AD-762 070

SYNTHETIC BATHYMETRIC PROFILING SYSTFM (SYNBAPS)

Roger J. Van Wyckhouse

Naval Oceanographic Office
Washington, D.C.

May 1973

DISTRIBUTED BY:

NTIS
National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

TR-233

AD 762070

TECHNICAL REPORT

SYNTHETIC BATHYMETRIC PROFILING SYSTEM (SYNBAPS)

ROGER J. VANWYCKHOUSE

MAY 1973

Approved for public release; distribution unlimited.

NAVAL OCEANOGRAPHIC OFFICE
WASHINGTON, D.C. 20373

A1

ABSTRACT

The Synthetic Bathymetric Profiling System (SYNBAPS) consists of 10 FORTRAN IV computer programs, a random-access storage device, and an initial bathymetric data base of over 3 million data points. SYNBAPS is designed for rapid generation of random omnidirectional bathymetric profiles in digital form along great-circle paths. The initial data base will cover most of the Northern Hemisphere and will be extended to other regions as suitable bathymetric contour charts become available.

Data derived from the bathymetric contour charts are structured into a gridded data surface by the application of a cubic spline algorithm. The gridded data are stored on a random-access storage device by 5-degree-square areas. An accessing program, initiated by a user's request, extracts the 5-degree-square blocks of data for processing. The interpolation of the final profile is accomplished by orienting a cubic spline algorithm along a great-circle path and interpolating the depth values from the 5-degree squares falling on the path. A status program checks the content and condition of the random-access storage device.

SYNBAPS will provide bathymetric profiles at about one-fifth the cost and one-hundredth the time of present semiautomated methods.

Ocean Science Department
Science and Engineering Center

A

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Ocean Science Department
U.S. Naval Oceanographic Office
Washington, D.C. 20373

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

N/A

3. REPORT TITLE

Synthetic Bathymetric Profiling System (SYNBAPS)

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Report

5. AUTHOR(S) (First name, middle initial, last name)

Roger J. Van Wyckhouse

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 1973	~~138~~ 147	15

8a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

TR-233

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

NONE

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

U.S. Naval Oceanographic Office
Washington, D.C. and Office of/20373
Naval Research, Washington, D.C.
20375

13. ABSTRACT

The Synthetic Bathymetric Profiling System (SYNBAPS) consists of 10 FORTRAN IV computer programs, a random-access storage device, and an initial bathymetric data base of over 3 million data points. SYNBAPS is designed for rapid generation of random omnidirectional bathymetric profiles in digital form along great-circle paths. The initial data base will cover most of the Northern Hemisphere and will be extended to other regions as suitable bathymetric contour charts become available.

Data derived from the bathymetric contour charts are structured into a gridded data surface by the application of a cubic spline algorithm. The gridded data are stored in a random-access storage device by 5-degree-square areas. An accessing program, initiated by a user's request, extracts the 5-degree-square blocks of data for processing. The interpolation of the final profile is accomplished by orienting a cubic spline algorithm along a great-circle path and interpolating the depth values from the 5-degree squares falling on the path. A status program checks the content and condition of the random-access storage device.

SYNBAPS will provide bathymetric profiles at about one-fifth the cost and one-hundredth the time of present semiautomated methods.

DD FORM 1473 1 NOV 65 (PAGE 1)

S/N 0101-807-6801

UNCLASSIFIED
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Bathymetry						
Bathymetric profiling						
Bathymetric data bank						
Acoustic propagation modeling						
Northern Hemisphere, bathymetry						
Digital computer software						
Great circle						
Random-access storage device						

DD FORM 1 NOV 65 1473 (BACK)
(PAGE 2)

FOREWORD

This report describes a computer system and programs that will establish a world-wide bathymetric data bank and generate computer-drawn bathymetric profiles. The research was performed by the Naval Oceanographic Office in support of the Office of Naval Research, Long Range Acoustic Propagation Project, which provided funding. It is part of a major bathymetric charting project covering the North Atlantic and North Pacific Oceans. Bathymetric data, usually in the form of profiles, are essential elements in the development of acoustic propagation models and predictions, which are required for naval planning, systems development, and operations. The computerized bathymetric profiling system and specialized data bank described here will generate computer-drawn bathymetric profiles at a small fraction of the time and cost of manually produced profiles. This specialized data bank will be operational when approximately 600 5-degree-square areas have been structured on a random-access storage device. Presently, the contour data required for the structuring procedure are being digitized under ONR-LRAPP contract No. N00014-72-C-0466.

P. V. Purkrabek

P. V. PURKRABEK
Captain, U.S. Navy
Commander
U.S. Naval Oceanographic Office

CONTENTS

ILLUSTRATIONS

TABLES

APPENDIX A

FIGURES - APPENDIX A

APPENDIX B

APPENDIX C

INTRODUCTION

The need for a computerized bathymetric data bank and techniques for rapidly manipulating large quantities of data became evident as demand upon the Naval Oceanographic Office for bathymetric profiles increased and became more urgent. It became increasingly difficult to satisfy these demands through manual compilation of depth soundings, contouring, and profile constructions. A massive recompilation and reanalysis of bathymetric data, systematic revision of all bathymetric charts in the North Atlantic and North Pacific Oceans, including extension of chart coverage to the equator, was underway. At the same time the impracticality of using the existing data bank of bathymetric soundings for machine generation of profiles became apparent. The need for a specialized bathymetric data bank to support acoustic - oceanographic modeling gave rise to development of a synthetic bathymetric profiling project using the new bathymetric contour charts as the data base. The project developed procedures for digitizing the contour charts, and computer programs and subroutines for data storage and retrieval and for profile generation. Mr. Thomas M. Davis, Naval Oceanographic Office, provided special assistance in developing programs SPLINT (SYNGRID), BURNS (SYNCON2R), BATHY (subroutine BATHY) and DAWHAT (SYNCHEX) and contributed to the basic philosophy regarding SYNBAPS. Mr. J.D. Brown, Naval Oceanographic Office, assisted in the software development and digitization of the test data.

Funds for this project were provided by the Office of Naval Research through the Long Range Acoustic Propagation Project.

One of the basic inputs to most Navy long-range, acoustic propagation models are bathymetric profiles in digital form. These profiles usually are plotted along a great-circle path (glossary) as a function of range versus depth. Two methods of generating such profiles generally have been employed. In the first, a ship sails a predetermined great-circle path collecting continuous bathymetry using a precision depth recorder (PDR). If the course is accurately adhered to, the PDR record can be merged with the navigational record to obtain the bathymetric profile. If the navigational record is poor, the track of the ship will have to be adjusted and normalized to obtain a satisfactory bathymetric profile. A profile thus produced is accurate and retains most of the high frequency information but is costly in ship time, hard to schedule, and usually results in only a single profile.

A second means of obtaining a bathymetric profile is to plot a great-circle path on a bathymetric contour chart, or series of charts, and digitize the range and depth at the intersection of the path with each bathymetric contour. When a large number of great-circle profiles, each several thousand miles long, involving dozens of bathymetric charts, are constructed, the labor costs are considerable. Profiles produced manually from charts tend to be schematic, blocky, and subject to human error. Most importantly, both of these methods are slow and cannot be achieved in real time.

Although various phases of both methods have been automated, within the Navy and elsewhere, no totally satisfactory solution has been achieved to the present time. The system proposed in this report is one approach to solving the above problems.

The Synthetic Bathymetric Profiling System (SYNBAPS) is a combination of digital computer software (programs) and a random-access storage file (presently a CDC 813 permanent disk) of gridded bathymetric data, employed to generate random, great-circle, bathymetric profiles suitable for acoustic propagation modeling. SYNBAPS is completely automatic, requiring only the input, via a control card, of the latitude and longitude of the beginning and end points to extract the desired profile. The profile also can be generated given the latitude and longitude of the beginning point, the bearing, and the maximum range. The generated profile is available in two forms. The first is a computer-drawn profile where range in whole nautical miles is plotted against depth, in either meters or fathoms; the second is a punched card deck of the same data. The profile outputs in card image are available on magnetic tape where large quantities of data are involved.

A bathymetric profile along a great-circle path of about 8,000 nm can be generated in approximately 3 minutes of computer time on a second generation computer and can be plotted in about 3 minutes on an incremental plotter. A cost comparison shows that, by present semiautomatic methods, a set of 19 short profiles totaling 9,000 nm required 144 man-hours at a cost of $900. The same profiles could be produced by SYNBAPS in 1.4 man-hours at a cost of $50, for a savings of 18:1 in dollars and 100:1 in time.

OUTLINE OF SYSTEM OPERATION

The SYNBAPS software can be broken down into three distinct program functions associated with structuring, accessing, and status (fig. 1).

The structuring programs create a gridded bathymetric data base and structure it on a random-access device in a precise form. The smallest cell of the data base is a 5-minute-square grid where the north-south side is in meridional minutes or parts

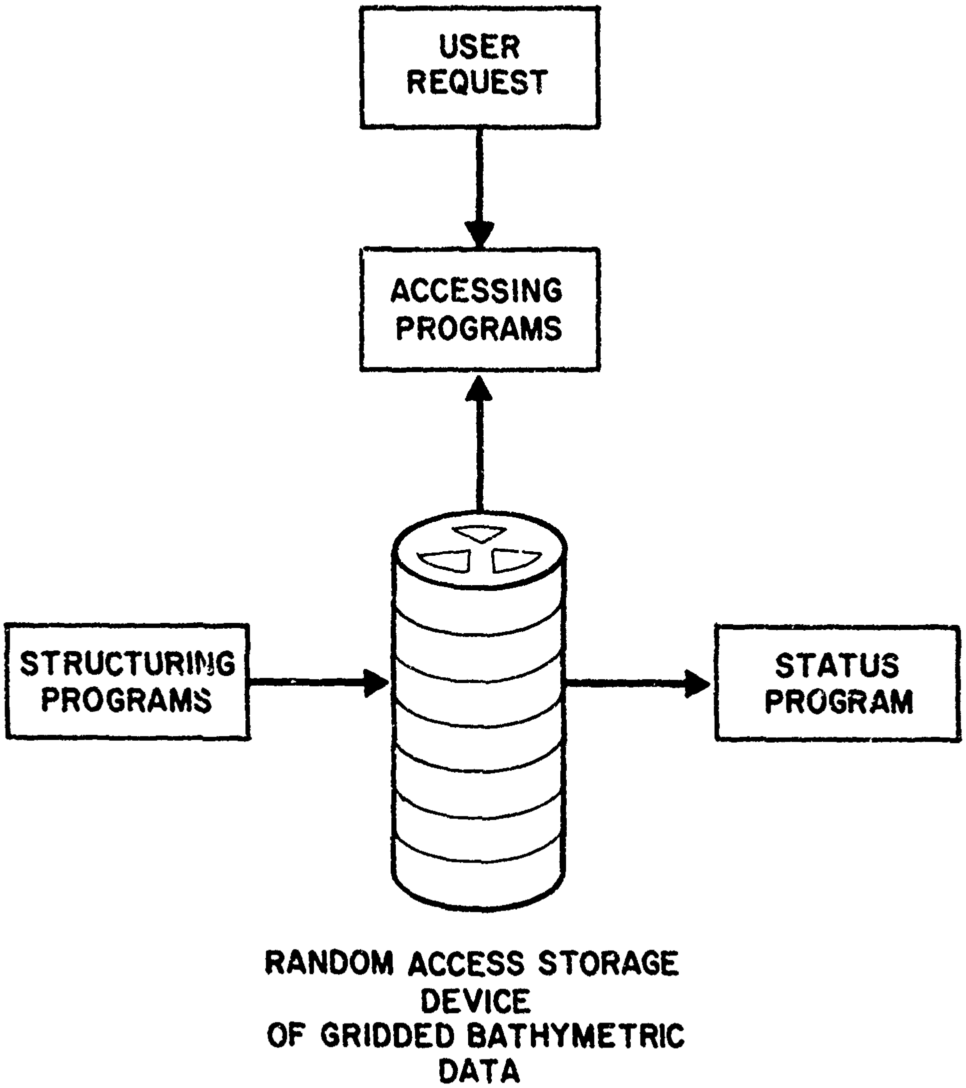


FIGURE 1. SYNTHETIC BATHYMETRIC PROFILING SYSTEM DIAGRAM

and the east-west side is in longitudinal minutes. On a Mercator projection contour chart this is a 5-minute rectangular grid. The bathymetric data are logically formatted to place depth values at the intersection of each 5-minute grid crossing as shown in figure 2.

The next level of structuring is to index the 5-minute cells into 5-degree squares called Marsden Square Locator numbers (MSQLOC) using the Marsden square system which divides the earth surface into 5-degree squares (fig. 3). Further subdivision of the Marsden square by quadrants is shown in figure 4. The MSQLOC is the quadrant number followed by the Marsden square number as follows:

Marsden square number+quadrant = MSQLOC

Example: 036+2 = 0362

The MSQLOC is a unique worldwide reference to each 5-degree square of gridded bathymetric data. The MSQLOC area includes a 5-minute overlap of all sides as shown in figure 5 for MSQLOC 0362.

The gridded bathymetric data base is created following the procedure used by Davis and Kontis (1970). However, accurate synthetic data derived from large and medium-scale bathymetric charts are used instead of original survey data. The synthetic track data are derived from charts by superimposing parallel track lines, 5 minutes apart, over the MSQLOC area. Extraction of the data usually starts from the lower left corner. The orientation of the track lines can be any direction from west-east (90° bearing) to nearly south-north (1° bearing), but not true north, which necessitates changing several statements in the gridding program. The only other restriction is that the first track be a west-east track across the MSQLOC area. The remaining tracks may be of any orientation and in any order.

The data are extracted from the chart by digitizing the intersections of the synthetic track with the contours sequentially along the track. Interpolated points must be extracted for the beginning and end of each full track. These tracks must extend 5 minutes beyond the MSQLOC area on all sides as shown in figure 6. Short tracks may be added to emphasize certain topographic characteristics such as spot elevations. These can be extracted at any orientation except true north-south as shown in figure 6B.

Each digitized track is assigned a sequence number, but the physical order of the tracks in the card deck is arbitrary after the first track. These digitized tracks are inputs to the gridding program. The output from that program is a punched deck of gridded bathymetric data with the point or origin in the lower left corner.

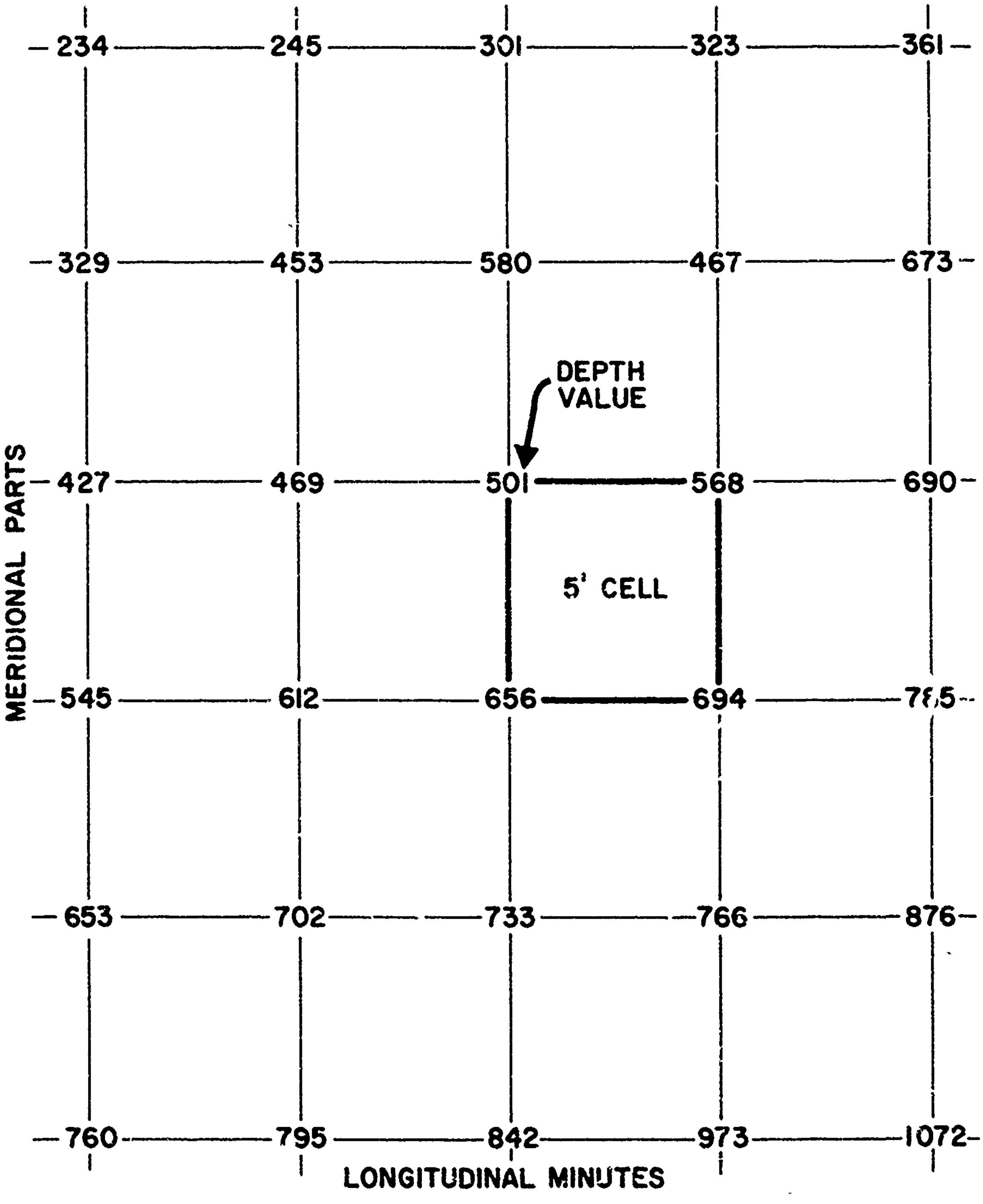


FIGURE 2. SYNBAPS LOGICAL DATA GRID

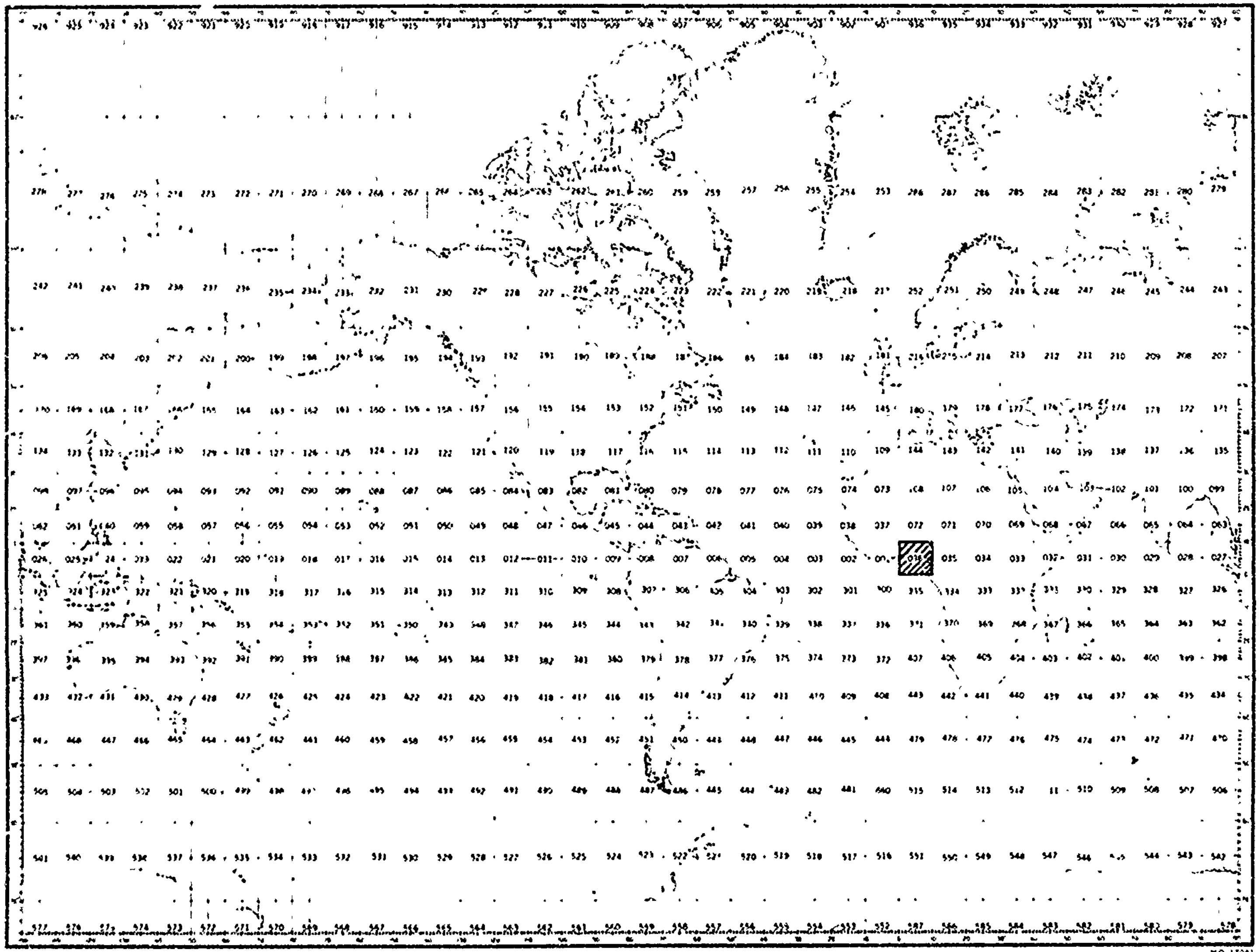


FIGURE 3. MARSDEN SQUARE CHART

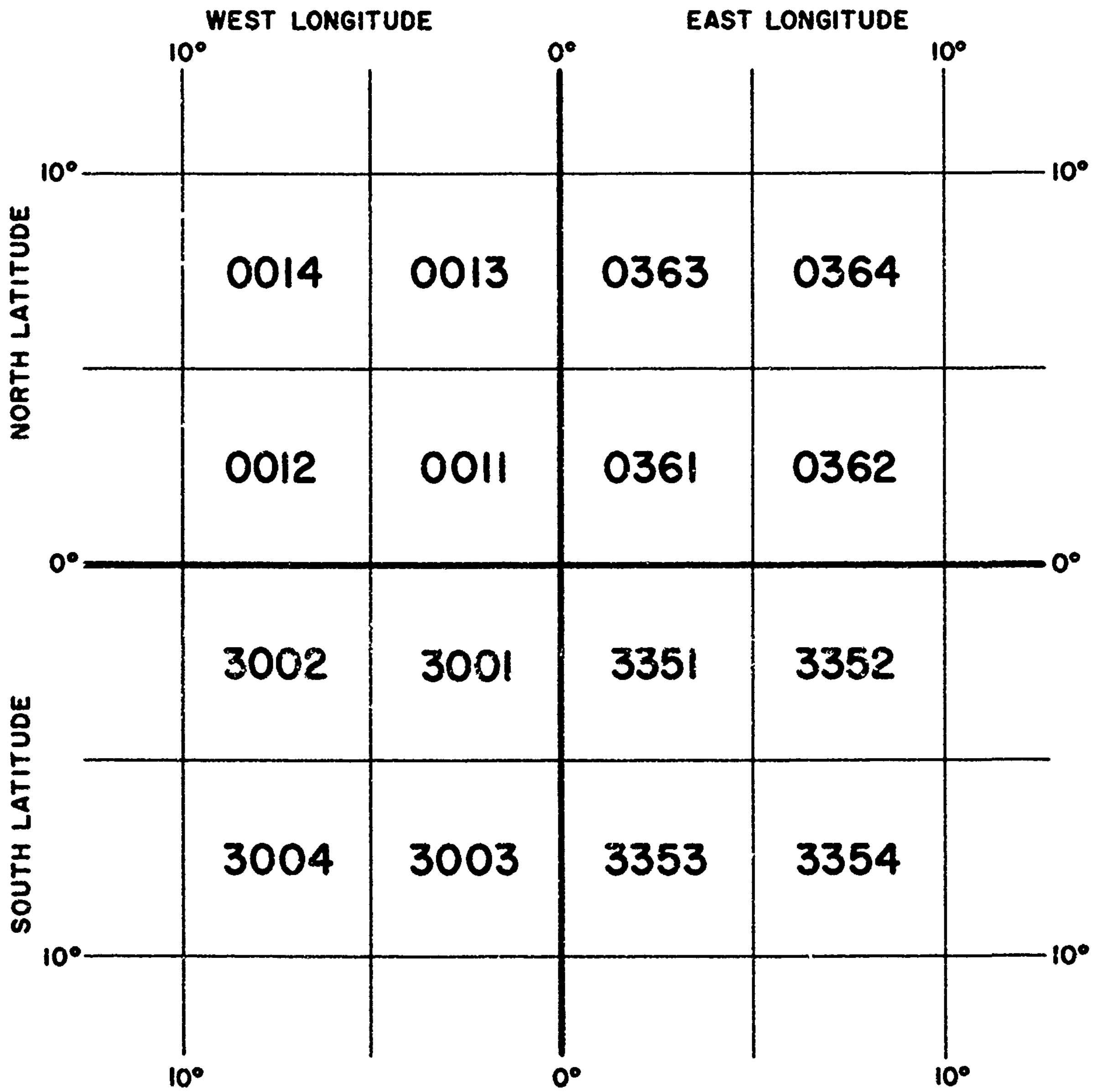


FIGURE 4. MARSDEN SQUARE QUADRANTS AS MSQLOC AREAS

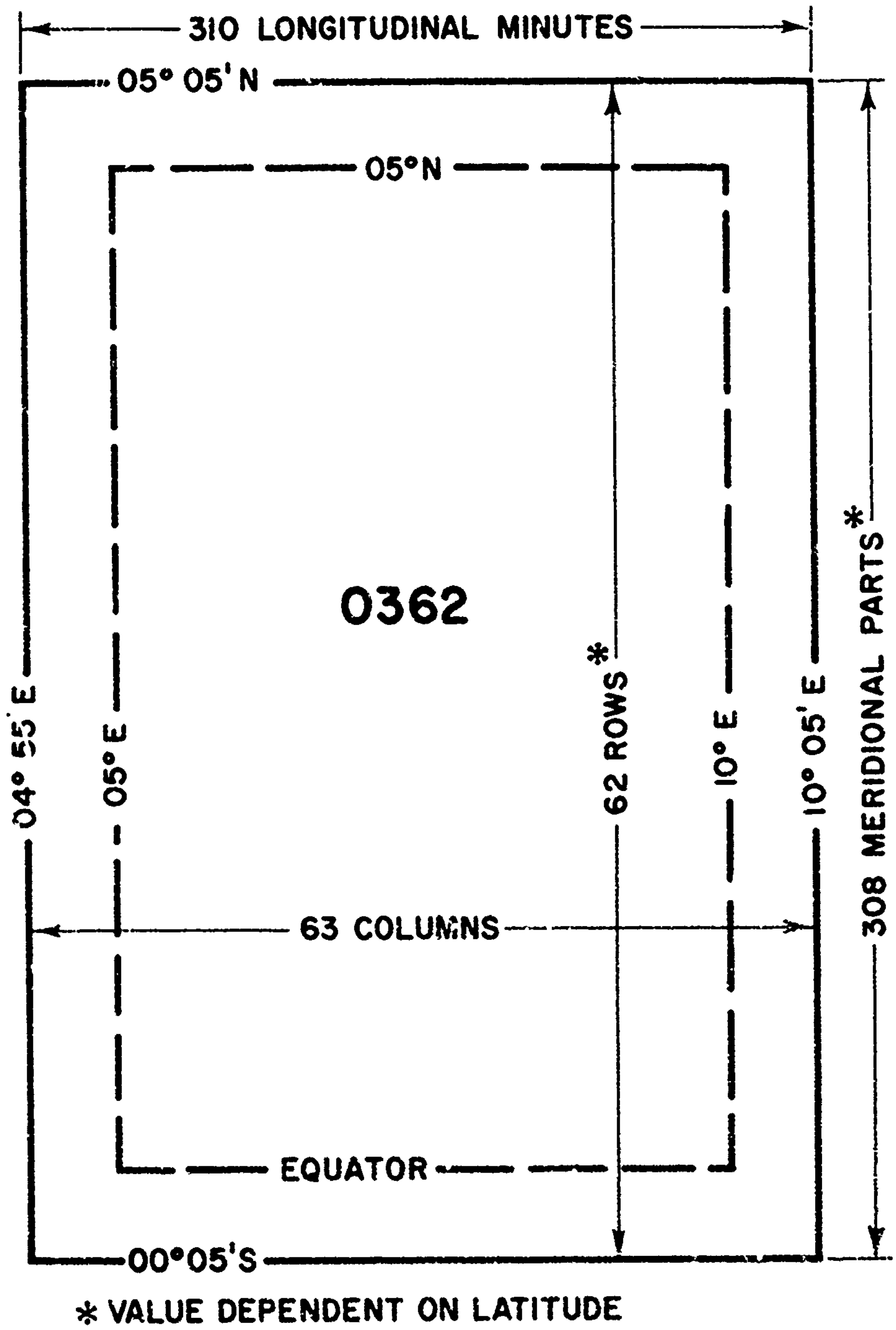


FIGURE 5. EXAMPLE OF MSQLOC AREA

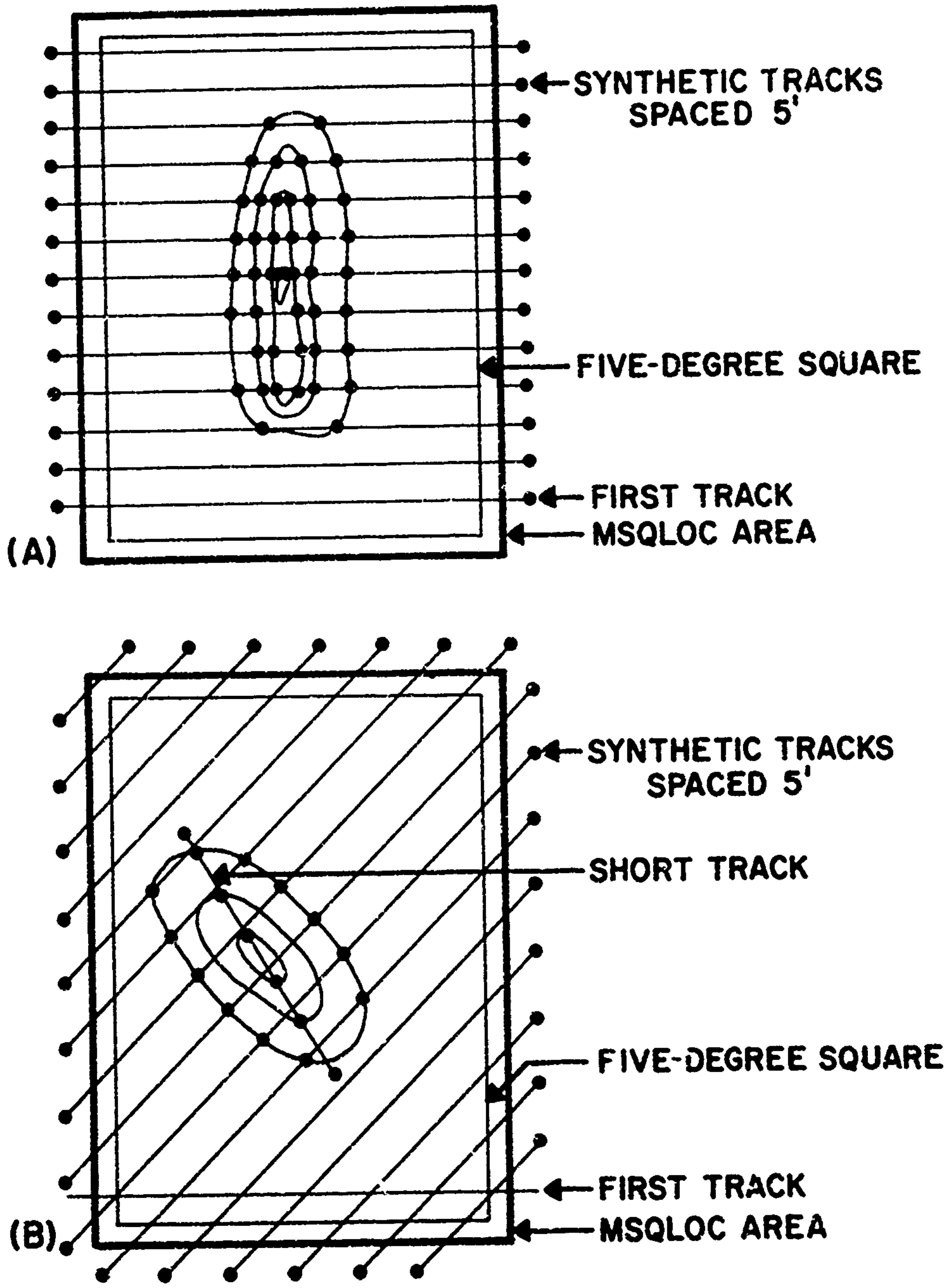


FIGURE 6. EXAMPLE OF SYNTHETIC TRACK ORIENTATION

These data are physically unformatted. A number of error checks are made before and after the gridded bathymetric data are created. The gridded data then are placed on a random-access storage device using a predetermined "look-up" table (list of acronyms). At this point the data are ready to be accessed.

At present, a bathymetric profile can be generated up to 8,000 nm long and crossing 30 MSQLOC areas. This limitation can be increased if necessary. The accessing is initiated by supplying the latitude and longitude of a beginning and end point or the latitude and longitude of a beginning point with the bearing and maximum range. Combinations of these accessing schemes can also be used.

The first step in retrieving a profile from the data bank is to generate its great-circle path. At the same time each MSQLOC area that the path crosses is identified and a search table of MSQLOC areas is created. For each MSQLOC area the search table contains the latitude and longitude of the first and last point in that MSQLOC area, the forward-looking bearing at both points, the accumulated range from zero for both points, and the MSQLOC area number. In turn, each MSQLOC area is called from the random-access storage device via the "look-up" table and the profile for that block of data is generated. This partial profile is then placed on a temporary magnetic tape. The next MSQLOC area is called from the random-access storage device and the cycle is repeated. At the end of profile generation the temporary magnetic tape is rewound. The plotting program is then called and the partial profiles are linked, punched on cards, and plotted and/or written on magnetic tape.

The accessing program is structured so that long profiles generally are processed faster than numerous short profiles that total the same mileage. The great-circle path generation requires about 10 seconds for an 8,000-nm profile plus about 5 seconds for each full MSQLOC area crossed for the interpolation.

The only maintenance to be performed to the system is the eventual updating of the gridded bathymetric data based on the random-access storage device. This is easily accomplished by recycling through the structuring phase of the system any MSQLOC area that requires updating and then replacing that block on the random-access storage device.

A status report can be generated to check any or all MSQLOC areas. This report includes the random-access device's compatible data block size, the actual column and row sizes, the date the data block was added to the random-access device, the MSQLOC area number, the relative address, and the actual data, if required.

SOURCE MATERIAL

Bathymetric contour charts instead of recorded water depths, are the source for the SYNBAPS data base. No computer algorithm (glossary) that can successfully handle all qualities of bathymetric-track-line data, resolve all navigational errors, and can apply a contouring philosophy to such data has been developed. These functions require the subjective judgement, based on knowledge of geologic processes, of the bathymetrist whose final product is the bathymetric contour chart. The bathymetrist's very subjectivity creates the data continuity which is a requisite element of SYNBAPS. A long profile requires on omnidirectional, continuous data base, something that is seldom achieved with either survey or random ship track line data alone. Using areas having high quality and dense data coverage as a framework, the bathymetrist extends, interpolates, and extrapolates regional trends into areas of lesser data to build a continuous picture of the submarine topography.

Although SYNBAPS is designed for worldwide application, initially a data base will be created only for the Northern Hemisphere, and possibly the Indian Ocean. Other regions will be added to the data base when sufficient continuous data become available. The charts used for the North Pacific Ocean will be large to medium scale (1:1,000,000 or larger) versions of the U.S. Naval Oceanographic Office H.O. Pubs. 1301, 1302, and 1303 (U.S. Naval Oceanographic Office 1969, 1971A and B). Recent unpublished large-to-medium scale charts compiled by the U.S. Naval Oceanographic Office will be utilized for the North Atlantic and Mediterranean Sea. Where applicable, classified data can be incorporated in the data base without compromising security. The gridded data point from a classified chart, which was contoured from classified data or from a mixture of classified and unclassified data, will be indistinguishable from a data point from an unclassified chart. Only the originator will know which depth values were created from classified data and that they may be more accurate than other points. The originator will keep a separate noncomputerized file, indexed by MSQLOC areas, showing the source of the contours, their evaluation, classification, and other pertinent information. There will be no reference to original track spacing, area limits, navigation, sounding device, or platform within the data base. Preparation of the charts for digitization is discussed in more detail in appendix A.

SYSTEM DESCRIPTION

A. Structuring Programs

The relationship between structuring programs is given in a flow diagram in figure 7. The main processing programs are SYNTRACK, SYNCARD, SYNCHEX, SYNGRID, SYNCON2R, and SYNBLOCK (list of acronyms). One additional program that is unique to this particular system is the digitizer scaling program (CALMA 485) which scales on a Mercator chart the latitude, longitude, and depth for each contour intersection along the track. The output from this program is a binary magnetic tape of scaled values. Any digitizer and/or digitizer processor program can be used as long as it generates the same program elements regardless of output mode.

The MSQLOC area to be digitized is mounted and leveled on the digitizer table (fig. 7). Starting in the lower left corner each track is scanned for data points from left to right and from bottom to top. The tracks are scanned an additional 5 minutes on each end to permit interpolation rather than extrapolation on end points in the gridding program. The MSQLOC identification and operator name are entered as a header information group before the data scanning is begun. The binary coded decimal (BCD) magnetic tape generated by the digitizer is processed by the CALMA 485 processor program to produce a binary magnetic tape of scaled latitude, longitude, and depth data. The binary tape is processed by SYNTRACK which:

- breaks up the data string into tracks,
- checks for missing data points,
- checks for operator errors,
- reformats the data to card image, and
- punches out a header card, track card, data cards (one point per card), and a blank card.

An illustration of this deck structure is given in figure 8. After errors have been corrected, the card deck generated by SYNTRACK is run through the SYNCARD program. This program checks to insure that the longitudes of contour intersections are not repeated, but either increase or decrease depending upon quadrant. In addition, this program tests the depth value to determine if it is within about plus or minus two times the contour interval. In regions of rapid depth change contours may be skipped if they are evenly spaced. All errors are flagged for correction.

After all corrections have been made, the card deck is run through the track plotting program (SYNCHEX). This program plots the tracks as they were digitized and annotates each contour intersection on the synthetic track line with cross ticks. This plot

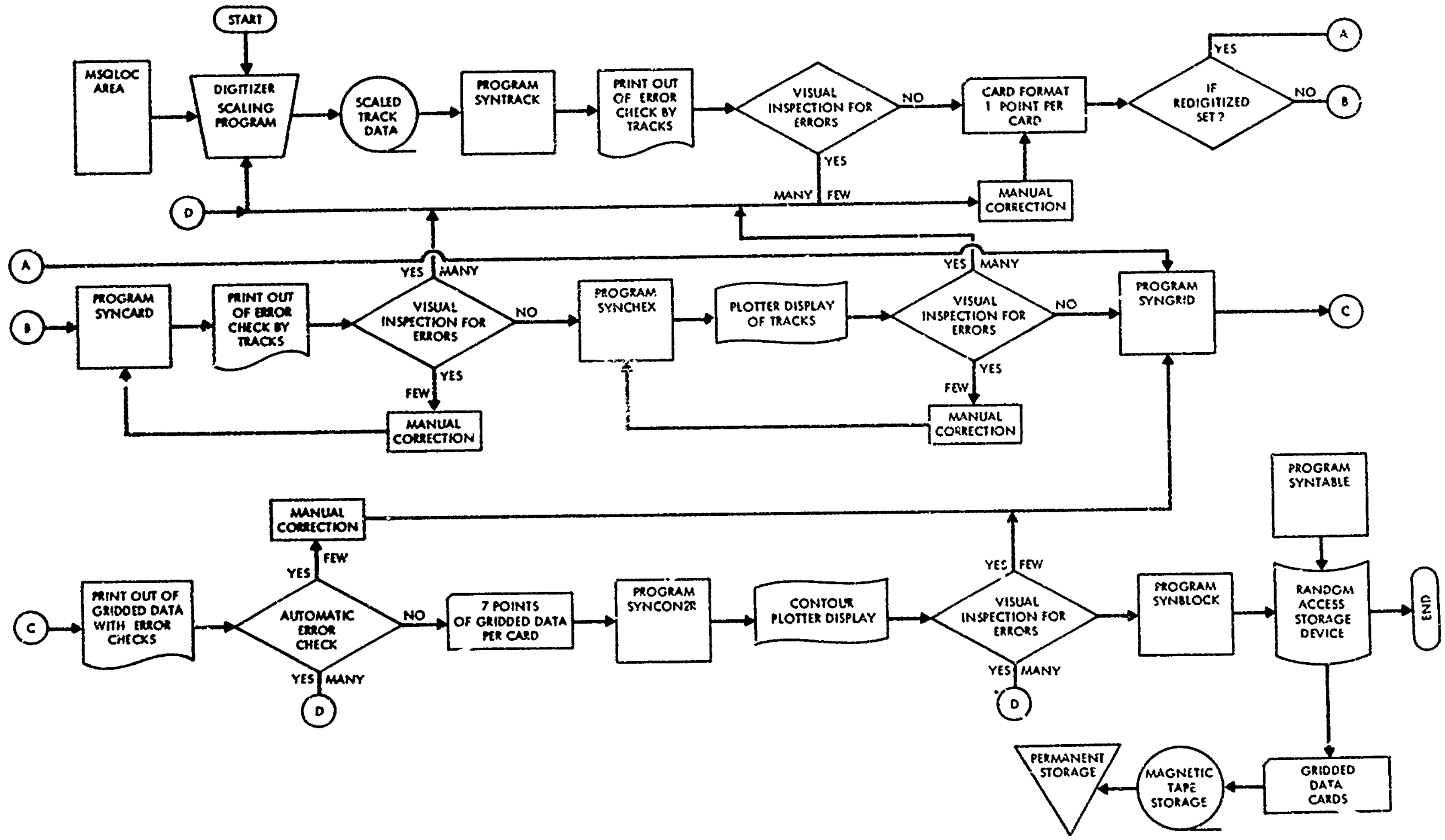


FIGURE 7. SYNBAPS STRUCTURING PROGRAMS FLOW DIAGRAM

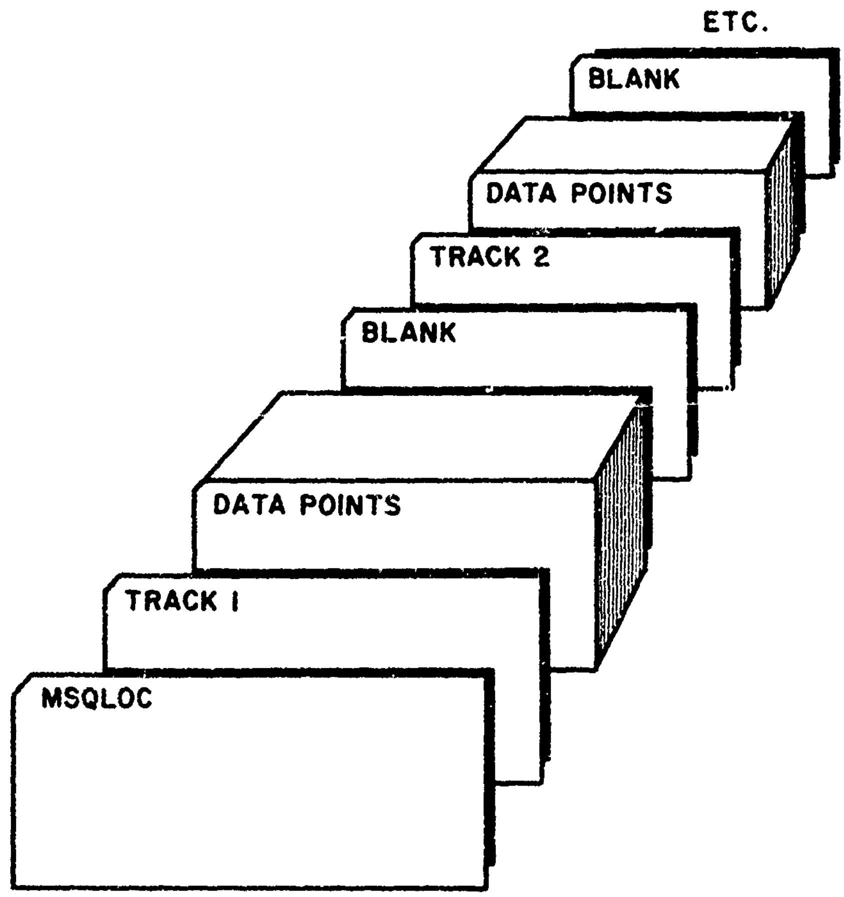


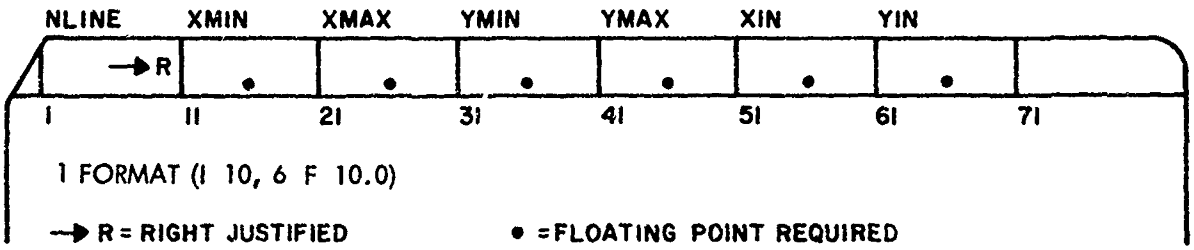
FIGURE 8. OUTPUT DECK STRUCTURE FROM SYNTRACK

insures that the proper and sufficient number of points have been extracted from the MSQLOC area. Additional tracks of data can be created at this time, if required by the complexity of the submarine topography. SYNCHEX requires a control card that is in reality the first data card. The format for this card is given in figure 9. If no further corrections or additions are to be made to the data deck, the MSQLOC area is ready for conversion to gridded bathymetry.

The SYNGRID program is fundamental to the structuring phase of SYNBAPS. SYNGRID transforms the synthetic track line data into gridded bathymetric data. The mathematical foundation and philosophy behind the one-dimensional cubic spline used to structure the gridded data is fully explained by Davis and Kontis (1970). SYNGRID is a modification by Davis of his original program (SPLINT) to handle bathymetric data instead of gravity data. SYNGRID is very flexible as it grids track-line-point data on either a Mercator projection or a Cartesian coordinate system and can compute mean data for various size cells on either system. Summarizing Davis and Kontis (1970), the value of this method lies not only in its ability to fit the observed data values but to retain the continuity of the first and second derivatives. This method might be considered the mathematical analog of the draftsman's plastic spline.

Because the cubic spline is a function of only one independent variable, the data obtained along a synthetic track line must be adjusted to lie on a straight line. Under most conditions this creates no problem as the data are digitized along straight lines. The interpolation formula used by Davis fits each data exactly, has continuous first and second derivations, and is a simple cubic polynomial in x within the interval between each pair of data points. The distance along the track line then may be interpreted as the independent variable. Therefore, taking the data from one track at a time, the position of the data points are converted into x, y coordinates and a least squares straight line is fitted to these locations. Because no statistical significance is attached to this operation, either x or y may be considered the independent variable. The computer program listed in appendix B considers x the equivalent longitude as the independent variable. If the survey tracks happen to run exactly north-south, the program should be modified to consider y as the independent variable.

The perpendicular distance be 'een the least squares straight line and each data point is determined and used to project the points orthogonally onto the line with an adjusted data value (based on an estimate of the local gradient) assumed to be a function of distance only. If the perpendicular distance between this point and the least squares line is less than predetermined



NLINE	=	TOTAL NUMBER OF TRACKS
XMIN	=	minimum number of minutes from prime meridian - east or west
XMAX	=	maximum number of minutes from prime meridian - east or west
YMIN	=	minimum number of meridional parts from equator - north or south
YMAX	=	maximum number of meridional parts from equator - north or south
XIN	=	east-west dimension of plot in inches
YIN	=	north-south dimension of plot in inches

NOTE:
1. North and east are positive, south and west are negative.
2. MSQLOC areas require 5° minutes of overlap on all sides.
3. Meridional parts are found in reference:
   Naval Oceanographic Office, 1962
   H.O. Pub. 9 - Table 5

FIGURE 9. SYNCHEX CONTROL CARD FOR TRACK PLOTTING OF MSQLOC AREA

pivot distance (usually set at 0.2 of a meridional part), the value associated with the data point is unchanged. If this distance is greater than the pivot distance, then the adjusted value associated with the mapped coordinates is computed.

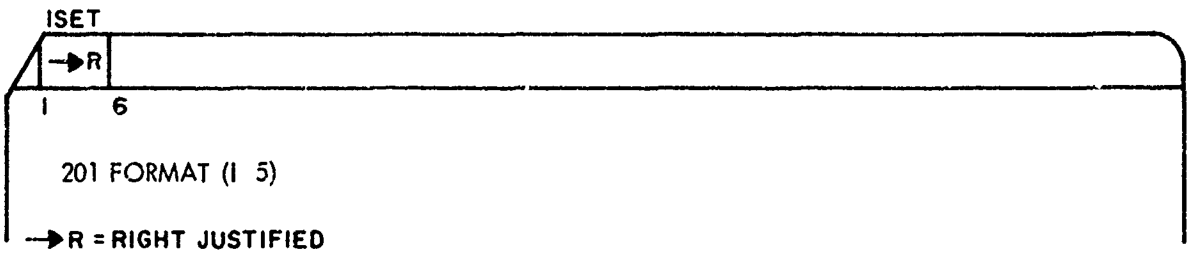
In the computer program for the cubic spline algorithm (SPLINE) contained in appendix B, the pivot distance is selectable via a control card. This pivot distance is usually equal to the maximum distance which one could move a data point without significantly changing its value. In order to minimize the error associated with the assumption that the gradient correction is independent of direction, continuous synthetic survey tracks which deviate appreciably from a stright line should be broken up into smaller segments with each segment treated as a separate track. The mapped coordinates and adjusted data values may be considered as irregularly spaced digital samples from a function whose independent variable is distance along the track from some arbitrary starting point, and whose dependent variable is the adjusted data values.

Utilizing the mapped data, the cubic spline is determined for each track. The cubic spline may then be used to interpolate data values at the intersections of the straight least square track lines and a set of parallel lines whose spacing is equal to the desired final grid spacing (5 minutes). If the direction of the survey tracks is predominantely east-west then the direction of the set of parallel lines is north-south. Similarly, for north-south tracks, the lines are run east-west.

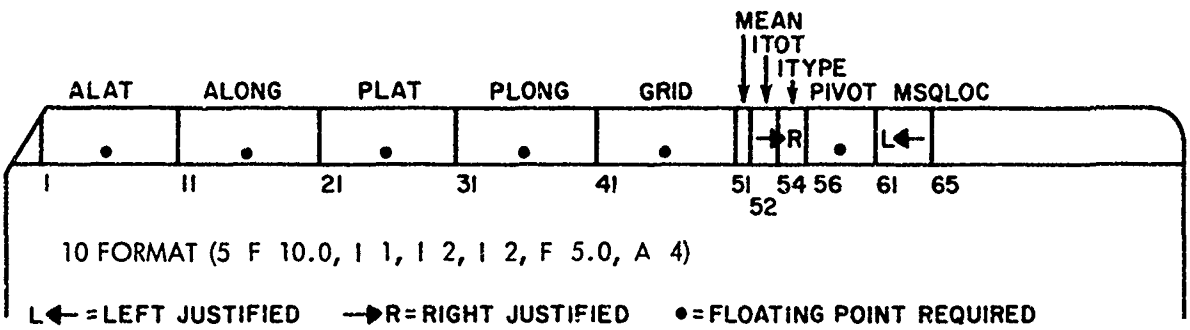
The computer program (app. B), is designed to operate on tracks in any direction, except exactly north-south. The direction of the set of parallel grid lines is controlled by the direction of track line number one. Since the track number designation is arbitrary, this feature allows the user to determine the desired orientation (N-S or E-W) of the parallel grid lines in order to obtain as many intersections as possible.

The interpolated data values generated as outlined in the preceding paragraph may be regarded as unequally spaced digital samples from a function whose independent variable is distance along each of the parallel lines. Application of the spline procedure in this cross track direction produces the final interpolated values at the desired grid points. If mean anomalies are desired, grid points are generated at one-half the final grid spacing and the resulting nine points are averaged to produce the mean value for each grid cell.

The control card formats for SYNGRID are given in figure 10. The output from SYNGRID is a new punched card deck of gridded bathymetry with seven points per card. The printout from SYNGRID



ISETS = number of MSQLOC areas to be processed during a computer run



ALAT	=	latitude of MSQLOC for lower lefthand corner in degrees
ALONG	=	longitude of MSQLOC for lower lefthand corner in degrees
PLAT	=	latitude of MSQLOC for upper righthand corner in degrees
PLONG	=	longitude of MSQLOC for upper righthand corner in degrees
GRID	=	grid spacing for output data in minutes
MEAN	=	blank, no mean computed; =1, mean computed
ITOT	=	total number of tracks of input data
ITYPE	=	1, grid is in Mercator projection; = -1, grid is in X and Y units
PIVOT	=	maximum distance from track for pivot test
MSQLOC	=	Marsden Square Locator area number

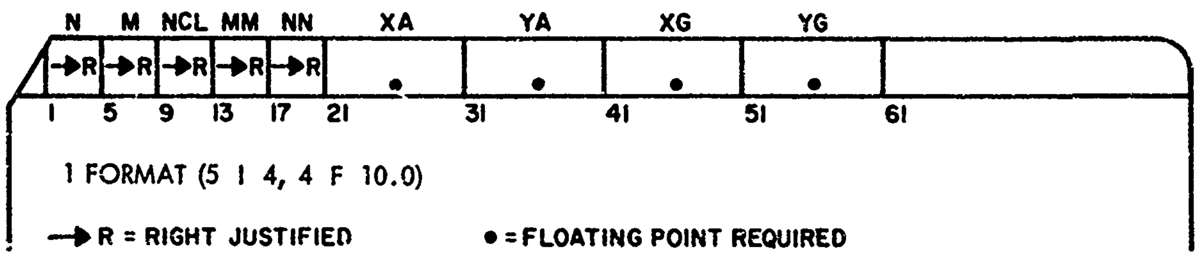
FIGURE 10. SYNGRID CONTROL CARDS FOR GRIDDING TRACK DATA

will indicate if the MSQLOC area has been structured correctly. An even more efficient method of checking is to pass the gridded bathymetric data through the SYNCON2R program.

The SYNCON2R program (fig. 7) plots contours of the gridded data on a Mercator projection at the same scale as the source manuscript. The source manuscript can be overlain by the gridded-data contour plot, for a comparison of content and form. This plotting check requires a 29-inch drum plotter or equivalent, while the SYNCON2R program itself requires a control card (fig. 11). In addition, the DATA statement variable (CL) requires a specification of the contour levels that will be plotted (see app. B). An optional DATA statement variable (LABELS) can be used if labels are desired (see app. B). If the SYNCON2R plot is satisfactory, the gridded bathymetry is loaded on the random-access storage device via the loading program (SYNBLOCK), fig. 7).

Before a block of gridded data can be loaded on the random-access storage device, the device must be primed with a traffic director program (SYNTABLE, fig. 7). SYNTABLE is a predetermined "look-up" table, which gives SYNBLOCK basic information that is needed to place a block of gridded data in its proper address on the device. Using the MSQLOC area number as the key, the table supplies the relative address, the actual block size to be transmitted, and a file key or name. The file key indicates by name in which file in the storage device a particular block of data is to be placed. An example of the "look-up" table printout is given in table 1. In the DATA statement N is equal to the number of MSQLOC areas now on the "look-up" table. The relative address is the physical location from the beginning of the file of the first word of the data block. The actual block size is the quantity of storage required to contain the data plus the identification groups and is an even multiple of 32 (Aiken, et al. 1970). The storage requirement for the actual block size is predetermined and is listed in table 2 by hemisphere latitude bands, which include the overlap.

Using the "look-up" table from SYNTABLE on the random-access storage device, a block of gridded bathymetry can now be loaded by SYNBLOCK. The punched deck of gridded data is preceded by two header cards. The first card contains the number of sets to be loaded and the second card, one for each set, specifies the MSQLOC area number and the column and row information obtained from table 2 (see app. B for exact card formats). The DATA statement N is equal to the number of MSQLOC's presently on the "look-up" table. SYNBLOCK then looks up the file, the relative address, and the block-size information from the preloaded table for each MSQLOC area and places the data in its proper location. An identification group containing the following is placed at the end of the data block:



N	=	number of columns of input data
M	=	number of rows of input data
NCL	=	number of contour levels
MM	=	row's maximum array size
NN	=	column's maximum array size
XA	=	1.0 = minimum number of rows
YA	=	1.0 = minimum number of columns
XG	=	x axis width of plot in inches
YG	=	y axis height of plot in inches

FIGURE 11. SYNCON2R CONTROL CARD FOR CONTOUR PLOTTING

SYNBAPS DISK FILE LOCATOR TABLE

MSQLOC	RELATIVE ADDRESS	SIZE OF BLOCK	FILE KEY
211	0	3936	EO8C
212	3936	3936	EO8C
213	7672	4000	EO8C
214	11872	4000	EO8C
571	15872	4064	EO8C
572	19936	4064	EO8C
573	24000	4128	EO8C
574	28128	4128	EO8C
931	32768	4256	EO8C
932	37024	4256	EO8C
933	41280	4448	EO8C
934	45728	4448	EO8C
1291	50176	4704	EO8C
1292	54880	4704	EO8C
1293	59584	4928	EO8C
1294	65536	4928	EO8C
1651	70464	4928	EO8C
1652	75776	5312	EO8C
1653	81088	5824	EO8C
1654	86912	5824	EO8C
2011	98304	6464	EO8C
2012	104768	6464	EO8C
2013	112096	7328	EO8C
2014	119424	7328	EO8C

TABLE I. EXAMPLE OF "LOOK UP" TABLE FROM SYNTABLE

	LATITUDE BAND (MSQLOC)	ARRAY SIZE COL./ROW	INITIAL STORAGE REQUIRED	ACTUAL STORAGE REQUIRED	APPROX. NO. OF MSQLOC/ BAND	INITIAL TOTAL STORAGE	ACTUAL TOTAL STORAGE
1	0°-5°	63x62	3906	3936	50	195,300	196,800
2	5°-10°	63x63	3936	4000	49	194,481	196,000
3	10°-15°	63x64	4032	4064	49	197,568	199,136
4	15°-20°	63x65	4095	4128	48	196,560	198,144
5	20°-25°	63x67	4221	4256	47	198,387	200,032
6	25°-30°	63x70	4410	4448	46	202,860	204,608
7	30°-35°	63x74	4662	4704	48	223,776	225,792
8	35°-40°	63x78	4914	4928	48	235,872	236,544
9	40°-45°	63x84	5292	5312	42	222,264	223,104
10	45°-50°	63x92	5796	5824	34	197,064	198,016
11	50°-55°	63x102	6426	6464	34	218,484	219,776
12	55°-60°	63x116	7308	7328	33	241,164	241,824
13	60°-65°	63x135	8505	8544	23	195,615	196,512
14	65°-70°	63x163	10269	10304	14	143,766	144,256
15	70°-75°	63x208	13104	13120	20	262,080	262,400
TOTALS			90909	91328	585	3,125,241	3,142,208

NOTE: Table is for the Northern Hemisphere only excluding the Indian Ocean

TABLE II. SYNBAPS WORD STORAGE REQUIREMENTS

NUM = actual size of storage block

ICOL = number of columns of array

IROW = number of rows or array

MSQLOC = Marsden Square Locator area number

IDAY = day that data were placed in storage

MONTH = month that data were placed in storage

IYEAR = year that data were placed in storage

LOCATE = relative address

This completes the structuring phase of SYNBAPS. The punched cards of gridded bathymetric data are loaded on magnetic tape with one MSQLOC area per file using a UTILITY program (Rozanski, et al. 1968). This magnetic tape is saved for backup to the random-access storage device.

B. Accessing Programs

The relationship between accessing programs is given in a flow diagram in figure 12. The two accessing programs are SYNBAPS1 and SYNPLOT (app. C). The request, in the form of control cards, is submitted to the SYNBAPS1 program (fig. 13). The formats for this request may be either all "BEARINGS" or all "POINTS" or can be a mixture of both, as long as the number of beams is correctly indicated for each set (the variable NOOFBM).

With the exception of SYNGRID, only a brief explanation of the program's operation was given in the structuring phase discussion. Because SYNBAPS1 and SYNPLOT may be used by others, they will be described in more detail.

Figure 14 contains a more detailed program flow diagram of SYNBAPS1. When a request is submitted to SYNBAPS1 the first operation is to call in the SEAARCH subroutine to generate the great-circle path to be followed by the profile. SEAARCH uses both the direction solution of the great circle, subroutine GCDIST, and the indirect solution GCPATH (Chang, 1969A and B) to create a latitude, longitude, forward bearing, and range for each nautical-mile point from the beginning to the end of a profile. In addition, subroutine MSQFQ is used to calculate the MSQLOC area for each of the points. SEAARCH then creates a range search table of only those points that start a profile, enter or exit a MSQLOC area, or terminate a profile. This table is printed out and also placed in COMMON.

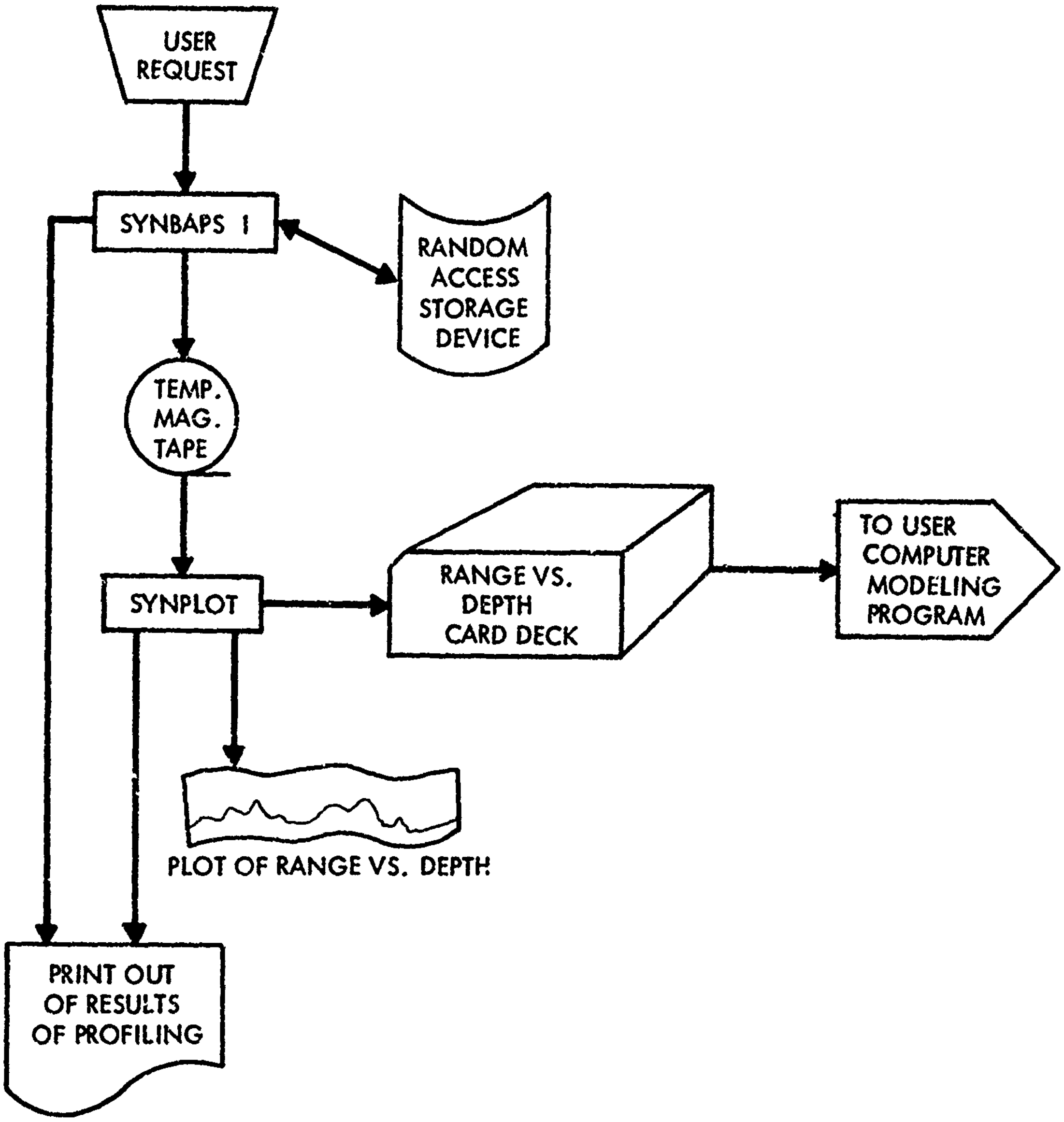
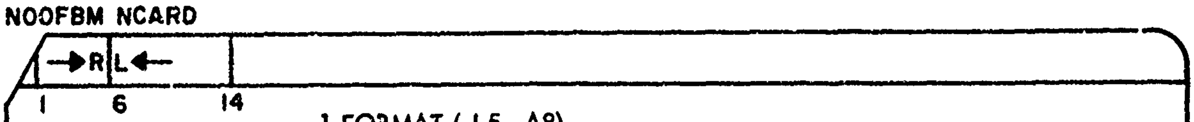


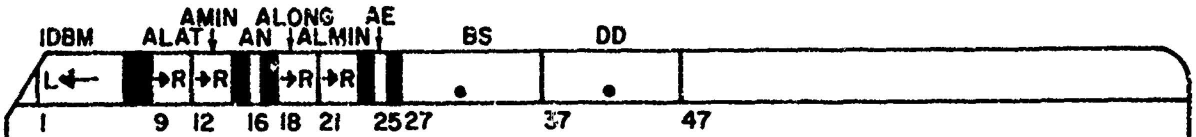
FIGURE 12. SYNBAPS ACCESSING PROGRAMS FLOW DIAGRAM



1 FORMAT (I 5, A8)

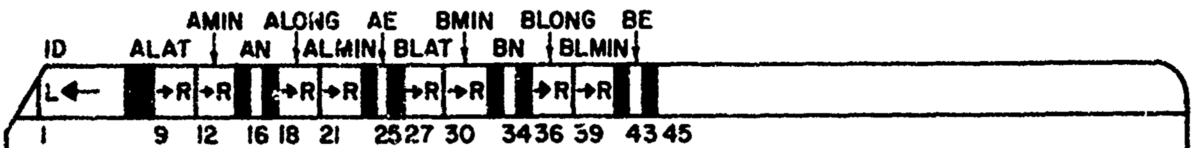
NOOFBM = number of profiles of NCARD type to be processed
NCARD = "BEARINGS" OR "POINTS"

L← = LEFT JUSTIFIED →R = RIGHT JUSTIFIED



1 FORMAT (A6, 2X, 2((F3.0, F3.0), 1X, A1, 1X), 2 F 10.0)
FOR NCARD = "BEARING"

L← = LEFT JUSTIFIED →R = RIGHT JUSTIFIED ● = FLOATING POINT REQUIRED



2 FORMAT (A6, 2X, 4(F3.0, F3.0, 1X, A1, 1X))

FOR NCARD = "POINTS"

IDBM	=	unique profile I.D. (alphanumeric)
ALAT	=	degree of latitude - start point
AMIN	=	minute of latitude - start point
AN, BN	=	hemisphere indicator, N or S
ALONG	=	degree of longitude - start point
ALMIN	=	minute of longitude - start point
AE, BE	=	hemisphere indicator, E or W
BS	=	bearing from start point for "BEARING" card only
DD	=	maximum range from start point for "BEARING" card only
BLAT	=	degree of latitude - end point
BMIN	=	minute of latitude - end point
BLONG	=	degree of longitude - end point
BLMIN	=	minute of longitude - end point

FIGURE 13. SYNBAPS1 PROFILE REQUEST CONTROL CARD

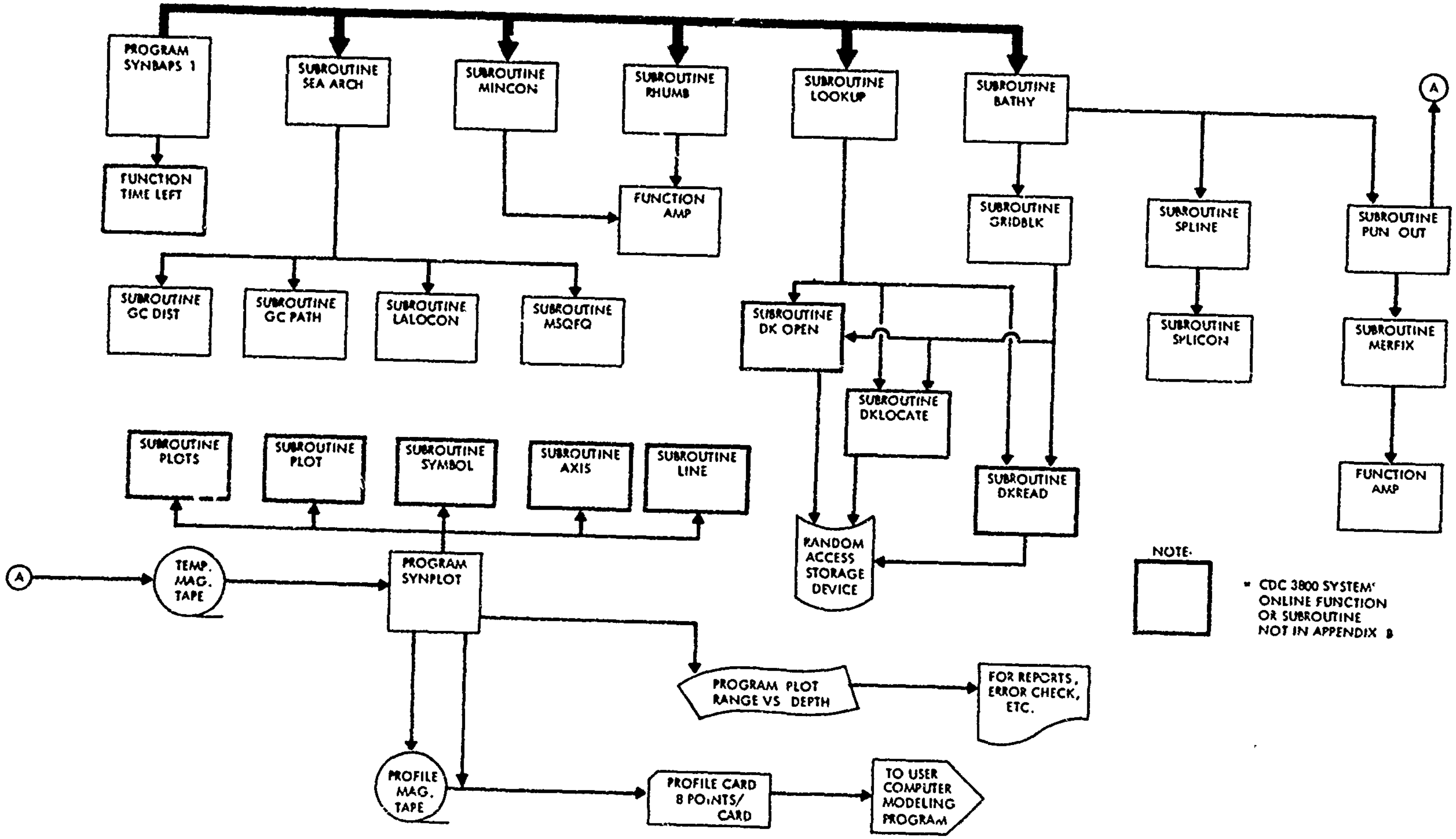


FIGURE 14. ACCESSING PROGRAMS DETAIL FLOW DIAGRAM

Subroutine MINCON is called in to calculate the starting point for the profile within MSQLOC in minutes from the lower left corner. MINCON uses the function AMP to calculate the meridional parts for the latitude component. The mathematical foundation for AMP is given in Thomas (1964) and in U.S. Naval Oceanographic Office (1962).

Subroutine RHUMB is called in to calculate, using AMP, the rhumb line bearing through the MSQLOC area. A rhumb line is used here because the subroutine BATHY can only interpolate along a straight line. The rhumb line approximates a chord of the great-circle path on a Mercator chart with the maximum deviation from the great circle at the approximate midpoint of that chord in the MSQLOC area. This deviation varies from zero to a maximum of about two nautical miles depending upon the great-circle path orientation. Maximum deviations occur in east-west paths in high latitudes, but are considered a necessary trade-off for the system's overall speed of operation.

The random-access storage device is queried by the subroutine LOOKUP, which passes through the SYNTABLE to find the file key and the relative address of the MSQLOC area, then extracts the actual block size and the column and row information. These parameters are used by the subroutine BATHY to extract gridded bathymetric data for the MSQLOC area.

From subroutine BATHY the subroutine GRIDBLK calls in the gridded data. Subroutine BATHY determines which quadrant the rhumb line will pass through so as to maximize the number of intersections for interpolation. This quadrant will determine whether or not the columns or the rows will be the independent variable for the cubic spline. The quadrant arrangement is shown in figure 15.

If the rhumb line falls in quadrants 2 or 4, the direction of the first interpolation is along a column and the independent variable is the distance from the origin along the column to the intersection of the rhumb line. If the rhumb line falls in quadrants 1 or 3, the interpolation will be along a row and the independent variable then is the distance from the origin along the row to the intersection with the rhumb line. At the intersection a value is interpolated by the cubic spline using the gridded data values along that column (or row) as the dependent variable.

When all the values have been interpolated at each intersection, the values now become the dependent variable while the distance along the rhumb line from the start of the profile becomes the independent variable. The cubic spline is used once more to interpolate the final profile values at distances

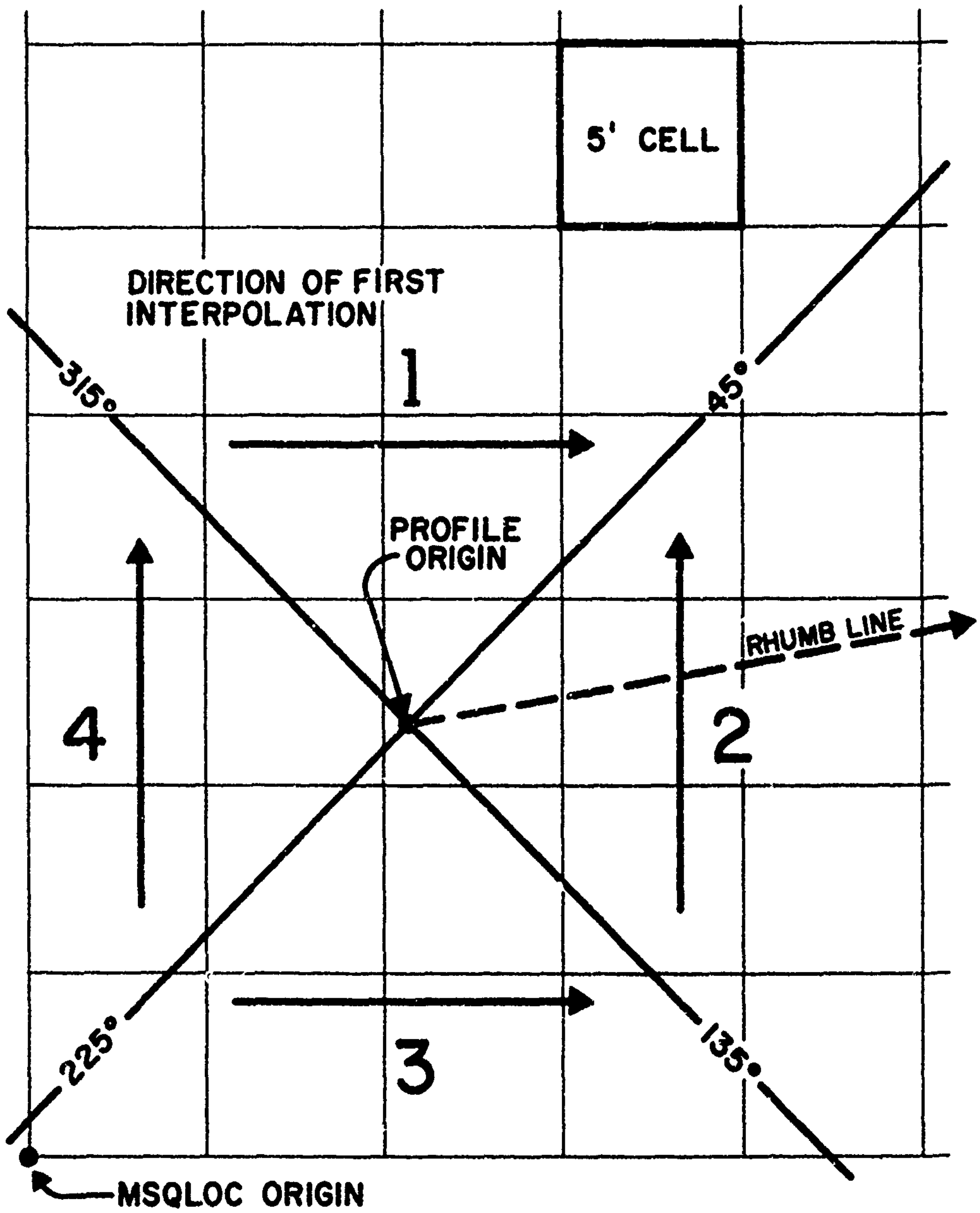


FIGURE 15. QUADRANTS FOR SUBROUTINE BATHY

corresponding to every meridional part along the rhumb line to the end of the MSQLOC area. An example of this rotation is given in figure 16. When a profile for a MSQLOC has been generated, BATHY calls the PUNOUT subroutine to put the MSQLOC profile data on a temporary magnetic tape. MERFIX and AMP are used by PUNOUT to calculate the rhumb line distance in meridional parts and set up a scaling factor. The parameters are used by PUNOUT to adjust the profile generated by BATHY, which is in meridional parts versus depth, to a profile which shows nautical miles versus depth by linear interpolation. Only when these operations are complete is the MSQLOC profile data written on the temporary magnetic tape and the next MSQLOC area or the next profile processed.

Each segment of a profile represents a single MSQLOC area. When the individual segments are written on the temporary magnetic tape the depth is in the same units as in the gridded data base and the range is in nautical miles from starting point within the MSQLOC area, which in each case is zero. At the end of the SYNBAPS1 program the temporary magnetic tape is rewound. The program SYNPLOT then reads this tape either on the same or a subsequent run. As each MSQLOC profile segment is read into SYNPLOT it is linked in sequence to the other MSQLOC areas to produce a great-circle profile. If geometric conversion to other depth units is required, it is performed at this point.

When the great-circle profile is complete, it is punched out on cards and the profile is plotted. This process is repeated for as many profiles as desired. Although the format for the punched profile cards is fixed at eight depth-versus-range points per card, the profile-plotting format is very flexible. This flexibility is attained through a control card for SYNPLOT, the format for which is given in figure 17. Generally, whenever SYNBAPS1 cannot find a MSQLOC block of gridded data on the random-access storage device or the plotting dimensions are not set for minimal limits (fig. 17), the processing will halt at that point and skip to the next profile, allowing the job run to continue while an error message is printed out.

The profiles generated by SYNBAPS are intended as input to long-range, acoustic propagation models. Although not necessarily accurate to geophysical or geodetic standards, the sythetic profiles are interpolated to the accuracy required by the models. A depth value is interpolated at each nautical-mile point from the starting point to the terminus of the profile along a great-circle path. Latitude and longitude values are rounded to the nearest minute, and the range is rounded to the nearest nautical mile.

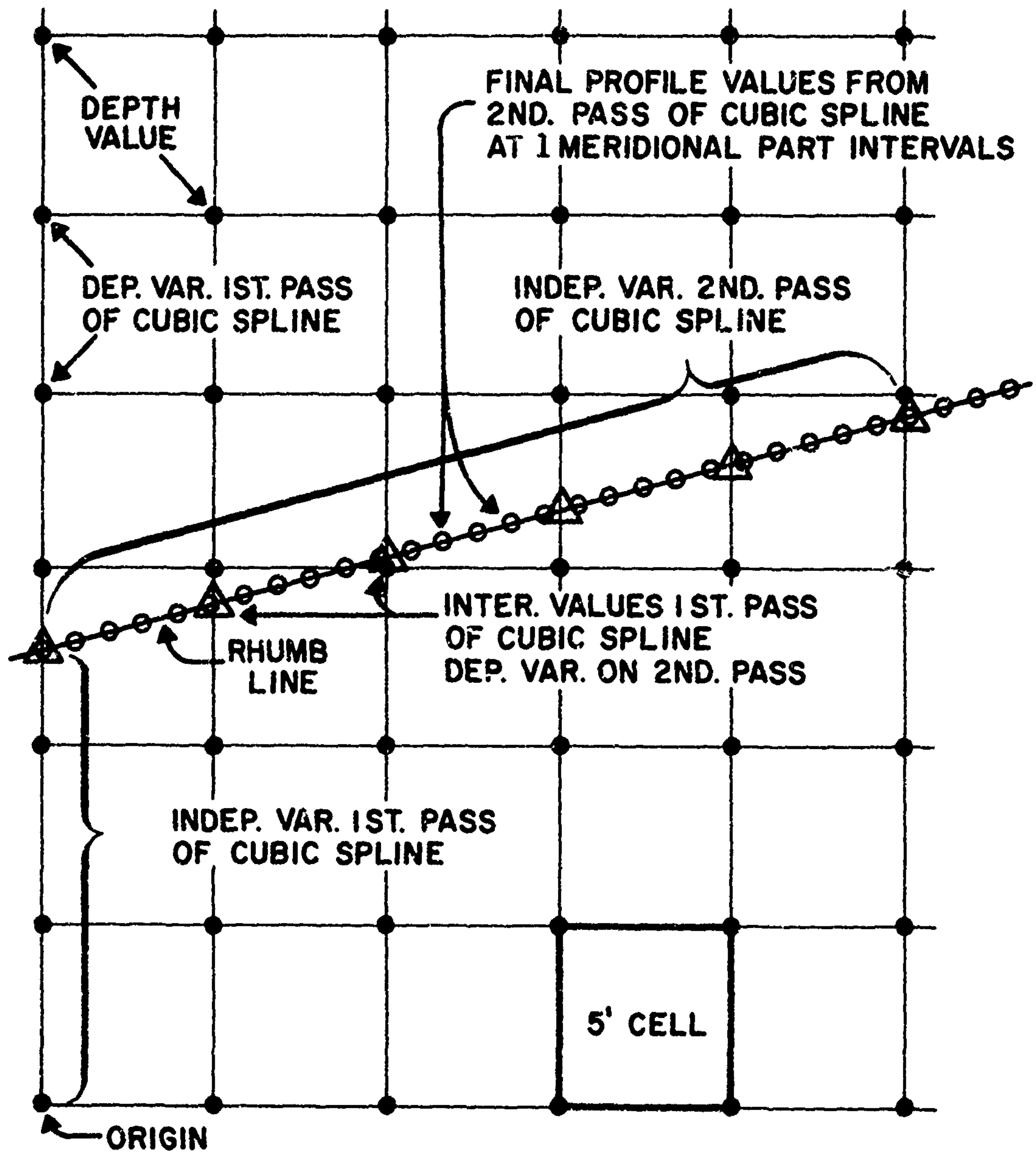
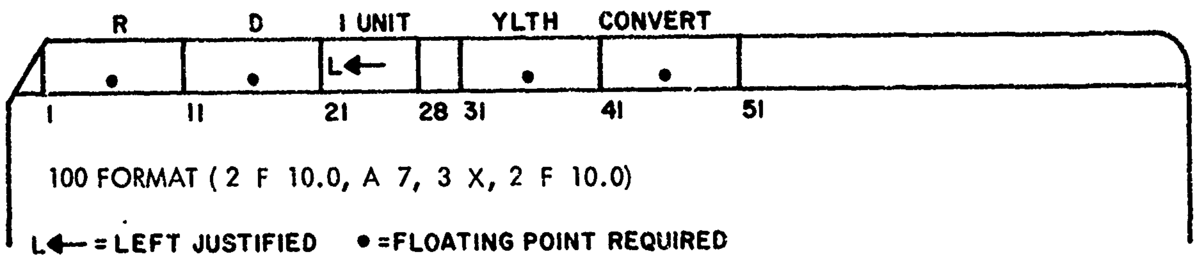


FIGURE 16. PROFILE EXTRACTION FROM GRIDDED DATA BASE



R	=	x axis scaling factor, nautical miles per inch
D	=	y axis scaling factor, meters or fathoms per inch
IUNITS	=	label for y axis (x axis is always in nautical miles)
YLTH	=	the total height of the y axis plot that will be displayed, the maximum is 10 inches. This usually set as a multiple of D. Example: when plotting at 500 fathoms/inch, to be able to display a profile that goes down to a depth of 4500 fathoms, YLTH would equal 9 inches. If not correctly set or if plot exceeds 13 feet on the x axis, that profile is omitted from plotting. However, the cards are still punched.
CONVERT	=	the data base is uncorrected for speed of sound in sea water. For fathoms the assumed standard is 800 fathoms per second, for meters it is 1500 meters per second. To convert from fathoms to meters CONVERT = 1.8750, from meters to fathoms CONVERT = 0.533---3. If no conversion needed CONVERT = blank or 0.0.

FIGURE 17. SYNPLOT CONTROL CARD

The great-circle subroutines are based upon a sphere 21,600 nm in diameter and can have a maximum error of 20 nm over a distance of 1 hemisphere (about 11,000 nm). This amounts to an error of about 2nm/1,000nm of range. For profiles of 1,000 nm or less this error is insignificant in propagation model applications, but it could be important at very long ranges. The magnitude of this error depends upon the difference in shape between the sphere and the oblate spheroid and on the method of path generation. Greater accuracy can be obtained by using a geodesic where the error is 1 m in latitude, longitude, and range and 0.035 sec. in bearing within a hemisphere (Thomas, 1965 and 1970).

Within each MSQLOC area there is a difference between the path followed by the great circle and the actual path along which the depths values are interpolated (fig. 18). Because SYNBAPS1 requires a straight line along which to interpolate depth values, a rhumb line between the first position entering a 5-degree square and the last position before leaving the square is used instead of the curved great-circle path. For all great circles that follow a meridian or the equator this difference is zero. For all other directions, the maximum difference is located at the approximate mid-point along a rhumb line within 5-degree square. Under the most unfavorable condition of high latitude and an east-west orientation, this difference rarely exceeds 2 nm.

Preliminary estimates of the accuracy of the interpolated depth values in the profile plane are $\pm$ 15 fm. This assumes that there are no positional errors in the great-circle path in the horizontal plane. A completed data bank, including regions of smooth to rough topography, will be needed before full error analysis can be undertaken.

C. Status Program

Program SYNSTAT queries the random-access storage device through the SYNTABLE for a listing of the identification group from each MSQLOC gridded data block. This listing includes the file key as in the following example:

MSQLOC	FILE KEY	RELATIVE ADDRESS	ACTUAL BLOCK SIZE	NO. OF COLUMNS	NO. OF ROWS	DATE ADDED TO RANDOM-ACCESS DEVICE
1. 1291	E08C	50176	4704	63	74	18 April 1972
2. 1292	E08C	54880	4704	63	74	19 April 1972

All MSQLOC gridded data blocks or selected ones can be listed. They are selectable through SYNSTAT control cards as shown in figure 19.

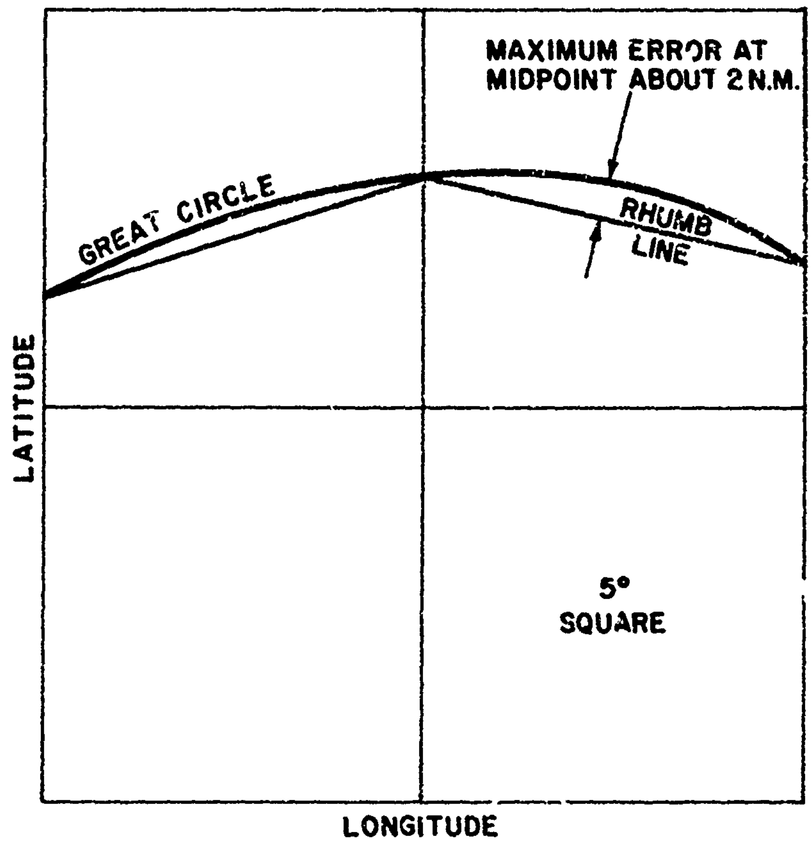
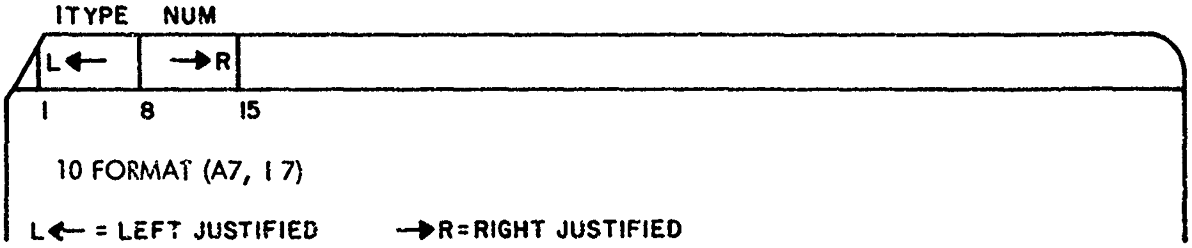
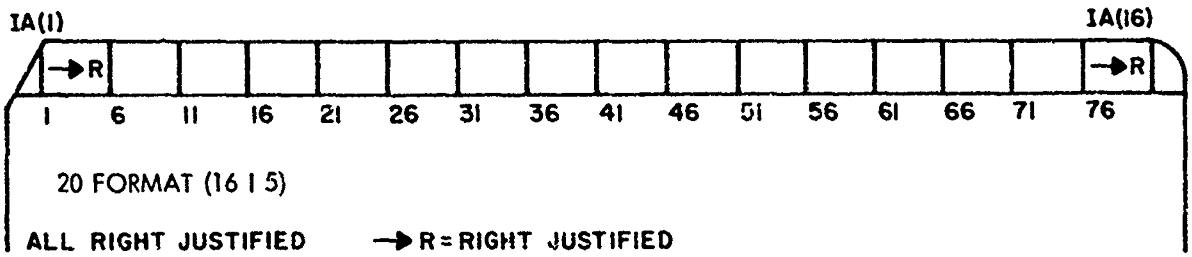


FIGURE 18. DIFFERENCE BETWEEN RHUMB LINE AND GREAT CIRCLE PATH WITHIN A FIVE-DEGREE SQUARE



ITYPE = "ALL", the contents of the complete random access storage device will be listed.
"PARTIAL", only those MSQLOC's listed on the following control cards will be listed out.

NUM = if blank, all the MSQLOC's listed; if present only that number of MSQLOC's on the following control cards will be listed.



IA = array of MSQLOC numbers.

FIGURE 19. SYNSTAT CONTROL CARDS

D. CDC 3800 System Subroutines and Functions

The subroutines used to open the file, position, read, and write on the CDC 813 permanent disk are on-line COMPASS language routines provided by the Naval Research Laboratory, Research Computation Center Staff (Aiken, et al., 1970). These subroutines are DKOPEN, DKLOCATE, DKREAD, and DKWRITE. The subroutine DATA is an off-line COMPASS language routine that retrieves the integer day, month, and year from the computer's internal clock (Houston, 1969). The function TIMELEFT is an on-line COMPASS language routine that retrieves time marks from the computer's internal clock. It is used to time various phases of the structuring and accessing programs operation (Shannon, 1968).

The on-line plotting subroutines PLOTS, PLOT, LINE, SYMBOL, and AXIS are FORTRAN language routines. With the possible exception of LINE and AXIS these routines are part of the standard Calcomp plotter package (Gossett, et al., pending).

Most of the previously mentioned subroutines and functions are unique to the NRL CDC 3800 computer system. However, these routines have counterparts on any large computer system, and their replacement should pose little or no problem.

PROFILE OUTPUT

Two adjoining MSQLOC areas, 1291 and 1292, in the western North Pacific Ocean were selected to test the computer program and were digitized, structured, and placed on the random-access storage device. The location of five test profiles along rhumb lines, subsequently shown in figures 21, 22, and 23, are indexed in figure 20. The contour chart used as an index chart shows only part of the contour data that will input to the data base; therefore, the test profiles show a slight difference in detail. Figure 21, a profile through both MSQLOC areas, shows that the link point between two data blocks is undetectable. This 530-nm profile was generated in 7 seconds.

In figure 22, composed of three profiles A, B, and C, a dashed line is superimposed on each profile. The dashed lines are profiles hand drawn by a bathymetrist, and the solid lines are the computer profiles. All the profiles used the same data base. Although the general shapes for both types of profiles are the same, the cubic spline profiles show details between the contour levels that would otherwise be lost if not captured by the surface of gridded bathymetric data. This is especially true in the more steeply sloping areas because the cubic spline considers data adjoining the profile path. The three profiles in figure 22 show the system's ability to start a profile inside a MSQLOC area. Figures 22A and 22C show profiles that terminate in gently sloping

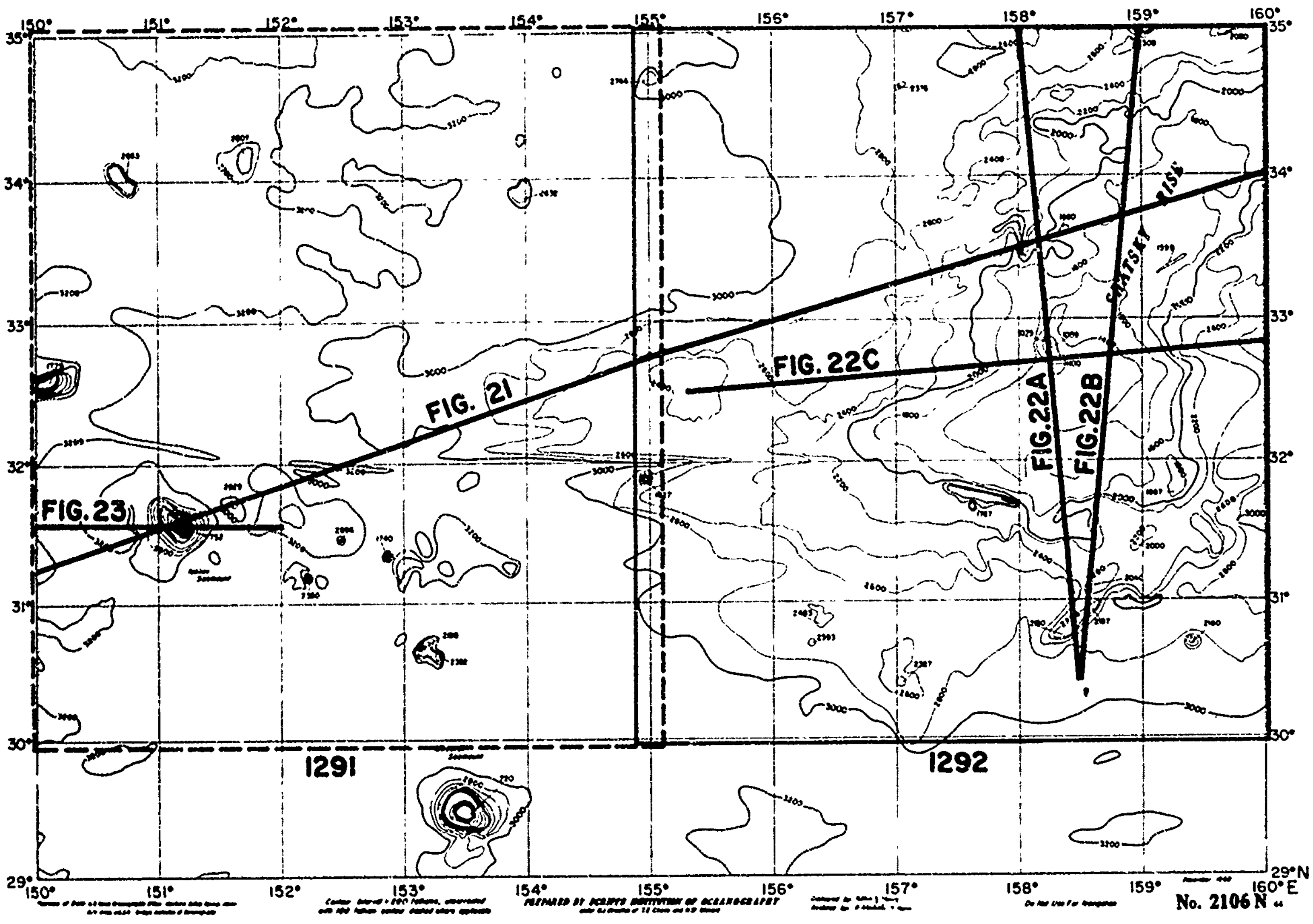


FIGURE 20. INDEX OF SAMPLE PROFILES

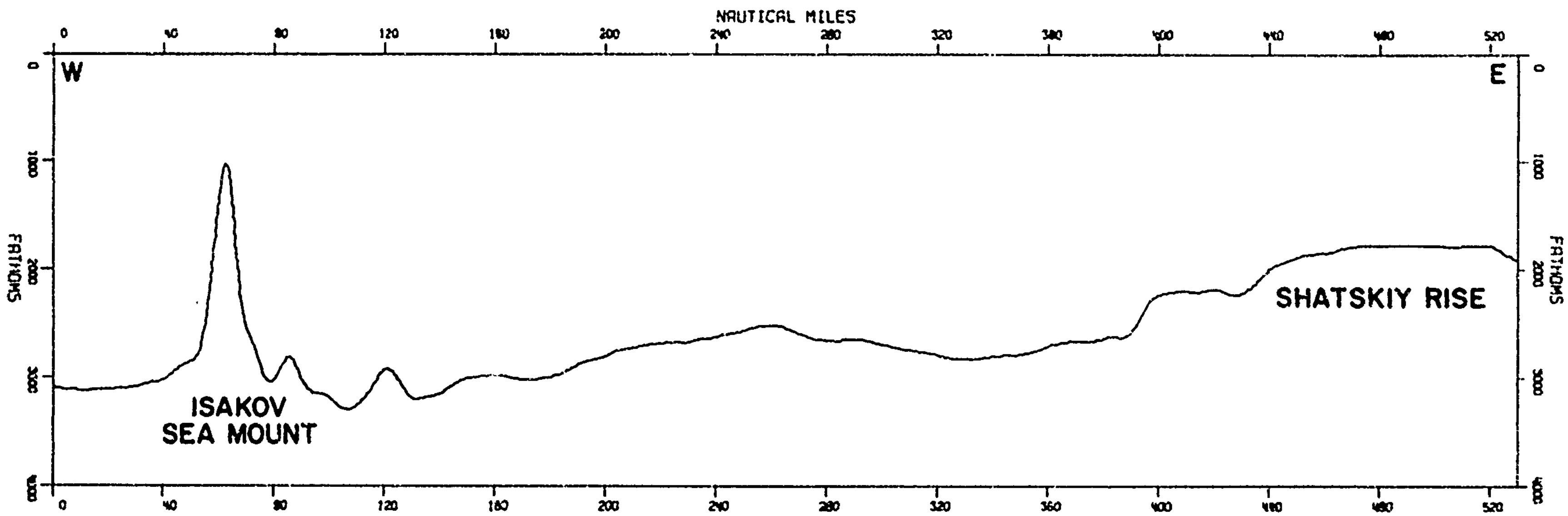


FIGURE 21. PROFILE PASSING THROUGH TWO MSQLOC AREAS

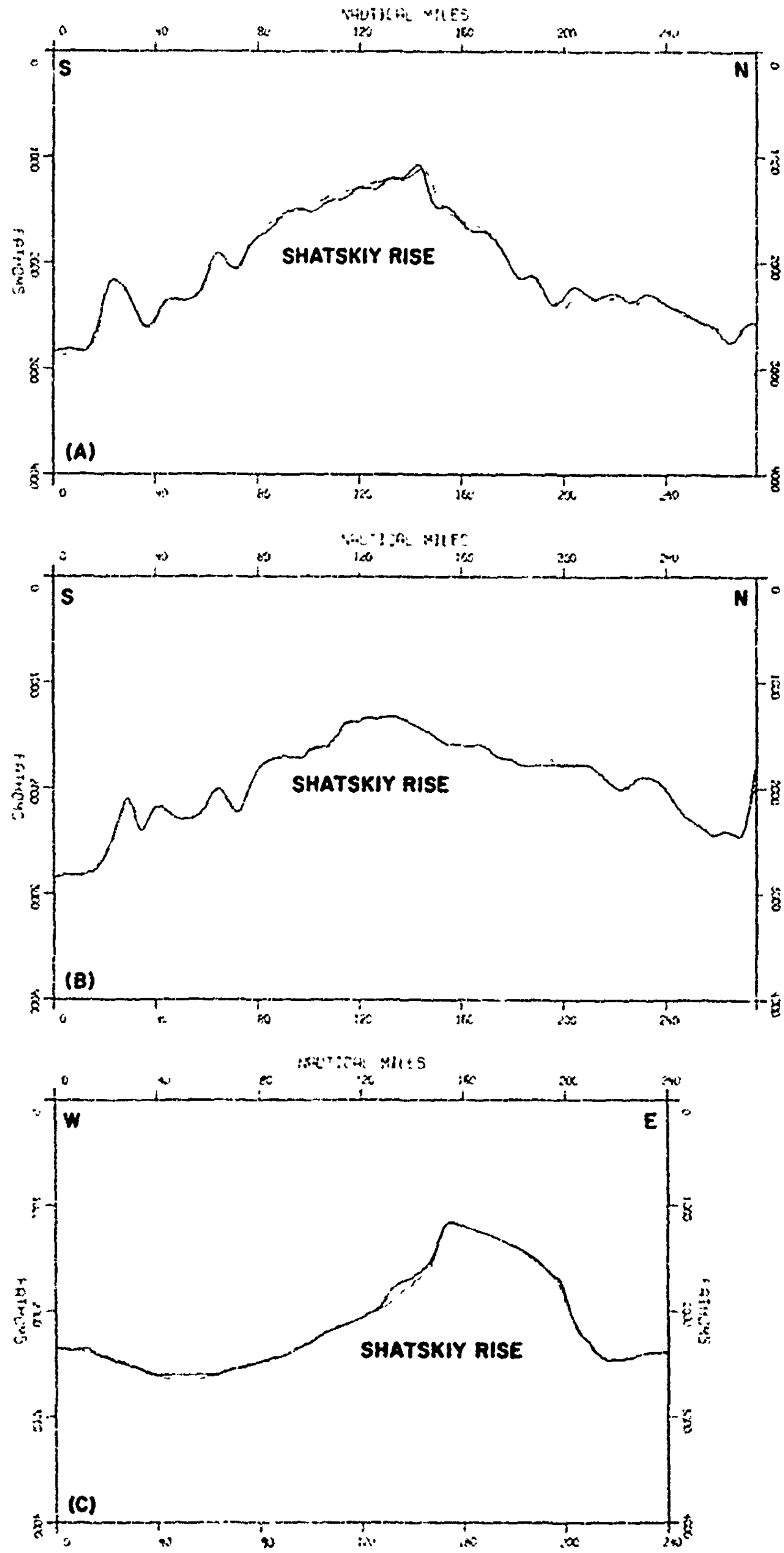


FIGURE 22. CUBIC SPLINE VS MANUAL PROFILES

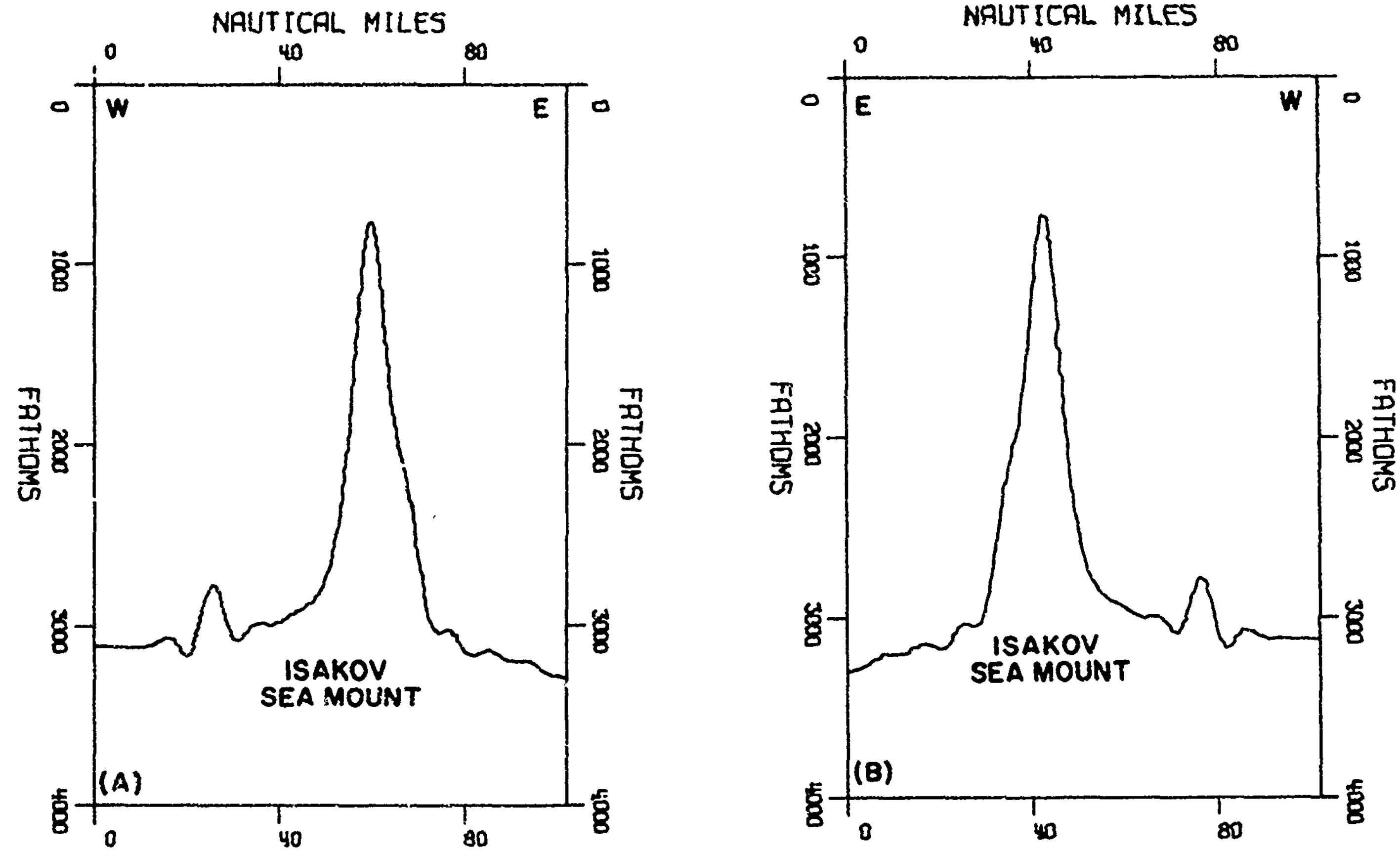


FIGURE 23. MIRROR-IMAGE PROFILES ALONG SAME PATH - DIFFERENT DIRECTIONS

areas, and figure 22B shows a profile terminating on the upslope side of a seamount in the next area to the north. Figures 22B and 22C show that the cubic spline can follow both convex and concave submarine topography equally well.

Figure 23 shows two mirror-image profiles, which illustrate the profile repeatability along the same path in either direction. Profile A was run from west to east, then profile B was run from east to west, both along the same path.

FURTHER MODIFICATIONS, ADDITIONS AND OTHER APPLICATIONS

The first modification to SYNBAPS will replace the great-circle subroutines, GCPATH and GCDIST, in the accessing phase with geodesic subroutines, GEODIST and GEOPATH. The argument list for the new routines will be the same as for the great circle routines. The second modification will replace the contour checking program, SYNCON2R, in the structuring phase with a smoother contour plotting program. A third modification will attempt to increase the overall efficiency (speed of operation) by simplifying the programs. One example is to use buffering statements when writing and reading on the temporary magnetic tape.

An additional program to operate on the SYNBAPS output will be an updated automatic depth correction routine based on Matthews' sound velocity correction tables. This will permit the use of depth values either corrected or uncorrected for speed of sound in sea water.

An additional version of SYNBAPS1, the accessing program, called SYNBAPS2 is being considered. This program will generate eight radial profiles simultaneously from one point to the edge of a MSQLOC area or an irregular chart area. This output could be useful for profile evaluation of site locations where greater detail is required. In addition, SYNBAPS1 can be merged with the NODC Ocean Station Data file to produce a composite plot of the bottom profile and selected sound velocity profiles along a great-circle path. Extending this concept one more step will produce profile plots of various acoustic environmental parameters, such as depth to the axis or bottom of the deep sound channel, by marrying SYNBAPS to an appropriate oceanographic data file or files. The number of possible combinations of oceanographic data with the depth data using SYNBAPS is almost infinite.

A system similar to SYNBAPS, but using land topography, could be applied in radar terrain studies and weather pattern models requiring elevation data.

SUMMARY AND CONCLUSIONS

The SYNBAPS data base was designed to meet the specific and immediate need for bathymetric profiles for acoustic modeling. However, properly used, it offers many applications beyond its preliminary designs.

Often in naval planning as well as in naval operations, speed is as important as accuracy when information is needed. SYNBAPS is not ideally suited to hydrographic charting because some high-frequency information is lost, but it provides very rapid responses. SYNBAPS has these additional features:

- Only data points are stored in the data bank,
- The locations of data points are logically structured on a Mercator projection by 5-minute intersections,
- Random access to the data is by large blocks (5-degree square),
- The data bank is updated by replacing blocks of data,
- The size of the data bank is fixed once it has been created for any ocean area,
- Classified survey data, in chart form, can be incorporated in the data base with no compromise of security,
- Highly compacted forms of the accessing program and the data bank can be used on shipboard.

LIST OF REFERENCES

Aiken, E.L., Denton, D.L., and Shannon, D.P., 1970, 813 disk file subroutine package: U.S. Naval Research Laboratory Memorandum Report 2104 or Computer Bulletin 19, Washington, D.C.

Chang, D., 1969A, A FORTRAN subroutine for locations and bearings at given distances from a starting point along a great circle path: U.S. Naval Research Laboratory Computer Note 33, Washington, D.C.

_____, 1969B, A FORTRAN subroutine for the great circle distance between two points and bearings at the points: U.S. Naval Research Laboratory Computer Note 32, Washington, D.C.

Davis, T.M. and Kontis, A.L., 1970, Spline interpolation algorithms for track-type survey data with application to the computation of mean gravity anomalies: U.S. Naval Oceanographic Office, Technical Report 226, Washington, D.C.

Gossett, D.E., Hill, S.L., Houston, J.H., Kroesh, J., Rogers, G.H., Ulrich, J.T., and Williams, M., (pending): 3800 Calcomp plotter subroutine package, U.S. Naval Research Laboratory Memorandum Report or Computer Bulletin 3, Washington, D.C.

Houston, J.H., 1969, 3800 Computer integer date request subroutine: U.S. Naval Research Laboratory Memorandum Report 2008 or Computer Bulletin 10 Washington, D.C.

Rozanski, T.L., Burgess, J.G., Gossett, D.E., and Shannon, D.P., 1968, Computer utility program: U.S. Naval Research Laboratory Memorandum Report 1935 or Computer Bulletin 1, Washington, D.C.

Shannon, D.P., 1968, 3800 Computer timeleft function: U.S. Naval Research Laboratory Computer Note 5, Washington, D.C.

Thomas, P.D., 1964, Conformal projections in geodesy and cartography: U.S. Dept. of Commerce, Coast and Geodetic Survey, Special Pub. No. 251, 4th edition, 142 p., Washington, D.C.

_____, October 1965, Mathematical models for navigation systems: U.S. Naval Oceanographic Office, Technical Report 182, including figures and tables, 142 p., Washington, D.C.

_____, 1970, Spheroidal geodesics, reference systems, and local geometry: U.S. Naval Oceanographic Office, Special Pub. No. 138, 165 p., Washington, D.C.

U.S. Naval Oceanographic Office, 1962, Tables from American Practical Navigator Bowditch: U.S. Naval Oceanographic Office, H.O. Pub. No. 9 Tables, Washington, D.C.

U.S. Naval Oceanographic Office, 1969, Bathymetric atlas of the northwestern Pacific Ocean, U.S. Naval Oceanographic Office, H.O. Pub. No. 1301, Washington, D.C.

U.S. Naval Oceanographic Office, 1971A, Bathymetric atlas of the northcentral Pacific Ocean, U.S. Naval Oceanographic Office, H.W. Pub. No. 1302, Washington, D.C.

U.S. Naval Oceanographic Office, 1971B, Bathymetric atlas of the northeastern Pacific Ocean, U.S. Naval Oceanographic Office, H.O. Pub. No. 1303, Washington, D.C.

BIBLIOGRAPHY

American Standards Association, 1966, USA Standard vocabulary for information processing, X3.12 - 1966, United States of America Standards Institute, New York, N.Y.

Baker, B.B., Jr., Deebel, W.R., Geisenderfer, R.D., Editors, 1966, Glossary of oceanographic terms: U.S. Naval Oceanographic Office, Special Pub. No. 35, 2nd edition, Washington, D.C.

Bhattacharyya, B.K., 1966, Bicubic spline interpolation as a method for treatment of potential field data: Geophysics, v. 34, p. 402-423.

Grim, P.J., Keller, G.H., and Barday, R.J., 1972, Computer produced profiles of micro-topography as a supplement to contour maps: International Hydrographic Review, v. 49 (1),

_____, 1968, Initialization for the 3800 Calcomp plotter package: U.S. Naval Research Laboratory Computer Note 10, Washington, D.C.

Hunt, L.M., and Groves, D.G., Editors, 1965, A glossary of ocean science and undersea technology terms, Compass Publication, Inc., Arlington, VA.

Pennington, R.H., 1965, Introductory computer methods and numerical analysis: The Macmillan Co., New York.

U.S. Army topographic Command, 1969, DOD glossary of mapping, charting, and geodetic terms: 2nd edition, prepared for Dept. of Defense by Dept. of the Army, Corps of Engineers, U.S. Army Topographic Command, Washington, D.C.

GLOSSARY OF SELECTED TERMS

Accuracy	The degree of freedom from error, that is, the degree of conformity to truth or to a rule. Accuracy is contrasted with precision, e.g., four-place numerals are less precise than six-place numerals; nevertheless a properly computed four-place numeral might be more accurate than an improperly computed six-place numeral.
Address	(1) An identification, as represented by a name, label, or number, for a register, location in storage, or any other data source or destination such as the location of a station in a communication network. (2) Loosely, any part of an instruction that specifies the location of an operand for the instruction.
Algorithm	A finite set of rules that gives a sequence of operations for solving a specific type of problem. It should have the following features, (1) Finiteness, (2) Definiteness, (3) Input, (4) Output, and (5) Effectiveness.
Alphanumeric	Pertaining to a character set that contains both letters and numerals, and usually other characters. Synonymous with Alphameric.
Argument list	List of the formal parameters of a subprogram used as an explicit transfer of information to or from a subprogram.
Band (Latitudinal band)	Any latitudinal strip, designated by accepted units of linear or angular measurement, which circumscribes the earth.
Bathymetric	Relating to the measurement of ocean depths.
Bathymetric chart	A topographic map of the floor of the ocean.

Bathymetry	The science of determining and interpreting ocean depths and topography.
Bearing	1. (general) The horizontal angle at a given point measured clockwise from a specific reference datum to a second point. Also called bearing angle. 2. (navigational) The horizontal direction of one terrestrial point from another, expressed as the angular distance from a reference direction. It is usually measured from 000° at the reference direction clockwise through 360°. The terms, bearing and azimuth are sometimes used interchangeably, but in navigation the former customarily applies to terrestrial objects and the latter to the direction of a point on the celestial sphere from a point on the earth
Binary	(1) Pertaining to a characteristic or property involving a selection, choice, or condition in which there are two possibilities. (2) pertaining to the numeration system with a radix of two.
Binary Coded Decimal (BCD)	Pertaining to a decimal notation in which the individual decimal digits are each represented by a group of binary digits, e.g., in the 8-4-2-1 binary coded decimal notation, the number 23 is represented as 0010 0011, whereas in binary notation, 23 is represented as 10111.
Block	A set of things, such as words, characters, or digits, handled as a unit.
Block diagram	A diagram of a system, instrument, computer, or program in which selected portions are represented by annotated boxes and interconnecting lines.

Cartesian coordinates	Values representing the location of a point in a plane in relation to two intersecting straight lines, called axes. The point is located by measuring its distance from each axis along a parallel to the other axis. If the axes are perpendicular to each other, the coordinates are rectangular; if not perpendicular, they are oblique coordinates. This system is extended to represent the location of points in three-dimensional space by referencing to three mutually perpendicular coordinate axes which intersect at a common point of origin.
COMMON	Is a specification statement, used during compilation rather than execution as a convenient method for passing values between main program and subprograms without mentioning them as arguments.
COMPASS	Control Data Corporation assembly language for CDC 3000- and 6000-series computers.
DATA	Is a specification statement, used during compilation rather than execution as a convenient method for entering data value into referenced storage areas.
Data	Any representations such as characters or analog quantities to which meaning might be assigned.
Deck	A collection of punched cards.
Dependent variable	A fixed variable given as a function of another variable, i.e., if y is given as a function of x, then, y is the dependent variable.
Digitize	(1) The conversion of graphical analog information or characters into digital form, usually for the purpose of rapid manipulation or storage by a digital computer (2) to express data in a digital form.

Field	In a record, a specified area used for a particular category of data, e.g., a group of card columns used to represent a wage rate or a set of bit locations in a computer word used to express the address of the operand.
File	A collection of related records treated as a unit. Thus in inventory control, one line of an invoice forms an item, a complete invoice forms a record, and the complete set of such records forms a file.
Fixed point	Pertaining to a numeration system in which the position of the point is fixed with respect to one end of the numerals, according to some convention.
Floating point	Pertaining to a numeration system in which the position of the point does not remain fixed with respect to one end of the numerals.
Flowchart	A graphical representation for the definition, analysis, or solution of a problem, in which symbols are used to represent operations, data, flow and equipment.
Geodesic	A line of shortest distance between any two points on any mathematically defined surface. A geodesic line is a line of double curvature, and usually lies between the two normal section lines which the two points determine. If the two terminal points are in nearly the same latitude, the geodesic line may cross one of the normal section lines. It should be noted that, except along the equator and along the meridians, the geodesic line is not a plane curve and cannot be sighted over directly. However, for conventional triangulation the lengths and directions of geodesic lines differ inappreciably from corresponding pairs of normal section lines. Also called geodesic line; geodetic line.

Great circle — A circle on the surface of the earth, the plane of which passes through the center of the earth.

Header card — The first card or cards of a deck of punched cards containing identification of fixed information about the punched cards of variable data that follow.

Independent variable — A variable whose assigned value(s) are arbitrary when defined as a function of another variable, i.e., if y is given as a function of x, then, x is the independent variable.

Input — (1) The data to be processed. (2) The stage or sequence of states occurring on a specified input channel. (3) The device or collective set of devices used for bringing data into another device. (4) A channel for impressing a state on a device or logic element. (5) The process of transferring data from an external storage to an internal storage.

Interpolation — To determine intermediate values between given fixed values. As applied to logical contouring to interpolate is to ratio vertical distances between given spot elevations.

Look-up table — An index file or array(s) which is usually used to access a main record file. It contains the identifier (or file key) and the storage address in sequential or non-sequential order. It may also contain critical information.

MARSDEN chart — A system introduced by Marsden early in the nineteenth century for showing the distribution of meteorological data on a chart; especially over the oceans. A Mercator map projection is used; the world between 90°N and 80°S being divided into Marsden "squares" each of 10° latitude by 10° longitude.

MARSDEN chart (Con.)	These squares are systematically numbered to indicate position. Each square may be divided into quarter squares, or into 100 1° subsquares numbered from 00 to 99 to give the position to the nearest degree.
Mercator projection	A conformal map projection of the cylindrical type. The equator is represented by a straight line true to scale; the geographic meridians are represented by parallel straight lines perpendicular to the line representing the equator; they are spaced according to their distance apart at the equator. The geographic parallels are represented by a second system of straight lines perpendicular to the family of lines representing the meridians and therefore parallel with the equator. Conformability is achieved by mathematical analysis, the spacing of the parallels being increased with increasing distance from the equator to conform with the expanding scale along the parallels resulting from the meridians being represented by parallel lines. Also called equatorial cylindrical orthomorphic map projection.
Merge	To combine two or more sets of items into one, usually in a specified sequence.
Meridional part	The length of the arc of a meridian between the equator and a given parallel on a Mercator chart, expressed in units of one minute of longitude at the equator.
Offline	Pertaining to equipment of devices not under direct control of the central processing unit.
Online	Pertaining to equipment or devices under direct control of the central processing unit.

Output	(1) Data that has been processed. (2) The state or sequence of states occurring on a specified output channel. (3) The device or collective set of devices used for taking data out of a device. (4) A channel for expressing a state of a device or logic element. (5) The process of transferring data from an internal storage to an external storage.
Precision	The degree of discrimination with which a quantity is stated, e.g., a three-digit numeral discriminates among 1,000 possibilities.
Profile	A vertical section of the surface of the ground, or of underlying strata, or both, along any fixed line.
Program element	The smallest field (group) of unique contiguous characters or digits.
Punched cards	(1) A card punched with a pattern of holes to represent data. (2) A card as in (1) before being punched.
Radix	A quantity whose successive integral powers are the implicit multipliers of the sequence of digits that represent a number. For example if the radix is 5, then 143.2 means 1 times 5 to the second power, plus 4 times 5 to the first power, plus 3 times 5 to the zero power, plus 2 times 5 to the minus one power.
Random access	(1) Pertaining to the process of obtaining data from, or placing data into, storage where the time required for such access is independent of the location of the data most recently obtained or placed in storage. (2) Pertaining to a storage device in which the access time is effectively independent of the location of the data.

Real Time	(1) Pertaining to the actual time during which a physical process transpires. (2) Pertaining to the performance of a computation during the actual time that the related physical process transpires in order that results of the computation can be used in guiding the physical process.
Relative address	Identifies a word in a subroutine or array with respect to its position. Relative addresses are translated into absolute addresses by the addition of some specific reference address, usually that at which the first word of the routine or array is stored.
Rhumb line	A line of the surface of the earth making the same angle with all meridians; a loxodrome or loxodromic curve spiraling toward the poles in a constant true direction. Parallels and meridians, which also maintain constant true directions, may be considered special cases of the rhumb line. A rhumb line is a straight line on a Mercator projection. Also called equiangular spiral; loxodrome, loxodromic curve; Mercator track.
Round off	To delete the least significant digit or digits of a numeral and to adjust the part retained in accordance with some rule.
Routine	A set of instructions arranged in proper sequence to cause a computer to perform a desired task.
Selection overlay	A tracing of selected map source detail compiled on transparent material; usually described by the name of the features or details depicted, such as contour overlay, vegetation overlay. Also called lift; pull up; trace.
Storage	(1) Pertaining to a device into which data can be entered, in which it can

Storage (Con.)	be held, and from which it can be retrieved at a later time. (2) Loosely, any device that can store data. (3) Synonymous with Memory.
Synthetic	Produced artifically; devised, arranged, or fabricated for special situations to imitate or replace usual realities.

LIST OF ACRONYMS USED IN COMPUTER PROGRAMS

AMP-	Function used in MINCON, MERFIX and RHUMB to calculate meridional parts for the latitude component.
AXIS-	Calcomp plotter subroutine to automatically scale and draw axes.
BATHY-	Subroutine which determines which quadrant the rhumb line will pass through, extracts the gridded data and calculates the profile for each MSQLOC area.
BURNS-	See SYNCON2R
CALMA 485-	(1) A large bed, graphical analog digitizer manufactured by the CALMA Corporation. (2) A processor program for (1) that initially scales the synthetic track from charts.
CDC-	Control Data Corporation
DATE-	COMPASS off-line subroutine which automatically calculates an integer day, month, year from the computer's interval clock.
DAWHAT-	See SYNCHEX
DKLOCATE-	Subroutine which positions read/write head at specified relative address.
DKOPEN-	Subroutine which opens disk file.
DKREAD-	Subroutine which reads blocks of data from the disk file in groups of 32 words or larger.
DKWRITE-	Subroutine which writes blocks of data on to disk file in groups of 32 words or larger.
GCDIST-	Subroutine used by SEAARCH for direct solution of the great circle.

Preceding page blank

GCPATH-	Subroutine used by SEAARCH for indirect solution of the great circle.
GEODIST-	Subroutine for the direct solution of the geodesic.
GEOPATH-	Subroutine for the indirect solution of the geodesic.
GRIDBLK-	Subroutine which calls in the gridded data from the random access storage device for BATHY.
LOOKUP-	Subroutine which "looks up" or extracts the relative address, block size and the column and row information for each MSQLOC area from the random access storage device previous to passing this information to BATHY.
LINE-	Calcomp plotter subroutine to automatically draw a line as a function of x and y.
MERFIX-	Subroutine which calculates the rhumb line distance and sets up a scaling factor for nautical miles along a profile.
MINCON-	Subroutine used to calculate the start point for a profile within a MSQLOC area.
MSQFQ-	Subroutine used to calculate in part the MSQLOC area numbers for points on the profile path.
MSQLOC-	Marsden Square Locator Number (Marsden square system is a numbered, 10 degree rectangular grid of the world which is subdivided further into 5 and 1 degree squares).
PLOT-	Calcomp plotter subroutine which moves pen in x and y direction.
PLOTS-	Calcomp plotter subroutine which initiates plotter action.
PUNOUT-	Subroutine which places each MSQLOC area profile on magnetic tape.

RHUMB-	Subroutine using AMP to compute the rhumb line (approximation of a chord of a great circle on a Mercator projection) bearing through an MSQLOC area.
SEAARCH-	Subroutine used to generate a great-circle path.
SPLICON-	Subroutine used by SPLINE for cubic spline calculations.
SPLINE-	Subroutine for the cubic spline algorithm.
SPLINT-	See SYNGRID
SYMBOL-	Calcomp plotter subroutine which plots alphanumeric characters and symbols.
SYNBAPS-	Synthetic Bathymetric Profiling System.
SYNBAPS1-	Accessing program which produces a depth range profile on magnetic tape for each MSQLOC area.
SYNBLOCK-	Program which loads gridded bathymetric data into the random access storage device.
SYNCARD-	Program which checks longitude of data points and depth values.
SYNCHEX-	Program which track plots data points on a Mercator projection at the scale of the source manuscript.
SYNCON2R-	Program which plots contours of the gridded data on a Mercator projection at the scale of the source manuscript.
SYNGRID-	Program which transforms synthetic track line data into gridded bathymetric data at seven points per card. This is the primary structuring program.
SYNPLOT-	Accessing program which links together the profiles on magnetic tape produced by SYNBAPS1 for each MSQLOC area to plot a great circle profile. This program is usually run linked to SYNBAPS1.

SYNSTAT- Status program which queries random access storage device for listing of file key, relative address, block size, number of rows and columns and date that data were added to storage and/or actual gridded data.

SYNTABLE- Traffic director program which supplies relative address, block size and file key to SYNBLOCK for the accurate placement of blocks of gridded bathymetric data on the random access storage device.

SYNTRACK- Program which outputs header, track, data and blank cards and conducts error checks. Input is a scaled data tape from the CALMA 485 processor program.

TIMELEFT- COMPASS on-line function which extends time mark from computer's interval clock.

UTILITY- Systems program which loads gridded bathymetric data cards on mangetic tape.

APPENDIX A

Preparation of Charts for Digitization

The 5-degree square unit, around which the data base is created, has been explained in the "Outline of the System" and in figure 3, 4, 5, and 6. Paper copies of the contour charts, which are on a Mercator projection, are used to prepare the basic manuscripts for digitizing. Sufficient overlap around each 5-degree square is required to provide 5 minutes on all sides for the MSQLOC area and an additional 5 minutes on all sides for interpolation of the track input data (fig. 6): The manuscript size is then at least 320 minutes by 320 minutes regardless of the chart scale. Ideally, the manuscript should consist of one easy-to-handle document. However, because chart formats vary, this is not always possible. A case in point is the addition of large scale survey of a newly discovered seamount to a regional chart.

One method of handling this is to digitize the two charts separately, then, substitute the synthetic tracks from the new seamount chart for those in the corresponding section of the older regional chart. A second method is to prepare a contour selection overlay for the seamount chart, photographically reduce it to the scale of the regional chart, make a print at that scale, attach the print to the regional chart and match the contours. This method also can be used with transparent media.

The smallest cell selected for SYNBAPS is a 5-minute (meridional part) square with a depth value at the four corner intersections. The synthetic tracks of input depth points are usually taken at a 5-minute spacing on a Mercator projection. In high frequency data areas, additional tracks of data at 1-, 2-, 3-, or 4-minute spacing can be input so as to improve the four cell depth values. However, there is a limit to how much improvement can be made without losing some of the high-frequency detail. One improvement would use a smaller cell size, but this makes random-access storage device data storage requirements very large. Thus, small features that fall within a 5-degree cell can be lost to the data base, especially if they are not picked up at the input or structuring phase.

It is necessary to interpolate the beginning and end points for each track in the overlap areas. This is not a requirement for short tracks within the body of the MSQLOC area. These points may be visually interpolated by the analyst or by an experienced digitizer operator. This interpolation need only be to the nearest 20 fathoms or about one-tenth the contour interval.

The output from the SYNCON2R program is a contour plot of the MSQLOC area. Although this output is not a primary product of the system, it is used for checking and may be a useful byproduct as rough automated contours. Because of the 5-minute cell size and the nature of the interpolation scheme, large flat areas tend to break up on the contour plot. This break up of contours is not an error in the data and does not affect the profile generation. To improve the contour output aesthetically, the interpolation can be improved by adding contours in key locations. In areas of rough topography this improvement will not be necessary. The first example, around seamounts or a seamount group, is shown in figure A-1. Usually the added contour is placed outside the base contour to cutoff or terminate the interpolation adjoining a flat area or to define the seamount base. The second example is for domes, rises, ridges or tablemounts (fig. A-2). Here the added contours are on the top of the structure in order to cutoff or terminate the interpolation on their flat or gently rounded summits. The third example is for noses or spurs (fig. A-3). Although this feature is similar to those in figures A-1 and A-2, short disconnected contours may be needed if the spur slopes are gentle. In all these examples, the track direction was assumed to be left to right.

The boundary condition is a special case of endpoint interpolation. Whenever an island or continent is encountered, the zero contour or sea level is handled as shown in figure A-4. On the SYNCON2R program the zero-contour level should never be plotted, but the 1-fathom or 1-meter contour should be interpolated to show the coast line. In profiling, the punched card depth values after the first zero usually are discarded and the profile terminated at that range.

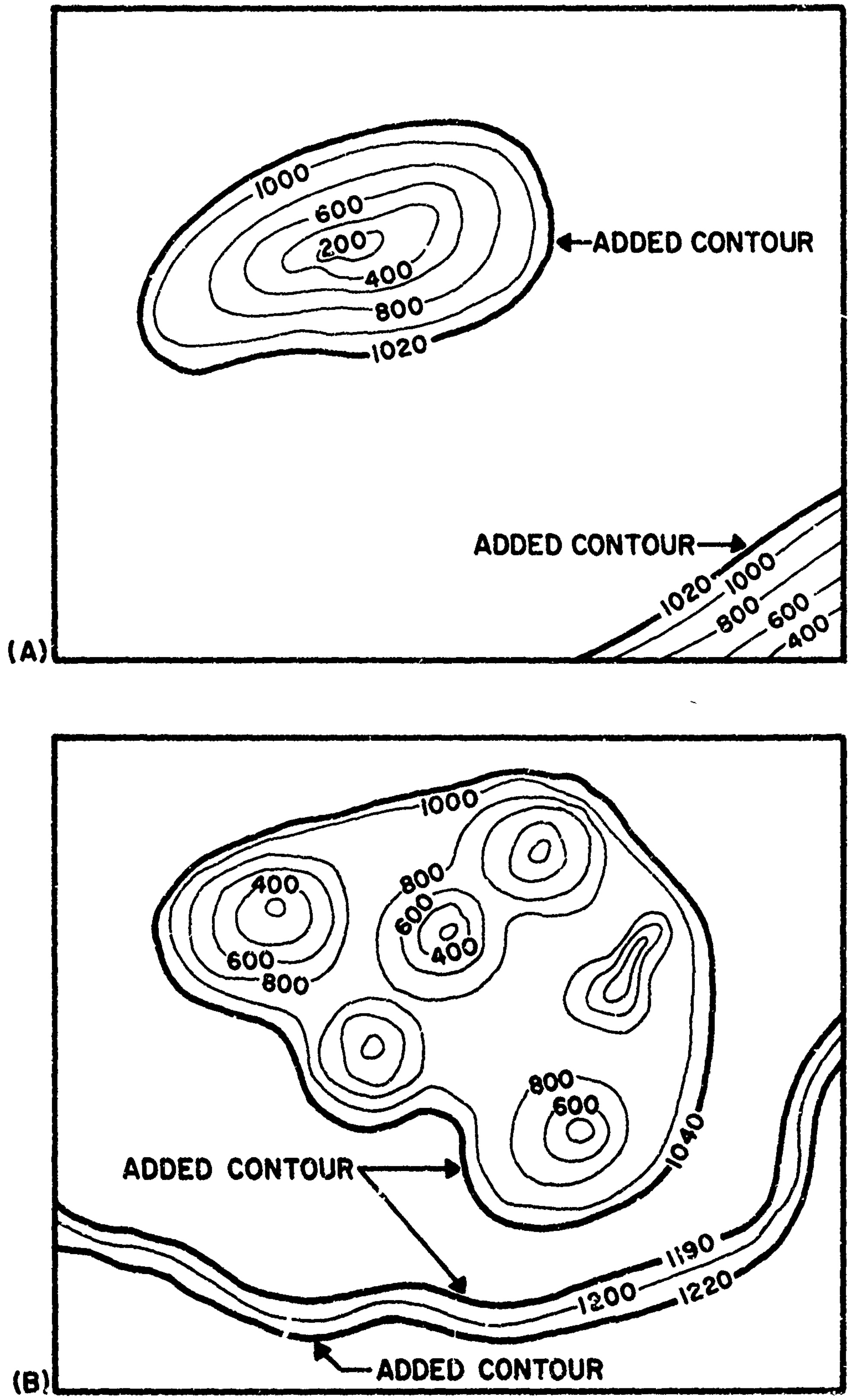


FIGURE A-1. ADDED CONTOURS AROUND SEAMOUNTS OR SEAMOUNT GROUP

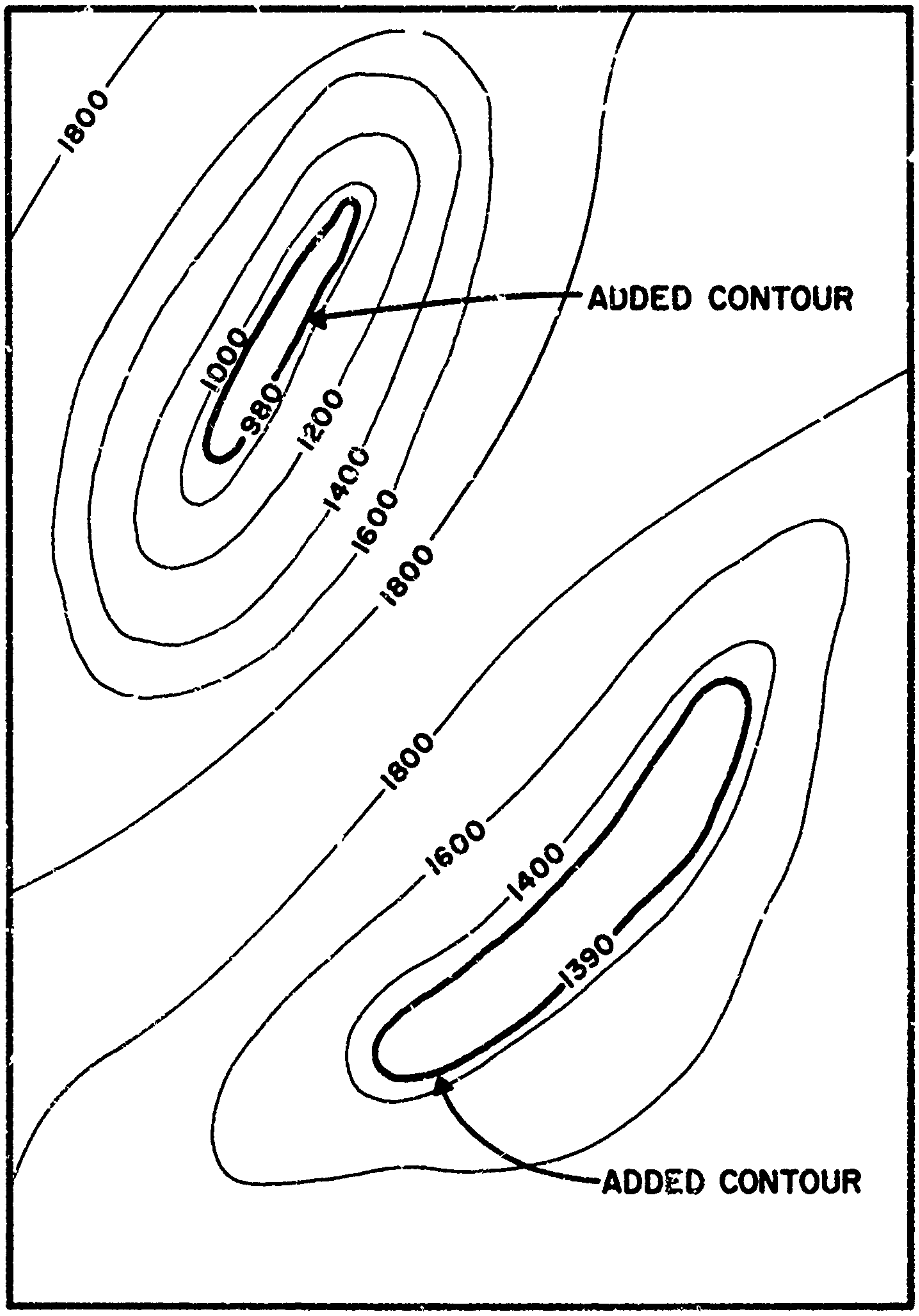


FIGURE A-2. ADDED CONTOURS ON DOMES OR RISES

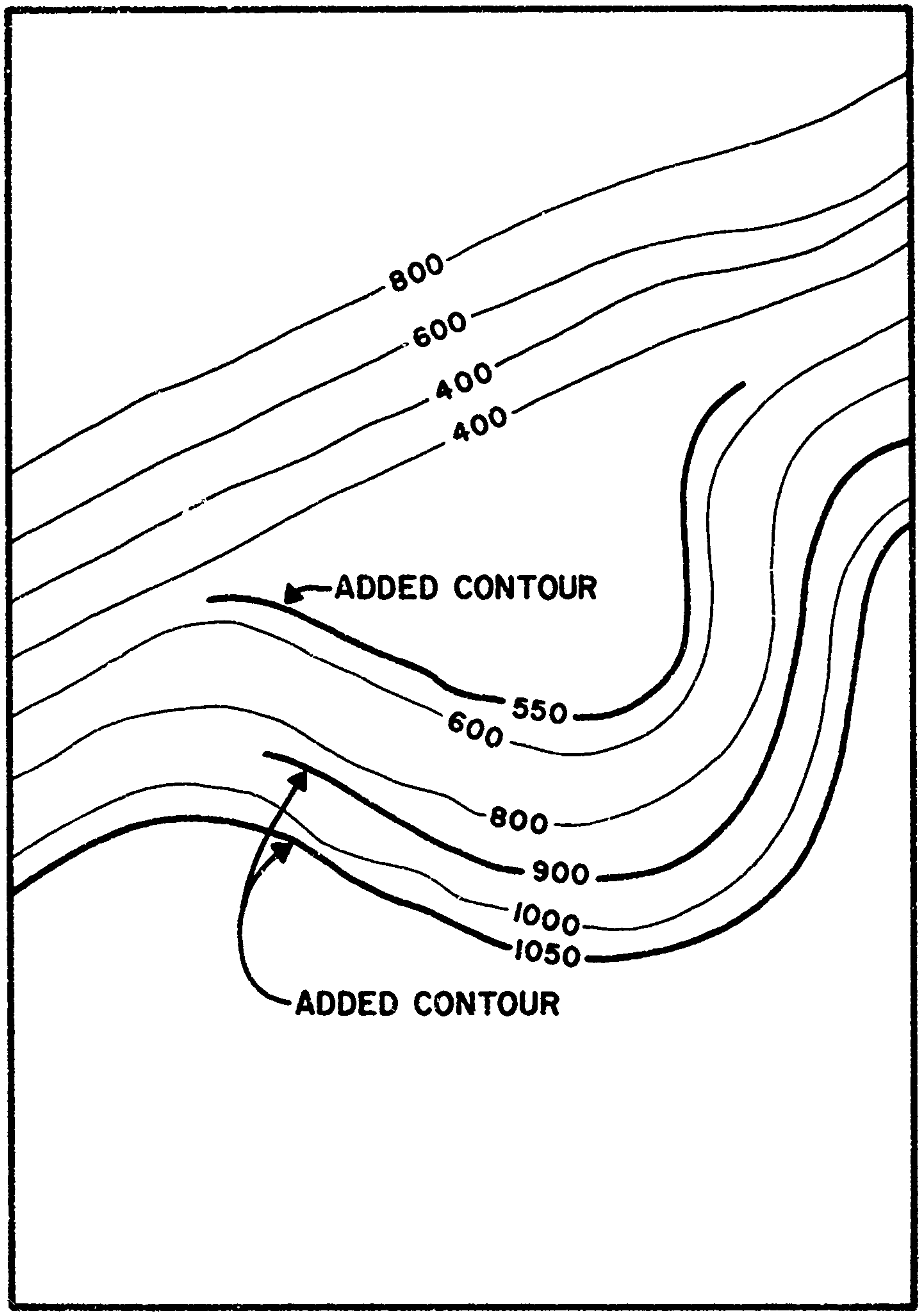


FIGURE A-3. ADDED CONTOURS AROUND A SPUR

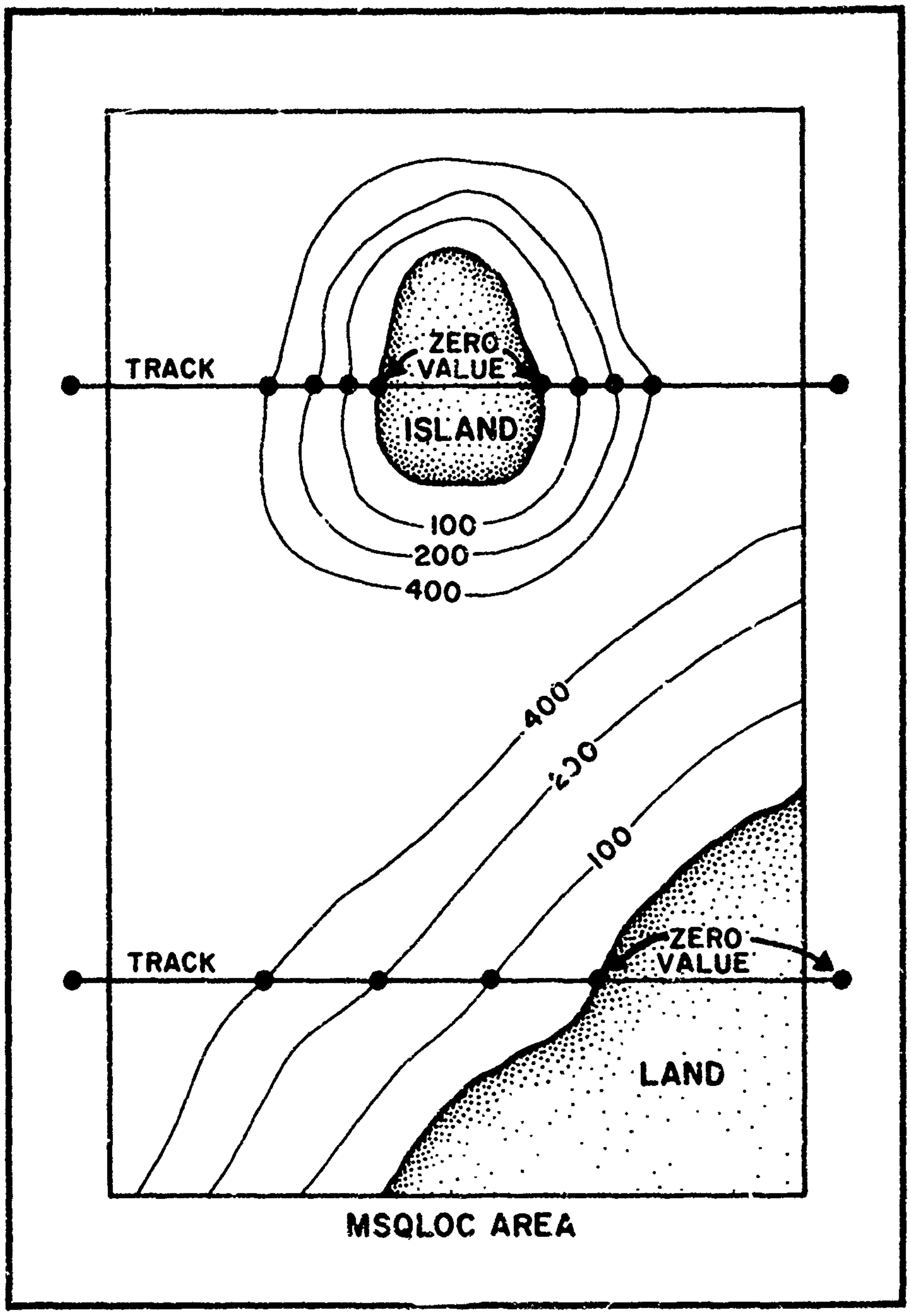


FIGURE A-4. BOUNDARY CONDITIONS FOR ZERO CONTOUR LEVEL

APPENDIX B

FORTRAN Programs for

Structuring SYNBAPS

All programs and subroutines listed in this appendix are subject to change without notice. Modifications within the programs and adoption of the system for other computers will necessitate major changes. The author should be contacted for the most recent versions of these programs.

```
      PROGRAM SYNTRACK
C
C                                   A SYNBAPS PROGRAM
C                                   *******************
C
C A PROGRAM TO PROCESS CALMA 485 DIGITIZER BINARY DATA TAPES OF
C  SYNTHETIC BATHYMETRIC TRACK LINE DATA FOR CARD INPUT TO SYNCARD.
C PROGRAM PAIRS-UP ALTERNATING RECORDS OF BCD (DEPTH) AND BINARY (LAT. AND LONG.
C ) INTO TRACK LINES BASED ON A LEFT TO RIGHT ORIENTATION WITHIN EACH MSQLOC
C AREA.
C
C REQUIRES SUBROUTINE LALOCON
C
C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
C
      DIMENSION LAT(6500),LONG(6500),IDEPTH(6500),MSQ5(2)
      TYPE INTEGER STTRK,DEPTH
      TYPE REAL LAT,LONG
C- REWIND 32  IF NOT ALREADY REWOUND.
      REWIND 32
C READ IN TRACK START POINT, STTRK=1 IF START OF NEW BLOCK
  200 READ 300,STTRK
      IF(EOF,60)1000,301
  300 FORMAT(I3)
C READ IN HEADER INFORMATION
  301 READ(32)N,NN,MSQ5(1),MSQ5(2)
      IF(IOCHECK,32)1,1
    1 DECODE(16,2,MSQ5(1)) MSQLOC, NAME
    2 FORMAT(4X,A4,A8)
      PUNCH 3, MSQLOC
    3 FORMAT(A4)
      J=1
C READ IN LAT-LONG AND DEPTH POINTS - CHECK FOR MISSING DATA
      DO 98 K=1,6500
      LAT(K)= 99.0
```

```
      LONG(K)= 999.0
      IDEPTH(K)=4H9999
      READ(32)N,NN,RLONG,RLAT
      IF(IOCHECK,32)4,4
    4 IF(EOF,32)5,6
    5 K=K-1 $ GO TO 9
    6 IF(N+NN-4) 77,10,77
   77 READ(32) KD,KKD
      GO TO 99
   10 READ(32)ND,NND,DEPTH
      IF(IOCHECK,32)7,7
    7 IF(EOF,32)5,13
   13 IF(ND+NND-1) 99,12,99
   12 DECODE(8,8,DEPTH) IDEPTH(K)
    8 FORMAT(A4,4X)
      LONG(K)=RLONG
      LAT(K)=RLAT
      GO TO 98
   99 CALL ALOCON(LAT(K-1),LONG(K-1),K,FLAT,FLON,FLATM,FLONM,NORT,IEST)
      PRINT 102, MSQLOC
  102 FORMAT(* ERROR IN MSQLOC *A4)
      PRINT 11,K,FLAT,FLATM,NORT,FLON,FLONM,IEST,IDEPTH(K-1)
   11 FORMAT(* ERROR++++ONE OR MORE POSITIONS AND/OR DEPTHS ARE MISSING
     1FROM TRACK AT LOCATION *I4//10X*MISSING POINT FOLLOWS POINT AT LAT
     2ITUDE  =*2F4.0,2X,A1/41X*LONGITUDE =*2F4.0,2X,A1/41X*DEPTH     = *
     3A4/* PROGRAM RUN WILL CONTINUE*/)
      J=J+1
   98 CONTINUE
    9 IF(J .GT. 1)14,15
   15 PRINT 16,MSQLOC
   16 FORMAT(1H1* INPUT OF SYNTHETIC TRACK DATA WAS ERROR FREE FOR MSQLO
     1C *A4*++++ PROGRAM RUN WILL CONTINUE*/)
   14 IF(STTRK .EQ. 1)17,18
   17 ITRACK=1
      GO TO 19
   18 ITRACK=STTRK
   19 L=M=1
```

```
C  FOR TESTING ONLY REMOVE FOR PRODUCTION
      PRINT 500,(KK,LAT(KK),LONG(KK),IDEPTH(KK),KK=1,K)
  500 FORMAT(1X,3(I5,2F10.5,2X,A4,10X))
C TEST FOR CHANGING TRACKS AND OUTPUT BY TRACKS
      DO 97  J=1,K
C FLIP (J+1) AND(J) FOR WEST LONGITUDE QUADRANT
      IF(AINT(LONG(J+1)).LT.AINT(LONG(J)).AND. LAT(J).NE. 99.)20,21
   20 PUNCH 300,ITRACK
      PRINT 25, ITRACK, MSQLOC
   25 FORMAT(1H ,10X,*THE FOLLOWING POINTS ARE FOR TRACK NUMBER *I3* OF
     1MSQLOC *A4/)
      ITRACK=ITRACK+1
   27 DO 96 N=M,L
      CALL LALOCON(LAT(N),LONG(N),N,FLAT,FLON,FLATM,FLONM,NORT,IEST)
      PUNCH 23,FLAT,FLATM,NORT,FLON,FLONM,IEST,IDEPTH(N)
   23 FORMAT(2(2F4.0,1X,A1),6X,A4)
  600 PRINT 33,N,FLAT,FLATM,NORT,FLON,FLONM,IEST,IDEPTH(N)
   33 FORMAT(1X,I6*,*2X,(2(2F4.0,1X,A1),6X,A4));
   96 CONTINUE
      M=L+1
      L=L+1
      PUNCH 24
   24 FORMAT(35X*BLANK*)
      GO TO 97
   21 L=L+1
C  IF J EQUALS K-1  PUNCH OUT LAST TRACK
      IF(J.EQ. K-1) 20,97
   97 CONTINUE
      PRINT 100,NAME,MSQLOC
  100 FORMAT(//* THE PRECEDING POSITIONS AND DEPTHS WERE DIGITIZED BY *A
     1A* FOR MSQLOC = *A4)
      PRINT 101
  101 FORMAT(1H1)
      GO TO 200
 1000 PRINT 103
  103 FORMAT(* ******** END OF TRACKING ROUTINE - SUBMIT CORRECTED CARDS
```

```
      ) TO SYNCARD*********)
       STOP
       END


       SUBROUTINE LALOCON(FINLAT,FINLON,IDBM,FLAT,FLON,FLATM,FLONM,NORT,
      1IEST)
C
C                                         A SYNBAPS SUBROUTINE
C                                        **********************
C
C A ROUTINE TO CONVERT INTERNAL LAT AND LONG TO DEGREEES,MINUTES AND HEMI-
C SPHERE FOR PRINTER OUTPUT.  ERROR MESSAGE VARIES WITH APPLICATION
C
C NORTH AND EAST ARE POSITIVE
C SOUTH AND WEST ARE NEGATIVE
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
       DIMENSION FINLAT(1),FINLON(1)
       FLAT= ABSF(AINT(FINLAT))
       FLON= ABSF(AINT(FINLON))
       FLATM=AINT(((ABSF(FINLAT)-FLAT)*60.)+.5)
       FLONM=AINT(((ABSF(FINLON)-FLON)*60.)+.5)
       IF(FLATM-60.) 11,10,11
    10 FLAT=FLAT+1.0
       FLATM=0.0
    11 IF(FLONM-60.) 13,12,13
    12 FLON=FLON+1.0
       FLONM=0.0
    13 IF(FINLAT)100,101,102
   100 NORT=1HS
       GO TO 104
   102 NORT=1HN
       GO TO 104
   101 PRINT 103,IDBM
```

```
103 FORMAT(* POSSIBLE ERROR IN POSITION CONVERSION OR POSITION FALLS O
   1N EQUATOR OR PRIME MERIDIAN AT POINT *I4/* PROGRAM RUN WILL CONTIN
   2UE *)
    NORT=1HN
    IF(FINLON.NE. 0.0) GO TO 104
    IEST=1HE
    RETURN
104 IF(FINLON)105,101,107
105 IEST=1HW
    RETURN
107 IEST=1HE
    RETURN
    END


    PROGRAM SYNCARD
C                              A SYNBAPS PROGRAM
C                              *******************
C
C  A PROGRAM USED TO CHECK CARD DECKS FOR DEPTH AND LONGITUDE AS OUTPUT FROM
C   SYNTRACK BEFORE ENTRY TO SYNCHEX.
C
C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7605
C
     DIMENSION LAT(325),LONG(325),LATM(325),LONGM(325),DEPTH(325)
     TYPE INTEGER DEPTH, TRKCNT, TRK, FLAG
     DATA(FLAG=400)
 200 READ 1,TRKCNT
   1 FORMAT(I3)
     IF(EOF,60) 1000,100
 100 READ 2, MSQLOC
   2 FORMAT(A4)
     NN=0
     PRINT 20, MSQLOC
  20 FORMAT(1H1,35X*THE FOLLOWING TRACKS ARE FOR MSQLOC *A4/)
```

```
      DO 99 L=1,TRKCNT
      READ 1, TRK
      DO 98 I=1,325
      READ 3,LAT(I),LATM(I),LONG(I),LONGM(I),DEPTH(I),IFLAG
    3 FORMAT(2(2I4,2X),6X,I4,5X,A5)
      II=I
      IF(IFLAG.EQ. 5HBLANK) 4,98
   98 CONTINUE
    4 II=II-1
      DO 97 J=1,II
      K=0
      IF(J.EQ. 1)27,5
    5 IF(DEPTH(J).GE.(DEPTH(J-1)+FLAG).OR.DEPTH(J).LE.(DEPTH(J-1)-FLAG))
     16,7
    7 IF(LONG(J).EQ.LONG(J-1).AND.LONGM(J).EQ.LONGM(J-1))9,8
    8 K=K+1   $ GO TO 30
    9 K=K+2   $ GO TO 30
    6 K=K+1
      IF(LONG(J).EQ.LONG(J-1).AND.LONGM(J).EQ.LONGM(J-1)) 11,10
   10 K=K+2   $ GO TO 30
   11 K=K+3   $ GO TO 30
   27 K=K+1
      PRINT 28,TRK
   28 FORMAT(1X,*THE FOLLOWING POSITIONS AND DEPTHS ARE FOR TRACK NUMBER
     1*I3/)
   30 GO TO (12,14,16,18)K
   12 PRINT 13,J,LAT(J),LATM(J),LONG(J),LONGM(J),DEPTH(J)
   13 FORMAT(1X,I3,2X,4I5,I6)
      GO TO 97
   14 PRINT 15,J,LAT(J),LATM(J),LONG(J),LONGM(J),DEPTH(J)
   15 FORMAT(1X,I3,2X,4I5,I6,2X*ERROR IN LONGITUDE ONLY*)
      GO TO 197
   16 PRINT 17,J,LAT(J),LATM(J),LONG(J),LONGM(J),DEPTH(J)
   17 FORMAT(1X,I3,2X,4I5,I6,2X*ERROR IN DEPTH ONLY*)
      GO TO 197
```

```
   18 PRINT 19,J,LAT(J),LATM(J),LONG(J),LONGM(J),DEPTH(J)
   19 FORMAT(1X,I3,2X,4I5,I6,2X*ERROR IN BOTH LONGITUDE AND DEPTH*)
  197 NN=NN+1
   97 CONTINUE
      PRINT 300
  300 FORMAT(5X*BLANK CARD*/)
   99 CONTINUE
      PRINT 2000,NN
 2000 FORMAT(* THERE ARE*I4* ERRORS IN THE PRECEDING MSQLOC BLOCK - CORR
     1ECT THESE BEFORE ENTRY TO SYNCHEX PROGRAM*)
      GO TO 200
 1000 PRINT 1001
 1001 FORMAT(1H1,* ***** END OF CHECKING ROUTINE - SUBMIT CORRECTED CARD
     1S TO SYNCHEX *****)
      STOP
      END


      PROGRAM SYNCHEX
C
C
C                                        A SYNBAPS PROGRAM
C                                        *******************
C
C PROGRAM TO PLOT X Y COORDINATES OF TRACKS OBTAINED FROM DIGITIZER
C NLINE IS TOTAL NUMBER OF TRACKS
C XIN AND YIN ARE DIMENSIONS OF PLOT (LARGE PLOTTER) IN INCHES
C
C PROGRAM WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION X(2000),Y(2000),Z(2000),A(2000),XA(2000),YA(2000)
      CALL PLOTS(A,2000,3,29)
      READ 1,NLINE,XMIN,XMAX,YMIN,YMAX,XIN,YIN
    1 FORMAT(I10,6F10.0)
      S=YIN/(YMAX-YMIN)
      R=XIN/(XMAX-XMIN)
```

```
      ORIGIN=(29.0-YIN)/2.0
      CALL PLOT(0.0,ORIGIN,-3)
      CALL PLOT((XMAX-XMIN)*R,0.0,2)
      CALL PLOT((XMAX-XMIN)*R,YIN,2)
      CALL PLOT(0.0,YIN,2)
      CALL PLOT(0.0,0.0,2)
   60 READ(60,33) ILINE
   33 FORMAT(I3)
      J=0
      PRINT 47,ILINE,J
   47 FORMAT(2I10)
   40 J=J+1
      READ (60,34) Y(J),YA(J),X(J),XA(J),Z(J)
   34 FORMAT(2(F4.0,F4.0,2X),F8.0)
      Y(J) = (YA(J)+Y(J)*60.)*.00029089
      X(J) = (XA(J)+X(J)*60.)
      AP = 0.785398
      TEMP = SIN(AP+Y(J)/2.)/COS(AP+Y(J)/2.)
      Y(J) =  7915.7045*ALOG10(TEMP)-23.268932*SIN(Y(J))
      X(J)=(X(J)-XMIN)*R
      Y(J)=(Y(J)-YMIN)*S
       PRINT 52, X(J),Y(J),Z(J),XMIN,YMIN
   52  FORMAT(5F10.2)
      IF(Z(J))38,39,38
   38 IF(J-1)42,41,42
   41  CALL SYMBOL(X(J),Y(J),.04,3,0.0,-1)
       CALL NUMBER(X(J),Y(J),.04,Z(J),0.,4HF3.0)
      CALL PLOT(X(J),Y(J),3)
      GO TO 40
   42  CALL SYMBOL(X(J),Y(J),.04,3,0.0,-2)
       CALL NUMBER(X(J),Y(J),.04,Z(J),0.,4HF3.0)
      CALL PLOT(X(J),Y(J),3)
      GO TO 40
   39 CALL NUMBER (X(J-1),Y(J-1),.08,ILINE,0.,2HI3)
      IF(ILINE-NLINE)60,69,69
   69 PRINT 100
  100 FORMAT(1H0,*    NORMAL EXIT*)
```

```
      CALL STOP PLOT
      STOP
      END



      PROGRAM SYNGRID
C
C                               A SYNBAPS PROGRAM
C                               *******************
C
C     PROGRAM TO GRID   RANDOM TRACK TYPE SURVEY DATA WITH OPTION TO
C     COMPUTE MEAN ANOMALIES, GRIDDING METHOD IS ONE DIMENSIONAL CUBIC
C     SPLINES. REF.BHATTACHARYYA, GEOPHYSICS V34,NO.3,JUNE 69. MEAN
C     ANOMALIES COMPUTED BY AVERAGING 9 PTS. IN EACH GRID INTERVAL
C     POSITION OF MEAN FOR EACH GRID CELL IS THE UPPER LEFT HAND
C      CORNER OF THE CELL
C     NOTE-INPUT DATA IS ASSUMED TO BE ERROR FREE. ALAT,ALONG=LOCATION
C     OF ORIGIN IN DEGREES AND TENTHS,SET AT LOWER LEFT CORNER,GRID=
C     DESIRED GRID INTERVAL IN MINUTES,PLAT,PLONG= COORDINATES(DEGREES
C     AND TENTHS) OF UPPER RIGHT HAND CORNER, MEAN=BLANK-NO MEAN ANOMALY
C     COMPUTED,=1-MEAN COMPUTED AT CENTER OF EACH GRID,NTRK=TRACK NUMBER
C     FROM 1 TO ITOT,LAST CARD IN EACH TRACK IS BLANK,1ST.CARD IS NTRK
C     CARD,NUMBERTRACKS IN ANY ORDER ACROSS AREA,PROGRAM REQUIRES
C     SPLINE, SPLICON AND SORTY SUBROUTINES
C     LAT,LONG OF INPUT DATA IS IN DEG, MIN, SEC(FORMAT 70) OR IN
C     DECIMAL X,Y (FORMAT 90)
C     IF ITYPE =BLANK GRID WILL BE EQUAL INTERVALS OF LAT AND LONG,IF
C     ITYPE =1 GRID WILL BE IN MERCATOR PROJECTION
C     ITYPE= -1 GRID, ORIGIN AND LIMITS OF INPUT DATA ARE IN X,Y UNITS
C     WITH X= LONG AND Y= LAT, PIVOT= MAX DISTANCE(X-Y UNITS) FROM
C     TRACK FOR PIVOT TEST
C
C     REF. FOR CUBIC SPLINE IS PENNINGTON,R.H. INTRODUCTORY COMPUTER
C     METHODS AND NUMERICAL ANALYSIS,MACMILLAN,1965
```

```
C
C     EXCEPT FOR ICT,VARIABLES WITH 1 DIMENSION SHOULD BE DIMENSIONED
C     TO THE MAX NO.OF POINTS ON ANY LINE OR THE MAX NO.OF GRID
C     INTERVALS IN THE LONGEST DIMENSION OF THE AREA WHICHEVER IS
C     GREATER. IN THE MAIN PROGRAM,VARIABLES WITH 2 DIMENSIONS
C     HAVE THE 2ND DIMENSION = TO ITOT AND THE 1ST DIMENSION = TO THE
C     MAX NO.OF GRID INTERVALS IN THE LONGEST DIMENSION OF THE AREA.
C     THE LARGEST DIMENSION OF 2 DIMENSIONAL VARIABLES IN THE
C     SUBROUTINES SHOULD BE AT LEAST EQUAL TO THE SIZE OF THE 1
C     DIMENSIONAL VARIABLES IN THE MAIN PROGRAM.
C
C PROGRAM WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION X(82),Y(82),Z(82),DIST(82),AX(82),AY(82),BX(82,90),
     1BY(82,90),BZ(82,90),ICT(90),AZ(82),AVE(82)
C     READ IN CONTROL CARD AND CONVERT ORIGIN AND LIMIT TO X-Y
      READ(60,201) ISETS
  201 FORMAT(I5)
      DO 67 LL=1,ISETS
      READ (60,10),ALAT,ALONG,PLAT,PLONG,GRID,MEAN,ITOT,ITYPE,PIVOT,
     1MSQLOC
   10  FORMAT(5F10.0,I1,I2,I2,F5.0,A4)
      PUNCH 202, MSQLOC
  202 FORMAT(A4)
      WRITE(61,203) ALAT,ALONG,PLAT,PLONG,GRID,MEAN,ITOT,ITYPE,PIVOT,
     1MSQLOC
  203 FORMAT(* CONTROL CARD *5F10.3,I1,2I2,F5.2,A4///)
      WRITE(61,200) MSQLOC
  200 FORMAT(20X,*OUTPUT- FOR FIVE DEGREE SQUARE NUMBER  *A4/)
      IF(ITYPE)91,71,72
   71 BLAT= ALAT *60.0
      BLONG=ALONG *60.0
      RLAT=(PLAT*60.0)-BLAT
      RLONG=BLONG -(PLONG*60.0)
      RLONG= -1.0*RLONG
C     CHANGE APPROPRIATE SIGN IF AREA IS IN WEST LONG OR SOUTH LAT
      GO TO 31
```

```
91 BLAT=ALAT
   BLONG=ALONG
   RLAT=PLAT-BLAT
   RLONG=PLONG-BLONG
   GO TO 31
72 BLAT=ALAT *.0174533
   AP=0.785398
   TEMP=SIN(AP+BLAT/2.0)/COS(AP+BLAT/2.0)
   BLAT=7915.7045*ALOG10(TEMP)-23.268932*SIN(BLAT)
   BLONG=ALONG*60.0
   RLAT=PLAT*.0174533
   TEMP=SIN(AP+RLAT/2.0)/ COS(AP+RLAT/2.0)
   RLAT=(7915.7045*ALOG10(TEMP)-23.268932*SIN(RLAT))-BLAT
   RLONG=BLONG-PLONG*60.0
   RLONG= -1.0*RLONG
C     READ TRACK NO.AND DATA .CONVERT TO X-Y
31  READ(60,20) NTRK
20 FORMAT(I3)
   ATER= 9999.99
   I=1
99 IF (ITYPE) 92,3,3
92  READ(60,90)  HLONG,HLAT,Z(I)
90  FORMAT(3F20.0)
    HLAT= HLAT* .00029089
   GO TO 93
 3 FLAT=0.0
   FLONG=0.0
   READ(60,70)DLAT,ELAT,DLONG,ELONG,Z(I)
70 FORMAT(2(F4.0,F4.0,2X),F10.0)
   HLAT=((FLAT/60.0)+ELAT+DLAT*60.0)*.00029089
   HLONG=(FLONG/60.0)+ELONG+DLONG*60.0
C     CHECK IF LAST CARD THIS TRACK
93 IF (HLAT+HLONG) 2,4,2
 2  X(I)=BLONG-HLONG
   X(I)= -1.0* X(I)
   IF (ITYPE) 94,95,95
94 X(I)= -1.0*X(I)
```

```
   95 IF(ITYPE) 73,73,75
   73 Y(I)=(HLAT/.00029089)-BLAT
      GO TO 74
   75 TEMP=SIN(AP+HLAT/2.0)/ COS(AP+HLAT/2.0)
      Y(I)=(7915.7045*ALOG10(TEMP)-23.268932*SIN(HLAT))-BLAT
   74 I=I+1
      GO TO 99
C     NOTE PROGRAM ASSUMES LONGITUDE IS EAST,CHANGE STATEMENT 2 IF
C     DESIRED ,NOW FIT LEAST SQUARES LINE TO POSITIONS
    4  N=I-1
C      CARDS FROM HERE TO 704 FOR USOC DATA BASE ONLY
       NB=N-1
       KC=1
       DO 701 KA=1,NB
       KB=KA+1
       ITEMP=(X(KB)-X(KA))/(1.0*GRID)
       IF(ITEMP.EQ.0) GO TO 702
       DTEMP=ITEMP+1
       AX(KC)=X(KA)
       AZ(KC)=Z(KA)
       AY(KC)=Y(KA)
       DEL1=(X(KB)-X(KA))/DTEMP
       DEL2=(Z(KB)-Z(KA))/DTEMP
        DEL3=(Y(KB)-Y(KA))/DTEMP
       DO 703 JC=1,ITEMP
       KD=KC+JC
       AC=JC
       AX(KD)=X(KA)+DEL1*AC
       AZ(KD)=Z(KA)+DEL2*AC
  703   AY (KD)=Y(KA)+DEL3*AC
       KC=KD+1
       GO TO 701
  702   AX(KC)=X(KA)
        AY(KC)=Y(KA)
        AZ(KC)=Z(KA)
       KC=KC+1
  701   CONTINUE
```

```
      DO 7 I=1,N
      IF (DIST(I)- PIVOT) 8,8,7
    7 CONTINUE
    8 A3= -1.0/A2
      AX(I)=(A1+A3*X(I)-Y(I))/(A3-A2)
      AX(KC)=X(KB)
      AY(KC)=Y(KB)
      AZ(KC)=Z(KB)
      N=KC
      DO 704 J=1,N
      X(J)=AX(J)
      Y(J)=AY(J)
  704 Z(J)=AZ(J)
      AN=N
      A=0.0
      B=0.0
      C=0.0
      D=0.0
      DO 5 I=1,N
      A=A+X(I)
      B=B+X(I)**2
      C=C+Y(I)
    5 D=D+Y(I)*X(I)
      A1=(C*B-D*A)/(AN*B-A**2)
      A2=(C-A1*AN)/A
      IF(ABS(A2).LT.0.000001 ) A2 = A2 + 0.00001
      WRITE (61,101) NTRK,A1,A2
  101  FORMAT(* TRACK NO *,I3,* TRACK LINE IS Y=*,F8.2,*+*,F8.2,*X*)
C     LEAST SQUARES LINE IS Y=A1+A2X
C     NOW SEARCH FOR A POINT LESS THAN PIVOT DISTANCE   FROM TRACK LINE
C     TO USE FOR 1ST PIVOT AND MAP PTS.ONTO TRACK WITH CORRECT Z VALUF
      DO 6 I=1,N
    6 DIST(I)=ABS((Y(I)-A2*X(I)-A1)/ (SQRT(A2**2+1.0)))
      AY(I)= A2*AX(I) +A1
      IA=I+1
C     NOW WORK BACKWARDS ON TRACK TO PICK UP POINTS THAT FAILED
C     PIVOT TEST
```

```
 11 J=I-1
    IF(J)12,12,9
  9 DELZ=(Z(J)-Z(I))/SQRT((X(I)-X(J))**2 +(Y(I)-Y(J))**2)
    AX(J)=(A1+A3*X(J)-Y(J))/(A3-A2)
    AY(J)=A2*AX(J)+A1
    Z(J)=(DELZ *SQRT((AX(I)-AX(J))**2+(AY(I)-AY(J))**2))+Z(I)
    I=J
    GO TO 11
C      NOW WORK FORWARD ON TRACK TO PICK UP REMAINING PTS.
 12 DO 13 I=IA,N
    IF (DIST(I)- PIVOT) 14,14,15
 14 AX(I)=(A1+A3*X(I) -Y(I))/(A3-A2)
    AY(I)= A2*AX(I) +A1
    GO TO 13
 15 J= I-1
    DELZ=(Z(I)-Z(J))/SQRT((X(I)-X(J))**2 +(Y(I)-Y(J))**2)
    AX(I)=(A1+A3*X(I)-Y(I))/(A3-A2)
    AY(I)=A2*AX(I) +A1
    Z(I)= (DELZ *SQRT((AX(I)-AX(J))**2+(AY(I)-AY(J))**2))+Z(J)
 13 CONTINUE
    WRITE(61,103) PIVOT
103 FORMAT(*        X           Y           Z    INPUT DATA MAPPED ONTO TRAC
   1K PIVOT DISTANCE =*,F4.1,*UNITS*)
C      IF THESE DATA ARE DESIRED REMOVE C FROM NEXT CARD
C       WRITE(61,102) (AX(I),AY(I),Z(I), I=1,N)
102     FORMAT(3F10.1)
C      AT THIS POINT WE HAVE MAPPED ALL INPUT PTS. ONTO TRACK WITH
C      CORRECT Z VALUES.NOW USE CUBIC SPLINE TO INTERPOLATE FOR GRID PTS.
C      INDEPENDENT VARIABLE IS DISTANCE DOWN TRACK FROM 1ST PT.
C      CHECK QUADRANT TO DETERMINE IF INTERPOLATION IS IN X OR Y
C      DIRECTION.THIS IS CONTROLLED BY TRACK NO.1 IN STATEMENT 32 AND 30
    IF(NTRK-1)83,83,30
 83 A4=A2
121  IF(MEAN) 122,30,122
122  GRID=GRID/2.0
 30 IF(ABS(A4)-1.0)16,16,17
 16 DELD=SQRT((A2*GRID)**2 +GRID **2)
```

```
      KX=AX(1)/GRID
      AKX=KX
      AKX=AKX*GRID
      AKY=A1+A2*AKX
      START=SQRT((AX(1)-AKX)**2+(AY(1)-AKY)**2)
      IF(AX(2)-AX(1))88,88,21
   21 IF(AX(1))98,98,22
   22 START =DELD-START
      AKX=AKX+GRID
      AKY=A1+A2*AKX
   98  SIGN=1.0
      GO TO 18
   17 DELD=SQRT((GRID/A2)**2 +GRID**2)
      KY=AY(1)/GRID
      AKY=KY
      AKY=AKY*GRID
      AKX=(AKY-A1)/A2
      START=SQRT((AX(1)-AKX)**2 +(AY(1)-AKY)**2)
      IF(AY(2)-AY(1))88,23,23
   23 IF(AY(1))98,98,24
   24 START=DELD-START
      AKY=AKY +GRID
      AKX=(AKY-A1)/A2
      SIGN= 1.0
      GO TO 18
   88  SIGN= -1.0
   18 DO 19 I=1,N
   19 DIST(I)=SQRT((AX(I)-AX(1))**2+(AY(I)-AY(1))**2)
      ICT(NTRK)= (DIST(N)/DELD)+1.0
      JCT= ICT(NTRK)
      WRITE(61,104)
  104 FORMAT(*        X           Y           Z    INTERPOLATED VALUES ALONG T
     1RACK AT EGUAL GRID SPACING*)
      DO 25 I=1,JCT
      AJ=I-1
      XINT=(AJ*DELD)+START
      CALL SPLINE( DIST,Z,N,XINT,ZINT,ATER)
```

```
C     NOW COMPUTE X AND Y VALUE FOR THIS INTERPOLATED VALUE OF Z
      IF(ABS(A4)-1.0) 26,26,27
   26 BX(I,NTRK)=AKX+AJ*GRID *SIGN
      BY(I,NTRK)= A1+A2*BX(I,NTRK)
      GO TO 28
   27 BY(I,NTRK)=AKY+AJ*GRID *SIGN
      BX(I,NTRK) = (BY(I,NTRK) -A1)/A2
   28 BZ(I,NTRK)=ZINT
C     IF THESE DATA ARE DESIRED REMOVE C FROM NEXT CARD
C     WRITE(61,102)(BX(I,NTRK),BY(I,NTRK),BZ(I,NTRK))
   25 CONTINUE
C     INTERPOLATED VALUES OF Z HAVE NOW BEEN COMPUTED AT EQUALLY SPACED
C     VALUES OF X OR Y DEPENDING ON TEST IN STATEMENT 32 AND 30 AND
C     STORED WITH 2ND INDEX = TRACK NO.,NOW COMPLETE ABOVE PROCESS
C     FOR ALL TRACKS ON THIS RUN
      IF(NTRK-ITOT) 31,32,32
C     NOW COMPUTE GRID VALUES,IF ABOVE INTERPOLATION WAS IN X,SORT
C     POINTS INTO INCREASING Y AND INTERPOLATE FOR GRID VALUES IN Y
C     DIRECTION ,IF ABOVE PROCESS WAS IN Y DIRECTION,INTERCHANGE X AND Y
   32 IF(ABS(A4)-1.0) 33,33,53
   33 DX=0.0
      ITEMP=(RLAT/GRID)+1.0
      DO 123 I=1,ITEMP
  123  AVE(I)=0.0
       KTEMP=1
   39 K=0
      ATER= 9999.99
      DO 34 J=1,ITOT
      JCT= ICT(J)
      DO 35 I=1,JCT
       IF(BX(I,J).LT.(DX+.001).AND.BX(I,J).GT.(DX-.001)) GO TO 36
   35 CONTINUE
      GO TO 34
   36 K=K+1
      Y(K)=BY(I,J)
      X(K)=DX
      Z(K)=BZ(I,J)
```

```
   36 CONTINUE
      IF(K.EQ.0) GO TO 108
       WRITE( 61,106) DX
  106 FORMAT(*        X           Y           Z   INPUT DATA FOR FINAL INTERP
     1OLATION IN Y DIRECTION FOR X = *,F7.1)
C     IF THESE DATA ARE DESIRED REMOVE C FROM NEXT CARD
C     WRITE(61,102) (X(I),Y(I),Z(I),I=1,K)
C     NOW CHECK IF THERE ARE ENOUGH PTS ON THIS LINE TO INTERPOLATE
      IF(K-9) 108,108,38
C     NOW SORT DATA INTO INCREASING Y FOR THIS VALUE OF X
   38 CALL SORTY(X,Y,Z,AX,AY,AZ,K,1,+1,GRID)
   41 DO 45 IA=1,ITEMP
      AJ=IA-1
      XINT= AJ*GRID
      CALL SPLINE(Y,Z,K,XINT,ZINT,ATER)
       AY(IA)= XINT
       AX(IA)= DX
   45  AZ(IA)= ZINT
      IF(MEAN) 112,112,124
  124  DO 125 IA=1,ITEMP
  125 AVE(IA)=AVE(IA)+(AZ(IA)/9.0)
      IF(KTEMP-3) 126,127,127
  127  DO 128 IA=3,ITEMP,2
      IB=IA-1
      IC=IA-2
  128 AVE(IC)=AVE(IA)+AVE(IB)+AVE(IC)
      WRITE(61,80)
   80 FORMAT(*    X       Y          Z        X       Y          Z       X
     1     Y              Z MEAN ANOMALY DATA*)
      DO 129 IA=1,ITEMP,2
      AX(IA)=AX(IA)-GRID *2.0
  129 AY(IA)=AY(IA)+GRID *2.0
      WRITE(61,50) (AX(I),AY(I),AVE(I), I=1,ITEMP,2)
      PUNCH 1000, (AVE(I), I=1,ITEMP,2)
      DO 131 J=1,ITEMP
  131 AVE(J)=0.0
      KTEMP=1
```

```
      GO TO 124
  126 KTEMP=KTEMP+1
      GO TO 109
C     OUTPUT THIS GRIDDED DATA
  112 WRITE(61,40)
   40 FORMAT(*    X          Y              Z         X          Y              Z        X
     1    Y              Z FINAL GRIDDED DATA*)
   37  WRITE(61,50)   (AX(I),AY(I),AZ(I), I=1,ITEMP)
      PUNCH 1000,(AZ(I),I=1,ITEMP)
 1000 FORMAT (7F11.2)
   50 FORMAT(1X,3(F7.1,F8.1,F11.2))
      GO TO 109
  108  WRITE(61,110) DX
  110 FORMAT(* NOT ENOUGH PTS FOR VALID INTERPOLATION ALONG X=*,F7.1)
  109 IF(RLONG-DX) 67,67,66
   46 DX=DX+GRID
      GO TO 39
   53 DY=0.0
      ITEMP=(RLONG/GRID)+1.0
      DO 132 I=1,ITEMP
  132  AVE(I)=0.0
      KTEMP=1
   59 K=0
      ATER= 9999.99
      DO 54 J=1,ITOT
      JCT= ICT(J)
      DO 55 I=1,JCT
       IF(BY(I,J).LT.(DY+.001).AND.BY(I,J).GT.(DY-.001)) GO TO 56
   55 CONTINUE
      GO TO 54
   56 K=K+1
      Y(K)=DY
      X(K)=BX(I,J)
      Z(K)=BZ(I,J)
   54 CONTINUE
      IF(K.EQ.0) GO TO 111
      WRITE(61,107) DY
```

```
  107 FORMAT(*        X            Y              Z    INPUT DATA FOR FINAL INTERP
     1OLATION IN X DIRECTION FOR Y = *,F7.1)
C       IF THESE DATA ARE DESIRED REMOVE C FROM NEXT CARD
C       WRITE(61,102) (X(I),Y(I),Z(I),I=1,K)
C       CHECK IF ENOUGH PTS TO INTERPOLATE
        IF(K-3) 111,111,58
C       NOW SORT DATA INTO INCREASING X FOR THIS VALUE OF Y
   58 CALL SORTY(Y,X,Z,AY,AX,AZ,K,1,+1,GRID)
   61 DO 65 IA=1,ITEMP
      AJ=IA-1
      XINT=AJ*GRID
      CALL SPLINE(X,Z,K,XINT,ZINT,ATER)
       AX(IA)= XINT
       AY(IA)= DY
   65  AZ(IA)= ZINT
      IF(MEAN) 113,113,133
  133 DO 134 IA=1,ITEMP
  134 AVE(IA)= AVE(IA)+(AZ(IA)/ 9.0)
      IF (KTEMP-3) 135,136,136
  136 DO 137 IA=3,ITEMP,2
      IB=IA-1
      IC=IA-2
  137 AVE(IC)= AVE(IA)+AVE(IB)+AVE(IC)
      WRITE (61,80)
      DO 138 IA=1,ITEMP,2
      AX(IA)= AX(IA)
  138 AY(IA)= AY(IA)
      WRITE(61,50) (AX(I),AY(I),AVE(I), I=1,ITEMP,2)
      PUNCH 1000, (AVE(I), I=1,ITEMP,2)
      DO 139 J=1,ITEMP
  139 AVE(J)=0.0
      KTEMP=1
      GO TO 133
  135 KTEMP= KTEMP+1
      GO TO 114
C       OUTPUT THIS GRIDDED DATA
  113  WRITE(61,40)
```

```
 57  WRITE(61,50)   (AX(I),AY(I),AZ(I), I=1,ITEMP)
     PUNCH 1000,(AZ(I),I=1,ITEMP)
     GO TO 114
111  WRITE(61,115) DY
115  FORMAT(* NOT ENOUGH PTS FOR VALID INTERPOLATION ALONG Y=*,F7.1)
114  IF(RLAT -DY)67,67,66
 66  DY=DY +GRID
     GO TO 59
 67  CONTINUE
     STOP
     END


     SUBROUTINE SPLINE (X,Y,M,XINT,YINT,ATER)
C
C                                    A SYNBAPS SUBROUTINE
C                                  **********************
C
C    SEE PENNINGTON REF. FOR DESCRIPTION OF THIS SUBROUTINE
C
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
     DIMENSION X(99),Y(99),C(4,82)
     IF(X(1)+Y(M)+Y(M-1)+X(M-1)+Y(M-2)-ATER) 10,3,10
 10  CALL SPLICON(X,Y,M,C)
     ATER= X(1)+Y(M)+Y(M-1)+X(M-1)+Y(M-2)
     K=1
  3  IF(XINT-X(1)) 70,1,2
 70  K=1
     GO TO 7
  1  YINT=Y(1)
     RETURN
  2  IF(XINT-X(K+1))6,4,5
  4  YINT=Y(K+1)
     RETURN
  5  K=K+1
```

```
      IF(M-K) 71,71,3
   71 K=M-1
      GO TO 7
    6 IF(XINT-X(K))13,12,11
   12 YINT=Y(K)
      RETURN
   13 K=K-1
      GO TO 6
   11 YINT=(X(K+1)-XINT)*(C(1,K)*(X(K+1)-XINT)**2+C(3,K))
      YINT=YINT+(XINT-X(K))*(C(2,K)*(XINT-X(K))**2+C(4,K))
      RETURN
    7  PRINT 101, XINT
  101 FORMAT(* CAUTION VALUE AT POSITION*,F10.2,* WAS EXTRAPOLATED*)
      GO TO 11
      END

      SUBROUTINE SPLICON(X,Y,M,C)
C
C                             A SYNBAPS SUBROUTINE
C                            **********************
C
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION X(99),Y(99),C(4,82),D(82),P(82),E(82),A(82,3),B(82),
     1Z(82)
      MM=M-1
      DO 2 K=1,MM
      D(K)=X(K+1)-X(K)
      P(K)=D(K)/6.
    2 E(K)=(Y(K+1)-Y(K))/D(K)
      DO 3 K=2,MM
    3 B(K)=E(K)-E(K-1)
      A(1,2)=-1.-D(1)/D(2)
      A(1,3)=D(1)/D(2)
      A(2,3)=P(2)-P(1)*A(1,3)
      A(2,2)=2.*(P(1)+P(2))-P(1)*A(1,2)
```

```
      A(2,3)=A(2,3)/A(2,2)
      B(2)=B(2)/A(2,2)
      DO 4 K=3,MM
      A(K,2)=2.*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
      B(K)=B(K)-P(K-1)*B(K-1)
      A(K,3)=P(K)/A(K,2)
    4 B(K)=B(K)/A(K,2)
      Q=D(M-2)/D(M-1)
      A(M,1)=1.+Q+A(M-2,3)
      A(M,2)=-Q-A(M,1)*A(M-1,3)
      B(M)=B(M-2)-A(M,1)*B(M-1)
      Z(M)=B(M)/A(M,2)
      MN=M-2
      DO 6 I=1,MN
      K=M-I
    6 Z(K)=B(K)-A(K,3)*Z(K+1)
      Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
      DO 7 K=1,MM
      Q=1./(6.*D(K))
      C(1,K)=Z(K)*Q
      C(2,K)=Z(K+1)*Q
      C(3,K)=Y(K)/D(K)-Z(K)*P(K)
    7 C(4,K)=Y(K+1)/D(K)-Z(K+1)*P(K)
      END
```

```
      SUBROUTINE SORTY(X,Y,Z,AX,AY,AZ,K,KODE,JCODE,GRID)
C
C                                A SYNBAPS SUBROUTINE
C                                **********************
C
C     Y=INPUT VARIABLE TO BE SORTED. X,Z=VALUES ASSOCIATED WITH Y
C     K=LENGTH OF Y,IF KODE=1,VALUES OF Y WHICH ARE WITHIN.25GRID INT OF
C     PREVIOUS VALUE ARE REMOVED,IF KODE=0 ALL VALUES OF Y ARE
C     RETAINED,IF JCODE=+1,Y IS SORTED IN INCREASING ORDER,IF JCODE=-1,
C     Y IS SORTED IN DECREASING ORDER,OUTPUT IS SORTED VALUES OF
C     Y WITH ASSOCIATED X AND Z
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
C
      DIMENSION Y(99),X(99),Z(99),AX(99),AY(99),AZ(99)
      KB = K
      CODE=JCODE
      J = 1
129   I = 1
      JCT=0
      AY(J) = Y(I)
  132 TEMP= CODE*(AY(J)-Y(I+1))
      IF((ABS(TEMP))-GRID/4.0)122,122,120
  120 IF(TEMP)121,122,123
121   I = I + 1
      IF((I + 1) - KB) 132,132,125
123   JCT = 1
      AY(J) = Y(I + 1)
      AX(J) = X(I + 1)
      AZ(J) = Z(I + 1)
      KT = I + 2
      GO TO 121
  122  IF(KODE) 121,121,136
  136  KD=I+2
      IF (KD-KB) 124,124,139
  139  K=K-1
      KB=KB-1
```

```
      GO TO 125
  124 DO 126 JD=KD,KB
      JF = JD - 1
      Y(JF) = Y(JD)
      X(JF) = X(JD)
  126 Z(JF) = Z(JD)
      KB = KB - 1
      K = K - 1
      GO TO 132
  125 IF(JCT) 127,127,128
  127 AY(J) = Y(1)
      AX(J) = X(1)
      AZ(J) = Z(1)
      KT = 2
  128 J = J + 1
      IF(J - K) 131,133,133
  131 DO 134 KA = KT,KB
      JT = KA - 1
      Y(JT) = Y(KA)
      X(JT) = X(KA)
  134 Z(JT) = Z(KA)
      KB = KB - 1
      GO TO 129
  133 IF(JCT) 137,137,138
  137 KB=KB+1
  138 AY(K) = Y(KB- 1)
      AX(K) = X(KB- 1)
      AZ(K) = Z(KB- 1)
      DO 135 I = 1,K
      Y(I) = AY(I)
      X(I) = AX(I)
  135 Z(I) = AZ(I)
      RETURN
      END
```

```
      PROGRAM SYNCON2R
C
C                               A SYNBAPS PROGRAM
C                               *****************
C
C PROGRAM PLOTS ROUGH CONTOURS OF MSQLOC AREA FOR CHECKING PURPOSES-
C THE PLOT IS AN OVERLAY AT THE SAME SCALE AS THE SOURCE CHART
C NO LABELS ARE USED
C
C REQUIRES SUBROUTINES CONTOUR, LABEL, INTERP, SCAN, TRACE, GET PT, AND FIT
C REQUIRES FUNCTION FX, FY
C
C PROGRAM WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION STORX(2000),LABELS(40),CL(50),A(100,20)
      EXTERNAL  FX
      EXTERNAL  FY
      COMMON/MATRIX/ Z(101,101)
      DATA(LABELS=8H        ,8H        ,8H        ,8H        ,8H
     1,8H        ,8H        ,8H        ,8H        ,8H        )
     2(CL=3400.,3200.,3000.,2800.,2600.,2400.,2200.,2000.,1800.,1600.,
     31400.,1200.,1000.,800.)
C     CL= DESIRED CONTOUR VALUES,NCL= NO.OF CONTOUR LEVELS
C     M=NO.OF ROWS INPUT,N=NO.OF COLS. YG=LENGTH OF Y AXIS,XG=LENGTH OF
C     X AXIS IN DECIMAL INCHES
C     FIRST INDEX ON Z IS ROWS(Y),2ND INDEX IS COLS(X)
      READ(60,1) N ,M ,NCL,MM,NN,XA,YA,XG,YG
    1 FORMAT(5I4,4F10.0)
      YB=FLOAT(M)
      XB=FLOAT(N)
      DO 69 K=1,N
      READ(60,50) (Z(J,K),J=1,M)
   50 FORMAT(7F11.2)
   69 CONTINUE
      CALL PLOTS(STORX,2000,3,29)
      CALL CONTOUR(M,N,MM,NN,XA,XB,YA,YB,XG,YG,NCL,CL,LABELS,FX,FY)
      CALL STOPPLOT
```

```
      STOP 77777
      END

      FUNCTION  FX(X)
      FX =X
      END

      FUNCTION  FY(Y)
      FY =Y
      END

      SUBROUTINE LABEL(ITITLE,FX,FY)
C**********SUBROUTINE LABEL CHANGED FROM ORIGINAL(NOW PLOTS ONLY AXIS LABELS)
      COMMON/TEMP/Z(101)                                                CON 8300
      COMMON/XYBOUNDS/XA,XB,YA,YB,XSIZE,YSIZE,HX,HY,                    CON 8400
     1XS,XSS,YS,YSS,FXA,FYA                                             CON 8500
      COMMON/INDICES/M,N,MM,NN/CLEVELS/NCL,NLV,CL(50)                   CON 8600
      DIMENSION ITITLE(1)
      J=0
      X=XA
      P=0
      G=XSIZE-.9
      XG=XSS*(FX(X)-FXA)
   15 CALL SYMBOL(0.-0.9,0.10,ITITLE(8),0,8)
   11 CALLSYMBOL(MAX1F(.2,.5*(XSIZE-2.84)),-.9,.14,ITITLE(5),0,24)      CON10200
      CALLSYMBOL(MAX1F(.2,.5*(XSIZE-5.70)),YSIZE+.30,.21,ITITLE,0,32)   CON10300
      END                                                               CON12800
```

```
      SUBROUTINE CONTOUR                                                CON 2700
C ROUTINE WRITTEN BY ATOMIC ENERGY COMMISSION PERSONNEL
     1(M,N,MM,NN,XA,XB,YA,YB,XG,YG,NCL,CL,ITITLE,FX,FY)                 CON 2800
      COMMON/INDICES/MROW,NCOL,MMROW,NNCOL                              CON 2900
      COMMON/XYBOUNDS/XMIN,XMAX,YMIN,YMAX,XSIZE,YSIZE,                  CON 3000
     1HX,HY,XS,XSS,YS,YSS,FXA,FYA                                       CON 3100
      COMMON/CLEVELS/NLVLS,NLV,CLEVEL(50)                               CON 3200
      COMMON/CAVIN/IDIM,DUM(4035)                                       CON 3300
      COMMON/MATRIX/Z(101,101)                                          CON 3400
      DIMENSION CL(1)                                                   CON 3500
C     Z(I,J) IS THE ORDINATE AT POINT X(J), Y(I)                        CON 3600
C     MXN IS THE SIZE OF THE CALCULATED X-Y GRID                        CON 3700
C     MMXNN IS THE SIZE OF THE EXPANDED(BY INTERPOLATION) X-Y GRID      CON 3800
C     XA,XB,YA,YB ARE THE MINIMUM AND MAXIMUM VALUES                    CON 3900
C          OF X AND Y,                                                  CON 4000
C     XG IS THE WIDTH OF THE GRAPH IN INCHES.                           CON 4100
C     YG IS THE HEIGHT OF THE GRAPH IN INCHES.                          CON 4200
C     NCL IS THE NO. OF CONTOUR LEVELS                                  CON 4300
C     CL(I) ARE THE CONTOUR LEVELS                                      CON 4400
C     ITITLE CONTAINS THE PLOT TITLES IN 80 BCD CHARACTERS,             CON 4500
C          PLOT NAME IS FIRST 4 WORDS,                                  CON 4600
C          X-AXIS LABEL IS NEXT 3 WORDS,                                CON 4700
C          Y-AXIS LABEL IS NEXT 3 WORDS,                                CON 4800
C     THE X(I) ARE ASSUMED TO BE EQUALLY SPACED, AND                    CON 4900
C     LIKEWISE, THE Y(I).                                               CON 5000
C     FX IS THE FUNCTION TO BE PLOTTED ALONG THE X-AXIS.                CON 5100
C     FY IS THE FUNCTION TO BE PLOTTED ALONG THE Y-AXIS.                CON 5200
      MROW=M $ NCOL=N $ MMROW=MM $ NNCOL=NN                             CON 5300
      XMIN=XA $ XMAX=XB $ YMIN=YA $ YMAX=YB                             CON 5400
      XSIZE=XG $ YSIZE=YG $ NLVLS=XABSF(NCL)                            CON 5500
      CALL PLOT(0.0,0.5*(29.0-YSIZE),-3)
      IF(NCL)1,9,9                                                      CON 5700
    1 CLEVEL=HX=Z $ L=0                                                 CON 5800
      DO15I=1,NCOL $ DO7J=1,MROW $ L=L+1                                CON 5900
      IF(Z(L).LT.CLEVEL)4,5                                             CON 6000
    4 CLEVEL=Z(L)                                                       CON 6100
    5 IF(Z(L).GT.HX)6,7                                                 CON 6200
```

```
    6 HX=Z(L)                                                        CON 6300
    7 CONTINUE                                                       CON 6400
   15 L=L-M+IDIM    $ HX=(HX-CLEVEL)/FLOATF(NLVLS-1)                 CON 6500
      DO8I=2,NLVLS                                                   CON 6600
    8 CLEVEL(I)=CLEVEL(I-1)+HX $ GOTO11                              CON 6700
    9 DO10I=1,NLVLS                                                  CON 6800
   10 CLEVEL (I)=CL(I)                                               CON 6900
   11 HX=(XMAX-XMIN)/FLOATF(NCOL-1)                                  CON 7000
      HY=(YMAX-YMIN)/FLOATF(MROW-1)                                  CON 7100
      XS=(XMAX-XMIN)/FLOATF(NNCOL-1)                                 CON 7200
      YS=(YMAX-YMIN)/FLOATF(MMROW-1)                                 CON 7300
      FXA=FX(XMIN) $ FYA=FY(YMIN)                                    CON 7400
      XSS=XG/(FX(XMAX)-FXA)                                          CON 7500
      YSS=YG/(FY(YMAX)-FYA)                                          CON 7600
    2 CALLINTERP                                                     CON 7700
      DO3NLV=1,NLVLS                                                 CON 7800
    3 CALLSCAN(FX,FY)
C    THE CALLABEL(ITITLE,FX,FY) HAS BEEN PULLED. PUT BACK FOR LABELLING.
C     PLACE TICK MARKS AT THE FOUR CORNERS OF THE GRAPH
      CALL PLOT(0.0,0.0,3)

      CALL PLOT(XG,0.,2)

      CALL PLOT(XG,YG,2)

      CALL PLOT(0.,YG,2)

      CALL PLOT(0,0,2)
      CALL PLOT(XSIZE + 5.0,0.0,-3)
      END                                                            CON 8100
```

```
      SUBROUTINE INTERP                                                 CON12900
      COMMON/MATRIX/AM(101,101)                                         CON13000
      COMMON/TEMP/Z(101)                                                CON13100
      COMMON/XYBOUNDS/XA,XB,YA,YB,XG,YG,HX,HY,                          CON13200
     1XS,XSS,YS,YSS,FXA,FYA                                             CON13300
      COMMON/INDICES/M,N,MM,NN                                          CON13400
      ZFUN(V)=A0+A1*V+A2*V**2                                           CON13500
      N1=N-1 $ M1=M-1                                                   CON13600
      IF(N-NN)16,15,14                                                  CON13700
   16 DO6I=1,M                                                          CON13800
      DO1J=1,N                                                          CON13900
    1 Z(J)=AM(I,J) $ XY=XA $ K=1 $ T=HX+XA                              CON14000
      DO3J=2,N1 $ CALLFIT(J,T,HX,A0,A1,A2)                              CON14100
    2 AM(I,K)=ZFUN(XY) $ XY=XY+XS $ K=K+1 $ IF(XY-T)2,2,3               CON14200
    3 T=T+HX                                                            CON14300
    4 IF(K-NN)5,5,6                                                     CON14400
    5 AM(I,K)=ZFUN(XY) $ K=K+1 $ XY=XY+XS $ GOTO4                       CON14500
    6 CONTINUE                                                          CON14600
   15 IF(M-MM)17,13,14                                                  CON14700
   17 DO12I=1,NN                                                        CON14800
      DO7J=1,M                                                          CON14900
    7 Z(J)=AM(J,I) $ K=1 $ XY=YA $ T=HY+YA                              CON1500
      DO9J=2,M1 $ CALLFIT(J,T,HY,A0,A1,A2)                              CON15100
    8 AM(K,I)=ZFUN(XY) $ XY=XY+YS $ K=K+1 $ IF(XY-T)8,8,9               CON15200
    9 T=T+HY                                                            CON15300
   10 IF(K-MM)11,11,12                                                  CON15400
   11 AM(K,I)=ZFUN(XY) $ K=K+1 $ XY=XY+YS $ GOTO10                      CON15500
   12 CONTINUE                                                          CON15600
   13 RETURN                                                            CON15700
   14 PRINT 999                                                         CON15720
  999 FORMAT(1H1,* PROGRAM TERMINATED BECAUSE OF INCORRECT INPUT PARAMETCON15740
     1ERS TO SUBROUTINE CONTOUR.*//* (EITHER M IS GREATER THAN MM OR N ICON15760
     2S GREATER THAN NN.)*)                                             CON15780
      STOP                                                              CON15800
      END                                                               CON15900
```

```
      SUBROUTINE SCAN(FX,FY)
C AM IS THE MATRIX TO BE CONTOURED. MT AND NT ARE ITS X AND Y DIMENSIONS
C CL(NLV) IS THE CONTOUR LEVEL.
C THE N  (X,Y)  VALUES OF ONE CONTOUR LINE ARE PLOTTED WHEN
C  THEY ARE AVAILABLE.
      DIMENSION AM(101,101)
      COMMON/MATRIX/AM/CLEVEL/ICL,NLV,CL(50)
      COMMON/INDICES/DUM(2),MT,NT
      COMMON/CAVIN/DIM, IX,IY,IDX,IDY,ISS,
     1 NP,N,CV,IS,IS0,IX0,IY0,DCP,
     2    INX(8),INY(8),REC(800),X(1603),Y(1603)
      TYPE INTEGER REC ,DIM
      DATA(INX=-1,-1,0,1,1,1,0,-1,0,1,1,1,0,-1,-1,-1)
      DATA(DIM=101)
      NP=ISS=0
      CV=CL(NLV)
      MT1=MT-1 $ NT1=NT-1
      DO 110 I=1,MT1
      IF(AM(I)   -CV)55,110,110
   55 IF(AM(I+1)   -CV)110,57,57
   57 IX0=IX=I+1 $ IY0=IY=IS0=IS=1 $ IDX=-1 $ IDY=0
      CALL TRACE(FX,FY)
  110 CONTINUE
      J=MT-DIM $ DO20I=1,NT1 $ J=J+DIM
      IF(AM(J)-CV)15,20,20
   15 IF(AM(J+DIM)-CV)20,17,17
   17 IX0=IX=MT $ IY0=IY=I+1 $ IDX=0 $ IDY=-1 $ IS0=IS=7
      CALL TRACE(FX,FY)
   20 CONTINUE
      J=MT+NT1*DIM+1 $ DO30I=1,MT1 $ J=J-1
      IF(AM(J)-CV)25,30,30
   25 IF(AM(J-1)-CV)30,27,27
   27 IX0=IX=MT-I  $IY0=IY=NT $ IDX=1 $ IDY=0 $ IS0=IS=5
      CALL TRACE(FX,FY)
   30 CONTINUE
      J=NT*DIM+1 $ DO40I=1,NT1 $ J=J-DIM
      IF(AM(J)-CV)35,40,40
```

```
35 IF(AM(J-DIM)-CV)40,37,37                                          CON19700
37 IX0=IX=1 $ IY0=IY=NT-I $ IDX=0 $ IDY=1 $ IS0=IS=3                 CON19800
   CALL TRACE(FX,FY)                                                 CON19900
40 CONTINUE                                                          CON20000
   ISS=1 $ L=0                                                       CON20100
   DO13J=2,NT1 $ L=L+DIM                                             CON20200
   DO10I=1,MT1 $ L=L+1                                               CON20300
   IF(AM(L)-CV)5,10,10                                               CON20400
 5 IF(AM(L+1)-CV)10,7,7                                              CON20500
 7 K=L+1                                                             CON20600
   DO 9 ID = 1,NP                                                    CON20700
   IF(REC(ID)-K)9,10,9                                               CON20800
 9 CONTINUE                                                          CON20900
   IX0=IX=I+1 $ IY0=IY=J $ IDX=-1 $ IDY=0 $ IS0=IS=1                 CON21000
   CALL TRACE(FX,FY)                                                 CON21100
10 CONTINUE                                                          CON21200
13 L=L-MT1                                                           CON21300
   END                                                               CON21400


   SUBROUTINE TRACE(FX,FY)                                           CON21500
   DIMENSION AM(101,101)                                             CON21600
   COMMON/MATRIX/AM/INDICES/DUM(2),MT,NT                             CON21700
   COMMON/XYBOUNDS/XA,XB,YA,YB,XSIZE,YSIZE,HX,HY,                    CON21800
  1XS,XSS,YS,YSS,FXA,FYA                                             CON21900
   COMMON/CAVIN/DIM, IX,IY,IDX,IDY,ISS,                              CON22000
  1 NP,N,CV,IS,IS0,IX0,IY0,DCP,                                      CON22100
  2    INX(8),INY(8),REC(800),X(1603),Y(1603)                        CON22200
   COMMON/CLEVELS/NCL,NLV,CL(50)                                     CON22300
   TYPE INTEGER REC,DIM                                              CON22400
   N=0 $ JY=DIM*(IY-1)+IX $ MY=DIM*IDY+IDX+JY                        CON22500
 2 N=N+1 $ IF(N-1600)3,3,32                                          CON22600
 3 IF(IDX)5,4,6                                                      CON22700
 4 X(N)=FLOATF(IY-1)+FLOATF(IDY)*(AM(JY)-CV)/(AM(JY)-AM(DIM*IC/+JY)) CON22800
   Y(N)=FLOATF(IX-1) $ GOTO7                                         CON22900
 5 NP=NP+1 $ REC(NP)=JY                                              CON23000
```

```
 6 Y(N)=FLOATF(IX-1)+FLOATF(IDX)*(AM(JY)-CV)/(AM(JY)-AM(JY+IDX))        CON23100
   X(N)=FLOATF(IY-1)                                                    CON23200
 7 IS=IS+1                                                              CON23300
 8 IF(IS-8)10,10,9                                                      CON23400
 9 IS=IS-8                                                              CON23500
10 IDX=INX(IS) $ IDY=INY(IS)                                            CON23600
   IX2=IX+IDX $ IY2=IY+IDY $ IR=IDX*IDY                                 CON23700
11 IF(ISS)13,15                                                         CON23800
13 IF(IS.NE.IS0.OR.IY.NE.IY0.OR.IX.NE.IX0)15,14                         CON23900
14 N=N+1 $ X(N)=X $ Y(N)=Y $ GOTO73                                     CON24000
15 IF(IX2.AND.IX2.LE.MT.AND.IY2.AND.IY2.LE.NT)16,73                     CON24100
16 MY=DIM*IDY+IDX+JY $ IF(IR)19,17,20                                   CON24200
17 IF(CV-AM(MY))18,18,2                                                  CON24300
18 IX=IX2 $ IY=IY2                                                      CON24400
81 IS=IS+5 $ JY=MY $ GOTO8                                              CON24500
19 KY=JY+IDX $ LY=MY-IDX $ GOTO21                                       CON24600
20 KY=MY-IDY $ LY=JY+IDY                                                CON24700
21 DCP=(AM(JY)+AM(KY)+AM(LY)+AM(MY))*.25 $ IF(CV-DCP)23,23,22           CON24800
22 CALL GETPT(JY) $ GOTO7                                               CON24900
23 IF(IR)24,25,25                                                       CON25000
24 IX=IX2 $ [illegible]DX $ CALL GETPT(KY)                              CON25100
   IY=IY2 $ [illegible]IDY $ GOTO26                                     CON25200
25 IY=IY2 $ [illegible]-IDY $ CALL GETPT(KY)                            CON25300
   IX=IX2 $ IDX=-IDX                                                    CON25400
26 IF(CV-AM(MY))81,81,28                                                CON25500
28 CALLGETPT(MY) $ IF(IR)29,30,30                                       CON25600
29 IX=IX+IDX $ IDX=-IDX $ GOTO31                                        CON25700
30 IY=IY+IDY $ IDY=-IDY                                                 CON25800
31 IF(CV-AM(LY))33,33,34                                                CON25900
33 IS=IS-1 $  JY=LY $ GOTO10                                            CON26000
34 CALLGETPT(LY) $ IF(IR)35,36,36                                       CON26100
35 IY=IY+IDY $ GOTO7                                                    CON26200
36 IX=IX+IDX $ GOTO7                                                    CON26300
32 PRINT103,CV                                                          CON26400
73 DO74I=1,N                                                            CON26500
   X(I)=XSS*(FX(X(I)*XS+XA)-FXA)                                        CON26600
74 Y(I)=YSS*(FY(Y(I)*YS+YA)-FYA)                                        CON26700
```

```
      CALL NUMBER(X,Y,.08,CV,0.,4HF5.0)
      CALL PLOT(X(1),Y(1),3)
      DO75I=1,N
   75 CALL PLOT(X(I), Y(I), 2)
      RETURN
  103 FORMAT(1H0,23HA CONTOUR LINE AT LEVEL,F10.5,
     1 41H WAS TERMINATED BECAUSE IT CONTAINED MORE,
     2 23H THAN 1600 PLOT POINTS.)
      END


      SUBROUTINE GET PT(J)
      COMMON/MATRIX/AM(101,101)
      COMMON/CAVIN/DIM, IX,IY,IDX,IDY,ISS,
     1NP,N,CV,IS,ISO,IXO,IYO,DCP,
     2    INX(8),INY(8),REC(800),X(1603),Y(1603)
      N=N+1 $ B=AM(J)-DCP $ IF(B)2,1
    1 V=.5 $ GOTO3
    2 V=.5*(AM(J)-CV)/B
    3 Y(N)=FLOATF(IX-1)+FLOATF(IDX)*V $ X(N)=FLOATF(IY-1)+FLOATF(IDY)*V
      END


      SUBROUTINE FIT(I,X,H,C,B,A)
      COMMON/TEMP/Z(101)
      W=0.5*(Z(I+1)-Z(I-1))/H
      A=0.5*(Z(I+1)+Z(I-1)-Z(I)-Z(I))/H**2
      C=Z(I)+X*(X*A-W) $ B=W-2.*X*A $ END
```

```
      PROGRAM SYNTABLE
C
C                                    A SYNBAPS PROGRAM
C                                    *******************
C
C PROGRAM CREATES A LOOK UP TABLE OF MSQLOC AREAS. THEIR FILE NAME. RELATIVE
C ADDRESS, AND BLOCK SIZE.
C
C REQUIRES SUBROUTINES DKOPEN, DKLOCATE, DKWRITE, DKREAD, AND FUNCTION TIMELEFT
C TABLE STARTS AT RELATIVE ADDRESS =128704 ON DISK FILE E08C
C
C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION LOCTAB(2368),LOCOUT(2368)
      DATA (N=24)
      NN=N*4
      IF((N/3)*3 .EQ. N) GO TO 3
      NC=N/3 +1
      GO TO 4
    3 NC=N/3
    4 M=12
      K=1
      DO 77 J=1,NC
      READ 1,(LOCTAB(I),I=K,M)
    1 FORMAT(3(3I6,2X,A4))
      K=K+12
   77 M=K+11
      T1=TIMELEFT(0)
      CALL DKOPEN(5,3HRAN,4HE08C)
      CALL DKLOCATE(128704)
      CALL DKWRITE(LOCTAB(1),LOCTAB(2368))
      CALL DKLOCATE(128704)
      CALL DKREAD(LOCOUT(1),LOCOUT(2368))
      T2=TIMELEFT(0) $ T=T1-T2
      PRINT 29
   29 FORMAT(17X,31HSYNBAPS DISK FILE LOCATOR TABLE,//)
      PRINT 30
```

```
 30 FORMAT(15X,6HMSQLOC,3X,8HRELATIVE,2X,7HSIZE OF,5X,4HFILE,/,24X,
   17HADDRESS,4X,5HBLOCK,6X,3HKEY,/)
    DO 88 K=1,NN,4
    IF((K/92)*92 .EQ. K) GO TO 50
    GO TO 88
 50 PRINT 100
100 FORMAT(1H1)
    PRINT 29
    PRINT 30
 88 PRINT 20,LOCOUT(K),LOCOUT(K+1),LOCOUT(K+2),LOCOUT(K+3)
 20 FORMAT(10X,3I10,6X,A4,/)
    PRINT 10, T
 10 FORMAT(1H1,* TIME FOR RANDOM ACCESSES=*F9.3* SECONDS*)
    STOP
    END


    PROGRAM SYNBLOCK
C
C                          A SYNBAPS PROGRAM
C                          *******************
C
C THIS PROGRAM STRUCTURES A BLOCK(S) OF 5 DEG. SQ. GRIDDED BATHYMETRIC DATA ON
C  PERMANENT DISK FILE AND LOOKSUP RELATIVE ADDRESS (LOCATE) AND SIZE (NUM) FOR
C  EACH BLOCK
C
C REQUIRES SUBROUTINES DKOPEN, DKLOCATE, DKWRITE, DKREAD, AND DATE
C REQUIRES FUNCTION TIMELEFT
C
C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
    DIMENSION Z(63,116),SFD(7328),RALOC(2368),LOC(7328)
    REAL LOC
    TYPE INTEGER RALOC
    EQUIVALENCE(SFD,LOC)
```

```
      DATA (N=24)
      READ(60,5) ISET
    5 FORMAT(I5)
      DO 99 LL=1,ISET
      READ(60,30) MSQLOC,ICOL,IROW
   30 FORMAT(3I4)
      DO 6 J=1,ICOL
    6 READ(60,13) (Z(J,I),I=1,IROW)
   13 FORMAT(7F11.2)
      M=1
      DO 7 K=1,ICOL
      DO 7 L=1,IROW
      SFD(M) = Z(K,L)
    7 M=M+1
      M=M-1
      MM= ICOL * IROW
      IF(M.NE.MM)2,3
    2 PRINT 12,M,MSQLOC
   12 FORMAT(* ERROR-ONLY *I4* GRIDDED DATA POINTS CONVERTED FROM CARDS
     1FOR BLOCK NO. *I4//)
      GO TO 99
C     LOOKUP ADDRESS FOR GRIDDED DATA BASED ON MSQLOC
    3 CALL DATE(MONTH,IDAY,IYEAR,JULDAY)
      T1=TIMELEFT(0)
      CALL DKOPEN(5,3HRAN,4HEOBC)
      CALL DKLOCATE(128704)
      CALL DKREAD(RALOC(1),RALOC(2368))
      T2=TIMELEFT(0)
      NC=N*4
      DO 8 N=1,NC,4
      IF(MSQLOC.EQ. RALOC(N))9,8
    9 PRINT 2000, RALOC(N), RALOC(N+1),RALOC(N+2),RALOC(N+3)
 2000 FORMAT(1X,3I10,4X,A6)
      LOCATE = RALOC(N+1)
      NUM=RALOC(N+2)
      KEY= RALOC(N+3)
      GO TO 11
    8 CONTINUE
```

```
11 SFD(NUM-7)= FLOAT(NUM)
   SFD(NUM-6)=FLOAT(ICOL)
   SFD(NUM-5)=FLOAT(IROW)
   SFD(NUM-4)=FLOAT(MSQLOC)
   SFD(NUM-3)=FLOAT(IDAY)
   SFD(NUM-2)=FLOAT(MONTH)
   SFD(NUM-1)= 1900.+ FLOAT(IYEAR)
   SFD(NUM)=FLOAT(LOCATE)
   T3= TIMELEFT(0)
   CALL DKOPEN(5,3HRAN,KEY)
   CALL DKLOCATE(LOCATE)
   CALL DKWRITE(SFD(1),SFD(NUM))
   T4= TIMELEFT(0)
   GO TO(101,102,103,104,105,106,107,108,109,110,111,112)MONTH
101 MON=7HJANUARY   $ GO TO 113
102 MON=8HFEBRUARY  $ GO TO 113
103 MON=5HMARCH     $ GO TO 113
104 MON=5HAPRIL     $ GO TO 113
105 MON=3HMAY       $ GO TO 113
106 MON=4HJUNE      $ GO TO 113
107 MON=4HJULY      $ GO TO 113
108 MON=6HAUGUST    $ GO TO 113
109 MON=5HSEPT.     $ GO TO 113
110 MON=7HOCTOBER   $ GO TO 113
111 MON=8HNOVEMBER  $ GO TO 113
112 MON=8HDECEMBER  $ GO TO 113
113 PRINT 10,MSQLOC,LOCATE,IDAY,MON,IYEAR
 10 FORMAT(* FIVE DEGREE SQUARE *I4* WAS ADDED TO DISK FILE STARTING A
  1T RELATIVE ADDRESS=*I8* ON *I2,1X,A8*,19*I2//)
   T5= TIMELEFT(0)
   CALL DKLOCATE(LOCATE)
   CALL DKREAD(LOC(1),LOC(NUM))
   T6= TIMELEFT(0)
   NK=1
   NJ=15
   NU=(NUM/15)+1
   DO 89 I=1,NU
```

```
      PRINT 14, (LOC(JJ),JJ=NK,NJ)
   14 FORMAT(1X,15F9.2)
      NK=NK+15
   89 NJ=NK+14
      T=(T1-T2)+(T3-T4)+(T5-T6)
      PRINT 1001,T
 1001 FORMAT(* TIME FOR RANDOM ACCESS =*F9.3* SECONDS*//)
   99 CONTINUE
      PRINT 1000
 1000 FORMAT(* ***END OF RUN*** ABOVE BLOCKS HAVE BEEN ADDED TO DISK FIL
     1E*)
      STOP
      END
```

APPENDIX C

FORTRAN Programs for

Accessing SYNBAPS

All programs and subroutines listed in this appendix are subject to change without notice. Modifications within the programs and adoption of the system for other computers will necessitate major changes. The author should be contacted for the most recent versions of these programs.

```
      PROGRAM SYNBAPS1
C
C                                         A SYNBAPS PROGRAM
C                                         *******************
C
C MAIN ACCESSING PROGRAM TO GENERATE RANDOM, OMNIDIRECTIONAL BATHYMETRIC
C PROFILES ALONG A GREAT CIRCLE PATH.

C REQUIRES SUBROUTINES SEAARCH, MINCON, RHUMB, LOOKUP, BATHY, GRIDBLK, SPLINE,
C SPLICON, PUNOUT, MERFIX, DKOPEN, DKLOCATE, DKREAD, GCDIST, GCPATH, LALOCON,
C AND MSQLOC.
C REQUIRES FUNCTIONS AMP, TIMELEFT

C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      COMMON A(60),B(60),C(60),D(60),IE(60),IDBM,LINK,IMSQ(1000),
     1DEEP(1000),KNT,KKNT,MILES,Z(100,100)
      NOBEAM=0
  500 READ 1,NOOFBM,NCARD
    1 FORMAT(I5,4X,A8)
      IF(EOF,60) 501,502
  502 DO 99 I=1,NOOFBM
      T3=TIMELEFT(0)
      CALL SEAARCH(NCARD,K,IERROR)
      LINK=0
      IF(IERROR .EQ. 999) GO TO 99
      NOBEAM=NOBEAM+1
      DO 44 LL=1,K,2
      T1=TIMELEFT(0)
      KNT=LL
      KKNT=0
      CALL MINCON(ZLAT,ZLON)
      CALL RHUMB(ZBER)
      CALL LOOKUP(ICOL,IROW,LOCATE,NUM,KEY)
      IF(KKNT.EQ. 888) GO TO 98
C     PRINT 2000, ZLAT,ZLON,ZBER,IE(LL),IDBM,ICOL,IROW,LOCATE,NUM,KEY
 2000 FORMAT(1X,3F15.10,I8,2X,A6,4I10,A4,/)
```

```
      CALL BATHY(ZLON,ZLAT,ZBER,IE(LL),ICOL,IROW,LOCATE,NUM,KEY)
      IF(KKNT.EQ. 888) GO TO 98
      T2=TIMELEFT(0)
      T=T1-T2
      WRITE(61,100) T
  100 FORMAT(1X,28HTIME FOR BLOCK GENERATION = ,F10.3,8H SECONDS,/)
   44 CONTINUE
   98 T4=TIMELEFT(0)
      TT=T3-T4
      WRITE(61,101) TT
  101 FORMAT(1X,36HTOTAL TIME FOR PROFILE GENERATION = ,F10.3,8H SECONDS
     1,/)
   99 CONTINUE
      GO TO 500
  501 ENDFILE 10
      REWIND 10
      PRINT 2001,NOBEAM
 2001 FORMAT(1H1,* END OF COMPUTER RUN*I10* BATHYMETRIC PROFILES PROCESS
     1ED*,////)
 2002 STOP
      END
```

```
      SUBROUTINE SEAARCH(NCARD,K,IERROR)
C
C                                       A SYNBARS SUBROUTINE
C                                       **********************
C
C ROUTINE GENERATES GREAT CIRCLE PATH AND CREATES RANGE SEARCH TABLE OF MSQLOC
C AREAS.
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION DIST(1000),FINLAT(1000),FINLON(1000),FINBER(1000),AL(5),
     1LA(1)
      COMMON A(60),B(60),C(60),D(60),IE(60),IDBM,LINK,IMSQ(1000),
     1DUMMY(1002),N,DUMDUM(10000)
      TYPE INTEGER AN,AE,BN,BE
      IERROR=0
      IF(NCARD .EQ. 8HPOINTS  ) GO TO 500
      IF(NCARD .EQ. 8HBEARINGS) GO TO 499
      WRITE(61,498)
  498 FORMAT(1X,*ERROR - NO INDICATION OF CONTROL CARD TYPE GIVEN - AN A
     1LPHANUMERIC FIELD OF EITHER POINTS OR BEARINGS STARTING IN COLUMN
     210*/*  IS REQUIRED FOR EACH BEAM SET*//)
      IERROR=999
      RETURN
  499 READ(60,1) IDBM,ALAT,AMIN,AN,ALONG,ALMIN,AE,BS,DD
    1 FORMAT(A6,2X,2(F3.0,F 3.0,1X,A1,1X),2F10.0)
      PAS=ALAT+AMIN/60.
      POS=ALONG+ALMIN/60.
      IF(AN.EQ. 1HS) PAS=-PAS
      IF(AE.EQ. 1HW) POS=-POS
      GO TO 501
  500 READ 2, IDBM,ALAT,AMIN,AN,ALONG,ALMIN,AE,BLAT,BMIN,BN,BLONG,
     1BLMIN,BE
    2 FORMAT(A6,2X,4(F3.0,F3.0,1X,A1,1X))
      PAS=ALAT+AMIN/60.
      POS=ALONG+ALMIN/60.
      PAF=BLAT+BMIN/60.
```

```
      POF=BLONG+BLMIN/60.
      IF(AN.EQ. 1HS) PAS=-PAS
      IF(AE.EQ. 1HW) POS=-POS
      IF(BN.EQ. 1HS) PAF=-PAF
      IF(BE.EQ. 1HW) POF=-POF
   17 CALL GCDIST(PAS,POS,PAF,POF,BS,BF,DD)
  501 N= INT(DD+.5)
   55 IF(N.LT. 8000) GO TO 19
   18 PRINT 3,IDBM,N
    3 FORMAT(* ERROR-MAXIMUM RANGE EXCEEDED IN BEAM NUMBER *A6,
     1I10,*NAUTICAL MILES*)
      IERROR=999
      RETURN
   19 IF((N/1000)*1000 .EQ. N) GO TO 20
      NUM=(N/1000)+1
      GO TO 21
   20 NUM= N/1000
   21 K=2
      KK=1
      DO 33 MT=1,NUM
      DIST(1)= FLOAT(MT*1000)-1000.
      DO 331 J=2,1000
      DIST(J)= DIST(J-1)+1.0
      NT=J
      IF(N.EQ. INT(DIST(J)))GO TO 332
  331 CONTINUE
  332 CALL GCPATH(PAS,POS,BS,DIST,NT,FINLAT,FINLON,FINBER)
      DO 88 M=1,NT
      CALL LALOCON(FINLAT(M),FINLON(M),IDBM,FLAT,FLON,FLATM,FLONM,NORT,
     1 IEST)
      FINLAT(M)=FLAT+(FLATM/60.)
      IF(NORT .EQ. 1HS) FINLAT(M)= -FINLAT(M)
      FINLON(M)= FLON+(FLONM/60.)
      IF(IEST .EQ. 1HW) FINLON(M)= -FINLON(M)
      LAT= INT(FINLAT(M))
      LONG= -(INT(FINLON(M)))
      IF(LAT)90,91,90
```

```
   91 IF(FINLAT(M))93,90,90
   93 LAT=LAT-1
   90 IF(LONG)25,92,25
   92 IF(FINLON(M))25,25,94
   94 LONG=LONG-1
   25 CALL MSQFQ(LAT,LONG,MSQ,MSQ5,MSQ1)
      IMSQ(M)=(MSQ*10)+MSQ5
   88 CONTINUE
      IF(MT .GT. 1) GO TO 57
   56 A(1)=FINLAT(1)
      B(1)=FINLON(1)
      C(1)=FINBER(1)
      D(1)=DIST(1)
      IE(1)=IMSG(1)
   57 DO 77 J=2,NT
      IF(IE(KK).EQ.IMSQ(J)) GO TO 77
   60 A(K)=FINLAT(J-1)
      B(K)=FINLON(J-1)
      C(K)=FINBER(J-1)
      D(K)=DIST(J-1)
      IE(K)=IMSG(J-1)
      A(K+1)=FINLAT(J)
      B(K+1)=FINLON(J)
      C(K+1)=FINBER(J)
      D(K+1)=DIST(J)
      IE(K+1)=IMSQ(J)
      KK=K+1
      K=K+2
   77 CONTINUE
      IF(MT .EQ. NUM ) GO TO 58
C     GO TO 61
      GO TO 33
   58 A(K)=FINLAT(NT)
      B(K)=FINLON(NT)
      C(K)=FINBER(NT)
      D(K)=DIST(NT)
      IE(K)=IMSQ(NT)
C  61 PRINT 1000,IDBM
```

```
 1000 FORMAT(27X,40HINDIVIDUAL RANGE POINTS FOR BEAM NUMBER ,A6//)
C      PRINT 1001
 1001 FORMAT(20X*LATITUDE   LONGITUDE   FINAL BEARING   RANGE N.M.  MSQL
     1OC*/)
C      DO 33 L=1,NT
C      CALL LALOCON(FINLAT(L),FINLON(L),IDBM,FLAT,FLON,FLATM,FLONM,NORT,
C     1IEST)
C      PRINT 1002,FLAT,FLATM,NORT,FLON,FLONM,IEST,FINBEH(L),DIST(L),
C     1IMSQ(L)
 1002 FORMAT(20X,2(F3.0,F3.0,1X,A1,2X),F17.9,F9.0,I10)
   33 CONTINUE
      KNT=K-1
      PRINT 1003,IDBM
 1003 FORMAT(1H1,30X*RANGE SEARCH TABLE FOR BEAM NUMBER *A6//)
      PRINT 1001
      DO  200 JJ=1,K
      CALL LALOCON(A(JJ),B(JJ),IDBM,FLAT,FLON,FLATM,FLONM,NORT,IEST)
      PRINT 1002,FLAT,FLATM,NORT,FLON,FLONM,IEST,C(JJ),D(JJ),IE(JJ)
  200 CONTINUE
   99 RETURN
      END
```

```
      SUBROUTINE GRIDBLK(MSQLOC,ICOL,IROW,LOCATE,NUM,KEY,IOMIT)
C
C                                     A SYNBAPS SUBROUTINE
C                                    **********************
C
C ROUTINE EXTRACTS MSQLOC AREA DATA BLOCK FROM DISK

C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION ZD(8000)
      COMMON DUMMY(300),IBEAM,DUMDUM(2002),KKNT,MILES,Z(100,100)
      IOMIT=0
      CALL DKOPEN(5,3HRAN,KEY)
  205 CALL DKLOCATE(LOCATE)
      CALL DKREAD(ZD(1),ZD(NUM))
      K=1
      DO 1 J=1,ICOL
      DO 1 I=1,IROW
      Z(J,I)=ZD(K)
      K=K+1
    1 CONTINUE
      IF(NUM.GT.ICOL*IROW) GO TO 208
  206 PRINT 207,MSQLOC,IBEAM
  207 FORMAT(* ERROR- DATA BLOCK NOT UNPACKED CORRECTLY FROM DISK FOR BL
     1OCK NO. *I4* FOR BEAM NO. *A6//)
      IOMIT= 999
      KKNT= 888
C 208 DO 300 J=1,ICOL
C 300 PRINT 10,(Z(J,I),I=1,IROW)
   10 FORMAT(1X,15F9.2)
  208 RETURN
      END
```

```
      SUBROUTINE MINCON(R,S)
C
C                                  A SYNBAPS SUBROUTINE
C                                  *********************
C
C ROUTINE CALCULATES ENTRY POINT OF MSQLOC AREA IN MINUTES OF X,Y FROM LOWER
C LEFT CORNER
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      COMMON A(60),B(60),DUMMY(2182),I,DUMDUM(10002)
      DATA(CON5= 5.08333),(CON6= 0.08333)
      S=AINT(((ABSF(B(I))-((AINT(ABSF(B(I))*0.1))*10.))*60.)+.5)
    4 IF(S.GE. 300.) GO TO 5
      S=S+5.0
      GO TO 6
    5 S=S-295.0
    6 IF(B(I)) 7,8,8
    7 S=310. -S
    8 AA=ABS(AINT(A(I)*0.1)*10.0)
      IF((ABS(A(I))-AA).GE. 5.0) AA=AA+5.0
      IF(A(I)) 9,10,10
    9 ALAT=(-(AA+CON5))
      GO TO 11
   10 ALAT=AA-CON6
   11 T = A(I)
      R= AINT((AMP(T)-AMP(ALAT))+0.5)
      RETURN
      END
```

```
      SUBROUTINE RHUMB(AZ)
C
C                              A SYNBAPS SUBROUTINE
C                              *********************
C
C ROUTINE CALCULATES RHUMB LINE BEARING FOR MSQLOC AREA
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      COMMON A(60),B(60),C(60)     ,D(60),DUM(2062),K,DUMMY(10002)
      REAL HP
      DATA (CON3= 100000.0),(CON4= 0.00001),(D2R= 0.0174532925199)
      K2=K+1
      AA= ABS(AMP(A(K2))-AMP(A(K)))
      BB= ABS(B(K2)-B(K))*60.
      HP=SQRT((AA**2)+(BB**2))
      CC=AA/HP
      IF(CC.GT. 1.0) CC=1.0
      AZ= ASIN(CC)/D2R
      IF(C(K).GE. 0.0.AND.C(K).LT.90.0) AZ=90.0-AZ
      IF(C(K).GE.90..AND.C(K).LT.180.) AZ=90.0+AZ
      IF(C(K).GE.180..AND.C(K).LT.270.) AZ=270.-AZ
      IF(C(K).GE.270..AND.C(K).LT.360.) AZ=270.+AZ
      AZ= ABS(AINT((AZ*CON3)+0.5)*CON4)
      RETURN
      END
```

```
      SUBROUTINE LOOKUP(ICOL,IROW,LOCATE,NUM,KEY)
C
C                                      A SYNBAPS SUBROUTINE
C                                      **********************
C
C ROUTINE EXTRACTS NEEDED PARAMETERS FROM THE LOOKUP TABLE
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION C(32),IB(2368)
      COMMON DUMMY(240),IE(60),IDBM,DUMDUM(2001),IA,KKNT,DUM(10001)
      DATA (NN=24)
      CALL DKOPEN(5,3HRAN,4HE08C)
      CALL DKLOCATE(128704)
      CALL DKREAD(IB(1),IB(2368))
      NC=NN*4
      DO 1 K=1,NC,4
      KK=K
      IF(IE(IA) .EQ. IB(K)) GO TO 2
    1 CONTINUE
      WRITE(61,10) IE(IA),IDBM
   10 FORMAT(* DATA BLOCK NOT FOUND ON DISK FOR BLOCK NO. *I4* FOR BEAM
     1NO. *A6/* THIS BEAM WILL TERMINATE HERE.  */* RUN WILL CONTINUE IF
     2 FURTHER BEAMS REQUIRE PROFILING.*)
      KKNT= 883
      RETURN
    2 N=IB(KK+2)
      KEY=IB(KK+3)
      LOCATE=IB(KK+1)
      L=IB(KK+1)+(N-32)
      CALL DKOPEN(5,3HRAN,KEY)
      CALL DKLOCATE(L)
      CALL DKREAD(C(1),C(32))
      NUM= INT(C(25))
      ICOL= INT(C(26))
      IROW= INT(C(27))
      IF(NUM.NE. N) GO TO 20
```

```
      RETURN
   20 WRITE(61,30) IE(IA),IDBM
   30 FORMAT(1X,22HERROR - FOR BLOCK NO. ,I4,13HFOR BEAM NO. ,A6,4X,
     157HBLOCK SIZES DO NOT MATCH BETWEEN TABLE AND STORAGE BLOCK )
      WRITE(61,10) IE(IA),IDBM
      KKNT=888
      RETURN
      END


      FUNCTION AMP(Y)
C
C                                  A SYNBAPS SUBROUTINE
C                                  *********************
C
C ROUTINE CALCULATES MERIDIONAL PARTS FOR ANY LATITUDE POINT

C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DATA (D2R= 0.0174532925199),(CON1= 7915.7045),(CON2= 23.268932)
      AP= 45.*D2R
      X= ABS(Y)*D2R
      TEMP= SIN(AP+X/2.0)/ COS(AP+X/2.0)
      AMP  = CON1*ALOG10(TEMP)-CON2* SIN(X)
      RETURN
      END
```

```
      SUBROUTINE BATHY(AX,AY,AANGLE,MSQLOC,ICOL,IROW,LOCATE,NUM,KEY)
C
C                           A SYNBAPS SUBROUTINE
C                           *********************
C
C     ROUTINE TO COMPUTE BATHYMETRIC PROFILES BY APPLICATION OF A CUBIC
C     SPLINE INTERPOLATION PROCEDURE TO GRIDDED BATHYMETRIC DATA.
C     OUTPUT IS UP TO 8 PROFILES. DATA BASE IS ASSUMED GRIDDED WITH AN
C     INTERVAL OF BGRID IN X-Y UNITS AND IS READ IN COLUMNWISE STARTING
C     WITH ORIGIN AT LL CORNER. IN CONTROL CARD,AX,AY IS ORIGIN FOR
C     PROFILES,ANGLE IS DIRECTION OF DESIRED PROFILES IN DECIMAL DEG.
C     CLOCKWISE FROM NORTH,PINT=DESIRED DATA SPACING ALONG PROFILES IN
C     X-Y UNITS,IROW,ICOL IS NO.OF ROWS AND COLS.OF INPUT DATA.
C
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION               ANGLE(9),SLOPE(4,8),TINDP(150),DIST(8,150),
     1VALUE(8,150),XPLOT(900),YPLOT(900),YLAB(4,8),TDV(150)
      COMMON DUMMY(2305),Z(100,100)
      DATA (ANGLE(2)=999.9),(ANGLE(9)=999.9),(ATER=9999.9),(PINT=1.0),
     1(BGRID=5.0)
      ANGLE(1)=AANGLE
      J1=0
      J2=0
      J3=0
      J4=0
C     CONVERT ANGLES TO SLOPES
      DO 2 J=1,8
      JA=J+1
      IF(ANGLE(J).GE.225.0.AND.ANGLE(J).LE.315.0)GO TO 3
      IF(ANGLE(J).GE.45.0.AND.ANGLE(J).LE.135.0) GO TO 4
      IF(ANGLE(J).GT.135.0.AND.ANGLE(J).LT.225.0) GO TO 5
      IF(ANGLE(J).EQ.0.0) ANGLE(J)= ANGLE(J)+.01
      J1=J1+1
      SLOPE(1,J1)=TANF(7.854 -ANGLE(J)/57.2958)
      YLAB(1,J1)=ANGLE(J)
      IF(ANGLE(JA).EQ.999.9) GO TO 41
```

```
      GO TO 2
    3 J4=J4+1
      SLOPE(4,J4)=TANF(7.854 -ANGLE(J)/57.2958)
      YLAB(4,J4)=ANGLE(J)
      IF(ANGLE(JA).EQ.999.9) GO TO 41
      GO TO 2
    4 J2=J2+1
      SLOPE(2,J2)=TANF(7.854 -ANGLE(J)/57.2958)
      YLAB(2,J2)=ANGLE(J)
      IF(ANGLE(JA).EQ.999.9) GO TO 41
      GO TO 2
    5 J3=J3+1
      SLOPE(3,J3)=TANF(7.854 -ANGLE(J)/57.2958)
      YLAB(3,J3)=ANGLE(J)
      IF(ANGLE(JA).EQ.999.9) GO TO 41
    2 CONTINUE
C     READ IN GRIDDED DATA BASE BY COLUMNS STARTING AT LL CORNER
   41 CALL GRIDBLK(MSQLOC,ICOL,IROW,LOCATE,NUM,KEY,IOMIT)
      IF(IOMIT .EQ. 999) GO TO 100
C     DO 300 J=1,ICOL
C 300 PRINT 10,(Z(J,I),I=1,IROW)
   10 FORMAT(1X,15F9.2)
      IQUAD=4
C     COMPUTE INTERSECTIONS OF PROFILES IN QUADRANT (IQUAD) WITH EACH
C     COL FROM DATA BASE,USE SPLINE TO INTERPOLATE VALUE WITH
C     INDEP.VARIABLE DISTANCE DOWN PROFILE FROM AX,AY POSITION
      KTEMP=1
      JCT=J4
      LROW=IROW
      IF(JCT.EQ.0) GO TO 73
      IACOL=1
      GO TO 54
   73  IACOL=(AX/BGRID)+1.0
      GO TO 63
   54 DO 7 J=1,IROW
      A=J-1
    7 TINDP(J)=A*BGRID
```

```
      AJT=IROW-1
      YMAX=AJT*BGRID
   12 DO 11 J=1,IROW
   11 TDV(J)=Z(IACOL,J)
      ACOL=IACOL-1
   33 DO 8 I=1,JCT
      IF(IQUAD.EQ.4.OR.IQUAD.EQ.2) GO TO 44
      AINT= AX+(1.0/SLOPE(IQUAD,I))*((ACOL*BGRID)-AY)
      DIST(I,KTEMP)=SQRT((AINT-AX)**2+((ACOL*BGRID)-AY)**2)
      GO TO 35
   44 AINT=SLOPE(IQUAD,I)*((ACOL*BGRID)-AX)  +AY
      DIST(I,KTEMP)=SQRT((ACOL*BGRID-AX)**2 +(AINT-AY)**2)
   35 IF(AINT.LT.0.0.OR.AINT.GT.YMAX) GO TO 9
      CALL SPLINE(TINDP,TDV,LROW,AINT,YINT,ATER)
      VALUE(I,KTEMP)=YINT
      GO TO 8
    9 VALUE(I,KTEMP) = 3500.
    8 CONTINUE
      IF(IQUAD.NE.4) GO TO 23
   25 IF((ACOL    )*BGRID.GT.AX)GO TO 213
      GO TO 27
  213 DO 214 I=1,JCT
  214  DIST(I,KTEMP)= -1.0* DIST(I,KTEMP)
      GO TO 13
   27 IACOL=IACOL +1
      KTEMP=KTEMP+1
      GO TO 12
   23 IF(IQUAD.NE.3) GO TO 24
      IF((ACOL    )*BGRID.GT.AY) GO TO 213
   37 IACOL=IACOL +1
      KTEMP =KTEMP+1
      GO TO 34
   24 IF(IQUAD.EQ.2) GO TO 26
      IF(IACOL.GE.IROW) GO TO 313
      GO TO 37
   26 IF(IACOL.GE.ICOL) GO TO  313
      GO TO 27
  313  DO 314 I=1,JCT
```

```
  314 DIST(I,1)= -1.0* DIST(I,1)
      GO TO 13
C     NOW INTERPOLATE ALONG EACH PROFILE TO OBTAIN POINTS FOR SMOOTH
C     PROFILE OUTPUT
   13 NTOT =KTEMP
      DO 14 I=1,JCT
      IF(IQUAD.LT.3) GO TO 21
      NCT=1
      GO TO 19
   21 NCT=NTOT
   19 JTOT=(DIST(I,NCT)/PINT) +1.0
C     PRODUCE 1 DIMENSIONAL DATA FOR SPLINE INPUT
      DO 16 K=1,NTOT
      IF(IQUAD.LT.3) GO TO 17
      INTOT=NTOT +1 -K
      GO TO 18
   17 INTOT=K
   18 TINDP(K)=DIST(I,INTOT)
   16 TDV(K)= VALUE(I,INTOT)
C     NOW PRODUCE PROFILE POINTS AND PLOT
      DO 15 J=1,JTOT
      A=J-1
      A=A*PINT
      CALL SPLINE (TINDP,TDV,NTOT,A,YINT,ATER)
      XPLOT(J)=A
   15 YPLOT(J)=YINT
C  WRITE OUTPUT TAPE FOR PLOTTING AND PUNCHED CARDS     LU=10
C     WRITE(61,500) (XPLOT(IT),YPLOT(IT),IT=1,JTOT)
  500 FORMAT(1X,12F10.2)
      CALL PUNOUT(XPLOT,YPLOT,JTOT,PINT)
   14 CONTINUE
C     AT THIS POINT ALL PROFILES FROM QUADRANT IQUAD HAVE BEEN PLOTTED
C     NOW SET UP FOR NEXT QUADRANT,ORDER IS 4,2,3,1
      IF(IQUAD.NE.4) GO TO 22
   63  IQUAD=2
      JCT=J2
      LROW=IROW
```

```
      IF(JCT.EQ.0) GO TO 22
      IACOL=IACOL-1
      KTEMP=1
      GO TO 54
   22 IF(IQUAD.EQ.3) GO TO 36
      IF(IQUAD.EQ.1) GO TO 100
      IQUAD=3
      JCT=J3
      LROW=ICOL
      IF(JCT.EQ.0) GO TO 74
      KTEMP=1
      IACOL=1
      GO TO 53
   74  IACOL=(AY/BGRID)+1.0
      GO TO 36
   53 DO 31 J=1,ICOL
      A=J-1
   31 TINDP(J)=A*BGRID
      YMAX= TINDP(ICOL)
   34 DO 32 J=1,ICOL
   32 TDV(J)=Z(J,IACOL)
      ACOL=IACOL-1
      GO TO 33
   36 IF (IQUAD.EQ.1) GO TO 100
      IQUAD=1
      JCT=J1
      IF(JCT.EQ. 0) GO TO 100
      LROW=ICOL
      KTEMP=1
      IACOL=IACOL-1
      GO TO 53
  100 RETURN
      END
```

```
      SUBROUTINE PUNOUT(X,Y,N,P)
C
C
C                                      A SYNBAPS SUBROUTINE
C                                      **********************
C ROUTINE WRITES PROFILE SEGMENTS ON TO TEMPORARY MAGNETIC TAPE AND SCALES
C PROFILES INTO NAUTICAL MILES
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION X(N),Y(N)
      COMMON DUMMY(180),D(60),IDUM(60),IDBM,LINK,IRANGE(1000),
     1DEPTH(1000),K,KK,MILES,DUMDUM(10000)
      DIST=AINT(D(K+1)-D(K))
      CALL MERFIX(DF,AHP)
      N= INT(AHP) +3
      DO 14 LL=1,N
   14 X(LL)=(X(LL)*DF)
      YHOLD=Y(1)
      DZ=1.0
      XHOLD=0.0
      NN=N-1
      IK=1
      DO 15 LK=2,NN
      IF(X(LK) .LE. DZ .AND. X(LK+1) .GE. DZ) GO TO 16
      GO TO 15
   16 Y(IK)=YHOLD
      X(IK)= XHOLD
      IK=IK+1
      YHOLD=Y(LK)+(((DZ-X(LK))/(X(LK+1)-X(LK)))*(-1.0*(Y(LK)-Y(LK+1))))
      XHOLD=DZ
      DZ=DZ+1.0
   15 CONTINUE
      N=IK
   13 IF(X(N).EQ.DIST) GO TO 12
      N=N-1
      GO TO 13
```

```
   12 L=1
      DO 3 J=1,N
      IRANGE(L)=X(J)
      DEPTH(L)=Y(J)
      I=LINK+L
    3 L=L+1
      LINK=I
      L=L-1
C   8 WRITE(61,100)IDBM,L,IRANGE(L)
  100 FORMAT(A6,2I10)
C     WRITE(61,200)(IRANGE(I),DEPTH(I),I=1,L)
  200 FORMAT(8(I4,1X,F4.0,1X))
      WRITE(10,100)IDBM,L,IRANGE(L)
      WRITE(10,200)(IRANGE(I),DEPTH(I),I=1,L)
      RETURN
      END
```

```
      SUBROUTINE MERFIX(DF,HP)
C
C                                   A SYNBARS SUBROUTINE
C                                   *********************
C
C ROUTINE CALCULATES RHUMB LINE DISTANCE AND SCALING FACTOR

C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      COMMON A(60),B(60),DUMDUM(60),D(60),DUM(2062),K,DUMMY(10002)
      REAL HP,KP
      K2=K+1
      R= AMP(A(K2))
      S= AMP(A(K))
      AA= R-S
      BB= ABS(B(K2)-B(K))*60.
      KP=SQRT((AA**2)+(BB**2))
      HP= AINT(KP+0.5)
      DIST=AINT(D(K2 )-D(K))
      DF=DIST/KP
C     WRITE(61,20) HP,S,R,DF,DIST
   20 FORMAT(1X,5F15.5)
      RETURN
      END
```

```
      SUBROUTINE SPLINE (X,Y,M,XINT,YINT,ATER)
C
C                                A SYNBAPS SUBROUTINE
C                                *********************
C
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 0610
C
C     SEE PENNINGTON REF. FOR DESCRIPTION OF THIS SUBROUTINE
C
      DIMENSION X(300),Y(300),C(4,400)
      IF(X(1)+Y(M)+Y(M-1)+X(M-1)+Y(M-2)-ATER) 10,3,10
   10 CALL SPLICON(X,Y,M,C)
      ATER= X(1)+Y(M)+Y(M-1)+X(M-1)+Y(M-2)
      K=1
    3 IF(XINT-X(1)) 70,1,2
   70 K=1
      GO TO 7
    1 YINT=Y(1)
      RETURN
    2 IF(XINT-X(K+1))6,4,5
    4 YINT=Y(K+1)
      RETURN
    5 K=K+1
      IF(M-K) 71,71,3
   71 K=M-1
      GO TO 7
    6 IF(XINT-X(K))13,12,11
   12 YINT=Y(K)
      RETURN
   13 K=K-1
      GO TO 6
   11 YINT=(X(K+1)-XINT)*(C(1,K)*(X(K+1)-XINT)**2+C(3,K))
      YINT=YINT+(XINT-X(K))*(C(2,K)*(XINT-X(K))**2+C(4,K))
      RETURN
    7  PRINT 101, XINT
  101 FORMAT(* CAUTION VALUE AT POSITION*,F10.2,* WAS EXTRAPOLATED*)
      GO TO 11
      END
```

```
      SUBROUTINE SPLICON(X,Y,M,C)
C
C                                      A SYNBAPS SUBROUTINE
C                                      **********************
C
C ROUTINE WRITTEN BY T.M. DAVIS,NAVOCEANO,GATP,CODE 061D
C
      DIMENSION X(300),Y(300),C(4,300),D(400),P(400),E(400),A(400,3),B(4
     100),Z(400)
      MM=M-1
      DO 2 K=1,MM
      D(K)=X(K+1)-X(K)
      P(K)=D(K)/6.
    2 E(K)=(Y(K+1)-Y(K))/D(K)
      DO 3 K=2,MM
    3 B(K)=E(K)-E(K-1)
      A(1,2)=-1.-D(1)/D(2)
      A(1,3)=D(1)/D(2)
      A(2,3)=P(2)-P(1)*A(1,3)
      A(2,2)=2.*(P(1)+P(2))-P(1)*A(1,2)
      A(2,3)=A(2,3)/A(2,2)
      B(2)=B(2)/A(2,2)
      DO 4 K=3,MM
      A(K,2)=2.*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
      B(K)=B(K)-P(K-1)*B(K-1)
      A(K,3)=P(K)/A(K,2)
    4 B(K)=B(K)/A(K,2)
      Q=D(M-2)/D(M-1)
      A(M,1)=1.+Q+A(M-2,3)
      A(M,2)=-Q-A(M,1)*A(M-1,3)
      B(M)=B(M-2)-A(M,1)*B(M-1)
      Z(M)=B(M)/A(M,2)
      MN=M-2
      DO 6 I=1,MN
      K=M-I
    6 Z(K)=B(K)-A(K,3)*Z(K+1)
      Z(1)=-A(1,2)*Z(2)-A(1,3)*Z(3)
```

```
      DO 7 K=1,NM
      Q=1./(6.*D(K))
      C(1,K)=Z(K)*Q
      C(2,K)=Z(K+1)*Q
      C(3,K)=Y(K)/D(K)-Z(K)*P(K)
    7 C(4,K)=Y(K+1)/D(K)-Z(K+1)*P(K)
      END


       SUBROUTINE MSQFQ(LAT,LONG,MSQ,MSQ5,MSQ1)
C
C                                           A SYNBAPS SUBROUTINE
C                                          **********************
C
C ROUTINE CALCULATES MARSDEN SQUARE NUMBER, FIVE DEGREE SQUARE NUMBER AND THE
C ONE DEGREE SQUARE NUMBER

C NORTH AND WEST ARE POSITIVE, SOUTH AND EAST ARE NEGATIVE

C ROUTINE WRITTEN BY OSCAR JACKSON,NAVOCEANO,CODE 08
C
      IF(LAT)70,71,20
   20 IF(LONG)75,76,40
   71 IF(LAT.AND.4000000000000000B)70,20
   70 LAT=IABS(LAT)
      GO TO 10
   76 IF(LONG.AND.4000000000000000B)75,40
   75 LONG=IABS(LONG)
      GO TO 30
C       QUARDRANT 2
   40 MSQ=36*(LAT/10)+LONG/10+1
      GO TO 60
   10 IF(LONG)90,95,55
   95 IF(LONG.AND.4000000000000000B)90,55
   90 LONG=IABS(LONG)
   50 MSQ=36*(LAT/10)-LONG/10+335
```

```
      GO TO 60
   55 MSQ=36*(LAT/10)+LONG/10+300
      GO TO 60
   30 MSQ=36*(LAT/10)-LONG/10+36
   60 IF(LAT.GT.79)61,62
   61 MSQ=MSQ+612
   62 MSQ5=1
      LTC=LAT-(LAT/10)*10
      LG1=LONG-(LONG/100)*100
      LG=LG1-(LG1/10)*10
      IF(LTC.LT.5)80,81
   80 IF(LG.GT.4)82,83
   82 MSQ5=2
      GO TO 83
   81 MSQ5=3
      IF(LG.LT.5)83,84
   84 MSQ5=4
   83 MSQ1=LTC*10+LG
      RETURN
      END

      SUBROUTINE LALOCON(FINLAT,FINLON,IDBM,FLAT,FLON,FLATM,FLONM,NORT,
C
C                                A SYNBAPS SUBROUTINE
C                                ********************
C
C A ROUTINE TO CONVERT INTERNAL LAT AND LONG TO DEGREEES,MINUTES AND HEMI-
C SPHERE FOR PRINTER OUTPUT.  ERROR MESSAGE VARIES WITH APPLICATION
C NORTH AND EAST ARE POSITIVE
C SOUTH AND WEST ARE NEGATIVE
C
C ROUTINE WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
     1IEST)
      DIMENSION FINLAT(1),FINLON(1)
      FLAT= ABSF(AINT(FINLAT))
```

```
      FLON= ABSF(AINT(FINLON))
      FLATM=AINT(((ABSF(FINLAT)-FLAT)*60.)+.5)
      FLONM=AINT(((ABSF(FINLON)-FLON)*60.)+.5)
      IF(FLATM-60.) 11,10,11
   10 FLAT=FLAT+1.0
      FLATM=0.0
   11 IF(FLONM-60.) 13,12,13
   12 FLON=FLON+1.0
      FLONM=0.0
   13 IF(FINLAT)100,101,102
  100 NORT=1HS
      GO TO 104
  102 NORT=1HN
      GO TO 104
  101 PRINT 103,IDBM
  103 FORMAT(* ERROR IN QUADRANT OUTPUT FOR BEAM NUMBER *A6* POINT ASSIG
     1NED TO QUADRANT ONE OR FOUR*)
      NORT=1HN
      IF(FINLON.NE. 0.0) GO TO 104
      IEST=1HE
      RETURN
  104 IF(FINLON)105,101,107
  105 IFST=1HW
      RETURN
  107 IEST=1HE
      RETURN
      END
```

```
      SUBROUTINE GCPATH(PAS,POS,BS,DIST,N,FINLAT,FINLON,FINBER)
C     IDENT NUMBER - T0002000
C     TITLE - GREAT CIRCLE PATH FROM A POINT
C     IDENT NAME - TO-NRL-GCPATH
C     LANGUAGE - FORTRAN
C     COMPUTER - CDC-3800
C     CONTRIBUTOR - DAVID CHANG, CODE 8170, PROPAGATION BRANCH,
C                              ACOUSTICS DIVISION
C     ORGANIZATION - NRL - NAVAL RESEARCH LABORATORY -
C                              WASHINGTON, D.C.   20390
C     DATE - 22 JULY 1969
C     PURPOSE - GIVEN A GREAT CIRCLE PATH SPECIFIED BY AN INITIAL
C     POINT AND BEARING, THIS SUBROUTINE FINDS THE LOCATIONS AND
C     BEARINGS OF POINTS AT A GIVEN ARRAY OF DISTANCES IN NAUTICAL
C     MILES ALONG THAT PATH.
C
C     THE EARTH IS A SPHERE WITH CIRCUMFERENCE 21,600. MILES
C
C     ALL ANGLES ARE IN FLOATING POINT DEGREES.
C     LATITUDES .GE. 90. ARE AT THE NORTH POLE
C     LATITUDES .LE.-90. ARE AT THE SOUTH POLE
C     ALL LONGITUDES MUST BE BETWEEN -180. (180 W) AND +180. (180 E).
C     ALL BEARINGS NOT AT POLES ARE BETWEEN 0. AND 360. DEGREES,
C     MEASURED CLOCKWISE FROM DUE NORTH.
C     ALL BEARINGS AT THE POLES ARE LONGITUDE LINES.
C     FOR DISTANCES .LE. 0. OR .GE. CIRCUMFERENCE, THE FINAL POINT IS
C     THE INITIAL POINT.
C
C     PAS = INITIAL LATITUDE
C     POS = INITIAL LONGITUDE
C     BS  = INITIAL BEARING
C     DIST   = ARRAY OF N DESIRED DISTANCES IN NAUTICAL MILES
C     N      = DIMENSION OF FOUR ARRAYS
C     FINLAT = ARRAY OF FINAL LATITUDES
C     FINLON = ARRAY OF FINAL LONGITUEDS
C     FINBER = ARRAY OF FINAL BEARINGS
C
```

```
      DIMENSION DIST(N),FINLAT(N),FINLON(N),FINBER(N)
      DATA(DTOR=17724357506504518)
      DATA(RTOD=20007122734064628)
      DATA(RADEARTH=20146555576261378)
      DATA(AMILPDEG=60.)
      IF(PAS .GE. 90.)  GO TO 101
      IF(PAS .LE.-90.)  GO TO 151
C  INITIAL POINT NOT AT POLES.
      IF(BS. EQ.0.)  GO TO 201
      IF(BS. EQ.180.)  GO TO 251
C  GREAT CIRCLE USED DOES NOT PASS THROUGH POLES
      DUMMY=BS*DTOR
      CBS=COS(DUMMY)
      LEFT=0
      IF(BS.GT.180.)  LEFT=1
      DUMMY=PAS*DTOR
      CS=SIN(DUMMY)
      SS=COS(DUMMY)
      DO 60  I=1,N
      DA=DIST(I)/AMILPDEG
      IF(DA.NE.180.)  GO TO 20
      FINLAT(I)=-PAS
      FINLON(I)=POS+180.
      IF(FINLON(I).GT.180.)  FINLON(I)=FINLON(I)-360.
      FINBER(I)=180.-BS
      IF(LEFT)  FINBER(I)=FINBER(I)+360.
      GO TO 60
20    IF(DA.LT.360. .AND. DA.GT.0.)  GO TO 30
      FINLAT(I)=PAS
      FINLON(I)=POS
      FINBER(I)=BS
      GO TO 60
30    D=DIST(I)/RADEARTH
      CD=COS(D)
      SD=SIN(D)
      CF=CS*CD+SS*SD*CBS
      PAF=ASIN(CF)
```

```
      SF=COS(PAF)
      CBF=(CF*CD-CS)/SF/SD
      CAP=(CD-CS*CF)/SS/SF
      BF=ACOS(CBF)*RTOD
      APO=ACOS(CAP)*RTOD
      IF(SD.LT.0.)  APO=360.-APO
      IF(LEFT)  GO TO 40
      DUMMY=POS+APO
      IF(DUMMY.GT. 180.)  DUMMY=DUMMY-360.
      GO TO 50
40    DUMMY=POS-APO
      IF(DUMMY.LT.-180.)  DUMMY=DUMMY+360.
      BF=360.-BF
50    FINLON(I)=DUMMY
      FINBER(I)=BF
      FINLAT(I)=PAF*RTOD
60    CONTINUE
      RETURN
101   DO 120  I=1,N
C  INITIAL POINT AT NORTH POLE
      D=DIST(I)/AMILPDEG
      IF(D-180.)  105,115,111
105   FINLAT(I)=90.-D
      FINLON(I)=BS
      FINBER(I)=180.
      GO TO 120
111   FINLAT(I)=D-270.
      FINLON(I)=BS+180.
      IF(FINLON(I).GT.180.)  FINLON(I)=FINLON(I)-360.
      FINBER(I)=0.
      GO TO 120
115   FINLAT(I)=-90.
      FINBER(I)=BS+180.
      IF(FINBER(I).GT.180.)  FINBER(I)=FINBER(I)-360.
      FINLON(I)=FINBER(I)
120   CONTINUE
      RETURN
151   DO 170  I=1,N
```

```
C  INITIAL POINT AT SOUTH POLE
      D=DIST(I)/AMILPDEG
      IF(D-180.)  155,165,161
155   FINLAT(I)=D-90.
      FINLON(I)=BS
      FINBER(I)=0.
      GO TO 170
161   FINLAT(I)=270.-D
      FINLON(I)=BS+180.
      IF(FINLON(I).GT.180.)  FINLON(I)=FINLON(I)-360.
      FINBER(I)=180.
      GO TO 170
165   FINLAT(I)= 90.
      FINBER(I)=BS+180.
      IF(FINBER(I).GT.180.)  FINBER(I)=FINBER(I)-360.
      FINLON(I)=FINBER(I)
170   CONTINUE
      RETURN
201   DNP=90.-PAS
C  GREAT CIRCLE PASSES THROUGH NORTH POLE, AND THEN SOUTH POLE
      DSP=180.+DNP
      DO 230  I=1,N
      D=DIST(I)/AMILPDEG
      IF(D-DNP)  211,203,208
203   FINLAT(I)=90.
      FINBER(I)=POS+180.
      IF(FINBER(I).GT.180.)  FINBER(I)=FINBER(I)-360.
      FINLON(I)=FINBER(I)
      GO TO 230
205   FINLAT(I)=-90.
      FINLON(I)=FINBER(I)=POS
      GO TO 230
  208 IF(D-DSP) 221,205,209
209   D=D-360.
211   FINLAT(I)=PAS+D
      FINLON(I)=POS
      FINBER(I)=0.
```

```
      GO TO 230
221   FINLAT(I)=180.-PAS-D
      FINLON(I)=POS+180.
      IF(FINLON(I).GT.180.)  FINLON(I)=FINLON(I)-360.
      FINBER(I)=180.
230   CONTINUE
      RETURN
251   DSP=90.+PAS
C  GREAT CIRCLE PASSES THROUGH SOUTH POLE, AND THEN NORTH POLE
      DNP=180.+DSP
      DO 280  I=1,N
      D=DIST(I)/AMILPDEG
      IF(D-DSP)  261,253,258
253   FINLAT(I)=-90.
      FINBER(I)=POS+180.
      IF(FINBER(I).GT.180.)  FINBER(I)=FINBER(I)-360.
      FINLON(I)=FINBER(I)
      GO TO 280
255   FINLAT(I)= 90.
      FINLON(I)=FINBER(I)=POS
      GO TO 280
258   IF(D-DNP)  271,255,259
259   D=D-360.
261   FINLAT(I)=PAS-D
      FINLON(I)=POS
      FINBER(I)=180.
      GO TO 280
271   FINLAT(I)=D-180.-PAS
      FINLON(I)=POS+180.
      IF(FINLON(I).GT.180.)  FINLON(I)=FINLON(I)-360.
      FINBER(I)=0.
280   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GCDIST(PAS,POS,PAF,POF,BS,BF,D)
C     IDENT NUMBER - T0001000
C     TITLE - GREAT CIRCLE DISTANCE BETWEEN TWO POINTS
C     IDENT NAME - TO-NRL-GCDIST
C     LANGUAGE - FORTRAN
C     COMPUTER - CDC-3800
C     CONTRIBUTOR - DAVID CHANG, CODE 8170, PROPAGATION BRANCH,
C                             ACOUSTICS DIVISION
C     ORGANIZATION - NRL - NAVAL RESEARCH LABORATORY -
C                             WASHINGTON, D.C.   20390
C     DATE - 22 JULY 1969
C     PURPOSE - THIS SUBROUTINE FINDS THE DISTANCE IN NAUTICAL MILES
C     ALONG THE GREAT CIRCLE PATH BETWEEN TWO POINTS ON THE EARTH, AND
C     THE INITIAL AND FINAL BEARINGS OF THAT PATH.
C
C     THE EARTH IS A SPHERE WITH CIRCUMFERENCE 21,600. MILES
C     ALL ANGLES ARE IN FLOATING POINT DEGREES.
C     LATITUDES .GE. 90. ARE AT THE NORTH POLE
C     LATITUDES .LE.-90. ARE AT THE SOUTH POLE
C     ALL LONGITUDES MUST BE BETWEEN -180. (180 W) AND +180. (180 E).
C     ALL BEARINGS NOT AT POLES ARE BETWEEN 0. AND 360. DEGREES,
C     MEASURED CLOCKWISE FROM DUE NORTH.
C     ALL BEARINGS AT THE POLES ARE LONGITUDE LINES.
C     FOR TWO DIAMETRICALLY OPPOSITE POINTS, THE PATH GOES OVER THE
C     NORTH POLE.
C
C     PAS = INITIAL LATITUDE
C     POS = INITIAL LONGITUDE
C     PAF = FINAL LATITUDE
C     POF = FINAL LONGITUDE
C     BS  = INITIAL BEARING
C     BF  = FINAL BEARING
C     D   = DISTANCE IN NAUTICAL MILES.
C
      DATA(DTOR=17724357506504518B)
      DATA(RTOD=20067122734064628B)
      DATA(RADEARTH=20146555576261378B)
```

```
      DATA(AMILPDEG=60.)
      IF(PAS.GE.90.)  GO TO 51
      IF(PAS.LE.-90.)  GO TO 52
      IF(PAF.GE.90.)  GO TO 53
      IF(PAF.LE.-90.)  GO TO 54
C  NEITHER POINT IS AT A POLE
      APO=POF-POS
      LEFT=1
      IF(APO)  4,31,5
4     APO=-APO
      LEFT=-1
5     IF(APO-180.)  10,41,6
C  GREAT CIRCLE DOES NOT PASS THROUGH POLES
6     APO=360.-APO
      LEFT=-LEFT
10    APO=APO*DTOR
      LEFT=1-LEFT
      CAP=COS(APO)
      SAP=SIN(APO)
      DUMMY=PAS*DTOR
      CS=SIN(DUMMY)
      SS=COS(DUMMY)
      DUMMY=PAF*DTOR
      CF=SIN(DUMMY)
      SF=COS(DUMMY)
      CD=CS*CF+SS*SF*CAP
      D=ACOS(CD)
      SD=SIN(D)
      CBS=(CF-CS*CD)/SS/SD
      CBF=(CF*CD-CS)/SF/SD
      BS=ACOS(CBS)*RTOD
      BF=ACOS(CBF)*RTOD
      IF(.NOT. LEFT)  GO TO 30
      BS=360.-BS
      BF=360.-BF
30    D=D*RADEARTH
      RETURN
```

```
31    D=PAF-PAS
C  POINTS HAVE SAME LONGITUDE
      IF(D.LT.0.)  GO TO 55
      BS=BF=0.
      D=D*AMILPDEG
      RETURN
55    BS=BF=180.
      D=-D*AMILPDEG
      RETURN
41    D=PAS+PAF
C  POINTS ARE 180 DEGREES OF LONGITUDE APART
      IF(D.LT.0.)  GO TO 45
      BS=0.
      BF=180.
      D=(180.-D)*AMILPDEG
      RETURN
45    BS=180.
      BF=0.
      D=(180.+D)*AMILPDEG
      RETURN
51    D=(90.-PAF)*AMILPDEG
C  INITIAL POINT IS AT NORTH POLE
      BS=POF
      BF=180.
      RETURN
52    D=(90.+PAF)*AMILPDEG
C  INITIAL POINT IS AT SOUTH POLE
      BS=POF
      BF=0.
      RETURN
53    D=(90.-PAS)*AMILPDEG
C  FINAL POINT IS AT NORTH POLE
      BS=0.
      BF=POS+180.
      IF(BF.GT.180.)  BF=BF-360.
      RETURN
54    D=(90.+PAS)*AMILPDEG
C  FINAL POINT IS AT SOUTH POLE
```

```
      BS=180.
      BF=POS+180.
      IF(BF.GT.180.)  BF=BF-360.
      RETURN
      END


      PROGRAM SYNPLOT
C
C                                    A SYNBAPS PROGRAM
C                                    *******************
C
C PROGRAM LINKS, RESCALES RANGE AND PLOTS PROFILE SEGMENTS FROM TEMPORARY
C MAGNETIC TAPE INTO FINAL PROFILE FORM
C
C PROGRAM WRITTEN BY R.J. VANWYCKHOUSE,NAVOCEANO,USOP,CODE7005
C
      DIMENSION ARRAY(254),RANGE(8000),DEPTH(8000)
C ASSUMES LU-10 IS REWOUND   USE FOR TAPE ONLY
      CALL PLOTS(ARRAY,254,11)
      CALL PLOT(0.0,10.0,-3)
C READ CONTROL CARD
      READ(60,100)R,D,IUNIT,YLTH,CONVERT
  100 FORMAT(2F10.0,A7,3X,2F10.0)
  699 READ(10,200)ID,ITOTAL,MILES
  200 FORMAT(A6,2I10)
      IF(IOCHECK,10) 201,201
  201 IF(EOF,10)1000,600
  600 ITOTAL2 =0
      NTOTAL=(ITOTAL/8)*8
      IF(NTOTAL.EQ.ITOTAL) GO TO 400
      NT=NTOTAL/8
      N=ITOTAL-NTOTAL
      NK=1
      NI=8
  299 DO 11 J=1,NT
```

```
      READ(10,300)(RANGE(I),DEPTH(I),I=NK,NI)
  300 FORMAT(16(F4.0,1X))
      IF(IOCHECK,10) 202,202
  202 NK=NI+1
   11 NI=NI+8
      NI=NI-(8-N)
      GO TO(1,2,3,4,5,6,7)N
    1 READ(10,301)(RANGE(I),DEPTH(I),I=NK,NI)
  301 FORMAT(2(F4.0,1X))
      IF(IOCHECK,10) 205,205
  205 GO TO 500
    2 READ(10,302)(RANGE(I),DEPTH(I),I=NK,NI)
  302 FORMAT(4(F4.0,1X))
      IF(IOCHECK,10) 206,206
  206 GO TO 500
    3 READ(10,303)(RANGE(I),DEPTH(I),I=NK,NI)
  303 FORMAT(6(F4.0,1X))
      IF(IOCHECK,10) 207,207
  207 GO TO 500
    4 READ(10,304)(RANGE(I),DEPTH(I),I=NK,NI)
  304 FORMAT(8(F4.0,1X))
      IF(IOCHECK,10) 208,208
  208 GO TO 500
    5 READ(10,305)(RANGE(I),DEPTH(I),I=NK,NI)
  305 FORMAT(10(F4.0,1X))
      IF(IOCHECK,10) 209,209
  209 GO TO 500
    6 READ(10,306)(RANGE(I),DEPTH(I),I=NK,NI)
  306 FORMAT(12(F4.0,1X))
      IF(IOCHECK,10) 210,210
  210 GO TO 500
    7 READ(10,307)(RANGE(I),DEPTH(I),I=NK,NI)
  307 FORMAT(14(F4.0,1X))
      IF(IOCHECK,10) 211,211
  211 GO TO 500
  400 NK=1
      NI=8
```

```
      IT=ITOTAL/8
  401 DO 22 J=1,IT
      READ(10,300)(RANGE(I),DEPTH(I),I=NK,NI)
      IF(IOCHECK,10) 203,203
  203 NK=NI+1
   22 NI=NI+8
      NI=NI-8
  500 READ(10,200) ID2,ITOTAL2,MILES2
      IF(IOCHECK,10) 204,204
  204 IF(EOF,10) 650,499
  499 IF(ID2-ID) 501,502,501
  501 BACKSPACE 10
      MILES2=0
      ITOTAL2=0
      GO TO 650
  502 NTOTAL=(ITOTAL2/8)*8
      IF(NTOTAL .EQ. ITOTAL2) GO TO 503
      NT=NTOTAL/8
      N=ITOTAL2 - NTOTAL
      NK=NI+1
      NI=NI+8
      GO TO 299
  503 NK=NI+1
      NI=NI+8
      IT=ITOTAL2/8
      GO TO 401
  650 MILES=NI-1
      ITOTAL=NI
      DO 55  K=1,ITOTAL
      IF(RANGE(K+1) .GT. RANGE(K)) GO TO 55
      RANGE(K+1) = RANGE(K) + 1.0
   55 CONTINUE
      IF(CONVERT .EQ. 0.0) GO TO 651
      DO 653 K=1,ITOTAL
  653 DEPTH(K)= DEPTH(K)*CONVERT
  651 WRITE(62,200) ID   ,ITOTAL,MILES
      WRITE(62,300)(RANGE(I),DEPTH(I),I=1,ITOTAL)
```

```
652 XDIS= FLOAT(MILES)/R
    IF(XDIS.GT.156.0) GO TO 800
    DO 33 J=1,ITOTAL
    IF(DEPTH(J).GT.(D*YLTH)) GO TO 800
 33 CONTINUE
    YDIS=-YLTH
    DO 44 J=1,ITOTAL
    RANGE(J)=RANGE(J)/R
 44 DEPTH(J)= -DEPTH(J)/D
    CALL AXIS(0.0,0.0,14HNAUTICAL MILES,+14,XDIS,0.0,1.0,0.0,R,4HF4.0)
    CALL AXIS(XDIS,0.0,IUNIT,+7,YLTH,-90.,1.0,0.0,D,4HF5.0)
    CALL AXIS(0.0,0.0,IUNIT,-7,YLTH,-90.,1.0,0.0,D,4HF5.0)
    CALL AXIS(0.0,YDIS,0,-1,XDIS,0.0,1.0,0.0,R,4HF4.0)
 45 CALL LINE(RANGE(1),DEPTH(1),ITOTAL,1,-1,0.0,0)
    XSYM=XDIS+1.0
    CALL SYMBOL(XSYM,0.0,0.525,ID,270.,6)
    XINCR=XDIS+5.0
    CALL PLOT(XINCR,10.0,-3)
    GO TO 699
800 WRITE(61,801) ID
801 FORMAT(1X,33HOVERFLOW OF X OR Y AXIS FOR PLOT ,A6,2X,43HCHECK CONT
   1ROL CARD - PLOTTING WILL CONTINUE)
    GO TO 699
1000 WRITE(61,802)
802 FORMAT(1X,20HEND OF SYNBAPS PLOTS)
    CALL STOPPLOT
    STOP
    END
```

DEPARTMENT OF THE NAVY
OFFICE OF NAVAL RESEARCH
875 NORTH RANDOLPH STREET
SUITE 1425
ARLINGTON VA 22203-1995

IN REPLY REFER TO:

5510/1
Ser 321OA/011/06
31 Jan 06

MEMORANDUM FOR DISTRIBUTION LIST

Subj: DECLASSIFICATION OF LONG RANGE ACOUSTIC PROPAGATION PROJECT (LRAPP) DOCUMENTS

Ref: (a) SECNAVINST 5510.36

Encl: (1) List of DECLASSIFIED LRAPP Documents

1. In accordance with reference (a), a declassification review has been conducted on a number of classified LRAPP documents.

2. The LRAPP documents listed in enclosure (1) have been downgraded to UNCLASSIFIED and have been approved for public release. These documents should be remarked as follows:

   Classification changed to UNCLASSIFIED by authority of the Chief of Naval Operations (N772) letter N772A/6U875630, 20 January 2006.

   DISTRIBUTION STATEMENT A: Approved for Public Release; Distribution is unlimited.

3. Questions may be directed to the undersigned on (703) 696-4619, DSN 426-4619.

BRIAN LINK
By direction

Subj: DECLASSIFICATION OF LONG RANGE ACOUSTIC PROPAGATION PROJECT (LRAPP) DOCUMENTS

DISTRIBUTION LIST:

NAVOCEANO (Code N121LC – Jaime Ratliff)
NRL Washington (Code 5596.3 – Mary Templeman)
PEO LMW Det San Diego (PMS 181)
DTIC-OCQ (Larry Downing)
ARL, U of Texas
Blue Sea Corporation (Dr.Roy Gaul)
ONR 32B (CAPT Paul Stewart)
ONR 321OA (Dr. Ellen Livingston)
APL, U of Washington
APL, Johns Hopkins University
ARL, Penn State University
MPL of Scripps Institution of Oceanography
WHOI
NAVSEA
NAVAIR
NUWC
SAIC

Declassified LRAPP Documents

Report Number	Personal Author	Title	Publication Source (Originator)	Pub. Date	Current Availability	Class.
Unavailable	Brancart, C. P.	TRANSMISSION REPORT, VIBROSEIS CW ACOUSTIC SOURCE, CHURCH ANCHOR EXERCISE, AUGUST AND SEPTEMBER 1973	B-K Dynamics, Inc.	730101	AD0528904	U
Unavailable	Daubin, S. C., et al.	LONG RANGE ACOUSTIC PROPAGATION PROJECT. BLAKE TEST SYNOPSIS REPORT	University of Miami, Rosenstiel School of Marine and Atmospheric Science	730101	AD0768995	U
NUSC TR NO. 4457	King, P. C., et al.	MOORED ACOUSTIC BUOY SYSTEM (MABS): SPECIFICATIONS AND DEPLOYMENTS	Naval Underwater Systems Center	730105	AD0756181; ND	U
MC-012	Unavailable	CHURCH GABBRO SYNOPSIS REPORT (U)	Maury Center for Ocean Science	730210	ND	U
Unavailable	Hecht, R. J., et al.	STATISTICAL ANALYSIS OF OCEAN NOISE	Underwater Systems, Inc.	730220	AD0526024	U
Raff rept 73-2	Bowen, J. I., et al.	EASTLANT SHIPPING DENSITIES	Raff Associates, Inc.	730227	ND	U
Unavailable	Sander, E. L.	SHIPPING SURVEILLANCE DATA FOR CHURCH GABBRO	Raff Associates, Inc.	730315	AD0765360	U
Unavailable	Wagstaff, R. A.	RANDI: RESEARCH AMBIENT NOISE DIRECTIONALITY MODEL	Naval Undersea Center	730401	AD0760692	U
Unavailable	Van Wyckhouse, R. J.	SYNTHETIC BATHYMETRIC PROFILING SYSTEM (SYNBAPS)	Naval Oceanographic Office	730501	AD0762070	U
MCPLAN012	Unavailable	SQUARE DEAL EXERCISE PLAN (U)	Maury Center for Ocean Science	730501	NS; ND	U
Unavailable	Marshall, S. W.	AMBIENT NOISE AND SIGNAL-TO-NOISE PROFILES IN IOMEDEX	Naval Research Laboratory	730601	AD0527037	U
Unavailable	Daubin, S. C.	CHURCH GABBRO TECHNICAL NOTE: SYSTEMS DESCRIPTION AND PERFORMANCE	University of Miami, Rosenstiel School of Marine and Atmospheric Science	730601	AD0763460	U
MC-011	Unavailable	CHURCH ANCHOR EXERCISE PLAN (U)	Maury Center for Ocean Science	730601	ND	U
Unavailable	Solosko, R. B.	SEMI-AUTOMATIC SYSTEM FOR DIGITIZING BATHYMETRY CHARTS	Calspan Corp.	730613	AD0761647	U
64	Jones, C. H.	LRAPP VERTICAL ARRAY- PHASE II	Westinghouse Research Laboratories	730613	AD0786239; ND	U
Unavailable	Koenigs, P. D., et al.	ANALYSIS OF PROPAGATION LOSS AND SIGNAL-TO-NOISE RATIOS FROM IOMEDEX	Naval Underwater Systems Center	730615	AD0526552	U
NUSC TR 4417	Perrone, A. J.	INFRASONIC AND LOW-FREQUENCY AMBIENT-NOISE MEASUREMENTS OFF NEWFOUNDLAND	Naval Underwater Systems Center	730619	ND AD 913268	U
USRD Cal. Report No. 3576	Unavailable	CALIBRATION OF FLIP-CHURCH ANCHOR TRANSDUCERS SERIALS 15 AND 19	Naval Research Laboratory	730716	ND	U