DEPARTMENT OF THE INTERIOR

U.S. GEOLOGICAL SURVEY

SAUDI ARABIAN MISSION

MISCELLANEOUS DOCUMENT 24

(INTERAGENCY REPORT 358)

PROGRAM PDP003: CTRL2D

by

M. M. Donzeau and M. E. Gettings

Open-File Report 81-1152

U.S. Geological Survey

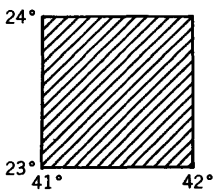
Jiddah, Saudi Arabia

1981

The work on which this report was based was performed in accordance with a cooperative agreement between the U.S. Geological Survey and the Ministry of Petroleum and Mineral Resources.

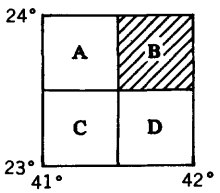
The report has not been edited or reviewed for conformity with U.S. Geological Survey standards and nomenclature. Product names used in this report are for descriptive purposes and in no way imply endorsement by the U.S. Geological Survey.

The quadrangle identification method used in U.S. Geological Survey Saudi Arabian Mission reports is shown below.



23/41

1-degree quadrangle



23/41 B

30-minute quadrangle

CONTENTS

	Page
ABSTRACT	1
SOURCE DECK LOCATION AND UPDATE STATUS	1
DESCRIPTION OF COMPUTED QUANTITIES	2
DESCRIPTION OF THE PROGRAM	7
Files used in the program	7
Control files	8
Control cards of user's control file	8
Control cards of new control file	15
Data files	20
User's input data file	20
New input data file	20
Program structure	21
Size of the program	37
PROGRAM OPERATION	38
Error messages on terminal	39
Error messages on printer file	40
EXAMPLE	43
SOURCE PROGRAM LISTING	49
REFERENCES CITED	80

ILLUSTRATIONS

Figure 1.	Structure of the control files	9
2-12.	Flow charts showing:	
2.	CRTL2D program	22
3.	CREATE subprogram	23
4.	EDITF subprogram	24
5.	READAL subprogram	27
6.	WRITAL subprogram	28
7.	WRIONE subprogram	30
8.	WRISEV subprogram	31
9.	RWDATP subprogram	33
10.	RWDATP subprogram entry RWDATG	34
11.	WRIPRE subprogram	35
12.	WRIPRE subprogram entry WRIGRI	36

(v follows)

Program Documentation Manual

Program Number	:	PDP003
Program Name	:	CTRL2D
Programmer	:	M. M. Donzeau and M. E. Gettings
Language	:	FORTRAN IV PLUS
Computer/System	:	PDP-11/45
Date	:	13 March 1979
Institution	:	U.S. Geological Survey Saudi Arabian Mission
Location	:	Jiddah, Saudi Arabia

PROGRAM PDP003:CRTL2D

by

M. M. Donzeau and M. E. Gettings

ABSTRACT

This program is designed to build a control file that contains all the control cards necessary to run the program CONTUR and to build a corresponding input data file.

In the first part of the program, a user's control file is created or edited by entering information through the keyboard. In the second part, the user's control file and the user's input data file are checked to assure that every desired map is acceptable by the program CONTUR. If the limits in the Y-direction exceed the plotter limit (27 inches), or if the data array to read or to be created is larger than the maximum size allowed by the program CONTUR (2500 points), the map will be divided in several parts. For each such part, the program CTRL2D will generate a series of control cards for the new control file and will write a smaller data set corresponding to the part of the map in the new input data file. If a map is acceptable in full, the corresponding control cards will be copied from the user's control file to the new control file and the corresponding data set from the user's input data file to the new input data file.

The user's input data file must be built before running the program CTRL2D.

SOURCE DECK LOCATION AND UPDATE STATUS

The FORTRAN source program is divided into three overlays: CTRL20.FTN, CTRL21.FTN, CTRL22.FTN. These three overlays and the task image CTRL2D.TSK are on the system disk of the DGMR PDP-11/45 computer under the UIC [22,50].

No updates have been made to the program at this time.

DESCRIPTION OF COMPUTED QUANTITIES

There are no quantities computed during the building and editing of the user's control file; a number of quantities are computed while building the new control and input data files. A description of the computed quantities follows.

1. FMAPX2 of first DRAW card (page 19)

This parameter corresponds to one part of a map and is only calculated when the input array or the array to be gridded is greater than the maximum size allowed by the program CONTUR.

The parameter NNX (number of columns) is labeled NNX2 for one part of a map, and FMAPX2 is calculated in subprogram WRISEV as follows.

$$FMAPX2 = (NNX2-1) * \frac{FMAPX}{NNX-1}$$

2. FMAPY2 of first DRAW card (page 19)

This parameter corresponds to one part of a map and is calculated either if the height of map is greater than the plotter height, or if the input array or the array to be created is greater than the maximum size allowed by the program CONTUR. The calculation takes place in subprogram WRISEV.

If the height of a map is greater than the plotter height, the first part of the map has a height FMAPY2 equal to the plotter height. The second part has a height calculated as follows.

$$FMAPY2 = (NNY2-1) * \frac{FMAPY}{NNY-1} \qquad (1)$$

where NNY2 is the number of remaining rows in the map. If this value of FMAPY2 is greater than the plotter height, the second part of the map will have a height equal to the plotter height, and the third part a height as calculated in equation (1) with the new value of NNY2, and so forth.

If the input array or the array to be created is greater than the maximum size allowed by the program CONTUR, the number of rows NNY2 is calculated for each part of map and the height of each part is calculated with equation (1).

3. IBXOPT of first DRAW card (page 19)

For multipart maps this parameter is increased by 2. Each part of the map will be drawn by the program CONTUR using an array smaller than the original by one row and one column on each side. This allows the contours to be continuous without sharp angles from one part to the next. The value IBXOPT = 0 should be avoided if the map is plotted in several parts.

4. XORIG, YORIG of second DRAW card (page 20)

These parameters are always calculated, since there is only one origin for all the maps.

If there is a series of maps to be drawn, each map in one part only, the maps are plotted one above the other beginning at the origin down left, until the upper part of the plotter is reached. (Each map is separated vertically from the other by two inches.) The next map is plotted starting at the lower part of the plotter, two inches to the right of the widest map of the first set. The next map is plotted above that map, and so on. The coordinates of the origin of each map, XORIG and YORIG, are calculated in the subprogram WRIONE.

If a map is in several parts, the coordinates of the origin are calculated in subprogram WRISEV so that one part is drawn next to the previous one, thus composing the desired map. As first row is always at top, the first part of the map is drawn starting at the upper part of the plotter.

5. FRMT2 of second FMT card (page 21)

The format is always calculated for pregridded data. If the number of data per record is NCOL, the following parameters are calculated.

```
NCARD = NCOL/6
NSUP  = NCOL-NCARD*6
```

If the parameter NSUP is not null, the format is defined as

(<NCARD> (6E13.7), <NSUP> E13.7).

If the parameter NSUP is null, NCARD is decreased by 1, NSUP is set to the value 6, and the format is defined by the same expression. NCARD and NSUP are coded in the format, using an ENCODE statement, in subprogram RWDATP.

NCOL is equal to NNY2 (number of rows for one part of a map) if the data are read in by columns, or to NNX2 (number of columns for one part of a map) if the data are read in by rows.

For ungridded data, the format is fixed to

(A4, 6X, 3(E13.7,1X)).

6. NPTS2 of GRID card (page 22)

The program first checks if the parameter NPT is negative, null, or positive in the subprogram WRITAL.

If NPT is negative, the type of data (pregridded or ungridded) that were previously stored on the intermediary file is determined by reading the first record of that file. If the data previously stored were of pregridded type, or if there were no previous data, the program tries to read the next data set from the input data file. If the data were of ungridded type, the number of input data points of the previous data set is given in the first record of the intermediary file and NPT is set to this value. The program then follows the same procedure as in the case of NPT equal to positive.

If NPT is null, the program reads the whole data set from the user's input data file and counts the number of (X,Y,Z) data points preceding the word "END" in the alphanumeric field or preceding the end of file. This number will become the value for NPT and the program then follows the same procedure as in the case of NPT positive.

If NPT is positive, the program checks in the subprogram RWDATG to see if NPT is below or over an arbitrarily fixed value (this value is presently equal to 2000). If NPT is below this limit, the whole data set of (X,Y,Z) input data points is copied onto the new input data file and NPTS2 is set to the value of NPT. If NPT is over the limit, the program checks to see if the X-value of each input data point is inside the limits X1, X2, and if the Y value is inside the limits Y1, Y2, defined as follows.

```
X1 = XMPMN2 - (XMPMX2 - XMPMN2) * 0.25
X2 = XMPMX2 + (XMPMX2 - XMPMN2) * 0.25
Y1 = YMPMN2 - (YMPMX2 - YMPMN2) * 0.25
Y2 = YMPMX2 + (YMPMX2 - YMPMN2) * 0.25
```

The program then copies the input data points which are inside these limits onto the new input data file. The number of such points is NPTS2.

7. NX2 of GRID card, NNX2 of RDGD card (pages 22 and 23)

These two parameters are related by the formula NNX2 = NX2+1. The parameter NNX2 is for one part of a map and is only calculated when the input array or the array to be created is greater than the maximum size allowed by the program CONTUR. If NNX is less than or equal to 75, NNX2 is equal to NNX. If NNX is greater than 75, the first part of the map will have a number of columns NNX2 equal to 50, the second part will have the remaining number of columns, unless it is greater than 75. In such a case, the second part will also have NNX2 equal to 50, and the third part the remaining number of columns, and so on. This calculation is performed in subprogram WRISEV.

8. NY2 of GRID card, NNY2 of RDGD card (pages 22 and 23)

These two parameters are related by the formula NNY2=NY2+1. The parameter NNY2 is for one part of a map and is calculated in subprogram WRISEV either if the height of the map is greater than the plotter height, or if the input array or the array to be created is bigger than the maximum size allowed by the program CONTUR.

When the height of a map is greater than the plotter height, the first part of the map has a number of rows equal to

$$NNY2 = \text{plotter height} * \frac{NNY-1}{FMAPY}. \qquad (2)$$

The second part of the map is set with NNY2 equal to the remaining number of rows, unless the corresponding height is greater than the plotter height. If that is the case, the second part of the map is set up with NNY2 as given in equation (2), and the third part with NNY2 equal to the remaining number of rows, and so on.

When the input array or the array to be created is greater than the maximum size allowed by the program CONTUR (MAXPGD), the number of columns NNX2 is calculated first, then the number of rows NNY2 is given by the equation

$$NNY2 = \frac{MAXPGD}{NNX2}.$$

9. XMPMN2, XMPMX2, YMPMN2, YMPMX2 of REGN card (pages 23 and 24)

If the REGN card is not supplied for an ungridded data set, the parameters XMPMN, XMPMX, YMPMN, and YMPMX are calculated in subprogram WRITAL. In this subprogram the entire data set is read from the intermediary file, and comparison statements allow one to find minima and maxima of X- and Y-values.

The parameters XMPMN2, XMPMX2, YMPMN2, and YMPMX2 are fixed for one part of a map and are calculated in subprogram RWDATG from the values of NX10, NX20, NY10, NY20. XMPMN2 is fixed by the equation

$$XMPMN2 = XMPMN + (NX10-1)* \frac{XMPMX - XMPMN}{NX}.$$

XMPMX2 is calculated by the same equation, replacing NX10 by NX20. YMPMN2 is fixed by the equation

$$YMPMN2 = YMPMX - (NY20 - 1)* \frac{YMPMX - YMPMN}{NY}.$$

YMPMX2 is calculated by the same equation, by replacing NY20 by NY10.

10. NX10, NX20, NY10, NY20

These quantities are used to write pregridded data on the new input data file or to calculate the X- and Y- limits of an ungridded data set. They are calculated in subprogram WRISEV.

If a map is in one part, the value of these four quantities are 1,NNX,1,NNY, respectively. If a map is in several parts, these quantities are calculated for each part from the values of NNX2 and NNY2, as follows.

```
NNX2 = NX20-NX10 + 1
NNY2 = NY20-NY10 + 1
```

The entire pregridded data set is read in from the intermediary file, and only the values of the array from columns NX10 to NX20 and from rows NY10 to NY20 are written on the new input file in subprogram RWDATP.

For ungridded data, the quantities NX10, NX20, NY20, and NY10 permit the calculation of XMPMN2, XMPMX2, YMPMN2, and YMPMX2, respectively.

DESCRIPTION OF THE PROGRAM

The files and contents of files used in the program are described first, then the structure of the program.

Files used in the program

One or two input files are necessary to run CTRL2D.

1. The user's control file, the name of which is supplied by the user, is needed in the case of the edit option only. It may have been created by a previous run of CTRL2D or by the EDI utility program.

2. The user's input data file, the name of which is supplied by the user, must have been built before running CTRL2D. It may contain several data sets; each set could be one of two kinds: (X, Y, Z) points from which a grid will be created in program CONTUR, or the Z-values of a rectangular array of pregridded data.

There are four output files for the program CTRL2D.

1. The user's control file, the name of which is supplied by the user, is created or edited by the program. It contains the control cards as they are described on pages 12 to 18.

2. The new control file, which is named CONTUR.CTR, contains the control cards as described on pages 12 and 19 to 24. During program execution each control card set of the user's control file is checked to see if the desired map can be drawn in one or several parts. A few calculations are performed, and then the control cards of the new control file contain the new values of the parameters. This file is ready to be used by the program CONTUR.

3. The new input data file is named DB1:CONTUR.DAT. For maps that must be drawn in several parts, one data set of the user's input data file will correspond to several data sets of the new input data file. The contents of this new input data file are described on page 24.

4. The printer file is named "CTRL2D.PNT" and shows the contents of both control files.

The input and output files are all sequential-formatted files with a variable record length.

There is one temporary file, which is created and deleted at each run. It is an unformatted direct-access file, named the intermediary file, and contains one data set at a time.

Control files

The structure of both control files is shown in figure 1. The five first records of each control file are as follows:

1. Input data filename (32 characters): this is the user's input data filename in the user's control file. It is set to "DB1:CONTUR.DAT" in the new control file.

2. Printer filename (32 characters): this file will be built by a run of CONTUR and will contain information about CONTUR program execution.

3. Output grid filename (32 characters): this file will be output from the program CONTUR if griddings are required and PNCH cards were specified in the control file.

4. Plotter filename (32 characters): this file will be output from the program CONTUR.

5. General title: this is a 40-character title that will be plotted by the program CONTUR at a direction of 90° before all other plots.

Note that if the user's control file is created with the utility program EDI, the last character of a filename must be a null character.

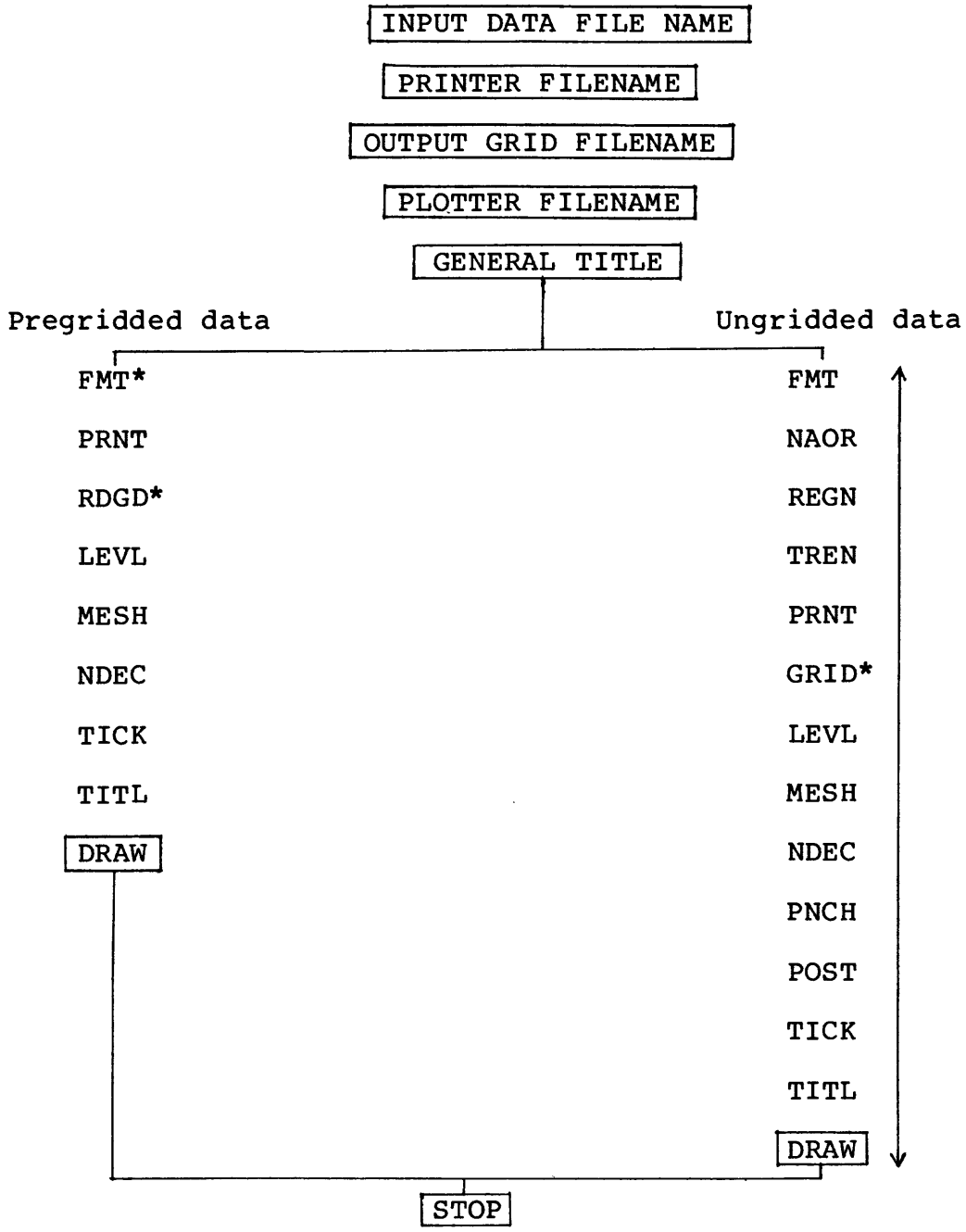
Following is a description of the other control cards, describing the parameters and the format used to write them, first for the user's control file, then for the new control file.

Control cards of user's control file.--Most of the control cards are identical to those described in the CONTOUR Applications Software manual (CALCOMP, 1974) and in the documentation of program CONTUR (PDP004) (Donzeau and others, 1981).

The formats described are those used by the program CTRL2D to write the control cards.

The format used to write most of the variables of the type REAL depends on the values of the variables. The field width is always 10 and the format is one of the following: F10.0, F10.1, ..., F10.8, or E10.4. This type of variable format is called FVAR in the following description.

For those control cards that are identical to the ones described in the documentation of program CONTUR (GRID, LEVL, MESH, NAOR, NDEC, PNCH, POST, PRNT, RDGD, REGN, STOP, TICK, TITL, and TREN) only the formats used to write them are indicated.



* This card must be present in the first control card set.

Figure 1. Structure of the control data file. Use as many control card sets of either type as the number of desired maps, boxed cards are essential.

DRAW card

It consists of two cards and initiates the drawing of a map.

First DRAW card

Variable	Parameter	Description
1	TYPE	DRAW
2	FMAPX	width of map to be drawn, in inches (default = 10)
3	FMAPY	height of map to be drawn, in inches (default = 10)
4	SN	number of divisions, in the X direction, that each grid cell is to be divided into (default = 4, max = 10)
5	SM	number of divisions, in the Y direction, that each grid cell is to be divided into (default = 4, max = 10)
6	IBXOPT	boundary parameter = 0, draw box boundary = 1, put a "+" at each corner, no boundary drawn (default = 0)
Format:		(A4, '1', 5X, 2"FVAR", 2 F10.0, I10)

Second DRAW card

Variable	Parameter	Description
1	TYPE	DRAW
2	ZBLANK	Z-value given to blank areas (where there will be no contours). If there is no blank area, enter a Z-value different from all those of the data set. For ungridded data, enter the value -1.E35. The program CONTUR wil assign this value of -1.E35 to points that cannot be defined because there are not enough neighbors around. However, the program CONTUR will take those points into account for the contouring if the user has not set the parameter ZBLANK to the value -1.E35.
Format:		(A4, '2', 5X, "FVAR")

FMT card

It consists of one card containing the TYPE parameter and one card containing the format. It describes the format used to read the user's input data file. This format remains in effect until changed by another FMT card.

First FMT card

Variable	Parameter	Description
1	TYPE	FMT
Format		(A4)

Second FMT card

1	FRMT	execution-time format statement that describes data to be read in - for pregridded data, the FMT card must be supplied, at least with the first map. The format should uniquely describe one row or column of the grid - for ungridded data, the format should uniquely describe one data point by its alphanumeric field and its three coordinates default = previous format, or (A4, 6X, 3F10.3)
Format:		(20A4)

GRID card

Format: (A4, 6X, 6I10)

LEVL card

Format:
- with a constant contour interval:
  (A4, 6X, 3"FVAR")
- with contour levels defined one by one:
  (A4, 6X, "FVAR", 10X, "FVAR")

If this card is omitted for one data set, no action is taken by program CTRL2D.

MESH card

Format: (A4, 6X, F10.4, I10)

NAOR card

Format: (A4, 6X, 2I10, "FVAR")

NDEC card

Format: (A4, 6X, I10)

PNCH card

Format: (A4)

POST card

Format: (A4, 6X, F10.4, I10)

PRNT card

Format: (A4, 6X, I10)

RDGD card

Format: (A4, 6X, 3I10)

REGN card

Format: (A4, 6X, 4"FVAR")

STOP card

Format: (A4)

TICK card

Format: (A4)

TITL card

Format: first TITL card: (A4)
second TITL card: (20A4)

Note: It is preferable to avoid a TITL card in case of a multipart map.

TREN card

Format: (A4)

Control cards of new control file.--The new control file is named CONTUR.CTR.

The cards which are identical to those of the user's control file are: LEVL, MESH, NAOR, NDEC, PNCH, POST, PRNT, STOP, TICK, TITL, and TREN.

The cards which are different from those of the user's control file are DRAW, FMT, GRID, RDGD, and REGN. The following is a description of these cards and the format used to write them.

Note that in the case of ungridded data the REGN card is always present in the new control file.

DRAW card

It consists of two cards and initiates the drawing of a (part of) map.

First DRAW card

Variable	Parameter	Description
1	TYPE	DRAW
2	FMAPX2	width of (part of) map to be drawn, in inches
3	FMAPY2	height of (part of) map to be drawn, in inches
4	SN	number of divisions, in the X direction, that each grid cell is to be divided into. Same value as in the user's control file
5	SM	number of divisions, in the Y direction, that each grid cell is to be divided into. Same value as in the user's control file
6	IBXOPT	boundary parameter = 0, draw box boundary (only for map in one part) = 1, put a "+" at each external corner, no boundary drawn (only for map in one part)

= 2, draw box boundary, and put a "+" at each internal corner (for one part of a multipart map)
= 3, put a "+" at each internal and external corner, no boundary drawn (for one part of a multipart map).

Format: (A4, '1', 5X, 2"FVAR", 2F10.0,I10)

Second DRAW card

Variable	Parameter	Description
1	TYPE	DRAW
2	ZBLANK	Z-value given to blank areas (where there will be no contours). Same value as in the user's control file
3	XORIG	X-coordinate of down-left corner of boundary of (part of) map, in inches
4	YORIG	Y-coordinate of down-left corner of boundary of (part of) map, in inches

Format: (A4, '2', 5X, 3"FVAR")

Number of DRAW cards: For each DRAW card of user's control file, there is one DRAW card in the new control file if the desired map is in one part and several DRAW cards if the desired map is in several parts.

Note: An external corner is one of the corners of the area defined by the parameters FMAPX2, FMAPY2. An internal corner is inside this area, at a distance of one row and column from the corresponding external corner.

FMT card

It consists of one card containing the TYPE parameter and one card containing the format. This card describes the format used to read the new input data file.

First FMT card

Variable	Parameter	Description
1	TYPE	FMT
Format:		(A4)

Second FMT card

1	FRMT2	execution-time format statement that describes the data to be read in by program CONTUR - pregridded data, = (<NCARD> (6E13.7), <NSUP>E13.7) (see description of NCARD, NSUP page 6) - ungridded data, = (A4, 6X, 3(E13.7,1X))
Format:		(20A4)
Number of FMT cards:		There is one FMT card in the new control file for each RDGD card (pregridded data) or GRID card (ungridded data) of new control file.

GRID card

This card initiates the creation of a grid for a (part of) map, and specifies that ungridded data points are to be read in.

Variable	Parameter	Description
1	TYPE	GRID
2	NPTS2	always positive, indicates that there are NPTS2 data records on new input data file
3	NX2	number of grid divisions in the X-direction that the (part of) map is to be divided into
4	NY2	number of grid divisions in the Y-direction that the (part of) map is to be divided into
5	IXPOS	= 1
6	IYPOS	= 2
7	IZPOS	= 3

Format: (A4, 6X, 6I10)

Number of GRID cards: 1) For each GRID card in the user's control file there is one GRID card in the new control file, if the desired map is in one part, and several GRID cards if the desired map is in several parts.
2) For each DRAW card without a GRID card in the user's control file, there is no GRID card in the new control file if both previous and actual maps are in one part. There is one GRID card if previous map is in several parts and actual map is in one part. There are several GRID cards if actual map is in several parts.

RDGD card

This card informs the program CONTUR that a pregridded Z-value array (part of) is to be read in and gives the dimensions of this array (part of).

Variable	Parameter	Description
1	TYPE	RDGD
2	NNX2	number of columns in the (part of) data array to be read in
3	NNY2	number of rows in the (part of) data array to be read in
4	MANNR	flag to tell how data is to be read in. = 1, read data in by columns, top to bottom, = 2, read data in by rows, left to right. same value as in the user's control file.

Format: (A4, 6X, 3I10)

Number of RDGD cards: 1) For each RDGD card in the user's control file there is one RDGD card in the new control file if the desired map is in one part, and several RDGD cards if the desired map is in several parts.
2) For each DRAW card without a RDGD card before it in the user's control file there is no RDGD card in the new control file if both previous and actual maps are in one part; there is one RDGD card if previous map is in several parts and actual map is in one part; there are several RDGD cards if actual map is in several parts.

REGN card

This card describes the (X,Y) limits of the (part of) grid that is to be produced.

Variable	Parameter	Description
1	TYPE	REGN
2	XMPMN2	minimum X-value of the (part of) grid to be produced, in data units

3	XMPMX2	maximum X-value of the (part of) grid to be produced, in data units
4	YMPMN2	minimum Y-value of the (part of) grid to be produced, in data units
5	YMPMX2	maximum Y-value of the (part of) grid to be produced, in data units

Format: (A4, 6X, 4"FVAR")

Number of REGN cards: There is one REGN card for each GRID card of new control file.

Data files

User's input data file.--This file contains two kinds of data sets. The first one is composed of records that contain the (X,Y,Z) points from which a grid will be created. The first field of the record is an alphanumeric field that the program CTRL2D examines to see whether the end of the data set has been found. The word END in the alphanumeric field tells the program that the last input point for the current data set was reached. The location of the (X,Y,Z) and alphanumeric fields is specified by the FMT card in the user's control file. The order in which the (X,Y,Z) fields appear in the records is specified on the GRID card in the user's control file.

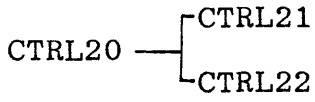
The second type of data set is composed of records that contain the Z-values of a rectangular array of pregridded data. The FMT and RDGD cards in the user's control file control the reading of this kind of data.

New input data file.--The new input data file is named DB1:CONTUR.DAT. This file contains the two following types of data records. The first type of data records contains the (X,Y,Z) points from which a (part of) grid will be created. The first field of the record is the same alphanumeric field as before. The location of the (X,Y,Z) and alphanumeric fields is specified on the FMT card in the new control file. This FMT card is identical for all the ungridded data sets. The number of data points, and the order in which the (X,Y,Z) fields appear in the records, are specified on the GRID card in the new control file.

The second type of data records contains the Z-values of a rectangular (part of) array of pregridded data. The FMT and RDGD cards in the new control file control the reading of this kind of data.

Program structure

The program consists of one main program (CTRL2D), three main routines (CREATE, EDITF, READAL), and eleven other routines. It is divided into three overlays, with the following structure.



The following is a brief description of the routines.

CTRL20: Main program

CTRL21: Create or edit user's control file

- CREATE : create user's control file
- EDITF : edit an existing control file
- SELECT : select entry in routine CCTR1
- CCTR1 : create one control card (entries CCTR2 to CCTR16)
- COPY : copy one file to another (entry COPIOW to copy on printer file)
- RWCOP : copy one file to another and vice versa

CTRL22: Read user's control and input data files, and create new control and input data files

- READAL : read control cards
- RCTR1 : read one control card (entries RCTR2 to RCTR16)
- WRITAL : read user's input data file, write data on intermediary file; check if map is in one or several parts
- WRIONE : calculate parameters when map is in one part
- WRISEV : calculate parameters when map is in several parts
- RWDATP : read intermediary data file and write on new input data file for pregridded data (entry RWDATG for ungridded data).
- WRIPRE : write control cards on new control file for pregridded data (entry WRIGRI for ungridded data)
- WCTR1 : write one control card on new control data file (entries WCTR2 to WCTR16)

The main program flow chart is shown in figure 2, and flow charts of most of the routines are shown in figures 3 to 12.

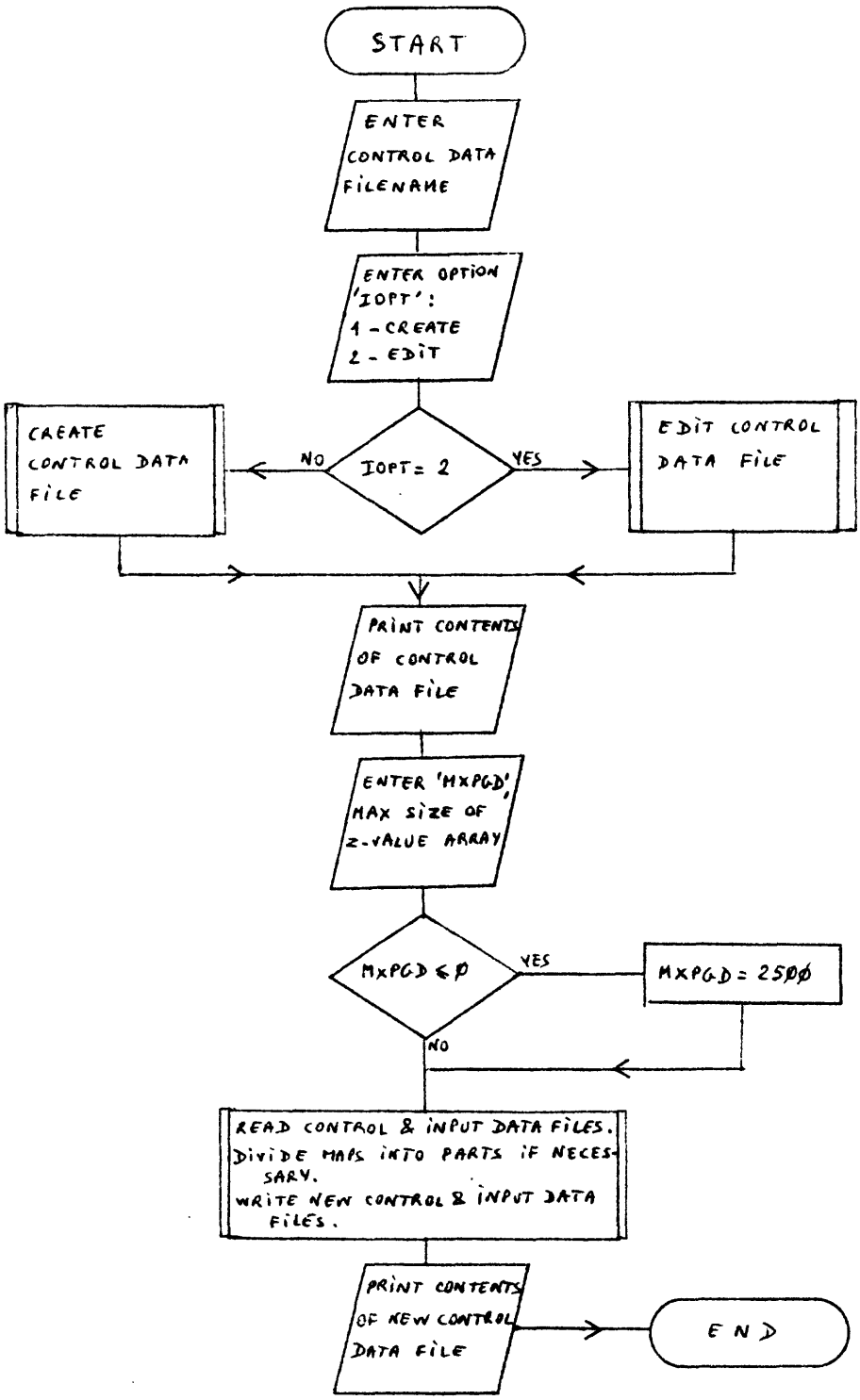


Figure 2.–CTRL2D program flow chart.

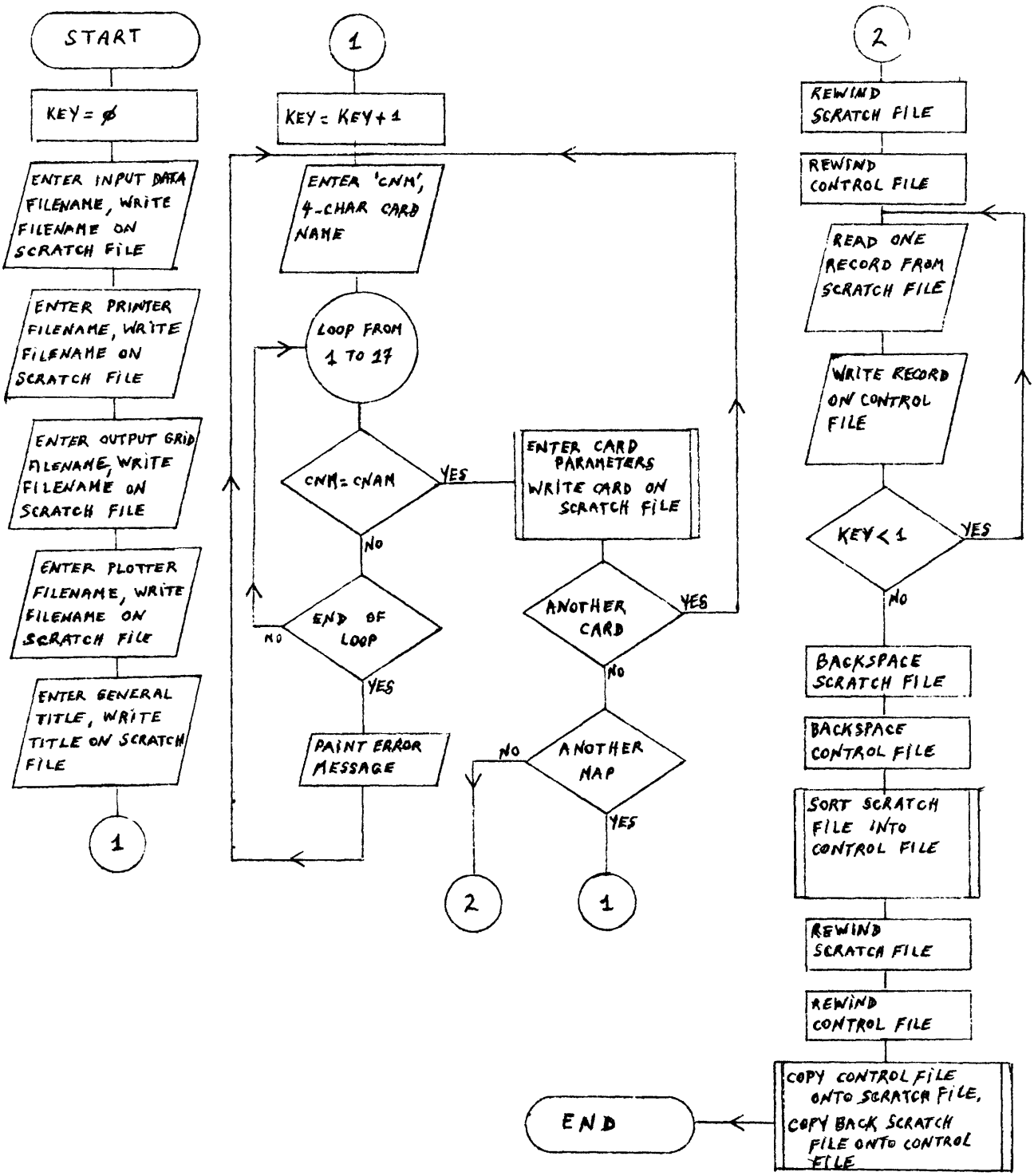


Figure 3.–CREATE subprogram flow chart.

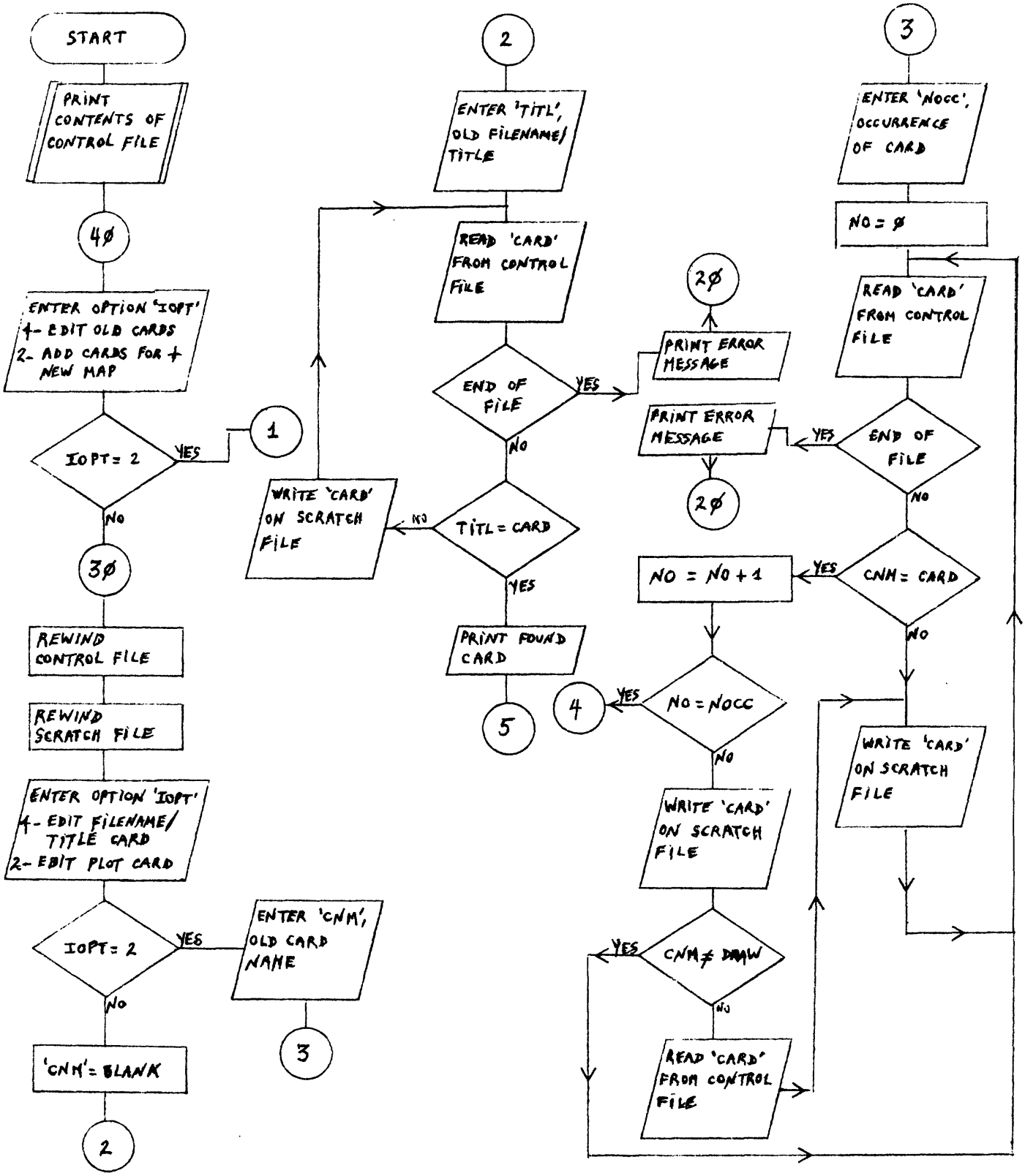


Figure 4.–EDITF subprogram flow chart.

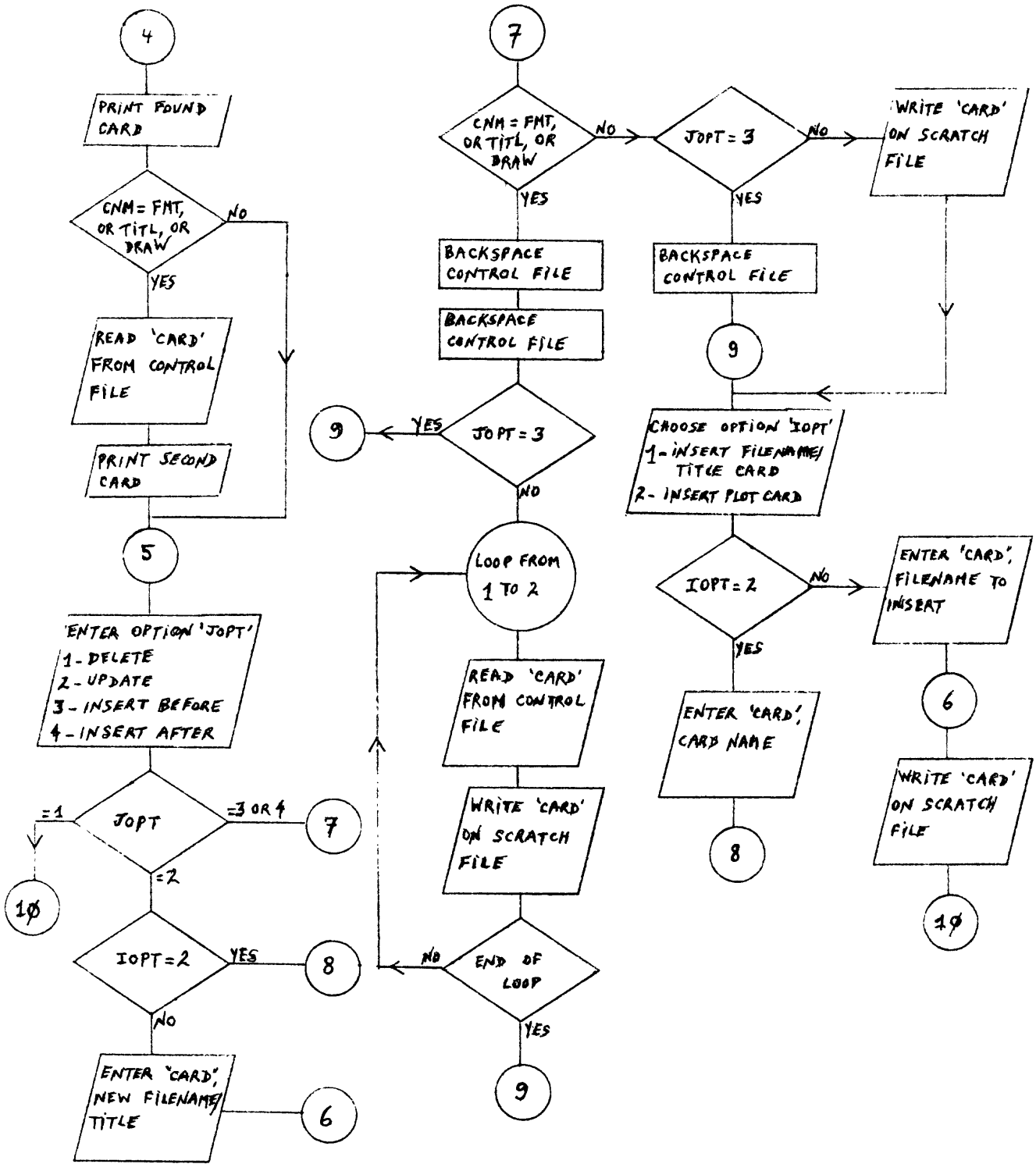


Figure 4.–Continued.

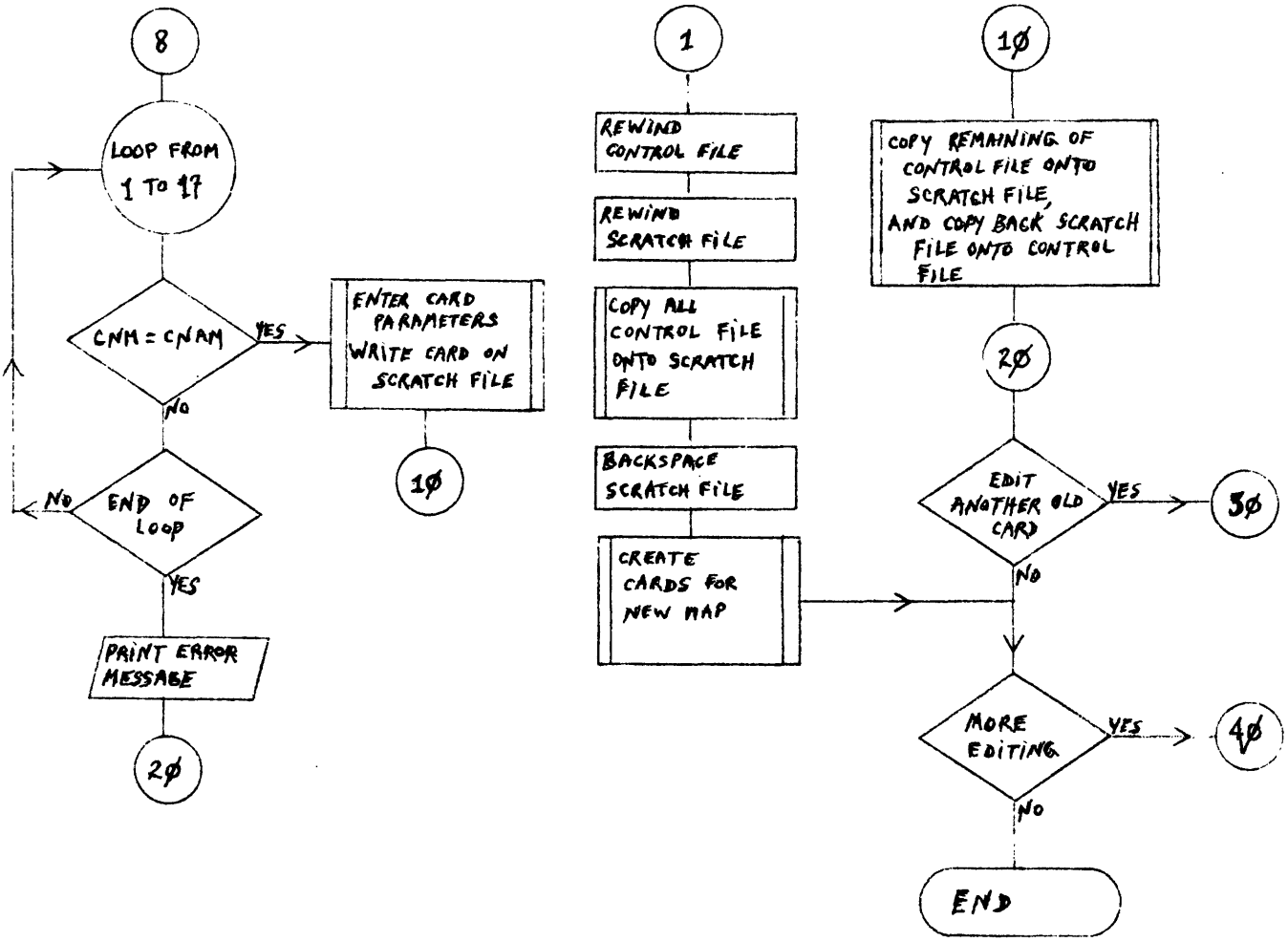


Figure 4.—Continued

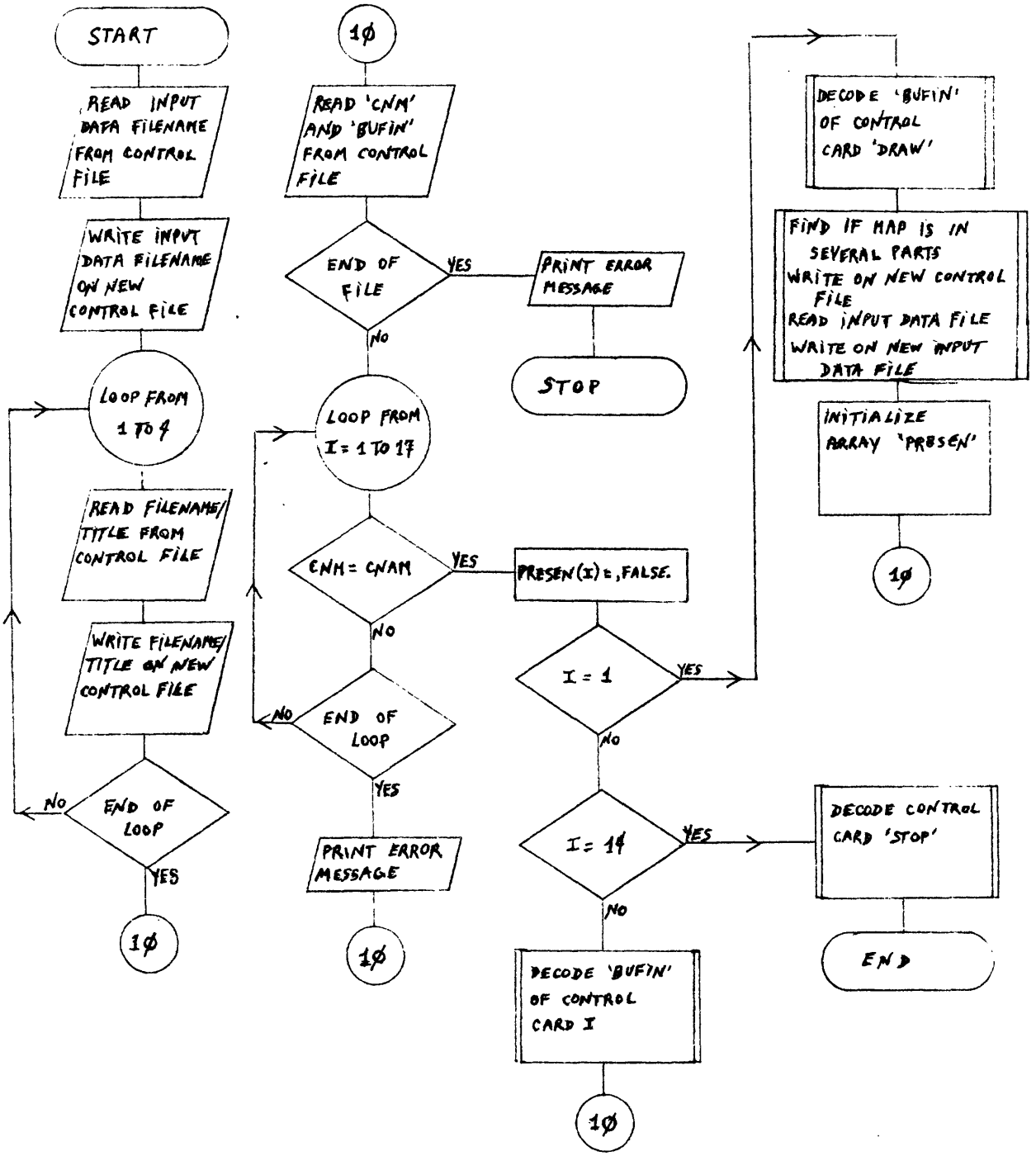


Figure 5.–READAL subprogram flow chart.

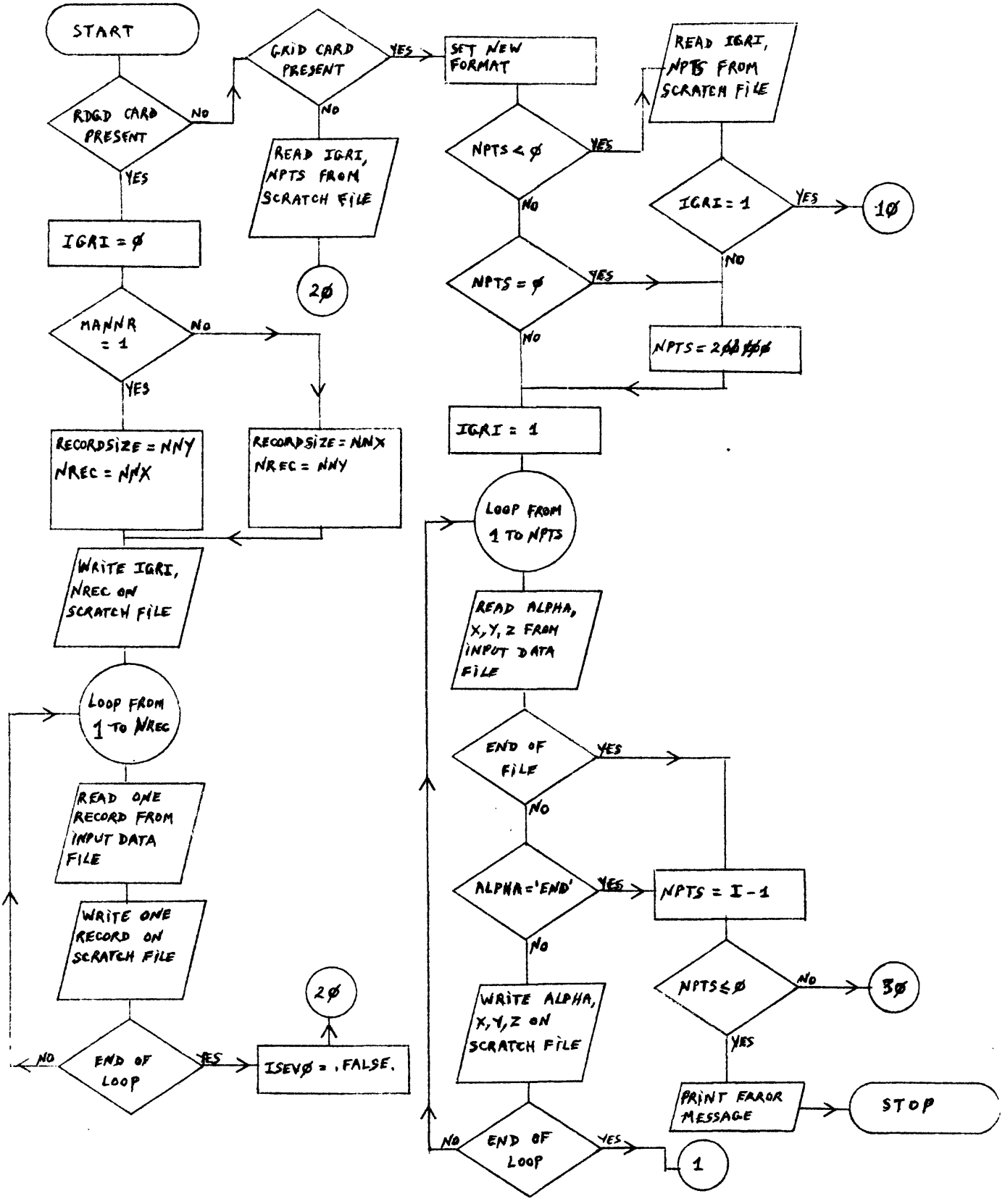


Figure 6.–WRITAL subprogram flow chart.

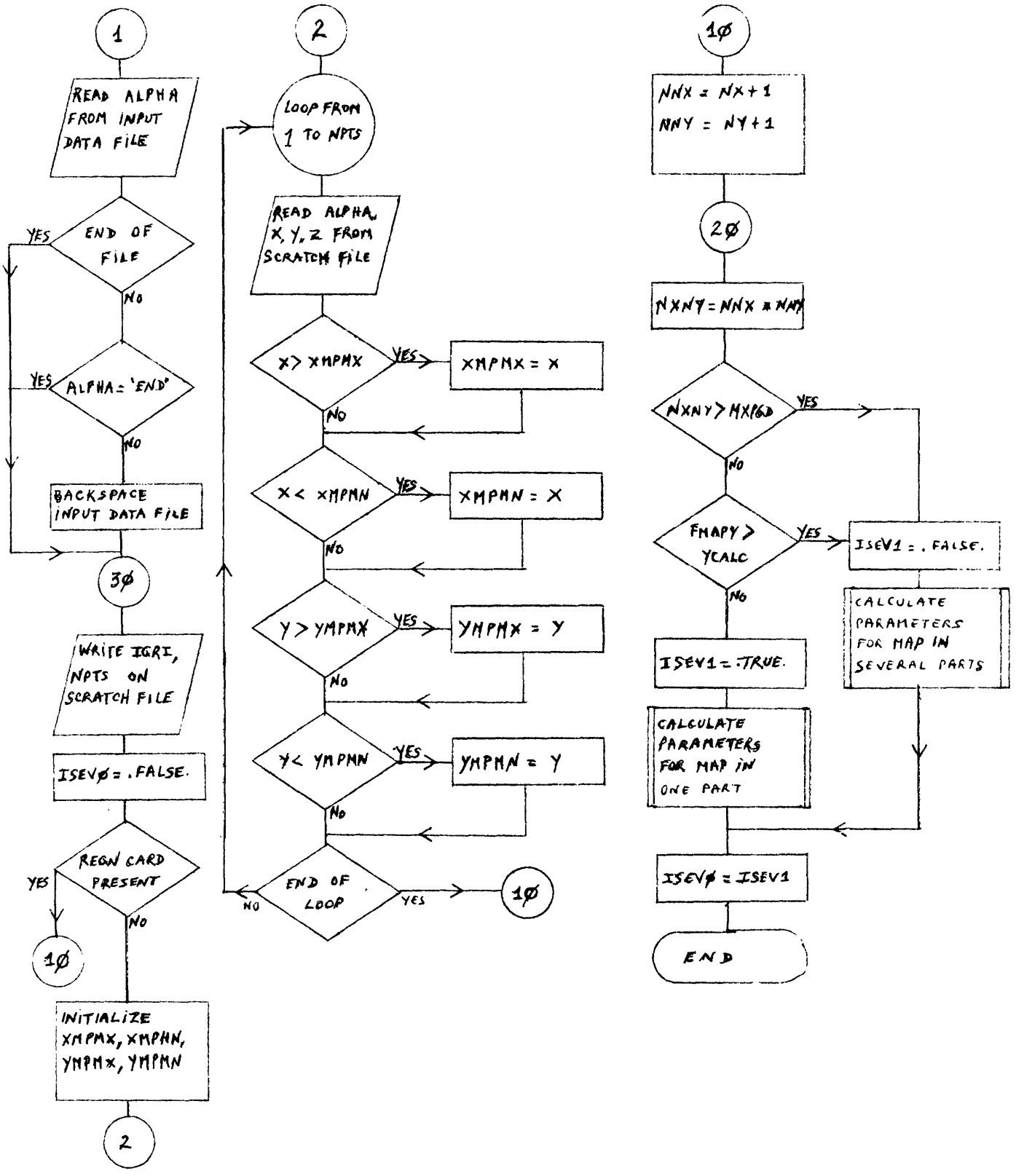


Figure 6.–Continued.

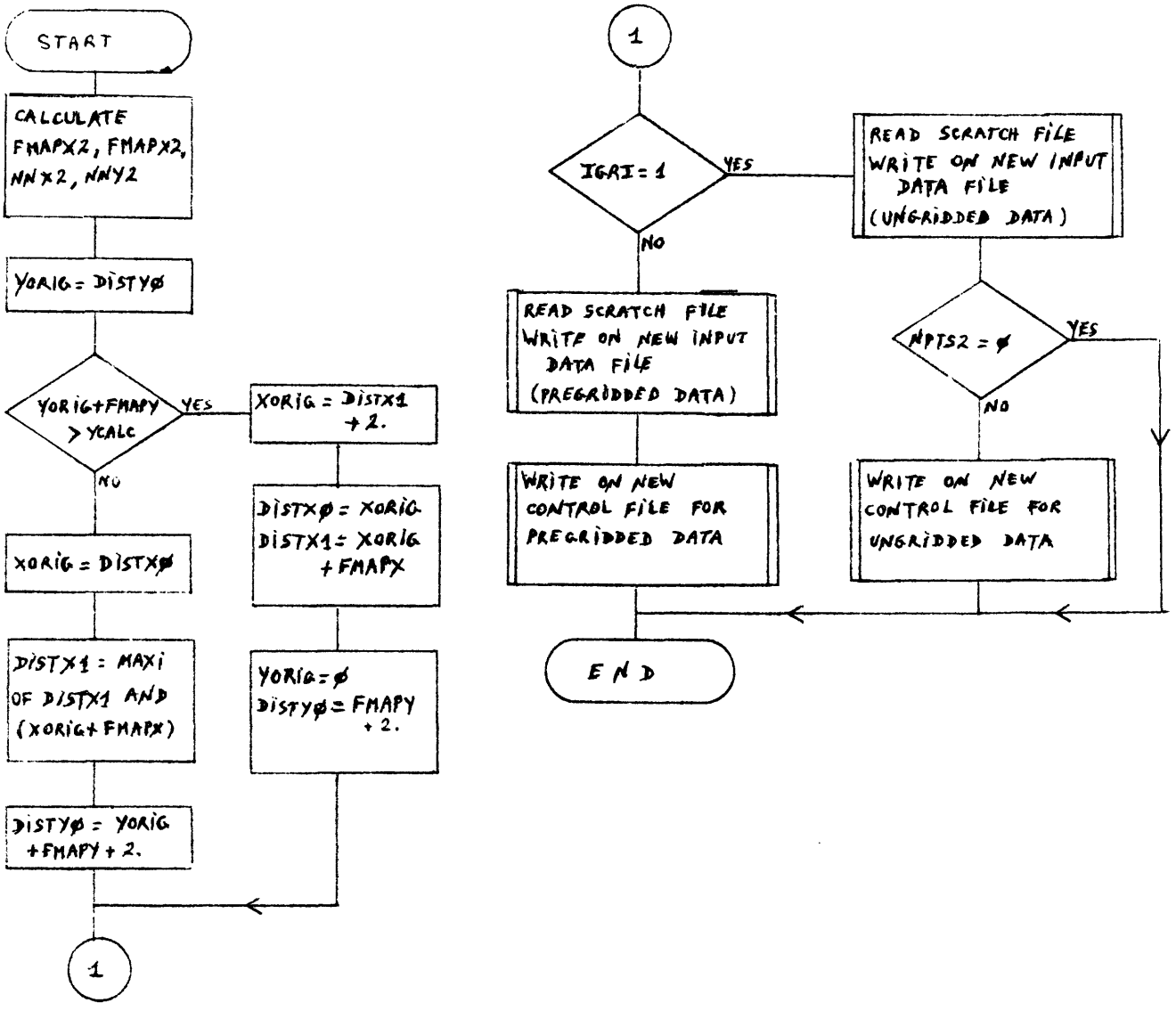


Figure 7.–WRIONE subprogram flow chart.

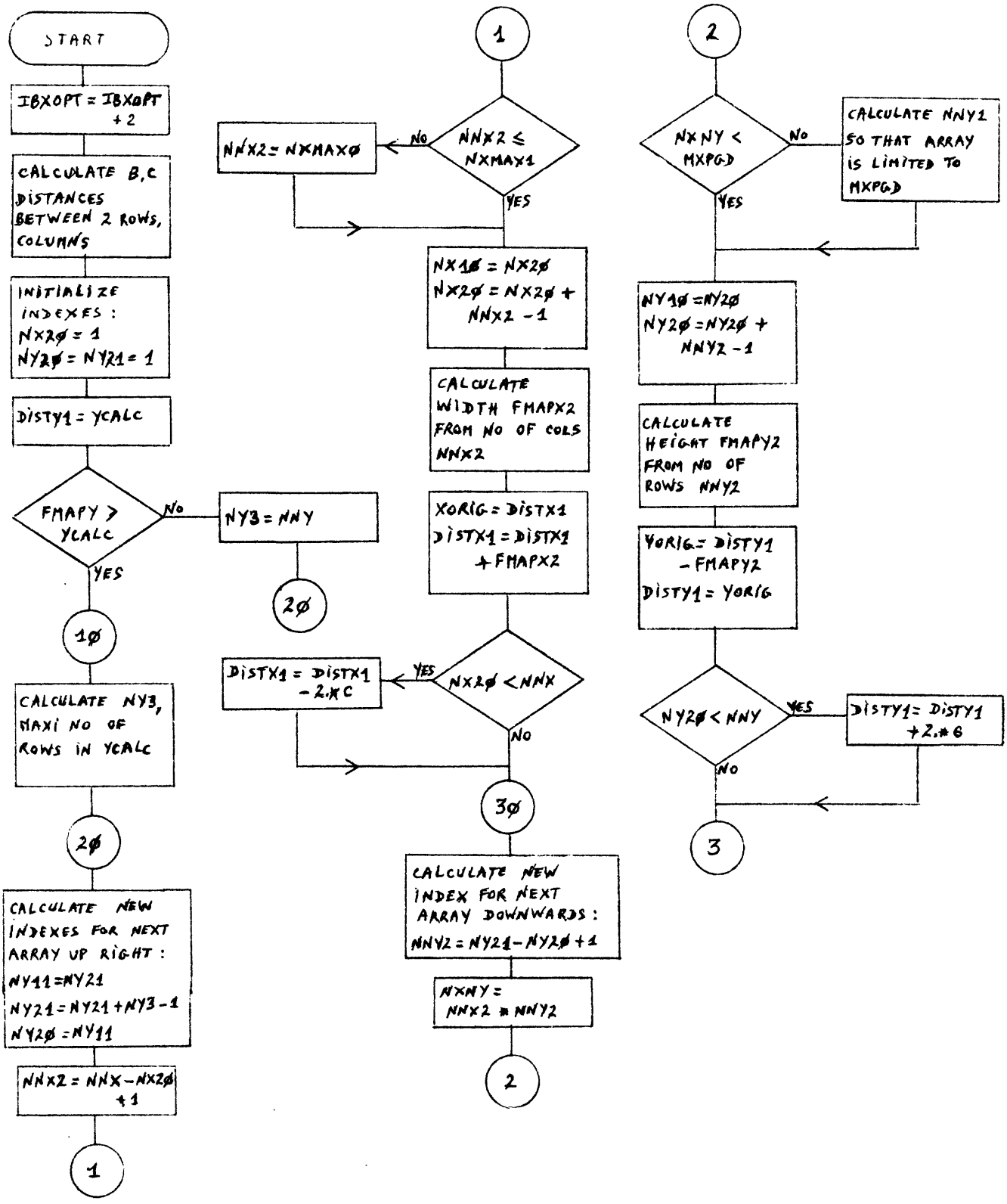


Figure 8.–WRISEV subprogram flow chart.

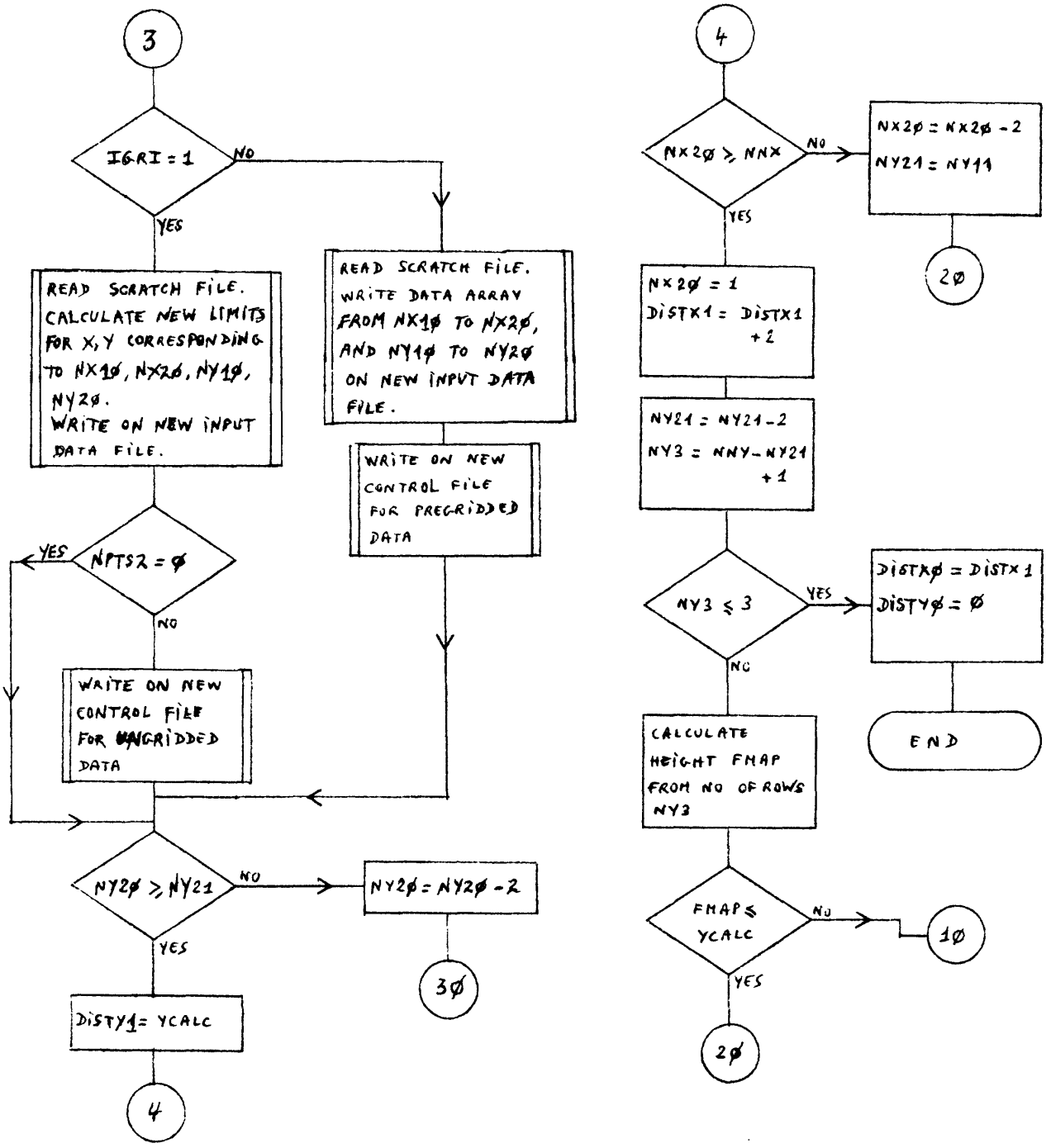


Figure 8.–Continued.

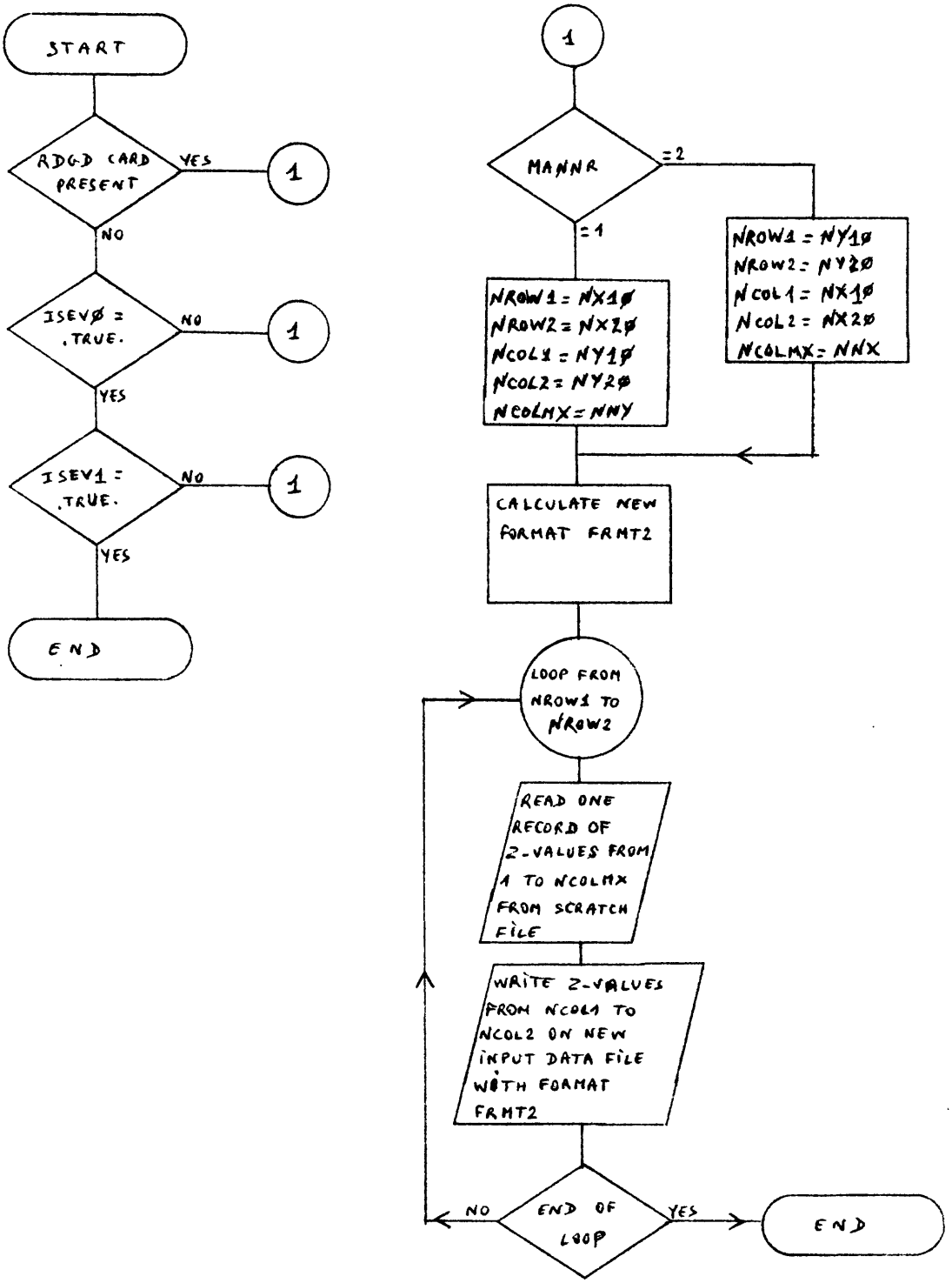


Figure 9.–RWDATP subprogram flow chart.

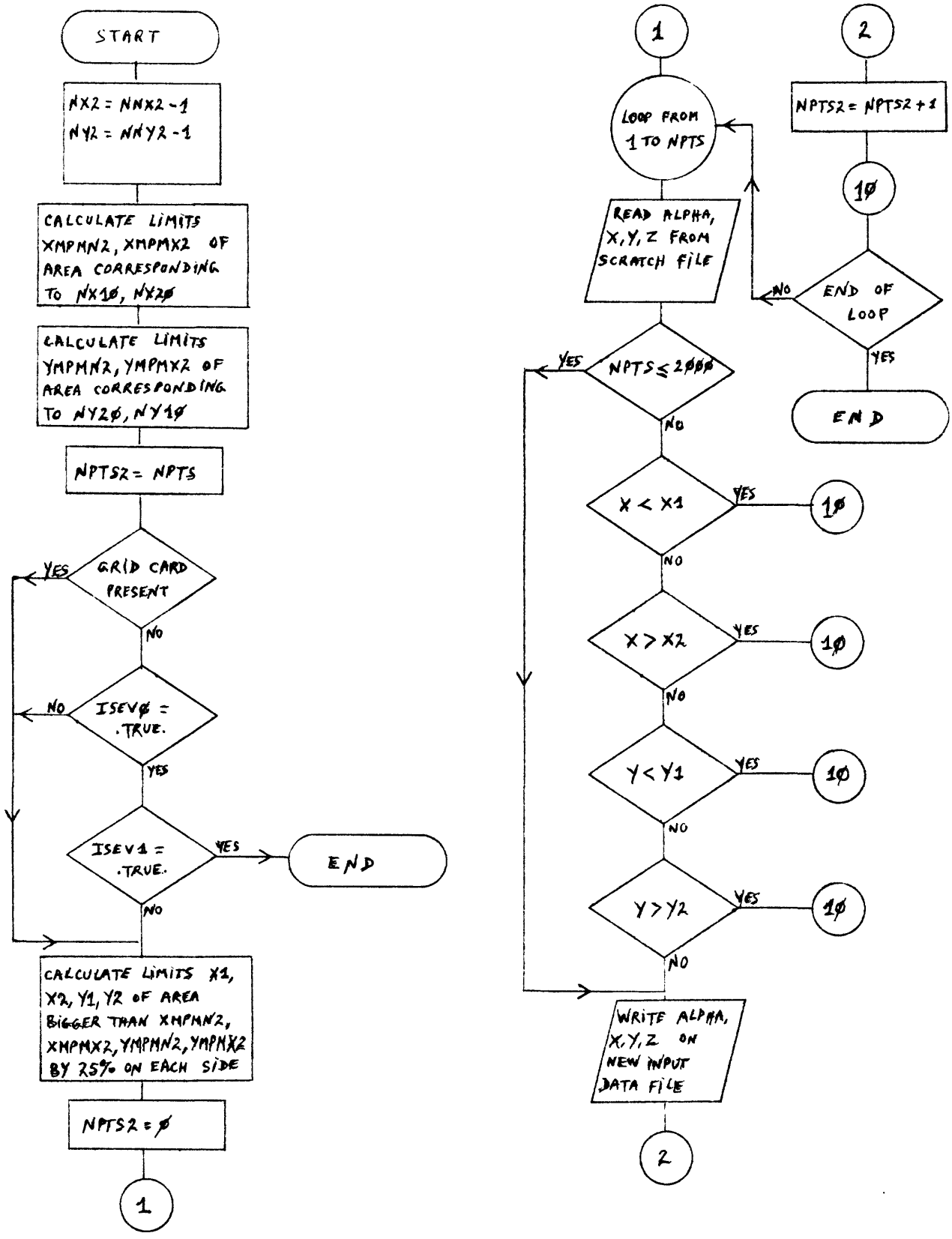


Figure 10.–RWDATP subprogram flow chart, entry RWDATG.

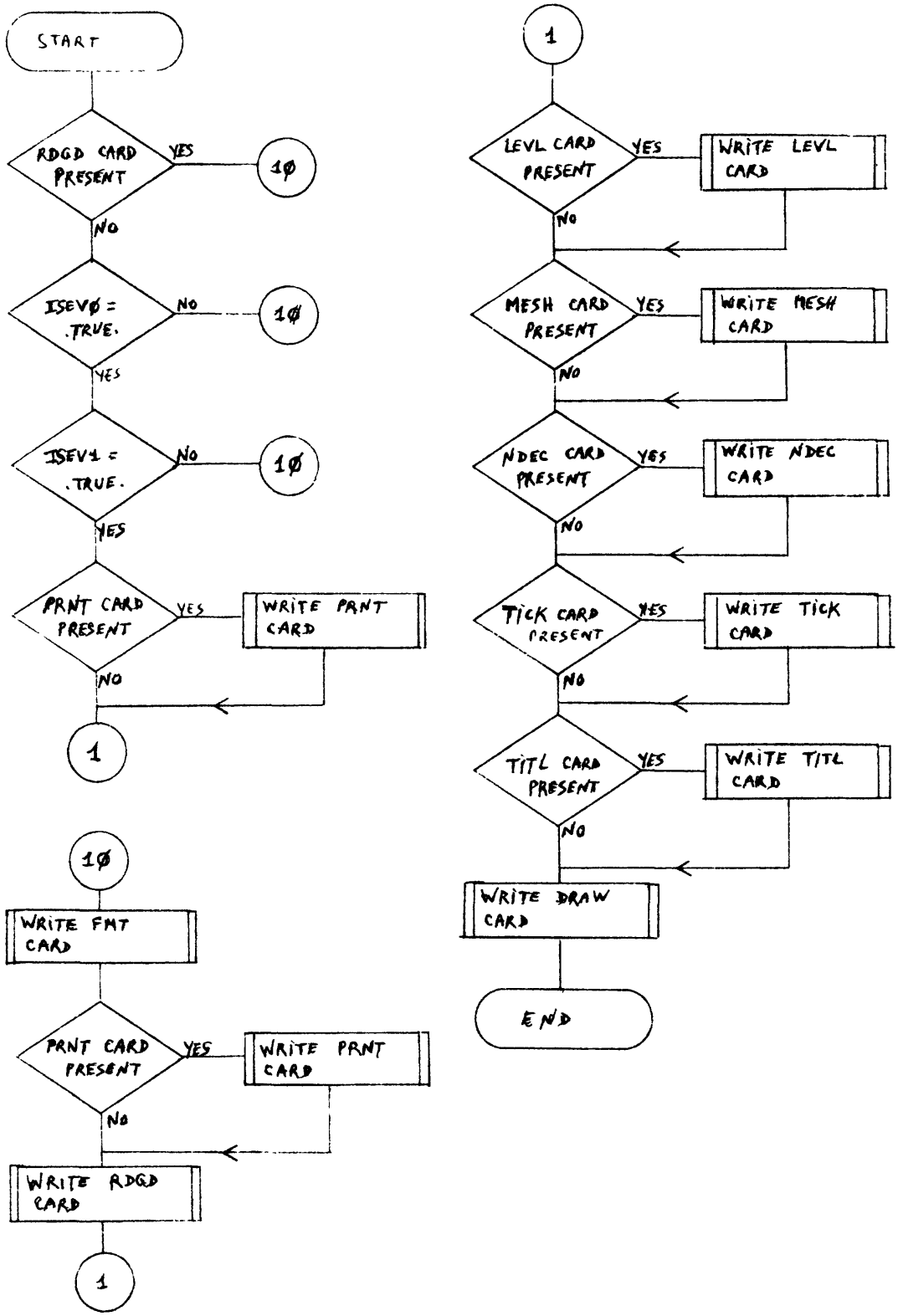


Figure 11.–WRIPRE subprogram flow chart.

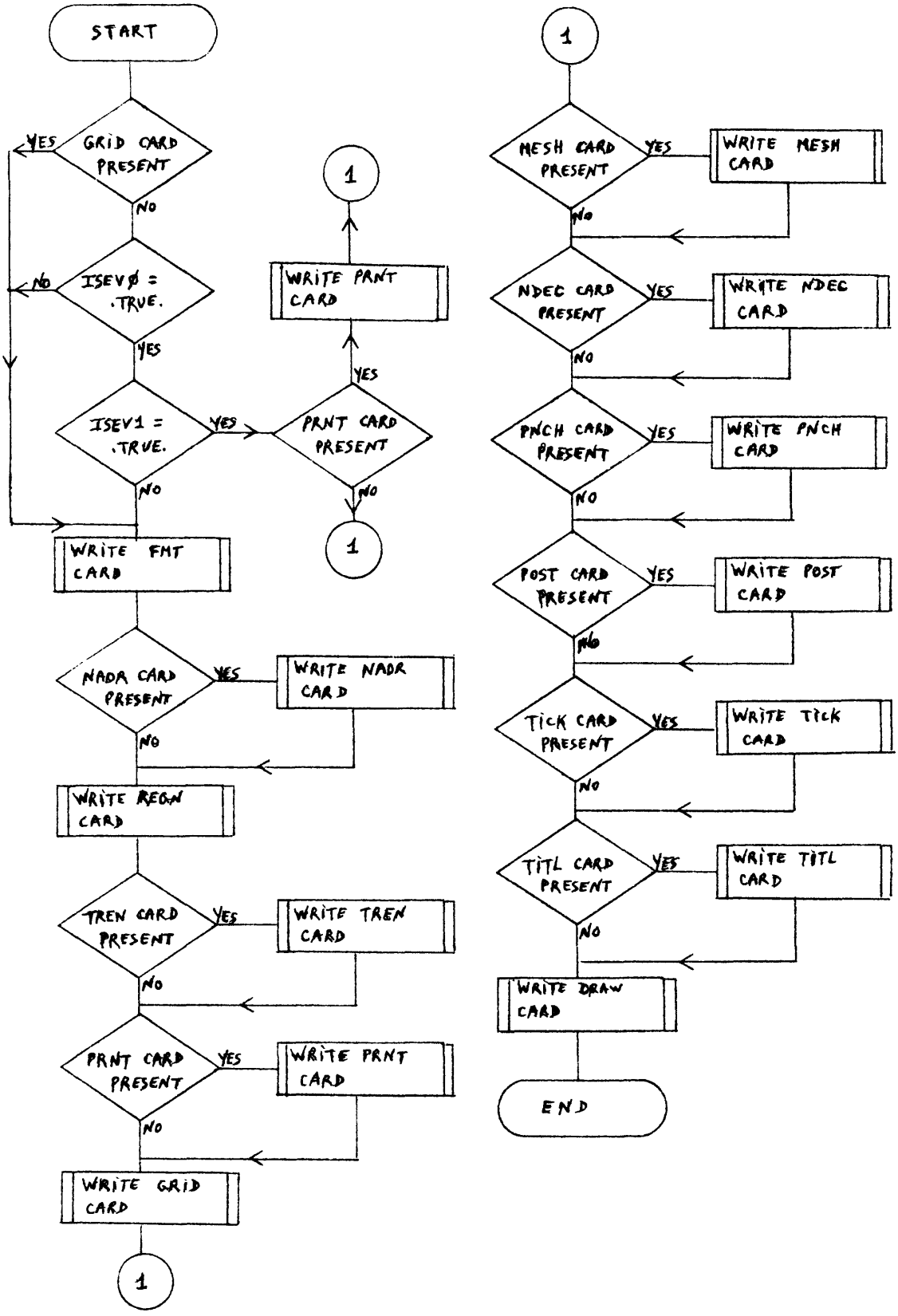


Figure 12.–WRIPRE subprogram flow chart, entry WRIGRI.

Size of the program

Control and execution of the program is accomplished interactively through the terminal display.

The source program, scattered in three overlays, requires 75 blocks of memory and the task image CTRL2D.TSK requires 158 blocks.

PROGRAM OPERATION

To run the program enter RUN CTRL2D($)[1] on the terminal. Operation of the program is largely self-explanatory. Enter information through the terminal keyboard in response to requests appearing on the terminal display.

Be sure that the input data file exists before running CTRL2D.

Note that a blank character must be entered following the three letters of the FMT card name, otherwise error number 1 will occur.

When creating or editing the user's control file, you may enter the control cards in any order, but all the cards for a single data set must be together.

If one map is to be divided into several parts, avoid the use of the TITL card, or enter a blank title.

The final question asked by the program is "maximum size of Z-value array for one part of map (default = 2500)". Enter 0 or 2500 to get the default value. In some cases it may be useful to enter a value less than the default value, for example when error number 17 occurs during the running of CONTUR.

The contents of the user's control file and the new control file named CONTUR.CTR are written on the printer file CTRL2D.PNT. during program execution. The Utility program PIP can be used to direct the printer file to the line printer and then to delete the printer file. The new input data file is named DB1:CONTUR.DAT. The new control and input data files are now ready to be used by the program CONTUR.

Following is the description of the error messages which appear on the terminal and on the printer file.

[1] The symbol ($) means ALT mode or ESC key.

Error messages on terminal

1) *** NO CONTROL CARD NAMED: XXXX ***

Routine:	CREATE, EDITF (CTRL21)
Explanation:	such a card name is not available
Program action:	1) option create: ask the same question again "4-char card name" 2) option edit: ask the new question "edit another old card?"
User action:	1) option create: enter the correct card name. Note that the FMT card name must be followed by a blank character to be accepted as a 4-character card name 2) option edit: answer "Y", and continue editing by entering the right card name

2) *** CARD NOT FOUND ***

Routine:	EDITF(CTRL21)
Explanation:	the filename or general title entered for editing was not found in the user's control file
Program action:	asks the new question "edit another old card?"
User action:	answer "Y", and continue editing by entering the right filename or general title

3) *** NO OCCURRENCE ≠ XX OF CARD XXXX FOUND ***

Routine:	EDITF(CTRL21)
Explanation:	there is no such card name with such an occurrence number in the user's control file
Program action:	ask the new question "edit another old card?"
User action:	answer "Y", and continue editing by entering the right old card name and(or) occurrence number

4) *** NO CONTROL CARD NAMED: XXXX ***

Routine: READAL (CTRL22)

Explanation: while reading the user's control file, the program found a control card name which is not allowed

Program action: read next control card

User action: none

5) *** NO 'STOP' CARD IN CONTROL FILE ***

Routine: READAL (CTRL22)

Explanation: the user's control file was not terminated by a STOP card

Program action: stop

User action: add the STOP card in the control file

6) *** END OF CONTROL FILE NOT EXPECTED ***

Routine: RCTR1 (CTRL22)

Explanation: the program found the end of the user's control file while trying to read a second DRAW card, or a second FMT card, or a LEVL card, or a second TITL card

Program action: stop

User action: correct the control file

7) *** ERROR WHILE READING PREGRIDDED DATA FROM INPUT DATA FILE ***, followed by the record number and the three first Z-values of array

Routine: WRITAL (CTRL22)

Explanation: message is self-explanatory

Program action: stop

User action: correct the input data file

8) *** ERROR WHILE READING UNGRIDDED DATA FROM INPUT DATA FILE ***, followed by the record number, alpha field, and (X,Y,Z) values.

Routine: WRITAL (CTRL22)

Explanation: message is self-explanatory

Program action: stop

User action: correct the input data file

9) *** END OF INPUT DATA FILE NOT EXPECTED WHEN READING PREGRIDDED DATA ***

Routine: WRITAL (CTRL22)

Explanation: the value of NNX or NNY is wrong, or there are not enough data

Program action: stop

User action: correct RDGD card of control file, or correct the input data file

10) *** ERROR WHILE WRITING ON INTERM FILE, ON RECORD NO: XXX ***

Routine: WRITAL (CTRL22)

Explanation: no more disk space available on DB1: to write the scratch file, or the DB1: disk is corrupted

Program action: stop

User action: see computer system manager

11) *** ERROR WHILE READING INTERM FILE, ON RECORD NO: XXXX ***

Routine: WRITAL (CTRL22)

Explanation: DB1: disk is corrupted

Program action: stop

User action: see computer system manager

12) *** NO MORE DATA SET ON INPUT DATA FILE, WHEN READING UNGRIDDED DATA ***

Routine: WRITAL (CTRL22)

Explanation: the program tried to read a new data set and could not find it

Program action: stop

User action: correct GRID card of control file, or correct the input data file

13) *** ERROR WHILE WRITING FINAL DATA FILE, ON ROW NO: XXXX ***

Routine: RWDATP, RWDATG (CTRL22)

Explanation: no more disk space available on DB1: to write the new input data file DB1:CONTUR.DAT, or the DB1: disk is corrupted

Program action: stop

User action: see computer system manager

EXAMPLE

Following is the printout resulting from a typical run of program CTRL2D.

The control file DB1:GM123.CTL was created with a previous run of CTRL2D. In this example, the control cards REGN, GRID, and DRAW are updated. The array to be created with program CONTUR is 174x85, which is not acceptable (greater than 2500). The program CTRL2D divides this array into eight parts, calculates the new values of parameters, and creates a new control file CONTUR.CTR, and a new data file DB1:CONTUR.DAT.

The time to run CTRL2D with this example was 20 minutes.

CALCOMP CONTOUR PROGRAM INPUT.   20-NOV-79
LIST OF CONTROL CARD DATASET READ FROM FILE : DB1:GM123.CTL

```
0.........1....... .2..........3.........4.........5.........6.........7.........8
DB1:GM123.SAV
DB1:GM123.PNT
DB1:BID.
DB1:GM123.PLT
GM123, 20 NOV 79, M.DONZEAU
FMT
(A1,3E13.7)
NADR               0          1   3.200000
REGN       119.1000   524.7000   1095.800   1430.600
GRID               0        169        124          0          0          0
LEVL       4800.000              0.0000E+00
LEVL       5200.000              0.0000E+00
LEVL       5600.000              0.0000E+00
LEVL       5700.000              0.0000E+00
LEVL       5800.000              0.0000E+00
LEVL       5900.000              0.0000E+00
LEVL       6000.000              0.0000E+00
LEVL       6100.000              0.0000E+00
LEVL       6200.000              0.0000E+00
LEVL       6600.000              0.0000E+00
LEVL       7000.000              0.0000E+00
POST         0.0200          0
DRAW1      32.18089   26.78795        10.        10.          1
DRAW2     -.1000E+06
STOP
0.........1....... .2..........3.........4.........5.........6.........7.........8
```

```
CALCOMP CONTOUR PROGRAM INPUT.                                    20-NOV-79
LIST OF EDITED CONTROL CARD DATASET WRITTEN ON FILE : DB1:GM123.CTL


0.........1.........2.........3.........4.........5.........6.........7.........8
DB1:GM123.SAV
DB1:GM123.PNT
DB1:BID.
DB1:GM123.PLT
GM123, 20 NOV 79, M.DONZEAU
FMT
(A1,3E13.7)
NAOR                0          1  3.200000
REGN         436.2000  851.4000  644.4000  871.2000
GRID                0       173        84         0         0         0
LEVL         4800.000           0.0000E+00
LEVL         5200.000           0.0000E+00
LEVL         5600.000           0.0000E+00
LEVL         5700.000           0.0000E+00
LEVL         5800.000           0.0000E+00
LEVL         5900.000           0.0000E+00
LEVL         6000.000           0.0000E+00
LEVL         6100.000           0.0000E+00
LEVL         6200.000           0.0000E+00
LEVL         6600.000           0.0000E+00
LEVL         7000.000           0.0000E+00
POST            0.020          0
DRAW1        32.96753  17.99756        10.       10.         1
DRAW2     -.1000E+36
STOP
0.........1.........2.........3.........4.........5.........6.........7.........8
```

CALCOMP CONTOUR PROGRAM INPUT. 20-NOV-79
LIST OF FINAL CONTROL CARD DATASET WRITTEN ON FILE : CONTUR.CTR

```
0.........1.........2.........3.........4.........5.........6.........7.........8
CONTUR.DAT
DB1:GM123.PNT
DB1:BID.
DB1:GM123.PLT
GM123, 20 NOV 79, M.DONZEAU
FMT
(A4,6X,3(E13.7,1X))
NADR                0          1  3.200000
REGN         436.2000  553.8000  741.6000  871.2000
GRID             1371         49        48         1         2         3
LEVL         4800.000   0.0000E+00
LEVL         5200.000   0.0000E+00
LEVL         5600.000   0.0000E+00
LEVL         5700.000   0.0000E+00
LEVL         5800.000   0.0000E+00
LEVL         5900.000   0.0000E+00
LEVL         6000.000   0.0000E+00
LEVL         6100.000   0.0000E+00
LEVL         6200.000   0.0000E+00
LEVL         6600.000   0.0000E+00
LEVL         7000.000   0.0000E+00
POST           0.0200          0
DRAW1        9.337624  10.28432       10.       10.         3
DRAW2    -.1000E+360  0.0000E+00  16.71568
FMT
(A4,6X,3(E13.7,1X))
NADR                0          1  3.200000
REGN         436.2000  553.8000  644.4000  747.0000
GRID              358         49        38         1         2         3
LEVL         4800.000   0.0000E+00
LEVL         5200.000   0.0000E+00
LEVL         5600.000   0.0000E+00
LEVL         5700.000   0.0000E+00
LEVL         5800.000   0.0000E+00
LEVL         5900.000   0.0000E+00
LEVL         6000.000   0.0000E+00
LEVL         6100.000   0.0000E+00
LEVL         6200.000   0.0000E+00
LEVL         6600.000   0.0000E+00
LEVL         7000.000   0.0000E+00
POST           0.0200          0
DRAW1        9.337624  8.141753       10.       10.         3
DRAW2    -.1000E+360  0.0000E+00  9.002439
FMT
(A4,6X,3(E13.7,1X))
NADR                0          1  3.200000
REGN         549.0000  566.6000  741.6000  871.2000
GRID             2819         49        48         1         2         3
LEVL         4800.000   0.0000E+00
LEVL         5200.000   0.0000E+00
LEVL         5600.000   0.0000E+00
LEVL         5700.000   0.0000E+00
LEVL         5800.000   0.0000E+00
LEVL         5900.000   0.0000E+00
LEVL         6000.000   0.0000E+00
LEVL         6100.000   0.0000E+00
LEVL         6200.000   0.0000E+00
LEVL         6600.000   0.0000E+00
LEVL         7000.000   0.0000E+00
POST           0.0200          0
```

```
DRAW1       9.337624  10.28432         10.        10.         3
DRAW2     -.1000E+06  8.956496   16.71568
FMT
(A4,6X,3(E13.7,1X))
NADR               0         1   3.200000
REGN        540.0000  666.6000   644.4000   747.0000
GRID            1810        49         38          1          2          3
LEVL        4800.000           0.0000E+00
LEVL        5200.000           0.0000E+00
LEVL        5600.000           0.0000E+00
LEVL        5700.000           0.0000E+00
LEVL        5800.000           0.0000E+00
LEVL        5900.000           0.0000E+00
LEVL        6000.000           0.0000E+00
LEVL        6100.000           0.0000E+00
LEVL        6200.000           0.0000E+00
LEVL        6600.000           0.0000E+00
LEVL        7000.000           0.0000E+00
POST          0.0200         0
DRAW1       9.337624  8.141753         10.        10.         3
DRAW2     -.1000E+06  8.956496   9.002439
FMT
(A4,6X,3(E13.7,1X))
NADR               0         1   3.200000
REGN        661.8000  779.4000   741.6000   871.2000
GRID            2836        49         48          1          2          3
LEVL        4800.000           0.0000E+00
LEVL        5200.000           0.0000E+00
LEVL        5600.000           0.0000E+00
LEVL        5700.000           0.0000E+00
LEVL        5800.000           0.0000E+00
LEVL        5900.000           0.0000E+00
LEVL        6000.000           0.0000E+00
LEVL        6100.000           0.0000E+00
LEVL        6200.000           0.0000E+00
LEVL        6600.000           0.0000E+00
LEVL        7000.000           0.0000E+00
POST          0.0200         0
DRAW1       9.337624  10.28432         10.        10.         3
DRAW2     -.1000E+06  17.91299   16.71568
FMT
(A4,6X,3(E13.7,1X))
NADR               0         1   3.200000
REGN        661.8000  779.4000   644.4000   747.0000
GRID            2277        49         38          1          2          3
LEVL        4800.000           0.0000E+00
LEVL        5200.000           0.0000E+00
LEVL        5600.000           0.0000E+00
LEVL        5700.000           0.0000E+00
LEVL        5800.000           0.0000E+00
LEVL        5900.000           0.0000E+00
LEVL        6000.000           0.0000E+00
LEVL        6100.000           0.0000E+00
LEVL        6200.000           0.0000E+00
LEVL        6600.000           0.0000E+00
```

```
LEVL        7000.000          0.0000E+00
POST          0.0200        0
DRAW1       9.337624  8.141753        10.        10.         3
DRAW2     -.1000E+36  17.91299  9.002439
FMT
(A4,6X,3(E13.7,1X))
NAOR             0         1  3.200000
REGN        774.6000  851.4000  671.4000  871.2000
GRID             336       32        74         1         2         3
LEVL        4800.000          0.0000E+00
LEVL        5200.000          0.0000E+00
LEVL        5600.000          0.0000E+00
LEVL        5700.000          0.0000E+00
LEVL        5800.000          0.0000E+00
LEVL        5900.000          0.0000E+00
LEVL        6000.000          0.0000E+00
LEVL        6100.000          0.0000E+00
LEVL        6200.000          0.0000E+00
LEVL        6600.000          0.0000E+00
LEVL        7000.000          0.0000E+00
POST          0.0200        0
DRAW1       6.098040  15.85499        10.        10.         3
DRAW2     -.1000E+36  26.86949  11.14501
FMT
(A4,6X,3(E13.7,1X))
NAOR             0         1  3.200000
REGN        774.6000  851.4000  644.4000  676.8000
GRID             704       32        12         1         2         3
LEVL        4800.000          0.0000E+00
LEVL        5200.000          0.0000E+00
LEVL        5600.000          0.0000E+00
LEVL        5700.000          0.0000E+00
LEVL        5800.000          0.0000E+00
LEVL        5900.000          0.0000E+00
LEVL        6000.000          0.0000E+00
LEVL        6100.000          0.0000E+00
LEVL        6200.000          0.0000E+00
LEVL        6600.000          0.0000E+00
LEVL        7000.000          0.0000E+00
POST          0.0200        0
DRAW1       6.098040  2.571080        10.        10.         3
DRAW2     -.1000E+36  26.86949  9.002439
$STOP
0.........1.........2.........3.........4.........5.........6.........7.........8
```

SOURCE PROGRAM LISTING

Routine	Overlay	Page
CTRL2D	CTRL20	57
CREATE	CTRL21	59
EDITF	"	61
SELECT	"	64
CCTR1	"	65
COPY	"	69
RWCOP	"	70
READAL	CTRL22	71
RCTR1	"	73
WRITAL	"	75
WRIONE	"	78
WRISEV	"	79
RWDATP	"	81
WRIPRE	"	84
WCTR1	"	85

```
0001          PROGRAM CTRL2D
      C       --------------
      C   CREATE CONTROL CARDS FOR 'CONTUR' PROGRAM, CHECK IF MAPS ARE IN ONE OR
      C   SEVERAL PARTS, REBUILD INPUT DATA FILE.
      C   M.E.GETTINGS, M.D. 15 OCT 1978
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
          A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0006          COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0007          COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0008          COMMON NNY2  ,NY    ,NY2   ,SH    ,SN    ,XMPMN ,XMPMX ,XORIG
0009          COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0010          COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0011          REAL*8 FMT
0012          INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD
      C
0013          DATA ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR/1,2,3,4,5,6,7/
0014          DATA CNAM/'DRAW','FMT','GRID','LABL','LEVL','MESH','NAOR','NDEC',
          B 'PNCH','POST','PRNT','RDGD','REGN','STOP','TICK','TITL','TREN'/
0015          DATA YCALC/27./,DISTX0/0./,DISTY0/0./
0016          DATA PRESEN/18*.TRUE./,NX,NY/0,0/,DISTX1/0./
0017          DATA FRMT/4H(A4,,4H6X,3,4HF10.,2H0),16*' '/
      C
0018    2000  FORMAT(' CONTROL DATA FILE GENERATION FOR CALCOMP CONTUR',
          A ' PROGRAM'/)
0019    2001  FORMAT(//' *** OUTPUT FOR LINE PRINTER IS ON FILE :',T55,
          1 'CTRL2D.PNT'/
          2 ' *** CONTROL DATA FILE TO RUN ''CONTUR'' IS ON FILE :',T55,
          3 'CONTUR.CTR'/
          4 ' *** INPUT DATA FILE TO RUN ''CONTUR'' IS ON FILE :',T55,
          5 'DB1:CONTUR.DAT'/)
0020    2200  FORMAT(1H1,'CALCOMP CONTOUR PROGRAM INPUT.',T70,5A2/' LIST OF',
          1 ' FINAL CONTROL CARD DATASET WRITTEN ON FILE : CONTUR.CTR'//)
0021    2201  FORMAT(1H1,'CALCOMP CONTOUR PROGRAM INPUT.',T70,5A2/' LIST OF',
          1 ' CREATED CONTROL CARD DATASET WRITTEN ON FILE : ',5A4/
          2 1X,15A4/)
      C
0022          A=0.
0023          CALL FORM (A,FMT,0,0,10)
      C       ---------
0024          CALL DATE (IDAT)
0025          WRITE(ITT,2000)
0026          OPEN (UNIT=INTUN,TYPE='SCRATCH')
0027          OPEN (UNIT=JPNTR,TYPE='NEW',NAME='CTRL2D.PNT')
0028          CALL TTINAA ('FILENAME OF CONTROL DATA FILE',29,FILEN2,80,ITT)
0029          OPEN (UNIT=ICARD,TYPE='UNKNOWN',NAME=FILEN2)
0030          CALL TTINSI ('WISH TO CREATE (1), OR EDIT (2) CONTROL DATA FILE',
          A 49,IOPT,ITT)
0031          IF (IOPT.EQ.2) GOTO 500
0032          CALL CREATE
      C       -----------
0033          WRITE(JPNTR,2201) IDAT,FILEN2
0034          GOTO 110
```

```
0035    500     CALL EDITF
        C       ----------
0036    110     CALL COPIOW (ICARD,JPNTR)
        C       ----------
0037            CLOSE (UNIT=INTUN)
0038            CLOSE (UNIT=ICARD)
0039            OPEN (UNIT=ICARD,TYPE='OLD',NAME=FILEN2)
0040            OPEN (UNIT=JCARD,TYPE='NEW',NAME='CONTUR.CTR')
0041            CALL TTINSI ('MAXIMUM SIZE OF Z-VALUE ARRAY FOR ONE PART OF MAP
               1 (DEFAULT=2500)',64,MXPGD,ITT)
0042            IF (MXPGD.LE.0) MXPGD=2500
0043            CALL READAL
        C       ----------
0044            WRITE(JPNTR,2200) IDAT
0045            CALL COPIOW (JCARD,JPNTR)
        C       ----------
0046            CLOSE (UNIT=JCARD)
0047            CLOSE (UNIT=JPNTR)
0048            WRITE(ITT,2001)
0049            STOP ' *** END OF PROGRAM CTRL2D ***'
0050            END
```

```
0001          SUBROUTINE CREATE
      C       ----------------
      C   CREATE CONTROL CARDS
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
     A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          DIMENSION FILEN1(20),TITL(10),BBB(22)
0006          LOGICAL*1 IAN,BID1
      C
0007    1000  FORMAT(20A4,I2)
0008    2000  FORMAT(20A4,I2,I3)
0009    2001  FORMAT(10A4,40X,I2,I3)
0010    2002  FORMAT(' *** NO CONTROL CARD NAMED : ',A4,' ***')
0011    2003  FORMAT(20A4)
      C
      C   CREATE FILENAME CONTROL CARDS
0012          KEY=0
0013          CALL TTINAA ('FILENAME OF INPUT DATA POINTS OR GRID FILE',42,
     1 FILEN1,80,ITT)
0014          I1=1
0015          WRITE(INTUN,2000) FILEN1,KEY,I1
0016          CALL TTINAA ('FILENAME OF PRINTER FILE',24,FILEN1,80,ITT)
0017          I1=2
0018          WRITE(INTUN,2000) FILEN1,KEY,I1
0019          CALL TTINAA ('FILENAME OF OUTPUT GRID FILE',28,FILEN1,80,ITT)
0020          I1=3
0021          WRITE(INTUN,2000) FILEN1,KEY,I1
0022          CALL TTINAA ('FILENAME OF PLOTTER FILE',24,FILEN1,80,ITT)
0023          I1=4
0024          WRITE(INTUN,2000) FILEN1,KEY,I1
0025          CALL TTINAA ('40-CHAR PLOT SERIES TITLE',25,TITL,40,ITT)
0026          I1=5
0027          WRITE(INTUN,2001) TITL,KEY,I1
      C
0028          ENTRY CREAT1
      C       ------------
      C   CREATE CONTROL CARDS FOR NEXT MAP
0029    360   KEY=KEY+1
0030    300   CALL TTINAA ('4-CHAR CARD NAME',16,CNM,4,ITT)
0031          DO 50 I=1,17
0032          IF (CNM.EQ.CNAM(I)) GOTO 301
0033    50    CONTINUE
0034          WRITE(ITT,2002) CNM
0035          GOTO 300
0036    301   CALL SELECT (I)
      C       -----------
0037          CALL TTINAA ('ANOTHER CARD FOR THIS MAP ?',27,IAN,1,ITT)
0038          IF (IAN.NE.'N') GOTO 300
0039          CALL TTINAA ('CREATE CONTROL CARDS FOR ANOTHER MAP ?',38,IAN,1,
     1 ITT)
0040          IF (IAN.NE.'N') GOTO 360
      C------------------- SEARCH FOR FIRST MAP
0041          REWIND INTUN
0042          REWIND ICARD
```

```
0043    210     READ (INTUN,1000) TTL,KEY
0044            WRITE(ICARD,2003) TTL
0045            IF (KEY.LT.1) GOTO 210
0046    200     BACKSPACE INTUN
0047            BACKSPACE ICARD
        C------------------- SORT SCRATCH FILE INTO CONTROL FILE
0048            CALL SORTR (INTUN,ICARD,ITT,BBB,85,81,85)
        C       ----------
0049            REWIND ICARD
0050            REWIND INTUN
0051            CALL RWCOP (ICARD,INTUN)
        C       ----------
0052            RETURN
0053            END
```

```
0001          SUBROUTINE EDITF
      C       ----------------
      C   EDIT AN EXISTING CONTROL FILE
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
          A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          DIMENSION TITL(20),CARD(20)
0006          LOGICAL*1 IAN,BID1
      C
0007    1000  FORMAT(20A4)
0008    2000  FORMAT(' *** CARD NOT FOUND ***')
0009    2001  FORMAT(' *** NO OCCURRENCE #',I2,' OF CARD ''',A4,''' FOUND ***')
0010    2002  FORMAT(' FOUND CARD :'/1X,20A4)
0011    2003  FORMAT(' *** NO CONTROL CARD NAMED : ',A4,' ***')
0012    2004  FORMAT(' CHOOSE BETWEEN THESE 4 OPTIONS :'/T10,'1- DELETE THIS',
          3 ' CARD'/T10,'2- UPDATE THIS CARD'/T10,'3- INSERT BEFORE THIS',
          4 ' CARD'/T10,'4- INSERT AFTER THIS CARD')
0013    2005  FORMAT(20A4,'00000')
0014    2200  FORMAT(1H1,'CALCOMP CONTOUR PROGRAM INPUT.',T70,5A2/' LIST OF',
          1 ' CONTROL CARD DATASET READ FROM FILE : ',5A4/
          2 15A4/)
0015    2201  FORMAT(1H1,'CALCOMP CONTOUR PROGRAM INPUT.',T70,5A2/' LIST OF',
          1 ' EDITED CONTROL CARD DATASET WRITTEN ON FILE : ',5A4/
          2 15A4/)
      C
0016          KEY=0
0017          WRITE(JPNTR,2200) IDAT,FILEN2
0018          CALL COPIOW (ICARD,JPNTR)
      C       -----------
0019    1500  CALL TTINSI ('EDIT OLD CARDS (1), OR ADD CARDS FOR A NEW MAP (2)',
          A 50,IOPT,ITT)
0020          IF (IOPT.EQ.2) GOTO 1700
      C-------------------- EDIT OLD CARDS
0021    1550  REWIND ICARD
0022          REWIND INTUN
0023          CALL TTINSI ('EDIT A FILENAME/TITLE CARD (1), OR A PLOT SERIES
          B CARD (2)',57,IOPT,ITT)
0024          IF (IOPT.EQ.2) GOTO 500
      C-------------------- FIND OLD FILENAME/TITLE
0025          CNM=1H
0026          CALL TTINAA ('OLD FILENAME OR TITLE',21,TITL,80,ITT)
0027    1530  READ (ICARD,1000,END=1540) CARD
0028          DO 1520 I=1,20
0029          IF (CARD(I).NE.TITL(I)) GOTO 1525
0030    1520  CONTINUE
0031          WRITE(ITT,2002) CARD
0032          GOTO 100
0033    1525  WRITE(INTUN,1000) CARD
0034          GOTO 1530
0035    1540  WRITE(ITT,2000)
0036          GOTO 600
      C-------------------- FIND OLD PLOT SERIES CARD
0037    500   CALL TTINAA ('4-CHAR OLD CARD NAME',20,CNM,4,ITT)
0038          CALL TTINSI ('WHICH OCCURRENCE OF THIS CARD',29,NOCC,ITT)
```

```
0039            NO=0
0040    510     READ (ICARD,1000,END=520) CARD
0041            IF (CARD(1).NE.CNM) GOTO 511
0042            NO=NO+1
0043            IF (NO.EQ.NOCC) GOTO 530
0044            WRITE(INTUN,1000) CARD
0045            IF (CNM.NE.'DRAW') GO TO 510
0046            READ (ICARD,1000,END=520) CARD
0047    511     WRITE(INTUN,1000) CARD
0048            GOTO 510
0049    520     WRITE(ITT,2001) NOCC,CNM
0050            GOTO 600
        C------------------- CHOOSE OPTIONS
0051    530     WRITE(ITT,2002) CARD
0052            IF ((CNM.NE.'FMT').AND.(CNM.NE.'TITL').AND.(CNM.NE.'DRAW'))
               A GOTO 100
0053            READ (ICARD,1000) CARD
0054            WRITE(ITT,2002) CARD
0055    100     WRITE(ITT,2004)
0056            CALL TTINSI ('ENTER 1, 2, 3, OR 4',19,JOPT,ITT)
0057            GOTO (540,1800,533,533), JOPT
        C------------------- REPLACE CARD
0058    1800    IF (IOPT.EQ.2) GOTO 1600
0059            CALL TTINAA ('NEW FILENAME OR TITLE',21,CARD,80,ITT)
0060            GOTO 1630
0061    1600    DO 65 I=1,17
0062            IF (CNM.EQ.CNAM(I)) GOTO 66
0063    65      CONTINUE
0064            WRITE(ITT,2003) CNM
0065            GOTO 600
0066    66      CALL SELECT (I)
        C       -----------
0067            GOTO 540
        C------------------- INSERT CARD
0068    533     IF ((CNM.EQ.'FMT').OR.(CNM.EQ.'TITL').OR.(CNM.EQ.'DRAW'))
               B GOTO 105
0069            IF (JOPT.EQ.3) GOTO 110
0070            WRITE(INTUN,1000) CARD
0071            GOTO 135
0072    110     BACKSPACE ICARD
0073            GOTO 135
0074    105     BACKSPACE ICARD
0075            BACKSPACE ICARD
0076            IF (JOPT.EQ.3) GOTO 135
0077            DO 115 I=1,2
0078            READ (ICARD,1000) CARD
0079    115     WRITE(INTUN,1000) CARD
0080    135     CALL TTINSI ('INSERT A FILENAME/TITLE CARD (1), OR A PLOT SERIES
               B CARD (2)',59,IOPT,ITT)
0081            IF (IOPT.EQ.2) GOTO 560
0082    120     CALL TTINAA ('FILENAME OR TITLE',17,CARD,80,ITT)
0083    1630    WRITE(INTUN,1000) CARD
0084            GOTO 540
0085    560     CALL TTINAA ('4-CHAR CARD NAME',16,CNM,4,ITT)
0086            GOTO 1600
        C------------------- COPY REMAINING OF ICARD ON INTUN, AND BACK
```

```
0087    540     CALL RWCOP (ICARD,INTUN)
        C       ----------
0088    600     CALL TTINAA ('EDIT ANOTHER OLD CARD ?',23,IAN,1,ITT)
0089            IF (IAN.NE.'N') GOTO 1550
0090    1710    CALL TTINAA ('MORE EDITING ?',14,IAN,1,ITT)
0091            IF (IAN.NE.'N') GOTO 1500
0092            WRITE(JPNTR,2201) IDAT,FILEN2
0093            RETURN
        C------------------- ADD NEW CARDS
0094    1700    REWIND ICARD
0095            REWIND INTUN
0096    210     READ (ICARD,1000,END=200) CARD
0097            WRITE(INTUN,2005) CARD
0098            GOTO 210
0099    200     CALL CREAT1
        C       -----------
0100            GOTO 1710
0101            END
```

```
0001            SUBROUTINE SELECT (I)
      C         ----------------
0002            GOTO (101,102,103,104,105,106,107,108,109,110,111,112,113,
     1 114,109,116,109), I
0003    101     CALL CCTR1
0004            RETURN
0005    102     CALL CCTR2
0006            RETURN
0007    103     CALL CCTR3
0008            RETURN
0009    104     CALL CCTR4
0010            RETURN
0011    105     CALL CCTR5
0012            RETURN
0013    106     CALL CCTR6
0014            RETURN
0015    107     CALL CCTR7
0016            RETURN
0017    108     CALL CCTR8
0018            RETURN
0019    109     CALL CCTR9 (I)
0020            RETURN
0021    110     CALL CCTR10
0022            RETURN
0023    111     CALL CCTR11
0024            RETURN
0025    112     CALL CCTR12
0026            RETURN
0027    113     CALL CCTR13
0028            RETURN
0029    114     CALL CCTR14
0030            RETURN
0031    116     CALL CCTR16
0032            RETURN
0033            END
```

```
0001          SUBROUTINE CCTR1
      C       ---------------                    DRAW CARD
      C   CREATE CONTROL CARDS
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
     A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          DIMENSION ICODE(20)
0006          LOGICAL*1 IAN,BID1
0007          INTEGER*4 NPTS
0008          REAL*8 FMT1(7),FMT2(5),FMT5(6),FMT6(6),FMT7(5),FMT13(7)
      C
      C
      C
0009          DATA FMT1/8H(A4,'1',,3H5X,,2$' ',8H,2F10.0,,8HI10,20X,,6HI2,I3)/
0010          DATA FMT2/8H(A4,'2',,3H5X,,' ',8H,60X,I2,,3HI3)/
0011          DATA FMT5/7H(A4,6X,,3$' ',8H,40X,I2,,3HI3)/
0012          DATA FMT6/7H(A4,6X,,' ',5H,10X,,' ',8H,40X,I2,,3HI3)/
0013          DATA FMT7/8H(A4,6X,2,4HI10,,' ',8H,40X,I2,,3HI3)/
0014          DATA FMT13/7H(A4,6X,,4$' ',8H,30X,I2,,3HI3)/
0015          DATA ICODE/490,100,190,500,200,420,110,430,440,450,180,190,120,
     B 600,470,480,130,101,481,491/
0016          DATA KEY0/99/
      C
      C
      C
0017          CALL TTINSR('WIDTH OF MAP TO BE DRAWN, IN INCHES',35,FMAPX,ITT)
0018          CALL TTINSR('HEIGHT OF MAP TO BE DRAWN, IN INCHES',36,FMAPY,
     1 ITT)
0019          CALL TTINSR('# DIVISIONS (4-10) IN X OF SUBGRID',34,SN,ITT)
0020          CALL TTINSR('# DIVISIONS (4-10) IN Y OF SUBGRID',34,SM,ITT)
0021          CALL TTINSI('DRAW BOUNDARY (0) OR JUST REGISTRATION MARKS (1)',
     2 48,IBXOPT,ITT)
0022          CALL TTINSR('Z-VALUE FOR BLANK AREAS (-1.E35 FOR UNGRIDDED
     1 DATA)',51,ZBLANK,ITT)
0023          CALL FORM (FMAPX,FMT1(3),0,1,10)
0024          CALL FORM (FMAPY,FMT1(4),1,1,10)
0025          WRITE(INTUN,FMT1) CNAM(1),FMAPX,FMAPY,SN,SM,IBXOPT,KEY,ICODE(1)
0026          CALL FORM (ZBLANK,FMT2(3),0,1,10)
0027          WRITE(INTUN,FMT2) CNAM(1),ZBLANK,KEY,ICODE(20)
0028          RETURN
0029          ENTRY CCTR2
      C       ------ -----                      FMT CARD
0030          WRITE(INTUN,2002) CNAM(2),KEY,ICODE(2)
0031    2002  FORMAT(A4,76X,I2,I3)
0032          CALL TTINAA ('FORMAT, INCLUDING PARENTHESES',29,FRMT,80,ITT)
0033          WRITE(INTUN,2000) FRMT,KEY,ICODE(18)
0034    2000  FORMAT(20A4,I2,I3)
0035          RETURN
0036          ENTRY CCTR3
      C       ----------                         GRID CARD
0037          CALL TTINSR('# DATA POINTS (NO LIMIT)',24,XNPTS,ITT)
0038          NPTS=XNPTS
0039          CALL TTINSI('# OF GRID DIVISIONS IN X (DOWNPAGE)',35,NX,ITT)
0040          CALL TTINSI('# OF GRID DIVISIONS IN Y (ACROSS PAGE)',38,NY,ITT)
```

```
0041            CALL TTINSI('POSITION (1-3) OF X IN DATA',27,IXPOS,ITT)
0042            CALL TTINSI('POSITION (1-3) OF Y IN DATA',27,IYPOS,ITT)
0043            CALL TTINSI('POSITION (1-3) OF Z IN DATA',27,IZPOS,ITT)
0044            WRITE(INTUN,2003) CNAM(3),NPTS,NX,NY,IXPOS,IYPOS,IZPOS,KEY,ICODE(3)
0045    2003    FORMAT(A4,6X,6I10,10X,I2,I3)
0046            RETURN
0047            ENTRY CCTR4
        C       -----------                         LABL CARD (NOT USED,ICODE=500)
        C       ENTRY LABL(KEY,IO2)
        C30     CALL TTINSR('HEIGHT (IN) OF TITLE LINES',26,HTI,ITT)
        C       CALL TTINSI('MAX # CHARACTERS IN TITLE (<81)',31,MXCI,ITT)
        C       XWD=MXCI*HTI*6/7+1.0
        C       CALL TTINSI('# LINES IN TITLE',16,NTI,ITT)
        C       CALL TTINSR('HEIGHT (IN) OF TEXT LINES',25,HTX,ITT)
        C       NLIN=INT((28.-NTI*1.5*HTI-1.0)/(1.5*HTX))
        C       WRITE(ITT,221)NLIN
        C221    FORMAT(' MAX # LINES TEXT WITH THESE HEIGHTS IS',I4,
        C       1 '.'/' DO YOU WANT TO CHANGE HEIGHTS?',$)
        C       READ(ITT,100)QR
        C       IF(QR.EQ.Y) GO TO 30
        C       CALL TTINSI('MAX # LINES OF TEXT (<MAX+1 ABOVE)',34,NTX,ITT)
        C       YWD=NTI*1.5*HTI+NTX*1.5*HTX+1.0
        C32     CALL TTINSI('MAX # CHARACTERS IN TEXT LINES (<81)',36,MXCX,ITT)
        C       IF(MXCX*HTX*6/7+1.0.LE.XWD) GO TO 31
        C       WRITE(ITT,222)MXCX
        C222    FORMAT(' MAX # TEXT CHAR PER LINE OF',I4,' TOO LARGE')
        C       GO TO 32
        C31     WRITE(IO2,223)XWD,YWD,KEY
        C223    FORMAT('LABL',6X,2F10.3,50X,I2,'300')
        C       WRITE(IO2,224)NTI,HTI,NTX,HTX,MXCI,MXCX,KEY
0048            RETURN
0049            ENTRY CCTR5
        C       -----------                           LEVL CARD
0050            CALL TTINAA('CONTOUR INTERVAL CONSTANT ?',27,IAN,1,ITT)
0051            IF (IAN.EQ.'N') GOTO 100
0052            CALL TTINSR('MINIMUM Z VALUE TO BE CONTOURED',31,SLVL,ITT)
0053            CALL TTINSR('CONTOUR INTERVAL (>0)',21,DLVL,ITT)
0054            CALL TTINSR('MAXIMUM Z VALUE TO BE CONTOURED',31,ELVL,ITT)
0055            CALL FORM (SLVL,FMT5(2),0,1,10)
0056            CALL FORM (DLVL,FMT5(3),1,1,10)
0057            CALL FORM (ELVL,FMT5(4),1,1,10)
0058            WRITE(INTUN,FMT5) CNAM(5),SLVL,DLVL,ELVL,KEY,ICODE(5)
0059            RETURN
0060    100     J=ICODE(5)-1
0061    110     J=J+1
0062            CALL TTINSR('Z VALUE TO BE CONTOURED',23,SLVL,ITT)
0063            CALL TTINSR('LABEL FOR THIS CONTOUR (0, IF = Z VALUE)',40,
              1 BLABEL,ITT)
0064            CALL FORM (SLVL,FMT6(2),0,1,10)
0065            CALL FORM (BLABEL,FMT6(4),1,1,10)
0066            WRITE(INTUN,FMT6) CNAM(5),SLVL,BLABEL,KEY,J
0067            CALL TTINAA('MORE CONTOUR LINES ?',20,IAN,1,ITT)
0068            IF (IAN.NE.'N') GOTO 110
0069            RETURN
0070            ENTRY CCTR6
        C       -----------                           MESH CARD
```

```
0071          CALL TTINSR('HEIGHT OF PLOTTED GRID VALUES, IN INCHES',40,HEIGT,
             3 ITT)
0072          CALL TTINSI('# DIGITS FOLLOWING DECIMAL POINT',32,NDEC,ITT)
0073          WRITE (INTUN,2006) CNAM(6),HEIGT,NDEC,KEY,ICODE(6)
0074    2006  FORMAT(A4,6X,F10.4,I10,50X,I2,I3)
0075          RETURN
0076          ENTRY CCTR7
      C       ----------                          NAOR CARD
0077          CALL TTINSI('# DATA POINTS USED TO DETERMINE EACH GRID POINT',
             3 47,NABRS,ITT)
0078          CALL TTINSI('ORDER OF BIVARIATE POLYNOMIAL (1-4) USED FOR GRID POI
             4NT DETERMINATION',69,IORD,ITT)
0079          CALL TTINSR ('RADIUS OF NEIGHBOURHOOD (DATA UNITS)',36,RADIUS,ITT)
0080          CALL FORM (RADIUS,FMT7(3),0,1,10)
0081          WRITE(INTUN,FMT7) CNAM(7),NABRS,IORD,RADIUS,KEY,ICODE(7)
0082          RETURN
0083          ENTRY CCTR8
      C       ----------                          NDEC CARD
0084          CALL TTINSI('# DIGITS FOLLOWING DECIMAL POINT IN PLOTTED #S',
             5 46,NDEC,ITT)
0085          WRITE(INTUN,2008) CNAM(8),NDEC,KEY,ICODE(8)
0086    2008  FORMAT(A4,6X,I10,60X,I2,I3)
0087          RETURN
0088          ENTRY CCTR9 (I)
      C       ------ ----                        PNCH, TICK, TREN CARDS
0089          WRITE(INTUN,2002) CNAM(I),KEY,ICODE(I)
0090          RETURN
0091          ENTRY CCTR10
      C       -----------                         POST CARD
0092          CALL TTINSR('HEIGHT OF PLOTTED INPUT DATA VALUES, IN INCHES',46,
             6 PSTHT,ITT)
0093          CALL TTINSI ('PLOT INPUT DATA VALUES (1), OR PUT A + MARK ONLY
             1 (0)',52,NVALU,ITT)
0094          WRITE(INTUN,2010) CNAM(10),PSTHT,NVALU,KEY,ICODE(10)
0095    2010  FORMAT(A4,6X,F10.4,I10,50X,I2,I3)
0096          RETURN
0097          ENTRY CCTR11
      C       -----------                         PRNT CARD
0098          CALL TTINSI('PRINT UNGRIDDED DATA POINTS (1), OR NOT (0)',43,IP,
             A ITT)
0099          WRITE(INTUN,2008) CNAM(11),IP,KEY,ICODE(11)
0100          RETURN
0101          ENTRY CCTR12
      C       -----------                         RDGD CARD
0102          CALL TTINSI('# COLUMNS IN DATA GRID (X)',26,NX,ITT)
0103          CALL TTINSI('# ROWS IN DATA GRID (Y)',23,NY,ITT)
0104          CALL TTINSI('1-READ BY COL,TOP TO BOTT; 2-READ BY ROWS,LEFT TO RIG
             7HT',55,MANNR,ITT)
0105          WRITE(INTUN,2012) CNAM(12),NX,NY,MANNR,KEY,ICODE(12)
0106    2012  FORMAT(A4,6X,3I10,40X,I2,I3)
0107          RETURN
0108          ENTRY CCTR13
      C       -----------                         REGN CARD
0109          CALL TTINSR('MINIMUM X VALUE OF GRID, IN DATA UNITS',38,XMPMN,
             7 ITT)
0110          CALL TTINSR('MAXIMUM X VALUE OF GRID, IN DATA UNITS',38,XMPMX,
```

```
          8 ITT)
0111          CALL TTINSR('MINIMUM Y VALUE OF GRID, IN DATA UNITS',38,YMPMN,
          9 ITT)
0112          CALL TTINSR('MAXIMUM Y VALUE OF GRID, IN DATA UNITS',38,YMPMX,
          1 ITT)
0113          CALL FORM (XMPMN,FMT13(2),0,1,10)
0114          CALL FORM (XMPMX,FMT13(3),1,1,10)
0115          CALL FORM (YMPMN,FMT13(4),1,1,10)
0116          CALL FORM (YMPMX,FMT13(5),1,1,10)
0117          WRITE(INTUN,FMT13) CNAM(13),XMPMN,XMPMX,YMPMN,YMPMX,KEY,ICODE(13)
0118          RETURN
0119          ENTRY CCTR14
      C       ------------                    STOP CARD
0120          WRITE(INTUN,2002) CNAM(14),KEY0,ICODE(14)
0121          RETURN
0122          ENTRY CCTR16
      C       ------------                    TITL CARD
0123          WRITE(INTUN,2002) CNAM(16),KEY,ICODE(16)
0124          CALL TTINAA ('80-CHAR TITLE FOR THE MAP',25,TTL,80,ITT)
0125          WRITE(INTUN,2000) TTL,KEY,ICODE(19)
0126          RETURN
0127          END
```

CTRL21.FTN /TR:BLOCKS/WR

```
0001            SUBROUTINE COPY (IO1,IO2)
        C       ---------------
0002            DIMENSION CARD(20)
0003    1000    FORMAT(20A4)
0004    2200    FORMAT(1X,20A4)
0005    2201    FORMAT(1X,'0.........1.........2.........3.........4.........',
               1 '5.........6.........7.........8')
        C
0006    500     READ (IO1,1000,END=9000) CARD
0007            WRITE(IO2,1000) CARD
0008            GOTO 500
0009    9000    RETURN
0010            ENTRY COPIOW (IO1,IOW)
        C       ------------
0011            REWIND IO1
0012            WRITE(IOW,2201)
0013    600     READ (IO1,1000,END=9100) CARD
0014            WRITE(IOW,2200) CARD
0015            GOTO 600
0016    9100    WRITE(IOW,2201)
0017            REWIND IO1
0018            RETURN
0019            END
```

```
0001            SUBROUTINE RWCOP (IO1,IO2)
        C       ----------------
0002            CALL COPY (IO1,IO2)
        C       ---------
0003            REWIND IO1
0004            REWIND IO2
0005            CALL COPY (IO2,IO1)
        C       ---------
0006            REWIND IO1
0007            REWIND IO2
0008            RETURN
0009            END
```

```
0001          SUBROUTINE READAL
      C       ----------------
      C   READ CONTROL CARDS AND DIRECT PROGRAM TO 'WRITAL' FOR EACH MAP
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
             A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          DIMENSION FILEN8(20),TITL(20),FILEN(20)
      C
0006          DATA FILEN/4HDB1:,4HCONT,4HUR.D,2HAT,'000,15*' '/
0007    1000  FORMAT(20A4)
0008    1001  FORMAT(A4,6X,6F10.0)
0009    2204  FORMAT(' *** NO CONTROL CARD NAMED : ',A4,' ***')
0010    2206  FORMAT(' *** NO ''STOP'' CARD IN CONTROL FILE ***')
      C
      C------------------- READ FILENAMES AND TITLE FROM CONTROL FILE, WRITE
      C                    ON NEW CONTROL FILE
0011          READ (ICARD,1000) FILEN8
0012          WRITE(JCARD,1000) FILEN
0013          DO 100 I=1,4
0014          READ (ICARD,1000) TITL
0015          WRITE(JCARD,1000) TITL
0016    100   CONTINUE
      C-------------------- OPEN FILES
0017          OPEN (UNIT=IUNIT,NAME=FILEN8,TYPE='OLD')
0018          OPEN (UNIT=IWRIT,NAME='DB1:CONTUR.DAT',TYPE='NEW')
0019          OPEN (UNIT=INTUN,NAME='DB1:FOR006.DAT',TYPE='SCRATCH',
             1 ACCESS='DIRECT',RECORDSIZE=10)
      C------------------- READ REMAINING OF CONTROL CARDS
0020    200   READ (ICARD,1001,END=9000) CNM,BUFIN
0021          DO 210 I=1,17
0022          IF (CNM.EQ.CNAM(I)) GOTO 300
0023    210   CONTINUE
0024          WRITE(JPNTR,2204) CNM
0025          GOTO 200
0026    300   PRESEN(I)=.FALSE.
0027          GOTO (301,302,303,200,305,306,306,306,306,306,306,312,313,314,315,
             1 316,317), I
0028    301   CALL RCTR1
0029          CALL WRITAL
      C       -----------
0030          DO 120 I=1,17
0031    120   PRESEN(I)=.TRUE.
0032          GOTO 200
0033    302   CALL RCTR2
0034          GOTO 200
0035    303   CALL RCTR3
0036          GOTO 200
0037    305   CALL RCTR5
0038          GOTO 200
0039    306   CALL RCTR6 (I-5)
0040          GOTO 200
0041    312   CALL RCTR12
0042          GOTO 200
0043    313   CALL RCTR13
```

```
0044            GOTO 200
0045    314     CALL RCTR6 (7)
0046            GOTO 400
0047    315     CALL RCTR6 (8)
0048            GOTO 200
0049    316     CALL RCTR16
0050            GOTO 200
0051    317     CALL RCTR6 (9)
0052            GOTO 200
        C------------------- CLOSE FILES
0053    400     CALL WCTR6 (7)
0054            CLOSE (UNIT=ICARD)
0055            CLOSE (UNIT=IUNIT)
0056            CLOSE (UNIT=IWRIT)
0057            RETURN
0058    9000    WRITE(JPNTR,2206)
0059            STOP ' *** ERROR DETECTED BY ROUTINE : READAL ***'
0060            END
```

```
0001          SUBROUTINE RCTR1
      C       ----------------                    DRAW CARD
      C   READ CONTROL CARDS
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
     A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0006          COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0007          COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0008          COMMON NNY2  ,NY    ,NY2   ,SM    ,SN    ,XMPMN ,XMPMX ,XORIG
0009          COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0010          COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0011          INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD
0012          COMMON/BIDON/Z(2500),SLVL(200),BLABEL(200)
      C
0013    1000  FORMAT(20A4)
0014    1001  FORMAT(A4,6X,6F10.0)
0015    1002  FORMAT(A4,6X,F10.0,10X,F10.0)
0016    2200  FORMAT(' *** END OF CONTROL FILE NOT EXPECTED ***')
      C
0017          FMAPX=BUFIN(1)
0018          IF (FMAPX.LE.0.) FMAPX=10.
0019          FMAPY=BUFIN(2)
0020          IF (FMAPY.LE.0.) FMAPY=10.
0021          SN=BUFIN(3)
0022          SM=BUFIN(4)
0023          IBXOPT=BUFIN(5)
0024          READ (ICARD,1001,END=9000) CNM,BUFIN
0025          ZBLANK=BUFIN(1)
0026          RETURN
0027          ENTRY RCTR2
      C       ----------                          FMT CARD
0028          READ (ICARD,1000,END=9000) FRMT
0029          RETURN
0030          ENTRY RCTR3
      C       -----------                         GRID CARD
0031          NPTS=BUFIN(1)
0032          IF (BUFIN(2).GT.0) GOTO 100
0033          IF (NX.GT.0) GOTO 110
0034          NX=20
0035          GOTO 110
0036    100   NX=BUFIN(2)
0037    110   IF (BUFIN(3).GT.0.) GOTO 120
0038          IF (NY.GT.0) GOTO 130
0039          NY=20
0040          GOTO 130
0041    120   NY=BUFIN(3)
0042    130   IF (NPTS.LT.0) RETURN
0043          IF (BUFIN(4).LE.0.) GOTO 140
0044          IXPOS=BUFIN(4)
0045          IYPOS=BUFIN(5)
0046          IZPOS=BUFIN(6)
0047          RETURN
0048    140   IXPOS=1
```

```
0049          ITYPOS=2
0050          ITYPOS=3
0051          RETURN
0052          ENTRY RCTR5
      C       -----------                    LEVL CARD
0053          KLEVL=1
0054          SLVL(1)=BUFIN(1)
0055          IF (BUFIN(2).LE.0) GOTO 200
0056          SLVL(2)=BUFIN(2)
0057          SLVL(3)=BUFIN(3)
0058          RETURN
0059    200   KLEVL=2
0060          BLABEL(1)=BUFIN(3)
0061          J=1
0062    210   J=J+1
0063          READ (ICARD,1002,END=9000) CNM,SLVL(J),BLABEL(J)
0064          IF (CNM.EQ.CNAM(5)) GOTO 210
0065          NLVL=J-1
0066          BACKSPACE ICARD
0067          RETURN
0068          ENTRY RCTR6 (I)
      C       -----------                    CARDS : MESH, NAOR, NDEC, PNCH, POST,
      C                                              PRNT, STOP, TICK, TREN
0069          BACKSPACE ICARD
0070          READ (ICARD,1000,END=9000) (AAA(K,I),K=1,20)
0071          RETURN
0072          ENTRY RCTR12
      C       ------------                   RDGD CARD
0073          NNX=BUFIN(1)
0074          NNY=BUFIN(2)
0075          MANNR=BUFIN(3)
0076          IF (MANNR.LE.0) MANNR=2
0077          RETURN
0078          ENTRY RCTR13
      C       ------------                   REGN CARD
0079          XMPMN=BUFIN(1)
0080          XMPMX=BUFIN(2)
0081          YMPMN=BUFIN(3)
0082          YMPMX=BUFIN(4)
0083          RETURN
0084          ENTRY RCTR16
      C       ------------                   TITL CARD
0085          READ (ICARD,1000,END=9000) TTL
0086          RETURN
0087    9000  WRITE(JPNTR,2200)
0088          STOP ' *** ERROR DETECTED BY ROUTINE : RCTR1 ***'
0089          END
```

```
0001          SUBROUTINE WRITAL
      C       ----------------
      C   READ DATA FILE, CHECK IF MAP IS IN SEVERAL PARTS
      C
0002          DIMENSION A(3)
0003          LOGICAL*1 PRESEN,ISEV0,ISEV1
0004          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0005          COMMON/BIDON/Z(2500)
0006          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
          A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0007          COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0008          COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0009          COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0010          COMMON NNY2  ,NY    ,NY2   ,SM    ,SN    ,XMPNN ,XMPMX ,XORIG
0011          COMMON YCALC ,YMPNN ,YMPMX ,YORIG ,ZBLANK
0012          COMMON XMPNN2,XMPMX2,YMPNN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0013          INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD,MXPGD1,MXPGD0,NXNY
0014          INTEGER*4 I
0015          DIMENSION FRMT20(20)
0016          DATA FRMT20/4H(A4,,4H6X,3,4H(E13,4H.7,1,3HX)),15*' '/
0017          DATA MXPGD1/2500/
      C
0018    2203  FORMAT(' *** ERROR WHILE READING PREGRIDDED DATA FROM INPUT DATA',
          3 ' FILE ***'/' RECORD NO :',I7,'   Z =',3E12.4)
0019    2204  FORMAT(' *** ERROR WHILE READING UNGRIDDED DATA FROM INPUT DATA',
          4 ' FILE ***'/' RECORD NO :',I7,'   ALPHA =',A4,'   X,Y,Z =',
          5 3E12.4)
0020    2205  FORMAT(' *** END OF INPUT DATA FILE NOT EXPECTED WHEN READING',
          5 ' PREGRIDDED DATA ***')
0021    2206  FORMAT(' *** ERROR WHILE WRITING ON INTERM FILE, ON RECORD NO :',
          6 I7,' ***')
0022    2207  FORMAT(' *** ERROR WHILE READING INTERM FILE, ON RECORD NO :',I7,
          7 ' ***')
0023    2208  FORMAT(' *** NO MORE DATA SET ON INPUT DATA FILE, WHEN READING',
          8 ' UNGRIDDED DATA ***')
      C
0024          MXPGD0=MXPGD
0025          IF (PRESEN(12).EQ..TRUE.) GOTO 300
      C------------------- PREGRIDDED DATA, RDGD CARD ENCOUNTERED
0026          IGRI=0
0027          MXPGD=MXPGD1
0028          GOTO (340,330) MANNR
0029    330   NR=NNX
0030          NREC=NNY
0031          GOTO 350
0032    340   NR=NNY
0033          NREC=NNX
0034    350   CLOSE (UNIT=INTUN)
0035          OPEN (UNIT=INTUN,NAME='DB1:FOR006.DAT',TYPE='SCRATCH',
          A ACCESS='DIRECT',RECORDSIZE=NR,ASSOCIATEVARIABLE=KNTUN)
0036          KNTUN=1
0037          WRITE(INTUN'KNTUN,ERR=9200) IGRI,NREC
0038          DO 100 I=1,NREC
0039          READ (IUNIT,FRMT,ERR=9000,END=9100) (Z(J),J=1,NR)
0040    100   WRITE(INTUN'KNTUN,ERR=9200) (Z(J),J=1,NR)
0041          ISEV0=.FALSE.
```

```
0042            NNXO=NNX
0043            NNYO=NNY
0044            GOTO 700
                C------------------- UNGRIDDED DATA, GRID CARD ENCOUNTERED
0045    300     IF (PRESEN(3).EQ..TRUE.) GOTO 800
0046            CALL MOVE (FRMT20,FRMT2,80)
0047            IF (NPTS.LT.0) GOTO 370
0048            CLOSE (UNIT=INTUN)
0049            OPEN (UNIT=INTUN,NAME='DB1:FOR006.DAT',TYPE='SCRATCH',
              1 ACCESS='DIRECT',RECORDSIZE=4,ASSOCIATEVARIABLE=KNTUN)
0050            IF (NPTS.EQ.0) GOTO 380
                C------------------- NPTS > 0
0051    390     IGRI=1
0052            KNTUN=2
0053            DO 110 I=1,NPTS
0054            READ (IUNIT,FRMT,ERR=9050,END=410) ALPHA,A
0055            IF (ALPHA.EQ.'END') GOTO 410
0056            Z(1)=A(IXPOS)
0057            Z(2)=A(IYPOS)
0058            Z(3)=A(IZPOS)
0059    110     WRITE(INTUN'KNTUN,ERR=9200) ALPHA,(Z(K),K=1,3)
0060            I=0
0061            READ (IUNIT,FRMT,ERR=9050,END=405) ALPHA
0062            IF (ALPHA.EQ.'END') GOTO 405
0063            BACKSPACE IUNIT
0064    405     KNTUN=1
0065            WRITE(INTUN'KNTUN,ERR=9200) IGRI,NPTS
0066            ISEVO=.FALSE.
0067            GOTO 600
0068    410     NPTS=I-1
0069            IF (NPTS.LE.0) GO TO 9500
0070            GO TO 405
                C------------------- NPTS = 0
0071    380     NPTS=200000
0072            GO TO 390
                C------------------- NPTS < 0
0073    370     KNTUN=1
0074            READ (INTUN'KNTUN,ERR=9400) IGRI,NPTS
0075            IF (IGRI.EQ.1) GOTO 610
0076            GOTO 380
                C------------------- TEST REGN CARD
0077    600     IF (PRESEN(13).EQ..FALSE.) GOTO 610
0078            XMPMX=-1.E14
0079            YMPMX=-1.E14
0080            XMPMN=1.E14
0081            YMPMN=1.E14
0082            KNTUN=1
0083            DO 650 I=1,NPTS
0084            READ (INTUN'KNTUN,ERR=9400) ALPHA,(Z(K),K=1,3)
0085            IF (Z(IXPOS).GT.XMPMX) XMPMX=Z(IXPOS)
0086            IF (Z(IXPOS).LT.XMPMN) XMPMN=Z(IXPOS)
0087            IF (Z(IYPOS).GT.YMPMX) YMPMX=Z(IYPOS)
0088            IF (Z(IYPOS).LT.YMPMN) YMPMN=Z(IYPOS)
0089    650     CONTINUE
0090    610     NNXO=NX+1
0091            NNYO=NY+1
```

```
0092          GOTO 700
      C---------- NO RDGD NOR GRID CARD
0093    800   KNTUN=1
0094          READ (INTUN'KNTUN,ERR=9400) IGRI,NPTS
0095          IF (IGRI.EQ.0) MXPGD=MXPGD1
      C---------- MAP IN ONE OR SEVERAL PARTS
0096    700   BNXNY=FLOATI(NNX0)*FLOATI(NNY0)
0097          NXNY=BNXNY
0098          IF (NXNY.GT.MXPGD) GO TO 720
0099          IF (FMAPY.GT.YCALC) GOTO 720
0100          ISEV1=.TRUE.
0101          CALL WRIONE
0102          GOTO 900
0103    720   ISEV1=.FALSE.
0104          CALL WRISEV
0105    900   ISEV0=ISEV1
0106          MXPGD=MXPGD0
0107          RETURN
      C---------- ERRORS
0108    9000  WRITE(JPNTR,2203) I,(Z(K),K=1,3)
0109          GOTO 9900
0110    9050  WRITE(JPNTR,2204) I,ALPHA,A
0111          GO TO 9900
0112    9100  WRITE(JPNTR,2205)
0113          GOTO 9900
0114    9200  WRITE(JPNTR,2206) KNTUN
0115          GOTO 9900
0116    9400  WRITE(JPNTR,2207) KNTUN
0117          GO TO 9900
0118    9500  WRITE(JPNTR,2208)
0119    9900  STOP ' *** ERROR DETECTED BY ROUTINE : WRITAL ***'
0120          END
```

```
0001            SUBROUTINE WRIONE
                C  ----------------
                C   MAP TO DRAW IS IN ONE PART ONLY
                C
0002            LOGICAL*1 PRESEN,ISEV0,ISEV1
0003            COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
               A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0004            COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0005            COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0006            COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0007            COMMON NNY2  ,NY    ,NY2   ,SM    ,SN    ,XMPMN ,XMPMX ,XORIG
0008            COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0009            COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0010            INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD
                C
0011            FMAPX2=FMAPX
0012            FMAPY2=FMAPY
0013            NNX2=NNX0
0014            NNY2=NNY0
0015            YORIG=DISTY0
0016            D=YORIG+FMAPY
0017            IF (D.GT.YCALC) GOTO 300
                C-------------------- MAP IS DRAWN UP THE PREVIOUS ONE
0018            XORIG=DISTX0
0019            X=XORIG+FMAPX
0020            DISTX1=AMAX1 (X,DISTX1)
0021            DISTY0=D+2.
0022            GOTO 310
                C-------------------- MAP IS DRAWN RIGHT THE PREVIOUS ONE, AND DOWN
0023    300     XORIG=DISTX1+2.
0024            DISTX0=XORIG
0025            DISTX1=XORIG+FMAPX
0026            YORIG=0.
0027            DISTY0=FMAPY+2.
                C-------------------- PREGRIDDED DATA
0028    310     IF (IGRI.EQ.1) GOTO 320
0029            CALL RWDATP (1,NNX2,1,NNY2)
                C  -----------
0030            CALL WRIPRE
                C  -----------
0031            RETURN
                C-------------------- DATA TO GRID
0032    320     CALL RWDATG (1,NNX2,1,NNY2)
                C  -----------
0033            IF (NPTS2.EQ.0) RETURN
0034            CALL WRIGRI
                C  -----------
0035            RETURN
0036            END
```

```
0001            SUBROUTINE WRISEV
                C       ----------------
                C   MAP TO DRAW IS IN SEVERAL PARTS
                C
0002            LOGICAL*1 PRESEN,ISEV0,ISEV1
0003            COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004            COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
               A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005            COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0006            COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,MPTS
0007            COMMON MPTS2 ,MNX   ,MNX0  ,MNX2  ,MX    ,MX2   ,MNY   ,MNY0
0008            COMMON MNY2  ,MY    ,MY2   ,SM    ,SN    ,XMPMN ,XMPMX ,XORIG
0009            COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0010            COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0011            INTEGER*4 MPTS,MPTS2,KNTUN,MXPGD,NXNY
0012            DATA NXMAX0/50/,NXMAX1/75/
                C
0013            IBXOPT=IBXOPT+2
0014            C=FMAPX/FLOATI(MNX0-1)
0015            B=FMAPY/FLOATI(MNY0-1)
0016            MX20=1
0017            MY20=1
0018            MY21=1
0019            DISTY1=YCALC
0020            IF (FMAPY-YCALC) 500,500,600
0021    600     MY3=YCALC/B-0.5
0022            GOTO 502
0023    500     MY3=MNY0
0024    502     MY11=MY21
0025            MY21=MY21+MY3-1
0026            MY20=MY11
                C--------------------
0027            MNX2=MNX0-MX20+1
0028            IF (MNX2.LE.MXMAX1) GOTO 510
0029            MNX2=MXMAX0
0030    510     MX10=MX20
0031            MX20=MX20+MNX2-1
0032            FMAPX2=FLOATI(MNX2-1)*C
0033            XORIG=DISTX1
0034            DISTX1=DISTX1+FMAPX2
0035            IF (MX20.LT.MNX0) DISTX1=DISTX1-2.*C
                C--------------------
0036    525     MNY2=MY21-MY20+1
0037            BMXNY=FLOATI(MNX2)*FLOATI(MNY2)
0038            MXNY=BMXNY
0039            IF (MXNY.LT.MXPGD) GOTO 515
0040            MNY2=FLOATJ(MXPGD)/FLOATI(MNX2)-0.5
0041    515     MY10=MY20
0042            MY20=MY20+MNY2-1
0043            FMAPY2=FLOATI(MNY2-1)*B
0044            YORIG=DISTY1-FMAPY2
0045            DISTY1=YORIG
0046            IF (MY20.LT.MNY0) DISTY1=DISTY1+2.*B
0047            IF (IGRI.EQ.1) GOTO 300
                C------------------- PREGRIDDED DATA
0048            CALL RWDATP (MX10,MX20,MY10,MY20)
```

```
        C       -----------
0049            CALL WRIPRE
        C       -----------
0050            GOTO 310
        C-------------------- DATA TO GRID
0051    300     CALL RWDATG (NX10,NX20,NY10,NY20)
        C       -----------
0052            IF (NPTS2.EQ.0) GO TO 310
0053            CALL WRIGRI
        C       -----------
        C--------------------
0054    310     IF (NY20.GE.NY21) GOTO 320
0055            NY20=NY20-2
0056            GOTO 525
0057    320     DISTY1=YCALC
0058            IF (NX20.GE.NNX0) GOTO 330
0059            NX20=NX20-2
0060            NY21=NY11
0061            GOTO 502
0062    330     NX20=1
0063            NY21=NY21-2
0064            NY3=NNY0-NY21+1
0065            DISTX1=DISTX1+2.
0066            IF (NY3.LE.3) GOTO 610
0067            FMAP=FLOATI(NY3-1)*B
0068            IF (FMAP-YCALC) 502,502,600
        C--------------------
0069    610     DISTX0=DISTX1
0070            DISTY0=0.
0071            RETURN
0072            END
```

```
0001          SUBROUTINE RWDATP (NX10,NX20,NY10,NY20)
      C       ---------------
      C   READ INTERMEDIARY DATA FILE, WRITE FINAL DATA FILE, FOR PREGRIDDED DATA
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/BIDON/Z(2500)
0005          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
             A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0006          COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0007          COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0008          COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0009          COMMON NNY2  ,NY    ,NY2   ,SM    ,SN    ,XMPMN ,XMPMX ,XORIG
0010          COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0011          COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0012          INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD
0013          INTEGER*4 I
0014          DIMENSION FRMT10(20),FRMT30(7)
0015          DATA FRMT10/1H(,' ',4H(6E1,4H3.7/,2H),,' ',4HE13.,2H7),12*' '/
0016          DATA FRMT30/1H(,' ',4HE13.,2H7,,' ',4HX,I5,4H,I3)/
0017          DATA NB6/6/
      C
0018    1000  FORMAT(I4)
0019    2000  FORMAT(6E13.7,2X,I5,I3)
0020    2001  FORMAT(80X,I5,I3)
0021    2200  FORMAT(' *** ERROR WHILE READING INTERM FILE, ON RECORD NO :',I7,
            1 ' ***')
0022    2203  FORMAT(' *** ERROR WHILE WRITING FINAL DATA FILE, ON ROW NO :',I7,
            3 ' ***')
      C
0023          FUNX(L)=XMPMN+(L-1)*(XMPMX-XMPMN)/NX
0024          FUNY(L)=YMPMX-(L-1)*(YMPMX-YMPMN)/NY
      C
0025          IF ((PRESEN(12).EQ..TRUE.).AND.(ISEV0.EQ..TRUE.).AND.
            1 (ISEV1.EQ..TRUE.)) RETURN
0026          GOTO (200,100) MANNR
      C-------------------- WRITE DATA BY ROWS
0027    100   NROW1=NY10
0028          NROW2=NY20
0029          NCOL=NNX2
0030          NCOL1=NX10
0031          NCOL2=NX20
0032          NCOLMX=NNX0
0033          GOTO 400
      C-------------------- WRITE DATA BY COLUMNS
0034    200   NROW1=NX10
0035          NROW2=NX20
0036          NCOL=NNY2
0037          NCOL1=NY10
0038          NCOL2=NY20
0039          NCOLMX=NNY0
      C--------------------
0040    400   NCARD=NCOL/6
0041          NSUP=NCOL-NCARD*6
0042          IF (NSUP.GT.0) GOTO 450
0043          NCARD2=NCARD-1
```

```
0044            ENCODE (4,1000,FRMT10(2)) NCARD2
0045            ENCODE (4,1000,FRMT10(6)) NB6
0046            GOTO 410
0047    450     ENCODE (4,1000,FRMT10(2)) NCARD
0048            ENCODE (4,1000,FRMT10(6)) NSUP
0049            NCARD1=NCARD+1
0050            NSPACE=80-NSUP*13
0051            ENCODE (4,1000,FRMT30(2)) NSUP
0052            ENCODE (4,1000,FRMT30(5)) NSPACE
0053    410     CALL MOVE (FRMT10,FRMT2,80)
0054            IROW=0
0055            KNTUN=NROW1+1
0056            DO 310 I=NROW1,NROW2
0057            IROW=IROW+1
0058            READ (INTUN'KNTUN,ERR=9000) (Z(K),K=1,NCOLMX)
0059            J1=NCOL1
0060            DO 420 J=1,NCARD
0061            J2=J1+5
0062            WRITE(IWRIT,2000,ERR=9200) (Z(K),K=J1,J2),IROW,J
0063    420     J1=J2+1
0064            IF (NSUP.EQ.0) GOTO 310
0065            WRITE(IWRIT,FRMT30,ERR=9200) (Z(K),K=J1,NCOL2),IROW,NCARD1
0066    310     CONTINUE
0067            RETURN
        C
0068            ENTRY RWDATG (NX10,NX20,NY10,NY20)
        C       ------------
        C   READ INTERMEDIARY FILE, WRITE FINAL DATA FILE, FOR UNGRIDDED DATA
        C
0069            NX2=NNX2-1
0070            NY2=NNY2-1
0071            XMPMN2=FUNX(NX10)
0072            XMPMX2=FUNX(NX20)
0073            YMPMN2=FUNY(NY20)
0074            YMPMX2=FUNY(NY10)
0075            NPTS2=NPTS
0076            IF ((PRESEN(3).EQ.,TRUE.).AND.(ISEV0.EQ.,TRUE.).AND.
              1 (ISEV1.EQ.,TRUE.)) RETURN
0077            X0=(XMPMX2-XMPMN2)*0.25
0078            X1=XMPMN2-X0
0079            X2=XMPMX2+X0
0080            Y0=(YMPMX2-YMPMN2)*0.25
0081            Y1=YMPMN2-Y0
0082            Y2=YMPMX2+Y0
0083            NPTS2=0
0084            KNTUN=2
0085            DO 110 I=1,NPTS
0086            READ (INTUN'KNTUN,ERR=9000) ALPHA,(Z(J),J=1,3)
0087            IF (NPTS.LE.2000) GO TO 120
0088            IF (Z(1).LT.X1) GO TO 110
0089            IF (Z(1).GT.X2) GO TO 110
0090            IF (Z(2).LT.Y1) GO TO 110
0091            IF (Z(2).GT.Y2) GO TO 110
0092    120     WRITE(IWRIT,FRMT2,ERR=9600) ALPHA,(Z(J),J=1,3)
0093            NPTS2=NPTS2+1
0094    110     CONTINUE
```

```
0095            RETURN
        C------------------- ERRORS
0096    9000    WRITE(JPNTR,2200) KNTUN
0097            GOTO 9900
0098    9200    WRITE(JPNTR,2203) IROW
0099            GOTO 9900
0100    9600    WRITE(JPNTR,2203) I
0101    9900    STOP ' *** ERROR DETECTED BY ROUTINE : RWDATP/RWDATG ***'
0102            END
```

```
0001            SUBROUTINE WRIPRE
        C       ----------------
        C   WRITE CONTROL CARDS FOR ONE PART OF MAP, FOR PREGRIDDED DATA
        C
0002            LOGICAL*1 PRESEN,ISEV0,ISEV1
0003            COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
               A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
        C
0004            IF ((PRESEN(12).EQ..TRUE.).AND.(ISEV0.EQ..TRUE.).AND.
               1 (ISEV1.EQ..TRUE.)) GOTO 100
0005            CALL WCTR2
0006            IF (PRESEN(11).EQ..FALSE.) CALL WCTR6 (6)
0007            CALL WCTR12
0008            GO TO 110
0009    100     IF (PRESEN(11).EQ..FALSE.) CALL WCTR6 (6)
0010    110     IF (PRESEN(5).EQ..FALSE.) CALL WCTR5
0011            IF (PRESEN(6).EQ..FALSE.) CALL WCTR6 (1)
0012            IF (PRESEN(8).EQ..FALSE.) CALL WCTR6 (3)
0013            IF (PRESEN(15).EQ..FALSE.) CALL WCTR6 (8)
0014            IF (PRESEN(16).EQ..FALSE.) CALL WCTR16
0015            CALL WCTR1
0016            RETURN
        C
0017            ENTRY WRIGRI
        C       ------------
        C   WRITE CONTROL CARDS FOR ONE PART OF MAP, FOR UNGRIDDED DATA
        C
0018            IF ((PRESEN(3).EQ..TRUE.).AND.(ISEV0.EQ..TRUE.).AND.
               1 (ISEV1.EQ..TRUE.)) GOTO 220
0019            CALL WCTR2
0020            IF (PRESEN(7).EQ..FALSE.) CALL WCTR6 (2)
0021            CALL WCTR13
0022            IF (PRESEN(17).EQ..FALSE.) CALL WCTR6 (9)
0023            IF (PRESEN(11).EQ..FALSE.) CALL WCTR6 (6)
0024            CALL WCTR3
0025            GO TO 230
0026    220     IF (PRESEN(11).EQ..FALSE.) CALL WCTR6 (6)
0027    230     IF (PRESEN(5).EQ..FALSE.) CALL WCTR5
0028            IF (PRESEN(6).EQ..FALSE.) CALL WCTR6 (1)
0029            IF (PRESEN(8).EQ..FALSE.) CALL WCTR6 (3)
0030            IF (PRESEN(9).EQ..FALSE.) CALL WCTR6 (4)
0031            IF (PRESEN(10).EQ..FALSE.) CALL WCTR6 (5)
0032            IF (PRESEN(15).EQ..FALSE.) CALL WCTR6 (8)
0033            IF (PRESEN(16).EQ..FALSE.) CALL WCTR16
0034            CALL WCTR1
0035            RETURN
0036            END
```

```
0001          SUBROUTINE WCTR1
      C       ----------------                    DRAW CARD
      C   WRITE CONTROL CARDS FOR ONE PART OF MAP
      C
0002          LOGICAL*1 PRESEN,ISEV0,ISEV1
0003          COMMON/UNITS/ICARD,JCARD,IUNIT,IWRIT,ITT,INTUN,JPNTR
0004          COMMON/ARRAYS/BUFIN(6),FRMT(20),FRMT2(20),TTL(20),CNAM(17),
             A PRESEN(18),IDAT(5),KEY,ISEV0,ISEV1,AAA(20,10),FILEN2(20)
0005          COMMON DISTX0,DISTX1,DISTY0,FMAPX ,FMAPX2,FMAPY ,FMAPY2,IBXOPT
0006          COMMON IGRI  ,IXPOS ,IYPOS ,IZPOS ,MANNR ,MXPGD ,NPTS
0007          COMMON NPTS2 ,NNX   ,NNX0  ,NNX2  ,NX    ,NX2   ,NNY   ,NNY0
0008          COMMON NNY2  ,NY    ,NY2   ,SM    ,SN    ,XMPMN ,XMPMX ,XORIG
0009          COMMON YCALC ,YMPMN ,YMPMX ,YORIG ,ZBLANK
0010          COMMON XMPMN2,XMPMX2,YMPMN2,YMPMX2,KLEVL ,NLVL  ,KNTUN
0011          INTEGER*4 NPTS,NPTS2,KNTUN,MXPGD
0012          COMMON/BIDON/Z(2500),SLVL(200),BLABEL(200)
0013          REAL*8 FMT1(6),FMT2(6),FMT5(5),FMT13(6)
      C
      C
0014          DATA FMT1/8H(A4,'1',,3H5X,,2*' ',8H,2F10.0,,4HI10)/
0015          DATA FMT2/8H(A4,'2',,3H5X,,3*' ',1H)/
0016          DATA FMT5/7H(A4,6X,,' ',5H,10X,,' ',1H)/
0017          DATA FMT13/6H(A4,6X,4*' ',1H)/
      C
      C
0018    2001  FORMAT(A4,6X,6I10)
0019    2002  FORMAT(20A4)
      C
      C
      C
0020          CALL FORM (FMAPX2,FMT1(3),0,1,10)
0021          CALL FORM (FMAPY2,FMT1(4),1,1,10)
0022          WRITE(JCARD,FMT1) CNAM(1),FMAPX2,FMAPY2,SN,SM,IBXOPT
0023          CALL FORM (ZBLANK,FMT2(3),0,1,10)
0024          CALL FORM (XORIG,FMT2(4),1,1,10)
0025          CALL FORM (YORIG,FMT2(5),1,1,10)
0026          WRITE(JCARD,FMT2) CNAM(1),ZBLANK,XORIG,YORIG
0027          RETURN
0028          ENTRY WCTR2
      C       -----------                         FMT CARD
0029          WRITE(JCARD,2001) CNAM(2)
0030          WRITE(JCARD,2002) FRMT2
0031          RETURN
0032          ENTRY WCTR3
      C       -----------                         GRID CARD
0033          WRITE(JCARD,2001) CNAM(3),NPTS2,NX2,NY2,IXPOS,IYPOS,IZPOS
0034          RETURN
0035          ENTRY WCTR5
      C       -----------                         LEVL CARD
0036          IF (KLEVL.EQ.2) GOTO 100
0037          DO 120 J=1,3
0038    120   CALL FORM (SLVL(J),FMT13(J+1),1,1,10)
0039          WRITE(JCARD,FMT13) CNAM(5),(SLVL(J),J=1,3)
0040          RETURN
0041    100   DO 110 J=1,NLVL
0042          CALL FORM (SLVL(J),FMT5(2),0,1,10)
```

```
0043          CALL FORM (BLABEL(J),FMT5(4),0,1,10)
0044    110   WRITE(JCARD,FMT5) CNAM(5),SLVL(J),BLABEL(J)
0045          RETURN
0046          ENTRY WCTR6 (I)
      C       -----------                    CARDS : MESH, MAOR, NDEC, PNCH, POST,
      C                                              PRNT, STOP, TICK, TREN
0047          WRITE(JCARD,2002) (AAA(K,I),K=1,20)
0048          RETURN
0049          ENTRY WCTR12
      C       ------------                   RDGD CARD
0050          WRITE(JCARD,2001) CNAM(12),NNX2,NNY2,MANNR
0051          RETURN
0052          ENTRY WCTR13
      C       ------------                   REGN CARD
0053          CALL FORM (XMPMN2,FMT13(2),1,1,10)
0054          CALL FORM (XMPMX2,FMT13(3),1,1,10)
0055          CALL FORM (YMPMN2,FMT13(4),1,1,10)
0056          CALL FORM (YMPMX2,FMT13(5),1,1,10)
0057          WRITE(JCARD,FMT13) CNAM(13),XMPMN2,XMPMX2,YMPMN2,YMPMX2
0058          RETURN
0059          ENTRY WCTR16
      C       ------------                   TITL CARD
0060          WRITE(JCARD,2001) CNAM(16)
0061          WRITE(JCARD,2002) TTL
0062          RETURN
0063          END
```

REFERENCES CITED

CALCOMP, 1974, CONTOUR, a basic contouring program: CALCOMP Applications Software, December 1974.

Donzeau, M. M., North, L. D., and Gettings, M. E., 1981, Program PDP004: CONTUR: U.S. Geological Survey Saudi Arabian Mission Miscellaneous Document 23 (Interagency Report 357), 57 p.