ANALYSIS AT TECHNOLOGY

AU-53
Copy No. ____
Series ____

TASSRAP INPUT MODULE

Analysis & Technology, Inc.
Report No. P-339-6-77

29 July 1977

Prepared by:

S. R. Elam
R. J. Bessette
M. F. Fleck

Prepared for:

Department of the Navy
Naval Ocean Research and Development Activity
Bay St. Louis, Mississippi, 39529
(Attn: Commander J. E. Paquin)

Analysis & Technology, Inc. — Technology Park, P. O. Box 220, North Stonington, Connecticut 06359 — (203) 599-3910

ANALYSIS A&T TECHNOLOGY

AU-53
Copy No. ____
Series ____

TASSRAP INPUT MODULE

Analysis & Technology, Inc.
Report No. P-339-6-77

29 July 1977

Prepared by:

S. R. Elam
R. J. Bessette
M. F. Fleck

Prepared for:

Department of the Navy
Naval Ocean Research and Development Activity
Bay St. Louis, Mississippi, 39529
(Attn: Commander J. E. Paquin)

Table of Contents

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

List of Illustrations

List of Tables

CHAPTER 1
Input Module Performance

1.1 INTRODUCTION This module is designed to maximize the ease with which the TASSRAP program may be operated and to minimize the possibility of input errors. There are two operating modes for the module, either automatic or interactive. In the automatic mode, the necessary operational information is specified along with options available to the operator. When these options are exercised, the module is considered to be in the interactive mode.

1.1.1 Scope This document is intended to describe the input module.

1.1.1.1 Identification. The nomenclature for this module is INPUT and is divided into two major segments - INPUT and INPUT:OV. INPUT calls the following major subroutines and functions: BTGRAPH, GETTGT, XNTF, GETSONAR, SLFRQ, and TRWND. The overlay INPUT:OV, is loaded by INPUT after completing all required tasks. Subroutines associated with INPUT:OV are: GETENV, TRWND, XNTERP, MERGE, XNTF, TWDPT, PFGRAPH and function WILSON.

1.1.1.2 Functional Summary. One of the principal design features of the input module is to accept all the data needed by the entire TASSRAP II program. These data are placed in common blocks for access by other modules. Information such as the date-time-group, latitude, and so forth are entered by the operator when requested by the program. On the other hand, sonar type, target type, and data of this nature are presented in a tabular form with the appropriate selection made by the operator. Based on input information, subroutine GETENV retrieves historical environmental data consisting of bottom reflectivity, salinity, and temperature as a function of depth. If an in situ BT is entered, these data are merged with the historical data. Wilson's equation is used to convert the data to a sound velocity profile (SVP). Target information such as speed, depth, radiated noise, and so forth are retrieved from a data file by subroutine GETTGT. Sonar characteristics are obtained by subroutine GETSONAR. Using the data retrieved by these two subprograms or appropriate data inserted by the operator, the subroutine SLFRQ selects those target frequencies that tend to maximize detection ranges. The subroutine TWDPT calculates the surface layer depth and deep sound channel axis.

1.2 DIGITAL SYSTEM REQUIREMENTS

1.2.1 General This section defines and specifies all functional, operational, and performance requirements as well as the design constraints and standards necessary to ensure the proper development and maintenance of the input module.

1.2.2 Program Description The input module is designed to accept operator inputs and/or retrieve from data files the data necessary for the other modules to function. These data are passed via labeled common blocks. INPUT is the first routine called by the driver module.

1.2.2.1 Peripheral Equipment Identification. Peripheral equipment with which the input module interfaces are: keyboard, cathode ray tube (CRT) display, and the disk drive unit.

1.2.2.2 Interface Identification. With the exception of passing data through the labeled common area, the input module interfaces only with the executive module.

1.3 FUNCTIONAL DESCRIPTION

1.3.1 Interface Block Diagram See Figure 1-1.

1.3.2 Program Interfaces All informational exchanges between the input module and other modules are transmitted via the primary communication area.

1.3.3 Function Description The major function of the input module is to accept from the operator and/or retrieve data necessary for the other modules to function. Initially there are 13 inputs, with the CRT displaying the information to be entered from the keyboard by the operator. Following the data entry, the operator selects a target type from a list presented or inputs his own frequency-source level pairs. Next, a table of target operational modes is presented from which the operator makes a choice from the table or has the option to enter a target depth directly. If frequency-source level information is not an input, this information is retrieved from a data file. Next, own-ship type of mission and sonar type are chosen from a list of available options.

BT data may be entered by the operator in either metric or English units. Bottom depth may also be an input, with the units being identical to those used when entering a BT or in meters if a BT is not input. Beam noise data may now be entered. Following this, sonar data are retrieved from the sonar file. The five optimum frequencies (if more than five are available) on which to base detection are selected by an optimization routine.

Bottom loss, historical BT, and shipping density information are retrieved from data files based upon operator inputs. If a BT is an input, these data are merged with the historical data. Retrieved data, input BT, and merged BT, if applicable, are displayed in tabular form. The operator then has the option of viewing a graphical representation of the BT data and/or the SVP calculated by the program.

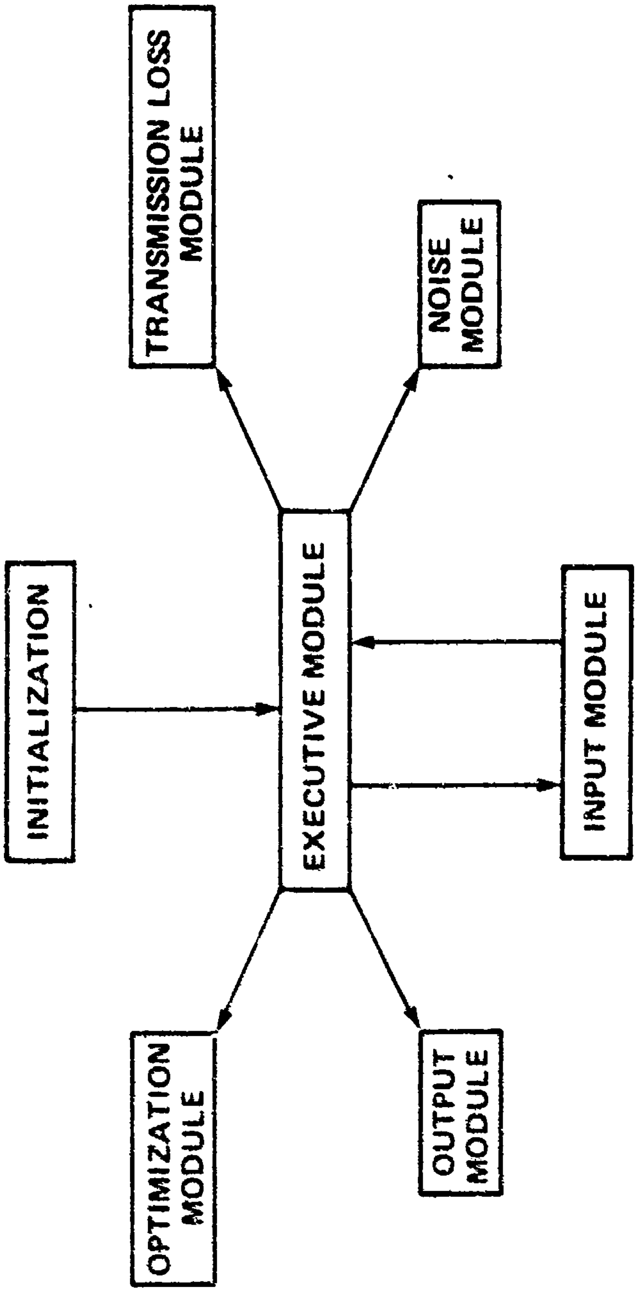


Figure 1-1. Interface Block Diagram

1.4 DETAILED FUNCTIONAL REQUIREMENTS

1.4.1 Functional Requirements Introduction

1.4.1.1 Inputs. The data input to the input module are as follows:

Data Input	Description
Identifier Label	20-character alphanumeric descriptor; entered via keyboard.
Day	1- or 2-digit number; entered via keyboard.
Month	1- or 2-digit number; entered via keyboard, checked to ascertain if input values are between 1 and 12 inclusive.
Year	1- or 2-digit number; entered via keyboard.
Time	4-digit number based on 24-hour clock; entered via keyboard.
Latitude	1- to 4-digit number with the last 2 digits representing minutes of latitude and the first 2 (if present) representing degrees of latitude; entered via keyboard.
North-South	This is a prompter. The operator responds by entering number 1 if the latitude previously entered is north latitude or 2 to designate south latitude; entered via keyboard.
Longitude	1- to 5-digit number with the last 2 digits representing minutes of longitude and the first 3 (if present) representing degrees of longitude; entered via keyboard.
East-West	This is a prompter. The operator responds by entering number 1 if the longitude previously entered

Data Input	Description
	is east longitude or 2 to designate west longitude; entered via keyboard.
Maximum Range	Number out to which propagation loss is to be calculated; entered via keyboard.
Wave Height	Number entered via keyboard.
Wind Speed	Number entered via keyboard.
Ship Speed	Number entered via keyboard.
Target Type	Selected from the following list by entering the corresponding number via the teletype: 1) Soviet nuclear submarine-Type 1 2) Soviet nuclear submarine-Type 2 3) Soviet nuclear submarine-Type 3 4) Soviet diesel Type 1 (F, R, W, Z) 5) Soviet diesel JULIETT (Type 2) 6) Soviet diesel FOXTROT (Type 3) 7) U.S. nuclear submarine-637 Class 8) Own source levels The number entered is checked to ascertain if its value is between 1 and 8, inclusive.
Own Source Levels	This optional input is entered via the keyboard in frequency-source level pairs with a maximum of five pairs acceptable.
Target Operational Mode	Selected from the following list by entering the corresponding number via the teletype: 1) Transit 2) Area search - ASW 3) Area search - surface ships 4) Barrier 5) Convoy penetration 6) Amphibious attack

Data Input	Description
	7) HVU attack 8) SSBN Operations 9) Surveillance - ASW 10) Surveillance - surface ships 11) Snorkel 12) Input own source depth The number entered is checked to ascertain that its value is between 1 and 12, inclusive.
Own Source Depth	This optional input is a number entered via the keyboard.
Own Ships Mission	Selected from the following list by inputting the corresponding number via the teletype: 1) Surveillance 2) Escort 3) Trail 4) Area Sanitization 5) Amphibious assault protection
Sonar	Selected from the following list by inputting the corresponding number via the teletype: 1) AN/SQR-15 2) AN/BQR-15 3) STASS 4) TACTASS 5) LAMBDA
BT	Optional input as depth, temperature pairs with the first depth being equal to zero and the last equal to or greater than 300 meters. The first and last depths are checked to ascertain if they comply with the above restrictions; entered via the keyboard.
Bottom Depth	Optional number entered via the keyboard. If no BT is entered, bottom depth units are meters. On

Data Input	Description
	the other hand, if a BT is entered, the same units are to be used for bottom depth.
Beam Noise	Optional data entered via the keyboard as a beam number followed by frequency-level pairs for that beam. A maximum of 24 beams and five frequency-level pairs for each beam are allowed with the appropriate checks being made.

1.4.1.2 Processing. Most of the data entered into the program are placed in the primary communication area for processing in other modules. Target type (for those cases when frequency-source level pairs are not entered) in conjunction with target operational mode are processed by subroutine GETTGT to retrieve target information. These two items determine the data block to be read. Information retrieved includes target radiated frequencies and source levels, target speed, broadband noise, operating depth (if not entered directly), reliability of radiated noise, standard deviation of noise levels for nuclear submarines, and engine RPM for diesel submarines.

Subroutine SLRFQ selects the optimum target frequencies that maximize acoustic performance. All frequencies emitted by the target are examined to determine the frequencies within the sonar frequency limits. If there are not more than five frequencies meeting this criteria, the subroutine returns to the main program. If SLFRQ has found more than five frequencies within the sonar limitations, some of those frequencies are eliminated.

In the elimination process the first step is to compare the previously selected frequencies. Should any of these frequencies be within 20 Hz of each other, one is eliminated. SLFRQ compares the SPLs and their reliabilities to decide which one to eliminate. If, at any time during this elimination process, the subroutine has reduced the number of frequencies to five, control is returned to the mainline program. Next in the elimination process (if the number of frequencies is still greater than five) is the selection of the five frequencies (from those remaining) that exhibit the highest reliabilities. SLFRQ then returns to the mainline program with this information.

Latitude, north-south, longitude, and east-west inputs are used to select the geographical area for bottom loss, environmental, and shipping density data files. The proper seasonal environmental data file is accessed on the basis of the input month.

Data retrieved includes: high and low frequency bottom loss information, historical salinity-temperature data, and shipping density. If a bottom depth is not entered, the last depth in the historical temperature profile is set equal to the bottom depth. If BT data had been entered, these data are merged with the retrieved data.

Merging techniques assume that the synoptic profile is valid from the surface to 1500 feet, and that the historical profile is valid at depths of 5000 feet and greater. Merging, therefore, occurs between the 1000- and 1500-foot depth of the synoptic BT and the 5000-foot depth of the historical data. This procedure is as follows:

1. The temperature difference (ΔT) between the synoptic BT (T_S) and the historical profile (T_H) at the bottom is determined:

$$\Delta T = T_S - T_H$$

2. Temperature at the next depth is computed by adjusting T according to a weighting factor in favor of the synoptic observation:

$$T_S = 0.70\ \Delta T$$

$$T_{S+1} = T_{H+1} + \Delta T_S\ .$$

A new temperature difference is computed by comparing T_{S+1} and T_{H+1}. This method continues until a depth of 5000 feet is reached. For example:

1500 ft $\quad T_S = 70.0 \qquad T_H = 68.0$

$T = 2.0 \qquad \Delta T_S = 1.4$

2500 ft $\quad T_{H+1} = 67.5$

$T_{S+1} = 67.5 + 1.4 = 68.9$

$\Delta T = 1.4 \qquad \Delta T_S = 1.0$

4000 ft $\quad T_{H+2} = 66.0$

$T_{S+2} = 66.0 + 10.0 = 67.0$

$\Delta T = 1.0 \qquad \Delta T_S = 0.7$

5000 ft $T_{H+3} = 60.0$

$T_{S+3} = 60.0 + 0.7 = 60.7$

If a bottom depth is entered, subroutine XNTERP is called to extrapolate values of temperature and salinity to that depth. NOPTS is the number of points in each array, and ZBOT is the bottom depth to which the values are extrapolated. It is assumed that ZBOT is deeper than the next-to-last point on the input depth array.

XPRESN is calculated as a weighting factor with

$$XPRESN = \frac{ZBOT - Z(NOPTS)}{Z(NOPTS) - Z(NOPTS-1)} .$$

Temperature and salinity at ZBOT equal:

$$T(NOPTS) + XPRESN\,[T(NOPTS) - T(NOPTS-1)]$$

$$S(NOPTS) + XPRESN\,[S(NOPTS) - S(NOPTS-1)] .$$

These extrapolated values and ZBOT are returned as the last points in their respective arrays.

Function XNTF interpolates the value of a parameter for a given depth. ZF is the depth at which the interpolated value is needed. ZA is the depth array over which the interpolation is performed, and TA is the array of values to be interpolated. NOPTS represents the number of points in the depth array.

Interpolation is accomplished by a do-loop from I = 2 to NOPTS. ZA(I) is compared with ZF until these values are equal, or until ZF is larger than ZA(I). When equal, XNTF is set equal to TA(I). For the case when ZF is larger:

$$XNTF = TA(I-1) + [TA(I) - TA(I-1)] \times \frac{ZF - ZA(I-1)}{ZA(I) - ZA(I-1)} .$$

Sound velocity profiles are calculated using function WILSON. This function is called with variables Z, T, and S representing depth, temperature, and salinity, respectively.

The value returned is:

WILSON = 1449.14 + SVP + SVT + SVS + STP where:

$$SVP = 1.60272 \times 10^{-1} P + 1.0268 \times 10^{-5} P^2 + 3.5216 \times 10^{-9} P^3 - 3.3603 \times 10^{-12} P^4$$

$$SVT = 4.5721T - 4.4532 \times 10^{-2} T^2 - 2.6045 \times 10^{-4} T^3 + 7.9851 \times 10^{-6} T^4$$

$$SVS = 1.39799(S35) + 1.69202 \times 10^{-3}(S35)^2$$

$$STP = 1.579T^{-} P(S35) + 7.7016 \times 10^{-5} P(S35) - 1.2943 \times 10^{-7} P^2 (S35) - 1.244 \times 10^{-2} T(S35) + 7.7711 \times 10^{-7} P^2 (S35) + 3.158 \times 10^{-8} TP(S35) + 4.5283 \times 10^{-8} T^3 P + 7.4812 \times 10^{-6} T^2 - 1.8607 \times 10^{-4} T P - 1.9646 \times 10^{-10} T P^3 + 1.8563 \times 10^{-9} T^2 P^2 - 2.5294 \times 10^{-7} T P^2$$

where:

$$S35 = S - 35$$

$$P = 1.03 + 0.1025Z + 2.5 \times 10^{-7} Z^2 .$$

Deep sound channel and surface layer depths are calculated by subroutine TWDPT. All the velocities are compared with each other to ascertain the one that is the minimum. The depth at which this velocity occurs is called the deep sound channel depth. Before proceeding a check is made to determine if the profile is essentially isovelocity. In this instance, the deep sound channel depth is set at the bottom. When this occurs, the layer depth is assigned to the surface. For the other cases, sound velocities from the surface to deep sound channel are compared to determine the maximum, with the surface layer depth set equal to the depth of maximum velocity.

1.4.1.3 Outputs. The data output by the input module includes a tabular presentation of the retrieved bottom loss, environmental, and shipping density, temperature graph, and sound velocity profiles. For examples, see pages 2-86 through 2-88.

1.5 PROGRAM DESIGN

1.5.1 Function Allocation The input module requests and accepts the TASSRAP II OB program from the operator. In addition there are provisions that enable the operator to enter data directly thereby countermanding retrieved data. Retrieved data such as bottom loss province, salinity, temperature versus depth, and shipping intensity, however, cannot be totally countermanded by the operator. In addition to accepting and retrieving data, the input module merges an input BT with the retrieved data and calculates sound velocity versus depth for the merged data, if applicable, or the historical data if no BT was entered.

1.5.2 Function Description Data input to the input module is listed below:

Data Input	Description
Identifier Label	20-character alphanumeric descriptor, entered via keyboard.
Day	1- or 2-digit number; entered via keyboard.
Month	1- or 2- digit number; entered via keyboard, checked to ascertain if input values are between 1 and 12, inclusive.
Year	1- or 2-digit number; entered via keyboard.
Time	4-digit number based on 24-hour clock; entered via keyboard.
Latitude	1- to 4-digit number with the last 2 digits representing minutes of latitude and the first 2 (if present) representing degrees of latitude; entered via keyboard.
North-South	This is a prompter. The operator responds by entering the number 1 if the latitude previously entered is north latitude or a 2 to designate south latitude; entered via keyboard.
Longitude	1- to 5-digit number with the last 2 digits representing minutes of

Data Input	Description
	longitude and the first 3 (if present) representing degrees of longitude; entered via keyboard.
East-West	This is a prompter. The operator responds by entering the number 1 if the longitude previously entered is east longitude or a 2 to designate west longitude, entered via keyboard.
Maximum Range	Number out to which propagation loss is to be calculated; entered via keyboard.
Wave Height	Number entered via keyboard.
Wind Speed	Number entered via keyboard.
Ship Speed	Number entered via keyboard.
Target Type	Selected from the following list by inputting the corresponding number via the teletype: 1) Soviet nuclear submarine-Type 1 2) Soviet nuclear submarine-Type 2 3) Soviet nuclear submarine-Type 3 4) Soviet diesel Type 1 (F, R, W, Z) 5) Soviet diesel JULIETT (Type 2) 6) Soviet diesel FOXTROT (Type 3) 7) U.S. nuclear submarine-637 Class 8) Own source levels The number entered is checked to ascertain if its value is between 1 and 8, inclusive.
Own Source Levels	This optional input is entered via the keyboard in frequency-source level pairs with a maximum of five pairs acceptable.
Target Operational Mode	Selected from the following list by inputting the corresponding number via the teletype:

Data Input	Description
	1) Transit 2) Area search - ASW 3) Area search - surface ships 4) Barrier 5) Convoy penetration 6) Amphibious attack 7) HVU attack 8) SSBN Operations 9) Surveillance - ASW 10) Surveillance - surface ships 11) Snorkel 12) Input own source depth The number input is checked to ascertain if its value is between 1 and 12, inclusive.
Own Source Depth	This optional input is a number entered via the keyboard.
Own Ships Mission	Selected from the following list by inputting the corresponding number via the teletype: 1) Surveillance 2) Escort 3) Trail 4) Area Sanitization 5) Amphibious assault protection
Sonar	Selected from the following list by inputting the corresponding number via the teletype: 1) AN/SQR-15 2) AN/BQR-15 3) STASS 4) TACTASS 5) LAMBDA
BT	Optional input as depth, temperature pairs with the first depth being equal to zero and the last equal to or greater than 300 meters. The first and last depths are checked to ascertain if they comply with the above restrictions; enter via keyboard.

Data Input	Description
Bottom Depth	Optional number entered via the keyboard. If no BT is entered, bottom depth units are meters. On the other hand, if a BT is entered, the same units are to be used for bottom depth.
Beam Noise	Optional data entered via the keyboard as a beam number followed by frequency-level pairs for that beam. A maximum of 24 beams and five frequency-level pairs for each beam are allowed with the appropriate checks being made.

Most of the data entered into the program are placed in the primary communication area for processing in other modules. Target type (for those cases when frequency-source level pairs are not entered) in conjunction with target operational mode are processed by subroutine GETTGT to retrieve target information. These two items determine the data block to be read. Information retrieved includes target radiated frequencies and source levels, target speed, broadband noise, operating depth (if not entered directly), reliability of radiated noise, standard deviation of noise levels for nuclear submarines, and engine RPM for diesel submarines.

Subroutine SLRFQ selects the optimum target frequencies that maximize acoustic performance. All frequencies emitted by the target are examined to determine the frequencies within the sonar frequency limits. If there are not more five frequencies meeting this criteria, the subroutine returns to the main program. If SLFRQ has found more than five frequencies within the sonar limitations, some of the frequencies are eliminated.

Data required by the other modules that are generated by the input module are stored in the primary communications area. The following describes these data as they appear in the primary communication area:

LABEL - Alphanumeric label of up to 20 characters including spaces; entered by operator.

ITIME - Time group. 24-hour clock; entered by operator.

IDATE - Date group. Day, month, and year; operator input.

LAT - Latitude. Four digits (0000-9000) with the last two being minutes; operator input.

INS	- Integer to denote north(1) or south(2) latitude; operator input.
LON	- Longitude. Up to five digits (00000-18000) with the last two being minutes; operator input.
IEW	- Integer to denote east(1) or west(2) longitude; operator input.
RANGE	- Maximum range in nautical miles; operator input.
WH	- Wave height in feet; operator input.
BOTZ	- Depth of ocean in meters; operator input.
SS	- Own-ship speed in knots; operator input.
WS	- Wind speed in knots; operator input.
IB	- Integer representation of the bottom loss class. Obtained from environmental file. Bits 8-11 of this variable contain the value of the low frequency bottom loss class; bits 12-15 contain the value of the high frequency bottom loss class.
ITGT	- Integer representation of the target type; operator input.
ITOM	- Integer representation of the target operational mode.
IST	- Integer representation of own-ship type mission.
ISONAR	- Integer representation of type sonar system; operator input.
FREQ	- Frequencies and SPLs on which to optimize; selected from target file based upon target type: maximum of five frequencies - row 1 contains frequencies; row 2 contains SPLs.
INUMFRQ	- The number of frequencies contained in the target frequency file and in the noise data file.
TGTDEP	- Target depth in feet.
TGTSPD	- Target speed in knots.
TGTBBN	- Target broadband noise.
TOWDP	- Array depths on which optimization is made; selected from sound velocity profile and limitations of array.

INUMDPS - The number of array depths contained in the tow depth file.

DSC - Depth of the sound channel axis in meters; selected from the sound velocity profile.

IPROF - Input BT or not: Yes = 1, No = 2.

SLD - Sonic layer depth in meters; selected from the sound velocity profile.

DMAX - Maximum array depth in meters.

Z - Depth of historical temperature and salinity; selected from environmental data file.

T - Historical temperatures for the various depths; selected from environmental data file.

S - Historical salinity for the various depths; selected from environmental data file.

ZO - Depths of the in situ BT in meters; obtained from the input BT depth.

TOB - Temperature versus depth in centigrade; obtained from the input BT.

ZM - Depths of merged temperature and salinity; obtained from historical data and input BT.

TM - Array of merged temperature versus depth; obtained from historical data and input BT.

SM - Array of salinity versus depth; obtained from historical data and interpolated for BT input depths.

VM - Velocity of sound versus depth; calculated by Wilson's equations.

DEP - Depths of the in situ BT; operator input in meters or feet.

TEMP - Array of input temperature versus depth; operator input in degrees centigrade or degrees Fahrenheit.

NOPTS - Number of data points in the retrieved data file; obtained from data file.

NDP - Number of points in the input BT; operator input.

NOPTM - Number of data points in the merged data file; obtained from data file and BT input.

MOE - An indicator which denotes whether the BT data was entered in metric or English units; 1 = metric, 2 = English.

SHPDEN - Shipping density for a 1-degree square retrieved from shipping density file.

NB - Number of beams for which beam noise was entered by the operator (maximum number is 24).

NF1 - Number of frequencies for which beam noise was input by the operator (maximum number is 5).

IBEAM - Beam numbers for beam noise entered by operator.

FREQN - Frequencies for beam noise data input by operator: column 1 contains the beams; columns 2 through 6 contain the frequencies.

LEVELN - Level of beam noise data entered by operator: column 1 contains the beams; columns 2 through 6 contain levels.

Processing by the input module includes selecting the optimum frequency for maximizing acoustic performance, merging the input BT with historical data, and calculating the sound velocity from merged or historical data.

1.5.3 Storage And Processing Allocation The input module when loaded into memory occupies 3468 blocks of storage.

1.5.4 Program Functional Flow Diagram This section presents the general system flow of program data and execution control in Figure 1-2.

1.6 QUALITY ASSURANCE PROVISIONS

1.6.1 General The input module and associated data files are tested, as they relate to various predictions provided by the program. It should be noted that obtaining a prediction based upon the various independent parameters is difficult. As a result, the best to be expected is that the prediction is a "good" one. Good predictions, as common sense dictates, are those that are "close" to the parameter being predicted. More precisely, the quality of the prediction is to be evaluated in terms of unbiasedness, consistency, efficiency, and sufficiency.

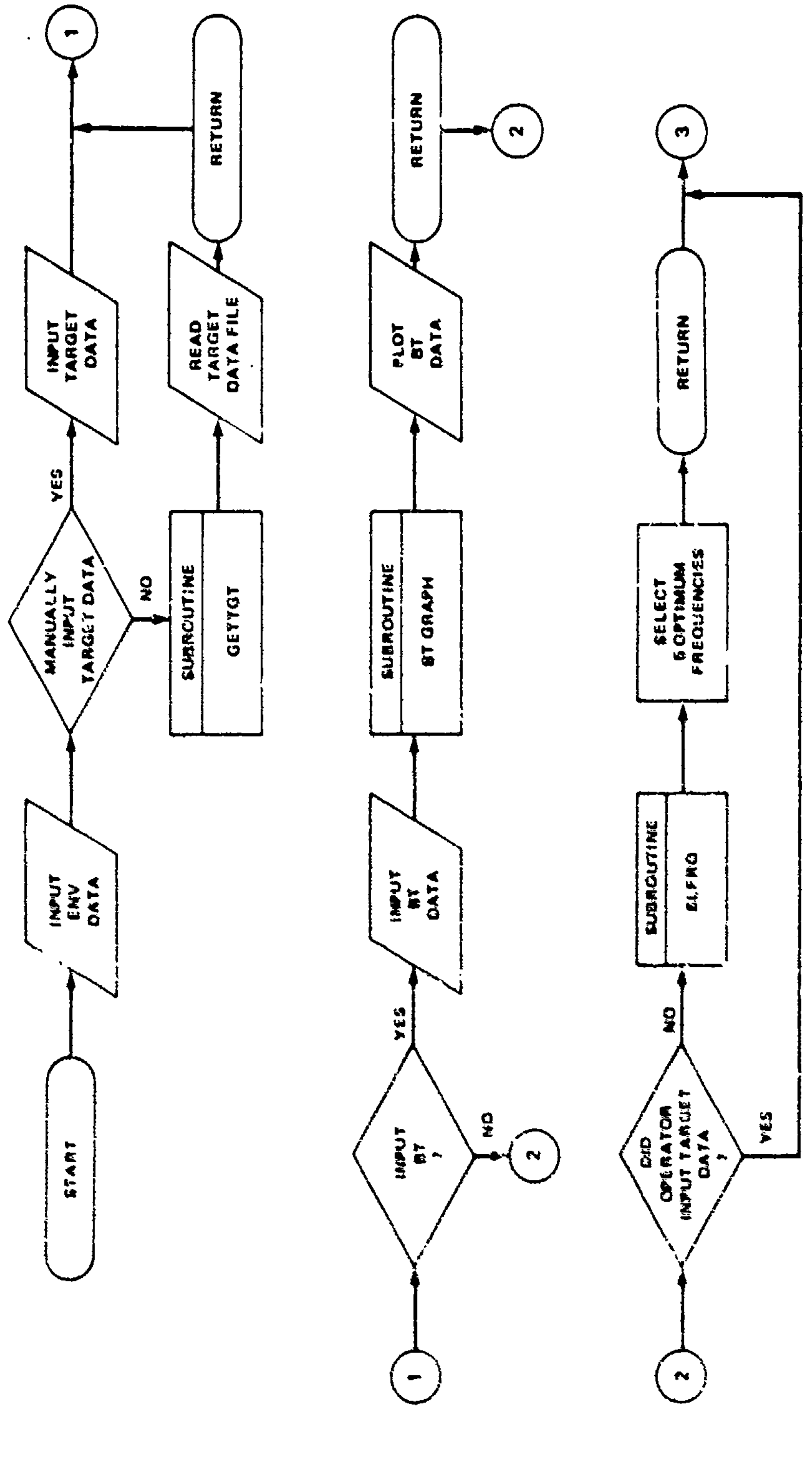


Figure 1-2. Program Data Flow and Execution Control

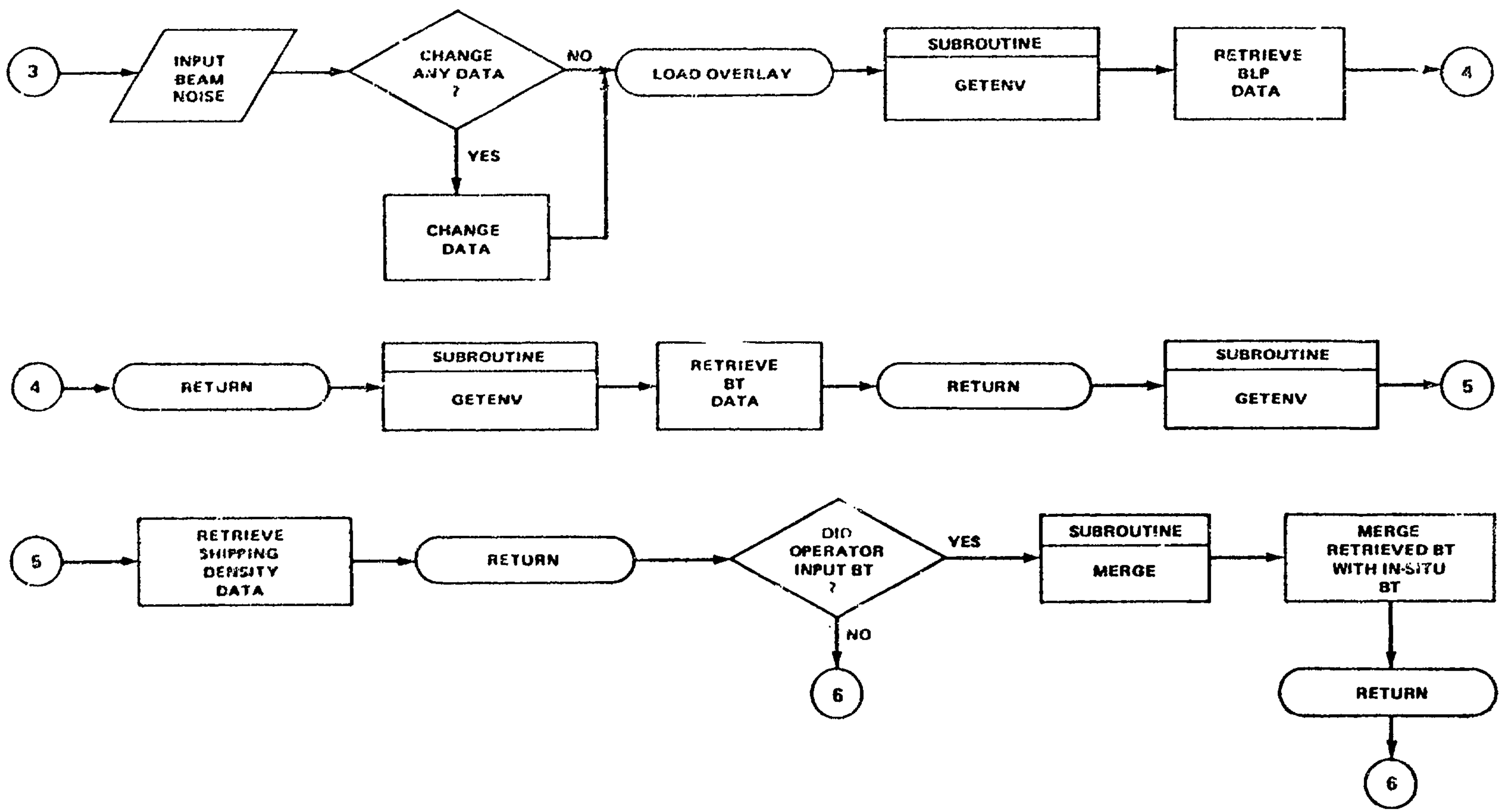


Figure 1-2. Program Data Flow and Execution Control (continued)

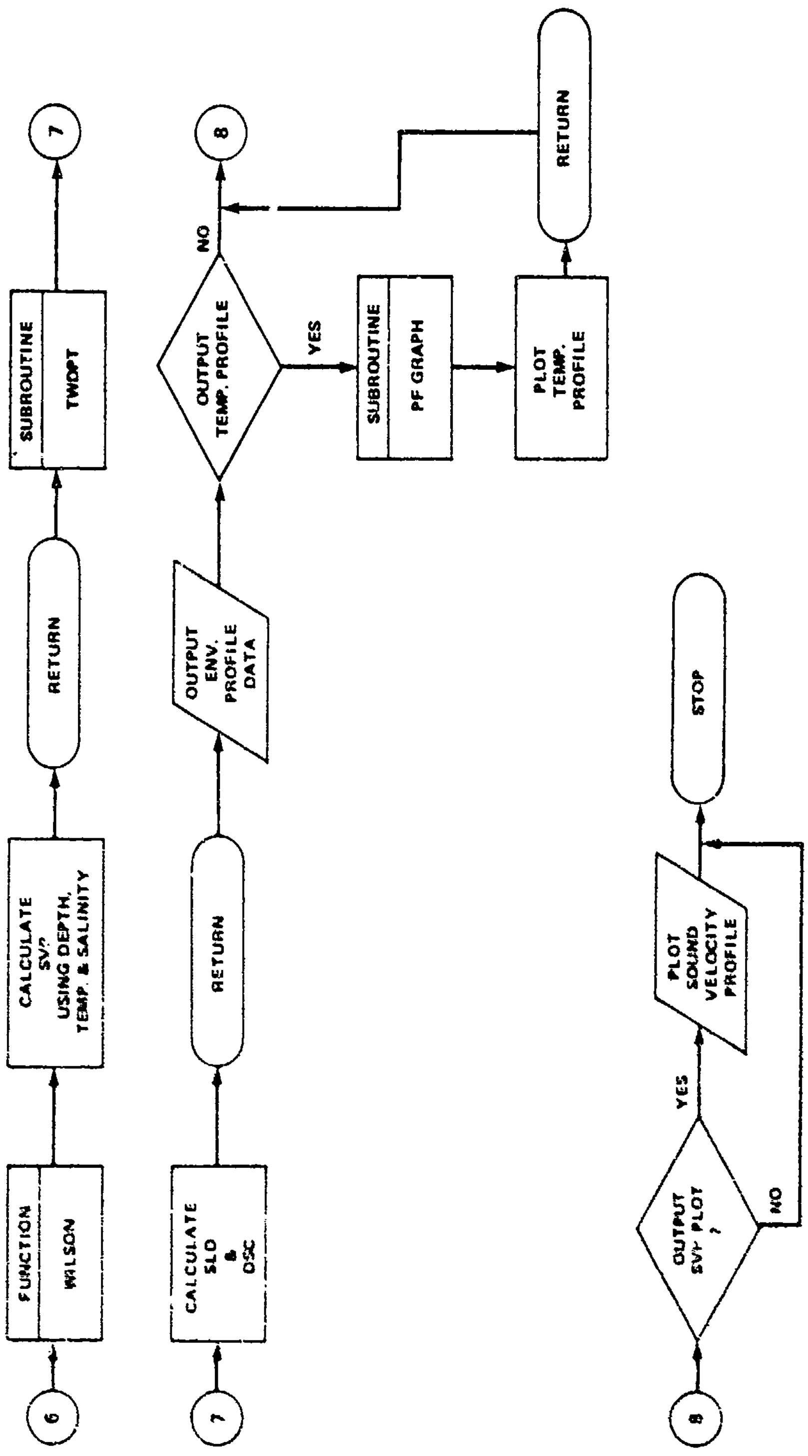


Figure 1-2. Program Data Flow and Execution Control (continued)

A prediction is unbiased if its expected value is identical with the parameter being predicted. If the probability for a prediction to approach the parameter being predicted is 1.0, as the population of the parameter approaches infinity, the prediction is consistent. One prediction is more efficient than another if the variance of the first is less than that of the second. The concept of sufficiency entails an accurate intuitive meaning. A prediction is sufficient if it conveys as much information as possible about the parameter being predicted, so that little additional information will be supplied by any other predictor.

Unbiasedness, consistency, efficiency, and sufficiency form the basic criteria for all the tests described in the succeeding pages. More quantitative criteria are applied to specific tests as necessary.

The main objective of the in-house testing is to establish whether the module will produce valid outputs for various inputs for purposes of attaining a specific objective.

1.6.2 Test Requirements While testing any module of the TASSRAP II program, the input module is tested. Testing of the target data file is to ascertain if the predicted levels are in concordance with available data. Output levels are inspected to determine whether they are within the acceptance criteria. Environmental data files are inspected to determine if any abnormalities exist in selected BT files. The TASSRAP II program is exercised for randomly selected areas of the world, and the retrieved BT and calculated SVP compared with historical FNWC data. If any abnormalities are found, NORDA should be requested to reconcile the differences.

1.6.3 Acceptance Test Requirements Acceptance of the target data file requires that all target levels be equal to the average level as reported in NWP 76-2, Submarine Acoustic Data Manual. For the BT data file, temperatures must agree within 0.5 degrees centigrade at and below the main thermocline.

CHAPTER 2
Module Description

2.1 INTRODUCTION

2.1.1 Purpose This chapter provides a detailed description of the input module that will enable a computer analyst to understand the module and easily modify it if necessary.

2.1.2 Scope This document is intended to provide a summary description of the structure and functioning of the input module.

2.1.2.1 Identification. The nomenclature for this module is INPUT and is divided into two major segments - INPUT and INPUT:OV. INPUT calls the following major subroutines and functions: BTGRAPH, GETTGT, XNTF, GETSONAR, SLFRQ, and TRWND. The overlay INPUT:OV, is loaded by INPUT after completing all required tasks. Subroutines associated with INPUT:OV are: GETENV, TRWND, XNTERP, MERGE, XNTF, TWDPT, PFGRAPH and function WILSON.

2.1.2.2 Module Tasks. One of the principal design features of the input module is to accept data needed by the entire TASSRAP II program. These data are placed in common blocks for access by other modules. Information such as the date-time-group, latitude, and so forth are entered by the operator when requested by the program. On the other hand, sonar type, target type, and data of this nature are presented in a tabular form with the appropriate selection made by the operator. Based on input information, subroutine GETENV retrieves historical environmental data consisting of bottom reflectivity, salinity, and temperature as a function of depth and shipping density. If an in situ BT is entered, these data are merged with the historical data. Wilson's equation is used to convert the data to a sound velocity profile (SVP). Target information such as speed, depth, radiated noise, and so forth are retrieved from a data file by subroutine GETTGT. Sonar characteristics are obtained by subroutine GETSONAR. Using the data retrieved by these two subprograms or appropriate data inserted by the operator, the subroutine SLFRQ selects those target frequencies that tend to maximize detection ranges. The subroutine TWDPT calculates the surface layer depth and deep sound channel axis.

2.2 REQUIREMENTS

2.2.1 Module Detailed Description

2.2.1.1 INPUT And INPUT:OV. In the initial portion of the module, data are input via accept statements. These statements are structured with a line number followed by the requested information (e.g., 2 Day =). Table 2-1 presents the input data along with line number and variable name.

Table 2-1. Initial Input Data For Input Module

Line Number	Data Requested	Variable Name
1	Label	LABEL
2	Day	IDA
3	Month	IMO
4	Year	IYR
5	Time	ITIME
6	Latitude	LAT
7	North-South	INS
8	Longitude	LON
9	East-West	IEW
10	Maximum range (nmi)	RANGE
11	Wave height (ft)	WH
12	Wind speed (kt)	WS
13	Ship speed (kt)	SS

Label, enables the operator to enter a 20-character identifier so that various outputs may be identified. Day, month, and year represent the time period for the information requested by the operator. Time is employed as another identifier. The next four inputs (latitude, north-south, longitude, and east-west) are used to determine the prediction area. Maximum range is the maximum range to which propagation loss calculations will be made. The remaining inputs are the parameter values at the time of the prediction. After entering the above, there is a provision in the routine that allows the operator to change any of the data. To alter the data, the operator types in the appropriate line number and the new value. This process continues until no more changes are desired.

Next the operator selects target type, target operational mode, own-ship type of mission, and sonar type. There are eight target types that may be selected by the operator:

1) Soviet nuclear submarine - Type 1
2) Soviet nuclear submarine - Type 2
3) Soviet nuclear submarine - Type 3
4) Soviet diesel Type 1 (F, R, W, Z)
5) Soviet diesel JULIETT (Type 2)
6) Soviet diesel FOXTROT (Type 3)
7) U.S. nuclear submarine - 637 Class
8) Own source levels

Should the operator select to enter source levels directly, a message is displayed on the CRT stating the maximum number of frequencies is five; the program then requests the number of frequencies to be entered. Frequency-source level data pairs are then entered. This data may be edited in the same manner as the initial inputs. When no further modifications to the data are desired, the program continues with the target operational mode selection.

There are 12 target operational modes selectable:

1) Transit
2) Area search - ASW
3) Area search - surface ships
4) Barrier
5) Convoy penetration
6) Amphibious attack
7) HVU attack
8) SSBN operations
9) Surveillance - ASW
10) Surveillance - surface ships
11) Snorkel
12) Input own source depth

If the operator previously entered his own source levels or elects to enter the source depth directly, the program automatically requests source depth. The subroutine GETTGT is called to retrieve target frequencies, source levels, speed, and broadband noise from the target data file if frequency-source level pairs are not entered. In addition, target depth is also retrieved if it is not a direct input. Next the operator is requested to select the type of own-ship mission. This selection provides the program with necessary information upon which to optimize array depths and search frequencies. There are five types of missions available to the operator:

1) Surveillance
2) Escort
3) Trail
4) Area sanitization
5) Amphibious assault protection

Finally, the operator inputs the particular towed array being used by own ship. At present, there are five arrays programmed into this routine with provisions to add additional arrays as they become operational. The five arrays are:

1) AN/SQR-15
2) AN/BQR-15

3) STASS
4) TACTASS
5) LAMBDA

Following these selections, there is an option which allows insertion of an in situ BT. If this option is exercised, the data may be entered in either English or metric units. There are two restrictions on the input data. First, the initial depth must be zero and second, the last depth must be greater than or equal to 300 meters. Bottom depth may be entered in the same units as the BT input.

Subroutine BTGRAPH is called, and it displays the input BT information on the CRT. In addition to the graph, the information is tabularized in a format of line number, depth, and temperature. This output is an aid to inspect the information for errors. The line numbers provide an easy method for correcting a line or lines without changing all data. This process is repeated until no changes to the BT data are necessary. After completing this, an input bottom depth is displayed, with the operator allowed to change this input.

If no BT information is entered, the above does not occur, and the bottom depth (in meters) is an optional input after the operator indicates an in situ BT input is not desired. At this point, measured beamed noise for five frequencies and 24 beams may be entered. The program accepts the number of frequencies and beams for which the data is entered. Beam noise data is inserted via the keyboard as a beam number, followed by all the frequency-level pairs for that beam.

Subroutine GETSONAR is called, following the beam noise section. This subprogram retrieves characteristics and restrictions of the sonar.

When target frequencies and source levels are not inputs, subroutine SLFRQ is accessed. This subroutine is designed to select the five target frequencies (if more than five are available) that maximize detection ranges based on frequency reliability, source level, and sonar frequency limitations.

At this point in the program structure, all the required tasks of INPUT have been performed. Most of the data input by the operator are displayed on the CRT with a provision to change any input. If a change is desired, the program goes to the appropriate section with the new data being entered. The program returns to the master display to allow further changes. This mode continues until no changes are to be made.

INPUT:OV is then overlayed on the first segment. Initially the second segment assigns values to four variables. These variables are file slots to be assigned to data files within the segment. Latitude, longitude, north-south, and east-west indicators that were entered in the first segment are used to calculate the appropriate ocean area. If there are no data files available for the input latitude and longitude, a message to that effect is displayed, and the operator is required to enter new values for latitude, north-south, longitude, and east-west After calculating the ocean area, the shipping density file is opened. The sub-area is then computed with the bottom loss file opened for the sub-area. Based upon the month input in the first segment, the appropriate seasonal data are opened for the sub-area.

Subroutine GETENV is called to retrieve bottom loss data. This subroutine addresses file slot LUNOS on which ROUGH was opened. Returned data are converted into information applicable to high and low frequencies for use in the propagation loss calculations. GETENV is called again to retrieve temperature and salinity versus depth data. For this iteration, the subroutine addresses file slot LUNAT on which the environmental file was opened. If bottom depth is not entered, the last depth in the data file is the bottom depth. Subroutine GETENV is called a third time to retrieve shipping density data.

If a BT is entered, the subroutine MERGE is called to merge the historical data with the input data. In the cases when no BT is input but a bottom depth is entered, subroutine XNTERP is called to extrapolate the historical data to the bottom depth. After establishing the STD (salinity, temperature, and depth) file, function WILSON is called to calculate the sound velocity profile. Layer depth and depth of the deep sound channel are calculated by subroutine TWDPT after the SVP has been calculated.

The output from this section displays high frequency bottom loss, low frequency bottom loss, and shipping density, with the remainder of the display being dependent on whether or not a BT is entered. Retrieved environmental data, calculated SVP, and indicators denoting surface layer and deep sound channel depths are presented in a tabularized output for the case of no input BT. On the other hand, when a BT has been entered, a tabularized output presents the input data, retrieved data, merged data, and sound velocity for the merged data. Also, the surface layer and deep sound channel depths are denoted.

Optional outputs from INPUT include a temperature profile consisting of the input BT, historical BT, merged BT, and a total temperature profile from the surface to the bottom if an in situ BT is entered. For the cases in which no BT is entered, the output is the historical temperature profile from surface to bottom. Also, a graph of the SVP can be displayed in either metric or English units from the surface to the bottom.

2.2.1.2 Subroutine BTGRAPH. Subroutine BTGRAPH is called by INPUT to display on the CRT, the BT input by the operator as an aid in editing errors. Transfer variables Z, T, NOPTS, and MOE are carried along with the call to BTGRAPH. Z is the depth array and T the temperature array input by the operator. The number of depth-temperature pairs is represented by NOPTS, and MOE is the input units - either in metric or English.

Graphical limits are set depending on the units employed. Offset, grid, and labeling routines are called, followed by labels for the appropriate units. Plotting is then performed with interpolation to 1500 feet (or 400 meters) if inputs are deeper.

2.2.1.3 Subroutine GETENV. Subroutine GETENV is used by INPUT:OV to retrieve data from bottom loss, environmental, and shipping density files. Transfer variables LUN, IBOT, and JSTAT are called with GETENV. LUN is the channel number from which the data is read. These channels will be LUNOS for bottom loss data, LUNAT for STD data, and LUNSN for shipping noise data. If IBOT equals 1, bottom loss information is retrieved; IBOT equaling 2 denotes STD data retrieval. Shipping density information is obtained if IBOT equals 3 or 4. A value of 3 denotes data for the Atlantic, Pacific, or Indian Oceans is to be calculated, while a value of 4 for IBOT denotes Mediterranean Sea shipping noise. Status of the subroutine execution is returned to INPUT:OV via JSTAT. Table 2-2 lists the values of JSTAT and the corresponding meaning.

Table 2-2. Execution Status Indicators (GETENV)

JSTAT	INTERPRETATION
1	Normal execution.
2	Data are outside file area, or wrong seasonal file is loaded.
3	Area requested is on land.
4	Read function on data block was not executed properly.
5	Information at beginning of a data block does not check.

Parameters IBUF and ILOC are initialized at the beginning of the program. Subroutine TR720 is called to read the first data block from channel LUN. The first 11 numbers are read into variables IBUF(1) - IBUF(11). IBUF(1) is checked to determine if the data file is a proper one. The next number is an ocean identifier (IHCW), and the season identifier (ISEA) is set equal to IBUG(3). Minimum and maximum latitude

(XLATMN and XLATMX, respectively) are set equal to the floating point equivalent of IBUF(4) and IBUF(5), respectively. Also, the floating point values of IBUF(6) and IBUF(7) are used for the minimum (XLONMN) and maximum (XLONMAX) longitudes, respectively. The number of data blocks in the second section of the file is IBUF(8) and is identified as NLBLK. IMAX is the degrees of longitude covered by the data file and is equal to IBUF(9). JMAX is the degrees of latitude encompassed by the file and is equal to IBUF(10). The number of data blocks in the third section (NDBLK) is equivalent to IBUF(11). For the bottom loss file and shipping density, there is no third section, and this number is always zero.

XLON, XLAT, and the month are checked to determine if the values are reasonable (i.e., the month is between 1 and 12, XLON is less than 360, and XLAT is less than 90). Also, the input latitude and longitude is compared with the data file latitude and longitude limits to verify that the proper data file has been accessed. If any data fail to pass the above checks, the appropriate error message is displayed. JSEA is compared with ISEA. If the two values are not equal, an error message is displayed. When using the subroutine to retrieve bottom loss and shipping density data, the seasonal comparison is omitted.

Following the data verification, the program proceeds to read the second section of the data file and places it in a one-dimensional array. Latitude and longitude inputs are converted to an index denoting the position of IREF in the array. If bottom loss data is being sought, IB, the bottom loss variable, is set equal to IREF and is carried through the program in the common block XDATA. Control is then returned to INPUT:OV. When seeking shipping information, IREF is divided by the area of a five-degree quadrangle to obtain shipping density for the Atlantic, Pacific, and Indian Oceans. For the Mediterranean Sea, IREF is divided by 1-degree quadrangle. SHPDEN is set equal to the resultant and carried in common block ENV. If IREF equals zero or 999, SHPDEN is assigned a default value with a provision permitting the operator to enter his own value.

For STD information, IREF designates the appropriate data block in the third section where the environmental profile data is located. The program then searches for this data block, and subroutine TR720 is called to read these data. Checks are performed on the first three elements to ensure the proper block has been accessed. The data are converted from fixed-point format to floating-point numbers and are returned in the labeled common ENV as variables Z, T, and S. In addition, the number of points on the profile (NOPTS) is also returned in this common area.

2.2.1.4 Subroutine GETTGT. The subroutine GETTGT is called by INPUT to retrieve target data. Transfer variables, LUNTG and ISTAT, are carried by the subroutine. LUNTG is the channel on which the target data file is opened, and ISTAT is the execution status of the subroutine. Table 2-3 presents the status numbers and corresponding designations.

Table 2-3. Execution Status Indicators (GETTGT)

ISTAT	INTERPRETATION
1	Normal retrieval accomplished
2	Invalid block number
3	Invalid target operating mode
4	Invalid limit information
5	Data file failure
6	End of file data not found
7	System I/O error

Information from the data file is read into a two-dimensional array named IBLOCK. Various checks are made on the data to ascertain that the correct information is being read properly. For example, target type (ITGT) and target operational mode (ITOM), input by the operator, are compared with the appropriate values in IBLOCK. Input values for ITGT and ITOM are contained in labeled common XDATA. Target depth (TGTDEP) equals the floating point conversion of IBLOCK (3, ITOM), and target speed (TGTSPD) is set equal to the floating point equivalent IBLOCK (4, ITOM). Target broadband noise level (TGTBBN), is one-tenth the value of IBLOCK (5, ITOM), with floating point conversion of IBLOCK (1, 5) being the prediction frequency (PRDFRQ) for this noise. The designation for nuclear or diesel target (IDN) is IBLOCK (1, 2), with IBLOCK (1, 3) being target type (ITYPE).

Row 6 of IBLOCK contains all the primary frequencies emitted by the target. The next row of data is comprised of SPL information which corresponds to the frequencies in the previous row. Row 8 of IBLOCK contains reliabilities corresponding to the frequencies, with the last row containing standard deviations for the SPLs. These data are placed into rows 1, 2, 3, and 4 of an array called IFRQ.

Values of IFRQ, IDN, ITYPE, and PRDFRQ leave the subroutine in a common block named TGT which is outside the primary communication area.

2.2.1.5 Subroutine HOLD. Subroutine HOLD is called from various locations throughout INPUT and INPUT:OV. HOLD acts similar to the pause command without displaying the word pause. Striking any key other than the RETURN key, erases and homes the display. If the RETURN key is depressed, a hardcopy of the CRT is made before erasing and initializing the screen.

2.2.1.6 Subroutine IOERR. The subroutine IOERR is called from numerous locations through INPUT, INPUT:OV, and associated subroutines. Variables called with IOERR are NAME, ISLOT, and IE. NAME is the file name where the error occurred, ISLOT is the slot or channel number on which the error occurred, and IE is the error number.

This subroutine writes to the CRT - "I/O error," IE; "on file," NAME; and "slot number," ISLOT.

2.2.1.7 Subroutine MERGE. The subroutine MERGE is called from INPUT:OV, as necessary, to combine an observed BT trace with the data retrieved from a historical file. Variables BOTZ and ISTAT are transferred with the subroutine. BOTZ is the bottom depth, and ISTAT is the subroutine execution status. Table 2-4 presents the values of ISTAT and their explanation which may be returned to INPUT:OV.

Table 2-4. Execution Status Indicators (MERGE)

ISTAT	EXPLANATION
1	Normal execution
2	First depth does not equal zero
3	Observed trace does not extend to 300 meters or 1000 feet
4	Observed trace is deeper depth than historical data

As shown in Table 2-4, there are several checks made in subroutine MERGE. The first point from the observed trace must equal zero, and the last point must be for a depth of 300 meters or greater. Also, the input BT cannot exceed the historical BT depth. Historical and input depth, temperatures, and salinities are contained in the labeled common ENV, along with the number of points in the historical profile (NOPTH).

Merging techniques in this subroutine assume that the synoptic profile is valid from the surface to 1500 feet. At depths of 5000 feet and greater, the historical profile is assumed to be valid. Merging, therefore, occurs between the 1000 and 1500 foot depth of the synoptic BT and the 5000 foot depth of the historical data. This procedure is as follows:

1. The temperature difference (ΔT) between the synoptic BT (T_S) and the historical profile (T_H) at the bottom is determined:

$$\Delta T = T_S - T_H$$

2. Temperature at the next depth is computed by adjusting T according to a weighting factor in favor of the synoptic observation:

$$\Delta T_S = 0.70\ \Delta T$$

$$T_{S+1} = T_{H+1} + \Delta T_S$$

A new temperature difference is computed by comparing T_{S+1} and T_{H+1}. This method continues until a depth of 5000 feet is reached. For example:

1500 ft $T_S = 70.0$ $T_H = 68.0$

$\Delta T = 2.0$ $\Delta T_S = 1.4$

2500 ft $T_{H+1} = 67.5$

$T_{S+1} = 67.5 + 1.4 = 68.9$

$\Delta T = 1.4$ $\Delta T_S = 1.0$

4000 ft $T_{H+2} = 66.0$

$T_{S+2} = 66.0 + 1.0 = 67.0$

$\Delta T = 1.0$ $\Delta T_S = 0.7$

5000 ft $T_{H+3} = 60.0$

$T_{S+3} = 60.0 + 0.7 = 60.7$

Subroutine XNTERP is called from MERGE as necessary. Salinity values for the merged data are calculated by function XNTF.

Merged values for depth, temperature, and salinity are returned as variables ZM, TM, and SM, respectively, in labeled common ENV. In addition, NOPTM, the number of points in the merged profile, is returned in the same common area.

2.2.1.8 Subroutine MOVFR. The subroutine MOVFR is used by subroutine GETENV to move the file pointer forward. Variables LUN, IMOVE, and ISTAT are transferred with this subroutine. LUN is the channel number corresponding to the file, IMOVE is the position to which the file pointer is to be moved, and ISTAT is the subroutine execution status.

Variable IRNO is brought to the subroutine via labeled common ALTIO; however, its value is set to zero by a data statement. IRNO is then set equal to IRNO + IMOV. This subroutine executes the position file command (FPSFL) on channel LUN for a record length of 288 bytes (2 x 144), with record number IRNO, zero byte count, and error code, IE.

If IE equals zero, control is returned to GETENV; otherwise, subroutine IOERR is called, and then control is returned.

2.2.1.9 Subroutine MOVBR. The subroutine MOVBR is called by GETENV to move the file pointer backwards. MOVBR is similar to MOVFR with the exceptions being that IRNO is not initialized, and its value is set to IRNO - IMOVE.

2.2.1.10 Subroutine PFGRAPH. The subroutine plots the BT data and is used by INPUT:OV. Much of this subroutine is involved with setting the graphical limits, drawing the grid, and placing the appropriate labels on the grid.

If the operator enters a BT trace, the output is three graphs simultaneously displayed on the CRT. The first graph shows the input BT and the first 400 meters of the retrieved BT. Input BTs in English units are converted to metric units. On the second graph the upper 400 meters of the merged BT are displayed, while the third graph illustrates the entire BT from surface to bottom. If a BT is not input, the retrieved data from surface to bottom is the only display.

2.2.1.11 Subroutine SLFRQ. The subroutine SLFRQ is called by INPUT to select the optimum target frequencies which maximize acoustic performance. Subroutine SLFRQ uses the transfer variables FREQ, INUMFRQ, LFRQLM, and UFRQLM. FREQ is an array containing the frequencies selected by SLFRQ. INUMFRQ is the number of frequencies in the FREQ array. LFRQLM and UFRQLM are the lower and upper frequency limits of the sonar. Another array used by this subroutine (IFRQ) is contained in the labeled common block TGT. This array contains frequency, SPL, and reliability information previously retrieved by subroutine GETTGT.

The IFRQ array (which contains all frequencies emitted by the target) is examined to determine the frequencies within the sonar frequency limits. If there are not more than five frequencies meeting this criteria, the subroutine returns to the main program. If SLFRQ has found more than five frequencies within the sonar limitations, some of those frequencies are eliminated.

In the elimination process, the first step is to compare the previously selected frequencies. Should any of these frequencies be within 20 Hz of each other, one is eliminated. SLFRQ compares the SPLs and reliabilities of the two frequencies in order to select the one to save. If, at any time during this elimination process, the subroutine reduces the number of frequencies to five, control is returned to the mainline program. Next in the elimination process (if the number of frequencies is still greater than five) is the selection of the five frequencies (from those remaining) exhibiting the highest reliabilities. SLFRQ then returns to the mainline program with this information.

INUMFRQ is set to zero if none of the available frequencies meet the limitations of the sonar.

2.2.1.12 Subroutine TRWND. Subroutine TRWND executes the rewind command on the transfer variable LUN, which is a channel number (file slot).

2.2.1.13 Subroutine TR720. Subroutine TR720 is called by GETENV to read data. Transfer variables with this subroutine are LUN, IBUF, and ISTAT. LUN is the channel number from which data are read, with IBUF being a buffer area into which the data are read. ISTAT indicates status of the subroutine execution.

IBUF is dimensionalized to 144, and the value of IRNO read from common area ALTIO is incremented by one and returned to ALTIO. The read file command (FRDFL) is executed on channel LUN, buffer area IBUF, with a maximum byte count of 2 x 144, actual byte count, dummy IABC, and error code, IE.

After performing the read operation, if IE equals zero, a normal return is accomplished, otherwise subroutine IOERR is called and then a return executed.

2.2.1.14 Subroutine TWDPT. Deep sound channel depth and surface layer depth are calculated by subroutine TWDPT. There is also a provision to calculate four tow depths which has been deactivated. TWDPT is called by INPUT:OV with one transfer variable (IND). There are only two values for IND, 0 or 1, with the former indicating tow depths are not to be calculated, whereas the latter indicates the opposite. Presently, TWDPT is called with IND always equal to zero.

Deep sound channel and layer depths are computed from the sound velocity profile carried into the subroutine in common block ENV as ZM(I) and VM(I). All the velocities are compared with each other to ascertain the minimum one. The depth at which this velocity occurs is called the deep sound channel depth. Before proceeding, a check is made to determine if the profile is essentially isovelocity. In this instance, the deep sound channel depth is set at the bottom. When this occurs, the layer depth is assigned to the surface. For the other cases, sound velocities from the surface to the deep sound channel are compared to determine the maximum value, with the surface layer depth set equal to the depth of maximum velocity.

Deep sound channel and surface layer depths are returned to INPUT as DSC and SLD, respectively, in common area XDATA.

Tow depths are a function of the surface layer depth, deep sound channel depth, and maximum depth attainable by the array, DMAX, which is contained in the common block ENV. The first computed tow depth is directly related to the surface layer depth, and the last three are based on the deep sound channel and DMAX. These depths are placed in labeled common XDATA as TOWDP for use in other portions of INPUT:OV.

2.2.1.15 Function WILSON. Function WILSON is used to compute the speed of sound in water according to Wilson's equations. This function is called with variables Z, T, and S representing depth, temperature, and salinity, respectively.

The value returned is:

$$WILSON = 1449.14 + SVP + SVT + SVS + STP$$

where:

$$SVP = 1.60272 \times 10^{-1} P + 1.0268 \times 10^{-5} P^2 + 3.5216 \times 10^{-9} P^3 - 3.3603 \times 10^{-12} P^4$$

$$SVT = 4.5721T - 4.4532 \times 10^{-2} T^2 - 2.6045 \times 10^{-4} T^3 + 7.9851 \times 10^{-6} T^4$$

$$SVS = 1.39799(S35) + 1.69202 \times 10^{-3} (S35)^2$$

$$STP = 1.579T^2 P(S35) + 7.7016 \times 10^{-5} p(S35) - 1.2943 \times 10^{-7} P^2 (S35) - 1.244 \times 10^{-2} T(S35) + 7.7711 \times 10^{-7} T^2 (S35) + 3.158 \times 10^{-8} TP(S35) + 4.5283 \times 10^{-8} T^3 P + 7.4812 \times 10^{-6} T^2 P - 1.8607 \times 10^{-4} T P - 1.9646 \times 10^{-10} T P^3 + 1.8563 \times 10^{-9} T^2 P^2 - 2.5294 \times 10^{-7} T P^2$$

where:

$$S35 = S - 35$$

$$P = 1.03 + 0.1025Z \times 10^{-7} Z^2$$

2.2.1.16 Subroutine XNTERP. Subroutine XNTERP is used by INPUT:OV and subroutine MERGE to extrapolate values of temperature and salinity to a bottom depth. Called along with XNTERP are transfer variables Z, T, NOPTS, and ZBOT. Depth, temperature, and salinity arrays are Z, T, and S, respectively; all are floating point variables in and out. NOPTS is the number of points in each array, and ZBOT is the bottom depth to which the values are extrapolated. It is assumed that ZBOT is deeper than the next-to-last point on the input depth array.

XPRESN is calculated as a weighting factor with

$$XPRESN = \frac{ZBOT - Z(NOPTS)}{Z(NOPTS) - Z(NOPTS - 1)}$$

Temperature and salinity at ZBOT equal:

$$T(NOPTS) + XPRESN\,[T(NOPTS) - T(NOPTS - 1)]$$

$$S(NOPTS) + XPRESN\,[S(NOPTS) - S(NOPTS - 1)]$$

These extrapolated values and ZBOT are returned as the last points in their respective arrays.

2.2.1.17 Function XNTF. Function XNTF interpolates the value of a parameter for a given depth and is used primarily by subroutine MERGE to calculate salinity. Transfer variables ZF, ZA, TA, NOPTS are carried along with the function. ZF is the depth at which the interpolated value is performed, and TA is the depth array over which the interpolated value is needed. ZA is the depth array of values to be interpolated. NOPTS represents the number of points in the depth array.

Interpolation is accomplished by a do-loop from I = 2 to NOPTS. ZA(I) is compared with ZF until these values are equal or ZF is larger than ZA(I). When equal, XNFT is set equal to TA(I). For the case when ZF is larger:

$$XNTF = TA(I - 1) + [TA(I) - TA(I - 1)] \times \left[\frac{ZF - ZA(I - 1)}{ZA(I) - ZA(I - 1)}\right]$$

2.2.2 Module Flow Diagrams This section contains flow diagrams for INPUT (Figure 2-1) and INPUT:OV (Figure 2-2), along with the major functions and subroutines in the input module.

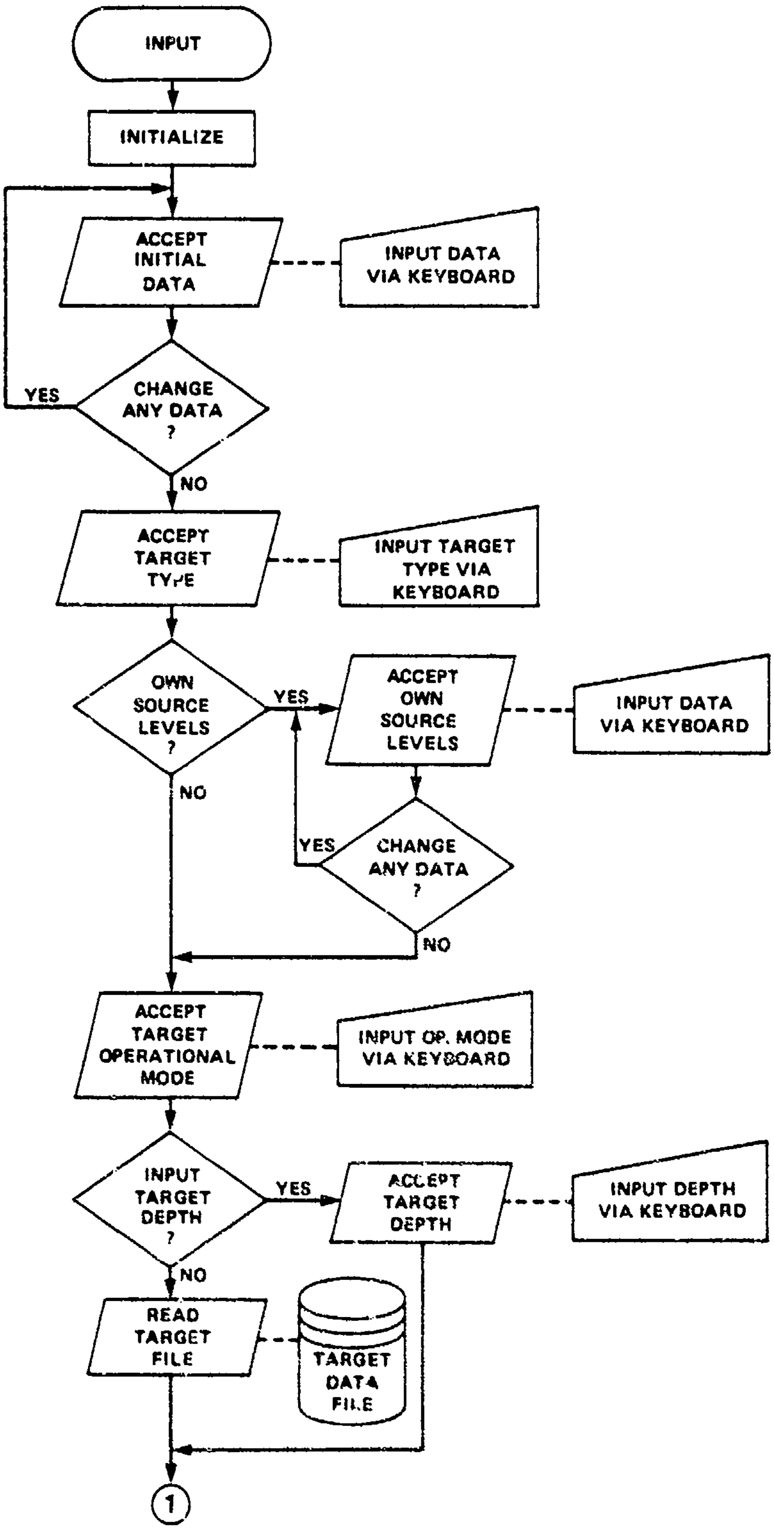


Figure 2-1. Flow Diagram of INPUT Routine

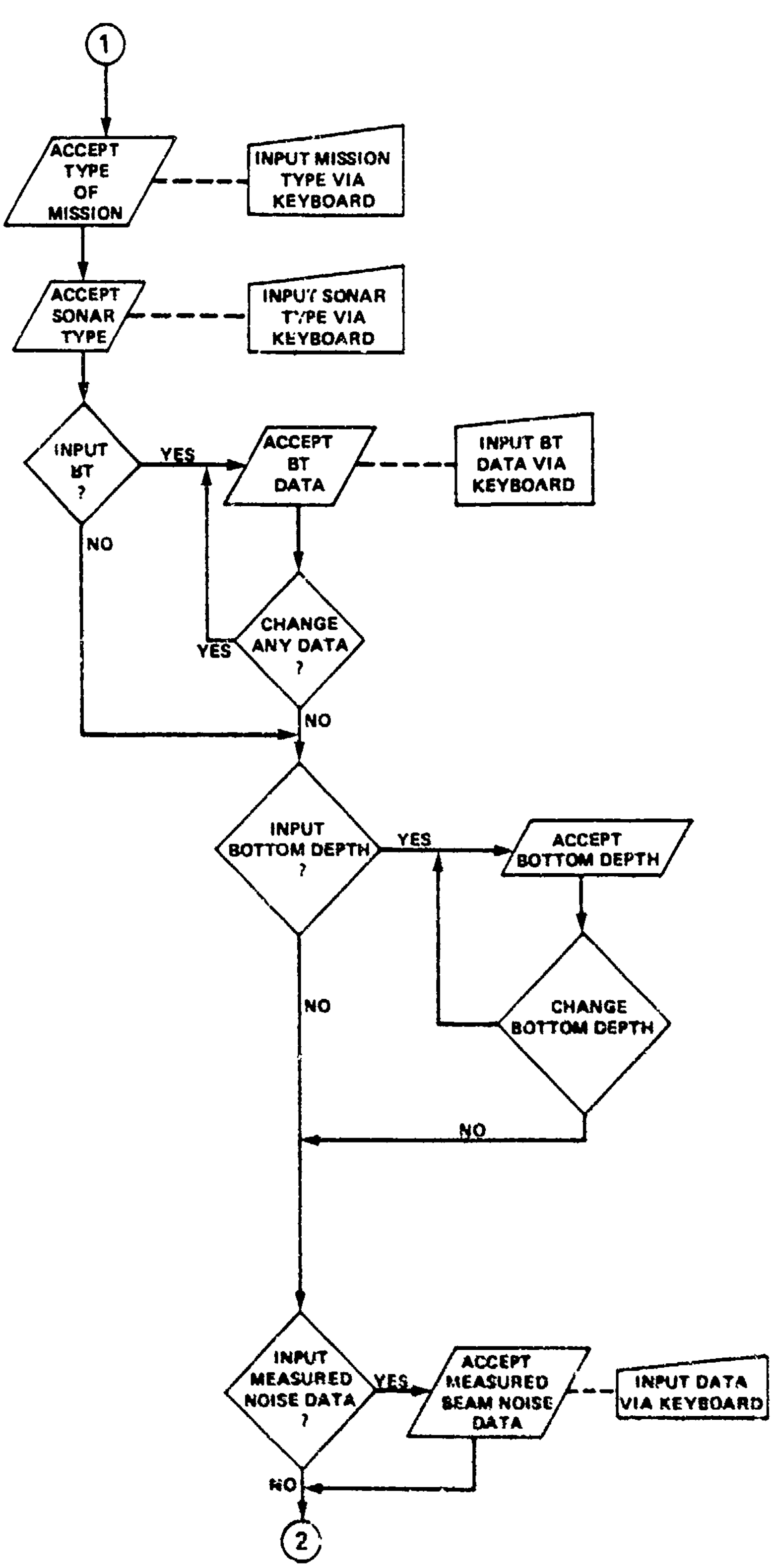


Figure 2-1. Flow Diagram of INPUT Routine (continued)

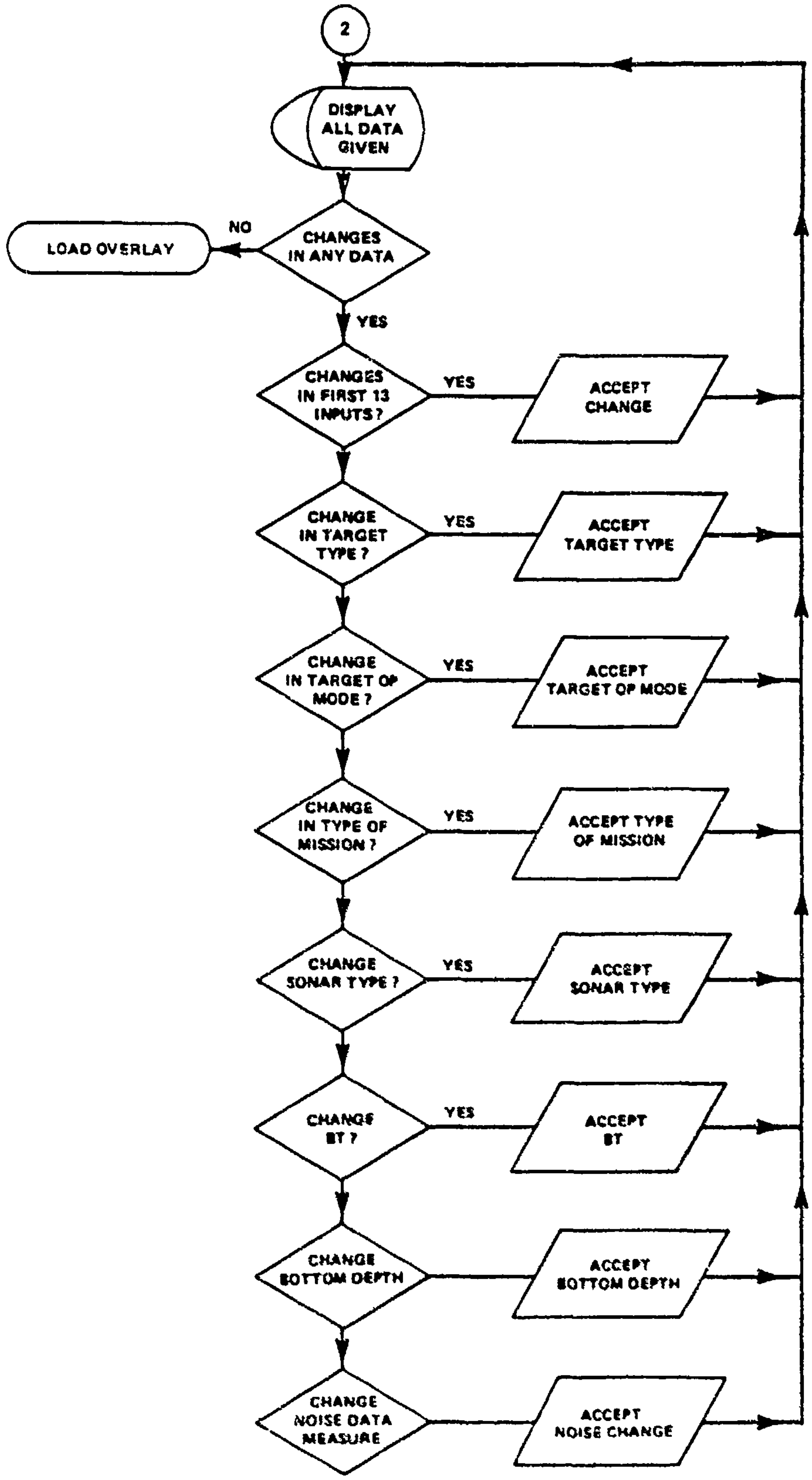


Figure 2-1. Flow Diagram of INPUT Routine (continued)

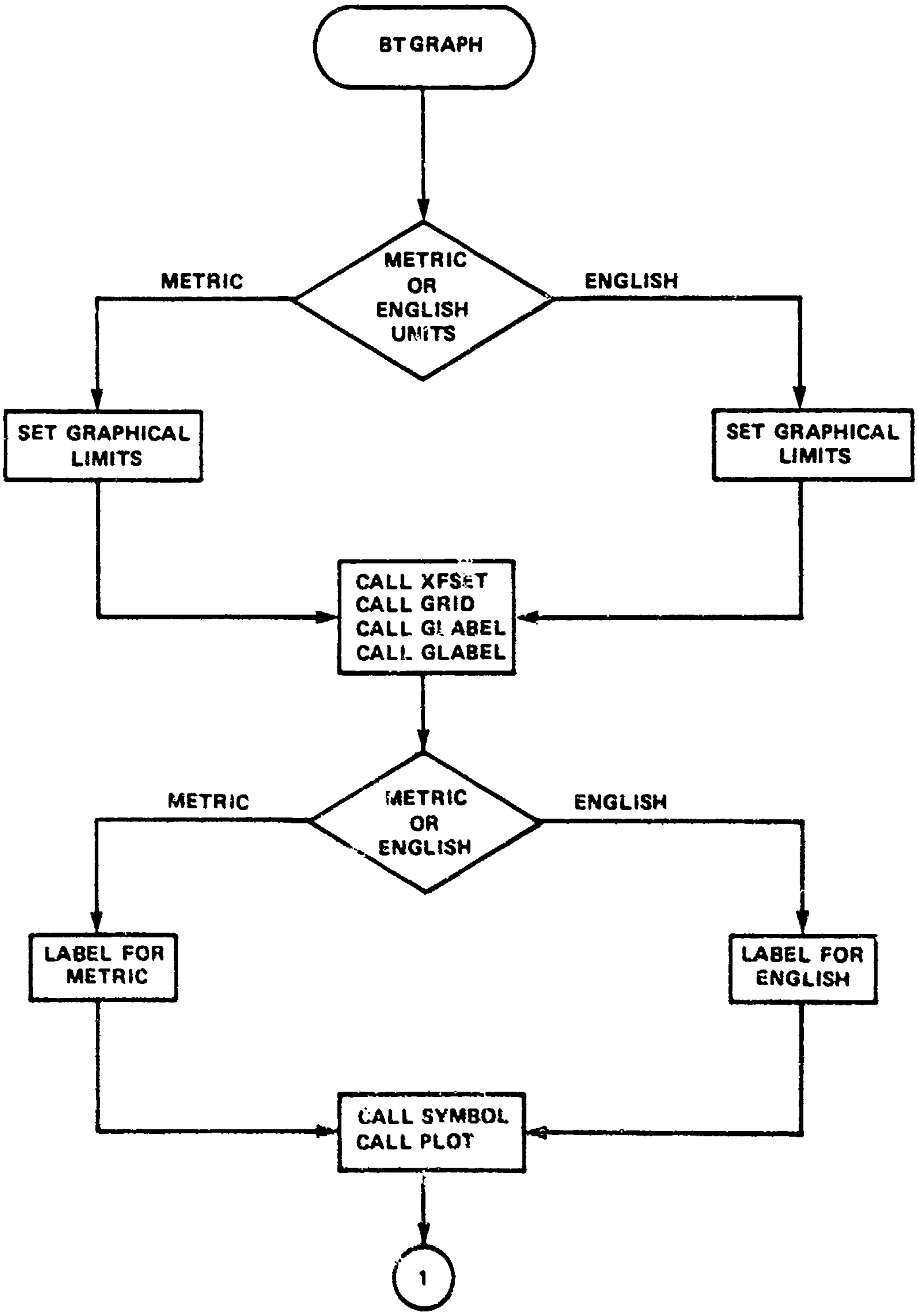


Figure 2-1. Flow Diagram of INPUT Routine (continued)

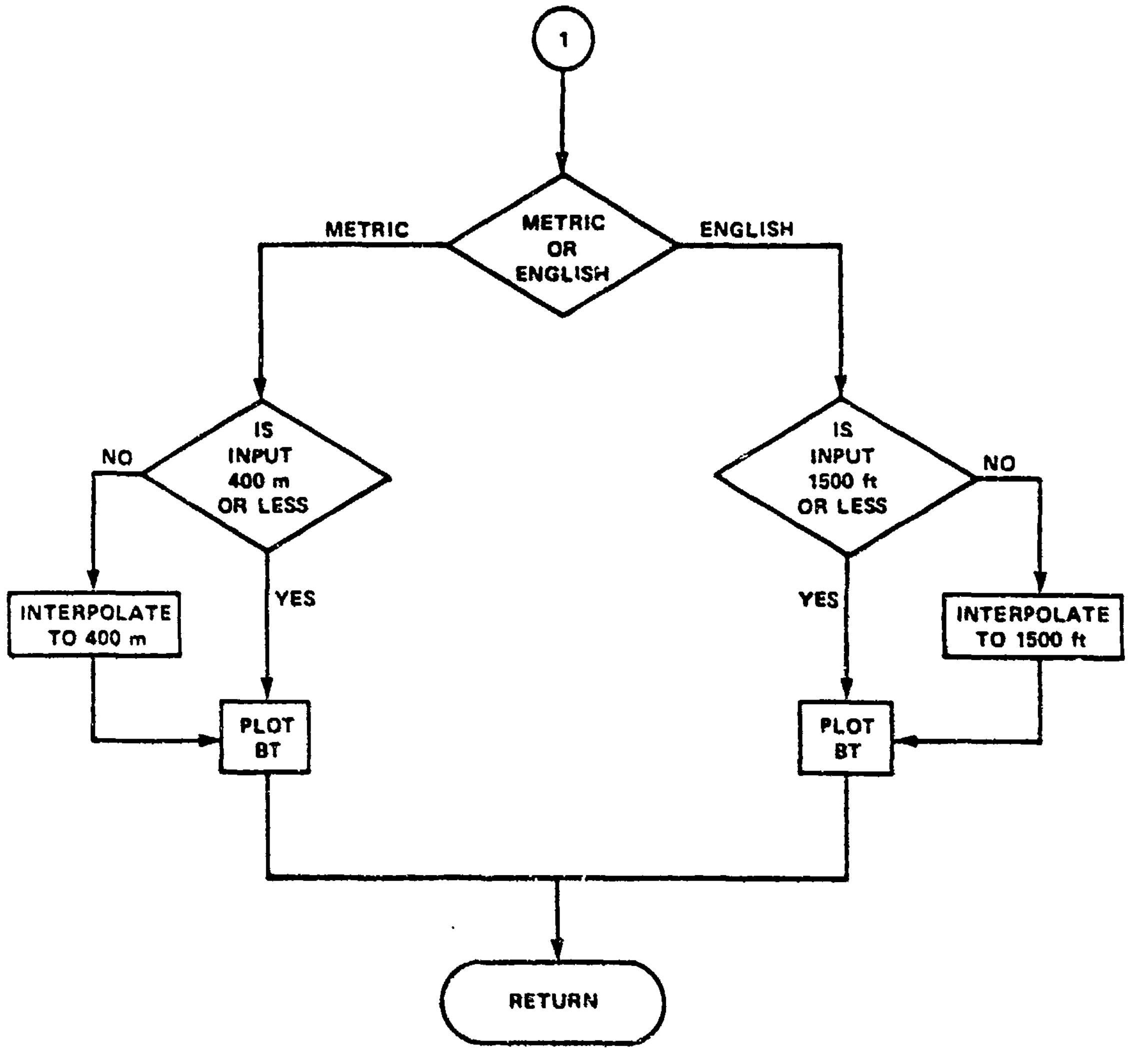


Figure 2-1. Flow Diagram of INPUT Routine (continued)

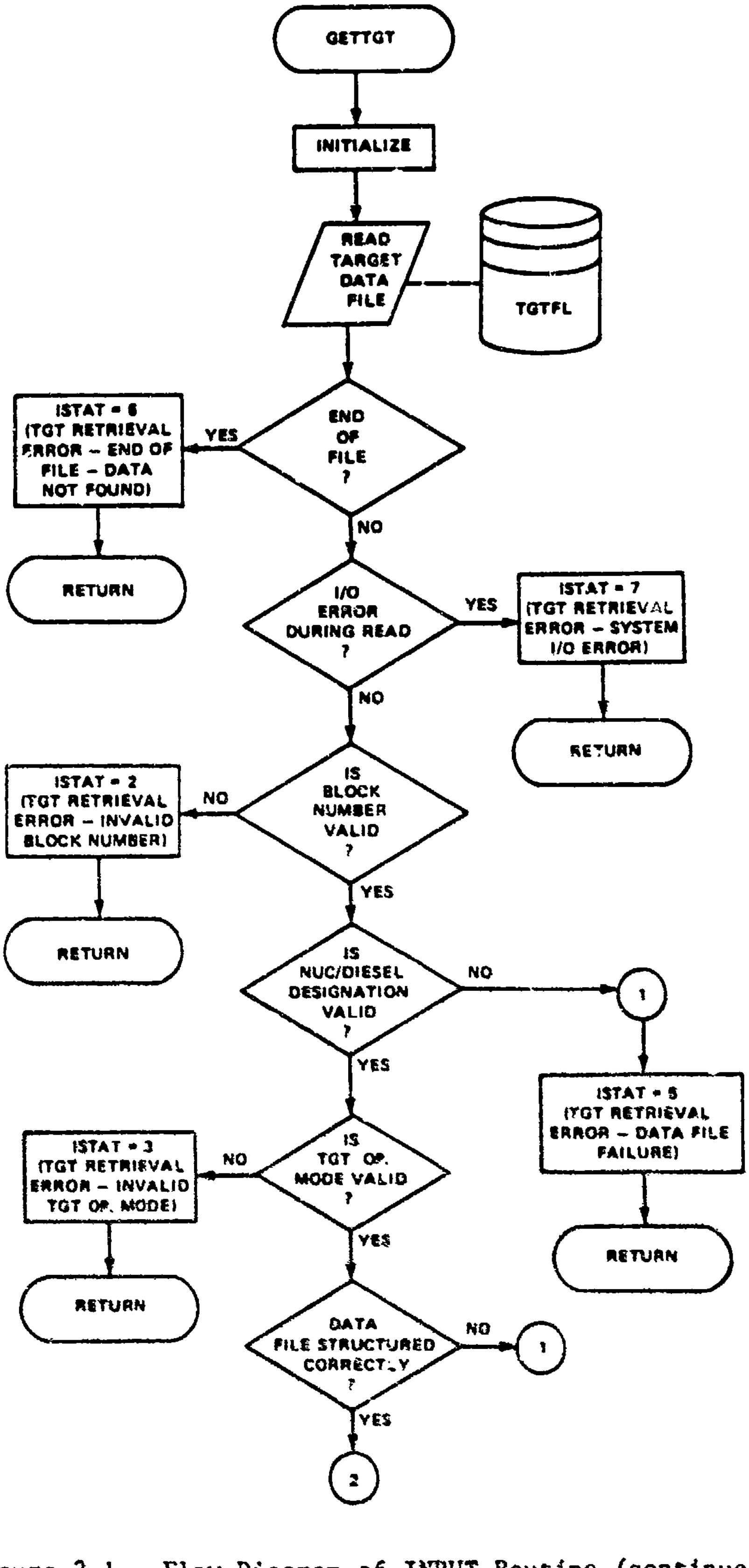


Figure 2-1. Flow Diagram of INPUT Routine (continued)

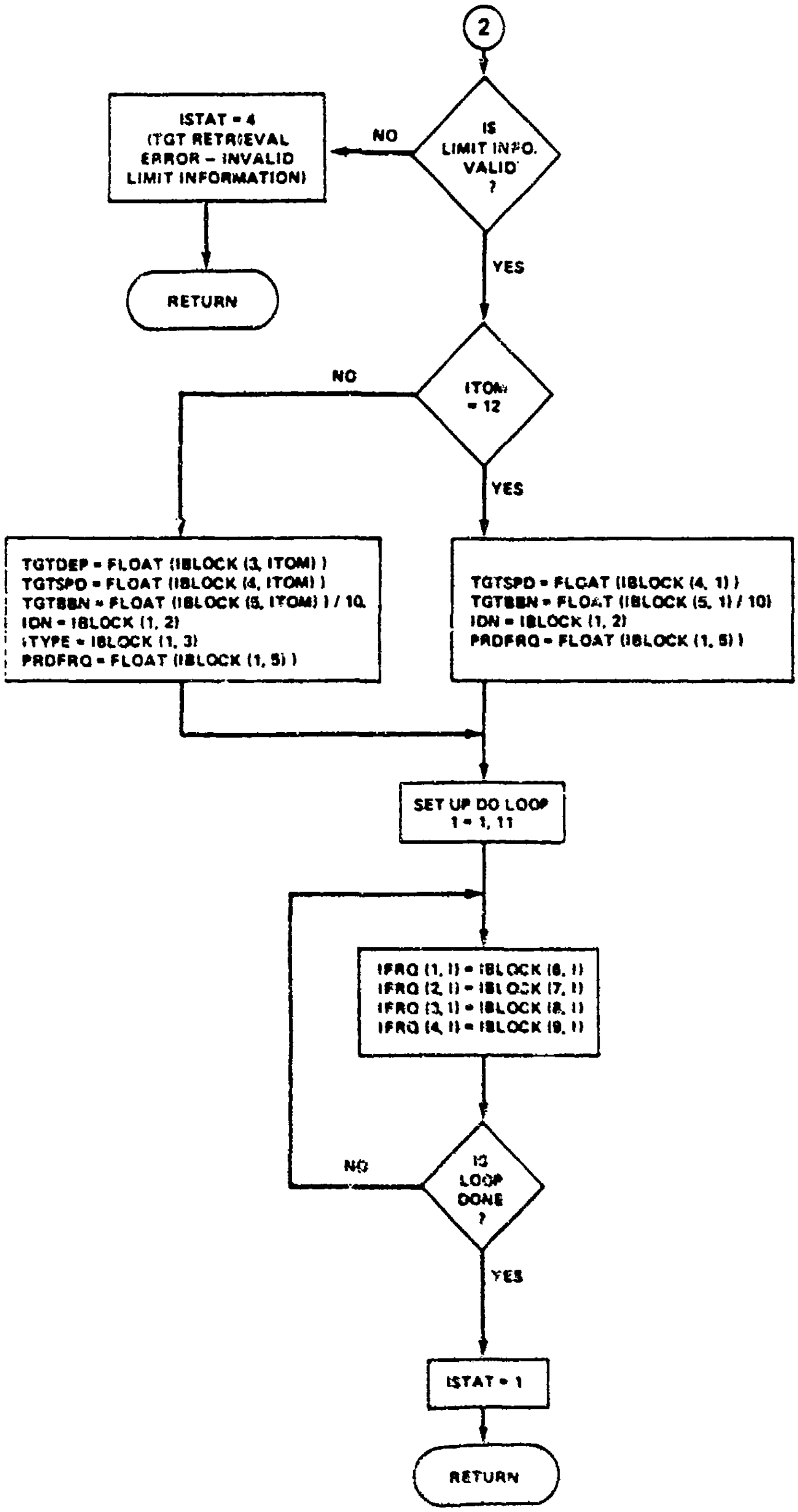


Figure 2-1. Flow Diagram of INPUT Routine (continued)

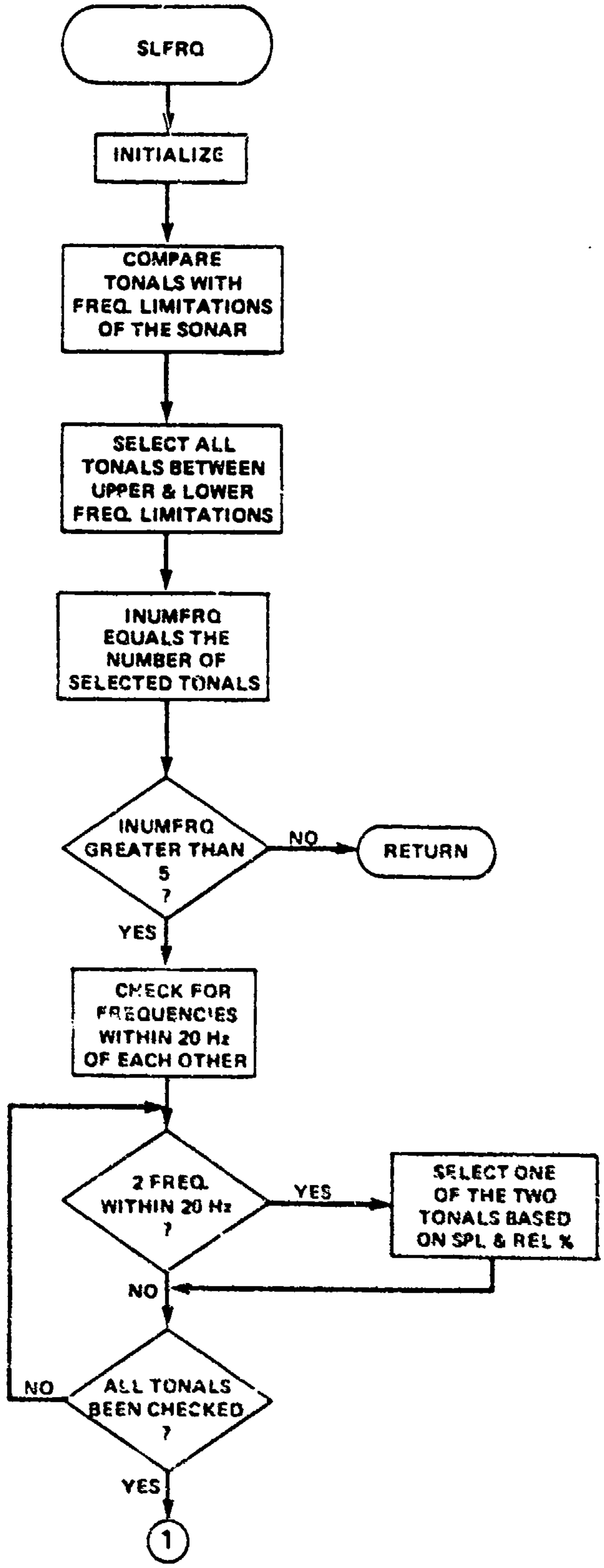


Figure 2-1. Flow Diagram of INPUT Routine (continued)

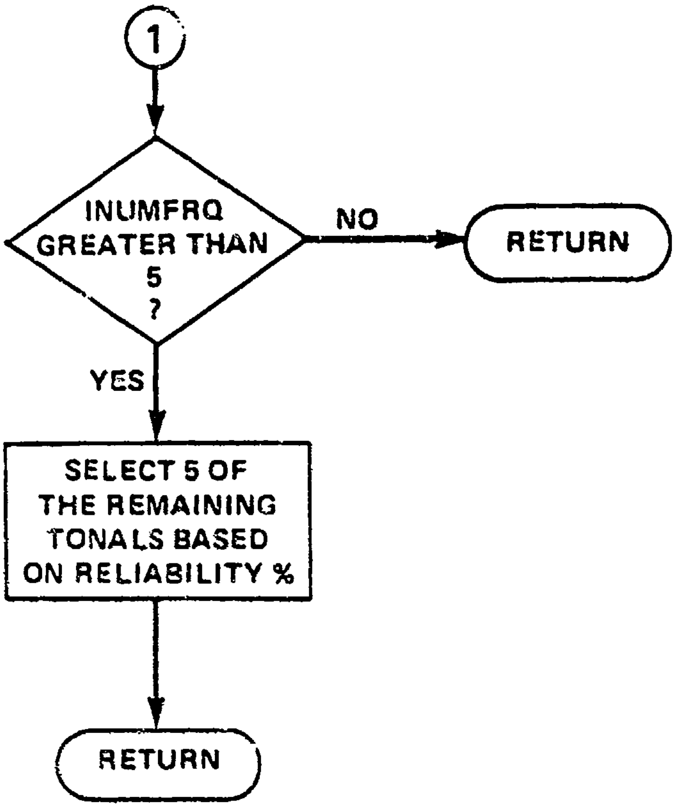


Figure 2-1. Flow Diagram of INPUT Routine (continued)

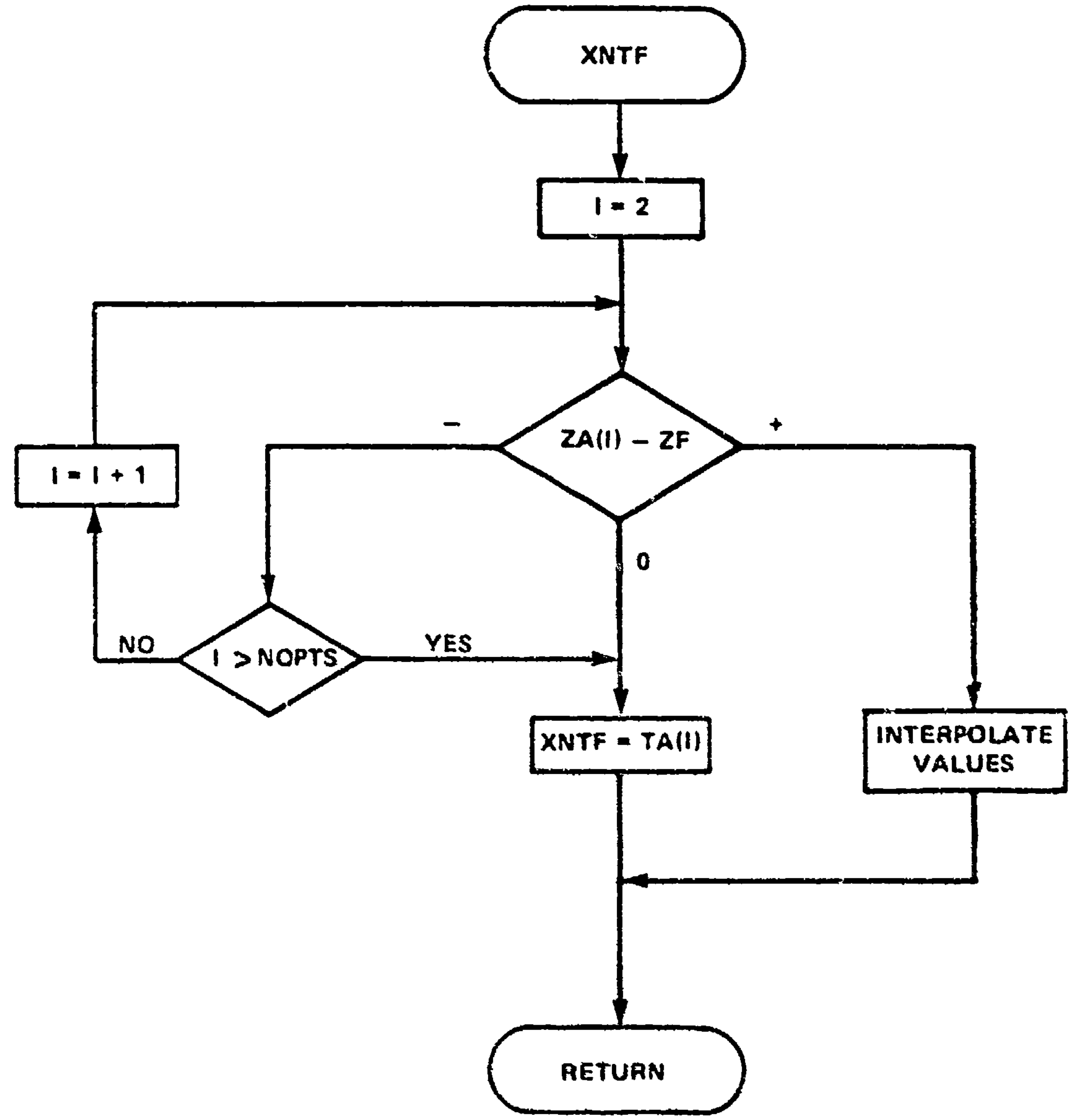


Figure 2-1. Flow Diagram of INPUT Routine (continued)

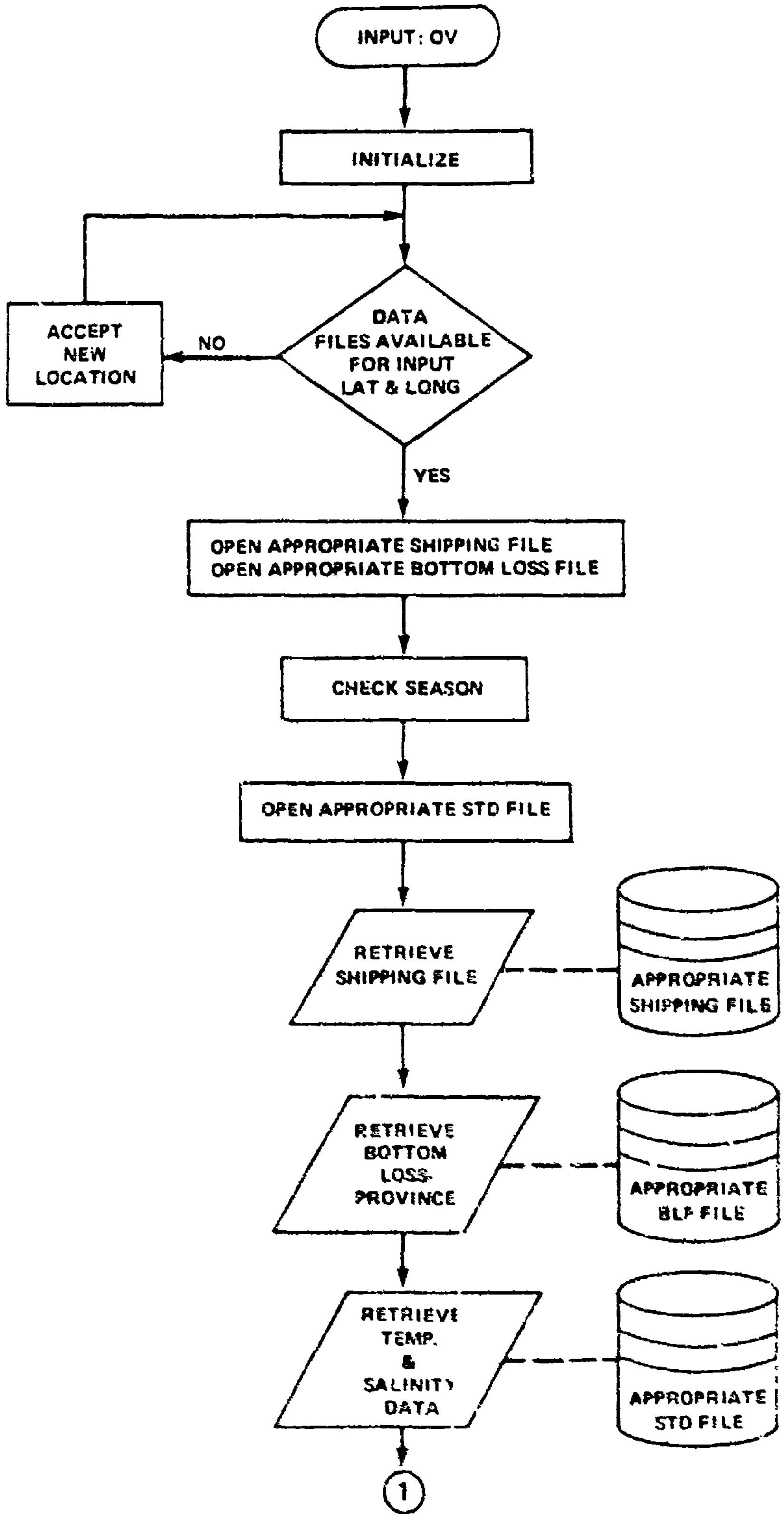


Figure 2-2. Flow Diagram of INPUT:OV Routine

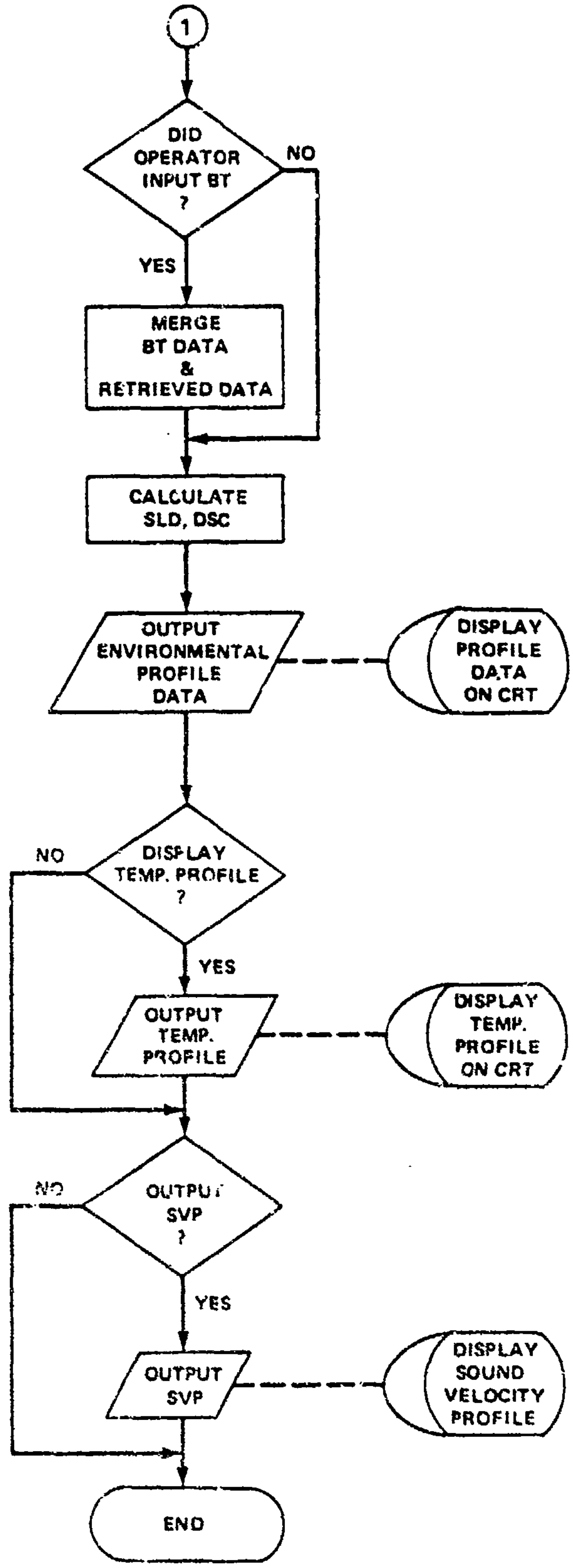


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

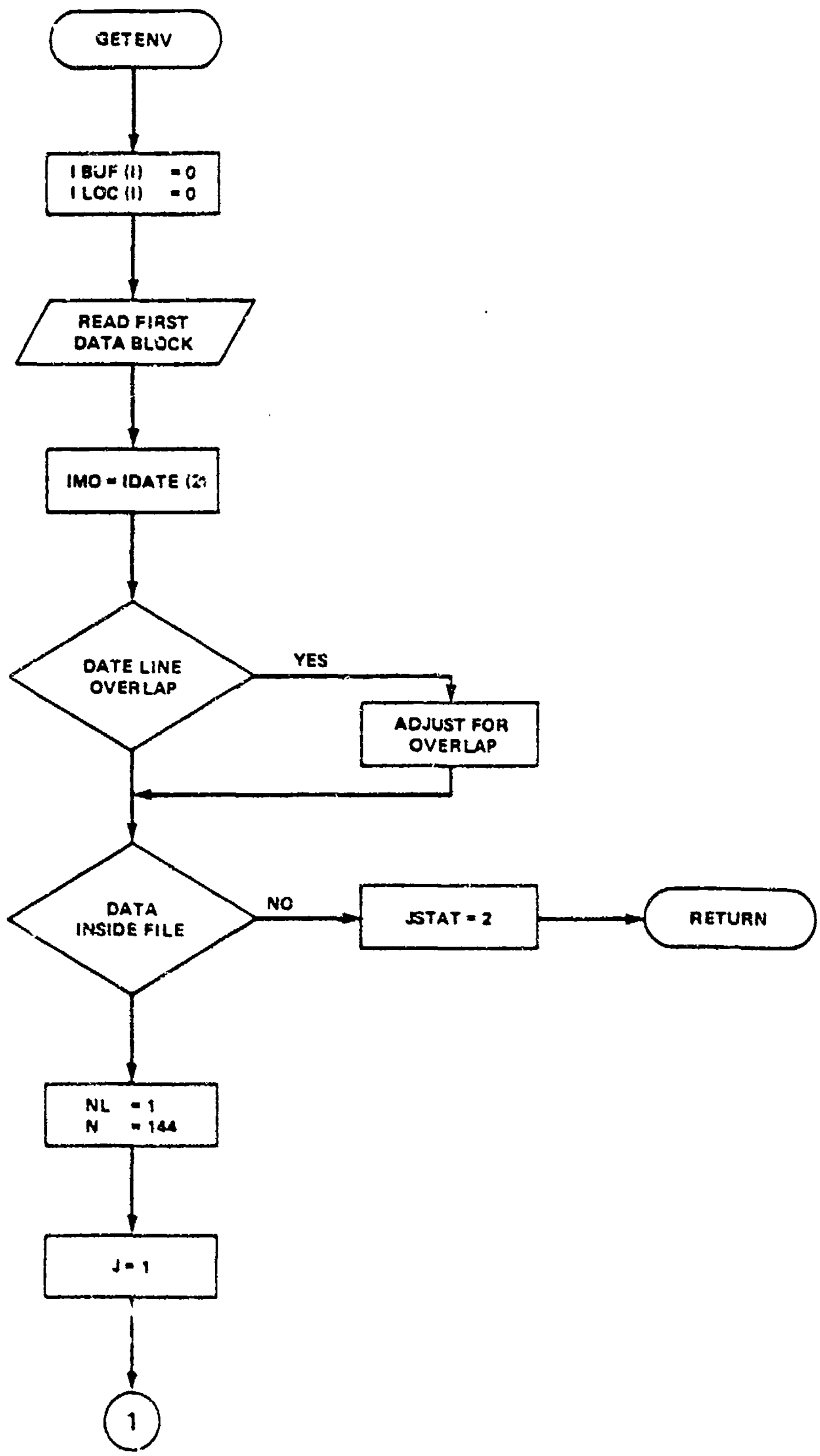


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

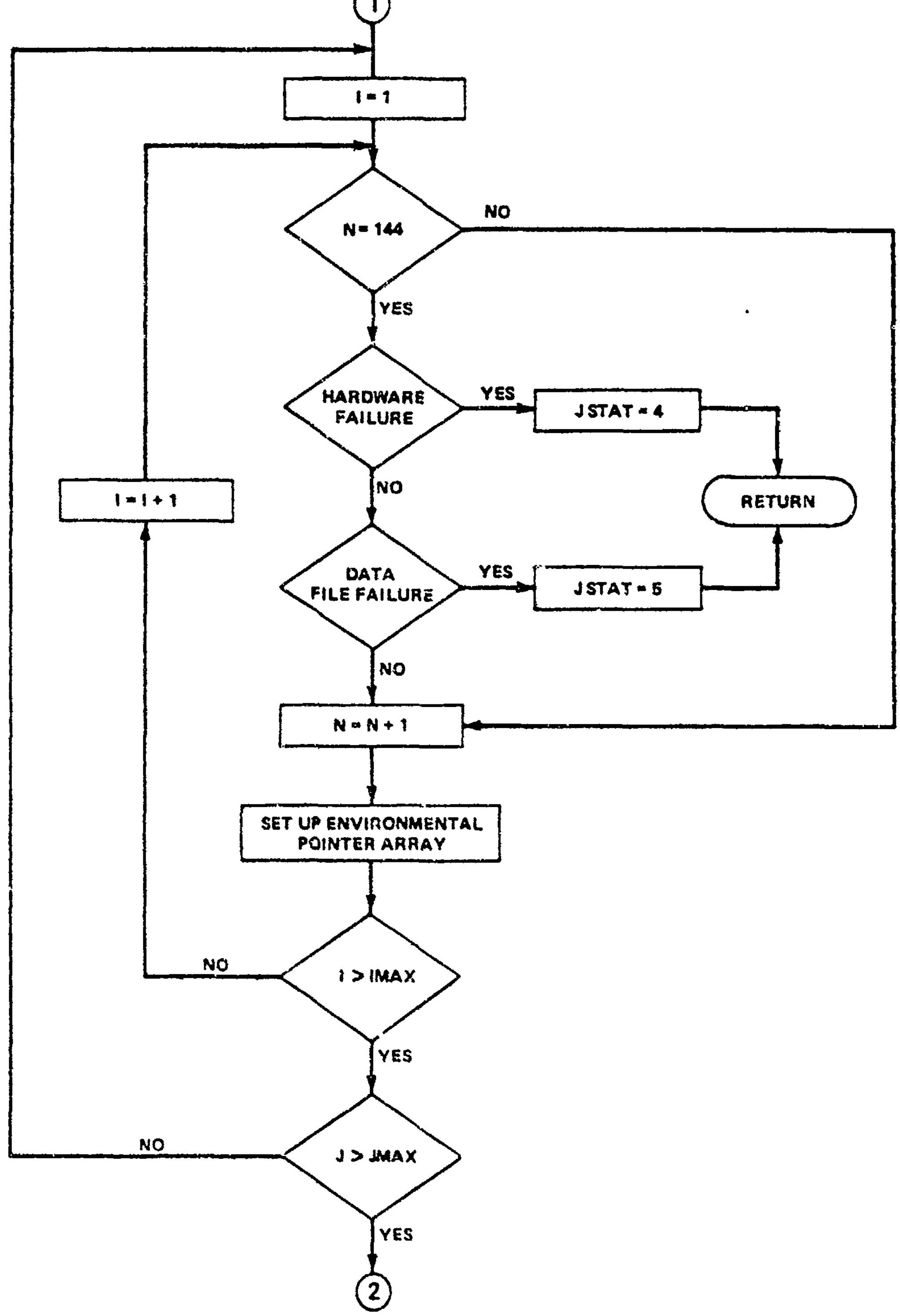


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

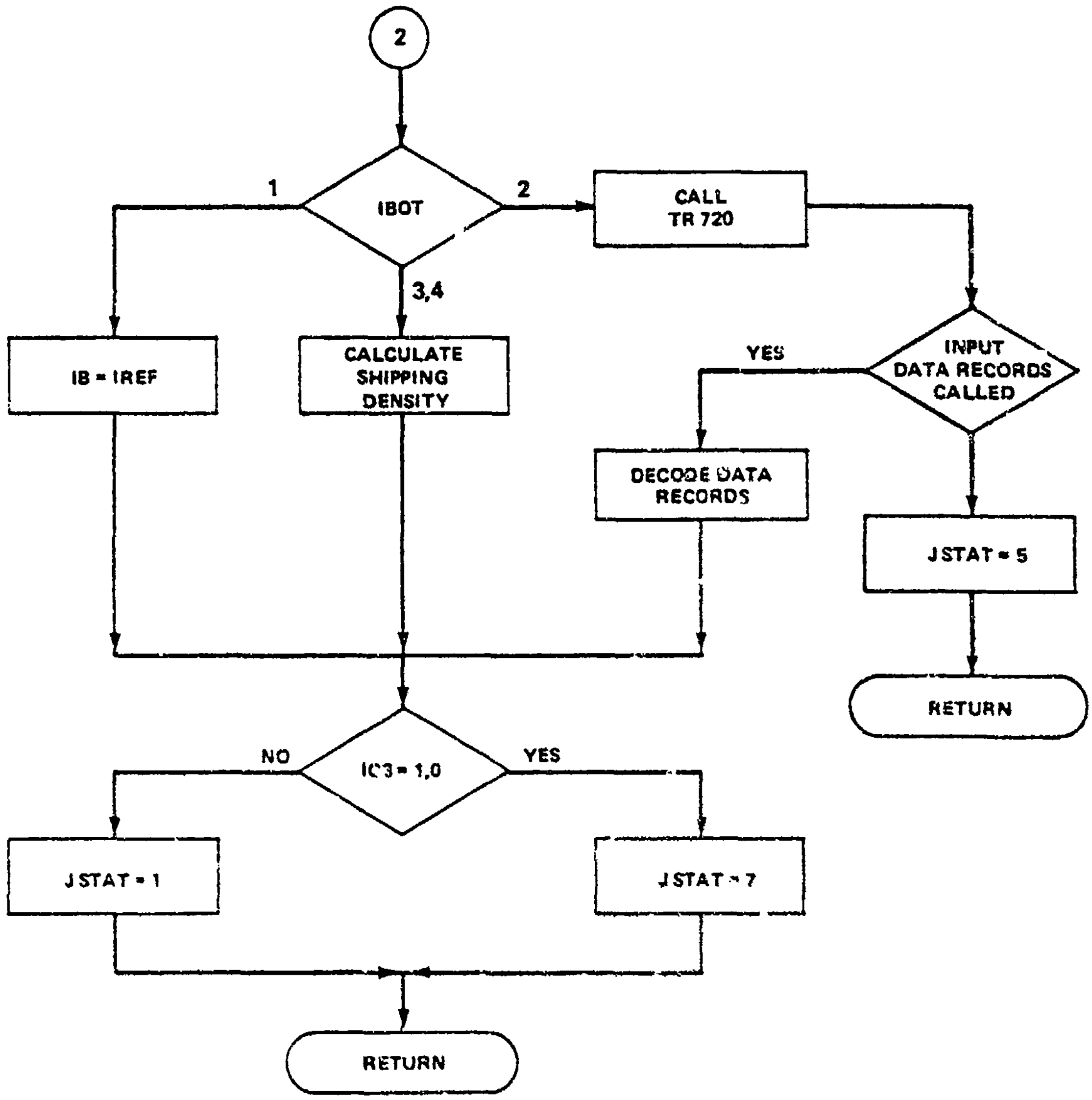


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

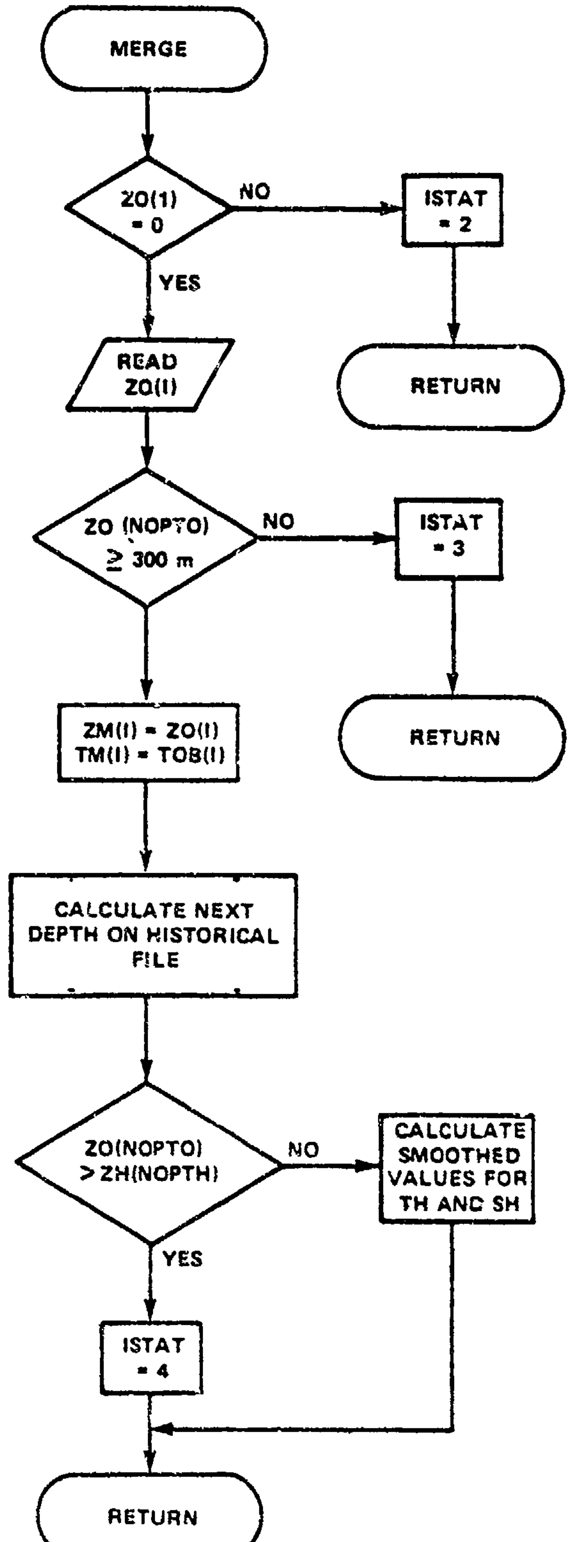


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

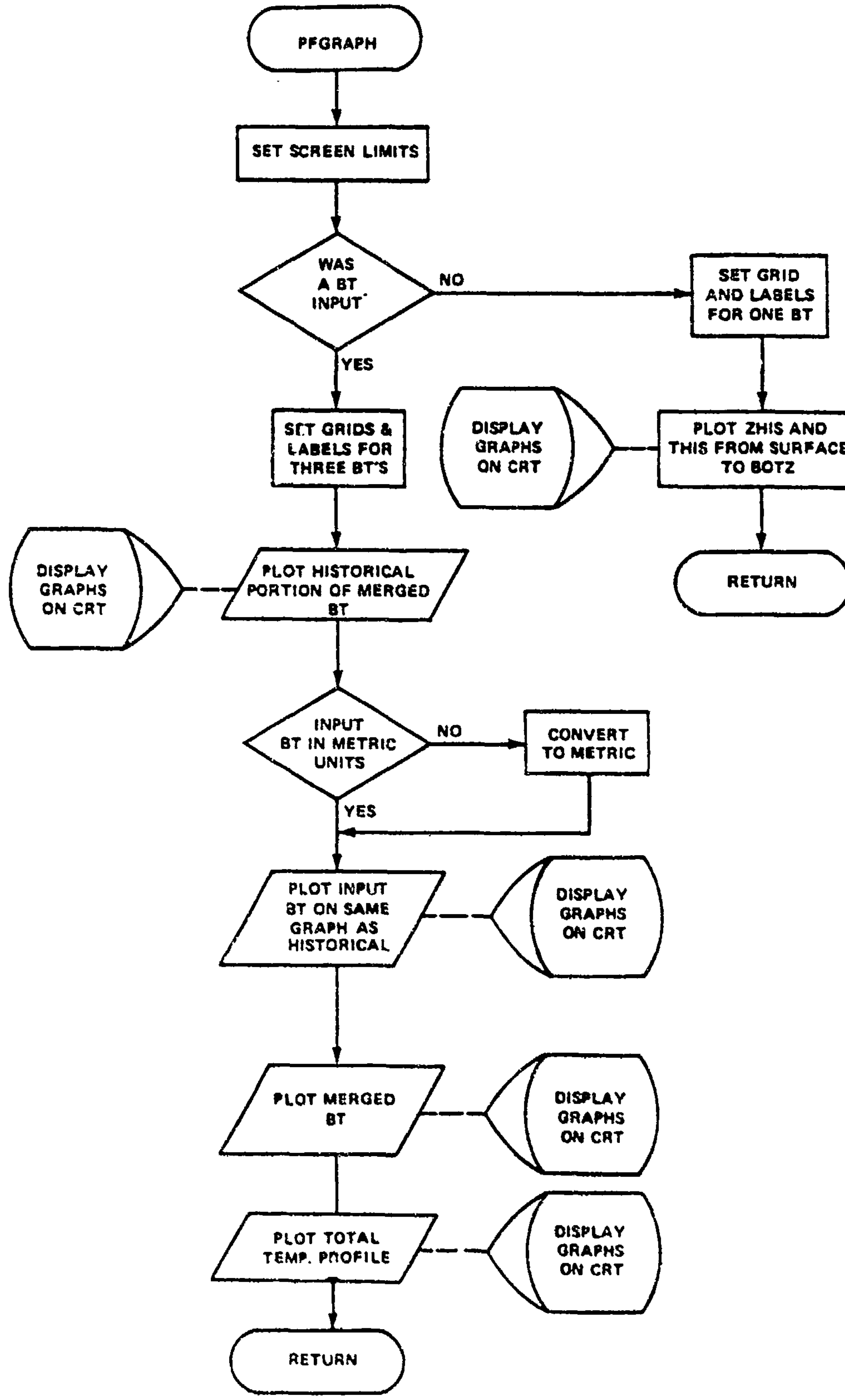


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

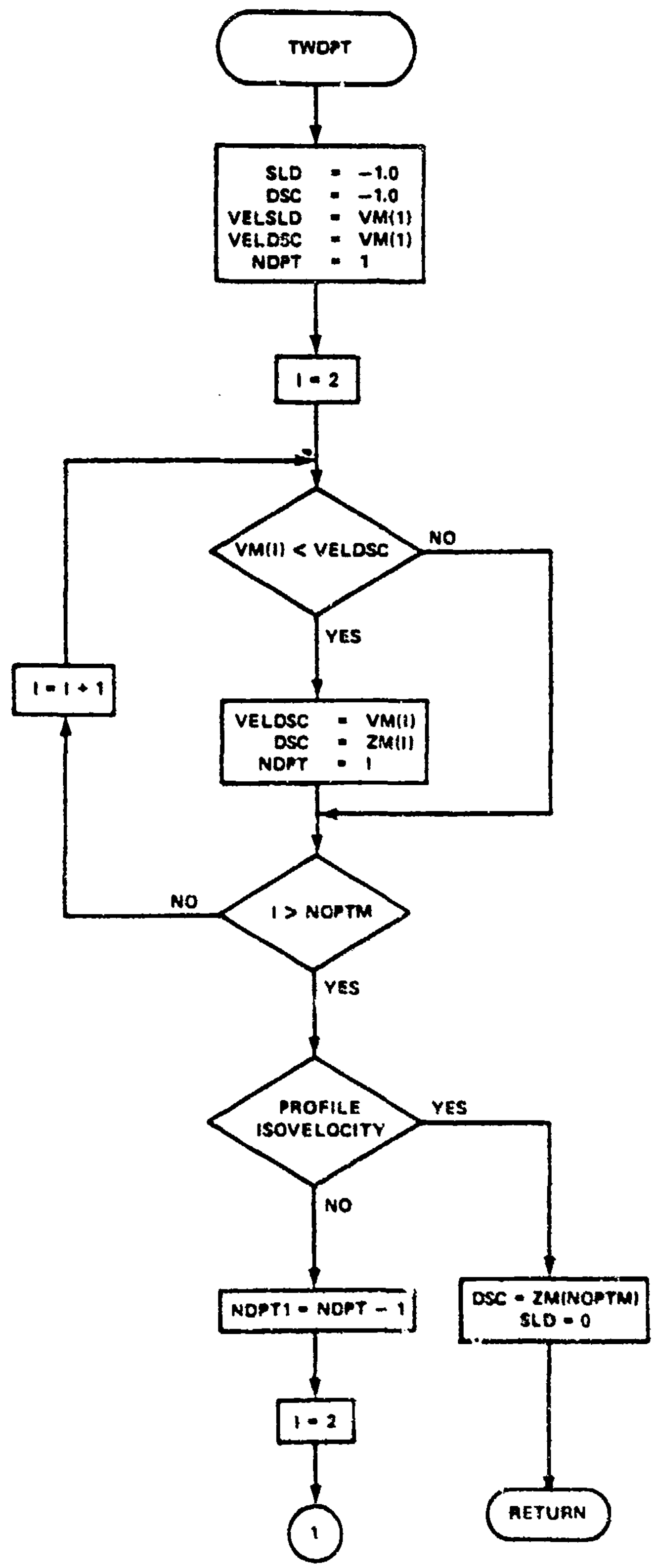


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

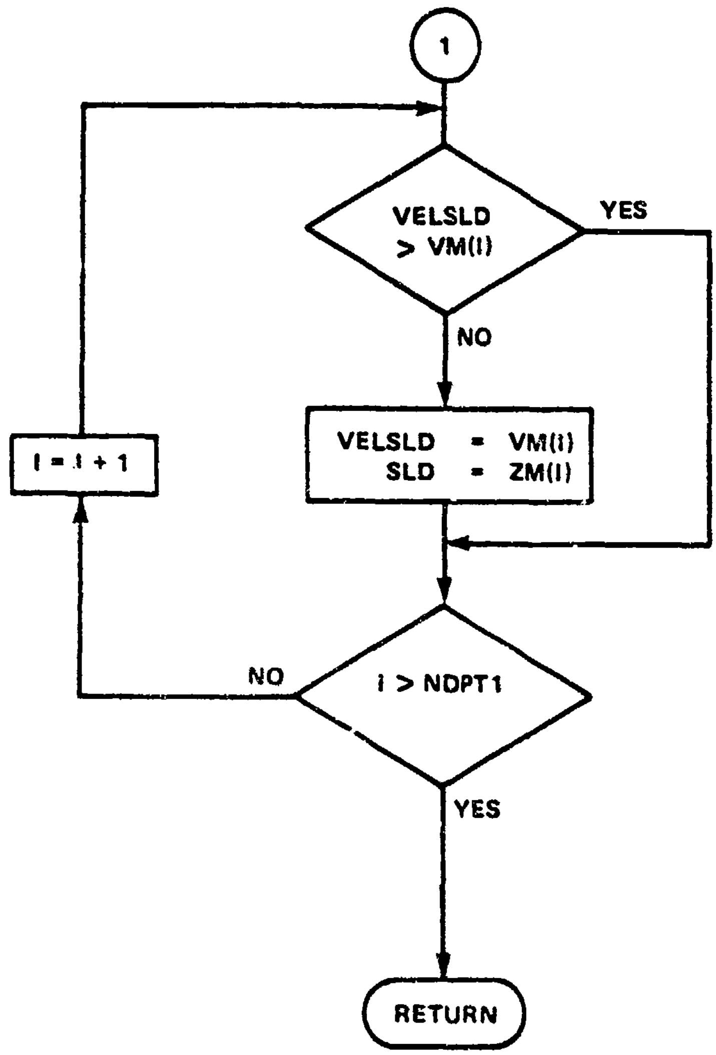


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

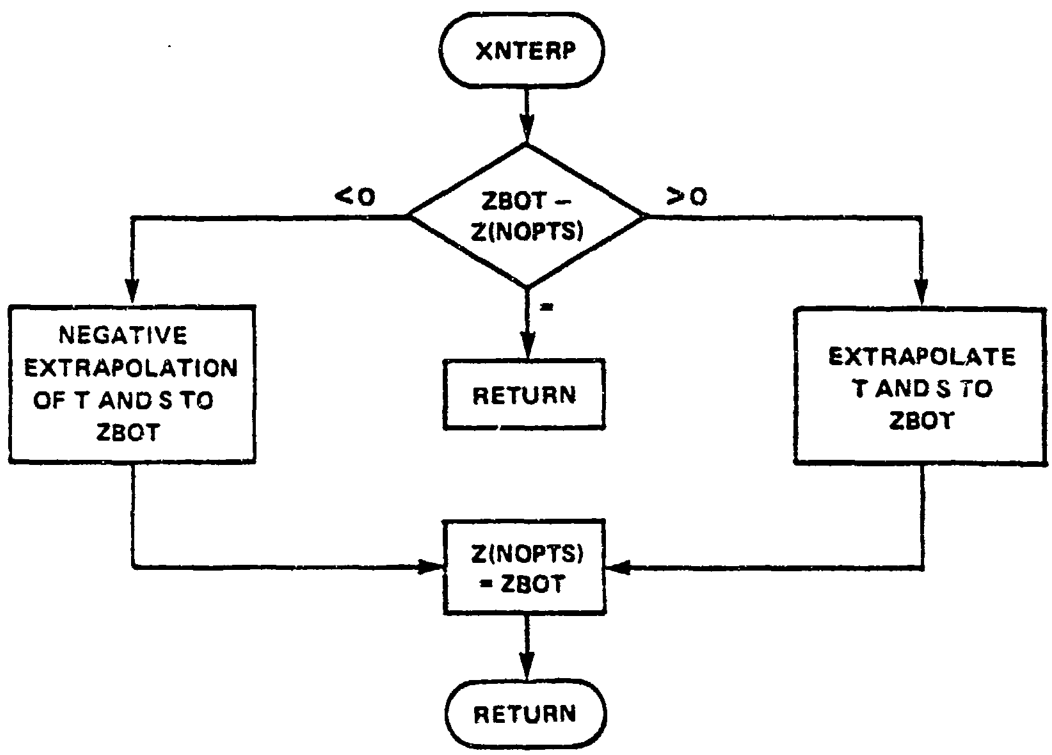


Figure 2-2. Flow Diagram of INPUT:OV Routine (continued)

2.2.3 Input Module Data Design

2.2.3.1 Data Files. There are four data files associated with the input module as shown in Table 2-5.

Table 2-5. Input Module Data Files

DATA FILE	DESCRIPTION
Target Data File	This file is a binary file containing integer precision data. Each record (block) of this file is 99 words long (9 rows by 11 columns).
Salinity-Temperature Versus Depth File	This file is a binary file containing integer precision data. Each record (block) of this file is 144 words in length.
Bottom Loss Data File	This file is a binary file containing integer precision data. Each record (block) of this file is 144 words in length. Bits 8-11 of the bottom index contain the value of the low frequency bottom loss province (BLP). Bits 12-15 contain the value of the high frequency BLP.
Shipping File	This file is a binary file containing integer precision data. Each record (block) of this file is 144 words in length.

2.2.3.1.1 Target Data File. The target data file is shown in Table 2-6. This data file contains target information as a function of target type and operational mode. There is one block of information for each target type with each block composed of 99 words of integer precision data. A block (or record) has 11 columns by 9 rows.

The first row of a data block contains administrative information. Column one is the block number while column two indicates whether the block contains data corresponding to a nuclear or diesel target. Target type (TYPE 1, 2, or 3) is found in column three. Column four contains an index corresponding to the number of valid operational modes for the target. The fifth column contains the frequency used for predicting

Table 2-6. Target Data File

COL. # / ROW #	1	2	3	4	5	6	7	8	9	10	11
1	BLOCK #	NUC/DIES.	TGT TYPE	# OF OP. MODES	PRED. FREQ. (BB)	# OF TONALS					
2	OP. MODE # 1	OP MODE # 2	OP. MODE # 3	OP. MODE # 4	OP. MODE # 5	OP. MODE # 6	OP. MODE # 7	OP. MODE # 8	OP. MODE # 9	OP. MODE # 10	OP. MODE # 11
3	OP DEPTH	OP DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH	OP. DEPTH
4	OP SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD	OP. SPD
5	L_S	L_S	L_S	L_S	L_S	L_S	L_S	L_S	L_S	L_S	L_S
6	TONAL # 1	TONAL # 2	TONAL # 3	TONAL # 4	TONAL # 5	TONAL # 6	TONAL # 7	TONAL # 8	TONAL # 9	TONAL # 10	TONAL # 11
7	SPL	SPL	SPL	SPL	SPL	SPL	SPL	SPL	SPL	SPL	SPL
8	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %	REL. %
9	STD DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD. DEV /RPM	STD. DEV /RPM

ROWS 5 9 ARE TEN TIMES GREATER THAN ACTUAL VALUE.

ROW 8 IS RELIABILITY RELIABILITY IS NOT AVAILABLE IF EQUAL TO ZERO.

ROW 9 IS STD DEV/RPM ROW 9 EQUALS STD. DEV FOR NUC'S OR ENGINE RPM FOR DIESELS.

IF EQUAL TO ZERO DATA NOT AVAILABLE.

detection ranges for broadband noise. The sixth column contains an index which indicates the number of frequencies in the target data block. Columns seven through 11 of the first row are not used.

Indexes corresponding to valid operational modes for the target are in the second row. If any column has an index of zero, this indicates an invalid operational mode for the target in question. The next row of the data block constitutes operational depths for the target. These are depths at which the target will be most commonly found for the corresponding operational mode. Row number four contains typical operating speeds for the target operational mode in question. The next row is broadband noise levels corresponding to the target operational mode.

The sixth row in the data block contains frequencies emitted by the target, with the next row being the SPLs that correspond to these frequencies. Reliabilities for the target emitted frequencies (0-100%) comprise the eighth row. The last row of the data block contains either standard deviations or values for engine revolutions per minute (RPM). If the target is a nuclear submarine, then this row contains standard deviations corresponding to the sound pressure levels, whereas this row contains engine RPM values (which determine the frequencies) for diesel submarine targets.

Rows five through nine are ten times greater than the actual values. Reliability (row 8) is equal to zero if no reliability value is available. Standard deviations or engine RPM values are not available if equal to zero.

2.2.3.1.2 Salinity-Temperature Versus Depth File. Seasonal environmental data files have been established for the northern hemisphere of the Atlantic and Pacific Oceans, the northern portion of the Indian Ocean, and the complete Mediterranean Sea. The major ocean basins are divided into convenient geographical areas, with each area further subdivided into 1-degree quadrangles. Each quadrangle is represented by an array of temperature and salinity values at standard depths from the surface to the bottom.

The northern hemisphere portion of the Atlantic Ocean is subdivided into five major areas, and the Pacific Ocean is subdivided into seven major areas. Boundaries of the North Atlantic and the North Pacific are shown in Figure 2-3, and the geographical locations of the environmental boundaries for these areas are shown in Table 2-7. The northern portion of the Indian Ocean is subdivided into two major ocean areas, and their boundaries are shown in Figure 2-4. As the Mediterranean Sea is small in comparison to the other areas, it is not subdivided into additional areas.

There are 60 salinity-temperature versus depth data files. Each file is assigned a file name comprised of seven letters based upon

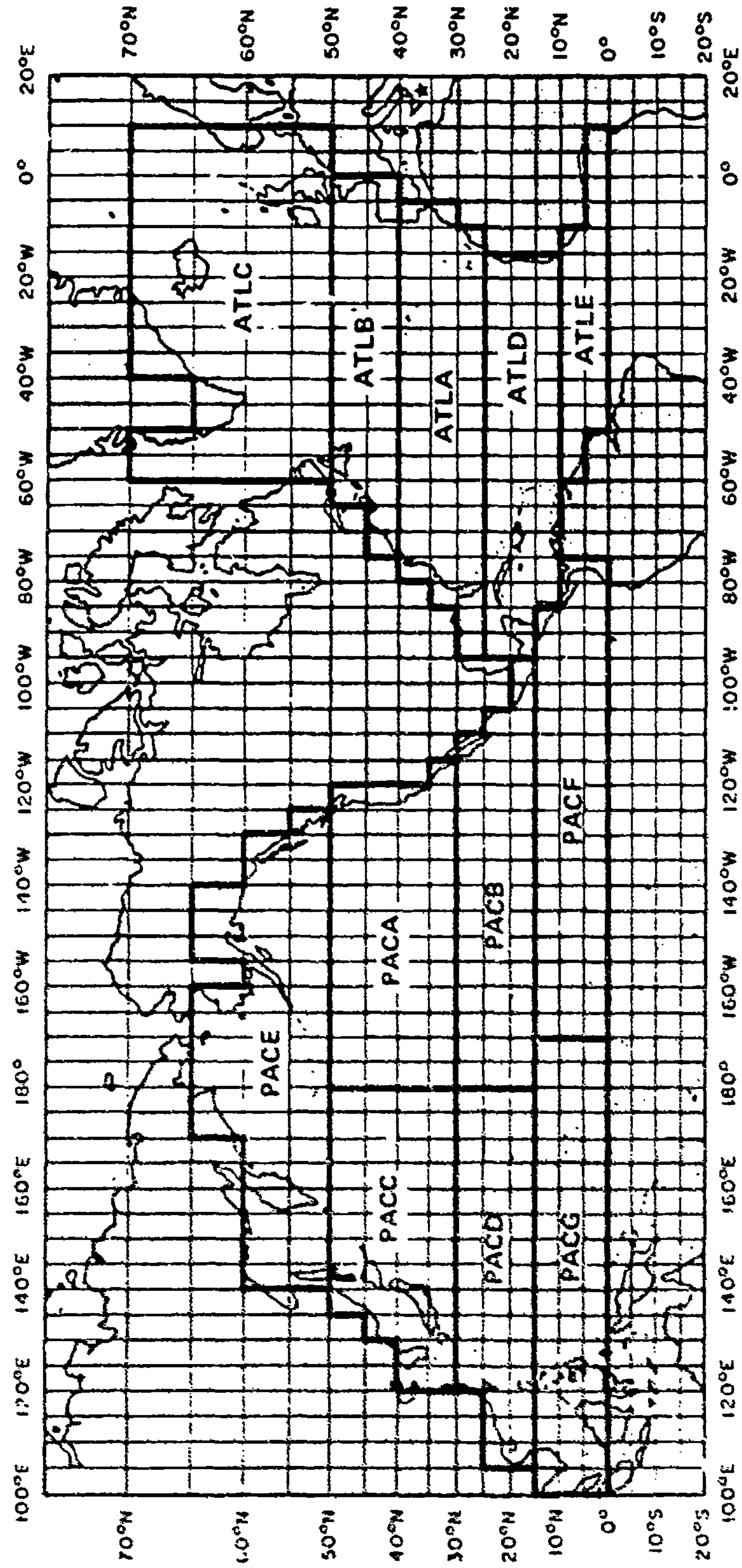


Figure 2-3. North Atlantic and North Pacific Environmental Area Boundaries

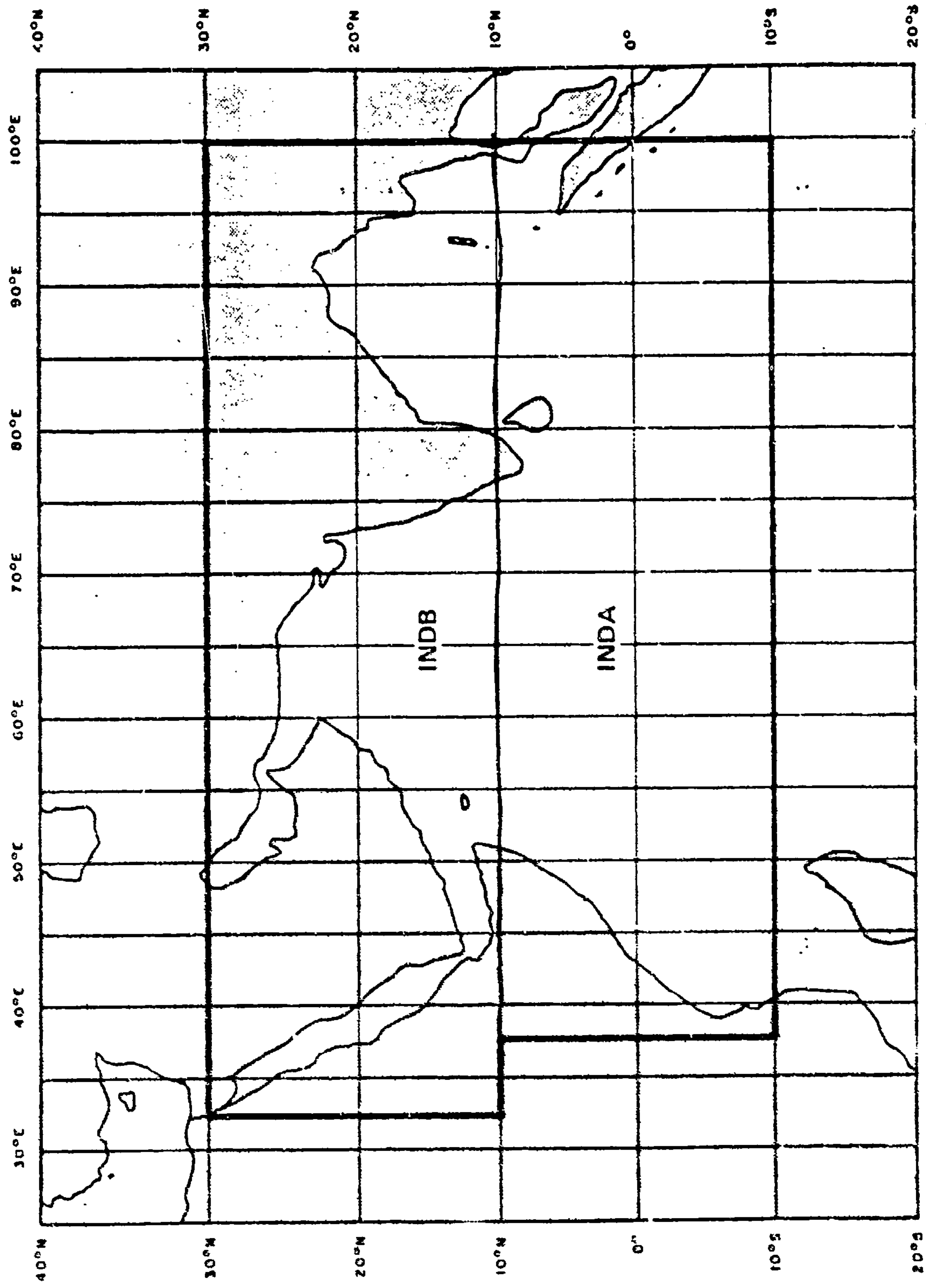


Figure 2-4. Indian Ocean Environmental Area Boundaries

geographic area and season. The first three letters refer to the ocean area (i.e., ATL-Atlantic, PAC-Pacific, IND-Indian, and MED-Mediterranean Sea), with the next letter indicating the ocean subdivision. Because the Mediterranean Sea is not subdivided, there are only six letters in its file name. A three-letter suffix specifies the season of the year (i.e., WIN-winter, SPR-spring, SUM-summer, and FAL-fall). Thus, the salinity-temperature data for the Pacific Ocean area B during the summer season would be found in the file named PACBSUM.

Table 2-7. Geographical Location of North Atlantic and North Pacific Environmental Data File

AREA	SOUTHERN BOUNDARY (deg-min)	NORTHERN BOUNDARY* (deg-min)	EASTERN BOUNDARY* (deg-min)	WESTERN BOUNDARY* (deg-min)
ATLA	25-00	39-59	NC	NC
ATLB	40-00	49-59	NC	NC
ATLC	50-00	NC	NC	NC
ATLD	10-00	24-59	NC	NC
ATLE	0-00	9-59	NC	NC
PACA	30-00	49-59	NC	179-59
PACB	15-00	29-59	NC	179-59
PACC	30-00	49-59	180-00	NC
PACD	15-00	29-59	180-00	NC
PACE	50-00	NC	NC	NC
PACF	0-00	14-59	NC	169-59
PACG	0-00	14-59	170-00	NC

*NC denotes no conflict with another boundary.

Each data file is composed of three sections. The first section is the "administrative" information necessary for the program to access the proper bathythermograph (BT) information. Section two consists of the reference numbers which the program uses as a pointer to extract the salinity, temperature, and depth (STD) information contained in the third section. All data blocks in each section are composed of 144 integers; unused locations are filled with zeroes. Each block is the same length which simplifies the programming necessary to extract the desired data.

One data block constitutes the first section of the file. Only the first 11 integers of this block are significant, with the remaining 133 integers being zero.

I1 Limit block = 18035 - signifies beginning of a data file

I2 Coding block identifying ocean area:

Pacific	=	29779
Atlantic	=	7395
Mediterranean	=	17748
Indian	=	-26283

I3 Season contained in data file

Winter = 1	January-March
Spring = 2	April-June
Summer = 3	July-September
Fall = 4	October-December

I4 Lowest limit of latitude covered by data file

I5 Maximum latitude limit of data file

I6 Lowest limit of longitude covered by data file

I7 Maximum longitude limit of data file

I8 Number of data blocks in second section

I9 Number of degrees longitude covered by data file

I10 Number of degrees latitude covered by data file

I11 Number of BT records in section three

In the second section, the number of data blocks varies due to the number of profiles necessary to provide complete coverage of the geographical area. For each block the first two integers are identifiers:

J1 Ocean area designator and is equivalent to I2 from section one

J2 Data block number

The remaining 142 integers are reference numbers for the profiles.

Depth, temperature, and salinity values are in the third section. As in the second section, the number of data blocks is variable. Information in each data block is as follows:

K1 Area identifier (equivalent to I2 and J1)

K2 Reference number of the data block

K3 Season identifier

K4 Number of depth, temperature, and salinity trios in the data block

K5 Water temperature at the surface in degrees centigrade multiplied by ten

K6 Salinity at the surface in parts per thousand (0/00) multiplied by ten

K7 Next profile depth in meters (this is usually 10 meters)

K8 Temperature at this depth in degrees centigrade times 10

K9 Salinity at this depth in 0/00 times 10

This sequence of depth, temperature, and salinity continues through the final set of numbers. Remaining spaces in the data block are filled with zeroes.

To access the requested profile, the computer arranges the reference numbers in a one-dimensional array. Based on the input latitude and longitude, the computer counts through the array until the requested 1-degree quadrangle is reached. The reference number in this location is assigned to a variable name. The computer searches the third block of data for the reference number. Upon finding the reference number, the data are read into a depth, temperature, and salinity file. This file is then used to generate an SVP.

2.2.3.1.3 Bottom Loss Data File. Data files containing bottom loss information have been developed for the same areas as the STD files. Construction of the bottom loss files is similar to that of the STD files. All areas, sub-areas, and quadrangle divisions are equivalent between the files. Each quadrangle is represented by the bottom loss classification associated with that location. Both the five-value classification used by the FACT model at low frequencies and the nine-value classification employed for higher frequencies are stored for each 1-degree quadrangle. Bottom loss files are named in the same manner as the STD files except there is no seasonal dependence, and the final three letters on each file are bottom loss province (BLP). Thus, the bottom loss file for the Atlantic Ocean area E would be named ATLEBLP.

Each data file is composed of two sections. As with the STD files, the first section contains the administrative information and is a single data block, and the second is the bottom loss values. In the administrative block, only the first ten integers are significant, with the remaining 134 being zeroes. These integers are equivalent to I1-I10 of the STD file, except for I3 which is the season identifier. Because there is no seasonality to the bottom loss files, this integer is equal to 5 in the bottom loss files. The first two integers of each block in the second section are equivalent to J1 and J2 of the STD file, with the remaining integers being the bottom loss values. These values are 16 times the bottom loss province for low frequency plus the high frequency bottom loss value.

To access the bottom loss information, the province numbers are arranged in a one-dimensional array, with the computer counting through the array, based on input longitude and latitude, until reaching the requested area.

2.2.3.1.4 Shipping Noise Data File. Shipping noise information is contained in data files for the Atlantic, Pacific, and Indian Oceans as well as the Mediterranean Sea. These files are similar in construction to the bottom loss files except there is only one data file for each ocean area. The Pacific, Atlantic, and Indian Ocean files have been subdivided into 5-degree quadrangles, with the Mediterranean Sea file having 1-degree subdivisions. Each quadrangle is represented by a historical average for the number of large merchant ships and fishing vessels. Shipping noise files are named in the same manner as the bottom loss files except there are no subdivisions within an ocean area, and the last four letters are: SHIP. Thus, the Pacific shipping noise file is named PACSHIP.

Each data file is composed of two sections with the first containing administrative information and the second the shipping values. In the administrative block only the first ten integers are meaningful with the remaining 134 being zeroes. These integers are equivalent to those corresponding in the bottom loss files. The first two integers in each block of the second section also correspond to those in the bottom loss files with the remaining being the shipping values. These values are merchants plus one-tenth of the fishing vessels; the sum times 100.

To access the shipping information, the province numbers are arranged in a one-dimensional array with the computer counting through the array, based on input longitude and latitude, until reaching the requested area.

2.2.3.2 Tables. The following are the data base tables and arrays used by the TASSRAP II input module with the size and type of each array denoted in the parenthesis (e.g., T (50) - 50-element, single-dimension array). Arrays with mnemonic names beginning with the letter I, J, K. L, M, or N, with the exception of LEVELN, contain integer precision data (one 16-bit word). All other arrays contain standard precision floating point data (two 16-bit words):

DEP (31) - Depths of the in situ BT; operator input in meters or feet.

FREQ (2, 5) - Frequencies and SPLs on which to optimize detection performance selected from target file based upon target type: row 1 contains frequencies, row 2 contains SPLs.

FREQ (24, 6) - Frequencies for beam noise data input by the operator: column 1 contains the beam numbers; columns 2 through 6 contain the frequencies.

IBEAM (24) - Beam numbers for beam noise; operator input.

IFRQ (4, 11) - An intermediate frequency file containing frequency, SPL, and reliability information.

LABEL (10) - Alphanumeric label up to 20 characters including spaces; operator input.

LEVELN (24, 6) - Level of beam noise for each frequency input by operator: column 1 contains beam number; columns 2 through 6 contain levels.

S (50) - Historical salinity in parts per thousand for the various depths; selected from environmental data file.

SM (50) - Array of salinity in parts per thousand versus depth; obtained from historical data and interpolated for BT input depths.

T (50) - Historical temperatures in degrees centigrade for the various depths; selected from environmental data file.

TEMP (31) - Array of input temperature versus depth; entered by operator in degrees centigrade or degrees Fahrenheit.

TM (50) - Array of merged temperature versus depth; obtained from historical data and input BT.

TOB (31) - Entered temperature versus depth in degrees centigrade.

VM (50) - Velocity of sound versus depth; calculated by Wilson's equations.

Z (50) - Depth of historical temperature and salinity; selected from environmental data file.

ZM (50) - Depths of merged temperature and salinity; obtained from historical data and input BT.

ZO (31) - Depths of the in situ BT in meters; obtained from the input BT depth.

2.2.3.3 Variables. Variables and constants in the data base used by the input module are included in the following list along with a detailed description of each. Names beginning with I, J, K, L, M, or N are integer precision variables (one 16-bit word); all others contain single precision floating point data (two 16-bit words). The variable LFRQLM also contains two-word floating point data.

BOTZ - Depth of ocean in meters or feet. This variable may be operator input or retrieved from data file.

DSC - Depth of deep sound channel.

IDA - Numerical value of the day; operator input.

IDATE - Date group (day, month, year); operator input.

IHFBLP - High frequency bottom loss province.

ILFBLP - Low frequency bottom loss province.

IMO - Numerical value of the month; operator input.

INUMFRQ - Number of target frequencies read in target data file.

ISEA - Season read from data file.

ITIME - Time group, twenty-four (24) hour clock; operator input.

ITYPE - Target type.

IYR - Numerical representation of year; operator input.

JMAX - Maximum number of degrees of latitude in data files.

JSEA - Numerical value of season calculated from input month.

LAT - Latitude, four digits (0000-9000) with the last two being minutes; operator input.

LFRQLM - A floating point variable which contains the lower frequency limit of the sonar.

LON - Longitude, up to five digits (00000-18000) with the last two being minutes; operator input.

NB - Number of beams for which measured noise is to be an input.

NDP - Number of data points in input BT.

NF - Number of input target frequencies.

NF1 - Number of input beam noise frequencies.

NPOINT - Number of points in historical array covered by input data.

NZP - NDP + 1

PRDFRQ - Predicted frequency.

RANGE - Maximum range in nautical miles; operator input.

SLD - Surface layer depth in meters; selected from sound velocity profile.

SHPDEN - Shipping density.

SS - Own-ship speed in knots; operator input.

TGTBBN - Target broadband noise; retrieved from target data file.

TGTDEP - Target depth in feet; retrieved from target data file or operator input.

TGTSPD - Target speed in knots; retrieved from target file.

UFRQLM - Upper frequency limit of a sonar.

VELDSC - Velocity at deep sound channel.

VELSLD - Velocity at surface layer depth.

WH - Wave in feet; operator input.

WS - Wind speed in knots; operator input.

XDEP - Depth of input BT modified from previous input.

XLATMN - Minimum latitude covered by a data file.

XLATMX - Maximum latitude covered by a data file.

XLONMN - Minimum longitude covered by a data file.

XLONMX - Maximum longitude covered by a data file.

XTEMP - Temperature value of input BT modified from previous inputs.

2.2.3.4 Flags. There are several flags used by the input module and associated subroutines in the data base. The following is a list and detailed description of each flag. All flags are integer precision variables:

IDN - Integer which indicates whether the target data retrieved from TGTFL are for a diesel or a nuclear submarine: D = diesel, N = nuclear.

IEW - Integer to denote east (1) or west (2) longitude; operator input.

INS - Integer to denote north (1) and south (2) latitude; operator input.

IPROF - Denotes whether or not a BT was entered: 1 = input, 2 = no input.

ITYPE - Integer correlating target type to the retrieved target data.

MOE - An indicator which denotes whether the BT data was entered in metric or English units: 1 = metric, 2 = English.

2.2.3.5 Indexes. All the indexes used in the data base are integer precision variables (one 16-bit word); each index is listed below along with a detailed description:

IB - Integer representation of the bottom loss class; obtained from environmental file. Bits 8-11 of this variable contain the value of the low frequency bottom loss class, and bits 12-15 contain the value of the high frequency bottom loss class.

INUMDPS - The number of array depths contained in the tow depth file.

INUMFRQ - The number of frequencies contained in the target frequency file and in the noise data file.

ISONAR - Integer representation of type of sonar system; operator input.

IST - Numerical value representing own-ship type of mission.

ITGT - Integer representation of the target type; operator input.

ITOM - Integer representation of the target operational mode; operator input.

JI (10) - An array of indexes used by the BT data input routine.

NOPTM - Number of data points in the merged data file; obtained from data file and BT input.

NOPTS - Number of data points in retrieved data file; obtained from data file.

NDP - Number of points in an input BT; operator input.

2.2.3.6 Common Data Base Reference. This subsection provides a list of all references to local and common data base items and location of each reference. The list is divided in three parts which parallel subsections 2.2.3.2, 2.2.3.3, and 2.2.3.4. Those items carried through in the primary communications area are denoted PCA.

DEP (31) - PCA, INPUT, BTGRAPH, INPUT:OV

FREQ (2, 5) - PCA, SLFRQ

FREQN (24, 6) - PCA, INPUT

IBEAM (24) - PCA, INPUT

IFRQ (4, 11) - Labeled common TGT, GETTGT, SLFRQ

LABEL (10) - PCA, INPUT

LEVELN (24, 6) - PCA, INPUT

S (50) - PCA, INPUT:OV, GETENV, MERGE, XNTERP, WILSON, XNTF

SM (50) - PCA, INPUT:OV, GETENV, MERGE, WILSON, XNTERP, XNTF

T (50) - PCA, INPUT:OV, GETENV, MERGE, XNTERP, XNTF, PFGRAPH, WILSON

TEMP (31) - PCA, INPUT, BTGRAPH, INPUT:OV

TM (50) - PCA, INPUT:OV, MERGE, XNTERP, XNTF, PFGRAPH, WILSON

TOB (31) - PCA, INPUT:OV, MERGE, PFGRAPH

VM - PCA, INPUT:OV, MERGE, WILSON, TWDPT

Z - PCA, INPUT:OV, GETENV, MERGE, WILSON, PFGRAPH, XNTERP, XNTF

ZM - PCA, INPUT:OV, MERGE, WILSON, PFGRAPH, XNTERP, XNTF

ZO	-	PCA, INPUT:OV, MERGE, PFGRAPH
BOTZ	-	PCA, INPUT, INPUT:OV, GETENV, PFGRAPH
DSC	-	PCA, INPUT:OV, TWDPT
IB	-	PCA, INPUT:OV, GETENV
IDA	-	PCA, INPUT
IDATE	-	PCA, INPUT
IHFBLP	-	INPUT:OV
ILFBLP	-	INPUT:OV
IMO	-	INPUT, INPUT:OV
INUMFRQ	-	PCA, GETTGT, SLFRQ
ITIME	-	PCA, INPUT
ITYPE	-	Common TGT, GETTGT
IYR	-	INPUT
JSEA	-	Common TEMP, INPUT:OV, GETENV
LAT	-	PCA, INPUT, INPUT:OV
LFRQLM	-	GETSONAR
LON	-	PCA, INPUT, INPUT:OV
NB	-	PCA, INPUT
NDP	-	PCA, INPUT, INPUT:OV
NF	-	INPUT
NF1	-	PCA, INPUT
NPOINT	-	MERGE
NZP	-	INPUT:OV
PRDFRQ	-	Common TGT, GETTGT, SLFRQ

RANGE	-	PCA, INPUT
SLD	-	PCA, TWDPT, INPUT:OV
SHPDEN	-	PCA, GETENV, INPUT:OV
SS	-	PCA, INPUT
TGTBBN	-	PCA, GETTGT
TGTDEP	-	PCA, GETTGT, INPUT
TGTSPD	-	PCA, GETTGT
UFRQLM	-	GETSONAR
VELDSC	-	TWDPT
VELSLD	-	TWDPT
WH	-	PCA, INPUT
WS	-	PCA, INPUT
XDEP	-	INPUT
XLATMN	-	GETENV
XLATMX	-	GETENV
XLONMN	-	GETENV
XLONMX	-	GETENV
XTEMP	-	INPUT
IB	-	PCA, GETENV, INPUT:OV
INUMFRQ	-	PCA, GETTGT, SLFRQ
ISONAR	-	PCA, INPUT
IST	-	PCA, INPUT
ITGT	-	PCA, INPUT
ITOM	-	PCA, INPUT, GETTGT

J1 (10) - INPUT

NOPTM - PCA, INPUT:OV, MERGE, TWDPT

NOPTS - PCA, INPUT:OV, GETENV, XNTERP, XNTF, PFGRAPH, TWDPT

NDP - INPUT

2.3 INPUT/OUTPUT FORMATS All inputs to the input module are entered via accept statements. Pages 2-52 through 2-62 present I/O when the operator is employing the automatic mode while pages 2-63 through 2-88 present I/O when every input option is exercised. To make entries, the operator answers the questions presented or responds to a prompter. The examples presented on the succeeding pages illustrate the program output and the appropriate operator response.

2.4 REQUIRED SYSTEM LIBRARY SUBROUTINES

SYSTEM SUBROUTINE NAME	USED	DOCUMENT REFERENCE
AINT (truncation of real number)	INPUT:OV	Data General FORTRAN IV User's Manual
AMAX1 (choose maximum value of real numbers)	INPUT:OV	Data General FORTRAN IV User's Manual
AMIN1 (choose minimum value of real numbers)	INPUT:OV	Data General FORTRAN IV User's Manual
FLOAT (convert from integer to real)	GETTGT GETENV SLFRQ	Data General FORTRAN IV User's Manual
IABS (absolute value of integer)	SLFRQ GETENV	Data General FORTRAN IV User's Manual
IFIX (convert from real to integer by truncation)	GETENV	Data General FORTRAN IV User's Manual
INT (convert from real to integer by multiplying the sign of the argument by the largest integer)	INPUT:OV	Data General FORTRAN IV User's Manual

2.5 CONDITIONS FOR INITIATION This section describes the system conditions that must be met for each subroutine to be initiated. For those routines that are always initiated, the word "UNCONDITIONAL" is shown.

```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL = TASSRAP
 2 DAY = 1
 3 MONTH = 10
 4 YEAR = 77
 5 TIME = 1000
 6 LATITUDE = 3000
 7 NORTH(1)-SOUTH(2) = 1
 8 LONGITUDE = 6500
 9 EAST(1)-WEST(2) = 2
10 MAXIMUM RANGE(NM) = 175
11 WAVE HEIGHT(FT) = 3
12 WIND SPEED(KT) = 10
13 SHIP SPEED(KTS) = 5

CHANGE ANY DATA?  1=YES 0=NO---0
```

```
                    ***** TASSRAP INPUT PROGRAM *****

**** TARGET TYPE ****

     1)SOVIET NUCLEAR TYPE 1
     2)SOVIET NUCLEAR TYPE 2
     3)SOVIET NUCLEAR TYPE 3
     4)SOVIET DIESEL TYPE 1 (F,R,W,Z)
     5)SOVIET DIESEL  JULIET (TYPE 2)
     6)SOVIET DIESEL FOXTROT (TYPE 3)
     7)US NUCLEAR SSN 637 CLASS
     8)OWN SOURCE LEVELS
WHICH TARGET TYPE?---1


**** TARGET OPERATIONAL MODE ****

     1)TRANSIT
     2)AREA SEARCH-ASW
     3)AREA SEARCH-SURFACE SHIPS
     4)BARRIER
     5)CONVOY PENETRATION
     6)AMPHIBIOUS ATTACK
     7)HVU ATTACK
     8)SSBN OPERATIONS
     9)SURVEILLANCE-ASW
     10)SURVEILLANCE-SURFACE SHIPS
     11)SNORKEL
     12)INPUTSOURCE DEPTH
WHICH TARGET OPERATION MODE?---1
```

```
##### TASSRAP INPUT PROGRAM #####

**** OWN SHIP TYPE OF MISSION ****

      1)SURVEILLANCE
      2)ESCORT
      3)TRAIL
      4)AREA SANITIZATION
      5)AMPHIBIOUS ASSAULT PROTECTION
WHICH TYPE OF MISSION?---1


**** SONAR TYPE ****
      1)AN/SQR-15
      2)AN/BQR-15
      3)STASS
      4)TACTASS
      5)LAMBDA
WHAT TYPE OF SONAR?---4
```

```
##### TASSRAP INPUT PROGRAM #####

INPUT BT?  1=YES Ø=NO---Ø
DO YOU WISH TO ENTER A BOTTOM DEPTH
YES(1)-NO(Ø) ?Ø
```

```
***** TASSRAP INPUT PROGRAM *****

 DO YOU WISH TO INPUT MEASURED BEAM NOISE DATA ?
YES (1) -NO (0) ? 0
```

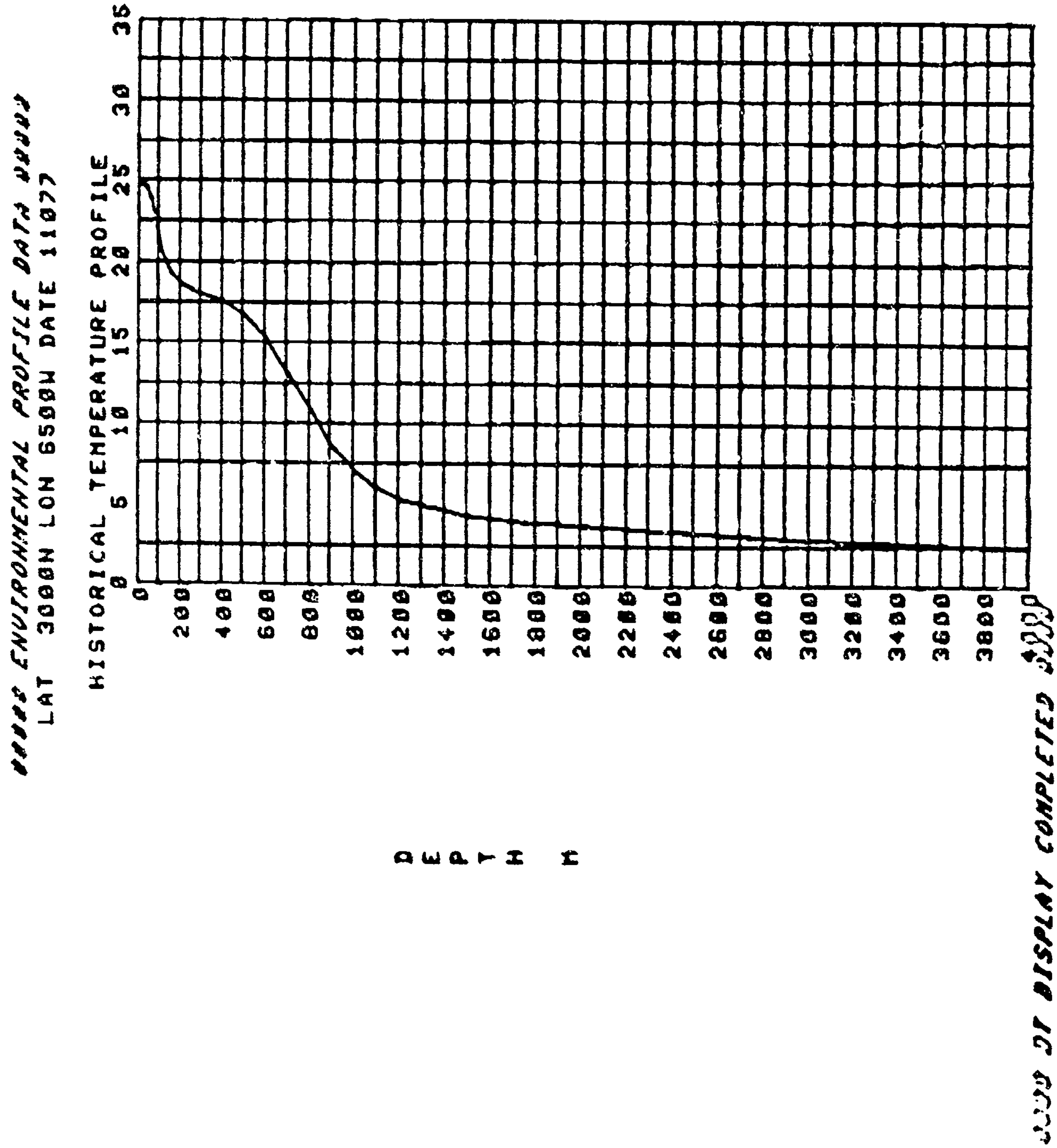
```
##### TASSRAP INPUT PROGRAM #####

1 LABEL = TASSRAP
2 DAY =     1
3 MONTH =    10
4 YEAR =    77
5 TIME =    1000
6 LATITUDE =    3000
7 NORTH(1)-SOUTH(2) =     1
8 LONGITUDE =    6500
9 EAST(1)-WEST(2) =     2
10 MAXIMUM RANGE(NM) =  175.0
11 WAVE HEIGHT(FT) =     3.0
12 WIND SPEED(KTS) =    10.0
13 SHIP SPEED(KTS) =     5.0
14 TARGET TYPE =     1
15 TARGET OP. MODE =     1
16 TYPE OF MISSION =     1
17 SONAR TYPE =     4
18 CHANGE BT INPUT
19 CHANGE BEAM NOISE DATA
20 BOTTOM DEPTH

CHANGE ANY DATA?  1=YES 0=NO---0
```

```
##### ENVIRONMENTAL PROFILE DATA #####
          LAT 3000N LON 6500W DATE  11077
H. FREQ BLP  7    L. FREQ BLP  3    SHIP. DEN.  1.0084500E -3
                        RETRIEVED DATA
```

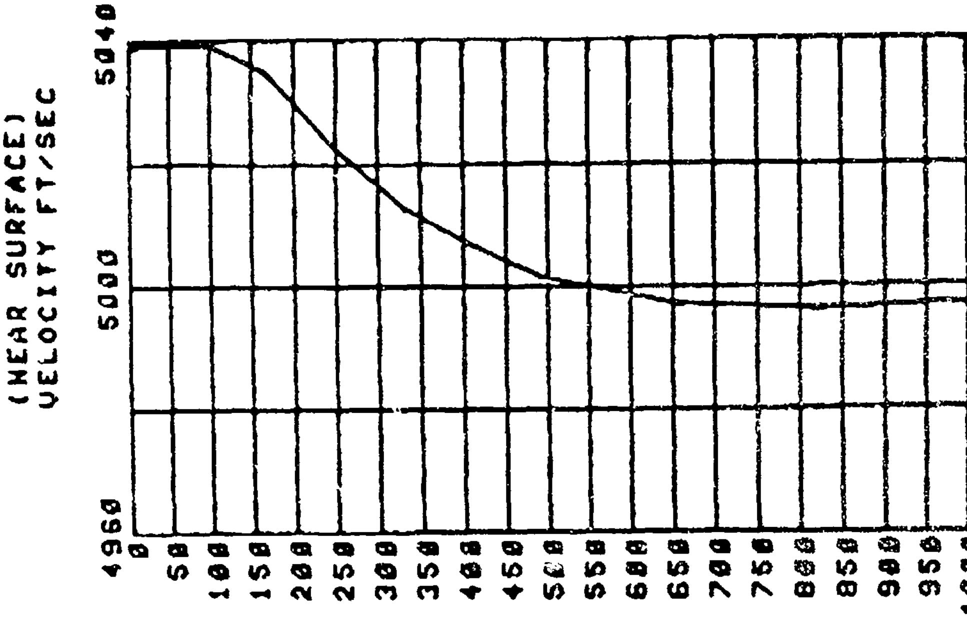
	DEP (M)	TEMP (C)	SAL (PPT)	VEL (M/SEC)
	0.	24.79	36.39	1535.92
	9.	24.79	36.39	1536.38
SLD>	19.	24.69	36.49	1536.13
	29.	24.59	36.49	1536.05
	50.	23.89	36.49	1534.71
	75.	22.19	36.59	1531.02
	99.	20.89	36.59	1528.06
	125.	19.99	36.59	1526.06
	149.	19.29	36.59	1524.54
	199.	18.59	36.49	1523.28
	250.	18.19	36.49	1522.96
	299.	17.99	36.49	1523.21
	399.	17.49	36.39	1523.27
	500.	16.69	36.29	1522.42
	599.	15.19	35.99	1519.09
	699.	13.09	35.69	1513.56
	799.	10.79	35.39	1506.91
	899.	8.49	35.19	1499.80
	999.	6.99	35.09	1495.65
	1099.	5.89	35.09	1492.96
	1199.	5.29	35.09	1492.20
DSC>	1299.	4.89	34.99	1492.10
	1399.	4.59	34.99	1492.54
	1499.	4.29	34.99	1492.98
	1749.	3.89	34.99	1495.52
	1999.	3.69	34.99	1498.91
	2500.	3.19	34.99	1505.32
	2999.	2.79	34.69	1512.09
	3995.	2.29	34.89	1527.42

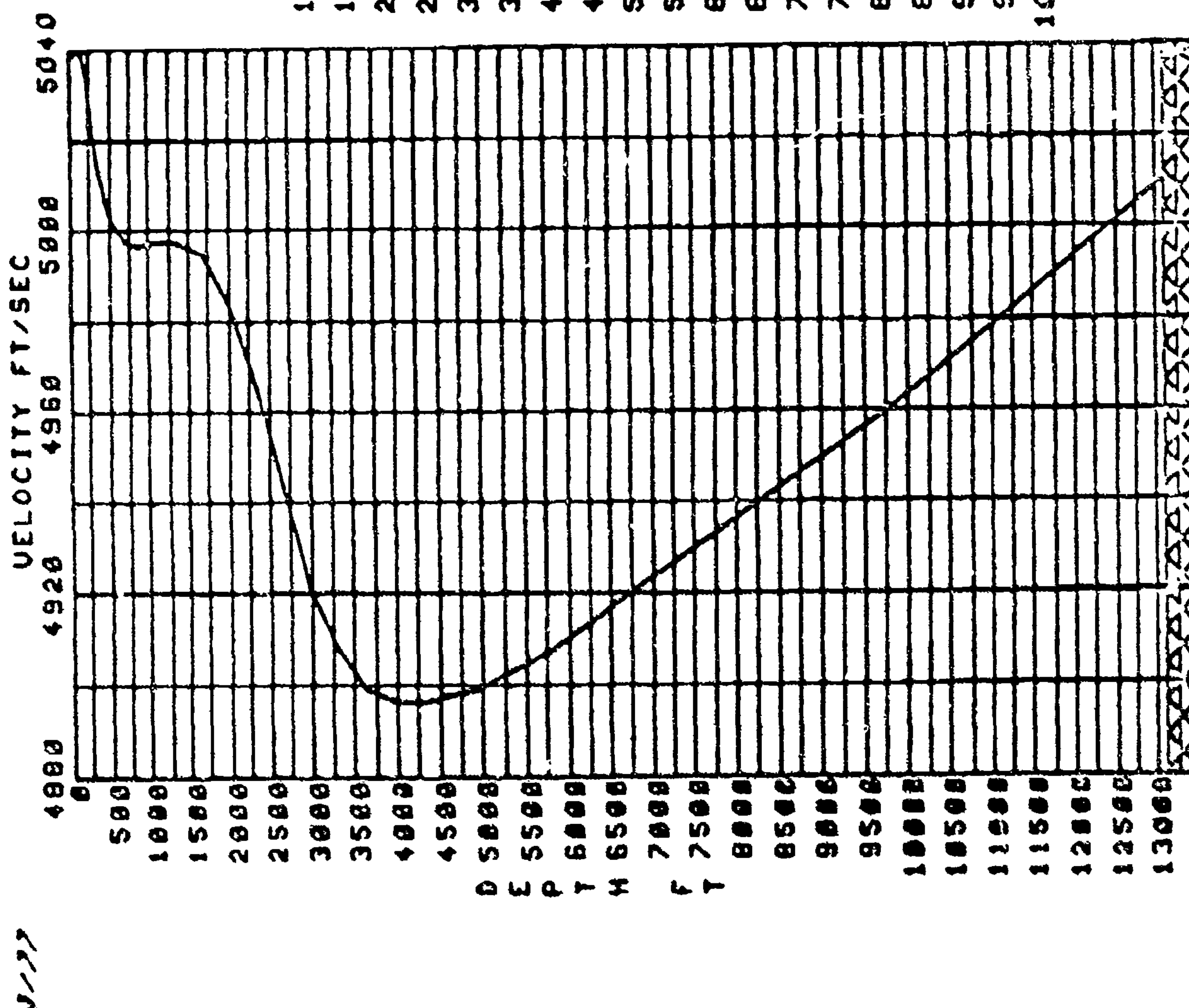
00000 PROFILE COMPLETE 00000

```
OUTPUT TEMP. PROFILE?  1=YES 0=NO---1
```


ENVIRONMENTAL PROFILE DATA
LAT 3000N LON 6500W DATE 11077
HISTORICAL TEMPERATURE PROFILE
0 5 10 15 20 25 30 35
0 200 400 600 800 1000 1200 1400 1600 1800 2000 2200 2400 2600 2800 3000 3200 3400 3600 3800 4000
DEPTH M
OF DISPLAY COMPLETED

```
OUTPUT SUP?  1=YES 0=NO---1
UNITS OF DATA, 1=METRIC, 2=ENGLISH---2
```


(NEAR SURFACE)
VELOCITY FT/SEC
4960
5000
5040
0
50
100
150
200
250
300
350
400
450
500
550
600
650
700
750
800
850
900
950
1000


VELOCITY FT/SEC
4880
4920
4960
5000
5040
DEPTH FT
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
5500
6000
6500
7000
7500
8000
8500
9000
9500
10000
10500
11000
11500
12000
12500
13000
13500

```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL = TASSRAP
 2 DAY = 26
 3 MONTH = 6
 4 YEAR = 77
 5 TIME = 1100
 6 LATITUDE = 3500                    6 LATITUDE = 3530
 7 NORTH(1)-SOUTH(2) = 1
 8 LONGITUDE = 3500
 9 EAST(1)-WEST(2) = 2
10 MAXIMUM RANGE(NM) = 250
11 WAVE HEIGHT(FT) = 3
12 WIND SPEED(KT) = 3
13 SHIP SPEED(KTS) = 3

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---6
```

```
***** TASSRAP INPUT PROGRAM *****

1 LABEL = TASSRAP
2 DAY =    26
3 MONTH =     6
4 YEAR =    77
5 TIME =   1100
6 LATITUDE =   3530
7 NORTH(1)-SOUTH(2) =     1
8 LONGITUDE =   3500
9 EAST(1)-WEST(2) =     2
10 MAXIMUM RANGE(NM) =  250.0
11 WAVE HEIGHT(FT) =    3.0
12 WIND SPEED(KTS) =    3.0
13 SHIP SPEED(KTS) =    3.0

CHANGE ANY DATA?  1=YES 0=NO---0
```

```
@@@@@ TASSRAP INPUT PROGRAM @@@@@

@@@@ TARGET TYPE @@@@

     1)SOVIET NUCLEAR TYPE 1
     2)SOVIET NUCLEAR TYPE 2
     3)SOVIET NUCLEAR TYPE 3
     4)SOVIET DIESEL TYPE 1 (F,R,W,Z)
     5)SOVIET DIESEL  JULIET (TYPE 2)
     6)SOVIET DIESEL FOXTROT (TYPE 3)
     7)US NUCLEAR SSN 637 CLASS
     8)OWN SOURCE LEVELS
WHICH TARGET TYPE?---8


MAXIMUM NUMBER OF FREQUENCIES = 5
NUMBER OF FREQUENCIES = 5
INPUT TARGET DATA IN FREQUENCY - SOURCE LEVEL PAIRS
20000,210
18000,200
14999,195
14000,190
12000,180
```

```
000 FREQUENCY INPUT DATA 000

LINE FREQUENCY LEVEL

    1    019959   210.0

    2    017999   200.0

    3    014999   195.0

    4    013999   190.0

    5    011999   180.0
CHANGE ANY OF THE  5
FREQUENCY-LEVEL PAIRS (1=YES,0=NO)?=1
NUMBER OF POINTS TO BE CORRECTED =3
INPUT LINE NUMBER AND CORRECT FREQUENCY-LEVEL PAIRS

01,21000,210

02,16000,195

03,15000,193
```

```
*** FREQUENCY INPUT DATA ***

LINE FREQUENCY LEVEL

    1    *20999   210.0

    2    *16999   195.0

    3    *14999   193.0

    4    *13999   190.0

    5    *11999   180.0
CHANGE ANY OF THE  5
FREQUENCY-LEVEL PAIRS (1=YES,0=NO)?=0
```

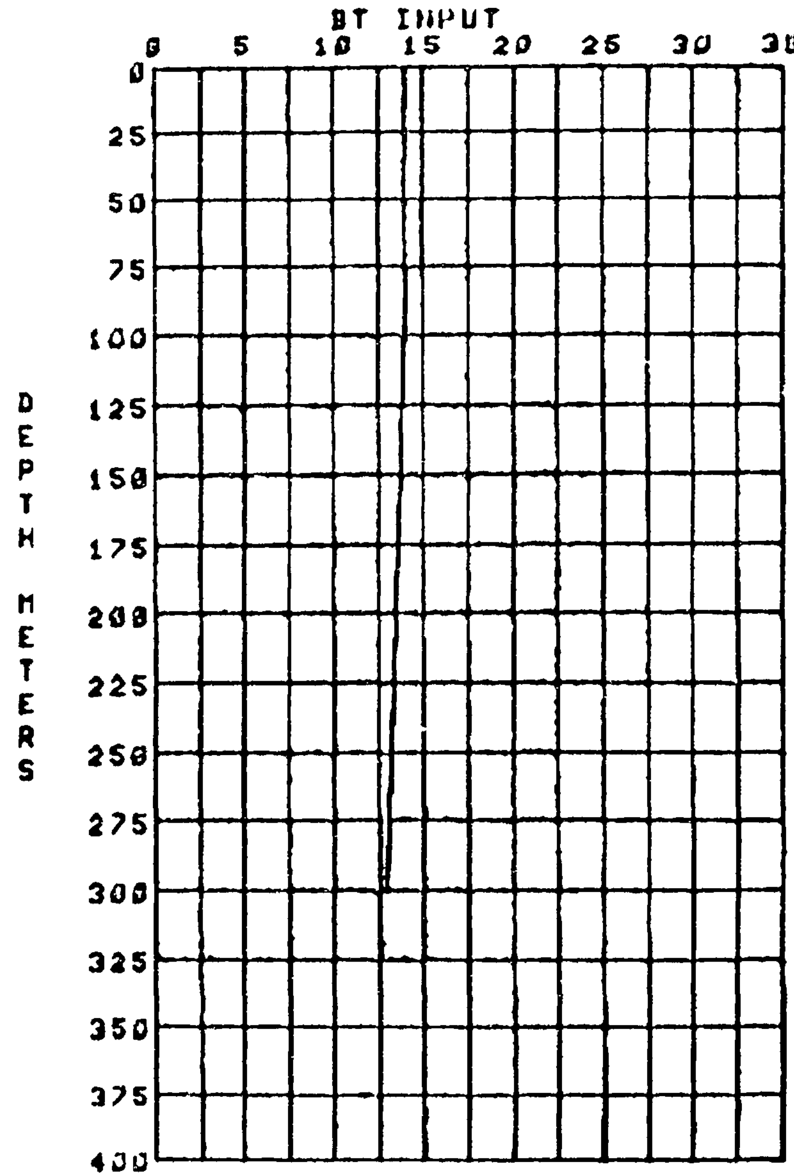
```
@@### TASSHAP INPUT PROGRAM #####

000# TARGET OPERATIONAL MODE ####

     1)TRANSIT
     2)AREA SEARCH-ASW
     3)AREA SEARCH-SURFACE SHIPS
     4)BARRIER
     5)CONVOY PENETRATION
     6)AMPHIBIOUS ATTACK
     7)HVU ATTACK
     8)SSBN OPERATIONS
     9)SURVEILLANCE-ASW
     10)SURVEILLANCE-SURFACE SHIPS
     11)SNORKEL
     12)INPUTSOURCE DEPTH
WHICH TARGET OPERATION MODE?---1
SOURCE DEPTH (FEET) = 100


000# OWN SHIP TYPE OF MISSION ####

     1)SURVEILLANCE
     2)ESCORT
     3)TRAIL
     4)AREA SANITIZATION
     5)AMPHIBIOUS ASSAULT PROTECTION
WHICH TYPE OF MISSION?---3
```

```
                 ***** TASSRAP INPUT PROGRAM *****

**** SONAR TYPE ****
     1)AN/SQR-15
     2)AN/BQR-15
     3)STASS
     4)TACTASS
     5)LAMBDA
WHAT TYPE OF SONAR?---4


INPUT BT?  1=YES 0=NO---1
THE FIRST DEPTH MUST BE 0, AND THE LAST INPUT
DEPTH MUST BE EQUAL TO OR GREATER THAN 300 METERS
FOR METRIC INPUT, OR EQUAL TO OR GREATER THAN 1,000
FEET FOR ENGLISH INPUT
NUMBER OF DATA POINTS IN PROFILE = 3
UNITS OF DATA, 1=METRIC, 2=ENGLISH---1
```

```
##### BATHYTHERMOGRAPH INPUT #####

INPUT PROFILE DATA IN DEPTH,TEMPERATURE PAIRS
 DATA POINT DEPTH,TEMPERATURE
       1       0,15

       2       100,15

       3       300,13

 DO YOU WISH TO ENTER A BOTTOM DEPTH
YES(1)-NO(0) ?1
 BOTTOM DEPTH UNITS MUST BE METERS
BOTTOM DEPTH = 5000
```

```
##### BATHYTHERMOGRAPH INPUT #####
GOODT DATA INPUT###

LINE     DEPTH     TEMP
  1         0.     14.0
  2       100.     14.0
  3       300.     13.0
CHANGE ANY OF THE   3
DEPTH-TEMPERATURE PAIRS?   1=YES 0=NO---0*
DO YOU WISH TO CHANGE BOTTOM DEPTH FROM
           5000.
YES=1,NO=0 ?0
```

*Any changes to be made follow the same procedure as changing frequency source level data.



```
***** TASSRAP INPUT PROGRAM *****

 DO YOU WISH TO INPUT MEASURED BEAM NOISE DATA ?
YES (1) -NO (0) ? 1
ENTER DATA WITH ONE DESIRED BEAM NUMBER
FOLLOWED BY THE DESIRED FREQUENCY LEVEL PAIRS WITH ALL
NUMBERS SEPARATED BY COMMAS.  AFTER DOING THIS
STRIKE THE RETURN KEY AND FOLLOW THE SAME
PROCEDURE FOR ANY ADDITIONAL FREQUENCIES.
*** NOTE MAXIMUM OF 5 FREQUENCIES AND 24 BEAMS
 ARE ALLOWED ***
 NUMBER OF FREQUENCIES ?2
 NUMBER OF BEAMS ?3
 BEAM NUMBER, FREQUENCY,LEVEL
3,1000,100,2000,200
5,1000,100,2000,200
7,1000,100,2000,200
```

```
##### TASSRAP INPUT PROGRAM #####

1 LABEL = TASSRAP
2 DAY =      1
3 MONTH =     8
4 YEAR =    77
5 TIME =    1000
6 LATITUDE =    3000
7 NORTH(1)-SOUTH(2) =     1
8 LONGITUDE =    6500
9 EAST(1)-WEST(2) =     2
10 MAXIMUM RANGE(NM) =  175.0
11 WAVE HEIGHT(FT) =     3.0
12 WIND SPEED(KTS) =   10.0
13 SHIP SPEED(KTS) =     5.0
14 TARGET TYPE =     1
15 TARGET OP.MODE =     1
16 TYPE OF MISSION =     1
17 SONAR TYPE =     4
18 CHANGE BY INPUT
19 CHANGE BEAM NOISE DATA
20 BOTTOM DEPTH = 13123.1
BOTTOM DEPTH IS IN FEET

CHANGE ANY DATA?  1=YES 0=NO---2

     ***** YOUR ENTRY IS INVALID *****
       *** HIT SPACE BAR TO CONTINUE ***
```

```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL = TASSRAP
 2 DAY =     1
 3 MONTH =    10                          3 MONTH = 8
 4 YEAR =    77
 5 TIME =    1000
 6 LATITUDE =    3000
 7 NORTH(1)-SOUTH(2) =     1
 8 LONGITUDE =    6500
 9 EAST(1)-WEST(2) =     2
 10 MAXIMUM RANGE(NM) =  175.0
 11 WAVE HEIGHT(FT) =    3.0
 12 WIND SPEED(KTS) =   10.0
 13 SHIP SPEED(KTS) =    5.0
 14 TARGET TYPE =     1
 15 TARGET OP.MODE =     1
 16 TYPE OF MISSION =     1
 17 SONAR TYPE =     4
 18 CHANGE BT INPUT
 19 CHANGE BEAM NOISE DATA
 20 BOTTOM DEPTH = 13123.1
 BOTTOM DEPTH .3 IN FEET

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---3
```

```
***** TASSRAP INPUT PROGRAM *****

1 LABEL = 1
2 DAY =     1
3 MONTH =     1
4 YEAR =     1
5 TIME =       1
6 LATITUDE =   3000
7 NORTH(1)-SOUTH(2) =     1
8 LONGITUDE =   6000
9 EAST(1)-WEST(2) =     2
10 MAXIMUM RANGE(NM) =     1.0
11 WAVE HEIGHT(FT) =     1.0
12 WIND SPEED(KTS) =     1.0
13 SHIP SPEED(KTS) =     1.0
14 TARGET TYPE =     1
15 TARGET OP.MODE =     1
16 TYPE OF MISSION =     3
17 SONAR TYPE =     4
18 CHANGE BT INPUT
19 CHANGE BEAM NOISE DATA
20 BOTTOM DEPTH = 18404.0
BOTTOM DEPTH IS IN FEET

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---15
```

```
***** TASSRAP INPUT PROGRAM *****

**** TARGET OPERATIONAL MODE ****

      1)TRANSIT
      2)AREA SEARCH-ASW
      3)AREA SEARCH-SURFACE SHIPS
      4)BARRIER
      5)CONVOY PENETRATION
      6)AMPHIBIOUS ATTACK
      7)HVU ATTACK
      8)SSBN OPERATIONS
      9)SURVEILLANCE-ASW
      10)SURVEILLANCE-SURFACE SHIPS
      11)SNORKEL
      12)INPUTSOURCE DEPTH
WHICH TARGET OPERATION MODE?---1
```

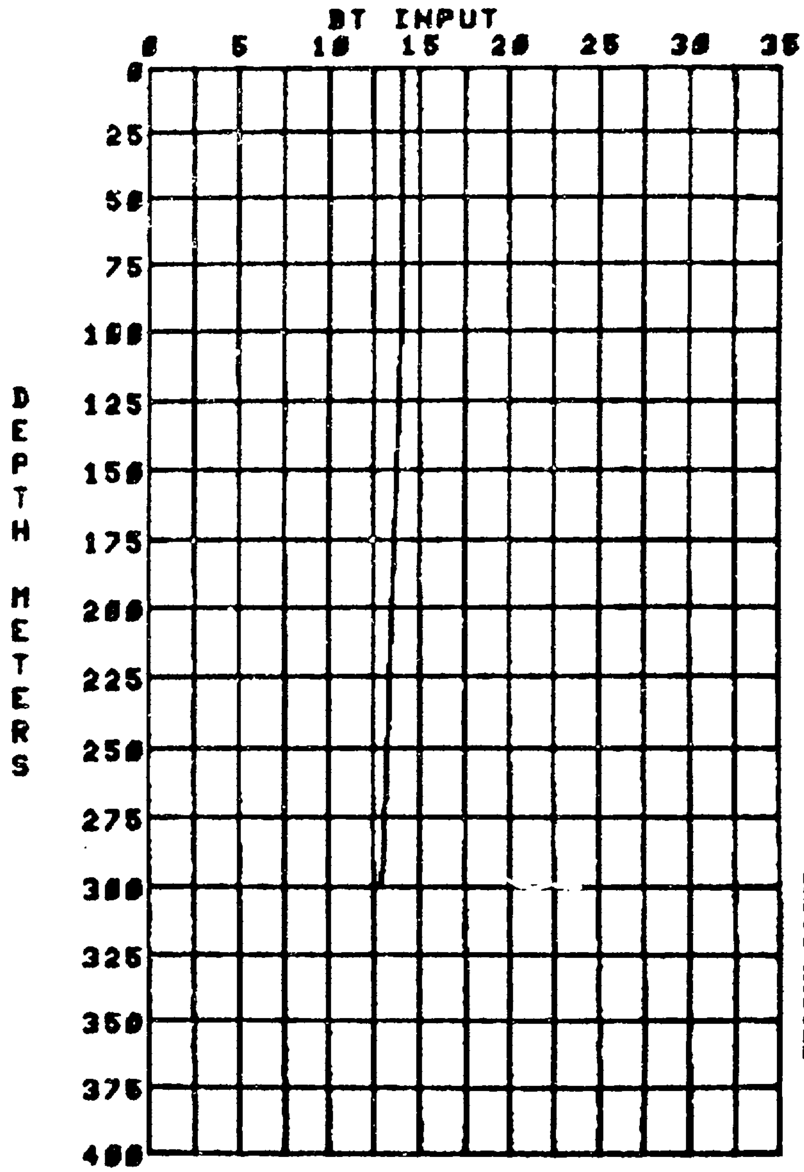
```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL =  1
 2 DAY =      1
 3 MONTH =      1
 4 YEAR =      1
 5 TIME =        1
 6 LATITUDE =    3000
 7 NORTH(1)-SOUTH(2) =       1
 8 LONGITUDE =    6000
 9 EAST(1)-WEST(2) =      2
 10 MAXIMUM RANGE(KM) =      1.0
 11 WAVE HEIGHT(FT) =      1.0
 12 WIND SPEED(KTS) =      1.0
 13 SHIP SPEED(KTS) =      1.0
 14 TARGET TYPE =      1
 15 TARGET OP.MODE =      1
 16 TYPE OF MISSION =      3
 17 SONAR TYPE =      4
 18 CHANGE BT INPUT
 19 CHANGE BEAM NOISE DATA
 20 BOTTOM DEPTH = 16404.0
 BOTTOM DEPTH IS IN FEET

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---16
```

```
                    ***** TASSRAP INPUT PROGRAM *****

**** OWN SHIP TYPE OF MISSION ****

     1)SURVEILLANCE
     2)ESCORT
     3)TRAIL
     4)AREA SANITIZATION
     5)AMPHIBIOUS ASSAULT PROTECTION
WHICH TYPE OF MISSION?---4
```

```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL = 1
 2 DAY =     1
 3 MONTH =     1
 4 YEAR =     1
 5 TIME =      1
 6 LATITUDE =   3000
 7 NORTH(1)-SOUTH(2) =     1
 8 LONGITUDE =   6000
 9 EAST(1)-WEST(2) =    2
 10 MAXIMUM RANGE(NM) =    1.0
 11 WAVE HEIGHT(FT) =    1.0
 12 WIND SPEED(KTS) =    1.0
 13 SHIP SPEED(KTS) =    1.0
 14 TARGET TYPE =    1
 15 TARGET OP.MODE =    1
 16 TYPE OF MISSION =    4
 17 SONAR TYPE =    4
 18 CHANGE BT INPUT
 19 CHANGE BEAM NOISE DATA
 20 BOTTOM DEPTH = 16404.0
 BOTTOM DEPTH IS IN FEET

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---10
```

BATHYTHERMOGRAPH INPUT

```
***BT DATA INPUT***

  LINE    DEPTH    TEMP
    1        0.    14.0
    2      100.    14.0
    3      300.    13.0
CHANGE ANY OF THE  3
DEPTH-TEMPERATURE PAIRS?  1=YES 0=NO---0
DO YOU WISH TO CHANGE BOTTOM DEPTH FROM
          5000.
YES=1,NO=0 ?9
```



```
***** TASSRAP INPUT PROGRAM *****

 1 LABEL = 1
 2 DAY =     1
 3 MONTH =     1
 4 YEAR =     1
 5 TIME =      1
 6 LATITUDE =    3000
 7 NORTH(1)-SOUTH(2) =     1
 8 LONGITUDE =    6000
 9 EAST(1)-WEST(2) =     2
 10 MAXIMUM RANGE(NM) =     1.0
 11 WAVE HEIGHT(FT) =     1.0
 12 WIND SPEED(KTS) =     1.0
 13 SHIP SPEED(KTS) =     1.0
 14 TARGET TYPE =     1
 15 TARGET OP.MODE =     1
 16 TYPE OF MISSION =     4
 17 SONAR TYPE =     4
 18 CHANGE BT INPUT
 19 CHANGE BEAM NOISE DATA
 20 BOTTOM DEPTH = 16404.0
 BOTTOM DEPTH IS IN FEET

CHANGE ANY DATA?  1=YES 0=NO---1
INPUT LINE NUMBER TO BE CHANGED ---19
```

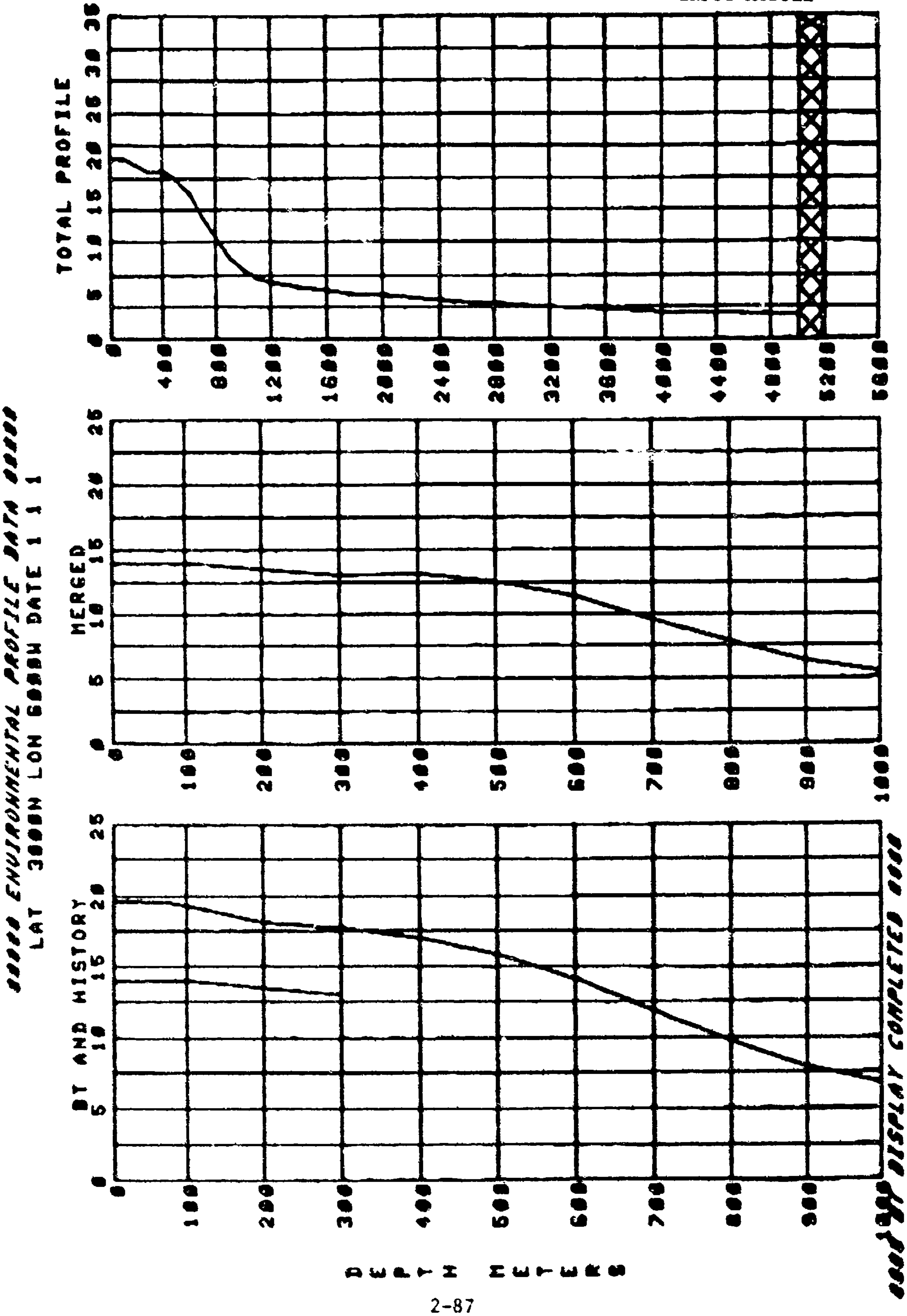
```
##### TASSRAP INPUT PROGRAM #####

BEAM NUMBER FREQUENCY LEVEL
     3          999.9      99.9
     3         1999.9     199.9
     5          999.9      99.9
     5         1999.9     199.9
     7          995.9      99.9
     7         1999.9     199.9
DO YOU WISH TO CHANGE ANY NOISE DATA YES=1 NO=0 1
 ON WHICH BEAM NUMBER IS THE ERROR 3
 INPUT BEAM NUMBER FOLLOWED BY ALL FREQUENCY
 LEVEL PAIRS FOR THAT BEAM
3,3000,100,3000,100
```

```
***** TASSRAP INPUT PROGRAM *****

 BEAM NUMBER FREQUENCY LEVEL
     3        2999.9     99.9
     3        2999.9     99.9
     6         599.9     99.9
     5        1999.9    199.9
     7         599.9     99.9
     7        1999.9    199.9
DO YOU WISH TO CHANGE ANY NOISE DAT  YES=1 NO=0 1
 ON WHICH BEAM NUMBER IS THE ERROR 9
 THERE IS NO BEAM NUMBER EQUAL TO YOUR INPUT
 DO YOU WISH TO INCLUDE THAT BEAM YES=1 NO=0 1
 INPUT BEAM NUMBER FOLLOWED BY ALL FREQUENCY
 LEVEL PAIRS FOR THAT BEAM
9.1000.100.2000.230
```

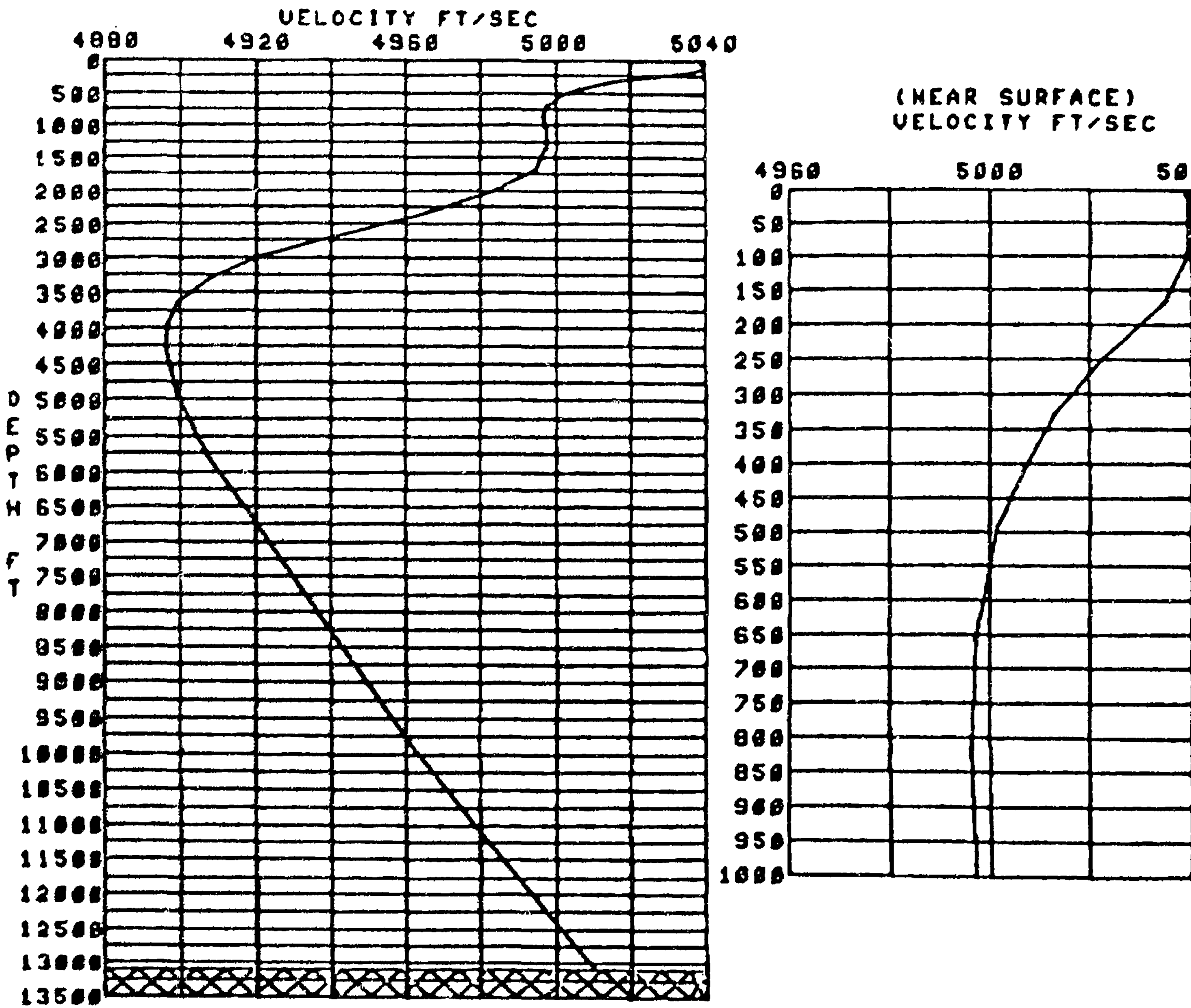
```
##### TASSRAP INPUT PROGRAM #####

BEAM NUMBER FREQUENCY LEVEL
     3       2999.9     99.9
     3       2999.9     99.9
     5        999.9     99.9
     5       1999.9    199.9
     7        999.9     99.9
     7       1999.9    199.9
     9        999.9     99.9
     9       1999.9    199.9
DO YOU WISH TO CHANGE ANY NOISE DATA YES=1 NO=0
```

```
                    ##### ENVIRONMENTAL PROFILE DATA #####
                      LAT 3000N LON12000W DATE  1  1  1
NO DATA FILES FOR LATITUDE AND LONGITUDE
 ENTERED
NEW LATITUDE VALUE =3000
NORTH = 1 OR SOUTH = 2    1
NEW LONGITUDE VALUE =6000
EAST =1 OR WEST = 2    2
```

```
                          ***** ENVIRONMENTAL PROFILE DATA *****
                             LAT 3000N LON 6000W DATE  1 1 1
     H. FREQ BLP  7        L. FREQ BLP  5      SHIP. DEN.  1.7287700E -3
   BT DATA                RETRIEVED DATA                           MERGED DATA
  DEP      TEMP             DEP      TEMP       SAL                DEP      TEMP       SAL        VEL
  (M)       (C)             (M)       (C)      (PPT)               (M)       (C)      (PPT)     (M/SEC)
    0.     14.00              0.     19.59     36.50                 0.     14.00     36.50     1506.04
  100.     14.00              9.     19.59     36.50               100.     14.00     36.59     1507.80
  300.     13.00             20.     19.59     36.50               300.     13.00     36.39     1507.54
                             30.     19.59     36.59        SLD>   400.     13.07     36.29     1509.31
                             50.     19.50     36.59               500.     12.52     36.09     1508.84
                             75.     19.50     36.59               600.     11.36     35.79     1506.11
                             99.     19.29     36.59               700.      9.51     35.50     1500.75
                            125.     19.00     36.59               800.      7.79     35.29     1495.67
                            149.     18.69     36.50               900.      6.20     35.19     1491.00
                            199.     18.09     36.50               999.      5.37     35.09     1489.16
                            250.     17.89     36.50        DSC>  1099.      4.78     35.09     1488.45
                            300.     17.69     36.39              1199.      4.47     35.09     1488.81
                            400.     17.00     36.29              1299.      4.22     35.09     1489.45
                            500.     15.79     36.09              1400.      4.05     35.09     1490.41
                            600.     14.09     35.79              1499.      3.86     35.00     1491.56
                            700.     11.79     35.50              1750.      3.54     35.00     1494.05
                            800.      9.69     35.29              1999.      3.42     35.00     1497.75
                            900.      7.79     35.19              2500.      2.98     35.00     1504.42
                            999.      6.69     35.09              3000.      2.63     34.89     1511.40
                           1099.      5.89     35.09              3999.      2.00     34.89     1526.49
                           1199.      5.39     35.09              5000.      1.91     34.89     1543.57
                           1299.      5.00     35.09              5000.      1.91     34.89     1543.57
                           1400.      4.69     35.09
                           1499.      4.49     35.00
                           1750.      3.99     35.00
                           1999.      3.79     35.00
                           2500.      3.29     35.00
                           3000.      2.89     34.89
                           3999.      2.29     34.89
                           5000.      2.09     34.89
***** PROFILE COMPLETE *****
```


***** ENVIRONMENTAL PROFILE DATA *****
LAT 3000N LON 6000W DATE 1 1 1
TOTAL PROFILE
MERGED
BT AND HISTORY
DEPTH METERS
**** DISPLAY COMPLETED ****

DATE 1/18/77
LAT 3300N
LON 6500W



0C330 TASSRAP SVP GRAPHIC

BTGRAPH - This subroutine is used whenever the operator elects to enter an in situ BT.

XNTF - This subprogram is only initiated if the operator elects to enter a BT.

GETTGT - This subroutine is not used when the operator chooses to enter target depth and source levels.

GETSONAR - UNCONDITIONAL

SLFRQ - This subroutine is not initiated if the operator enters source levels (i.e., TARGET TYPE = 8).

IOERR - This subroutine is initiated whenever an input/output error is detected.

GETENV - UNCONDITIONAL

XNTERP - UNCONDITIONAL

MERGE - This subroutine is only used if an in situ BT was entered.

WILSON - UNCONDITIONAL

TWDPT - UNCONDITIONAL

PFGRAPH - This subroutine is initiated when the operator elects to obtain a temperature profile graph.

TRWND - UNCONDITIONAL

MOVFR - UNCONDITIONAL

MOVBR - UNCONDITIONAL

TR720 - UNCONDITIONAL

2.6 MODULE LIMITATIONS

2.6.1 Input Module The following information pertains to limitations and units corresponding to the parameters which are entered by the TASSRAP II operator.

All "units of data" questions have the following codes for responses:

1 = metric

2 = English

All yes/no questions have the following codes for responses:

1 = yes

Ø = no

These responses are checked to ascertain that a 1 or 0 has been entered. If the check fails, an error message is printed with the operator required to reenter his response.

All inputs are listed below along with any limitations:

Label	- An alphanumeric label up to 20 characters.
Day	- An integer value corresponding to the day (1-31).
Month	- Numerical value of the month (1-12). This value is used to calculate the season in the retrieval of environmental data and is checked to determine its value between 1 and 12.
Year	- A two-digit integer corresponding to the year (e.g., 77).
Time	- Integer representation of the time group, 24-hour clock (e.g., 1500).
Latitude	- The latitude in degrees and minutes; a four-digit integer between 0000 and 9000 (e.g., 4400 = 44 degrees 00 minutes).
North(1)-South(2)	- Denotes north or south latitude: 1 = north, 2 = south.
Longitude	- The longitude in degrees and minutes, a five-digit integer between 00000 and 18000 (e.g., 15930 = 159 degrees 30 minutes).
East(1)-West(2)	- Denotes east or west longitude: 1 = east, 2 = west.
Maximum Range	- The maximum range to be used in calculating acoustic performance predictions. The units of this input are nautical miles (e.g., 100.0).

Wave Height	-	The wave height in feet (e.g., 4.5).
Wind Speed	-	The wind speed in knots (e.g., 15.3).
Ship Speed	-	Own-ship speed in knots (e.g., 4.8).
Target Type	-	A numerical index (1-8) corresponding to the target type of interest (e.g., target type 1 = Soviet nuclear Type 1). If the operator enters an invalid index, the program will display an error message and allow the operator to reenter his selection.
Target Operational Mode	-	A numerical index (1-13) corresponding to the target operational mode of interest (e.g., target operational mode 1 = transit). If an invalid entry is made, the program displays an error message and allows the operator to reenter his selection.
Type of Mission	-	A numerical index (1-5) corresponding to own-ship type of mission (e.g., type of mission 1 = surveillance). If an invalid entry is made, the program displays an error message and allows the operator to reenter the mission type.
Sonar Type	-	A numerical index (1-5) corresponding to own-ship sonar type (e.g., sonar type 1 = AN/SQR-15). If the operator enters an invalid index, the program will display an error message and allow the operator to reenter the sonar type.
BT Input	-	If the operator chooses to input his own BT, the following restrictions apply to the data: (1) The first depth entered must be zero. (2) The last depth entered must be 300 meters or greater, or 1000 feet or greater. The operator can enter data in either metric or English units. Data should be accurate to the nearest foot or meter and to a tenth of a degree.

Bottom Depth - Optional input for ocean depth. If the operator entered a BT, ocean depth is entered in the same units as the BT. If no BT is entered, the units are meters.

Input Own Source Levels - If the operator chooses to enter his own source levels, he responds with a target type index of 8. The operator enters the data in frequency-source level pairs. Data should be accurate to the nearest tenth of a decibel. The operator may enter a maximum of five frequencies.

Input Measured Noise - The operator may enter measured beam noise for up to five frequencies and up to a maximum of 24 beams. Data should be accurate to a tenth of a hertz and to the nearest tenth of a decibel.

Input latitude and longitude are compared with the data file limits. If no data files exist for these inputs, the operator is required to enter new values.

2.6.2 Subroutine GETENV There are numerous checks performed throughout this subroutine. When the first block of a data file is read, the first number is checked to determine if a valid file is being read. Longitude must have values between -360 and 360; latitude is also examined to have values between -90 and 90, the only ones accepted. If one of the above conditions is not satisfied, control is returned to INPUT:OV. As the data from the second section are read, the first two numbers are checked against predetermined values. For unequal conditions, an error status message is printed and control returned to INPUT:OV. The same applies for data read from the third section, except that the first three numbers are checked.

2.6.3 Subroutine MERGE In this subroutine, there are three conditions which must be met. First, the initial depth of the input BT equals zero. Second, the synoptic BT is required to be greater than or equal to 300 meters. Finally, the BT cannot exceed the retrieved data.

2.6.4 Subroutine GETTGT Several tests are made in this subroutine of the data retrieved. The first piece of data read is checked to determine if it is equal to the type of target selected. Should the condition not be fulfilled, an error message is printed out stating an invalid block number was read. Next, the data are checked to be either an "N" or "D" with the message "data file failure" printed if there is no verification. Target operational mode is tested to be between 1 and 11, while the third data should equal the target operational mode entered by the operator. For neither condition being true, "Invalid tgt op. mode" is an output. Finally, the number of frequencies is checked to ascertain if a data file exists.

2.6.5 Subroutine IOERR   This subroutine is employed whenever there is an error in executing the commands FGTS, FOPFL, and FCLFL and is described in section 3.1.

CHAPTER 3
Data Base Design

3.1 INTRODUCTION This chapter discusses the tables, variables, indexes, flags, and constants employed in the input module. These items have been assigned a mnenomic which for the most part follows a labeling convention of abbreviating the original name. There are some items, however, named during program editing with the only criteria placed on the name being that of singularity.

3.1.1 Purpose It is the purpose of this chapter to provide a detailed description of the variables, indexes, flags, and constants employed in the input module.

3.1.2 Scope The descriptions in this chapter in conjunction with the other chapters in this document are designed to enable a program analyst to fully understand the input module.

3.2 TABLES This section contains a detailed description of each table used in the input module.

3.2.1 Table Name

Depth	(DEP)
Frequency	(FREQ)
Frequency of beam noise	(FREQN)
Beam	(IBEAM)
Block of data	(IBLOCK)
Buffer	(IBUF)
Intermediate frequency file	(IFRQ)
Working storage for frequencies	(IWSFRQ)
Label	(LABEL)
Level of beam noise data	(LEVELN)
Salinity	(S)
Merged salinity	(SM)

Temperature	(T)
Temperature input	(TEMP)
Historical temperature	(THIS)
Merged temperature	(TM)
Input temperature in degrees centigrade	(TOB)
Velocity	(VM)
Depth	(Z)
Historical depths	(ZHIS)
Merged depth	(ZM)
Depth	(ZO)

3.2.2 Purpose And Type

DEP - Depths of the in situ BT; operator input in meters or feet; variable length.

FREQ - Frequencies and SPLs on which to optimize detection performance selected from target file based upon target type; variable length.

FREQN - Frequencies for beam noise data entered by the operator; variable length.

IBEAM - Beam numbers for beam noise entered by operator; variable length.

IBLOCK - An input buffer used by subroutine GETTGT when retrieving target data; fixed length.

IBUF - An input buffer used by subroutine GETENV when retrieving environmental data; fixed length.

IFRQ	- An intermediate frequency file containing frequency, SPL, and reliability information; fixed length.
IWSFRQ	- A working storage area used in subroutine SLFRQ; fixed length.
LABEL	- Alphanumeric label up to 20 characters including spaces entered by operator; variable length.
LEVELN	- Level of beam noise for each frequency entered by operator; variable length.
S	- Historical salinity in parts per thousand for the various depths; selected from environmental data file; variable length.
SM	- Array of salinity in parts per thousand versus depth; obtained from historical data and interpolated for BT input depths; variable length.
T	- Historical temperatures in degrees centigrade for the various depths; selected from environmental file; variable length.
TEMP	- Array of input temperature versus depth; input by operator in degrees centigrade or degrees Fahrenheit; variable length.
THIS	- Historical temperature in degrees centigrade for various depths; variable length.
TM	- Array of merged temperature versus depth; obtained from historical data and input BT; variable length.
TOB	- Input temperature in degrees centigrade versus depth; variable length.
VM	- Velocity of sound versus depth calculated by Wilson's equations; variable length.

Z - Depth of historical temperature and salinity; selected from environmental data file; variable length.

ZHIS - Depth of historical temperature array; variable length.

ZM - Depths of merged temperature and salinity; obtained from historical data and input BT; variable length.

ZO - Depths of the in situ BT in meters; obtained from the input BT depth; variable length.

3.2.3 Size And Indexing Procedure Listed below are the tables and arrays with the size and type of each array denoted in parenthesis (e.g., T (50) -50 element, single-dimension array). Arrays with mnemonic names beginning with the letter I, J, K, L, M, or N, with the exception of LEVELN, contain integer precision data (one 16-bit word). All other arrays contain standard precision floating point data (two 16-bit words).

DEP (31)

FREQ (2, 5) Row 1 contains frequencies and row 2 contains SPLs.

FREQN (24, 6) Column 1 contains beam numbers and columns 2 through 6 contain frequencies.

IBEAM (24)

IBLOCK (9, 11)

IBUF (145)

IFRQ (4, 11) Row 1 is frequency, row 2 is SPL, row 3 is reliability, and row 4 is standard deviation or RPM.

IWSFRQ (3, 11) Row 1 is frequency, row 2 is SPL, and row 3 is reliability.

LABEL (10)

LEVELN (24, 6) Column 1 contains beam numbers, and columns 2 through 6 contain levels.

S (50)

SM (50)

T (50)

TEMP (3)

THIS (50)

TOB (31)

VM (50)

Z (50)

ZHIS (50)

ZM (50)

ZO (31)

3.3 VARIABLES This section contains a detailed description of each variable included in common or file.

3.3.1 Variable Name

Bottom depth	(BOTZ)
Bottom depth	(BOTZ1)
Maximum array depth	(DMAX)
Maximum depth	(DMAX1)
Deep sound channel	(DSC)
File	(IBOT)
Change beam	(ICHB)
Day	(IDA)
Date	(IDATE)
Delta source level	(IDELTALS)
Delta reliability	(IDELTARL)

End	(IEND)
Ocean area	(IHCW)
High frequency bottom loss province	(IHFBLP)
Low frequency bottom loss province	(ILFBLP)
Maximum	(IMAX)
Minimum	(IMIN)
Month	(IMO)
Move number	(IMOV)
Move number	(IMOVE)
Number of frequencies	(INUMFRQ)
Number of operational modes	(INUMOP)
Reference number	(IREF)
Record number	(IRNO)
Season	(ISEA)
File slot	(ISLOT)
Temporary	(ITEMP)
Time	(ITIME)
Target type	(ITYPE)
Year	(IYR)
North-South	(JLIN)
East-West	(JLIN1)
Maximum number	(JMAX)
Season	(JSEA)

Latitude	(LAT)
Low frequency limit	(LFRQLM)
Longitude	(LON)
Channel number	(LUN)
Target file channel number	(LUNTG)
Number of beams	(NB)
Number of data blocks	(NDBLK)
Number of data points	(NDP)
Number of frequencies	(NF)
Number of frequencies	(NF1)
Number of data points	(NNDP)
Number of points	(NPOINT)
Number of horizontal increments	(NX)
Number of vertical increments	(NY)
Number of depth points	(NZP)
Predicted frequency	(PRDFRQ)
Maximum range	(RANGE)
Salinity minus 35	(S35)
Surface layer depth	(SLD)
Shipping density	(SHPDEN)
Ship's speed	(SS)
Dummy variable	(TA)
Temperature difference	(TDEL)
Target broadband noise	(TGTBBN)

Target depth	(TGTDEP)
Target speed	(TGTSPD)
Highest temperature	(THI)
Lowest temperature	(TLO)
Maximum temperature	(TMAX)
Minimum temperature	(TMIN)
Upper frequency limit	(UFRQLM)
Velocity at DSC	(VELDSC)
Velocity at SLD	(VELSLD)
Wave height	(WH)
Wind speed	(WS)
Depth	(XDEP)
Maximum horizontal value	(XHI)
Latitude	(XLAT)
Minimum latitude	(XLATMN)
Maximum latitude	(XLATMX)
Minimum horizontal value	(XLO)
Longitude	(XLON)
Minimum longitude	(XLONMN)
Maximum longitude	(XLONMX)
Maximum horizontal value	(XMAX)
Horizontal increments	(XMDUL)
Minimum velocity	(XMIN)
Minimum depth	(XMINDP)

Temperature	(XTEMP)
Maximum vertical value	(YHI)
Minimum vertical value	(YLO)
Bottom depth	(ZBOT)
Dummy depth	(ZF)

3.3.2 Purpose And Type

BOTZ	- Depth of ocean in meters or feet. This variable may be entered by operator or retrieved from data file; floating point real.
BOTZ1	- Depth of ocean in feet, converted if necessary from BOTZ; floating point real.
DMAX	- Maximum array depth in meters, retrieved from sonar file; floating point real.
DMAX1	- Maximum depth for near-surface portion of SVP graph, set in BTGRAPH; floating point real.
DSC	- Depth of deep sound channel, calculated in TWDPT: floating point real.
IBOT	- Used as a transfer variable with GETENV to denote which data file is to be addressed; numeric integer.
ICHB	- Beam number to be changed, input by operator; numeric integer.
IDA	- Numerical value of the day-input by operator; fixed length.
IDATE	- Date group (day, month, year), assigned values of IDA, IMO, and IYR; fixed length.

IDELTALS - A weighted difference between source levels of two frequencies. Assigned values in subroutine SLFRQ; numeric integer.

IDELTARL - A weighted difference between reliabilities of two frequencies. Assigned values in subroutine SLFRQ; numeric integer.

IEND - A variable which indicates the maximum value of a "do loop" used in SLFRQ; numeric integer data.

IHCW - Ocean area designator, read from data file; integer data.

IHFBLP - High frequency bottom loss province; calculated in INPUT:OV from value obtained in bottom loss data file; integer data.

ILFBLP - Low frequency bottom loss province, calculated in INPUT:OV from value obtained in bottom loss data file; integer data.

IMAX - Used in GETENV, equals maximum number of degrees of longitude in data file; integer data. Used by SLFRQ to denote the maximum value for a "do loop."

IMIN - Used by SLFRQ to denote the starting value for a "do loop;" integer precision data.

IMO - Numerical value of the month entered by the operator; integer data.

IMOV - Number of data blocks to skip to retrieve proper STD file; calculated in GETENV; integer data.

IMOVE - Equals IMOV, used in MOVBR or MOVFR; integer data.

INUMFRQ - Number of target frequencies read in target data file; integer data.

INUMOP	- Number of operational modes available for target type selected; integer data.
IREF	- Reference number in data file for latitude and longitude input; integer data.
IRNO	- Used by GETENV as a record number indicator; integer data.
ISEA	- Season read from data file; integer data.
ISLOT	- File slot number on which an error has occurred; integer data.
ITEMP	- A temporary storage location used by SLFRQ subroutine; numeric integer data.
ITIME	- Time group, 24-hour clock input by operator; integer data.
ITYPE	- Target type read from target data file; integer data.
IYR	- Numerical representation of year; input by operator; integer data.
JLIN	- Numerical value of north-south indicator; converted from alphanumeric value in INPUT:OV; integer data.
JLIN1	- Numerical value of east-west indicator, converted from alphanumeric value in INPUT:OV; integer data.
JMAX	- Maximum number of degrees of latitude in data files, read from data file; integer data.
JSEA	- Numerical value of season calculated from input month in INPUT:OV; integer data.

LAT	- Latitude, four digits (0000-9000) with the last two being minutes; entered by operator; integer data.
LFRQLM	- A floating point variable which contains the lower frequency limit of the sonar; read from sonar file; floating point real.
LON	- Longitude, up to five digits (00000-18000) with the last two being minutes; input by operator; integer data.
LUN	- Transfer variable for various subroutines, always equal to a channel number; integer data.
LUNTG	- Channel on which file TGTFL is opened; integer data.
NB	- Number of beams for which measured noise is to be input; entered by operator; integer data.
NDBLK	- Number of data blocks in data file; read from data file; integer data.
NDP	- Number of data points in input BT: entered by operator; integer data.
NF	- Number of input target frequencies; entered by operator; integer data.
NF1	- Number of input beam noise frequencies; entered by operator; integer data.
NNDP	- NDP + 1; used in INPUT:OV; integer data.
NPOINT	- Number of points in historical array covered by input data; used in MERGE; integer data.
NX	- Number of increments in X direction; used in plotting routines; integer data.

NY	-	Number of increments in Y direction; used in plotting routines; integer data.
NZP	-	NDP + 1; used as page counter; integer data.
PRDFRQ	-	Predicted broadband frequency; read from target file; floating point real.
RANGE	-	Maximum range in nautical miles; entered by operator; floating point real.
S35	-	Salinity - 35; used in WILSON; floating point real.
SLD	-	Surface layer depth in meters; calculated from sound velocity profile in TWDPT; floating point real.
SHPDEN	-	Shipping density calculcated in GETENV; floating point real.
SS	-	Own-ship speed in knots; input by operator; floating point real.
TA	-	Interpolated value of a variable; used in XNTF; floating point real.
TDEL	-	Temperature difference calculated in MERGE; floating point real.
TGTBBN	-	Target broadband noise; retrieved from target data file; floating point real.
TGTDEP	-	Target depth in feet; retrieved from target data file or entered by operator; floating point real.
TGTSPD	-	Target speed in knots; retrieved from target file; floating point real.
THI	-	Highest temperature to be plotted; used in PFGRAPH; floating point real.

TLO	-	Lowest temperature to be plotted; used in PFGRAPH; floating point real.
TMAX	-	Maximum temperature value to be labeled; used in PFGRAPH; floating point real.
TMIN	-	Minimum temperature value to be labeled; used in PFGRAPH; floating point real.
UFRQLM	-	Upper frequency limit of a sonar; read from sonar file; floating point real.
VELDSC	-	Velocity at deep sound channel; calculated in TWDPT; floating point real.
VELSLD	-	Velocity at surface layer depth; calculated in TWDPT; floating point real.
WH	-	Wave height in feet; entered by operator; floating point real.
WS	-	Wind speed in knots; entered by operator; floating point real.
XDEP	-	Depth of input BT modified from previous input; entered by operator; floating point real.
XHI	-	Maximum value in X direction to be labeled; used in plotting routines; floating point real.
XLAT	-	Latitude converted into a real number in INPUT:OV; floating point real.
XLATMN	-	Minimum latitude covered by a data file; read from data file; floating point real.
XLATMX	-	Maximum latitude covered by a data file; read from data file; floating point real.

XLO - Minimum value in X direction to be labeled; used in plotting routines; floating point real.

XLON - Longitude converted into a real number in INPUT:OV; floating point real.

XLONMN - Minimum longitude covered by a data file; read from data file; floating point real.

XLONMX - Maximum longitude covered by a data file; read from data file; floating point real.

XMAX - Maximum velocity; used in INPUT:OV for SVP plot; floating point real.

XMDUL - Number of X increments to be plotted; used in plotting routines; floating point real.

XMIN - Minimum velocity; used in INPUT:OV for SVP plot; floating point real.

XMINDP - Minimum depth for which a BT may be input; floating point real.

XTEMP - Temperature value of input BT modified from previous inputs; operator input; floating point real.

YHI - Highest value of Y to be labeled on grid; used in plotting routines; floating point real.

YLO - Lowest value of Y to be labeled on grid; used in plotting routines; floating point real.

ZBOT - Bottom depth, variable used in MERGE; floating point real.

ZF - Depth value to be interpolated; used in XNTF; floating point real.

3.3.3 Size Variable names beginning with I, J, K, L, M, or N are integer precision variables (one 16-bit word); all other contain single-precision, floating point data (two 16-bit words). The variable LFRQLM also contains two-word floating point data. Only the variable IDATE contains more than one word. This variable is comprised of three words with

IDATE(1) = IDA

IDATE(2) = IMO

IDATE(3) = IYR

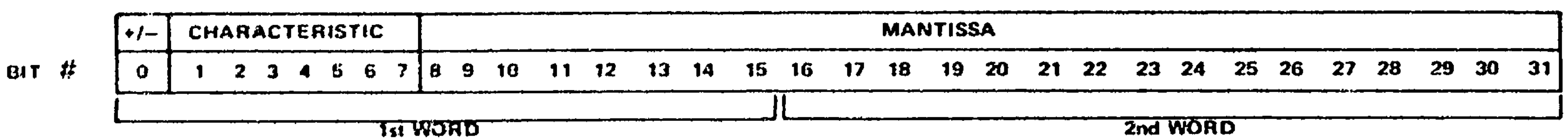
Figure 3-1 illustrates integer and floating point data construction.

3.3.4 Binary Point All integer precision data have a one-to-one correspondence with a binary number.

3.3.5 Range Of Values And Initial Condition Conceivably, all variables with the exception of IMO, JSEA, and XMINDP may assume values from -32,767 to +32,767. Only IMO is checked to determine that its value is between 1 and 12. JSEA is calculated from IMO; therefore, its value is between 1 and 4. XMINDP is set equal to 299 or 999. Reasonableness should prevail for the remaining variables.

Initial conditions for the variables are listed below:

BOTZ	0
BOTZ1	Not initialized
DMAX	0
DMAX1	Not initialized
DSC	0
IBOT	Not initialized
ICHB	Not initialized
IDA	Not initialized
IDATE	0
IDELTALS	Not initialized
IDELTARL	Not initialized

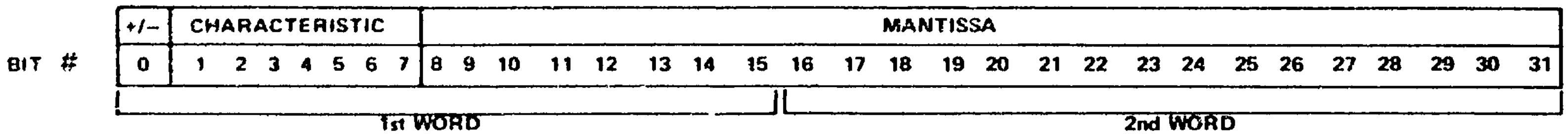


FLOATING POINT DATA

NUMERIC INTEGER DATA

NOTE: NEGATIVE INTEGERS ARE REPRESENTED USING 2's COMPLEMENT.

Figure 3-1. Integer and Floating Point Data Construction



FLOATING POINT DATA

NUMERIC INTEGER DATA

NOTE: NEGATIVE INTEGERS ARE REPRESENTED USING 2's COMPLEMENT.

Figure 3-1. Integer and Floating Point Data Construction

IEND	Not initialized
IHCW	Not initialized
IHFBLP	Not initialized
LFBLP	Not initialized
IMAX	Not initialized
IMIN	Not initialized
IMO	0
IMOV	Not initialized
IMOVE	Not initialized
INUMFRQ	0
INUMOP	Not initialized
IREF	Not initialized
IRNO	Not initialized
ISEA	Not initialized
ISLOT	Not initialized
ITEMP	Not initialized
ITIME	0
ITYPE	0
IYR	0
JLIN	Not initialized
JLIN1	Not initialized
JMAX	Not initialized
JSEA	Not initialized
LAT	0

LFRQLM	Not initialized
LON	0
LUN	Not initialized
LUNTG	Not initialized
NB	0
NDBLK	Not initialized
NDP	0
NF	Not initialized
NNDP	Not initialized
NPOINT	Not initialized
NX	Not initialized
NY	Not initialized
NZP	Not initialized
PRDFRQ	0
RANGE	0
S35	Not initialized
SLD	0
SHPDEN	0
SS	0
TA	Not initialized
TDEL	Not initialized
TGTBBN	0
TGTDEP	0
TGTSPD	0

THI	Not initialized
TLO	Not initialized
TMAX	Not initialized
TMIN	Not initialized
UFRQLM	Not initialized
VELDSC	Not initialized
VELSLD	Not initialized
WH	0
WS	0
XDEP	Not initialized
XHI	Not initialized
XLAT	0
XLATMN	Not initialized
XLATMX	Not initialized
XLO	Not initialized
XLON	0
XLONMN	Not initialized
XLONMX	Not initialized
XMAX	Not initialized
XMDUL	Not initialized
XMIN	Not initialized
XMINDP	Not initialized
XTEMP	Not initialized
YHI	Not initialized

YLO	Not initialized
ZBOT	Not initialized
ZF	Not initialized

3.3.6 Static/Dynamic Dynamic variables are those over the operator has direct control of its value. This may occur due to required input or optional input. The operator has no control over the value of static variables. Dynamic variables have been denoted in subsection 3.3.2 by the statement that the operator may enter the variable.

3.3.7 Structure And Bit Layout All floating point variables are made up of two 16-bit words, and numeric integer variables are made up of one 16-bit word, Figure 3-1 shows the structure and bit configuration of these two types of variables.

3.4 CONSTANTS This section contains a detailed description of each constant found in common or file.

3.4.1 Constant Name

Fahrenheit to centigrade	FRCENT
Feet to meters	FTMT
Channel number	LUNAT
Channel number	LUNIM
Channel number	LUNOS
Channel number	LUNP
Channel number	LUNSN

3.4.2 Purpose

FRCENT	- Employed in conversion of degrees Fahrenheit to degrees centigrade; equals 0.5555556; floating point real.
FTMT	- Multiplying this by depth in feet converts that depth to meters; equals 0.3048; floating point real.
LUNAT	- File slot on which STD file is to be opened, equals 0; integer data.

LUNIM - File slot on which scratch file Z9997AS:IM is to be opened; value equals 1; integer data.

LUNOS - File slot allocated for BLP file; equals 2; integer data.

LUNP - Channel number for CRT; equals 11; integer data.

LUNSN - File slot for shipping noise file; equals 3; integer data.

3.4.3 Initial Condition None of the constants employed in the input module are initialized.

3.4.4 Structure And Bit Layout All integers are single-precision 16-bit words with a binary equivalent. Floating point numbers are two 16-bit words with bit 0 being the sign, bits 1 through 7 the characteristic, and the remainder being a proper fraction.

3.5 FLAGS This section contains a detailed description of each flag used in the input module.

3.5.1 Flag Name

Change bottom	(ICBOT)
Change input	(IC1)
Change value	(IC3)
Change data	(ICHNG)
Input depth	(ICQ1)
Input depth	(ICQ2)
Diesel or nuclear	(IDN)
Error	(IE)
New environment	(IENV)
East-West	(IEW)
Type of shipping file	(ILIN)

No depths	(IND)
Noise change	(INOIS)
North-South	(INS)
Include number	(INUB)
Page full	(IPAGE)
Prefer a BT	(IPRF)
BT input	(IPROF)
Plot	(IQ)
Plot units	(IO1)
Plot	(IQ6)
Status	(ISTAT)
Change mode	(KCHNG)
Bottom depth	(LINDA)
Mike Fleck	(MF)
Metric or English	(MOE)
English or metric	(Q)

3.5.2 Flag Definition

ICBOT - Denotes whether or not the operator desired to change the input bottom depth: 1 = yes, Ø = no.

IC1 - A variable which indicates whether or not the operator desires to change any of the initial inputs: 1 = yes, Ø = no.

IC3 - Signifies if the default value for shipping density is to be changed: 1 = yes, Ø = no.

ICHNG - Indicator which denotes whether the operator desires to change any BT data or not: 1 = yes, Ø = no.

ICQ1 - Indicates bottom depth input following BT input: 1 = yes, Ø = no.

ICQ2 - Connotes if the operator wishes to input a bottom depth when no BT has been entered: 1 = yes, Ø = no.

IDN - Integer which indicates whether the target data retrieved from TGTFL is for a diesel or a nuclear submarine: D = diesel, N = nuclear.

IE - An error flag which is used whenever calling system subroutines. If IE = 0 then the call was successfully completed.

IENV - Flag relaying information between input module and TRANSMISSION LOSS DRIVER: Ø = no action required by either module, 1 = changes have been entered in the input module requiring a new propagation loss calculation, 2 = a new BT input is required in order to successfully complete propagation loss calculation.

IEW - Integer to denote east (1) or west (2) longitude; entered by operator.

ILIN - Denotes the type of shipping file to be assessed. 3 = file based on 5-degree quadrangle (i.e., Atlantic, Pacific, and Indian Oceans), 4 = file based on 1-degree quadrangle (i.e., Mediterranean Sea).

IND - An indicator which is used by subroutine TWDPT. If IND = 0 then the subroutine calculates the surface layer depth and the depth of the deep sound channel. If IND = 1 then the subroutine also calculates four tow depths.

INOIS	-	Signifies if the operator desires to change any input beam noise data: 1 = yes, Ø = no.
INS	-	Integer to denote north (1) and south (2) latitude; entered by operator.
INUB	-	During the process of changing input beam noise, if the operator wishes to change data for a beam number not previously entered, INUB signifies whether or not this new beam number is to be included in the data base: 1 = yes, Ø = no.
IPAGE	-	An indicator used to denote when the CRT screen is full. If IPAGE is greater than or equal to 30, then the screen is full.
IPRF	-	Enter a BT or not: 1 = yes, Ø = no.
IPROF	-	Denotes whether or not a BT was entered: 1 = input, 2 = no input.
IQ	-	Plot sound velocity profile: 1 = yes, Ø = no.
IQ1	-	Plot SVP in metric or English units: 1 = metric, 2 = English.
IQ6	-	Output temperature profile: 1 = yes, Ø = no.
ISTAT	-	A status indicator used by subroutines MERGE, GETENV, and GETTGT. If ISTAT = 1, the call was successfully completed.
KCHNG	-	Signifies if the first segment of INPUT is being initially accuated or if in cycle, allowing operator to change initial inputs: Ø = initial run, 1 = change mode.
LINDA	-	Used in GETENV to denote if a bottom depth were entered: Ø = bottom depth was entered, 1 = no bottom depth entered.

MF	-	Indicates initial run of INPUT or recalled by executive routine: Ø = initial, 1 = recalled.
MOE	-	An indicator which denotes whether the BT data was entered in metric or English units: 1 = metric, 2 = English.
Q	-	A floating point variable enabling an SVP plotting routine to be used for either metric or English units: equals 2 for metric and 5 for English units.

3.5.3 Initial Condition None of the flags are initialized.

3.5.4 Structure And Bit Layout All flags are integer precision data with the exception of flag Q which contains standard floating point data (two 16-bit words).

3.6 INDEX This section contains a detailed description of each index employed by the input module.

3.6.1 Index Name

Bottom	(IB)
Change	(IC)
Place to go	(IC2)
Place to go	(ICQ)
Sort	(IH)
Sort	(IH1)
Sort	(IL)
Sort	(IL1)
Number of change	(INCHNG)
Sonar	(ISONAR)
Own ship	(IST)

Target	(ITGT)
Target operational mode	(ITOM)
Array	(JI)
Number of points	(NOPTM)
Number of points	(NOPTS)
Number of data points	(NDP)

3.6.2 Index Definition

IB - Integer representation of the bottom loss class obtained from environmental file. Bits 8-11 of this variable contain the value of the low frequency bottom loss class, and bits 12-15 contain the value of the high frequency bottom loss class.

IC - An index corresponding to an input parameter the operator desires to change.

IC2 - Used for displayed placement on CRT: equals IC + 9.

ICQ - Employed in computed go to statement: equals IC - 13.

IH - Used in sort routine.

IH1 - Used in sort routine.

IL - Used in sort routine.

IL1 - Used in sort routine.

INCHNG - Number of data points to be changed in input BT.

ISONAR - Integer representation of type of sonar system; entered by operator.

IST - Numerical value representing own-ship type of mission.

ITGT	-	Integer representation of the target type; entered by operator.
ITOM	-	Integer representation of the target operational mode; entered by operator.
JI	-	An array of indexes used by the BT data input routine.
NOPTM	-	Number of data points in the merged data file; obtained from data file and BT input.
NOPTS	-	Number of data points in retrieved data file; obtained from data file.
NDP	-	Number of points in an input BT; entered by operator.

3.7 SUBPROGRAM REFERENCE (SET/USED) This section presents all the items discussed in the preceding sections and subsections in tabular form, cross-referenced with the major referencing routines. The letters S, U, and B are employed to indicate values set, used, or both (set and used), respectively. Items on the tabular listing that are followed by an asterisk are as follows:

DMAX	Set in GETSONAR
IMOVE	Used in MOVBR, MOVFR
IRNO	Used in TR720
ISLOT	Used in IOERR
LUN	Used in TRWND, MOVFR, MOVBR, TR720
NX	Used in plotting routines
NY	Used in plotting routines
XHI	Used in plotting routines
XLO	Used in plotting routines
XMDUL	Used in plotting routines
YHI	Used in plotting routines

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
TABLES												
DEP	S	U	U									
FREQ								S				
FREQN	S											
IBEAM	S											
IBLOCK					B							
IBUF				B								
IFRQ					S			U				
IWSFRQ								B				
LABLE	S	U										
LEVELN	S											
S		U		S		U					U	U
SM		S				S				U		
T		U		S		U	U				U	U
TEMP	S		U									
ZHIS							U					

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
TM		U				S				U		U
TOB		S				U	U					
VM		U							U	S		
Z		U		S		U					U	U
ZHIS							U					
ZM		U				S			U	U		
ZO		S				U						
VARIABLES												
BOTZ	B	U		S		U	U					
BOTZ1	B											
DMAX*												
DMAX1			B									
DSC		U							S			
IBOT				U								
ICHB	B											
IDA	S											
IDATE	S	U					U					

DATA ITEM	ROUTINES INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
IDELTALS								B				
IDELTARL								B				
IEND								B				
IHCW				B								
IHFBLP		B										
ILFBLP		B										
IMAX				B				B				
IMIN								B				
IMG	S	U										
IMOV				S								
IMOVE*												
INUMFRQ								B				
INUMOP					S			U				
IREF				B								
IRNO*		S										
ISEA				B								

DATA ITEM	ROUTINES INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
ISLOT*												
ITEMP								B				
ITIME	S											
ITYPE					S							
IYR	S											
JLIN		B										
JLIN1		B										
JMAX				B								
JSEA		B		U								
LAT	S	U					U					
LFRQLM*												
LON	S	U					U					
LUN*				U								
LUNTG	S				U							
NB	B											
NDBLK				B								

DATA ITEM	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
	ROUTINES											
NDP	S	U	U									
NF	B											
NF1	B											
NNDP		B										
NPOINT						B						
NX*												
NY*												
NZP		B										
PRDFRQ					B							
RANGE	S											
S35										B		
SLD		U							S			
SHPDEN		U		S								
SS	S											
TA												B
TDEL						B						

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
TGTBBN					S							
TGTDEP					S							
TGTSPD					S							
THI							B					
TLO							B					
TMAX							B					
TMIN							B					
UFRQLM												
VELDSC									B			
VELSLD									B			
WH	S											
WS	S											U
XDEP	B											
XHI*												
XLAT		B		U								
XLATMN				B								

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
XLATMX				B								
XLO*												
XLON		B		U								
XLONMN				B								
XLONMX				B								
XMAX		B										
XMDUL*		B										
XMIN		B										
XMINDP	B											
XTEMP	B											
YHI*												
YLO*												
ZBOT						U						
ZF												U
<u>CONSTANTS</u>												
FRCENT		B										

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
FTMT	B	B										
LUNAT		B										
LUNIM		B										
LUNOS		B										
LUNP		B										
LUNSN		B										
FLAGS												
ICBOT	B											
IC1	B											
IC3				B								
ICHNG	B											
ICQ1	B											
ICQ2	B											
IDN					B							
IE*	S	S		S	S							
IENV	B											

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTCRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
IEW	S	U										
ILIN		S										
IND		S										
INOIS	B											
INS	S	U										
INUB	B											
IPAGE		B										
IPRF	S											
IPROF	S	U										
IQ		B										
IQ1		B										
IQ6		B										
ISTAT				U	U	U						
KCHNG	B											
LINDA				B								
MF	B											

DATA ITEM	ROUTINES											
	INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
MOE	B	U					U					
Q		B										
INDEXES												
IB		U		S								
IC	B											
IC2	B											
ICQ	B											
IH								B				
IH1								B				
IL								B				
IL1								B				
INCHNG	B											
ISONAR	S											
IST	S											
ITGT	S				U							
ITOM	S				U							

DATA ITEM	ROUTINES INPUT	INPUT:OV	BTGRAPH	GETENV	GETTGT	MERGE	PFGRAPH	SLFRQ	TWDPT	WILSON	XNTERP	XNTF
JI	B											
NOPTM		U				B				U		
NOPTS		U		B		U						
NDP	B											

YLO	Used in plotting routines
LFRQLM	Set in GETSONAR
UFRQLM	Set in GETSONAR
IE	Used in IOERR

3.8 NOTES This section is a list of all subroutines and functions utilized within the input module.

MNEMONIC LABEL	TEXT NAME
BTGRAPH	Balhythernograph graphic
GETENV	Get environmental
GETSONAR	Get sonar
GETTGT	Get target
HOLD	Hold
IOERR	Input/output/error
MERGE	Merge
MOVBR	Move backwards
MOVFR	Move forwards
PFGRAPH	Profile graph
SLFRQ	Select frequency
TR720	Tape read 720 integers of data
TRWND	Tape rewind
TWDPT	Tow depth
WILSON	Wilson
XNTERP	Extrapolate
XNTF	Interpolate

CHAPTER 4
Program Package

4.1 INTRODUCTION This chapter and related material enable persons with a machine configuration identical to the TASSRAP II OB System to execute the input module. This configuration requires a Data General NOVA 800 Series CPU, Xebec XMD 5000 Disk Formatter/Controller, Caelus Model 303/2 Disk Drive, TEKTRONIX 4002A Graphic terminal, and DICOM 344 Cassette tape system. In addition, disk packs must be dual density, 16 sectors, with 192 words/sector.

4.1.1 Purpose It is the intent of the Program Package to disseminate the input module to authorized installations in a form suitable for loading and execution.

4.1.2 Scope This document has been structured so that systems personnel can obtain a complete understanding of the input module.

4.2 SOURCE DIGITAL PROCESSOR PROGRAM This Program Package item is a source form of the input module on a cassette tape and disk pack.

4.3 OBJECT PROGRAM TAPE This Program Package item is a relocatable binary form of the input module and data items on cassette tape and disk pack. Using this item enables the operator to load and execute the input module.

4.4 SOURCE PROGRAM LISTING This Program Package item is a listing of the source language program. Organization of the listing is INPUT followed by associated subroutines in alphabetical order, then INPUT:OV followed by associated subroutines and functions.

```
C      ********************************************************
C      ********************************************************
C      ******                                            ******
C      ******      INPUT PROGRAM FIRST SEGMENT           ******
C      ******                                            ******
C      ********************************************************
C      ********************************************************
C
C
C      THIS SUBROUTINE PROVIDES THE INPUT MECHANISM
C      FOR ALL DATA REQUIRED TO RUN THE TASSRAP PROGRAM
C
C
C
       COMMON/TEMP/XLAT,XLON
       COMMON/TERM/NXO,NYO,NPX,NPY
       COMMON/PLO/MODEX,IXX,IYX
       COMMON/CMBTGF/MDUM(30)
       COMMON/ATLIO/IRNO
       COMMON/TGT/IFRQ(4,11),IDN,ITYPE,PRDFRQ
       COMMON/XXXEC/IXXEC
       COMMON/ARCH/IENU,IC0(41)
       COMMON/XDATA/LABEL(10),ITIME,IDATE(3),LAT,INS,LON,IEW,
     1RANGE,WH,BOTZ,SS,WS,IB,ITGT,ITOM,IST,ISONAR,FREQ(2,5),
     2INUMFRQ,TGTDEP,TGTSPD,TGTBBN,TOWDP(5),INUMDPS,DSC,IPROF,SLD,DMAX
       COMMON/ENV/Z(50),T(50),S(50),ZO(31),TOB(31),ZM(50),TM(50),
     1SM(50),VM(50),DEP(31),TEMP(31),NOPTS,MDP,NOPTM,MOE,SHPDEN
       COMMON/NOISE/NB,NF1,IBEAM(24),FREQN(24,5),LEVELN(24,5)
       DIMENSION JI(10),XTEMP(10),XDEP(10),XF(10),D(50),TAB(800)
       REAL LFREQLM,LEVELN
C
C
       EQUIVALENCE (IDA,IDATE(1)),(IMO,,IDATE(2)),(IYR,IDATE(3))
C
C
C
```

```
C         CHECK FOR FIRST TIME THROUGH
C
          IF(IENV.GE.0) GO TO 5000
C
C         IT'S FOR REAL - THIS IS THE BEGINNING
C
          CALL FRSFL
          IC=0
          KCHNG=0
          CALL CHAR(31)
          CALL HOLD
          WRITE(11,900)
900       FORMAT(5X,///)
          CALL CHAR(14)
          TYPE '                         ***** TASSRAP INPUT PROGRAM *****'
          CALL CHAR(31)
          WRITE(11,900)
          TYPE ' 1 LABEL = '
          READ (10,1000)LABEL(1)
1000      FORMAT (S20)
          ACCEPT ' 2 DAY = ',IDA
          ACCEPT ' 3 MONTH = ',IMO
          ACCEPT ' 4 YEAR = ',IYR
          ACCEPT ' 5 TIME = ',ITIME
          ACCEPT ' 6 LATITUDE = ',LAT
          ACCEPT ' 7 NORTH(1)-SOUTH(2) = ',INS
          ACCEPT ' 8 LONGITUDE = ',LON
          ACCEPT ' 9 EAST(1)-WEST(2) = ',IEW
          ACCEPT '10 MAXIMUM RANGE(NM) = ',RANGE
          RANGE=RANGE+.001
          ACCEPT '11 WAVE HEIGHT(FT) = ',WH
          WH=WH+.001
          ACCEPT '12 WIND SPEED(KT) = ',WS
          WS=WS+.001
          ACCEPT '13 SHIP SPEED(KTS) = ',SS
```

```
      SS=SS+.001
      IENV=1
10    CALL CHAR (10)
      IF((IMO.LT.1).OR.(IMO.GT.12)) WRITE(11,1005)
1005  FORMAT(" YOUR SELECTION FOR THE MONTH IS INVALID")
      IF((INS.NE.1).AND.(INS.NE.2)) WRITE(11,200)
      IF((IEW.NE.1).AND.(IEW.NE.2)) WRITE(11,200)
200   FORMAT("YOUR SELECTION OF DIRECTION IS INVALID")
      ACCEPT "CHANGE ANY DATA?  1=YES 0=NO---",IC1
      IF((IC1.NE.0).AND.(IC1.NE.1)) WRITE(11,1027)
      IF((IC1.NE.0).AND.(IC1.NE.1)) GO TO 30
      IF(IC1.EQ.1) GO TO 20
      IF(KCHNG.EQ.0) GO TO 40
      IF(ITGT.EQ.8) GO TO 8889
      GO TO 443
20    ACCEPT 'INPUT LINE NUMBER TO BE CHANGED ---',IC
      IF((IC.EQ.3).OR.(IC.EQ.6).OR.(IC.EQ.7).OR.
     *(IC.EQ.8).OR.(IC.EQ.9).OR.(IC.EQ.12))IENV=1
      IF((IC.EQ.6).OR.(IC.EQ.7).OR.(IC.EQ.8).OR.(IC.EQ.9))
     * BOTZ=0.
      IF((IC.GT.20).OR.(IC.LT.1))WRITE(11,1027)
      IF((IC.GT.20).OR.(IC.LT.1)) GO TO 30
      IF((IC.GT.13).AND.(KCHNG.EQ.0)) WRITE(11,1027)
      IF((IC.GT.13).AND.(KCHNG.EQ.0)) GO TO 20
      IF(IC.LT.14) GO TO 23
      ICQ=IC-13
      GO TO (40,442,1035,4441,9971,4459,21)ICQ
21    IF(IPROF.EQ.1) GO TO 9945
      GO TO 70
23    CALL CHAR(1)
      IC2=IC+9
      DO 25 I=1,IC2
      WRITE(11,910)
910   FORMAT(5X)
25    CONTINUE
```

```
      GO TO  (1,2,3,4,5,6,7,8,9,10,11,12,13)IC
1     CALL FCNOT ("                                 1 LABEL = ")
      READ(10,1000)LABEL(1)
      GO TO 30
2     CALL FCNOT ("                                 2 DAY = ")
      READ(10)IDA
      GO TO 30
3     CALL FCNOT ("                                 3 MONTH = ")
      READ(10)IMO
      GO TO 30
4     CALL FCNOT ("                                 4 YEAR = ")
      READ(10)IYR
      GO TO 30
5     CALL FCNOT ("                                 5 TIME = ")
      READ (10)ITIME
      GO TO 30
6     CALL FCNOT ("                                 6 LATITUDE = ")
      READ (10)LAT
      GO TO 30
7     CALL FCNOT ("                                 7 NORTH(1)-SOUTH(2) =
      READ (10)INS
      GO TO 30
8     CALL FCNOT ("                                 8 LONGITUDE = ")
      READ (10)LON
      GO TO 30
9     CALL FCNOT ("                                 9 EAST(1)-WEST(2) = "
      READ(10)IEW
      GO TO 30
10    CALL FCNOT ("                                10 MAXIMUM RANGE = ")
      READ(10)RANGE
      RANGE=RANGE+.001
      GO TO 30
11    CALL FCNOT ("                                11 WAVE HEIGHT = ")
      READ (10)WH
      WH=WH+.001
```

```
      GO TO 30
12    CALL FCHOT ("                                    12 WIND SPEED = ")
      READ (10)WS
      WS=WS+.001
      GO TO 30
13    CALL FCHOT ("                                    13 SHIP SPEED = ")
      READ (10)SS
      SS=SS+.001
      GO TO 30
C
C     ENTRANCE FOR SUCCESSIVE RUNS OF INPUTS
C
5000  KCHNG = 1
      MF = 1
C
30    CALL CHAR(31)
      IF(IENV.EQ.2) GO TO 9977
      BOTZ1=BOTZ
      IF(MF.EQ.1) BOTZ1=BOTZ103.2808
      CALL HOLD
      WRITE(11,900)
      CALL CHAR(14)
      TYPE '                     ***** TASSRAP INPUT PROGRAM *****'
      CALL CHAR(31)
      WRITE(11,900)
      WRITE (11,1010)LABEL(1),IDA,IMO,IYR,ITIME,LAT,INS,LON,IEW,
     1RANGE,WH,WS,SS
1010  FORMAT (' 1 LABEL = ',S20,/,' 2 DAY = ',I4,/,' 3 MONTH = ',I4,
     1/,' 4 YEAR = ',I4,/,' 5 TIME = ',I6,/,' 6 LATITUDE = ',I6,/,
     2' 7 NORTH(1)-SOUTH(2) = ',I4,/,' 8 LONGITUDE = ',I8,/,' 9 EAST
     3(1)-WEST(2) = ',I4,/,' 10 MAXIMUM RANGE(NM) = ',F8.1,/,'
     4 11 WAVE HEIGHT(FT) = ',F6.1,/,
     5' 12 WIND SPEED(KTS) = ',F6.1,/,' 13 SHIP SPEED(KTS) = ',F8.1)
      IF((KCHNG.EQ.1).AND.(BOTZ.NE.0.))
     1 WRITE(11,1011)ITGT,ITOM,IST,ISONAR,BOTZ1
```

```
1011     FORMAT(" 14 TARGET TYPE = ",I4,/," 15 TARGET",
       2 " OP.MODE = ",I4,/," 16 TYPE OF MISSION = ",I4,/,
       3 " 17 SONAR TYPE = ",I4,/," 18 CHANGE BT INPUT",/,
       4 " 19 CHANGE BEAM NOISE DATA",/," 20 BOTTOM",
       5 " DEPTH = ",F7.1)
         IF((MF.EQ.1).AND.(KCHNG.EQ.1).AND.(BOTZ.NE.0))
      1 TYPE" BOTTOM DEPTH IS IN FEET"
         IF((KCHNG.EQ.1).AND.(BOTZ.EQ.0.))
       1 WRITE(11,1012)ITGT,ITOM,IST,ISONAR
1012     FORMAT(" 14 TARGET TYPE = ",I4,/," 15 TARGET",
       2 " OP. MODE = ",I4,/," 16 TYPE OF MISSION = ",I4,/,
       3 " 17 SONAR TYPE = ",I4,/," 18 CHANGE BT INPUT",/,
       4 " 19 CHANGE BEAM NOISE DATA",/," 20 BOTTOM DEPTH",/,)
         GO TO 10
40       CALL HOLD
         CALL CHAR(14)
         TYPE '                         ***** TASSRAP INPUT PROGRAM *****'
         CALL CHAR(31)
         WRITE(11,1020)
         READ (10)ITGT
1020     FORMAT (///,"**** TARGET TYPE ****",///,5X,"1)SOVIET NUCLEAR
       1 TYPE 1",/,5X,
       1"2)SOVIET NUCLEAR TYPE 2",/,5X,"3)SOVIET NUCLEAR TYPE 3",/,
       25X,"4)SOVIET DIESEL TYPE 1 (F,R,W,Z)",/,5X,"5)SOVIET DIESEL
       2 JULIET (TYPE 2)",/,5X,"6)SOVIET DIESEL FOXTROT (TYPE 3)",/,
       35X,"7)US NUCLEAR SSN 637 CLASS",/,5X,"8)OWN SOURCE LEVELS
       4",/,"WHICH TARGET TYPE?---")
         IF(ITGT.LT.1.OR.ITGT.GT.8) WRITE(11,1027)
1027     FORMAT(//,5X,"***** YOUR ENTRY IS INVALID *****",/,
       17X,"*** HIT SPACE BAR TO CONTINUE ***")
         IF(ITGT.LT.1.OR.ITGT.GT.8) GO TO 40
         IF (ITGT.EQ.8) GO TO 41
         IF(KCHNG.NE.0) GO TO 30
         GO TO 441
41       WRITE(11,1025)
```

```
1000    FORMAT(//,5X)
        TYPE"MAXIMUM NUMBER OF FREQUENCIES = 5"
        ACCEPT"NUMBER OF FREQUENCIES = ",NF
        IF((NF.LT.1).OR.(NF.GT.5)) WRITE(11,1027)
        IF((NF.LT.1).OR.(NF.GT.5)) GO TO 41
        IF(NF.GT.5) WRITE(11,1027)
        IF(NF.GT.5) GO TO 41
        ACCEPT "INPUT TARGET DATA IN FREQUENCY - SOURCE LEVEL
      $ PAIRS","<15>",(FREQ(1,I),FREQ(2,I),'<15>',I=1,NF)
        DO 39 I=1,NF
        FREQ(1,I)=FREQ(1,I)+.001
        FREQ(2,I)=FREQ(2,I)+.001
39      CONTINUE
42      CALL CHAR(12)
        WRITE(11,10260)
10260   FORMAT(//,1X,"*** FREQUENCY INPUT DATA ***",
      $//," LINE FREQUENCY LEVEL")
        WRITE(11,1026)(I,FREQ(1,I),FREQ(2,I),I=1,NF)
1026    FORMAT(/,2X,I3,3X,F6.1,2X,F6.1)
        WRITE (11,1000)NF
        ACCEPT"FREQUENCY-LEVEL PAIRS (1=YES,0=NO)?=",ICHNG
        IF((ICHNG.NE.0).AND.(ICHNG.NE.1)) WRITE(11,1027)
        IF((ICHNG.NE.0).AND.(ICHNG.NE.1)) GO TO 42
        IF(ICHNG.NE.1) GO TO 442
        ACCEPT "NUMBER OF POINTS TO BE CORRECTED =",INCHNG
        IF((INCHNG.LT.1).OR.(INCHNG.GT.5)) WRITE(11,1028)
        IF((INCHNG.LT.1).OR.(INCHNG.GT.5)) GO TO 42
1028    FORMAT(//,5X,"***** YOUR ENTRY IS INVALID*****")
        ACCEPT"INPUT LINE NUMBER AND CORRECT FREQUENCY-LEVEL
      $ PAIRS","<15>",("$",JI(I),XDEP(I),XTEMP(I),"<15>",I=1,INCHNG)
        DO 43 I=1,INCHNG
        N=JI(I)
        FREQ(1,N)=XDEP(I)
        FREQ(2,N)=XTEMP(I)
        FREQ(1,N)=FREQ(1,N)+.001
```

```
          FREQ(2,N)=FREQ(2,N)+.001
          IF(N.GT.NF+1) GO TO 44
          IF (N.GT. NF) NF = N
43        CONTINUE
          GO TO 42
44        WRITE(11,1098)
          GO TO 42
442       CALL HOLD
          CALL CHAR(14)
          TYPE"     ***** TASSRAP INPUT PROGRAM *****"
441       CALL CHAR(31)
          CALL CHAR(10)
          WRITE(11,1030)
          READ (10)ITOM
 1030 FORMAT (///,"*** TARGET OPERATIONAL MODE ***",//,5X,"1)TRANSI
     1T",/,5X,"2)AREA SEARCH-ASW",/,5X,"3)AREA SEARCH-SURFACE SHIPS",/,5X
     2RRIER",/,5X,"5)CONVOY PENETRATION",/,5X,"6)AMPHIBIOUS ATTACK",/,5X,
     3 ATTACK",/,5X,"8)SSBN OPERATIONS",/,5X,"9)SURVEILLANCE-ASW",/,5X,
     4"10)SURVEILLANCE-SURFACE SHIPS",/,5X,"11)SNORKEL",/,5X,"12)INPUT
     5SOURCE DEPTH",/,"WHICH TARGET OPERATION MODE?---")
          IF(ITOM.LT.1.OR.ITOM.GT.12) WRITE(11,1027)
          IF(ITOM.LT.1.OR.ITOM.GT.12) GO TO 442
          IF(ITOM.EQ.12.OR.ITGT.EQ.8) GO TO 4425
          IF(KCHNG.NE.0) GO TO 30
          GO TO 443
4425      ACCEPT"SOURCE DEPTH (FEET) = ",TGTDEP
          IF(KCHNG.NE.0) GO TO 30
          IF(ITGT.NE.8) GO TO 443
          GO TO 444
443       CALL FGTFS(LUNTG,IE)
          IF(IE.NE.0) CALL IOERR("TGTFL",LUNTG,IE)
          CALL FOPFL("TGTFL",LUNTG,IE)
          IF(IE.NE.0) CALL IOERR("TGTFL",LUNTG,IE)
          CALL GETTGT(LUNTG,ISTAT)
          CALL FCLFL(LUNTG,IE)
```

```
      IF(IE.NE.0) CALL IOERR("TGTFL",LUNTG,IE)
      IF(ISTAT.EQ.1) GO TO 1034
      IF(ISTAT.EQ.3.AND.KCHNG.NE.0) GO TO 30
      IF(ISTAT.EQ.3) GO TO 40
      TYPE "***FATAL TGT RETRIEVAL ERROR***"
      CALL FRSFL
      STOP
1034  IF(KCHNG.NE.0) GO TO 80
1035  CALL HOLD
      CALL CHAR(12)
      CALL CHAR(14)
      TYPE '                    ***** TASSRAP INPUT PROGRAM *****'
444   CALL CHAR(31)
      WRITE(11,1040)
      READ (10)IST
 1040 FORMAT (///,"**** OWN SHIP TYPE OF MISSION ****",//,5X,"1)SURVEIL
     *LANCE",/,5X
     1,"2)ESCORT",/,5X,"3)TRAIL",/,5X,"4)AREA SANITIZATION",/,5X,"
     *5)AM
     2PHIBIOUS ASSAULT PROTECTION",/,"WHICH TYPE OF MISSION?---")
      IF(IST.LT.1.OR.IST.GT.5) WRITE(11,1027)
      IF(IST.LT.1.OR.IST.GT.5) GO TO 1035
      IF(KCHNG.NE.0) GO TO 30
      IF(ITGT.LT.8) GO TO 445
4441  CALL HOLD
      CALL CHAR(12)
      CALL CHAR(14)
      TYPE "                  ***** TASSRAP INPUT PROGRAM *****"
      CALL CHAR(31)
445   WRITE(11,1050)
      READ(10)ISONAR
 1050 FORMAT (///,"**** SONAR TYPE ****",/,5X,"1)AN/SQR-15",/,5X,"2)AN
     1/BQR-15",/,5X,"3)STASS",/,5X,"4)TACTASS",/,5X,"5)LAMBDA"
     2,/,"WHAT TYPE OF SONAR?---")
      IF(ISONAR.LT.1.OR.ISONAR.GT.5) WRITE(11,1027)
```

```
      IF(ISONAR.LT.1.OR.ISONAR.GT.5) GO TO 4441
      TYPE
      TYPE
      IF(KCHNG.NE.0) GO TO 30
      IF(ITGT.EQ.0) GO TO 9977
9971  CALL HOLD
      TYPE
      CALL CHAR(14)
      TYPE "                         ***** TASSRAP INPUT PROGRAM *****"
      CALL CHAR(31)
      TYPE
      IF(KCHNG.EQ.0) GO TO 9977
      IF(IPROF.EQ.1) GO TO 45
      IPROF=1
      GO TO 9972
9977  ACCEPT "INPUT BT?  1=YES 0=NO---",IPRF
      IF((IPRF.NE.1).AND.(IPRF.NE.0)) WRITE(11,1027)
      IF((IPRF.NE.1).AND.(IPRF.NE.0)) GO TO 9971
      CALL CHAR(31)
      IPROF=2
      IF(IPRF.EQ.1)IPROF=1
      IF (IPRF.EQ.0) GO TO 70
944   TYPE"THE FIRST DEPTH MUST BE 0, AND THE LAST INPUT"
      TYPE"DEPTH MUST BE EQUAL TO OR GREATER THAN 300 METERS"
      TYPE"FOR METRIC INPUT, OR EQUAL TO OR GREATER THAN 1,000"
      TYPE"FEET FOR ENGLISH INPUT"
9972  ACCEPT"NUMBER OF DATA POINTS IN PROFILE = ",NDP
      IF(NDP.LT.2) CALL CHAR(25)
      IF(NDP.LT.2) WRITE(11,10290)
10290 FORMAT(" ***** YOU ARE A DUMBSHIT *****")
      CALL CHAR(31)
      IF(NDP.LT.2) WRITE(11,1029)
1029  FORMAT(//,5X,"**** YOUR ENTRY FOR NUMBER OF DATA
     &POINTS IS INVALID",/,5X,"***** HIT SPACE BAR TO CONTINUE")
      IF(NDP.LT.2) GO TO 944
```

```
201     CALL HOLD
        ACCEPT"UNITS OF DATA, 1=METRIC, 2=ENGLISH---",MOE
        IF((MOE.NE.1).AND.(MOE.NE.2)) WRITE(11,1027)
        IF((MOE.NE.1).AND.(MOE.NE.2)) GO TO 201
        CALL HOLD
        WRITE(11,900)
        CALL CHAR(14)
        TYPE '                 ***** BATHYTHERMOGRAPH INPUT *****'
        CALL CHAR(31)
        WRITE(11,900)
        TYPE"INPUT PROFILE DATA IN DEPTH,TEMPERATURE PAIRS"
        TYPE" DATA POINT DEPTH,TEMPERATURE"
        DO 9943 I=1,NDP
        WRITE(11,1052)I
1052    FORMAT(4X,I3,6X,Z)
        ACCEPT DEP(I),TEMP(I),"<15>"
9943    CONTINUE
        XMINDP=299.9
        IF(MOE.EQ.2) XMINDP=999.9
        IF(DEP(1).EQ.0.0.AND.DEP(NDP).GE.XMINDP) GO TO 9945
        TYPE "*** INVALID BT INPUT ***  CHECK DEPTH-TEMP PAIRS"
        GO TO 944
202     CALL HOLD
9945    TYPE" DO YOU WISH TO ENTER A BOTTOM DEPTH"
        ACCEPT"YES(1)-NO(0) ?",ICQ1
        IF((ICQ1.NE.0).AND.(ICQ1.NE.1)) WRITE(11,1028)
        IF((ICQ1.NE.0).AND.(ICQ1.NE.1)) GO TO 202
        MF=0
        IF(ICQ1.EQ.0) BOTZ = 0.
        IF(ICQ1.EQ.0) GO TO 9455
        IF(MOE.EQ.1) TYPE" BOTTOM DEPTH UNITS MUST BE METERS"
        IF(MOE.EQ.1) GO TO 9454
        IF(MOE.EQ.2) TYPE" BOTTOM DEPTH UNITS MUST BE FEET"
9454    ACCEPT"BOTTOM DEPTH = ",BOTZ
9455    IF(KCHNG.NE.0.AND.IC.EQ.20) GO TO 30
```

```
45      CONTINUE
        CALL HOLD
        CALL BTGRAPH (DEP,TEMP,NDP,MOE)
        CALL HOME
        CALL CHAR(14)
        TYPE '                    ***** BATHYTHERMOGRAPH INPUT *****'
        CALL CHAR(31)
        DO 9946 I=2,NDP
9946    DEP(I)=DEP(I)+.001
        DO 9947 I=1,NDP
9947    TEMP(I)=TEMP(I)+.001
        WRITE(11,1070)(I,DEP(I),TEMP(I),I=1,NDP)
 1070   FORMAT("  ***BT DATA INPUT***",//,"  LINE   DEPTH   TEMP",/,
     1(I5,3X,F6.0,3X,F4.1))
203     CALL HOLD
        WRITE (11,1080)NDP
1080    FORMAT ("CHANGE ANY OF THE",I3)
        ACCEPT "DEPTH-TEMPERATURE PAIRS?  1=YES 0=NO---",ICHNG
        IF((ICHNG.NE.0).AND.(ICHNG.NE.1)) WRITE(11,1027)
        IF((ICHNG.NE.0).AND.(ICHNG.NE.1)) GO TO 203
        IF((ICHNG.NE.1).AND.(BOTZ.NE.0.)) GO TO 71
        IF(ICHNG.NE.1) GO TO 75
        ACCEPT"NUMBER OF PAIRS TO BE CHANGED = ",INCHNG
        TYPE
        TYPE "INPUT---LINE*,DEPTH,TEMP"
        DO 50 I=1,INCHNG
        ACCEPT "* ",JI(I),XDEP(I),XTEMP(I)
        N=JI(I)
        DEP(N)=XDEP(I)
        TEMP(N)=XTEMP(I)
        IF(N.GT.NDP+1) GO TO 60
        IF(N.GT.NDP)NDP=N
 50     CONTINUE
        GO TO 45
 60     WRITE(11,1090)
```

```
1000    FORMAT(///,"##### LINE SEQUENCE ERROR TRY AGAIN @@@@@")
        GO TO 45
204     CALL HOLD
70      TYPE"DO YOU WISH TO ENTER A BOTTOM DEPTH"
        ACCEPT"YES(1)-NO(0) ?",ICQ2
        IF((ICQ2.NE.0).AND.(ICQ2.NE.1)) WRITE(11,1027)
        IF((ICQ2.NE.0).AND.(ICQ2.NE.1)) GO TO 204
        MF=0
        IF(ICQ2.EQ.0) BOTZ =0.
        IF(ICQ2.EQ.0.) GO TO 75
        ACCEPT"BOTTOM DEPTH(METERS) = ",BOTZ
        GO TO 75
71      BOTZ=BOTZ+.01
72      TYPE"DO YOU WISH TO CHANGE BOTTOM DEPTH FROM"
        WRITE(11,2000)BOTZ
2000    FORMAT(9X,F8.0)
        ACCEPT "YES=1,NO=0 ?",ICBOT
        IF((ICBOT.NE.0).AND.(ICBOT.NE.1))WRITE(11,1028)
        IF((ICBOT.NE.0).AND.(ICBOT.NE.1)) GO TO 72
        IF (ICBOT.NE.1) GO TO 75
        TYPE"INPUT NEW BOTTOM DEPTH IN THE SAME UNITS"
        ACCEPT"    AS BT =",BOTZ
75      CONTINUE
        IF(KCHNG.NE.0) GO TO 30
        CALL FRSFL
C
C       ###### LOAD SECOND SEGMENT OF THE TASSRAP INPUT PROGRAM @@@@@
C
4459    CALL HOLD
        TYPE
        CALL CHAR(14)
        TYPE "                    ##### TASSRAP INPUT PROGRAM @@@@@"
        TYPE
        CALL CHAR(31)
        CALL CHAR(10)
```

```
      IF(KCHNG.NE.0.OR.IC.EQ.19) GO TO 790
      TYPE" DO YOU WISH TO INPUT MEASURED BEAM NOISE DATA ?"
      ACCEPT"YES (1) -NO (0) ? ",IC2
      IF((IC2.NE.0).AND.(IC2.NE.1)) WRITE(11,1027)
      IF((IC2.NE.0).AND.(IC2.NE.1)) GO TO 4459
      IF(IC2.EQ.0) GO TO 80
      TYPE"ENTER DATA WITH ONE DESIRED BEAM NUMBER"
      TYPE"FOLLOWED BY THE DESIRED FREQUENCY LEVEL PAIRS WITH ALL"
      TYPE"NUMBERS SEPARATED BY COMMAS.  AFTER DOING THIS"
      TYPE"STRIKE THE RETURN KEY AND FOLLOW THE SAME "
      TYPE"PROCEDURE FOR ANY ADDITIONAL FREQUENCIES."
      TYPE"*** NOTE MAXIMUM OF 5 FREQUENCIES AND 24 BEAMS"
      TYPE" ARE ALLOWED ***"
      ACCEPT" NUMBER OF FREQUENCIES ?",NF1
      ACCEPT" NUMBER OF BEAMS ?",NB
      ACCEPT" BEAM NUMBER, FREQUENCY,LEVEL","<15>",
     *(IBEAM(I),(FREQN(I,J),LEVELN(I,J),J=1,NF1),"<15>",
     *I=1,NB)
      IF(KCHNG.EQ.0) GO TO 80
790   WRITE(11,1091)
1091  FORMAT(1X,"BEAM NUMBER FREQUENCY LEVEL")
      DO 792 I=1,NB
      DO 794 J=1,NF1
      A=1.
      WRITE(11,1092)(IBEAM(I),(FREQN(I,J),LEVELN(I,J)))
1092  FORMAT(3X,I3,7X,F8.1,3X,F6.1)
      IF(A.EQ.25.) CALL HOLD
      IF(A.EQ.25.) A=0.
      A=A+1.
794   CONTINUE
792   CONTINUE
      ACCEPT"DO YOU WISH TO CHANGE ANY NOISE
     * DATA YES=1 NO=0 ",INOIS
      IF((INOIS.LT.0).OR.(INOIS.GT.1)) WRITE(11,1027)
      IF((INOIS.LT.0).OR.(INOIS.GT.1)) GO TO 4459
```

```
      IF(INOIS.EQ.0) GO TO 30
      ACCEPT" ON WHICH BEAM NUMBER IS THE ERROR "
     #,ICHB
      K=1
      DO 795 I=1,NB
      IF(IBEAM(I).EQ.ICHB) GO TO 800
      K=K+1
795   CONTINUE
      TYPE" THERE IS NO BEAM NUMBER EQUAL TO YOUR INPUT"
      ACCEPT" DO YOU WISH TO INCLUDE THAT BEAM YES=1 NO=0
     # ",INUB
      IF((INUB.LT.0).OR.(INUB.GT.1)) WRITE(11,1027)
      IF((INUB.LT.0).OR.(INUB.GT.1)) GO TO 4459
      IF(INUB.EQ.0) GO TO 4459
      NB=NB+1
800   TYPE" INPUT BEAM NUMBER FOLLOWED BY ALL FREQUENCY"
      ACCEPT" LEVEL PAIRS FOR THAT BEAM","<15>",(IBEAM(I),
     #(FREQN(K,J),LEVELN(K,J),J=1,NF1))
      CALL CHAR(31)
      GO TO 4459
      IF(KCHNG.NE.0) GO TO 30
30    CALL GETSONAR(DMAX,LFRQ(   JFRQLM)
      IF(ITGT.EQ.8) GO TO 8886
      CALL SLFRQ(FREQ,INUMFRQ,LFRQLM,UFRQLM)
8888  IF(KCHNG.NE.0) GO TO 8889
      KCHNG=1
      GO TO 30
8889  IF (LON.EQ.10000) IEW = 2
C
C
C     CALL TO SECOND INPUT SEGMENT
C
      CALL FRNOU("INPUT:OV",IE)
C
      TYPE "(7) "
```

```
TYPE "*** OVERLAY ERROR ***"
TYPE "SYSTEM ERROR *",IE
STOP
END
```

```
      SUBROUTINE BTGRAPH(Z,T,NOPTS,MOE)
C*****PURPOSE* THIS SUBROUTINE DISPLAYS ON THE CRT THE BT BEING
C*****         INPUT TO PROFGEN TO AID IN EDITING FOR ERRORS
C*****USES* XFSET,GLABEL,SYMBOL,PLOT,XNTF
C*****ARGUMENTS*Z-ARRAY OF DEPTH POINTS,METERS OR FEET,FLOATING-IN
C*****          T-ARRAY OF TEMPERATURE POINTS,CENTIGRADE OR
C*****            FARENHEIT,FLOATING-IN
C*****          NOPTS-NUMBER OF TRACE POINTS,FIXED-IN
C*****          MOE-METRIC OR ENGLISH UNITS INDICATER
C*****            1-METRIC UNITS
C*****            2-ENGLISH UNITS
C
C
      COMMON/CMBTGF/MMDEP(12),MFDEP(10),MBTIN(8)
      DIMENSION XF(10),Z(31),T(31)
      DATA MMDEP(1)/68,69,80,84,72,32,77,69,84,69,82,83/
      DATA MFDEP(1)/68,69,80,84,72,32,70,69,69,84/
      DATA MBTIN(1)/66,84,32,73,78,80,85,84/
C
C               BRANCH TO APPROPRIATE SECTION FOR
C                METRIC OR ENGLISH UNITS
C
      GO TO(10,15),MOE
C
C               SETUP FOR METRIC UNITS
C
   10 X0    = 0.0
      Y0    = 0.0
      XLIM  = 35.0
      YLIM  = 400.0
      XINC  = 2.5
      YINC  = 25.0
      NX    = 14
      NY    = 16
      L1ST  = 0
      L1INC = 5
      NL1   = 8
      L2ST  = 0
```

```
      L2INC = 25
      NL2   = 17
      GO TO 20
C
C           SETUP FOR ENGLISH DATA
C
   15 X0    = 30.0
      Y0    = 0.0
      XLIM  = 90.0
      YLIM  = 1500.0
      XINC  = 5.0
      YINC  = 100.0
      NX    = 12
      NY    = 15
      L1ST  = 30
      L1INC = 10
      NL1   = 7
      L2ST  = 0
      L2INC = 100
      NL2   = 16
C
C           OUTPUT GRID AND LABELS
C
   20 CALL XFSET(X0,Y0,XLIM,YLIM,650,650,1000,20,XF)
      CALL GRID(XF,XINC,YINC,NX,NY)
      CALL GLABEL(L1ST,L1INC,NL1,1,XF)
      CALL GLAMEL(L2ST,L2INC,NL2,4,XF)
      GO TO(22,23),MOE
C
C           LABEL FOR METRIC
C
   22 CALL SYMBOL(575,450,MMDEP,12,1)
      GO TO 25
C
C           LABEL FOR ENGLISH
C
   23 CALL SYMBOL(575,450,MFDEP,10,1)
C
C           THIS SECTION USED BY BOTH METRIC AND ENGLISH
```

```
C
   25   CALL SYMBOL(75Ø,675,M3TIN,8,Ø)
        CALL PLOT(T(1),Z(1),XF,Ø)
C
C              BRANCH ON MOE AGAIN
C
        GO TO(4Ø,3Ø),MOE
C
C              ENGLISH INPUT DATA AGAIN
C
   3Ø   DO 35 I = 2,NOPTS
        IF(Z(I) .LT. 15ØØ.)GO TO 33
C
C              INTERPOLATE TO 15ØØ FEET IF DEEPER
C
        TINTR = XNTF(15ØØ.,Z,T,NOPTS)
        CALL PLOT(TINTR,15ØØ.,XF,1)
        GO TO 5Ø
   33   CALL PLOT(T(I),Z(I),XF,1)
   35   CONTINUE
        GO TO 5Ø
C
C              METRIC INPUT DATA AGAIN
C
   4Ø   DO 45 I = 2,NOPTS
        IF(Z(I) .LT. 4ØØ.)GO TO 43
C
C              INTERPOLATE TO 4ØØ METERS IF DEEPER
C
        TINTR = XNTF(4ØØ.,Z,T,NOPTS)
        CALL PLOT(TINTR,4ØØ.,XF,1)
        GO TO 5Ø
   43   CALL PLOT(T(I),Z(I),XF,1)
   45   CONTINUE
   5Ø   CALL HOME
        RETURN
        END
```

```
      SUBROUTINE GETSONAR(DMAX,LFRQLM,UFRQLM)
C
C*************************************************************
C*************************************************************
C*************************************************************
C**********      THIS ROUTINE HAS NOT BEEN      *************
C**********      CODED AS OF YET                ************
C**********                                     ************
C*************************************************************
C*************************************************************
C
C
X     CALL HOLD
X     TYPE"ENTERED GETSONAR"
      DMAX=3000.    ;   DMAX = MAX DEPTH FOR THE HYDROPHONE (3000. FE
      LFRQLM=10.    ;   LFRQLM = LOWER FREQ. LIMIT FOR THE SONAR IN U
      UFRQLM=1000. ;   UFRQLM = UPPER FREQ. LIMIT FOR THE SONAR IN U
C
C
      RETURN
      END


>>
```

```
C         THE VALUES OF ISTAT WHICH MAY BE RETURNED TO THE MAINLINE PROG
C         ARE AS FOLLOWS.  THE VALUE OF ISTAT DESIGNATES THE TYPE OF ERR
C         ENCOUNTERED.
C
C**********************************************************
C*****************                              **********
C*****************  1 - RETRIEVAL OK            **********
C*****************  2 - INVALID BLOCK NUMBER    **********
C*****************  3 - INVALID TGT OP. MODE    **********
C*****************  4 - INVALID LIMIT INFORMATION  *******
C*****************  5 - DATAFILE FAILURE        **********
C*****************  6 - END OF FILE-DATA NOT FOUND *******
C*****************  7 - SYSTEM I/O ERROR        **********
C*****************                              **********
C**********************************************************
C
C
          SUBROUTINE GETTGT(LUNTG,ISTAT)    ; GET-TGT  "GET TARGET DATA"
C
C         OVERLAY-COMMUNICATIONS LABELED COMMON  (XDATA)
C
          COMMON/XDATA/LABEL(10),ITIME,IDATE(3),LAT,INS,LON,IEW,
        & RANGE,WH,BOTZ,SS,WS,IB,ITGT,ITOM,IST,ISONAR,FREQ(2,5),
        & INUMFRQ,TGTDEP,TGTSPD,TGTBDN,TOWDP(5),INUMDPS,DSC,IPROF,SLD,DM
C
C
          COMMON/TGT/IFRQ(4,11),IDN,ITYPE,PRDFRQ
C
          DIMENSION IBLOCK(9,11)
C
X         CALL HOLD
X         TYPE"ENTERED GETTGT"
          DO 10 I=1,ITGT
          READ BINARY(LUNTG,END=960,ERR=970) ((IBLOCK(J,K),K=1,11),J=1,9
10        CONTINUE
C***      TEST BLOCK NUMBER
          IF(IBLOCK(1,1).NE.ITGT) GO TO 920
C***      TEST DIESEL/NUC DESIGNATION
```

```
      IF(IBLOCK(1,2).NE."N ".AND.IBLOCK(1,2).NE."D ") GO TO 950
C***  TEST TGT OP. MODE
      IF(ITOM.EQ.12) GO TO 15
      IF(ITOM.GT.11.OR.ITOM.LT.1) GO TO 930
      IF(IBLOCK(2,ITOM).EQ.0) GO TO 930
      IF(IBLOCK(2,ITOM).NE.ITOM) GO TO 950
      IF(IBLOCK(5,ITOM).EQ.0) GO TO 950
15    INUMOP=0
      DO 20 I=1,11
      IF(IBLOCK(2,I).EQ.I) INUMOP=INUMOP+1
      IF(IBLOCK(2,I).EQ.0.OR.IBLOCK(2,I).EQ.I) GO TO 20
C***  DATAFILE FAILURE
      GO TO 950
20    CONTINUE
C***  CHECK LIMIT INFO
      IF(INUMOP.NE.IBLOCK(1,4)) GO TO 945
      INUMFRQ=IBLOCK(1,6)
      DO 40 I=1,11
      IF(I.GT.INUMFRQ) GO TO 30
      IF(IBLOCK(6,I).EQ.0) GO TO 940
C***  CHECK FOR DATAFILE FAILURE
      IF(IBLOCK(7,I).EQ.0) GO TO 950
      GO TO 40
30    IF(IBLOCK(8,I).NE.0) GO TO 940
40    CONTINUE
C
C     ALL POSSIBLE FAILURES HAVE BEEN CHECKED
C
      IF(ITOM.NE.12) GO TO 41
      TGTSPD=FLOAT(IBLOCK(4,1))
      TGTBRN=FLOAT(IBLOCK(5,1))/10.
      GO TO 45
41    TGTDEP=FLOAT(IBLOCK(3,ITOM))
      TGTSPD=FLOAT(IBLOCK(4,ITOM))
      TGTBRN=FLOAT(IBLOCK(5,ITOM))/10.
45    IBN=IBLOCK(1,2)
      ITYPE=IBLOCK(1,3)
      PRDFRQ=FLOAT(IBLOCK(1,5))
      DO 50 I=1,11
```

```
      IFRQ(1,I)=IBLOCK(6,I)
      IFRQ(2,I)=IBLOCK(7,I)
      IFRQ(3,I)=IBLOCK(8,I)
      IFRQ(4,I)=IBLOCK(9,I)
50    CONTINUE
      ISTAT=1
C
C     CALL SUCESSFULLY COMPLETED
C
      RETURN    ! TO MAIN-LINE PROGRAM
C
C***  INV. BLOCK NO.
920   ISTAT=2
      TYPE "<7>***TGT RETRIEVAL ERROR*** INVALID BLOCK NUMBER"
      RETURN
C
C***  INV. TGT OP. MODE
930   ISTAT=3
      TYPE "<7>***TGT RETRIEVAL ERROR*** INVALID TGT OP. MODE"
      RETURN
C
C***  INV. LIMIT INFO.
940   ISTAT=4
      TYPE "<7>***TGT RETRIEVAL ERROR*** INVALID LIMIT INFORMATION"
      RETURN
C
C***  DATAFILE FAILURE
950   ISTAT=5
      TYPE "<7>***TGT RETRIEVAL ERROR*** DATAFILE FAILURE"
      RETURN
C
C***  END OF FILE-DATA NOT FOUND
960   ISTAT=6
      TYPE "<7>***TGT RETRIEVAL ERROR*** END OF FILE-DATA NOT FOUND"
      RETURN
C
C***  SYSTEM I/O ERROR
970   ISTAT=7
      TYPE "<7>***TGT RETRIEVAL ERROR*** SYSTEM I/O ERROR"
```

```
RETURN
END ; OF SUBROUTINE

>>
```

```
        SUBROUTINE HOLD
        CALL CHAR(7)
A               SUBO    Ø,Ø
A               NIOS    TTI
A               SKPDN   TTI
A               JMP     .-1
A               DIA     Ø,TTI
A               STA     Ø,RESP
A               LDA     1,CTRLD
A               SUB     Ø,1,SZR
A               JMP     CTRLD+1
        TYPE "OPERATOR BREAK *"
        STOP
A       RESP:   ØØØØØØ
A       CTRLD:  ØØØ2Ø4
A               LDA     Ø,RESP
A               LDA     1,HRDCPY
A               SUB     Ø,1,SZR
A               JMP     HRDCPY+1
        CALL CHAR(23)
        CALL CHAR(12)
A               IORST
        RETURN
A       HRDCPY: ØØØ215
        CALL CHAR(12)
A               IORST
        RETURN
        END

>>
```

```
      SUBROUTINE SLFRQ(FREQ,INUMFRQ,LFRQLM,UFRQLM)
      REAL LFRQLM
      COMMON/TGT/IFRQ(4,11),IDN,ITYPE,PRDFRQ
      DIMENSION IWSFRQ(3,11),FREQ(2,5)
X     CALL HOLD
X     TYPE"ENTERED SLFRQ"
      IMIN=0
      IMAX=0
      DO 5 I=1,10
      FREQ(I)=0.0
5     CONTINUE
      DO 10 I=1,11
      IF(IFRQ(1,I).EQ.0) GO TO 20
      IF(IMIN.NE.0) GO TO 6
      IF(FLOAT(IFRQ(1,I))/10..LT.LFRQLM) GO TO 10
      IMIN=I
      GO TO 10
6     IF(FLOAT(IFRQ(1,I))/10..GT.UFRQLM) GO TO 10
      IMAX=I
10    CONTINUE
20    IF(IMIN.NE.0.AND.IMAX.GE.IMIN) GO TO 30
      INUMFRQ=0
      RETURN
30    INUMFRQ=(IMAX-IMIN)+1
      IF(INUMFRQ.GT.5) GO TO 35
      DO 32 I=IMIN,IMAX
      FREQ(1,(I-IMIN)+1)=FLOAT(IFRQ(1,I))/10.
      FREQ(2,(I-IMIN)+1)=FLOAT(IFRQ(2,I))/10.
32    CONTINUE
      RETURN
35    DO 40 I=IMIN,IMAX
      IWSFRQ(1,(I-IMIN)+1)=IFRQ(1,I)
      IWSFRQ(2,(I-IMIN)+1)=IFRQ(2,I)
      IWSFRQ(3,(I-IMIN)+1)=IFRQ(3,I)
40    CONTINUE
      IEND=INUMFRQ-1
      DO 45 I=1,IEND
      IF(IABS(IWSFRQ(1,I)-IWSFRQ(1,I+1)).GT.200) GO TO 45
```

```
      IDELTALS=(IWSFRQ(2,I+1)-IWSFRQ(2,I))**2
      IDELTARL=IWSFRQ(3,I+1)-IWSFRQ(3,I)
      IF((IDELTALS+IDELTARL).GT.0) GO TO 43
      DO 42 J=1,3
      ITEMP=IWSFRQ(J,I)
      IWSFRQ(J,I)=IWSFRQ(J,I+1)
      IWSFRQ(J,I+1)=ITEMP
42    CONTINUE
43    INUMFRQ=INUMFRQ-1
      ISTART=I
      DO 44 II=ISTART,INUMFRQ
      IWSFRQ(1,II)=IWSFRQ(1,II+1)
      IWSFRQ(2,II)=IWSFRQ(2,II+1)
      IWSFRQ(3,II)=IWSFRQ(3,II+1)
44    CONTINUE
      IWSFRQ(1,INUMFRQ+1)=0
      IWSFRQ(2,INUMFRQ+1)=0
      IWSFRQ(3,INUMFRQ+1)=0
      IF(ISTART.GE.INUMFRQ) GO TO 50
      IF(INUMFRQ.LE.5) GO TO 70
      I=I-1
45    CONTINUE
50    IF(INUMFRQ.LE.5) GO TO 70
      ID=1
51    ID=ID+1
      I=INUMFRQ/ID
      IF(I.EQ.0) GO TO 60
      IL=1
      IH=I+1
52    IF(IH.GT.INUMFRQ) GO TO 51
      IL1=IL
      IH1=IH
53    IF(IWSFRQ(3,IH).LT.IWSFRQ(3,IL)) GO TO 55
      DO 54 J=1,3
      ITEMP=IWSFRQ(J,IL)
      IWSFRQ(J,IL)=IWSFRQ(J,IH)
      IWSFRQ(J,IH)=ITEMP
54    CONTINUE
      IL=IL-I
```

```
      IF(IL.LT.1) GO TO 55
      IH=IH-1
      GO TO 53
55    IL=IL1+1
      IH=IH1+1
      GO TO 52
60    INUMFRQ=5
70    DO 80 I=1,INUMFRQ
      FREQ(1,I)=FLOAT(IWSFRQ(1,I))/10.
      FREQ(2,I)=FLOAT(IWSFRQ(2,I))/10.
80    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRUND(LUN)
      IUN = LUN
      REWIND IUN
      RETURN
      END
      SUBROUTINE MOVFR(LUN,IMOVE,ISTAT)
      COMMON/ATLIO/IRNO
      DATA IRNO/0/
      IRNO = IRNO+IMOVE
      CALL FPSFL(LUN,2*144,IRNO,0,IE)
      IF(IE.EQ.0)GO TO 10
      ISTAT = 0
      CALL IOERR('ATLAS',LUN,IE)
      TYPE 'RECORD NUMBER',IRNO
      GO TO 20
   10 ISTAT = 1
   20 RETURN
      END
      SUBROUTINE MOVBR(LUN,IMOVE,ISTAT)
      COMMON/ATLIO/IRNO
      IRNO = IRNO-IMOVE
      CALL FPSFL(LUN,2*144,IRNO,0,IE)
      IF(IE.EQ.0) GO TO 10
      ISTAT = 0
      CALL IOERR('ATLAS',LUN,IE)
      TYPE 'RECORD NUMBER',IRNO
      GO TO 20
   10 ISTAT = 1
   20 RETURN
      END
      SUBROUTINE TR720(LUN,IBUF,ISTAT)
      COMMON/ATLIO/IRNO
      DIMENSION IBUF(144)
      IRNO = IRNO+1
      CALL FRDFL(LUN,IBUF,2*144,IABC,IE)
      IF(IE.EQ.0) GO TO 10
      ISTAT = 0
      CALL IOERR('ATLAS',LUN,IE)
```

```
      TYPE 'RECORD NUMBER',IRNO
      GO TO 20
   10 ISTAT = 1
   20 RETURN
      END
      PARAMETER LUNP = 11
      SUBROUTINE IOERR(NAME,ISLOT,IE)
      WRITE(LUNP,9000)IE,NAME,ISLOT
 9000 FORMAT(' *****I/O ERROR',OI5,' ON FILE ',S16,' SLOT NUMBER ',I2,
     1*****')
      RETURN
      END
```

```
      FUNCTION XNTF(ZF,ZA,TA,NOPTS)
C*****PURPOSE* THIS FUNCTION INTERPOLATES THE VALUE OF SOME OTHER
C*****   PARAMETER FOR A GIVEN VALUE OF DEPTH
C*****ARGUMENTS*ZF-DEPTH VALUE TO BE INTERPOLATED FOR,FLOATING-IN
C*****            ZA-DEPTH ARRAY,FLOATING-IN
      DIMENSION ZA(1),TA(1)
      DO 10 II = 2,NOPTS
        I=II
      IF(ZA(I)-ZF)10,20,30
   10 CONTINUE
   20 XNTF = TA(I)
      GO TO 40
   30 XNTF = TA(I-1)+((TA(I)-TA(I-1))*(ZF-ZA(I-1)))/(ZA(I)-ZA(I-1))
   40 RETURN
      END
```

))

```
C        ************************************************
C        ************************************************
C        ******                                    ******
C        ******      INPUT PROGRAM SECOND SEGMENT  ******
C        ******                                    ******
C        ************************************************
C        ************************************************
C
C
C        THIS SUBROUTINE PROVIDES THE INPUT MECHANISM
C
C        FOR ALL DATA REQUIRED TO RUN THE TASSRAP PROGRAM
C
         COMMON/TEMP/XLAT,XLON,JSEA
         COMMON/TERM/NXO,NYO,NPX,NPY
         COMMON/PLO/MODEX,IXX,IYX
         COMMON/ATLIO/IRNC
         COMMON/XXXEC/IXXEC
         COMMON/ARCH/IENU,IC0(41)
         COMMON/XDATA/LABEL(10),ITIME,IDATE(3),LAT,INS,LON,IEW,
     1RANGE,WH,BOTZ,SS,WS,IB,ITGT,ITOM,IST,ISONAR,FREQ(2,5),
     2INUMFRQ,TGTDEP,TGTSPD,TGTBBN,TOWDP(5),INUMDPS,DSC,IPROF,SLD,DM
         COMMON/ENU/Z(50),T(50),S(50),Z0(31),TOB(31),ZM(50),TM(50),
     1SM(50),UM(50),DEP(31),TEMP(31),NOPTS,NDP,NOPTM,MOE,SHPDEN
         COMMON/NOISE/NB,NF1,IBEAM(24),FREQN(24,5),LEVELN(24,5)
         DIMENSION JI(10),XTEMP(10),XDEP(10),XF(10),D(50),TAB(600)
         REAL LFRQLM,LEVELN
         CALL FRSFL
         IC=0
         CALL CHAR(31)
         LUNAT=0
         LUNIM=1
         LUNOS=2
         LUNSN=3
         CALL CHAR(31)
```

```
CB        ACCEPT "INPUT OWN RECUR. DEPTHS?  1=YES 0=NO---",IANS
CB        IF(IANS.EQ.1) GO TO 1000
CB        IND=1
CB        CALL HOLD
CB        GO TO 1099
CB1000    IND=0
CB        ACCEPT "UNITS OF DATA,  1=METRIC, 2=ENGLISH---",IMOE
CB1072    ACCEPT "NUMBER OF RECUR. DEPTHS = ",INUMDPS
CB        IF(INUMDPS.LE.5) GO TO 1076
CB        TYPE "***MAX. # OF DEPTHS = 5***"
CB        GO TO 1072
CB1076    CALL HOLD
CB        TYPE
CB        CALL CHAR(14)
CB        TYPE "                          ***** RECEIVER DEPTH INPUT *****"
CB        CALL CHAR(31)
CB        TYPE
CB        ACCEPT "INPUT RECUR. DEPTHS","<15>",(TOWDP(I),"<15>",I=1,INUMDPS)
CB1070    CALL HOLD
CB        TYPE
CB        CALL CHAR(14)
CB        TYPE "                          ***** RECUEIVER DEPTH INPUT *****"
CB        CALL CHAR(31)
CB        TYPE
CB        DO 1077 I=1,INUMDPS
CB        TOWDP(I)=TOWDP(I)+.01
CB1077    CONTINUE
CB        WRITE(11,1002) (I,TOWDP(I),I=1,INUMDPS)
CB1002    FORMAT(" ***RECUR DEPTHS***",/,"  LINE    DEPTH",/,(I5,4X,F6.0))
CB        WRITE(11,1003) INUMDPS
CB1003    FORMAT("CHANGE ANY OF THE",I3)
CB        ACCEPT"RECUR. DEPTHS?  1=YES 0=NO---",IC1
CB        IF(IC1.NE.1) GO TO 1097
CB        ACCEPT "NUMBER OF DEPTHS TO BE CHANGED = ",INC1
CB        TYPE
```

```
CU        TYPE "INPUT---LINE *,DEPTH"
CB        DO 1086 I=1,INC1
CB        ACCEPT "J ",J,TOWDP(J)
CB1086    CONTINUE
CB        GO TO 1078
CB1097    IF(IMOE.EQ.2) GO TO 10985
CB        DO 1098 I=1,INUMDPS
CB        TOWDP(I)=TOWDP(I)*3.2808
CB1098    CONTINUE
CB10985 CALL HOLD
CB1099    NNDP=NDP+1
          NNDP=NDP+1
11000     CALL HOLD
          JLIN = IEW
        IEW = 'E'
        IF(JLIN.EQ.2)IEW = 'W'
        JLIN1 = INS
        INS = 'N'
        IF(JLIN1.EQ.2)INS = 'S'
          LUNP=11
          CALL CHAR(14)
          TYPE "                    ***** ENVIRONMENTAL PROFILE DATA *****"
          CALL CHAR(31)
        WRITE(LUNP,1100)LAT,INS,LON,IEW,IDATE
1100       FORMAT(23X,'LAT',I5,A1,' LON',I5,A1,' DATE',1X,3I2)
          XLAT=FLOAT(LAT/100)+FLOAT(MOD(LAT,100))/60.
          XLON=FLOAT(LON/100)+FLOAT(MOD(LON,100))/60.
          IF(INS.EQ."S") XLAT=-XLAT
          IF(IEW.EQ."W") XLON=-XLON
          JSEA=((IDATE(2)-1)/3)+1
C******TEST FOR PACIFIC OCEAN*****
          IF (((XLON.GE.-180.).AND.(XLON.LT.-95)).OR.
       1((XLON.GE.100.).AND.(XLON.LE.180)))GO TO 11100
          IF ((XLON.GE.-95.).AND.(XLON.LT.-75.).AND.(XLAT.LT.10.))
       1GO TO 11100
```

```
      IF ((XLON.GE.-95.).AND.(XLON.LY.-06.).AND.
     1(XLAT.LT.15.)) GO TO 11100
C00000 TEST FOR MED SEA*****
      IF ((XLON.GE.0.).AND.(XLON.LT.40.).AND.(XLAT.GE.30.)
     1.AND.(XLAT.LT.47.)) GO TO 11300
      IF ((XLON.GE.-5.).AND.(XLAT.GE.34.).AND.(XLAT.LT.40.))
     1 GO TO 11300
C***** TEST FOR ATLANTIC OCEAN*****
C        THIS TEST FOLLOWS FROM THE TWO ABOVE
      IF((XLON.GE.-55.).AND.(XLON.LT.10.).AND.
     1(XLAT.GE.0.).AND.(XLAT.LT.70.)) GO TO 11500
C***** TEST FOR INDIAN OCEAN *****
C        THIS TEST FOLLOWS FROM ALL OF THE ABOVE
      IF ((XLON.GE.32.).AND.(XLON.LT.100.).AND.
     1(XLAT.GE.-10.).AND.(XLAT.LT.30.)) GO TO 11700
11005 TYPE"NO DATA FILES FOR LATITUDE AND LONGITUDE".
      TYPE" ENTERED"
      ACCEPT"NEW LATITUDE VALUE =",LAT
      ACCEPT"NORTH = 1 OR SOUTH = 2   ",INS
      ACCEPT"NEW LONGITUDE VALUE =",LON
      ACCEPT"EAST =1 OR WEST = 2   ",IEW
      CALL FRSFL
      GO TO 11000
C*****PACIFIC SUBAREA TEST  *****
11100 IF((XLAT.GE.65.).OR.(XLAT.LT.0.)) GO TO 11005
      CALL FOPFL("PACSHIP",LUNSN,IE)
      IF(IE.NE.0) CALL IOERR("PACSHIP",LUNSN,IE)
      ILIN=3
      IF(XLAT.LT.15.) GO TO 11130
      IF(XLAT.LT.30.) GO TO 11180
      IF(XLAT.LT.50.) GO TO 11230
C*****BY DEFAULT IT MUST BE PACE
      CALL FOPFL("PACEBLP",LUNOS,IE)
      IF(IE.NE.0) CALL IOERR("PACEBLP",LUNOS,IE)
      GO TO (11110,11115,11120,11125)JSEA
```

```
11110   CALL FOPFL("PACEWIN",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACEWIN",LUNAT,IE)
        GO TO 1102
11115   CALL FOPFL("PACESPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACESPR",LUNAT,IE)
        GO TO 1102
11120   CALL FOPFL("PACESUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACESUM",LUNAT,IE)
        GO TO 1102
11125   CALL FOPFL("PACEFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACEFAL",LUNAT,IE)
        GO TO 1102
C***** EITHER PACG OR PACF
11130   IF((XLON.GE.-170.).AND.(XLON.LT.-75.)) GO TO 11155
C***** PACG
        CALL FOPFL("PACGBLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("PACGBLP",LUNOS,IE)
        GO TO (11135,11140,11145,11150)JSEA
11135   CALL FOPFL("PACGWIN",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACGWIN",LUNAT,IE)
        GO TO 1102
11140   CALL FOPFL("PACGSPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACGSPR",LUNAT,IE)
        GO TO 1102
11145   CALL FOPFL("PACGSUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACGSUM",LUNAT,IE)
        GO TO 1102
11150   CALL FOPFL("PACGFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACGFAL",LUNAT,IE)
        GO TO 1102
C***** PACF
11155   CALL FOPFL("PACFBLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("PACFBLP",LUNOS,IE)
        GO TO (11160,11165,11170,11175)JSEA
11160   CALL FOPFL("PACFWIN",LUNAT,IE)
```

```
      IF(IE.NE.0) CALL IOERR("PACFWIN",LUNAT,IE)
      GO TO 1102
11165 CALL FOPFL("PACFSPR",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACFSPR",LUNAT,IE)
      GO TO 1102
11170 CALL FOPFL("PACFSUM",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACFSUM",LUNAT,IE)
      GO TO 1102
11175 CALL FOPFL("PACFFAL",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACFFAL",LUNAT,IE)
      GO TO 1102
C**** EITHER PACD OR PACB
11180 IF((XLON.GE.-100.).AND.(XLON.LT.-95)) GO TO 11205
C**** PACD
      CALL FOPFL("PACDBLP",LUNOS,IE)
      IF(IE.NE.0) CALL IOERR('PACDBLP",LUNOS,IE)
      GO TO (11185,11190,11195,11200)JSEA
11185 CALL FOPFL("PACDWIN",LUNAT,IE)
      IF(IE.NE.0)CALL IOERR("PACDWIN",LUNAT,IE)
      GO TO 1102
11190 CALL FOPFL("PACDSPR",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACDSPR",LUNAT,IE)
      GO TO 1102
11195 CALL FOPFL("PACDSUM",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACDSUM",LUNAT,IE)
      GO TO 1102
11200 CALL FOPFL("PACDFAL",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACDFAL",LUNAT,IE)
      GO TO 1102
C**** PACB
11205 CALL FOPFL("PACBBLP",LUNOS,IE)
      IF(IE.NE.0) CALL IOERR("PACBBLP",LUNOS,IE)
      GO TO (11210,11215,11220,11225)JSEA
11210 CALL FOPFL("PACBWIN",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACBWIN",LUNAT,IE)
```

```
      GO TO 1102
11215 CALL FOPFL("PACBSPR",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACBSPR",LUNAT,IE)
      GO TO 1102
11220 CALL FOPFL("PACBSUM",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACBSUM",LUNAT,IE)
      GO TO 1102
11225 CALL FOPFL("PACBFAL",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACBFAL",LUNAT,IE)
      GO TO 1102
C**** EITHER PACC OR PACA
11230 IF((XLON.GE.-160.).AND.(XLON.LT.-115)) GO TO 11255
C**** PACC
      CALL FOPFL("PACCBLP",LUNOS,IE)
      IF(IE.NE.0) CALL IOERR("PACCBLP",LUNOS,IE)
      GO TO (11235,11240,11245,11250)JSEA
11235 CALL FOPFL("PACCWIN",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACCWIN",LUNAT,IE)
      GO TO 1102
11240 CALL FOPFL("PACCSPR",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACCSPR",LUNAT,IE)
      GO TO 1102
11245 CALL FOPFL("PACCSUM",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACCSUM",LUNAT,IE)
      GO TO 1102
11250 CALL FOPFL("PACCFAL",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACCFAL",LUNAT,IE)
      GO TO 1102
C**** PACA
11255 CALL FOPFL("PACABLP",LUNOS,IE)
      IF(IE.NE.0) CALL IOERR("PACABLP",LUNOS,IE)
      GO TO (11260,11265,11270,11275)JSEA
11260 CALL FOPFL("PACAWIN",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("PACAWIN",LUNAT,IE)
      GO TO 1102
```

```
11265   CALL FOPFL("PACASPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACASPR",LUNAT,IE)
        GO TO 1102
11270   CALL FOPFL("PACASUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACASUM",LUNAT,IE)
        GO TO 1102
11275   CALL FOPFL("PACAFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("PACAFAL",LUNAT,IE)
        GO TO 1102
C
C***** MED SEA *****
11300   CALL FOPFL("MEDSHIP",LUNSN,IE)
        IF (IE.NE.0) CALL IOERR("MEDSHIP",LUNSN,IE)
        ILIN=4
        CALL FOPFL("MEDBLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("MEDBLP",LUNOS,IE)
        GO TO (11305,11310,11315,11320)JSEA
11305   CALL FOPFL("MEDWIN",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("MEDWIN",LUNAT,IE)
        GO TO 1102
11310   CALL FOPFL("MEDSPR",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("MEDSPR",LUNAT,IE)
        GO TO 1102
11315   CALL FOPFL("MEDSUM",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("MEDSUM",LUNAT,IE)
        GO TO 1102
11320   CALL FOPFL("MEDFAL",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("MEDFAL",LUNAT,IE)
        GO TO 1102
C***** ATLANTIC OCEAN *****
11500   CALL FOPFL("ATLSHIP",LUNSN,IE)
        IF (IE.NE.0) CALL IOERR("ATLSHIP",LUNSN,IE)
        ILIN=3
        IF (XLAT.LT.10.) GO TO 11525
        IF (XLAT.LT.25.) GO TO 11550
```

```
      IF (XLAT.LT.40.) GO TO 11575
      IF (XLAT.LT.50.) GO TO 11600
C***** ATLC BY DEFAULT *****
      CALL FOPFL("ATLCBLP",LUNOS,IE)
      IF (IE.NE.0) CALL IOERR("ATLCBLP",LUNOS,IE)
      GO TO (11505,11510,11515,11520)JSEA
11505 CALL FOPFL("ATLCWIN",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLCWIN",LUNAT,IE)
      GO TO 1102
11510 CALL FOPFL("ATLCSPR",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLCSPR",LUNAT,IE)
      GO TO 1102
11515 CALL FOPFL("ATLCSUM",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLCSUM",LUNAT,IE)
      GO TO 1102
11520 CALL FOPFL("ATLCFAL",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLCFAL",LUNAT,IE)
      GO TO 1102
C***** ATLE *****
11525 CALL FOPFL("ATLEBLP",LUNOS,IE)
      IF (IE.NE.0) CALL IOERR("ATLEBLP",LUNOS,IE)
      GO TO (11530,11535,11540,11545)JSEA
11530 CALL FOPFL("ATLEWIN",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLEWIN",LUNAT,IE)
      GO TO 1102
11535 CALL FOPFL("ATLESPR",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLESPR",LUNAT,IE)
      GO TO 1102
11540 CALL FOPFL("ATLESUM",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLESUM",LUNAT,IE)
      GO TO 1102
11545 CALL FOPFL("ATLEFAL",LUNAT,IE)
      IF (IE.NE.0) CALL IOERR("ATLEFAL",LUNAT,IE)
      GO TO 1102
C***** ATLD *****
```

```
11550   CALL FOPFL("ATLDBLP",LUNOS,IE)
        IF (IE.NE.0) CALL IOERR("ATLDBLP",LUNOS,IE)
        GO TO (11555,11560,11565,11570)JSEA
11555   CALL FOPFL("ATLDWIN",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("ATLDWIN",LUNAT,IE)
        GO TO 1102
11560   CALL FOPFL("ATLDSPR",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("ATLDSPR",LUNAT,IE)
        GO TO 1102
11565   CALL FOPFL("ATLDSUM",LUNAT,IE)
        IF (IE.NE.0) CALL IOERR("ATLDSUM",LUNAT,IE)
        GO TO 1102
11570   CALL FOPFL("ATLDFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLDFAL",LUNAT,IE)
        GO TO 1102
C****ATLA
11575   CALL FOPFL("ATLABLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("ATLABLP",LUNOS,IE)
        GO TO(11580,11585,11590,11595)JSEA
11580   CALL FOPFL("ATLAWIN",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLAWIN",LUNAT,IE)
        GO TO 1102
11585   CALL FOPFL("ATLASPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLASPR",LUNAT,IE)
        GO TO 1102
11590   CALL FOPFL("ATLASUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLASUM",LUNAT,IE)
        GO TO 1102
11595   CALL FOPFL("ATLAFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLAFAL",LUNAT,IE)
        GO TO 1102
C****ATLB
11600   CALL FOPFL("ATLBBLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("ATLBBLP",LUNAT,IE)
        GO TO (11605,11610,11615,11620)JSEA
```

```
11605   CALL FOPFL("ATLBWIN",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLBWIN",LUNAT,IE)
        GO TO 1102
11610   CALL FOPFL("ATLBSPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLBSPR",LUNAT,IE)
        GO TO 1102
11615   CALL FOPFL("ATLBSUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLBSUM",LUNAT,IE)
        GO TO 1102
11620   CALL FOPFL("ATLBFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("ATLBFAL",LUNAT,IE)
        GO TO 1102
C00000 INDIAN OCEAN
11700   CALL FOPFL("INDSHIP",LUNSN,IE)
        IF(IE.NE.0) CALL IOERR("INDSHIP",LUNSN,IE)
        ILIN=3
        IF(XLAT.GE.10) GO TO 11725
C INDA BY DEFAULT
        CALL FOPFL("INDABLP",LUNOS,IE)
        IF(IE.NE.0) CALL IOERR("INDABLP",LUNOS,IE)
        GO TO (11705,11710,11715,11720)JSEA
11705   CALL FOPFL("INDAWIN",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("INDAWIN",LUNAT,IE)
        GO TO 1102
11710   CALL FOPFL("INDASPR",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("INDASPR",LUNAT,IE)
        GO TO 1102
11715   CALL FOPFL("INDASUM",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("INDASUM",LUNAT,IE)
        GO TO 1102
11720   CALL FOPFL("INDAFAL",LUNAT,IE)
        IF(IE.NE.0) CALL IOERR("INDAFAL",LUNAT,IE)
        GO TO 1102
C00000INDB
11725   CALL FOPFL("INDBBLP",LUNOS,IE)
```

```
      IF(IE.NE.0) CALL IOERR("INDDBLP",LUNOS,IE)
      GO TO (11730,11735,11740,11745)JSEA
11730 CALL FOPFL("INDBWIN",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("INDBWIN",LUNAT,IE)
      GO TO 1102
11735 CALL FOPFL("INDBSPR",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("INDBSPR",LUNAT,IE)
      GO TO 1102
11740 CALL FOPFL("INDBSUM",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("INDBSUM",LUNAT,IE)
      GO TO 1102
11745 CALL FOPFL("INDBFAL",LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("INDBFAL",LUNAT,IE)
      GO TO 1102
1102  IRNO=0
      CALL GETENU(LUNOS,1,ISTAT)
      CALL FCLFL(LUNOS,IE)
      IF(IE.NE.0) CALL IOERR("ROUGH",LUNOS,IE)
      IF(ISTAT.EQ.2) GO TO 11005
      IF(ISTAT.EQ.3) GO TO 11005
      IF(ISTAT.NE.1) GO TO 220
      ILFBLP=IB/16
      IHFBLP=IB-(16*ILFBLP)
      IRNO=0
      CALL GETENU(LUNAT,2,ISTAT)
      CALL FCLFL(LUNAT,IE)
      IF(IE.NE.0) CALL IOERR("ATLAS",LUNAT,IE)
      IF((ISTAT.EQ.2).OR.(ISTAT.EQ.3)) GO TO 11005
      IF(ISTAT.NE.1) GO TO 220
      IRNO=0
      CALL GETENU(LUNSH,ILIN,ISTAT)
      CALL FCLFL(LUNSH,IE)
      IF(IE.NE.0) CALL IOERR("SHIPPING",LUNSH,IE)
      IF((ISTAT.EQ.2).OR.(ISTAT.EQ.3)) GO TO 11005
      IF(ISTAT.EQ.7) GO TO 1103
```

```
      IF(ISTAT.NE.1) GO TO 220
      GO TO 1104
1103  CALL HOLD
      WRITE(LUNP,1100)LAT,INS,LON,IEW,IDATE
1104  WRITE(LUNP,1105) IHFBLP,ILFBLP,SHPDEN
1105  FORMAT(7X,"H. FREQ BLP ",I2,4X,"L. FREQ BLP ",I2,4X,"SHIP. DEN. "
      GO TO (110,80)IPROF
80    DO 90 I=1,NOPTS
      IF(Z(I) .LT. BOTZ)GO TO 90
      NOPTS = I-1
      IF(Z(I) .NE. BOTZ)GO TO 100
      NOPTS = NOPTS + 1
      GO TO 43
90    CONTINUE
100   CALL XNTERP(Z,T,S,NOPTS,BOTZ)
43    NOPTM = NOPTS
      IPAGE=0
      WRITE(LUNP,1110)
1110  FORMAT(32X,'RETRIEVED DATA',/,25X,'DEP',4X,'TEMP',4X,'SAL',8X,
     &'VEL',/,25X,'(M)',5X,'(C)',3X,'(PPT)',3X,'(M/SEC)')
      DO 44 I=1,NOPTM
      ZM(I)=Z(I)
      TM(I)=T(I)
      SM(I)=S(I)
      VM(I)=WILSON(ZM(I),TM(I),SM(I))
      VM(I) = 0.1*AINT(10.*VM(I)+0.5)
44    CONTINUE
      CALL TWDPT(0)
      DO 51 I=1,NOPTM
      IPAGE=IPAGE+1
      IF(IPAGE.LT.31) GO TO 450
      CALL HOLD
      IPAGE=0
      CALL CHAR(14)
      TYPE "                    ***** ENVIRONMENTAL PROFILE DATA *****"
```

```
      CALL CHAR(31)
      WRITE(LUNP,1110)
400   IF(Z(I).EQ.SLD) GO TO 46
      IF(Z(I).EQ.DSC) GO TO 47
      WRITE (11,8010)(Z(I),T(I),S(I),VM(I))
8010  FORMAT(22X,F6.0,2F8.2,2X,F8.2)
      GO TO 51
46    WRITE(11,8011) (Z(I),T(I),S(I),VM(I))
8011  FORMAT(18X,"SLD>",F6.0,2F8.2,2X,F8.2)
      GO TO 51
47    WRITE(11,8012) (Z(I),T(I),S(I),VM(I))
8012  FORMAT(18X,"DSC>",F6.0,2F8.2,2X,F8.2)
51    CONTINUE
      JSTAT=1
      GO TO 200
110   WRITE(LUNP,1130)
1130  FORMAT (6X,"BT DATA",11X,"RETRIEVED DATA",21X,
     @"MERGED DATA")
      WRITE(LUNP,1140)
1140  FORMAT (4X,3HDEP,4X,4HTEMP,7X,3HDEP,4X,4HTEMP,
     @4X,3HSAL,12X,3HDEP,4X,4HTEMP,4X,3HSAL,4X,3HVEL)
      IPAGE=0
      IF(MOE.EQ.1) GO TO 130
      WRITE(LUNP,1150)
1150  FORMAT (3X,4H(FT),5X,3H(F),7X,3H(M),5X,3H(C),3X,5H(PPT),
     @11X,3H(M),5X,3H(C),3X,5H(PPT),3X,7H(M/SEC))
      FTMT=.3048
      FRCENT=.5555556
      DO 120 I=1,NDP
      ZO(I)=DEP(I)*FTMT
      TOB(I)=(TEMP(I)-32.0)*FRCENT
120   CONTINUE
      BOTZ=BOTZ*FTMT
      ZO(NNDP)=0.0
      GO TO 150
```

```
130     CONTINUE
        ZO(NNDP)=0.0
        DO 140 I=1,NDP
        ZO(I)=DEP(I)
        TOB(I)=TEMP(I)
140     CONTINUE
        WRITE(LUNP,1160)
1160    FORMAT(4X,3H(M),5X,3H(C),7X,3H(M),5X,3H(C),3X,
     *5H(PPT),11X,3H(M),5X,3H(C),3X,5H(PPT),3X,7H(M/SEC))
150     CALL MERGE(BOTZ,JSTAT)
        IF (JSTAT.NE.1) GO TO 210
        DO 160 I=1,NOPTM
        VM(I)=WILSON(ZM(I),TM(I),SM(I))
160     CONTINUE
        CALL TWDPT(0)
        DO 165 I=1,NDP
        IPAGE=IPAGE+1
        IF(IPAGE.LT.30) GO TO 9160
        CALL HOLD
        IPAGE=0
        CALL CHAR(14)
        TYPE "                    ***** ENVIRONMENTAL PROFILE DATA *****"
        CALL CHAR(31)
        WRITE(LUNP,1130)
        WRITE(LUNP,1140)
        IF(MOE.EQ.1) GO TO 9161
        WRITE(LUNP,1150)
        GO TO 9160
9161    WRITE(LUNP,1160)
9160    IF(ZM(I).EQ.SLD) GO TO 161
        IF(ZM(I).EQ.DSC) GO TO 162
        WRITE(LUNP,1170)(DEP(I),TEMP(I),Z(I),T(I),S(I),ZM(I),
     *TM(I),SM(I),VM(I))
1170    FORMAT(F7.0,F8.2,3X,F7.0,2F8.2,7X,F6.0,2F8.2,2X,F8.2)
        GO TO 165
```

```
161     WRITE(LUNP,1171) (DEP(I),TEMP(I),Z(I),T(I),S(I),ZM(I),
       0TM(I),SM(I),UM(I))
1171    FORMAT(F7.0,F8.2,3X,F7.0,2F8.2,3X,"SLD>",F6.0,2F8.2,2X,F8.2)
        GO TO 165
162     WRITE(LUNP,1172) (DEP(I),TEMP(I),Z(I),T(I),S(I),ZM(I),
       *TM(I),SM(I),UM(I))
1172    FORMAT(F7.0,F8.2,3X,F7.0,2F8.2,3X,"DSC>",F6.0,2F8.2,2X,F8.2)
165     CONTINUE
        NZP=NDP+1
        IF(NOPTS-NOPTM)190,180,170
170     DO 175 I=NZP,NOPTM
        IPAGE=IPAGE+1
        IF(IPAGE.LT.30) GO TO 9170
        CALL HOLD
        IPAGE=0
        CALL CHAR(14)
        TYPE "                      ***** ENVIRONMENTAL PROFILE DATA *****"
        CALL CHAR(31)
        WRITE(LUNP,1130)
        WRITE(LUNP,1140)
        IF(MOE.EQ.1) GO TO 9171
        WRITE(LUNP,1150)
        GO TO 9170
9171    WRITE(LUNP,1160)
9170    IF(ZM(I).EQ.SLD) GO TO 171
        IF(ZM(I).EQ.DSC) GO TO 172
        WRITE(LUNP,1190)(Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
        GO TO 175
171     WRITE(LUNP,1191) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
        GO TO 175
172     WRITE(LUNP,1192) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
175     CONTINUE
        NZP=NOPTM+1
        WRITE(LUNP,1100)(Z(I),T(I),S(I),I=NZP,NOPTS)
1100    FORMAT(10X,F7.0,2F8.2)
```

```
      GO TO 200
180   DO 185 I=NZP,NOPTS
      IPAGE=IPAGE+1
      IF(IPAGE.LT.30) GO TO 9180
      CALL HOLD
      IPAGE=0
      CALL CHAR(14)
      TYPE "                         ***** ENVIRONMENTAL PROFILE DATA *****"
      CALL CHAR(31)
      WRITE(LUNP,1130)
      WRITE(LUNP,1140)
      IF(MOE.EQ.1) GO TO 9181
      WRITE(LUNP,1150)
      GO TO 9180
9181  WRITE(LUNP,1160)
9180  IF(ZM(I).EQ.SLD) GO TO 181
      IF(ZM(I).EQ.DSC) GO TO 182
      WRITE(LUNP,1190)(Z(I),T(I),S(I),ZM(I),TM(I),SM(I),VM(I))
1190  FORMAT(10X,F7.0,2F8.2,7X,F6.0,2F8.2,2X,F8.2)
      GO TO 185
181   WRITE(LUNP,1191) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),VM(I))
1191  FORMAT(10X,F7.0,2F8.2,3X,"SLD>",F6.0,2F8.2,2X,F8.2)
      GO TO 185
182   WRITE(LUNP,1192) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),VM(I))
1192  FORMAT(10X,F7.0,2F8.2,3X,"DSC>",F6.0,2F8.2,2X,F8.2)
185   CONTINUE
      GO TO 200
190   DO 195 I=NZP,NOPTS
      IPAGE=IPAGE+1
      IF(IPAGE.LT.30) GO TO 9190
      CALL HOLD
      IPAGE=0
      CALL CHAR(14)
      TYPE "                         ***** ENVIRONMENTAL PROFILE DATA *****"
      CALL CHAR(31)
```

```
          WRITE(LUNP,1130)
          WRITE(LUNP,1140)
          IF(MOE.EQ.1) GO TO 9191
          WRITE(LUNP,1150)
          GO TO 9190
9191      WRITE(LUNP,1160)
9190      IF(ZM(I).EQ.SLD) GO TO 191
          IF(ZM(I).EQ.DSC) GO TO 192
          WRITE(LUNP,1190)(Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
          GO TO 195
191       WRITE(LUNP,1191) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
          GO TO 195
192       WRITE(LUNP,1192) (Z(I),T(I),S(I),ZM(I),TM(I),SM(I),UM(I))
195       CONTINUE
          NZP=NOPTS+1
          DO 199 I=NZP,NOPTM
          IPAGE=IPAGE+1
          IF(IPAGE.LT.30) GO TO 9195
          CALL HOLD
          IPAGE=0
          CALL CHAR(14)
          TYPE "                    ***** ENVIRONMENTAL PROFILE DATA *****"
          CALL CHAR(31)
          WRITE(LUNP,1130)
          WRITE(LUNP,1140)
          IF(MOE.EQ.1) GO TO 9196
          WRITE(LUNP,1150)
          GO TO 9195
9196      WRITE(LUNP,1160)
9195      IF(ZM(I).EQ.SLD) GO TO 196
          IF(ZM(I).EQ.DSC) GO TO 197
          WRITE(LUNP,1200)(ZM(I),TM(I),SM(I),UM(I))
1200      FORMAT(48X,F6.0,2F8.2,2X,F8.2)
          GO TO 199
196       WRITE(LUNP,1201) (ZM(I),TM(I),SM(I),UM(I))
```

```
1201    FORMAT(44X,"SLD>",F6.0,2F8.2,2X,F8.2)
        GO TO 199
107     WRITE(LUNP,1202) (ZM(I),TM(I),SM(I),UM(I))
1202    FORMAT(44X,"DSC>",F6.0,2F8.2,2X,F8.2)
199     CONTINUE
200     CONTINUE
        WRITE(LUNP,1210)
1210    FORMAT("***** PROFILE COMPLETE *****")
        GO TO 230
210     WRITE(LUNP,1220)JSTAT
1220    FORMAT(" MERGE FAILURE CHECK STATUS INDICATOR STATUS=",I2)
        GO TO 230
220     WRITE(LUNP,1230)ISTAT
1230    FORMAT("RETRIEVAL FAILURE CHECK STATUS INDICATOR
     * STATUS A = "I2)
230     CONTINUE
        CALL FRSFL
C       SOUND VELOCITY PROFILE GRAPHIC PROGRAM
C*********PRINT HEADER
500     CALL HOLD
        ACCEPT "OUTPUT TEMP. PROFILE?  1=YES 0=NO---",IQ6
        IF((IQ6.NE.0).AND.(IQ6.NE.1)) WRITE(11,501)
501     FORMAT(//,5X,"*****YOUR ENTRY IS INVALID*****",/,
     17X,"***HIT SPACE BAR TO CONTINUE***")
        IF((IQ6.NE.0).AND.(IQ6.NE.1)) GO TO 500
        IF(IQ6.NE.1) GO TO 5010
        CALL CHAR(12)
        CALL PFGRAPH
502     CALL HOLD
5010    ACCEPT "OUTPUT SUP?  1=YES 0=NO---",IQ
        IF((IQ.NE.0).AND.(IQ.NE.1)) WRITE(11,501)
        IF((IQ.NE.0).AND.(IQ.NE.1)) GO TO 502
        IF(IQ.NE.1) GO TO 5999
        ACCEPT "UNITS OF DATA, 1=METRIC, 2=ENGLISH---",IQ1
        IF((IQ1.NE.1).AND.(IQ1.NE.2)) WRITE(11,501)
```

```
      IF((IQ1.NE.1).AND.(IQ1.NE.2)) GO TO 6[illegible]
      GO TO (5025,5015)IQ1
5015  FTMT=.3048
      DO 5020 I=1,NOPTM
      ZM(I)=ZM(I)/FTMT
      UM(I)=UM(I)/FTMT
      TM(I)=(TM(I)*1.8)+32.
5020  CONTINUE
5025  CALL HOLD
      WRITE(LUNP,6011) IDATE,LAT,INS,LON,IEW
6011  FORMAT(1X,'DATE ',2(I2"/"),I2,/,1X,'LAT'I5,A1,/,1X,'LON'I5,A1)
      CALL HOME
      GO TO (5100,5110)IQ1
5100  WRITE(LUNP,14010)
14010 FORMAT(31X,'VELOCITY  M/SEC',///,66X,
     *'(NEAR SURFACE)',/,66X,'VELOCITY  M/SEC')
      GO TO 5120
5110  WRITE(LUNP,14020)
14020 FORMAT(31X,'VELOCITY FT/SEC',///,66X,
     *'(NEAR SURFACE)',/,66X,'VELOCITY FT/SEC')
5120  CONTINUE
C     GET GRAPH LIMITS
      XMIN=UM(1)
      XMAX=XMIN
      DO 5030 I=2,NOPTM
      XMIN=AMIN1(XMIN,UM(I))
      XMAX=AMAX1(XMAX,UM(I))
5030  CONTINUE
      XLO=AINT(XMIN)
      XHI=AINT(XMAX)
      XMDUL = IQ1*20.
      XLO = AINT((XLO/XMDUL)+.5)*XMDUL
      IF(XLO.GT.XMIN)XLO = XLO-XMDUL
      XHI = AINT((XHI/XMDUL)+.5)*XMDUL
      IF(XHI.LT.XMAX) XHI = XHI+XMDUL
```

```
      NX=INT((XHI-XLO)/XMDUL)
C
C***** Q DEFINED BELOW IS A DUMMY CHARACTER ENABLING
C***** THIS PROGRAM TO PLOT EITHER METRIC OR ENGLISH
C***** UNITS FOR AN SUP
C
      IF(IQ1.NE.1)GO TO 5035
      Q=2.
      GO TO 5036
5035  Q=5.
5036  Q1=Q/2.
      YLO=0.
      YHI = AINT(ZM(NOPTM))
      YHI = AINT((YHI/(Q*100.))+.5)*(Q*100.)
      IF(YHI.LT.ZM(NOPTM))YHI = YHI+(Q*100.)
      NY=INT(YHI/(Q*100.))
      WRITE(LUNP,14050)
14050 FORMAT(//)
      CALL XFSET(XLO,YLO,XHI,YHI,252,695,662,30,XF)
      CALL GRID(XF,XMDUL/2.,(Q1*100.),2*NX,2*NY)
      CALL GLABEL(INT(XLO),INT(XMDUL),NX+1,1,XF)
      CALL GLABEL(INT(YLO),INT(Q*100.),NY+1,4,XF)
      IF(IQ1.NE.1) GO TO 5038
      CALL SYMBOL(184,444,' D E P T H   M  ',8,1)
      GO TO 5039
5038  CALL SYMBOL(184,444,' D E P T H   F T',8,1)
5039  CONTINUE
      CALL GBOT(XF,ZM(NOPTM))
      CALL PLOT(VM(1),ZM(1),XF,0)
      DO 5040 I=2,NOPTM
      CALL PLOT(VM(I),ZM(I),XF,1)
5040  CONTINUE
C     COMPUTE LIMITS FOR NEAR SURFACE PORTION
      DMAXI=300.
      IF(IQ1.NE.1)DMAXI=1000.
```

```
          XMIN=UM(1)
          XMAX=XMIN
          D(1)=ZM(1)
          S(1)=UM(1)
          DO 5050 I=2,NOPTM
          D(I)=ZM(I)
          S(I)=UM(I)
          K=I
          KK=K
          XMIN=AMIN1(XMIN,UM(I))
          XMAX=AMAX1(XMAX,UM(I))
          IF(ZM(K).EQ.DMAXI)GO TO 5070
          IF (ZM(K).GT.DMAXI)GO TO 5060
5050      CONTINUE
          GO TO 5070
5060      D(KK)=DMAXI
          S(KK)=((DMAXI-ZM(K-1))/(ZM(K)-ZM(K-1)))*
        *(UM(K)-UM(K-1))+UM(K-1)
          XMIN=AMIN1(XMIN,S(KK))
          XMAX=AMAX1(XMAX,S(KK))
5070      XLO=AINT(XMIN)
          XHI=AINT(XMAX)
          XMDUL = IQ1*20.
          XLO = AINT((XLO/XMDUL)+.5)*XMDUL
          IF(XLO.GT.XMIN)XLO = XLO-XMDUL
          XHI = AINT((XHI/XMDUL)+.5)*XMDUL
          IF(XHI.LT.XMAX)XHI = XHI+XMDUL
          NX=INT((XHI-XLO)/XMDUL)
          YLO=0.
          YHI=DMAXI
          YHI = AINT((YHI/(Q*10.))+.5)*(Q*10.)
          IF(YHI.LT.DMAXI)YHI = YHI+(Q*10.)
          NY=INT(YHI/(Q*10.))
          CALL XFSET(XLO,YLO,XHI,YHI,720,600,995,120,XF)
          IF(IQ1.NE.1)GO TO 5077
```

```
      CALL GRID(XF,XMDUL/2.,10.,2*NX,2*NY)
      GO TO 5070
5077  CALL GRID(XF,XMDUL/2.,50.,2*NX,NY)
5070  CONTINUE
      CALL GLABEL(INT(XLO),INT(XMDUL),NX+1,1,XF)
      CALL GLABEL(INT(YLO),INT(Q*10.),NY+1,4,XF)
      IF(IQ1.NE.1)GO TO 5072
      CALL SYMBOL(670,444,' D E P T H   M  ',8,1)
      GO TO 5073
5072  CALL SYMBOL(670,444,'D E P T H   F T',8,1)
5073  CONTINUE
      CALL PLOT(S(1),D(1),XF,0)
      DO 5080 I=1,KK
      CALL PLOT(S(I),D(I),XF,1)
5080  CONTINUE
      CALL TPLOT(0,0,21)
      CALL CHAR(14)
      TYPE "  ***** TASSRAP SUP GRAPHIC *****"
      CALL CHAR(31)
      CALL HOLD
5999  CONTINUE
      IF (IQ1.EQ.2) GO TO 6002
      DO 6001 I=1,NOPTM
      ZM(I)=ZM(I)*3.2808
      UM(I)=UM(I)*3.2808
      TM(I)=(TM(I)*1.8)+32
6001  CONTINUE
6002  SLD=SLD*3.2808
      DSC=DSC*3.2808
      INS=JLIN1
      IEW=JLIN
      CALL CHAR(31)
      CALL FOPFL("Z999TASS:IM",LUNIM,IE)
      IF(IE.NE.0) CALL IOERR("Z999TASS:IM",LUNIM,IE)
      REWIND LUNIM
```

```
      WRITE BINARY(LUNIM) IDATE,LAT,INS,LON,IEW,NOPTM,
     *(ZM(I),TM(I),SM(I),VM(I),I=1,NOPTM)
      CALL FCLFL(LUNIM,IE)
      IF(IE.NE.0) CALL IOERR("Z999TASS:IM",LUNIM,IE)
      CALL FRNOV("EXECP",IE)
      IF(IE.NE.0) TYPE"INPUTOV OVERLAY ERROR=",IE
      STOP
      END
```

```
      SUBROUTINE GETENV(LUN,IBOT,JSTAT)
C*****PURPOSE* THIS IS THE HISTORICAL FILE DATA RETRIEVAL SUBROUTINE
C*****USES* TR720,TRWND,MOVFR,MOVBR
C*****ARGUMENTS*LAT-LATITUDE IN DEGREES AND MINUTES,I4,FIXED-IN
C*****          INS-LATITUDE INDICATOR ,N-NORTH,S-SOUTH,A1,ALPHA-IN
C*****          LON-LONGITUDE IN DEGREES AND MINUTES,I5,FIXED-IN
C*****          IEW-LONGITUDE INDICATOR,E-EAST,W-WEST,A1,ALPHA-IN
C*****          IDATE-DATE, DAMOYR,I6,FIXED-IN
C*****          NOPTS-NUMBER OF TRACE POINTS, FIXED-OUT
C*****          Z-ARRAY OF DEPTH POINTS, METERS,FLOATING-OUT
C*****          T-ARRAY OF TEMPERATURES, DEGREES CENTIGRADE,FLOATING-O
C*****          S-ARRAY OF SALINITIES,PARTS PER THOUSAND,FLOATING-OUT
C*****          JSTAT-STATUS RETURN WORD
C*****              1-RETRIEVAL OK
C*****              2-DATA NOT ON FILE TAPE
C*****              3-LAND AREA-DATA NOT AVAILABLE
C*****              4-HARDWARE FAILURE-CHECK NOPTS FOR ADDITIONAL STAT
C*****              5-DATAFILE FAILURE-CHECK NOPTS FOR ADDITIONAL STAT
C*****              6-CALL LIST PARAMETER BAD
C*****REMARKS* WHEN JSTAT IS NOT 1, NOPTS CONTAINS ADDITIONAL STATUS
C*****                INFORMATION
      COMMON/TEMP/XLAT,XLON,ISEA
      COMMON/RTRV/LIMAV,LOCAV,EOFBF,IBUF(145),ILOC(1420)
      COMMON/XXXEC/IXXEC
      COMMON/ARCH/IENV,IC0(41)
      COMMON/XDATA/IBOB(11),IDATE(3),LAT,INS,LON,IEW,SAM(2),BOTZ,
     *ELAM(2),IB,
     1ISAM(4),BOB(10),INUMFRQ,FLECK(8),IDM,DM,IPROF,SLD,DMAX
      COMMON/ENV/Z(50),T(50),S(50),RAY(324),NOPTS,IRAY(2),MOE,SHPDEN
      COMMON/NOISE/NB,NF1,IBEAM(24),FREQN(24,5),LEVELN(24,5)
      REAL LEVELN
      LOGICAL LIMAV,LOCAV,EOFBF
      LIMAV=.FALSE.
      LOCAV=.FALSE.
      EOFBF=.FALSE.
      DO 10 I=1,145
      IBUF(I)=0
10    CONTINUE
```

```
      DO 11 I=1,1420
      ILOC(I)=0
 11   CONTINUE
C00HAS A LIMIT BLOCK BEEN READ? NO,READ ONE-YES,PROCEED
      IF(LIMAV)GO TO 20
      CALL TRWND(LUN)
C00INPUT LIMIT BLOCK
      CALL TR720(LUN,IBUF,ISTAT)
C00CHECK INPUT OF LIMIT BLOCK-OK,PROCEED-OTHERWISE ERROR
      IF(ISTAT .NE. 1)GO TO 980
      LOCAV = .FALSE.
      IF(IBUF(1).NE.21631K .AND. IBUF(1) .NE. 43163K) GO TO 981
C00DECODE LIMIT BLOCK
      IHCW =IBUF(2)
      ISEA  =IBUF(3)
      XLATMN =FLOAT(IBUF(4))
      XLATMX =FLOAT(IBUF(5))
      XLONMN =FLOAT(IBUF(6))
      XLONMX=FLOAT(IBUF(7))
      NLBLK =IBUF(8)
      IMAX  =IBUF(9)
      JMAX  =IBUF(10)
      NDBLK =IBUF(11)
      LIMAV =.TRUE.
C00CHECK CALL LIST PARAMETERS FOR ERROR
20      IMO=IDATE(2)
C00CHECK AND ADJUST FOR DATELINE OVERLAP
        YLON = XLON
      IF((XLONMN.LT.-180.).AND.(XLON.GT.0.))YLON = XLON-360.
      IF((XLONMX.GT.180.).AND.(XLON.LT.0.))YLON = XLON+360.
C00DATA SOUGHT IN THIS FILE? NO,INPUT NEXT FILE LIMIT BLOCK-YES,PROCEE
      IF((XLAT.GE.XLATMN).AND.(XLAT.LE.XLATMX)
     1.AND.(YLON.GE.XLONMN).AND.(YLON.LE.XLONMX))GO TO 30
C00DATA OUTSIDE FILE AREA
      GO TO 985
C00LOCATION BLOCK AVAILABLE  NO,INPUT ONE-YES, PROCEED
   30 IF(LOCAV)GO TO 40
C00INPUT LOCATION BLOCKS
        IF (IBOT.EQ.3) IMAX=IFIX(FLOAT(IMAX)/5.+.99)
```

```
        IF (IBOT.EQ.3) JMAX=IFIX(FLOAT(JMAX)/5.+.59)
      NL =1
      N= 144
      DO 35 J=1,JMAX
      DO 34 I=1,IMAX
      IF(N .NE.144)GO TO 33
      CALL TR720(LUN,IBUF, ISTAT)
C**CHECK INPUT OF LOCATION BLOCK-OK,PROCEED-OTHERWISE ERROR
      IF(ISTAT .NE. 1)GO TO 986
      IF(IBUF(1) .NE. IHCW)GO TO 987
      IF(IBUF(2) .NE. NL)GO TO 988
      NL =NL+1
      N = 2
   33 N=N+1
      IL = (J-1)*IMAX+I
      ILOC(IL) = IBUF(N)
   34 CONTINUE
   35 CONTINUE
      LOCAV =.TRUE.
C**SET DATA RECORD POINTER TO 1
      IDPNTR = 1
C**CALCULATE BY POSITION AND ACCESS DATA RECORD NUMBER
40      IF (IBOT.EQ.3) GO TO 4010
      I=IFIX(YLON-XLONMN+1.)
      J=IFIX(XLAT-XLATMN+1.)
      GO TO 4020
4010    I=((IFIX (YLON-XLONMN))/5)+1
        J=((IFIX (XLAT-XLATMN))/5)+1
4020    IL = (J-1)*IMAX+I
      IREF = ILOC(IL)
C**IF DATA ON LAND-RETURN
        IF((IBOT.EQ.3).OR.(IBOT.EQ.4)) GO TO 400
      IF(IREF.EQ.0)GO TO 989
400     GO TO (401,402,403,404)IBOT
401     IB=IREF
        GO TO 994
402     CONTINUE
C**CALCULATE POSITION OF DATA RECORD
      IMOV= IREF -IDPNTR
```

```
      IF (IBOT.EQ.3) JMAX=IFIX(FLOAT(JMAX)/5.+.99)
     NL =1
     N= 144
     DO 35 J=1,JMAX
     DO 34 I=1,IMAX
     IF(N .NE.144)GO TO 33
     CALL TR720(LUN,IBUF, ISTAT)
C**CHECK INPUT OF LOCATION BLOCK-OK,PROCEED-OTHERWISE ERROR
     IF(ISTAT .NE. 1)GO TO 986
     IF(IBUF(1) .NE. IHCH)GO TO 987
     IF(IBUF(2) .NE. NL)GO TO 988
     NL =NL+1
     N = 2
  33 N=N+1
     IL = (J-1)*IMAX+I
     ILOC(IL) = IBUF(N)
  34 CONTINUE
  35 CONTINUE
     LOCAU =.TRUE.
C**SET DATA RECORD POINTER TO 1
     IDPNTR = 1
C**CALCULATE BY POSITION AND ACCESS DATA RECORD NUMBER
40      IF (IBOT.EQ.3) GO TO 4010
     I=IFIX(YLON-XLONMN+1.)
     J=IFIX(XLAT-XLATMN+1.)
     GO TO 4020
4010    I=((IFIX (YLON-XLONMN))/5)+1
        J=((IFIX (XLAT-XLATMN))/5)+1
4020    IL = (J-1)*IMAX+I
     IREF = ILOC(IL)
C**IF DATA ON LAND-RETURN
        IF((IBOT.EQ.3).OR.(IBOT.EQ.4)) GO TO 400
     IF(IREF.EQ.0)GO TO 989
400     GO TO (401,402,403,404)IBOT
401     IB=IREF
        GO TO 994
402     CONTINUE
C**CALCULATE POSITION OF DATA RECORD
     IMOV= IREF -IDPNTR
```

```
C**POSITION TAPE
      IF(IMOV)41,50,43
   41 IMOV =IABS(IMOV)
      CALL MOVBR(LUN,IMOV,ISTAT)
      IF(ISTAT .NE. 1)GO TO 990
      GO TO 50
   43 CALL MOVFR(LUN,IMOV,ISTAT)
      IF(ISTAT .NE. 1)GO TO 991
      GO TO 50
C**INPUT DATA RECORD
   50 CALL TR720(LUN,IBUF,ISTAT)
C**CHECK INPUT OF DATA RECORD-OK,PROCEED-OTHERWISE ERROR
      IF(ISTAT .NE. 1)GO TO 992
      IF(IBUF(1) .NE. IHCH)GO TO 993
      IF(IBUF(2) .NE. IREF)GO TO 993
      IF(IBUF(3) .NE. ISEA)GO TO 993
C**DECODE DATA RECORD-FORMAT DATA-RETURN
      NOPTS =IBUF(4)
      IDPNTR =IREF + 1
      Z(1)= 0.
      T(1)= FLOAT(IBUF(5))/10.
      S(1)= FLOAT(IBUF(6))/10.
      DO 55 I=2,NOPTS
      Z(I) = FLOAT(IBUF(I*3+1))
      T(I) = FLOAT(IBUF(I*3+2))/10.
      S(I) = FLOAT(IBUF(I*3+3))/10.
   55 CONTINUE
        IF(BOTZ.EQ.0.) BOTZ=Z(NOPTS)
        IF(MOE.EQ.2) BOTZ=BOTZ*3.2808
      GO TO 994
C       CALCULATE SHIPPING DENSITY
C       SHIPPING DENSITY FOR ATLANTIC,PACIFIC, AND INDIAN OCEANS
403     IF ((IREF.EQ.999).OR.(IREF.EQ.0)) GO TO 407
        SHPDEN=((FLOAT(IREF))/100.)/(300.*300.*COS(XLAT))
        GO TO 994
C       SHIPPING DENSITY FOR MEDITERRANEAN SEA
404     IF((IREF.EQ.999).OR.(IREF.EQ.0)) GO TO 407
        SHPDEN=((FLOAT(IREF))/100.)/(60.*60.*COS(XLAT))
        GO TO 994
```

```
407       TYPE"NO SHIPPING INFORMATION AVAILABLE"
          IREF=0.1
          IF (IBOT.EQ.3) SHIPDEN=((FLOAT(IREF))/100.)/(300.*300.*COS(XLA
          IF (IBOT.EQ.4) SHIPDEN=((FLOAT(IREF))/100.)/(60.*60.*COS(XLAT)
          TYPE"DEFAULT VALUE=",SHIPDEN
          TYPE"DO YOU WISH TO CHANGE THE DEFAULT"
          ACCEPT"VALUE-YES=1,NO=0",IC3
          IF (IC3.EQ.1) ACCEPT"SHIPPING DENSITY=",SHPDEN
          GO TO 994
C**SET STATUS WORDS
980       JSTAT=4
          GO TO 995
981       JSTAT=5
          GO TO 995
985       JSTAT = 2
          GO TO 995
906       JSTAT=4
          GO TO 995
907       JSTAT=5
          GO TO 995
908       JSTAT=5
          GO TO 995
C**LAND AREA
909       JSTAT = 3
          GO TO 996
990       JSTAT = 4
          GO TO 995
991       JSTAT = 4
          GO TO 995
992       JSTAT = 4
          GO TO 996
993       JSTAT = 5
          GO TO 995
C**NORMAL RETURN
994       JSTAT = 1
          IF((IC3.EQ.1).OR.(IC3.EQ.0)) JSTAT=7
          GO TO 996
995       LIMAV = .FALSE.
```

END

```
      SUBROUTINE MERGE(BOTZ,ISTAT)
C*****PURPOSE* THIS SUBROUTINE MERGES AN OBSERVED BT TRACE WITH A TRAC
C*****FROM AN HISTORICAL FILE
C*****USES* XNTF,XNTERP
C*****ARGUMENTS*ZO-DEPTH ARRAY OBSERVED,METERS,FLOATING-IN
C*****          TO-TEMPERATURE ARRAY OBSERVED,DEGREES C,FLOATING-IN
C*****          ZH-DEPTH ARRAY HISTORICAL,METERS,FLOATING-IN
C*****          TH-TEMPERATURE ARRAY HISTORICAL,DEGREES C,FLOATING-IN
C*****          SH-SALINITY ARRAY HISTORICAL,PARTS PER THOUSAND,FLOATIN
C*****          NOPTSH-NUMBER OF POINTS HISTORICAL TRACE ARRAYS,FIXED-I
C*****              3-OBSERVED TRACE DOES NOT EXTEND TO 300 METERS
C*****              4-OBSERVED TRACE DEEPER THAN HISTORY
C*****REMARKS* MCCAULEY MERGE TECHNIQUE USED WITH SMOOTH PARAMETER OF
C*****   THE ONLY DATA REQUIREMENTS ARE THAT THE UNITS OF ZO,ZH,AND BO
C*****   BE THE SAME AND THAT THE UNITS OF TO AND TH BE THE SAME
       COMMON/ENU/ZH(50),TH(50),SH(50),ZO(31),TOB(31),ZM(50),TM(50),S
     1            BOB2(112),NOPTH,IBOB2,NOPTM,IBOB3,SHPDEN
       NOPTO = 30
      IF(ZO(1))991,5,991
    5 DO 10 I=2,30
      IF(ZO(I))10,15,10
   15 NOPTO = I-1
      GO TO 20
   10 CONTINUE
20     NOPTM = NOPTO
      IF(ZO(NOPTO).LT.300.)GO TO 992
      DO 30 I=1,NOPTO
      ZM(I) =ZO(I)
      TM(I) =TOB(I)
   30 CONTINUE
      TBEL = TOB(NOPTO)-XNTF(ZO(NOPTO),ZH,TH,NOPTH)
      DO 35 I=1,NOPTH
      IF((ZH(I)-50.).LT.ZO(NOPTO)) GO TO 35
      NPOINT = I
      GO TO 39
   35 CONTINUE
      GO TO 994
   39 DO 40 I=NPOINT,NOPTH
```

```
      NOPTM = NOPTM+1
      TODEL = TDEL*.835
      TDEL  = TODEL
      ZM(NOPTM) = ZH(I)
      TM(NOPTM) = TH(I) + TODEL
      IF(ZM(NOPTM).GE.BOTZ) GO TO 50
   40 CONTINUE
   50 DO 60 I=1,NOPTM
      SM(I) = XNTF(ZM(I),ZH,SH,NOPTH)
   60 CONTINUE
      CALL XNTERP(ZM,TM,SM,NOPTM,BOTZ)
      ISTAT = 1
  999 RETURN
C**FIRST TRACE POINT NOT ZERO
  991 ISTAT = 2
      GO TO 999
C**SYNOPTIC TRACE NOT DEEP ENOUGH
  992 ISTAT = 3
      GO TO 999
C**SYNOPTIC TRACE DEEPER THAN HISTORY
  994 ISTAT = 4
      GO TO 999
      END
```

```
      SUBROUTINE PFGRAPH
C
C     ****************************************
C     ****** THIS SUBROUTINE PLOTS THREE B/T GRAPHS
C     ****************************************
C     ****************************************
C
      COMMON/PGGRAF/MDEP(12),NBTH(14),NMER(6),NTOT(13)
      COMMON/XXXEC/IXXEC
        COMMON/ARCH/IENV,IC0(41)
        COMMON/XDATA/LABEL(10),ITIME,IDATE(3),LAT,INS,LON,IEW,RANGE,
     1  WH,BOTZ,SS,WS,IB,ITGT,ITOM,IST,ISONAR,FREQ(2,5),
     2  INUMFRQ,TGTDEP,TGTSPD,TGTBBN,TOWDP(5),INUMDPS,DSC,IPROF,SLD,DM
        COMMON/ENV/ZHIS(50),THIS(50),SHIS(50),ZO(31),TOB(31),Z(50),T(5
     1  S(50),UM(50),BTDEP(31),BTEMP(31),NOPTS,HDP,KY,MOE,SHPDEN
        COMMON/NOISE/NB,NF1,IBEAM(24),FREQN(24,5),LEVELN(24,5)
        REAL LEVELN
      DIMENSION XF(10)
      DATA MDEP(1)/68,69,80,84,72,32,77,69,84,69,82,83/
      DATA NBTH(1)/66,84,32,66,78,68,32,72,73,83,84,79,82,89/
      DATA NMER(1)/77,69,82,71,69,68/
      DATA NTOT(1)/84,79,84,65,76,32,80,82,79,70,73,76,69/
        LUNP=11
C BT AND MERGED PROFILES
        CALL CHAR(14)
        TYPE "                    ***** ENVIRONMENTAL PROFILE DATA ***
        CALL CHAR(31)
        WRITE (LUNP,1000)LAT,INS,LON,IEW,IDATE
1000    FORMAT(23X,"LAT ",I5,A1," LON",I5,A1," DATE",3I2)
      TLO = 0.0
      TUP = 25.0
        TXINC = 2.5
      ITLO = 0
      ITUP = 25
        ITINC = 5
      TYLIM = 5600.
      TYINC = 400.
      LBINCT = 400
```

```
      ITOP = 640
      GO TO(10,400),IPROF
C     TEMP PROFILE 1
10      CONTINUE
      CALL XFSET(TLO,0.,TUP,1000.,95,ITOP,370,20,XF)
      CALL GRID(XF,TXINC,100.,10,10)
      CALL GLABEL(ITLO,ITINC, 6,1,XF)
      CALL GLABEL(0,100,11,4,XF)
      CALL SYMBOL(20,440,MDEP,12,1)
      CALL SYMBOL(150,ITOP+25,NBTH,14,0)
      CALL PLOT(THIS(1),ZHIS(1),XF,0)
      DO 306 I=2,NOPTS
      IF(ZHIS(I).LE.1000.) GO TO 305
        TEMP = XNTF(1000.,ZHIS,THIS,NOPTS)
        CALL PLOT(TEMP,1000.,XF,1)
        GO TO 307
  305 CALL PLOT(THIS(I),ZHIS(I),XF,1)
  306 CONTINUE
307     IF(MOE.EQ.1) GO TO 309
        DO 308 I=1,NDP
        BTEMP(I) = (BTEMP(I)-32.0)*.5555556
        BTDEP(I) = BTDEP(I)*.3048
308     CONTINUE
  309 CALL PLOT(BTEMP(1),BTDEP(1),XF,0)
      DO 311 I=2,NDP
      IF(BTDEP(I).LE.1000.)GO TO 310
        TEMP = XNTF(1000.,BTDEP,BTEMP,NDP)
        CALL PLOT(TEMP,1000.,XF,1)
        GO TO 312
  310 CALL PLOT(BTEMP(I),BTDEP(I),XF,1)
  311 CONTINUE
C     TEMP PROFILE 2
  312 CALL XFSET(TLO,0.,TUP,1000.,433,ITOP,685,20,XF)
      CALL GRID(XF,TXINC,100.,10,10)
      CALL GLABEL(ITLO,ITINC, 6,1,XF)
      CALL GLABEL(0,100,11,4,XF)
      CALL SYMBOL(520,ITOP+25,NMER,6,0)
      CALL PLOT(T(1),Z(1),XF,0)
      DO 316 I=2,KY
```

```
      IF(Z(I) .LE. 1000.) GO TO 315
        TEMP = XNTF(1000.,Z,T,KY)
        CALL PLOT(TEMP,1000.,XF,1)
        GO TO 317
  315 CALL PLOT(T(I),Z(I),XF,1)
  318 CONTINUE
C     TEMP PROFILE 3
317     CALL XFSET(TLO,0.,TUP,TYLIM,748,ITOP,1000,20,XF)
      CALL GRID(XF,TXINC,TYINC,10,14)
      CALL GLABEL(ITLO,ITINC, 8,1,XF)
      CALL GLABEL(0,LBINCT,15,4,XF)
      CALL GBOT(XF,BOTZ)
      CALL SYMBOL(800,ITOP+35,NTOT,13,0)
      CALL PLOT(T(1),Z(1),XF,0)
      DO 350 I=2,KY
      CALL PLOT(T(I),Z(I),XF,1)
  350   CONTINUE
        GO TO 450
400     CONTINUE
        WRITE(LUMP,2000)
2000    FORMAT(/,25X,'HISTORICAL TEMPERATURE PROFILE')
        CALL CHAR(31)
C   GET GRAPH LIMITS
        THIN=THIS(1)
        THAX=THIN
        DO 4030 I=2,NOPTS
        THIN=AMIN1(THIN,THIS(I))
        THAX=AMAX1(THAX,THIS(I))
4030    CONTINUE
        TLO=AINT(THIN)
        THI=AINT(THAX)
        XMDUL=.5
        TLO=AINT((TLO/XMDUL)+0.5)*XMDUL
        IF(TLO.GT.THIN) TLO=TLO-XMDUL
        THI=AINT((THI/XMDUL)+0.5)*XMDUL
        IF(THI.LT.THAX)THI=THI+XMDUL
        IF((TLO.GE.0.).AND.(THI.LE.35.)) TLO=0.0
        IF(THI.GT.35.) TLO=THI-35.
        IF(TLO.LT.0.) TLO=-5.
```

```
      YLO=0.
      YHI=AINT(Z(NOPTS))
      YHI=AINT((YHI/200.)+.5)*200.
      IF(YHI.LT.Z(NOPTM)) YHI=YHI+200.
      NY=INT(YHI/200.)
      IF(NY.LT.1) NY=1
      CALL XFSET(TLO,0.,TLO+35.,YHI,370,ITOP,740,20,XF)
      CALL GRID(XF,TXINC,100.,14,2*NY)
      CALL GLABEL(INT(TLO),ITINC,0,1,XF)
      CALL GLABEL(0,200,NY+1,4,XF)
      CALL SYMBOL(184,444," D E P T H   M  ",0,1)
      CALL GBOT(XF,BOTZ)
      CALL PLOT(THIS(1),ZHIS(1),XF,0)
      DO 410 I=2,NOPTS
      CALL PLOT(THIS(I),ZHIS(I),XF,1)
410   CONTINUE
450   CALL TPLOT(0,0,21)
      CALL CHAR(14)
      TYPE "**** BT DISPLAY COMPLETED ****"
      RETURN
      END

))
```

```
      SUBROUTINE TWDPT(IND)
      COMMON/XXXEC/IXXEC
      COMMON/ARCH/IENV,IC0(41)
      COMMON/XDATA/LABEL(10),ITIME,IDATE(3),LAT,INS,LON,IEW,
     & RANGE,WH,BOTZ,SS,WS,IB,ITGT,ITOM,IST,ISONAR,FREQ(2,5),
     & INUMFRQ,TGTDEP,TGTSPD,TGTBBN,TOWDP(5),INUMDPS,DSC,IPROF,SLD,DM
      COMMON/ENV/Z(50),T(50),S(50),ZO(31),TOB(31),ZM(50),TM(50),
     & SM(50),VM(50),DEP(31),TEMP(31),NOPTS,NDP,NOPTM,MOE,SHPDEN
      COMMON/NOISE/NB,NF1,IBEAM(24),FREQN(24,5),LEVELN(24,5)
      REAL LEVELN
      SLD=-1.0
      DSC=-1.0
      FTMT=3.2808
      VELSLD=VM(1)
      VELDSC=VM(1)
      NDPT=1
      DO 10 I=2,NOPTM
      IF(VM(I).GE.VELDSC) GO TO 10
      NDPT=I
      VELDSC=VM(I)
      DSC=ZM(I)
10    CONTINUE
      IF((VM(1)-VELDSC).GT..5) GO TO 11
      DSC=-1
      NDPT=1
11    IF(NDPT.LT.2) GO TO 101
      NDPT1=NDPT-1
      DO 100 I=2,NDPT1
      IF(VM(I).LT.VELSLD) GO TO 100
      VELSLD=VM(I)
      SLD=ZM(I)
100   CONTINUE
101   IF(NOPTM.LT.1) GO TO 120
      IF(SLD.EQ.-1.0) SLD=ZM(1)
      IF(DSC.EQ.-1.0) DSC=ZM(NOPTM)
      GO TO 130
120   DSC=0.
      SLD=0.
```

```
100     IF(IND.EQ.0) RETURN
        SLDFT=SLD*FTMT
        DSCFT=DSC*FTMT
        IF(SLDFT.LE.50.) TOWDP(1)=100.
        IF(SLDFT.GT.50..AND.SLDFT.LE.70.) TOWDP(1)=50.
        IF(SLDFT.GT.70.) TOWDP(1)=0.75*SLDFT
        IF(DMAX.GT.DSCFT) GO TO 200
        TOWDP(2)=0.5*DMAX
        TOWDP(3)=0.75*DMAX
        TOWDP(4)=DMAX
        RETURN
200     TOWDP(2)=0.5*DSCFT
        TOWDP(3)=DSCFT
        TOWDP(4)=DMAX
        RETURN
        END

>>
```

```
      REAL FUNCTION WILSON(Z,T,S)
      S35 = S-35.
      P=(.1025+2.5E-7*Z)*Z + 1.03
      SUP=(((-3.3603E-12*P+3.5216E-9)*P+1.0268E-5)*P+1.60272E-1)*P
      SUT=(((7.9851E-6*T-2.6045E-4)*T-4.4532E-2)*T+4.5721)*T
      SUS=((1.69202E-3)*S35+1.39799)*S35
      TP = T*P
      STP=((7.7711E-7*T-1.1244E-2)*T+(-1.2943E-7*P+7.7316E-5)*P+(1.579
     1*T+3.158E-8)*TP)*S35+((4.5283E-8*T+7.4812E-6)*T-1.8607E-4)*TP
     2+(-1.9646E-10*P+1.8563E-9*T-2.5294E-7)*TP*P
        WILSON = 1449.14 + SUP+SUT+SUS+STP
      RETURN
      END
```

```
      SUBROUTINE XNTERP(Z,T,S,NOPTS,ZBOT)
C*****PURPOSE*THIS SUBROUTINE EXTRAPOLATES VALUES OF TEMPERATURE AND
C*****    SALINITY FOR A BOTTOM DEPTH
C*****ARGUMENTS*Z-DEPTH ARRAY,FLOATING-IN AND OUT
C*****          T-TEMPERATURE ARRAY,FLOATING-IN AND OUT
C*****          S-SALINITY ARRAY,FLOATING-IN AND OUT
C*****          NOPTS-NUMBER OF POINTS IN EACH ARRAY,FIXED-IN AND OUT
C*****          ZBOT-BOTTOM POINT OF TRACE TO BE EXTRAPOLATED TO,FLOAT
C*****REMARKS* IT IS ASSUMED THAT ZBOT IS DEEPER THAN THE NEXT TO LAST
C*****    POINT ON THE INPUT DEPTH ARRAY
      DIMENSION Z(50),T(50),S(50)
      IF(ZBOT-Z(NOPTS))5,10,5
    5 CONTINUE
      IF(ZBOT .LT. Z(NOPTS))K=NOPTS
      IF(ZBOT .GT. Z(NOPTS))K=NOPTS+1
      XPRESN =(ZBOT-Z(NOPTS))/(Z(NOPTS)-Z(NOPTS-1))
      T(K) =T(NOPTS)+ XPRESN*(T(NOPTS)-T(NOPTS-1))
      S(K) =S(NOPTS)+ XPRESN*(S(NOPTS)-S(NOPTS-1))
      Z(K) =ZBOT
      NOPTS =K
   10 RETURN
      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.
DASC 012-C-77	Unavailable	LRAPP PACIFIC DYNAMIC ARCHIVE (U) SEPTEMBER 1976	Daniel Analytical Services Corporation	770201	NS; ND	U
SAI-78-527-WA	Spofford, C. W.	NELANT DATA ASSESSMENT APPENDIX III-MODELING REPORT	Science Applications, Inc.	770225	ND ADA 047 680	U
PSI TR 036049	Barnes, A. E., et al.	OCEAN ROUTE ENVELOPES	Planning Systems Inc.	770419	ND	U
Unavailable	Unavailable	TAP II BEAMFORMING SYSTEM SOFTWARE FINAL REPORT	Bunker-Ramo Corp. Electronic Systems Division	770501	ADC011789	U
S01037C8	Unavailable	TAP 2 PROCESSING SYSTEM FINAL REPORT HARDWARE DOCUMENTATION (U)	Bunker-Ramo Corp. Electronic Systems Division	770501	ADC011790; NS; ND	U
Unavailable	Weinberg, H.	GENERIC FACT	Naval Underwater Systems Center	770601	ADB019907	U
Unavailable	Unavailable	TASSRAP II OB SYSTEM TEST	Analysis and Technology, Inc.	770614	ADA955352	U
Unavailable	Unavailable	LRAPP TECHNICAL SUPPORT	Texas Instruments, Inc.	770624	ND	U
Unavailable	Bessette, R. J., et al.	TASSRAP INPUT MODULE	Analysis and Technology, Inc.	770729	ADA955340	U
Unavailable	Unavailable	TAP-II PHASE II FINAL REPORT	Bunker-Ramo Corp. Electronic Systems Division	770901	ADC011791	U
Unavailable	Unavailable	LONG RANGE ACOUSTIC PROPAGATION PROJECT (LRAPP)	Xonics, Inc.	770930	ADA076269	U
SAI78696WA	Unavailable	REVIEW OF MODELS OF BEAM-NOISE STATISTICS (U)	Science Applications Inc.	771101	NS; ND	U
TRACORT77RV109 C	Unavailable	FINAL REPORT FOR CONTRACT N00014-76-C-0066 (U)	Tracor Sciences and Systems	771130	ADC012607; NS; ND	U
Unavailable	Unavailable	LONG RANGE ACOUSTIC PROPAGATION PROJECT (LRAPP)	Xonics, Inc.	771231	ADB041703	U
Unavailable	Homer, C. I.	SUS SOURCE LEVEL ERROR ANALYSIS	Underwater Systems, Inc.	780120	ND	U
Unavailable	Fitzgerald, R. M.	LOW-FREQUENCY LIMITATION OF FACT	Naval Research Laboratory	780131	ADA054371	U
Unavailable	Unavailable	MIDWATER ACOUSTIC MEASUREMENT SYSTEM - PAR AND ACODAC	Texas Instruments, Inc.	780228	ADB039924	U
ORI TR 1245	Moses, E. J.	OPTIONS, REQUIREMENTS, AND RECOMMENDATIONS FOR AN LRAPP ACOUSTIC ARRAY PERFORMANCE MODEL	ORI, Inc.	780331	ND	U
Unavailable	Hosmer, R. F., et al.	COMBINED ACOUSTIC PROPAGATION IN EASTPAC REGION (EXERCISE CAPER): INITIAL ACOUSTIC ANALYSIS	Naval Ocean Systems Center	780601	ADB032496	U
LRAPPRC78023	Watrous, B. A.	LRAPP EXERCISE ENVIRONMENTAL DATA INVENTORY, JUNE 1978 (U)	Naval Ocean R&D Activity	780601	NS; ND	U
TR052085	Solomon, L. P., et al.	HISTORICAL TEMPORAL SHIPPING (U)	Planning Systems Inc.	780628	NS; ND	U