Record 1977/17

A GUIDE TO THE USE AND OPERATION OF PROGRAM CONTOR

by

A.S. MURRAY

CONTENTS

SUMMARY

Program CONTOR produces plotted and/or contoured maps from randomly distributed discrete observations. The program is written in Fortran IV for a Cyber 76 computer and uses plotting subroutines written for CALCOMP plotters. This document describes the operation of the program with a guide to its use and maintenance.

1. INTRODUCTION

This Record describes edition 6.3 of program CONTOR dated 25-10-76. The development history of the program has been described by Murray (1974).

The function of CONTOR is to produce plotted and/or contoured maps from randomly distributed discrete observations. The density of the data does not have to be uniform over the area of the map. The program can contour any number of observations in the map area. Any type of data in the form of discrete observations can be contoured provided that each observation has the following items; observation number, latitude, longitude and value to be contoured. The data may be read from a card image file or from a binary record file in the standard BMR gravity data bank format.

The program plots and annotates a map base, and plots the position of each observation found in the map area during the search of the input data file. The input data are transformed, if necessary, to obtain the value to be contoured and the observations plotted are annotated as required. The input values are put onto a rectangular grid and these values are then transformed by successive iterations of a smoothing function so that the imaginary surface passing through the grid values approaches a surface of minimum curvature. The smoothed grid values are used to calculate the intersections of the required contour levels with the grid. A cubic spline function is used to produce smooth contour lines from the intersection points.

The following chapters describe the program in detail. Chapter 2 contains information required by the occasional user and chapter 3 describes the program structure and the most effective way to produce maps, given a certain data distribution. The Appendices list the diagnostics which the program generates, give examples of decks required for various mapping requirements, and list the standard map references.

2. USER'S GUIDE

This chapter describes the functions of CONTOR in general terms. A simple example card deck is listed and described. The control cards which may be used are described, and their format is defined.

2.1 General Description

The following paragraphs describe the basic operations of CONTOR and give a summary of the technical file arrangement.

2.1.1. Plotting and annotating the map base.

The projection, name, number, co-ordinate limits, scale and graticule spacing of the required map are read from the input card deck. Using the data supplied, the program calls subroutines to define the parameters of the map, and the parameters needed to convert latitude and longitude into plotter increment units.

Five map projections are available; Standard Mercator, Transverse Mercator, simple Conic, Lambert Conformal Conic, and Plane (Rectangular).

The maximum width in longitude of a map for the drum plotter is 83.8 cm (33 inches), and the maximum dimensions for the flatbed plotter are 147.3 cm by 114.3 cm (58 by 45 inches). The map borders and annotation should be allowed for in calculating the map size; add 5.1 cm (2 inches) to the width and 20.3 cm (8 inches) to the map base dimensions. The program will print a warning if the map is too wide for the drum plotter and terminate the map if the dimensions exceed the flatbed limits. The axes can be interchanged on the flatbed plotter but not on the drum plotter.

The map graticule spacing is defined separately for latitude and longitude. The graticule automatically coincides with the ICAO standard mapsheet boundaries irrespective of the actual limits of the map. Registration marks, four + symbols,

are plotted at the limits of the graticule or at co-ordinates specified by the user.

If a standard 1:250 000 or 1:1 million scale map is required the map can be specified by name or number, all the other parameters being automatically selected. A reference table of the standard map names and numbers which can be recognised by the program is contained in Appendix C.

2.1.2. Data input, selection and posting.

The type of data to be input, map title, border width, Bouguer density and symbols, size and frequency of labels are read from the input deck. Data are read from the data input unit, and each observation lying within the required area and having the required parameters is accepted for the map. The data may be read in binary or BCD format; the BCD format is card images with up to eight ten-column variables per card. The items for each observation are assumed to be station number, latitude, longitude, meter height, observed gravity, ground height and terrain correction respectively. The co-ordinates are assumed to be in decimal degrees unless otherwise specified. The value to be contoured may be any one of the items for each observation or the free air or Bouguer anomaly calculated from the items for each observation.

The position of each observation may be plotted using one of 13 available symbols, two of which, the dot and the arrow, cannot be annotated. The arrow is used to indicate an observation with incomplete data for which a contour value cannot be calculated. The observations lying within the map base may be annotated with the survey code, station number, anomaly and/or height, or contouring value. The annotation frequency, label size, and survey number codes can be specified.

For map areas which cover both land and sea a composite anomaly map can be calculated using Bouguer anomalies on land and free air anomalies at sea.

The data on the input file may be searched in the following ways: (a) all the data may be searched, (b) particular segments (survey area blocks) may be searched, or (c) individual survey numbers can be selectively searched or selectively omitted from the search.

2.1.3. Gridding the input data.

Information on the grid spacing, contour cutoff distance, number of iterations for the smoothing operation, and amount of printer listing required is read from the input card deck. The data are searched to define the starting values for the grid points; flags and coefficients are defined when an observation lies close to a grid point. All grid points which do not have observations within their grid square are defined by spreading the values outward from the originally defined grid points.

The grid is defined by a latitude grid spacing in minutes and the ratio of latitude to longitude grid spacing. Each grid spacing is automatically adjusted so that the map base is covered by an integral number of grid spacings in each direction. The maximum number of grid points allowed in the latitude direction is 1000 and in the longitude direction is 500. The maximum number of grid points allowed on each map is 100 000. If these limits are exceeded the grid spacing will be repeatedly doubled by the program until they are met.

The mean and range of the input data values are calculated, and if the range is less than 0.001 the map is terminated. The data are searched for values lying within half a grid spacing of each grid point; these values are averaged and this average value is assigned to the grid point. The grid point is flagged with the average position of the averaged observations. If any value near a grid point differs from the average value by more than 0.1 of the range on the map the value may be in error, and the value, the average, and the position of the grid point in minutes from the top left corner of the map are printed.

Values are assigned to grid points without flags by spreading values from points with flags. When an unflagged point is equidistant from several flagged points it assumes the average value of all those points. After the spreading has moved out past the number of spacings specified as the contour cutoff distance, all the grid points are flagged with a default flag.

2.1.4. Forming the surface of minimum curvature.

The grid values are modified iteratively to make them approach a surface of minimum curvature as described by Briggs (1974). After the final iteration, grid points outside the contour cutoff distance (default flagged points) are assigned a large default value which is ignored in the contouring operation.

The grid point values are then redefined to approach the surface of minimum curvature by an iterative process which relates each grid value to its neighbouring grid values. There are three types of iteration referred to as linear, fast, and slow. The linear and fast iterations redefine only those grid points without an observation flag; the slow iterations redefine the flagged points also, using a weighting function based on the average observation value and average position of observations near the grid point.

The smoothing process may be stopped after any number of iterations in order to save the intermediate values for more iterations in a later job. When all required iterations have been completed, the grid values are prepared for contouring.

2.1.5. Contouring the gridded data.

Information on the contour levels, range of contours required, inhibition distance, plotter pens, and amount of printout, is read from the input card deck. The locations of the intersections of grid lines and contour levels are calculated and stored. These intersections are joined using cubic splines and the resultant contour lines are written on the plotter tape.

The contour levels which are required for the map are defined by two contour intervals and, if required, a minimum and maximum contour level. If a multi-pen mode is selected, the first and second contour levels are drawn with different pens and in different plot blocks.

Several contouring operations may be done using the same gridded data. The contour lines found for each pass are written as separate blocks on the plotter tape. Each plot block written on the plotter tape may be plotted using one of four pens selected automatically by the program, or manually at the time the tape is being plotted.

The contour lines will terminate at the edge of the map base or when a default grid value is sensed. The default values define the areas of the map beyond the extrapolation cutoff distance from an observation as described in 2.1.3.

The contour lines are supressed if they approach each other closer than the inhibition distance specified in the input deck.

2.1.6. File arrangement.

The standard file for data input is TAPE40. The plotter instructions are written on TAPE42, which should be assigned to a 7 track magnetic tape for the CALCOMP plotters currently being used by BMR. The results of the iterations are written on TAPE47 and TAPE48. During odd-numbered iterations the grid values are read from TAPE48 and new values are written on TAPE47; during even-numbered iterations the values are read from TAPE47 and written on TAPE48. The grid value files, TAPE47 and TAPE48, may be written on magnetic tape for use in a later job; these saved tapes are called restart tapes. The restart tapes are assigned to TAPE40 in place of the input data file when a contour job is restarted.

When the processing of a map terminates abnormally because of an error in a data control card, bad data on the input file, or a machine error, the program will terminate the current plotter block and search the input deck for the next map (indic-

ated by a MAPPLOT card); if an end of section (EOS) card is read during the search, the job will terminate.

The plotter instructions written on the plotter tape are divided into five types of plotter blocks as follows:

1. The definition of the plotter unit, scale and internal buffers.
2. The required origin and map registration marks.
3. The graticule, borders and annotation of the map base,
4. The symbols and annotation for the observations, and the title and legends for the map.
5. The contour lines.

For station plot maps, only plotter block types 1 to 4 are written. Each block may be plotted using different pens by manual pen selection. All blocks except type 5 blocks are plotted with pen 1. The pen number may be specified for the contour line blocks of type 5. For any map, several blocks of type 4 and type 5 may be written, to avoid problems with very large plotter blocks or to separate the plotter instructions into segments for convenience. Plotter blocks of type 1 and type 2 must be plotted first, but types 3, 4 or 5 may be omitted if the map base, observation plots or contours respectively are not required.

2.2 Sample Deck

This example shows a typical card deck required to produce a contoured Bouguer anomaly map from a standard format gravity data bank tape. The card deck is listed and briefly described here; a complete description of the parameters on these cards will be found in 2.3 below. Each line of the listing represents a computer card with the information starting in the first column of each card.

```
.CYJ, PROJECT=DMRXRC. (6,7,8,9 multiple punch in column 1)
AMGRVPT, T100, ML1, NL1.
COMMENT.   A. MURRAY    EXT.223
MOUNT, SN=DMR3922, VSN=PMC510.
```

```
ATTACH, CONTOR, CONTOR6, ID=DMRXRA, SN=DMR3922.
STAGE, TAPE40, PE, VSN =1216.
STAGE, TAPE42, HI, POST.
LABEL, TAPE42, W, L = PLOTMAP.
CONTOR.
EXIT, U.
RETURN, CONTOR.
ATTACH, TRS, TRS.
LIBRARY, *TRS.
LOAN, O/PLOTMAP.
EXIT,S.
RETURN, CONTOR.
end of section (7,8,9 multiple punch in column 1)
*MAPPLOT
DOBBYN
*BAPLOT
*GRID 1.0
*CONTOUR 3 5.00 20.0
end of information (6,7,8,9 multiple punch in column 1)
```

The program is stored on the BMR disc pack DMR3922 under the name CONTOR6 and is assigned to the file named CONTOR for this job. The input data are read from a nine-track magnetic tape assigned to the file named TAPE40. The plotter instructions are written on the file named TAPE42, which is copied to a seven track magnetic tape, with the label PLOTMAP, when the job is completed.

The only parameter defining the required map is the name, DOBBYN, so the program will automatically assume a standard mapsheet is required. In this case the map is at 1:250 000 scale, Transverse Mercator projection on the Australian National Spheroid with graticule lines every 15 minutes. The BAPLOT card indicates a Bouguer anomaly map and the density is assumed to be 2.67 t/m^3. All the data on file TAPE40 will be searched, and all stations will be allocated a symbol and survey code and will be plotted and labelled.

The grid spacing is 1 minute in latitude and longitude and the contours are drawn in 3 colours at intervals of 5.0 and 20.0 milligals. The negative 5.0 levels, the positive 5.0 levels, and all the levels which are multiples of 20.0,will each be drawn with a different pen.

2.3 Job Control Cards

All the cards placed before the first end of section (EOS) card in the deck are used by the Cyber 76 SCOPE monitor to control the execution of the job. This section of the deck, which varies little from the job to job, is termed the 'Job Control Section'. The job control cards which may be used in a normal job are listed below and each card is described in the following paragraphs. Actual examples are given for most of the cards but each user will need to make changes to project code, job name, time limit, tape names and numbers, etc. as necessary. The optional parameters and default values are given where appropriate. Full details of the job control cards are given in the Cyber 76 SCOPE Reference Manual.

Sample decks and brief descriptions of frequently used types of job are listed in Appendix B.

The following cards may be used in the job control section and should be used in the order listed. The information is usually punched starting in column 1 and without any enclosed blank columns, however, blanks are ignored on these cards. The numbers on the left indicate the paragraphs describing each card which follow this list.

1. .CYJ, PROJECT = DMRXRC.
2. AMGRVPT, T100, ML1, NL3.
3. COMMENT. Users name and extension.
4. MOUNT, SN=DMR3922, VSN=PMC510.
5. ATTACH, CONTOR, CONTOR6, ID=DMRXRA, SN=DMR3922.
6. STAGE, TAPE40, PE, VSN = 1216.
7. LABEL, TAPE40, R,L=INPUT
8. STAGE, TAPE42, HI, POST.
9. LABEL, TAPE42, W,L=PLOTMAP.

```
10.  STAGE, TAPE47, PE, POST.
11.  LABEL, TAPE47, W,L = GRID1.
12.  STAGE, TAPE48, PE, POST.
13.  LABEL, TAPE48, W,L = GRID2.
14.  CONTOR.
15.  EXIT, U.
16.  RETURN, CONTOR.
17.  ATTACH, TRS, TRS.
18.  LIBRARY, *, TRS.
19.  LOAN, O/PLOTMAP.
20.  EXIT,S.
21.  RETURN, CONTOR.
```

1. .CYJ, PROJECT = DMRXRC requests that the job with project code DMRXRC is placed in the Cyber 76 execution queue. The project code is required to indicate that the job originates from the BMR computer terminal and that the listing is to be printed at BMR. The RC in the code indicates a mapping job belonging to the Regional Gravity Group.

2. AMGRVPT, T100, ML1, NL3 defines the job name and sets a time limit of 100 octal (64) seconds of central processor time for the job. The number of seven track and nine track magnetic tapes required by the job are indicated by the ML and NL parameters. In this case one 7 track and three 9 track tapes are requested. If the numbers following ML and NL are less than the number of 7 and 9 track tapes used, the job will terminate.

3. COMMENT is a control word which copies all characters punched on this card to the output listing. The information on this card is useful in identifying the user if any problems arise during the input or output of the job.

4. MOUNT, SN=DMR3922, VSN=PMC510 requests access to the BMR disc pack which is permanently mounted in the Cyber 76 set of peripheral devices. The set name is DMR3922 and the serial number is PMC510. This card is always needed when reading the program from this disc pack.

5. ATTACH, CONTOR, CONTOR6, ID=DMRXRA, SN=DMR3922. assigns the program (permanent file name CONTOR6) on the disc pack to

the file named CONTOR. This card is necessary for every job which reads the program from the BMR disc pack.

6. STAGE, TAPE40, PE, VSN=1216. requests that the 9 track tape with serial number 1216 and a recording density of 1600 bits per inch (PE) is to be assigned to unit 40 as the data input for this job. STAGE indicates that the magnetic tape is not read on-line but is copied to a disc file named TAPE40 when the first reference is made to unit 40 in the program.

7. LABEL, TAPE40, R, L=INPUT. defines the label of the tape specified by card 6. This card is only required if the tape label is to be checked; if the label on the tape is not the same as the label specified on the card, in this case 'INPUT', the job will terminate.

8. STAGE, TAPE42, HI, POST. requests that the plotter instructions written on file TAPE42 be transferred to a 7 track magnetic tape with a recording density of 556 bits per inch (HI) when the job is completed. This card is necessary when plotting is required. The tape generated by the job will be assigned a serial number by the system and this number will be printed on the job listing. When the tape is received at BMR, on loan from CSIRO, it can be used to generate plotted maps on the CALCOMP drum or flat bed plotters.

9. LABEL, TAPE42, W,L = PLOTMAP. specifies the label to be written on the 7 track magnetic tape which will contain the plotter instructions.

10. STAGE, TAPE47, PE, POST. requests that the contents of file TAPE47 are written on a 9 track tape when the job is completed. The serial number of the assigned tape will be printed on the listing. TAPE47 contains the grid values produced by an odd number of iterations.

11. LABEL, TAPE47, W,L=GRID1. specifies the label 'GRID1' to be written on the tape requested by card 10. A label should be specified for all tapes written by the job so that the tapes can be easily identified.

12. STAGE, TAPE48,PE,POST. requests that the contents of file TAPE48 are written on a 9 track tape when the job is completed. TAPE48 contains the grid values defined by the

gridding of the input data or the values produced by an even number of iterations. The iteration process transfers data between TAPE47 and TAPE48, the input and output files transpose after each iteration. Usually only one of the files TAPE47 or TAPE48 is saved depending on the number of iterations specified, odd or even respectively.

13. LABEL, TAPE48,W,L=GRID2. specifies the label, 'GRID2', to be written on the tape requested by Card 12.

14. CONTOR. causes the file containing the binary program to be loaded into memory and executed. This file name is defined by card 5.

15 & 16. EXIT,U. and RETURN, CONTOR. ensure that the file CONTOR, which was attached from the disc pack, is released iffthe job terminates during the execution of the program.

17.18&19. ATTACH, TRS, TRS., LIBRARY, *, TRS. and LOAN, O/PLOTMAP. request that the tape labelled PLOTMAP created by this job is sent to BMR on loan from CSIRO. The tape reservation system TRS is a library of programs for manipulating the magnetic tape directories. These 3 cards must be used together and in the specified order to obtain a tape on loan automatically. The importance of unique labels is obvious as the label is the only parameter to identify the plotter output tape at the time the job is executed. The LOAN card (19) will request all magnetic tapes with the job's charge-code and the label PLOTMAP co be sent to BMR.

20 & 21. EXIT,S. and RETURN, CONTOR. ensure that the file CONTOR is released if the job terminates during the loading of the program.

2.4 Data Control Cards

The cards which are placed after the first end of section card in the job deck are read by program CONTOR and are termed the 'Data Control Section'. The data control cards which are recognized by the program are listed below and the format and parameters of each card are described in the following paragraphs. Figure 1 illustrates the card formats. Each card is referred to by its control word (e.g. MAPPLOT) or by its function

FIGURE 1

BMR DATA CODING FORM

NAME			DATE
CONTOR	Ed: 6.4	DATA CONTROL CARD FORMATS	DECEMBER 1976 SHEET 1 OF 1

Card	Entry (columns 1–80)	Annotations
1	*MAPPLOT CONIC 1600 2400 11730 12900,	projection; corner marks, map name or number
2	CANNING BASIN WESTERN AUSTRALIA E+F/51-2 1540 2420 11710 12920	map name; map area; map number; map limits
3	AUSTRALIAN NATIONAL SPHEROID 6378160. 298.25 1800 3600 2500000. 60. 90.	spheroid name; earth radius; flattening; standard parallels; scale; graticule space
4	*BAPLOT, 60, 3 METRES, MINUTES, MARINE,	data type; file item; height units; coordinates; environment
5	*DATUM FAVO,	plotted data
6	*TITLE BOUGUER ANOMALIES 1965 ISOGAL DATUM	main title; sub title
7	*BORDER 15.0 2.67 2 99 500 1	border; density; label size; default symbol; plot flush size; marine sea air
8	*SURVEY 6006 A 1 5 6106 B 10 10 6491 I 5 11 6205 C 3 6804 D 4 6805 E 5 2 6914 F 6 5	survey code symbol label
9	*SEGMENTS 5 10 1, 6206.001,	all; station anomaly height frequencies; survey area no.
10	*GRID 4.00 2.5 5 10 10 6 1, 2	latitude grid; lat/long grid ratio; cutoff; fast iterations; slow iterations; linear; prem. term.; print
11	*CONTOUR 3 5.0 20.0 60.0 80.0 0.1 4, 2	multi pen; interval-1; interval-2; min level; max level; inhibition; pen; print
12	*RESTART ITERATE,	procedure
13	EOS	

CALCOMP SYMBOLS

≤-1	0	1	2	3	4	5	6	7	8	9	10	11	≥12
blank	default	□	+	△	⧗	X	◇	Y	O	Z	⌧	✳	■

0452 21A

(e.g. mapname). The optional parameters and default values are given where appropriate. The cards should be used in the order shown (except the RESTART card) for each map required. Several maps may be plotted in a job subject to the job time limit specified, and provided that the grid values for all the maps except the last one are not required to be saved. The grid value tapes are rewound and rewritten each time a new map is started.

The information on each card must be punched in the columns indicated in the description of each card. The number on the left in the following list refers to the paragraph which describes each card.

1. *MAPPLOT
2. Mapname
3. Spheroid

4a. *BAPLOT
4b. *FAPLOT
4c. *HTPLOT
4d. *XYPLOT

5. *DATUM
6. *TITLE
7. *BORDER
8. *SURVEY
9. *SEGMENTS
10. *GRID
11. *CONTOUR
12. *RESTART
13. end of section (7,8,9 multiple punch in column 1)

In the following card descriptions several abbreviations have been used: N for necessary card, O for optional card, D = value for default parameter and Col for card column numbers. All numeric parameters should be right justified and alphabetic parameters left justified.

1. MAPPLOT card. (N for jobs with plotter output)

The MAPPLOT card initiates the map drawing procedure and defines the projection and registration mark positions, map name or map number. The projection is specified by one of the names MERCATOR, TRANSVERSE, CONIC, LAMBERT or PLANE which represent Standard Mercator, Transverse Mercator, Simple Conic, Lambert Conformal Conic and Plane (Rectangular) respectively. If the projection name is omitted the program assumes a Transverse Mercator projection. Four + signs (registration marks) define a "rectangle" in latitude and longitude which is used to register the map position when the graticule is not plotted, and may be inside or outside the map base. If the registration mark co-ordinates are not specified the marks will be drawn at the corners of the map base. The map name or map number may be specified on the MAPPLOT card or on the map name card as described in the next paragraph (2).

Col 1-8 : *MAPPLOT

Col 11-20 : projection name which may be MERCATOR, TRANSVERSE, CONIC, LAMBERT or PLANE. All names must start in column 11. (D = TRANSVERSE)

Col 21-50 : registration mark limits, map name or map number. The format for registration mark limits is:

Col 21-23:	Northern	latitude	limit,	degree component
24-25:	"	"	"	minute "
26-28:	Southern	"	"	degree "
29-30:	"	"	"	minute "
31-33:	Western	longitude	"	degree "
34-35:	"	"	"	minute "
36-38:	Eastern	"	"	degree "
39-40:	"	"	"	minute "

Positive latitudes are assumed to be southern hemisphere, positive longitudes are east of Greenwich. The program will automatically distinguish between registration mark limits, map names or map numbers. The map name or number may be punched

anywhere in the columns 21 to 50 but must be one of the standard map names or numbers recognized by the program. Microfiche lists of recognized names and numbers are available (D = blank).

2. Map name card. (N for non-standard map names or numbers).

The mapname card defines the map name, area, number and co-ordinate limits. If the map limits are omitted the map name or number will be used to define the map. Given a map name or number the program will search a reference list for this name or number. If the map is not found in the list a message will be printed, and the job will search for the next MAPPLOT card. If the map is found the program defines the standard name, number, area and limits automatically.

Col 1-30: Map name. (e.g. DOBBYN)
Col 31-50: Map area. (e.g. QUEENSLAND)
Col 51-60: Map number (e.g. E54-14)
Col 61-80: Map limits. The limits have the following format:

Col 61-63:	Northern	latitude	limit,	degree component
64-65:	"	"	"	minute "
66-68:	Southern	"	"	degree "
69-70:	"	"	"	minute "
71-73:	Western	longitude	"	degree "
74-75:	"	"	"	minute "
76-78:	Eastern	"	"	degree "
79-80:	"	"	"	minute "

If the map limits are omitted the map name field columns 1 to 30 must contain a standard map name or number.

This card is omitted when there is no MAPPLOT card or when a standard name or number is specified on the MAPPLOT card, otherwise it is necessary.

3. Spheroid card. (A)

The spheroid card defines the spheroid name and parameters, the scale and the graticule spacing. This card is not

required for standard maps specified by a name or number and is optional when the scale is 1:250 000. When this card is absent the program defines the spheroid parameters for the Australian National Spheroid (ANS) and uses a graticule spacing of 15 minutes for every multiple of 250 000 in scale Factor (e.g. 1 degree at 1:1 million scale). For Standard Mercator projection the latitude of the defined scale is the central latitude on the map. For the conic projections the default standard parallels are 18 and 36 degrees South latitude. For the Plane projection no spheroid is used and the parameters supplied are specified as plotter units per degree of latitude and degree of longitude. The graticule spacing should not be less than 1/50 of the map dimensions in either direction as a maximum of 50 graticule lines can be drawn in each direction. Any excess lines will not be plotted.

Col 1-30: Spheroid name. (D = AUSTRALIAN NATIONAL SPHEROID)
Col 31-40: Radius of Earth (D = 6378160.0) or latitude scale (PLANE : D = 2.0)
Col 41-50: Flattening factor (D = 298.25) or latitude scale units (PLANE: D = inches)
Col 51-60: Origin and zone width (TRANSVERSE: D = -183.0, 6.0)
Latitude of standard scale (MERCATOR: D = Equator)
Standard parallels (CONIC, LAMBERT: D = 18.0, 36.0)
Longitude scale (PLANE: D = 2.0)
Col 61-70: Scale (D = 250000.0) or longitude scale units (PLANE: D = inches)
Col 71-80: Graticule spacing in minutes with the following format.
Col 71-75: latitude graticule spacing (D = 15.0/250000)
Col 76-80: longitude " " "

The origin and zone width for Traverse Mercator projection and the two standard parallels for the conic projections are punched in columns 51-55 and 56-60 respectively. The standard latitude for Standard Mercator projection is punched in columns 51-55. The scale units for the plane projection are punched in columns 42-43 and 62-63 as one of the following

two letter codes: IN for inches, FT for feet, CM for centimetres, MM for millimetres. The scale for the plane projection is expressed as the number of plotter units per degree (D = 2.0 inches/degree).

4. Data plot card. (N for data input, searching and plotting)

The BAPLOT, FAPLOT, HTPLOT or XYPLOT card initiates the data input and posting procedure. This card specifies the data input file, the parameter to be contoured and the units and type of the input data.

4a. BAPLOT is used for Bouguer anomalies
4b. FAPLOT is used for free air anomalies
4c. HTPLOT is used for ground heights
4d. XYPLOT is used for card input data

If the data input file is not specified the file named TAPE40 will be used. If the data is to be read from the input card deck the unit number specified is 60. When cards 4a and 4b are present the contour parameter is automatically the Bouguer or free air anomaly respectively. When 4c or 4d are present the parameter to be contoured may be any one of the items pertaining to the observation except the observation number or, the parameter may be either type of anomaly. For calculating gravity anomalies the items (principal facts) for each observation are assumed to be observation number, latitude, longitude, meter height, observed gravity, ground height and terrain correction respectively. When the item number is not specified the ground height is used with a HTPLOT card or the value in the meter height position with an XYPLOT card. The units of the input heights and co-ordinates are assumed to be metres and degrees unless specified separately as FEET and MINUTES respectively. When MINUTES is specified for the co-ordinates, the input data are read as minutes for cards 4a, 4b and 4c and degrees and minutes for 4d. If the input data are marine the word MARINE is specified which causes all anomaly calculations to use a marine reduction formula.

Col 1-7: *BAPLOT for Bouguer anomalies
*FAPLOT for free air anomalies
*HTPLOT for heights or other input variables
*XYPLOT for data in card image format.

Col 11-12: Data input unit. Any number between 01 and 99 except 30, 42, 47, 48, 61 and 62 (D = 40).

Col 15 : A number indicating the variable to be used as the contour value. This number has the following meanings for standard gravity principal facts:
1 = latitude, 2 = longitude, 3 = meter height, 4 = observed gravity, 5 = ground height, 6 = terrain correction, 8,9 = gravity anomaly (8 indicates variable 3 (meter height) is used as the height for both meter and ground heights and number 9 indicates variables 3 and 5 are used as the meter and ground height respectively). On card image input the number indicates the columns containing the required variable as follows: 1 = col 11-20, 2=22-30, 3 = 31-40, 5 = 51-60, 6=61-70 and 7=71-80. (HTPLOT: D = 5, XYPLOT: D = 3, others D = blank).

Col 17-22: Elevation units; METRES or FEET (D = METRES)

Col 25-31: Co-ordinate units: DEGREES or MINUTES (D = DEGREES) For XYPLOT, MINUTES means degrees and minutes are punched on the data cards not decimal minutes.

Col 33-38: MARINE indicates the input data are marine readings (D = blank)

5. DATUM card. (O)

The DATUM card is used when a non-standard format of data annotation or contouring value is required. The standard units are metres and milligals for heights and gravity values. The standard gravity datum is the May 1965 Isogal datum which uses the 1930 ellipsoid for calculation of the normal gravity. The alternative gravity datums use the 1967 ellipsoid.

Col 1-6 : *DATUM
Col 9 : + for 6 digit numbers in addition to the survey code, otherwise = for standard 4 digit numbers
Col 10 : D for degrees (co-ordinates)
Col 11 : M for metres, F for feet (height units)
Col 12 : A for AHD heights, otherwise blank
Col 13 : M for milligals, U for micrometres/sec^2
Col 14 : O for May 65 Isogal, P for provisional IGSN71 datum and N for 1976 Isogal datum.

6. TITLE Card. (O)

The TITLE card provides two lines of annotation to be written under the map in a central position. If the title is not specified the program will select a title appropriate to the type of procedure 4a, b or c. The automatically defined titles are; Bouguer Anomalies, Free Air Anomalies, and Heights. No title is automatically defined for card 4d.

Col 1-6 : *TITLE
Col 11-50 : Main title. This information is plotted as a line of annotation directly underneath the map base.
Col 51-80 : Sub-title. This information is written as a second line of annotation below the main title but above the survey symbol and code table. Default titles for the various data plot cards are:
BAPLOT: BOUGUER ANOMALIES
FAPLOT: FREE AIR ANOMALIES
HTPLOT: HEIGHTS
XYPLOT: No title

7. BORDER Card. (O)

The BORDER card specifies the border width in minutes, Bouguer density in t/m^3, label size, default symbol, observation plot block size and marine data free air flag. The border width

defines the area surrounding the map base in which any observations found will be used in the calculation of the grid values. These observations will not be plotted but their values will be used as the edge control in the calculation of the contour lines. The Bouguer density is used only when the variable required for contouring is the Bouguer anomaly. If the density is specified as 0.0 then free air anomalies will be calculated. The label size defines the size of the characters in the annotation beside each station symbol. The dafault symbol number defines the symbol to be plotted at the position of all observations with survey numbers which have not been assigned a particular symbol on a SURVEY card.

The station plot block size defines the number of station (observation) symbols and annotations which are to be written in each block of plotter instructions on TAPE42. The marine free air flag causes all anomaly calculations for marine observations to be made using the free air correction only. Bouguer anomalies will be calculated, if requested, for the land stations.

Col 1-7: *BORDER

Col 11-15: Border width in minutes (D = 0.0 if BORDER card is present otherwise D = 15.0).

Col 16-20: Density for Bouguer reduction (D = 2.67)

Col 21-25: Labelling size for annotation in units of 0.07 inches letter height. (D = 1)

Col 26-30: Default symbol for surveys not specified on a SURVEY card. The symbol numbers which can be used and their meanings are as follows: negative integer = skip all surveys not specified on a SURVEY card; 0 or omitted use the standard symbols (numbers 1 to 9) in rotation as each new survey is encountered (excluding those surveys specified on a SURVEY card); 1 = square; 2 = +; 3 = triangle; 4 = bow; 5 = X; 6 = diamond; 7 = Y; 8 = octagon; 9 = Z; 10 = pointed square; 11 = large star; 12 and above = dot. (D = 0). (see Fig 1)

Col 31-35:Station plot block size. This parameter defines the amount of symbol and annotation information written on the plotter tape in each plot block. The program increments a counter by 1 for each dot plotted, 3 for each symbol plotted and 10 for each line of annotation. This parameter defines the maximum value which the counter can assume for each plot block. (D = 10000).

Col 36-40:Marine free air flag. If this parameter is not zero all marine observations will have a free air anomaly calculated instead of a Bouguer anomaly. (D = 0)

8. SURVEY card. (O)

The SURVEY card defines the code letters, symbols and individual labelling frequencies to be assigned to the specified surveys. The four digit survey prefix of each observatior or station number is replaced by a one or two letter code when the annotation is plotted on the map, thus reducing the amount of lettering on the map. The symbol number defines the symbol to be plotted at the position of all observations of the specified survey which are within the map base. The individual labelling frequency for a survey may be modified by the parameter specified on this card but it also depends on the labelling frequency specified on the SEGMENT card. A maximum of 100 surveys per map can be assigned specific codes and symbols on SURVEY cards. Any observation assigned the symbol blank, dot or arrow will not be annotated.

Col 1-7: *SURVEY

Col 11-20,21-30,.....71-80: Survey number, code, symbol number and label frequency in each 10 column field as shown below for columns 11 to 20; all the other fields are analogous.

Col 11-14: Survey prefix. The standard 4 digit survey number as used in the gravity data files or the first four digits of an 8 digit observation number.

Col 15-16: A two letter code which will replace the survey prefix in annotations. If only one letter is required this should be punched in column 16.

Col 17-18: The symbol number for the specified survey. If this number is negative this survey will be omitted from the data search. If the number is 0 or omitted no symbol or labels will be written for stations in this survey, however the values obtained from the stations lying in the search area will be used in the contouring proceedure. If the number is positive the symbol used will be as listed above in 7.

Col 19-20: The survey labelling frequency enables specific surveys to have labelling differing from the general labelling frequencies specified on the SEGMENT card. If this labelling frequency is negative no labels will be written for stations of this survey. If this frequency is 0 or omitted the frequencies on the SEGMENT card will be used. If this frequency is a positive integer the labelling depends on the frequencies on the SEGMENT card as follows:

(i) If the SEGMENT frequencies are all zero or omitted then the SURVEY frequency will be used for all labels for the survey.

(ii) If any of the three SEGMENT frequencies are positive then the frequency of each label will be the product of the SEGMENT frequency and the SURVEY frequency.

9. SEGMENT card. (0)

The SEGMENT card specifies the data block on the input file which is to be searched and the annotation frequency for the observations extracted from the block. If a SEGMENTS card (9a; note the final S) is used all the data blocks on the input file will be searched. The segment to be searched is specified by the survey area number for the standard tape library input format or by the first 8 characters of the block name for card image input. Any number of SEGMENT cards may be specified for each map but only one SEGMENTS cards is allowed.

If card image data are to be read from the card reader the data cards should be preceded by an area name and the corresponding SEGMENT card. The data cards are terminated by one end of section card if only one block of data is being searched or by two end of section cards if all the data is being searched (by the use of a SEGMENTS card).

The labelling frequency defines the number of stations plotted per annotation. The variables in the annotation can be identified by referring to the legend below the map.

Col 1-8: *SEGMENT

Col 9 : S indicates that all segments on the input file are to be searched.

Col 11-13:Station number labelling frequency.

Col 14-16:Anomaly labelling frequency.

Col 17-19:Height or contouring value labelling frequency.
If any of the frequencies are zero or omitted the corresponding label will not be written. If the specified frequency has the value n then every n'th station will have the corresponding label. The labelling frequency for specified surveys may be modified by the frequency on the SURVEY card as described above in 8.

Col 21-28:The input data block number or standard survey area number which defines the segment of the input file which is to be searched. For card image input each block of data is preceded by a header card. The first

8 characters of the header of the required block are specified here. This parameter is ignored when an S in punched in column 9 of the card.

10. GRID card. (N for gridding and smoothing data)

The GRID card defines the grid spacing, extrapolation cutoff distance, number of iterations in the smoothing procedure and the listing option. The grid spacing is specified by a latitude grid spacing on minutes and the ratio of the latitude to longitude grid spacings. The longitude grid is defined by dividing the latitude grid by the specified grid ratio. This ratio is specified only when a non-square grid is required, for example, when contouring anisotropic data such as marine or airborne linear traverses. If the grid spacing is too small to satisfy the limit of the maximum number of grid points in the grid as a whole (100 000), in latitude (1000) or in longitude (500), the grid spacing will be repeatedly doubled until all these limits are met. Both grid spacings are automatically adjusted by the program so that the required grid area is covered by an integral number of grid points. The actual grid spacings are printed on the output listing. The extrapolation cutoff distance is the number of grid spacings away from any observation to which the contour lines will extend. There are three types of iteration; linear, fast and slow. In linear and fast iterations only the grid points without observations lying within half a grid spacing in either direction are changed to produce a smoother surface. In slow iterations the grid points with observations within half a grid space are also changed.

Col 1-5: *GRID
Col 6-10: Latitude grid spacing in minutes (D = half the average station spacing)
Col 11-15: Latitude/longitude grid spacing ratio. (D = 1.0)
Col 16-20: Contour extrapolation cutoff distance expressed in grid spacings. (D = 5).

Col 21-25:Number of fast iterations in the smoothing process (D = 5)

Col 26-30:Number of slow iterations in the smoothing process (D = 5)

Col 31-35:Number of linear iterations in the smoothing process (D = 2)

Col 38 :Restart option. If this number is 1 the smoothing process is prematurely terminated.

Col 40 :Listing option. If this number is 1 a summary of the gridding and iteration procedures is printed. If the number is 2 or more a full list of grid values, flags and coefficients is printed. (D = 0; only a minimum of information is printed).

11. CONTOUR Card. (N for contouring procedure)

The CONTOUR card defines the multi-pen option, contour intervals, range of contour levels required, contour line inhibition distance, pen number and listing option. Each contouring pass of the gridded data may proceed in one of three modes which are the following:

(i) Single pen mode: all contour lines are drawn with the same pen; one contour plot block is written.

(ii) Two pen mode: The levels of each specified interval are drawn with a different pen; two contour plot blocks are written.

(iii) Three pen mode: the levels of the second specified interval, the negative contours of the first interval, and the positive contours of the first interval, are each drawn with a different pen; three contour plot blocks are written.

The first contour level should be smaller than the second level. The minimum and maximum levels define the limits between which contour lines are drawn. The inhibition distance, expressed in minutes, is the minimum distance allowed between two consecutive levels. The pen number defines which pen is to be used for the contours in the plot block, it is ignored with

the multi-pen modes. Each separate CONTOUR card processed by the job will begin a new contour plot block. The separation of contours into different plot blocks allows manual pen changes and selective drawing of particular levels.

Col 1-8: *CONTOUR

Col 10 : Multiple pen option. This parameter may have the following values:

1. All contours will be drawn with pen 1.
2. The second interval levels will be drawn with pen 1 and the first interval levels with pen 2.
3. The second interval levels will be drawn with pen 1, the negative contours of the first interval with pen 2 and the positive contours of the first interval with pen 3.

0 or omitted: All contours will be drawn with the pen specified in column 38. The second interval contours will be drawn twice to produce a thicker line.

Col 11-15: First contour interval. All levels which are a multiple of this interval but not a multiple of the second interval will be drawn. If this value is negative, zero or omitted an interval is defined by the program so that about 20 multiples of the interval are drawn on the map.

Col 16-20: Second contour interval. All levels which are multiples of this interval are drawn. If this value is omitted or zero then only the first level multiples are drawn. If this value is negative the multiples of this interval will not be drawn. (D = 0.0).

Col 21-25: Minimum level to be drawn (D = lowest value).

Col 26-30: Maximum " " " " (D = highest value).

The limiting levels define the values between which the data is contoured. Any values lying outside these limits are not represented by contour lines.

Col 31-35: Inhibition distance expressed in minutes. If the contour lines approach each other closer than this distance the drawing of the lines is suppressed.

Col 38 : Pen number. This number may be an integer between 1 and 4. The plotting equipment will automatically change the pen from the standard pen 1 to the specified pen to draw the contour lines produced by this CONTOUR card. (D = 1). If the multipen option was specified this pen number is ignored.

Col 40: : Listing option. If this parameter is specified a list of contour block size, edge values and contour cuts is produced. This listing is automatically produced if the listing option was specified on a preceding GRID card.

12. RESTART Card. (N for restart jobs)

The RESTART card is used when the gridded data are stored on magnetic tape and the job is to be restarted at the iteration or contouring stage of the processing. If the job is restarted to make further iterations in the smoothing of the grid surface and no contouring is to be done then the RESTART card is the first card in the data control section otherwise this card must replace card number 4. RESTART is followed by a GRID or CONTOUR card depending on the parameter ITERATE or CONTOUR specified on the RESTART card. The input grid values are read from file TAPE40 during the restarted job and the smoothed grid values are alternately written on files TAPE 47 and TAPE 48 as for the normal gridding and iterating procedure. A restarted job will terminate if further iterations are requested for a tape of grid values which have been prepared for contouring in a previous job or if contours are requested from a tape of grid values which have not been prepared for contouring.

Col 1-8: *RESTART

Col 11-17: ITERATE or CONTOUR. ITERATE is specified when restarting a job after the data has been gridded or if the smoothing

process was prematurely terminated. In this case the RESTART card is followed by a GRID card (N.B. the grid parameters cannot be redefined by the restarted job). CONTOUR is specified when restarting a job after the smoothed grid has been prepared for contouring. In this case the RESTART card is followed by a CONTOUR card. (n.b. the map limits specified for the restarted job must be the same as for the job which created the gridded values).

13. End of Section Card.

An end of section card in the data control deck will terminate the job except in the following cases:

(i) An end of section may be used to terminate a group of SEGMENT cards.

(ii) When reading data from the input card deck at end of section is required to terminate each data block. If a SEGMENTS card is used, two consecutive end of section cards are required to terminate the data blocks.

(iii) An end of section may replace card number 10 when station plots only are required.

Col 1: Multiple punching 7, 8, 9.

3. OPERATING INFORMATION

This Section describes the structure of CONTOR, the configuration of logical units, the factors influencing the selection of parameters, and the means of increasing the efficiency of particular types of job.

3.1 Program Structure

CONTOR consists of four overlays, two of which each have two segments. This division into overlays and segments limits the amount of central processor core required during execution of the program package. Each overlay and segment is responsible for one of the sequential steps in the production of a contour map.

Overlay 0:Program CONTOR6 controls the execution of the job by calling the other overlays into core as required. This overlay defines the common data blocks and subroutines used by the other overlays.

Overlay 1:Program MAPDATA controls the execution of programs MAPPLOT and BAHMAP.

Segment 1:Program MAPPLOT establishes the map parameters and draws and labels the map base.

Segment 2:Program BAHMAP reads the data file, transforms the data as required, stores, plots and annotates the data values.

Overlay 2:Program GRID establishes the regular grid and controls the execution of GRIDDY and ITERATE.

Segment 1:Program GRIDDY grids the irregularly spaced input data onto the regular grid.

Segment 2:Program ITERATE smooths the grid values to approach a surface of minimum curvature.

Overlay 3:Program CONTOUR controls the drawing of contour lines from the gridded surface.

Most of the programs named above have several subroutines as outlined in the following list. This list contains the 52 sections which comprise the binary file of CONTOR.

1. CONTOR, 0,0	14. NUMBER	27. EDITEXT	40. CALCO
2. CONTOR6	15. SCALE	28. MAPSEEK	41. CONTOR,2,1
3. BEGINIT	16. SYMBOL	29. CONTOR,1,2	42. GRIDDY
4. ORICHEK	17. FACTOR	30. BAHMAP	43. CONTOR,2,2
5. PARAM	18. NEWPEN	31. AREFIND	44. ITERATE
6. PROJECT	19. WHERE	32. PARIX	45. LINEATE
7. XYMRCTR	20. PLOTS	33. FLOTE	46. VSOP
8. XYTRANS	21. PLOT	34. DASHIFT	47. CONTOR,3,0
9. XYCONIC	22. BUFF	35. RECORD	48. CONTOUG
10. XYLMBRT	23. CONTOR,1,0	36. IBYTE	49. LINKUT
11. XYPLANE	24. MAPDATA	37. CONTOR,2,0	50. DRAWLIN
12. AXIS	25. CONTOR,1,1	38. GRID	51. ISQRT
13. LINE	26. MAPPLOT	39. BLOCK	52. LGL

The following paragraphs briefly describe the purpose of each section listed.

1. CONTOR, 0, 0Oidentifies overlay 0. Each overlay or segment includes all sections following the identifier up to the next identifier or the end of the binary file.

2. Program CONTOR6 calls the other overlays.

3. Subroutine BEGINIT prints timing information on the job output.

4. Subroutine ORICHEK plots a symbol several times during a job which enables detection of plotter origin shifts.

5. Subroutine PARAM annotates the plotted map with information on the contouring parameters.

6. Subroutine PROJECT controls the operation of following five co-ordinate conversion subroutines which transform latitude and longitude into plotter units.

7. Subroutine XYMRCTR - Standard Mercator projection.

8. Subroutine XYTRANS - Transverse Mercator.

9. Subroutine XYCONIC - Simple Conic.

10. Subroutine XYLMBRT - Lambert Conformal Conic.

11. Subroutine XYPLANE - Plane projection.

12-22. Subroutines AXIS to BUFF are proprietory CALCOMP subroutines which convert plotting commands into machine instructions for the CALCOMP drum and flat-bed plotters used by BMR.

23. CONTOR, 1, 0 identifies overlay 1.
24. Program MAPDATA controls MAPPLOT and BAHMAP
25. CONTOR, 1,1 identifies segment 1 of overlay 1.
26. Program MAPPLOT defines the map.
27. Subroutine EDITEXT edits the map name and number
28. Subroutine MAPSEEK searches a reference table of standard map names and numbers.
29. CONTOR, 1,2 identifies segment 2 of overlay 1.
30. Program BAHMAP reads, searches, stores and plots the data.
31-36. Subroutines AREFIND to IBYTE manipulate the data read from standard gravity data bank files.
37. CONTOR, 2, 0 identifies overlay 2.
38. Program GRID establishes the grid.
39. Subroutine BLOCK defines the grid block variables.
40. Subroutine CALCO calculates the coefficients for the 625 grid cells surrounding each grid point.
41. CONTOR, 2, 1 identifies segment 1 of overlay 2.
42. Program GRIDDY grids the data.
43. CONTOR, 2,2 identifies segment 2 of overlay 2.
44. Program ITERATE controls the smoothing process.
45. Subroutine LINEATE smooths the grid values linearly.
46. Subroutine VSOP smooths the grid values using bicubic splines.
47. CONTOR, 3,0 identifies overlay 3.
48. Program CONTOUR controls the drawing of contour lines.
49. Subroutine LINKUT calculates the contour line intersections with the grid.
50. Subroutine DRAWLIN joins the contour cuts using cubic splines to produce the contours.
51. Subroutine ISQRT calculates integer roots quickly.
52. Subroutine LGL handles single bit variables.

All these programs and subroutines are written in Fortran IV except BUFF which is in compass assembly language. The overlay and segment identifiers are single cards which precede their appropriate program in the Fortran card deck.

3.2 File Use

The files used by the program have the following names in the Cyber 76 file information tables: TAPE 1, TAPE 2, TAPE 30, TAPE 40, TAPE 42, TAPE 47, TAPE 48, TAPE 60 (= INPUT) and TAPE 61 (= OUTPUT).

The standard file for data input in TAPE40 which is referred to as logical unit number 40 in the Fortran program. The plotter instructions are written on TAPE42 which should be assigned to a 7 track magnetic tape, for the CAPCOMP plotters currently used by BMR. The selected data from the search of the input data file is written on TAPE 30 as three values per observation; latitude, longitude and value for contouring. The initial grid values before smoothing are written on TAPE48. The results of the smoothing iterations are written alternately on TAPE47 and TAPE48. During odd numbered iterations the grid values are read from TAPE48 and the new values are written on TAPE47, and during even iterations the values are read from TAPE47 and written on TAPE48. At the beginning of the contouring phase the final grid values are read from TAPE47 or TAPE48 as appropriate, separated into blocks, and written on TAPE30.

The grid value files, TAPE47 and TAPE48, may be saved on magnetic tape as restart tapes for use in a later job. The restart tape is assigned to TAPE40 in the restart job in place of the data input file.

When a new map is commenced by the program all the information on files TAPE30, TAPE47 and TAPE48 is destroyed.

The program uses TAPE60, also called INPUT, as the data control card file and TAPE61, also called output, as the file for writing the job listing. The SCOPE monitor automatically assigns the section of input card deck following the first end of section card to file INPUT and sends file OUTPUT to the line printer.

The file named TAPE1 is available as an alternative data input file. File TAPE2 is used by the CALCOMP plotting subroutines. If a data input file different from those listed above is required the file name will have to be added to the CONTOR6 program card and the program recompiled and reloaded.

3.3 Parameter Selection Criteria

This section describes the technical significance of the parameters on each card, indicates the range of values allowed, and gives the criteria for choosing the parameters for optimising the job. The card numbers referred to here were defined in the Users Guide.

The job control section parameters are as follows:

1. CYJ, PROJECT = project code. The project code for BMR users is a six letter code DMRX -- where -- indicates the users Section. (e.g. DMRXRC for Regional Gravity Group).

2. Job name, time limit, peripheral devices. The job name is between one and seven characters in length and may contain any alphanumeric characters provided that the first is a letter. In the Regional Gravity Group this name is usually uuGRVcc where uu are the users initials and cc is a two letter identifier (e.g. AMGRVPT). The time limit is expressed as an octal number of seconds prefixed by T. (eg T100 = 64 seconds). This time limit is the maximum amount of CPU time to be used on the Cyber 76 computer. For each map in a job the time required can be roughly calculated as follows: Allow 1 second for every 10 000 stations on the input data file, 1 second for every 2000 stations plotted and 1 second for every 250 stations fully annotated. The gridding time in seconds is the number of observations by the number of grid points/500 000. The contouring time in seconds is the number of contour lines on the map by the number of grid points/10 000. The peripheral devices may be seven track tapes (MLn), nine track tapes (NLn) or demountable disc packs (YL1) where n is the number of devices used, only 1 demountable disc pack may be used.

3. COMMENT. Users name and extension. After the COMMENT. the information may be punched in free format on the remainder of the card.

4. MOUNT, SN = set name, VSN = disc serial number. The set name and serial number define the disc pack to be used by the job. One of these cards is required for every pack used. If the pack is demountable the YL1 parameter must be specified on card 2.

In a normal job where CONTOR6 is being attached from the BMR permanently mounted pack on which it usually resides the set name is DMR3922 and the serial number is PMC510. If the program or some data is being attached from the Regional Gravity demountable pack the set name is DMR8119 and the serial number is C030.

5. ATTACH, CONTOR, CONTOR6, ID=DMRXRA, SN = set name. The file name CONTOR is arbitrary but is standardised for convenience. This file name must be identical to that on card 14. The permanent file name CONTOR6 and the account code DMRXRA are standard for this edition of the program. The set name is generally DMR3922 but may be DMR8119. A mount card (4) must be present for the set name used.

6 & 7. STAGE, TAPE40, PE, VSN= serial number. LABEL, TAPE40, R, L = INPUT. The LABEL card is optional and should not be used unless a positive tape label check is required. TAPE40 is the default data input file name and must be changed if the file parameter on data control card 4 is not 40 or blank. All data tapes should be 9 track tapes (PE) for efficiency. The serial number is the CSIRO tape library number. If the data input is read from the job card deck then these cards are omitted. For data stored on the disc pack these cards may be replaced by ATTACH, TAPE40, INPUT, ID = account code, SN = set name. In this case INPUT is the permanent file name. This disc pack must have been requested by a preceding MOUNT card.

8 & 9. STAGE, TAPE42, HI, POST. LABEL, TAPE42, E,L = label. Both these cards are required if plotter output is wanted; they specify that the plotter instruction file named TAPE42 written by the program is copied to a seven-track magnetic tape with the given label on completion of the job. Card 8 is invariant. The label on card 9 should enable easy identification of the plotting tape and may be any contiguous string of alphanumeric characters. If the label contains blanks or special characters it is enclosed by $ signs (e.g. L = $ MAP NO. I/54$).

10 to 13. The STAGE and LABEL cards for the grid value files named TAPE47 and TAPE48 are only required when the grid values are to be saved on magnetic tape. If the grid values are required on tape only two of these cards, either 10 and 11

or 12 and 13, are normally used. Cards 10 and 12 are invariant and the only variable on 11 and 13 is the label which follows the guidelines in 9 above. If the grid values are to be saved after gridding the input data or after an even number of iterations cards 10 and 11 are used. If the grid values are to be saved after an odd number of iterations cards 12 and 13 are used. The number of iterations is the sum of the fast, slow and linear iterations which if not specified on the GRID card have the default values 5, 5 and 2 respectively.

14. CONTOR. This card causes the file named CONTOR to be loaded and executed. The name must be that specified on card 5. Implicit in this card is a print limit of 5000 lines and the file names being as defined in the program. If a larger print limit is required, for example a large grid value listing, this card may be changed to CONTOR, PL = 10000. where the print limit is 10000 lines. If the file names are to be changed from those defined in the program the CONTOR card contains a list of the new file names separated by commas in the order in which they are defined in the program. The program file definition is equivalent to CONTOR, INPUT, OUTPUT, TAPE60, TAPE61, TAPE30, TAPE40, TAPE42, TAPE47, TAPE48. If the data input file has the name A and the plotter file is to be written on B this card will become CONTOR, ,,,,,, A, B. The remaining files are unchanged as indicated by the appropriate number of contiguous commas.

15 to 21. These cards are invariant except for the label of the plotter tape on card 19. This label may be represented by between 2 and 10 contiguous characters corresponding to the first characters of the label on card 9. Cards 17 to 19 are omitted if an automatic loan of the plotter tape is not required.

By convention within the Regional Gravity Group card 1 is punched on a pink data card, card 2 is blue, card 3 is orange, and cards 4 to 21 are green. The end of section card, which follows the job control section, is blue.

The data control section parameters are as follows (see 2.4 for key):

1. The MAPPLOT card specifies the projection and may specify corner marks or map name or number. The default projection is Transverse Mercator. The standard map projections are Transverse Mercator for scales of less than 1 to 1 million (eg 1 to 250 000), Lambert Conformal Conic to overlay the ICAO series or Transverse Mercator at 1 to 1 million, and Simple Conic for scales of more than 1 to 1 million (eg 1 to 2½ million). The corner marks are used to position the map if the graticule is not plotted. These marks may be drawn outside, at or inside the map graticule limit. If the corner marks are outside the plotter size limits a diagnostic is printed and the marks will be drawn at the graticule limits. A standard map name or number must match exactly a name or number in the reference table of standard names and numbers. The name or number may be punched anywhere in the 30 column field as it is automatically left justified. The program distinguishes between the corner mark parameters and the name or number by the numeric and alphabetic character of the respective variables.

2. The map name, area, and number parameters correspond to the standard mapsheet name, State (s), and international reference number (eg J55-1). These variables are automatically centred in their respective fields of 30, 20 and 10 columns. The ma limits are precise to 1 minute. The program can handle all latitude values ranging from -90 degrees at the North Pole to 90 degrees at the South Pole but only longitudes between -99 degrees (99W) to 180 degrees (180E). Longitudes between -180 degrees (180W) and -100 degrees (100W) may be mapped by adding 360 degrees to the limits (ie 180 to 260).

3. The spheroid card is standard for each scale and projection. The currently accepted standard spheroid is the Australian National Spheroid with an equatorial radius of 6378160 metres and a flattening factor of 298.25. The origin and zone width for Transverse Mercator projection on this spheroid is --183 degrees and 6 degrees respectively. The standard parallels for the Australian continent for the conic projections are 18 degrees and 36 degrees south latitude. Several other spheroids are available; the most commonly used is the Clark 1858 Spheroid with radius of 6378293 m, flattening of 294.26, zone origin of

111 degrees east, and width 5 degrees. The standard parallels should be roughly 1/6 of the latitude range from the top and bottom of the map so that 2/3 of the map area lies between. The scale is expressed as the reduction factor (eg 250000.0 for a 1 to 250 000 map). The graticule spacing is usually 5 minutes for 1 to 100 000, 15 minutes for 1 to 250 000 and 60 minutes latitude by 90 minutes longitude for 1 to 1 million and larger scales. The program will automatically register the graticule lines of coincide with the standard 1 to 250 000 map boundaries. A maximum of 50 lines may be drawn in either direction.

4. The data plot card usually has no parameters. The input data file number is normally redefined only when data is read from the input card deck in which case it is 60. The only other file numbers currently allowed are 1 and 40, the latter is the default. The remaining parameters should not be specified when reading data bank tapes as double unit conversions may result.

5. The DATUM card is required only if the values to be plotted on the map are of different units and to a different datum from the standard. The standard units and datum are milligals for gravity values to the 1965 Isogal Datum and metres for elevations to the Australian Height Datum. The gravity units and datum will be changed to micrometres/sec^2 and the 1976 Isogal Datum in a later edition of the program.

6. The title is specified only when the standard title supplied by the program is not suitable.

7. The border is usually 15 minutes for 1 to 250 000 scale, 30 minutes for 1 to 1 million and 60 minutes for 1 to 2½ million. The border should be at least 10 grid spaces and no less than twice the average station spacing to ensure good edge matching of adjoining maps. If the BORDER card is omitted a border of 15 minutes and Bouguer density of 2.67 t/m^3 is assumed. The other variables on the border card are usually omitted (see also notes after card 9 below).

8. The SURVEY card contains code, symbol and labelling information for up to seven surveys. Several SURVEY cards may be used and any field on any of the cards may be omitted. A maximum of 100 surveys may be specified. Surveys which are not

defined on SURVEY cards will be assigned a default code, symbol and labelling parameter provided that the number of surveys given a code and symbol does not exceed 100. Stations of any additional surveys are indicated by a dot or not plotted depending on the default symbol (see notes below). The convention for assigning symbols and codes is as follows (see 2.4 for symbol key):

Benchmarks - symbols 1 and 8

Regional helicopter - symbols 3, 4 and 6

Semi-detailed - symbols 2,5, 7 and 9

Detailed - symbol 99, no code.

Isogal - symbols 10 and 11, code letters I to N

9. Survey area numbers are specified on SEGMENT cards if selective searching of specific blocks is required, for example when only stations from a particular survey are to be plotted. When a small number of survey areas are not wanted on a map the best way of skipping these is by using the SURVEY card rather than several SEGMENT cards (see (i) below). The usual labelling frequency for 1 to 250 000 sheets is 1 for regional helicopter coverage, 5 for semi-detailed coverage and 10-100 for detailed surveys. These frequencies can be scaled in proportion to the map scale. On maps of scales larger than 1 to 1 million labelling is usually of little use and the default symbol number 99 (.) should be selected.

Notes on the selection of survey symbols and labelling.

(i) Station selection by survey number: If the default symbol number on the BORDER card (7) is negative only surveys specified on a SURVEY card (8) with a non-negative symbol number will be used. If the default symbol number is omitted, zero or positive, all surveys will be used except those specified on a SURVEY card with a negative symbol number.

(ii) Symbol selection: If the default symbol on the BORDER card is zero or omitted the program will assign symbols to surveys which are not specified on a SURVEY card. The program will use symbols 1 to 9 in rotation for each new survey found. If the symbol number for a particular survey on a SURVEY card is 0 or omitted stations of that survey will not be plotted but their values will be used in the contouring.

(iii) Labelling: No annotation is written for stations of surveys which are not used, are not plotted or are plotted with the dot symbol. If the label frequency for a survey specified on a SURVEY card is negative no annotation is written for that survey; if it is zero or omitted the frequencies on the SEGMENT card (9) are used. If the frequency on the SURVEY card is positive and the frequencies on the SEGMENT card are omitted or zero then the SURVEY card frequency will be used for all 3 annotation types for stations of the specified survey. If the SURVEY card frequency is positive and one or more of the SEGMENT frequencies are positive the product of the respective SEGMENT frequencies with the SURVEY frequency will be used as the respective annotation frequencies for the specified survey.

10. The GRID card contains information which largely determines the cost of computing a particular contour map. The job time is roughly proportional to the inverse square of the grid spacing. The iteration time is proportional to the number of iterations and is not so critical. From preliminary results the optimum grid spacing seems to lie between $\frac{1}{2}$ and 1/5 of the average station spacing. If the grid is larger than the station spacing averaging of several stations will occur. If the grid spacing is too small the contours will become wavy or square. The number of iterations should be at least twice the average number of grid spacings between observations. The number of linear iterations should be about equal to the number of grid points between ovservations. The number of fast and slow iterations are usually the same and between 1 and 2 times the number of grid points between observations. The latitude to longitude grid ratio is specified only when contouring anisotropic data such as marine or airborne traverses where the station spacing in the latitude and longitude are significantly different, otherwise the grid is assumed to be isotopic.

11. The contour levels are usually chosen so that between 20 and 50 levels are drawn on the map. The standard contour intervals for gravity anomalies are 5 and 20 mGal (50 and 200 um/s^2). The maximum and minimum levels are useful for

separating ranges of levels in the preparation of colour masks. The inhibition distance is normally omitted.

By convention cards 1, 4, 6, 7, 9, 10, 11 and 12 are punched on orange data cards and cards 2, 3, 5 and 8 on white cards. The end of section card 13 is blue.

3.4 Program Maintenance

The program is stored in binary form as a permanent file named CONTOR6 with account code DMRXRA and multi-read password ASM on the disc packs with set names DMR3922 and DMR8119.

To reload the program onto the disc from scratch and to remove the old edition, if present, the following job is required:

```
.CYJ,PROJECT=DMRXRC.
AMGRVBC, T20, YL1.
COMMENT.
MOUNT, SN=DMR3922, VSN=PMC510.
REQUEST, KONTOR, *PF, SN=DMR3922.
FTN, B=KONTOR.
PURGE, CN, CONTOR6, ID=DMRXRA, PW=ASM, SN=DMR3922.
EXIT, U.
CATALOG, KONTOR, CONTOR6, ID=DMRXRA, PW=ASM.
EXIT, U.
RETURN, CN, KONTOR.
EXIT, S.
RETURN, CN, KONTOR.
end of section
Fortran decks as listed in 3.1
end of information
```

The binary program is divided into 52 sections on the permanent file CONTOR6 as described in section 3.1. If, for example, subroutine LINEATE is to be recompiled the following job would be used:

```
.CYJ, PROJECT = DMRXRC.
AMGRVBC, T20, YL1.
COMMENT.
MOUNT, SN=DMR3922, VSN=PMC510.
REQUEST,KONTOR, *PF, SN=DMR3922
PURGE, CN, CONTOR6, ID=DMRXRA, PW=ASM, SN=DMR3922.
COPYS, CN, KONTOR, 44.
COPYS, CN, C, 1.
FTN, B=KONTOR, SL.
COPY, CN, KONTOR.
CATALOG, KONTOR, CONTOR6, ID=DMRXRA, XR=ASM.
EXIT, U.
RETURN, CN, KONTOR.
EXIT, S.
RETURN, CN, KONTOR.
end of section
Fortran deck of LINEATE
end of information
```

4. REFERENCES

BRIGGS, I.C., 1974 , Machine Contouring using Minimum Curvature Geophysics, v.39, p 39-48.

MURRAY, A.S., 1974, The Australian National Gravity Repository Computer System: Bureau of Mineral Resources Record 1974/68 (unpubl.).

APPENDIX A : DIAGNOSTIC MESSAGES

1. MAPPLOT CARD MISSING name WAS READ

Program MAPPLOT has read a card with a name other than MAPPLOT or RESTART. The program searches for the next MAPPLOT or end of section card.

2. UNKNOWN PROJECTION

The projection given on the MAPPLOT card cannot be recognized. Columns 11 to 20 must be blank or contain MERCATOR, TRANSVERSE, CONIC, LAMBERT or PLANE starting in Column 11. Execution is terminated.

3. Projection name PROJECTION IS NOT YET AVAILABLE

The projection name specified on the MAPPLOT card is not yet available for use although the name is recognized by the program. This message will occur only temporarily while a new projection is being implemented. Execution is terminated.

4. UNRECOGNIZED MAP NAME ON MAPPLOT CARD = name

The map name or number specified on the MAPPLOT card or on the map name card cannot be found in the table of standard names. This diagnostic should appear only when defining the required map by name or number only. It may indicate that the map limits have been omitted from the map name card or that the registration mark co-ordinates have been mis-punched. The map is terminated.

5. ILLEGAL KEY WORD AT index IN REFERENCE TABLE.

5a. CROSS REFERENCE INDEX number IS MISSING

These messages indicate an error in the standard map reference tables in subroutine MAPSEEK. The subroutine will need correcting and recompiling to remove the error.

6. CONTROL CARD READ FOR MAPNAME. FIRST WORD = name

The program has read a card with a * in column 1 where the map name card was expected. The map is terminated.

7. Number CHARACTERS IN NAME. ONLY 100 WILL BE USED

This message will only appear if the program is malfunctioning. All the map parameters should be checked for errors which may not have been detected by the program.

8. ERROR IN MAP LIMITS. LIMITS GIVEN AS -

The latitude or longitude limits on the map name card are in the wrong order or omitted. The first latitude and longitude must be less than the second. The program searches for the next map.

9. MISSING SPHEROID PARAMETERS GIVEN DEFAULT VALUES

One or more of the parameters required to define the spheroid are not specified or have illegal values. The default values assumed by the program are the parameters for the Australian National Spheroid at a scale of 1:250,000.

10. ILLEGAL ORIGIN OR ZONE WIDTH FOR TRANSVERSE MERCATOR PROJECTION

The origin is too far west or too far east for the map required or the zone width is too small, too large, or negative. The map is terminated.

11. ERROR IN GRATICULE SPACINGS GIVEN AS minutes AND minutes SPACINGS REDEFINED AS 60.0 AND 90.0 MINUTES

The specified graticule spacings are too small for the map or they have been omitted. The spacings are redefined and processing continues.

12. MAP SIZE IS TOO SMALL - CHECK THE SCALE

The size of the map base is less than 2.5 cm (1 inch) by 1.3 cm (½ inch). The scale or the map limits may be in error. The map is terminated.

13. WARNING - MAP IS TOO WIDE FOR DRUM PLOTTER

The width of the map in longitude is more than 83.8 cm (33 inches). The map will not fit on the drum plotter but may fit on the flat bed. Execution continues.

14. MAP SIZE TOO LARGE FOR BOTH PLOTTERS

The map dimensions are more than 114.3 cm (45 inches) by 147.3 cm (58 inches) which exceed the limits of the flat bed plotter. The program searches for the next map.

15. REGISTRATION MARKS OUTSIDE PLOTTER LIMITS

The specified registration mark co-ordinates lie outside the range of the plotters. If the marks will fit on the flat bed plotter they will be drawn as specified, otherwise the marks will be drawn at the corners of the map base. Execution continues.

16. CONTROL CARD MISSING IN BAHMAP

Program BAHMAP has read a card with a name other than BAPLOT, FAPLOT, HTPLOT, XYPLOT or RESTART. The map is terminated and the program searches for the next map.

17. EOF READ FOR TAPENAME ON UNIT - number

An end of file or end of section was read on the designated unit. This error may be due to one of the following: (a) Input data unit was specified incorrectly or omitted. When cards are to be read from the card reader unit 60 should be specified. (b) An empty file on the input data unit. (c) The data unit was not rewound after it had been used previously in the job (this applies only to BCD data).

18. WRONG TAPE - name/SHOULD BE - name

The name on the input tape (first name printed) does not match the name given in the data control deck (last name printed). Normally the tapename is not specified in the data control deck unless the identity of the data tape is in doubt. The program will search for the next map.

19. 100 SURVEYS HAVE BEEN USED, ANY OTHERS WILL BE INDICATED BY symbol AND NOT LISTED

Only 100 survey numbers may be differentiated by an individual symbol and code. Any other surveys will not be annotated on the map and the stations used will be indicated only by a dot or no symbol depending on the default symbol. Execution continues.

20. NO DATA FOUND FOR THIS MAP

No stations with all the necessary principal facts were found in the map or border area specified. The program will search for the next map.

21. AREA NAMED - name - IS NOT ON THE TAPE

The survey area number specified on the SEGMENT card

cannot be found on the input data unit. The segment card will be ignored, and the job continues.

22. PARITY ERROR ON INPUT UNIT - number

This message will occur only if there is a hardware malfunction. The job should continue normally.

23. ILLEGAL STATION NUMBER IN BCD INPUT DATA - number

The datum for the input binary data is incorrectly specified or there is an error in a station number for BCD input data. All station numbers must be numeric. The station is not used for the map and the job continues.

24. ERROR OCCURRED IN AREANAME (AREAFIND). NAME USED - name

A parity error occurred during the input of the area name of a data block on the input unit. This error will occur only if there is a hardware malfunction, and the job will continue.

25. ERROR OCCURRED AFTER -TH STATION NUMBERED number

A parity error occurred during data input causing the loss of some information. See remarks for previous error message.

26. ILLEGAL STATION IN OUTPUT DATA number SKIPPED.

26a. Number OUTPUT ERRORS DETECTED. PROBABLY DUE TO UNDEFINED INPUT DATA TYPE

26b. ERROR IN FUNCTION IBYTE

These errors should not appear if the program is running normally, and they probably indicate a hardware malfunction. The job should be run again and if these error messages reoccur the program must be reloaded.

27. ILLEGAL DATUM WORD - datum, DATUM USED IS - datum

The datum on the input tape or the DATUM card is incorrectly specified. A default datum is used and execution continues.

33. END OF FILE OR PARITY ERROR ON RESTART UNIT number

This message indicates a hardware malfunction in a normal job or a blank tape in a restart job. The restart data input should be assigned to unit 40 and this unit should be rewound if it is used previously in the job. The program will search for the next map.

34. DATA IS IN CONTOUR FORMAT, NO MORE ITERATIONS ALLOWED

All the grid values lying outside the cutoff distance from the observations have been given the default value and no more iterations can be performed. If more iterations are required after saving the grid values, the iteration process must be stopped by specifying the restart option on the GRID card. The program searches for the next map.

35. ABOVE OBSERVATION AT OR BEYOND EDGE

The observation lies too close to the edge of the grid block to be used as a control value. Execution continues.

36. DIFFERING VALUES AT LAT value MINS, LONG value MINS
NEW VALUE = value, OLD VALUE = value

The observation at the position indicated, in minutes from the top left corner of the map, has a value differing from the existing average value at the nearest grid point by more than 0.1 of the range of values on the map. The observation value and the existing average value are listed. Processing continues with all flagged values being used.

37. ERROR ON RESTART TAPE number

An error has been detected on the restart unit during iteration. This will occur only if there is a hardware malfunction. The program will search for the next map.

38. CONTOUR CARD ERROR. FIRST WORD WAS - name

The iteration process was stopped before the grid values were prepared for contouring. The program searches for the next map.

40. MORE THAN 500 LEVELS ON MAP. RANGE = value LEVEL = value

The large number of contour levels required for the map indicates an error in the specified contour interval or an error in the data causing an excessive range. The program terminates the map without contouring and searches for the next map.

41. MORE THAN 200 LINES OR 7000 CUTS IN THIS BLOCK

The contour lines in this contour block are too numerous for the size of the arrays. The program will write out the lines in several operations. Processing will continue.

42. NEGATIVE ARGUMENT IN ISQRT

An error has occurred during calculation of the contour line cuts. This message should not appear during a normal job but processing continues.

43. VALUE TO BE INSERTED IS TOO LARGE. OCTAL VALUE IS value

An array has probably overflowed in subroutine LINKUT and caused this error. A program correction is necessary.

APPENDIX B. JOB EXAMPLES

Two examples of frequently used types of jobs are shown in Figures 2 and 3.

Example 1 (Figure 2). The data to be mapped are stored on more than one file. Program LOCABA, which locates data within specified limits, is used to form a merged data file on TAPE40; this file is then read by CONTOR. The cards starting at *LOCATE and ending at the next end of section are read by program LOCABA and a description of their format can be found in a guide for that program. In brief, LOCABA searches two magnetic tape files named TAPE1 and TAPE2 for data lying within the area bounded by 19 and 29 degrees latitude and 113 and 123 degrees longitude. The located block of data is given the area name PILBARA BLOCK DATA. The CONTOR data control cards request a 1 to 2.5 million simple conic map base; the Bouguer anomaly at density 2.67 tonne/m^3 as the contour value and 3 pens to be used in drawing the contour lines.

Example 2 (Figure 3). Four station plots are required. The maps are defined by standard names and are assumed to be at scale 1 to 250 000 and Transverse Mercator projection. Every 5th station on each map is fully annotated.

FIGURE 2

```
*CY,CPDMRXRC,AMGRVSU
AMGRVSU,T400,YL1,ML1,NL3.
COMMENT,********************** CCOMRXRC ********** A.MURRAY *** EXT 223.
MOUNT,SN=DMR3922,VSN=PMC510.
MOUNT,SN=DMR8119,VSN=C030.
ATTACH,LOCABA,LOCABA5,ID=DMRXRA,SN=DMR3922.
ATTACH,CONTOR,CONTOR6,ID=DMRXRA,SN=DMR8119.
STAGE,TAPE1,PE,VSN=3660.
STAGE,TAPE2,PE,VSN=5371.
STAGE,TAPE42,HI,POST.
LABEL,TAPE42,W,L=PILBARA.
STAGE,TAPE48,PE,POST.
LABEL,TAPE48,W,L=PILGRID.
LOCABA.
CONTOR.
EXIT,U.
RETURN,LOCABA.
RETURN,CONTOR.
ATTACH,TRS,TRS.
LIBRARY,*,TRS.
LOAN,M/PIL.
EXIT,S.
RETURN,LOCABA.
RETURN,CONTOR.
>  END OF SECTION
*LOCATE  42
*INPUT  2
PILBARA BLOCK DATA
*LIMITS      19 20.00  29 28.00 113 [illegible] 121 [illegible] 1 2          1
PILBARA BLOCK DATA
>  END OF SECTION
*MAPPLOT  CONIC
     PILBARA  BLOCK            WESTERN AUSTRALIA   F.G/50-1   2000 2800 1400 1200
AUSTRALIAN NATIONAL SPHEROID  6378160.  298.25     1800 3600 2500000.  60.  90.
*BAPLOT
*BORDER   15.0 2.67          -1
*SURVEY   6491 I 1  6792 J 0  6793 K 0  7000 A 6 57141 H 6 56205 C 1 56914 B 1 5
*SURVEY   6915 E 1 57130 F 3 57141 G 3 57142 H 3 57143 L 3 57130 M 3 57130 N 5 5
*SURVEY   7335 O 7436009 P 6 56824 Q 4 56805 R 0 56918 S 1 5
*SEGMENTS      1
*GRID 4.00          5   10   10
*CONTOUR 35.00 20.00
>  END OF SECTION
>  END OF INFORMATION
```

FIGURE 3

```
*CY,CPDMRXRC,AMGRVPW
AMGRVPW,T600,ML1,NL1.
COMMENT,********************** CCDMRXRC ********** A.MURRAY *** EXT 223.
MOUNT,SN=DMR3922,VSN=PMC510.
ATTACH,CONTOR,CONTOR6,ID=DMRXRA,SN=DMR3922.
STAGE,TAPE40,PE,VSN=6898.
STAGE,TAPE42,HI,POST.
LABEL,TAPE42,W,L=BHP.
CONTOR.
EXIT,U.
ATTACH,TRS,TRS.
LIBRARY,*,TRS.
LOAN,0/BHP.
RETURN,CONTOR.
EXIT,S.
RETURN,CONTOR.
>  END OF SECTION
*MAPPLOT
DEVONPORT
*BAPLOT
*TITLE                STATION PLOT
*SEGMENTS    5   5   5
*MAPPLOT
(LAUNCESTON)
*BAPLOT
*TITLE                STATION PLOT
*SEGMENTS    5   5   5
*MAPPLOT
SMITHTON
*BAPLOT
*TITLE                STATION PLOT
*SEGMENTS    5   5   5
*MAPPLOT
NARRANDERA
*BAPLOT
*TITLE                STATION PLOT
*SEGMENTS    5   5   5
>  END OF SECTION
>  END OF INFORMATION
```

```
      OVERLAY(LUNTOR,0,0)
      PROGRAM CONTOR7(INPUT,OUTPUT,TAPE60=INPUT,TAPE1,TAPE2,TAPE30,
     1               TAPE40,TAPE42,TAPE46,TAPE47,TAPE48,TAPE49,TAPE50)
C * * * THIS PROGRAM CONTROLS THE PLOTTING AND CONTOURING OF RANDOM DATA.* * * C
C * * * DATA CONTROL DECK CONSISTS OF THE FOLLOWING TEN CONTROL CARDS. * * * * C
C * * * 1.MAPPLOT,2.MAPNAME,3 PROJECTION,4.BAPLOT,FAPLOT,HTPLOT OR XYPLOT. * * *
C * * * 5.TITLE, 6.BORDER, 7.SURVEY, 8.SEGMENT(S), 9.GRID, 10.CONTOUR. * * * * C
C * * * AN EOS CARD WILL TERMINATE PROCESSING IF READ AFTER CONTOUR OR IT  * * * C
C * * * WILL TERMINATE THE MAP ELSEWHERE (EXCEPT AFTER A SEGMENT CARD).* * * * C
C . . . FILES USED - INPUT IS CARD READER, OUTPUT IS LINE PRINTER.
C . . .  TAPE1,2 ARE OPTIONAL DATA INPUT, TAPE40 IS STANDARD DATA INPUT.
C . . .  TAPE42 IS PLOTTER OUTPUT,TAPE50 IS CUMULATIVE GRID SAVE OUTPUT.
C . . . TAPE30 IS X Y Z VALUE AND CONTOUR GRID WINDOW STORE.
C . . . TAPE46,49 ARE TEMPORARY STORES, TAPE47,48 STORE THE GRID VALUES.
C - - - THIS EDITION IS DATED 26/ 9/77.  THIS VERSION DATED 01/08/78.
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT ALN,XX,YY,XL,XR,YT YB,S00
      DATA (LOTX=0),(LOTY=0),(NEDIT=7),(NEDATE=10H1978/08/01),(NEVERS=6)
      DATA (NRE=9(1H )) , (LUN=0) , (NOPB=0)
      LD=30 $ LQ=46 $ LR=47 $ LS=48 $ LT=49 $ INMAIN=0 $ NRE=8HCN7/5 ,
C . . . SET INPUT ERROR CHECK AND DUMP OUTPUT FILE TO PRINTER.
      CALL ERRSET (IERR,200)
      LOP=6LOUTPUT $ CALL RETURN (LOP) $ REWIND LOP
   10 LOTX=LOTX+1
      TUN = TIMEF(CLOK)
C . . . CALL MAPDATA TO DRAW MAP AND LOCATE AND POST OBSERVATIONS.
      CALL OVERLAY (6HCONTOR,1,0)
      IF(NAME.EQ.7HCONTOUR)GO TO 20 $ IF(NAME.EQ.5HNOMAP)GO TO 10
      IF (LOTX.GT.1000) GO TO 30
C . . . CALL GRID TO GRID THE DATA AND SMOOTH THE GRID SURFACE BY ITERATION.
      CALL OVERLAY (6HCONTOR,2,0)
      IF (LOTX.GT.1000) GO TO 30
C . . . CALL CONTOUR TO DRAW THE CONTOUR LINES FROM THE GRIDDED DATA.
   20 CALL OVERLAY (6HCONTOR,3,0)
      IF (LOTX.LT.1000) GO TO 10
C . . . ERROR DETECTED. ABANDON THIS MAP AND START NEXT MAP.
   30 LOTX=LOTX-1000 $ GO TO 10
      END
      SUBROUTINE BEGINIT
C * * * THIS ROUTINE PRINTS THE TIME OF ENTRY AND EXIT FROM ROUTINE NAME.* * * C
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      TODAY = DATE(0.)
      DECODE(10,10,TODAY)D,M,Y
   10 FORMAT(3(1X,R2),1X)
C . . . PRINTED FROM BEGINNING OF ROUTINE NAME.
      PRINT 20,D,M,Y
   20 FORMAT(1H1/1H0,61X,2H* ,R2,2(1H/R2),2H */1H0)
      TIMES = TIME (CLOK)
      TYM = TIMEF(CLOK) - TUN
      IF(LOTX.NE.LOTY)PRINT 40,NEDIT,NEVERS,NEDATE,LOTX $ LOTY=LOTX
   40 FORMAT(1H0,11X*PROGRAM - CONTOR *I2* . VERSION *I3* DATED - *A10,
     1       *.  THIS IS MAP NUMBER *I3* OF THIS JOB.*)
      PRINT 50,NAME,NAB, TYM ,TIMES
   50 FORMAT(1H0,11X,R9,*COM*,A6 16X,A10,24X,*JOB TIME IS *,
     1       -3PF10.3,* SECOND.
```

```
      RETURN
C . . . CALLED FROM FINISH OF ROUTINE NAME.
      ENTRY FINIT
      NAB=6HPLETED
      GO TO 30
      END
      SUBROUTINE JRICHEK (N)
C * * * THIS ROUTINE DRAWS ORIGIN REGISTRATION MARKS FOR EACH PLOT BLOCK.* * * C
      COMMON/CONTROL/NAIM(8),.NMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION IP(3)
      SIZE=KSIZE
      CALL FACTOR (1.0)
      IF(N.GE.5)GO TO 10
C . . . DRAW THE + AND RELEVANT LETTER.
      CALL PLOT (XR-1.4*SIZE,-2.7*SIZE,3)
      CALL PLOT (XR-1.2*SIZE,-2.7*SIZE,2)
      CALL PLOT (XR-1.3*SIZE,-2.6*SIZE,3)
      CALL PLOT (XR-1.3*SIZE,-2.8*SIZE,2)
      NX = N-2*(N/2)  $  NY=N/3
      XO=XR-(1.26 +NX*0.1)*SIZE $ YO=-(2.685+NY*0.1)*SIZE
      M=SHIFT(N,54)
      CALL SYMBOL (XO,YO,0.07*SIZE M,0,1)
      RETURN
C . . . ENCLOSE THE + AND LETTERS, ANNOTATE THE TIME AND CLOSE THE PLOT FILE.
   10 IF(NAME.EQ.5HNOMAP)GO TO 170 $ IF(LUN.EQ.0)GO TO 50
      CALL PLOT (XR-1.4*SIZE  2.6*SIZE,3)
      CALL PLOT (XR-1.2*SIZE  2.6*SIZE,2)
      CALL PLOT (XR-1.2*SIZE,-2.8*SIZE,2)
      CALL PLOT (XR-1.4*SIZE,-2.8*SIZE,2)
      CALL PLOT (XR-1.4*SIZE,-2.6*SIZE,2)
      ENCODE (10,15,IP) NEDIT,NEVERS
   15 FORMAT (I2,1H.,I3)
      CALL SYMBOL (XR-1.4*SIZE,-0.5-2.7*SIZE,0.0 ,IP,0,6)
      ENCODE (20,20,IP) TIMES
   20 FORMAT(*JOB TIME =*,-3PF6.2,* SEC*)
      CALL SYMBOL (XR-SIZE,-0.5-2.7*SIZE,0.07,IP,0,20)
      IF(LOTX.EQ.LOTY)PRINT 25,NRE
   25 FORMAT (1H0,* PROGRAM VERSIONS USED - *,9A8)
      IF(N.LT.6)GO TO 40
      IF(LOT.EQ.3HEOF)GO TO 50
      LOTX=LOTX+1000 $ IF(NAME.EQ.6HNEWMAP)GO TO 40 $ PRINT 30
   30 FORMAT(23X,1H*/23X,3H***/10X,18(1H*)/10X,20(1H*),* MAP TERMINATED*
     1      /10X,18(1H*)/23X,3H***/23X,1H*)
C . . . ANOTHER MAP IS REQUIRED.
   40 IF(LOT.EQ.3HEND)GO TO 50 $ RETURN
C . . . NO MORE MAPS REQUIRED. CLOSE THE PLOT FILE.
   50 IF(LUN.EQ.0)STOP $ CALL PLOT (0,0,999) $ NOPB=998
      ENTRY TAGBLOK
  100 NOPB=NOPB+1
      WRITE (LT,110) NOPB $ IF(NOPB.LT.999)RETURN
  110 FORMAT (* MARK *,I3,1X,20(1H-))
      ENDFILE LT $ REWIND LT $ PRINT 120
  120 FORMAT (1H1/5X,*SUMMARY OF PLOT BLOCKS ON*//6X,*TAPE NUMBER */1X)
  130 READ (LT,140) IP $ IF(EOF(LT))170,150
  140 FORMAT (3A10)
  150 PRINT 140,IP $ GO TO 130
  170 IF(LOT.EQ.3HEND.OR.LOT.EQ.3HEOF)STOP
      END
```

```
      SUBROUTINE PARAM (DX,DY,BOR,NIP,NAL)
C * * * THIS ROUTINE ANNOTATES THE MAP WITH THE CONTOURING PARAMETERS USED.* * C
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION MUF(2),C(2)
      SIZE=KSIZE $ YS=0.07*SIZE
      IF(DX.EQ.0.0)GO TO 4
C . . . LATITUDE GRID IN MINUTES.
      X1=XR-1.0*SIZE $ Y1=0.1-2.7*SIZE $ MUF=4HLAT
      ENCODE (20,1,MUF) MUF(1),DY
    1 FORMAT(A4,* GRID =*,F5.2,* MIN*)
      CALL SYMBOL (X1,Y1,YS/2,MUF,0,20) $ Y1=Y1-YS
C . . . LONGITUDE GRID IN MINUTES.
      MUF=4HLONG $ ENCODE (20,1,MUF) MUF(1),DX
      CALL SYMBOL (X1,Y1,YS/2,MUF,0,20) $ Y1=Y1-YS
C . . . FAST / TOTAL NUMBER OF ITERATIONS.
      ENCODE (20,2,MUF)NIP,NAL
    2 FORMAT(*ITERATIONS = *,I3,*/*,I3)
      CALL SYMBOL (X1,Y1,YS/2,MUF,0,20) $ Y1=Y1-YS
C . . . WIDTH OF AREA OF BORDER DATA IN MINUTES.
      ENCODE (20,3,MUF)BOR
    3 FORMAT(*BORDER  =  *F5.2* MIN*)
      CALL SYMBOL (X1,Y1,YS/2,MUF,0,20)
      RETURN
C . . . CONTOUR LEVELS USED FOR MAP.
    4 IF(NAL.GT.1)GO TO 5
      X1=XR-1.6*SIZE $ Y1=-2.27*SIZE $ X2=XR-0.2*SIZE
      MUF(1)=10H  CONTOUR  $ MUF(2)=10HLEVELS AT
      CALL SYMBOL (X1,Y1,YS,MUF,0,20)
    5 CALL NEWPEN(NIP) $ C(1)=DY $ C(2)=BOR
        DO 7 I=1,2 $ IF(C(I).EQ.0.0)GO TO 7
        ENCODE (7,6,MUF) C(I)
    6   FORMAT (F7.2)
        CALL SYMBOL (X2,Y1,YS,MUF,0,7)
        Y1=Y1-2.0*YS
    7   CONTINUE
      RETURN
      END
      FUNCTION ISQRT (N)
C . . . CALCULATE THE INTEGER SQUARE ROOT OF AN INTEGER BY ITERATION.
      IF(N.GE.0)GO TO 20
      PRINT 10,N $ ISQRT=0 $ RETURN
   10 FORMAT (4H0***,* NEGATIVE ARGUMENT IN ISQRT = *,I6)
   20 IF(N.GE.2)GO TO 30 $ ISQRT=N $ RETURN
   30 M=N/2
        DO 40 K=1,10000
        MM=((M+N/M)+1)/2 $ IF(MM.EQ.M)GO TO 50 $ M=MM
   40   CONTINUE
   50 ISQRT=MM
      END
      SUBROUTINE PROJECT (JECTOR)
C * * * THIS ROUTINE CONTROLS THE CALLING OF XYCOORDS FOR VARIOUS PROJECTIONS.
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      J=9
      IF(JECTOR.EQ.10H          )GO TO 10
      IF(JECTOR.EQ.10HTRANSVERSE)GO TO 10
      IF(JECTOR.EQ.10HCONIC     )GO TO 20
      IF(JECTOR.EQ.10HPLANE     )GO TO 30
      IF(JECTOR.EQ.10HLAMBERT   )GO TO 40
      IF(JECTOR.EQ.10HMERCATOR  )GO TO 50
      PRINT 5
```

```
    5 FORMAT (4H0***,* UNKNOWN PROJECTION *)
      STOP
   10 JECTOR=J=1 $ RETURN
   20 JECTOR=J=2 $ RETURN
   30 JECTOR=J=3 $ RETURN
   40 JECTOR=J=4 $ RETURN
   50 JECTOR=J=5 $ RETURN
   80 PRINT 90,JECTOR $ STOP
   90 FORMAT (4H0***,1X,A10,* PROJECTION IS NOT YET AVAILABLE *)
C . . . ESTABLISH PROJECTION PARAMETERS.
      ENTRY XYCOORD
      IF(J.GE.9) GO TO 190
      IF(INMAIN.NE.0)GO TO 105
      READ 100,NAIM $ IF(EOF(60))400,105
  100 FORMAT (8A10)
  105 INMAIN=0 $ K=NAIM.AND.MASK(6) $ IF(K.EQ.1L*)INMAIN=1
      GO TO (110,120,130,140,150, 80, 80, 80)J
  110 CALL XYTRANS $ RETURN
  120 CALL XYCONIC $ RETURN
  130 CALL XYPLANE $ RETURN
  140 CALL XYLMBRT $ RETURN
  150 CALL XYMRCTR $ RETURN
  190 RETURN
C . . . CONVERT COORDINATES TO PLOTTER UNITS.
      ENTRY COORDS
      IF(J.GE.9) GO TO 290
      GO TO (210,220,230,240,250, 80, 80, 80)J
  210 CALL COORDTM $ RETURN
  220 CALL COORDCO $ RETURN
  230 CALL COORDXY $ RETURN
  240 CALL COORDLC $ RETURN
  250 CALL COORDMC $ RETURN
  290 RETURN
C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY XYDATA
      IF(J.GE.9) GO TO 390
      GO TO (310,320,330,340,350, 80, 80, 80)J
  310 CALL TMDATA $ RETURN
  320 CALL CODATA $ RETURN
  330 CALL DATAXY $ RETURN
  340 CALL LCDATA $ RETURN
  350 CALL MCDATA $ RETURN
  390 RETURN
  400 LOT=3HEOF $ CALL CRICHEK(6)
      END
      SUBROUTINE XYMRCTR

C       THIS VERSION OF XYCOORDS COMPUTES EASTINGS AND NORTHINGS ON
C       THE STANDARD MERCATOR PROJECTION

      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION SFEROID(3),MAPTYPE(3),MAPZONE(3)
      INTEGER SFEROID

C . . . ESTABLISH PROJECTION PARAMETERS.
      IF(INMAIN.NE.0)GO TO 11
      DECODE (80,10,NAIM) SFEROID,A,CLATD,CLATM,SCALE,XSPACE,YSPACE
   10 FORMAT (3A10,F10.0,10X,F3.0,F2.0,5X,F10.0,2F5.0)
      F=0.0
   11 IF(A.LE.0.0)F=A=6378160.0
```

```
      IF(SCALE.LE.0.0)F=SCALE=250000.0
      IF(XSPACE*YSPACE.LE.0.0)F=XSPACE=YSPACE=K=6*SCALE/100000
      IF(XSPACE+YSPACE.LT.0.1)F=XSPACE=YSPACE=1.0
      IF(F.NE.0.0)PRINT 14
   14 FORMAT(1H0,9X,*MISSING SPHEROID PARAMETERS GIVEN DEFAULT VALUES*)

   15 PI=57.2957795 $ PI2=2.0*PI
      IF(CLATD.LT.0.0)CLATM=-CLATM $ CLAT=(CLATD+CLATM/60.0)/PI
      RADIUS=A*COS(CLAT)/(SCALE*0.0254)
      ALPHA=RADIUS/PI $ BETA=45.0/PI

      RETURN

C . . . CONVERT COORDINATES TO PLOTTER UNITS.
      ENTRY COORDMC

      XX=ALPHA*ALN/60.0
      YY=-RADIUS*ALOG(TAN(BETA+ALT/(PI2*60.0)))

      RETURN

C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY MCDATA

      SIZE = KSIZE

      ENCODE(30,50,MAPTYPE
   50 FORMAT (30HSTANDARD MERCATOR PROJECTION  )
      I=CLATD $ J=CLATM $ K=1RS $ IF(I.GE.0)GO TO 55
      I=-I $ J=-J $ K=1RN
   55 ENCODE (30,60,MAPZONE) I,J,K
   60 FORMAT (21HSCALE DEFINED AT LAT ,2I3,1X,R1,1H.)

      PRINT 70, SFEROID,MAPTYPE,MAPZONE
   70 FORMAT(//20X,3A10,2(/,20X,3A10))

      OFFSET = -0.4*SIZE
      CALL SYMBOL (OFFSET,-1.0*SIZE,0.07*SIZE,SFEROID,0,30)
      CALL SYMBOL (OFFSET,-1.2*SIZE,0.07*SIZE,MAPTYPE,0,30)
      CALL SYMBOL (OFFSET,-1.4*SIZE,0.07*SIZE,MAPZONE,0,30)

      RETURN

      END
      SUBROUTINE XYTRANS                                              XYC 001

C       THIS VERSION OF XYCOORDS COMPUTES EASTINGS AND NORTHINGS ON
C       THE TRANSVERSE MERCATOR PROJECTION

      COMMON/CONTROL/NAME(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION SFEROID(3),MAPTYPE(3),MAPZONE(3)
      INTEGER SFEROID

C . . . ESTABLISH PROJECTION PARAMETERS.
      IF(INMAIN.NE.0)GO TO 11
      DECODE (80,10,NAME) SFEROID,A,RECIPF,ORIGIN,WIDTH,SCALE,XSPACE,
     1                    YSPACE
   10 FORMAT(3A10  ,2F10.0,2F5.0,F10.0,2F5.0)
      F=0.0
   11 IF(A.LE.0.0)F=A=6378160.0
```

```
      IF(RECIPF.LE.0.0)F=RECIPF=298.25
      IF(ORIGIN.EQ.0.0)F=ORIGIN=-183.0
      IF(WIDTH.LE.0.0)F=WIDTH=6.0
      IF(SCALE.LE.0.0)F=SCALE=250000.0
      IF(XSPACE*YSPACE.LE.0.0)F=XSPACE=YSPACE=K*6*SCALE/100000
      IF(XSPACE+YSPACE.LT.0.1)F=XSPACE=YSPACE=1.0
      IF(F.NE.0.0)PRINT 14
   14 FORMAT(1H0,9X,*MISSING SPHEROID PARAMETERS GIVEN DEFAULT VALUES*)

   15 AVGLONG = 0.5*(WLN+ELN)
      IF(ORIGIN+0.5*WIDTH.LT.-180.0)GO TO 16
      ORIGIN = 60.0*ORIGIN                                              XYC 017
      WIDTH = 60.0*WIDTH                                                XYC 018
      IF(AVGLONG.GE.(ORIGIN-0.5*WIDTH))GO TO 19
   16 PRINT 17 $ SCALE=0.0 $ RETURN
   17 FORMAT(1H0,9X,*ILLEGAL ORIGIN OR ZONE WIDTH FOR TRANSVERSE MERCATO
     1           *OR PROJECTION*)

   19 ZONE = -1.0                                                       XYC 020
   20 ZONE = ZONE+1.0                                                   XYC 021
      CM = ORIGIN+ZONE*WIDTH                                            XYC 022
      IF(ABS(AVGLONG-CM).LE. 0.5*WIDTH) 30,20
   30 CONTINUE                                                          XYC 024

      RADIUS = A/(SCALE*0.0254)                                         XYC 026
      F = 1.0/RECIPF                                                    XYC 027
      ESQ = 2.0*F-F**2                                                  XYC 028
      ALPHA = 1.0-ESQ                                                   XYC 029

      B1 = RADIUS*(1.0-ESQ/4.0-3.0*ESQ**2/64.0-5.0*ESQ**3/256.0)        XYC 031
      B2 = 3.0*RADIUS*(ESQ+ESQ**2/4.0+15.0*ESQ**3/128.0)/8.0            XYC 032
      B3 = 15.0*RADIUS*(ESQ**2+3.0*ESQ**3/4.0)/256.0                    XYC 033
      B4 = 35.0*RADIUS*ESQ**3/3072.0                                    XYC 034

      RETURN

C . . . CONVERT COORDINATES TO PLOTTER UNITS.
      ENTRY  COORDTM                                                    XYC 038

      PHI = ALT/3437.7468
      W = (ALN-CM)/3437.7468
      C = COS(PHI)
      S = SIN(PHI)
      U = (W*C)**2                                                      XYC 044
      V = (S/C)**2                                                      XYC 045
      BETA = 1.0-ESQ*S**2                                               XYC 046
      RPV = RADIUS / SQRT(BETA)
      PSI = BETA/ALPHA                                                  XYC 048

      DM = B1*PHI-B2*SIN(2.0*PHI)+B3*SIN(4.0*PHI)-B4*SIN(6.0*PHI)

      X1 = 0.9996*RPV*W*C                                               XYC 052
      X2 = U*(PSI-V)/6.0                                                XYC 053
      X3 = U**2*(4.0*PSI**3*(1.0-6.0*V)+PSI**2*(1.0+8.0*V)-PSI*2.0*V    XYC 054
     1*V**2)/120.0                                                      XYC 055
      X4 = U**3*(61.0-479.0*V+179.0*V**2-V**3)/5040.0                   XYC 056
      XX = X1*(1.0+X2+X3+X4)

      Y1 = 0.9996*DM                                                    XYC 059
      Y2 = 0.5*X1*W*S                                                   XYC 060
      Y3 = U*(4.0*PSI**2+PSI-V)/12.0                                    XYC 061
      Y4 = U**2*(8.0*PSI**4*(11.0-24.0*V)-28.0*PSI**3*(1.0-6.0*V)+PSI**2 XYC 062
```

```
     1*(1.0-32.0*V)-PSI*2.0*V*V**2)/360.0                                XYC 063
      Y5 = U**3*(1385.0-3111.0*V+543.0*V**2-V**3)/20160.0                XYC 064
      YY = -Y1-Y2*(1.0+Y3*Y4+Y5)

      RETURN

C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY TMDATA                                                       XYC 069

      SIZE = KSIZE

      ENCODE(30,50,MAPTYPE)
   50 FORMAT(30HTRANSVERSE MERCATOR PROJECTION)
      IZONE = ZONE
      ENCODE (30,60,MAPZONE(1) )  IZONE
   60 FORMAT (5HZONE ,I2,19H (AUSTRALIA SERIES),4X)

      PRINT 70, SFEROID,MAPTYPE,MAPZONE
   70 FORMAT(//20X,3A10,2(//,20X,3A10))

      OFFSET = -0.4*SIZE
      CALL SYMBOL (OFFSET,-1.0*SIZE,0.07*SIZE,SFEROID,0,30)
      CALL SYMBOL (OFFSET,-1.2*SIZE,0.07*SIZE,MAPTYPE,0,30)
      CALL SYMBOL (OFFSET,-1.4*SIZE,0.07*SIZE,MAPZONE,0,30)

      RETURN

      END
      SUBROUTINE XYCONIC

C       THIS VERSION OF XYCOORDS COMPUTES EASTINGS AND NORTHINGS ON
C       A CONICAL PROJECTION WITH TWO STANDARD PARALLELS

      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,SOO
      DIMENSION SFEROID(3),MAPTYPE(3),MAPZONE(4)
      INTEGER SFEROID

C . . . ESTABLISH PROJECTION PARAMETERS.
      IF(INMAIN.NE.0)GO TO 11
      DECODE(80,10,NAIM)SFEROID,A,RECIPF,TLATD,TLATM,BLATD,BLATM,SCALE,
     1                  XSPACE,YSPACE
   10 FORMAT(3A10,2F10.0,2(F3.0,F2.0),F10.0,2F5.0)
      F=0.0
   11 IF(A.LE.0.0)F=A=6378160.0
      IF(RECIPF.LE.0.0)F=RECIPF=298.25
      IF(SCALE.LE.0.0)F=SCALE=250000.0
      IF(XSPACE*YSPACE.LE.0.0)F=XSPACE=YSPACE=K=6*SCALE/100000
      IF(XSPACE+YSPACE.LT.0.1)F=XSPACE=YSPACE=1.0
      IF(ABS(TLATD)+ABS(TLATM)+ABS(BLATD)+ABS(BLATM).NE.0.0)GO TO 13
      F=TLATD=18.0 $ BLATD=36.0
   13 IF(F.NE.0.0)PRINT 14
   14 FORMAT(1H0,9X,*MISSING SPHEROID PARAMETERS GIVEN DEFAULT VALUES*)
   15 IF(TLATD.LT.0.0)TLATM=-ABS(TLATM)
      IF(BLATD.LT.0.0)BLATM=-ABS(BLATM)

      CM=0.5*(WLN+ELN)
      RADIUS=A/(SCALE*0.0254)
      F = 1.0/RECIPF
      ESQ=2.0*F-F**2
```

```
      B1=RADIUS*(1.0-ESQ/4.0-3.0*ESQ**2/64.0-5.0*ESQ**3/256.0)
      B2=3.0*RADIUS*(ESQ+ESQ**2/4.0+15.0*ESQ**3/128.0)/8.0
      B3=15.0*RADIUS*(ESQ**2+3.0*ESQ**3/4.0)/256.0
      B4=35.0*RADIUS*ESQ**3/3072.0

      PHI1=(TLATD+TLATM/60.0)/57.2957795
      SINPHI1=SIN(PHI1)
      COSPHI1=COS(PHI1)
      BETA1=1.0-ESQ*SINPHI1**2
      RPV1 = RADIUS / SQRT(BETA1)

      PHI2=(BLATD+BLATM/60.0)/57.2957795
      SINPHI2=SIN(PHI2)
      COSPHI2=COS(PHI2)
      BETA2=1.0-ESQ*SINPHI2**2
      RPV2 = RADIUS / SQRT(BETA2)

      P1=RPV1*COSPHI1
      DM1=B1*PHI1-B2*SIN(2.0*PHI1)+B3*SIN(4.0*PHI1)-B4*SIN(6.0*PHI1)
      P2=RPV2*COSPHI2
      DM2=B1*PHI2-B2*SIN(2.0*PHI2)+B3*SIN(4.0*PHI2)-B4*SIN(6.0*PHI2)
      ORIGIN=DM1*P1-(DM2-DM1)/(P1-P2)
      THETA1=P1/(ORIGIN-DM1)

      RETURN
C
C . . . CONVERT COORDINATES TO PLOTTER UNITS.
       ENTRY COORDCO

      PHI = ALT/3437.7468
      W = (ALN-CM)/3437.7468
      DM=B1*PHI-B2*SIN(2.0*PHI)+B3*SIN(4.0*PHI)-B4*SIN(6.0*PHI)
      THETA=W*THETA1
      DY=(ORIGIN-DM)*(1.0-COS(THETA))

      XX=(ORIGIN-DM)*SIN(THETA)
      YY=-DM-DY

      RETURN

C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY CODATA

      SIZE = KSIZE
      ENCODE (30,20,MAPTYPE)
   20 FORMAT (25HSIMPLE CONICAL PROJECTION,5X)
      ENCODE (40,30,MAPZONE)TLATD,TLATM,BLATD,BLATM
   30 FORMAT (19HSTANDARD PARALLELS ,F3,F3,4HAND ,F3,F3,5HSOUTH)

      PRINT 40,SFEROID,MAPTYPE,MAPZONE
   40 FORMAT (//,20X,3A10//,20X,3A10//,20X,4A10 )

      OFFSET = -0.4*SIZE
      CALL SYMBOL (OFFSET,-1.0*SIZE,0.07*SIZE,SFEROID,0,30)
      CALL SYMBOL (OFFSET,-1.2*SIZE,0.07*SIZE,MAPTYPE,0,30)
      CALL SYMBOL (OFFSET,-1.4*SIZE,0.07*SIZE,MAPZONE,0,40)

      RETURN

      END
      SUBROUTINE XYLMBRT
```

```
C       THIS VERSION OF XYCOORDS COMPUTES EASTINGS AND NORTHINGS ON      LCCOD003
C       LAMBERTS CONFORMAL PROJECTION WITH TWO STANDARD PARALLELS        LCCOD004

      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION SFEROID(3),MAPTYPE(3),MAPZONE(4)
      INTEGER SFEROID                                                   LCCOD010

C . . . ESTABLISH PROJECTION PARAMETERS.
      IF(INMAIN.NE.0)GO TO 11
      DECODE(80,10,NAIM)SFEROID,A,RECIPF,TLATD,TLATM,BLATD,BLATM,SCALE,
     1                  XSPACE,YSPACE
   10 FORMAT(3A10,2F10.0,2(F3.0,F2.0),F10.0,2F5.0)                      LCCOD014
      F=0.0
   11 IF(A.LE.0.0)F=A=6378160.0
      IF(RECIPF.   0.0)F=RECIPF=298.25
      IF(SCALE.LE. 0.0)F=SCALE=250000.0
      IF(XSPACE*YSPACE.LE.0.0)F=XSPACE=YSPACE=K=6*SCALE/100000
      IF(XSPACE+YSPACE.LT.0.1)F=XSPACE=YSPACE=1.0
      IF(ABS(TLATD)+ABS(TLATM)+ABS(BLATD)+ABS(BLATM).NE.0.0)GO TO 13
      F=TLATD=18.0 $ BLATD=36.0
   13 IF(F.NE.0.0)PRINT 14
   14 FORMAT(1H0,9X,*MISSING SPHEROID PARAMETERS GIVEN DEFAULT VALUES*)
   15 IF(TLATD.LT.0.0)TLATM=-ABS(TLATM)
      IF(BLATD.LT.0.0)BLATM=-ABS(BLATM)

      CM = 0.5*(WLN+ELN)
      RADIUS = A/(SCALE*0.0254)                                         LCCOD020
      F = 1.0/RECIPF                                                    LCCOD021
      ESQ = 2.0*F-F**2                                                  LCCOD022
      E = SQRT(ESQ)
      ALPHA = 1.0-ESQ                                                   LCCOD024
      PHI1=(TLATD+TLATM/60.0)/57.2957795                                LCCOD026
      SINPHI1=SIN(PHI1)                                                 LCCOD027
      COSPHI1=COS(PHI1)                                                 LCCOD028
      BETA1=1.0-ESQ*SINPHI1**2                                          LCCOD029
      RPV1=RADIUS/SQRT(BETA1)                                           LCCOD030
      GAMMA1=((1.0-SINPHI1)/(1.0+SINPHI1))*                             LCCOD031
     1       ((1.0+E*SINPHI1)/(1.0-E*SINPHI1))**E                       LCCOD032
      PHI2=(BLATD+BLATM/60.0)/57.2957795                                LCCOD033

      SINPHI2=SIN(PHI2)                                                 LCCOD035
      COSPHI2=COS(PHI2)                                                 LCCOD036
      BETA2=1.0-ESQ*SINPHI2**2                                          LCCOD037
      RPV2=RADIUS/SQRT(BETA2)                                           LCCOD038
      GAMMA2= ((1.0-SINPHI2)/(1.0+SINPHI2))*                            LCCOD039
     1        ((1.0+E*SINPHI2)/(1.0-E*SINPHI2))**E                      LCCOD040

      C1=ALOG((RPV1*COSPHI1)/(RPV2*COSPHI2))/ALOG(GAMMA1/GAMMA2)
      C2 = 2.0*C1                                                       LCCOD043
      C3=(RPV1*COSPHI1)/(C2*GAMMA1**C1)                                 LCCOD044

      RETURN                                                            LCCOD050

C . . . CONVERT COORDINATES TO PLOTTER UNITS.
      ENTRY COORDLC

      PHI=ALT/3437.7468
      W=(ALN-CM)/3437.7468
      SINFPHI = SIN(PHI)
      GAMMA  = ((1.0-SINFPHI )/(1.0+SINFPHI ))*                         LCCOD058
```

```
     1         ((1.0+E*SINFPHI )/(1.0-E*SINFPHI ))**E                    LCCOD059
      R = C3*GAMMA**C1                                                   LCCOD060
      THETA = C2*W                                                       LCCOD061

      XX = R * SIN(THETA)
      YY = R * COS(THETA)

      RETURN                                                             LCCOD066

C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY LCDATA

      SIZE=KSIZE
      ENCODE (30,20,MAPTYPE)
   20 FORMAT (30HLAMBERTS CONFORMAL PROJECTION )
      ENCODE (40,30,MAPZONE)TLATD,TLATM,BLATD,BLATM
   30 FORMAT (19HSTANDARD PARALLELS ,2F3,4HAND ,2F3,5HSOUTH)
      PRINT 40,SFEROID,MAPTYPE,MAPZONE
   40 FORMAT (//,20X,3A10,//,20X,3A10,//,20X,4A10)

      OFFSET = -0.4*SIZE
      CALL SYMBOL (OFFSET,-1.0*SIZE,0.07*SIZE,SFEROID,0,30)
      CALL SYMBOL (OFFSET,-1.2*SIZE,0.07*SIZE,MAPTYPE,0,30)
      CALL SYMBOL (OFFSET,-1.4*SIZE,0.07*SIZE,MAPZONE,0,40)

      RETURN                                                             LCCOD072

      END                                                                LCCOD074
      SUBROUTINE XYPLANE

C       THIS VERSION OF XYCOORDS COMPUTES EASTINGS AND NORTHINGS ON
C       A RECTANGULAR PROJECTION SPECIFIED IN PLOTTER UNITS / DEGREE.

      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION SFEROID(3),MAPTYPE(3),MAPZONE(4),UNIT(4),FACTOR(4)
      INTEGER SFEROID,XUNITS,YUNITS,UNIT
      DATA (UNIT=4H MM ,4H CM ,4H IN ,4H FT )
      DATA (FACTOR=25.3998,2.53998,1.00,0.083333)

C . . . ESTABLISH PROJECTION PARAMETERS.
      IF(INMAIN.NE.0)GO TO 11
      DECODE (80,10,NAIM) SFEROID,XSCALE,XUNITS,YSCALE,YUNITS,XSPACE,
     1                    YSPACE
   10 FORMAT (3A10  ,2(F10.2,A4,6X),2F5.2)
   11 IF(XSCALE.EQ.0.0)XSCALE=YSCALE=2.0
      IF(YSCALE.EQ.0.0)YSCALE=XSCALE
      IF(YUNITS.EQ.1H )YUNITS=XUNITS
   15 SCALE=YSCALE

         DO 20 J=1,4
         IF(XUNITS.EQ.UNIT(J))GO TO 30
   20    CONTINUE $ J=3 $ XUNITS=UNIT(3)
   30 XSCALE=XSCALE/(FACTOR(J)*60.0)

         DO 40 J=1,4
         IF(YUNITS.EQ.UNIT(J))GO TO 50
   40    CONTINUE $ J=3 $ YUNITS=UNIT(3)
   50 YSCALE=YSCALE/(FACTOR(J)*60.0)

      RETURN
```

```
C . . . CONVERT COORDINATES TO PLOTTER UNITS
      ENTRY COORDXY

      XX=ALT*XSCALE
      YY=ALN*YSCALE

      RETURN

C . . . ANNOTATE MAP WITH PROJECTION DESCRIPTION.
      ENTRY DATAXY

      SIZE = KSIZE  $  XYRATIO = XSCALE / YSCALE

      ENCODE (30,60,MAPTYPE) SCALE,UNITS
   60 FORMAT (16HLATITUDE SCALE =,F7.2,A3,4H/DEG)
      ENCODE (32,70,MAPZONE) XYRATIO
   70 FORMAT (26HLONGITUDE/LATITUDE RATIO =,F6.2)
      PRINT 80,SFEROID,MAPTYPE,MAPZONE
   80 FORMAT (//20X,3A10  //20X,3A10//20X,4A10)

      OFFSET = -0.4*SIZE  $  SCALE = 0.0
      CALL SYMBOL (OFFSET,-1.0*SIZE,0.07*SIZE,SFEROID,0,30)
      CALL SYMBOL (OFFSET,-1.2*SIZE,0.07*SIZE,MAPTYPE,0,30)
      CALL SYMBOL (OFFSET,-1.4*SIZE,0.07*SIZE,MAPZONE,0,32)

      RETURN

      END
      SUBROUTINE AXIS(XPAGE,YPAGE,IBCD,NCHAR,AXLEN,ANGLE,FIRSTV,DELTAV) AXIS0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972          AXIS0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                      AXIS0003
C.....     XPAGE,YPAGE  COORDINATES OF STARTING POINT OF AXIS, IN INCHESAXIS0004
C.....     IBCD         AXIS TITLE.                                    AXIS0005
C.....     NCHAR        NUMBER OF CHARACTERS IN TITLE. + FOR C.C-W SIDE.AXIS0006
C.....     AXLEN        FLOATING POINT AXIS LENGTH IN INCHES.          AXIS0007
C.....     ANGLE        ANGLE OF AXIS FROM THE X-DIRECTION, IN DEGREES. AXIS0008
C.....     FIRSTV       SCALE VALUE AT THE FIRST TIC MARK.             AXIS0009
C.....     DELTAV       CHANGE IN SCALE BETWEEN TIC MARKS ONE INCH APARTAXIS0010
      DIMENSION IBCD(2)                                                AXIS0011
      KN=NCHAR                                                         AXIS0012
      A=1.0                                                            AXIS0013
      IF (KN) 1,2,2                                                    AXIS0014
   1  A=-A                                                             AXIS0015
      KN=-KN                                                           AXIS0016
   2  EX=0.0                                                           AXIS0017
      ADX= ABS  (DELTAV)                                               AXIS0018
      IF (ADX) 3,7,3                                                   AXIS0019
   3  IF (ADX- 99.0) 6,4,4                                             AXIS0020
   4  ADX=ADX/10.0                                                     AXIS0021
      EX=EX+1.0                                                        AXIS0022
      GO TO 3                                                          AXIS0023
   5  ADX=ADX*10.0                                                     AXIS0024
      EX=EX-1.0                                                        AXIS0025
   6  IF (ADX-0.01) 5,7,7                                              AXIS0026
   7  XVAL=FIRSTV*10.0**(-EX)                                          AXIS0027
      ADX= DELTAV*10.0**(-EX)                                          AXIS0028
      STH=ANGLE*0.0174533                                              AXIS0029
      CTH=COS(STH)                                                     AXIS0030
      STH=SIN(STH)                                                     AXIS0031
      DXB=-0.1                                                         AXIS0032
      DYB=0.15*A-0.05                                                  AXIS0033
```

```
      XN=XPAGE+DXB*CTH-DYB*STH                                          AXIS0034
      YN=YPAGE+DYB*CTH+DXB*STH                                          AXIS0035
      NTIC=AXLEN+1.0                                                    AXIS0036
      NT=NTIC/2                                                         AXIS0037
      DO  20  I=1,NTIC                                                  AXIS0038
        CALL NUMBER(XN,YN,0.105,XVAL,ANGLE,2)                           AXIS0039
        XVAL=XVAL+ADX                                                   AXIS0040
        XN=XN+CTH                                                       AXIS0041
        YN=YN+STH                                                       AXIS0042
        IF (NT) 20,11,20                                                AXIS0043
11      Z=KN                                                            AXIS0044
        IF (EX)  12,13,12                                               AXIS0045
12      Z=Z+7.0                                                         AXIS0046
13      DXB=-.07*Z+AXLEN*0.5                                            AXIS0047
        DYB=0.325*A-0.075                                               AXIS0048
        XT=XPAGE+DXB*CTH-DYB*STH                                        AXIS0049
        YT=YPAGE+DYB*CTH+DXB*STH                                        AXIS0050
        CALL SYMBOL(XT,YT,0.14,IBCD(1),ANGLE,KN)                        AXIS0051
        IF (EX)  14,20,14                                               AXIS0052
14        Z=KN+2                                                        AXIS0053
          XT=XT+Z*CTH*0.14                                              AXIS0054
          YT=YT+Z*STH*0.14                                              AXIS0055
          CALL SYMBOL(XT,YT,0.14,3H*10,ANGLE,3)                         AXIS0056
          XT=XT+(3.0*CTH-0.8*STH)*0.14                                  AXIS0057
          YT=YT+(3.0*STH+0.8*CTH)*0.14                                  AXIS0058
          CALL NUMBER(XT,YT,0.07,EX,ANGLE,-1)                           AXIS0059
20    NT=NT-1                                                           AXIS0060
      CALL PLOT(XPAGE+AXLEN*CTH,YPAGE+AXLEN*STH,3)                      AXIS0061
      DXB=-0.07*A*STH                                                   AXIS0062
      DYB=+0.07*A*CTH                                                   AXIS0063
      A=NTIC-1                                                          AXIS0064
      XN=XPAGE+A*CTH                                                    AXIS0065
      YN=YPAGE+A*STH                                                    AXIS0066
      DO  30  I=1,NTIC                                                  AXIS0067
        CALL PLOT(XN,YN,2)                                              AXIS0068
        CALL PLOT(XN+DXB,YN+DYB,2)                                      AXIS0069
        CALL PLOT(XN,YN,2)                                              AXIS0070
        XN=XN-CTH                                                       AXIS0071
        YN=YN-STH                                                       AXIS0072
30    CONTINUE                                                          AXIS0073
      RETURN                                                            AXIS0074
      END                                                               AXIS0075
      SUBROUTINE LINE (XARRAY,YARRAY,NPTS,INC,LINTYP,INTEQ)             LINE0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 3000 SERIES  OCT.,1972          LINE0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                      LINE0003
C.....     XARRAY  NAME OF ARRAY CONTAINING ABSCISSA OR X VALUES.       LINE0004
C.....     YARRAY  NAME OF ARRAY CONTAINING ORDINATE OR Y VALUES.       LINE0005
C.....     NPTS    NUMBER OF POINTS TO BE PLOTTED.                      LINE0006
C.....     INC     INCREMENT OF LOCATION OF SUCCESSIVE POINTS.          LINE0007
C.....     LINTYP  CONTROL TYPE OF LINE--SYMBOLS, LINE, OR COMBINATION. LINE0008
C.....     INTEQ   INTEGER EQUIVALENT OF SYMBOL TO BE USED, IF ANY.     LINE0009
      DIMENSION XARRAY(1),YARRAY(1)                                     LINE0010
      LMIN = NPTS*INC+1                                                 LINE0011
      LDX  = LMIN+INC                                                   LINE0012
      NL   = LMIN-INC                                                   LINE0013
      FIRSTX = XARRAY(LMIN)                                             LINE0014
      DELTAX = XARRAY(LDX)                                              LINE0015
      FIRSTY = YARRAY(LMIN)                                             LINE0016
      DELTAY = YARRAY(LDX)                                              LINE0017
      CALL WHERE (XN,YN,DF)                                             LINE0018
      DF=AMAX1(ABS((XARRAY( 1)-FIRSTX)/DELTAX-XN),                      LINE0019
     1         ABS((YARRAY( 1)-FIRSTY)/DELTAY-YN) )                     LINE0020
```

```
      DL=AMAX1(ABS((XARRAY(NL)-FIRSTX)/DELTAX-XN),                      LINE0021
     1         ABS((YARRAY(NL)-FIRSTY)/DELTAY-YN))                      LINE0022
       IPEN = 3                                                         LINE0023
       ICODE = -1                                                       LINE0024
       NT =IABS(LINTYP)                                                 LINE0025
       IF (LINTYP) 7,6,7                                                LINE0026
     6 NT = 1                                                           LINE0027
     7 IF (DF-DL) 9,9,8                                                 LINE0028
     8 NF = NL                                                          LINE0029
       NA = ((NPTS-1)/NT)*NT+NT-(NPTS-1)                                LINE0030
       KK = -INC                                                        LINE0031
       GO TO 10                                                         LINE0032
     9 NF = 1                                                           LINE0033
       NA = NT                                                          LINE0034
       KK = INC                                                         LINE0035
    10 IF (LINTYP) 11,12,13                                             LINE0036
    11 IPENA = 3                                                        LINE0037
       ICODEA = -1                                                      LINE0038
       LSW = 1                                                          LINE0039
       GO TO 15                                                         LINE0040
    12 NA=LDX                                                           LINE0041
    13 IPENA = 2                                                        LINE0042
       ICODEA = -2                                                      LINE0043
       LSW=0                                                            LINE0044
    15 DO 30 I =1,NPTS                                                  LINE0045
       XN = (XARRAY(NF)-FIRSTX)/DELTAX                                  LINE0046
       YN = (YARRAY(NF)-FIRSTY)/DELTAY                                  LINE0047
       IF (NA-NT) 20,21,22                                              LINE0048
    20 IF (LSW) 23,22,23                                                LINE0049
    21 CALL SYMBOL (XN,YN,0.08,INTEQ,0.0,ICODE)                         LINE0050
       NA = 1                                                           LINE0051
       GO TO 25                                                         LINE0052
    22 CALL PLOT (XN,YN,IPEN)                                           LINE0053
    23 NA = NA + 1                                                      LINE0054
    25 NF = NF+KK                                                       LINE0055
       ICODE = ICODEA                                                   LINE0056
    30 IPEN = IPENA                                                     LINE0057
       RETURN                                                           LINE0058
       END                                                              LINE0059
       SUBROUTINE NUMBER (XPAGE,YPAGE,HEIGHT,FPN,ANGLE,NDEC)            NUMB0001
CA_COMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972          NUMB0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                      NUMB0003
C.....     XPAGE,YPAGE COORDINATES OF LOWER LEFT CORNER OF NUMBER.      NUMB0004
C.....     HEIGHT   HEIGHT OF PLOTTED NUMBER.                           NUMB0005
C.....     FPN      FLOATING POINT NUMBER TO BE PLOTTED.                NUMB0006
C.....     ANGLE    ANGLE AT WHICH NUMBER IS PLOTTED, IN DEGREES.       NUMB0007
C.....     NDEC     NUMBER OF DECIMAL PLACES TO BE DRAWN.               NUMB0008
C.....     THIS VERSION OF NUMBER REQUIRES THE SYMBOL VERSION WITH      NUMB0009
C.....     999. X, Y FEATURE, AND  NC = 0 FEATURE.                     NUMB0010
       DIMENSION NUM(20)                                                NUMB0011
       DATA MINUS/38/,IPER/47/,IZERO/27/                                NUMB0012
       II=0                                                             NUMB0013
       FPV = FPN                                                        NUMB0014
       N = NDEC                                                         NUMB0015
       MAXN = 9                                                         NUMB0016
       IF (N - MAXN) 11, 11, 10                                         NUMB0017
    10 N = MAXN                                                         NUMB0018
    11 IF (N + MAXN) 12, 20, 20                                         NUMB0019
    12 N = -MAXN                                                        NUMB0020
    20 IF (FPV) 21, 30, 30                                              NUMB0021
21     II=II+1                                                          NUMB0022
       NUM(II)=  MUS                                                    NUMB0023
```

```
   30 MN = -N                                                          NUMB0024
      IF (N) 31, 32, 32                                                NUMB0025
   31 MN = MN - 1                                                      NUMB0026
   32 FPV = ABS(FPV) + (0.5 * 10. ** MN)                               NUMB0027
      I = ALOG10(FPV) + 1.0                                            NUMB0028
      ILP = I                                                          NUMB0029
      IF (N + 1)  40, 41, 41                                           NUMB0030
   40 ILP = ILP + N + 1                                                NUMB0031
   41 IF (ILP)  50, 50, 51                                             NUMB0032
50    II=II+1                                                          NUMB0033
      NUM(II)=IZERO                                                    NUMB0034
      GO TO 61                                                         NUMB0035
51    IF (ILP+N-18) 54,54,52                                           NUMB0036
52    N=-1                                                             NUMB0037
      IF (ILP-19) 54,54,53                                             NUMB0038
53    ILP=19                                                           NUMB0039
54    DO 60 J=1,ILP                                                    NUMB0040
        K = FPV * 10. ** (J - I)                                       NUMB0041
      II=II+1                                                          NUMB0042
      NUM(II)=K+IZERO                                                  NUMB0043
        FPV = FPV - (FLOAT(K) * 10. ** (I - J))                        NUMB0044
60    CONTINUE                                                         NUMB0045
   61 IF (N) 99, 70, 70                                                NUMB0046
70    II=II+1                                                          NUMB0047
      NUM(II)=IPER                                                     NUMB0048
      IF (N)  99, 99, 80                                               NUMB0049
   80 DO 90 J = 1, N                                                   NUMB0050
        K = FPV * 10.                                                  NUMB0051
      II=II+1                                                          NUMB0052
      NUM(II)=K+IZERO                                                  NUMB0053
   90   FPV = FPV * 10. - FLOAT(K)                                     NUMB0054
99    CALL SYMBOL (XPAGE,YPAGE,HEIGHT,NUM,ANGLE,II+1000)               NUMB0055
      RETURN                                                           NUMB0056
      END                                                              NUMB0057
      SUBROUTINE  SCALE   (ARRAY,AXLEN,NPTS,INC)                       SCAL0001
CALCOMP HCRS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972         SCAL0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                     SCAL0003
C.....     ARRAY   NAME OF ARRAY CONTAINING VALUES TO BE SCALED.       SCAL0004
C.....     AXLEN   LENGTH IN INCHES OVER WHICH ARRAY IS TO BE SCALED.  SCAL0005
C.....     NPTS    NUMBER OF POINTS TO BE SCALED.                      SCAL0006
C.....     INC     INCREMENT OF LOCATION OF SUCCESSIVE POINTS.         SCAL0007
      DIMENSION  ARRAY(1),SAVE(7)                                      SCAL0008
      SAVE(1)=1.0                                                      SCAL0009
      SAVE(2)=2.0                                                      SCAL0010
      SAVE(3)=4.0                                                      SCAL0011
      SAVE(4)=5.0                                                      SCAL0012
      SAVE(5)=8.0                                                      SCAL0013
      SAVE(6)=10.0                                                     SCAL0014
      SAVE(7)=20.                                                      SCAL0015
      FAD=0.01                                                         SCAL0016
      K=IABS(INC)                                                      SCAL0017
      N=NPTS*K                                                         SCAL0018
      YO=ARRAY(1)                                                      SCAL0019
      YN=YO                                                            SCAL0020
      DO  25  I=1,N,K                                                  SCAL0021
      YS=ARRAY(I)                                                      SCAL0022
      IF  (YO-YS)  22,22,21                                            SCAL0023
21    YO=YS                                                            SCAL0024
      GO  TO  25                                                       SCAL0025
22    IF  (YS-YN)  25,25,24                                            SCAL0026
24    YN=YS                                                            SCAL0027
25    CONTINUE                                                         SCAL0028
```

```
      FIRSTV=Y0                                                          SCAL0029
      IF  (Y0)  34,35,35                                                 SCAL0030
34    FAD=FAD-1.0                                                        SCAL0031
35    DELTAV=(YN-FIRSTV)/AXLEN                                           SCAL0032
      IF (DELTAV) 70,70,40                                               SCAL0033
40    I=ALOG10(DELTAV)*1000.0                                            SCAL0034
      P=10.0**(I-1000)                                                   SCAL0035
      DELTAV=DELTAV/P-0.01                                               SCAL0036
      DO  45  I=1,6                                                      SCAL0037
      IS=I                                                               SCAL0038
      IF  (SAVE(I)-DELTAV)  45,50,50                                     SCAL0039
45    CONTINUE                                                           SCAL0040
50    DELTAV=SAVE(IS)*P                                                  SCAL0041
      FIRSTV=DELTAV*AINT(Y0/DELTAV+FAD)                                  SCAL0042
      T=FIRSTV+(AXLEN+0.01)*DELTAV                                       SCAL0043
      IF (T-YN)  55,57,57                                                SCAL0044
 55   FIRSTV=P*AINT(Y0/P+FAD)                                            SCAL0045
      T=FIRSTV+(AXLEN+.01)*DELTAV                                        SCAL0046
      IF (T-YN) 56,57,57                                                 SCAL0047
 56   IS=IS+1                                                            SCAL0048
      GO  TO  50                                                         SCAL0049
57    FIRSTV=FIRSTV-AINT((AXLEN+(FIRSTV-YN)/DELTAV)/2.0)*DELTAV          SCAL0050
      IF (Y0*FIRSTV) 58,58,59                                            SCAL0051
58    FIRSTV=0.0                                                         SCAL0052
59    IF  (INC) 61,61,65                                                 SCAL0053
 61   FIRSTV=FIRSTV+AINT(AXLEN+.5)*DELTAV                                SCAL0054
      DELTAV=-DELTAV                                                     SCAL0055
65    N=N+1                                                              SCAL0056
      ARRAY(N)=FIRSTV                                                    SCAL0057
      N=N+K                                                              SCAL0058
      ARRAY(N)=DELTAV                                                    SCAL0059
67    RETURN                                                             SCAL0060
 70   DELTAV=2.0*FIRSTV                                                  SCAL0061
      DELTAV=ABS(DELTAV/AXLEN)+1.                                        SCAL0062
      GO TO 40                                                           SCAL0063
      END                                                                SCAL0064
      SUBROUTINE SYMBOL (XPAGE,YPAGE,HEIGH,IBCD,ANGLE,NCHAR)             SYMB0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972           SYMB0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                       SYMB0003
      DIMENSION IBCD(1)                                                  SYMB0004
      DATA OLDTH,OLDFT/2*999.0/,XO,YO,XNC/3*0.0/,ICENT/15/               SYMB0005
C                                                                        SYMB0006
C        ((2**NBC-1)/NCW)*NCW     NCS/NCW                               SYMB0007
C              V                    V                                   SYMB0008
      DATA NCS/60/             ,NWS/6/                                   SYMB0009
   20 NCNT=0                                                             SYMB0010
      IPEN=3                                                             SYMB0011
      NT=NCHAR                                                           SYMB0012
      NC=1                                                               SYMB0013
      II=1                                                               SYMB0014
      IF(ABS(XPAGE-999.0)-0.1) 106,105,105                               SYMB0015
105   XO=XPAGE                                                           SYMB0016
      GO TO 110                                                          SYMB0017
106   XO=XO+XNC*XA*.875                                                  SYMB0018
 110  IF(ABS(YPAGE-999.0)-0.1) 116,115,115                               SYMB0019
115   YO=YPAGE                                                           SYMB0020
      GO TO 120                                                          SYMB0021
116   YO=YO+XNC*YA*.875                                                  SYMB0022
  120 IF (NT) 200,400,300                                                SYMB0023
  200 IF (NT+1) 210,220,220                                              SYMB0024
  210 IPEN=2                                                             SYMB0025
  220 IF (IBCD(1)-ICENT) 230,230,400                                     SYMB0026
```

```
  230 FCT=HEIGH+HEIGH                                                    SYMB0027
      GO TO 410                                                          SYMB0028
  300 NC=MOD(NT,1000)                                                    SYMB0029
C                          1.1428571=8.0/7.0                           SYMB0030
  400 FCT=HEIGH*1.1428571                                                SYMB0031
410   CALL PLOT (XO,YO,IPEN)                                           SYMB0032
      XNC=NC                                                           SYMB0033
      IF (ANGLE-OLDTH) 420,415,420                                     SYMB0034
  415 IF (FCT-OLDFT)425,417,425                                        SYMB0035
  417 IC=1004                                                          SYMB0036
      GO TO 430                                                        SYMB0037
420   OLDTH=ANGLE                                                      SYMB0038
      TH1=OLDTH*0.017453                                               SYMB0039
      SINTH=SIN(TH1)                                                   SYMB0040
      COSTH=COS(TH1)                                                   SYMB0041
425   OLDFT=FCT                                                        SYMB0042
      XA=FCT*COSTH                                                     SYMB0043
      YA=FCT*SINTH                                                     SYMB0044
      IC=1003                                                          SYMB0045
  430 IF (NC-NCS) 445,445,440                                          SYMB0046
  440 NC=NCS                                                           SYMB0047
  445 ICNT=NC+8                                                        SYMB0048
      CALL BUFF(ICNT,ICNT,-1)                                          SYMB0049
      CALL PLOT(XA,YA,IC)                                              SYMB0050
      IF (NT) 500,450,470                                              SYMB0051
  450 NCNT=-1                                                          SYMB0052
      GO TO 500                                                        SYMB0053
  470 NCNT=NC                                                          SYMB0054
      IF (NT-1000) 500,450,480                                         SYMB0055
  480 NCNT=-NC                                                         SYMB0056
  500 CALL BUFF (IBCD(II),NCNT,1)                                      SYMB0057
      NT=NT-NC                                                         SYMB0058
      IF (NT) 600,600,510                                              SYMB0059
  510 IF(NT-1000)530,600.520                                           SYMB0060
  520 II=II+NCS                                                        SYMB0061
      NC=NT-1000                                                       SYMB0062
      GO TO 430                                                        SYMB0063
  530 II=II+NWS                                                        SYMB0064
      NC=NT                                                            SYMB0065
      GO TO 430                                                        SYMB0066
  600 RETURN                                                           SYMB0067
      END                                                              SYMB0068
      SUBROUTINE FACTOR(FACT)                                          FACT0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972         FACT0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                     FACT0003
      CALL PLOT  (FACT,FACT,1000)                                      FACT0004
      RETURN                                                           FACT0005
      END                                                              FACT0006
      SUBROUTINE NEWPEN(INP)                                           NEWP0000
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972         NEWP0001
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                     NEWP0002
C     THIS SUBROUTINE SELECTS A PEN ON 718/7180 PLOTTERS.              NEWP0003
      DATA IOLDPN/1/                                                   NEWP0004
      INP1 = INP                                                       NEWP0005
      IF (INP - IOLDPN) 15,99,15                                       NEWP0006
 15   INP1 = MOD(INP1,9)                                               NEWP0007
      IF (INP1) 20,99,25                                               NEWP0008
 20   INP1 = -INP1                                                     NEWP0009
 25   IOLDPN = INP1                                                    NEWP0010
      CALL BUFF(0,2,-1)                                                NEWP0011
      CALL BUFF(4,1,0)                                                 NEWP0012
      CALL BUFF(INP1,1,0)                                              NEWP0013
```

```
      CALL PLOT(0.0,0.0,1005)                                                  NEWP0014
 99   RETURN                                                                   NEWP0015
      END                                                                      NEWP0016
      SUBROUTINE WHERE(XPAGE,YPAGE,RFACT)                                      WHER0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972                 WHER0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                             WHER0003
      CALL PLOT(XPAGE,YPAGE,1001)                                              WHER0004
      CALL PLOT(RFACT,RFACT,1002)                                              WHER0005
      RETURN                                                                   WHER0006
      END                                                                      WHER0007
      SUBROUTINE PLOTS (I,J,LDEV)                                              PLTS0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972                 PLTS0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                             PLTS0003
      IDENT=1110                                                               PLTS0004
      IDENT2=9901                                                              PLTS0005
      CALL BUFF (LDEV,1,-2)                                                    PLTS0006
      CALL BUFF (IDENT,3,0)                                                    PLTS0007
      CALL BUFF(IDENT2,3,0)                                                    PLTS0008
      CALL BUFF (1000,1000,-1)                                                 PLTS0009
      CALL BUFF (1,1,0)                                                        PLTS0010
      CALL BUFF (1,3,0)                                                        PLTS0011
      CALL BUFF (257,2,0)                                                      PLTS0012
      RETURN                                                                   PLTS0013
      END                                                                      PLTS0014
      SUBROUTINE PLOT(XPAGE,YPAGE,IPEN)                                        PLOT0001
CALCOMP HCBS  PN 711072/911072  FOR CDC 6000 SERIES  OCT.,1972                 PLOT0002
COPYRIGHT  1972  CALIFORNIA COMPUTER PRODUCTS,INC.                             PLOT0003
C***IS(1) SHOULD BE 2**(NBC-1)-1 WHERE NBC IS THE NO. OF BITS/CHARACTER.       PLOT0004
C***IS(3) WILL BE 2**(NBA-1)-1 WHERE NBA IS THE NO. OF BITS/TWO WORDS          PLOT0005
C***IN 900.                                                                    PLOT0006
C***  SIGN(X,Y) = X*(Y/ABS(Y))                                                 PLOT0007
      DIMENSION IS(3)                                                          PLOT0008
      DATA IS(1),IS(2),IS(3)/31,2047,131071/                                   PLOT0009
      DATA MAX/500/,IOLDX,IOLDY,IOPEN,ICFLG/4*0/,MAXDX/32768/                  PLOT0010
      DATA IBLK,IFFLAG/2*1/,FACT/1.0/,STEPS/10000./,STEPS1/10000./             PLOT0011
      DATA ISFLG,MDX,MDY/3*0/                                                  PLOT0012
20    INEWP=IPEN                                                               PLOT0013
      ICOD1=16                                                                 PLOT0014
      ICOD=2                                                                   PLOT0015
      IF(INEWP-IOPEN) 25,40,25                                                 PLOT0016
  25  IF(INEWP-2) 27,37,31                                                     PLOT0017
  27  IF(INEWP) 29,40,40                                                       PLOT0018
  29  ISFLG=-1                                                                 PLOT0019
      INEWP=-INEWP                                                             PLOT0020
      GO TO 25                                                                 PLOT0021
  31  IF(INEWP-999) 35,33,1000                                                 PLOT0022
  33  ICFLG=-1                                                                 PLOT0023
      IBLK=9998                                                                PLOT0024
      ISFLG=-1                                                                 PLOT0025
  35  IF(IOPEN-3) 36,40,36                                                     PLOT0026
  36  INEWP=3                                                                  PLOT0027
      ICOD=3                                                                   PLOT0028
  37  CALL BUFF(ICOD,1,0)                                                      PLOT0029
      IOPEN=INEWP                                                              PLOT0030
40    IX=XPAGE*STEPS+SIGN(0.5,XPAGE)                                           PLOT0031
      IDX=IX-IOLDX                                                             PLOT0032
      IADX=IABS(IDX)                                                           PLOT0033
      IOLDX=IX                                                                 PLOT0034
      IY=YPAGE*STEPS+SIGN(0.5,YPAGE)                                           PLOT0035
      IDY=IY-IOLDY                                                             PLOT0036
      IADY=IABS(IDY)                                                           PLOT0037
      IOLDY=IY                                                                 PLOT0038
```

```
 49  IF(IADX-IADY) 50,51,51                                          PLOT0039
 50  NDIV=IADY/MAXDX                                                 PLOT0040
     GO TO 52                                                        PLOT0041
 51  NDIV=IADX/MAXDX                                                 PLOT0042
 52  IF(NDIV) 44,44,152                                              PLOT0043
152  NA=NDIV                                                         PLOT0044
     NT=NA+NA                                                        PLOT0045
     NB=NA                                                           PLOT0046
     NDIV=NB+1                                                       PLOT0047
     ISX=ISIGN(1,IDX)                                                PLOT0048
     ISY=ISIGN(1,IDY)                                                PLOT0049
     IDX=IDX/NDIV                                                    PLOT0050
     IDY=IDY/NDIV                                                    PLOT0051
     NR=2*(IADX-IABS(IDX)*NDIV)                                      PLOT0052
     NS=2*(IADY-IABS(IDY)*NDIV)                                      PLOT0053
     IMFLG=0                                                         PLOT0054
 43  IADX=IABS(IDX+MDX)                                              PLOT0055
     IADY=IABS(IDY+MDY)                                              PLOT0056
 44  IXCNT=0                                                         PLOT0057
     IYCNT=0                                                         PLOT0058
     IF(IADX) 62,62,61                                               PLOT0059
 61  IXCNT=IXCNT+1                                                   PLOT0060
     IF(IADX-IS(IXCNT)) 62,161,61                                    PLOT0061
161  IMFLG=1                                                         PLOT0062
 62  IF(IADY) 64,64,63                                               PLOT0063
 63  IYCNT=IYCNT+1                                                   PLOT0064
     IF(IADY-IS(IYCNT)) 64,163,63                                    PLOT0065
163  IMFLG=1                                                         PLOT0066
 64  ICOD=IXCNT+IYCNT+1                                              PLOT0067
     CALL BUFF(ICOD,ICOD,-1)                                         PLOT0068
     ICOD=IXCNT*4+IYCNT+ICOD1                                        PLOT0069
     IF (ICOD-16) 66,65,66                                           PLOT0070
 65  IF (ISFLG) 66,67,66                                             PLOT0071
 66  CALL BUFF(ICOD,1,0)                                             PLOT0072
     IT=IDX+MDX                                                      PLOT0073
     IF (IT) 160,170,160                                             PLOT0074
160  CALL BUFF (IT,IXCNT,0)                                          PLOT0075
170  IT=IDY+MDY                                                      PLOT0076
     IF (IT) 200,190,200                                             PLOT0077
200  CALL BUFF (IT,IYCNT,0)                                          PLOT0078
190  CONTINUE                                                        PLOT0079
 67  MDX=0                                                           PLOT0080
     MDY=0                                                           PLOT0081
     NDIV=NDIV-1                                                     PLOT0082
     IF (NDIV) 70,70,53                                              PLOT0083
 53  NA=NA-NR                                                        PLOT0084
     NB=NB-NS                                                        PLOT0085
     IF (NA) 54,55,55                                                PLOT0086
 54  MDX=ISX                                                         PLOT0087
     NA=NA+NT                                                        PLOT0088
 55  IF (NB) 56,57,57                                                PLOT0089
 56  MDY=ISY                                                         PLOT0090
     NB=NB+NT                                                        PLOT0091
 57  IF(IMFLG) 64,64,43                                              PLOT0092
 70  IF (ISFLG) 75,999,71                                            PLOT0093
 71  IF (IPEN-1000) 998,999,999                                      PLOT0094
 75  CALL BUFF(MAX,MAX,-1)                                           PLOT0095
     ISFLG=0                                                         PLOT0096
     IOLDX=0                                                         PLOT0097
     IOLDY=0                                                         PLOT0098
     IOPEN=0                                                         PLOT0099
     IBLK=IBLK+1                                                     PLOT0100
```

```
      CALL BUFF(1,1,0)                                                  PLOT0101
      CALL BUFF(IBLK,3,0)                                               PLOT0102
      IFFLG=1                                                           PLOT0103
      IF(ICFLG) 80,999,999                                              PLOT0104
80    CALL BUFF(MAX,MAX,-1)                                             PLOT0105
      ICFLG=0                                                           PLOT0106
 998  ISFLG=0                                                           PLOT0107
999   RETURN                                                            PLOT0108
1000  IF(IPEN-1001) 1010,1020,1030                                      PLOT0109
1010  FACT=XPAGE                                                        PLOT0110
      STEPS=STEPS1*FACT                                                 PLOT0111
      IFFLG=1                                                           PLOT0112
      RETURN                                                            PLOT0113
1020  XPAGE=FLOAT(IOLDX)/STEPS                                          PLOT0114
      YPAGE=FLOAT(IOLDY)/STEPS                                          PLOT0115
      RETURN                                                            PLOT0116
1030  IF (IPEN-1003) 1032,1040,1050                                     PLOT0117
1032  XPAGE=FACT                                                        PLOT0118
      RETURN                                                            PLOT0119
1040  IDX=XPAGE*STEPS+SIGN(0.5,XPAGE)                                   PLOT0120
      IDY=YPAGE*STEPS+SIGN(0.5,YPAGE)                                   PLOT0121
      XPAGE=FLOAT(IDX)/STEPS                                            PLOT0122
      YPAGE=FLOAT(IDY)/STEPS                                            PLOT0123
      IADX=IABS(IDX)                                                    PLOT0124
      IADY=IABS(IDY)                                                    PLOT0125
      NDIV=1                                                            PLOT0126
      IFFLG=0                                                           PLOT0127
      ICOD1=32                                                          PLOT0128
      ISFLG=1                                                           PLOT0129
      IF(3*MAX0(IADX,IADY)-MAXDX) 44,44,1051                            PLOT0130
 1050 IF(IPEN-1005) 1051,1060,1060                                      PLOT0131
 1060 IOPEN=0                                                           PLOT0132
      RETURN                                                            PLOT0133
 1051 IF(IFFLG) 1040,1052,1040                                          PLOT0134
 1052 IDX=0                                                             PLOT0135
      IDY=0                                                             PLOT0136
      IADX=0                                                            PLOT0137
      IADY=0                                                            PLOT0138
      IXCNT=0                                                           PLOT0139
      IYCNT=0                                                           PLOT0140
      ICOD1=32                                                          PLOT0141
      ISFLG=1                                                           PLOT0142
      NDIV=1                                                            PLOT0143
      GO TO 64                                                          PLOT0144
      END                                                               PLOT0145
            IDENT  BUFF                                                 BUFF0001
*   CALCOMP HCBS PN 756012/956012  FOR CDC 7000 SERIES                  BUFF0002
*   COPYRIGHT  1973  CALIFORNIA COMPUTER PRODUCTS,INC.                  BUFF0003
            ENTRY BUFF                                                  BUFF0004
            ENTRY    CLOSE
 BUFF       DATA 0                                                      BUFF0005
            SX6  A0                                                     BUFF0006
            SA6  TEMPA0          SAVE A0                                BUFF0007
            SA2  A1              PICK UP ADDR. OF ARG 1                 BUFF0008
            SB1  X2              SAVE IT                                BUFF0009
            SA2  A1+1            PICK UP ADDR. OF ARG 2                 BUFF0010
            SB2  X2              SAVE IT                                BUFF0011
            SA2  A1+2            PICK UP ADDR. OF ARG 3                 BUFF0012
            SB3  X2              SAVE IT                                BUFF0013
            SA1  ILOC2           INITIALIZE PART WORD(ILOC)             BUFF0014
            BX7  X1                                                     BUFF0015
            SA1  IBUF1                                                  BUFF0016
```

```
         SA0   X1                                                    BUFF0017
         SA1   B1              GET LOC(1) IN X1                      BUFF0018
         SA2   B2              GET NCNT IN X2                        BUFF0019
         SA3   B3              GET ICNT IN X3                        BUFF0020
         SB4   X2                                                    BUFF0021
         PL    X2,BF40                                               BUFF0022
         SB4   B0-B4                                                 BUFF0023
BF40     SX4   B4              X4=JCNT=ABS(NCNT)                     BUFF0024
         SB4   X3+2                                                  BUFF0025
         LT    B4,B0,RSTOR     IF ICNT.LE.-3,RETURN                  BUFF0026
         NE    B4,B0,BF80      IS ICNT.EQ.-2                         BUFF0027
BF60     RJ    TAPNAM          YES.INITIALIZE TAPE WRITE SETUP       BUFF0028
         JP    RSTOR           RETURN                                BUFF0029
BF80     SB4   X3+1            NO                                    BUFF0030
         NE    B4,B0,BF200     IS ICNT.EQ.-1                         BUFF0031
BF100    SA3   JMAX            YES.GET JMAX                          BUFF0032
         SA5   J               GET J                                 BUFF0033
         IX3   X3-X5           JMAX-J                                BUFF0034
         PX3   B0,X3                                                 BUFF0035
         NX3   B0,X3           FLOAT(JMAX-J)                         BUFF0036
         SA5   NCW                                                   BUFF0037
         PX5   B0,X5                                                 BUFF0038
         NX5   B0,X5           FLOAT(NCW)                            BUFF0039
         FX5   X3*X5           (JMAX-J)*NCW                          BUFF0040
         UX5   B3,X5                                                 BUFF0041
         LX5   B3,X5           FIX IT                                BUFF0042
         SA3   JSH             GET JSH                               BUFF0043
         IX5   X5+X3           L=(JMAX-J)*NCW+JSH                    BUFF0044
         IX3   X4-X5           JCNT-L                                BUFF0045
         NG    X3,RSTOR        L.LE.JCNT.IF NO,EXIT                  BUFF0046
         SX4   0               YES.JCNT=0                            BUFF0047
         SX3   15              X3=15(STOP PLOT)=ILOC1                BUFF0048
         BX6   X3                                                    BUFF0049
         SA6   ILOC1           ILOC1=15(STOP PLOT)                   BUFF0050
         EQ    B0,B0,BF1000                                          BUFF0051
BF140    SA3   JSH             GET JSH                               BUFF0052
         SA5   NCW             GET NCW                               BUFF0053
         IX3   X3-X5                                                 BUFF0054
         ZR    X3,BF160        IS JSH=NCW                            BUFF0055
         SX3   15              NO.SET STOP PLOT                      BUFF0056
         JP    BF1000                                                BUFF0057
BF160    SX4   1               YES.X4=JCNT=1                         BUFF0058
         SA3   J                                                     BUFF0059
         SX3   X3-3                                                  BUFF0060
         PL    X3,BF1030       IS J.GT.2(KF)                         BUFF0061
         EQ    B0,B0,EXIT      NO.EXIT                               BUFF0062
BF200    NZ    X3,BF2000       IS ICNT.EQ.0                          BUFF0063
         ZR    X4,RSTOR        YES.IF JCNT.EQ.0,EXIT                 BUFF0064
         SA3   NCW             GET NCW                               BUFF0065
         SB7   X3+1                                                  BUFF0066
         SB6   X4                                                    BUFF0067
         SB3   B7-B6           B3=NCW+1-JCNT=K                       BUFF0068
         SX6   0               IPF=0                                 BUFF0069
         SA6   IPF                                                   BUFF0070
         PL    X1,BF500                                              BUFF0071
         SX1   X1+1            SET TO 2 S COMPLEMENT                 BUFF0072
         NZ    X1,BF500                                              BUFF0073
         SA1   ONES                                                  BUFF0074
BF500    RJ    PACK1           SETS X3=ILOC1                         BUFF0075
         SA5   IPF                                                   BUFF0076
         ZR    X5,BF1000                                             BUFF0077
         SB2   X3                                                    BUFF0078
```

```
         SA3   ITAB+B2        ILOC1=ITAB(ILOC1)                          BUFF0079
BF1000   RJ    PACK2                                                     BUFF0080
         SX6   B4-1           JSH=JSH-1                                  BUFF0081
         SA6   JSH                                                       BUFF0082
         SX6   X6-1                                                      BUFF0083
         SA5   J                                                         BUFF0084
         NG    X6,BF1040      IS JSH.LT.1                                BUFF0085
         NZ    X6,BF1100      NO.IS JSH.GT.1                             BUFF0086
BF1010   SA3   JMAX           NO.JSH=1                                   BUFF0087
         IX3   X3-X5                                                     BUFF0088
         NZ    X3,BF1100      IS J=JMAX                                  BUFF0089
         SB5   X5-1           YES.SAVE J                                 BUFF0090
         SX3   15                                                        BUFF0091
         RJ    PACK2          STORE STOP PLOT                            BUFF0092
         SA7   A0+B5          IBUF(J)=ILOC2                              BUFF0093
BF1030   RJ    FDUMP          WRITE IBUF                                 BUFF0094
         SX6   2                                                         BUFF0095
         SA6   J              J=KF                                       BUFF0096
         JP    BF1050                                                    BUFF0097
BF1040   SB5   X5-1                                                      BUFF0098
         SA7   A0+B5          IBUF(J)=ILOC2                              BUFF0099
         SX6   X5+1           J=J+1                                      BUFF0100
         SA6   J              STORE J                                    BUFF0101
BF1050   SX7   B0             ILOC2=0                                    BUFF0102
         SA5   NCW                                                       BUFF0103
         SX6   X5                                                        BUFF0104
         SA6   JSH            JSH=NCW                                    BUFF0105
BF1100   SX4   X4-1           JCNT=JCNT-1                                BUFF0106
         SB5   X4                                                        BUFF0107
         EQ    B5,B0,EXIT     EXIT IF JCNT=0                             BUFF0108
         LT    B5,B0,BF140    IS JCNT.LT.0                               BUFF0109
         SB3   B3+1           NO.JCNT.GT.0. K=K+1                        BUFF0110
         SB5   B3-1           K-1                                        BUFF0111
         SA5   NCW                                                       BUFF0112
         SB4   X5             GET NCW                                    BUFF0113
         EQ    B5,B4,BF1130   IS (K-1) =NCW                              BUFF0114
         LT    B5,B4,BF500    NO.IS (K-1) .LT.NCW                        BUFF0115
         JP    BF2200         NO. (K-1) .GT.NCW                          BUFF0116
BF1130   SA1   NK                                                        BUFF0117
         JP    BF2240                                                    BUFF0118
BF2000   SA5   NCW                                                       BUFF0119
         SB3   X5+1           K=B3=NCW+1                                 BUFF0120
         BX3   X4             ILOC1=X3=JCNT                              BUFF0121
         SX4   X4+1           JCNT=JCNT+1                                BUFF0122
         NZ    X3,BF1000      IS ILOC1=0                                 BUFF0123
         SX4   2              YES.JCNT=2                                 BUFF0124
BF2020   SA5   MSK377                                                    BUFF0125
         BX1   X1*X5          NK=MOD(LOC(1),128)                         BUFF0126
         SX5   X1-32                                                     BUFF0127
         NG    X5,BF1000      IS NK.LT.32.SKIP IF YES                    BUFF0128
         SX5   X1-96          NO                                         BUFF0129
         PL    X5,BF2060      IS NK.LT.96.IF NOT,IT IS.LT.128            BUFF0130
         SX1   X1-32          YES                                        BUFF0131
         SX3   1                                                         BUFF0132
         JP    BF1000                                                    BUFF0133
BF2060   SX1   X1-54                                                     BUFF0134
         JP    BF1000                                                    BUFF0135
BF2200   SX6   0                                                         BUFF0136
         SA6   IPF            IPF=0                                      BUFF0137
         NZ    X2,BF2220      IS NCNT=0                                  BUFF0138
         BX3   X1             YES.ILOC1=NK                               BUFF0139
         JP    BF1000                                                    BUFF0140
```

```
BF2220   PL   X2,BF2230      IS NCNT.GT.0                          BUFF0141
         SA1  NCW             NO.NCNT.LT.0.NK=NCW                   BUFF0142
         JP   BF2240                                                BUFF0143
BF2230   SX1  1               YES.NK=1                              BUFF0144
BF2240   SB3  X1              B3=K=NK                               BUFF0145
         BX6  X1                                                    BUFF0146
         SA6  NK              SAVE NK                               BUFF0147
         SA5  IPF                                                   BUFF0148
         SX6  X5+1                                                  BUFF0149
         SA6  IPF             IPF=IPF+1                             BUFF0150
         SB2  X6-1                                                  BUFF0151
         SA1  B1+B2           ILOC=LOC(IPF)                         BUFF0152
         JP   BF500                                                 BUFF0153
EXIT     SA7  ILOC2                                                 BUFF0154
         EQ   B0,B0,RSTOR                                           BUFF0155
RSTOR    SA4  TEMPA0                                                BUFF0156
         SA0  X4              RESTORE A0                            BUFF0157
         EQ   B0,B0,BUFF      EXIT                                  BUFF0158
PACK1    DATA 0                                                     BUFF0159
         SX3  B3              GET K                                 BUFF0160
         SX5  77B             SET MASK                              BUFF0161
         SB2  54                                                    BUFF0162
PK1      SX3  X3-1            TEST K                                BUFF0163
         ZR   X3,PK2                                                BUFF0164
         SB2  B2-6            GET CORRECT SHIFT                     BUFF0165
         EQ   B0,B0,PK1                                             BUFF0166
PK2      AX3  B2,X1           ILOC IS IN X1                         BUFF0167
         BX3  X3*X5           MASK IT(ILOC1)                        BUFF0168
         JP   PACK1                                                 BUFF0169
PACK2    DATA 0                                                     BUFF0170
         SA5  JSH             GET JSH                               BUFF0171
         SB2  0                                                     BUFF0172
         SB4  X5              SAVE JSH                              BUFF0173
PK3      SX5  X5-1            TEST JSH                              BUFF0174
         ZR   X5,PK4                                                BUFF0175
         SB2  B2+6            GET CORRECT SHIFT FOR OUTPUT WORD     BUFF0176
         EQ   B0,B0,PK3                                             BUFF0177
PK4      SX5  77B                                                   BUFF0178
         BX3  X3*X5           MASK IT                               BUFF0179
         LX3  B2,X3           SHIFT TO PROPER POSITION              BUFF0180
         BX7  X7+X3           COMBINE WITH PART WORD                BUFF0181
         JP   PACK2                                                 BUFF0182
*                                                                   BUFF0183
*     INTERNAL SUBROUTINE FDUMP                                    BUFF0184
*     FDUMP CONSTRUCTS THE I/O LIST AND INITIATES THE OUTPUT       BUFF0185
*                                                                   BUFF0186
 FDUMP    DATA 0                                                    BUFF0187
          SX6    A7+1            LWA + 1                          BUFF0188
          SA5    IBUF1           FWA                              BUFF0189
          IX7    X6-X5           RECORD LENGTH IN WORDS           BUFF0190
          PUTW   TAPE2,IBUF,X7                                    BUFF0191
          SX7    B0              ILOC2=0                          BUFF0192
          JP     FDUMP                                            BUFF0193
*                                                                  BUFF0194
*     INTERNAL SUBROUTINE TAPNAM                                   BUFF0195
*     TAPNAM CONSTRUCTS THE PLOT FILE NAME,                        BUFF0196
*     INITIALIZES THE I/O LIST, AND STORES                         BUFF0197
*     THE SYNC CODES IN EACH BUFFER                                BUFF0198
*                                                                  BUFF0199
 TAPNAM   DATA 0                                                   BUFF0200
          PX1  B0,X1           FLOAT LOGICAL TAPE                BUFF0201
          NX1  B0,X1                                             BUFF0202
```

```
          SA2  F10           GET FL.PT. 10                                  BUFF0203
          FX4  X1/X2         LTAPE/10                                       BUFF0204
          UX4  B3,X4         FIX RESULT OF DIVISION                         BUFF0205
          LX4  B3,X4                                                        BUFF0206
          PX3  B0,X4         FLOAT FIXED VALUE                              BUFF020?
          NX3  B0,X3                                                        BUFF020?
          FX3  X3*X2         REMULTIPLY                                     BUFF020?
          FX5  X1-X3         COMPUTE REMAINDER                              BUFF021?
          UX5  B3,X5         FIX REMAINDER                                  BUFF0211
          LX5  B3,X5                                                        BUFF0212
          SX5  X5+27         CONVERT TO DISPLAY CODE                        BUFF0213
          NZ   X4,CHKBIG                                                    BUFF0214
          BX4  X5                                                           BUFF0215
          SX5  B0            MASK IN BLANK                                  BUFF0216
          EQ   B0,B0,TOK1    SKIP HIGH ORDER CHECK                          BUFF0217
CHKBIG    SB1  X4-10                                                        BUFF0218
          LT   B1,B0,TOK                                                    BUFF0219
          SX4  B0                                                           BUFF0220
TOK       SX4  X4+27         CONVERT TO DISPLAY CODE                        BUFF0221
TOK1      SA2  TAPE          PICK UP TAPE PARAMETER                         BUFF0222
          LX4  30            SHIFT HIGH ORDER TO POSITION                   BUFF0223
          LX5  24            SHIFT LOW ORDER DIGIT                          BUFF0224
          BX6  X4+X2         COMBINE HIGH ORDER AND TAPE                    BUFF0225
          BX6  X6+X5                                                        BUFF0226
          MX3    42                                                         BUFF0227
          BX6    X6*X3           FILENAME                                   BUFF0228
          STORE  TAPE2,LFN=X6                                               BUFF0229
          OPENM  TAPE2,I-O,R    OPEN FILE,INITIALLY REWOUND
          SX7    IBUF                                                       BUFF0231
          SA7    IBUF1          SET LOCATION OF BUFFER                      BUFF0232
          SA2    SYNCH                                                      BUFF0233
          BX6    X2                                                         BUFF0234
          SA6    IBUF           STORE SYNCH CODE                            BUFF0235
          EQ   B0,B0,TAPNAM                                                 BUFF0236
CLOSE     BSSZ     1
          SX6      A0                  SAVE A0
          SA6      TEMPA0
          CLOSEM   TAPE2               CLOSE FILE
          SA5      TEMPA0
          SA0      X5
          EQ       CLOSE               RETURN
F10       DATA 10.0                                                         BUFF0237
IBUF1     DATA 0                                                            BUFF0238
ILOC1     DATA 0                                                            BUFF0239
ILOC2     DATA 0                                                            BUFF0240
IPF       DATA 1                                                            BUFF0241
J         DATA 2                                                            BUFF0242
JMAX      DATA 48                                                           BUFF0243
JSH       DATA 10                                                           BUFF0244
MSK377    DATA 377B                                                         BUFF0245
ONES      DATA 777777777777777777B                                          BUFF0246
NCW       DATA 10                                                           BUFF0247
NK        DATA 0                                                            BUFF0248
SYNCH     DATA 37313137B                                                    BUFF0249
TAPE      VFD    24/4HTAPE,36/0                                             BUFF0250
*                                                                           BUFF0251
*     FILE INFORMATION TABLE (FIT) FOR PLOT FILE                            BUFF0252
*                                                                           BUFF0253
          FILE  LFN=TAPE2,FO=SQ,PD=I-O,BT=C,RT=S,MBL=5120,WSA=IBUF
*                                                                           BUFF0255
          DATA 0                                                            BUFF0256
          DATA 33                                                           BUFF0257
```

DATA 34	BUFF0258
DATA 35	BUFF0259
DATA 36	BUFF0260
DATA 37	BUFF0261
DATA 38	BUFF0262
DATA 39	BUFF0263
DATA 40	BUFF0264
DATA 41	BUFF0265
DATA 42	BUFF0266
DATA 43	BUFF0267
DATA 44	BUFF0268
DATA 45	BUFF0269
DATA 46	BUFF0270
DATA 47	BUFF0271
DATA 48	BUFF0272
DATA 49	BUFF0273
DATA 50	BUFF0274
DATA 51	BUFF0275
DATA 52	BUFF0276
DATA 53	BUFF0277
DATA 54	BUFF0278
DATA 55	BUFF0279
DATA 56	BUFF0280
DATA 57	BUFF0281
DATA 58	BUFF0282
DATA 16	BUFF0283
DATA 17	BUFF0284
DATA 18	BUFF0285
DATA 19	BUFF0286
DATA 20	BUFF0287
DATA 21	BUFF0288
DATA 22	BUFF0289
DATA 23	BUFF0290
DATA 24	BUFF0291
DATA 25	BUFF0292
DATA 11	BUFF0293
DATA 13	BUFF0294
DATA 10	BUFF0295
DATA 15	BUFF0296
DATA 8	BUFF0297
DATA 9	BUFF0298
DATA 4	BUFF0299
DATA 29	BUFF0300
DATA 0	BUFF0301
DATA 12	BUFF0302
DATA 14	BUFF0303
DATA 2	BUFF0304
DATA 59	BUFF0305
DATA 61	BUFF0306
DATA 26	BUFF0307
DATA 1	BUFF0308
DATA 3	BUFF0309
DATA 5	BUFF0310
DATA 62	BUFF0311
DATA 6	BUFF0312
DATA 7	BUFF0313
DATA 28	BUFF0314
DATA 30	BUFF0315
DATA 31	BUFF0316
DATA 32	BUFF0317
DATA 63	BUFF0318
DATA 27	BUFF0319

```
TEMPA0   DATA 0                                                          BUFF0320
IBUF     BSSZ   48                                                       BUFF0321
         END                                                             BUFF0322
      OVERLAY(1,0)
      PROGRAM  MAPDATA

C * * * THIS PROGRAM CONTROLS THE PLOTTING AND ANNOTATING OF MAP AND DATA. * * C

      COMMON/CONTROL/NAIM(8).INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/MAINB/LAREA,SCAX,SCAY.OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00

C . . . CALL MAPPLOT TO DRAW THE BASE MAP.
      CALL OVERLAY (6HCONTOR,1,1)

C . . . IF MAPPLOT CARD IS NOT PRESENT RETURN TO CONTOR5.
      IF(NAME.NE.8H*MAPPLOT)RETURN

C . . . CALL BAHMAP TO PLOT AND ANNOTATE DATA AND STORE REQUIRED XYZ VALUES.
      CALL OVERLAY (6HCONTOR,1,2)

C . . . IF MAPPLOT CARD WAS READ BY BAHMAP, START NEW MAP.
      IF(NAIM.NE.8H*MAPPLOT)RETURN $ NAME=6HNEWMAP $ CALL ORICHEK(6)

      END
      OVERLAY(1,1)
      PROGRAM MAPPLOT

C * * * THIS PROGRAM READS INFORMATION ON THE PROJECTION, SCALE AND GRATICULE. C
C * * * SUBROUTINE PROJECT IS CALLED TO DEFINE PROJECTION PARAMETERS.  * * * * C
C * * * THE GRATICULE, BORDER AND MAP NAME AND DATA IS WRITTEN ON THE PLOT FILE.

      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/MAPSEEK/MAPNAME(3),MAPAREA(2),MAPNUM(2),KORN(3),MAP,JECTOR
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00

      DIMENSION GRIDLAT(50),GRIDLON(50),GRIDX(50,50),GRIDY(50,50)
      DIMENSION MAPSCAL(2),MAPDATE(2),CORN(8),IH(4)
      INTEGER  TLAT1,TLAT2,BLAT1,BLAT2,WLON1,WLON2,ELON1,ELON2
      DATA (MARK=7H0     /)

C . . . READ MAPPLOT, PROJECTION, MAP NAME OR NUMBER OR CORNER MARK COORDINATES.
      AL=0.0 $ AM=YT+7.5+2.0*KSIZE $ IF(INMAIN.EQ.0)GO TO 10 $ INMAIN=0
      DECODE(80,15,NAIM)NAME,JECTOR,KORN $ GO TO 60
   10 READ (60,15) NAME,JECTOR,KORN
   15 FORMAT (5A10)
      IF (EOF(60)) 20,30
   20 LOT=3HEOF
   25 CALL ORICHEK(6) $ RETURN
   30 IF(NAME.EQ.8H*MAPPLOT)GO TO 60
      IF(NAME.NE.8H*RESTART)GO TO 40
      NAME=JECTOR $ PRINT 35,NAME $ RETURN
   35 FORMAT (1H0,20(1H*),* RESTART *,A8)
   40 PRINT 50,NAME $ GO TO 10
   50 FORMAT (4H3***,* MAPPLOT CARD MISSING. *,A8,* WAS READ.*)
```

```
   60 NRE(2)=8HMAP/8 ,   $ CALL BEGINIT $ CALL PROJECT (JECTOR)
      IF(KORN.NE.2H  )GO TO 73

C . . . READ MAP NAME INFORMATION AND COORDINATES OF THE AREA TO BE MAPPED.
   65 READ 70,MAPNAME,MAPAREA,MAPNUM(1),TLAT1,TLAT2,BLAT1,BLAT2,
     1        WLON1,WLON2,ELON1,ELON2,NAIM,IH
   70 FORMAT (6A10,4(I3,I2),T1,8A10,T57,4(4X,A1))
      MAP=NAIM.AND.MASK(6) $ IF(MAP.EQ.1L*)GO TO 370

C . . . CHECK MAP COORDINATES FOR VALIDITY.
      IF(WLON1.GT.180.0)WLON1=WLON1-360.0
      IF(ELON1.GT.180.0)ELON1=ELON1-360.0
      IF(IH(1).EQ.1H-.OR.TLAT1.LT.0.0)TLAT2=-TLAT2 $ TLT=60*TLAT1+TLAT2
      IF(IH(2).EQ.1H-.OR.BLAT1.LT.0.0)BLAT2=-BLAT2 $ BLT=60*BLAT1+BLAT2
      IF(IH(3).EQ.1H-.OR.WLON1.LT.0.0)WLON2=-WLON2 $ WLN=60*WLON1+WLON2
      IF(IH(4).EQ.1H-.OR.ELON1.LT.0.0)ELON2=-ELON2 $ ELN=60*ELON1+ELON2
      IF(TLT.LT.BLT.AND.WLN.LT.ELN)GO TO 75
      IF((TLT+BLT+WLN+ELN).NE.0.0)GO TO 71
      KORN(1)=MAPNAME(1) $ KORN(2)=MAPNAME(2) $ KORN(3)=MAPNAME(3)
      CALL MAPSEEK $ GO TO (71,75,130) MAP
   71 PRINT 72,TLT,BLT,WLN,ELN $ GO TO 130
   72 FORMAT (4H0***,* ERROR IN MAP LIMITS. LIMITS GIVEN AS *4F10.4)

C . . . DEFINE PARAMETERS OF THE PROJECTION.
   73 CALL MAPSEEK $ GO TO (65,75,130) MAP
   75 CALL XYCOORD $ IF(SCALE.LE.0.0)GO TO 130 $ ISM=MAP-1

      DECODE (20,76,KORN) CORN,IH
   76 FORMAT (4(F3.0,F2.0),T1,4(A1,4X))
           DO 77 I=1,4
           J=2*I-1 $ IF(CORN(J).GT.180.0)CORN(J)=CORN(J)-360.0
           IF(CORN(J).LT.0.OR.IH(I).EQ.1H-)CORN(J+1)=-CORN(J+1)
   77      CORN(I)=CORN(J)*60.0+CORN(J+1)

C . . . CALCULATE COORDINATES OF THE GRATICULE.
      IF(XSPACE*YSPACE.GT.0.1)GO TO 79
      PRINT 78,XSPACE,YSPACE $ XSPACE=60.0 $ YSPACE=90.0
   78 FORMAT(1H0,9X,*ERROR IN GRATICULE SPACINGS GIVEN AS*F6.2* AND*F6.2
     1      *. SPACINGS REDEFINED AS 60.0 AND 90.0 MINUTES*)
   79 I=YSPACE+0.5 $ IF(I.LT.1)I=15 $ J=XSPACE+0.5 $ IF(J.LT.1)J=15
      M=180 $ L=M/I $ IF(L*I.NE.M)M=(I+30)/60*60 $ L=WLN/M $ RLN=L*M
      M = (WLN-RLN) /YSPACE
      RLN = RLN + M * YSPACE
      IF(RLN.GT.WLN)RLN=RLN-YSPACE
      M=60 $ L=M/J $ IF(L*J.NE.M)M=(J+30)/60*60 $ L=BLT/M $ RLT=L*M
      M = (BLT-RLT) /XSPACE
      RLT = RLT + M * XSPACE  $ IF(RLT.LT.BLT)RLT=RLT+XSPACE
      L = M = 1  $  GRIDLON(1) = WLN  $  GRIDLAT(1)  RLT
   80 L = L+1 $ IF(L.EQ.50)GO TO 82
      GRIDLON(L) =RLN+(L-1)*YSPACE
      IF(GRIDLON(L).LT.ELN)GO TO 80
   82 GRIDLON(L) = ELN

   85 M = M+1 $ IF(M.EQ.50)GO TO 88
      GRIDLAT(M) =RLT-(M-1)*XSPACE
      IF (GRIDLAT(M).GT.TLT)GO TO 85
   88 GRIDLAT(M)=TLT

      DO 95 I=1,L                                                    MAP 038
      ALN = GRIDLON(I)
      DO 90 J=1,M                                                    MAP 039
      ALT = GRIDLAT(J)
```

```
      CALL COORDS                                                        MAP 042
      GRIDX(I,J) = XX
      GRIDY(I,J) = YY
   90 CONTINUE
   95 CONTINUE

      X1 = SQRT ((GRIDX(1,M)-GRIDX(1,1))**2+(GRIDY(1,M)-GRIDY(1,1))**2)
      X2 = SQRT ((GRIDX(L,M)-GRIDX(L,1))**2+(GRIDY(L,M)-GRIDY(L,1))**2)
      X3 = SQRT ((GRIDX(L,M)-GRIDX(1,M))**2+(GRIDY(L,M)-GRIDY(1,M))**2)
      X4 = SQRT ((GRIDX(L,1)-GRIDX(1,1))**2+(GRIDY(L,1)-GRIDY(1,1))**2)
      S00=4*(ELN-WLN)*(BLT-TLT)/((X1+X2) (X3+X4))

      IF(GRIDX(L,1).GT.GRIDX(1,1))GO TO 96 $ SR=1.0 $ CR=0.0 $ GO TO 97
   96 ROTASHN = ATAN((GRIDY(L,1)-GRIDY(1,1))/(GRIDX(L,1)-GRIDX(1,1)))
      SR = SIN(ROTASHN)
      CR = COS(ROTASHN)
   97 XL = GRIDX(1,1)
      YB = GRIDY(1,1)                                                    MAP 057

      DO 99  I=1,L                                                       MAP 059
      DO 98  J=1,M                                                       MAP 060
      XX = GRIDX(I,J)-XL                                                 MAP 061
      YY = GRIDY(I,J)-YB                                                 MAP 062
      GRIDX(I,J) = SR*YY+CR*XX
      GRIDY(I,J) = CR*YY-SR*XX
   98 CONTINUE                                                           MAP 065
   99 CONTINUE                                                           MAP 066
      XR = GRIDX(L,1)                                                    MAP 067
      YT = GRIDY(1,M)                                                    MAP 068

C . . . CHECK THE WIDTH OF THE MAP.
      W1=H1=60.0 $ W2=H2=0.0
          DO 101 I=1,M
          IF(GRIDX(1,I).LT.W1)W1=GRIDX(1,I)
          IF(GRIDX(L,I).GT.W2)W2=GRIDX(L,I)
   10     CONTINUE
          DO 102 I=1,L
          IF(GRIDY(I,1).LT.H1)H1=GRIDY(I,1)
          IF(GRIDY(I,M).GT.H2)H2=GRIDY(I,M)
  102     CONTINUE
      H=H2-H1+8.0 $ W=W2-W1
      H2=H2-0.25 $ H1=H1+0.25 $ W2=W2-0.1 $ W1=W1+0.1
      IF(W.GT.1.0.AND.  GT.8.5)GO TO 110
      PRINT 105 $ GO TO 130
  105 FORMAT(1H0,9X,*MAP SIZE IS TOO SMALL - CHECK THE SCALE *)
  110 IF(W.LT.33.0)GO TO 135 $ PRINT 115
  115 FORMAT (1H0,9X,*WARNING - MAP IS TOO WIDE FOR DRUM PLOTTER*)
      IF(W.GT.58.0.OR.H.GT.58.0)GO TO 120
      IF(W.GT.45.0.AND.H.GT.45.0)120,135
  120 PRINT 125
  125 FORMAT(1H0,9X,*MAP SIZE TOO LARGE FOR BOTH PLOTTERS*)
  130 NAME=5HNOMAP $ GO TO 25

C . . . OPEN THE PLOT FILE AND DEFINE THE ORIGIN.
  135 LUN=42 $ KSIZE=2 $ IF(W.LT.9.0)KSIZE=1
      SIZE = KSIZE $ WSIZE=0.07*KSIZE $ GSIZE=WSIZE/2.0
      IF(LOTX.NE.1)GO TO 140 $ AL=1.0-W1 $ AM=8.0
      CALL PLOTS (IBUF,1000,LUN) $ CALL TAGBLOK
  140 WRITE (LT,141) LOTX $ CALL TAGBLOK
  141 FORMAT (* DEFINE ORIGIN OF MAP*,I3,6X)
      CALL PLOT (AL,AM,-3) $ CALL ORICHEK (1)
```

```
C . . . PLOT CORNER MARKS.
      IF(CORN(1)+CORN(2).EQ.0.0)GO TO 180 $ K=7 $ N=-2
          DO 150 I=1,2 $ ALT=CORN(I)
              DO 145 J=3,4 $ ALN=CORN(J)
              CALL COORDS
              CORN(K)=XX $ CORN(K+1)=YY $ K=K+N
  145         CONTINUE $ K=1 $ N=2
  150     CONTINUE $ K=1

          DO 160 I=1,2
              DO 155 J=1,2
              XX=CORN(K)-XL $ YY=CORN(K+1)-YB
              CORN(K)=SR*YY+CR*XX
              CORN(K+1)=CR*YY-SR*XX
  155         K=K+2
  160     CONTINUE
      W=CORN(5)-CORN(7) $ H=CORN(8)-CORN(2)
      IF(W.LT.34.0)GO TO 190 $ PRINT 170 $ PRINT 115
  170 FORMAT(1H0,9X,*REGISTRATION MARKS OUTSIDE PLOTTER LIMITS*)
      IF((W.LT.58.0.AND.H.LT.58.0).AND.(W.LT.45.0.OR.H.LT.45.))GO TO 190
      PRINT 125
  180 CORN(1)=GRIDX(1,1) $ CORN(2)=GRIDY(1,1)
      CORN(3)=GRIDX(L,1) $ CORN(4)=GRIDY(L,1)
      CORN(5)=GRIDX(L,M) $ CORN(6)=GRIDY(L,M)
      CORN(7)=GRIDX(1,M) $ CORN(8)=GRIDY(1,M)
  190     DO 200 J=1,7,2
  200     CALL SYMBOL (CORN(J),CORN(J+1),.05,3,0.,-1)
      WRITE (LT,205) $ CALL TAGBLOK
  205 FORMAT (* REGISTRATION MARKS *,10X)
      CALL PLOT (0.0,0.0,-3)

C . . . PLOT THE GRATICULE.
          DO 220 J=1,M $ K=3
              DO 210 II=1,L $ I=II $ IF((J/2)*2.EQ.J)I=L+1-II
              CALL PLOT (GRIDX(I,J),GRIDY(I,J),K) $ K=2
  210         CONTINUE
  220     CONTINUE

          DO 240 II=1,L $ I=L+1-II $ IF((M/2)*2.EQ.M)I=II $ K=3
              DO 230 JJ=1,M $ J=M+1-JJ $ IF((II/2)*2.EQ.II)J=JJ
              CALL PLOT (GRIDX(I,J),GRIDY(I,J),K) $ K=2
  230         CONTINUE
  240     CONTINUE

C . . . PLOT THE MAP BOUNDARY.
      IF(GRIDX(1,1).GT.W1.AND.GRIDX(1,M).GT.W1)GO TO 250
      IF(GRIDX(L,1).LT.W2.AND.GRIDX(L,M).LT.W2)GO TO 250
      IF(GRIDY(1,1).GT.H1.AND.GRIDY(L,1).GT.H1)GO TO 250
      IF(GRIDY(1,M).LT.H2.AND.GRIDY(L,M).LT.H2)GO TO 250
      BOUNDRY = 0.4* SIZE
      CALL PLOT(GRIDX(1,1)-BOUNDRY,GRIDY(1,1)-BOUNDRY,3)
      CALL PLOT(GRIDX(L,1)+BOUNDRY,GRIDY(L,1)-BOUNDRY,2)
      CALL PLOT(GRIDX(L,M)+BOUNDRY,GRIDY(L,M)+BOUNDRY,2)
      CALL PLOT(GRIDX(1,M)-BOUNDRY,GRIDY(1,M)+BOUNDRY,2)
      CALL PLOT(GRIDX(1,1)-BOUNDRY,GRIDY(1,1)-BOUNDRY,2)

C . . . LABEL GRATICULE COORDINATES.
  250     DO 260 I=1,L
          LONG1 = GRIDLON(I)/60.0
          LONG2 = GRIDLON(I) - 40.0*LONG1
          ENCODE(10,270,LONG2)LONG2 $ ENCODE(6,290,MUF)LONG1,LONG2
          XX=GRIDX(I,1)-3.0*GSIZE $ YY=GRIDY(I,1)-6.0*GSIZE
```

```
          CALL SYMBOL (XX,YY,GSIZE,MUF,0,6)
          CALL SYMBOL (XX+3.0*GSIZE,YY+1.0*GSIZE,GSIZE/2.0,MARK,0,7)
  260     CONTINUE

          DO 280 J=1,M
          LAT1 = GRIDLAT(J)/60.0
          LAT2 = GRIDLAT(J)+40.0*LAT1
          ENCODE(10,270,LAT2) LAT2 $ ENCODE(6,290,MUF)LAT1,LAT2
  270     FORMAT (2X,I8)
          XX=GRIDX(L,J)+1.5*GSIZE $ YY=GRIDY(L,J)-0.5*GSIZE
          CALL SYMBOL (XX,YY,GSIZE,MUF,0,6)
          CALL SYMBOL (XX+3.0*GSIZE,YY+1.0*GSIZE,GSIZE/2.0,MARK,0,7)
  280     CONTINUE

          DO 300 II=1,L $ I=L+1-II
          LONG1 = GRIDLON(I)/60.0
          LONG2 = GRIDLON(I) + 40.0*LONG1
          ENCODE(10,270,LONG2)LONG2 $ ENCODE(6,290,MUF)LONG1,LONG2
  290     FORMAT(.3,1X,R2)
          XX=GRIDX(I,M)-3.0*GSIZE $ YY=GRIDY(I,M)+5.0*GSIZE
          CALL SYMBOL (XX,YY,GSIZE,MUF,0,6)
          CALL SYMBOL (XX+3.0*GSIZE,YY+1.0*GSIZE,GSIZE/2.0,MARK,0,7)
  300     CONTINUE

          DO 320 JJ=1,M $ J=M+1-JJ
          LAT1 = GRIDLAT(J)/60.0
          LAT2 = GRIDLAT(J)+40.0*LAT1
          ENCODE(10,270,LAT2) LAT2 $ ENCODE(6,290,MUF)LAT1,LAT2
          XX=GRIDX(1,J)-9.5*GSIZE $ YY=GRIDY(1,J)-0.5*GSIZE
          CALL SYMBOL (XX,YY,GSIZE,MUF,0,6)
          CALL SYMBOL (XX+3.0*GSIZE,YY+1.0*GSIZE,GSIZE/2.0,MARK,0,7)
  320     CONTINUE

      CALL XYDATA

C . . . ANNOTATE MAP WITH NAME, NUMBER, SCALE AND DATE.
      IF(MAP.NE.2)GO TO 325
      TLAT1=TLT/60 $ TLAT2=TLT-TLAT1*60
      BLAT1=BLT/60 $ BLAT2=BLT-BLAT1*60
      WLON1=WLN/60 $ WLON2=WLN-WLON1*60
      ELON1=ELN/60 $ ELON2=ELN-ELON1*60
  325 CALL DATE(TODAY)
      IF(KORN(3).EQ.1H )KORN(3)=5HSCALE $ J=SCALE
      ENCODE (10,330,J) J $ CALL LJUST (J,1)
  330 FORMAT (I10)
      ENCODE (20,335,MAPSCAL(1)) KORN(3),J
  335 FORMAT (A10,*1:*,A8)
      IF(SCALE.LE.0)MAPSCAL(1)=MAPSCAL(2)=8H
      ENCODE(17,340,MAPDATE(1))  TODAY
  340 FORMAT(7HPLOTTED,A10)
      CALL LJUST (MAPNAME,3)
      MAP=30 $ IF(MAPNAME(3).EQ.1H )MAP=20 $ J=MAP/10 $ RLN=0.14*MAP
      CALL CENTRE (MAPNAME,J) $ CALL CENTRE (MAPAREA,2)
      CALL CENTRE (MAPNUM,2)
      PRINT 350,MAPSCAL,MAPNAME,MAPAREA,MAPNUM,TLAT1,TLAT2,BLAT1,BLAT2,
     1          WLON1,WLON2,ELON1,ELON2
  350 FORMAT(//,20X,2A10,//,40X,3A10,//,40X,2A10,//,40X,2A10.//,38X,
     1*LATITUDE   *,I3,* TO  *,I3,//38X,*LONGITUDE  *,I3,* TO  *,I3)
      PRINT 360,X1,X2,X3,X4,MAPDATE
  360 FORMAT(1H0,45X,*SHRINKAGE CHECK*,//,42X,*LEFT SIDE*,F12.2,/,42X,
     1     *RIGHT SIDE*,F11.2,/,42X,*TOP SIDE ,F13.2,/,42X,*BOTTOM SIDE*,
     2     F10.2,//,44X,A10,A6,///)
```

```
      CALL SYMBOL (XR+BOUNDRY-RLN*SIZE, -1.5*SIZE,2*WSIZE,MAPNAME,0,MAP)
      CALL SYMBOL (XR+BOUNDRY-2.1*SIZE, -1.7*SIZE,WSIZE,MAPAREA,0,20)
      CALL SYMBOL (XR+BOUNDRY-2.1*SIZE, -2.0*SIZE,WSIZE,MAPNUM,0,20)
      CALL SYMBOL (XR+BOUNDRY-2.1*SIZE, -3.3*SIZE,WSIZE,MAPDATE,0,16)
      RLN=0.105*MAP $ CALL RJUST (MAPNUM,2)
      CALL SYMBOL (-BOUNDRY, YT+0.5*SIZE,WSIZE,MAPSCAL,0,20)
      CALL SYMBOL(0.5*XR-RLN*SIZE,YT+0.5*SIZE,3*WSIZE,MAPNAME,0,MAP)
      CALL SYMBOL (XR+BOUNDRY-1.4*SIZE,YT+0.5*SIZE,WSIZE,MAPNUM,0,20)

      WRITE (LT,365) MAPNAME $ CALL TAGBLOK
  365 FORMAT (* GRATICULE AND LABELS OF MAP -*/1X,3A10)
      CALL FINIT $ CALL PLOT (0.0,0.0,-3) $ RETURN

C . . . A CONTROL CARD WAS READ BY MAPPLOT.
  370 PRINT 380,MAPNAME(1) $ INMAIN=1 $ GO TO 130
  380 FORMAT(4H0***,* CONTROL CARD READ FOR MAP NAME. FIRST WORD = *A10)

      END
      SUBROUTINE EDITEXT (NAME,NOW)
C * * * ARRAY NAME IS SEARCHED FOR TEXT WHICH IS CENTRED OR JUSTIFIED. * * * * C
      DIMENSION KAR(100),NEW(100)
      DATA (NIL=1R )
      ENTRY CENTRE
      JUL=JUR=0
   10 NLB=NNB=J=0 $ NOC=NOW*10 $ IF(NOC.GT.100)GO TO 90
   15 DECODE (NOC,20,NAME) (KAR(I),I=1,NOC)
   20 FORMAT (100R1)
         DO 50 I=1,NOC $ NEW(I)=NIL
         IF(KAR(I).EQ.0)GO TO 50
         IF(KAR(I).NE.NIL)GO TO 30
         IF(NNB.EQ.0)50,40
   30    IF(KAR(I).EQ.1R*.AND.JUL.EQ.0)GO TO 60 $ NNB=J+1
   40    J=J+1 $ NEW(J)=KAR(I)
   50    CONTINUE
   60 NLB=(NOC-NNB)/2 $ IF(JUL.NE.0)GO TO 70 $ IF(JUR.NE.0)NLB=NOC-NNB
      IF(NLB.NE.0)GO TO 80
   70 ENCODE (NOC,20,NAME) (NEW(I),I=1,NOC) $ RETURN
   80 NNB=NOC-NLB $ IF(NNB.LE.0)GO TO 70
      ENCODE (NOC,20,NAME) (NIL,I=1,NLB),(NEW(I),I=1,NNB) $ RETURN
      ENTRY LJUST
      JUL=1 $ GO TO 10
      ENTRY RJUST
      JUR=1 $ GO TO 10
   90 PRINT 95,NOC $ NOC=100 $ GO TO 15
   95 FORMAT(4H0***,10X,I3,* CHARACTERS IN NAME. ONLY 100 WILL BE USED*)
      END
      SUBROUTINE MAPSEEK
C * * * ARRAY KORN IS TESTED FOR A NAME, SHEET REFERENCE OR CORNER LIMITS. * * C
C * * * IF KORN IS NUMERIC MAP IS SET TO 1 FOR CORNER LIMITS.
C * * * IF KORN IS ALPHANUMERIC THE SHEET REFERENCE TABLE IS SEARCHED.
C * * * IF KORN IS ALPHABETIC THE MAP NAME TABLE IS SEARCHED.
C * * * IF THE MAPSHEET IS IDENTIFIED THE NAME, AREA AND NUMBER ARE DEFINED, * C
C * * * AND THE LIMITS, SCALE AND PROJECTION PARAMETERS ARE DEFINED. MAP=2 * * C
C * * * IF THE MAP IS UNIDENTIFIED MAP IS SET TO 3.
C - - - THIS SUBROUTINE WRITTEN FOR CONTOR 6 IN SEPTEMBER 1976.
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAPSEEK/MAPNAME(3),MAPAREA(2),MAPNUM(2),KORN(3),MAP,JECTOR
      COMMON/TABLES/NAMES,NUMBS,LIMIT,KEY,INDEX,NUMB2,NAR
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION NAMES(1500),NUMBS(1000),LIMIT(2,1000),KEY(1000),XY(2,2)
      DIMENSION NM1(600),NM2(600),NM3(300),INDEX(100),NUMB2(2,100)
```

```
      DIMENSION KAR(30),NEW(30),NAR(2,20),MAPL(7),MAPR(7,2),MAPS(6)
      EQUIVALENCE (NAMES,NM1),(NAMES(601),NM2),(NAMES(1201),NM3)
      EQUIVALENCE (MAPL,MAPNAME)
      DATA(KEY(I),I=1,131)/1050004,1040006,1050008,1060026,1060028,        1-  5
     A     1050032,1080034,1080046,1020050,1050052,1020054,1060056,        6- 12
     B     1080062,1060068,1040070,1040072,1060074,1040078,1080080,       13- 19
     C     1080082,1080086,1040088,1030112,1060116,1030118,1120120,       20- 26
     D     1040122,1050124,1060126,1020128,1040130,1050144,1020146,       27- 33
     E     1040156,1080164,1020166,1060168,1060170,1030172,1060174,       34- 40
     F     1040176,1080180,1040182,1050184,1080186,1040188,1020206,       41- 47
     G     1060208,1040218,1020220,1040224,24420226,1060230,1040234,      48- 54
     H     1060242,1060246,1040256,1060262,1040264,1040268,1040270,       55- 61
     I     1060272,1070274,1050276,1060282,1060278,1040306,1050308,       62- 68
     J     1040320,1100322,1040326,1060328,1080340,1040352,1020388,       69- 75
     K     1020354,1060364,1050368,1040370,1050374,1040378,1040384,       75- 82
     L     1060390,1040402,1030406,1060408,1050416,1060410,1040412,       83- 89
     M     1040418,1050422,1060424,1050430,1050432,1060434,1060436,       90- 96
     N     1060444,1040446,1060448,1060450,1060452,1040454,1040480,       97-103
     O     1050456,1050458,1050482,1060486,1080494,1080504,1120506,      104-110
     P     1060508,1070510,1040516,1020532,1060518,1020522,1060526,      111-117
     Q     1040530,1040534,1060538,1060540,1060542,1040566,1040568,      118-124
     R     1060570,1060574,1060572,1040576,1080578,1050580,1040582    /  125-131
      DATA(KEY(I),I=132,262)/1020584,1040586,1060588,20760594,1060596,   132-136
     A     1040600,1050602,1080632,1080642,1020648,1060650,1080656,      137-143
     B     1050652,1050658,1050680,1040682,1040684,1050688,1040692,      144-150
     C     1060694,1040696,1050698,1020700,1060702,1060704,1020712,      151-157
     D     1080714,1020716,1060724,1040726,1080752,1030756,1040760,      158-164
     E     1020764,1020766,1080768,1060770,1080774,1060776,1060788,      165-171
     F     1070790,1040796,1040798,1030800,1080806,1040808,1090810,      172-178
     G     1060816,1040842,1050844,1040846,1020848,1040850,1020854,      179-185
     H     1060876,1040880,1020882,1040896,1060888,1060921,1080936,      186-192
     I     1050948,1050954,1050956,1060958,1060960,1080966,1060968,      193-199
     J     1060992,1050996,1061012,1081014,1071016,1061018,1041020,      200-206
     K     1061024,1061026,1081030,1051032,1051034,1061036,1021058,      207-213
     L     1041050,1061052,1061060,1061084,1041086,1051122,1041088,      214-220
     M     1041090,1021092,1061096,1061098,1051104,1061106,1141110,      221-227
     N     1021112,1061114,1041118,1021120,1041122,1061126,1051130,      228-234
     O     1061134,1051140,1031162,1021172,1061176,1141176,1081178,      235-241
     P     1041180,1021182,1061184,1061186,1041190,1041200,1061202,      242-248
     Q     1021204,1041206,1061212,1041214,26851268,22601284,26861270,   249-255
     R     26871272,26881274,26911278,26941282,1041240,26951286,26961288/ 256-262
      DATA(KEY(I),I=263,393)/26971290,27001294,27011296,22671300,1081256 263-267
     A     ,27041302,27051304,1041306,1061308,1061310,1041314,22751146,  268-274
     B     1081144,22771150,1041148,1061336,1081340,1051342,1061344,     275-281
     C     1021346,1021348,1021350,1061352,1021354,1061360,1061362,      282-288
     D     1061364,1061368,1051372,1061374,1061376,1051378,1041384,      289-295
     E     1061388,1061390,1051392,1051396,1021398,1021400,1071424,      296-302
     F     1051426,1061428,1051430,1051434,1041436,1051438,1031440,      303-309
     G     1051464,1061496,1061468,1081470,1061474,1061478,1051482,      310-316
     H     1061484,1061494,1081504,1061506,1051508,1061510,1051514,      317-323
     I     1021516,27131544,27151546,1061606,1071524,23281548,27161550,  324-330
     J     1081530,1081604,23311552,1031532,27171554,1041542,1061558,    331-337
     K     1071582,1041584,1061586,1061610,1041612,1051614,1041616,      338-344
     L     1061618,1061620,1081638,1041638,1061634,1061650,1061652,      345-351
     M     1061654,1061658,1061694,1061676,1031686,1041696,1061700,      352-358
     N     1061702,1061706,1021712,1061714,1071716,1041722,1061726,      359-365
     O     1041724,1061728,27281730,27291732,1141746,1051742,1041744,    366-372
     P     1021748,1061762,1061766,1021774,1051770,1041772,1081776,      373-379
     Q     1041780,1051784,1041788,27411794,1061802,1041808,1081810,     380-386
     R     1041812,1051814,27451816,1041824,1061828,27461834,27471840 /  387-393
      DATA(KEY(I),I=394,520)/1021846,1061848,1041850,1081852,1061858,    394-398
     A     1021854,27511868,1061872,1041878,1061882,1021874,1081876,     399-405
```

```
B     1031880,1031884,1031886,1061890,1051892,1131894,1081896,          406-412
C     1081900,1061918,1061902,1041903,1051906,27521908,1051926,         413-419
D     1041928,1021932,1061934,1061936,1041940,1051948,1041950,          420-426
E     1021952,1051962,1061964,1041970,1061972,1051974,1061976,          427-433
F     1061978,1061980,1021982,1051984,1061986,1061988,1061990,          434-440
G     1061992,1061994,1061998,1090002,1090024,1090048,1090058,          441-447
H     1090066,1090200,1150202,29210258,1090266,1090310,27600438,        448-454
I     27630440,1150492,1090502,1090638,1090686,1070812,1090923,         455-461
J     1090926,27730928,27730930,1090938,1090946,1090962,27681040,       462-468
K     27721044,1091082,1091094,1151102,1091128,1091142,1091370,         469-475
L     1091602,27751136,1091338,1091512,27781556,1091600,1091646,        476-482
M     1091670,24851674,1091672,1091684,1091688,1091690,1091792,         483-489
N     1091838,1091856,27801862,1091866,1091898,1091920,27831966,        490-496
O     1091996,26060136,26370498,26050132,26100158,26170248,26010036,    497-503
P     26520780,26560838,26590890,26630950,26651008,22011023,26661038    504-510
Q     ,26671042,27841486,26580884,1091704,1091764,1050945,1070791,      511-517
R     27870654,28100721,26300391                                  /     518-520
 DATA(KEY(I),I=521,550)/5(0),1060856,1060857,1060858,1060859,           521-529
A     1060860,1060861,1050862,1050863,1050864,1050865,1050868,          530-536
B     1050866,1050869,1050867,1050870,1050901,1050902,1050903,          537-543
C     1050904,1050905,1050906,1050907,1050908,1030909,1030910     /     544-550
 DATA(KEY(I),I=551,600)/2010010,2010030,28310038,2010060,2010198,      551-555
A     2010222,2010236,2010244,2010324,2010356,2010380,2010382,          556-562
B     2010386,2010420,2010496,2010536,2010590,2010604,28370748,         563-569
C     2010758,2011164,2011386,2011498,2011736,2011750,2011786,          570-576
D     2011790,2011938,28551818,2160372,29260394,2090640,25880794,       577-583
E     2090940,2090990,2091598,28621380,2091922, 0     , 0     ,         584-590
F     1030912,1030914,1030915,1030916,1030917,1030913,1070975,          591-597
G     1070976,1070977,1070978                                     /     598-600
 DATA(KEY(I),I=601,731)/1080040,1080044,1080064,1060114,1080134,       601-605
A     1060138,26080148,1080150,1080154,1060160,1050178,1080190,         606-612
B     1080192,1080204,1040232,1020238,1060250,1080252,1080312,          613-619
C     1060314,1060316,1080330,1040332,1040344,1040350,1040358,          620-626
D     21390362,1040366,1040376,1020392,1080393,1020404,1050414,         627-633
E     1020428,1080484,1060490,1030500,1040512,1060514,1040520,          634-640
F     1040544,1080631,1070636,26430634,1040706,1130708,1060710,         641-647
G     1080719,26560762,1080772,1080778,1040782,1060852,1080786,         648-654
H     1080792,1080840,1040878,1060886,1040892,1080894,1060942,          655-661
I     1060944,1070952,1050964,1080994,1060998,1081002,1081010,          662-668
J     1061022,1061046,1061100,1061108,1041132,1061156,21561158,         669-675
K     1081168,26761165,1061196,26781194,1061198,1061218,26811264,       676-682
L     1061220,26831266,1061222,1061226,1041228,1081230,1081232,         683-689
M     26891276,1061234,1061236,26921280,1081238,1081242,1081244,        690-696
N     1081246,1061248,26981292,1061250,1081252,1081254,27021298,        697-703
O     1041258,1081260,1051312,1061152,27071154,1081356,1051462,         704-710
P     1061466,27841488,1051518,1081500,1041522,1061526,1081534,         711-717
Q     1061540,1061622,1031580,1061608,1081624,1041626,1081640,          718-724
R     1041656,1081692,1081708,1031678,1131680,1051738,1141768     /     725-731
 DATA(KEY(I),I=732,830)/1081710,1061718,1081778,27371734,1081720,      732-736
A     1041682,1081782,27441804,1111844,1081796,1041798,1081800,         737-743
B     1041806,1041820,1081836,1061842,27491942,1081944,1081860,         744-750
C     1021870,1081910,1041916,1021924,1061930,1061946,1050653,          751-757
D     29190014,1090084,1090334,29200022,1090191,1090336,29210214,       758-764
E     29210260,27700442,29220216,1091000,27770628,1090338,27700342,     765-771
F     1091006,1090932,27730934,1091138,1090690,1090630,1091536,         772-778
G     27821830,1091864,27851954,1091832,1091968,1041490,1091958,        779-785
H     1060598,1040655,1020718,1040226,1041216,1040851,1080646,          786-792
I     1041905,27931904,1060331,1061636,1061366, 0     , 0     ,         793-799
J     28100722,1060802,24600814,1070953,1070637,1071017,1071583,        800-806
K     1071525,1050925,28080924,1080720,1070979,1070980,1070981,         807-813
L     1070982,1070983,1070984,1070985,1070986,1070987,1020911,          814-820
M     1020918,1020871,1020872,1020873,1020874,1020822,1020823,          821-827
```

```
N    1040733,1040734,1040666                                          /  828-830
 DATA(KEY(I),I=831,900)/2010042,2010240,2010254,2010348,2010360,      831-835
A    2010426,2010750,2010754,2010804,2010922,2011004,2011160,         836-842
B    2011170,28431166,2011192,2011358,2011394,2011422,2011432,        843-849
C    2011502,2011520,2011628,2011642,2011648,2151822,2011826,         850-856
D    2090016,28570012,29260396,2160723,2171056,2171382,28611054,      857-863
E    2091538,2091960,28651956, 0      , 0      , 0      , 0      ,    864-870
F    2180460,2180461,2180462,2180550,2180551,2170557,2180610,         871-877
G    2170620,2180855,2180897,2190898,2190899,2190900,2180970,         878-884
H    2190971,2190972,2190973,2190974,2170988,2180730,2191070,         885-891
I    2191071,2191072,2170925,2170875,2170735,2170667,2170621,0,0 /    892-900
 DATA(KEY(I),I=901,1000)/1061476,1061224,1081914,1060592,1020524,     901-905
A    1060318,1041492,1060162,1080194,29120488,29081528,1060140,       906-912
B    1081644,1061048,29140528,1080895,1080152,1081472,1090018,        913-919
C    1090020,1090210,1090212, 0      , 0      , 0      ,2180398,4(0),  920-930
D    1060662,1060731,1060732,1060820,1060821,1040663,1040664,         931-937
E    1040665,1040617,1040618,1040619,1040555,1040556,1080465,         938-944
F    1080466,1060463,1060464,1060552,1060553,1060554,1060614,         945-951
G    1060615,1060616,1060611,1060612,1060613,1060660,1060661,         952-958
H    1090090,1090091,1090095,1090467,1090468,1090469,1090099,         959-965
I    1090103,1090092,1090096,1090093,1090097,1090100,1090285,         966-972
J    1090286,1090101,1090094,1090098,1090102,1090104,1090287,         973-979
K    1090294,1090105,1090106,1090288,1090289,1090290,1090291,         980-986
L    1090292,1090293,1090295,1090296,1160297,1090470,1090471,         987-993
M    1090472,1090473,1090474,1090475,      0,      0,      0     /   994-000
 DATA((LIMIT(I,J),I=1,2),J=1,70) / 260012,1333018,250012,1440018,      1-  2
A    340012,1380018,270012,1133024,350012,1170018,260012,1320018,      3-  6
B    220012,1333018,190012,1350018,330012,1410018,300012,1363018,      7-  10
C    290012,1470018,200012,1213018,120012,1350018,150012,1260018,     11-  14
D    170012,1440018,250012,1453018,340012,1140018,130012,1410018,     15-  18
E    150012,1290018,200012,1363018,250012,1303018,190012,1470018,     19-  22
F    370012,1470018,320012,1230018,370012,1423018,340012,1423018,     23-  26
G    240012,1483018,350012,1380018,290012,1183018,310012,1440018,     27-  30
H    260012,1410018,300012,1320018,330012,1483018,240012,1380018,     31-  34
I    170012,1333018,360012,1483018,260012,1170018,300012,1170018,     35-  38
J    360012,1440018,250012,1260018,250012,1393018,190012,1273018,     39-  42
K    250012,1380018,270012,1290018,180012,1290018,240012,1440018,     43-  46
L    330012,1440018,310012,1200018,220012,1393018,300012,1453018,     47-  50
M    200012,1470018,      0,      0,340012,1183018,270012,1530018,     51-  54
N    170012,1213018,250012,1243018,210012,1453018,240012,1200018,     55-  58
O    280012,1423018,240012,1513018,170012,1393018,320012,1260018,     59-  62
P    404512,1445024,330012,1380018,260012,1153018,330012,1140018,     63-  66
Q    160012,1453018,290012,1393018,190012,1380018,350012,1483018 /    67-  70
 DATA((LIMIT(I,J),I=1,2),J=71,140)/250012,1410018,340012,1230018,     71-  72
A    130012,1290018,100012,1423018,310012,1453018,330012,1453018,     73-  76
B    160012,1243018,310012,1333018,260012,1500018,330012,1393018,     77-  80
C    220012,1470018,200012,1393018,240012,1260018,130012,1423018,     81-  84
D    380012,1423018,330012,1153018,300012,1290018,240012,1183018,     85-  88
E    240012,1410018,150012,1440018,310012,1290018,260012,1273018,     89-  92
F    300012,1380018,260012,1393018,200012,1260018,320012,1170018,     93-  96
G    190012,1243018,180012,1410018,270012,1170018,320012,1243018,     97-100
H    300012,1230018,280012,1453018,270012,1500018,290012,1363018,    101-104
I    310012,1393018,260012,1350018,200012,1153018,120012,1303018,    105-108
J    150012,1303018,350012,1440018,170012,1230018,404515,1453018,    109-112
K    190012,1393018,320012,1483018,290012,1140018,300012,1513018,    113-116
L    140012,1260018,230012,1483018,290012,1393018,270012,1213018,    117-120
M    330012,1170018,200012,1243018,140012,1423018,250012,1470018,    121-124
N    260012,1123018,230012,1153018,290012,1213018,180012,1440018,    125-128
O    210012,1350018,330012,1333018,230012,1470018,290012,1453018,    129-132
P    260012,1423018,330012,1213018,      0,      0,310012,1273018,    133-136
Q    280012,1440018,270012,1320018,250012,1333018,120012,1290018 /   137-140
 DATA((LIMIT(I,J),I=1,2),J=141,210)/330012,1470018,300012,1273018,   141-142
```

```
A    200012,1350018,310012,1320018,300012,1?73018,310012,1350018,    143-146
B    160012,1410018,220012,1453018,270012,1380018,180012,1423018,    147-150
C    280012,1140018,190012,1423018,280012,1320018,310012,1483018,    151-154
D    250012,1153018,260012,1183018,340012,1483018,120012,1363018,    155-158
E    290012,1513018,230012,1213018,260012,1513018,240012,1350018,    159-162
F    370012,1410018,150012,1423018,310012,1513018,340012,1440018,    163-166
G    230012,1363018,210012,1260018,240012,1320018,250012,1230018,    167-170
H    300012,1140018,424512,1463024,140012,1410018,270012,1470018,    171-174
I    360012,1410018,220012,1350018,200012,1440018, 60012,1470018,    175-178
J    320012,1183018,180012,1453018,270012,1393018,170012,1453018,    179-182
K    290012,1500018,270012,1513018,320012,1440018,300012,1183018,    183-186
L    230012,1453018,350012,1453018,240012,1423018,290012,1260018,    187-190
M    300012,1200018,140012,1320018,330012,1350018,300012,1350018,    191-194
N    350012,1363018,260012,1213018,280012,1170018,250012,1320018,    195-198
O    300012,1213018,180012,1213018,280012,1363018,170012,1260018,    199-202
P    150012,1320018,404512,1463024,280012,1213018,180012,1380018,    203-206
Q    270012,1260018,280012,1200018,170012,1290018,340012,1350018 /   207-210
 DATA((LIMIT(I,J),I=1,2),J=211,295)/270012,1303018,160012,1273018,   211-212
A    300012,1440018,230012,1440018,300012,1260018,200012,1273018,    213-216
B    230012,1273018,240012,1393018,260012,1290018,210012,1483018,    217-220
C    220012,1410018,290012,1530018,240012,1230018,310012,1260018,    221-224
D    340012,1363018,330012,1230018,370012,1483018,320012,1423018,    225-228
E    190012,1200018,230012,1423018,300012,1500018,210012,1423018,    229-232
F    210012,1183018,290012,1380018,290012,1273018,290012,1303018,    233-236
G    370012,1440018,320012,1410018,290012,1200018,340012,1410018,    237-240
H    120012,1333018,190012,1410018,290012,1410018,290012,1230018,    241-244
I    230012,1123018,260012,1470018,240012,1500018,300012,1153018,    245-248
J    290012,1483018,160012,1380018,230012,1243018,160012,1440018,    249-252
K    14(0),200012,1380018,12(0),210012,1303018,4(0),250012,1500018,  253-270
L    190012,1213018,270012,1153018,220012,1440018,0,0,250012,        271-274
M    1350018,0,0,   210012,1410018,250012,1200018,220012,1320018,    275-279
N    360012,1391521,310012,1243018,300012,1483018,340012,1453018,    280-283
O    320012,1470018,280012,1243018,320012,1513018,330012,1183018,    284-287
P    230012,1183018,220012,1123018,290012,1170018,270012,1363018,    288-291
Q    320012,1273018,180012,1243018,280012,1290018,170012,1410018 /   292-295
 DATA((LIMIT(I,J),I=1,2),J=296,370)/320012,1213018,210012,120001?,   296-297
A    310012,1303018,320012,1320018,320012,1453018,310012,1470018,    298-301
B    414512,1463024,320012,1393018,210012,1134022,270012,1350018,    302-305
C    300012,1303018,110012,1423018,320012,1380018,350012,1410018,    306-309
D    310012,1380018,310012,1151022,250012,1183018,150012,1363018,    310-313
E    340012,1153018,160012,1213018,370014,1393018,210012,1243018,    314-317
F    210012,1153018,130012,1303018,320012,1153018,350012,1393018,    318-321
G    290012,1243018,260012,1363018,330012,1423018,4(0),280012,       332-326
H    1230018,424512,1443024,4(0),140012,1290018,190012,1363018,      327-332
I    2(0),380012,1410018,2(0),200012,1483018,210012,1170018,         333-337
J    414512,1443024,260012,1440018,240012,1123018,240012,1273018,    338-341
K    170012,1423018,340012,1393018,200012,1423018,260012,1230018,    342-345
L    230012,1200018,240012,1333018,260012,1483018,200012,1170018,    346-349
M    220012,1183018,220012,1213018,230012,1230018,230012,1260018,    350-353
N    270012,1183018,210012,1230018,380012,1470018,240012,1530018,    354-357
O    250012,1273018,300012,1243018,250012,1123018,220012,1500018,    358-361
P    270012,1200018,404016,1440018,240012,1470018,250012,1213018,    362-365
Q    230012,1393018,210012,1273018,4(0),350012,1423018          /    366-370
 DATA((LIMIT(I,J),I=1,2),J=371,443)/280012,1393018,270012,1483018,   371-372
A    330012,1500022,220012,1230018,260012,1260018,310012,1500018,    373-376
B    290012,1320018,240012,1453018,190012,1290018,210012,1440018,    377-380
C    300012,1333018,250012,1483018,2(0),270012,1230018,280012,       381-384
D    1410018,220012,1363018,270012,1440018,310012,1363018,2(0),      385-389
E    190012,1453018,240012,1213018,4(0),350012,1500018,220012,       390-394
F    1243018,210012,1380018,140012,1333018,280012,1260018,290012,    395-398
G    1423018,2(0),270012,1273018,160012,1423018,280012,1273018,      399-403
H    300012,1470018,170012,1350018,360012,1453018,370012,1453018,    404-407
```

```
I     380012,1453018,240012,1243018,280012,1350018,280012,1513018,    408-411
J     160012,1290018,170012,1303018,270012,1243018,220012,1273018,    412-415
K     120012,1410018,280012,1303018,     0,      0,330012,1363018,    416-419
L     250012,1530018,310012,1423018,220012,1260018,260012,1200018,    420-423
M     250012,1423018,270012,1333018,220012,1423018,340012,1500018,    424-427
N     260012,1303018,250012,1140018,270012,1453018,220012,1153018,    428-431
O     290012,1290018,280012,1153018,160012,1230018,220012,1140018,    432-435
P     290012,1440018,320012,1350018,260012,1140018,210012,1153018,    436-439
Q     200012,1200018,280012,1383018,260012,1243018,310012,1230018 /   440-443
 DATA((LIMIT(I,J),I=1,2),J=444,550)/100012,1483018,30012,1410018,     444-445
A     40012,1423018,60012,1483018,90012,1453018,40012,1440018,90012,  446-449
B     1410018,2(0),80012,1470018,110012,1513018,4(0),90012,1423018,   450-456
C     100012,1513018,80012,1410018,60012,1500018,430012,1470018,      457-460
D     100012,1470018,60012,1440018,4(0),20012,1495020,70012,1440018,  461-466
E     80012,1423018,4(0),20012,1513018,50012,1453018,90012,1440018,   467-472
F     60012,1453018,80012,1410018,10012,1440018,50012,1440018,2(0),   473-477
G     30012,1510018,50012,1513018,2(0),60012,1410018,104515,1530018,  478-482
H     110012,1500018,2(0),50012,1470018,70012,1470018,100012,1500018  483-487
I     ,30012,1523018,40012,1560018,90012,1483018,20012,1410018,2(0),  488-492
J     50012,1423018,70012,1453018,30012,1423018,2(0),80012,1453018,   493-497
K     32(0),30012,1440018,50012,1500018,360012,1363018,420012,        498-516
L     147(0)18,16(0),320012,1140018,350012,1140018,350012,1183018,    517-528
M     330012,1243018,340012,1200018,340012,1243018,320012,1290018,    529-532
N     320012,1303018,330012,1320018,340012,1320018,350012,1320018,    533-536
O     340012,1333018,350012,1333018,340012,1350018,350012,1350018,    537-540
P     360012,1380018,360012,1393018,370012,1380018,370012,1393018,    541-544
Q     380012,1380018,380012,1393018,390012,1380018,390012,1393018,    545-548
R     390012,1410018,390012,1423018                                /   549-550
 DATA((LIMIT(I,J),I=1,2),J=551,600)/320048,1380072,320048,1140072,0  551-552
A     ,0,280048,1500072,360048,1500072,280048,1440072,240048,1500072  553-557
B     ,160012,1200072,320048,1440072,240048,1080072,200048,1440072,   558-561
C     200048,1080072,200048,1380072,120048,1440072,120048,1260072,    562-565
D     360048,1320072,320048,1200072,320048,1260072,     0,      0,    566-569
E     360048,1380072,360048,1440072,160048,1380072,280048,1140072,    570-573
F     400048,1380072,320048,1500072,280048,1320072,400048,1440072,    574-577
G     240048,1200072,0,0,40048,1560072,0,0,40048,1380072,0,0,48,      578-583
H     1500072,40048,1440072,40048,1500072,0,0,48,1380072,             584-588
I          0,      0,      0,      0,380012,1453018,390012,1440018,   589-592
J     390012,1453018,390012,1470018,390012,1483018,380012,1483018,    593-596
K     400012,1440018,400012,1453018,400012,1470018,400012,1483018 /   597-600
 DATA((LIMIT(I,J),I=1,2),J=601,672)/230012,1333018,120012,1320018,   601-602
A     120012,1350018,220012,1200018,210012,1333018,200012,1140018,0,  603-606
B     0,110012,1290018,160012,1350018,190012,1183018,290012,1350018,  607-611
C     240012,1290018,130012,1350018,200012,1333018,230012,1410018,    612-615
D     310012,1410018,140012,1230018,180012,1350018,170012,1363018,    616-619
E     150012,1273018,150012,1230018,140012,1363018,140012,1440018,    620-623
F     160012,1393018,120012,1423018,260012,1453018,     0,      0,    624-627
G     200012,1433018,190012,1440018,300012,1410018,105014,1320018,    628-631
H     300012,1530018,290012,1333018,340012,1470018,160012,1320018,    632-635
I     202020,1153018,390012,1460018,280012,1470018,170012,1273018,    636-639
J     180012,1393018,270012,1410018,140012,1303018,393015,1463024,0,  640-643
K     0,220012,1380018,280012,1500018,180012,1273018,190012,1320018,  644-648
L          0,      0,180012,1303018,230012,1320018,230012,1513018,    649-652
M     350012,1153018,210012,1290018,150012,1333018,230012,1350018,    653-656
N     110012,1410018,200012,1230018,200012,1410018,110012,1333018,    657-660
O     310012,1170018,240012,1140018,394512,1443018,280012,1380018,    661 664
P     240012,1303018,320012,1200018,220012,1290018,200012,1320018,    665-668
Q     170012,1243018,130012,1260018,310018,1260018,330012,1230018 /   669-672
 DATA((LIMIT(I,J),I=1,2),J=673,750)/250012,1513018,140012,1273018,0  673-674
A     ,0,110012,1303018,0,0,340012,1213018,0,0,140012,1243018,180012  675-680
B     ,1230018,0,0,190012,1260018,0,0,340012,1170018,220012,1170018,  681-686
C     210012,1470018,220012,1303018,180012,1363018,0,0,240012,        687-690
```

```
D     1170018,160012,1260018,0,0,130012,1320018,230012,1303018,        691-695
E     130012,1333018,210012,1320018,240012,1153018,0,0,180012,         696-699
F     1260018,230012,1290018,200012,1303018,0,0,230012,1380018,        700-704
G     150012,1350018,280012,1333018,190012,1230018,0,0,170012,         705-708
H     1320018,260012,1380018,210012,1213018,0,0,320012,1363018,        709-713
I     250012,1290018,220012,1500018,200012,1183018,130012,1363018,     714-717
J     150012,1243018,250012,1170018,380012,1440018,330012,1200018,     718-721
K     160012,1363018,230012,1500018,140012,1350018,150012,1410018,     722-725
L     210012,1363018,240012,1363018,360012,1423018,280012,1483018,     726-729
M     320012,1333018,360012,1470018,250012,1363018,310012,1183018,     730-733
N     190012,1303018,0,0,180012,1320018,220012,1483018,160012,         734-737
O     1333018,0,0,280012,1530018,190012,1333018,270012,1423018,        738-742
P     200012,1290018,100012,1490018,100012,1414518,110012,1363018,     743-746
Q     230012,1170018,      0,      0,180012,1303018,160012,1303018 /   747-750
 DATA((LIMIT(I,J),I=1,2),J=751,830)/350012,1470018,110012,1350018,     751-752
A     170012,1380018,300012,1423018,310012,1213018,230012,1140018,     753-756
B     310015,1320018,0,0,70012,1423018,80012,1483018,0,0,50012,        757-761
C     1410018,50012,1483018,8(0),60012,1423018,0,0,40012,1523018,0,0   762-771
D     ,70012,1410018,40012,1453018,0,0,10012,1490018,40012,1510018,    772-776
E     90012,1500018,90012,1470018,0,0,40012,1483018,0,0,80012,         777-781
F     1500018,10012,1423018,210012,1500018,83012,1514019,310015,       782-785
G     1273018,243018,1523015,290012,1513024,200012,1470024,160012,     786-789
H     1440024,270012,1525018,120012,1300024,160012,1390012,0,0,        790-794
I     332014,1145012,201521,1170018,220012,1133024,6(0),280012,        795-800
J     1123018,0,0,392016,1434020,392017,1471018,404016,1470018,        801-805
K     420012,1450024,430012,1453018,351512,1363020,      0,      0,    806-809
L     133012,1360015,410012,1440018,410012,1453018,410012,1470018,     810-813
M     410012,1483018,420012,1440018,420012,1453018,420012,1483018,     814-817
N     430012,1440018,430012,1483018,360012,1483018,360012,1500018,     818-821
O     330012,1500018,330012,1513018,340012,1513018,350012,1513018,     822-825
P     310012,1513018,310012,1530018,260012,1513018,260012,1530018,     826-829
Q     220012,1513018                                            /      830
 DATA((LIMIT(I,J),I=1,2),J=831,900)/200048,1320072,280048,1380072,     831-832
A     120048,1200072,80048,1320072,240048,1440072,240048,1380072,      833-836
B     160048,1260072,200048,1140072,280048,1080072,280048,1200072,     837-840
C     200048,1260072,240048,1140072,80048,1260072,0,0,120048,1380072   841-845
D     ,160048,1320072,280048,1260072,200048,1200072,240048,1320072,    846-849
E     240048,1260072,320048,1320072,200048,1500072,120048,1320072,     850-853
F     120048,1140072,80048,1380072,160048,1440072,48,1440072,4(0),     854-859
G     80048,1560072,2800 8,1560072,280048,1620072,0,0,80048,1440072,   860-864
H     80048,1500072,10(0),80048,1080072,80048,1140072,80048,1200072,   865-873
I     120048,1080072,120048,1140072,120048,1500072,160048,1080072,     874-877
J     160048,1500072,360048,1080072,360048,1080072,360048,1140072,     878-881
K     360048,1200072,360048,1260072,400048,1080072,400048,1140072,     882-885
L     400048,1200072,400048,1260072,400048,1320072,400048,1500072,     886-889
M     240048,1020072,440048,1380072,440048,1440072,440048,1500072,     890-893
N     360048,1560072,320048,1560072,240048,1560072,200048,1560072,     894-897
O     160048,1560072,      0,      0,      0,      0              /    898-900
 DATA((LIMIT(I,J),I=1,2),J=901,930)/340015,1153018,340015,1170018,     901-902
A     110012,1350024,330015,1213018,300012,1513024,150012,1241521,     903-906
B     213018,1500018,194515,1183018,130012,1350024,4(0),200012,        907-911
C     1151022,140012,1350024,134010,1260018,0,0,114016,1333018,        912-916
D     110012,1390018,152020,1363018,14515,1470018,14515,1453018,       917-920
E     50012,1543018,60012,1543018,4(0),120048,960072,8(0)       /      921-930
 DATA((LIMIT(I,J),I=1,2),J=931,1000)/210012,1140018,270012,1123018,    931-932
A     270012,1140018,310012,1140018,310012,1153018,200012,1500018,     933-936
B     200012,1513018,210012,1513018,160012,1470018,170012,1470018,     937-940
C     180012,1470018,120012,1470018,130012,1440018,100012,1320018,     941-944
D     110012,1320018,110012,1260018,110012,1273018,120012,1260018,     945-948
E     120012,1273018,130012,1273018,160012,1200018,170012,1200018,     949-952
F     180012,1200018,190012,1140018,190012,1153018,190012,1170018,     953-956
G     200012,1123018,210012,1123018, 10012,1410018, 20012,1423018,     957-960
```

```
H    20012,1440018,80012,1440018,100012,1440018,100012,1453018,      961-964
I    30012,1453018,20012,1500018,10012,1453018,20012,1453018,10012,  965-968
J    1470018,20012,1470018,30012,1470018,40012,1470018,70012,14830   969-973
K    18,30012,1483018,10012,1483018,20012,1483018,10012,1500018,     974-977
L    30012,1500018,40012,1500018,70012,1500018,30012,1513018,30012,  978-981
M    1513018,40012,1513018,40012,1530018,40012,1543018,50012,15300   982-986
N    18,60012,1513018,60012,1530018,70012,1513018,70012,1530018,     987-990
O    70012,1543018,80012,1513018,80012,1530018,90012,1513018,90012,  991-994
P    1530018,100012,1530018,110012,.530018,6(0)                  /   995-000
 DATA (NM1(I),I=1,99) /7HABMINGA,7HADAVALE,8HADELAIDE,5HAJANA,          1-  4
A    6HALBANY,7HALBERGA,7HALCOOTA,5HALROY,10HANA BRANCH,9HANDAMOOKA     5- 10
B    ,9HANGLEDOOL,8HANKETELL,10HARNHEM BAY,6HASHTON,8HATHERTON,        11- 15
C    10HAUGATHELLA,7HAUGUSTA,7HAURUKUN,8HAUVERGNE,10HAVON DOWNS,       16- 20
D    10HAYERS ROCK,3HAYR,10HBAIRNSDALE,10HBALLADONIA,8HBALLARAT,       21- 25
E    9HBALRANALD,8HBARALABA,6HBARKER,6HBARLEE,7HBARNATO,8HBARROLKA,    26- 31
F    6HBARTON,8HBATHURST,8HBEDOURIE,8HBEETALOO,4HBEGA,6HBELELE,        32- 37
G    9HBENCUBBIN,7HBENDIGO,7HBENTLEY,7HBETOOTA,9HBILLILUNA,            38- 42
H    10HBIRDSVILLE,9HBIRKSGATE,10HBIRRINDUDU,8HBLACKALL,8HBOOLIGAL,    43- 47
I    9HBOORABBIN,6HBOULIA,6HBOURKE,5HBOWEN,7HBREADEN,10HBREMER BAY,    48- 53
J    8HBRISBANE,6HBROOME,6HBROWNE,8HBUCHANAN,6HBULLER,6HBULLOO,        54- 59
K    9HBUNDABERG,9HBURKETOWN,9HBURNABBIE,6HBURNIE,5HBURRA,4HBYRO,      60- 65
L    9HBUSSELTON,6HCAIRNS,10HCALLABONNA,9HCAMOOWEAL,8HCANBERRA,        66- 70
M    10HCANTERBURY,9HCAPE ARID,10HCAPE SCOTT,9HCAPE YORK,5HCOBAR,      71- 75
N    10HCARGELLIGO,8HCHARNLEY,8HCHILDARA,10HCHINCHILLA,8HCHOWILLA,     76- 80
O    8HCLERMONT,9HCLONCURRY,4HCOBB,4HCOEN,5HCOLAC,6HCOLLIE,4HCOOK,     81- 87
P    7HCOLLIER,9HCONNEMARA,8HCOOKTOWN,8HCOOMPANA,6HCOOPER,6HCOPLEY,    88- 93
Q    8HCORDILLO,7HCORNISH,8HCORRIGIN,9HCROSSLAND,7HCROYDON,3HCUE /     94- 99
 DATA(NM1(I),I=100,199)/6HCULVER,9HCUNDEELEE,10HCUNNAMULLA,5HDALBY,  100-103
A    10HCURDIMURKA,9HCURNAMONA,9HDALHOUSIE,7HDAMPIER,6HDARWIN,        104-108
B    8HDELAMERE,10HDENILIQUIN,5HDERBY,9HDEVONPORT,6HDOBBYN,5HDUBBO,   109-114
C    7HDONGARA,7HDORRIGO,8HDRYSDALE,8HDUARINGA,7HDUCHESS,7HDUKETON,   115-120
D    10HDUMBLEYUNG,6HDUMMER,8HEBAGOOLA,9HEDDYSTONE,4HEDEL,6HEDMUND,   121-126
E    8HEDJUDINA,10HEINASLEIGH,7HELKEDRA,8HELLISTON,7HEMERALD,         127-131
F    8HENNGONIA,8HEROMANGA,9HESPERANCE,9HEUABALONG,5HEUCLA,4HEULO,    132-137
G    7HEVERARD,5HFINKE,7HFOG BAY,6HFORBES,7HFORREST,10HFREW RIVER,    138-143
H    6HFOWLER,5HFROME,8HGAIRDNER,9HGALBRAITH,7HGALILEE,5HGASON,       144-149
I    10HGEORGETOWN,9HGERALDTON,9HGILBERTON,5HGILES,9HGILGANDRA,       150-154
J    9HGLENBURGH,9HGLENGARRY,8HGOULBURN,4HGOVE,7HGRAFTON,7HGUNANYA,   155-160
K    6HGYMPIE,10HHALE RIVER,8HHAMILTON,10HHANN RIVER,8HHASTINGS,      161-165
L    3HHAY,9HHAY RIVER,6HHELENA,7HHENBURY,7HHERBERT,10HHILL RIVER,    166-171
M    6HHOBART,7HHOLROYD,8HHOMEBOIN,7HHORSHAM,8HHUCKITTA,9HHUGHENDEN   172-177
N    ,4HHUON,5HHYDEN,6HINGHAM,10HINNAMINCKA,9HINNISFAIL,8HINVERELL,   178-183
O    7HIPSWICH,7HIVANHOE,7HJACKSON,7HJERICHO,10HJERILDERIE,6HJUNDAH   184-189
P    ,7HJUBILEE,10HKALGOORLIE,9HKATHERINE,5HKIMBA,9HKINGOONYA,        190-194
Q    9HKINGSCOTE,8HKINGSTON,10HKIRKALOCKA,7HKULGERA,8HKURNALPI /      195-199
 DATA(NM1(I),I=200,301)/9HLA GRANGE,9HLAKE EYRE,9HLANSDOWNE,         200-202
A    8HLARRIMAH,10HLAUNCESTON,8HLAVERTON,9HLAWN HILL,6HLENNIS,        203-207
B    7HLEONORA,8HLIMBUNYA,7HLINCOLN,7HLINDSAY,9HLISSADELL,5HLOUTH,    208-213
C    9HLONGREACH,8HLOONGANA,5HLUCAS,9HMACDONALD,9HMACHATTIE,4HMANN,   214-219
D    6HMACKAY,8HMACKUNDA,7HMACLEAN,6HMADLEY,6HMADURA,8HMAITLAND,      220-225
E    7HMALCOLM,10HMALLACOOTA,6HMANARA,7HMANDORA,7HMANEROO,7HMANILLA   226-231
F    ,6HMANUKA,10HMARBLE BAR,6HMARREE,5HMASON,7HMAURICE,9HMELBOURNE   232-237
G    ,8HMENINDEE,7HMENZIES,7HMILDURA,10HMILINGIMBI,10HMILLUNGERA,     238-242
H    10HMILPARINKA,8HMINIGWAL,7HMINILYA,8HMITCHELL,5HMONTO,5HMOORA,   243-248
I    5HMOREE,10HMORNINGTON,6HMORRIS,7HMOSSMAN,9HMT BARKER,6HMT ISA,   249-254
J    8HMT BRUCE,9HMT COOLON,9HMT DOREEN,10HMT EGERTON,9HMT EVELYN,    255-259
K    9HMOUNT ISA,9HMT LIEBIG,10HMT MARUMBA,8HMT PEAKE,9HMT RAMSAY,    260-264
L    9HMT RENNIE,7HMT THEO,10HMOUNT THEO,9HMT WHELAN,8HMT YOUNG,      265-269
M    10HMUNDUBBERA,5HMUNRO,6HMURGOO,10HMUTTABURRA,7HMCDILLS,          270-274
N    8HMC DILLS,8HMCKINLAY,9HMC KINLAY,7HNABBERU,8HNAPPERBY,          275-279
O    10HNARACOORTE,7HNARETHA,8HNARRABRI,10HNARRANDERA,9HNARROMINE,    280-284
P    5HNEALE,9HNEWCASTLE,9HNEWDEGATE,6HNEWMAN,8HNINGALOO,7HNINGHAN,  285-290
```

```
Q     9HNOOLYEANA,8HNOONAERA,10HNOONKANBAH,7HNOORINA,9HNORMANTON,     291-295
R     8HNORSEMAN,9HNULLAGINE,9HNULLARBOR,5HNUYTS,7HNYMAGEE,6HNYNGAN/  296-301
 DATA(NM1(I),I=302,398)/8HOATLANDS,5HOLARY,6HONSLOW,10HOODNADATTA,    302-305
A     6HOOLDEA,10HORFORD BAY,7HORROROO,5HOUYEN,10HPARACHILNA,5HPERTH  306-311
B     ,9HPEAK HILL.6HPELLEW,9HPEMBERTON,6HPENDER,6HPENOLA,8HPERCIVAL  312-317
C     ,9HPERENJORI,10HPINE CREEK,8HPINJARRA,8HPINNAROO,9HPLUMRIDGE,   318-322
D     10HPOOLOWANNA,9HPOONCARIE,10HPT AUGUSTA,10HPT CLINTON,5HRASON,  323-327
E     10HPORT DAVEY,8HPT DAVEY,10HPT HEDLAND,10HPORT KEATS,6HRANKEN,  328-332
F     8HPT KEATS,8HPORTLAND,10HPT LANGDON,10HPROSERPINE,7HPYRAMID,    333-337
G     10HQUEENSTOWN,7HQUILPIE,6HQUOBBA,9HRAWLINSON,9HRED RIVER,       338-342
H     7HRENMARK,8HRICHMOND,6HROBERT,9HROBERTSON,7HRODINGA,4HROMA,     343-348
I     9HROEBOURNE,8HROY HILL,6HRUDALL,6HRUNTON,4HRYAN,9HSANDSTONE,    349-354
J     6HSAHARA,4HSALE,10HSANDY CAPE,5HSCOTT,7HSEEMORE,9HSHARK BAY,    355-360
K     9HSINGLETON,10HSIR SAMUEL,8HSMITHTON,10HSPRINGSURE,7HSTANLEY,   361-365
L     10HSPRINGVALE,9HSTANSMORE,9HST ARNAUD,9HST GEORGE,9HSWAN HILL,  366-370
M     10HSTRZELECKI,5HSURAT,6HSYDNEY,8HTABLETOP,6HTALBOT,8HTAMWORTH,  371-376
N     10HTALLARINGA,5HTAMBO,6HTANAMI,8HTANGORIN,8HTARCOOLA,6HTAROOM,   377-382
O     10HTENNANT CK,9HTHROSSELL,9HTICKALARA,9HTOBERMORY,8HTOOMPINE,   383-387
P     7HTORRENS,9HTORRES ST,10HTOWNSVILLE,7HTRAINOR,9HTRUANT IS,      388-392
Q     8HTUREE CK,9HULLADULLA,4HURAL,9HURANDANGI,8HURAPUNGA,6HVERNON/  393-398
 DATA(NM1(I),I=399,500)/7HURISINO,5HWAGGA,6HWAIGEN,5HWALSH,5HWANNA,   399-403
A     7HWALGETT,10HWALLHALLOW,10HWANGARATTA,9HWARBURTON,8HWARRAGUL,    404-408
B     5HWARRI,7HWARRINA,7HWARWICK,8HWATERLOO,9HWAVE HILL,8HWESTWOOD,   409-414
C     4HWEBB,5HWEIPA,5HWELLS,9HWESSEL IS,7HWHYALLA,8HWIDE BAY,        415-420
D     9HWILCANNIA,6HWILSON,6HWILUNA,8HWINDORAH,8HWINTINNA,6HWINTON,    421-426
E     10HWOLLONGONG,9HWOODROFFE,8HWOORAMEL,7HWYANDRA,5HWYLOO,5HWYOLA   427-432
F     ,6HYALGOO,5HYAMPI,6HYANREY,10HYANTABULLA,6HYARDEA,7HYARINGA,     433-438
G     10HYARRALOOLA,6HYARRIE,7HYOUANMI,7HYOWALGA,7HZANTHUS,4HABAU,     439-444
H     6HAITAPE,7HAMBUNTI,5HARAWE,4HAROA,5HBOGIA,5HBOIGU,7HBUKA IS,    445-451
I     4HBUNA,8HCALVADOS,9HCP NELSON,9HCP RAOULT,4HDARU,7HDEBOYNE,     452-457
J     9HFLY RIVER,7HGASMATA,6H(HUON),4HKALO,7HKARAMUI,10HKAR KAR IS,  458-463
K     9HKARKAR IS,7HKAVIENG,6HKIKORI,5HKIWAI,8HL KUTUBU,8HL MURRAY,   464-469
L     5HMABUA,6HMADANG,4HMAER,7HMARKHAM,9HMAY RIVER,6HNINIGO,4HRAMU,  470-476
M     9HMASSAU IS,9HNAMATANAI,5HPOMIO,10HPT MORESBY,5HRAGGI,6HROSSEL  477-482
N     ,5HSABLE,6HSAGSAG,7HSAG SAG,8HSALAMAUA,7HSAMARAI,4HSAMO,4HTAUU  483-489
O     ,4HTUFI,6HVANIMO,7HVITU IS,5HWABAG,3HWAU,5HWEWAK,9HWUVULU IS,   490-496
P     4HYULE,10HBARROW IS ,7HDEAL IS,9HBARROW CK                      / 497-500
 DATA(NM1(I),I=501,550)/9HBEDOUT IS,9HBROWSE IS,9HALICE SPR,          501-503
A     8HHERON IS,10HILLOGWA CK,8HJULIA CK,7HKING IS,9HL AMADEUS,      504-508
B     6HL EYRE,10HL JOHNSTON,8HL MACKAY,8HPERCY IS,10HJOANNA SPR,     509-513
C     5HSEPIK,7HTALASEA,8HKERSAINT,8H(HOBART),9HFRASER IS,            514-518
D     9HGROOTE IS,6HCOBHAM,30(1H )                                  / 519-550
 DATA(NM1(I),I=551,600)/9HADELAIDE*,7HALBANY*,10HALICE SPR*,          551-553
A     8HARMIDALE,7HBODALLA,7HBOURKE*,9HBRISBANE*,7HBROOME*,           554-558
B     9HCANBERRA*,9HCARNARVON,9HCLERMONT*,7HCLOATES,10HCLONCURRY*,    559-563
C     9HCOOKTOWN*,7HDARWIN*,10HDU COUEDIC,10HESPERANCE*,4HEYRE,       564-568
D     8HHALLS CK,9HHAMILTON*,10HMELBOURNE*,10HNORMANTON*,6HPERTH*,    569-573
E     6HSTOKES,7HSYDNEY*,9HTARCOOLA*,8HTASMANIA,7HWILUNA*,            574-578
F     10HTORRES ST*,8HCHOISEUL,8HCOCOS IS,10HFLY RIVER*,9HHOLLANDIA,  579-883
G     8HKAVIENG*,3HLAE,6HRABAUL,10HNORFOLK IS,6HWEWAK*,12(1H )      / 584-600
 DATA(NM2(I),I=1,106)/10HALICE SPRI,3HNGS,10HALLIGATOR ,5HRIVER,       1-  4
A     10HARNHEM BAY,7H - GOVE,10HBALFOUR DO,3HWS,10HBARROW CRE,2HEK    5-  10
B     ,10HBARROW ISL,3HAND,10HBATHURST I,1HS,10HBATHURST I,5HSLAND,    11-  16
C     10HBAUHINIA D,4HOWNS,10HBEDOUT ISL,3HAND,10HBILLAKALIN,1HA,      17-  22
D     10HBLOODS RAN,2HGE,10HBLUE MUD B,2HAY,10HBONNEY WEL,1HL,         23-  28
E     10HBRIGHTON D,4HOWNS,10HBROKEN HIL,1HL,10HBROWSE ISL,3HAND,      29-  34
F     10HBRUNETTE D,4HOWNS,10HCALVERT HI,3HLLS,10HCAMBRIDGE ,4HGULF,   35-  40
G     10HCAMDEN SOU,2HND,10HCAPE BEATR,3HICE,10HCAPE MELVI,3HLLE,      41-  46
H     10HCAPE VAN D,5HIEMEN,10HCAPE WEYMO,3HUTH,10HCHARLEVILL,1HE,     47-  52
I     10HCHARLOTTE ,6HWATERS,10HCHARTERS T,5HOWERS,10HCLARKE RIV,      53-  57
J     2HER,10HCOBHAM LAK,1HE,10HCOBURG PEN,7HINSULAR,10HCOFFS HARB,   58-  63
K     3HOUR,10HCOOBER PED,1HY,10HCOOTAMUNDR,1HA,10HDALY WATER,1HS,     64-  70
```

```
L     10HDAMPIER - ,10HYARRALOOLA,10HDEAL ISLAN,1HD,10HDIRRANBAND,     71- 75
M     1HI,10HDIXON RANG,1HE,10HDONORS HIL,1HL,10HDURHAM DOW,2HNS,      76- 82
N     10HFERGUSSON ,5HRIVER,10HFLINDERS I,5HSLAND,10HFLINDERS I,1HS,   83- 88
O     10HGLENORMIST,2HON,10HGOONDIWIND,1HI,10HGORDON DOW,2HNS,         89- 94
P     10HGREEN SWAM,6HP WELL,10HHARDING SP,4HRING,10HHELEN SPRI,       95- 99
Q     3HNGS,10HHERMANNSBU,2HRG,10HHERON ISLA,2HND,10HIRWIN INLE,1HT/ 100-106
 DATA(NM2(I),I=107,198)/10HHIGHLAND R,4HOCKS,10HHODGSON DO,3HWNS,    107-110
A     10HILLOGWA CR,3HEEK,10HJARDINE RI,3HVER,10HJOANNA SPR,3HING,    111-116
B     10HJULIA CREE,1HK,10HJUNCTION B,2HAY,10HKELLERBERR,2HIN,        117-122
C     10HKENNEDY RA,3HNGE,10HKING ISLAN,1HD,10HKOPPERAMAN,2HNA,       123-128
D     10HLAKE AMADE,2HUS,10HLAKE JOHNS,3HTON,10HLAKE MACKA,1HY,       129-134
E     10HLANDER RIV,2HER,10HLENNARD RI,3HVER,10HLONDONDERR,1HY,       135-140
F     10HMADURA - B,8HURNABBIE,10HMALCOLM - ,9HCAPE ARID,             141-144
G     10HMARYBOROUG,1HH,10HMEDUSA BAN,2HKS,10HMEEKATHARR,1HA,         145-150
H     10HMELVILLE I,5HSLAND,10HMELVILLE I,1HS,10HMONDRAIN I,5HSLAND,  151-156
I     10HMONDRAIN I,1HS,10HMONTAGUE S,4HOUND,10HMOUNT ANDE,4HRSON,    157-162
J     10HMT ANDERSO,1HN,10HMOUNT BANN,5HERMAN,10HMT BANNERM,2HAN,     163-168
K     10HMOUNT BARK,2HER,10HMOUNT BRUC,1HE,10HMOUNT COOL,2HON,        169-174
L     10HMOUNT DORE,2HEN,10HMOUNT DRUM,4HMOND,10HMT DRUMMON,1HD,      175-180
M     10HMOUNT EGER,3HTON,10HMOUNT ELIZ,5HABETH,10HMT ELIZABE,2HTH,   181-186
N     10HMOUNT EVEL,2HYN,10HMOUNT LIEB,2HIG,10HMOUNT MARU,3HMBA,      187-192
O     10HMOUNT PEAK,1HE,10HMOUNT PHIL,4HLIPS,10HMT PHILLIP,1HS      / 193-198
 DATA(NM2(I),I=199,298)/10HMOUNT RAMS,2HAY,10HMOUNT RENN,2HIE,       199-202
A     10HMOUNT SOLI,5HTAIRE,10HMT SOLITAI,2HRE,10HMOUNT WHEL,2HAN,    203-208
B     10HMOUNT YOUN,1HG,10HMURLOOCOPP,2HIE,10HMC LARTY H,4HILLS,      209-214
C     10HMCLARTY HI,3HLLS,10HNEWCASTLE ,6HWATERS,10HPANDIE PAN,3HDIE  215-220
D     ,10HPATERSON R,4HANGE,10HPERCY ISLA,3HNDS,10HPORT AUGUS,2HTA,   221-226
E     10HPETERMANN ,6HRANGE,10HPORT CLINT,2HON,10HPORT HEDLA,2HND,    227-232
F     10HPORT LANGD,2HON,10HPRINCE REG,3HENT,10HROBINSON R,4HANGE,    233-238
G     10HQUEENSCLIF,1HF,10HRAVENSTHOR,2HPE,10HROBINSON R,4HIVER,      239-244
H     10HROCKHAMPTO,1HN,10HROPER RIVE,1HR,10HRUTLAND PL,4HAINS,       245-250
I     10HSANDOVER R,4HIVER,10HSIMPSON DE,10HSERT NORTH,10HSAINT ARNA  251-255
J     ,2HUD,10HSAINT GEOR,2HGE,10HSTREAKY BA,1HY,10HTALLANGATT,1HA,   256-262
K     10HSIMPSON DE,10HSERT SOUTH,10HSOUTHERN C,4HROSS,10HTANAMI EAS  263-267
L     ,1HT,10HST LAWRENC,1HE,10HSOUTH LAKE,6H WOODS,10HSAINT LAWR,    268-273
M     4HENCE,10HTANUMBIRIN,1HI,10HTHURSDAY I,1HS,10HTWEED HEAD,1HS,   274-280
N     10HTENNANT CR,3HEEK,10HTHARGOMIND,2HAH,10HTHE GRANIT,2HES,      281-286
O     10HTHURSDAY I,5HSLAND,10HTORRES STR,3HAIT,10HTRUANT ISL,3HAND,  287-292
P     10HTUREE CREE,1HK,10HWINNECKE C,1HK,10HWINNECKE C,4HREEK     /  293-298
 DATA(NM2(I),I=299,404)/10HVICTORIA R,10HIVER DOWNS,10HWAGGA WAGG,    299-301
A     1HA,10HWESSEL ISL,4HANDS,10HWESTMORELA,2HND,10HWHITE CLIF,2HFS  302-308
B     ,10HWIDGIEMOOL,3HTHA,10HWINNING PO,2HOL,10HFOWLER - N,4HUYTS,   309-314
C     10HADMIRALTY ,7HIS EAST,10HAWORRA RIV,2HER,10HCAPE NELSO,1HN,   315-320
D     10HADMIRALTY ,7HIS WEST,10HBLUCHER RA,3HNGE,10HCAPE RAOUL,1HT,  321-326
E     10HBOUGAINVIL,8HLE NORTH,10HBUKA ISLAN,1HD,10HCP ST GEOR,2HGE,  327-332
F     10HBOUGAINVIL,8HLE SOUTH,10HLAKE KUTUB,1HU,10HFERGUSSON ,2HIS,  333-338
G     10HCAPE SAINT,7H GEORGE,10HCAPE ST GE,4HORGE,10HLAKE MURRA,1HY  339-344
H     ,10HKAR KAR IS,4HLAND,10HKARKAR ISL,3HAND,10HMASSAU ISL,3HAND,  345-350
I     10HGAZELLE PE,7HNINSULA,10HFERGUSSON ,6HISLAND,10HPORT MORES,   351-355
J     2HBY,10HTROBRIAND ,2HIS,10HVITU ISLAN,2HDS,10HWOODLARK I,1HS,   356-362
K     10HTROBRIAND ,7HISLANDS,10HWUVULU ISL,3HAND,10HPERCY ISLE,1HS,  363-368
L     10HWOODLARK I,5HSLAND,10HEUCLA - NO,6HONAERA,10HFRASER ISL,     369-373
M     3HAND,9HGRAFTON -,7HMACLEAN,10HBOWEN - PR,8HOSERPINE,           374-378
N     9HMOSSMAN -,6HCAIRNS,9HIPSWICH -,8HBRISBANE,9HFOG BAY -,        379-383
O     6HDARWIN,9HWELLESLEY,7HISLANDS,9HWELLESLEY,2HIS,10HCAPE LEEUW,  384-389
P     2HIN,9HROEBOURNE,9H- PYRAMID,10HNINGALOO -,7H YANREY,4(1H ),    390-398
Q     10HGROOTE ISL,3HAND,10HHOUTMAN AB,6HROLHOS,15HHUON (TASMANIA)/  399-404
 DATA(NM2(I),I=405,460)/10H(KING ISLA,3HND),9H(FLINDERS,7HISLAND),    405-408
A     10H(LAUNCESTO,2HN),10H(QUEENSTOW,2HN),10H(PORT DAVE,2HY),       409-414
B     10HKANGAROO I,5HSLAND,10HKANGAROO I,1HS,10HGROOTE EYL,4HANDT,   415-420
C     40(1H )                                                      /  421-460
 DATA(NM2(I),I=461,600)/10HALICE SPRI,4HNGS*,10HBROKEN HIL,2HL*,     461-464
```

```
A     10HBRUNSWICK ,3HBAY,10HCAPE WESSE,1HL,10HCHARLEVILL,2HE*,        465-470
B     10HCOOPER CRE,2HEK,10HHALLS CREE,1HK,10HHAMERSLEY ,5HRANGE,      471-476
C     10HHOUTMAN AB,7HROLHOS*,10HKALGOORLIE,1H*,10HLAKE MACKA,2HY*,    477-482
D     10HMEEKATHARR,2HA*,10HMELVILLE I,6HSLAND*,10HMELVILLE I,2HS*,    483-488
E     10HMITCHELL R,4HIVER,9HNEWCASTLE,7HWATERS*,9HNULLARBOR,5HPLAIN   489-494
F     ,10HOAKOVER RI,3HVER,10HOODNADATTA,1H*,9HPETERMANN,7HRANGES*,    495-500
G     10HPORT AUGUS,3HTA*,10HROCKHAMPTO,2HN*,10HROPER RIVE,2HR*,       501-506
H     10HROWLEY SHO,3HALS,10HTORRES STR,4HAIT*,10HTOWNSVILLE,1H*,      507-512
I     10HADMIRALTY ,7HISLANDS,9HADMIRALTY,2HIS,10HCOCOS ISLA,3HNDS,    513-518
J     10HGUADALCANA,1HL,9HLORD HOWE,6HISLAND,10HNORFOLK IS,4HLAND,     519-524
K     9HLORD HOWE,2HIS,10HPORT MORES,3HBY*,10HWOODLARK I,6HSLAND*,     525-530
L     10HWOODLARK I,2HS*,68(1H )                                  /    531-600
 DATA(NM3(I),I=1,300)/9HPEMBERTON,10H- IRWIN IN,3HLET,10HMOUNT BARK     1-  4
A     ,10HER - ALBAN,1HY,10HWESSEL ISL,10HANDS - TRU,10HANT ISLAND,      5-  9
B     9HESPERANCE,10H- MONDRAIN,7H ISLAND,9HDORRIGO -,10HCOFFS HARB,    10- 14
C     3HOUR,10HCAMDEN SOU,10HND - PRINC,8HE REGENT,10HPERCY ISLE,       15- 19
D     10HS - PORT C,6HLINTON,10HBEDOUT ISL,10HAND - PORT,8H HEDLAND,    20- 24
E     10HBLUE MUD B,9HAY - PORT,7HLANGDON,9HDAMPIER -,10HBARROW ISL,    25- 29
F     3HAND,10HPORT HEDLA,10HND - BEDOU,8HT ISLAND,10HBARROW ISL,       30- 34
G     10HAND - DAMP,3HIER,10HROPER RIVE,10HR - CAPE B,7HEATRICE,        35- 39
H     10HLONDONDERR,10HY - DRYSDA,2HLE,10HDRYSDALE -,10H LONDONDER,     40- 44
I     2HRY,10HJUNCTION B,10HAY - MILIN,5HGIMBI,10HBATHURST I,           45- 49
J     10HS - MELVIL,5HLE IS,10HPELLEW - R,10HOBINSON RI,3HVER,          50- 54
K     9HADMIRALTY,10HISLANDS EA,2HST,9HADMIRALTY,10HISLANDS WE,2HST     55- 60
L     ,10HBOUGAINVIL,9HLE ISLAND,5HNORTH,10HBOUGAINVIL,9HLE ISLAND,    61- 65
M     5HSOUTH,9(1H ),10HCOCOS (KEE,10HLING) ISLA,3HNDS,222(1H )   /    66-300
 DATA(NUMBS(I),I=1,131)/6HG53-10,5HG55-5,5HI54-9,3HREF,6HI50-15,         1-  5
A     5HG53-9,6HF53-10,6HE53-15,5HI54-7,6HH53-12,5HH55-7,5HF51-2,       6- 12
B     5HD53-3,6HD52-13,5HE55-5,5HG55-6,5HI50-9,5HD54-7,6HD52-15,        13- 19
C     5HF53-4,5HG52-8,6HE55-15,5HJ55-7,5HI51-3,5HJ54-8,6HI54-12,        20- 26
D     5HG55-4,6HI54-13,5HH50-8,6HH55-13,6HG54-11,5HH53-9,5HI55-8,       27- 33
E     5HG54-1,5HE53-6,5HREF  ,6HG50-11,6HH50-11,5HJ55-1,5HG52-5,        34- 40
F     5HG54-6,6HE52-14,5HG54-5,6HG52-15,6HE52-11,5HG55-1,5HI55-5,       41- 47
G     6HH51-13,6HF54-10,6HH55-10,5HF55-3,1H ,6HI50-12,6HG56-15,         48- 54
H     5HE51-6,5HG51-8,5HF55-6,5HG51-1,5HH54-4,5HG56-2,5HE54-6,          55- 61
I     5HI52-1,7H(K55-3),5HI54-5,6HG50-10,5HI50-5,5HE55-2,5HH54-6,       62- 68
J     6HE54-13,6HI55-16,5HG54-7,6HI51-11,5HD52-7,6HC54-12,6HH55-14,     69- 75
K     5HI55-6,5HE51-4,6HH53-14,5HG56-9,5HI54-6,6HF55-11,5HF54-2,        76- 82
L     5HG52-1,5HD54-8,6HJ54-12,5HI50-6,6HH52-11,5HG50-4,5HG54-3,        83- 89
M     6HD55-13,6HH52-15,6HG52-10,5HH54-9,6HG54-10,5HF52-1,5HI50-3,      90- 96
N     6HE51-16,6HE54-11,6HG50-15,5HI51-4,6HH51-11,5HH55-2,6HG56-13,     97-103
O     5HH53-8,6HH54-14,6HG53-11,5HF50-2,5HD52-4,6HD52-16,6HI55-13,     104-110
P     5HE51-7,7HK55-6+2,6HE54-14,5HI55-4,5HH50-5,6HH56-10,5HD52-9,     111-117
Q     6HF55-16,5HF54-6,6HG51-14,5HI50-7,5HF51-4,6HD54-12,5HG55-7,      118-124
R     6HG49-12,6HF50-14,5HH51-6,5HE55-9,5HF53-7,5HI53-6,6HF55-15  /    125-131
 DATA(NUMBS(I),I=132,275)/5HH55-6,6HG54-12,5HI51-6,1H ,6HH52-14,       132-136
A     5HH55-1,6HG53-13,5HG53-6,5HD52-3,5HI55-7,6HH52-10,5HF53-3,       137-143
B     6HH53-13,6HH54-10,6HH53-15,5HE54-3,6HF55-10,6HG54-13,6HE54-12,   144-150
C     5HH50-1,6HE54-16,5HH53-1,6HH55-16,5HG50-6,6HG50-12,6HI55-12,     151-157
D     5HD53-4,5HH56-6,6HF51-14,9HG56-10+11,5HG53-3,5HJ54-7,6HD54-16,   158-164
E     9HH56-14+15,5HI55-9,6HF53-16,5HF52-5,5HG53-1,5HG51-7,5HH50-9,    165-171
F     7H(K55-8),6HD54-11,6HG55-15,5HJ54-3,6HF53-11,5HF55-1,6HB55-11,   172-178
G     5HI50-4,6HE55-10,6HG54-14,5HE55-6,5HH56-5,6HG56-14,5HI55-1,      179-185
H     6HH50-12,6HF55-14,6HI55-14,5HG54-4,5HH52-5,5HH51-9,5HD53-9,      186-192
I     5HI53-7,6HH53-11,6HI53-16,6HG51-10,5HH50-3,5HG53-5,6HH51-10,     193-199
J     6HE51-10,5HH53-4,5HE52-5,6HD53-13,7H(K55-4),5HH51-2,5HE54-9,     200-206
K     6HG52-13,5HH51-1,5HE52-7,9HI53-11+15,6HG52-16,5HE52-2,5HH55-9,   207-213
L     6HE55-13,5HH52-9,5HF52-2,6HF52-14,5HG54-2,6HG52-11,5HF55-8,      214-220
M     6HF54-11,5HH56-7,5HG51-3,6HH52-13,6HI53-12,5HI51-7,5HJ55-8,      221-227
N     5HI54-4,6HE51-13,6HF54-16,5HH56-9,5HF54-8,5HF50-8,5HH54-5,       228-234
O     5HH52-6,5HH52-8,5HJ55-5,5HI54-3,5HH51-5,6HI54-11,5HD53-2,        235-241
P     6HE54-15,5HH54-7,5HH51-7,6HF49-16,6HG55-11,5HG56-1,6HH50-10,     242-248
```

```
Q     5HH55-8,5HE54-1,6HF51-16,5HE55-1,7(1H ),5HF54-1,6(1H ),                249-266
R     5HF52-8,2(1H ),5HG56-5,6HE51-14,6HG50-14,5HF55-9,1H ,5HG53-7/          267-275
 DATA(NUMBS(I),I=276,412)/1H ,5HF54-7,5HG51-5,5HF53-9,7HJ54-2+1,             276-280
A     6HH51-16,6HH55-12,6HI55-10,5HI55-3,5HH51-4,5HI56-2,5HI50-8,            281-287
B     6HF50-16,6HF49-12,5HH50-7,6HG53-16,5HI52-2,6HE51-12,5HH52-3,           288-294
C     5HE54-7,5HI51-2,5HF51-5,6HH52-16,5HI53-1,5HI55-2,6HH55-15,             295-301
D     7H(K55-6),5HI54-2,3HREF,6HG53-15,6HH52-12,6HC54-16,5HI54-1,            302-308
E     6HI54-15,6HH54-13,9HH50-13+14,5HG50-8,6HD53-16,6HI50-10,               309-314
F     5HE51-2,8HJ54-6+10,5HF51-8,5HH50-6,5HD52-8,5HI50-2,6HI54-14,           315-321
G     5HH51-8,6HG53-12,5HI54-8,2(1H ),5HH51-3,7H(K55-7),2(1H ),              322-330
H     6HD52-11,6HE53-16,1H ,6HJ54-11,1H ,5HF55-4,5HF50-7,7H(K55-5),          331-338
I     5HG55-9,5HG49-4,5HG52-2,5HE54-8,6HI54-10,5HF54-4,6HG51-11,             339-345
J     6HF51-13,5HG53-2,6HG55-12,5HF50-3,6HF50-12,6HF51-10,6HF51-15,          346-352
K     6HF52-13,6HG50-16,5HF51-7,6HJ55-11,5HG56-3,5HG52-6,6HH51-12,           353-359
L     5HG49-8,5HI56-1,6HG51-13,7HK55-5+1,5HG55-7,5HG51-6,6HF54-14,           360-366
M     5HF52-6,2(1H ),6HI54-16,5HH54-2,6HG55-16,...I56-5+6,6HF51-11,          367-374
N     5HG52-9,6HH56-13,5HH53-5,5HG55-2,6HE52-15,5HF55-5,6HH53-10,            375-381
O     5HG55-8,1H ,6HG51-15,5HH54-3,6HF53-12,6HG55-13,6HH53-16,1H ,           382-389
P     6HE55-14,5HG51-2,2(1H ),6HI56-13,6HF51-12,5HF54-5,6HD53-10,            390-397
Q     5HH52-1,5HH54-8,1H ,6HG52-14,5HE54-4,5HH52-2,6HH55-11,5HE53-7,         398-405
R     5HJ55-2,5HJ55-6,9HJ55-10+14,5HG51-4,5HH53-3,5HH56-2,5HE52-3 /          406-412
 DATA(NUMBS(I),I=413,520)/5HE52-8,6HG51-16,6HF52-10,5HD54-3,5HH52-4          413-417
A     ,1H ,5HI53-8,5HG56-7,6HH54-16,5HF52-9,5HG51-9,5HG54-8,6HG53-14         418-425
B     ,6HF54-12,5HI56-9,6HG52-12,5HG50-5,6HG55-14,6HF50-10,5HH52-7,          426-432
C     5HH50-2,5HE51-3,5HF50-9,5HH55-5,5HI53-3,5HG50-9,5HF50-6,               433-439
D     5HF51-1,5HH50-4,6HG51-12,6HH51-15,6HC55-12,6HA54-15,5HB54-4,           440-446
E     6HB55-12,5HC55-6,5HB55-1,5HC54-7,1H ,5HC55-3,6HC56-14,2(1H ),          447-455
F     5HC54-8,6HC56-10,5HC54-3,5HB56-9,6HK55-15 ,6HC55-11,5HB55-9,           456-462
G     2(1H ),3HREF,6HB55-13,5HC54-4,2(1H ),6HA56-10,5HB55-6,5HC55-5,         463-472
H     6HB55-10,5HB54-3,5HA55-5,5HB55-5,1H ,8H(A56-14),5HB56-6,1H ,           473-480
I     6HB54-11,9HC56-15+11,6HC56-13,1H ,5HB55-7,6HB55-15,5HC56-9,            481-487
J     8H(A56-15),5HB57-1,5HC55-8,6HA54-11,1H ,5HB54-8,6HB55-14,              488-494
K     6HA54-16,1H ,5HC55-2,16(1H ),6HA55-13,5HB56-5,5HJ53-4,                 495-516
L     6HK55-11,3(1H )                                                 /      517-520
 DATA(NUMBS(I),I=521,550)/5(1H ),5HI50-1,6HI50-13,6HI50-16,5HI51-8,          521-529
A     5HI51-9,6HI51-12,5HI52-3,5HI52-4,5HI53-5,5HI53-9,6HI53-13,             530-536
B     6HI53-10,6HI53-14,6HI53-11,6HI53-15,5HJ54-1,5HJ54-2,5HJ54-5,           537-543
C     5HJ54-6,5HJ54-9,6HJ54-10,6HJ54-13,6HJ54-14,6HJ54-15,6HJ54-16/          544-550
 DATA(NUMBS(I),I=551,600)/3HI54,3HI50,1H ,3HH56,3HJ56,3HH55,3HG56,          551-557
A     3HE51,3HI55,3HG49,3HF55,3HF49,3HF54,3HD55,3HD52,3HJ53,3HI51,           558-567
B     3HI52,1H ,3HJ54,3HJ55,3HE54,3HH50,3HK54,3HI56,3HH53,3HK55,             568-577
C      3HG51,1H ,3HB57,1H ,3HB54,1H ,3HA56,3HB55,3HB56,1H ,3HA54,            578-588
D     2(1H ),6HJ55-10,6HJ55-13,6HJ55-14,6HJ55-15,6HJ55-16,6HJ55-12,          589-596
E     5HK55-1,5HK55-2,5HK55-3,5HK55-4                                 /      597-600
 DATA(NUMBS(I),I=601,739)/6HF53-14,5HD53-1,7HD53-3+4,5HF51-9,                601-604
A     5HF53-6,5HF50-1,1H ,6HC52-15,5HE53-3,6HE50-16,5HH53-7,5HG52-3,         605-612
B     5HD53-7,5HF53-2,6HF54-15,6HH54-15,6HD51-11,6HE53-11,5HE53-8,           613-619
C     6HD52-14,6HD51-15,6HD53-12,5HD55-9 ,5HE54-2,5HD54-4,6HG55-10           620-626
D     ,1H ,5HF55-2,6HE55-13,6HH54-11,8HC53-13+9,6HH56-11,5HH53-6,            627-633
E     6HI55-11,5HC53-1,7HF50-2+6,8H(J55-15),5HH55-2,5HE52-6,6HE54-10         634-640
F     ,6HG54-15,6HD52-12,7H(K55-2),1H ,5HF54-9,5HH56-1,6HE52-10,             641-647
G     6HE53-13,1H ,6HE53-10,6HF53-13,6HF56-14,6HI50-14,5HF52-7,              648-654
H     6HD53-14,6HF53-15,6HC54-15,5HF51-3,5HF54-3,6HC53-14,6HH50-15,          655-661
I     5HG50-1,7H(K55-1),5HH54-1,5HG52-4,5HI51-1,6HF52-11,5HF53-1,            662-668
J     5HE51-8,5HD52-5,3HREF,8HI51-7+11,5HG56-6,6HD52-10,1H ,6HC52-16         669-676
K     ,1H ,6HI51-10,1H ,6HD51-12,6HE51-11,1H ,6HE52-13,1H ,6HI50-11,         677-685
L     6HF50-11,5HF55-7,6HF52-12,6HE53-12,1H ,5HG50-3,5HE52-1,1H ,            686-693
M     5HD53-5,6HF52-16,5HD53-6,5HF53-5,5HG50-2,1H ,5HE52-9,6HF52-15,         694-701
N     6HF52-4 ,1H ,6HF54-13,5HD53-15,5HH53-2,6HE51-15,1H ,5HE53-5,           702-709
O     5HG54-9,5HF51-6,1H ,6HI53-4 ,5HG52-7,5HF56-9,5HF50-4,5HD53-8,          710-717
P     6HD51-16,5HG50-7,5HJ55-9,5HI51-5,5HE53-4,6HF56-13,6HD53-11,            718-724
Q     6HD54-15,5HF53-8,5HG53-4,5HJ54-4,5HH55-4,5HI53-2,5HJ55-3,              725-731
```

```
      R    5HG53-8,6HH50-16,6HE52-16,1H ,5HE53-9,6HF55-12,5HE53-2,1H /     732-739
       DATA(NUMBS(I),I=740,810)/5HH56-3,6HE53-14,6HG54-16,5HF52-3,          740-743
      A    6HC54-11,9HC54-11+12,6HC53-16,6HF50-15,1H ,6HE52-12,5HE52-4,     744-750
      B    6HI55-15,6HC53-15,5HE54-5,6HH54-12,6HH51-14,6HF50-13,3HREF,1H    751-758
      C    ,6HB54-16,5HC55-4,1H ,5HB54-7,5HB55-8,1H ,3(1H ),6HB54-12,1H ,   759-769
      D    7H(B56-3),1H ,6HB54-15,5HB55-2,1H ,7H(A55-8),7H(B56-2),5HC56-5    770-777
      F    ,5HC55-7,1H ,5HB55-4,1H ,5HC56-1,5HA5[illegible]8,5HF56-5,7H(C56-6)     778-785
      F    ,2(3HREF),7HH56-6+7,7HF55-3+4,7HE55-1+2,9HG56-14+15,7HD52-3+4,   786-792
      G    7HE54-1+2,1H ,7HI50-5+9,7HF50-3+7,3HREF,3(1H ),5HH49-4,1H ,      79[illegible] 802
      H    2(3HREF),7HK55-7+3,8HK55-9+10,6HK55-14,3HREF,1H ,3HREF      /    803-810
       DATA(NUMBS(I),I=811,830)/5HK55-5,5HK55-6,5HK55-7,5HK55-8,5HK55-9,    811-815
      A    6HK55-10,6HK55-12,6HK55-13,6HK55-16,5HJ55-4,5HJ56-1,5HI56-5,     816-82[illegible]
      B    5HI56-6,6HI56-10,6HI56-14,6HH56-14,6HH56-15,6HG56-10,6HG56-11    823-829
      C    ,6HF56-10                                                  /     830
       DATA(NUMBS(I),I=831,870)/3HF53,3HH54,3HD51,3HC53,3HG55,3HG54,3HE52   831-837
      A    ,3HF50,3HH49,3HH51,3HF52,3HG50,3HC52,1H ,3HD54,3HE53,3HH52,      838-847
      B    3HF51,3HG53,3HG52,3HI53,3HF56,3HD53,3HE50,3HC54,3HE55,3HA55,     848-857
      C    2(1H ),3HC57,3HH57,3HH58,1H ,3HC55,3HC56,5(1H )            /     858-870
       DATA(NUMBS(I),I=871,900)/3HC49,3HC50,3HC51,3HD49,3HD50,3HD56,3HE49   871-877
      A    ,3HE56,3HI49,3HJ49,3HJ50,3HJ51,3HJ52,3HK49,3HK50,3HK51,3HK52,    878-887
      B    3HK53,3HK56,3HG48,3HL54,3HL55,3HL56,3HJ57,3HI57,3HG57,3HF57,     888-897
      C    3HE57,3H    ,3H                                            /     898-900
       DATA(NUMBS(I),I=901,1000)/9HI50-10+14,9HI50-11+15,9HC53-15+16,       901-903
      A    8HI56-6+10,9HH56-10+11,9HD51-15+16,7HF56-5+9,3HREF,7HD53-7+8,     904-909
      B    2(1H ),7HF50-1+2,9HD53-11+12,7HD52-5+9,1H ,3HREF,9HC52-15+16,     910-917
      C    3HREF,8H(A55-11),8H(A55-10),5HB56-8,6HB56-12,3(1H ),3HD47,        918-926
      D    4(1H ),5HF50-5,6HG49-16,6HG50-13,6HH50-13,6HH50-14,5HF56-1,       927-936
      E    5HF56-2,5HF56-6,5HE55-3,5HE55-7,6HE55-11,5HD55-4,5HD55-5,         937-943
      F    5HC53-9,6HC53-13,6HC52-13,6HC52-14,5HD52-1,5HD52-2,5HD52-6,       944-950
      G    5HE51-1,5HE51-5,5HE51-9,6HE50-13,6HE50-14,6HE50-15,5HF49-4,       951-957
      H    5HF49-8,5HA54-7,6HA54-12,5HA55-9,5HC55-1,5HC55-9,6HC55-10,        958-964
      I    6HA55-14,5HA56-9,5HA55-6,6HA55-10,5HA55-7,6HA55-11,6HA55-15,      965-971
      J    5HB55-2,6HB55-16,6HA55-16,5HA55-8,6HA55-12,5HA56-5,6HA56-13,      972-978
      K    5HB56-1,6HB56-13,6HA56-14,6HA56-15,5HB56-2,5HB56-3,5HB56-4,       979-985
      L    5HB56-7,6HB56-10,6HB56-11,6HB56-14,6HB56-15,6HB56-16,5HC56-2,     986-992
      M    5HC56-3,5HC56-6,5HC56-7,6HC56-11,6HC56-15,3(1H )           /     993-000
       DATA(INDEX(I),I=1,20) / 4,36,465,304,671,757,786,787,000,797,000,    1- 11
      A    918,000,000,916,908,803,804,808,810                        /     12- 20
       DATA((NUMBZ(I,J),I=1,2),J=1,20)/10HG49-16/G50,3H-13,10HJ55-4/J56-,    11- 21
      A    1H1,10HA56-5/A55-,2H1?,10HF50-5/F49-,1H8,10HH52-13/I52,2H-1,      22- 52
      B    10HH53-13/I53,2H-1,10HH52-14/I52,2H-2,10HG56-2+3+6+,1H7,          61- 82
      C    10HE51-13/F51,2H-1,10HF49-12/F50,2H-9,10HF56-14/G56,2H-2,         91-112
      D    10HD53-16/E53,2H-4,10HF49-16/F50,3H-13,10HC53-15/D53,2H-3,        121-142
      E    10HC53-14/D53,2H-2,10HE50-16/F50,2H-4,10HJ54/K54/J5,              151-171
      F    10H5-13/K55-1,10HJ55-15+16/,7HK55-3+4,10HI53-16/J53,              172-191
      G    10H-1/I54/J54,10HD53-7+8+11,3H+12                          /      192-202
       DATA NAR / 9HAUSTRALIA,1H ,9HNEW SOUTH,5HWALES,8HVICTORIA,1H ,        11- 32
      A    10HQUEENSLAND,1H ,10HSOUTH AUST,5HRALIA,10HWESTERN AU,            41- 61
      B    7HSTRALIA,8HTASMANIA,1H ,10HNORTHERN T,8HERRITORY,9HPAPUA NEW,    62- 91
      C    6HGUINEA,10HA.C.T. / N,5H.S.W.,10HN.S.W. / Q,5HLD,10HN.S.W. /     92-121
      DV   ,7HICTORIA,10HQLD / N.S.,2HW.,10HVICTORIA /,7H W.S.W.,           122-142
      E    9HAUSTRALIA,8H/ P.N.G.,10HSOLOMON IS,5HLANDS,10HTASMAN SEA,1H     151-172
      F    ,10HINDIAN OCE,2HAN,10HSOUTHERN O,4HCEAN,10HANTARCTICA,1H    /    181-202
       DATA MAPS / 9H15.0 15.0,9H60.0 90.0,8HZ50000.0,9H1000000.0,5H 40.0
      A    ,5H 10.0 /,MAPR / 10HAUSTRALIAN,9H NATIONAL,9HSPHEROID,
      B    9H6378160.0,6HZ98.25,9H-183. 6.0,1H ,10HPLANE PROJ,6HECTION,
      C    2(1H ),3H CM,1H ,3H CM /
       DATA NDA,NDS,NDR /   000,1000000,20000000 /
C . . . TEST ARRAY KORN FOR THE TYPE OF INFORMATION.
       DECODE (30,10,KORN) KAR
   10  FORMAT (30R1)
       NUM=LET=LIM=J=0 $ MAP=3
```

```
          DO 50 I=1,30 $ NEW(I)=1R
          IF(KAR(I).EQ.1R )GO TO 50
          IF(KAR(I).GT.1RZ)GO TO 20 $ LET=LET+1 $ GO TO 40
   20     IF(KAR(I).GT.1R9)GO TO 30 $ NUM=NUM+1 $ GO TO 40
   30     IF(NUM*LET.EQ.0) GO TO 50 $ IF(LIM.EQ.0)KAR(I)=1R-
          LIM=LIM+1
   40     J=J+1 $ NEW(J)=KAR(I)
   50     CONTINUE
      IF(LIM>LET.GT.0)GO TO 60
C . . . ONLY NUMBERS WERE FOUND SO KORN IS CORNER LIMITS.
      MAP=1 $ RETURN
C . . . LEFT JUSTIFY THE NAME READY FOR THE REFERENCE SEARCH.
   60 IF(NUM.NE.0)GO TO 120 $ CALL LJUST (KORN,3)
C . . . CHECK FOR A SINGLE WORD NAME.
      IF(KORN(2).NE.1H )GO TO 80
          DO 70 L=1,600 $ I=L
          IF(KORN(1).EQ.NM1(I))GO TO 160
   70     CONTINUE
      GO TO 270
C . . . CHECK FOR A DOUBLE WORD NAME.
   80 J=1 $ IF(KORN(3).NE.1H )GO TO 100
          DO 90 L=601,900 $ I=L
          IF(KORN(1).NE.NM2( J ))GO TO 90
          IF(KORN(2).EQ.NM2(J+1))GO TO 160
   90     J=J+2
      GO TO 270
C . . . CHECK FOR A TREBLE WORD NAME.
  100     DO 110 L=901,1000 $ I=L
          IF(KORN(1).NE.NM3( J ))GO TO 110
          IF(KORN(2).NE.NM3(J+1))GO TO 110
          IF(KORN(3).EQ.NM3(J+2))GO TO 160
  110     J=J+3
      GO TO 270
C . . . CHECK FOR SINGLE WORD NUMBER.
  120 ENCODE (30,10,KORN) NEW
      IF(KORN(2).NE.1H )GO TO 140
          DO 130 L=1,1000 $ I=L
          IF(KORN(1).EQ.NUMBS(I))GO TO 160
  130     CONTINUE
      GO TO 270
C . . . CHECK FOR DOUBLE WORD NUMBER.
  140     DO 150 L=1,100
          IF(KORN(1).NE.NUMBZ(1,L))GO TO 150
          IF(KORN(2).NE.NUMBZ(2,L))GO TO 150
          I=INDEX(L) $ GO TO 160
  150     CONTINUE
      GO TO 270
C . . . DECODE THE KEY WORD AND RESET THE NAMES.
  160 KI=KEY(I) $ IF(KI.EQ.0)GO TO 280
      IF(KI.LT.NDR)GO TO 170 $ I=KI/NDA-2000 $ KI=KEY(I)
  170 L=1 $ IF(JECTOR.EQ.3)L=2
          DO 180 J=1,7
          MAPL(J)=1H  $ NAIM(J)=MAPR(J,L)
  180     CONTINUE
      NS=KI/NDS $ KI=KI-NS*NDS $ NAIM(8)=MAPS(NS)
      NA=KI/NDA $ L=I/601 $ M=I/901
      J=I-1+L*(I-601)+M*(I-901) $ K=1+L+M
C . . . DEFINE THE MAP NAME.
          DO 190 L=1,K
  190     MAPNAME(L)=NAMES(J+L)
C . . . DEFINE THE MAP AREA.
      MAPAREA(1)=NAR(1,NA) $ MAPAREA(2)=NAR(2,NA)
```

```
      KORN(3)=1H  $ IF(NA.LE.15.AND.NA.NE.9)KORN(3)=NAR(1,1)
C . . . DEFINE THE MAP NUMBER.
      MAPNUM(1)=NUMBS(I) $ IF(MAPNUM.NE.3HREF)GO TO 210
C . . . SEARCH THE CROSS REFERENCE LIST FOR A TWO WORD SHEET NUMBER.
           DO 200 J=1,100
           IF(I.NE.INDEX(J))GO TO 200
           MAPNUM(1)=NUMB2(1,J) $ MAPNUM(2)=NUMB2(2,J) $ GO TO 210
  200      CONTINUE
      GO TO 290
C . . . IF THERE IS NO MAP NAME THE MAP NUMBER IS USED AS THE NAME.
  210 IF(MAPNAME.NE.1H )GO TO 230
           DO 220 J=1,2
           MAPNAME(J)=MAPNUM(J) $ MAPNUM(J)=1H
  220      CONTINUE
C . . . DEFINE THE MAP LIMITS.
  230      DO 240 J=1,2 $ KI=LIMIT(J,I)
           K=KI/NDA $ L=KI-K*NDA $ M=L/100
           XY(1,J)=K*60.0+M $ XY(2,J)=XY(1,J)+(L-M*100)*5.0
  240      CONTINUE
      TLT=XY(1,1) $ BLT=XY(2,1) $ WLN=XY(1,2) $ ELN=XY(2,2)
      MAP=2 $ KORN(1)=KORN(2)=1H $ INMAIN=1
C . . . DEFINE THE SPHEROID PARAMETERS.
      IF(JECTOR.EQ.3)GO TO 260 $ NAIM(7)=MAPS(NS+2)
      IF(JECTOR.EQ.1)RETURN
      IF(JECTOR.EQ.5)GO TO 265
C . . . STANDARD PARALLELS FOR CONIC PROJECTIONS.
      SP2=10.0*NS**2 $ SP1=TLT+SP2 $ SP2=BLT-SP2
      I=SP1/60.0 $ J=SP1-I*60.0 $ K=SP2/60.0 $ L=SP2-K*60.0
  245 ENCODE (10,250,NAIM(6)) I,J,K,L $ RETURN
  250 FORMAT (2(I3,I2))
C . . . SCALE FACTORS FOR PLANE PROJECTION.
  260 NAIM(4)=NAIM(6)=MAPS(NS+4) $ RETURN
C . . . CENTRAL PARALLEL FOR MERCATOR PROJECTION.
  265 SP2=(TLT+BLT)/2 $ I=SP2/60.0 $ J=SP2-I*60.0 $ K=L=0 $ GO TO 245
C . . . THE NAME ON THE MAPPLOT CARD CAN NOT BE RECOGNIZED.
  270 PRINT 275,KORN $ RETURN
  275 FORMAT(4H0***,10X,*UNRECOGNIZED MAP NAME ON MAPPLOT CARD = *,3A10)
C . . . INCORRECT PARAMETERS IN THE REFERENCE TABLES.
  280 PRINT 285,I $ RETURN
  285 FORMAT(4H0***,10X,*ILLEGAL KEY WORD AT *I4* IN REFERENCE TABLE *)
  290 PRINT 295,I
  295 FORMAT(4H0***,10X,*CROSS REFERENCE INDEX *I4* IS MISSING *)
      END
      OVERLAY(1,2)
      PROGRAM BAHMAP6
C * * * CONTROLS PLOT AND CONTOUR MAPS OF ANOMALIES AND HEIGHTS FROM NEW * * * C
C * * * STANDARD DATUM PRINCIPAL FACTS TAPES INPUT ON UNIT LX.  04/08/72 * * * C
C * * * THERE ARE SIX TYPES OF CONTROL CARD. ONLY TYPE 1 IS NECESSARY. * * * * C
C - - -  1. *BAPLOT , *FAPLOT , *HTPLOT OR *XYPLOT - TYPE OF DATA WANTED - - - C
C - - -  2. *TITLE - PROVIDES 72 CHARACTER TITLE BENEATH MAP IN 2 LINES. - - - C
C - - -  3. TAPE HEADER - IS CHECKED WITH INPUT TAPE UNLESS BLANK OR ABSENT. - C
C - - -  4. *SURVEY - DEFINES IDENTIFIERS AND SYMBOLS FOR GIVEN SURVEY NOS.- - C
C - - -  5. *BORDER - BORDER WIDTH (MINS.),DENSITY,MARKSIZE,DEFAULT SYMBOL.- - C
C - - -  6. *SEGMENT(S) - STATION,BA,HT FREQUENCIES AND AREA NO. ONE/AREA. - - C
C - - - - - - - - - THIS EDITION OF PROGRAM DATED 06/07/78  - - - - - - - - - C
C          -          -          - DEFINE   ARRAYS  -          -          -      C
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/MAPPLOT/XSPACE,YSPACE,KSIZE,SCALE,ISM
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
```

```
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      COMMON/AF:COS/KARJP,IBFJP,JARNO,KARNO,NOAR,IFOUND,JARK
      COMMON/AREFIN/IN,IOUT,NAMECR(10),LPS,LSS,LDM,LAM,LOS,LV
      COMMON/CONSTAN/SIXTY, P3048, TEN, PI, NTL, NXN
      COMMON/DATUM/IDATUM,JDATUM,KDATUM,GDATUM,IDATE,DEFAULT,LANK,SCRUB
      COMMON/MATCH/MORD,NOBS,MATS(11),METS(3),MATA(4),MAT(2),NCN,NOCS
      COMMON/STOREA/BUFF(2400),C(960),LW(11467),NSURVEY(100),NUSED(101),
     1   IDENT(100),MARK(100),LABEL(100),NYA,NYC,NYMBL(18),MAPDAT(3),LY,
     2   MAPTITL(8),RATA(12),N,NAMEMT(12),NAMEAR(12),STN,NDM,FACTS(9),ID
      DIMENSION MARKIND(12),KONTROL(12),GNC(4),LEVU(3),MILI(4),LEGA(12)
      DIMENSION MAGRAV(3),NAMEC(8),MAFREE(3),MAPNAM(8),OHR(3)
      EQUIVALENCE (ITM,IDATUM),(NAMEC,NATM)
C * * * MARKS = SQR,+,TRIANGLE,BOW,X,DI   JND,Y,OCTAGON,Z,BIGSQR,STAR,DOT * * * C
      DATA (MARKIND = 0,3,2,12,4,5,9,1,8,10,11,46)
      DATA (NYA=4HNONE),(NYB=7HDEFAULT),(NYC=5HBLANK),(NYD=8HNOT USED)
      DATA (NYMBL=6HSQUARE,1H+,8HTRIANGLE,3HBOW,1HX,7HDIAMOND,1HY,
     1            7HOCTAGON,1HZ,10HBIG SQUARE,4HSTAR,3HDOT,6(1H ))
      DATA (KONTROL=5H*BAPL,5H*FAPL,5H*HTPL,5H*XYPL,5H*TITL,5H*SURV,
     1            5H*BORD,5H*SEGM,5H*DATU,5H*GRID,5H*REST,5H*MAPP)
      DATA (MAPNAM=10H BOUGUER  ,9HANOMALIES,10H FREE AIR ,9HANOMALIES,
     1            10H      HEIG,9HHTS       ,10H    STATIO,9HN PLOT   )
      DATA (LEGA=8HLEGEND  ,10HA1234 ; ST,10HATION NUMB,2HER,
     1           8H        ,10HAVITY ANOM,3HALY,10H 456  ; GR,
     2           10HOUND ELEVA,4HTION,10H 567. ; GR,10H 56.7 ; GR)
      DATA (LEVU=8H FEET   ,8H METRES ,8HUNITS   ),(LANK=8H         )
      DATA (MAFREE=10HFREE AIR A,10HNOMALY AT ,4HSEA.)
      DATA (MAGRAV=10HPROVISIONA,10HL 1973 VAL,4HUES.)
      DATA (MILI=10HMILLIGALS ,10HMICROMETRE,1H ,10HS/SEC/SEC )
      DATA (DEFAULT=-9.9E5),(SCRUB=9.9E9),(NTL=0),(NXN=6000),(LYM=12)
      DATA (GNC=978031.8,978049.0,0.0053024,0.0052884)
      DATA (SIXTY=60.0),(P3048=0.3048),(TEN=10.0),(PI=3.1415926536)
      DATA (OHR=0.0,1.027,0.917)
C          -          - BUFFER IN TAPENAME - SET COUNTERS.  -          -          C
      NB=N=M=NEWY=UNITS=1.0 $ NOSU=NOSY=NOSEG=NOAR=KON=KEY=NOTES=NUBSA=0
      LY=JARNO=KARNO=KARJP=IBFJP=JARK=MFA=NZ=NODAS=NOR=MD=ME=0
      BORDER=15.0 $ RHO=2.67 $ NUBSA(2)=1000 $ INDENT=0.00B $ NKO=-2
      JDATUM=IDATUM=6H=DM MO $ METS(1)=8H(8F10.6) $ METS(2)=METS(3)=LANK
      SUM=RANGE=AVERAG=0 $ SCAX=-SCRUB $ SCAY=SCRUB $ NAB=2999
C          -          READ CONTROL CARD, MAP AND TAPE NAMES.          -          C
      IF(INMAIN.EQ.0)GO TO 10 $ INMAIN=0
      DECODE(80,15,NAMEC)MUF,NAMECR $ GO TO 20
   10 IF(NOAR.EQ.2)GO TO 560
      READ 15,MUF,NAMECR,NAMEC
   15 FORMAT(A5,T1,10A8,T1,8A10)
      IF (EOF(60)) 555,20
   20 NRE(3)=8HBAP/11,
         DO 25 I=1,12
         IF(KONTROL(I).NE.MUF)GO TO 25
         KON=I  $  IF(KON.LE.4) GO TO 30
         IF(KON.LE.9.AND.KEY.EQ.0)GO TO 560
         GO TO (40,65,80,90,86,550,597,550)KON-4
   25    CONTINUE
      IF(KEY.NE.0) 50,560
C - - - - -  -          FIRST CONTROL CARD          -          - - - - - C
   30 MEASURE=NAMECR(3) $ IF(MEASURE.EQ.LANK)MEASURE=6HMETRES
      KEY=KON $ IF(MEASURE.EQ.4HFEET)UNITS=.P3048 $ N=-1 $ NAME=NAMECR
      YA = SIXTY  $ LDM=0 $ IF(NAMECR(4).EQ.7HMINUTES)LDM=YA=1.0
      LAM=MARSTN=0 $ IF(NAMECR(5).EQ.6HMARINE)MARSTN=LAM=1RM
      DECODE(80,35,NAMEC(2))LX,L
   35 FORMAT (I2,2X,I1)
      IF(LX.EQ.0)LX=40 $ IF(KEY.LE.2)L=9 $ LL=L-2
      IF(L.GE.8)LL=2*L-13 $ IF(L.LE.0)LL=5-2*(KEY/4)
```

```
      CALL ORICHEK (2) $ REWIND LD $ I=UNIT(LD)
      CALL BEGINIT $ IF(KEY.EQ.4)GO TO 10
      IN=LX $ CALL LUNSET (1,LX)  $  CALL RECORD (1,LX)
      IF(N)10,540,10
C - - - - -          READ MAPTITLE CARD          -          - - - - - C
   40     DO 45 I=1,7
   45     MAPTITL(I) = NAMEC (I+1)
      LY=1 $ GO TO 10
C          -          -          -  CHECK TAPENAME  -          -          -          C
   50 IF(M.EQ.2)RETURN
         DO 55 I=1,9
         IF(NAMECR(I).EQ.LANK)GO TO 55
         IF(NAMECR(I).NE.NAMEMT(I))GO TO 580
   55    CONTINUE
      PRINT 135 $ PRINT 60,NAMEMT $ GO TO 10
   60 FORMAT(1H0,* INPUT TAPE NAMED - *10A8,2X,A8,*  19*A8)
C - - - - -          -          SURVEY AND SYMBOL DATA          -          - - - - - C
   65     DO 75 J=2,8 $ IF(NOSY.GE.100) GO TO 10
         IF(NAMEC(J).EQ.1H )GO TO 75 $ NOSY=NOSY+1
         DECODE(10,70,NAMEC(J))NSURVEY(NOSY),IDENT(NOSY),MARK(NOSY),
     1                          LABEL(NOSY)
   70    FORMAT (I4,R2,2I2)
         IF(MARK(NOSY).GT.LYM)MARK(NOSY)=LYM
         IF(MARK(NOSY).LT.0)MARK(NOSY)=-1
   75    CONTINUE $ NOSEG=0 $ GO TO 10
C - - - - - BORDER IN MINUTES, DENSITY IN GM/CC, LABEL SIZE IN 1/10 INCH - - - C
C - - - - - DEFAULT SYMBOL NUMBER, PLOT BLOCK SIZE, MARINE FREE AIR. - - - - - C
   80 DECODE(80,85,NAMEC)BORDER,RHO,ME,MSIZE,NOMARK,NORM,MD,INK,MFA
   85 FORMAT(10X,2F5.2,I1,I4,2I5,I1,2I2)
      GO TO 10
C - - - - -          -          -          NEW DATUM          -          -          - - - - - C
   86 IDATUM=NAMECR(2) $ GO TO 10
C - - - - -          -          LIST PARAMETERS USED FOR MAP          -          - - - - - C
   90 IF(NOSEG.NE.0) GO TO 100  $ NOSEG=1 $ NYM=LYM
      IF(NOSY.LE.0)GO TO 95 $ PRINT 135 $ PRINT 91
   91 FORMAT(1H0,13X,* SURVEY  LABEL  SYMBOL      LABELLING *)
         DO 94 J=1,NOSY $ MARKNUM=NYMBL(MARK(J))
         ENCODE(10,92,MUF)LABEL(J)
   92    FORMAT (*EVERY *I2)
         IF(LABEL(J).LT.0)MUF=NYA $ IF(LABEL(J).EQ.0)MUF=LANK
         IF(MARK(J).LT.0)MUF=NYD
         PRINT 93,NSURVEY(J),IDENT(J),MARKNUM,MUF
   93    FORMAT (16X,I4,4X,R2,4X,A10,2X,A8)
   94    CONTINUE
   95 IF(NOMARK.GT.LYM)NOMARK=LYM $ IF(NOMARK.LT.0)NOMARK=NYM--1
      IF(NORM.EQ.0)NORM=10000  $ IF(KEY.EQ.2)RHO=0.0 $ ME=1-ME
      IF(MSIZE.EQ.0)MSIZE=1   $ CALL MATCH (IDATUM)
      IF(INK.LT.1.OR.INK.GT.4)INK=1 $ CALL NEWPEN (INK)
      MUF=NYMBL(NOMARK) $ IF(NOMARK.EQ.0)MUF=NYB
      PRINT 96,RHO,BORDER,MSIZE,MUF,IDATUM
   96 FORMAT(1H0,10X,30(1H*)/1H0,10X,* DENSITY = *F5.2* GM/CC.(0 IS *,
     1*FREE AIR)*5X*BORDER = *F5.2* MINUTES.  LABEL SIZE = *I2/1H0,11X,
     1*THE SYMBOL FOR UNLISTED SURVEYS IS - *,A10,6X,*THE DATUM IS *,A8)
      IF(MFA.NE.0)PRINT 97,MAFREE $ M=2
   97 FORMAT (1H0,10X,* BOUGUER ANOMALY ON LAND AND *,3A10)
C          -          -          -          SET  VARIABLES -          -          -          C
      MG = 17 - IBYTE(ITM,0,6,3)
      MH = IBYTE(ITM,0,3,3)/6 $ IF(MH.EQ.1)UNITS=P3048
      GF=10**NOGS $ IF(KEY.LT.4)GO TO 98
      NCN=7 $ IF(LDM.EQ.0)GO TO 99 $ NCN=9
      METS(1)=10H(F10.6,2(F $ METS(2)=10H4.0,F6.4), $ METS(3)=7H5F10.6)
      GO TO 99
```

```
   98 IF(NAMEMT(11).NE.LANK.AND.NAMEMT(11).NE.0)JDATUM=NAMEMT(11)
   99 TLIMIT = TLT - BORDER  $  BLIMIT = BLT + BORDER
      WLIMIT = WLN - BORDER  $  ELIMIT = ELN + BORDER
      SIZE= KSIZE $ WSIZE=0.07*KSIZE $ QSIZE=0.07*MSIZE
      XSIZE=0.025*SIZE $ YSIZE=0.02*SIZE $ IF(INMAIN.GE.1)GO TO 102
C - - - - -         -     DEFINE LABELLING FREQUENCIES      -          - - - - - C
  100 DECODE(80,101,NAMEC )K,MYSTN,MYBAN,MYHGT,NAMECR(2)
  101 FORMAT(8X,R1,1X,3I3,1X,A8)
  102 PRINT 135
      ANS=0.3 $ MYLAB=0 $ IF(MYSTN+MYBAN+MYHGT.LE.0)MYLAB=1
      IF(MYSTN.GT.0)GO TO 104 $ MYSTN=10000000 $ ANS=ANS-0.1
  104 IF(MYBAN.GT.0)GO TO 105 $ MYBAN=10000000 $ ANS=ANS-0.1
  105 IF(MYHGT.GT.0)GO TO 106 $ MYHGT=10000000 $ ANS=ANS-0.1
  106 ISNFREQ=MYSTN $ IBAFREQ=MYBAN $ IHTFREQ=MYHGT
      IF(MYLAB.EQ.1)MYSTN=MYBAN=MYHGT=1
      PRINT 107,ISNFREQ,IBAFREQ,IHTFREQ
  107 FORMAT(1H0,10X*DEFAULT LABEL FREQUENCY - STATION*I3* .   ANOMALY*,
     1       I3* .   HEIGHT*I3)
      ANI=0.4*MSIZE $ IF(ANS.LE.0.001)ANS=0.3 $ ANS=ANS*MSIZE
      IF(XR/ANI.GT.100.0)ANI=XR/100.0 $ IF(YT/ANS.GT.114.0)ANS=YT/114.0
C - - - - -         -          AREA BLOCK LOCATION.          -          - - - - - C
      NOAR=2  $  IF(K.EQ.1RS) GO TO 110  $  NOAR = 1
      IF(NAMECR(2).NE.LANK)GO TO 110
      READ 15,MUF,NAMECR
      NOAR = 0  $  IF (EOF(60))  540,110
  110 IFEND=0 $ PRINT 135 $ IF(KEY.EQ.4)GO TO 120
      CALL AREFIND  $  IF(IFOUND.LE.1) 10,150
  120 CALL BCDFIND (LX) $ IF(IFOUND.LE.1)10,130
  130 PRINT 140,(NAMEAR(J),J=1,10) $ UNITS=1.0
  135 FORMAT (1H0,10X,30(1H*))
  140 FORMAT(1H0,10A8/1H0)
C - - - - -         -      SKIP THIS BLOCK ON INPUT UNIT.     -          - - - - - C
  150 IF(K.NE.1R-)GO TO 200 $ PRINT 160
  160 FORMAT (1H0,9X,* THIS BLOCK HAS BEEN SKIPPED *)
      IF(KEY.EQ.4)GO TO 180 $ CALL SKIPF (2,IN) $ GO TO 10
  180 CALL BCDSKIP $ GO TO 10
C           -           -            INPUT STATION DATA           -          -          C
  200     IF(KEY.NE.4)GO TO 230
          READ (LX,METS) STATION,(FACTS(I),I=1,NCN)
          IF(EOF(LX)) 380,210
  210         DO 220 J=1,NCN
              IF(FACTS(J).NE.0.0)GO TO 220
              IF(LDM.EQ.1.AND.(J.EQ.1.OR.J.EQ.3))GO TO 220
              IF((FACTS(J).AND.77B).NE.0)FACTS(J)=DEFAULT
  220         CONTINUE
          IF(STATION.LT.1.0)STATION=ABS(STATION)+1.0
          IF(LDM.EQ.0)GO TO 240
          ALT=FACTS(1)*SIXTY+FACTS(2) $ ALN=FACTS(3)*SIXTY+FACTS(4)
          ELEV=FACTS(5) $ OBSGRAV=FACTS(6) $ HEIGHT=FACTS(LL+2)
          TC=FACTS(8) $ GO TO 245
  230     CALL RECORD (3,LX) $ IF(N)235,380,230
  235     CALL DASHIFT (JDATUM)
          STATION=STN $ MARSTN=IBYTE(NDM,0,8,3).OR.LAM
  240     ALT = FACTS(1)*YA  $  ALN = FACTS(2)*YA
          ELEV   = FACTS(3)  $  OBSGRAV = FACTS(4)
          HEIGHT = FACTS(LL) $  TC = FACTS(6)
C         -           -          IS DATA WITHIN LIMITS.          -          -          C
  245     IF(ALT.LE.TLIMIT.OR.ALT.GE.BLIMIT)GO TO 200
          IF(ALN.LE.WLIMIT.OR.ALN.GE.ELIMIT)GO TO 200
          INBORD=0
          IF(ALT.LT.TLT.OR.ALT.GT.BLT)INBORD=1
          IF(ALN.LT.WLN.OR.ALN.GT.ELN)INBORD=1
```

```
      CALL COORDS $ XX=XX-XL $ YY=YY-YB
      X=SR*YY+CR*XX $ Y=CR*YY-SR*XX
C     -       -          FIND STATION SYMBOL          -       -       C
      IF(NOR.LT.NORM)GO TO 250 $ CALL PLOT (0.0,0.0,-3)
      WRITE (LT,247) NB,INK $ CALL TAGBLOK
  247 FORMAT (* STATION BLOCK *I3,* PEN *I1,6X)
      NB=NB+1 $ NOR=0
  250 ISURVEY = STATION  $  MARKNUM=NYM
          DO 255 NT = 1,NOSY   $  NS=NOSY-NT+1
          IF ( ISURVEY .EQ. NSURVEY(NS) ) GO TO 275
  255     CONTINUE $ NS=101 $ IF(NOSY.GE.100)GO TO 280
      IF(NOMARK.LT.0)GO TO 200
      IF(INBORD.EQ.1)GO TO 280
      NOSY=NS=NOSY+1 $ IDENT(NS) = INDENT + NEWY $ MARK(NS)=NOMARK
      IF(NOMARK.EQ.0)MARK(NS)=NS-(NS-1)/9*9
      NSURVEY(NS)=ISURVEY
      IF(NEWY.LT.26)GO TO 260 $ NEWY=0 $ INDENT=INDENT+100B
  260 NEWY=NEWY+1 $ MUF=NYMBL(MARK(NS)) $ LABEL(NS)=0
      PRINT 265,ISURVEY,IDENT(NS),MUF
  265 FORMAT(1H0,20X,*SURVEY *I5* GIVEN IDENT *R2* AND SYMBOL *,A10)
      MUF=NYMBL(MARKNUM) $ IF(NS.EQ.100)PRINT 270,MUF
  270 FORMAT(* 100 SURVEYS HAVE BEEN USED, ANY OTHERS WILL BE*,
     1       *INDICATED BY *A4,* AND NOT LISTED *)
  275 MARKNUM=MARK(NS)
  280 IF(MARKNUM.LT.0)GO TO 200 $ IF(L.LT.8)GO TO 305
C     -       -          BOUGUER ANOMALY WANTED       -       -       C
      IF(ELEV.EQ.DEFAULT.OR.OBSGRAV.EQ.DEFAULT)GO TO 300
      IF(MFA.NE.0.AND.MARSTN.EQ.1RM)GO TO 285
      IF(HEIGHT.EQ.DEFAULT.AND.RHO.NE.0.0)GO TO 300
  285 IF(TC.EQ.DEFAULT)TC=0.0  $  TC=TC*RHO/2.0
      HEIGHT=HEIGHT*UNITS $ ELEV=ELEV*UNITS
      ANGLE = ALT * PI / 10800.0
      P = SIN(ANGLE) $  Q = SIN(2*ANGLE)
      GRAVN = GNC (MG) * (1.0 + GNC(MG+2)*P**2 - 5.9E-6*Q**2) + GF
      A=OHR(18/MARSTN+1) $ B=0.3086
      IF(ELEV.LT.0.0.AND.MARSTN.EQ.1RM)B=0.2223
  290 ANOM=OBSGRAV -GRAVN +ELEV *B  *GF
      IF(MFA.NE.0.AND.MARSTN.EQ.1RM)GO TO 295
      IF(RHO.NE.0)ANOM=ANOM+TC-0.0419*GF*(HEIGHT*(RHO-A)+ELEV*A)
  295 VALUE = ANOM $ GO TO 310
C     -       -       STATION WITH MISSING DATA.      -       -       C
  300 IF(INBORD.EQ.1)GO TO 200 $ NODAS=NODAS+1 $ NOR=NOR+3
      IF(MARKNUM.EQ.0)GO TO 200 $ IF(MARKNUM.GE.LYM)GO TO 375
      IF(MD.NE.0)CALL SYMBOL (X,Y,0.04*SIZE,6,0,-1) $ GO TO 200
C     -       -          GROUND  HEIGHTS WANTED       -       -       C
  305 IF(HEIGHT.EQ.DEFAULT)GO TO 300
      IF(MH.EQ.1)HEIGHT=HEIGHT/P3048
      VALUE = HEIGHT
C     -       -        STORE POSITION AND VALUE       -       -       C
  310 NKQ = NKQ + 3  $  NQ = NKQ + 3
      XYW(NKQ  ) = ALN - WLN
      XYW(NKQ+1) = ALT - TLT
      XYW(NKQ+2) = VALUE $ NOSU=NOSU+1 $ SUM=SUM+VALUE
      IF(SCAX.LT.VALUE)SCAX=VALUE $ IF(SCAY.GT.VALUE)SCAY=VALUE
      IF(NQ.GE.NAB)GO TO 400
  320 IF(INBORD.EQ.1)GO TO 200 $ NOTES=NOTES+1
C     -       -          PLOT AND LABEL POINT         -       -       C
      IF(NUSED(NS).EQ.0)NUSED(NS)=1
      IF(MARKNUM.EQ.0)GO TO 200 $ IF(MARKNUM.GE.LYM)GO TO 375
      NT=2 $ IF(MARKNUM.GE.10)NT=4 $ NOR=NOR+3 $ LINE=-MSIZE
      CALL SYMBOL (X,Y,NT*YSIZE,MARKIND(MARKNUM),0,-1)
      MYLAB=LABEL(NS) $ IF(MYLAB.LT.0)GO TO 200
```

```
          NX=X/ANI+1 $ NSM=Y/ANS+1 $ NT=NX+NSM*100
          IF(LW(NT)+LW(NT+1)+LW(NT-100)+LW(NT-99).NE.0)GO TO 20
          JSNFREQ=ISNFREQ $ IF(MYLAB.GT.0)JSNFREQ=MYLAB*MYSTN
          IF(MOD(NOTES,JSNFREQ).NE.0) GO TO 340
          ENCODE (10,330,JSTAT) STATION $ LINE=LINE+MSIZE
  330     FORMAT (4PF10.1)
          ENCODE(9,335,MUF)IDENT(NS),JSTAT $ NUSED(NS)=2
  335     FORMAT(X,R2,R6)
          Z=X $ IF((IDENT(NS).AND.7700B).EQ.5500B)Z=X-QSIZE
          CALL SYMBOL (Z,Y,QSIZE,MUF,0,7) $ NOR=NOR+10
  340     IF(L.LT.8)GO TO 360
          JSNFREQ=IBAFREQ $ IF(MYLAB.GT.0)JSNFREQ=MYLAB*MYBAN
          IF(MOD(NOTES,JSNFREQ) .NE. 0) GO TO 360
          IF(NOGS.EQ.0)ENCODE (6,350,MUF)ANOM  $  LINE= LINE + MSIZE
  350     FORMAT (F6.1)
          IF(NOGS.NE.0) ENCODE(6,355,MUF) ANOM
  355     FORMAT (F6.0)
          CALL SYMBOL (X,Y-0.1*LINE,QSIZE,MUF,0,6) $ NOR=NOR+10
  360     JSNFREQ=IHTFREQ $ IF(MYLAB.GT.0)JSNFREQ=MYLAB*MYHGT
          IF(MOD(NOTES,JSNFREQ) .NE. 0) GO TO 370
          ENCODE (8,365,MUF)HEIGHT  $ LINE = LINE + MSIZE
  365     FORMAT (F8.1)
          CALL SYMBOL (X,Y-0.1*LINE,QSIZE,MUF,0,6) $ NOR=NOR+10
  370     IF(LINE.LT.-0.01)GO TO 200
          LW(NT)=LW(NT+1)=LW(NT-100)=LW(NT-99)=ME $ GO TO 200
  375     CALL SYMBOL (X-XSIZE,Y-YSIZE,2*YSIZE,MARKIND(12),0,-1)
          NOR=NOR+1 $ GO TO 200
  380     PRINT 385,NOSU $ NOSU=0
  385     FORMAT (1H0,9X,* NUMBER OF STATIONS USED - *I10)
          IF(NQ.LT.NAB)GO TO 440
  390     IFEND=1 $ IF(NKQ.LT.0)GO TO 440
C - - - - -       -     TRANSFER DATA BLOCK TO UNIT LD     -          - - - - - C
  400     NZ=NZ+1 $ NUBSA(3)=(NQ-1)/3 $ NKQ=-2 $ NUBSA=NUBSA+NUBSA(3)
          BUFFER OUT (LD,1) (XYW,XYW(3000))
  410     I=UNIT(LD) $ IF(IFEND)440,320
  440     CONTINUE
        GO TO (10,10,110,445)NOAR+1
C         -         -         -   END SEGMENTS   -         -         -         C
C         -         -         -        ++       -         -         -         C
C         -         -         -   ANNOTATE MAP   -         -         -         C
  445 NDB=NZ  $  OFFSET = 2.1 * SIZE $ ANI=-1.0 $ NSM=25 $ NX=7
      WRITE (LT,247) NB,INK $ IF(INK.EQ.1)GO TO 446
      CALL NEWPEN(1) $ CALL TAGBLOK $ CALL PLOT(0.0,0.0,-3)
  446 YSIZE=YSIZE*2.0 $ MAPDAT=10HSURVEY KEY
      CALL SYMBOL (OFFSET+0.15*SIZE,ANI-0.6*SIZE,YSIZE,MAPDAT,0,10)
      MAPDAT=10HSYMBOL LET $ MAPDAT(2)=10HTER BMR NU $ MAPDAT(3)=4HMBER
      CALL SYMBOL (OFFSET-0.1*SIZE,ANI-0.7*SIZE,YSIZE,MAPDAT,0,24)
          DO 460 NS=1,NOSY $ IF(NUSED(NS).EQ.0)GO TO 460
          NX=NX+1 $ ANS=ANI-SIZE*0.1*NX
          IF(NUSED(NS).NE.2)IDENT(NS)=LANK
          ENCODE(12,450,MAPDAT)IDENT(NS),NSURVEY(NS)
  450     FORMAT(R2,I10)
          CALL SYMBOL (OFFSET+0.15*SIZE,ANS,YSIZE,MAPDAT,0,12)
          MARKNUM=MARK(NS) $ IF(MARKNUM.EQ.0)GO TO 455
          NT=1+MARKNUM/10-MARKNUM/12
          CALL SYMBOL (OFFSET,ANS+XSIZE,NT*YSIZE,MARKIND(MARKNUM),0,-1)
  455     IF (NX.LT.NSM) GO TO 460
          NSM=NSM+20 $ ANI=ANI+2.0*SIZE $ OFFSET=OFFSET+SIZE
  460     CONTINUE
      IF(LY.NE.0)GO TO 462 $ IF(KEY.EQ.4)GO TO 464
      LY=2*KEY-1 $ IF(IMMAIN.EQ.12)LY=7
      ENCODE (30,461,MAPTITL(5)) MAPNAM(LY),MAPNAM(LY+1) $ GO TO 463
```

```
  461 FORMAT (5X,2A10,5X)
  462 CALL SYMBOL (0.5*XR-2.8*SIZE,-.75*SIZE,2*WSIZE,MAPTITL(1),0,40)
  463 CALL SYMBOL (0.5*XR-2.1*SIZE,-1.0*SIZE,2*WSIZE,MAPTITL(5),0,30)
  464 IF(MG.GE.2)GO TO 465
      CALL SYMBOL (XR-1.4*SIZE,-SIZE,WSIZE,MAGRAV,0,24)
  465 GO TO (470,480,490,485) KEY
  470 IF(ISM.EQ.0.OR.RHO.NE.2.67)GO TO 473 $ MAPDAT=10HGRAVITY SE
      MAPDAT(2)=4HRIES $ CALL SYMBOL(SIZE,YT+0.5*SIZE,WSIZE,MAPDAT,0,14)
  473 ENCODE (24,475,MAPDAT)RHO $ GO TO 500
  475 FORMAT (*DENSITY =*,F5.2,* GMS/CC.  *)
  480 MAPDAT=10HFREE  AIR  $ MAPDAT(2)=10H ANOMALY  $ MAPDAT(3)=LANK
      GO TO 500
  485 MAPDAT(1)=MAPDAT(2)=MAPDAT(3)=LANK $ GO TO 500
  490 MAPDAT=10HGROUND  HE $ MAPDAT(2)=10HIGHTS  IN $ MAPDAT(3)=LEVU(MH)
      PRINT 505, LANK , MAPDAT
  500 CALL SYMBOL (XR-1.4*SIZE,-1.1*SIZE,WSIZE,MAPDAT,0,30)
      MAPDAT=LEVU(3)
      CALL SYMBOL (-0.15*SIZE,-2.75*SIZE,WSIZE,MAPDAT,0,5)
      MAPDAT(2)=LEVU(MH) $ MAPDAT=9HHEIGHT  ;
      PRINT 505,LEVU(3),MAPDAT(1),MAPDAT(2)
  505 FORMAT (1H0,10X,4A10)
      CALL SYMBOL (-0.40*SIZE,-2.90*SIZE,WSIZE,MAPDAT,0,18)
      IF (KEY.GE.3) GO TO 510
      MAPDAT=9HGRAVITY ; $ MAPDAT(2)=MILI(1+NOGS)
      MAPDAT(3)=MILI(3+NOGS) $ PRINT 505,LANK,MAPDAT
      CALL SYMBOL (-0.40*SIZE,-3.05*SIZE,WSIZE,MAPDAT,0,25)
      IF (NOGS.NE.0) CALL SYMBOL (999.0,-3.02*SIZE,0.6*WSIZE,1H2,0,1)
      IF(MFA.NE.0)CALL SYMBOL(XR-1.4*SIZE,-1.2*SIZE,WSIZE,MAFREE,0,24)
C - - - - -       -       -   DRAW LEGEND.   -       -       - - - - - C
  510 CALL PLOT (-0.4*SIZE,-1.8*SIZE,3)
      CALL PLOT (1.6*SIZE,-1.8*SIZE,2)
      CALL PLOT (1.6*SIZE,-2.5*SIZE,2)
      CALL PLOT (-.4*SIZE,-2.5*SIZE,2)
      CALL PLOT (-.4*SIZE,-1.8*SIZE,2)
      CALL SYMBOL (0.2*SIZE,-1.96*SIZE,WSIZE,LEGA,0,6)
      CALL PLOT (0.2*SIZE,-1.98*SIZE,3)
      CALL PLOT (0.6*SIZE,-1.98*SIZE,2)
      CALL SYMBOL (-0.25*SIZE,-2.18*SIZE,WSIZE,LEGA(2),0,24)
      IF(KEY.GE.3)GO TO 512 $ LEGA(5)=LEGA(12-NOGS)
      CALL SYMBOL (-0.25*SIZE,-2.31*SIZE,WSIZE,LEGA(5),0,24)
  512 CALL SYMBOL (-0.25*SIZE,-2.44*SIZE,WSIZE,LEGA(8),0,24)
      PRINT 135 $ NOST=NOTES
      PRINT 515,NOTES,NUBSA(1),NODAS
  515 FORMAT(1H0,9X,*NUMBER OF USABLE STATIONS PLOTTED ON MAP IS *I6/
     1       10X,*NUMBER OF USABLE STATIONS IN MAP AND BORDER *I6/
     2       10X,*NUMBER OF STATIONS ON MAP WITH MISSING DATA *I6)
      PRINT 520,NB $ IF(NUBSA.GT.0)AVERAG=SUM/NUBSA
  520 FORMAT (1H0,9X,I3,* STATION PLOT BLOCKS WRITTEN *)
      CALL FINIT $ REWIND LD $ RANGE=SCAX-SCAY
      WRITE (LT,522) $ CALL TAGBLOK $ OLAT=TLT $ OLON=WLN
  522 FORMAT (* MAP ANNOTATION *,14X)
      CALL PLOT (0.0,0.0,-3)
      IF(NDB.NE.0) GO TO 530   $  PRINT 525    $  GO TO 590
  525 FORMAT(1H-,20X,* NO DATA FOUND FOR THIS MAP*)
  530 RETURN
C         -         -         -   INPUT ERRORS   -         -         -         C
  540 PRINT 545,LX $ STOP
  545 FORMAT (4H0***,* EOF READ FOR TAPENAME ON UNIT - *I3)
  550 INMAIN=KON $ IF(NOSEG.NE.0)GO TO 555
      K=1RS $ MYSTN=MYBAN=MYHGT=1 $ GO TO 90
  555 IF(KEY.NE.0)GO TO 560 $ NAME=6HNEWMAP $ GO TO 590
  560 NOAR=3 $ IF(KEY.NE.0)GO TO 390
```

```
      PRINT 570 $ GO TO 590
  570 FORMAT (4H0***,* CONTROL CARD MISSING IN BAHMAP *)
  580 PRINT 585,(NAMEMT(I),I=1,10),NAMECR
  585 FORMAT (4H0***,* WRONG TAPE - *10A8/10X,* SHOULD BE - * 10A8)
  590 CALL ORICHEK (6) $ RETURN
C - - - - -         -        RESTART CARD WAS READ        -          - - - - - C
  597 NAME=NAMEC(2) $ PRINT 598,NAME
  598 FORMAT (1H0,20(1H*),*RESTART *,A8)
      END
      SUBROUTINE AREFIND
C * * * SEARCHES UNIT IN FOR AREA GIVEN BY NAMECR.                    T * * * C
      COMMON/ARECOS/KARJP,IBFJP,JARNO,KARNO,NOAR,IFOUND,JARK
      COMMON/AREFIN/IN,IOUT,NAMECR(10),LPS,LSS,LDM,LAM,LOS,LD
      COMMON/DATUM/IDATUM,JDATUM,KDATUM,GDATUM,IDATE,DEFAULT,LANK,SCRUB
      COMMON/STOREA/AB(15372),N,NAMEMT(12),NAMEAR(12),DATA(12)
      IFOUND=0
      NAMEAR(11)=JDATUM   $ NAMEAR(12)=IDATE
 1020 IF(IBFJP.NE.0)GO TO 1070
      CALL RECORD (2,IN)
      JARNO=JARNO+1
 1030 IF(N) 1070,1040,1140
 1040 N=UNIT(IN) $ REWIND IN $ CALL RESET(2)
      CALL RECORD (1,IN) $ JDATUM=NAMEMT(11) $ CALL DACHEK (JDATUM)
      JARNO=IBFJP=0 $ IF(NOAR.EQ.2)GO TO 1065
      IF(JARNO.NE.KARNO.OR.JARK.EQ.0)GO TO 1020
 1050 PRINT 1060,NAMECR
 1060 FORMAT(4H0***,* AREA NAMED - *10A8* - IS NOT ON THE TAPE.*)
 1065 KARNO=JARNO+1   $ KARJP=IFOUND=1 $ JARK=0
      RETURN
 1070 IF(KARJP.EQ.1)GO TO 1080
      IF(JARNO.NE.KARNO)GO TO 1080
      IBFJP = JARK  $  IF(JARK.NE.0) GO TO 1050  $  JARK =1
 1080 KARJP = IBFJP = 0  $  IF(NOAR.NE.0) GO TO 1110
          DO 1090 J=1,9
          IF(NAMECR(J).NE.NAMEAR(J))GO TO 1130
 1090     CONTINUE $ IFOUND=2
 1100 PRINT 1105,NAMEAR $ KARNO=JARNO+1 $ JARK=0
 1105 FORMAT(1H0,11A8,A10/1H0)
      RETURN
 1110 IF(NOAR.EQ.2)GO TO 1120
      IF(NAMEAR(1).NE.NAMECR(2))GO TO 1130
 1120 IFOUND=3
      GO TO 1100
 1130 CALL RESET(2)
      IF(UNIT(IN))1135,1020,1135
 1135 CALL SKIPF(2,IN)
      GO TO 1020
 1140 PRINT 1150,NAMEAR $ GO TO 1020
 1150 FORMAT (* ERROR OCCURRED IN ARE' 'ME (AREAFIND). NAME USED - *,
     1         20X,12A8)
      END
      SUBROUTINE PARIX(K)
      COMMON/ARECOS/KARJP,IBFJP,JARNO,KARNO,NOAR,IFOUND,JARK
      COMMON/AREFIN/IN,IOUT,NAMECR(10),LPS,LSS,LDM,LAM,LOS,LD
      COMMON/STOREA/AB(15372),N,NAMEMT(12),NAMEAR(10),DATA(14)
      DATA (NEW=1)
      PRINT1420,K $ N=4
 1420 FORMAT(4H0***,* PARITY ERROR ON INPUT UNIT - *,I2)
 1430 RETURN
      ENTRY BCDSKIP
 1450 READ (K,1500) NAMEAR $ IF(EOF(K))1460,1450
 1460 RETURN
```

```
      ENTRY BCDFIND
      IFOUND=0 $ IF(NEW.EQ.0.OR.K.EQ.60)GO TO 1530
      REWIND K $ READ (K,1500) NAMEAR $ IF(EOF(K))1510,1530
 1500 FORMAT (10A8)
 1510 PRINT 1520,K $ RETURN
 1520 FORMAT (*0 NO DATA ON UNIT *,I2)
 1530 READ (K,1500) NAMEAR $ IF(EOF(K))1540,1560
 1540 IF(NOAR.EQ.0)PRINT 1550,NAMECR
 1550 FORMAT (4H0***,* SEGMENT NAMED - *,10A8,* - IS NOT ON THE FILE*)
      IF(NOAR.EQ.1)PRINT 1550,NAMECR(2),(NAMECR(10),I=1,9)
      IFOUND=NEW=1 $ RETURN
 1560 IF(NOAR.EQ.2)GO TO 1580 $ IF(NOAR.EQ.1)GO TO 1590
          DO 1570 I=1,9
          IF(NAMECR(I).NE.NAMEAR(I))GO TO 1600
 1570     CONTINUE
 1580 IFOUND=3 $ RETURN
 1590 IF(NAMECR(2).EQ.NAMEAR(1))GO TO 1580
 1600 READ (K,1500) NAMEAR $ IF(EOF(K))1530,1600
      END
      SUBROUTINE FLOTE (S,I)
      COMMON/CONSTAN/SIXTY,P3048,TEN,PI,NTL,NXN
      COMMON/DATUM/IDATUM,JDATUM,KDATUM,GDATUM,IDATE,DEFAULT,LANK,SCRUB
      DIMENSION I(2),MN(2)
      MN(1)=I(1)  $  MN(2) = I(2)
      S=0
      IFN = SHIFT(IDATUM,6).AND.77B
      K=7  $  IF ( IFN .EQ. 45B) K= 5
          DO 4050 J = 1,4
          IJ=(IBYTE(MN,0,J,3)-27).AND.77B $ IF(IJ.EQ.22B)IJ=0
          IF(IJ.GT.9)GO TO 4070
 4050     S=S*10+IJ
          DO 4060  J=K, 10
          IJ=(IBYTE(MN(2),0,J,3)-27).AND.77B
          IF(IJ.EQ.22B)IJ=0 $ IF(IJ.GT.9)GO TO 4070 $ V=IJ
 4060     S=S+V/10**(J-K+1)
      RETURN
 4070 IF (K .NE. 5) GO TO 4080
      MN(2) = SHIFT (MN(2),12) + 5555B
 4080 PRINT 4085,MN $ S=-1 $ RETURN
 4085 FORMAT(4H0***,* ILLEGAL STATION NUMBER IN BCD INPUT DATA - *A4,R6)
      ENTRY EIGHTEN
      MN=S $ V=S-MN $ S=MN+V/100
      RETURN
      END
      SUBROUTINE DASHIFT (NM)
C * * * PERFORMS DATUM SHIFT TO DATA IN COMMON BLOCK STOREA TO DATUM IDATUM. * C
      COMMON/CONSTAN/SIXTY,P3048,TEN,PI,NTL,NXN
      COMMON/DATUM/IDATUM,JDATUM,KDATUM,GDATUM,IDATE,DEFAULT,LANK,SCRUB
      COMMON/MATCH/MORD,NOGS,MATS(11),METS(5),MATA(2),MAT(2),NCN,NOCS
      COMMON/STOREA/AB(15397),MSTN(2),EACH(6),KNEE(4)
      DIMENSION STN(2),KIN(4),CON(3),CST(4)
      DIMENSION KAR(4,6)
      EQUIVALENCE (CON,SIXTY)
      EQUIVALENCE (STN,MSTN),(ITM,IDATUM),(NTM,NE),(KEM,MSTN(2))
      DATA (MATS=4H(X,F,2(1H ),8H,X),F9.2,1H ,8H,2X,F9.2,1H ,8H,2(2X,A8,
     1           8H),2X,3A8,7H,2X,R1,,6HX,A10)),(MAT=6H(F7.2,  4(F7.1,)
      DATA (METS=8H9.4,2X,2,8H11.6  ,2,8H(I4,F6.2,8H(1X,F9.4,8H,2X,F8.2)
      DATA (KIN=2,3,5,6),(CON=60.,.3048,10.)
      DATA (KAR=1H0,1H=,1H+,1H0,1HD,4(1HM),3(1HF),1H ,3(1HA),1HM,
     1          3(1HU),1HO,1HP,2(1HI))
      DATA(NEM=6H0MM MO)
      DATA (CST=1.0005118,979685.74,0.9994885,979671.86)
```

```
      DATA (MASK=777777007777000000000B)
C . . . . . ITM,NTM,KEM WERE PREVIOUSLY TYPE BYTE (/6), DIMENSION 8  . . . . . C
      J=NM.AND.MASK $ KJ=IDATUM.AND.MASK
      IF(J.EQ.KJ)GO TO 4135
      NE=NM $  NTM = IBYTE(NTM,KEM,4,1) $ NTM = IBYTE(NTM,KEM,8,1)
      IFN=SHIFT(MSTN,6).AND.77B $ IF(IFN.GT.26.AND.IFN.LT.37)NE=NEM
      IF(IBYTE(NTM,0,1,3) .NE. 27) GO TO 4105
      NTM = IBYTE(NTM,55B,4,4)
      NTM = IBYTE(NTM,55B,8,4)
      IF(IBYTE(ITM,0,1,3) .NE. 27) GO TO 4100
      IF(IBYTE(NTM,ITM,1,2) .EQ.0) GO TO 4110
      ENCODE(8,4095,MSTN) STN(1)
 4095 FORMAT (4PF8.0)
      MSTN(2) = SHIFT (MSTN(1),24)
      GO TO 4110
 4100 MSTN(2) = SHIFT (MSTN(2),24) .AND. 7777B
      CALL FLOTE(STN,MSTN)  $ GO TO 4110
 4105 IF(IBYTE(ITM,NTM,1,2) .GT. 0) CALL EIGHTEN(STN)
      IF(IBYTE(ITM,NTM,1,2) .GE. 0) GO TO 4110
      J=STN(1) $ STN(2)=STN(1)-J $ STN(1)=1.0*J + STN(2)/100.0
 4110     DO 4130  J=1, 4
          KJ=KIN(J)
          I = IBYTE(ITM,NTM,KJ,2)
          IF(I.EQ.0) GO TO 4130
          IF(J.LT.3)GO TO 4120
          IF(EACH(4).EQ.DEFAULT)GO TO 4130
          IF(J.EQ.3)EACH(4)=EACH(4)*10.0**I
          IF(J.NE.4)GO TO 4130
          EACH(4)=(EACH(4)-CST(3-I)*GF)*CST(2-I)*CST(I+3)*GF
          GO TO 4130
 4120     IF(EACH(2*J-1).NE.DEFAULT)EACH(2*J-1)=EACH(2*J-1)*CON(J)**I
          IF(EACH(3*J-1).NE.DEFAULT)EACH(3*J-1)=EACH(3*J-1)*CON(J)**I
 4130 CONTINUE
      IF(IBYTE(ITM,0,1,3) .EQ. 27) RETURN
      MSTN(2) = IDATUM
      KEM = IBYTE(KEM,NTM,4,1)   $   KEM = IBYTE(KEM,NTM,8,1)
      KNEE(4) = IDATE
      ENTRY DATHERE
 4135 KEM = IBYTE(KEM,0,7,4)
          DO 4150 J=1,6
          IF(EACH(J).GT.SCRUB  .OR. EACH(J).LT.-9E5) EACH(J)=DEFAULT
 4140     IF(EACH(J).EQ.DEFAULT) GO TO 4150
          IFN = IBYTE(KEM,0,7,3) + 2 ** (6-J)
          KEM = IBYTE(KEM,IFN,7,4)
 4150     CONTINUE
      RETURN
      ENTRY MATCH
C * * * * DEFINES FORMAT FOR PRINTING STATION DATA WITH A GIVEN DATUM. * * * * C
C . . . . . LISTING FORMAT FOR FINAL TAPE PRINTOUT . . . . . C
      MATS(1)=4H(X,F    $ MATS(8) = 8H,2(2X,A8
 4160 NE = NM  $ MATS(2) =METS(1)
      IF(IBYTE(NTM,0,1,5) .EQ. 1H+) MATS(2) = METS(2)
      MORD = 1
      IF(IBYTE(NTM,0,2,5) .EQ. 1HM) MORD = 0
      MATS(3)=METS(3+MORD)    $   NOCS = 4- 2*MORD
      NOGS = 1
      IF(IBYTE(NTM,0,5,5) .EQ. 1HM) NOGS = 0
      MATS(5)=MATS(6)+NOGS*10000B  $ MATS(7)=METS(5)+NOGS*10000B
      GF=10**NOGS
      RETURN
C . . . . . LISTING FORMAT FOR BOUGUER ANOMALIES . . . . . C
      ENTRY MATCHB
```

```
      MATS=4H(X,F   $  MATS(8)=8H,7(XF9.2   $  GO TO 4160
C . . . . . LISTING FORMAT FOR RAW PRINCIPAL FACTS . . . . . C
      ENTRY MATCHD
      MATS(1)=5H(A5,F   $ MATS(8)=8H,1(2X,A8   $  GO TO 4160
      ENTRY DACHEK
C * * * CHECKS DATUM FOR LEGALITY.
      NE=NM $ I=1 $ IF(NE.EQ.0.OR.NE.EQ.LANK)GO TO 4176
          DO 4170 J=1,6
              DO 4165 KJ=1,4
              IF(IBYTE(NTM,0,J,5).EQ.KAR(KJ,J)) GO TO 4170
 4165         CONTINUE $ I=0
 4170     CONTINUE
      IF(I.NE.0) RETURN   $ PRINT 4175,NE,NEM
 4175 FORMAT (4H0***,* ILLEGAL DATUM WORD - *A8*, DATUM USED IS - *A8)
 4176 NM=NEM
      RETURN
      END
      SUBROUTINE RECORD (M,LUN)
C * * * DOES THE INPUT/OUTPUT BETWEEN UNIT LUN AND ARRAY AB = A,B OR BUFF. * * C
C - - - ROUTINE WRITTEN FOR STANDARD DATUM TAPES. THIS EDITION 01/11/73. - - - C
      COMMON/AREFIN/IN,IOUT,NAMECR(10),LPS,LSS,LDM,LAM,LOS,LD
      COMMON/ARSORT/IKA,IKO,IKP,NAST,NEST,LUSK,LREC,INK,ITK
      COMMON/DREAD/ISTAT(2),IDAT(6),IND(5),MAL,LAR,MOD,NMS,NOVAR,NODATA
      COMMON/STOREA/BUFF(2400),B(12,80),A(12,1000),RATA(12),N,
     1              NAMEMT(12),NAMEAR(12),DATA(12)
      DIMENSION AB(15409)
      DIMENSION LI(6),IRL(6),LRL(6),NR(6),LNR(6)
      DIMENSION LIA(6),LBUF(3),LDE(5),LEK(3),LER(5)
      EQUIVALENCE (AB,BUFF)
      DATA (LIA=2400,0,4400,2400,1200,4400), (LBUF = 480,600,480)
      DATA (LDE=15397,15373,15385,15397,15360),(NOES=0)
      DATA (NR=6(0)),(LI=6(0)),(LEK=480,512,480),(LER=0,0,12,24,0)
      DATA (NOMAX=50)
C * * * INPUT FROM LUN. M=0 TO DATA, 1 NAMEMT, 2 NAMEAR, 3 DATA, 4 RATA. * * * C
      LVL=0 $ NB=IN
 4500 MM=M $ I=(M+5)/3 $ K=LDE(M+1) $ IP=1 $ N=-1
      IF(I.NE.2.OR.LUN.EQ.NB)GO TO 4505
      MM = IP = 0  $  I=1
      IF(LAR.NE.0)  IP = 1 - LAR/3   $    GO TO 4510
 4505 IF(LUN.NE.MAL)GO TO 4510
      KBUF=LBUF(2) $ KREC=LEK(2) $ LE=24 $ GO TO 4515
 4510 KREC = LEK(I)  $  LE=LER(MM+1)  $  KBUF = LBUF(I)
 4515 I=I+LVL $ LIX=LIA(I) $ LB=LI(I) $ IR=IRL(I) $ LR=LRL(I)
      IF(LVL)4700,4520
 4520 IF(LB)4550,4530,4580
 4530 IR=1+LIX $ LR=IR+KREC-1
 4540 AB(IR+10)=AB(IR+11)=0 $ J=UNIT(LUN)
      BUFFER IN (LUN,IP)(AB(IR),AB(LR)) $ IRL(I)=IR $ LRL(I)=LR
      IF(LB)4870,4550,4580
 4550 IF(UNIT(LUN)) 4560,4600,4610
 4560 LI(I) = LB = 1
      LEL =((LENGTH(LUN)+3)/12) * 12 + 3
      IF(N.EQ.4) GO TO 4620
      IF(LEL.GE.KREC)GO TO 4570 $ AB(IR+LEL)=-1.0 $ IF(LEL.LE.LE)LE=0
      KREC=LEL
 4570 NR(I)=IR+LE-13  $  LNR(I)=IR+KREC-13
      IR=2*LIX+KBUF-IR+2 $ LR=IR+KREC-1
      GO TO 4540
 4580 NR(I)=NB=NR(I)+12 $ IF(AB(1+NB).EQ.-2.0)GO TO 4640
      IF(M.EQ.1.OR.M.EQ.2.OR.LEL.LT.24)GO TO 4585
      IF(AB(1+NB).LT. 0.0)GO TO 4550
 4585     DO 4590 J=1,12
```

```
 4590      AB(J+K)=AB(J+NB)
      IF(NB+8.LT.LNR(I))RETURN
      LI(I) = -1
      RETURN
 4600 LI(I)=N=0 $ BUFF(11)=BUFF(12)=BUFF(23)=BUFF(24)=0
      RETURN
 4610 CALL PARIX (LUN) $ GO TO 4560
 4620 NB=IR+LEL-12 $ J=(NB-LIX)/12+1 $ PRINT 4630,J,AB(NB)
 4630 FORMAT(20X,*ERROR OCCURRED AFTER *I3*TH STATION, NUMBERED *F11.6)
      AB(IR+LEL)=-2.0 $ GO TO 4570
 4640 J=1 $ LI(I)=0 $ RETURN
      ENTRY OUTPUT
C * * * OUTPUT TO LUN. M HAS SAME VALUES AND MEANING AS FOR INPUT PHASE. * * * C
      LVL=3 $ NB=IOUT $ GO TO 4500
 4700 IF(LB)4720,4710
 4710 IRL(I)=IR=1+LIX $ LRL(I)=IR+KREC-1 $ NR(I)=IR+LE-13
      LNR(I)=IR+KREC-13
 4720 NR(I)=NB=NR(I)+12 $ LI(I)=LB=1
      IF (AB(1+K).GT.0.0.AND.AB(1+K).LT.10000.0) GO TO 4725
      IF(M.NE.1.AND.M.NE.2)GO TO 5050
 4725     DO 4730 J=1,12
 4730     AB(J+NB)=AB(J+K)
      IF(NB+8.LT.LNR(I))RETURN
      IF(LE .NE. 0) GO TO 4770
 4740 J = UNIT(LUN)
 4750 BUFFER OUT (LUN,IP) (AB(IR),AB(LR))
      IF(LB)4760,4820
 4760 IRL(I) = IR = 2+LIX+KBUF-IR+2  $ LRL(I)=IR+KREC-1
      NR(I)=IR+LE-13 $ LNR(I)=IR+KREC-13 $ LI(I)=-1 $ RETURN
 4770 AB(NR(I)+13)=-1.0  $  AB(NR(I)+14)=(KREC-LE)/12  $  GO TO 4740
      ENTRY LUNSET
C * * * FINISHES OUTPUT, ENDFILES, REWINDS, INITIALS AND/OR BEGINS INPUT * * *
      IF(M.EQ.5)GO TO 4880 $ IP=1
      I=(M+5)/3 $ LB=LI(I+3) $ LE=LER(I+I/2+2) $ KREC=LEK(I)
      KBUF = LBUF(I)
      IF(M-1)4800,4840
 4800 IF(LB)4820,4820,4810
 4810 J=I+3 $ LB=LI(J)=0 $ IR=IRL(J) $ LR=LRL(J)
      AB(NR(J)+13) = -1.0
      IF(LE.NE.0) AB(LRL(J)-6) = (KREC-LE-LRL(J)+NR(J)) / 12
      IF(NR(J).GE.(IR+LE-1))GO TO 4740
 4820 J = UNIT(LUN)
 4830 ENDFILE LUN
 4840 IF(M.EQ.2)GO TO 4870
      J = UNIT(LUN) $ REWIND LUN $ IF(I-2)4860,4850
 4850 LB=LI(I)=-1 $ LIX=LIA(I) $ GO TO 4530
 4860 LI(I)=0 $ BUFF(11)=BUFF(12)=BUFF(23)=BUFF(24)=0
 4870 LI(I+3)=0 $ RETURN
 4880 REWIND LUN $ LI(5)=0 $ RETURN
      ENTRY OLDMAT
      IF(M.NE.0) GO TO 5000  $  LEK(2)=12  $ LER(3)=LER(4)=0 $  RETURN
 5000 M=LEK(2) $ RETURN
      ENTRY SKIPF
      IR=IRL(M) $ LR=LRL(M)
 5010 IF(UNIT(LUN))5020,5030,5020
 5020 BUFFER IN (LUN,1)(AB(IR),AB(LR)) $ GO TO 5010
      ENTRY RESET
 5030 LI(M)=0 $ RETURN
C . . . . . ILLEGAL STATION NUMBER ON OUTPUT . . . . . C
 5050 PRINT 5060,AB(1+K) $ NOES=NOES+1
 5060 FORMAT(4HO***,* ILLEGAL STATION IN OUTPUT DATA - *G17.8* SKIPPED*)
      IF(NOES.LT.NOMAX)GO TO 5100
```

```
      PRINT 5070,NOES $ STOP
 5070 FORMAT(1H0/9H0********I5,* OUTPUT ERRORS DETECTED, PROBABLY DUE *,
     1      *TO UNDEFINED INPUT DATA TYPE */1H1,*EXECUTION TERMINATED.*)
      ENTRY ILLSET
      NOMAX=M $ IF(NOMAX.EQ.0)NOMAX=50
      RETURN
C . . . . . .       BACKSPACE BUFFER POINTER       .         . . . . . C
      ENTRY BACKREC
 5100 NR(I)=NR(I)-12 $ RETURN
      END
      INTEGER  FUNCTION IBYTE (TARGET,SOURCE,LOC,FN)                    IBY 001
C
      INTEGER BYTE                                                      IBY 002
      INTEGER TARGET,SOURCE,FN,T1,T2,T3,S1,S2,TS                        IBY 003
C
C*****    FUNCTION BYTE ALLOWS HANDLING OF 6-BIT BYTES
C*****    FUNCTIONS ARE
C               1.  TARGET(LOC) = SOURCE(LOC)
C               2.  COMPARE TARGET(LOC) AND SOURCE(LOC)
C                   RETURN   -1  FOR TARGET.LT.SOURCE
C                             0  FOR EQUAL
C                            +1  FOR TARGET.GT.SOURCE
C               3.  RETURN TARGET(LOC) RIGHT JUSTIFIED IN WORD
C               4.  REPLACE TARGET(LOC) WITH CONSTANT
C               5.  RETURN TARGET(LOC) LEFT JUSTIFIED WITH BLANK FILL
C
      IF(FN.LT.1 .OR. FN.GT.5)  GO TO 500                               IBY 004
C
C**      MASKS TO ISOLATE LOW-ORDER 6-BIT BYTE
      M = MASK(54)                                                      IBY 005
      NM = COMPL(M)                                                     IBY 006
C
C**   CALCULATE NUMBER OF LEFT SHIFTS TO PUT REQUIRED BYTE AT RHS
      I1 = LOC * 6                                                      IBY 007
      I2 = 60 - I1                                                      IBY 008
C
C**   SHIFT BYTE TO LOW ORDER END
      T1 = SHIFT(TARGET,I1)                                             IBY 009
C
      GO TO (100,200,300,400,450) ,FN                                   IBY 010
C
C**   ASSIGN SOURCE(LOC)  TO TARGET(LOC)
  100 T2 = AND (T1,M)                                                   IBY 011
      S1 = SHIFT(SOURCE,I1)                                             IBY 012
      S2 = AND (S1,NM)                                                  IBY 013
      TS = OR(T2,S2)                                                    IBY 014
      BYTE = SHIFT(TS,I2)                                               IBY 015
      IBYTE = BYTE                                                      IBY 016
      RETURN                                                            IBY 017
C
C**       FUNCTION 2: COMPARE TARGET WITH SOURCE
  200 T2= AND(T1,NM)                                                    IBY 018
      S1 = SHIFT(SOURCE,I1)                                             IBY 019
      S2=AND(S1,NM)                                                     IBY 020
      IF(T2.EQ.S2) GO TO 2010                                           IBY 021
      BYTE = (T2 - S2)/IABS(T2 - S2)                                    IBY 022
      IBYTE = BYTE                                                      IBY 023
 2005 RETURN                                                            IBY 024
 2010 IBYTE = 0                                                         IBY 025
      GO TO 2005                                                        IBY 026
C
C**       FUNCTION = 3:  RETURN BYTE RIGHT JUSTIFIED,ZERO FILLED
```

```
  300 T2 = AND(T1,NM)                                                    IBY 027
      BYTE = T2                                                          IBY 028
      IBYTE = BYTE                                                       IBY 029
      RETURN                                                             IBY  30
C
C**       FUNCTION = 4  ASSIGN CONSTANT VALUE TO TARGET
  400 IF(SOURCE.LT.0  .OR.  SOURCE.GT.63) GO TO 500                      IBY 031
      T2 = AND(T1,M)                                                     IBY 032
      T3 = OR(SOURCE,T2)                                                 IBY 033
      BYTE = SHIFT(T3,12)                                                IBY 034
      IBYTE = BYTE                                                       IBY 035
      RETURN                                                             IBY 036
C
C***      FUNCTION 5 -  LEFT JUSTIFIED, BLANK FILL
C
C***      CREATE MASK OF BLANKS
C
  450 M1 = 10H                                                           IBY 037
      MB = AND(M1,M)                                                     IBY 038
      T2 = AND(T1,NM)                                                    IBY 039
      T3 = OR(MB,T2)                                                     IBY 040
      BYTE = SHIFT(T3,54)                                                IBY 041
      IBYTE = BYTE                                                       IBY 042
      RETURN                                                             IBY 043
C
  500 PRINT 510                                                          IBY 044
  510 FORMAT(1H1,10X,* --- ERROR IN FUNCTION BYTE --- *)                 IBY 045
      PRINT 520, FN, SOURCE                                              IBY 046
  520 FORMAT (//20X,* FUNCTION = *,I10,/20X,* SOURCE = *,A10)            IBY 047
      RETURN                                                             IBY 048
      END                                                                IBY 049
      OVERLAY(2,0)
      PROGRAM GRID
C * * * THIS PROGRAM CONTROLS THE GRIDDING OF IRREGULARLY SPACED OBSERVATIONS. C
C * * * THE PARAMETERS OF THE GRID ARE DEFINED IN SUBROUTINE BLOCK.      * * * C
C * * * THE RAW GRID AND COEFFICIENTS ARE ESTABLISHED IN PROGRAM GRIDDY. * * * C
C * * * THE GRID VALUES ARE SMOOTHED BY ITERATIONS IN PROGRAM ITERATE.   * * * C
C * * * THIS EDITION WRITTEN FEB.75 TO INCLUDE RESTARTING FACILITY.      * * * C
C - - - LATEST EDITION OF PROGRAM IS 30-11-76. THIS VERSION DATED 01/08/78.
      COMMON/CONTROL/NAIM(3),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/GRIDB/DX,DY,IAX,JAM,JAR,JAXLST,NAP,NIGP,NLP,NUMPS,X0,Y0,YXR
      COMMON/ITERB/INDIV,KG,LE,LEUN,NAG,NIP,NAL,NLA,NBB,NO,F,FL,H
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/TT/NAME,TIMES,LLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      DIMENSION KOEB(6000,3),F(6000),FL(6000),H(6000)
      DIMENSION MC(20),MR(4),CO(4),LC(4)
      EQUIVALENCE (F,KOEB)
      INTEGER FL
      REAL LE,KOEB
      LE=1.0 $ LO=LR $ INDIV=1000*LE $ KG=120
      NRE(4)=8HGRD/9 , $ IF(NAME.EQ.7HITERATE)GO TO 500
C . . . DEFINE GRIDDING PARAMETERS FROM INPUT CARD OR ASSIGN DEFAULT VALUES. . C
      IF(INMAIN.NE.0)GO TO 25 $ READ 5,NAIM
    5 FORMAT (8A10)
      IF(EOF(60))15,10
   10 IF(NAIM.NE.8H*MAPPLOT)GO TO 25 $ INMAIN=1
   15 NAME=6HNEWMAP
   20 CALL ORICHEK (6) $ RETURN
   25 DECODE(80,30,NAIM)NAME,DY,YXR,NAG,NIP,NAL,NLA,I50,NBB,IPRNT
```

```
   30 FORMAT (A5,2F5.2,4I5,I1,2I2)
      INMAIN=0 $ IF(NAME.EQ.5H*GRID)GO TO 50
   35 PRINT 40,NAME $ GO TO 20
   40 FORMAT (1H1,10X,*GRID CARD ERROR.  FIRST WORD WAS - *,A5)
C . . . SET DEFAULT VALUES FOR ABSENT DATA AND LIST PARAMETERS.
   50 CALL BEGINIT $ CALL ORICHEK (3)
      IF(DY.EQ.0)DY=SQRT((BLT-TLT)*(ELN-WLN)/NOST)*0.5
      IF(YXR.LE.0.0)YXR=1.0
      IF(NIP.EQ.0)NIP=5 $ IF(NIP.LT.0)NIP=0
      IF(NAL.EQ.0)NAL=5 $ IF(NAL.LT.0)NAL=0
      IF(NLA.EQ.0)NLA=2 $ IF(NLA.LT.0)NLA=0
      PRINT 60,DY,YXR,NIP,NAL,NLA
   60 FORMAT(1H0,20X*LATITUDE GRID = *F5.2* MINUTES, GRID RATIO LAT/LONG
     1 = *         F5.3/1H0,20X*ITERATIONS - FAST = *I6*,  SLOW = *I6/
     2           21X,*NUMBER OF LINEAR SMOOTHING PASSES = *I6)
      NIP=NIP+NLA $ NAL=NIP+NAL
C . . . PRINT AND CHECK DATA PARAMETERS.
      PRINT 150,NUBSA(1),AVERAG,SCAX,SCAY
  150 FORMAT (1H0,5X,*NUMBER OF OBSERVATIONS = *I9*, AVERAGE VALUE = *,
     1       G12.2,*, MAXIMUM VALUE = *,G12.2,*, MINIMUM VALUE = *G12.2)
      IF(RANGE.GT.0.001)GO TO 170
      PRINT 160,RANGE $ GO TO 20
  160 FORMAT (* RANGE OF VALUES IS ONLY*F10.7*, THIS IS TOO SMALL TO*,
     1        * CONTOUR *)
C . . . CALL BLOCK TO CALCULATE GRID PARAMETERS AND LIST THESE IF REQUIRED.
  170 TIMES=TIMEF(CLOK)-TUN $ PRINT 180,TIMES
  180 FORMAT (1H0,19X,* THE JOB TIME IS *,-3PF10.3,* SECONDS *)
      NAME=5HBLOCK $ PRINT 190,NAME
  190 FORMAT (1H0,9X,* ENTERED *,A8/1X)
      CALL BLOCK
C . . . PRINT GRID PARAMETERS ON MAP AND LISTING.
      CALL PARAM (DX,DY,BORDER,NIP,NAL)
      PRINT 200,NAP,M1,L1,NIGP,IAX,JAM,NUMPS,JAR,JAXLST
  200 FORMAT (6X,*NUMBER OF GRID BLOCKS  =*I6*. MAP ORIGIN AT ROW *I3,
     1       * AND COLUMN *I3/6X,*NUMBER OF GRID POINTS  =*I6*. WIDTH *,
     2       *OF GRID =*I4*. HEIGHT OF GRID =*I4/6X,*NUMBER OF POINTS/*,
     3       *BLOCK =*I6*. HEIGHT OF BLOCKS =*I4*. HEIGHT OF BOTTOM *,
     4       *BLOCK =*I4)
      PRINT 210,DY,DX
  210 FORMAT(6X,*ACTUAL GRID SPACING IS *F7.4* MINUTES LATITUDE AND *,
     1       F7.4,* MINUTES LONGITUDE *)
C . . . OPEN RESTART TAPE AND CALL GRIDDY TO GRID THE DATA.
      LAREA=4HGRID $ REWIND LS $ I=UNIT(LS) $ REWIND LR $ I=UNIT(LR)
      BUFFER OUT (LS,1)(LAREA,RANGE)  $ I=UNIT(LS)
      BUFFER OUT (LS,1)( DX , YXR )   $ I=UNIT(LS)
      TIMES=TIMEF(CLOK)-TUN $ PRINT 180,TIMES
      NAME=6HGRIDDY $ PRINT 190,NAME $ LI=LS
      CALL OVERLAY (6HCONTOR,2,1)
      TIMES=TIMEF(CLOK)-TUN $ PRINT 180,TIMES
      IF(NBB.GE.0)GO TO 250
  220 PRINT 230
  230 FORMAT (1H0,9X,* JOB TERMINATED AFTER GRIDDING OF DATA.*)
  240 LOT=3HEND $ IF(IPRNT.LE.1)20,280
C . . . CALL ITERATE TO SMOOTH THE GRID SURFACE.
  250 NAME=7HITERATE $ PRINT 190,NAME
      CALL OVERLAY (6HCONTOR,2,2)
      IF(LOTX.LT.1000)GO TO 255 $ RETURN
  255 TIMES=TIMEF(CLOK)-TUN $ PRINT 180,TIMES
      IF(NBB.EQ.0)GO TO 270
      PRINT 260,NAL $ GO TO 240
  260 FORMAT (1H0,9X,* JOB TERMINATED AFTER *,I3,* ITERATIONS.*)
  270 LOT=NAME=5H*GRID $ IF(IPRNT.GT.1)280,460
```

```
C . . . PRINT THE GRID COEFFICIENTS IF REQUIRED.
  280 REWIND LI $ J=UNIT(LI) $ PRINT 290
  290 FORMAT (1H1,19X,*GRID COEFFICIENTS.*//1X,
     1        4(*  GRID ROW COL  COEFFICIENT POSN.*))
      NN=LL=0 $ JAX=JAR-2 $ NRX=NAP+1 $ M=2 $ K=1
          DO 360 NR=1,NRX $ NUMPS=IAX*JAX
              DO 300 L=1,M
              BUFFER IN (LI,1)(KOEB(1,L),KOEB(NUMPS,L)) $ J=UNIT(LI)
  300         CONTINUE $ M=3
          IF(NR.EQ.1)GO TO 350
              DO 340 J=K,JAX $ LL=LL+1
                  DO 330 I=1,IAX $ NC=(J-1)*IAX+I
                  IF(FL(NC).LE.0)GO TO 330
                  NN=NN+1 $ MC(NN)=I $ MR(NN)=LL
                  CO(NN)=H(NC) $ LC(NN)=FL(NC) $ IF(NN.LT.4)GO TO 330
                  PRINT 320,(MR(NN),MC(NN),CO(NN),LC(NN),NN=1,4) $ NN=0
  320             FORMAT (4(7X,2I4,F12.3,I5))
  330             CONTINUE
  340         CONTINUE $ K=3
  350     IF(NR.EQ.NAP)JAX=JAXLST
  360     CONTINUE
      IF(NN.NE.0)PRINT 320,(MR(J),MC(J),CO(J),LC(J),J=1,NN)
C . . . PRINT GRID VALUES
      PRINT 370
  370 FORMAT (1H1,5X,*GRID VALUES X 10. ROWS SHOWN AT LEFT ARE NUMBERED*
     1              * FROM TOP OF MAP.*)
      REWIND LI $ J=UNIT(LI) $ LL=M=NO=0
  380 NO=NO+1
          DO 390 I=1,2
          BUFFER IN (LI,1) (XYW,XYW(5)) $ J=UNIT(LI)
  390     CONTINUE
      JAX=JAR-4 $ IF(NO.EQ.NAP)JAX=JAXLST
      NUMPS=IAX*JAX $ LL=LL+JAX $ L=M+1 $ M=L+NUMPS-1 $ NRX=0
      BUFFER IN (LI,1) (KOEB(L),KOEB(M)) $ I=UNIT(LI)
      IF(LL.GT.2*JAR.OR.NO.EQ.NAP)400,380
  400 NR=30 $ IF(LL.LT.30)NR=LL $ LL=LL-NR
      NC=IAX/20 $ NN=1 $ NNN=20
          DO 440 I=1,NC
          PRINT 410,(N,N=NN,NNN)
  410     FORMAT (8X,20(1X,*C*,I3,1X))
              DO 430 J=1,NR $ MM=MM+1 $ K=NRX+(J-1)*IAX
                  DO 415 N=NN,NNN
  415             MC(N-NN+1)=KOEB(K+N)*10.0
              PRINT 420,MM,(MC(N),N=1,20)
  420         FORMAT (*0 R*,I4,1X,20I6)
  430         CONTINUE
          PRINT 435 $ NN=NN+20 $ NNN=NNN+20 $ MM=MM-NR
  435     FORMAT (1H1)
  440     CONTINUE
      MM=MM+NR $ NRX=NRX+NR*IAX $ IF(LL.GE.30)GO TO 400
      M=N=LL*IAX $ IF(NO.LT.NAP)GO TO 445 $ IF(LL.EQ.0)455,400
  445     DO 450 J=1,N
  450     KOEB(J)=KOEB(J+NRX)
      GO TO 380
C . . . END OF GRIDDING. RETURN TO CONTOR.
  455 TIMES=TIMEF(CLOK)-TUN $ PRINT 180,TIMES
  460 IF(I50.EQ.0)GO TO 495
C . . . WRITE OUTPUT GRID ON TAPE50.
      REWIND LI $ J=UNIT(LI) $ PRINT 465 $ NUMPS=IAX*JAR
  465 FORMAT (1H0,10X,*GRID FILE WRITTEN ON TAPE50.*)
  470 BUFFER IN (LI,1)(KOEB(1),KOEB(NUMPS)) $ IF(UNIT(LI))480,490,480
  480 NNN=LENGTH(LI)
```

```
      BUFFER OUT (50,1)(KOEB(1),KOEB( NNN )) $ J=UNIT(50) $ GO TO 470
  490 ENDFILE 50
      IF(LOT.EQ.3HEND)GO TO 20
  495 CALL FINIT $ RETURN
C       RELOAD MEMORY FOR RESTARTING ITERATIONS
  500 READ 30,NAME,ALT,ALN,NAG,NN,NNN,NC,I50,NBB,IPRNT
      IF(EOF(60))20,510
  510 IF(NAME.NE.5H*GRID)GO TO 35 $ CALL BEGINIT $ LI=40
      BUFFER IN (LI,1)(LAREA,RANGE)  $ IF(UNIT(LI))520,530
  520 BUFFER IN (LI,1)( DX , YXR )   $ IF(UNIT(LI))550,530
  530 PRINT 540,LI $ GO TO 20
  540 FORMAT (5X,* END OF FILE OR PARITY ERROR ON RESTART UNIT *,I3)
  550 IF(LAREA.EQ.4HGRID)GO TO 560 $ PRINT 555 $ GO TO 20
  555 FORMAT(6X*DATA IS IN CONTOUR FORMAT, NO MORE ITERATIONS ALLOWED*)
  560 NIP=NIGP $ NAL=NUMPS $ BORDER=X0 $ NLA=0
      IF(NN.NE.0)NIP=NN $ IF(NNN.NE.0)NAL=NNN
      PRINT 60,DY,YXR,NIP,NAL,NLA $ IF(ALT.EQ.0.0)GO TO 570
      CALL ORICHEK (3) $ CALL PARAM (DX,DY,BORDER,NIP,NAL)
  570 IF(NBB.GE.0)250,220
      END
      SUBROUTINE BLOCK
C * * * THIS ROUTINE CALCULATES THE PARAMETERS OF THE GRID.          * * * * * C
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/GRIDB/DX,DY,IAX,JAM,JAR,JAXLST,NAP,NIGP,NLP,NUMPS,X0,Y0,YXR
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      NLP=4 $ KFDIM=6000
      YD=BLT-TLT $ XD=ELN-WLN
  700 LD=XD*YXR/DY+0.5 $ DX=XD/LD $ SCAX=1/DX
      MD=YD/DY+0.5 $ DY=YD/MD $ SCAY=1/DY
      L1=LD*BORDER/XD+3.5 $ L2=L1+LD $ L3=L2+L1-1
      M1=MD*BORDER/YD+3.5 $ M2=M1+MD $ M3=M2+M1-1
      X0=(L1-1)*DX $ Y0=(M1-1)*DY $ MUL=(L3-1)/20+1
      IAX=MUL*20 $ IF(IAX.GT.500)GO TO 710 $ JAM=M3
      JAR=KFDIM/IAX $ IF(JAR.GT.JAM)JAR=JAM
      NUMPS=IAX*JAR
      NAP=(JAM-NLP-1)/(JAR-NLP)+1
      NAX=NAP*(JAR-NLP)+NLP
      JAXLST=JAR+JAM-NAX
      NIGP=IAX*JAM
      RETURN
C . . . GRID TOO FINE. INCREASE THE GRID SPACING.
  710 PRINT 720
  720 FORMAT(10X,*TOO MANY GRID POINTS ACROSS MAP*)
      DY=DY*2.0 $ PRINT 730,DY $ GO TO 700
  730 FORMAT(21X,*NEW GRID SPACING = *F9.2)
      END
      OVERLAY(2,1)
      PROGRAM GRIDBY
C * * * SETS GRID VALUES FROM OBSERVATIONS AND PREPARES FOR ITERATION. * * * * C
C - - - PROGRAM REWRITTEN TO AVOID OVERFLOWS AND INCREASE ACCURACY NOV 74. - - C
C - - - PROGRAM MODIFIED TO ALLOW SPREADING VALUES ACROSS BLOCK BORDERS. - - - C
C - - - LATEST EDITION OF PROGRAM 05/02/77. THIS VERSION DATED 04/11/77.
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),XYW(3000)
      COMMON/GRIDB/DX,DY,IAX,JAM,JAR,JAXLST,NAP,NIGP,NLP,NUMPS,X0,Y0,YXR
      COMMON/ITERB/INDIV,KG,LE,LEUN,NAG,NIP,NAL,NLA,NBB,NQ,F,FL,H
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      DIMENSION KOEB(6000,3),F(6000),FL(6000),H(6000)
      DIMENSION NN(4),U(4),V(4),Z(4,2),GL(6000),DN(6000)
      EQUIVALENCE (F,KOEB) , (DN,GL)
```

```
      INTEGER FL,GL
      REAL LESS,LARGE,LE
      DATA (Z=1.0,-1.0,-1.0,1.0,1.0,1.0,-1.0,-1.0)
      NN(1)=0 $ NN(2)=1 $ NN(3)=1+IAX $ NN(4)=IAX $ NRE(5)=8HGDY/9 ,
      LIMIT=1000 $ LARGE=1.0E12 $ LESS=1.0E6 $ IXS=0
      IF(IPRNT.GE.2)PRINT 20
   20 FORMAT(18X*NO*6X*LAT(MINS)LONG*6X*VALUE*8X*ROW    COLUMN  DELTA*,
     1        * X   DELTA Y GRID PT  QTR  TOTAL    AVERAGE*/1X)
C . . . FOR EACH GRID BLOCK SEARCH ALL THE DATA FOR OBSERVATIONS IN THE BLOCK.
      REWIND LQ $ IF(NAG.LE.0)NAG=5 $ IAXM=IAX-1 $ IAXMM=IAX-2
      IX4=IAX*NLP $ IX2=IX4/2 $ JAX=JAR $ JAXM=JAX-1 $ JAXMM=JAX-2
          DO 450 NO=1,NAP
          IF(NO.EQ.NAP)JAX=JAXLST
          NUMPS=IAX*JAX
          NU=NO*(JAR-NLP)+NLP
          NL=(NO-1)*(JAR-NLP)+1
              DO 30 L=1,NUMPS
              F(L)=LARGE $ FL(L)=0
   30         CONTINUE
              DO 40 L=IXS,NUMPS
   40         GL(L)=0 $ IXS=IX2+1
          K=0 $ REWIND LD $ L=UNIT(LD)
              DO 200 NZ=1,NDB
   50         BUFFER IN (LD,1) (XYW,XYW(3000))
              IF(UNIT(LD))60,50,50
   60         NUBS=NUBSA(2) $ IF(NZ.EQ.NDB)NUBS=NUBSA(3)
                  DO 190 NNN=1,NUBS $ N=3*NNN-2
                  YSH=XYW(N+1)+YO $ QY=YSH/DY
                  JY=QY $ JG=JY+1 $ J=JG-NL+1
                  JUD=(NU-JG-1)*(JG-NL) $ IF(JUD.LT.0)GO TO 190
                  XSH=XYW(N)+XO $ QX=XSH/DX
                  IX=QX $ I=IX+1 $ M=I+(J-1)*IAX
                  U(1)=QX-IX $ V(1)=QY-JY $ XYWI=XYW(N+2)
                  U(2)=U(3)=1.0-U(1) $ U(4)=U(1)
                  V(3)=V(4)=1.0-V(1) $ V(2)=V(1)
                  IF(IPRNT.GE.2)PRINT 80,NNN,XYW(N+1),XYW(N),XYWI,JG,I
   80             FORMAT (15X,I5,2F10.2,F12.3,2I9)
                  IF(I.LE.1.OR.I.GE.IAXM)GO TO 90
                  IF(J.LE.1.OR.J.GE.JAXM)90,110
   90             IF(IPRNT.GE.2)PRINT 100
  100             FORMAT (10X,*ABOVE OBSERVATION AT OR BEYOND EDGE*)
                  GO TO 190
  110                 DO 180 L=1,4
                      LU=((J-2)**2+((L-1)*(L-2))**2)
     1                  *((I-2)**2+((L-1)*(L-4))**2)
     2                  *((JAXMM-J)**2+((L-3)*(L-4))**2)
     3                  *((IAXMM-I)**2+((L-2)*(L-3))**2)
                      IF(LU.EQ.0)180,120
  120                 MT=NN(L)+M
                      IF(U(L).GT.0.5.OR.V(L).GT.0.5)GO TO 180
                      IF(GL(MT).LT.1000)GO TO 140
                      IF(IPRNT.GE.2)PRINT 130 $ GO TO 180
  130                 FORMAT(1H+70X* VALUE NOT USED - GRID PT HAS 1000*)
  140                 GL(MT)=GL(MT)+1 $ K=K+1
                      XY=(XYWI+(GL(MT)-1)*F(MT))/GL(MT)
                      IF(IPRNT.GE.2)PRINT 150,U(1),V(1),MT,L,GL(MT),XY
  150                 FORMAT (1H+,69X,2F10.5,I7,2I6,F12.3)
                      IF(GL(MT).EQ.1)GO TO 170
                      IF(ABS(F(MT)-XYWI).LT.0.1*RANGE)GO TO 170
                      PRINT 160,XYW(N+1),XYW(N),XYWI,F(MT),MT
  160                 FORMAT(16X*VALUE AT*F6.2* AND*F6.2* =*F10.3,1X,
     1                   *WHICH DIFFERS FROM THE EXISTING AVERAGE OF *,
```

```
     2                  F10.3,* AT GRID PT *I4)
   170               F(MT)=XY
                     NX=LIMIT*(0.5+U(L)*Z(L,1))
                     NY=LIMIT*(0.5+V(L)*Z(L,2))
                     FL(MT)=FL(MT)+NX*LESS+NY $ GO TO 190
   180               CONTINUE
   190           CONTINUE
   200       CONTINUE
         PRINT 220,K,NO,NAG
   220     FORMAT(10X,I6,* OBSERVATIONS USED FOR BLOCK *,I2,*. CONTOUR*,
     1          * CUT-OFF AT *I2* GRID SPACES *)
         LU=(IAX-4)*(JAX-4) $ N=NNN=0 $ IF(K.LE.0)GO TO 370
C . . . ASSIGN COEFFICIENT NUMBER AND WEIGHTING FACTOR FOR USED GRID POINTS.
   240       DO 250 M=IXS,NUMPS
             NV=GL(M) $ IF(NV.EQ.0)GO TO 250 $ NVLT=NV*LIMIT
             XY=FL(M)/LESS $ NX=XY $ NNN=NNN+1 $ N=N+NV
             X=XY/NVLT $ Y=(XY-NX)*LESS/NVLT
             NX=25*X+1 $ NX=NX-NX/26 $ SX=(NX-13)/25.0
             NY=25*Y+1 $ NY=NY-NY/26 $ SY=(NY-13)/25.0
             FL(M)=NX+(NY-1)*25
             H(M)=F(M) $ GL(M)=-1
             IF(FL(M).EQ.313)GO TO 250
             P=ABS(SX)+ABS(SY)
             H(M)=4*H(M)/(P*P*P)
   250       CONTINUE
         PRINT 260,N,NNN,NUMPS
   260     FORMAT(10X,I6* VALUES USED TO FIX *I4* GRID POINTS OUT OF*I5)
C . . . SPREAD VALUES FROM USED GRID POINTS TO COVER WHOLE BLOCK.
             DO 350 JAG=1,500 $ KAM=0
                 DO 345 N=IXS,NUMPS $ IF(GL(N).GE.0)GO TO 345
                 J=1+(N-1)/IAX $ I=N-(J-1)*IAX $ OBS=F(N)
                 IL=I-JAG $ IU=I+JAG $ JL=NX=J-JAG $ JU=NY=J+JAG
                 IF(IL.LT.2)IL=1 $ IF(IU.GT.IAXM)IU=IAX
                 IF(NX.LT.2)JL=1 $ IF(NY.GT.JAXM)JU=JAX
                     DO 300 II=IL,IU $ LAG=(II-I)**2+(JAG**2)
                     MM=II+(JL-1)*IAX $ ML=II+(JU-1)*IAX
                     IF(LAG+FL(MM))275,265,270
   265               NV=KOEB(MM,3)=KOEB(MM,3)+1
                     F(MM)=(F(MM)*(NV-1)+OBS)/NV $ GO TO 280
   270               IF(F(MM).LT.LARGE)GO TO 280
   275               KAM=1 $ F(MM)=OBS $ FL(MM)=-LAG $ KOEB(MM,3)=1
   280               IF(LAG+FL(ML))295,285,290
   285               NV=KOEB(ML,3)=KOEB(ML,3)+1
                     F(ML)=(F(ML)*(NV-1)+OBS)/NV $ GO TO 300
   290               IF(F(ML).LT.LARGE)GO TO 300
   295               KAM=1 $ F(ML)=OBS $ FL(ML)=-LAG $ KOEB(ML,3)=1
   300               CONTINUE
                 JL=NX+1 $ IF(JL.LT.2)JL=1 $ JU=NY-1
                 IF(JU.GT.JAXM)JU=JAX $ IF(JU.LT.JL)GO TO 345
                     DO 340 JJ=JL,JU $ LAG=( JJ-J )**2+JAG**2
                     MRM=(JJ-1)*IAX $ MM=IL+MRM $ ML=IU+MRM
                     IF(LAG+FL(MM))315,305,310
   305               NV=KOEB(MM,3)=KOEB(MM,3)+1
                     F(MM)=(F(MM)*(NV-1)+OBS)/NV $ GO TO 320
   310               IF(F(MM).LT.LARGE)GO TO 320
   315               KAM=1 $ F(MM)=OBS $ FL(MM)=-LAG $ KOEB(MM,3)=1
   320               IF(LAG+FL(ML))335,325,330
   325               NV=KOEB(ML,3)=KOEB(ML,3)+1
                     F(ML)=(F(ML)*(NV-1)+OBS)/NV $ GO TO 340
   330               IF(F(ML).LT.LARGE)GO TO 340
   335               KAM=1 $ F(ML)=OBS $ FL(ML)=-LAG $ KOEB(ML,3)=1
   340               CONTINUE
```

```
  345             CONTINUE
                IF(KAM.EQ.0)355,350
  350           CONTINUE
  355       PRINT 360,JAG
  360       FORMAT (10X,*SPREADING DISTANCE WAS *I2* GRID SPACES*)
C . . . SPREAD DOWN OVER BORDER.
  370       IF(NQ.EQ.1)GO TO 410
                DO 395 I=1,IAX $ II=GL(I) $ OBS=DN(I+IAX)
                IF(II.EQ.0)II=LESS $ IF(II.LT.0)II=0
                    DO 390 J=3,JAX $ MM=I+(J-1)*IAX
                    IF(II+FL(MM))385,375,380
  375               NV=KOEB(MM,3)=KOEB(MM,3)+1
                    F(MM)=(F(MM)*(NV-1)+OBS)/NV $ GO TO 390
  380               IF(F(MM).LT.LARGE)GO TO 395
  385               F(MM)=OBS $ FL(MM)=-II $ KOEB(MM,3)=1
  390               II=II+ISQRT(II)*2+1
  395           CONTINUE $ I=IAX+1 $ IF(K.NE.0)GO TO 405
                DO 400 II=I,IX2 $ F(II)=DN(II)
  400           FL(II)=-GL(II-IAX)
C . . . STORE LINES 2,3+4 ON LQ FOR SPREAD UPWARD.
  405       BUFFER OUT (LQ,1)(F(I),F(IX4)) $ II=UNIT(LQ)
            BUFFER OUT (LQ,1)(FL(I),FL(IX4)) $ II=UNIT(LQ)
C . . . STORE BOTTOM LINE FOR NEXT BLOCK IN GL AND DN.
  410       IF(NQ.EQ.NAP)GO TO 420 $ M=JAXMM*IAX
                DO 415 I=1,IAX $ DN(I+IAX)=F(M+I)
  415           GL(I)=-FL(M+I)
C . . . STORE BLOCK ON UNIT LR.
  420       H(NUMPS)=NNN $ H(NUMPS-1)=LU-NNN $ IF(NAP.EQ.1)GO TO 460
                DO 430 L=1,3
                BUFFER OUT (LR,1)(KOEB(1,L),KOEB(NUMPS,L)) $ I=UNIT(LR)
  430           CONTINUE
  450       CONTINUE
  460 REWIND LR $ REWIND LQ $ I=UNIT(LR) $ I=UNIT(LQ)
      JAX=JAR $ ML=IAX*JAXMM
C . . . READ BLOCKS FROM LR FOR SPREADING UP AND OUTPUT ON LS.
            DO 600 NQ=1,NAP $ IF(NAP.EQ.1)GO TO 515
            IF(NQ.EQ.NAP)JAX=JAXLST $ NUMPS=IAX*JAX
                DO 470 L=1,3
                BUFFER IN (LR,1)(KOEB(1,L),KOEB(NUMPS,L)) $ I=UNIT(LR)
  470           CONTINUE
            IF(NQ.EQ.NAP)GO TO 515 $ L=IAX+IX2 $ I=L+1 $ J=L+L
            BUFFER IN (LQ,1) (DN(I),DN(J)) $ II=UNIT(LQ)
            BUFFER IN (LQ,1) (GL(1),GL(L)) $ II=UNIT(LQ)
                DO 500 I=1,IAX $ II=-GL(I) $ OBS=DN(L+I)
                    DO 490 J=3,JAX $ MM=I+(JAX-J)*IAX
                    IF(II+FL(MM))485,475,480
  475               NV=KOEB(MM,3)=KOEB(MM,3)+1
                    F(MM)=(F(MM)*(NV-1)+OBS)/NV $ GO TO 490
  480               IF(F(MM).LT.LARGE)GO TO 500
  485               F(MM)=OBS $ FL(MM)=-II $ KOEB(MM,3)=1
  490               II=II+2*ISQRT(II)+1
  500           CONTINUE
C . . . DEFINE ROWS N-1 AND N.
                DO 510 I=1,IX2 $ FL(ML+I)=GL(IAX+I)
  510           F(ML+I)=DN(IX4+I)
C . . . DEFINE ROWS 1 AND 2.
  515       IF(NQ.EQ.1)GO TO 530 $ J=ML-IX2
                DO 520 I=1,IX2 $ FL(I)=GL(J+I)
  520           F(I)=DN(ML+I)
C . . . STORE ROWS N-2 AND N-3.
  530       IF(NQ.EQ.NAP)GO TO 550 $ J=ML-IX2
                DO 540 I=1,IX2 $ GL(J+I)=FL(J+I)
```

```
  540         DN(ML+I)=F(J+I)
  550     LAG=-NAG**2
C . . . PREPARE GRID BLOCK FOR SMOOTHING
            DO 560 L=1,NUMPS
            IF(FL(L).LT.LAG)FL(L)=0
  560       CONTINUE
C . . . STORE GRID VALUES ON LS.
            DO 570 L=1,3
            BUFFER OUT (LS,1)(KOEB(1,L),KOEB(NUMPS,L)) $ I=UNIT(LS)
  570       CONTINUE
  600     CONTINUE
      ENDFILE LS $ I=UNIT(LS) $ REWIND LS
      END
      OVERLAY(2,2)
      PROGRAM ITERATE
C * * * THIS PROGRAM SMOOTHS THE GRID VALUES TO FIT A SURFACE OF MINIMUM * * * C
C * * * CURVATURE. GRID VALUES ARE SAVED AFTER NBB ITERATIONS IF REQUIRED. * * C
      COMMON/COEFB/COEF(625,6),SUM(2),SUM2(2),VAL,VAL2,ADJM
      COMMON/DATAB/NOST,BORDER,NDB,NUBSA(3),X(1000),Y(1000),W(1000)
      COMMON/GRIDB/DX,DY,IAX,JAM,JAR,JAXLST,NAP,NIGP,NLP,NUMPS,X0,Y0,YXR
      COMMON/ITERB/INDIV,KG,LE,NN,NAG,NIP,NAL,NLA,NBB,NO,F,FL,H
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      DIMENSION KOEB(6000,3),F(6000),FL(6000),H(6000),ITYP(3)
      EQUIVALENCE (F,KOEB)
      INTEGER FL,Y
      REAL LE,LARGE
      DATA (ITYP=4HLINE,4HFAST,4HSLOW)
      OMEGA=KG/100.0 $ LARGE=10.0E10 $ NRE(6)=8HITA/9 ,
C . . . CALCULATE COEFFICIENTS AND PRINT HEADING.
      IX4=IAX*NLP $ IX2=IX4/2 $ CALL CALCO
      IF(IPRNT.GE.1)PRINT 10
   10 FORMAT(1HO,43X*FREE POINTS*9X*FIXED POINTS*/15X*ITERATION  TYPE*,
     1              *  BLOCK*5X*AV.ADJ.    RMS*7X*AV.ADJ.    RMS*,6X,
     2              *MAX.ADJ.    TIME*/1X)
C . . . ITERATION LOOP
          DO 250 NN=1,NAL
          REWIND LI $ I=UNIT(LI) $ JAX=JAR
              DO 20 L=1,2
              BUFFER IN (LI,1)(F(1),F(5)) $ IF (UNIT(LI))20,260,260
   20         CONTINUE
          IF(NN.LT.NAL)GO TO 30 $ IF(NBB.EQ.1)GO TO 30
          LAREA=7HCONTOUR $ NIGP=NIP $ NUMPS=NAL $ X0=BORDER
   30     REWIND LO $ I=UNIT(LO)
          BUFFER OUT (LO,1)(LAREA,RANGE) $ I=UNIT(LO)
          BUFFER OUT (LO,1)(DX,YXR) $ I=UNIT(LO)
C . . . ITERATE EACH BLOCK IN TURN.
              DO 240 NO=1,NAP $ IF(NO.EQ.NAP)JAX=JAXLST $ NUMPS=IAX*JAX
              IF(NN.NE.NAL)GO TO 40 $ NP=(JAR-NLP)*IAX
   40             DO 50 L=1,3
                  BUFFER IN (LI,1) (KOEB(1,L),KOEB(NUMPS,L))
                  IF(UNIT(LI))50,260,260
   50             CONTINUE $ NFIX=H(NUMPS) $ NFREE=H(NUMPS-1)
              SUM(1)=SUM(2)=SUM2(1)=SUM2(2)=ADJM=0
C . . RESET OVERLAPPING GRID VALUES AND CALL CURVATE TO ITERATE THE GRID BLOCK
              IF(NO.EQ.1)GO TO 90
   70             DO 80 L=1,IX2
                  F(L)=X(L)  $ IF(Y(L).NE.0)FL(L)=Y(L)
   80             CONTINUE
   90         IF(NN.GT.NLA)GO TO 110
              CALL LINEATE (IAX,JAX) $ GO TO 115
```

```
  110          CALL CURVATE (IAX,JAX,NAP,OMEGA)
  115          IF(NN.LT.NAL)GO TO 180 $ IF(NBB.EQ.1)GO TO 180
C . . . LAST ITERATION. SET GRID VALUES OUTSIDE CUTOFF TO THE DEFAULT VALUE.
               IF(NQ.LT.NAP)GO TO 120 $ NP=JAXLST*IAX
  120              DO 170 J=1,NP
                   IF(FL(J).EQ.0)F(J)=LARGE
  170              CONTINUE
C . . . OUTPUT THE NEW GRID VALUES.
  180              DO 185 L=1,3
                   BUFFER OUT (LO,1) (KOEP'1,L),KOEB(NUMPS,L))
                   J=UNIT(LO)
  185              CONTINUE
C . . . STORE THE OVERLAPPING VALUES.
               J=IAX*(JAX-NLP)
                   DO 190 L=1,IX2
                   X(L)=F(J+L) $ Y(L)=FL(J+L)
  190              CONTINUE
C . . . PRINT THE ITERATION STATISTICS.
               IF(IPRNT.EQ.0)GO TO 240 $ L=1 $ IF(NN.GT.NLA)L=2
               IF(NN.GT.NIP)L=3 $ TIMES=TIMEF(CLOK)-TUN
               VAL=SUM/NFREE $ VAL2=SQRT(SUM2/NFREE) $ ADJM=SQRT(ADJM)
               PRINT 200,NN,ITYP(L),NQ,VAL,VAL2,ADJM,TIMES
  200          FORMAT(15X,I5,6X,A4,I5,5X,2F10.3,20X,F10.3,-3PF10.3)
               IF(NN.LE.NIP.OR.NFIX.EQ.0)GO TO 240
               VAL=SUM(2)/NFIX $ VAL2=SQRT(SUM2(2)/NFIX)
               PRINT 210,VAL,VAL2
  210          FORMAT(1H+,59X,2F10.3)
  240          CONTINUE
           ENDFILE LO $ J=UNIT(LO) $ LI=LO $ LO=LR+LS-LI
  250      CONTINUE $ RETURN
  260 PRINT 270,LI $ CALL ORICHEK(6)
  270 FORMAT (5X,*ERROR ON RESTART TAPE *I3)
       END
       SUBROUTINE CALCO
C * * * THIS ROUTINE CALCULATES THE COEFFICIENTS FOR THE GRID SUB-SQUARES. * * C
       COMMON/COEFB/COEF(625,6),SUM(2),SUM2(2),VAL,VAL2,ADJM
       COMMON/ITERB/INDIV,KG,LE,NN,NAG,NIP,NAL,NLA,NBB,NQ,F,FL,H
       REAL LE
       SG=1.0/25.0
           DO 20 J=1,25 $ L=(J-1)*25
           DO 20 I=1,25 $ K=I+L $ IF(K.EQ.313)GO TO 20
           EX=(I-13)*SG $ EY=(J-13)*SG
           DX=ABS(EX) $ DY=ABS(EY) $ P=DX+DY
           COEF(K,1)=LE*(4/P)*(1+DX*DY/(1+P))
           COEF(K,2)=LE*(1-2*DX*(1+DX)/(P+P*P)  )
           COEF(K,3)=LE*2*(1-2*DY/(1+P))
           COEF(K,4)=LE*2*(1-2*DX/(1+P))
           COEF(K,5)=LE*(1-2*DY*(1+DY)/(P+P*P))
           IF(J.LT.13)GO TO 10
           M=1 $ IF(I.LT.13)M=2 $ GO TO 15
    10     M=4 $ IF(I.LT.13)M=3
    15     COEF(K,6)=M
    20     CONTINUE
       END
       SUBROUTINE LINEATE (I,J)
C * * * THIS ROUTINE LINEARLY SMOOTHS A GRID OF I BY J VALUES BY AVERAGING.* * C
       COMMON/COEFB/COEF(625,6),SUM(2),SUM2(2),VAL,VAL2,ADJM
       COMMON/ITERB/INDIV,KG,LE,NN,NAG,NIP,NAL,NLA,NBB,NQ,F,FL,H
       DIMENSION F(6000),FL(6000),H(6000)
       INTEGER FL
       II=JJ=3 $ IL=I-2 $ JL=J-2 $ KK=0
C . . . FIND THE AVERAGE OF NEAREST NEIGHBOURS FOR UNFLAGGED POINTS.
```

```
          DO 220 K=JJ,JL $ KK=KK+I
              DO 210 L=II,IL $ LL=KK+L
              IF(FL(LL).GE.1)GO TO 210
              H(LL)=(F(LL-1)+F(LL+1)+F(LL-I)+F(LL+I))/4.0
  210         CONTINUE
  220       CONTINUE
C . . . REDEFINE THE UNFLAGGED GRID VALUES WITH THE AVERAGE VALUE.
        KK=0
          DO 240 K=JJ,JL $ KK=KK+I
              DO 230 L=II,IL $ LL=KK+L
              IF(FL(LL).GE.1)GO TO 230
C . . . STORE THE ADJUSTMENT, ADJUSTMENT SQUARED AND MAXIMUM ADJUSTMENT. . . . C
              VAL=H(LL)-F(LL) $ VAL2=VAL**2 $ IF(ADJM.LT.VAL2)ADJM=VAL2
              SUM=SUM+VAL $ SUM2=SUM2+VAL2
              F(LL)=H(LL)
  230         CONTINUE
  240       CONTINUE
        END
        SUBROUTINE CURVATE (KXX,KYY,NAPP,EGA)
C * * * THIS ROUTINE SMOOTHS THE GRID SURFACE BY SUCCESSIVE ITERATIONS.  * * * C
        COMMON/COEFB/COEF(625,6),SUM(2),SUM2(2),VAL,VAL2,ADJM
        COMMON/ITERB/INDIV,KG,LE,NN,NAG,NIP,NAL,NLA,NBB,NQ,F,FL,H
        DIMENSION KOEB(6000,3),F(6000),FL(6000),H(6000)
        DIMENSION IB(4),JB(4)
        EQUIVALENCE (F,KOEB)
        INTEGER FL
        DATA (IB=1,-1,-1,1)
        DATA (JB=1,1,-1,-1)
        KX=KXX $ KY=KYY $ NIT=1
        KXM2=KX-2 $ KXM1=KX-1 $ KXP1=KX+1 $ KXP2=KX+2
        KX2=KX*2  $ KX2M2=KX2-2 $ KX2M1=KX2-1 $ KX2P1=KX2+1 $ KX2P2=KX2+2
        KX3=KX*3  $ KX3M1=KX3-1 $ KX3P1=KX3+1 $ KX3P2=KX3+2
        KX4=KX*4  $ KX4M1=KX4-1
        KK =KX*KY $ KK1=KK-KX $ KK2=KK-KX2 $ KK3=KK-KX3 $ KK4=KK-KX4
        KK1P2=KK1+2 $ KK2P2=KK2+2 $ KK3P2=KK3+2 $ KK4P2=KK4+2
            DO 200 LIT=1,NIT $ L=KXZM2
            IF(KY.LT.5.OR.KX.LT.5)GO TO 60
C . . . INTERNAL GRID POINTS.
              DO 50 J=5,KY $ N=1
              L=L+4 $ LG1=L+KX2 $ LG2=L+KX $ LG3=L-KX $ LG4=L-KX2
                  DO 40 I=5,KX
                  L=L+1 $ LG1=LG1+1 $ LG2=LG2+1 $ LG3=LG3+1 $ LG4=LG4+1
                  VAL=F(L) $ M=FL(L) $ IF(M.GE.1)20,10
   10             F(L)=(8.0*(F(L+1)+F(L-1)+F(LG2)+F(LG3))
     1                -2.0*(F(LG2+1)+F(LG2-1)+F(LG3+1)+F(LG3-1))
     2                -F(LG1)-F(LG4)-F(L+2)-F(L-2))/20.0
                  GO TO 30
   20             IF(NN.LE.NIP.OR.M.EQ.313)GO TO 30 $ N=2
C . . . MODIFY GRID VALUE WITH AN OBSERVATION. SLOW ITERATION.
                  LB3=COEF(M,6) $ LB1=IB(LB3) $ LB2=KX*JB(LB3)
                  FHS=H(L)+COEF(M,2)*F(L+LB1-LB2)+COEF(M,3)*F(L-LB2)+
     1                COEF(M,4)*F(L-LB1)+COEF(M,5)*F(L-LB1+LB2)
                  F(L)=(FHS+(F(L+1)+F(L-1)+F(LG2)+F(LG3))-0.25*(F(LG1)+
     1                 F(LG4)+F(L+2)+F(L-2))-0.5*(F(LG2-1)+F(LG2+1)+
     2                 F(LG3-1)+F(LG3+1)))/(1.0+COEF(M,1))
C . . . STORE THE ADJUSTMENT, ADJUSTMENT SQUARED AND MAXIMUM ADJUSTMENT. . . . C
   30             VAL=F(L)-VAL $ VAL2=VAL**2 $ IF(ADJM.LT.VAL2)ADJM=VAL2
                  SUM(N)=SUM(N)+VAL $ SUM2(N)=SUM2(N)+VAL2
   40             CONTINUE
   50           CONTINUE
   60       IF(NQ.NE.1)GO TO 100 $ IF(KX.LT.5)GO TO 70
C . . . TOP BORDER
```

```
          L=KXP2 $ LG1=KX3P2 $ LG2=KX2P2 $ LG3=2
              DO 65 I=5,KX $ L=L+1 $ LG1=LG1+1 $ LG2=LG2+1 $ LG3=LG3+1
              F(L)=(8.0*(F(L+1)+F(L-1)+F(LG2))+4.0*(F(LG3))
     1             -2.0*(F(LG2-1)+F(LG2+1))-F(LG1)-F(L-2)-F(L+2)
     2             -F(LG3-1)-F(LG3+1))/19.0
    65        CONTINUE
    70        F(KXP2)=(8.0*(F(KX+3)+F(KX2P2))+4.0*(F(2)+F(KXP1))-F(KX3P2)
     1             -F(KX+4)-F(3)-F(KX2P1)-2.0*F(KX2+3))/18.0
              F(KX2M1)=(8.0*(F(KX2M2)+F(KX3M1))+4.0*(F(KXM1)+F(KX2))-F(KX3)
     1             -F(KX4M1)-F(KX2-3)-F(KXM2)-2.0*F(KX3-2))/18.0
              IF(KX.LT.5)GO TO 90 $ LG1=KX2P2 $ LG2=KXP2 $ L=2
              DO 80 I=5,KX $ L=L+1 $ LG1=LG1+1 $ LG2=LG2+1
              F(L)=(4.0*(F(L-1)+F(L+1)+F(LG2))-F(LG1)-F(L-2)-F(L+2)
     1             -F(LG2-1)-F(LG2+1))/7.0
    80        CONTINUE
C . . . TOP CORNERS
    90        F(2)=(4.0*(F(3)+F(KXP2))+2.0*F(1)-F(KX2P2)-F(KX+3)-F(KXP1)
     1           -F(4))/6.0
              F(KXP1)=(4.0*(F(KX2P1)+F(KXP2))+2.0*F(1)-F(KX2P2)-F(KX+3)
     1             -F(KX3P1)-F(2))/6.0
              F(KXM1)=(4.0*(F(KX2M1)+F(KXM2))+2.0*F(KX)-F(KX2M2)-F(KX-3)
     1             -F(KX3M1)-F(KX2))/6.0
              F(KX2)=(4.0*(F(KX3)+F(KX2M1))+2.0*F(KX)-F(KX2M2)-F(KXM1)
     1            -F(KX3M1)-F(KX4))/6.0
              F(1)=(2.0*(F(2)+F(KXP1))-F(3)-F(KX2P1))/2.0
              F(KX)=(2.0*(F(KXM1)+(KX2))-F(KXM2)-F(KX3))/2.0
   100        IF(KY.LT.5)GO TO 130
C . . . SIDE BORDERS
              LG1=KX3P2 $ LG2=KX2P2 $ L=KXP2 $ LG3=2 $ LG4=-KXM2
              LB1=KX4M1 $ LB2=KX3M1 $ LB3=KX2M1 $ LB4=KXM1 $ LB5=-1
                  DO 110 J=5,KY $ LB5=LB5+KX
                  LG1=LG1+KX $ LG2=LG2+KX $ L=L+KX $ LG3=LG3+KX $ LG4=LG4+KX
                  LB1=LB1+KX $ LB2=LB2+KX $ LB3=LB3+KX $ LB4=LB4+KX
                  F(L)=(8.0*(F(LG2)+F(LG3)+F(L+1)+4.0*F(L-1)-2.0*(F(LG2+1)+
     1                 F(LG3+1))-F(LG1)-F(LG4)-F(LG2-1)-F(LG3-1)-F(L+2))/19.
                  F(LB3)=(8.0*(F(LB2)+F(LB4)+F(LB3-1))+4.0*F(LB3+1)-F(LB3-2)
     1                    -2.0*(F(LB2-1)+F(LB4-1))-F(LB1)-F(LB5)-F(LB2+1)
     2                    -F(LB4-1))/19.0
   110            CONTINUE
              LG1=KX3P1 $ LG2=KX2P1 $ L=KXP1 $ LG3=1 $ LG4=-KXM1
              LB1=KX4 $ LB2=KX3 $ LB3=KX2 $ LB4=KX $ LB5=0
                  DO 120 J=5,KY $ LB5=LB5+KX
                  LG1=LG1+KX $ LG2=LG2+KX $ L=L+KX $ LG3=LG3+KX $ LG4=LG4+KX
                  LB1=LB1+KX $ LB2=LB2+KX $ LB3=LB3+KX $ LB4=LB4+KX
                  F(L)=(4.0*(F(LG2)+F(LG3)+F(L+1))-F(LG1)-F(LG4)-F(LG3+1)
     1                -F(LG2+1)-F(L+2))/7.0
                  F(LB3)=(4.0*(F(LB2)+F(LB4)+F(LB3-1))-F(LB1)-F(LB5)
     1                    -F(LB2-1)-F(LB4-1)-F(LB3-2))/7.0
   120            CONTINUE
   130        IF(NO.NE.NAPP)GO TO 200 $ IF(KX.LT.5)GO TO 150
C . . . BOTTOM BORDER
              LG2=KK1P2 $ L=KK2P2 $ LG3=KK3P2 $ LG4=KK4P2
                  DO 140 I=5,KX
                  LG2=LG2+1 $ L=L+1 $ LG3=LG3+1 $ LG4=LG4+1
                  F(L)=(8.0*(F(L-1)+F(L-1)+F(LG3))+4.0*F(LG2)-2.0*(F(LG3-1)+
     1                 F(LG3+1))-F(LG4)-F(L-2)-F(L+2)-F(LG2+1)-F(LG2-1))/19.
   140            CONTINUE
   150        F(KK2P2)=(8.0*(F(KK3P2)+F(KK2+3))+4.0*(F(KK2+1)+F(KK1P2))-
     1                2.0*F(KK3+3)-F(KK2+4)-F(KK4P2)-F(KK3+1)-F(KK1+3))/18.
              F(KK1-1)=(8.0*(F(KK1-2)+F(KK2-1))+4.0*(F(KK1)+F(KK-1))-2.0*
     1                 F(KK2-2)-F(KK3-1)-F(KK1-3)-F(KK2)-F(KK-2))/18.0
              IF(KX.LT.5)GO TO 170 $ LG1=KK3P2 $ LG2=KK2P2 $ L=KK1P2
```

```
            DO 160 I=5,KX $ LG1=LG1+1 $ LG2=LG2+1 $ L=L+1
            F(L)=(4.0*(F(L-1)+F(L+1)+F(LG2))-F(LG1)-F(L-2)-F(L+2)
     1           -F(LG2-1)-F(LG2+1))/7.0
  160       CONTINUE
C . . . BOTTOM CORNERS
  170     F(KK2+1)=(4.0*(F(KK2P2)+F(KK3+1))+2.0*F(KK1+1)-F(KK2+3)
     1           -F(KK3P2)-F(KK1P2)-F(KK4+1))/6.0
          F(KK1P2)=(4.0*(F(KK2P2)+F(KK1+3))+2.0*F(KK1+1)-F(KK2+3)
     1           -F(KK3P2)-F(KK2+1)-F(KK1+4))/6.0
          F(KK1)=(4.0*(F(KK2)+F(KK1-1))+2.0*F(KK)-F(KK1-2)-F(KK2-1)
     1          -F(KK-1)-F(KK3))/6.0
          F(KK-1)=(4.0*(F(KK-2)+F(KK1-1))+2.0*F(KK)-F(KK1-2)-F(KK2-1)
     1          -F(KK-3)-F(KK1))/6.0
          F(KK1+1)=(2.0*(F(KK2+1)+F(KK1+2))-F(KK3+1)-F(KK1+3))/2.0
          F(KK)=(2.0*(F(KK-1)+F(KK1))-F(KK-2)-F(KK2))/2.0
  200     CONTINUE
      END
      OVERLAY(3,0)
      PROGRAM CONTOUR
C * * * THIS PROGRAM CONTROLS THE DRAWING OF CONTOUR LINES FROM GRIDDED DATA.* C
C - - - WINDOWING OPTION ADDED IN JUNE 1977.
C - - -  - - - - - - - - THIS EDITION OF PROGRAM DATED 29/06/77  - - - - - - - - - - C
      COMMON/CONTROL/NAIM(8),INMAIN,NEDATE,NEDIT,NEVERS,LOTY
      COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
      COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
      COMMON/UNITB/LD,LI,LO,LQ,LR,LS,LT,LUN,NOPB,IBUF(1000)
      COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
      COMMON/ALAKP/A,LA,KP,MULP
      COMMON/INTER/CONVL(2),SINHIB,ILF,IRT,JTP,JBT
      COMMON/SCOL/SCINT,SCALE,OIST,PIST,NAP,IAX,JAX,KULN,INHIB,IT
      DIMENSION A(2000),LA(7500),KP(40),C(8000),F(6000),NUCS(2),MULP(40)
      DIMENSION NA(8),NOM(4)
      EQUIVALENCE (F,LA) , (DX,KP(1)) , (DY,KP(2)) , (NA,C(5))
      INTEGER A
      DATA (NOM=8H OMITTED,10H- INFINITY,10H ARE HEAVY,10H+ INFINITY)

      DTL=DBL=DWL=DEL=0.0 $ NRE(7)=8HCON/11,
      IF(NAME.EQ.7HCONTOUR)LI=40 $ NMR=0 $ BIG=10.0E+0 $ SMALL=0.0001
C . . . DEFINE CONTOURING PARAMETERS FROM INPUT CARD OR ASSIGN DEFAULT VALUES. C
    5 IF(INMAIN.NE.0)GO TO 50 $ READ 10,NAME,NAIM
   10 FORMAT (A8,T1,8A10)
      IF(EOF(60))450,20
   20 IF(NAME.EQ.8H*CONTOUR)GO TO 50
      IF(NAME.NE.8H*MAPPLOT)GO TO 25 $ INMAIN=1 $ GO TO 455
   25 PRINT 30,NAME
   30 FORMAT(1H1,10X,*CONTOUR CARD ERROR.  FIRST WORD WAS - *,A10)
   40 CALL ORICHEK (6) $ RETURN
   50 INMAIN=0 $ IF(NAIM(2).EQ.6HWINDOW)GO TO 500
      DECODE (80,60,NAIM) NAME,NUMBA,CONVL,CMIN,CMAX,SINHIB,IT,INK,JPRNT
   60 FORMAT (A8,I2,5F5.2,I1,2I2)
      NUR=0 $ IF(NMR.NE.0)GO TO 65  $ CALL BEGINIT $ CALL ORICHEK (4)
   65 IF(NMR.GT.0)GO TO 200 $ NMR=-NMR
      IF(IPRNT.EQ.0)IPRNT=JPRNT $ JPRNT=IPRNT/2

C . . . READ IN VARIABLES FROM RESTART TAPE AND DEFINE CONTOUR BLOCKS.
      REWIND LI $ REWIND LD $ J=UNIT(LI) $ J=UNIT(LD)
      BUFFER IN (LI,1)(LAREA,RANGE) $ IF(UNIT(LI))70,80,80
   70 BUFFER IN (LI,1)(KP,KP(11))   $ IF(UNIT(LI))100,80,80
   80 PRINT 90,LI $ GO TO 40
   90 FORMAT (5X,* END OF FILE OR PARITY ERROR ON RESTART UNIT *I3)
  100 IF(LAREA.EQ.7HCONTOUR)GO TO 120 $ PRINT 110 $ GO TO 40
  110 FORMAT (6X,*DATA NOT IN CONTOUR FORMAT, FINAL ITERATION NEEDED*)
```

```
  120 IAX=KP(3) $ JAR=KP(5) $ JAXLST=KP(6) $ NAP=KP(7) $ NLP=KP(9)
      IF(LI.EQ.40)CALL PARAM (KP(1),KP(2),KP(11),KP(8),KP(10))
      SCALE=10.0*SQRT(SCAX*SCAY*S00)
      ABMAX=AVERAG+RANGE $ ABMIN=AVERAG-RANGE
      JAX=JAR-NLP $ NUMPS=IAX*JAR
      LONOS=INT(DWL/DX+0.5)-2 $ ILF=L1+LONOS $ IRT=L2+2+INT(DEL/DX-0.5)
      LATOS=INT(DTL/DY+0.5)-2 $ JTP=M1+LATOS $ JBT=M2+2+INT(DBL/DY-0.5)
      IHG=IRT-ILF+1 $ JHG=JBT-JTP+1 $ MAX=JHG $ IF(MAX.GT.40)MAX=40
      JBL=8000/IHG $ IF(JBL.GT.MAX)JBL=MAX
      IBL=2000/JBL $ IF(IBL.GT.IHG)IBL=IHG
      NBJ=LINE=JU=0 $ LAP=5 $ NUMC=JBL*IHG $ LVJ=JBL
      NBI=(IHG-LAP-1)/(IBL-LAP)+1 $ LVI=IHG-(NBI-1)*(IBL-LAP)
          DO 190 NO=1,NAP $ IF(NO.EQ.NAP)JAX=JAXLST $ JL=ILF-1
          BUFFER IN (LI,1)( F, F(NUMPS)) $ IF(UNIT(LI))130,80
  130     BUFFER IN (LI,1)(KP(1),KP(10)) $ IF(UNIT(LI))140,80
  140     BUFFER IN (LI,1)(KP(1),KP(10)) $ IF(UNIT(LI))150,80
C . . . REFORMAT GRID VALUES ONTO CONTOUR BLOCKS AND WRITE ON FILE LD.
  150         DO 180 J=1,JAX $ LINE=LINE+1
              IF(LINE.LT.JTP)GO TO 180 $ IF(LINE.GT.JBT)GO TO 195
                  DO 160 I=1,IHG
  160             C(JU+I)=F(JL+I)
              JU=JU+IHG $ IF(JU.LT.NUMC)GO TO 180
              BUFFER OUT (LD,1) (C,C(NUMC)) $ I=UNIT(LD)
              JU=5*IHG $ NBJ=NBJ+1
                  DO 170 I=1,JU
  170             C(I)=C(I+NUMC-JU)
  180         JL=JL+IAX
  190     CONTINUE
  195 IF(JU.LE.5*IHG)GO TO 200 $ LVJ=JU/IHG $ NBJ=NBJ+1
      BUFFER OUT (LD,1) (C,C(JU)) $ I=UNIT(LD)
      ENDFILE LD $ I=UNIT(LD)

  200 REWIND LD $ I=UNIT(LD) $ NMR=NMR+1
      IF(SINHIB.LE.0.0)SINHIB=0.05/SCAY
      IF(NUMBA.EQ.0)GO TO 215
      IF(NUR.GT.0)GO TO 205 $ IF(NUMBA.EQ.1)GO TO 210
      ONE=CONVL(1) $ TWO=CONVL(2) $ UP=CMAX $ DN=CMIN
      CONVL(1)=TWO $ CONVL(2)=0.0 $ IF(NUMBA.GT.3)NUMBA=3
      IF(CMAX-CMIN.GT.SMALL)GO TO 210 $ UP=BIG $ DN=-BIG $ GO TO 210
  205 CONVL(1)=ONE $ CONVL(2)=-TWO $ CMIN=DN $ CMAX=UP
      IF(NUR.EQ.2.AND.CMIN.LT.0.0)CMIN=0.0
      IF(NUMBA.EQ.2.AND.CMAX.GT.0.0)CMAX=0.0
  210 NUMBA=NUMBA-1 $ INK=NUR=NUR+1  $  IF(CMAX.LT.CMIN)GO TO 445
      IF(CONVL.GE.0.0)GO TO 215 $ TWO=-TWO $ IF(NUMBA.GT.0)GO TO 205
  215 IF(NMR.EQ.1)GO TO 225
C . . . SET NEW PLOT BLOCK FOR ANOTHER CONTOUR LEVEL.
      PRINT 220,NMR $ CALL TAGBLOK
  220 FORMAT (1H0,11X,30(1H*),* START CONTOUR PLOT BLOCK *,I3)
      CALL PLOT (0.0,0.0,-3) $ CALL NEWPEN (1)
  225 IF(CONVL.GT.0.0)GO TO 250 $ RS=1.0
  230 RG=RANGE*RS
      IF(RG.GE.1.00)GO TO 235 $ RS=RS*10.0 $ GO TO 230
  235 IF(RG.LT.10.0)GO TO 240 $ RS=RS/10.0 $ GO TO 230
C . . . IF CONTOUR LEVELS ARE UNDEFINED CALCULATE LEVELS TO SUIT THIS DATA.
  240 CONVL=0.5
      IF(RG.LE.4.0)CONVL=0.2
      IF(RG.LE.2.0)CONVL=0.1
      CONVL=CONVL/RS $ CONVL(2)=CONVL*10.0
  250 IF(RANGE.GT.500.0*CONVL)GO TO 460 $ JL=2
      IF(CMAX.GT.CMIN.AND.CMAX.LE.0)CONVL=-CONVL
      I=1 $ IF(CONVL(1).LT.0.0)I=-1
      J=1 $ IF(CONVL(2).LT.0.0)J=-1
```

```
  255 ZINT=CONVL(2)*I*J $ IF(ZINT.NE.0.0)GO TO 260 $ CONVL(2)=BIG $ JL=1
  260 PRINT 265,CONVL(1),ZINT,NOM(2*J),SINHIB
  265 FORMAT(1H-,20X,*CONTOUR LINE EVERY*F10.3* , CONTOURS AT *F10.3,A10
     1       /1H0,20X,*LINES INHIBITED IF CLOSER THAN*F7.3* MINUTES*)
      ENCODE (20,267,NUCS) CMIN,CMAX
  267 FORMAT (2F10.3)
      IF(CMIN.LE.-BIG)NUCS(1)=NOM(2) $ IF(CMAX.GE.BIG)NUCS(2)=NOM(4)
      IF(CMAX.GT.CMIN.AND.CMAX-CMIN.LT.2*BIG)PRINT 270,NUCS
  270 FORMAT(1H0,20X,*CONTOUR BETWEEN * A10 * AND * A10 * ONLY*)
      IF(CMAX.GT.ABMAX)CMAX=ABMAX $ IF(CMIN.LT.ABMIN)CMIN=ABMIN
      IF(INK.LT.1.OR.INK.GT.4)INK=1
      PRINT 275,INK
  275 FORMAT(1H0,2 X,*THE PEN NUMBER IS *I2)
      CALL PARAM (0.0,CONVL(1),ZINT*(J+1)/2,INK,NMR)
      NUCS(1)=4H AND $ IF(J.LT.0)NUCS(1)=4H NOT
      WRITE (LT,277) CONVL(1),NUCS(1),ZINT,INK
  277 FORMAT (* LEVEL*F10.3,A4,F10.3/* PEN NUMBER *,I1,17X)
      CALL PLOT (0.0,0.0,3)
C . . . FLOATING GRID VALUES ARE SCALED BY ZINT TO INTEGERS FOR LINKUT.  . . . C
      CONVL(1)=ABS(CONVL(1)) $ JAX=JBL
      ZINT=8000.0/CONVL $ ZINHIB=ZINT/(SINHIB*SCAY)
      IF(IPRNT.NE.0)PRINT 280,IHG,JHG,NBI,NBJ,IBL,JBL,LVI,LVJ,ZINT
  280 FORMAT(1X/5X,*IHG=*I5* JHG=*I5* NBI=*I3* NBJ=*I3* IBL=*I5,
     1       * JBL=*I5* LVI=*I5* LVJ=*I5/5X,*SCALE FACTOR IS *,F8.2)
C . . . SEARCH DATA TO FIND NUMBER OF LINES NEEDED.
          DO 440 J=1,NBJ
          BUFFER IN (LD,1) (C,C(NUMC)) $ IF(UNIT(LD))290,80
  290     IF(IPRNT.EQ.0)GO TO 300
          TIMES=TIMEF(CLOK)-TUN $ PRINT 295,TIMES
  295     FORMAT (*0 TIME IS ....*-3PF10.3* SECOND*)
  300     IAX=IBL * F(J.EQ.NBJ)JAX=LVJ $ PIST-(J-1)*(JBL-LAP)+LATOS
              DO      I=1,NBI $ GRT=-BIG $ EAS=BIG
              K=(I-1)*(IBL-LAP) $ QIST=K+LONOS $ IF(I.EQ.NBI)IAX=LVI
              IF(JPRNT.NE.0)PRINT 310,I,J,IAX,JAX,QIST,PIST
  310         FORMAT(5X*I=*I3* J=*I3* IAX=*I5* JAX=*I5,
     1                 * QIST=*F5.0* PIST=*F5.0)
                  DO 340 JJ=1,JAX
                  M=(JJ-1)*IAX $ ML=(JJ-1)*IHG+K
                      DO 320 II=1,IAX $ L=II+ML
                      A(II+M)=C(L)*ZINT
                      IF(C(L).GT.GRT.AND.C(L).LT.BIG)GRT=C(L)
                      IF(C(L).LT.EAS)EAS=C(L)
  320                 CONTINUE
                  IF(JPRNT.NE.0)PRINT 330,C(ML+1),C(L)
  330             FORMAT(20X,*EDGES*20X,E10.2,10X,E10.2)
  340             CONTINUE
C . . . IF NO LINES ARE NEEDED SKIP THIS BLOCK.
              IF(GRT.GT.EAS)GO TO 350
  345         PRINT 346 $ GO TO 430
  346         FORMAT (1H0,20X,*NO CONTOURS IN THIS BLOCK*)
  350         IF(ABS(CMAX-CMIN).LE.SMALL)GO TO 355
              IF(CMIN.GT.GRT.OR.CMAX.LT.EAS)GO TO 345
              IF(CMIN.GT.EAS)EAS=CMIN
              IF(CMAX.LT.GRT)GRT=CMAX
C . . . CALCULATE AND STORE THE REQUIRED LEVELS AND THEIR WEIGHT.
  355         GRT=GRT+SMALL $ EAS=EAS-SMALL
              LG=ABS(GRT)/CONVL $ LGB=ABS(GRT)/CONVL(2)
              LL=ABS(EAS)/CONVL $ LLB=ABS(EAS)/CONVL(2)
              IF(GRT.GT.0.0)GO TO 360 $ LG=-LG-1 $ LGB=-LGB-1
  360         IF(EAS.GT.0.0)GO TO 370 $ LL=-LL-1 $ LLB=-LLB-1
  370         IF(JPRNT.NE.0)PRINT 380,LG,LL,LGB,LLB
  380         FORMAT (2X*LG=*I5* LL=*I5* LGB=*I5* LLB=*I5)
```

```
            NUCS=LG-LL $ NUCS(2)=LGB-LLB
                DO 420 K=1,JL * IF(CONVL(K).LE.0.0)GO TO 420
                IF(NUCS(K).EQ.0.GO TO 420
385             KULN=0 $ INHIB=ZINHIB*CONVL(K)
                    DO 400 KK=1,40
                    CP=CONVL(K)*(LL+KK)
                    KP(KK)=CP*ZINT $ MULP(KK)=K
                    IF(K.EQ.2.OR.JL.EQ.1)GO TO 390
                    L=CP/CONVL(2) $ IF(L*CONVL(2).EQ.CP)MULP(KK)=0
390                 IF(IPRNT.NE.0)PRINT 395,CP,KK
395                 FORMAT (5X,*CP=*F8.0*   KK=*I3)
                    KULN=KULN+1 $ IF(KK.GE.NUCS(K))GO TO 410
400                 CONTINUE
C . . . CALL LINKUT TO FIND THE CONTOUR LINE CUTS.
410             NUCS(K)=NUCS(K)-KULN $ LL=LL+KULN
                CALL LINKUT
                IF(NUCS(K).GT.0)GO TO 385 $ LL=LLB
420             CONTINUE
430         CONTINUE
440     CONTINUE
445 IF(NUMBA.GT.0)GO TO 200 $ GO TO 5
450 LOT=3HEND
455 NAME=8H*CONTOUR $ CALL FINIT $ CALL TAGBLOK
    CALL PLOT (0.0,0.0,-3) $ CALL NEWPEN (1)
    CALL ORICHEK (5) $ RETURN
460 PRINT 470,RANGE,CONVL(1) $ GO TO 445
470 FORMAT(6X*MORE THAN 500 LEVELS ON MAP. RANGE *G12.2* LEVEL *F10.2)
500 DECODE (80,510,NAIM) NAME,LAREA,NA $ NMR=-IABS(NMR)
510 FORMAT (2A10,4(I3,I2))
        DO 520 K=1,7,2
        IF(NA(K).LT.0)NA(K+1)=-NA(K+1)
        C(K/2+1)=60.0*NA(K)+NA(K+1)
520     CONTINUE
    IF(C(1).GE.C(2).OR.C(3).GE.C(4))GO TO 540
    IF(C(1).LT.TLT)C(1)=TLT $ IF(C(2).GT.BLT)C(2)=BLT
    IF(C(3).LT.WLN)C(3)=WLN $ IF(C(4).GT.ELN)C(4)=ELN
    DTL=C(1)-TLT $ DBL=C(2)-BLT $ DWL=C(3)-WLN $ DEL=C(4)-ELN
    PRINT 530,NA $ WRITE (LT,535) NA $ GO TO 5
530 FORMAT (1H0,11X,60(1H*)/1H0,11X,*WINDOW AT LATITUDE *2I3* TO *2I3,
   1        * AND LONGITUDE *2I3* TO *2I3)
535 FORMAT (1H ,29(1H-)/* WINDOW *2(I3,I2),1H/,2(I3,I2))
540 PRINT 550 $ WRITE (LT,550) $ DTL=DBL=DWL=DEL=0.0 $ GO TO 5
550 FORMAT (* WINDOW CANCELLED *,12X,30(1H<))
    END
    SUBROUTINE LINKUT
    COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LOT,LOTX,NRE(9)
    COMMON/ALAKP/F(2000),KUT(7500),LVL(40),MOL(40)
    COMMON/CUBIS/JX,KON,I,J,INTS,IAXM,JAXM
    COMMON/DRAW/NOP,LOOP,LDF,LIN,LKI,LRF,PEN
    COMMON/LINX/ LIX,LIY,LFX,LFY
    COMMON/SCOL/SCINT,SCALE,OX,OY,LINE ,IAX,JAX,NOC,INHIB,IT
C . . . CALCULATES THE POSITIONS OF CONTOUR LINES INTERSECTIONS WITH THE GRID. C
C . . .  POSITIONS OF CONTOURS CUTTING GRID DIAGONALS ADDED IN JAN 1978.  . . . C
C . . .  LIX = INITIAL X FOR LINE.     LFX = FINAL X FOR LINE.
C . . .  LIY = INITIAL Y FOR LINE.     LFY = FINAL Y FOR LINE.
C . . .  LDF = LINE DIRECTION FLAG.    LKI = LINE KUT INDEX.
C . . .  LOF = LAST OR FIRST POINT.    LRF = LINE REPETITION FACTOR.
C - - - THIS EDITION WRITTEN MARCH 1977. THIS VERSION DATED 12/01/78.
    DIMENSION NOP(200),LOOP(200),LDF(200),LIN(200),LKI(200),LRF(200)
    DIMENSION LIX(200),LIY(200),LFX(200),LFY(200),IDI(200)
    DIMENSION DN(80),FL(80),PEN(80),USED(4)
    DIMENSION NL(2),NE(4),NM(4),MM(12),KM(6),IM(6)
```

```
      INTEGER F,DN,FL,PEN,USED
      DATA (NE=2,3,4,1),(KM=2,1,2,1,2,1),(IM=0,0,1,0,1,1)
      DATA (NM=1,0,-1,0),(MM=0,0,0,-1,-1,0,0,0,1,0,1,0)
      DATA (INTS=1000),(LARGE=1000000000),(LIM=3),(SCINT=1000.0)

      NRE(8)=8HKUT/9 ,
      TIMES=TIMEF(CLOK)-TUN $ IF(IPRNT.NE.0)PRINT 10,TIMES
   10 FORMAT (* LINKUT ENTERED AT TIME .... *F10.0)
      NL(1)=IAXMM=IAX-2 $ NL(2)=JAX-3 $ MM(6)=IAXM=IAX-1 $ MM(3)=-MM(6)
      NM(2)=MM(8)=MM(12)=IAX $ NM(4)=MM(1)=MM(2)=-IAX
      NUMPS=IAX*JAX $ JAXM=JAX-1 $ JAXMM=JAXM-1 $ IAX2=IAX*2
C . . . SET EDGE FLAGS.
          DO 30 J=1,JAX
              DO 20 I=1,IAX $ M=I+(J-1)*IAX $ NX=NY=0
              IF(I.LE.2.OR.J.LE.2)NX=NY=1
              IF(I.GE.IAXMM.OR.J.GE.JAXM)NX=1
              IF(J.GE.JAXMM.OR.I.GE.IAXM)NY=1
              CALL LGL (DN,M,1,NX) $ CALL LGL (DN,M,2,NY)
   20         CONTINUE
   30     CONTINUE
C . . . SET INHIBITION FLAGS ON.
          DO 80 J=2,JAXM
              DO 70 I=2,IAXM $ M=I+(J-1)*IAX $ IF(F(M).GE.LARGE)GO TO 60
   50         IF(F(M+IAX)+F(M-IAX)+F(M+1)+F(M-1).GE.LARGE)GO TO 70
              L=ISQRT((F(M+IAX)-F(M-IAX))**2+(F(M+1)-F(M-1))**2)/2
              IF(L.LE.INHIB)GO TO 70
   60         CALL LGL (DN,M,1,1) $ CALL LGL (DN,M-1,1,1)
              CALL LGL (DN,M,2,1) $ CALL LGL (DN,M-IAX,2,1)
   70         CONTINUE
   80     CONTINUE
C . . . CONTOUR LEVEL SEARCH.
      KT=-1 $ LINE=NI=0
          DO 800 KK=1,NOC $ IF(MOL(KK).EQ.0)GO TO 800 $ KON=LVL(KK)
          NL(1)=IAXMM $ NL(2)=JAX-3
          IF(IPRNT.EQ.0)GO TO 100
          TIMES=TIMEF(CLOK)-TUN $ PRINT 90,KON,TIMES
   90     FORMAT (10X,*CONTOUR LEVEL *I10*   STARTED AT TIME ...*F10.0)
  100         DO 110 M=1,NUMPS
              CALL LGL (FL,M,1,0) $ CALL LGL (FL,M,2,0)
              IF(IABS(KON-F(M)).GT.LIM)GO TO 110
C . . . CONTOUR LEVEL THROUGH A GRID POINT.
              K=-LIM $ IF(F(M).GE.KON)K=LIM $ F(M)=F(M)+K
  110         CONTINUE
C . . . SET CONTOUR CUT FLAGS.
              DO 140 J=2,JAXM
                  DO 130 I=2,IAXM $ M=I+(J-1)*IAX
                  IF(I.GE.IAXM)GO TO 120 $ MP=M+1
                  IF(F(M)+F(MP).GE.LARGE.OR.F(M).EQ.F(MP))GO TO 120
                  L=(KON-F(M))*(F(MP)-KON)
                  IF(L.GE.0)CALL LGL (FL,M,1,1)
  120             IF(J.GE.JAXM)GO TO 130 $ MP=M+IAX
                  IF(F(M)+F(MP).GE.LARGE.OR.F(M).EQ.F(MP))GO TO 130
                  L=(KON-F(M))*(F(MP)-KON)
                  IF(L.GE.0)CALL LGL (FL,M,2,1)
  130             CONTINUE
  140         CONTINUE
C . . . SEARCH FOR CONTOUR CUT FLAGS
          MU=IAX+1 $ L=0 $ NW=1 $ MAX=NL(1)  $ KTST=0
  150     IF(KTST.GT.0)GO TO 700 $ M=MU+NM(NW) $ L=L+1
              DO 160 K=1,2 $ CALL LGC (FL,M,K,MP) $ IF(MP.EQ.1)GO TO 170
  160         CONTINUE $ K=0
  170     IF(L.LT.MAX)GO TO 190 $ IN=0
```

```
  180     NW=NE(NW) $ L=NW-NW/3*2 $ MAX=NL(L)=NL(L)-1 $ L=0
          IF(MAX.GE.1)GO TO 190 $ IN=1-IN $ IF(IN.EQ.1)180,780
  190     MU=M $ IF(K.EQ.0)GO TO 150 $ NP=M*2+K $ IN=N=0
C . . . SEARCH END OF LINE
  200     IK=K/2*3 $ IJ=IN*6+IK $ NT=0 $ N=N+1
              DO 210 II=1,3 $ MN=M+MM(II+IJ) $ KN=KM(II+IK)
              CALL LGC (FL,MN,KN,MP) $ IF(MP.EQ.1)NT=NT+II
  210         CONTINUE $ IF(NT.GE.1)GO TO 215
          IN=1-IN $ LP=0 $ GO TO 220
  215     IF(NT.GT.3)GO TO 216 $ IN=IM(NT+IN*3)
          M=M+MM(NT+IJ) $ K=KM(NT+IK)
          LP=M*2+K $ IF(LP.EQ.NP.OR.N.GT.1000)220,200
  216     ND=MN $ IF(MM(IJ+1).EQ.0)ND=MN+NM(KN)
          CALL MIDF (M,ND) $ IF(JX.EQ.KON)JX=KON+10  $ NT=1
          JY=(KON-F(M))*(JX-KON) $ IF(JY.LE.0)NT=3 $ GO TO 215
C . . . BEGIN LINE
  220     J=(M-1)/IAX+1 $ I=M-(J-1)*IAX $ N=M+NM(K)
          JX=INTS*(KON-F(M))/(F(N)-F(M)) $ LINE=LINE+1
          CALL CUSP (M,N) $ NO=1
          NX=NY=0 $ IF(F(N).LT.F(M))NY=1 $ IF(N-M.EQ.IAX)NY=1-NY
          LRF(LINE)=MOL(KK) $ LKI(LINE)=KT=KT+2 $ NP=(KT+1)/2
          KUT(KT)=LIX(LINE)=INTS*(I-1)+JX*(2-K)
          KUT(KT+1)=LIY(LINE)=INTS*(J-1)+JX*(K-1)
          CALL LGC (DN,M,K,MP) $ CALL LGL (PEN,NP,0,MP)
          MP=1 $ IF(LP.EQ.0)MP=0 $ LOOP(LINE)=MP $ CALL LGL (FL,M,K,MP)
  230     IK=K/2*3 $ IJ=IN*6+IK $ NT=0
              DO 240 II=1,3 $ MN=M+MM(II+IJ) $ KN=KM(II+IK)
              CALL LGC (FL,MN,KN,MP) $ IF(MP.EQ.1)NT=NT+II
  240         CONTINUE $ IF(NT.GE.1)GO TO 270
  250     NOP(LINE)=MP=NP-NI $ IF(MP.GE.2)GO TO 260
C . . . LINE IS TOO SHORT
          LINE=LINE-1 $ KT=KT-2*MP $ GO TO 150
C . . . FINISH LINE
  260     LFX(LINE)=LEX $ LFY(LINE)=LEY $ NI=NP
          IF(LINE.GE.200.OR.KT.GE.7000)810,150
C . . . FIND NEXT CUT.
  270     IF(NX.NE.0)GO TO 280 $ NX=2-NY
          IF(MN.EQ.M.OR.MN.EQ.M+1.OR.MN.EQ.M+IAX)NX=1+NY $ IDI(LINE)=NX
C . . . CALCULATE THE CUTS ACROSS EACH DIAGONAL TO THE EXIT EDGE.
  280     MN=M+MM(IJ+1) $ MO=M+MM(IJ+3) $ KN=KM(IK+1) $ KO=KM(IK+3)
          NN=MN+NM(KN) $ ND=MO+NM(KO) $ IF(M.NE.MN)ND=MO
C . . . FIND THE MIDPOINT  VALUE  OF THE GRID SQUARE AND THE DIAGONAL CUT.
  290     CALL MIDF (M,ND) $ MID=JX $ JY=(KON-F(M))*(MID-KON)
          MQ=1 $ IF(JY.LT.0)MQ=3 $ IF(NT.EQ.6)NT=MQ $ JS=1
  400     MN=M=M+MM(IJ+NT) $ KN=K=KM(IK+NT) $ NN=MN+NM(KN)
          CALL LGC (DN,M,K,MP)
          J=(MN-1)/IAX+1 $ I=MN-(J-1)*IAX $ IN=IM(NT+IN*3)
          JX=INTS*(KON-F(MN))/(F(NN)-F(MN)) $ CALL CUSP (MN,NN)
          LEX=INTS*(I-1)+JX*(2-KN) $ LEY=INTS*(J-1)+JX*(KN-1)
C . . . ARE TWO SUCCESSIVE POINTS TOO CLOSE TOGETHER.
          JX=(LEX-KUT(KT))**2+(LEY-KUT(KT+1))**2 $  IF(JX.LT.50)GO TO 410
          KT=KT+2 $ NP=NP+1 $ GO TO 420
  410     IF((NO*(I-3)*(J-3)*(I-IAXMM)*(J-JAXMM)).NE.0)GO TO 670
  420     KUT(KT)=LEX $ KUT(KT+1)=LEY $ CALL LGL (PEN,NP,0,MP)
C. . .  DOES THIS LINE CLOSE ON ITSELF. IF SO THEN TERMINATE IT.
          JX=(LEX-LIX(LINE))**2+(LEY-LIY(LINE))**2 $ IF(JX.LT.LIM)JS=-1
          IF(KT.LE.7490.AND.NP-NI.LE.195)GO TO 670
C . . . SUSPEND LINE AND STORE PARAMETERS.
          LOOP(LINE)=0 $ KTST=KT-6 $ NST=N $ KST=K $ JS=-1
  670     NO=0 $ IF(I.EQ.3.OR.J.EQ.3.OR.I.EQ.IAXMM.OR.J.EQ.JAXMM)NO=2
          CALL LGL (FL,M,K,0) $ IF(JS.LT.0)250,230
C . . . RECOMMENCE LINE AND RECALL PARAMETERS.
```

```
  700      LINE=LINE+1 $ LRF(LINE)=MOL(KK) $ LKI(LINE)=KT=KT+2
           NP=(KT+1)/2 $ N=NST $ NST=(KTST+1)/2 $ LOOP(LINE)=0
           KUT(KT)=LIX(LINE)=KUT(KTST) $ KUT(KT+1)=LIY(LINE)=KUT(KTST+1)
           CALL LGC (PEN,NST,0,MP) $ CALL LGL (PEN,NP,0,MP)
               DO 720 K=1,2 $ KT=KT+2 $ NP=NP+1
               KUT(KT)=KUT(KTST+2*K) $ KUT(KT+1)=KUT(KTST+2*K+1)
               CALL LGC (PEN,NST+K,0,MP) $ CALL LGL (PEN,NP,0,MP)
  720          CONTINUE
           KTST=0 $ K=KST $ GO TO 230
  780      IF(IPRNT.NE.0)PRINT 790
  790      FORMAT (* NO MORE CUTS *)
  800      CONTINUE $ KT=-1
  810  IF(IPRNT.NE.0)PRINT 820,LINE $ IF(LINE.EQ.0)GO TO 960
  820  FORMAT (* NUMBER OF LINES = *I3)
C . . . MINIMISE PEN MOVEMENTS.
           DO 830 J=1,4
  830      USED(J)=0
       JX=JY=0 $ LIN(1)=1 $ LDF(1)=IDI(1) $ IF(LINE.EQ.1)GO TO 870
           DO 860 J=1,LINE $ I=LARGE
               DO 850 K=1,LINE
               CALL LGC (USED,K,0,MP) $ IF(MP.EQ.1)GO TO 850
               KN=ISQRT((JX-LIX(K))**2+(JY-LIY(K))**2)
               KO=ISQRT((JX-LFX(K))**2+(JY-LFY(K))**2)
               IF(KN.GE.I)GO TO 840 $ I=KN $ LOF=1 $ N=K
  840          IF(KO.GE.I)GO TO 850 $ I=KO $ LOF=0 $ N=K
  850          CONTINUE $ IF(I.GE.LARGE)GO TO 860
           CALL LGL (USED,N,0,1) $ LIN(J)=N
           LDF(J)=IDI(N) $ IF(LOF.EQ.0)LDF(J)=LDF(J)-3
           IF(LRF(N).EQ.LRF(N)/2*2)LOF=1-LOF
           JX=(1-LOF)*LIX(N)+LOF*LFX(N) $ JY=(1-LOF)*LIY(N)+LOF*LFY(N)
  860      CONTINUE
  870  IF(IPRNT.EQ.0)GO TO 940
       IF(JPRNT.EQ.0)GO TO 920 $ PRINT 890
  890  FORMAT (1H0,15X*LINE*6X*X(1)*6X*Y(1)*6X*X(N)*6X*Y(N)*7X*CUT*4X,
      1        *POINTS*5X,1H*,5X*ORDER*6X*LINE*5X*DIRECTION*)
           DO 910 K=1,LINE
           PRINT 900,K,LIX(K),LIY(K),LFX(K),LFY(K),LKI(K),NOP(K)
      1             ,K,LIN(K),LDF(K)
  900      FORMAT (10X,7I10,5X,1H*,3I10)
  910      CONTINUE
  920  TIMES=TIMEF(CLOK)-TUN $ PRINT 930,TIMES
  930  FORMAT (* DRAWLIN ENTERED AT TIME ....*F10.0)
  940  CALL DRAWLIN
       IF(KT.LT.0)GO TO 960 $ LINE=NI=0 $ KT=-1 $ PRINT 950 $ GO TO 150
  950  FORMAT (* MORE THAN 200 LINES OR 7000 CUTS IN BLOCK*)
  960  IF(IPRNT.EQ.0)RETURN $ TIMES=TIMEF(CLOK)-TUN $ PRINT 970,TIMES
  970  FORMAT (* LINKUT FINISHED AT TIME ....*F10.0)
       END
       SUBROUTINE DRAWLIN
C * * * THIS ROUTINE JOINS UP THE CONTOUR CUTS TO FORM T  PLOTTED LINES.* * * C
C - - - THIS EDITION DATED 25.3.77     THIS VERSION 21/10/77.
       COMMON/MAINB/LAREA,SCAX,SCAY,OLAT,OLON,L1,L2,M1,M2,AVERAG,RANGE
       COMMON/TT/NAME,TIMES,CLOK,IPRNT,JPRNT,TUN,LGT,LOTX,NRE(9)
       COMMON/XYCOORD/TLT,BLT,WLN,ELN,SR,CR,ALT,ALN,XX,YY,XL,XR,YT,YB,S00
       COMMON/ALAKP/F(2000),KUT(7500),LVL(40),MOL(40)
       COMMON/DRAW/NOP,LOOP,LDF,LIN,LKI,LRF,PEN
       COMMON/LINX/X2,Y2,X3,Y3
       COMMON/SCOL/SCINT,SCALE,OX,OY,LINES,IAX,JAX,NOC,INHIB,IT
       DIMENSION NOP(200),LOOP(200),LDF(200),LIN(200),LKI(200),LRF(200)
       DIMENSION X(200),Y(200),JPEN(200),H(200),H2(200),S(200),XDEL(200)
       DIMENSION YDEL(200),X2(200),Y2(200),X3(200),Y3(200),PEN(80)
       DATA (EPSLN=.005),(S(1)=0.),(X2(1)=0.),(Y2(1)=0.)
```

```
      NRE(9)=&HDRL/14,  $ NTK=0
          DO 250 L=1,LINES $ LINE=LIN(L) $ NT=LRF(LINE)
          IF(IPRNT.NE.0)PRINT 10,LINE,LDF(L),LOOP(LINE)
   10     FORMAT (15X,*LINE = *I4* DIRECTION = *I4* LOOP = *I4)
          IF(NT.EQ.0)GO TO 250 $ N=NOP(LINE) $ N1=N-1 $ N2=N-2
          KT=LKI(LINE)-2 $ NP=(KT+1)/2 $ X2(N)=Y2(N)=0.0
C . . . TRANSFORM THE CUT COORDINATES INTO PLOTTER UNITS.
              DO 20 J=1,N $ KT=KT+2 $ NP=NP+1
              CALL LGC (PEN,NP,0,MP) $ JPEN(J)=MP
              ALN=(KUT( KT )/SCINT+OX)/SCAX+WLN
              ALT=(KUT(KT+1)/SCINT+OY)/SCAY+TLT
              CALL COORDS $ XX=XX-XL $ YY=YY-YB
              X(J)=SR*YY+CR*XX $ Y(J)=CR*YY-SR*XX
   20         CONTINUE  $ IF(LOOP(LINE).EQ.0)GO TO 25
          N2=N $ N1=N+1 $ N=N+2 $ X(N)=X(3) $ Y(N)=Y(3)
          X(N1)=X(2) $ Y(N1)=Y(2) $ JPEN(1)=JPEN(N)=1 $ JPEN(N1)=JPEN(2)
C . . . CALCULATE THE SLOPE OF THE LINE AND THE SEGMENT LENGTHS.
   25         DO 35 J=1,N1
              H(J)=SQRT((X(J+1)-X(J))**2+(Y(J+1)-Y(J))**2)
              S(J+1)=H(J)+S(J)
              IF(H(J).NE.0)GO TO 30
              XDEL(J)=YDEL(J)=0.0 $ GO TO 35
   30         XDEL(J)=(X(J+1)-X(J))/H(J)
              YDEL(J)=(Y(J+1)-Y(J))/H(J)
   35         CONTINUE
              DO 45 J=2,N1
              H2(J)=H(J-1)+H(J) $ IF(H2(J).NE.0)GO TO 40
              X2(J)=Y2(J)=X3(J)=Y3(J)=0 $ GO TO 45
   40         X2(J)=(XDEL(J)-XDEL(J-1))/H2(J) $ X3(J)=3.0*X2(J)
              Y2(J)=(YDEL(J)-YDEL(J-1))/H2(J) $ Y3(J)=3.0*Y2(J)
              H2(J)=H(J-1)/H2(J)
   45         CONTINUE
   50     ETA=0.0
              DO 60 J=2,N1
              XW=(X3(J)-X2(J+1)+H2(J)*(X2(J+1)-X2(J-1))-2*X2(J))*1.0718
              IF(ABS(XW).LE.ETA)GO TO 60
              ETA=ABS(XW) $ X2(J)=X2(J)+XW/2.0
   60         CONTINUE
          IF(ETA.GE.EPSLN)GO TO 50
   70     ETA=0.0
              DO 80 J=2,N1
              YW=(Y3(J)-Y2(J+1)+H2(J)*(Y2(J+1)-Y2(J-1))-2*Y2(J))*1.0718
              IF(ABS(YW).LE.ETA)GO TO 80
              ETA=ABS(YW) $ Y2(J)=Y2(J)+YW/2.0
   80         CONTINUE
          IF(ETA.GE.EPSLN)GO TO 70
              DO 100 J=1,N1
              IF(H(J).NE.0)GO TO 90
              X3(J)=Y3(J)=0.0 $ GO TO 100
   90         X3(J)=(X2(J+1)-X2(J))/H(J)
              Y3(J)=(Y2(J+1)-Y2(J))/H(J)
  100         CONTINUE
          NH=3-2*IABS(LDF(L)) $ IF(LDF(L).LT.0)GO TO 170
C . . . SEARCH CUTS FORWARD.
  110     CALL PLOT (X(2),Y(2),3)
C . . . CALCULATE THE INCREMENTAL STEPS ALONG THE LINE AND MOVE THE PEN.
  130         DO 160 J=2,N2 $ NP=0
              IF(JPEN(J).NE.0.OR.JPEN(J+1).NE.0)GO TO 160
              IF(JPEN(J-1).NE.0)CALL PLOT (X(J),Y(J),3)
              LTK=H(J)*SCALE-0.5
              IF(LTK.LE.0)GO TO 150
              DIS=H(J)/(LTK+1) $ HT1=0 $ IF(IT.EQ.1)NTK=LTK/2-1
```

```
                DO 140 K=1,LTK
                HT1=HT1+DIS
                HT2=HT1*(HT1-H(J))
                XX2=(2.0*X2(J)+X2(J+1)+HT1*X3(J))/3.0
                XP=X(J)+HT1*XDEL(J)+XX2*HT2
                YY2=(2.0*Y2(J)+Y2(J+1)+HT1*Y3(J))/3.0
                YP=Y(J)+HT1*YDEL(J)+YY2*HT2
                CALL PLOT (XP,YP,2) $ IF(K.NE.NTK)GO TO 140
                IF(NP.EQ.0)GO TO 135
                XO=XO-XP $ YO=YP-YO $ ETA=40.0*SQRT(XO**2+YO**2)
                CALL PLOT (XP+YO/ETA*NH,YP+XO/ETA*NH,2)
                CALL PLOT (XP,YP,2) $ GO TO 140
  135           XO=XP $ YO=YP $ NP=1 $ NTK=NTK+1
  140           CONTINUE
  150         CALL PLOT (X(J+1),Y(J+1),2)
  160         CONTINUE
            NT=NT-1 $ IF(NT.EQ.0)GO TO 250
            IF(LOOP(LINE).EQ.1)GO TO 130 $ NH=0
C . . . SEARCH CUTS BACKWARD.
  170       CALL PLOT (X(N1),Y(N1),3)
C . . . CALCULATE THE INCREMENTAL STEPS ALONG THE LINE AND MOVE THE PEN.
  190         DO 220 JJ=2,NZ $ J=N-JJ+1 $ NP=0
              IF(JPEN(J).NE.0.OR.JPEN(J-1).NE.0)GO TO 220
              IF(JPEN(J+1).NE.0)CALL PLOT (X(J),Y(J),3)
              LTK=H(J-1)*SCALE-0.5
              IF(LTK.LE.0)GO TO 210
              DIS=H(J-1)/(LTK+1) $ HT2=0 $ IF(IT.EQ.1)NTK=LTK/2-1
                  DO 200 K=1,LTK
                  HT2=HT2-DIS
                  HT1=HT2+H(J-1)
                  XX2=(2.0*X2(J-1)+X2(J)+HT1*X3(J-1))/3.0
                  XP=X(J-1)+HT1*XDEL(J-1)+HT1*HT2*XX2
                  YY2=(2.0*Y2(J-1)+Y2(J)+HT1*Y3(J-1))/3.0
                  YP =Y(J-1)+HT1*YDEL(J-1)+HT1*HT2*YY2
                  CALL PLOT (XP,YP,2) $ IF(K.NE.NTK)GO TO 200
                  IF(NP.EQ.0)GO TO 195
                  XO=XO-XP $ YO=YP-YO $ ETA=40.0*SQRT(XO**2+YO**2)
                  CALL PLOT (XP+YO/ETA*NH,YP+XO/ETA*NH,2)
                  CALL PLOT (XP,YP,2) $ GO TO 200
  195             XO=XP $ YO=YP $ NP=1 $ NTK=NTK+1
  200             CONTINUE
  210           CALL PLOT (X(J-1),Y(J-1),2)
  220           CONTINUE
              NT=NT-1 $ IF(NT.EQ.0)GO TO 250
              IF(LOOP(LINE).EQ.1)GO TO 170 $ NH=0 $ GO TO 110
  250       CONTINUE
        END
        SUBROUTINE LGL (M,I,J,N)
C . . . LOGICAL ARRAYS WITH 60 VALUES PER WORD ARE ADDRESSED BY BITS.     . . . C
C . . . VARIABLES ARE ARRAY M, INDICES I + J, LOGICAL VALUE N.
C . . . IF J = 0 ARRAY IS SINGLE DIMENSIONED.
C . . . ENTRY LGL LOADS BIT N INTO M(J,I), LGC CALLS BIT N FROM M(J,I).
C - - - ROUTINE WRITTEN 24.3.77
        DIMENSION M(1)
   10   L=1
C . . . CALCULATE WORD BIT ADDRESS.
        LOC=I-1 $ IF(J.NE.0)LOC=2*LOC+J-1
        IND=LOC/60 $ LOC=LOC-IND*60
        K=SHIFT(M(IND+1),LOC)$ IF(N.GE.0)GO TO 20
        N=L.AND.K $ RETURN
   20   IF(N.NE.0)N=L $ K=K.AND.(.NOT.L) $ K=K.OR.N
        LOC=60-LOC $ M(IND+1)=SHIFT(K,LOC) $ RETURN
```

```
      ENTRY LGC
      N=-1 $ GO TO 10
      END
      SUBROUTINE CUSP (M,N)
C . . . CALCULATES THE CUBIC SPLINE CUT OF CONTOUR ACROSS GRID
      COMMON/ALAKP/F(2000),KUT(7500),LVL(40),MOL(40)
      COMMON/CUBIS/JX,KON,I,INTS,IAXM,JAXM
      INTEGER F,FX,X,DX,DDX
      INTH=INTS
   10 IF(I.LT.2.OR.I.GE.IAXM.OR.J.LT.2.OR.J.GE.JAXM)RETURN
C . . . CALCULATE CUT POINT BY NEWTONS RULE
      DDX=INTS $ I2=INTS*INTS $ I3=I2*INTS $ X=JX
      G1=(F(M)-F(2*M-N))/2 $ G2=(F(N)-F(M))/2 $ G3=(F(2*N-M)-F(N))/2
      A=(G1-2*G2+G3)/3 $ B=G2-G1 $ C=(2*G1+5*G2-G3)/3 $ D=F(M)
      IF(G1*G3.LT.0)GO TO 40
   20 X2=X*X $ X3=X2*X $ FX=KON-A*X3/I3-B*X2/I2-C*X/INTS-D
      DELFX=-3*A*X2/I2-2*B*X/INTS-C $ IF(ABS(DELFX).LT.1)RETURN
      DX=-INTS*FX/DELFX $ X=X+DX
      IF(IABS(DX).LT.10)GO TO 30 $ IF(IABS(DX).GT.IABS(DDX))RETURN
      DDX=DX $ GO TO 20
   30 IF(X.GE.0.AND.X.LE.INTH)JX=X
      RETURN
      ENTRY CUSPH $ INTH=INTS/2 $ GO TO 10
C . . . CHANGE OF GRADIENT - CALCULATE THE TURNING POINT.
   40 FX=B**2-3*A*C $ IF(FX.LT.0)GO TO 20
      IF(ABS(A).GT.1)GO TO 50 $ X=-INTS*C/(2*B) $ GO TO 60
   50 FX=ISQRT(FX) $ X=-INTS*(B-FX)/(3*A)
      IF(X.LE.INTS.AND.X.GE.0)GO TO 70 $ X=-INTS*(B+FX)/(3*A)
   60 IF(X.LE.INTS.AND.X.GE.0)GO TO 70 $ X=JX $ GO TO 20
   70 FX=A*X**3/I3+B*X**2/I2+C*X/INTS+D
      IF(IABS(KON-FX).LT.2)GO TO 30
      KF=F(M) $ DX=0 $ IF((F(M)-KON)*(KON-FX).GE.0)GO TO 80
      KF=F(N) $ DX=INTS
   80 X=DX+(X-DX)*(KON-KF)/(FX-KF) $ GO TO 20
      ENTRY MIDF
      DX=(N-M)/(IAXM+1)*(IAXM+1) $ JX=(9*(F(M)+F(M+DX)+F(N)+F(N-DX))
     1            -F(2*M-N)-F(2*M-N+3*DX)-F(2*N-M)-F(2*N-M-3*DX))/32
      END
```

INDEX

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
1	ABMINGA	G53-10	26	0	27	0	133	30	135	0	SOUTH AUSTRALIA	1/250000	4
2	ADAVALE	G55-5	25	0	26	0	144	0	145	30	QUEENSLAND	1/250000	6
3	ADELAIDE	I54-9	34	0	35	0	138	0	139	30	SOUTH AUSTRALIA	1/250000	8
4	AJANA	REF	27	0	28	0	113	30	115	30	WESTERN AUSTRALIA	1/250000	26
5	ALBANY	I50-15	35	0	36	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	28
6	ALBERGA	G53-9	26	0	27	0	132	0	133	30	SOUTH AUSTRALIA	1/250000	32
7	ALCOOTA	F53-10	22	0	23	0	134	30	135	0	NORTHERN TERRITORY	1/250000	34
8	ALROY	E53-15	19	0	20	0	135	0	136	30	NORTHERN TERRITORY	1/250000	46
9	ANA BRANCH	I54-7	33	0	34	0	141	0	142	30	NEW SOUTH WALES	1/250000	50
10	ANDAMOOKA	H53-12	30	0	31	0	136	30	138	0	SOUTH AUSTRALIA	1/250000	52
11	ANGLEDOOL	H55-7	29	0	30	0	147	0	148	30	NEW SOUTH WALES	1/250000	54
12	ANKETELL	F51-2	20	0	21	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	56
13	ARNHEM BAY	D53-3	12	0	13	0	135	0	136	30	NORTHERN TERRITORY	1/250000	62
14	ASHTON	D52-13	15	0	16	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	68
15	ATHERTON	E55-5	17	0	18	0	144	0	145	30	QUEENSLAND	1/250000	70
16	AUGATHELLA	G55-6	25	0	26	0	145	30	147	0	QUEENSLAND	1/250000	72
17	AUGUSTA	I50-9	34	0	35	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	74
18	AURUKUN	D54-7	13	0	14	0	141	0	142	30	QUEENSLAND	1/250000	78
19	AUVERGNE	D52-15	15	0	16	0	129	0	130	30	NORTHERN TERRITORY	1/250000	80
20	AVON DOWNS	F53-4	20	0	21	0	136	30	138	0	NORTHERN TERRITORY	1/250000	82
21	AYERS ROCK	G52-8	25	0	26	0	130	30	132	0	NORTHERN TERRITORY	1/250000	86
22	AYR	E55-15	19	0	20	0	147	0	148	30	QUEENSLAND	1/250000	88
23	BAIRNSDALE	J55-7	37	0	38	0	147	0	148	30	VICTORIA	1/250000	112
24	BALLADONIA	I51-3	32	0	33	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	116
25	BALLARAT	J54-8	37	0	38	0	142	30	144	0	VICTORIA	1/250000	118
26	BALRANALD	I54-12	34	0	35	0	142	30	144	0	N.S.W. / VICTORIA	1/250000	120
27	BARALABA	G55-4	24	0	25	0	148	30	150	0	QUEENSLAND	1/250000	122
28	BARKER	I54-13	35	0	36	0	138	0	139	30	SOUTH AUSTRALIA	1/250000	124
29	BARLEE	H50-8	29	0	30	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	126
30	BARNATO	H55-13	31	0	32	0	144	0	145	30	NEW SOUTH WALES	1/250000	128
31	BARROLKA	G54-11	26	0	27	0	141	0	142	30	QUEENSLAND	1/250000	130
32	BARTON	H53-9	30	0	31	0	132	0	133	30	SOUTH AUSTRALIA	1/250000	144
33	BATHURST	I55-8	33	0	34	0	148	30	150	0	NEW SOUTH WALES	1/250000	146
34	BEDOURIE	G54-1	24	0	25	0	138	0	139	30	QUEENSLAND	1/250000	156
35	BEETALOO	E53-6	17	0	18	0	133	30	135	0	NORTHERN TERRITORY	1/250000	164
36	BEGA	REF	36	0	37	0	148	30	150	0	NEW SOUTH WALES	1/250000	166
37	BELELE	G50-11	26	0	27	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	168
38	BENCUBBIN	H50-11	30	0	31	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	170
39	BENDIGO	J55-1	36	0	37	0	144	0	145	30	VICTORIA	1/250000	172
40	BENTLEY	G52-5	25	0	26	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	174
41	BETOOTA	G54-6	25	0	26	0	139	30	141	0	QUEENSLAND	1/250000	176
42	BILLILUNA	E52-14	19	0	20	0	127	30	129	0	NORTHERN TERRITORY	1/250000	180
43	BIRDSVILLE	G54-5	25	0	26	0	138	0	139	30	QUEENSLAND	1/250000	182
44	BIRKSGATE	G52-15	27	0	28	0	129	0	130	30	SOUTH AUSTRALIA	1/250000	184
45	BIRRINDUDU	E52-11	18	0	19	0	129	0	130	30	NORTHERN TERRITORY	1/250000	186
46	BLACKALL	G55-1	24	0	25	0	144	0	145	30	QUEENSLAND	1/250000	188
47	BOOLIGAL	I55-5	33	0	34	0	144	0	145	30	NEW SOUTH WALES	1/250000	206
48	BOORABBIN	H51-13	31	0	32	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	208
49	BOULIA	F54-10	22	0	23	0	139	30	141	0	QUEENSLAND	1/250000	218
50	BOURKE	H55-10	30	0	31	0	145	30	147	0	NEW SOUTH WALES	1/250000	220

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
51	BOWEN	F55-3	20	0	21	0	147	0	148	30	QUEENSLAND	1/250000	224
52	BREADEN	SEE 442											228
53	BREMER BAY	I50-12	34	0	35	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	230
54	BRISBANE	G56-15	27	0	28	0	153	0	154	30	QUEENSLAND	1/250000	234
55	BROOME	E51-6	17	0	18	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	242
56	BROWNE	G51-8	25	0	26	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	246
57	BUCHANAN	F55-6	21	0	22	0	145	30	147	0	QUEENSLAND	1/250000	256
58	BULLER	G51-1	24	0	25	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	262
59	BULLOO	H54-4	28	0	29	0	142	30	144	0	QUEENSLAND	1/250000	264
60	BUNDABERG	G56-2	24	0	25	0	151	30	153	0	QUEENSLAND	1/250000	268
61	BURKETOWN	E54-6	17	0	18	0	139	30	141	0	QUEENSLAND	1/250000	270
62	BURNABBIE	I52-1	32	0	33	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	272
63	BURNIE	(K55-3)	40	45	41	45	144	50	146	50	TASMANIA	1/250000	274
64	BURRA	I54-5	33	0	34	0	138	0	139	30	SOUTH AUSTRALIA	1/250000	276
65	BYRO	G50-10	26	0	27	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	282
66	BUSSELTON	I50-5	33	0	34	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	278
67	CAIRNS	E55-2	16	0	17	0	145	30	147	0	QUEENSLAND	1/250000	306
68	CALLABONNA	H54-6	29	0	30	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	308
69	CAMOOWEAL	E54-13	19	0	20	0	138	0	139	30	QUEENSLAND	1/250000	320
70	CANBERRA	I55-16	35	0	36	0	148	30	150	0	A.C.T. / N.S.W.	1/250000	322
71	CANTERBURY	G54-7	25	0	26	0	141	0	142	30	QUEENSLAND	1/250000	326
72	CAPE ARID	I51-11	34	0	35	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	328
73	CAPE SCOTT	D52-7	13	0	14	0	129	0	130	30	NORTHERN TERRITORY	1/250000	340
74	CAPE YORK	C54-12	10	0	11	0	142	30	144	0	QUEENSLAND	1/250000	352
75	COBAR	H55-14	31	0	32	0	145	30	147	0	NEW SOUTH WALES	1/250000	388
76	CARGELLIGO	I55-6	33	0	34	0	145	30	147	0	NEW SOUTH WALES	1/250000	354
77	CHARNLEY	E51-4	16	0	17	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	364
78	CHILDARA	H53-14	31	0	32	0	133	30	135	0	SOUTH AUSTRALIA	1/250000	368
79	CHINCHILLA	G56-9	26	0	27	0	150	0	151	30	QUEENSLAND	1/250000	370
80	CHOWILLA	I54-6	33	0	34	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	374
81	CLERMONT	F55-11	22	0	23	0	147	0	148	30	QUEENSLAND	1/250000	378
82	CLONCURRY	F54-2	20	0	21	0	139	30	141	0	QUEENSLAND	1/250000	384
83	COBB	G52-1	24	0	25	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	390
84	COEN	D54-8	13	0	14	0	142	30	144	0	QUEENSLAND	1/250000	402
85	COLAC	J54-12	38	0	39	0	142	30	144	0	VICTORIA	1/250000	406
86	COLLIE	I50-6	33	0	34	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	408
87	COOK	H52-11	30	0	31	0	129	0	130	30	SOUTH AUSTRALIA	1/250000	416
88	COLLIER	G50-4	24	0	25	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	410
89	CONNEMARA	G54-3	24	0	25	0	141	0	142	30	QUEENSLAND	1/250000	412
90	COOKTOWN	D55-13	15	0	16	0	144	0	145	30	QUEENSLAND	1/250000	418
91	COOMPANA	H52-15	31	0	32	0	129	0	130	30	SOUTH AUSTRALIA	1/250000	422
92	COOPER	G52-10	26	0	27	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	424
93	COPLEY	H54-9	30	0	31	0	138	0	139	30	SOUTH AUSTRALIA	1/250000	430
94	CORDILLO	G54-10	26	0	27	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	432
95	CORNISH	F52-1	20	0	21	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	434
96	CORRIGIN	I50-3	32	0	33	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	436
97	CROSSLAND	E51-16	19	0	20	0	124	0	125	30	WESTERN AUSTRALIA	1/250000	444
98	CROYDON	E54-11	18	0	19	0	141	0	142	30	QUEENSLAND	1/250000	446
99	CUE	G50-15	27	0	28	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	448
100	CULVER	I51-4	32	0	33	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	450

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
101	CUNDEELEE	H51-11	30	0	31	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	452
102	CUNNAMULLA	H55-2	28	0	29	0	145	30	147	0	QUEENSLAND	1/250000	454
103	DALBY	G56-13	27	0	28	0	150	0	151	30	QUEENSLAND	1/250000	480
104	CURDIMURKA	H53-8	29	0	30	0	136	30	138	0	SOUTH AUSTRALIA	1/250000	456
105	CURNAMONA	H54-14	31	0	32	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	458
106	DALHOUSIE	G53-11	26	0	27	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	482
107	DAMPIER	F50-2	20	0	21	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	486
108	DARWIN	D52-4	12	0	13	0	130	30	132	0	NORTHERN TERRITORY	1/250000	494
109	DELAMERE	D52-16	15	0	16	0	130	30	132	0	NORTHERN TERRITORY	1/250000	504
110	DENILIQUIN	I55-13	35	0	36	0	144	0	145	30	N.S.W. / VICTORIA	1/250000	506
111	DERBY	E51-7	17	0	18	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	508
112	DEVONPORT	K55-6+2	40	45	42	0	145	30	147	0	TASMANIA	1/250000	510
113	DOBBYN	F54-14	19	0	20	0	139	30	141	0	QUEENSLAND	1/250000	516
114	DUBBO	5-4	32	0	33	0	148	30	150	0	NEW SOUTH WALES	1/250000	532
115	DONGARA	1-5	29	0	30	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	518
116	DORRIGO	H56-10	30	0	31	0	151	30	153	0	NEW SOUTH WALES	1/250000	522
117	DRYSDALE	D52-9	14	0	15	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	526
118	DUARINGA	F55-16	23	0	24	0	148	30	150	0	QUEENSLAND	1/250000	530
119	DUCHESS	F54-6	29	0	30	0	139	30	141	0	QUEENSLAND	1/250000	534
120	DUKETON	G51-1	27	0	28	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	538
121	DUMBLEYUNG	I50-7	33	0	34	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	540
122	DUMMER	F51-4	20	0	21	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	542
123	EBAGOOLA	D54-12	14	0	15	0	142	30	144	0	QUEENSLAND	1/250000	566
124	EDDYSTONE	G55-7	25	0	26	0	147	0	148	30	QUEENSLAND	1/250000	568
125	EDEL	G49-12	26	0	27	0	112	30	114	0	WESTERN AUSTRALIA	1/250000	570
126	EDMUND	F50-14	23	0	24	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	574
127	EDJUDINA	H51-6	29	0	30	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	572
128	EINASLEIGH	E55-9	18	0	19	0	144	0	145	30	QUEENSLAND	1/250000	576
129	ELKEDRA	F53-7	21	0	22	0	135	0	136	30	NORTHERN TERRITORY	1/250000	578
130	ELLISTON	I53-6	33	0	34	0	133	30	135	0	SOUTH AUSTRALIA	1/250000	580
131	EMERALD	F55-15	23	0	24	0	147	0	148	30	QUEENSLAND	1/250000	582
132	ENNU A	H55-6	29	0	30	0	145	30	147	0	NEW SOUTH WALES	1/250000	584
133	EROMANGA	G54-12	26	0	27	0	142	30	144	0	QUEENSLAND	1/250000	586
134	ESPERANCE	I51-6	33	0	34	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	588
135	EUABALONG	SEE 76											594
136	EUCLA	H52-14	31	0	32	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	596
137	EULO	H55-1	28	0	29	0	144	0	145	30	QUEENSLAND	1/250000	600
138	EVERARD	G53-13	27	0	28	0	132	0	133	30	SOUTH AUSTRALIA	1/250000	602
139	FINKE	G53-6	25	0	26	0	133	30	135	0	NORTHERN TERRITORY	1/250000	632
140	FOG BAY	D52-3	12	0	13	0	129	0	130	30	NORTHERN TERRITORY	1/250000	642
141	FORBES	I55-7	33	0	34	0	147	0	148	30	NEW SOUTH WALES	1/250000	648
142	FORREST	H52-10	30	0	31	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	650
143	FREW RIVER	F53-3	20	0	21	0	135	0	136	30	NORTHERN TERRITORY	1/250000	656
144	FOWLER	H53-13	31	0	32	0	132	0	133	30	SOUTH AUSTRALIA	1/250000	652
145	FROME	H54-10	30	0	31	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	658
146	GAIRDNER	H53-15	31	0	32	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	680
147	GALBRAITH	E54-3	16	0	17	0	141	0	142	30	QUEENSLAND	1/250000	682
148	GALILEE	F55-10	22	0	23	0	145	30	147	0	QUEENSLAND	1/250000	684
149	GASON	G54-13	27	0	28	0	138	0	139	30	SOUTH AUSTRALIA	1/250000	688
150	GEORGETOWN	E54-12	18	0	19	0	142	30	144	0	QUEENSLAND	1/250000	692

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
151	GERALDTON	H50-1	28	0	29	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	694
152	GILBERTON	E54-16	19	0	20	0	142	30	144	0	QUEENSLAND	1/250000	696
153	GILES	H53-1	28	0	29	0	132	0	133	30	SOUTH AUSTRALIA	1/250000	698
154	GILGANDRA	H55-16	31	0	32	0	148	30	150	0	NEW SOUTH WALES	1/250000	700
155	GLENBURGH	G50-6	25	0	26	0	116	30	117	0	WESTERN AUSTRALIA	1/250000	702
156	GLENGARRY	G50-12	26	0	27	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	704
157	GOULBURN	I55-12	34	0	35	0	148	30	150	0	NEW SOUTH WALES	1/250000	712
158	GOVE	D53-4	12	0	13	0	136	30	138	0	NORTHERN TERRITORY	1/250000	714
159	GRAFTON	H56-6	29	0	30	0	151	30	153	0	NEW SOUTH WALES	1/250000	716
160	GUNANYA	F51-14	23	0	24	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	724
161	GYMPIE	E56-10+11	26	0	27	0	151	30	153	0	QUEENSLAND	1/250000	726
162	HALE RIVER	G53-3	24	0	25	0	135	0	136	30	NORTHERN TERRITORY	1/250000	752
163	HAMILTON	J54-7	37	0	38	0	141	0	142	30	VICTORIA	1/250000	756
164	HANN RIVER	D54-16	15	0	16	0	142	30	144	0	QUEENSLAND	1/250000	760
165	HASTINGS	H56-14+15	31	0	32	0	151	30	153	0	NEW SOUTH WALES	1/250000	764
166	HAY	I55-9	34	0	35	0	144	0	145	30	NEW SOUTH WALES	1/250000	766
167	HAY RIVER	F53-16	23	0	24	0	136	30	138	0	NORTHERN TERRITORY	1/250000	768
168	HELENA	F52-5	21	0	22	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	770
169	HENBURY	G53-1	24	0	25	0	132	0	133	30	NORTHERN TERRITORY	1/250000	774
170	HERBERT	G51-7	25	0	26	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	776
171	HILL RIVER	H50-9	30	0	31	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	788
172	HOBART	(K55-8)	42	45	43	45	146	30	148	30	TASMANIA	1/250000	790
173	HOLROYD	D54-11	14	0	15	0	141	0	142	30	QUEENSLAND	1/250000	796
174	HOMEBOIN	G55-15	27	0	28	0	147	0	148	30	QUEENSLAND	1/250000	798
175	HORSHAM	J54-3	36	0	37	0	141	0	142	30	VICTORIA	1/250000	800
176	HUCKITTA	F53-11	22	0	23	0	135	0	136	30	NORTHERN TERRITORY	1/250000	806
177	HUGHENDEN	F55-1	20	0	21	0	144	0	145	30	QUEENSLAND	1/250000	808
178	HUON	B55-11	6	0	7	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	810
179	HYDEN	I50-4	32	0	33	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	816
180	INGHAM	E55-10	18	0	19	0	145	30	147	0	QUEENSLAND	1/250000	842
181	INNAMINCKA	G54-14	27	0	28	0	139	30	141	0	SOUTH AUSTRALIA	1/250000	844
182	INNISFAIL	E55-6	17	0	18	0	145	30	147	0	QUEENSLAND	1/250000	846
183	INVERELL	H56-5	29	0	30	0	150	0	151	30	NEW SOUTH WALES	1/250000	848
184	IPSWICH	G56-14	27	0	28	0	151	30	153	0	QUEENSLAND	1/250000	850
185	IVANHOE	I55-1	32	0	33	0	144	0	145	30	NEW SOUTH WALES	1/250000	854
186	JACKSON	H50-12	30	0	31	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	870
187	JERICHO	F55-14	23	0	24	0	145	30	147	0	QUEENSLAND	1/250000	880
188	JERILDERIE	I55-14	35	0	36	0	145	30	147	0	NEW SOUTH WALES	1/250000	882
189	JUNDAH	G54-4	24	0	25	0	142	30	144	0	QUEENSLAND	1/250000	896
190	JUBILEE	H52-5	29	0	30	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	888
191	KALGOORLIE	H51-9	30	0	31	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	921
192	KATHERINE	D53-9	14	0	15	0	132	0	133	30	NORTHERN TERRITORY	1/250000	936
193	KIMBA	I53-7	33	0	34	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	948
194	KINGOONYA	H53-11	30	0	31	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	954
195	KINGSCOTE	I53-16	35	0	36	0	136	30	138	0	SOUTH AUSTRALIA	1/250000	956
196	KINGSTON	G51-10	26	0	27	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	958
197	KIRKALOCKA	H50-3	28	0	29	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	960
198	KULGERA	G53-5	25	0	26	0	132	0	133	30	NORTHERN TERRITORY	1/250000	966
199	KURNALPI	H51-10	30	0	31	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	968
200	LA GRANGE	E51-10	18	0	19	0	121	30	123	0	WESTERN AUSTRALIA	1/250000	992

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
201	LAKE EYRE	H53-4	28 0 29 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	996
202	LANSDOWNE	F52-5	17 0 18 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1012
203	LARRIMAH	D53-13	15 0 16 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1014
204	LAUNCESTON	(K55-4)	40 45 41 45 146 30 148 30	TASMANIA	1/250000	1016
205	LAVERTON	H51-2	28 0 29 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1018
206	LAWN HILL	E54-9	18 0 19 0 138 0 139 30	QUEENSLAND	1/250000	1020
207	LENNIS	G52-13	27 0 28 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1024
208	LEONORA	H51-1	28 0 29 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1026
209	LIMBUNYA	E52-7	17 0 18 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1030
210	LINCOLN	I53-11+15	34 0 35 0 135 0 136 30	SOUTH AUSTRALIA	1/250000	1032
211	LINDSAY	G52-16	27 0 28 0 130 30 132 0	SOUTH AUSTRALIA	1/250000	1034
212	LISSADELL	E52-2	16 0 17 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1036
213	LOUTH	H55-9	30 0 31 0 144 0 145 30	NEW SOUTH WALES	1/250000	1058
214	LONGREACH	F55-13	23 0 24 0 144 0 145 30	QUEENSLAND	1/250000	1050
215	LOONGANA	H52-9	30 0 31 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1052
216	LUCAS	F52-2	20 0 21 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1060
217	MACDONALD	F52-14	23 0 24 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1084
218	MACHATTIE	G54-2	24 0 25 0 139 30 141 0	QUEENSLAND	1/250000	1086
219	MANN	G52-11	26 0 27 0 129 0 130 30	SOUTH AUSTRALIA	1/250000	1122
220	MACKAY	F55-8	21 0 22 0 148 30 150 0	QUEENSLAND	1/250000	1088
221	MACKUNDA	F54-11	22 0 23 0 141 0 142 30	QUEENSLAND	1/250000	1090
222	MACLEAN	H56-7	29 0 30 0 153 0 154 30	NEW SOUTH WALES	1/250000	1092
223	MADLEY	G51-3	24 0 25 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1096
224	MADURA	H52-13	31 0 32 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1098
225	MAITLAND	I53-12	34 0 35 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	1104
226	MALCOLM	I51-7	33 0 34 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1106
227	MALLACOOTA	J55-8	37 0 38 0 148 30 150 0	VICTORIA / N.S.W.	1/250000	1110
228	MANARA	I54-4	32 0 33 0 142 30 144 0	NEW SOUTH WALES	1/250000	1112
229	MANDORA	E51-13	19 0 20 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1114
230	MANEROO	F54-16	23 0 24 0 142 30 144 0	QUEENSLAND	1/250000	1118
231	MANILLA	H56-9	30 0 31 0 150 0 151 30	NEW SOUTH WALES	1/250000	1120
232	MANUKA	F54-8	21 0 22 0 142 30 144 0	QUEENSLAND	1/250000	1124
233	MARBLE BAR	F50-8	21 0 22 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1126
234	MARREE	H54-5	29 0 30 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	1130
235	MASON	H52-6	29 0 30 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1134
236	MAURICE	H52-8	29 0 30 0 130 30 132 0	SOUTH AUSTRALIA	1/250000	1140
237	MELBOURNE	J55-5	37 0 38 0 144 0 145 30	VICTORIA	1/250000	1162
238	MENINDEE	I54-3	32 0 33 0 141 0 142 30	NEW SOUTH WALES	1/250000	1172
239	MENZIES	H51-5	29 0 30 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1174
240	MILDURA	I54-11	34 0 35 0 141 0 142 30	VICTORIA / N.S.W.	1/250000	1176
241	MILINGIMBI	D53-2	12 0 13 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1178
242	MILLUNGERA	E54-15	19 0 20 0 141 0 142 30	QUEENSLAND	1/250000	1180
243	MILPARINKA	H54-7	29 0 30 0 141 0 142 30	NEW SOUTH WALES	1/250000	1182
244	MINIGWAL	H51-7	29 0 30 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1184
245	MINILYA	F49-16	23 0 24 0 112 30 114 0	WESTERN AUSTRALIA	1/250000	1186
246	MITCHELL	G55-11	26 0 27 0 147 0 148 30	QUEENSLAND	1/250000	1190
247	MONTO	G56-1	24 0 25 0 150 0 151 30	QUEENSLAND	1/250000	1200
248	MOORA	H50-10	30 0 31 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1202
249	MOREE	H55-8	29 0 30 0 148 30 150 0	NEW SOUTH WALES	1/250000	1204
250	MORNINGTON	E54-1	16 0 17 0 138 0 139 30	QUEENSLAND	1/250000	1206

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
251	MORRIS	F51-16	23 0 24 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1212
252	MOSSMAN	E55-1	16 0 17 0 144 0 145 30	QUEENSLAND	1/250000	1214
253	MT BARKER	SEE 685				1268
254	MT ISA	SEE 260				1284
255	MT BRUCE	SEE 686				1270
256	MT COOLON	SEE 687				1272
257	MT DOREEN	SEE 688				1274
258	MT EGERTON	SEE 691				1278
259	MT EVELYN	SEE 694				1282
260	MOUNT ISA	F54-1	20 0 21 0 138 0 139 30	QUEENSLAND	1/250000	1240
261	MT LIEBIG	SEE 695				1286
262	MT MARUMBA	SEE 696				1288
263	MT PEAKE	SEE 697				1290
264	MT RAMSAY	SEE 700				1294
265	MT RENNIE	SEE 701				1296
266	MT THEO	SEE 267				1300
267	MOUNT THEO	F52-8	21 0 22 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1256
268	MT WHELAN	SEE 704				1302
269	MT YOUNG	SEE 705				1304
270	MUNDUBBERA	G56-5	25 0 26 0 150 0 151 30	QUEENSLAND	1/250000	1306
271	MUNRO	E51-14	19 0 20 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1308
272	MURGOO	G50-14	27 0 28 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1310
273	MUTTABURRA	F55-9	22 0 23 0 144 0 145 30	QUEENSLAND	1/250000	1314
274	MCDILLS	SEE 275				1146
275	MC DILLS	G53-7	25 0 26 0 135 0 136 30	NORTHERN TERRITORY	1/250000	1144
276	MCKINLAY	SEE 277				1150
277	MC KINLAY	F54-7	21 0 22 0 141 0 142 30	QUEENSLAND	1/250000	1148
278	NABBERU	G51-5	25 0 26 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1336
279	NAPPERBY	F53-9	22 0 23 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1340
280	NARACOORTE	J54-2+1	36 0 37 0 139 15 141 0	SOUTH AUSTRALIA	1/250000	1342
281	NARETHA	H51-16	31 0 32 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1344
282	NARRABRI	H55-12	30 0 31 0 148 30 150 0	NEW SOUTH WALES	1/250000	1346
283	NARRANDERA	I55-10	34 0 35 0 145 30 147 0	NEW SOUTH WALES	1/250000	1348
284	NARROMINE	I55-3	32 0 33 0 147 0 148 30	NEW SOUTH WALES	1/250000	1350
285	NEALE	H51-4	28 0 29 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1352
286	NEWCASTLE	I56-2	32 0 33 0 151 30 153 0	NEW SOUTH WALES	1/250000	1354
287	NEWDEGATE	I50-8	33 0 34 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1360
288	NEWMAN	F50-16	23 0 24 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1362
289	NINGALOO	F49-12	22 0 23 0 112 30 114 0	WESTERN AUSTRALIA	1/250000	1364
290	NINGHAN	H50-7	29 0 30 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1368
291	NOOLYEANA	G53-16	27 0 28 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	1372
292	NOONAERA	I52-2	32 0 33 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1374
293	NOONKANBAH	E51-12	18 0 19 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1376
294	NOORINA	H52-3	28 0 29 0 129 0 130 30	SOUTH AUSTRALIA	1/250000	1378
295	NORMANTON	E54-7	17 0 18 0 141 0 142 30	QUEENSLAND	1/250000	1384
296	NORSEMAN	I51-2	32 0 33 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1388
297	NULLAGINE	F51-5	21 0 22 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1390
298	NULLARBOR	H52-16	31 0 32 0 130 30 132 0	SOUTH AUSTRALIA	1/250000	1392
299	NUYTS	I53-1	32 0 33 0 132 0 133 30	SOUTH AUSTRALIA	1/250000	1396
300	NYMAGEE	I55-2	32 0 33 0 145 30 147 0	NEW SOUTH WALES	1/250000	1398

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
301	WYNGAN	H55-15	31 0 32 0 147 0 148 30	NEW SOUTH WALES	1/250000	1400
302	OATLANDS	(K55-6)	41 45 42 45 146 30 148 30	TASMANIA	1/250000	1424
303	OLARY	I54-2	32 0 33 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	1426
304	ONSLOW	REF	21 0 22 0 113 40 115 30	WESTERN AUSTRALIA	1/250000	1428
305	OODNADATTA	G53-15	27 0 28 0 135 0 136 30	SOUTH AUSTRALIA	1/250000	1430
306	OOLDEA	H52-12	30 0 31 0 130 30 132 0	SOUTH AUSTRALIA	1/250000	1434
307	ORFORD BAY	C54-16	11 0 12 0 142 30 144 0	QUEENSLAND	1/250000	1436
308	ORROROO	I54-1	32 0 33 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	1438
309	OUYEN	I54-15	35 0 36 0 141 0 142 30	VICTORIA	1/250000	1440
310	PARACHILNA	H54-13	31 0 32 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	1464
311	PERTH	H50-13+14	31 0 32 0 115 10 117 0	WESTERN AUSTRALIA	1/250000	1496
312	PEAK HILL	G50-8	25 0 26 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1468
313	PELLEW	D53-16	15 0 16 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1470
314	PEMBERTON	I50-10	34 0 35 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1474
315	PENDER	E51-2	16 0 17 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1478
316	PENOLA	J54-6+10	37 0 38 10 139 30 141 0	SOUTH AUSTRALIA	1/250000	1482
317	PERCIVAL	F51-8	21 0 22 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1484
318	PERENJORI	H50-6	21 0 22 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1494
319	PINE CREEK	D52-8	13 0 14 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1504
320	PINJARRA	I50-2	32 0 33 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1506
321	PINNAROO	I54-14	35 0 36 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	1508
322	PLUMRIDGE	H51-8	29 0 30 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1510
323	POOLOWANNA	G53-12	26 0 27 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	1514
324	POONCARIE	I54-8	33 0 34 0 142 30 144 0	NEW SOUTH WALES	1/250000	1516
325	PT AUGUSTA	SEE 713				1544
326	PT CLINTON	SEE 715				1546
327	RASON	H51-3	28 0 29 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1606
328	PORT DAVEY	(K55-7)	42 45 43 45 144 30 146 30	TASMANIA	1/250000	1524
329	PT DAVEY	SEE 328				1548
330	PT HEDLAND	SEE 716				1550
331	PORT KEATS	D52-11	14 0 15 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1530
332	RANKEN	E53-16	19 0 20 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1604
333	PT KEATS	SEE 331				1552
334	PORTLAND	J54-11	38 0 39 0 141 0 142 30	VICTORIA	1/250000	1532
335	PT LANGDON	SEE 717				1554
336	PROSERPINE	F55-4	20 0 21 0 148 30 150 0	QUEENSLAND	1/250000	1542
337	PYRAMID	F50-7	21 0 22 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1558
338	QUEENSTOWN	(K55-5)	41 45 42 45 144 30 146 30	TASMANIA	1/250000	1582
339	QUILPIE	G55-9	26 0 27 0 144 0 145 30	QUEENSLAND	1/250000	1584
340	QUOBBA	G49-4	24 0 25 0 112 30 114 0	WESTERN AUSTRALIA	1/250000	1586
341	RAWLINSON	G52-2	24 0 25 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1610
342	RED RIVER	E54-8	17 0 18 0 142 30 144 0	QUEENSLAND	1/250000	1612
343	RENMARK	I54-10	34 0 35 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	1614
344	RICHMOND	F54-4	20 0 21 0 142 30 144 0	QUEENSLAND	1/250000	1616
345	ROBERT	G51-11	26 0 27 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1618
346	ROBERTSON	F51-13	23 0 24 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1620
347	RODINGA	G53-2	24 0 25 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1632
348	ROMA	G55-12	26 0 27 0 148 30 150 0	QUEENSLAND	1/250000	1638
349	ROEBOURNE	F50-3	20 0 21 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1634
350	ROY HILL	F50-12	22 0 23 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1650

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
351	RUDALL	F51-10	22 0 23 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1652
352	RUNTON	F51-15	23 0 24 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1654
353	RYAN	F52-13	23 0 24 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1658
354	SANDSTONE	G50-16	27 0 28 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1694
355	SAHARA	F51-7	21 0 22 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1676
356	SALE	J55-11	38 0 39 0 147 0 148 30	VICTORIA	1/250000	1686
357	SANDY CAPE	G56-3	24 0 25 0 153 0 154 30	QUEENSLAND	1/250000	1696
358	SCOTT	G52-6	25 0 26 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1700
359	SEEMORE	H51-12	30 0 31 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1702
360	SHARK BAY	G49-8	25 0 26 0 112 30 114 0	WESTERN AUSTRALIA	1/250000	1706
361	SINGLETON	I56-1	32 0 33 0 150 0 151 30	NEW SOUTH WALES	1/250000	1712
362	SIR SAMUEL	G51-13	27 0 28 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1714
363	SMITHTON	K55-5+1	40 40 42 0 144 0 145 30	TASMANIA	1/250000	1716
364	SPRINGSURE	G55-3	24 0 25 0 147 0 148 30	QUEENSLAND	1/250000	1722
365	STANLEY	G51-6	25 0 26 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1726
366	SPRINGVALE	F54-14	23 0 24 0 139 30 141 0	QUEENSLAND	1/250000	1724
367	STANSMORE	F52-6	21 0 22 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1728
368	ST ARNAUD		SEE 728			1730
369	ST GEORGE		SEE 729			1732
370	SWAN HILL	I54-16	35 0 36 0 142 30 144 0	VICTORIA / N.S.W.	1/250000	1746
371	STRZELECKI	H54-2	28 0 29 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	1742
372	SURAT	G55-16	27 0 28 0 148 30 150 0	QUEENSLAND	1/250000	1744
373	SYDNEY	I56-5+6	33 0 34 0 150 0 151 50	NEW SOUTH WALES	1/250000	1748
374	TABLETOP	F51-11	22 0 23 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1762
375	TALBOT	G52-9	26 0 27 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1766
376	TAMWORTH	H56-13	31 0 32 0 150 0 151 30	NEW SOUTH WALES	1/250000	1774
377	TALLARINGA	H53-5	29 0 30 0 132 0 133 30	SOUTH AUSTRALIA	1/250000	1770
378	TAMBO	G55-2	24 0 25 0 145 30 147 0	QUEENSLAND	1/250000	1772
379	TANAMI	E52-15	19 0 20 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1776
380	TANGORIN	F55-5	21 0 22 0 144 0 145 30	QUEENSLAND	1/250000	1780
381	TARCOOLA	H53-10	30 0 31 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	1784
382	TAROOM	G55-8	25 0 26 0 148 30 150 0	QUEENSLAND	1/250000	1788
383	TENNANT CK		SEE 741			1794
384	THROSSELL	G51-15	27 0 28 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1802
385	TICKALARA	H54-3	28 0 29 0 141 0 142 30	QUEENSLAND	1/250000	1808
386	TOBERMORY	F53-12	22 0 23 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1810
387	TOOMPINE	G55-13	27 0 28 0 144 0 145 30	QUEENSLAND	1/250000	1812
388	TORRENS	H53-16	31 0 32 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	1814
389	TORRES ST		SEE 745			1816
390	TOWNSVILLE	E55-14	19 0 20 0 145 30 147 0	QUEENSLAND	1/250000	1824
391	TRAINOR	G51-2	24 0 25 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1828
392	TRUANT IS		SEE 746			1834
393	TUREE CK		SEE 747			1840
394	ULLADULLA	I56-13	35 0 36 0 150 0 151 30	NEW SOUTH WALES	1/250000	1846
395	URAL	F51-12	22 0 23 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1848
396	URANDANGI	F54-5	21 0 22 0 138 0 139 30	QUEENSLAND	1/250000	1850
397	URAPUNGA	D53-10	14 0 15 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1852
398	VERNON	H52-1	28 0 29 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1858
399	URISINO	H54-8	29 0 30 0 142 30 144 0	NEW SOUTH WALES	1/250000	1854
400	WAGGA		SEE 751			1868

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
401	WAIGEN	G52-14	27	0	28	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	1872
402	WALSH	E54-4	16	0	17	0	142	30	144	0	QUEENSLAND	1/250000	1878
403	WANNA	H52-2	28	0	29	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	1882
404	WALGETT	H55-11	30	0	31	0	147	0	148	30	NEW SOUTH WALES	1/250000	1874
405	WALLHALLOW	E53-7	17	0	18	0	135	0	136	30	NORTHERN TERRITORY	1/250000	1876
406	WANGARATTA	J55-2	36	0	37	0	145	30	147	0	VICTORIA	1/250000	1880
407	WARBURTON	J55-6	37	0	38	0	145	30	147	0	VICTORIA	1/250000	1884
408	WARRAGUL	J55-10+14	38	0	39	0	145	30	147	0	VICTORIA	1/250000	1886
409	WARRI	G51-4	24	0	25	0	125	30	126	0	WESTERN AUSTRALIA	1/250000	1890
410	WARRINA	H53-3	28	0	29	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	1892
411	WARWICK	H56-2	28	0	29	0	151	30	153	0	QLD / N.S.W.	1/250000	1894
412	WATERLOO	E52-3	16	0	17	0	129	0	130	30	NORTHERN TERRITORY	1/250000	1896
413	WAVE HILL	E52-8	17	0	18	0	130	30	132	0	NORTHERN TERRITORY	1/250000	1900
414	WESTWOOD	G51-16	27	0	28	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	1918
415	WEBB	F52-10	22	0	23	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	1902
416	WEIPA	D54-3	12	0	13	0	141	0	142	30	QUEENSLAND	1/250000	1903
417	WELLS	H52-4	28	0	29	0	130	30	132	0	SOUTH AUSTRALIA	1/250000	1906
418	WESSEL IS		SEE 752										1908
419	WHYALLA	I53-8	33	0	34	0	136	30	138	0	SOUTH AUSTRALIA	1/250000	1926
420	WIDE BAY	G56-7	25	0	26	0	153	0	154	30	QUEENSLAND	1/250000	1928
421	WILCANNIA	H54-16	31	0	32	0	142	30	144	0	NEW SOUTH WALES	1/250000	1932
422	WILSON	F52-9	22	0	23	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	1934
423	WILUNA	G51-9	26	0	27	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	1936
424	WINDORAH	G54-8	25	0	26	0	142	30	144	0	QUEENSLAND	1/250000	1940
425	WINTINNA	G53-14	27	0	28	0	133	30	135	0	SOUTH AUSTRALIA	1/250000	1948
426	WINTON	F54-12	22	0	23	0	142	30	144	0	QUEENSLAND	1/250000	1950
427	WOLLONGONG	I56-9	34	0	35	0	150	0	151	30	NEW SOUTH WALES	1/250000	1952
428	WOODROFFE	G52-12	26	0	27	0	130	30	132	0	SOUTH AUSTRALIA	1/250000	1962
429	WOORAMEL	G50-5	25	0	26	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	1964
430	WYANDRA	G55-14	27	0	28	0	145	30	147	0	QUEENSLAND	1/250000	1970
431	WYLOO	F50-10	22	0	23	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	1972
432	WYOLA	H52-7	29	0	30	0	129	0	130	30	SOUTH AUSTRALIA	1/250000	1974
433	YALGOO	H50-2	28	0	29	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	1976
434	YAMPI	E51-3	16	0	17	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	1978
435	YANREY	F50-9	22	0	23	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	1980
436	YANTABULLA	H55-5	29	0	30	0	144	0	145	30	NEW SOUTH WALES	1/250000	1982
437	YARDEA	I53-3	32	0	33	0	135	0	136	30	SOUTH AUSTRALIA	1/250000	1984
438	YARINGA	G50-9	26	0	27	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	1986
439	YARRALOOLA	F50-6	21	0	22	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	1988
440	YARRIE	F51-1	20	0	21	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	1990
441	YOUANMI	H50-4	28	0	29	0	138	30	140	0	WESTERN AUSTRALIA	1/250000	1992
442	YOWALGA	G51-12	26	0	27	0	124	30	126	0	WESTERN AUSTRALIA	1/250000	1994
443	ZANTHUS	H51-15	31	0	32	0	123	0	124	30	WESTERN AUSTRALIA	1/250000	1998
444	ABAU	C55-12	10	0	11	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	2
445	AITAPE	A54-15	3	0	4	0	141	0	142	30	PAPUA NEW GUINEA	1/250000	24
446	AMBUNTI	B54-4	4	0	5	0	142	30	144	0	PAPUA NEW GUINEA	1/250000	48
447	ARAWE	B55-12	6	0	7	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	58
448	AROA	C55-6	9	0	10	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	66
449	BOGIA	B55-1	4	0	5	0	144	0	145	30	PAPUA NEW GUINEA	1/250000	200
450	BOIGU	C54-7	9	0	10	0	141	0	142	30	AUSTRALIA / P.N.G.	1/250000	202

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
451	BUKA IS	SEE 921				258
452	BUNA	C55-3	8 0 9 0 147 0 148 30	PAPUA NEW GUINEA	1/250000	266
453	CALVADOS	C56-14	11 0 12 0 151 30 153 0	PAPUA NEW GUINEA	1/250000	310
454	CP NELSON	SEE 760				438
455	CP RAOULT	SEE 763				440
456	DARU	C54-8	9 0 10 0 142 30 144 0	AUSTRALIA / P.N.G.	1/250000	492
457	DEBOYNE	C56-10	10 0 11 0 151 30 153 0	PAPUA NEW GUINEA	1/250000	502
458	FLY RIVER	C54-3	8 0 9 0 141 0 142 30	PAPUA NEW GUINEA	1/250000	638
459	GASMATA	B56-9	6 0 7 0 150 0 151 30	PAPUA NEW GUINEA	1/250000	686
460	(HUON)	K55-15	43 0 44 0 147 0 148 30	TASMANIA	1/250000	812
461	KALO	C55-11	10 0 11 0 147 0 148 30	PAPUA NEW GUINEA	1/250000	923
462	KARAMUI	B55-9	6 0 7 0 144 0 145 30	PAPUA NEW GUINEA	1/250000	926
463	KAR KAR IS	SEE 773				928
464	KARKAR IS	SEE 773				930
465	KAVIENG	REF	2 0 3 0 149 50 151 30	PAPUA NEW GUINEA	1/250000	938
466	KIKORI	B55-13	7 0 8 0 144 0 145 30	PAPUA NEW GUINEA	1/250000	946
467	KIWAI	C54-4	8 0 9 0 142 30 144 0	PAPUA NEW GUINEA	1/250000	962
468	L KUTUBU	SEE 768				1040
469	L MURRAY	SEE 772				1044
470	MABUA	A56-10	2 0 3 0 151 30 153 0	PAPUA NEW GUINEA	1/250000	1082
471	MADANG	B55-6	5 0 6 0 145 30 147 0	PAPUA NEW GUINEA	1/250000	1094
472	MAER	C55-5	9 0 10 0 144 0 145 30	AUSTRALIA / P.N.G.	1/250000	1102
473	MARKHAM	B55-10	6 0 7 0 145 30 147 0	PAPUA NEW GUINEA	1/250000	1128
474	MAY RIVER	B54-3	4 0 5 0 141 0 142 30	PAPUA NEW GUINEA	1/250000	1142
475	NINIGO	A55-5	1 0 2 0 144 0 145 30	PAPUA NEW GUINEA	1/250000	1370
476	RAMU	B55-5	5 0 6 0 144 0 145 30	PAPUA NEW GUINEA	1/250000	1602
477	MASSAU IS	SEE 775				1136
478	NAMATANAI	(A56-14)	3 0 4 0 151 0 152 30	PAPUA NEW GUINEA	1/250000	1338
479	POMIO	B56-6	5 0 6 0 151 30 153 0	PAPUA NEW GUINEA	1/250000	1512
480	PT MORESBY	SEE 778				1556
481	RAGGI	B54-11	6 0 7 0 141 0 142 30	PAPUA NEW GUINEA	1/250000	1600
482	ROSSEL	C56-15+11	10 45 12 0 153 0 154 30	PAPUA NEW GUINEA	1/250000	1646
483	SABLE	C56-13	11 0 12 0 150 0 151 30	PAPUA NEW GUINEA	1/250000	1670
484	SAGSAG	SEE 485				1674
485	SAG SAG	B55-7	5 0 6 0 147 0 148 30	PAPUA NEW GUINEA	1/250000	1672
486	SALAMAUA	B55-15	7 0 8 0 147 0 148 30	PAPUA NEW GUINEA	1/250000	1684
487	SAMARAI	C56-9	10 0 11 0 150 0 151 30	PAPUA NEW GUINEA	1/250000	1688
488	SAMO	(A56-15)	3 0 4 0 152 30 154 0	PAPUA NEW GUINEA	1/250000	1690
489	TAUU	B57-1	4 0 5 0 156 0 157 30	PAPUA NEW GUINEA	1/250000	1792
490	TUFI	C55-8	9 0 10 0 148 30 150 0	PAPUA NEW GUINEA	1/250000	1838
491	VANIMO	A54-11	2 0 3 0 141 0 142 30	PAPUA NEW GUINEA	1/250000	1856
492	VITU IS	SEE 780				1862
493	WABAG	B54-8	5 0 6 0 142 30 144 0	PAPUA NEW GUINEA	1/250000	1866
494	WAU	B55-14	7 0 8 0 145 30 147 0	PAPUA NEW GUINEA	1/250000	1898
495	WEWAK	A54-16	3 0 4 0 142 30 144 0	PAPUA NEW GUINEA	1/250000	1920
496	WUVULU IS	SEE 783				1966
497	YULE	C55-2	8 0 9 0 145 30 147 0	PAPUA NEW GUINEA	1/250000	1996
498	BARROW IS	SEE 606				136
499	DEAL IS	SEE 637				498
500	BARROW CK	SEE 605				132

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
501	BEDOUT IS		SEE 610			158
502	BROWSE IS		SEE 617			248
503	ALICE SPR		SEE 601			36
504	HERON IS		SEE 652			780
505	ILLOGWA CK		SEE 656			838
506	JULIA CK		SEE 659			890
507	KING IS		SEE 663			950
508	L AMADEUS		SEE 665			1008
509	L EYRE		SEE 201			1028
510	L JOHNSTON		SEE 666			1038
511	L MACKAY		SEE 667			1042
512	PERCY IS		SEE 784			1486
513	JOANNA SPR		SEE 658			884
514	SEPIK	A55-13	3 0 4 0 144 0 145 30	PAPUA NEW GUINEA	1/250000	1704
515	TALASEA	B56-5	5 0 6 0 150 0 151 30	PAPUA NEW GUINEA	1/250000	1764
516	KERSAINT	J53-4	36 0 37 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	945
517	(HOBART)	K55-11	42 0 43 0 147 0 148 30	TASMANIA	1/250000	791
518	FRASER IS		SEE 787			654
519	GROOTE IS		SEE 810			721
520	COBHAM		SEE 630			391
521						
522						
523						
524						
525						
526		I50-1	32 0 33 0 114 0 115 30	WESTERN AUSTRALIA	1/250000	856
527		I50-13	35 0 36 0 114 0 115 30	WESTERN AUSTRALIA	1/250000	857
528		I50-16	35 0 36 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	858
529		I51-8	33 0 34 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	859
530		I51-9	34 0 35 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	860
531		I51-12	34 0 35 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	861
532		I52-3	32 0 33 0 129 0 130 30	SOUTH AUSTRALIA	1/250000	862
533		I52-4	32 0 33 0 130 30 132 0	SOUTH AUSTRALIA	1/250000	863
534		I53-5	33 0 34 0 132 0 133 30	SOUTH AUSTRALIA	1/250000	864
535		I53-9	34 0 35 0 132 0 133 30	SOUTH AUSTRALIA	1/250000	865
536		I53-13	35 0 36 0 132 0 133 30	SOUTH AUSTRALIA	1/250000	868
537		I53-10	34 0 35 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	866
538		I53-14	35 0 36 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	869
539		I53-11	34 0 35 0 135 0 136 30	SOUTH AUSTRALIA	1/250000	867
540		I53-15	35 0 36 0 135 0 136 30	SOUTH AUSTRALIA	1/250000	870
541		J54-1	36 0 37 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	901
542		J54-2	36 0 37 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	902
543		J54-5	37 0 38 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	903
544		J54-6	37 0 38 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	904
545		J54-9	38 0 39 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	905
546		J54-10	38 0 39 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	906
547		J54-13	39 0 40 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	907
548		J54-14	39 0 40 0 139 30 141 0	SOUTH AUSTRALIA	1/250000	908
549		J54-15	39 0 40 0 141 0 142 30	VICTORIA	1/250000	909
550		J54-16	39 0 40 0 142 30 144 0	VICTORIA	1/250000	910

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
551	ADELAIDE*	I54	32 0 36 0 138 0 144 0	AUSTRALIA	1/1000000	10
552	ALBANY*	I50	32 0 36 0 114 0 120 0	AUSTRALIA	1/1000000	30
553	ALICE SPR*		SEE 871			38
554	ARMIDALE	H56	28 0 32 0 150 0 156 0	AUSTRALIA	1/1000000	60
555	BODALLA	J56	36 0 40 0 150 0 156 0	AUSTRALIA	1/1000000	198
556	BOURKE*	H55	28 0 32 0 144 0 150 0	AUSTRALIA	1/1000000	222
557	BRISBANE*	G56	24 0 28 0 150 0 156 0	AUSTRALIA	1/1000000	236
558	BROOME*	E51	16 0 17 0 120 0 126 0	AUSTRALIA	1/1000000	244
559	CANBERRA*	I55	32 0 36 0 144 0 150 0	AUSTRALIA	1/1000000	324
560	CARNARVON	G49	24 0 28 0 108 0 114 0	AUSTRALIA	1/1000000	356
561	CLERMONT*	F55	20 0 24 0 144 0 150 0	AUSTRALIA	1/1000000	380
562	CLOATES	F49	20 0 24 0 108 0 114 0	AUSTRALIA	1/1000000	382
563	CLONCURRY*	F54	20 0 24 0 138 0 144 0	AUSTRALIA	1/1000000	386
564	COOKTOWN*	D55	12 0 16 0 144 0 150 0	AUSTRALIA	1/1000000	420
565	DARWIN*	D52	12 0 16 0 126 0 132 0	AUSTRALIA	1/1000000	496
566	DU COUEDIC	J53	36 0 40 0 132 0 138 0	AUSTRALIA	1/1000000	536
567	ESPERANCE*	I51	32 0 36 0 120 0 126 0	AUSTRALIA	1/1000000	590
568	EYRE	I52	32 0 36 0 126 0 132 0	AUSTRALIA	1/1000000	604
569	HALLS CK		SEE 837			748
570	HAMILTON*	J54	36 0 40 0 138 0 144 0	AUSTRALIA	1/1000000	758
571	MELBOURNE*	J55	36 0 40 0 144 0 150 0	AUSTRALIA	1/1000000	1164
572	NORMANTON*	E54	16 0 20 0 138 0 144 0	AUSTRALIA	1/1000000	1386
573	PERTH*	H50	28 0 32 0 114 0 120 0	AUSTRALIA	1/1000000	1498
574	STOKES	K54	40 0 44 0 138 0 144 0	AUSTRALIA	1/1000000	1736
575	SYDNEY*	I56	32 0 36 0 150 0 156 0	AUSTRALIA	1/1000000	1750
576	TARCOOLA*	H53	28 0 32 0 132 0 138 0	AUSTRALIA	1/1000000	1786
577	TASMANIA	K55	40 0 44 0 144 0 150 0	AUSTRALIA	1/1000000	1790
578	WILUNA*	G51	24 0 28 0 120 0 126 0	AUSTRALIA	1/1000000	1938
579	TORRES ST*		SEE 855			1818
580	CHOISEUL	B57	4 0 8 0 156 0 162 0	SOLOMON ISLANDS	1/1000000	372
581	COCOS IS		SEE 926			394
582	FLY RIVER*	B54	4 0 8 0 138 0 144 0	PAPUA NEW GUINEA	1/1000000	640
583	HOLLANDIA		SEE 588			794
584	KAVIENG*	A56	0 0 4 0 150 0 156 0	PAPUA NEW GUINEA	1/1000000	940
585	LAE	B55	4 0 8 0 144 0 150 0	PAPUA NEW GUINEA	1/1000000	990
586	RABAUL	B56	4 0 8 0 150 0 156 0	PAPUA NEW GUINEA	1/1000000	1598
587	NORFOLK IS		SEE 862			1380
588	WEWAK*	A54	0 0 4 0 138 0 144 0	PAPUA NEW GUINEA	1/1000000	1922
589						
590						
591		J55-10	38 0 39 0 145 30 147 0	VICTORIA	1/250000	912
592		J55-13	39 0 40 0 144 0 145 30	VICTORIA	1/250000	914
593		J55-14	39 0 40 0 145 30 147 0	VICTORIA	1/250000	915
594		J55-15	39 0 40 0 147 0 148 30	VICTORIA	1/250000	916
595		J55-16	39 0 40 0 148 30 150 0	VICTORIA	1/250000	917
596		J55-12	38 0 39 0 148 30 150 0	VICTORIA	1/250000	913
597		K55-1	40 0 41 0 144 0 145 30	TASMANIA	1/250000	975
598		K55-2	40 0 41 0 145 30 147 0	TASMANIA	1/250000	976
599		K55-3	40 0 41 0 147 0 148 30	TASMANIA	1/250000	977
600		K55-4	40 0 41 0 148 30 150 0	TASMANIA	1/250000	978

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
601	ALICE SPRINGS	F53-14	23 0 24 0 133 30 135 0	NORTHERN TERRITORY	1/250000	40
602	ALLIGATOR RIVER	D53-1	12 0 13 0 132 0 133 30	NORTHERN TERRITORY	1/250000	44
603	ARNHEM BAY - GOVE	D53-3+4	12 0 13 0 135 0 136 30	NORTHERN TERRITORY	1/250000	64
604	BALFOUR DOWNS	F51-9	22 0 23 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	114
605	BARROW CREEK	F53-6	21 0 22 0 133 30 135 0	NORTHERN TERRITORY	1/250000	134
606	BARROW ISLAND	F50-1	20 0 21 0 114 0 115 30	WESTERN AUSTRALIA	1/250000	138
607	BATHURST IS	SEE 608				148
608	BATHURST ISLAND	C52-15	11 0 12 0 129 0 130 30	NORTHERN TERRITORY	1/250000	150
609	BAUHINIA DOWNS	E53-3	16 0 17 0 135 0 136 30	NORTHERN TERRITORY	1/250000	154
610	BEDOUT ISLAND	E50-16	19 0 20 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	160
611	BILLAKALINA	H53-7	29 0 30 0 135 0 136 30	SOUTH AUSTRALIA	1/250000	178
612	BLOODS RANGE	G52-3	24 0 25 0 129 0 130 30	NORTHERN TERRITORY	1/250000	190
613	BLUE MUD BAY	D53-7	13 0 14 0 135 0 136 30	NORTHERN TERRITORY	1/250000	192
614	BONNEY WELL	F53-2	20 0 21 0 133 30 135 0	NORTHERN TERRITORY	1/250000	204
615	BRIGHTON DOWNS	F54-15	23 0 24 0 141 0 142 30	QUEENSLAND	1/250000	232
616	BROKEN HILL	H54-15	31 0 32 0 141 0 142 30	NEW SOUTH WALES	1/250000	238
617	BROWSE ISLAND	D51-11	14 0 15 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	250
618	BRUNETTE DOWNS	E53-11	18 0 19 0 135 0 136 30	NORTHERN TERRITORY	1/250000	252
619	CALVERT HILLS	E53-8	17 0 18 0 136 30 138 0	NORTHERN TERRITORY	1/250000	312
620	CAMBRIDGE GULF	D52-14	15 0 16 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	314
621	CAMDEN SOUND	D51-15	15 0 16 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	316
622	CAPE BEATRICE	D53-12	14 0 15 0 136 30 138 0	NORTHERN TERRITORY	1/250000	330
623	CAPE MELVILLE	D55-9	14 0 15 0 144 0 145 30	QUEENSLAND	1/250000	332
624	CAPE VAN DIEMEN	E54-2	16 0 17 0 139 30 141 0	QUEENSLAND	1/250000	344
625	CAPE WEYMOUTH	D54-4	12 0 13 0 142 30 144 0	QUEENSLAND	1/250000	350
626	CHARLEVILLE	G55-10	26 0 27 0 145 30 147 0	QUEENSLAND	1/250000	358
627	CHARLOTTE WATERS	SEE 139				360
628	CHARTERS TOWERS	F55-2	20 0 21 0 143 30 145 0	QUEENSLAND	1/250000	366
629	CLARKE RIVER	E55-13	19 0 20 0 144 0 145 30	QUEENSLAND	1/250000	376
630	COBHAM LAKE	H54-11	30 0 31 0 141 0 142 30	NEW SOUTH WALES	1/250000	392
631	COBURG PENINSULAR	C53-13+9	10 50 12 0 132 0 133 30	NORTHERN TERRITORY	1/250000	393
632	COFFS HARBOUR	H56-11	30 0 31 0 153 0 154 30	NEW SOUTH WALES	1/250000	404
633	COOBER PEDY	H53-6	29 0 30 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	414
634	COOTAMUNDRA	I55-11	34 0 35 0 147 0 148 30	NEW SOUTH WALES	1/250000	428
635	DALY WATERS	E53-1	16 0 17 0 132 0 133 30	NORTHERN TERRITORY	1/250000	484
636	DAMPIER - YARRALOOLA	F50-2+6	20 20 22 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	490
637	DEAL ISLAND	(J55-15)	39 0 40 0 146 0 147 30	VICTORIA	1/250000	500
638	DIRRANBANDI	H55-3	28 0 29 0 147 0 148 30	QUEENSLAND	1/250000	512
639	DIXON RANGE	E52-6	17 0 18 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	514
640	DONORS HILL	E54-10	18 0 19 0 139 30 141 0	QUEENSLAND	1/250000	520
641	DURHAM DOWNS	G54-15	27 0 28 0 141 0 142 30	QUEENSLAND	1/250000	544
642	FERGUSSON RIVER	D52-12	14 0 15 0 130 30 132 0	NORTHERN TERRITORY	1/250000	631
643	FLINDERS ISLAND	(K55-2)	39 30 40 45 146 30 148 30	TASMANIA	1/250000	636
644	FLINDERS IS	SEE 643				634
645	GLENORMISTON	F54-9	22 0 23 0 138 0 139 30	QUEENSLAND	1/250000	706
646	GOONDIWINDI	H56-1	28 0 29 0 150 0 151 30	QLD / N.S.W.	1/250000	708
647	GORDON DOWNS	E52-10	18 0 19 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	710
648	GREEN SWAMP WELL	E53-13	19 0 20 0 132 0 133 30	NORTHERN TERRITORY	1/250000	719
649	HARDING SPRING	SEE 656				762
650	HELEN SPRINGS	E53-10	18 0 19 0 130 30 132 0	NORTHERN TERRITORY	1/250000	772

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
651	HERMANNSBURG	F53-13	23 0 24 0 132 0 133 30	NORTHERN TERRITORY	1/250000	778
652	HERON ISLAND	F56-14	23 0 24 0 151 30 153 0	QUEENSLAND	1/250000	782
653	IRWIN INLET	I50-14	35 0 36 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	852
654	HIGHLAND ROCKS	F52-7	21 0 22 0 129 0 130 30	NORTHERN TERRITORY	1/250000	786
655	HODGSON DOWNS	D53-14	15 0 16 0 135 30 135 0	NORTHERN TERRITORY	1/250000	792
656	ILLOGWA CREEK	F53-15	23 0 24 0 135 0 136 30	NORTHERN TERRITORY	1/250000	840
657	JARDINE RIVER	C54-15	11 0 12 0 141 0 142 30	QUEENSLAND	1/250000	878
658	JOANNA SPRING	F51-3	20 0 21 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	886
659	JULIA CREEK	F54-3	20 0 21 0 141 0 142 30	QUEENSLAND	1/250000	892
660	JUNCTION BAY	C53-14	11 0 12 0 133 30 135 0	NORTHERN TERRITORY	1/250000	894
661	KELLERBERRIN	H50-15	31 0 32 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	942
662	KENNEDY RANGE	G50-1	24 0 25 0 114 0 115 30	WESTERN AUSTRALIA	1/250000	944
663	KING ISLAND	(K55-1)	39 45 40 45 144 30 146 0	TASMANIA	1/250000	957
664	KOPPERAMANNA	H54-1	28 0 29 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	964
665	LAKE AMADEUS	G52-4	24 0 25 0 130 30 132 0	NORTHERN TERRITORY	1/250000	994
666	LAKE JOHNSTON	I51-1	32 0 33 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	998
667	LAKE MACKAY	F52-11	22 0 23 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1002
668	LANDER RIVER	F53-1	20 0 21 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1010
669	LENNARD RIVER	E51-8	17 0 18 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1022
670	LONDONDERRY	D52-5	13 0 14 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1046
671	MADURA - BURNABIE	REF	31 0 32 30 126 0 127 30	WESTERN AUSTRALIA	1/250000	1100
672	MALCOLM - CAPE ARID	I51-7+11	33 0 34 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1108
673	MARYBOROUGH	G56-6	25 0 26 0 151 30 153 0	QUEENSLAND	1/250000	1132
674	MEDUSA BANKS	D52-10	14 0 15 0 127 30 129 0	WESTERN AUSTRALIA	1/250000	1156
675	MEEKATHARRA		SEE 156			1158
676	MELVILLE ISLAND	C52-16	11 0 12 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1168
677	MELVILLE IS		SEE 676			1165
678	MONDRAIN ISLAND	I51-10	34 0 35 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1196
679	MONDRAIN IS		SEE 678			1194
680	MONTAGUE SOUND	D51-12	14 0 15 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1198
681	MOUNT ANDERSON	E51-11	18 0 19 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1218
682	MT ANDERSON		SEE 681			1264
683	MOUNT BANNERMAN	E52-13	19 0 20 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1220
684	MT BANNERMAN		SEE 683			1266
685	MOUNT BARKER	I50-11	34 0 35 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1222
686	MOUNT BRUCE	F50-1	22 0 23 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1226
687	MOUNT COOLON	F55-7	21 0 22 0 147 0 148 30	QUEENSLAND	1/250000	1228
688	MOUNT DOREEN	F52-12	22 0 23 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1230
689	MOUNT DRUMMOND	E53-12	18 0 19 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1232
690	MT DRUMMOND		SEE 689			1276
691	MOUNT EGERTON	G50-3	24 0 25 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1234
692	MOUNT ELIZABETH	E52-1	16 0 17 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1236
693	MT ELIZABETH		SEE 692			1280
694	MOUNT EVELYN	D53-5	13 0 14 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1238
695	MOUNT LIEBIG	F52-16	23 0 24 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1242
696	MOUNT MARUMBA	D53-6	13 0 14 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1244
697	MOUNT PEAKE	F53-5	21 0 22 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1246
698	MOUNT PHILLIPS	G50-2	24 0 25 0 115 30 117 0	WESTERN AUSTRALIA	1/250000	1248
699	MT PHILLIPS		SEE 698			1292
700	MOUNT RAMSAY	E52-9	18 0 19 0 126 0 127 30	WESTERN AUSTRALIA	1/250000	1250

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
701	MOUNT RENNIE	F52-15	23 0 24 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1252
702	MOUNT SOLITAIRE	F52-4	20 0 21 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1254
703	MT SOLITAIRE	SEE 702				1298
704	MOUNT WHELAN	F54-13	23 0 24 0 138 0 139 30	QUEENSLAND	1/250000	1258
705	MOUNT YOUNG	D53-15	15 0 16 0 135 0 136 30	NORTHERN TERRITORY	1/250000	1260
706	MURLOOCOPPIE	H53-2	28 0 29 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	1312
707	MC LARTY HILLS	E51-15	19 0 20 0 123 0 124 30	WESTERN AUSTRALIA	1/250000	1152
708	MCLARTY HILLS	SEE 707				1154
709	NEWCASTLE WATERS	E53-5	17 0 18 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1356
710	PANDIE PANDIE	G54-9	26 0 27 0 138 0 139 30	SOUTH AUSTRALIA	1/250000	1462
711	PATERSON RANGE	F51-6	21 0 22 0 121 30 123 0	WESTERN AUSTRALIA	1/250000	1466
712	PERCY ISLANDS	SEE 784				1488
713	PORT AUGUSTA	I53-4	32 0 33 0 136 30 138 0	SOUTH AUSTRALIA	1/250000	1518
714	PETERMANN RANGES	G52-7	25 0 26 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1500
715	PORT CLINTON	F56-9	22 0 23 0 150 0 151 30	QUEENSLAND	1/250000	1522
716	PORT HEDLAND	F50-4	20 0 21 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1526
717	PORT LANGDON	D53-8	13 0 14 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1534
718	PRINCE REGENT	D51-16	15 0 16 0 124 30 126 0	WESTERN AUSTRALIA	1/250000	1540
719	ROBINSON RANGE	G50-7	25 0 26 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1622
720	QUEENSCLIFF	J55-9	38 0 39 0 144 0 145 30	VICTORIA	1/250000	1580
721	RAVENSTHORPE	I51-5	33 0 34 0 120 0 121 30	WESTERN AUSTRALIA	1/250000	1608
722	ROPER RIVER	E53-4	16 0 17 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1624
723	ROCKHAMPTON	F56-13	23 0 24 0 150 0 151 30	QUEENSLAND	1/250000	1626
724	ROPER RIVER	D53-11	14 0 15 0 135 0 136 30	NORTHERN TERRITORY	1/250000	1640
725	RUTLAND PLAINS	D54-15	15 0 16 0 141 0 142 30	QUEENSLAND	1/250000	1656
726	SANDOVER RIVER	F53-8	21 0 22 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1692
727	SIMPSON DESERT NORTH	G53-4	24 0 25 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1708
728	SAINT ARNAUD	J54-4	36 0 37 0 142 30 147 0	VICTORIA	1/250000	1678
729	SAINT GEORGE	H55-4	28 0 29 0 148 30 150 0	QLD / N.S.W.	1/250000	1680
730	STREAKY BAY	I53-2	32 0 33 0 133 30 135 0	SOUTH AUSTRALIA	1/250000	1738
731	TALLANGATTA	J55-3	36 0 37 0 147 0 148 30	VICTORIA / N.S.W.	1/250000	1768
732	SIMPSON DESERT SOUTH	G53-8	25 0 26 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1710
733	SOUTHERN CROSS	H50-16	31 0 32 0 118 30 120 0	WESTERN AUSTRALIA	1/250000	1718
734	TANAMI EAST	E52-16	19 0 20 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1778
735	ST LAWRENCE	SEE 737				1734
736	SOUTH LAKE WOODS	E53-9	18 0 19 0 132 0 133 30	NORTHERN TERRITORY	1/250000	1720
737	SAINT LAWRENCE	F55-12	22 0 23 0 148 30 150 0	QUEENSLAND	1/250000	1682
738	TANUMBIRINI	E53-2	16 0 17 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1782
739	THURSDAY IS	SEE 744				1804
740	TWEED HEADS	H56-3	28 0 29 0 153 0 154 30	N.S.W. / QLD	1/250000	1844
741	TENNANT CREEK	E53-14	19 0 20 0 133 30 135 0	NORTHERN TERRITORY	1/250000	1796
742	THARGOMINDAH	G54-16	27 0 28 0 142 30 144 0	QUEENSLAND	1/250000	1798
743	THE GRANITES	F52-3	20 0 21 0 129 0 130 30	NORTHERN TERRITORY	1/250000	1800
744	THURSDAY ISLAND	C54-11	10 0 11 0 149 0 150 30	QUEENSLAND	1/250000	1806
745	TORRES STRAIT	C54-11+12	10 0 11 0 141 45 143 15	QUEENSLAND	1/250000	1820
746	TRUANT ISLAND	C53-16	11 0 12 0 136 30 138 0	NORTHERN TERRITORY	1/250000	1836
747	TUREE CREEK	F50-15	23 0 24 0 117 0 118 30	WESTERN AUSTRALIA	1/250000	1842
748	WINNECKE CK	SEE 749				1942
749	WINNECKE CREEK	E52-12	18 0 19 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1944
750	VICTORIA RIVER DOWNS	E52-4	16 0 17 0 130 30 132 0	NORTHERN TERRITORY	1/250000	1860

INDEX	NAME	NUMBER	LIMITS				REGION	SCALE	SEQUENCE
751	WAGGA WAGGA	I55-15	35 0	36 0	147 0	148 30	NEW SOUTH WALES	1/250000	1870
752	WESSEL ISLANDS	C53-15	11 0	12 0	135 0	136 30	NORTHERN TERRITORY	1/250000	1910
753	WESTMORELAND	E54-5	17 0	18 0	138 0	139 30	QUEENSLAND	1/250000	1916
754	WHITE CLIFFS	H54-12	30 0	31 0	142 30	144 0	NEW SOUTH WALES	1/250000	1924
755	WIDGIEMOOLTHA	H51-14	31 0	32 0	121 30	123 0	WESTERN AUSTRALIA	1/250000	1930
756	WINNING POOL	F50-13	23 0	24 0	114 0	115 30	WESTERN AUSTRALIA	1/250000	1946
757	FOWLER - NUYTS	REF	31 0	32 15	132 0	133 30	SOUTH AUSTRALIA	1/250000	653
758	ADMIRALTY IS EAST		SEE 919						14
759	AWORRA RIVER	B54-16	7 0	8 0	142 30	144 0	PAPUA NEW GUINEA	1/250000	84
760	CAPE NELSON	C55-4	8 0	9 0	148 30	150 0	PAPUA NEW GUINEA	1/250000	334
761	ADMIRALTY IS WEST		SEE 920						22
762	BLUCHER RANGE	B54-7	5 0	6 0	141 0	142 30	PAPUA NEW GUINEA	1/250000	191
763	CAPE RAOULT	B55-8	5 0	6 0	148 30	150 0	PAPUA NEW GUINEA	1/250000	336
764	BOUGAINVILLE NORTH		SEE 921						214
765	BUKA ISLAND		SEE 921						260
766	CP ST GEORGE		SEE 770						442
767	BOUGAINVILLE SOUTH		SEE 922						216
768	LAKE KUTUBU	B54-12	6 0	7 0	142 30	144 0	PAPUA NEW GUINEA	1/250000	1000
769	FERGUSSON IS		SEE 777						628
770	CAPE SAINT GEORGE	(B56-3)	4 0	5 0	152 30	154 0	PAPUA NEW GUINEA	1/250000	318
771	CAPE ST GEORGE		SEE 770						342
772	LAKE MURRAY	B54-15	7 0	8 0	141 0	142 30	PAPUA NEW GUINEA	1/250000	1006
773	KAR KAR ISLAND	B55-2	4 0	5 0	145 30	147 0	PAPUA NEW GUINEA	1/250000	932
774	KARKAR ISLAND		SEE 773						934
775	MASSAU ISLAND	(A55-8)	1 0	2 0	149 0	150 30	PAPUA NEW GUINEA	1/250000	1138
776	GAZELLE PENINSULA	(B56-2)	4 0	5 0	151 0	152 30	PAPUA NEW GUINEA	1/250000	690
777	FERGUSSON ISLAND	C56-5	9 0	10 0	150 0	151 30	PAPUA NEW GUINEA	1/250000	630
778	PORT MORESBY	C55-7	9 0	10 0	147 0	148 30	PAPUA NEW GUINEA	1/250000	1536
779	TROBRIAND IS		SEE 782						1830
780	VITU ISLANDS	B55-4	4 0	5 0	148 0	150 0	PAPUA NEW GUINEA	1/250000	1864
781	WOODLARK IS		SEE 785						1954
782	TROBRIAND ISLANDS	C56-1	8 0	9 0	150 0	151 30	PAPUA NEW GUINEA	1/250000	1832
783	WUVULU ISLAND	A54-8	1 0	2 0	142 30	144 0	PAPUA NEW GUINEA	1/250000	1968
784	PERCY ISLES	F56-5	21 0	22 0	150 0	151 30	QUEENSLAND	1/250000	1490
785	WOODLARK ISLAND	(C56-6)	8 30	9 30	151 45	153 15	PAPUA NEW GUINEA	1/250000	1958
786	EUCLA - NOONAERA	REF	31 0	32 15	127 30	129 0	WESTERN AUSTRALIA	1/250000	598
787	FRASER ISLAND	REF	24 30	26 0	152 30	153 45	QUEENSLAND	1/250000	655
788	GRAFTON - MACLEAN	H56-6+7	29 0	30 0	151 30	153 30	NEW SOUTH WALES	1/250000	718
789	BOWEN - PROSERPINE	F55-3+4	20 0	21 0	147 0	149 0	QUEENSLAND	1/250000	226
790	MOSSMAN - CAIRNS	E55-1+2	16 0	17 0	144 0	146 0	QUEENSLAND	1/250000	1216
791	IPSWICH - BRISBANE	G56-14+15	27 0	28 0	152 50	154 20	QUEENSLAND	1/250000	851
792	FOG BAY - DARWIN	D52-3+4	12 0	13 0	130 0	132 0	NORTHERN TERRITORY	1/250000	646
793	WELLESLEY ISLANDS	E54-1+2	16 0	17 0	139 0	140 0	QUEENSLAND	1/250000	1905
794	WELLESLEY IS		SEE 793						1904
795	CAPE LEEUWIN	I50-5+9	33 20	34 30	114 50	115 50	WESTERN AUSTRALIA	1/250000	331
796	ROEBOURNE - PYRAMID	F50-3+7	20 15	22 0	117 0	118 30	WESTERN AUSTRALIA	1/250000	1636
797	NINGALOO - YANREY	REF	22 0	23 0	113 30	115 30	WESTERN AUSTRALIA	1/250000	1366
798									
799									
800	GROOTE ISLAND		SEE 810						722

INDEX	NAME	NUMBER	LIMITS				REGION	SCALE	SEQUENCE
801	HOUTMAN ABROLHOS	H49-4	28 0	29 0	112 30	114 0	WESTERN AUSTRALIA	1/250000	802
802	HUON (TASMANIA)	SEE 460							814
803	(KING ISLAND)	REF	39 20	40 40	143 40	145 20	TASMANIA	1/250000	953
804	(FLINDERS ISLAND)	REF	39 20	40 45	147 10	148 40	TASMANIA	1/250000	637
805	(LAUNCESTON)	K55-7+3	40 40	42 0	147 0	148 30	TASMANIA	1/250000	1017
806	(QUEENSTOWN)	K55-9+10	42 0	43 0	145 0	147 0	TASMANIA	1/250000	1583
807	(PORT DAVEY)	K55-14	43 0	44 0	145 30	147 0	TASMANIA	1/250000	1525
808	KANGAROO ISLAND	REF	35 15	36 15	136 30	138 10	SOUTH AUSTRALIA	1/250000	925
809	KANGAROO IS	SEE 808							924
810	GROOTE EYLANDT	REF	13 30	14 30	136 0	137 15	NORTHERN TERRITORY	1/250000	720
811		K55-5	41 0	42 0	144 0	145 30	TASMANIA	1/250000	979
812		K55-6	41 0	42 0	145 30	147 0	TASMANIA	1/250000	980
813		K55-7	41 0	42 0	147 0	148 30	TASMANIA	1/250000	981
814		K55-8	41 0	42 0	148 30	150 0	TASMANIA	1/250000	982
815		K55-9	42 0	43 0	144 0	145 30	TASMANIA	1/250000	983
816		K55-10	42 0	43 0	145 30	147 0	TASMANIA	1/250000	984
817		K55-12	42 0	43 0	148 30	150 0	TASMANIA	1/250000	985
818		K55-13	43 0	44 0	144 0	145 30	TASMANIA	1/250000	986
819		K55-16	43 0	44 0	148 30	150 0	TASMANIA	1/250000	987
820		J55-4	36 0	37 0	148 30	150 0	NEW SOUTH WALES	1/250000	911
821		J56-1	36 0	37 0	150 0	151 30	NEW SOUTH WALES	1/250000	918
822		I56-5	33 0	34 0	150 0	151 30	NEW SOUTH WALES	1/250000	871
823		I56-6	33 0	34 0	151 30	153 0	NEW SOUTH WALES	1/250000	872
824		I56-10	34 0	35 0	151 30	153 0	NEW SOUTH WALES	1/250000	873
825		I56-14	35 0	36 0	151 30	153 0	NEW SOUTH WALES	1/250000	874
826		H56-14	31 0	32 0	151 30	153 0	NEW SOUTH WALES	1/250000	822
827		H56-15	31 0	32 0	153 0	154 30	NEW SOUTH WALES	1/250000	823
828		G56-10	26 0	27 0	151 30	153 0	QUEENSLAND	1/250000	733
829		G56-11	26 0	27 0	153 0	154 30	QUEENSLAND	1/250000	734
830		F56-10	22 0	23 0	151 30	153 0	QUEENSLAND	1/250000	666
831	ALICE SPRINGS*	F53	20 0	24 0	132 0	138 0	AUSTRALIA	1/1000000	42
832	BROKEN HILL*	H54	28 0	32 0	138 0	144 0	AUSTRALIA	1/1000000	240
833	BRUNSWICK BAY	D51	12 0	16 0	120 0	126 0	AUSTRALIA	1/1000000	254
834	CAPE WESSEL	C53	8 0	12 0	132 0	138 0	AUSTRALIA	1/1000000	348
835	CHARLEVILLE*	G55	24 0	28 0	144 0	150 0	AUSTRALIA	1/1000000	360
836	COOPER CREEK	G54	24 0	28 0	138 0	144 0	AUSTRALIA	1/1000000	426
837	HALLS CREEK	E52	16 0	20 0	126 0	132 0	AUSTRALIA	1/1000000	750
838	HAMERSLEY RANGE	F50	20 0	24 0	114 0	120 0	AUSTRALIA	1/1000000	754
839	HOUTMAN ABROLHOS*	H49	28 0	32 0	108 0	114 0	AUSTRALIA	1/1000000	804
840	KALGOORLIE*	H51	28 0	32 0	120 0	126 0	AUSTRALIA	1/1000000	837
841	LAKE MACKAY*	F52	20 0	24 0	126 0	132 0	AUSTRALIA	1/1000000	1004
842	MEEKATHARRA*	G50	24 0	28 0	114 0	120 0	AUSTRALIA	1/1000000	1160
843	MELVILLE ISLAND*	C52	8 0	12 0	126 0	132 0	AUSTRALIA	1/1000000	1170
844	MELVILLE IS*	SEE 843							1166
845	MITCHELL RIVER	D54	12 0	16 0	138 0	144 0	AUSTRALIA	1/1000000	1192
846	NEWCASTLE WATERS*	E53	16 0	20 0	132 0	138 0	AUSTRALIA	1/1000000	1358
847	NULLARBOR PLAIN	H52	28 0	32 0	126 0	132 0	AUSTRALIA	1/1000000	1394
848	OAKOVER RIVER	F51	20 0	24 0	120 0	126 0	AUSTRALIA	1/1000000	1422
849	OODNADATTA*	G53	24 0	28 0	132 0	138 0	AUSTRALIA	1/1000000	1432
850	PETERMANN RANGES*	G52	24 0	28 0	126 0	132 0	AUSTRALIA	1/1000000	1502

INDEX	NAME	NUMBER	LIMITS	REGION	SCALE	SEQUENCE
851	PORT AUGUSTA*	I53	32 0 36 0 132 0 138 0	AUSTRALIA	1/1000000	1520
852	ROCKHAMPTON*	F56	20 0 24 0 150 0 156 0	AUSTRALIA	1/1000000	1628
853	ROPER RIVER*	D53	12 0 16 0 132 0 138 0	AUSTRALIA	1/1000000	1642
854	ROWLEY SHOALS	E50	12 0 16 0 114 0 120 0	AUSTRALIA	1/1000000	1648
855	TORRES STRAIT*	C54	8 0 12 0 138 0 144 0	AUSTRALIA / P.N.G.	1/1000000	1822
856	TOWNSVILLE*	E55	16 0 20 0 144 0 150 0	AUSTRALIA	1/1000000	1826
857	ADMIRALTY ISLANDS	A55	0 0 4 0 144 0 150 0	PAPUA NEW GUINEA	1/1000000	16
858	ADMIRALTY IS		SEE 857			12
859	COCOS ISLANDS		SEE 926			396
860	GUADALCANAL	C57	8 0 12 0 156 0 162 0	SOLOMON ISLANDS	1/1000000	723
861	LORD HOWE ISLAND	H57	28 0 32 0 156 0 162 0	TASMAN SEA	1/1000000	1056
862	NORFOLK ISLAND	H58	28 0 32 0 162 0 168 0	TASMAN SEA	1/1000000	1382
863	LORD HOWE IS		SEE 861			1054
864	PORT MORESBY*	C55	8 0 12 0 144 0 150 0	PAPUA NEW GUINEA	1/1000000	1538
865	WOODLARK ISLAND*	C56	8 0 12 0 150 0 156 0	PAPUA NEW GUINEA	1/1000000	1960
866	WOODLARK IS*		SEE 865			1956
867						
868						
869						
870						
871		C49	8 0 12 0 108 0 114 0	INDIAN OCEAN	1/1000000	460
872		C50	8 0 12 0 114 0 120 0	INDIAN OCEAN	1/1000000	461
873		C51	8 0 12 0 120 0 126 0	INDIAN OCEAN	1/1000000	462
874		D49	12 0 16 0 108 0 114 0	INDIAN OCEAN	1/1000000	550
875		D50	12 0 16 0 114 0 120 0	INDIAN OCEAN	1/1000000	551
876		D56	12 0 16 0 150 0 156 0	TASMAN SEA	1/1000000	557
877		E49	16 0 20 0 108 0 114 0	INDIAN OCEAN	1/1000000	610
878		E56	16 0 20 0 150 0 156 0	TASMAN SEA	1/1000000	620
879		I49	32 0 36 0 108 0 114 0	INDIAN OCEAN	1/1000000	855
880		J49	36 0 40 0 108 0 114 0	INDIAN OCEAN	1/1000000	897
881		J50	36 0 40 0 114 0 120 0	SOUTHERN OCEAN	1/1000000	898
882		J51	36 0 40 0 120 0 126 0	SOUTHERN OCEAN	1/1000000	899
883		J52	36 0 40 0 126 0 132 0	SOUTHERN OCEAN	1/1000000	900
884		K49	40 0 44 0 108 0 114 0	INDIAN OCEAN	1/1000000	970
885		K50	40 0 44 0 114 0 120 0	SOUTHERN OCEAN	1/1000000	971
886		K51	40 0 44 0 120 0 126 0	SOUTHERN OCEAN	1/1000000	972
887		K52	40 0 44 0 126 0 132 0	SOUTHERN OCEAN	1/1000000	973
888		K53	40 0 44 0 132 0 138 0	SOUTHERN OCEAN	1/1000000	974
889		K56	40 0 44 0 150 0 156 0	TASMAN SEA	1/1000000	988
890		G48	24 0 28 0 102 0 108 0	INDIAN OCEAN	1/1000000	730
891		L54	44 0 48 0 138 0 144 0	SOUTHERN OCEAN	1/1000000	1070
892		L55	44 0 48 0 144 0 150 0	SOUTHERN OCEAN	1/1000000	1071
893		L56	44 0 48 0 150 0 156 0	SOUTHERN OCEAN	1/1000000	1072
894		J57	36 0 40 0 156 0 162 0	TASMAN SEA	1/1000000	920
895		I57	32 0 36 0 156 0 162 0	TASMAN SEA	1/1000000	875
896		G57	24 0 28 0 156 0 162 0	TASMAN SEA	1/1000000	735
897		F57	20 0 24 0 156 0 162 0	TASMAN SEA	1/1000000	667
898		E57	16 0 20 0 156 0 162 0	TASMAN SEA	1/1000000	621
899						
900						

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
901	PEMBERTON - IRWIN INLET	I50-10+14	34	0	35	15	115	30	117	0	WESTERN AUSTRALIA	1/250000	1476
902	MOUNT BARKER - ALBANY	I50-11+15	34	0	35	15	117	0	118	30	WESTERN AUSTRALIA	1/250000	1224
903	WESSEL ISLANDS - TRUANT ISLAND	C53-15+16	11	0	12	0	135	0	137	0	NORTHERN TERRITORY	1/250000	1914
904	ESPERANCE - MONDRAIN ISLAND	I56-6+10	33	0	34	15	121	30	123	0	WESTERN AUSTRALIA	1/250000	592
905	DORRIGO - COFFS HARBOUR	H56-10+11	30	0	31	0	151	30	153	30	NEW SOUTH WALES	1/250000	524
906	CAMDEN SOUND - PRINCE REGENT	D51-15+16	15	0	16	0	124	15	126	0	WESTERN AUSTRALIA	1/250000	318
907	PERCY ISLES - PORT CLINTON	F56-5+9	21	30	23	0	150	0	151	30	QUEENSLAND	1/250000	1492
908	BEDOUT ISLAND - PORT HEDLAND	REF	19	45	21	0	118	30	120	0	WESTERN AUSTRALIA	1/250000	162
909	BLUE MUD BAY - PORT LANGDON	D53-7+8	13	0	14	0	135	0	137	0	NORTHERN TERRITORY	1/250000	194
910	DAMPIER - BARROW ISLAND		SEE 912										488
911	PORT HEDLAND - BEDOUT ISLAND		SEE 908										1528
912	BARROW ISLAND - DAMPIER	-50-1+2	20	0	21	0	115	10	117	0	WESTERN AUSTRALIA	1/250000	140
913	ROPER RIVER - CAPE BEATRICE	D53-11+12	14	0	15	0	135	0	137	0	NORTHERN TERRITORY	1/250000	1644
914	LONDONDERRY - DRYSDALE	D52-5+9	13	40	15	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	1048
915	DRYSDALE - LONDONDERRY		SEE 914										528
916	JUNCTION BAY - MILINGIMBI	REF	11	40	13	0	133	30	135	0	NORTHERN TERRITORY	1/250000	895
917	BATHURST IS - MELVILLE IS	C52-15+16	11	0	12	0	130	0	131	30	NORTHERN TERRITORY	1/250000	152
918	PELLEW - ROBINSON RIVER	REF	15	20	17	0	136	30	138	0	NORTHERN TERRITORY	1/250000	1472
919	ADMIRALTY ISLANDS EAST	(A55-11)	1	45	3	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	18
920	ADMIRALTY ISLANDS WEST	(A55-10)	1	45	3	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	20
921	BOUGAINVILLE ISLAND NORTH	B56-8	5	0	6	0	154	30	156	0	PAPUA NEW GUINEA	1/250000	210
922	BOUGAINVILLE ISLAND SOUTH	B56-12	6	0	7	0	154	30	156	0	PAPUA NEW GUINEA	1/250000	212
923													
924													
925													
926	COCOS (KEELING) ISLANDS	D47	12	0	16	0	96	0	102	0	INDIAN OCEAN	1/1000000	398
927													
928													
929													
930													
931		F50-5	21	0	22	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	662
932		G49-16	27	0	28	0	112	30	114	0	WESTERN AUSTRALIA	1/250000	731
933		G50-13	27	0	28	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	732
934		H50-13	31	0	32	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	820
935		H50-14	31	0	32	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	821
936		F56-1	20	0	21	0	150	0	151	30	QUEENSLAND	1/250000	663
937		F56-2	20	0	21	0	151	30	153	0	QUEENSLAND	1/250000	664
938		F56-6	21	0	22	0	151	30	153	0	QUEENSLAND	1/250000	665
939		E55-3	16	0	17	0	147	0	148	30	QUEENSLAND	1/250000	617
940		E55-7	17	0	18	0	147	0	148	30	QUEENSLAND	1/250000	618
941		E55-11	18	0	19	0	147	0	148	30	QUEENSLAND	1/250000	619
942		D55-1	12	0	13	0	144	0	145	30	QUEENSLAND	1/250000	555
943		D55-5	13	0	14	0	144	0	145	30	QUEENSLAND	1/250000	556
944		C53-9	10	0	11	0	132	0	133	30	NORTHERN TERRITORY	1/250000	465
945		C53-13	11	0	12	0	132	0	133	30	NORTHERN TERRITORY	1/250000	466
946		C52-13	11	0	12	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	463
947		C52-14	11	0	12	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	464
948		D52-1	12	0	13	0	126	0	127	30	WESTERN AUSTRALIA	1/250000	552
949		D52-2	12	0	13	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	553
950		D52-6	13	0	14	0	127	30	129	0	WESTERN AUSTRALIA	1/250000	554

INDEX	NAME	NUMBER	LIMITS								REGION	SCALE	SEQUENCE
951		E51-1	16	0	17	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	614
952		E51-5	17	0	18	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	615
953		E51-9	18	0	19	0	120	0	121	30	WESTERN AUSTRALIA	1/250000	616
954		E50-13	19	0	20	0	114	0	115	30	WESTERN AUSTRALIA	1/250000	611
955		E50-14	19	0	20	0	115	30	117	0	WESTERN AUSTRALIA	1/250000	612
956		E50-15	19	0	20	0	117	0	118	30	WESTERN AUSTRALIA	1/250000	613
957		F49-4	20	0	21	0	112	30	114	0	WESTERN AUSTRALIA	1/250000	660
958		F49-8	21	0	22	0	112	30	114	0	WESTERN AUSTRALIA	1/250000	661
959		A54-7	1	0	2	0	141	0	142	30	PAPUA NEW GUINEA	1/250000	90
960		A54-12	2	0	3	0	142	30	144	0	PAPUA NEW GUINEA	1/250000	91
961		A55-9	2	0	3	0	144	0	145	30	PAPUA NEW GUINEA	1/250000	95
962		C55-1	8	0	9	0	144	0	145	30	PAPUA NEW GUINEA	1/250000	467
963		C55-9	10	0	11	0	144	0	145	30	PAPUA NEW GUINEA	1/250000	468
964		C55-10	10	0	11	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	469
965		A55-14	3	0	4	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	99
966		A56-9	2	0	3	0	150	0	151	30	PAPUA NEW GUINEA	1/250000	103
967		A55-6	1	0	2	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	92
968		A55-10	2	0	3	0	145	30	147	0	PAPUA NEW GUINEA	1/250000	96
969		A55-7	1	0	2	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	93
970		A55-11	2	0	3	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	97
971		A55-15	3	0	4	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	100
972		B55-3	4	0	5	0	147	0	148	30	PAPUA NEW GUINEA	1/250000	285
973		B55-16	7	0	8	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	286
974		A55-16	3	0	4	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	101
975		A55-8	1	0	2	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	94
976		A55-12	2	0	3	0	148	30	150	0	PAPUA NEW GUINEA	1/250000	98
977		A56-5	1	0	2	0	150	0	151	30	PAPUA NEW GUINEA	1/250000	102
978		A56-13	3	0	4	0	150	0	151	30	PAPUA NEW GUINEA	1/250000	104
979		B56-1	4	0	5	0	150	0	151	30	PAPUA NEW GUINEA	1/250000	287
980		B56-13	7	0	8	0	150	0	151	30	PAPUA NEW GUINEA	1/250000	294
981		A56-14	3	0	4	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	105
982		A56-15	3	0	4	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	106
983		B56-2	4	0	5	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	288
984		B56-3	4	0	5	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	289
985		B56-4	4	0	5	0	154	30	156	0	PAPUA NEW GUINEA	1/250000	290
986		B56-7	5	0	6	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	291
987		B56-10	6	0	7	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	292
988		B56-11	6	0	7	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	293
989		B56-14	7	0	8	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	295
990		B56-15	7	0	8	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	296
991		B56-16	7	0	8	0	154	30	156	0	SOLOMON ISLANDS	1/250000	297
992		C56-2	8	0	9	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	470
993		C56-3	8	0	9	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	471
994		C56-6	9	0	10	0	151	30	153	0	PAPUA NEW GUINEA	1/250000	472
995		C56-7	9	0	10	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	473
996		C56-11	10	0	11	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	474
997		C56-15	11	0	12	0	153	0	154	30	PAPUA NEW GUINEA	1/250000	475
998													
999													
1000													

CROSS REFERENCE TABLE

INDEX	NUMBER
4	G49-16/G50-13
36	J55-4/J56-1
465	A56-5/A55-12
304	F50-5/F49-8
671	H52-13/I52-1
757	H53-13/I53-1
786	H52-14/I52-2
787	G56-2+3+6+7
0	E51-13/F51-1
797	F49-12/F50-9
0	F56-14/G56-2
918	D53-16/E53-4
0	F49-16/F50-13
0	C53-15/D53-3
916	C53-14/D53-2
908	E50-16/F50-4
803	J54/K54/J55-13/K55-1
804	J55-15+16/K55-3+4
808	I53-16/J53-1/I54/J54
810	D53-7+8+11+12

444 ABAU
1 ABMINGA
2 ADAVALE
3 ADELAIDE
551 ADELAIDE*
858 ADMIRALTY IS
758 ADMIRALTY IS EAST
857 ADMIRALTY ISLANDS
919 ADMIRALTY ISLANDS EAST
920 ADMIRALTY ISLANDS WEST
761 ADMIRALTY IS WEST
445 AITAPE
4 AJANA
5 ALBANY
552 ALBANY*
6 ALBERGA
7 ALCOOTA
503 ALICE SPR
553 ALICE SPR*
601 ALICE SPRINGS
831 ALICE SPRINGS*
602 ALLIGATOR RIVER
8 ALROY
446 AMBUNTI
9 ANA BRANCH
10 ANDAMOOKA
11 ANGLEDOOL
12 ANKETELL
447 ARAWE
554 ARMIDALE
13 ARNHEM BAY
603 ARNHEM BAY - GOVE
448 AROA
14 ASHTON
15 ATHERTON
16 AUGATHELLA
17 AUGUSTA
18 AURUKUN
19 AUVERGNE
20 AVON DOWNS
759 AWORRA RIVER
21 AYERS ROCK
22 AYR
23 BAIRNSDALE
604 BALFOUR DOWNS
24 BALLADONIA
25 BALLARAT
26 BALRANALD
27 BARALABA
28 BARKER
29 BARLEE
30 BARNATO
31 BARROLKA
500 BARROW CK
605 BARROW CREEK
498 BARROW IS
606 BARROW ISLAND
912 BARROW ISLAND - DAMPIER
32 BARTON
33 BATHURST
607 BATHURST IS
608 BATHURST ISLAND
917 BATHURST IS - MELVILLE IS
609 BAUHINIA DOWNS
34 BEDOURIE
501 BEDOUT IS
610 BEDOUT ISLAND
908 BEDOUT ISLAND - PORT HEDLAND
35 BEETALOO
36 BEGA
37 BELELE
38 BENCUBBIN
39 BENDIGO
40 BENTLEY
41 BETOOTA
611 BILLAKALINA
42 BILLILUNA
43 BIRDSVILLE
44 BIRKSGATE
45 BIRRINDUDU
46 BLACKALL
612 BLOODS RANGE
762 BLUCHER RANGE
613 BLUE MUD BAY
909 BLUE MUD BAY - PORT LANGDON
555 BODALLA
449 BOGIA
450 BOIGU
614 BONNEY WELL
47 BOOLIGAL
48 BOORABBIN
921 BOUGAINVILLE ISLAND NORTH
922 BOUGAINVILLE ISLAND SOUTH
764 BOUGAINVILLE NORTH
767 BOUGAINVILLE SOUTH
49 BOULIA
50 BOURKE
556 BOURKE*
51 BOWEN
789 BOWEN - PROSERPINE
52 BREADEN
53 BREMER BAY
615 BRIGHTON DOWNS
54 BRISBANE
557 BRISBANE*
616 BROKEN HILL
832 BROKEN HILL*
55 BROOME
558 BROOME*
56 BROWNE
502 BROWSE IS
617 BROWSE ISLAND
618 BRUNETTE DOWNS
833 BRUNSWICK BAY
57 BUCHANAN
451 BUKA IS
765 BUKA ISLAND
58 BULLER
59 BULLOO
452 BUNA
60 BUNDABERG
61 BURKETOWN
62 BURNABBIE
63 BURNIE
64 BURRA
66 BUSSELTON
65 BYRO
67 CAIRNS
68 CALLABONNA
453 CALVADOS
619 CALVERT HILLS
620 CAMBRIDGE GULF
621 CAMDEN SOUND
906 CAMDEN SOUND - PRINCE REGENT
69 CAMOOWEAL
70 CANBERRA
559 CANBERRA*
71 CANTERBURY
72 CAPE ARID
622 CAPE BEATRICE
795 CAPE LEEUWIN
623 CAPE MELVILLE
760 CAPE NELSON
763 CAPE RAOULT
770 CAPE SAINT GEORGE
73 CAPE SCOTT
771 CAPE ST GEORGE
624 CAPE VAN DIEMEN
834 CAPE WESSEL
625 CAPE WEYMOUTH

74 CAPE YORK
76 CARGELLIGO
560 CARNARVON
626 CHARLEVILLE
835 CHARLEVILLE*
627 CHARLOTTE WATERS
77 CHARNLEY
628 CHARTERS TOWERS
78 CHILDARA
79 CHINCHILLA
580 CHOISEUL
80 CHOWILLA
629 CLARKE RIVER
81 CLERMONT
561 CLERMONT*
562 CLOATES
82 CLONCURRY
563 CLONCURRY*
75 COBAR
83 COBB
520 COBHAM
630 COBHAM LAKE
631 COBURG PENINSULAR
581 COCOS IS
859 COCOS ISLANDS
926 COCOS (KEELING) ISLANDS
84 COEN
632 COFFS HARBOUR
85 COLAC
86 COLLIE
88 COLLIER
89 CONNEMARA
633 COOBER PEDY
87 COOK
90 COOKTOWN
564 COOKTOWN*
91 COOMPANA
92 COOPER
836 COOPER CREEK
634 COOTAMUNDRA
93 COPLEY
94 CORDILLO
95 CORNISH
96 CORRIGIN
454 CP NELSON
455 CP RAOULT
756 CP ST GEORGE
97 CROSSLAND
98 CROYDON
99 CUE
100 CULVER
101 CUNDEELEE
102 CUNNAMULLA
104 CURDIMURKA
105 CURNAMONA
103 DALBY
106 DALHOUSIE
635 DALY WATERS
107 DAMPIER
910 DAMPIER - BARROW ISLAND
636 DAMPIER - YARRALOOLA
456 DARU
108 DARWIN
565 DARWIN*
499 DEAL IS
637 DEAL ISLAND
457 DEBOYNE
109 DELAMERE
110 DENILIQUIN
111 DERBY
112 DEVONPORT
638 DIRRANBANDI
639 DIXON RANGE
113 DOBBYN
115 DONGARA
640 DONORS HILL
116 DORRIGO
905 DORRIGO - COFFS HARBOUR
117 DRYSDALE
915 DRYSDALE - LONDONDERRY
118 DUARINGA
114 DUBBO
119 DUCHESS
566 DU COUEDIC
120 DUKETON
121 DUMBLEYUNG
122 DUMMER
641 DURHAM DOWNS
123 EBAGOOLA
124 EDDYSTONE
125 EDEL
127 EDJUDINA
126 EDMUND
128 EINASLEIGH
129 ELKEDRA
130 ELLISTON
131 EMERALD
132 ENNGONIA
133 EROMANGA
134 ESPERANCE
567 ESPERANCE*
904 ESPERANCE - MONDRAIN ISLAND
135 EUABALONG
136 EUCLA
786 EUCLA - NOONAERA
137 EULO
138 EVERARD
568 EYRE
769 FERGUSSON IS
777 FERGUSSON ISLAND
642 FERGUSSON RIVER
139 FINKE
644 FLINDERS IS
643 FLINDERS ISLAND
804 (FLINDERS ISLAND)
458 FLY RIVER
582 FLY RIVER*
140 FOG BAY
792 FOG BAY - DARWIN
141 FORBES
142 FORREST
144 FOWLER
757 FOWLER - NUYTS
518 FRASER IS
787 FRASER ISLAND
143 FREW RIVER
145 FROME
146 GAIRDNER
147 GALBRAITH
148 GALILEE
459 GASMATA
149 GASON
776 GAZELLE PENINSULA
150 GEORGETOWN
151 GERALDTON
152 GILBERTON
153 GILES
154 GILGANDRA
155 GLENBURGH
156 GLENGARRY
645 GLENORMISTON
646 GOONDIWINDI
647 GORDON DOWNS
157 GOULBURN
158 GOVE
159 GRAFTON
788 GRAFTON - MACLEAN
648 GREEN SWAMP WELL
810 GROOTE EYLANDT
519 GROOTE IS

800 GROOTE ISLAND
860 GUADALCANAL
160 GUNANYA
161 GYMPIE
569 HALLS CK
837 HALLS CREEK
162 HALE RIVER
838 HAMERSLEY RANGE
163 HAMILTON
570 HAMILTON*
164 HANN RIVER
649 HARDING SPRING
165 HASTINGS
166 HAY
167 HAY RIVER
168 HELENA
650 HELEN SPRINGS
169 HENBURY
170 HERBERT
651 HERMANNSBURG
504 HERON IS
652 HERON ISLAND
654 HIGHLAND ROCKS
171 HILL RIVER
172 HOBART
517 (HOBART)
655 HODGSON DOWNS
583 HOLLANDIA
173 HOLROYD
174 HOMEBOIN
175 HORSHAM
801 HOUTMAN ABROLHOS
839 HOUTMAN ABROLHOS*
176 HUCKITTA
177 HUGHENDEN
178 HUON
460 (HUON)
802 HUON (TASMANIA)
179 HYDEN
505 ILLOGWA CK
656 ILLOGWA CREEK
180 INGHAM
181 INNAMINCKA
182 INNISFAIL
183 INVERELL
184 IPSWICH
791 IPSWICH - BRISBANE
653 IRWIN INLET
185 IVANHOE
186 JACKSON
657 JARDINE RIVER
187 JERICHO
188 JERILDERIE
513 JOANNA SPR
658 JOANNA SPRING
190 JUBILEE
506 JULIA CK
659 JULIA CREEK
660 JUNCTION BAY
916 JUNCTION BAY - MILINGIMBI
189 JUNDAH
191 KALGOORLIE
840 KALGOORLIE*
461 KALO
809 KANGAROO IS
808 KANGAROO ISLAND
462 KARAMUI
463 KAR KAR IS
464 KARKAR IS
773 KAR KAR ISLAND
774 KARKAR ISLAND
192 KATHERINE
465 KAVIENG
584 KAVIENG*
661 KELLERBERRIN
662 KENNEDY RANGE
516 KERSAINT
466 KIKORI
193 KIMBA
507 KING IS
663 KING ISLAND
803 (KING ISLAND)
194 KINGOONYA
195 KINGSCOTE
196 KINGSTON
197 KIRKALOCKA
467 KIWAI
664 KOPPERAMANNA
198 KULGERA
199 KURNALPI
585 LAE
200 LA GRANGE
665 LAKE AMADEUS
201 LAKE EYRE
666 LAKE JOHNSTON
768 LAKE KUTUBU
667 LAKE MACKAY
841 LAKE MACKAY*
772 LAKE MURRAY
508 L AMADEUS
668 LANDER RIVER
202 LANSDOWNE
203 LARRIMAH
204 LAUNCESTON
805 (LAUNCESTON)
205 LAVERTON
206 LAWN HILL
669 LENNARD RIVER
207 LENNIS
208 LEONORA
509 L EYRE
209 LIMBUNYA
210 LINCOLN
211 LINDSAY
212 LISSADELL
510 L JOHNSTON
468 L KUTUBU
511 L MACKAY
469 L MURRAY
670 LONDONDERRY
914 LONDONDERRY - DRYSDALE
214 LONGREACH
215 LOONGANA
863 LORD HOWE IS
861 LORD HOWE ISLAND
213 LOUTH
216 LUCAS
470 MABUA
217 MACDONALD
218 MACHATTIE
220 MACKAY
221 MACKUNDA
222 MACLEAN
471 MADANG
223 MADLEY
224 MADURA
671 MADURA - BURNABBIE
472 MAER
225 MAITLAND
226 MALCOLM
672 MALCOLM - CAPE ARID
227 MALLACOOTA
228 MANARA
229 MANDORA
230 MANEROO
231 MANILLA
219 MANN
232 MANUKA
233 MARBLE BAR
473 MARKHAM

234 MARREE
673 MARYBOROUGH
235 MASON
477 MASSAU IS
775 MASSAU ISLAND
236 MAURICE
474 MAY RIVER
275 MC DILLS
274 MCDILLS
277 MC KINLAY
276 MCKINLAY
707 MC LARTY HILLS
708 MCLARTY HILLS
674 MEDUSA BANKS
675 MEEKATHARRA
842 MEEKATHARRA*
237 MELBOURNE
571 MELBOURNE*
677 MELVILLE IS
844 MELVILLE IS*
676 MELVILLE ISLAND
843 MELVILLE ISLAND*
238 MENINDEE
239 MENZIES
240 MILDURA
241 MILINGIMBI
242 MILLUNGERA
243 MILPARINKA
244 MINIGWAL
245 MINILYA
246 MITCHELL
845 MITCHELL RIVER
679 MONDRAIN IS
678 MONDRAIN ISLAND
680 MONTAGUE SOUND
247 MONTO
248 MOORA
249 MOREE
250 MORNINGTON
251 MORRIS
252 MOSSMAN
790 MOSSMAN - CAIRNS
681 MOUNT ANDERSON
683 MOUNT BANNERMAN
685 MOUNT BARKER
902 MOUNT BARKER - ALBANY
686 MOUNT BRUCE
687 MOUNT COOLON
688 MOUNT DOREEN
689 MOUNT DRUMMOND
691 MOUNT EGERTON
692 MOUNT ELIZABETH
694 MOUNT EVELYN
260 MOUNT ISA
695 MOUNT LIEBIG
696 MOUNT MARUMBA
697 MOUNT PEAKE
698 MOUNT PHILLIPS
700 MOUNT RAMSAY
701 MOUNT RENNIE
702 MOUNT SOLITAIRE
267 MOUNT THEO
704 MOUNT WHELAN
705 MOUNT YOUNG
682 MT ANDERSON
684 MT BANNERMAN
253 MT BARKER
255 MT BRUCE
256 MT COOLON
257 MT DOREEN
690 MT DRUMMOND
258 MT EGERTON
693 MT ELIZABETH
259 MT EVELYN
254 MT ISA
261 MT LIEBIG
262 MT MARUMBA
263 MT PEAKE
699 MT PHILLIPS
264 MT RAMSAY
265 MT RENNIE
703 MT SOLITAIRE
266 MT THEO
268 MT WHELAN
269 MT YOUNG
270 MUNDUBBERA
271 MUNRO
272 MURGOO
706 MURLOOCOPPIE
273 MUTTABURRA
278 NABBERU
478 NAMATANAI
279 NAPPERBY
280 NARACOORTE
281 NARETHA
282 NARRABRI
283 NARRANDERA
284 NARROMINE
285 NEALE
286 NEWCASTLE
709 NEWCASTLE WATERS
846 NEWCASTLE WATERS*
287 NEWDEGATE
288 NEWMAN
289 NINGALOO
797 NINGALOO - YANREY
290 NINGHAN
475 NINIGO
291 NOOLYEANA
292 NOONAERA
293 NOONKANBAH
294 NOORINA
587 NORFOLK IS
862 NORFOLK ISLAND
295 NORMANTON
572 NORMANTON*
296 NORSEMAN
297 NULLAGINE
298 NULLARBOR
847 NULLARBOR PLAIN
299 NUYTS
300 NYMAGEE
301 NYNGAN
848 OAKOVER RIVER
302 OATLANDS
303 OLARY
304 ONSLOW
305 OODNADATTA
849 OODNADATTA*
306 OOLDEA
307 ORFORD BAY
308 ORRORDO
309 OUYEN
710 PANDIE PANDIE
310 PARACHILNA
711 PATERSON RANGE
312 PEAK HILL
313 PELLEW
918 PELLEW - ROBINSON RIVER
314 PEMBERTON
901 PEMBERTON - IRWIN INLET
315 PENDER
316 PENOLA
317 PERCIVAL
312 PERCY IS
712 PERCY ISLANDS
784 PERCY ISLES
907 PERCY ISLES - PORT CLINTON
318 PERENJORI
311 PERTH

573 PERTH*
714 PETERMANN RANGES
850 PETERMANN RANGES*
319 PINE CREEK
320 PINJARRA
321 PINNAROO
322 PLUMRIDGE
479 POMIO
323 POOLOWANNA
324 POONCARIE
713 PORT AUGUSTA
851 PORT AUGUSTA*
715 PORT CLINTON
328 PORT DAVEY
807 (PORT DAVEY)
716 PORT HEDLAND
911 PORT HEDLAND - BEDOUT ISLAND
331 PORT KEATS
334 PORTLAND
717 PORT LANGDON
778 PORT MORESBY
864 PORT MORESBY*
718 PRINCE REGENT
336 PROSERPINE
325 PT AUGUSTA
326 PT CLINTON
329 PT DAVEY
330 PT HEDLAND
333 PT KEATS
335 PT LANGDON
480 PT MORESBY
337 PYRAMID
720 QUEENSCLIFF
338 QUEENSTOWN
80((QUEENSTOWN)
339 QUILPIE
340 QUOBBA
586 RABAUL
481 RAGGI
476 RAMU
332 RANKEN
327 RASON
721 RAVENSTHORPE
341 RAWLINSON
342 RED RIVER
343 RENMARK
344 RICHMOND
345 ROBERT
346 ROBERTSON
719 ROBINSON RANGE
722 ROBINSON RIVER
723 ROCKHAMPTON
852 ROCKHAMPTON*
347 RODINGA
349 ROEBOURNE
796 ROEBOURNE - PYRAMID
348 ROMA
724 ROPER RIVER
853 ROPER RIVER*
913 ROPER RIVER - CAPE BEATRICE
482 ROSSEL
854 ROWLEY SHOALS
350 ROY HILL
351 RUDALL
352 RUNTON
725 RUTLAND PLAINS
353 RYAN
483 SABLE
485 SAG SAG
484 SAGSAG
355 SAHARA
728 SAINT ARNAUD
729 SAINT GEORGE
737 SAINT LAWRENCE
486 SALAMAUA
356 SALE
487 SAMARAI
488 SAMO
726 SANDOVER RIVER
354 SANDSTONE
357 SANDY CAPE
358 SCOTT
359 SEEMORE
514 SEPIK
360 SHARK BAY
727 SIMPSON DESERT NORTH
732 SIMPSON DESERT SOUTH
361 SINGLETON
362 SIR SAMUEL
363 SMITHTON
733 SOUTHERN CROSS
736 SOUTH LAKE WOODS
364 SPRINGSURE
366 SPRINGVALE
365 STANLEY
367 STANSMORE
368 ST ARNAUD
369 ST GEORGE
735 ST LAWRENCE
574 STOKES
730 STREAKY BAY
371 STRZELECKI
372 SURAT
370 SWAN HILL
373 SYDNEY
575 SYDNEY*
374 TABLETOP
515 TALASEA
375 TALBOT
731 TALLANGATTA
377 TALLARINGA
378 TAMBO
376 TAMWORTH
379 TANAMI
734 TANAMI EAST
380 TANGORIN
738 TANUMBIRINI
381 TARCOOLA
576 TARCOOLA*
382 TAROOM
577 TASMANIA
489 TAUU
383 TENNANT CK
741 TENNANT CREEK
742 THARGOMINDAH
743 THE GRANITES
384 THROSSELL
739 THURSDAY IS
744 THURSDAY ISLAND
385 TICKALARA
386 TOBERMORY
387 TOOMPINE
388 TORRENS
389 TORRES ST
579 TORRES ST*
745 TORRES STRAIT
855 TORRES STRAIT*
390 TOWNSVILLE
856 TOWNSVILLE*
391 TRAINOR
779 TROBRIAND IS
782 TROBRIAND ISLANDS
392 TRUANT IS
746 TRUANT ISLAND
490 TUFT
393 TUREE CK
747 TUREE CREEK
740 TWEED HEADS
394 ULLADULLA
395 URAL

396 URANDANGI
397 URAPUNGA
399 URISINO
491 VANIMO
398 VERNON
750 VICTORIA RIVER DOWNS
492 VITU IS
780 VITU ISLANDS
493 WABAG
400 WAGGA
751 WAGGA WAGGA
401 WAIGEN
404 WALGETT
405 WALLHALLOW
402 WALSH
406 WANGARATTA
403 WANNA
407 WARBURTON
408 WARRAGUL
409 WARRI
410 WARRINA
411 WARWICK
412 WATERLOO
494 WAU
413 WAVE HILL
415 WEBB
416 WEIPA
794 WELLESLEY IS
793 WELLESLEY ISLANDS
417 WELLS
418 WESSEL IS
752 WESSEL ISLANDS
903 WESSEL ISLANDS - TRUANT ISLAND
753 WESTMORELAND
414 WESTWOOD
495 WEWAK
588 WEWAK*
754 WHITE CLIFFS
419 WHYALLA
420 WIDE BAY
755 WIDGIEMOOLTHA
421 WILCANNIA
422 WILSON
423 WILUNA
578 WILUNA*
424 WINDORAH
748 WINNECKE CK
749 WINNECKE CREEK
756 WINNING POOL
425 WINTINNA
426 WINTON
427 WOLLONGONG
781 WOODLARK IS
866 WOODLARK IS*
785 WOODLARK ISLAND
865 WOODLARK ISLAND*
428 WOODROFFE
429 WOORAMEL
496 WUVULU IS
783 WUVULU ISLAND
430 WYANDRA
431 WYLOO
432 WYOLA
433 YALGOO
434 YAMPI
435 YANREY
436 YANTABULLA
437 YARDEA
438 YARINGA
439 YARRALOOLA
440 YARRIE
441 YOUANMI
442 YOWALGA
497 YULE
443 ZANTHUS

588	A54
959	A54-7
783	A54-8
491	A54-11
960	A54-12
445	A54-15
495	A54-16
857	A55
475	A55-5
967	A55-6
969	A55-7
975	A55-8
961	A55-9
968	A55-10
970	A55-11
976	A55-12
514	A55-13
965	A55-14
971	A55-15
974	A55-16
584	A56
977	A56-5
966	A56-9
470	A56-10
978	A56-13
981	A56-14
982	A56-15
582	B54
474	B54-3
446	B54-4
762	B54-7
483	B54-8
481	B54-11
768	B54-12
772	B54-15
759	B54-16
585	B55
449	B55-1
773	B55-2
972	B55-3
780	B55-4
476	B55-5
471	B55-6
485	B55-7
763	B55-8
462	B55-9
473	B55-10
178	B55-11
447	B55-12
466	B55-13
494	B55-14
486	B55-15
973	B55-16
586	B56
979	B56-1
983	B56-2
984	B56-3
985	B56-4
515	B56-5
479	B56-6
986	B56-7
921	B56-8
459	B56-9
987	B56-10
988	B56-11
922	B56-12
980	B56-13
989	B56-14
990	B56-15
991	B56-16
580	B57
489	B57-1
871	C49
872	C50
873	C51
843	C52
946	C52-13
947	C52-14
608	C52-15
676	C52-16
834	C53
944	C53-9
945	C53-13
660	C53-14
752	C53-15
746	C53-16
855	C54
458	C54-3
467	C54-4
450	C54-7
456	C54-8
744	C54-11
74	C54-12
657	C54-15
307	C54-16
864	C55
962	C55-1
497	C55-2
452	C55-3
760	C55-4
472	C55-5
448	C55-6
778	C55-7
490	C55-8
963	C55-9
964	C55-10
461	C55-11
444	C55-12
865	C56
782	C56-1
992	C56-2
993	C56-3
777	C56-5
994	C56-6
995	C56-7
487	C56-9
457	C56-10
996	C56-11
483	C56-13
453	C56-14
997	C56-15
860	C57
926	D47
874	D49
875	D50
833	D51
617	D51-11
680	D51-12
621	D51-13
718	D51-14
565	D52
948	D52-1
949	D52-2
140	D52-3
108	D52-4
670	D52-5
950	D52-6
73	D52-7
319	D52-8
117	D52-9
674	D52-10
331	D52-11
642	D52-12
84	D52-13
620	D52-14
19	D52-15
109	D52-16
853	D53
102	D53-1
241	D53-2
13	D53-3
158	D53-4
694	D53-5
696	D53-6
613	D53-7
717	D53-8
192	D53-9
397	D53-10
724	D53-11
622	D53-12
203	D53-13
655	D53-14
705	D53-15
313	D53-16
845	D54
416	D54-3
625	D54-4
18	D54-7
84	D54-8
173	D54-11
123	D54-12
725	D54-15
164	D54-16
564	D55
942	D55-1
943	D55-5
623	D55-9
90	D55-13
876	D56
877	E49
854	E50
954	E50-13
955	E50-14
956	E50-15
610	E50-16
558	E51
951	E51-1
315	E51-2
434	E51-3
77	E51-4
952	E51-5
55	E51-6
111	E51-7
669	E51-8
953	E51-9
200	E51-10
681	E51-11
293	E51-12
229	E51-13
271	E51-14
707	E51-15
97	E51-16
837	E52
692	E52-1
212	E52-2
412	E52-3
750	E52-4
202	E52-5
639	E52-6
209	E52-7
413	E52-8
700	E52-9
647	E52-10
45	E52-11
749	E52-12
683	E52-13
42	E52-14
379	E52-15
734	E52-16
846	E53
635	E53-1
738	E53-2
609	E53-3
722	E53-4
709	E53-5
35	E53-6
405	E53-7
619	E53-8
736	E53-9
650	E53-10
618	E53-11
689	E53-12
648	E53-13
741	E53-14
8	E53-15
332	E53-16
572	E54
250	E54-1
624	E54-2
147	E54-3
402	E54-4
753	E54-5
61	E54-6
295	E54-7
342	E54-8
206	E54-9
640	E54-10
98	E54-11
150	E54-12
69	E54-13
113	E54-14
242	E54-15
152	E54-16
856	E55
252	E55-1
67	E55-2
939	E55-3
15	E55-5
182	E55-6
940	E55-7
128	E55-9
180	E55-10
941	E55-11
629	E55-13
390	E55-14
22	E55-15
878	E56
898	E57
562	F49
957	F49-4
958	F49-8
289	F49-12
245	F49-16
838	F50
606	F50-1
107	F50-2
349	F50-3
716	F50-4
931	F50-5
439	F50-6
337	F50-7
233	F50-8
435	F50-9
431	F50-10
686	F50-11
350	F50-12
756	F50-13
126	F50-14
747	F50-15
288	F50-16
848	F51
440	F51-1
12	F51-2
658	F51-3
122	F51-4
297	F51-5
711	F51-6
355	F51-7
317	F51-8
604	F51-9

351	F51-10
374	F51-11
395	F51-12
346	F51-13
160	F51-14
352	F51-15
251	F51-16
841	F52
95	F52-1
216	F52-2
743	F52-3
702	F52-4
168	F52-5
367	F52-6
654	F52-7
267	F52-8
422	F52-9
415	F52-10
667	F52-11
688	F52-12
353	F52-13
217	F52-14
701	F52-15
695	F52-16
831	F53
668	F53-1
614	F53-2
143	F53-3
20	F53-4
697	F53-5
605	F53-6
129	F53-7
726	F53-8
679	F53-9
7	F53-10
176	F53-11
386	F53-12
651	F53-13
601	F53-14
656	F53-15
167	F53-16
563	F54
260	F54-1
82	F54-2
659	F54-3
344	F54-4
396	F54-5
119	F54-6
277	F54-7
232	F54-8
645	F54-9
49	F54-10
221	F54-11
426	F54-12
704	F54-13
366	F54-14
615	F54-15
230	F54-16
561	F55
177	F55-1
628	F55-2
51	F55-3
336	F55-4
380	F55-5
57	F55-6
687	F55-7
220	F55-8
273	F55-9
148	F55-10
81	F55-11
737	F55-12
214	F55-13
187	F55-14
131	F55-15
118	F55-16
852	F56
936	F56-1
937	F56-2
784	F56-5
938	F56-6
715	F56-9
830	F56-10
723	F56-13
652	F56-14
897	F57
890	G48
560	G49
340	G49-4
360	G49-8
125	G49-12
932	G49-16
842	G50
662	G50-1
698	G50-2
691	G50-3
88	G50-4
429	G50-5
155	G50-6
719	G50-7
312	G50-8
438	G50-9
65	G50-10
37	G50-11
156	G50-12
933	G50-13
272	G50-14
99	G50-15
554	G50-16
578	G51
58	G51-1
391	G51-2
223	G51-3
409	G51-4
278	G51-5
365	G51-6
170	G51-7
56	G51-8
423	G51-9
196	G51-10
345	G51-11
442	G51-12
362	G51-13
120	G51-14
384	G51-15
414	G51-16
850	G52
83	G52-1
341	G52-2
612	G52-3
665	G52-4
40	G52-5
358	G52-6
714	G52-7
21	G52-8
375	G52-9
92	G52-10
219	G52-11
428	G52-12
207	G52-13
401	G52-14
44	G52-15
211	G52-16
849	G53
169	G53-1
347	G53-2
162	G53-3
727	G53-4
198	G53-5
139	G53-6
275	G53-7
732	G53-8
6	G53-9
1	G53-10
106	G53-11
323	G53-12
138	G53-13
425	G53-14
305	G53-15
291	G53-16
836	G54
34	G54-1
218	G54-2
89	G54-3
189	G54-4
43	G54-5
41	G54-6
71	G54-7
424	G54-8
710	G54-9
94	G54-10
31	G54-11
133	G54-12
149	G54-13
181	G54-14
641	G54-15
742	G54-16
835	G55
46	G55-1
378	G55-2
364	G55-3
27	G55-4
2	G55-5
16	G55-6
124	G55-7
382	G55-8
339	G55-9
626	G55-10
246	G55-11
48	G55-12
387	G55-13
430	G55-14
174	G55-15
372	G55-16
557	G56
247	G56-1
60	G56-2
357	G56-3
270	G56-5
673	G56-6
420	G56-7
79	G56-9
828	G56-10
829	G56-11
103	G56-13
184	G56-14
54	G56-15
896	G57
839	H49
801	H49-4
573	H50
151	H50-1
433	H50-2
197	H50-3
441	H50-4
115	H50-5
318	H50-6
290	H50-7
29	H50-8
171	H50-9
248	H50-10
38	H50-11
186	H50-12
934	H50-13
935	H50-14
661	H50-15
733	H50-16
840	H51
208	H51-1
205	H51-2
327	H51-3
285	H51-4
239	H51-5
127	H51-6
244	H51-7
322	H51-8
91	H51-9
99	H51-10
101	H51-11
159	H51-12
88	H51-13
75	H51-14
449	H51-15
281	H51-16
847	H52
398	H52-1
403	H52-2
294	H52-3
417	H52-4
190	H52-5
235	H52-6
432	H52-7
236	H52-8
215	H52-9
142	H52-10
87	H52-11
306	H52-12
224	H52-13
136	H52-14
91	H52-15
298	H52-16
576	H53
153	H53-1
706	H53-2
410	H53-3
201	H53-4
377	H53-5
633	H53-6
611	H53-7
104	H53-8
32	H53-9
381	H53-10
194	H53-11
10	H53-12
144	H53-13
78	H53-14
146	H53-15
388	H53-16
832	H54
664	H54-1
371	H54-2
385	H54-3
59	H54-4
234	H54-5
68	H54-6
243	H54-7
399	H54-8
93	H54-9
145	H54-10
630	H54-11
754	H54-12
310	H54-13
105	H54-14
616	H54-15
421	H54-16
556	H55
137	H55-1
102	H55-2
638	H55-3
729	H55-4
436	H55-5

132	H55-6
11	H55-7
249	H55-8
213	H55-9
50	H55-10
404	H55-11
282	H55-12
30	H55-13
75	H55-14
301	H55-15
154	H55-16
554	H56
646	H56-1
411	H56-2
740	H56-3
183	H56-5
159	H56-6
222	H56-7
231	H56-9
116	H56-10
632	H56-11
376	H56-13
826	H56-14
827	H56-15
861	H57
862	H58
879	I49
552	I50
526	I50-1
320	I50-2
96	I50-3
179	I50-4
66	I50-5
86	I50-6
121	I50-7
287	I50-8
17	I50-9
314	I50-10
685	I50-11
53	I50-12
527	I50-13
653	I50-14
5	I50-15
528	I50-16
567	I51
666	I51-1
296	I51-2
24	I51-3
100	I51-4
721	I51-5
134	I51-6
226	I51-7
529	I51-8
530	I51-9
678	I51-10
72	I51-11
531	I51-12
568	I52
62	I52-1
292	I52-2
532	I52-3
533	I52-4
851	I53
299	I53-1
730	I53-2
437	I53-3
713	I53-4
534	I53-5
130	I53-6
193	I53-7
419	I53-8
535	I53-9
537	I53-10
539	I53-11
225	I53-12
536	I53-13
538	I53-14
540	I53-15
195	I53-16
551	I54
308	I54-1
303	I54-2
238	I54-3
228	I54-4
64	I54-5
80	I54-6
9	I54-7
324	I54-8
3	I54-9
343	I54-10
240	I54-11
26	I54-12
28	I54-13
321	I54-14
309	I54-15
370	I54-16
559	I55
185	I55-1
300	I55-2
284	I55-3
114	I55-4
47	I55-5
76	I55-6
141	I55-7
33	I55-8
166	I55-9
283	I55-10
634	I55-11
157	I55-12
110	I55-13
188	I55-14
751	I55-15
70	I55-16
575	I56
361	I56-1
286	I56-2
822	I56-5
823	I56-6
427	I56-9
824	I56-10
394	I56-13
825	I56-14
895	I57
880	J49
881	J50
882	J51
883	J52
566	J53
516	J53-4
570	J54
541	J54-1
542	J54-2
175	J54-3
728	J54-4
543	J54-5
544	J54-6
163	J54-7
25	J54-8
545	J54-9
546	J54-10
334	J54-11
85	J54-12
547	J54-13
548	J54-14
549	J54-15
550	J54-16
571	J55
39	J55-1
406	J55-2
731	J55-3
820	J55-4
237	J55-5
407	J55-6
23	J55-7
227	J55-8
720	J55-9
591	J55-10
356	J55-11
596	J55-12
592	J55-13
593	J55-14
594	J55-15
595	J55-16
555	J56
821	J56-1
894	J57
884	K49
885	K50
886	K51
887	K52
888	K53
574	K54
577	K55
597	K55-1
598	K55-2
599	K55-3
600	K55-4
811	K55-5
812	K55-6
813	K55-7
814	K55-8
815	K55-9
816	K55-10
517	K55-11
817	K55-12
818	K55-13
807	K55-14
460	K55-15
819	K55-16
889	K56
891	L54
892	L55
893	L56

INDEX

LOC	CONTENT	LOC	CONTENT	LOC	CONTENT	LOC	CONTENT	LOC	CONTENT
B01	ABMINGA								
C01	BOWEN								
D01	CUNDEELEE								
E01	GERALDTON								
F01	LAKE EYRE								
G01	MORRIS								
H01	NYNGAN								
I01	RUDALL								
J01	WAIGEN								
K01	BUKA IS								
L01	BEDOUT IS								
M01	ADELAIDE*								
N01	ALICE SPRI								
O01	HERMANNSBU								
P01	MOUNT RENN								
B02	WAGGA WAGG								
C02	HOUTMAN AB								
D02	PORT AUGUS								
E02	PEMBERTON								
F02									
G02	A56-5/A55								
H02	ABAU								
I02	CAPE YORK								
J02	GROOTE ISL								
K02	MARREE								
L02	PERTH*								
M02	URANDANGI								
N02	A54								
O02	F51-10								
P02	H55-6								