the range of the data values, their coordinates and about the format of the data file. Since EDMAP is not interactive, this information, along with other specifications and parameters, must be submitted in another disk file. An interactive, menu driven program, RUNEDMAP, is provided to assist the user in selecting parameters and saving them in a command file to run EDMAP. RUNEDMAP is described in the next section.

The primary functions of EDMAP are listed below.

1. Load data values from a source file provided by the user.
2. Analyze the data to define a gridfield at the nodes, or grid points, of a rectangular grid.
3. Modify or combine gridfields created during separate runs.
4. Digitize contours on a gridfield and create a file of contour coordinates.
5. Organize the parameters and elements needed to make a plot and create a file of generic plot calls.

Other programs are provided to translate the plot calls to coded instructions for specific plotters or a screen display. They will be described in a later section.

RUNEDMAP

RUNEDMAP is a program to facilitate the selection and organization of parameters for EDMAP in a menu driven, interactive environment. Run parameters selected in this way are saved in command files. A command file is the interface between the user and EDMAP and contains all of the parameters it needs at run time. RUNEDMAP serves as a shell from which the user can create command files and submit them to EDMAP for execution.

RUNEDMAP comes up in the Plot Selection Menu. Here the user can choose one or more of the primary plot options. Secondary plot options and specification of other plot parameters will be made in the Run Parameters Menu. When the options are selected, they can be saved in a command file. In the Action Menu the user can view and edit the command file, execute EDMAP and display the plot on the screen.

Figure 1

PLOT SELECTION MENU

Move cursor to the desired line and press <ENTER> .

- _ Contour Plot (horizontal map or vertical section)
- _ XY Data Plot (data values or symbols at data locations)
- _ Grid Data Plot (data values at grid point locations)
- _ Vector Plot (vector arrows at grid point locations)
- _ Overlay Plot (superimposed coastlines, legends, etc)
- _ Concatenate (links added data to one of the above plots)
- _ Change to Run Parameters menu
- _ Change to Action menu

Press <ESC> to exit program, <PGUP> to set Run Parameters or <PGDN> to run EDMAP (Action Menu)

The plot selection menu is shown in Figure 1. The first five items are the primary plot options. One or more of these options must be selected to make a plot. The only restriction is that grid data and vector data cannot be plotted together because they use the same internal file variable. The concatenate option must be used in conjunction with one of the five primary plot options.

After a plot option is selected, the cursor will move automatically to the 'Change to RunParameters menu' item. However, the user can move the cursor up or down to any position with the arrow keys. The last two menu items are exit options. The <ESC>, <PGUP> and <PGDN> keys provide alternate means of exit from the menu.

The run parameters menu is shown in Figure 2. Each item deals with a group of associated parameters. Some parameters, including those specifying grid coordinates and dimensions, are mandatory. Others can be selected, as desired, by moving the cursor to the appropriate line with the up and down arrows. The selection of each menu item brings up a new screen with prompts for parameters relevant to that item. When the requisite responses for an item have all been entered, the run parameters menu will reappear.

The first five items in this menu may be selected repeatedly, if necessary, to change or correct previous entries. The sixth item (Exit menu. Create file...) will cause RUNEDMAP to create a command file containing the selected parameters and to bring up the action menu.

Figure 2

RUN PARAMETERS MENU

Move cursor to the desired line and press <ENTER>.

- _ Select the Plot coordinates.
- _ Specify the Grid dimensions.
- _ Identify the source (data or overlay) file(s).
- _ Set specifications for the Plot (tic marks, labels, etc).
- _ Select Gridfield options (save or load the gridfield, smooth, print, modify, etc.).
- _ Exit menu. Create the file of run parameters.
- _ Go to Action menu. Create and/or display a plot.
- _ Return to Plot Selection menu.

Press <ESC> to exit program, or <PGDN> for Action Menu.

The action menu, Figure 3, can be brought up at any time from either of the other menus, by selecting the appropriate menu item, or by pressing the page down key, <PGDN>. It is also possible to change back to the main menu from the action menu, however, this is equivalent to restarting the program, since all program variables and parameters will be reset to their default

condition.

Figure 3

ACTION MENU

Move cursor to desired line and press <ENTER>

- _ Run EDMAP and create a plot file
 - _ Run DSPLPLOT and display a plot on the screen
 - _ View or edit a file of run parameters
 - _ Execute a DOS command and return to this menu
 - _ Exit to plot selection menu. Reset run parameters.
 - _ Exit to DOS
-

EDMAP can be executed by selecting the first item. If a command file was created or viewed or edited in the current session of RUNEDMAP, its name will be known to EDMAP automatically. Otherwise, the user will be prompted to enter the name of the command file. When EDMAP has been executed successfully, the screen will display a message stating the name of the plot file (the primary output). The program will then return to the action menu. If the message does not appear, something has gone wrong. In that case, the user can examine the logfile that is automatically created during every run and will be found in RUNEDMAP directory with the name 'EDMAP.LOG'. This file will contain most of the messages that appeared on the screen during the run and may reveal the cause of the problem. EDMAP detects some run time errors but may not suspend execution, and error messages can scroll off the screen, unseen. Run time errors are usually caused by incorrect parameter values in the command file.

Plots created by EDMAP and saved in plot files can be viewed with DSPLPLOT by selecting the second item in the action menu. This program will prompt the user for the name of the plot file. Plot file names always have the extension '.PLT', and are displayed on the screen at the completion of each successful run of EDMAP. DSPLPLOT has an option to color the areas between contours on a contour plot. A prerequisite for this option is saving the gridfield to a file during the run in which the plot file was created. The saved file must have the same stem name as the plot file and the extension '.GRD'. The parameters needed to save the file can be entered from the 'Set Gridfield options etc.' item in the Run Parameters Menu, or directly into the command file with EDLIN or any text editor. You can do this by inserting the command option 'SAVEF2' just below the ANALYZE command and by inserting a filename parameter 'FIELD2={stem name}.GRD' in the command file with the other filename parameters. Another prerequisite is that every contour in the plot must terminate at a border line or at another plot feature or close on itself. Otherwise the color will bleed from one area into another.

The third item in the action menu allows the user to view and edit his command file. It is good practice to view a newly created command file before submitting it to EDMAP. The editing option is provided by the DOS program, EDLIN, a primitive but convenient line editor. It provides the capability to make minor corrections in a command file without leaving the RUNEDMAP environment.

The fourth item is an option to execute a DOS command without leaving RUNEDMAP. The user can view a directory with <DIR> to check on a filename or invoke a different editor to change a command file. When execution of the command is completed, control will return to the action menu.

The last two items provide exits from the action menu. A return to the main menu is equivalent to an exit to DOS and then rerunning RUNEDMAP. It is necessary to do this if the user wishes to work with a new or different command file.

Before using RUNEDMAP to create a command file, the user should be prepared to furnish information about details of the plot he/she wishes to make and about the format of the source data. EDMAP treats the plot surface as a two dimensional grid. The grid is specified by six parameters: the maximum and minimum values for each coordinate (X and Y), and the interval between grid lines along each coordinate. The grid points are located at the intersections of grid lines. During execution, values of the field at grid points are computed by interpolation from the source data. The resolution of features revealed by plotting contours will depend on both the grid interval and the distribution of data points within the grid. Maximum resolution results when the grid intervals are equal to the smallest distances between data points.

EDMAP designates the lower left corner of the grid as the grid origin, with coordinates (I,J) = (1,1). Grid coordinates increase through the grid to maximum values, (I,J) = (IMX,JMX), at the upper right corner. The user's (X,Y) coordinates can increase in either the same, or opposite, sense as the grid coordinates. The program will transform user coordinates to grid coordinates correctly, in either case.

The standard format for source data submitted to EDMAP is a text file (ASCII), with record lengths of 128 characters or less, and a constant number of fields per record. The file may contain header records which, in this context, are records that contain information common to a group of data records. The first field of each header record must contain an identical string constant, at least eight characters in length. Each header record must precede the group of data records with which it is associated.

The format of data records may be either fixed-field or free-field. In fixed-field format, each data field occupies the same range of columns (character positions) in every record, but field separators are not required. In free-field format, the data fields do not necessarily occupy the same respective positions in every record, but must be separated from adjacent fields by one or more spaces.

EDMAP also supports two special formats for source data. One is a binary format for CalCOFI data in the SWFSC oceanographic data base. The other is an ASCII format for data from spreadsheet programs (or other sources) with row and column coordinates implied from the

locations of data values in a file.

Optimum selection of run parameters, particularly for features such as tic marks, labels, contour selection, etc. may require experimentation. With DSPLPLOT and EDLIN, a plot file can be displayed on the screen, parameter values changed and the command file resubmitted to EDMAP to create a new plot file, without leaving the action menu. DSPLPLOT, however, has shortcomings as a plot preview tool. Characters and numbers, for example, are always the size of the text character box in the current graphics screen mode. The specification of character height in the command file has no effect, consequently strings in may not be properly located. Also, rotation of character strings, which can be specified as an option for overlay plots, is not supported by DSPLPLOT. Distortion can also be a problem. DSPLPLOT overrides plot scale parameters in the command file to make the plot as large as possible on the screen, approximately 8 inches wide by 6 inches high.

Hard copies of plots created by EDMAP can be obtained with two other programs, HPDRIVER and WPDRIVER. These programs both convert plot calls from an EDMAP plot file into HP-GL graphics code used by Hewlett Packard 7470A pen plotters. HPDRIVER sends the code directly to an HP plotter through a serial port. WPDRIVER writes the code to a disk file which can be loaded by Word Perfect 5.1 in graphics mode and sent to a printer. These program are included with the RUNEDMAP software but they cannot be run from the RUNEDMAP environment.

RUNEDMAP builds a command file of run parameters for EDMAP based on the user's menu selections and on responses to prompts and questions. The run parameters are described in the following sections. Reference to these sections may be useful to users who wish to know how the parameters in their command files contribute to their plots, and how to change the parameters to modify a plot.

PROGRAM PARAMETERS

EDMAP is not an interactive program. It performs under the control of parameters assembled and saved in disk files prior to submission at run time. EDMAP can be executed from the menu driven program, RUNEDMAP or from the program EDMAP.EXE. Both programs prompt the user for the name of a command file of run parameters. Run parameters in command files are organized in sets. Each set consists of a plot title (optional) and one or more command options followed by specifications and filename parameters (in that order). The structure of a set is outlined below.

```
Title string (1 to 30 char) or <blank> or #
Option parameters (commands)
#
Specification parameters
#
File name parameters
#
QUIT
```

A command file may contain multiple sets of parameters. In the outline above, additional sets would be placed between the last <#> and the keyword QUIT. Each set initiates one run. If a title is included, it will be plotted above the top border. The title can be suppressed on the plot by entering a blank or the symbol <#>, but the title string must be present in the command file. The <#> symbol is used as a delimiter between groups of parameters, and must be inserted in the first position of a new line after each group. The last set of parameters in a command file must be followed by the keyword <QUIT>.

The order in which certain command options are listed in a command file may affect program flow. The order in which specification or filename parameters are listed has no effect. The EDMAP flowchart (Figure 4) shows the sequence in which the option parameters are processed. Reference to the flowchart can be especially useful when using LOADF_ or SAVEF_ options to retrieve or save grid files.

OPTION PARAMETERS

Option parameters consist of primary commands and command qualifiers. Command qualifiers modify or channel the actions of primary commands. Primary commands are generally entered on separate lines. Qualifier commands are entered on the same line as the command they modify and are separated from the primary command, and each other, by slashes </>. Some commands require additional information, which must be entered after the qualifiers and enclosed in parentheses.

YMX, XGD and YGD. The converted coordinates, along with their respective data values are stored in a binary file identified internally by the file variable, IJVAL.

Qualifier Options:

XYDAT. This is the general purpose format designator. A source file can be read by EDMAP under this option provided that: (1) it is an ASCII coded file and (2) each record contains one or more data values associated with one set of (X,Y) coordinates. The syntax of the command line is:

```
LOADSORC/XYDAT(vrbl,nflds,xloc,yloc,vloc,HDRS,FMT,selc,dfld)/  
FORMAT(field_length_list)
```

The first five parameters inside the parentheses are mandatory. The others, beginning with HDRS, and the second qualifier (FORMAT) need be entered only if applicable.

vrbl is the data identifier; it can be entered as one of four string constants, <DATA>, <CCFI>, <VECT> or <WIND>. DATA is the default entry. CCFI is used if the file format conforms to the SWFSC CalCOFI format in ASCII code. VECT and WIND are used for vector data.

nflds is the number of fields in each data record.

xloc is the sequence number of the field containing the X coordinate, counting from left to right. If the letter H (upper case) is prefixed to xloc, the X coordinate will be extracted from header records.

yloc is the sequence number of the field containing the Y coordinate. If the letter H (upper case) is prefixed to yloc, the Y coordinate will be extracted from header records. When either coordinate is obtained from a header record, that coordinate will apply to all data records encountered prior to the next header record.

vloc is the sequence number of the field containing the data value to be extracted. Data values cannot be extracted from header records.

HDRS must be entered if the file contains header records, otherwise, it should be omitted. If HDRS is entered, EDMAP expects the first record in the file to be a header record. It stores the first eight characters of the header record into a string variable and uses this to identify subsequent header records. Consequently, the first eight characters of each header record must be identical.

FMT indicates that the format of data records is fixed-field. Fixed-field means adjacent fields need not be separated by spaces, but each field must occupy the same range of positions in every record. If FMT is entered, the keyword, FORMAT, must also be entered as a second qualifier, accompanied by a field_length_list in parentheses, as described below.

selc is a numeric search key used for selective extraction of data records. When submitted, it will be compared to data values in a designated field and only those records for which a match is found will be accepted. The entry must not exceed 6 characters (including decimal point).

dflid is the sequence number of the designated field to be searched for selc.

field length list specifies the format for fixed length records. It is a list of numbers indicating the length and position of the decimal point for each field in the data records. The integer part of each format specifier indicates the field width. If the format specifier has a fractional part, the latter is multiplied by 10 to indicate the number of implied decimal places in the associated data values. If the data values have embedded periods to mark the decimal points, the fractional part of the field specifier should be omitted. The length of the list should conform to the number of fields given by nflds. A field length list applies only to data records, not to header records.

Free field format is the default mode for records in XYDAT files. In this mode, data fields in a record must be separated by one or more blank characters, and each field must contain at least one nonblank character. Column alignment of fields from record to record is not required. When source file records are in free field format, the keywords FMT and FORMAT and the field length list should be omitted.

OCNDAT. This is the CalCOFI oceanographic binary file format for data at standard depths (0 to 500 meters) assembled at SWFSC. Each oceanographic station is represented by a header record followed by a set of data records. OCNDAT files are maintained in cruise order, however they can be reordered by station code. When in cruise order, EDMAP can select a single cruise from a file or process all cruises on a file. If an OCNDAT file is converted from binary to ASCII format, it can be processed with the XYDAT option.

The syntax of the command line is:

LOADSORC/OCNDAT(vrbl,ycor,xcor,crd,selc,dflid)

The first four parameters of additional information are mandatory. The last two, selc and dflid, need be submitted only if selective extraction of records from the source file is desired.

vrbl is the data identifier and must be selected from the following list: <TMP>, <SAL>, <OXY>, <SIGT>, <OSAT>, <DYNA> and <DEPTH>.

ycor is the Y coordinate. It must be entered as one of the following: <TMP>, <SAL>, <OXY>, <SIGT>, <DEPTH>, <DISTN> or <DISTP>.

xcor is the X coordinate. It must be entered as one of the following: <TMP>, <SAL>, <OXY>, <SIGT>, <TIME>, <DISTN> or <DISTP>.

crcd is a cruise code for data selection. The cruise code occupies the last field in OCNDAT header records, in files ordered by cruise. When cruise codes are present in header records, a single cruise can be selected for processing by entering the four digit (year-month) code. For example, 8410 would be entered to select October 1984. To disable selection by cruise code, the entry for crcd should be 9099.

selc is a search key used for selective extraction of data records, as described for XYDAT. When submitted, it will be compared to data values in a designated field and only those records for which a match is found will be accepted. The value to be entered for selc depends on the coordinates. When the Y coordinate is DEPTH and the X coordinate is DISTN or DISTP or TIME, the corresponding values for selc must conform to those in the following table. In these cases, the designated field will be picked automatically and the number entered for dfln (if any) will be ignored. Otherwise, a designated field must be submitted with selc.

X coord.	Y coord.	Select Parameter
DISTN	DEPTH	Line (3 digit code) * 1000
DISTP	DEPTH	Station (3 digit code)
TIME	DEPTH	Line_station (6 digit code)

dfln is the designated field number associated with selc. It can point to any field in the data records.

MWXOB. This is a special format in which ship's synoptic surface weather observations, received from Fleet Numerical Oceanographic Center, are stored at SWFSC. The syntax of the command line is:

```
LOADSORC/MWXOB(vrbl,yrmo,day_from,<->,day_to)
```

All of the fields in additional information are mandatory, and are described below.

vrbl is the data identifier. It must be entered as one of the following: <SST>, <WIND>, <WSPD>, <PRES>, <CLDS>, <AIR>, <DEW>, and <EVAP>.

yrmo is a four digit year-month code; two digits for year followed by two digits for month.

day_from-day to designates the period for which data is to be extracted from the source file. It is entered as two digits, dash, two digits, with no embedded spaces.

RCDAT. This is a special format option for data from spreadsheet programs, saved in ASCII files. The files should not contain header records. The X and Y coordinates are assumed to be column and row numbers, respectively. However, coordinate values are not included in data records; they are determined from the locations of data values in the

files. This requires that the spreadsheet values be copied to a file in a particular order, starting with the first row and column, and proceeding left to right across the columns and down the rows. The data values in each record must be separated by one or more spaces and must represent successive columns in a single row. Data records must not exceed 255 characters. If all of the columns in a row cannot be accommodated in a single record, then the spreadsheet can be divided into groups of columns, or ensembles, for the purpose of creating the data file. Some spreadsheet programs do this automatically.

The syntax of the command line is:

LOADSORC/RCDAT(vrbl,cpe,cols,rows,actn)

All of the parameters in additional information are mandatory except the last one, actn.

vrbl must be entered as <ROCO>.

cpe is the number of columns (or fields) per ensemble, as described above.

cols is the total number of columns (or fields) per row.

rows is the total number of rows in the data set. An incorrect value entered for this, or the two preceding parameters will confuse the row and column identification of data values.

actn can be entered as one of two options, EXY or EEZ. EXY provides the option of exchanging row and column coordinates at the time the gridfield is defined. When entered, it causes row number to be the X coordinate and column number to be the Y coordinate. EEZ must be used when processing the NOAA/NOS bathymetric data for the exclusive economic zone. It causes the row, column coordinates to be converted to latitude, longitude coordinates.

MOREDATA

Provides a means for accumulating input data from more than one source file. Although a primary option, it is meaningful only when it is submitted with <LOADSORC>. MOREDATA tells EDMAP to terminate the current run after loading the source data and to look for another set of run parameters with another LOADSORC command. The LOADSORC command can reference only one source file. Therefore, a new set of run parameters is required for each additional source file to be processed.

ANALYZE/CENTER/SETNUL/GUESSFLD

Data from user's source files are reformatted and saved in binary files identified, internally, by the file variable IJVAL. The command ANALYZE causes the data from these files to be fitted to the nodes of a two-dimensional grid, represented by an array in memory. The three option qualifiers affect the way this is done. The process is implemented in two phases: (1)

direct assignment of data values to grid points, and (2) interpolation of values for grid points not affected by direct assignment. Direct assignment applies only to grid points within one grid interval of each data value. The interpolation can be carried out across spans of varying length, which is determined by the specification parameter, ANL. The qualifiers, all of which are optional, are described below.

CENTER. Forces the assignment of each data value to the nearest grid point. Otherwise, each data value makes a weighted contribution to each of the four surrounding grid points.

SETNUL. Assigns the value submitted for the specification parameter, NUL, to grid points not assigned values in either phase 1 or 2, above. It's purpose is to mark undefined areas of the gridfield in order to preclude extrapolation of contours into such areas.

GUESSFLD. Provides a way to examine and reject data values which fail to satisfy selection criteria. This option is useful for processing data which have not been subjected to quality control. When GUESSFLD is submitted, the gridfield in memory must be initialized before analysis by loading a grid file created in a previous run. This can be done by placing a LOADF1 command ahead of the ANALYZE command in the command file. Then, when the data are processed, each data value is compared to the initialized gridfield, interpolated to the actual location of the data point. If the absolute value of the difference exceeds the value of the specification parameter, REJ, the data value is rejected.

SMOOTH

Invokes a two-dimensional smoothing function which modifies the gridfield in memory at each grid point. A choice of smoothing options is provided by different combinations of two 9-point operators which are selected by the specification parameter, SMH. An explanation of SMH and the available smoothing options is given in the section on specification parameters.

SAVEF1,SAVEF2,SAVEF3,SAVEF4

These four commands each create a disk file which contains the contents of the gridfield in memory at the time the command was activated. For each SAVEF_ command, there is an associated file identifier, FIELD_. Whenever a SAVEF_ command appears in a command file, the associated file identifier must be equated to an actual file name in a filename parameter. The provision of four different grid file identifiers makes it possible to have multiple grid files opened at the time, and thereby to support operations which modify or combine gridfields. Reference to the flow chart (Figure 4) will assist in the placement of SAVEF_ and LOADF_ commands to get desired results.

SAVEFLD

Saves the gridfield in memory to a grid file, just as the SAVEF_ commands do. In this case, however, the name of the grid file must be submitted in a filename parameter with the file

identifier GRDFIL. SAVEFLD must be used when either of the commands DIGITPLT or VECTORPLT are submitted in a command file.

LOADF1,LOADF2,LOADF3,LOADF4

These four commands are used to retrieve gridfields previously saved on disk with one of the SAVEF_ commands described above. As with the SAVEF_ commands, the LOADF_ commands are also associated with the file identifiers, FIELD1, Field2, FIELD3 and FIELD4. A retrieved file replaces or may be combined with, the gridfield in memory. The same file can be retrieved more than once in the same run. Reference to the flow chart (Figure 4) will assist in the placement of LOADF_ and SAVEF_ commands in the command file.

EDITFLD/String_Parameter

This command invokes a call, during execution, to a procedure named EDITFLD, which has access to the gridfield in memory. This procedure contains routines which can be used to modify or transform the gridfield in ways not possible with the other command options. To implement this command, the special routines must be incorporated into EDITFLD's source code, which must then be recompiled and relinked to EDMAP. The String_Parameter will be passed to EDITFLD at runtime. EDITFLD will interpret the first six characters of the String_Parameter as the name of the routine to be executed. A more detailed description of this process is given in a later section.

SUBTRACT

This command is used with, and must be preceded by a LOADF_ command. It causes a grid file, identified by FIELD_ in a filename parameter, to be loaded and subtracted point by point from the gridfield in memory. The result of the subtraction replaces the gridfield in memory.

ADDFLD

Combines a previously saved gridfield with the gridfield in memory. As with SUBTRACT, it must be preceded by a retrieve command, LOADF_, to access a grid file on disk identified by FIELD_ in a filename parameter. The grid point values from the grid file are combined with those in the gridfield in memory, point by point. The result is a weighted average of the two fields and replaces the gridfield in memory. ADDFLD and SUBTRACT cannot be entered in the same run.

PRINTFLD

Creates a printer file of the gridfield in memory, and its associated count field. The resulting file is an ASCII text file, with a maximum line length of 72 characters. It contains row and column labels in the user's (X, Y) coordinates. The file identifier, LPRINT, must be associated with the ASCII text file in a filename parameter.

TRACKCNTRS/SETCNT(Contour list)

This command initiates the contour tracking routine on the gridfield in memory. The gridfield is scanned for selected contours by examining every pair of adjacent grid points. When a contour segment is encountered, the scanning is suspended while the contour is tracked and its coordinates recorded. Contour selection may be made with the specification parameters, COG, CIN and CTX, or with a contour list specified as (additional info.) in the command line following the qualifier option, SETCNT (see below). The degree of smoothness of contours is affected by the value of the specification parameter SQD. If this value is greater than 1, it is squared to define a subsquare matrix for each unit grid area through which a contour is tracked. Coordinates are computed by curvilinear interpolation at each point where the contour intersects a subsquare boundary. The contour coordinates are organized by contour segments and stored in a file associated with the file identifier, CTRFIL, in a filename parameter. The digitized contour segments will be retrieved with the command PLOT CNTRS and reorganized as array parameters to be passed to CALLPSUB, a procedure that creates a file of generic plot calls.

The optional qualifier, SETCNT, specifies that contour selection is to be made solely from the contour list inside the parentheses. The elements in the list should be unscaled values in ascending order, separated either by spaces or commas. The list may occupy more than one line.

PLOT CNTRS

One of five primary commands which result in the creation of a plot file. The others are XYDATAPLT, DIGITPLT, VECTORPLT and OVERLAYPLT. Each of these commands is associated with specific kinds of plot features. A plot file contains generic plot calls which can represent contributions from one or more of the primary plot commands. Programs provided to convert these plot calls to device-specific code include DSPLPLOT, HPDRIVER AND WPDRIVER. These programs are described in the section on RUNEDMAP.

PLOT CNTRS reads contour coordinates for digitized contour segments from a disk file created by TRACKCNTRS, associated with the file identifier, CTRFIL. Contour labels can be inserted in selected contours at staggered locations, as specified by the parameters, CIN, COG and CTL. Tic marks and labels will be plotted along the plot border in accordance with the specification parameters XTC, YTC, XLB, YLB, XLP, YLP, XLD and YLD. The title string submitted for the current run (if any) will be centered above the top border. The height of the title and of all contour and axis labels are specified by the parameter HGT. The size of the plot will be determined by the scaling parameters MFX and MFY.

XYDATAPLT/CONNECT

Causes data values from the data file associated with the file identifier, IJVAL (created by LOADSORC), to be plotted at their actual coordinates. Alternately, the data values can be replaced by a symbol, selected by the value of the specification parameter SYM. If SYM is zero, the data values will be plotted. If it is greater than zero, it will be used to select a symbol from the ASCII character code. If the SYM parameter specifies symbols, the qualifier option, CONNECT will cause the plotted symbols to be connected with straight line segments. This

option makes it possible to plot mathematically defined curves. XYDATAPLT can be used together with PLOT CNTRS, VECTORPLT, CONCATPLT or OVERLAYPLT.

DIGITPLT

Causes the grid values from a grid file, equated to the file identifier, GRDFIL, to be plotted at grid point locations. The number of decimal places to be plotted for each value is determined by the parameter DPL. If DPL is zero, the decimal point will be suppressed. DIGITPLT can be used together with PLOT CNTRS, CONCATPLT or OVERLAYPLT.

VECTORPLT

Produces a vector plot depicted by arrows and numbers representing directions and magnitudes of wind vectors at grid points. The vector data are read from a grid file equated to the file identifier GRDFIL, which is also referenced by DIGITPLT. Consequently, these two commands cannot be submitted in the same run. Vector data is formatted in a different way from scalar data in grid files. Vector gridfields cannot be processed by PRINTFLD to create printer files.

The specification parameter VEC provides control over the length of the arrows and determines whether the wind magnitude will be plotted. It is described in the section on specification parameters. VECTORPLT can be used together with PLOT CNTRS, XYDATAPLT, CONCATPLT or OVERLAYPLT.

OVERLAYPLT

This command provides a capability to superimpose special features on a plot, such as coastlines (for geographical plots), custom labels, scales, titles and simple graphics. The overlay plot data must be stored in a disk file equated to the file identifier, CSTFIL, at run time. Overlay plot files are created with a program named MAKOVRLA, which is described in a subsequent section. OVERLAYPLT can be used together with any of the other plot commands.

CONCATPLT/Qualifier

A secondary plot command which provides a capability to concatenate plot features. Each of the primary plot commands produces a particular type of plot and is associated with a particular file identifier. The CONCATPLT qualifier must be selected from the following list to designate which plot type will be used for concatenation:

DATPLT, CTRPLT, GRDPLT, VECPLT, OVLPLT.

Plot data from up to 999 files can be concatenated. They must all have the same name, except for extension. File switching, during a run, is keyed to the filename extensions of the files to be concatenated. These extensions must consist of consecutive, three-digit numbers, starting with 001. The filename with the largest extension number must be equated in a filename parameter to CONCAT.

For example, assume that a contour plot is to be made from four files of contour coordinates, whose filenames are: PLOT.ONE, AREA.001, AREA.002, and AREA.003. The requisite filename parameters would be: PLTFIL=PLOT.ONE and CONCAT=AREA.003. The program would first process the data in PLOT.ONE, and then proceed, in turn, with AREA.001, AREA.002 and AREA.003.

TIMEPLT/Qualifier(additional info.)

Provides a special mode of handling alternative units and labeling options for the X coordinate of a plot when the X coordinate is TIME. It is applicable only when the source file type is OCNDAT. The qualifier option is required and must be entered as one of three keywords:

YRMO, MNTH or DAYS.

The selection must reflect the actual units of X coordinate values in the source file. When the qualifier option is entered as MNTH or DAYS, the plot is presumed to depict an annual cycle with a range of 1..12 for MNTH and 1..365 for DAYS. The actual range of the coordinates in the source file need not be restricted to these values, however, and the specification parameters XZO and XMX may be set to allow any number of repetitions of the annual cycle.

When the qualifier option is MNTH, the parameter XGD should be entered with a value of 1. When the option is DAYS, the value submitted for XGD should be a simple fraction, or multiple, of 30.42. The X coordinates of data from the source data will automatically be converted to Mnth with the formula:

$$\text{Mnth} = (\text{Xcoord} + 15.21) / 30.42.$$

The (additional info.) portion of the command line specifies whether the X axis labels will be plotted as letters or digits. The valid entries are LTRS and DGTS. If LTRS is submitted, X axis labels will be selected from the month abbreviations: JAN, FEB, MAR,...,DEC. In addition, when the qualifier option is YRMO, the appropriate year will be plotted below each JUL. If DGTS is submitted as (Additional info.), and the qualifier option is YRMO, the appropriate year-month code will be plotted as X axis labels. Otherwise, two-digit codes will be selected from the sequence 01,02,...,12.

SPECIFICATION PARAMETERS

Specification parameters provide information for defining the gridfield and for carrying out certain command functions. Each parameter entered into the command file must be preceded by a specification keyword and an equal sign, for example: XZO=10. Specifications must be separated by one or more spaces.

Specification parameters can be classified in four groups, each of which relates to one or more of the primary functions of EDMAP. They are:

1. Parameters for defining a gridfield.
2. Parameters for analyzing source data to, or modifying a gridfield.

3. Parameters for selecting and tracking contours.
4. Parameters for making a plot.

Parameters for defining a gridfield.

XZO, YZO, XMX, YMX

These four parameters specify the left, lower, right and upper boundaries, respectively, of the gridfield. They must be entered in units of the X and Y coordinates of the data field. If the data field is geographical (with coordinates of latitude and longitude), one of the letters N, S, E or W must be appended to each parameter value, as appropriate.

XGD, YGD

These parameters specify the interval between grid lines (grid interval) along the X and Y coordinates, respectively. They must be entered in the same units as the boundary specifications. EDMAP designates the lower left corner of the grid (defined by XZO and YZO) as the grid origin, with coordinates (I,J) = (1,1). The grid coordinates at the upper right corner (defined by XMX and YMX) are (I,J) = (IMX, JMX). The values of IMX and JMX, therefore, are the grid dimensions. They are computed from the grid specifications as follows:

$$\begin{aligned} \text{IMX} &= \text{TRUNC}((\text{XMX}-\text{XZO})/\text{XGD} + 1.5) \\ \text{JMX} &= \text{TRUNC}((\text{YMX}-\text{YZO})/\text{YGD} + 1.5) \end{aligned}$$

where TRUNC means the evaluated expression is truncated to an integer.

PJC

Identifies the type of coordinates as geographical, oceanographic or other. The range of valid parameters is 1 to 5, as defined below:

1. Geographical coordinates (Mercator projection).
2. Geographical coordinates (Lambert's Equal Area projection).
3. Geographical coordinates (no projection).
4. Oceanographic coordinates with depth as the Y coordinate.
5. Any linear rectangular coordinates.

MPR

Specifies a reference longitude for geographical coordinates when PJC <= 3. The sum of the longitudinal distances from MPR to XZO and XMX must not exceed 180 deg. The value entered should be followed by W or E for west longitude or east longitude, respectively.

2. Parameters for fitting data values to a grid.

ANL

Specifies the scope of interpolation for grid points which were not assigned values directly from source data. The interpolation for such grid points is carried out after all the values in the data file have been processed. The method is described in a later section. The value submitted for ANL is interpreted as the maximum number of grid intervals across which interpolation can be performed in the attempt to assign values to target grid points. If ANL is submitted as an integer, it applies to both the I (horizontal) and J (vertical) directions. If it is submitted as a mixed number, the integer part will apply to the I direction and the fractional part, multiplied by 10, will apply to the J direction, except if the fractional part is equal to 0.1 it will be interpreted as 10 rather than 1.

Certain values of ANL invoke special actions. If ANL = 0, the analysis procedure will be bypassed and the command option ANALYZE nullified; if ANL=1, the analysis procedure will be terminated after phase 1, the direct assignment of the data values to gridpoints. Otherwise, values submitted in the range 2..10 will set a search area for interpolation that will be approximately proportional to the parameter value squared. The default value of ANL is 1.

SCL

Specifies the precision to be maintained in fitting the source data to the gridfield. It should be entered as the number of decimal places to be preserved in the computation of gridpoint values. SCL will be interpreted as a power of ten for scaling the source data.

NUL

Specifies a value to be assigned to grid points which are not defined during the analysis procedure, either by direct assignment of data values in phase 1 or by interpolation in phase 2. This action occurs only if the qualifier option SETNUL is submitted with ANALYZE. The NUL value is used by PRINTFLD to suppress printing of values at undefined gridpoints. It is also used to suppress contour tracking through undefined portions of a gridfield by TRACKCNTRS. The value submitted for NUL should be outside the range of valid data values.

REJ

Specifies a window for accepting or rejecting data values during the analysis process when editing is invoked by the qualifier option GUESSFLD, submitted with ANALYZE. It requires prior initialization of the gridfield from a grid file created in a previous run. The grid file is presumed to be a reference field applicable to the source data. Editing is performed by comparing each data value with the value of the initialized gridfield interpolated to the coordinates of the data point. If the absolute value of the difference exceeds the value of REJ, the data value will be rejected. No editing will be performed if REJ = 0 (default value).

SMH

Specifies a selection code for smoothing the gridfield when the command option SMOOTH is submitted. The smoothing is applied point by point to the gridfield in memory, and the result replaces the gridfield in memory. Smoothing is employed to reduce or eliminate small scale irregularities in contoured fields while retaining the essence of large scale features.

EDMAP uses three nine point, two dimensional smoothing operators. Each is equivalent to a three point operator applied in a direction parallel to the X axis first, then to the Y axis. The array of weights defined for each nine point operator is shown below.

(1)	(2)	(3)
1 2 1	1 -6 1	1 8 1
2 4 2	-6 36 -6	8 64 8
1 2 1	1 -6 1	1 8 1

The first of these is used as the primary operator. The effective weights that would result from repeated applications of this operator can be found in a table of binomial coefficients. The weighted contributions of all nine gridpoints are used to smooth the value at the central point. The sum of the weights, in this case, is 16. The original (unsmoothed) value at a grid point therefore contributes 4/16, or 25% to the smoothed value at that point after one pass. A secondary operator, with the weights shown under (2), above is provided to restore weight to the original (unsmoothed) value. The contribution of the latter to a smoothed grid point value after one pass of each of these operators would be just over 39%.

A third operator, with weights shown under (3) above, was designed for patchy, discontinuous fields of discrete elements. Such fields may require some smoothing to avoid overloading EDMAP's tracking capacity with too many contour segments, but would be oversmoothed by the first two operators. The contribution of an original grid point value to a smoothed value after one pass of this operator would be 64%.

The results obtained with different values of SMH are described below.

- 0 No smoothing occurs; the SMOOTH command is ignored.
- 1 The primary smoothing operator (1) is applied once
- 2 The primary operator is applied followed by the secondary operator (2); equivalent to applying a single 25-point operator.
- 3 The primary smoothing operator is applied twice; equivalent to applying a single 25-point operator.
- 10 The third smoothing operator (3) is applied once.

The actions described for SMH = 2 or 3 can be repeated by adding multiples of 3 to either code. For example, entering 5 for SMH would cause the smoothing described for code 2 to be applied twice.

3. Parameters for tracking contours.

COG

Specifies the lowest valued contour that will be selected in the tracking process. The value entered for COG should, therefore, be small enough to ensure that all desired contours in a gridfield will be selected. There is no corresponding upper limit. The default value of COG is 0.

CIN

Specifies a fixed interval for contour selection, provided that the value of CTX equals 0 (see below) and that contour selection is not superseded by the qualifier option SETCNT. When CIN is the determining parameter, contours will be selected from the following sequence:

COG, COG+CIN, COG+2*CIN, COG+3*CIN,...etc.

The default value for CIN=1.

CTX

Specifies a base for selecting exponentially varying contours. When the value submitted for CTX is greater than 1 and contour selection is not superseded by the qualifier option SETCNT, contours will be selected from the following sequence:

COG, COG+1, COG+CTX, COG+CTX², COG+CTX³,...etc.,

where $\langle \wedge \rangle$ denotes exponentiation. When CTX is used to determine contour intervals, CIN is ignored and COG should be greater than or equal to zero.

SQD

Specifies a subsquare division factor which controls the resolution with which contours are tracked through grid squares. When a grid square is entered, in the tracking process, it is divided into a number of subsquares determined by the value of SQD squared. Coordinates will be computed for a contour at points of intersection with subsquare boundaries, by means of a curvilinear interpolation procedure. Values submitted for SQD should be in the range 1..10. The larger values result in smoother contours and increased CPU time. The default value of SQD is 1.

FXD

Invokes a routine in EDITFLD that makes a slight adjustment to gridpoint values when they are exactly equal to contours that were selected for tracking with TRACKCNTRS. It reduces the probability of getting caught in an infinite loop during the tracking process. The specification is enabled when submitted with a value of 1. Otherwise it is disabled.

4. Parameters for making the plot.

ORG

Sets the grid origin with reference to the origin of the physical device (plotter), or screen, on which the plot will be drawn or displayed. The grid origin is the point corresponding to the grid coordinates (I,J) = (1,1). Moving the grid origin makes it possible to plot labels or captions outside the grid boundary and remain inside the boundaries of the plotter surface. The displacement is in inches and is applied to both axes, moving the grid origin up and to the right on the plotting surface. The default value is 1.0.

SPB

Suppresses the plot border when submitted with a value of 1. Otherwise, the border will be plotted. When SPB = 1, all plot parameters affecting tic marks or labels along the plot border will be ignored. The default value is 0.

MFX,MFY

These parameters are scaling factors and determine the size of the plot. They must be entered as inches per unit of X and Y, respectively, where X and Y are the coordinates of the user's source data. The plot width is determined by multiplying the range of X by MFX. The plot height is determined by multiplying the range of Y by MFY, except when the plot is a geographical map projection (when PJC = 1 or 2). In that case, the plot height will be determined automatically as a function of plot width, and MFY will be ignored. The values to be entered for MFX and MFY for any desired plot width and height can be obtained with the following formulas:

$$\begin{aligned} \text{MFX} &= \text{Plot_width}/\text{ABS}(\text{XMX}-\text{XZO}) \\ \text{MFY} &= \text{Plot_height}/\text{ABS}(\text{YMX}-\text{YZO}). \end{aligned}$$

HGT

Specifies the height (in inches) of axis labels, contour labels, and characters in the title string. The default value of HGT is 0.1.

XTC,YTC

Specify how tic marks will be plotted along the plot borders. Tic marks can be plotted only at grid points, that is, at intervals specified by the parameters XGD and YGD. The locations of tic marks and labels along both X and Y axes are determined with reference to the grid origin, where (I,J) = (1,1). If XTC=0 or YTC=0 (default values), the corresponding tic marks will not be plotted. Otherwise, tic marks will be plotted at every XTCth grid point along the X axis, and at every YTCth grid point along the Y axis.

XLB,YLB

Specify the spacing between labels plotted along the X and Y axes. The placement of the first label relative to the grid origin, on either axis, will be determined by the values of XLP and YLP. The number of decimal places in axis labels will be determined by the parameters XLD and YLD. Values entered for XLB and YLB will be interpreted as the number of grid intervals between labels. If the value of either XLB or YLB is 0 (default value), the corresponding labels will not be plotted.

XLP,YLP

Specify starting points for plotting labels along the X and Y axes. The parameter values will be interpreted as the number of grid intervals to skip before plotting the first label. For example, if XLP=2, the first label on the X axis would be plotted at the third grid point, counting from the grid origin. Subsequent labels would be plotted at intervals specified by XLB. Selection of tic mark and label specifications should be coordinated to assure desired alignment of these features.

XLD,YLD

Specify the number of digits to be plotted to the right of the decimal point in X and Y axis labels. If either parameter is entered as zero, the corresponding decimal points will not be plotted.

DPL

Specifies the number of decimal digits to be plotted to the right of the decimal point for grid or data values when either DIGITPLT or XYDATAPLT is submitted. In the latter case, however, DPL would be ignored if the value of SYM was greater than 0. The default value of DPL is 0.

SEG

Suppresses plotting of short segments on contour plots, if submitted with a value > 1 . Contour segments with fewer than SEG points will not be plotted.

CTL

Controls the plotting of contour labels. If the value of CTL is 0 no labels will be plotted. If CTL = 1, every contour will be labeled; otherwise every CTLth contour will be labeled. The number of decimal places in contour labels will be determined automatically from the value of CTL entered for the current run, and from values of COG and CIN that were specified for the run in which the plot file was created. The default value for CTL is 0.

SYM

Specifies whether data values or special symbols are to be plotted at the locations of source data points, when the command option XYDATAPLT is invoked. If the parameter value is 0 (default value), then the data values will be plotted as decimal numbers with the number of

decimal places specified by the parameter DPL. Otherwise, the value for SYM may be entered as a mixed number, in which case the integer part will be used to select a symbol from the ASCII character set. The fractional part will be multiplied by the value of the parameter HGT to determine the height of the plotted symbol. If the fractional part is zero, it will be interpreted as 1 for this purpose.

At run time, the current value of SYM is stored in the header record of the data file equated to IJVAL when that file is created by LOADSORC. It is recovered when IJVAL is opened by XYDATAPLT and used as described above. This provides a means for assigning different symbols for different sets of data that are to be plotted (using CONCATPLT) on the same plot.

VEC

Specifies whether the magnitudes of wind vectors will be plotted or suppressed when the command option VECTORPLT is invoked. It also controls the lengths of arrows depicting the wind directions. The parameter value may be entered as a mixed number. The integer part must be either 1 (to plot the magnitudes) or 0 (to suppress them). If VEC = 1 the mean vector magnitude will be plotted above the arithmetic mean at each grid point. The ratio of the vector mean to the arithmetic mean is a measure of the steadiness of the wind when the number of data values represented is more than one. The magnitudes will be plotted as rounded integers.

The fractional part of the parameter value will be multiplied by the mesh length of the grid along the X axis to determine the length of the arrow shafts. If the fractional part is 0 or missing, it will be interpreted as 1 for this purpose. The mesh length is the product of the specification parameters MFX and XGD.

XPV, YPV, ROT

These three parameters provide for rotation of the (I,J) coordinates with respect to the (X,Y) coordinates when a plot is created. This capability has particular application in mapping data on the CalCOFI grid. It allows the use of station and line numbers as X and Y coordinates for analyzing CalCOFI data to a gridfield and facilitates conversion to latitude and longitude coordinates when making a plot.

The values submitted for XPV and YPV will specify the (X,Y) coordinates of the pivot point. The value of ROT specifies the amount of rotation in degrees (positive counterclockwise). The geographical orientation of the CalCOFI grid represents a counterclockwise rotation of 30 degrees from a north-south direction, around a pivot point about 44 nautical miles off Point Conception, on the California coast. The rotation parameters applicable to CalCOFI grid rotation are:

```
XPV=121.15W
YPV=34.15N
ROT=30.0
```

These values should be used only for the CalCOFI grid, since they result in a transformation of coordinates as well as axis rotation. However, the rotation parameters can be used in plotting

any gridfield by submitting a nonzero value for ROT, and appropriate values for the pivot point parameters, XPV and YPV.

FILENAME PARAMETERS

Filename parameters equate the names of actual disk files to program file identifiers. EDMAP uses four kinds of external files: source files, data files, grid files and files of contour segments, coastline segments or other overlay elements. Source files contain user's data, and are for input only. Data files are created from source files and contain a subset of the data values and their coordinates. Grid files contain the array of grid point values that define a gridfield. They are created with certain option parameters, and can be retrieved to restore those values to a gridfield in memory at a later time. Contour coordinate files are created by the contour tracking routine. Overlay files are created by the program MAKOVRLA.

The syntax for filename parameters is:

FILID=Filename

where FILID is one of fourteen keywords (listed below) and Filename is the actual filename. Each of the fourteen keywords is associated with a program file variable and, with the exception of LPRINT, falls in one of the four file type categories mentioned above.

File Identifier (keyword)	File Type
1. FIELD1.....	Grid file
2. FIELD2.....	Grid file
3. FIELD3.....	Grid file
4. FIELD4.....	Grid file
5. IJVAL.....	Data file
6. CTRFIL.....	File of contour coordinates
7. GRDFIL.....	Grid file
8. CSTFIL.....	File of overlay plot features
9. CONCAT.....	Any of the above
10. XYDAT.....	Source file
11. MWXOB.....	Source file
12. MARDAT.....	Source file
13. OCNDAT.....	Source file
14. LPRINT.....	Printer file of gridpoint values

GRIDFIELD MODIFICATION WITH EDITFLD

The command options provide a limited number of ways to modify or combine gridfields created by EDMAP. The option EDITFLD can be used to perform other kinds of gridfield modifications, with customized routines which can be added to the source code of one of the components (EDITFLD) which comprise EDMAP.

The source code of EDMAP is written in Turbo Pascal 6.0. Custom routines are Turbo Pascal procedures, nested in procedure EDITFLD. EDITFLD's only function is to pass control to one of these nested routines when the EDITFLD option is activated. The string parameter submitted with the EDITFLD option is passed to the EDITFLD procedure at run time. The first field (six characters) of the string parameter must name one of the nested procedures. EDITFLD will strip the name field from the string parameter and then call the named procedure and pass the remainder of the string parameter to it. It is up to the called procedure to know what other fields, if any, the string parameter contains.

Several predeclared variables and utilities in EDITFLD are available to the custom routines, to facilitate access to the gridfield in memory. These include the variables IC, JC, IMX, JMX and FTR of type INTEGER, NULVAL and GPV of type LONGINT, and two procedures: GETV(IC,JC, GPV) and PUTV(IC,JC,GPV). The gridfield in memory is represented as a two dimensional array of LONGINT (integers with range from -2147483648 to 2147483647) with dimensions given by IMX and JMX. The array elements are scaled grid point values. FTR is the scale factor. NULVAL is the scaled null value for undefined grid points, submitted as the specification parameter NUL. IC and JC are variables which can be used for indexing the gridfield array. GPV can be used to store and retrieve array elements. However any LONGINT variable can be used for this purpose. The procedure GETV assigns the gridfield array element at (IC,JC) to the parameter GPV. The procedure PUTV assigns the value of GPV to the array element at (IC,JC).

A predeclared function, GETFLD(Str_Parm), is available to extract the remaining fields, if any, from the string parameter passed to the custom routine. A field, in this context, is a contiguous sequence of non blank characters, delimited by blank(s). GETFLD is of type STRING. Each time it is called, it returns the first field in Str_Parm and then deletes that field. If Str_Parm is blank or empty, it returns one blank character.

CREATING OVERLAY PLOT FILES WITH MAKOVRLA

The command option **OVERLAYPLT** provide a means of superimposing special titles, captions, labels, scales, coastlines and other features on a plot. The overlay elements must be available at run time in an overlay plot file equated to the file identifier, **CSTFIL**, in a filename parameter.

MAKOVRLA is a program written to facilitate preparation of overlay plot files. It consists of two parts. The first part assembles a set of overlay element specifications from user responses to prompts, and stores them in an ASCII file on disk. The second part creates a binary file containing the overlay elements in the format required for input to **EDMAP**. The second file is the one that should be linked to **CSTFIL** in the filename parameter.

MAKOVRLA provides nine types of overlay elements, identified by the following keywords:

**SYMBOL, NUMBER, RECTANGLE, DIAMOND,
POLYGON, BAR, LINES, SCALES and COAST.**

Each element has its own parameters, which are described below. When the user selects an element, **MAKOVRLA** prompts for the parameters required for that element. At the end of the interactive session, **MAKOVRLA** saves the overlay specifications in an ASCII file, then automatically proceeds to the second part, to create the binary file. The user can make changes or corrections to the ASCII file with any text editor, if desired, and rerun **MAKOVRLA** with the option to execute only the second part to make a new binary file.

Overlay elements are located on a plot with reference to the user's X and Y coordinates and (with the exception of **COAST**) may be placed anywhere on the plot, inside or outside the plot border. Overlay elements associated with **COAST** are coastline or other contour segments and, when plotted, are clipped at the plot borders.

The binary file created by **MAKOVRLA** includes a header record with identification fields and parameters relating to the gridfield for which the file was created. When running in the first part of **MAKOVRLA**, the user will be asked to submit values for **XZO, YZO, XMX, YMX, XGD, YGD, PJC** and **MPR**. Responses should conform to those submitted for the corresponding parameters in the **EDMAP** command file. However, if the grid parameters do not agree, the coordinates in the overlay file will be adjusted, if possible, so that the overlay elements are plotted in the right locations.

The element **COAST** is provided to add coastline segments in geographical plots from data submitted to **MAKOVRLA** in a file of coastline coordinates. The coastline file must be organized in segments. Each segment must start with a header record. The first field of every header record must contain the number of records in the associated segment. The remainder of the header record will be interpreted as a segment identification field. Each data record, following a header record, must contain the latitude and longitude of one coastline point. However, other fields can also be present. The user will be asked to enter the number of fields in each record, and the position (field sequence number) of latitude and longitude.

The latitudes and longitudes in some coastline files are stored as integer values with implied decimal points. MAKOVRLA will ask if this is the case and, if so, prompt for the number of implied decimal places. Quadrants are commonly indicated by the use of negative values for latitude and longitude. The convention for latitude is positive north, negative south. The convention for longitude can be either way, and the user will be asked whether negative longitudes, in the coastline file, are east or west.

The remainder of this section describes the parameters required for each type of overlay element. These descriptions are intended primarily to assist in correcting or modifying an existing file of overlay specifications. When starting anew, a session in the first part of MAKOVRLA is recommended as the best way to become familiar with the process.

Overlay parameters must be entered in sets. Each set begins on a new line, with one of the element identifiers. There are no restrictions on the number of sets or on the order in which they are entered. In the following descriptions, element identifiers are enclosed by < > symbols to distinguish them from parameter identifiers that will be replaced by actual numeric or string constants in the overlay file. Parameters may be separated by spaces or commas.

OVERLAY ELEMENTS

<SYMBOL> N,HT,ROT,X,Y,STRING

Plots a character string.

N is the number of characters in the string (including embedded spaces).

HT is height of the characters in inches.

ROT is the rotation (in degrees, counterclockwise) of the string around (X,Y).

X,Y are the coordinates of the lower left corner of the first string character.

STRING is the string to be plotted; it must be entered on a separate line.

Example: SYMBOL 13 0.12 0 122.5W 37.7N
SAN FRANCISCO

<NUMBER> N,HT,ROT (followed by)

N sets of: X,Y,VAL,F.

Specifies a sequence of numbers, where

N is the number of numbers to be plotted.

HT is the height of the numbers in inches.

ROT is the rotation (in degrees, counterclockwise) of each number around (X,Y).

For each number:

X and Y are the coordinates of the lower left corner of the first digit.

VAL is the number to be plotted.

F is the number of fractional digits to be plotted for VAL.

Example: NUMBER 3 0.16 90

3 1 300 0

6 1 400 0

9 1 500 0

<RECTANGLE> X,Y,RX,RY

Specifies a rectangle, where

X and Y are the coordinates of the center.

RX is the half-width in inches.

RY is the half-height in inches.

Example: RECTANGLE 5 3 2.0 1.5

<DIAMOND> X,Y,RX,RY

Specifies a diamond, where

X and Y are the coordinates of the center.

RX is the half-width in inches.

RY is the half-height in inches.

Example: DIAMOND 5 3 2.0 1.5

<POLYGON> N,X,Y,R

Specifies a polygon with a vertex at the top, where

N is the number of sides in the polygon.

X and Y are the coordinates of the center.

R is the distance from the center to any vertex in inches.

Example: POLYGON 5 13.3 21.4 5.0

<BAR> hatch_option,N,SL,SP (followed by)

N sets of: X,Y,DX,DY.

Specifies one or more bars for a bar graph, where hatch_option must be entered as <HATCH> or <EMPTY> to select hatching or no hatching.

N is the number of bars to be plotted.

SL sets the slant of the hatch lines (range = 0..10). The actual angle of the hatch lines will depend on the grid and plot size parameters.

SP sets the density of the hatching (range = 0..10). Enter 0 when hatch_option is <EMPTY>, 10 for maximum density.

For each bar:

X and Y are the coordinates of the lower left corner.

DX is the width, in X coordinate units.

DY is the height, in Y coordinate units.

Example: BAR HATCH 3 4 1

```
1 0 0.5 8
2 0 0.5 13
3 0 0.5 26
```

<LINES> N (followed by)

N sets of: X,Y.

Specifies line segments connecting a sequence of points, where

N is the number of points.

X and Y are the coordinates.

Example: LINES 5

13 20
18 20
18 25
13 25
13 20 (draws a rectangle)

<SCALES> N,DX,DY,X,Y

Specifies a horizontal scale with vertical tic marks. where N is the number of tic marks.

DX is the spacing between tic marks in X coordinate units.

DY is the height of the 1st, 3rd, 5th,...tic marks, expressed in Y coordinate units; the intermediate ones will have a height of DY/2.

X and Y are the coordinates of one end of the horizontal line depicting the scale. The line will be drawn to the right of (X,Y) if DX is positive, and to the left of (X,Y) if DX is negative. The tic marks will be plotted above the line if DY is positive, and below the line if DY is negative. The length of the line will be equal to: $DX*(N-1)$.

Example: SCALE 6 3 1 10 15

<COAST> IDP,NEGLON (followed by)

FILENAME
NFLDS, XLOC, YLOC

where

IDP is the implied number of decimal places in latitudes and longitudes,

NEGLON is either EAST or WEST for negative longitude,

FILENAME is the name of the coastline file,

NFLDS is the number of fields in each record (headers excluded),

XLOC is the field sequence number for longitude and

YLOC is the field sequence number for latitude.

<QUIT>

This keyword must be entered as the last line in an overlay specification file.

EXAMPLE The following is an example of an overlay specification file for **MAKOVRLA**.

```
MAKOVRLA COMMAND FILE
122W 32N 117W 35N
1.0 1.0 1 160 (gridfield parameters)

SYMBOL 13 0.16 0 120.5W 34.5N
PT CONCEPTION (strings start in col 1)

SYMBOL 9 0.16 0 118.3W 33.7N
SAN PEDRO

SYMBOL 9 0.16 0 117.2W 33.2N
SAN DIEGO

SCALES 7 0.167 0.75 118.5W 34.5N

NUMBER 4 0.12 0
118.5W 35.3N 0 0
118.167W 35.3N 20 0
117.833W 35.3N 40 0
117.5W 35.3N 60 0

SYMBOL 14 0.12 0 118.5W 34.0N
nautical miles

COAST 0 EAST (no implied decimal places, east longitudes are neg.)

COASTLINE.DAT (name of coastline file)
2 2 1 (two fields in each record, 2nd field is longitude, 1st field is latitude)

QUIT
```

From these specifications, **MAKOVRLA** would create an overlay plot file with the following features:

1. California coastline from 32N to 35N and the channel islands (from a file named **COASTLINE.DAT**).
2. The names **PT CONCEPTION**, **SAN PEDRO** and **SAN DIEGO**, at those locations.
3. A scale of nautical miles from 0 to 60 in the upper right corner of the plot, with tic marks at intervals of 10, labels at every other tic mark, and the string <nautical miles> below it.

FITTING RANDOMLY DISTRIBUTED DATA TO A GRIDFIELD

The technique used by EDMAP for digitizing contours requires that the source data be interpolated to the nodes of a rectangular grid. The gridfield resulting from this process is explicitly defined only at the grid points and information which the source data may have contained about features between grid points is not retained. In addition, the method used for fitting data values to grid points introduces some spatial distortion. These facts should be considered in selecting the mesh lengths (XGD, YGD) for a gridfield. The mesh lengths should be compatible with the spatial distribution of the data values. If the gridfield is too coarse, it cannot reflect the full resolution afforded by the data field. If it is too fine, extraneous processing will be performed with no added benefit.

In EDMAP, the source data values are assigned directly to one or more grid points within one mesh length of the respective data points. Grid points which are not assigned values directly, become target points of an interpolation process. The procedures are described below.

1. Direct assignment of data values to grid points.

Each data value contributes to the definition of the gridfield at one or more grid points in its immediate proximity. If the qualifier option, CENTER, is submitted with ANALYZE, the I and J coordinates for each data value will be rounded to the nearest integer, effectively assigning the data values to the nearest grid points. Otherwise, if $PJC < > 4$, a weight, $W(i)$, will be computed for the grid point at each corner (i) of the grid square containing each data point, using the following formula:

$$W(i) = (1 - delI(i)^2 + delJ(i)^2) / \text{SUM}(1 - delI(i)^2 + delJ(i)^2), \\ i = 1..4$$

where delI and delJ are the incremental distances from the data point to the grid points, $<^>$ denotes exponentiation and SUM designates summation for $i = 1..4$. This formula gives each data value a total weight of 1.0, whether it contributes to one, two, three or four grid point values. The data value is multiplied by the weight for each of the adjacent grid points and the products are accumulated for each grid point, along with the associated weights, until all of the data values have been processed. If the accumulated weights for a grid point reaches 100, no additional contributions from data values are accepted for that point. When the assignment of data to the gridfield is completed, a value for each grid point affected is computed by dividing the accumulated weighted products for that point by the sum of the accumulated weights.

A different procedure is used for the special case where $PJC = 4$, which designates a gridfield as an oceanographic section, for which the Y coordinate is depth. Data values in the source file are expected to be ordered by depth (from the surface downward) at discrete locations along the X axis. In the analysis process, linear interpolation to horizontal grid lines will be performed using the vertically distributed values at each such location (or station). Horizontal interpolation will then be performed to determine values at vertical grid lines.

When all of the data values have been processed by one of these procedures, the second phase of analysis will be initiated, unless the value of the parameter ANL = 1. In that case the analysis procedure would terminate. This option provides an opportunity to save the gridfield

in a grid file before interpolated values are added. The saved grid file can be retrieved in a subsequent run to continue the analysis if so desired.

2. Interpolation to grid points not directly defined by data.

In this phase of the analysis process, the gridfield is scanned twice in an attempt to interpolate values for target grid points, which remained undefined after phase 1. In the first scan, only those grid points which were assigned values in phase 1 are used for interpolation. This renders the results independent of the starting point and direction of the scan. In the second scan, the interpolated values added in the first scan are used also.

The extent of the search area, around each target point, will be determined by the value of ANL. If this parameter is submitted as a mixed number, the integer part will be used to set the horizontal search range, ANLH, and the fractional part (multiplied by 10) will be used to set the vertical search range, ANLV. If ANL is an integer, its value will be assigned to both ANLH and ANLV. For a target grid point at coordinates (I,J) and a reference grid point (with value assigned) at coordinates (I+L,J+K), the boundaries of the search area are defined as:

$$(I (+or-) (ANLH-1)), (J (+or-) (ANLV-1))$$

with the restriction that $(L*L+K*K)$ must be less than or equal to the larger of:

$$(ANLH-1)^2 + 4 \text{ or } (ANLV-1)^2 + 4$$

where $\langle \rangle$ denotes exponentiation. The effect of the restriction is to cut off the corners of the rectangular search area when ANLH and ANLV are large.

The search is conducted by selecting up to 12 reference grid point values in the search area. Selected grid point values will include those on either side of, and closest to, the target grid point on both the horizontal and vertical grid lines that pass through the target grid point. They also include the closest reference grid point values in the upper left, upper right, lower left and lower right sectors of the search area. If the two closest grid points in any sector are equidistant from the target grid point, both will be selected.

The selected values are used to compute intercepts on the horizontal and vertical grid lines passing through the target grid point. An intercept can be determined by a grid point value on one of these grid lines, or by linear interpolation between grid point values in adjacent corner sectors. Each intercept is assigned a weight, based on the sum of the squared distances of the associated reference points to the target grid point. Each intercept is also assigned a category: Pos_K, Neg_K, Pos_L or Neg_L. The intercept with the lowest weight is chosen from each category for the final interpolations, provided that the associated grid points satisfy the range criteria. The latter are applied to the sum of the distances of the two grid points (used to compute the intercept) to the grid line on which the intercept is located. This sum must be less than or equal to ANLH for an intercept on the vertical line, or to ANLV for an intercept on the horizontal grid line.

Finally, two interpolations are made for the target grid point, using the (Pos_K,Neg_K) pair

for one and the (Pos_L, Neg_L) pair for the other. Each valid interpolation yields a value for the target grid point. If both values are obtained, a weighted average is computed from the weights associated with the respective intercepts.

In essence, this process attempts to find values for undefined grid points with a linear interpolation technique, using reference values at nearby grid points. The parameter ANL allows the user to specify the scope of "nearby". It does not extrapolate into areas for which no data exists, and will not introduce local maxima or minima at target grid points.

TRACKING CONTOURS THROUGH A GRIDFIELD

The procedure used by EDMAP to compute contours employs a search strategy based on a systematic scan of the gridfield, conducted in order to intercept specified contours and track them to their endpoints.

The gridfield scan begins at the grid origin, where $(I,J) = (1,1)$, and proceeds around the grid perimeter in a counterclockwise sense. The scan is controlled by two program variables: T, the perimeter level indicator and P, the orientation of the scan vector. The scan vector spans one pair of grid points on the current scan line. The values of T and P determine the starting point and the direction of scan lines, which extend across the gridfield, and are coincident with grid lines. The orientation of the scan vector is downward when $P=0$, to the right when $P=1$, upward when $P=2$ and to the left when $P=3$. The perimeter level indicator, T, is initialized with the value 1 for the outermost perimeter (the external boundary) of the grid. It is incremented by 1 each time the current scan line shifts inward, toward the center of the grid. The scan vector starts at the grid origin and progresses along the current scan line between successive pairs of grid points. At each position, the grid point values at the tip and base of the scan vector are tested to determine whether a contour has been intercepted. If so, the gridfield scan is temporarily interrupted and the tracking routine is initiated. Otherwise, the scan vector advances to the next position along the scan line.

To initiate tracking, the coordinates of the contour intercept, on the scan line, are stored and marked as the initial point of contour segment. The contour is then tracked into the grid square on the left side of the scan vector. The selected grid square is expanded and divided into subsquares, according to the value of the parameter SQD. In the expanded square, the values of the field at subsquare corners are computed with an interpolation formula derived from the equation of a third order polynomial and McLaurin's series:

$$P(i) = P_2 + i*(P_3 - P_2) + i*(i-1)*(P_1 - P_2 + P_4 - P_3)/4 + \\ I*(i-1)*(i-0.5)*(P_4 - 3*P_3 + 3*P_2 - P_1)/7.5$$

where P_1, P_2, P_3, P_4 are the values of four successive grid points along a horizontal or vertical grid line, and $P(i)$ is the interpolated value at a subsquare corner at the incremental distance, i , between the second and third of the four grid points. If either of the outside values (P_1 or P_4) is undefined (equal to NUL) and the parameter, FXD, was submitted with a value of 1, the interpolation becomes linear. Four vertical interpolations followed by SQD-1 horizontal interpolations are required to compute values for all subsquare corners in the expanded grid square. The use of curvilinear interpolation for subsquare points produces smoother contours.

The main program parameters involved in the tracking routine are grid indices $(I1, J1)$, pointing to the expanded grid square defined above, subsquare indices $(I2, J2)$, pointing to the current subsquare through which the contour is being tracked, a subsquare scan vector orientation Q, which is initialized by P when an initial contour point is found, and the interpolated gridfield values at the base and tip of the subsquare scan vector. The latter spans two adjacent subsquare points. Each contour point is an intercept of the contour with a subsquare boundary and marks its entry into a new subsquare. With each such entry, the grid square and subsquare indices are

adjusted as necessary and the scan vector is shifted from the side through which the new subsquare was entered, to each of the other sides, in turn, until the exit point of the contour from that subsquare is found. The process is straightforward, until the exit point coincides with a subsquare corner. In that case, the tracking routine must determine which of three subsquares, sharing the exit corner point, the contour is entering. It accomplishes this by searching each of the three subsquares, in turn, for an exit.

When the configuration of the gridfield in a local area includes a hyperbolic, or saddle point, the contour may be tracked into a grid square with more than one exit. When the tracking routine detects a multiple exit condition, it will attempt to track the contour straight through the hyperbolic point (which is a point of intersection of two contours with the same value). However, there can be situations with multiple exits, involving subsquare corner points, which the tracking routine cannot resolve, due to the proximity of external or internal boundaries (internal boundaries set off areas which contain undefined grid points). In such cases, the tracking routine may prematurely terminate the contour being tracked, or it may retrace part of another contour. In general, however, the routine will follow a contour to an end point, which may be an external or internal boundary or, if the contour closes on itself, its own initial point.

When an end point is reached, the tracking indices, I1,J1,I2,J2 and scan vector parameter Q, are reset so that the tracking of the current contour can be resumed from its initial point in the opposite direction (outward), and continues until the second end point of the contour is reached.

As the perimeter scan proceeds through the gridfield, shifting inward from level to level, the same contour may be encountered many times. In order to insure that the same contour is not tracked and saved (as output in a disk file) as a consequence of each new encounter, a reference array is maintained which contains a record of contours already tracked. Each time a new initial point is found during the perimeter scan, it is checked against those in the reference array before tracking is initiated. Each time the perimeter scan shifts (inward) to a new level, the reference array is purged of all entries representing points outside of the new perimeter. In this way, its size is kept at a minimum to reduce search time and memory requirements.

The contour coordinates computed by the tracking routine are accumulated in segments and stored in a disk file associated with the file identifier CTRFIL. This file is accessed in EDMAP by PLOTCONTRS, which organizes all of the elements associated with a plot and creates a ASCII file of generic plot calls. Edmap does not drive plotters or screen displays. Other programs must be employed to convert the generic plot calls to device specific codes to send the plot to a screen or to a plotter.

APPENDIX A. Summary of option and specification parameters

Option parameters (commands).

- 1 **ADDFLD** Combines a gridfield, saved on disk in a previous run, with the gridfield in memory.
- 2 **ANALYZE** Fits source data, converted to (I,J) coordinates by **LOADSORC**, to a rectangular grid.
- 3 **CENTER** Qualifier option for **ANALYZE**; causes data values to be assigned to the nearest grid point.
- 4 **CONCATPLT** Concatenates plot features from more than one run of **EDMAP** onto one plot.
- 5 **DIGITPLT** Causes gridpoint values to be plotted as numbers at grid point locations.
- 6 **EDITFLD** Modifies a gridfield at run time by invoking special routines loaded into the **EDMAP** source code.
- 7 **SAVEFLD** Causes the gridfield in memory to be saved in a grid file for subsequent access by **DIGITPLT** or **VECTORPLT**.
- 8 **GUESSFLD** Qualifier option for **ANALYZE**; invokes a reference gridfield to screen source data values during analysis.
- 9-12 **LOADF1, LOADF2, LOADF3, LOADF4** Causes a gridfield, previously saved on disk, to be loaded into memory.
- 13 **LOADSORC** Loads source data from user's file and converts (X,Y) coordinates to (I,J) grid coordinates.
- 14 **MOREDATA** Option used with **LOADSORC**; causes data to be added from a another source file before proceeding with analysis.
- 15 **OVERLAYPLT** Causes special plot features stored in an overlay plot file to be superimposed on a plot.
- 16 **PLOTCNTRS** Produces a contour plot file.
- 17 **PRINTFLD** Creates a printer file of the gridfield in memory, complete with row and column labels.
- 18-21 **SAVEF1, SAVEF2, SAVEF3, SAVEF4** Causes the gridfield in memory to be saved in a disk file.

- 22 SETNUL Qualifier option for ANALYZE; causes the null value NUL, to be assigned to grid points not defined by the source data.
- 23 FORMAT Qualifier option for LOADSORC; specifies the format for fixed field data records.
- 24 SMOOTH Modifies the gridfield in memory with a smoothing procedure.
- 25 SUBTRACT Subtracts a gridfield, saved on disk in a previous run, from the gridfield in memory.
- 26 TIMEPLT Specifies alternative labeling options for the X axis when the X coordinate is TIME and the source file format is designated as OCNDAT.
- 27 TRACKCNTRS Tracks selected contours through the grid- field in memory and saves the coordinates in a disk file.
- 28 VECTORPLT Plots wind data fitted to a gridfield, as arrows (direction) and numbers (speed) at grid points.
- 29 XYDATAPLT Plots data values, or symbols, at the actual locations of the data points.
- 30 IJVOPN Qualifier option for MOREDATA; automatically invoked to permit accumulation of additional source in a disk file associated with the file variable, IJVAL.
- 31 SETCNT Qualifier option for TRACKCNTRS; provides an alternate mode of contour selection.
- 32-36 DATPLT, CTRPLT, GRDPLT, VECPLT, OVLPLT Qualifier options for CONCATPLT; designates plot file type for concatenation.

Specification parameters

- 1-2 **IMX, JMX** Dimensions of the gridfield, computed automatically and used to control operations on the grid.
- 3-4 **XZO, YZO** (X,Y) coordinates of the lower, left corner of the gridfield.
- 5-6 **XXM, YMX** (X,Y) coordinates of the upper, right corner of the gridfield.
- 7-8 **XGD, YGD** Specifies grid interval, or mesh length, in units of the X and Y coordinates, respectively.
- 9 **PJC** Code value from 1 to 5 specifying type of coordinates.
- 10 **MPR** Reference longitude for converting the X coordinate values to the I grid coordinate when PJC is 1,2 or 3.
- 11 **NUL** Specifies a value to be assigned to undefined grid points when SETNUL is true.
- 12 **SCL** Specifies a power of 10 to scale data values when analyzing the gridfield, to preserve precision. Gridpoint values are processed as long integers.
- 13 **SYM** Code value designating data value or symbol to be plotted by XYDATAPLT.
- 14 **SQD** Subsquare division factor specifying resolution of points computed when tracking contours through a gridfield.
- 15 **LRL** Logical record length determined automatically and used internally by EDMAP to block records when creating grid files.
- 16 **SMH** Code value to select smoothing operator when SMOOTH is invoked.
- 17 **FXD** Enables a call to a routine in EDITFLD, when entered as 1, to make minor adjustments to gridpoint values as needed by TRACKCNTRS for digitizing contours. The default value is 0.
- 18 **ANL** Code value specifying scope of search area when interpolating values for target grid points.
- 19 **REJ** Specifies a range, relative to a reference field, for accepting, or rejecting, data values when ANALYZE and GUESSFLD are invoked.
- 20 **CIN** Specifies a contour interval for contour selection when TRACKCNTRS is elected.
- 21 **COG** Specifies a contour origin to be used with CIN for selecting contours. It also sets the minimum contour value.

- 22 CTL Code value used to control labeling of contours when PLOTCNTRS is invoked.
- 23-24 MFX, MFY Map scale parameters which determine the size of a plot.
- 25 HGT Specifies the height of the plot title and of axis and contour labels.
- 26-27 XTC, YTC Specify where tick marks will be plotted along the X and Y axes, respectively.
- 28-29 XLB, YLB Specify where labels will be plotted along the X and Y axes, respectively.
- 30-32 ROT, XPV, YPV Specify an angle for plot rotation and X and Y coordinates of the pivot point.
- 33 SPB Code value to permit or suppress plotting of the plot border.
- 34 ORG Specifies a diagonal displacement of the plot origin, $(I,J) = (1,1)$, up and to the right.
- 35 CTX Specifies a base for selection of contour intervals exponentially.
- 36 VEC Controls the length of arrows and plotting of magnitudes on vector (wind) plots.
- 37-38 XLP, YLP Specify starting points for plotting labels along the X and Y axes, respectively.
- 39-40 XLD, YLD Specify the number of decimal places to be plotted in labels along the X and Y axes, respectively.
- 41 DPL Specifies the number of decimal places to be plotted in data or grid values when XYDATAPLT or DIGITPLT are selected.
- 42 SEG Threshold count for coordinates in contour segments. Those with fewer points than specified by SEG will not be plotted.

APPENDIX B. Plot parameters for CALLPSUB

CALLPSUB is a procedure which creates the file of generic plot calls from parameters and specifications generated by the primary plot routines.

CALLPSUB constructs plot calls for five basic plot functions:

ORIGIN, PLOT, SYMBOL, NUMBER and LINE.

The parameters passed to CALLPSUB are described below.

PNAM Plot call type with one of five values: ORIG, PLOT, SYMB, NUMB or LINE.

X,Y Floating point arrays with index ranges of 1..240, containing plot coordinates in inches. The first element in each array, X[1],Y[1], are the move (pen up) coordinates for all plot commands. The remaining elements contain point coordinates for calls to LINE.

PN Type integer array with index range 1..240. The contents provide pen control for PLOT and LINE plot calls. The first element, PN[1], specifies length of a string parameter for SYMB and number of decimal places for NUMB.

HT Specifies height of characters for SYMB or NUMB. Specifies an array index value for LINE indicating the from position in arrays X and Y of a contour segment.

NUM The number to be plotted with plot type NUMB, or a symbol code for SYMB. Specifies an array index value for LINE indicating the to position in arrays X and Y of a contour segment.

STR The string to be plotted with SYMB. The length of the string is contained in PN[1].

THETA Specifies a rotation for plotting strings or numbers, in degrees (positive counterclockwise). The pivot point for rotation is the lower left corner of the first character in the string or number. The coordinates of this point are given by X[1],Y[1].

The plot calls constructed by CALLPSUB are described below. Each call starts with the keyword that identifies it and, except for LINE, is completed on one line. All coordinates are given in inches from the grid origin, except those submitted with ORIG. They specify the offset (in inches) of the grid origin with respect to the origin of the plotter or screen.

ORIG X Y WIDTH HEIGHT CIN4 COG4

Sets the grid origin relative to the plot origin.

X is the right offset of grid origin

Y is the upward offset of grid origin

WIDTH is the width of the plot in inches.

HEIGHT is the height of the plot in inches.

CIN4 is the scaled value of the contour interval $CIN * FTR$.

COG4 is the scaled value of contour origin $(COG * FTR)$.

WIDTH and HEIGHT may be reduced by DSPLPLOT if necessary to keep the plot inside the screen window.

CIN4 and COG4 are special parameters used by the paint option in DSPLPLOT to paint areas between contours on contour plots.

PLOT X Y PN

Moves pen to coordinates (X,Y)

PN = 2 means pen down

PN = 3 means pen up.

SYMB X Y PN HT THETA STR

Plots a string.

X and Y are coordinates of the lower, left corner of the first character.

PN = 0 means a single character will be plotted. The ASCII code for the character is given by THETA. The string parameter, STR will be empty.

PN = 1..72 is the number of characters to be plotted.

HT is the height of the characters, in inches.

THETA is the rotation of the string in degrees (except when PN = 0). Rotation is counterclockwise when THETA is positive. The pivot point for rotation is given by X and Y.

STR is the string to be plotted. The length can be up to 72 characters.

NUMB X Y PN HT THETA NUM

Plots a number

X and Y are coordinates of the lower, left corner of the first digit in NUM.

PN is the number of decimal places to be plotted in the number. If PN = 0 the decimal point will be suppressed.

HT is the height of the digits, in inches.

THETA is the rotation of the number, in degrees. Rotation is counterclockwise

when THETA is positive. The pivot point for rotation is given by X and Y.
NUM is the number to be plotted.

LINE N CNTR

Plots a connected sequence of line segments.

X1 Y1

X2 Y2

X3 Y3

. .

. .

XN YN

N is the number of points in the sequence.

CNTR is a contour value or segment identifier.

X1,Y1..XN,YN are the coordinates of the points.

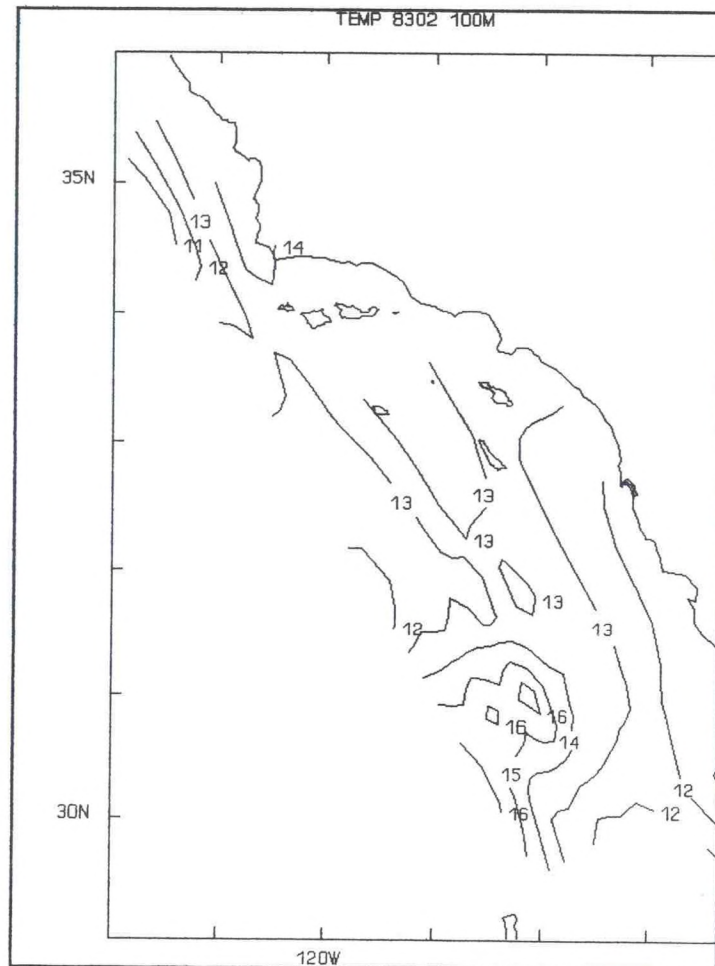
The first record in a plot file will contain the ORIG statement. Otherwise, the plot calls may appear in any order. The maximum number of points in a call to LINE is 240. Contour labels (CNTR) are entered as NUM parameters in plot calls to NUMB at run time.

APPENDIX C. Examples of plots and associated command files

The examples presented here demonstrate some of the plot capabilities of EDMAP. Most of them were created from data in the CalCOFI oceanographic data base. The plot files created by EDMAP were converted to Hewlett Packard (HP7470A) code and submitted to Word Perfect 5.0 for printing.

Example 1. Temperature distribution at 100 meters in Feb. 1983. The coastline and island features were taken from an overlay file. The command contains two sets of run parameters. In the first run, the temperature data were analyzed with CalCOFI (line and station) coordinates and the contours computed with those coordinates. In the second run, the coordinates were transformed to latitude and longitude for plotting.

```
#
LOADSORC/OCNDAT(TMP DISTP DISTN 8302 100 03)
ANALYZE/CENTER/SETNUL
TRACKCNTRS
#
XZO=60 YZO=110 XMX=30 YMX=70 XGD=2.5 YGD=3.3333
SCL=3 PJC=5 ANL=8 NUL=99
COG=0 CIN=1 CTX=0 CTL=1
SQD=2 SYM=0
#
OCNDAT=CC8302.OCN
IJVAL=CC8302.IJV
CTRFIL=CC8302.CTR
#
TEMP 8302 100M
OVERLAYPLT
PLOT CNTRS
#
XZO=122W YZO=29N XMX=116W
YMX=36N XGD=1 YGD=1
SCL=3 PJC=1 ANL=4 NUL=99
COG=0 CIN=1 CTX=0 CTL=1
ORG=1.0 HGT=0.1 SPB=0
SEG=1 SQD=2
XTC=1 YTC=1 XLB=5 YLB=5
XLP=2 YLP=1 XLD=0 YLD=0
MFX=.75 MFY=.75 ROT=30
XPV=121.15 YPV=34.15
#
CTRFIL=CC8302.CTR
CSTFIL=PACOAST.OVL
PLTFIL=CC8302.PLT
#
QUIT
```

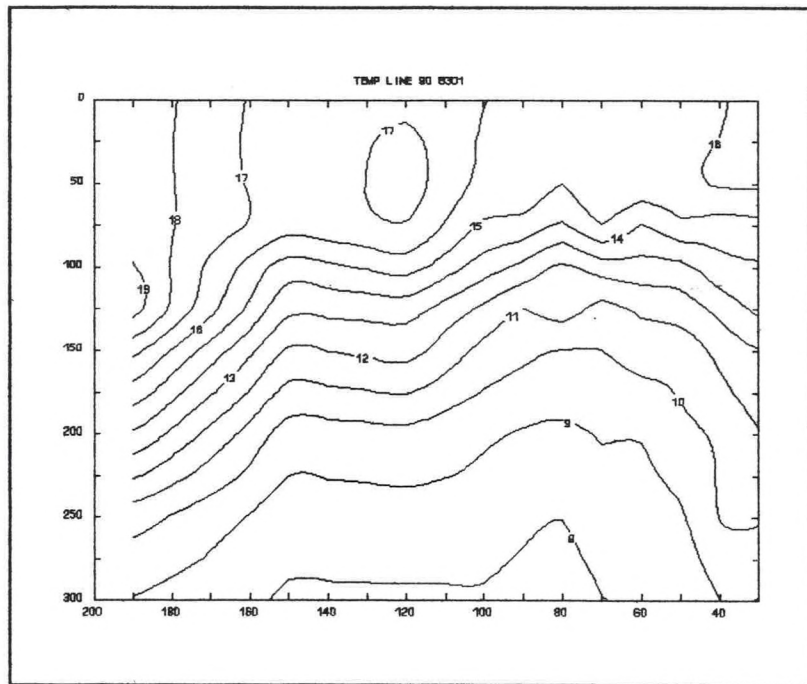


Example 2. Distribution of temperature with depth along CalCOFI line 90 in Jan. 1983. Distance offshore increases from 10 naut. miles at the right border to 690 naut. miles at the left border. Depths plotted along the left border are given in meters.

```

TEMP LINE 90 8301
LOADSORC/OCNDAT(TMP
DEPTH DISTN 8301 90000)
ANALYZE/CENTER/SETNUL
TRACKCNTRS
PLOTCNTRS
#
XZO = 200  YZO = 300
XMX=30 YMX=0 XGD=10
YGD=25
SCL=3  PJC=4  ANL=4
NUL=99
COG=0  CIN=1  CTX=0
CTL=1 SQD=6
SYM=0  SEG=1  ORG=1.0
HGT=0.1 SPB=0
XTC=1  YTC=1  XLB=2
YLB=2
XLP=0  YLP=0  XLD=0
YLD=0
MFX=0.047058 MFY=.02
#
OCNDAT=CC1983.SUB
IJVAL=CC1983.IJV
CTRFIL=CC1983.CTR
PLTFIL=CC1983.PLT
#
QUIT

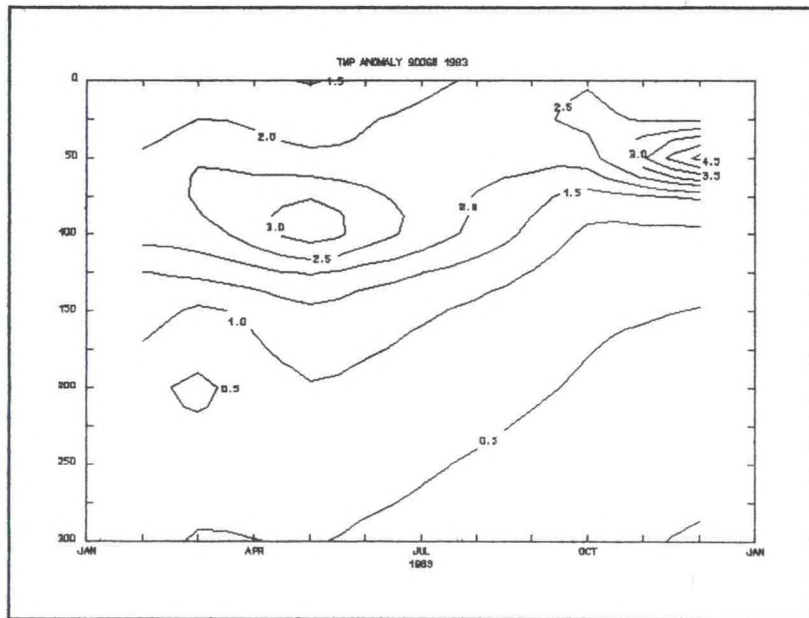
```



Example 3. Time section of temperature anomaly at CalCOFI station 90060 from 0 to 300 meters during 1983. Month and year labels along the bottom show the result of using /YRMO(LTRS) with TIMEPLT.

```
TMP ANOMALY 90060 1983
LOADSORC/OCNDAT(TMP DEPTH TIME 9099 90060)
TIMEPLT/YRMO(LTRS)
ANALYZE/CENTER/SETNUL
TRACKCNTRS
PLOTCNTRS
```

```
#
XZO=8301  YZO=300
XMX=8401  YMX=0  XGD=1
YGD=25
SCL=3  PJC=4  ANL=5
NUL=9
COG=-6  CIN=0.5  CTX=0
CTL=1  SQD=2
SYM=0  SEG=1  ORG=1.0
HGT=0.1  SPB=0
XTC=1  YTC=1  XLB=3
YLB=2
XLP=0  YLP=0  XLD=0
YLD=0
```



```
MFX=.7272727  MFY=.02
#
OCNDAT=CC1983.DEV
IJVAL=CC1983.IJV
CTRFIL=CC1983.CTR
PLTFIL=CC1983.PLT
#
QUIT
```

Example 4. Temperature profiles from 0 to 300 meters at CalCOFI station 90060 for Feb. 1983 (right profile) and the Feb. long term mean (left profile). Word perfect failed to reproduce the (+) and (o) symbols which mark standard depths on the profiles, possibly because they were designated to be plotted at half height.

```
#
LOADSORC/OCNDAT(TMP DEPTH TMP 9099 090060 01)
```

```
#
XZO=5 YZO=300 XMX=20 YMX=0 XGD=1 YGD=25
SCL=3 PJC=5 ANL=4 NUL=0
COG=0 CIN=1 CTX=0 SQD=2
SYM=111 DPL=0
```

```
#
OCNDAT=90060.MEA
IJVAL=CCMEATS.001
```

```
#
TMP 90060 NORMAL (o)
8302 (+)
```

```
LOADSORC/OCNDAT(TMP
DEPTH TMP 8302 090060
01)
```

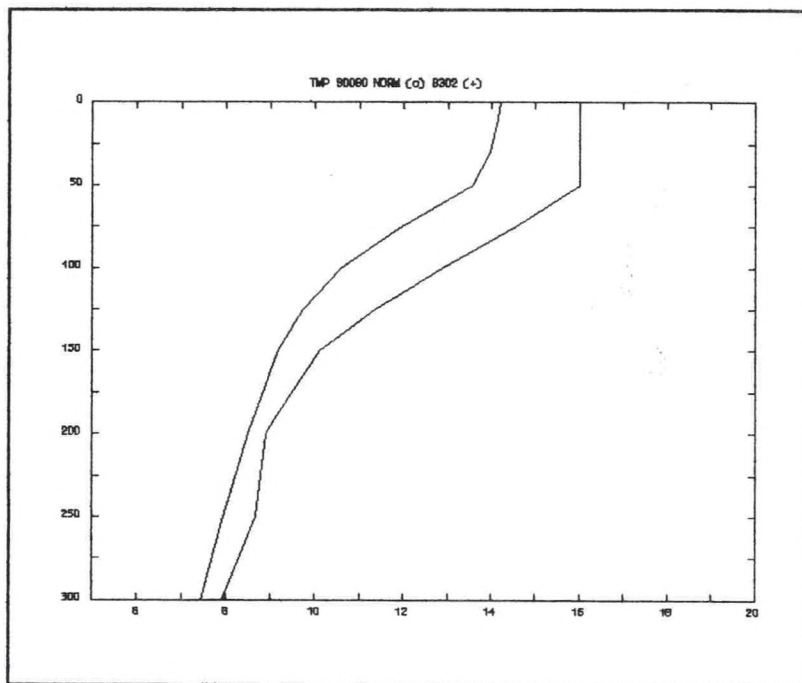
```
XYDATAPLT/CONNECT
CONCATPLT/DATPLT
```

```
#
XZO=5 YZO=300 XMX=20
YMX=0 XGD=1 YGD=25
SCL=.3 PJC=5 ANL=4
NUL=0
COG=0 CIN=1 CTX=0
SQD=2
SYM=43 DPL=0
```

```
ORG=1.0 HGT=0.1 SPB=0
XTC=1 YTC=1 XLB=2 YLB=2
XLP=1 YLP=0 XLD=0 YLD=0
MFX=.5333334 MFY=.02
```

```
#
OCNDAT=CC1983.SUB
IJVAL=CC83TS.IJV
CONCAT=CCMEATS.001
PLTFIL=CC83TS.PLT
```

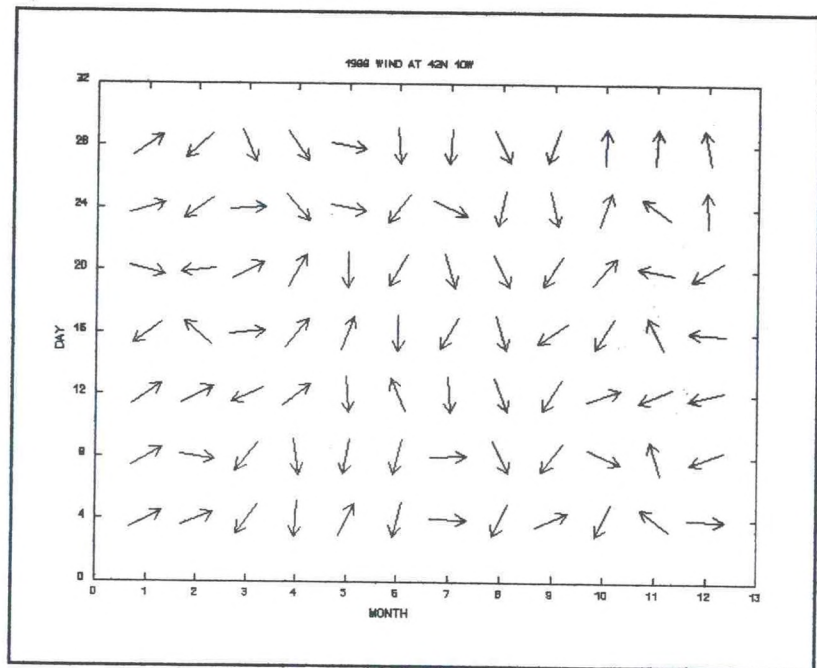
```
#
QUIT
```



Example 5. Vector plot of geostrophic wind at a location off northern Portugal during 1988. Each arrow represents a four day average of six hourly values. Wind directions, in the source file, are given in degrees clockwise from north. The overlay option was used to plot the captions along the left and lower borders.

```
1988 WIND AT 42N 10W
LOADSORC/XYDAT(VE 6 1 2 5 )
ANALYZE/CENTER/SETNUL
SAVEFLD
OVERLAYPLT
VECTORPLT
```

```
#
XZO=0 YZO=0 XMX=13
YMX=32 XGD=1 YGD=4
SCL=1 PJC=5 ANL=4
NUL=-1
COG=0 CIN=1 CTX=0
SQD=2
SYM=0 DPL=0 VEC=0.7
ORG=1.0 HGT=0.1 SPB=0
SEG=1
XTC=1 YTC=1 XLB=1
YLB=1
XLP=0 YLP=0 XLD=0
YLD=0
MFX=.6153846 MFY=.1875
```

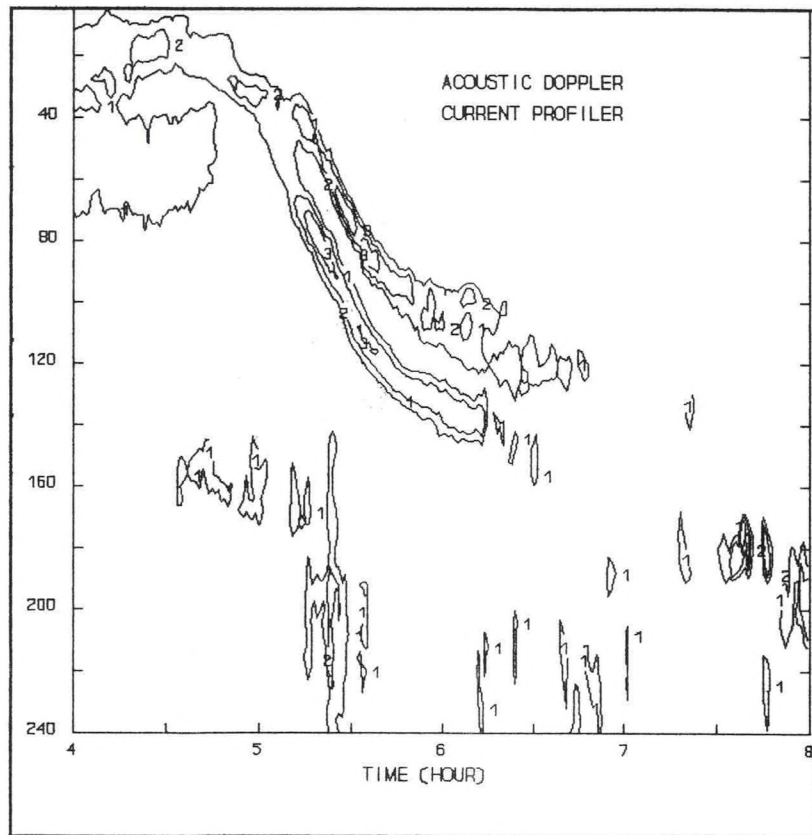


```
#
XYDAT=WIND88.DAT
IJVAL=WIND88.IJV
GRDFIL=WIND88.GRD
CSTFIL=WIND88.OVL
PLTFIL=WIND88.PLT
```

```
#
QUIT
```

Example 6. Data from an acoustic doppler current profiler processed by a spreadsheet program. Labels along the left border indicate depths in meters. In the spreadsheet source file, rows represented time at one minute intervals and columns represented depth at four meter intervals. The exchange rows and columns parameter, EXY, was used for this plot. The overlay option was used to plot the captions below the lower border and in the upper right portion of the figure.

```
#
LOADSORC/RCDAT(ROCO 6 60 240 EXY )
ANALYZE/CENTER/SETNUL
SMOOTH
TRACKCNTRS
OVERLAYPLT
PLOTCNTRS
#
XZO=4 YZO=240 XMX=8
YMX=0 XGD=0.016667
YGD=4
SCL=3 PJC=5 ANL=4
NUL=99 SMH=10
COG=1 CIN=1 CTX=0
CTL=1 SQD=1
SYM=0 ORG=1.25
HGT=0.1 SPB=0 SEG=12
XTC=30 YTC=5 XLB=60
YLB=10
XLP=0 YLP=0 XLD=0
YLD=0
MFX=1.5 MFY=.025
#
XYDAT=a0704n.asc
IJVAL=a0704n.IJV
CTRFIL=a0704n.CTR
CSTFIL=A0704N.OVL
PLTFIL=a0704n.PLT
#
QUIT
```



Example 7. Mathematical function plot (archimedian spirals). The source file contains coordinates computed for successive points along each curve. The curves were drawn using the CONNECT qualifier with the XYDATAPLT plot option.

```
SPIRAL
LOADSORC/XYDAT(DA 5 4 5 1)
XYDATAPLT/CONNECT
#
XZO=-6 YZO=-6 XMX=6 YMX=6 XGD=1 YGD=1
SCL=3 PJC=5 ANL=4 NUL=99
COG=0 CIN=1 CTX=0
SQD=2
SYM=46 DPL=0
ORG=1.0 HGT=0.1 SPB=0
XTC=1 YTC=1 XLB=2
YLB=2
XLP=0 YLP=0 XLD=0
YLD=0
MFX=.5 MFY=.5
#
XYDAT=SPIRALC.DAT
IJVAL=SPIRAL.IJV
PLTFIL=SPIRAL.PLT
#
QUIT
```

