

AD-A103 892

WOODS HOLE OCEANOGRAPHIC INSTITUTION MA

F/G 8/10

VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM C--ETC(U)

AUG 81 N A BRAY

N00014-76-C-0197

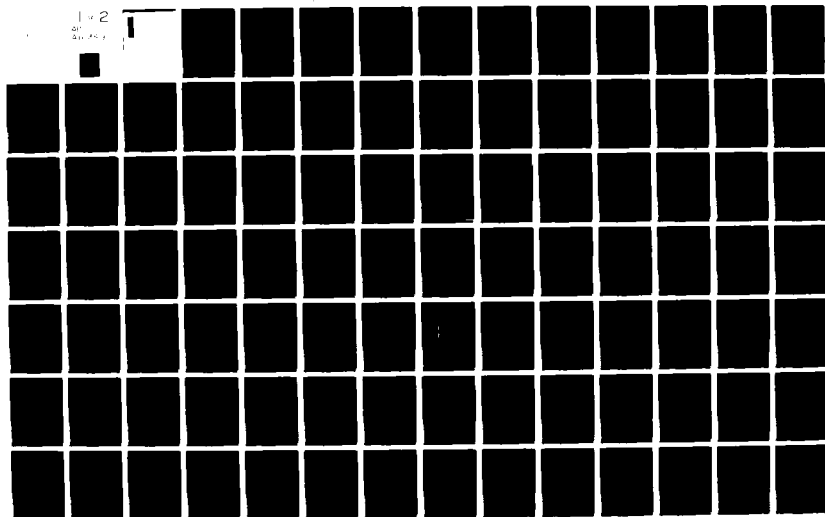
WHOI-81-70

NL

UNCLASSIFIED

1 of 2

01
10/1/81



UNCLASSIFIED 8/81

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER WHOI-81-70 ✓	2. GOVT ACCESSION NO. AD-A103 892	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM CTD DATA	5. TYPE OF REPORT & PERIOD COVERED Technical rept.	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Nancy Amanda Bray	8. CONTRACT OR GRANT NUMBER(s) * N00014-76-C-0197 ** N00014-79-C-0071 OCE-77-19403	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Woods Hole Oceanographic Institution Woods Hole, Massachusetts 02543	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS * NR 083-400 ** NR 083-004	
11. CONTROLLING OFFICE NAME AND ADDRESS NORDA/National Space Technology Laboratory Bay St. Louis, MS 39529	12. REPORT DATE August 1981	
13. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 140	13. NUMBER OF PAGES 135 pages	
	14. SECURITY CLASS. (of this report) Unclassified	
15. DECLASSIFICATION/DOWNGRADING SCHEDULE		
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES This report should be cited as: Woods Hole Oceanog. Inst. Tech. Rept. WHOI-81-70.		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) 1. Oceanic available potential energy 2. Definitions of horizontal density surfaces 3. Hydrographic data		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Fofonoff (1981). The report includes examples of how the programs may be used, as well as complete listings of all the required FORTRAN files.		

DD FORM 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED 8/81

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

381000

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
1st	Avail and/or Special
A	

WHOI-81-70

VAX-11 PROGRAMS FOR COMPUTING AVAILABLE
POTENTIAL ENERGY FROM CTD DATA

by

Nancy Amanda Bray

WOODS HOLE OCEANOGRAPHIC INSTITUTION
Woods Hole, Massachusetts 02543

August 1981

TECHNICAL REPORT

*Prepared for the Office of Naval Research under Contracts
N00014-76-C-197; NR 083-400 and N00014-79-C-0071; NR 083-
004 and by the National Science Foundation under Grant
OCE-77-19403.*

*Reproduction in whole or in part is permitted for any pur-
pose of the United States Government. This report should
be cited as: Woods Hole Oceanog. Inst. Tech. Rept. WHOI-
81-70.*

Approved for public release; distribution unlimited.

Approved for Distribution:

Valentine Worthington
Valentine Worthington, Chairman
Department of Physical Oceanography

DTIC
ELECTED
SEP 5 1981

D

TABLE OF CONTENTS

	Page
Abstract	3
List of Tables	4
List of Figures	5
Introduction	6
1. General Structure of programs	7
2. POTEN Documentation	13
3. PEPLT Documentation	27
4. Modification of POTEN to accept data in other than CTD78 disc format	51
Acknowledgements	53
References	54
Appendix A: Command files for generating standard output from POTEN and PEPLT	55
A1 ENERGY.COM	56
A2 POTEN.COM	67
A3 PEPLT.COM	68
A4 TABLE.COM	69
A5 DYNHT.COM	70
Appendix B: Program Listings for POTEN	71
B1 POTEN.DOC	72
B2 COMPOTEN.FOR	73
B3 IDXREC.DIM	74
B4 POTEN.FOR	75
B5 PTENS.FOR	76
B6 COMPS.FOR	81
B7 AVRCP.FOR	85
B8 DATA.FOR	91
B9 POTENSUB.FOR	95

		Page
Appendix C:	Program Listings for PEPLT	102
C1	PEPLS.DOC	103
C2	PEPLS3.DOC	104
C3	AVRGS.DOC	105
C4	COMPEPLT.FOR	106
C5	PEPLT.FOR	107
C6	PEPLS.FOR	108
C7	AVRGS.FOR	110
C8	TABLE.FOR	118
C9	TDATA.FOR	121
C10	PEPLTSUB.FOR	123

Abstract

This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Fofonoff (1981). The report includes examples of how the programs may be used, as well as complete listings of all the required FORTRAN files.

List of Tables

Table	Subject	Page
1	FORTRAN and Object files	11
2	Accessory files	12
3	Example of a JSHP.PTN file	16
4	POTEN Data Output Variables	18
5	POTEN Parameters	22
6	PEPLT Variables	31
7	PEPLT:PEPLS Parameters	40
8	PEPLT:AVRGS Parameters	43

List of Figures

Figure	Title	Page
1	POTEN Schematic	8
2	PEPLT Schematic	9
3	AVRGS Schematic	10
4	Example plots from TABLE subroutine	
	a. Buoyancy frequency	28
	b. Vertical displacements	29
5	Example plot from ENERGY.COM - APE	59
6	Example plot from ENERGY.COM - Rms displacement	60
7	- Salinity	61
8	- Potential temperature....	62
9	- Buoyancy frequency	63
10	- Reference steric anomaly.	64
11	- θ vs. S	65
12	- deep θ vs S	66

Introduction

This report describes the structure and usage of programs designed for calculating and displaying available potential energy (APE), adiabatically leveled steric surfaces, and related variables from a group of CTD stations. For a general discussion of the technique it is strongly recommended that the reader refer to Bray and Fofonoff (1981). The programs have an inherent requirement that the input CTD data be an even series in pressure, although the input pressure interval may be specified. This report describes specifically the structure of the programs as used on the W.H.O.I. VAX-11, with input data in the standard CTD78 disc format (Millard, et al (1978)). Other input formats can be accommodated through modification of the data input subroutine as described in section 4.

The calculation and display are divided into separate programs. POTential ENergy (POTEN) reads the input data, calculates the adiabatically leveled reference steric field (see Bray and Fofonoff, 1981) and variables related to the leveled field. Potential Energy PLoT (PEPLT) calculates variables derived from the leveled field variables and displays POTEN output in the form of lists and plots.

This report is divided into four sections. The first, General Structure, covers the non-FORTRAN aspects of the programs: file structure, linkage and general usage. The second and third sections contain detailed documentation for POTEN and PEPLT. The fourth section describes modifications to the data read subroutine in POTEN, to allow input data in other than CTD78 disc format. Documented examples of how to run the programs interactively and in batch mode on the VAX-11 are found in Appendix A. Listings of programs appear in Appendices B and C.

1. General Structure of Programs

Both POTEN and PEPLT are accessed through a short main program which performs initializations of parameters as requested by the user. Control is then transferred to one of three major subroutines, from which point the user is free to access different branches within that subroutine, or request entrance into either of the other two major subroutines. The various branches are described in detail in the following sections. Schematics of POTEN and PEPLT are shown in Figures 1, 2 and 3. The remainder of POTEN and PEPLT consist of secondary subroutines: data read, physical properties of seawater, etc., which are accessed as part of the various branches available to the user in the major subroutines. The file structure reflects the program structure (Table 1). POTEN and PEPLT are linked by linking the object files in Table 1. Accessory files are listed in Table 2.

The input data in CTD78 disc format is accessed using subroutines from CTDATA/LIB, and the plots in PEPLT are created using the NCAR plot package. The plot package creates a file on logical unit 8 which must be read and translated into plot(s) by a Metacode translator. Those translators are available both for the high speed Calcomp plotter and for various screens, for plot previewing. The absolute plot dimensions may be altered after the file is created, and the plots can be plotted as many times as desired. The use of the translators is described at the end of section 3.

The multiple branch structure of the programs provides an extremely powerful and flexible framework for computations which are often not routine; however useful documentation of such programs is correspondingly difficult. It is suggested that the new user begin by studying Figs. 1., 2 and 3. A documented command file (ENERGY.COM) for a routine computation and display is found in Appendix A. This file allows the new user to become familiar gradually with the options available in the programs. After studying and experimenting with the command file, the user may wish to explore other options available by referring to the detailed branch descriptions found in sections 2 and 3 of this report.

POTEN SCHEMATIC

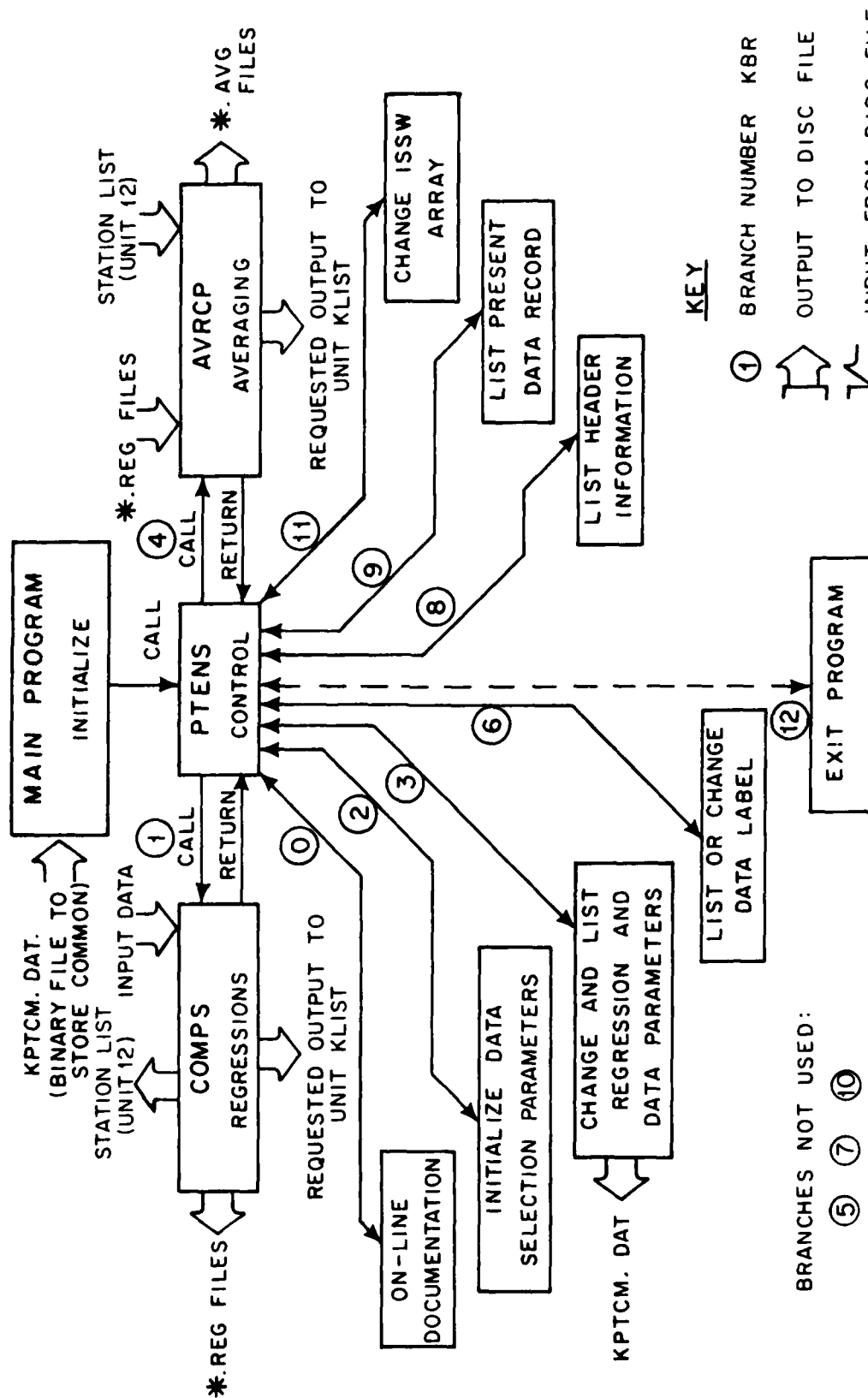


Figure 1: POTEN Schematic

PEPLT SCHEMATIC

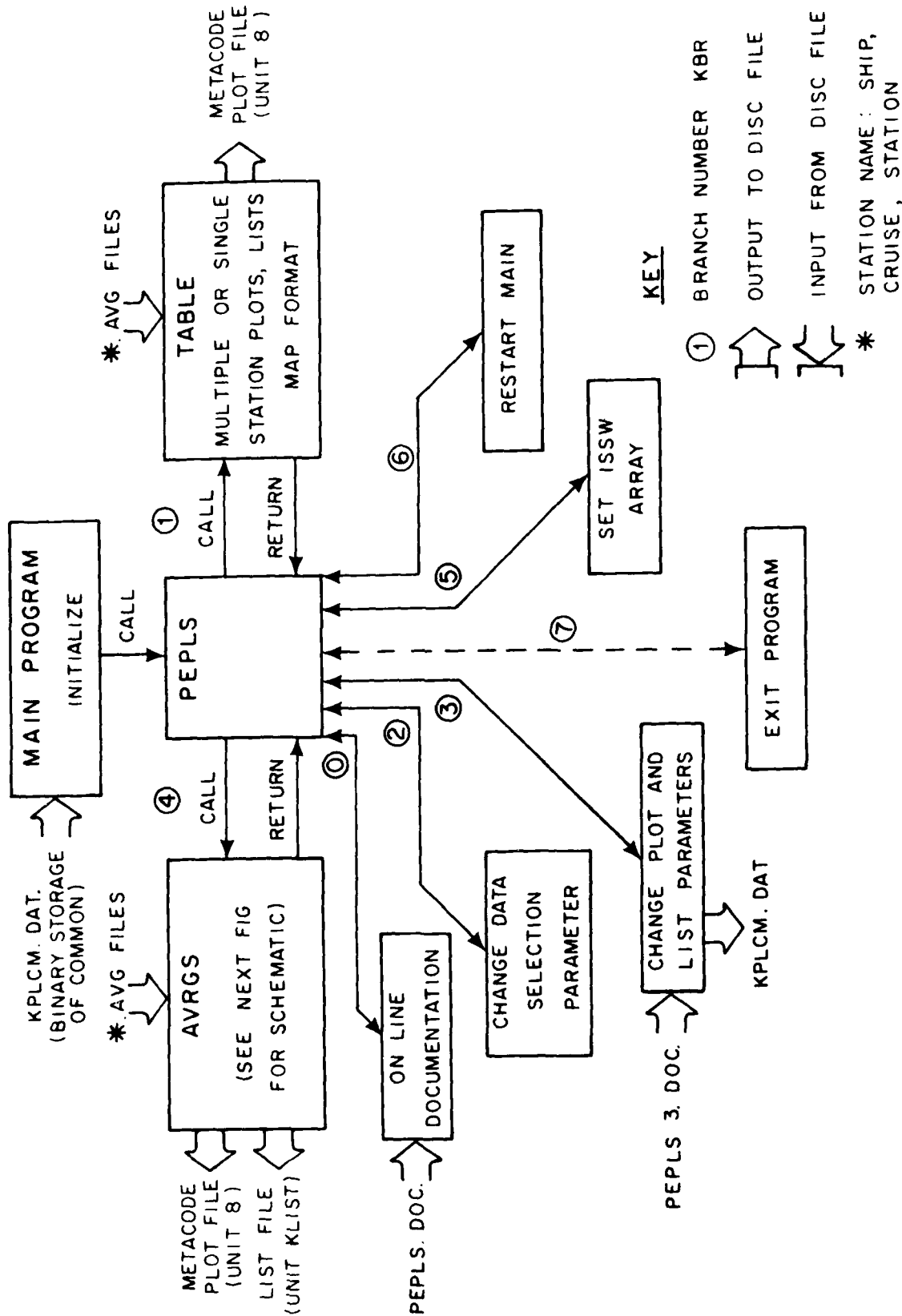


Figure 2: PEPLT Schematic

AVRGS SCHEMATIC

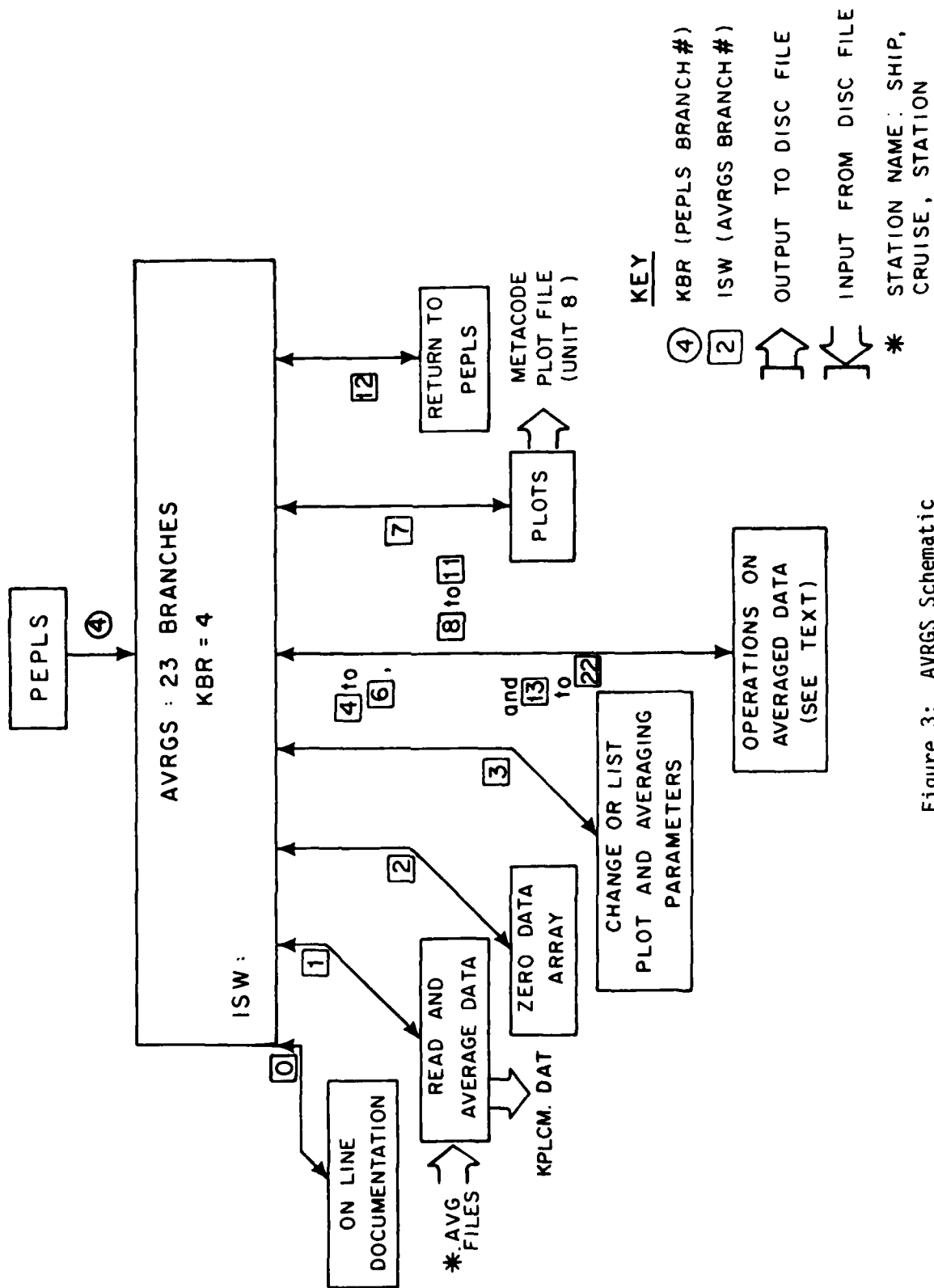


Figure 3: AVRGS Schematic

TABLE 1: FORTRAN and Object Files

	<u>POTEN Files</u>	<u>PEPLT Files</u>
Main program	POTEN	PEPLT
Major subroutines	PTENS	PEPLS
	COMPS	AVRGS
	AVRCP	TABLE
Data read subroutine	DATA	TDATA
Secondary subroutines	POTENSUB	PEPLTSUB
		POTENSUB
Library subroutines	CTDATA/LIB	
System subroutines (associated		AUTOGRAPH
with the NCAR plot package)		DASHCHAR
		NCAR

TABLE 2

<u>Accessory Files</u>	<u>POTEN</u>	<u>PEPLT</u>
Common, dimension and equivalence required for compilation (FORTRAN)	COMPOTEN.FOR	COMPEPLT.FOR
Station lists (JSHP.PTN is generated by COMPS -- unit 12)	JSHP.PTN	JSHP.PTN
Input data	*.CTD	*.AVG
*.CTD is CTD file	*.REG	
*.REG is generated by COMPS		
*.AVG is generated by AVRCP		
* is the station identifier		
Common storage (binary file)	KPTCM.DAT	KPLCM.DAT
POTEN and PEPLT generate these if they do not exist in the directory		
Documentation files (formatted)	POTEN.DOC	PEPLS.DOC PEPLS3.DOC AVRGS.DOC
Command file used to set up assignments and start an interactive job	POTEN.COM	PEPLT.COM
Command file used to run POTEN and PEPLT sequentially in batch mode, supplying standard output.	ENERGY.COM	
Command file to create plots shown in Fig. 4 using TABLE subroutine.		TABLE.COM
Command file to compute dynamic height station by station and output in map format.		DYNHT.COM

2. Documentation of POTEN

In this section, the major subroutines in POTEN described above are documented in detail. They are all structured around the multiple GO TO statement of the form

GO TO (#1, #2, #3,.... n) BRANCH

with #1 through #n FORTRAN statement numbers, and the BRANCH # an index such that BRANCH # = n transfers control to Statement #n. The BRANCH is input by the user following an appropriate program prompt. The branch numbers are keyed to various computations which may be accessed at the user's option. The keys are listed later in this section, and short versions of them may be obtained on the terminal any time the program prompts the user for branch number input, by typing 0/.

Within each branch there may be options which are accessible by varying parameters input by the user at the time the branch number is input. These options are also listed in the branch keys.

In addition to input parameter options, there is an array called ISSW with 16 elements found in both programs. Within the different branches, different elements of ISSW may be tested for values of -1 or 0, and options either accessed or skipped depending upon the value. In general, ISSW elements determine whether a given type of output is generated. (Historically, the ISSW array derives from the binary switches available on the shipboard computer, the HP2100 series.) The elements of ISSW may be altered by accessing the appropriate branch in both POTEN and PEPLT as described below.

As described earlier, POTEN is accessed as a short main program which initializes parameters if requested by the user, or reads from a binary file KPTCM.DAT the most recently stored parameters, if no initialization is requested. The main program POTEN then transfers control to the major subroutine PTENS which, as shown in the schematic Fig. 1, controls the various branches available to the user. PTENS is the only component of POTEN in which branches may be accessed. The two remaining major subroutines are COMPS, in which the regressions are performed, and AVRCP, in which the horizontal averaging is performed.

This subsection charts the branches available to the user in detail, and describes briefly the working of COMPS and AVRCP. Short versions of the branch documentation are found in the Appendix, and may also be printed on the screen while the program is (interactively) on line by typing Ø/ whenever the program prompts for branch input.

2a. Main Program:

The main program queries the user 'Initialize common (YES or NO)?'. A NO response causes the present elements of KPTCM.DAT, the binary storage file, to be read into common. (If no file KPTCM.DAT exists in the directory, the program will create a new file named KPTCM.DAT, but if the response to the initialization query was NO, an 'end of file during read' error will result. Therefore, the proper sequence of commands to create a new KPTCM.DAT file is to run POTEN, respond YES to the initialization, thereby creating a new KPTCM.DAT, but not attempting to read from it. Later in the program (in branch 3) common may be stored to the newly created file, for use next time the program is run.) A YES response initializes the data selection parameters (subroutine DATA), and certain other parameters not related to the regressions.

Following this query, control is transferred to PTENS, and the user is asked: 'Initialize regression parameters (YES or NO)?'. A YES response initializes the regression parameters. A NO response reads them from KPTCM.DAT. (Again, with a newly created KPTCM.DAT file, the correct response is YES.) Finally, PTENS asks for the resolution of the input data, before going to branch mode. At this point the user may input up to 7 variables, as listed in the program prompt. The current values of the variables are printed on the screen along with the prompt list. The variables are: KBR, the branch number; ISW and JSW, which may access different options in branch KBR; KLIST, usually the list output logical device number (reset to 6 each time the prompt is printed); KOUT and KTP, the data output and input logical device numbers (note that the program uses named files for data input and output via OPEN statements which use

KTP and KOUT as unit numbers); KIN, the program input device for screen or command file. (Changing the value of KIN to 6 part way through a COM file transfers control to the screen, allowing interactive mode -- see Appendix A for an example command file, POTEN.COM.)

2b. Branches (KBR)

0: Short documentation printed on screen. See Appendix B for a listing of this documentation.

SUMMARY - POTEN:PTENS: KBR = 0

Function: List on terminal the short documentation for PTENS

ISW, JSW options: None

Output device: unit KTTX

Input device: None

ISSW options: None

- 1: Calls COMPS subroutine, which performs the following sequence:
 - a. Calls subroutine DATA, which opens the subindex directory for the default file specifications of the input data. (Those specifications may be changed by calling KBR = 13, which is identical to branch 1 except for allowing file specifications to be changed.) Then the header for the ISWth sequential station in that subindex file is examined to see if it meets data selection criteria. If so, a file name is written to file corresponding to logical unit 12. Throughout this report that file is called JSHP.PTN; an example is given in Table 3. The temperature and salinity data are transferred to array DATA, using Millard subroutine GETDAT. Pressure is stored in the zeroth element of DATA, which is equivalenced to array PRESS. PRESS is used through COMPS and AVRCP. The total number of scans (NTOT) is also noted. The above occurs in subroutine DATA, after which control returns to COMPS.
 - b. COMPS then sets up the regression for the first interval using parameters which may be changed using branch 3 and continues the computation through all the intervals requested,

TABLE 3

Example of a JSHP.PTN file generated by COMPS

<u>Consec.</u> <u>Number</u>	<u>Station I.D.</u>	<u>Weight</u>
1	GY001002	1.0
2	GY001003	1.0
3	GY001004	1.0

or until the end of the data (determined by NTOT) is reached. For each interval potential temperature and steric anomaly referred to p_f (the level pressure) are calculated for each data scan to be used in the regression. Potential temperature is calculated according to Fofonoff (1977), using the polynomial formula of Bryden (1973) for the adiabatic temperature gradient. Steric anomaly δ is calculated as:

$$\delta = 10^5 \times (\alpha(p, \theta(p, T, S, p_f), \delta) - \alpha(p, 0, 35))$$

with α the specific volume calculated according to the SCOR Working Group 51 new equation of state for seawater (Millero, et al, 1980), for which an algorithm is given by Fofonoff (1981). Within each interval an editing process occurs in which points exceeding three standard deviations of the regression estimate at a given steric anomaly are flagged. Temperature and salinity are then regressed against pressure over the interval. Any points in T or S which exceed three standard deviations are replaced by the regression estimate. The regression of steric anomaly is performed again and rechecked. The number of standard deviations for both tests may be changed -- see $KBR = 3$. The interpolated scans are printed out on unit KLIST and data scans which are flagged but not interpolated are also listed as such on KLIST if $ISSW(3)$ is set to -1. ($ISSW$ values may be changed using branch 5.) Pressure p and potential temperature θ (referred to the level pressure p_f) are regressed against steric volume anomaly (also referred to p_f) and the coefficients for both p and θ are stored in arrays CP and CT for each interval. Data output occurs if $ISSW(13) = -1$, and is written into a file with the name *.REG, where * identifies the station, a two character (alpha) ship name, a 3 digit cruise number and a 3 digit station number. The format of the output file is a header of 150 words equivalenced to an $I*4$ array followed by a variable number of data records (each 46 words, also an $I*4$ array), one record per level

TABLE 4
POTEN Data Output Variables

HEADER RECORD: 150 WORDS

VARIABLE

NAMEDESCRIPTION

LTYPE	Identifies record as header record (LTYPE = 1)
MHDR	Number of elements in header
ICON	Sequential number of station (in POTEN calculation)
ISHP	Ship name (A2 format)
KCAST	Station number
IDAY	Julian year day
IPR	First pressure
LPR	Last pressure
XLAT	Latitude of station
XLONG	Longitude of station
WGT	Weight
XLTD	Latitude of origin for distance computations in kilometers (negative for south latitude)
XLGD	Longitude of origin (negative for west)
LBBL(3)	Short station label (3A4 format)
LBL(13)	Run identification label (13A4 format)
NSC(60)	Regression parameters } see text
NPR(60)	Regression parameters
NSECTION	Number of sections in the water column

DATA RECORD: 46 WORDS

KTYPE	Identifies record as data record (KTYPE = 0)
MBUF	Number of elements in data record
IREC	Level number
N	Polynomial order
NDP	Number of data scans used in regression
KSW	Not used

TABLE 4 (continued)

<u>NAME</u>	<u>DESCRIPTION</u>
L1	Not used
L2	Not used
PF	Level pressure
T0,S0,DV0	Temperature, salinity and steric anomaly from input data, averaged about PF $\neq$ PDIFF (see branch 3 description and Table 5)
PI	Pressure of the reference steric anomaly (DVF) in the unlevelled or initial field
THF	Local potential temperature (referred to PF) as estimated by the regressions: $\theta_f(P_f)$
DVI	Steric anomaly corresponding to PF in the initial field
DVF	Steric anomaly corresponding to PF in the leveled field
PM,THM,SM,DVM	Average of pressure, potential temperature, salinity and steric anomaly over the regression interval.
DH	ds/dp based on the averaged regression coefficients
PE	Potential energy anomaly }
XPE	Horizontally average PE
	Recommended that these not be used, but calculated in PEPLT
CP(8)	Pressure vs. steric anomaly coefficients
Z1	Standard deviation of regression pressure estimate (Fofonoff and Bryden, 1975)
CT(8)	Potential temperature vs. steric anomaly regression coefficients
Z2	Standard deviation of regression temperature estimate
F1,F2,F3	Steric volume minimum, maximum and average over regression interval
XL0:	Latitude of origin: default is 40.0
XLG0:	Longitude of origin: default is -70.0

p_f . The output is in binary (unformatted) files. The variables output are identified in Table 4. Some information at each level may be output to unit KLIST if ISSW (12) = -1, for purposes of checking. Header information is output to unit KLIST if ISSW (11) = -1. The input data scans are output to unit KLIST if ISSW (5) = -1 and the regression coefficients and residuals are output to unit KLIST if ISSW (10) = -1. If ISSW (6) = -1 statistics of the coefficients are printed on unit KLIST. The ratio of each coefficient to its standard deviation (see Fofonoff and Bryden, 1975, Appendix) is computed. For an infinite number of degrees of freedom, at 95% confidence that ratio should equal or exceed 1.96. The statistic which is listed is (a_i the coefficients):

$$\frac{a_i}{\text{std dev } (a_i)} \cdot$$

When stations with subindex reference number (sequential number) ISW through JSW have been tested for data selection criteria and either been skipped or have gone through the regression calculation, COMPS returns control to PTENS.

SUMMARY - POTEN:PTENS: KBR = 1

Function: calls COMPS subroutine

ISW, JSW Options: ISW to JSW are the station reference numbers

Output device: data goes automatically to *.REG file if
ISSW (13) = -1; other information output goes
to unit KLIST, as requested by elements of ISSW

ISSW options: 3 = -1 Print out interpolated scans

(to unit KLIST) 5 = -1 Print out input data scans

6 = -1 Print out coefficient statistics

10 = -1 Print out regression coefficients for
each scan

11 = -1 Print out header information

12 = -1 Print out selected data following
regression

13 = -1 Data output to *.REG

2: Initializes data selection parameters described in Table 5

SUMMARY - POTEN:PTENS: KBR = 2

Function: Initialize data selection parameters

ISW, JSW Options: None

Output device: None

ISSW Options: None

3: Changes or lists regression and data selection parameters described in Table 5. The data selection parameters are straightforward. For the regression parameters the water column is divided into a maximum of nine sections, each of which may have a number of levels whose regression parameters are the same. The regression parameters consist of the total number of sections; in each section, the interval between leveled surfaces, the interval over which the regression is performed, the polynomial order, and start and end pressures for the section. All of these parameters are input using subroutine PARAM, which branch 3 calls. The prompts are (hopefully) self-explanatory. After parameters have been entered for all sections, PARAM translates them into internal parameters which control the way the program performs the regressions. These internal parameters are stored in arrays NPR and NSC. Since the arrays NPR and NSC are included in common stored to KPTCM.DAT, the user form parameters need be entered only once, until a change is required. The old parameters may be retrieved by responding 'NO' to the initial query in PTENS 'Initialize regression parameters?'. Stored common is written to KPTCM.DAT at the end of branch 3, so any changes in regression parameters will overwrite the most recent ones in KPTCM.DAT, provided branch 3 is completed. It is not possible to change only a single regression parameter; if a change is required, all the parameters must be re-entered. (This is because the internal parameters NPR and NSC have elements whose value depends upon parameters for more than

TABLE 5
 POTEN Parameters: Branch KBR = 3

<u>Parameter</u>	<u>Definition</u>	<u>Default if Initialized</u>
ICON	Consecutive number	1 for first station. Increments with stations processed
KSW	Not used	1
A2	Number of standard deviations allowed for a regression point in $p(\delta)$ before flagging.	3.
A3	Number of standard deviations allowed for a regression point in $T(p)$ and $S(p)$ before interpolation	3.
WGT	Weight	1.
POIFF	Interval (db) about P_f for averaging $T\theta, S\theta, P\theta$	6.
DELP	Pressure series interval for input CTD data (db)	2.

REGRESSION Parameters -- as described in program prompts

Data selection parameters: windows such that data inside all windows is
 accessed; all other data skipped

IDAY1	: Minimum Julian year day	0
IDAY2	: Maximum Julian year day	365
JDO	: Additive constant to actual day	0
XEMN	: Minimum longitude	-180.0
XEMX	: Maximum longitude	180.0
XMN	: Minimum longitude	- 90.0
XNMX	: Maximum latitude	90.0

one section. PARAM requires that parameters be input sequentially.) It is not necessary to understand how NPR and NSC work in order to run the program (that is the purpose of the PARAM subroutine); however, modifications of the program may require that the programmer know how these arrays function. A brief description is therefore presented here. The pressure p_f for each level is given by:

For IREC less than NPR(section #)

$$PF = NPR(\text{section \#} + \text{total number of sections}) \times$$

$$(\text{IREC} - \text{NPR}(\text{section \#} + 2 \times \text{total number of sections}))$$

NPR (section # + total number of sections) contains the interval between pressure levels; NPR(section # + 2 x total number of sections) contains an index which allows the correct p_f to be determined, while NPR(section #) contains the level number at which the section commences. Some care should be taken to assure that the parameters input are consistent.

Specifically, the first level of a new section must have a pressure p_f such that p_f is some integral multiple of the pressure interval between leveled surfaces in that section. The use of the total number of sections allows the program to treat NPR as a variable length two-dimensional array, even though it is in fact singly dimensioned. Subroutine PARAM adds an additional 'dummy' section below those input by the user to assure that COMPS does not continue below the desired depth. Thus, the total number of sections (NSECTION) will always be one greater than the number input by the user.

Array NSC contains the remainder of the parameters: start pressure in NSC(section#), polynomial order in NSC(NSECTION + section #), number of data scans in the regression interval in NSC(2*NSECTION + section #).

SUMMARY - POTEN:PTENS: KBR = 3

Function: Change or list regression and data selection parameters
ISW, JSW Options: ISW = 0: short list only

ISW = 1: full list
 JSW: no options
 Input device: unit KIN
 Output device: unit KLIST
 ISSW Options: None

- 4: Call AVRCP - averaging subroutine. The pressure and potential temperature coefficients from the regressions performed in COMPS are averaged horizontally, level by level. The average pressure polynomial at each p_f is set equal to p_f (corresponding to a mass conservation constraint between the initial and leveled fields) and the resultant polynomial is inverted to obtain the reference steric anomaly (δ_f) corresponding to that p_f . (See Bray and Fofonoff, 1981 for a more detailed discussion.)

The averaging is actually done in two 'passes' through the data, but a single call to AVRCP with ISSW(7) = 0 will automatically average and output new station data files based on the leveled field. (Data output occurs if ISSW(13) = -1, as in COMPS. The new files are called *.AVG with * as before the station identifier.) Information about the averaged pressure coefficients is output to unit KLIST if ISSW(12) = -1. Information about the averaged steric field is output to unit KLIST if ISSW(11) = -1.

The two averaging 'passes' may be accessed individually, and separately from the data output by setting ISSW(7) = -1 and entering KBR = 4, ISW = 1 for the first pass, KBR = 4, ISW = 2 for the second pass and KBR = 4, ISW = 3 to output the new station data files. However, since the second pass must be performed directly after the first, and the output directly after the averaging it is recommended that the automatic access be used (ISSW(7) = 0). If no output is desired, ISSW(13) should be set to 0.

SUMMARY - POTEN:PTENS: KBR = 4

Function: Call AVRCP averaging subroutine

ISW, JSW options: If ISSW(7) = -1 ISW = 1: First averaging pass
 ISW = 2: Second averaging pass
 ISW = 3: Output of data to
 *.AVG files if
 ISSW(13) = -1
 If ISSW(7) = 0: ISW = 1: Averaging and output
 performed
 automatically.

Input files: *.REG

Output files, data: *.AVG

Output files, lists: unit KLIST

ISSW Options: ISSW(7) = -1: individual access of averaging passes
 ISSW(11) = -1: List of averaged steric field on
 unit KLIST
 ISSW(12) = -1: List of averaged pressure
 coefficients on unit KLIST
 ISSW(13) = -1: Levelled field based data output to
 *.AVG files

5: Not used

6: Print data label. This label is input by the user in branch 3,
 and is carried in both the *.REG and *.AVG files as an identifier
 of the group of stations, the version of the POTEN run, etc. Its
 format is 13 A4 or a total of 52 characters. Branch 6 lists this
 label to unit KLIST.

SUMMARY - POTEN:PTENS: KBR = 6

Function: Write data label

ISW, JSW options: None

Output device: unit KLIST

ISSW options: None

7: Not used

8: Write header record to unit KLIST: Station label, position,
 origin, LTYPE, MHDR, ICON, ISHP, ICAST, JDAY, IPR, LPR. This is
 also done automatically in subroutine DATA when COMPS accesses the
 station, provided ISSW(11) = -1.

SUMMARY - POTEN:PTENS: KBR = 8

Function: Write station header information

ISW, JSW options: None

Output device: unit KLIST

ISSW options: None

- 9: Write *.REG or *.AVG single data record to unit KLIST. Of doubtful usefulness, this branch was part of the original program.

SUMMARY - POTEN:PTENS: KBR = 9

Function: Write single output data record to unit KLIST

ISW, JSW options: None

Output device: unit KLIST

ISSW options: None

- 10: Not used

- 11: Set the values of the ISSW array. One call allows up to 16 inputs. Each input consists of element number followed by a comma and the value to assign to that element. Whenever input is complete, if less than 16, the branch may be terminated with a /.

SUMMARY - POTEN:PTENS: KBR = 11

Function: Set ISSW array

ISW, JSW options: None

Output device: unit KTTX

Input device: unit KIN

ISSW options: None

- 12: Exit program. Program queries 'Exit program '. A YES response results in a FORTRAN stop statement execution. A NO response returns the PTENS branch prompt.

SUMMARY - POTEN:PTENS: KBR = 12

Function: Exit program

Input device: unit KIN

If a value of KBR greater than 12 or less than 0 is entered, the short documentaton is printed on the screen.

3. PEPLT Documentation

Like POTEN, PEPLT is accessed through a short main program, which initializes parameters as requested by the user, and then transfers control to a major subroutine, PEPLS. From PEPLS, the user may call subroutine TABLE, which plots and lists station by station, and subroutine AVRGS which computes and displays horizontally averaged quantities as a function of depth. Subroutine AVRGS has its own set of internal branches, one of which returns program control to PEPLS. Subroutine TABLE has no internal branches. As in POTEN, short documentation can be displayed on the screen while the program is running interactively, by typing `D/` as a response to branch prompts in either PEPLS or AVRGS.

3a. Main Program: PEPLT

The main program queries 'Load in previously stored common?'. A YES response causes the elements of the binary array KPLCM.DAT to be read into common, beginning with the common element KTX. a 'NO' response causes no action by the program. Control is then transferred to subroutine PEPLS.

3b. Branches - PEPLT

- 1: Calls subroutine TABLE. TABLE plots and lists station by station. It also outputs requested information in a format appropriate as input to objective mapping programs. The plot section of TABLE is designed to permit a number of stations to be plotted on the same frame, with the origin of each station within the larger frame. Examples are shown in Fig. 4. In Fig. 4a the buoyancy frequency N is plotted as a function of geographical position (relative to an origin at 37°N , 69.65°W), the coordinates of the frame; and, for each station, as a function of depth, where the station axes represent 0 to 3000 db vertically and -3 to 3 cph horizontally. This is accomplished by scaling the buoyancy frequency, and adding it to the X-coordinate (in

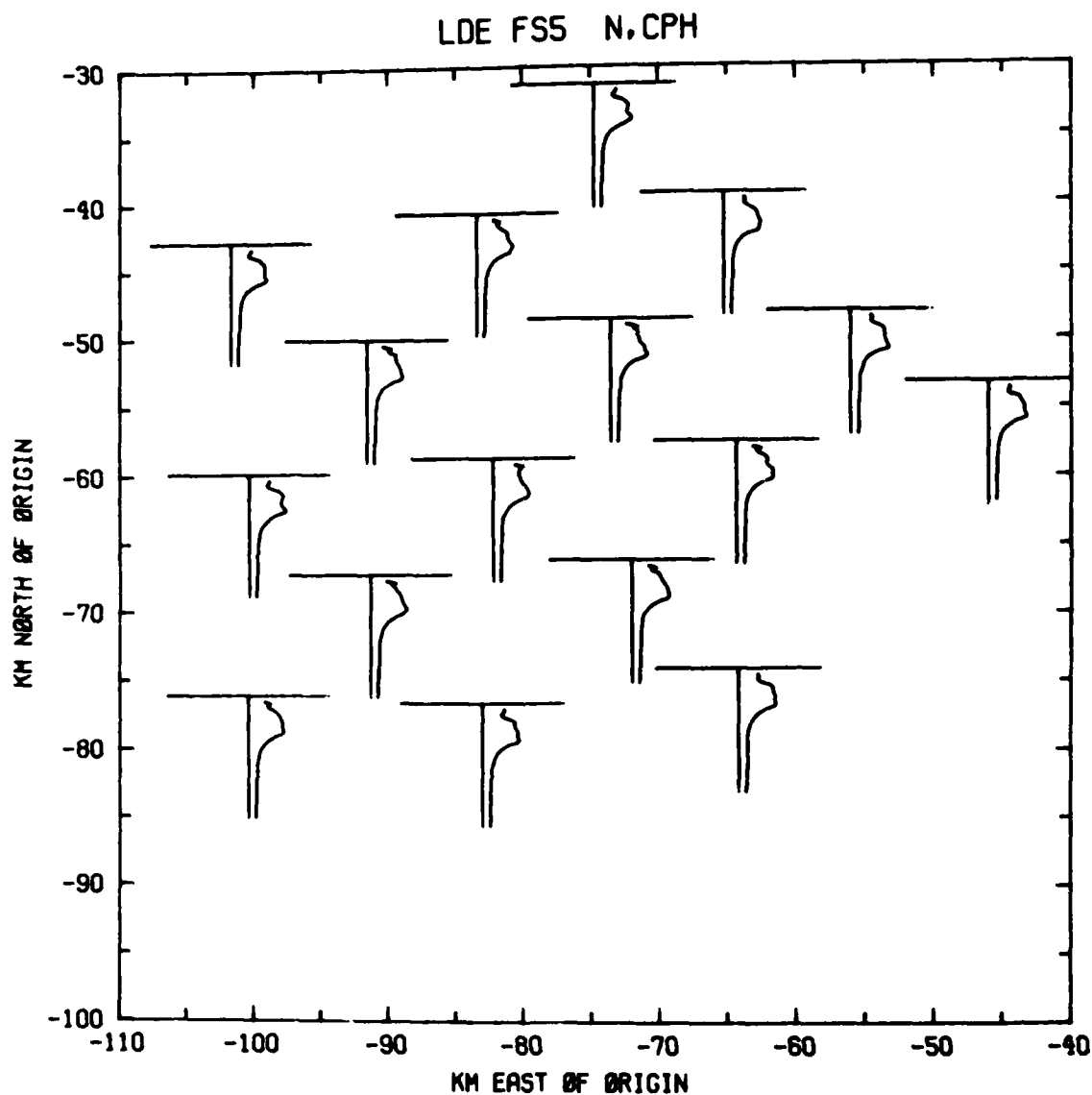
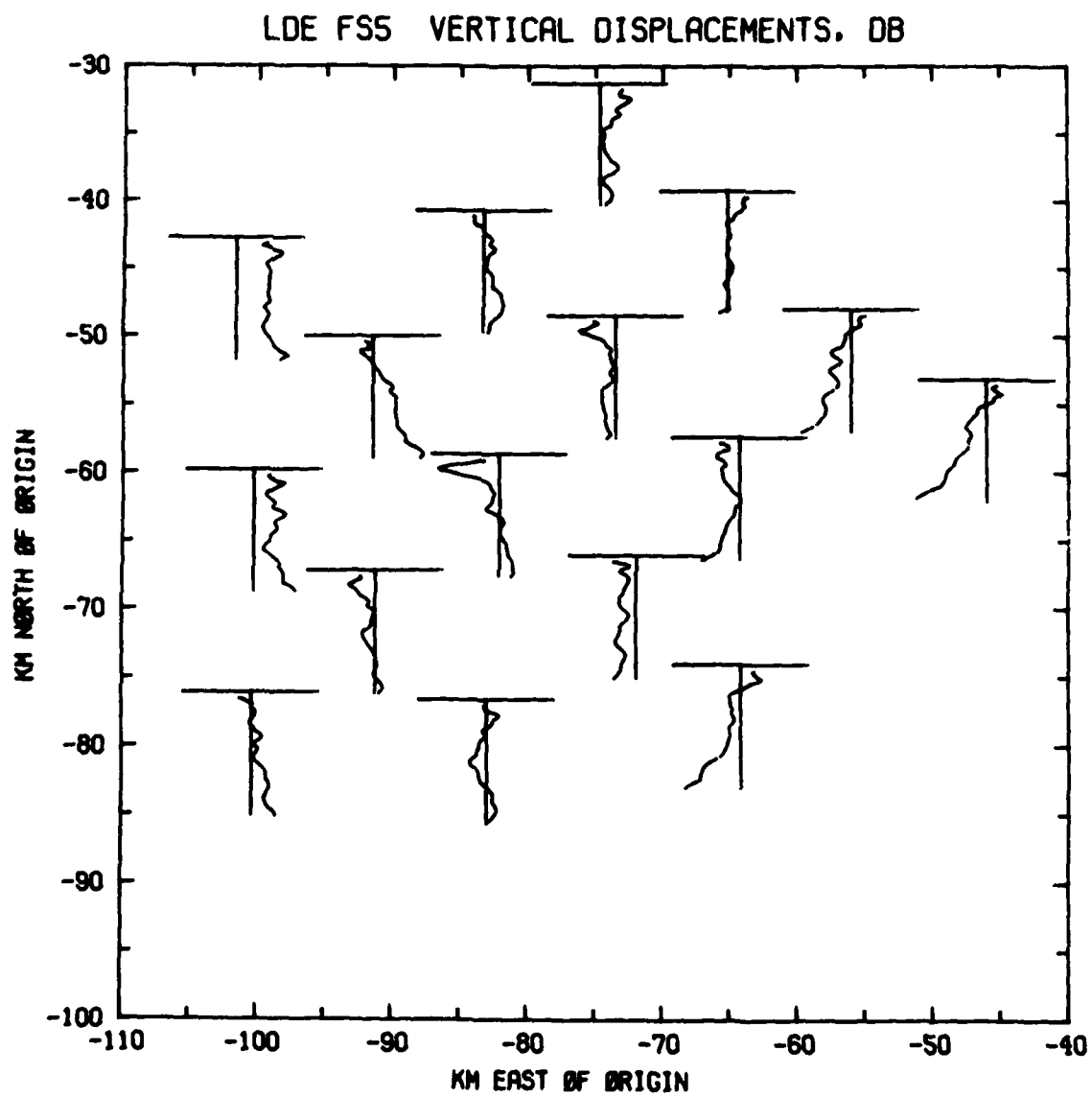


Figure 4: Example plots from TABLE.COM

- 4a. Buoyancy Frequency, N in cph. Inset axes represent ± 3 cph (horizontal), 0 to 3000 db in pressure (vertical). The origin on the inset axes (0,0) represents the station location.



4b. Vertical displacement π in db. Inset axes represent ± 100 db (horizontal) and 0 to 3000 db (vertical).

this example, kilometers east of the origin) and scaling the pressure and subtracting from the Y-coordinate (here km north of the origin). The program allows up to four variables to be included in such a sum for X and Y. The length of the station axes are determined by the variables X2DIM and Y2DIM, which are input in user units (i.e. cph in Fig. 4a) for the variables being plotted. In the example, Y2DIM is 3000 (db) and X2DIM is 3 (cph). One frame is created for each call to TABLE; it will encompass ND station plots. The plot parameters may be initialized by calling TABLE (KBR = 1) with ISW $\geq$ 2. Control then returns to PEPLS. Plot parameters may be changed by calling TABLE with ISW = 1. Control again returns to PEPLS. Plotting commences only when TABLE is called with ISW = 0. Figure 4b is the same type of plot as 4a, with vertical displacements plotted instead of N. Both of these plots were created using the documented command file TABLE.COM found in Appendix A.

A number of variables relating to the leveled field, the initial field, and the location, and time of each station may be examined using PEPLT. A list of these variables is found in Table 6; they are computed in function subroutine VRBL, coded by number. Thus a call to VRBL (3) returns the latitude of the station being examined (variable XLAT). Subroutine TABLE plots the following for x and y:

$$\begin{aligned} x &= A1*VRBL(NX1) + A2*VRBL(NX2) + A3*VRBL(NX3) \\ &\quad + A4*C(IREC,1) \\ y &= B1*VRBL(NY1) + B2*VRBL(NY2) + B3*VRBL(NY3) \\ &\quad + B4*C(IREC,2) \end{aligned}$$

Here C(IREC,n) refers to an array which may be filled using AVRGS subroutine (see branch 4). A1 to A4, B1 to B4, NX1 to NX3, and NY1 to NY3 may be changed by accessing branch 3. The default values (initialized by KBR = 1, ISW = 2) are:

TABLE 6 -- PEPLT Variables
(Nomenclature follows that of Bray and Fofonoff, 1981)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
-1	1	None	Returns the number 1 (counts number of observations at each level).
0	0	None	Returns 0
1	XPL	km	Zonal distance from origin (XLT0,XLG0)
2	YPL	km	Meridional distance from origin
3	XLAT	degrees	Signed decimal latitude (south negative)
4	XLONG	degrees	Signed decimal longitude (west negative)
5	ICON	None	Consecutive station number in POTEN computation
6	WGT	None	Averaging weight
7	JDAY	days	Julian year day
8	ISHP	None	Ship code
9	ICAST	None	Station number
10	N	None	Polynomial order
11	NDP	None	Number of data scans in regression interval
12	PF	db	Level pressure
13	T0	°C	T, S, σ averaged over the interval PF $\neq$ PDIFF
14	S0	ppt	
15	DV0	$10^{-5}\text{cm}^3\cdot\text{gm}^{-1}$	
16	PI	db	Pressure of reference steric anomaly (δ_f) in the initial field

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
17	THF	°C	<u>Local</u> potential temperature referred to PF. (See 65.)
18	DVI	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}$	Initial steric anomaly δ_i on PF
19	DVF	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}$	Reference steric anomaly (δ_f) on PF
20	PM	db	Pressure, local σ , salinity and steric anomaly referred to P_f averaged over regression interval
21	THM	°C	
22	SM	ppt	
23	DVM	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}$	
24	$\alpha_{P_f}^*$	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1} \text{db}^{-1}$	$d\delta/dp$ based on averaged regression coefficients
25	x		Potential energy anomaly calculated by POTEN
26	APE		APE calculated by POTEN
27 to 34	CP(1) to CP(8)	(db) ⁻¹	Regression coefficients for pressure
35	Z1	db	Standard deviation of pressure regression estimate
36 to 43	CT(1) to CP(8)	(°C) ⁻¹	Regression coefficients for local potential temperature
44	Z2	°C	Standard deviation of local potential temperature regression estimate
46,47,48	F1,F2,F3	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}$	Minimum, maximum and average values of steric anomaly over regression interval

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
48	π^*	db	'Boussinesq' displacements: $\pi^* = -(\delta_i - \bar{\delta}_i)/(\frac{d\delta}{dp})^*$
49	PI*	db	$\pi^* + PF$
50	π	db	Displacement of initial field from reference field PI-PF. Positive implies downward.
51	$\Delta\delta$	$10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}$	$\delta_i - \delta_f$
52	Vortex stretching	db	$PIX \frac{\sin(XLAT)}{\sin(XLAT)}$
53	$\frac{APE_B^*}{g}$		$\frac{1}{g}$ 'Boussinesq' APE with true displacements $-\frac{1}{2g} \alpha_p^* \pi^{*2}$
54			$gPF\Delta D - PE$
55	$\frac{APE_B}{g}$		$\frac{1}{g}$ Boussinesq APE with Boussinesq displacements $-\frac{1}{2g} \alpha_p^* \pi^{*2}$
56	$E\theta$	$(10^{-5} \text{cm}^3 \cdot \text{gm}^{-1})^{-1} \text{db}$	Inverse of local specific volume gradient with pressure $(\frac{dp}{d\delta})$
57	$1/E\theta$	$(10^{-5} \text{cm}^3 \cdot \text{gm}^{-1}) \text{db}^{-1}$	Local specific volume gradient with pressure
58	N^2	$10^{-6} (\text{rad} \cdot \text{sec}^{-1})^2$	Squared buoyancy frequency
59	e_p	$^{\circ}\text{C db}^{-1}$	Potential temperature gradient $\frac{de_f}{dp}$

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
60	S_p	ppt db ⁻¹	Salinity gradient $\frac{dS_f}{dp}$
61	S_f	ppt	Salinity corresponding to θ_f, p_f, δ_f
62	$\frac{dS_f}{d\theta_f}$	ppt(°C) ⁻¹	Gradient of salinity with potential temperature
63	π^2	(db) ²	Squared displacement of initial field from reference field
64	N	cph	Buoyancy frequency
65	θ_f	°C	Potential temperature at δ_f referred to zero pressure
66	σ_θ	10 ⁻³ gm-cm ⁻³	Sigma theta of $p_f,$ S_f, θ_f
67	α_f	cm ³ ·gm ⁻¹	Specific volume anomaly in reference field
68	$-\frac{1}{2} \Gamma_k \pi^2$		Vertical gradient of compressibility contribution to GPE
69	Not used		
70	Γ_k	10 ⁻⁵ cm ³ ·gm ⁻¹ ·db ²	$\frac{d\kappa}{dp} - \left(\frac{\partial\kappa}{\partial p}\right)_a$ (see Bray and Fofonoff, 1981)
71	Not used		

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
72	ΔS_f	ppt	Salinity anomaly from cubic spline fit to Worthington-Metcalf and Iselin θ -s curves $\Delta S(p_f, \theta_f, S_f)$ (Armi and Bray, 1981)
73	ΔS_i	ppt	Same as 72 but using the initial field $\Delta S(P_i, \theta(T\theta), S(\theta))$
74	θ_f^2	($^{\circ}\text{C}$) ²	Leveled field potential temperature squared. (Used in calculating horizontal standard deviation using AVRGS branch ISW = 21.)
75	RME	($10^{-4} \text{ J} \cdot \text{kg}^{-1}$) ²	Random measurement error (based on pressure error of $\pm 5\text{db}$ magnitude, temperature error of $.007^{\circ}\text{C}$, salinity error of $.005 \text{ ppt}$) for APE. See Bray and Fofonoff, 1981, Appendix, for details of error calculations. $V(\alpha_p * \pi^2 / 2)$
76	Not used		
77	RFE	($10^{-4} \text{ J} \cdot \text{kg}^{-1}$) ²	Same as 78 but π
78	RFE*	($10^{-4} \text{ J} \cdot \text{kg}^{-1}$) ²	Random finestructure error: (based on 3xZ1 as error in π^*): $V(\alpha_p * \pi^2 / 2)$

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
79	RME	$(10^{-4} \text{ J} \cdot \text{kg}^{-1})^2$	Same as 75 but pressure error only
80	RFEC	$(10^{-4} \text{ J} \cdot \text{kg}^{-1})^2$	Random finestructure errors in the vertical compressibility term (must be integrated using AVRGS branch ISW = 17) $V(\Gamma_k \pi^2/2)$
81	RMEC	$(10^{-4} \text{ J} \cdot \text{kg}^{-1})^2$	Random measurement error in the vertical compressibility term (pressure error only)
82,83	Not used		
84	κ	$10^{-5} \text{ cm}^3 \cdot \text{gm}^{-1} \cdot \text{db}^{-1}$	Compressibility $(\frac{\partial \delta}{\partial p})_a$
85	κ_S	$10^{-5} \text{ cm}^3 \cdot \text{gm}^{-1} \cdot \text{ppt}^{-1}$	Derivative of specific volume with respect to salinity; temperature and pressure held constant: $(\frac{\partial \delta}{\partial S})_{P,T}$
86	$\kappa \pi$		Contribution to GPE from horizontal gradients of compressibility
87	APE_B	$10^{-4} \text{ J} \cdot \text{kg}^{-1}$	'Boussinesq' APE per unit mass with true displacements
88-90	Not used		
91	$(\delta_i - \bar{\delta}_i)$	$10^{-5} \text{ cm}^3 \cdot \text{gm}^{-1}$	
92	$(-\theta_p \pi)^{-1}$	$(^\circ\text{C})^{-1}$	
93	Not used		

TABLE 6 (Continued)

VARIABLE NUMBER	NAME or SYMBOL	UNITS	DESCRIPTION
94	θ_i^2		Potential temperature corresponding to p_f in initial field squared
95	$-(\theta_i - \bar{\theta}_i)/\theta_p \pi$ (if $\bar{\theta}_i$ is in column 4)	None	
96	$-(\theta_f - \bar{\theta}_f)/\theta_p \pi$ (if $\bar{\theta}_f$ is in column 4)	None	
97	θ_i	$^{\circ}\text{C}$	Local potential temperature at δ_i
98	Not used		

A1 = 1	B1 = 1	NX1 = 1	NY1 = 2
A2 = 2	B2 = -.003	NX2 = 64	NY2 = 12
A3 = 0	B3 = 0	NX3 = 0	NY3 = 0
A4 = 0	B4 = 0		

These values will cause the buoyancy frequency in cph to be plotted as a function of meridional position (y-axis), time (x-axis) and pressure (station axis). X2DIM defaults to 3 (cph) and Y2DIM to 3000 (db), resulting in station axes representing ± 3 cph for the displacements and 0 to 3000 db for the pressure. The default number of stations (variable ND) is 1 and may be changed by calling KBR = 1, ISW = 1. The plot information is stored in the file corresponding to unit 8. It must be read and translated by a Metacode translator. PEPLT may be run on any terminal, but the translators are only available for graphics terminals and the Calcomp plotter. See the last part of this section for instructions on the access of the translators. The origin co-ordinates may be changed in PEPLS branch 2.

Branch 1 with ISW = 0 may be used to change PMIN and PMAX, thereby selecting a range in pressure over which data will be used (all other data is excluded), X2DIM and Y2DIM, described above, JMIN, the level number corresponding to the pressure at which the plot is to start (this allows the user to skip over shallow points which may have anomalous values), and various plot parameters. The plot parameters include PLABL, the overall plot label; XMIN, XMAX, YMIN, YMAX, the axis limits; XLABL and YLABL the x and y-axis labels, respectively.

In addition to plots, if ISSW(10) = -1 TABLE outputs to unit KOUT the following list of variables in format (GF8.3):

PF, XPL, YPL, (VRBL(NV(K))), K = 1,6).

(See Table 6 for descriptions of these variables.)

If ISSW(12) = -1, a short list of variables is output to unit KLIST: pressure (PF), and the variables x, y and z, z given by

$$z = C1*VRBL(NZ1) + C2*VRBL(NZ2) + C3*VRBL(NZ3) + C4*C(IREC,3)$$

SUMMARY - PEPLT:PEPLS: KBR = 1

Function: Call subroutine TABLE - multiple station plots, map
format output, lists by station.

ISW, JSW options: ISW = 2 Initialize plot parameters
ISW = 1 Change plot, map
format and list
parameters
ISW = 0 Plot, list, map

format output

	JSW	No options
ISSW options:	ISSW(5) = -1	No interior axes on plot
	ISSW(6) = -1	No plot
	ISSW(10) = -1	List variables
	ISSW(12) = -1	List p, x, y, z.

- 2: Change data selection variables. Calls subroutine to change time and space windows and origin co-ordinates.
- 3: Change plot and list parameters. This branch prints a short documentation on the screen each time it is called. Parameters which may be changed and their descriptions are listed in Table 7. This branch has internal branches 1 through 8, which are prompted by '**: PARAMETERS: KBR3, ISW3, KX, MV, MW'. Only KBR3 and ISW3 have any effect in this branch. KX is the total number of parameter input branches (5). To return to PEPLS from branch 3 the user must enter KBR3 = 1, ISW3 = 0 followed by /. This will cause the new parameter values to be written on unit KLIST, and stored common to be written to KPLCM.DAT.

SUMMARY - PEPLT:PEPLS: KBR = 3

Function: Change or list plot and listing parameters

ISW, JSW options: None
Input device: KIN
Output device: KLIST
ISSW options: None

TABLE 7
PEPLT: PEPLS Branch 3 Parameters

<u>VARIABLE</u>	<u>DEFAULT</u>	<u>DESCRIPTION</u>
NX1	12	Variable codes for VRBL used in AVRGS and TABLE computations
NX2	0	
NX3	0	
NY1	19	
NY2	0	
NY3	0	
NZ1	25	
NZ2	0	
NZ3	0	
A1 B1 C1	1.	Scaling factors used in AVRGS and TABLE computations
A2-A6, B2-B6, C2-C6	0	
D1 to D6	1.	
TMIN to YT	None	Not used
SMIN to ST	None	Not used

- 4: Calls AVRGS subroutine. This subroutine calculates horizontal averages, allows operations such as vertical integration and column addition, multiplication, exponentiation and division. There are 23 internal branches in AVRGS, accessed with different values of ISW(0 to 22). These internal branches are described below, with a summary at the end of each. As an overview, AVRGS reads the requested data from *.AVG files into a two-dimensional array C(100,6). The rows (1 to $\leq$ 100) correspond to the pressure levels and the columns to variables requested by the user and computed in function subroutine VRBL (see PEPLS branch 1 for a description of VRBL). As each successive station is read, the elements of C are added to, forming sums of all data available at all levels. These sums must then be divided by the total number of observations at each level, to obtain the average values. For reasons of flexibility, the reading/summing and division are performed in separate ISW branches within AVRGS. Once the array C is filled (one column of which must be the number of observations) and averaged, then a number of operations can be performed on the averages. The remaining ISW branches of AVRGS are devoted to these operations.

AVRGS has its own prompt 'AVRGS:KBR,ISW,JSW,KLIST', and control does not return to PEPLS unless KBR = 4 ISW = 12 is accessed. Therefore, only four variables (or < 4 followed by a /) need be input following the AVRGS prompt. In order to keep track of the operations performed in AVRGS, if ISSW(2) = -1 the four parameters are written to unit 4 each time an AVRGS branch is accessed, along with other pertinent information. This ISSW option will not be noted in the summaries.

Branches in AVRGS: (ISW)

- ISW = 0: Prints short documentation on unit KTTX
 ISW = 1: Reads station data into C array. Variables corresponding to NV(JSW) to NV(KLIST) (maximum of six) are read into columns

JSW to KLIST of array C for ND number of stations from file JSHP.PTN (logical unit 12), starting with the first station in that file. All data between PMIN and PMAX is accessed for each station. The array C is stored to KPLCM.DAT before returning to the AVRGS prompt. If ISSW(15) = -1, the weights (WT) from JSHP.PTN file are used; otherwise a weight of 1. is used. Each element of C is a sum of

$$C(I\text{REC},I) = C(I\text{REC},I) + D(I)*WT*(AV*VRBL(NV(I))) \\ + (BV+CV*VRBL(NV(I))*VRBL(NX(I)))$$

The default parameters are set such that

$$C(I\text{REC},I) = C(I\text{REC},I) + WT*VRBL(NV(I))$$

Some of the parameters used by this branch may be changed in branch 3 of AVRGS, and some in branch 3 of PEPLS.

SUMMARY - PEPLT:AVRGS: KBR = 4: ISW = 1

Function: Read and store data to C array

JSW, KLIST options: JSW is first column, KLIST last column

Output device: Array is stored to KPLCM.DAT for emergency retrieval. No other output.

ISSW options: None

ISW = 2: Zeros columns JSW to KLIST of array C

ISW = 3: Changes or lists parameters. Parameters involved are listed in Table 8. JSW = 1 initializes the parameters (defaults also in Table 9) before allowing changes; JSW = 0 retains previous values. (The first access to this branch must initialize.)

SUMMARY - PEPLT:AVRGS: KBR = 4: ISW = 3

Function: Change parameters

JSW, KLIST option: JSW = 1 initializes

JSW = 0 prints current values

Output device: KTTX

ISSW options: None

ISW = 4: Average table: divide columns JSW to KLIST by column 6, which should have the number of observations at each level.

TABLE 8
PEPLT: AVRGS Branch ISW = 3 Parameters

<u>VARIABLE</u>	<u>DEFAULT</u> <u>(Initialized)</u>	<u>DESCRIPTION</u>
ND	1	Number of stations to be processed
NV(1)	51	Variables to compute for C array as VRBL(NV(I)) in column I. See Table 6 for VRBL codes.
NV(2)	68	
NV(3)	86	
NV(4)	87	
NV(5)	63	
NV(6)	-1	
JREF	50	Number of levels to be calculated
JMAX	55	Level number corresponding to reference pressure for integrations over pressure
NX(I), I=1,6	0	Optional additive quantities in C array element calculation (see text).
A1	1.	X and Y scaling factors for plots
A2	0.	
A3	0.	
B1	1.	Initialized when PEPLS is called by responding YES to 'Initialize common '
B2	0.	
NX1	12	Optional plot parameters (see text for AVRGS branch ISW = 7).
NX2	0	
NY1	19	
NY2	0	Initialized in PEPLS as above

ISW = 5: Add column JSW vertically, starting from level 2 and going to JMAX:

$$C(IREC,JSW) = C(IREC - 1,JSW)$$

ISW = 6: List C array to unit KLIST. Includes data label, parameters, level number and pressure, and C array.

ISW = 7: Plot one frame. Up to six curves allowed per frame. NCAR plot package outputs to unit 8 a file which must be read and translated into a plot by a Metacode translator. PEPLT may be run on any terminal, but the plot files may only be translated on graphics terminals and the Calcomp plotter. Instructions for running the translators are found at the end of this section of the report. The plot branch asks for the number of curves (default 1, maximum 6), the level number for the first point (default 1), the plot label, the minimum and maximum coordinates for x and y (unless the user opts to have the NCAR plot package compute the scales, by responding YES to the query 'Use default axis parameters?'), x and y axis labels, and the column number to be plotted.

The program actually plots:

$$x = B1*C(J,JSW) + B2*C(J,NX2) + B3*PF$$

$$y = A1*PF + A2*C(J,NV1) + A3*C(J,NY2)$$

The default values of the parameters plots

$$C(J,JSW) \text{ vs } PF \text{ (pressure).}$$

However, if for example the user wished to plot potential temperature θ vs salinity S, with θ (VRBL(65)) in column 1 and S(VRBL(61)) in column 2, then the values of the above parameters should be changed (using AVRGS branch 3)

$$A1 = 0. \quad B1 = 1 \quad NY1 = 1$$

$$A2 = 1. \quad B2 = 0$$

$$A3 = 0. \quad B3 = 0.$$

The y-axis runs backwards (maximum at the bottom to minimum at the top) unless A1 is equal to 0. An example is given in ENERGY.COM -- see Appendix A.

Characters of the user's choice which mark the actual data points may also be plotted if $ISSW(5) = -1$. Note should be made that these are not centered characters, so that the data point actually occurs wherever the plotter commences drawing the character.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 7

Function: Plot one frame containing up to six curves.

JSW option: JSW is the column number to be plotted. It may be changed while in the plotting branch.

Output device: Plot information goes to Metacode file, unit 8.

ISSW options: $ISSW(5) = -1$ plots character to mark actual data points. Character is requested while in plot branch.

ISW = 8: Calculates gravitational available potential energy per unit mass (GPE) and per unit area (TGPE), from the horizontal averaged steric volume $\overline{DVI}$ (VRBL(18)) in column 1 and for the reference steric volume DVF (VRBL(19)) in column 2, except for a constant of integration. GPE and TGPE relative to some reference pressure are calculated by subtracting from GPE and TGPE at each level the value at the level corresponding to the desired reference pressure (denoted by level number JREF) in AVRGS branch ISW = 10. GPE is stored in column 1, TGPE in column 2. The units are 10^{-4} J.kg⁻¹ and 10^{+4} J.m⁻², respectively.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 8

Function: Calculate GPE and TGPE except for a constant of integration

JSW options: None

Output device: None (GPE and TGPE replace $\overline{DVI}$ and DVF in columns 1 and 2, respectively, of array C.)

ISSW options: None

ISW = 9: Integrate over pressure columns JSW to KLIST. This is an alternate method for calculating GPE and subsequently TGPE,

with $\overline{DVI} - DVF$ (VR8L(51)) in column JSW. It may also be used to compute the compressibility effects in the GPE calculation (see equation 28 in Bray and Fofonoff, 1981).

The integration is performed starting with the first element in the column, and continuing to the last; the reference value must be subtracted in a separate operation, using AVRGS branch ISW = 10.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 9

Function: Integration over pressure (except for a constant) of columns JSW to KLIST

JSW options: Columns JSW to KLIST are integrated

Output device: None

ISSW options: None

ISW = 10: Subtract value at reference pressure (level corresponding to JREF) from all other elements in columns JSW to KLIST

Output device: None

ISSW option: None

ISW = 11: Add up to four scaled columns, according to

$J = IREC$

$$C(J, JC1) = CR1 * C(J, JC1) + CR2 * C(J, JC2) + CR3 * C(J, JC3) + CR4 * C(JREF, JC4)$$

If JSW = 1, JC1, CR1 to JC4, CR4 are entered; no addition is performed.

If JSW = 0, addition is performed using most recently input parameters.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 11

Function: Add up to four scaled columns, row by row

JSW Option: 0: perform addition

1: input scaling and column parameters

Output device: None

ISSW option: None

ISW = 12: Return to PEPLS

ISW = 13: Multiply up to three scaled columns, row by row according to

$$C(IREC,I) = CON1 * C(IREC,I) * \{CON2 * C(IREC,J) * [CON3 * C(IREC,K)]\}$$

 If I = -1 no operation is performed.
 If J = -1 then the expression in {} is set to 1; if
 K = -1, the expression in [] is set to one, allowing one,
 two or three scaled columns to be multiplied together. The
 parameters may be changed when the branch is accessed. The
 default values are I,J,K = -1; CON1, CON2, CON3 = 1.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 13

Function: Multiply up to three columns, row by row

JSW options: None

Output device: None

ISSW options: None

ISW = 14: Output in map format to unit KTO. Branch requests output
 file name and level number (JREC) desired. Variables output
 are:

IDSTN (station identifier: ship, station), XLAT, XLONG,
 (VRBL(NV(K)), K = 1,3), (C(JREC,K), K = 4,5)
 in format (1H ,A5,2(F8.2),5F(8.3)).

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 14

Function: Output in map format

JSW option: None

Output device: Unit KTO (may be changed in branch; default
 is 60)

ISSW options: None

ISW = 15: Not used

ISW = 16: Take any single column to any power, row by row. Operations
 are performed on the absolute value of all elements. If
 JSW = 1, exponent and column inputs are prompted. If
 JSW = 0, exponentiation is performed. The call to JSW = 0
 should immediately follow that to JSW = 1, as the variables
 used for exponent and column number are not unique to this
 branch.

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 16

Function: Exponentiation of a single column

JSW options: JSW = 0: Operation performed
 JSW = 1: Exponent and column entered

Output device: None

ISSW options: None

ISW = 17: Integration of error terms: interval pressure squared as the integration variable. This is intended for the calculation of measurement and finestructure errors in GPE and TGPE; as such it may be used on columns containing averaged values of VRBL (75 and 77 through 81) -- see Table 6. This branch uses the same algorithm as AVRGS branch ISW = 9, with ΔP^2 instead of ΔP as the integration variable. See AVRGS branch ISW = 9 for a summary.

ISW = 18: Writes into column 5 the difference in pressure between each pair of levels, beginning at the top.

ISW = 19: Exchange columns JSW and KLIST.

ISW = 20: Input a new single element of C. Branch prompts for column and row of element to be changed.

ISW = 21: Compute the standard deviation and store in column 1 of any quantity X for which $\bar{X}$ (the average value) is stored in column 4 and $\bar{X}^2$ in column 3.

ISW = 22: Compute the dynamic height for each station at any range of levels referred to level JREF and output in map format. Branch prompts for output device (default is 60), and level numbers (JREC1, JREC2) for dynamic height calculation. Reference level JREF may be changed in AVRGS branch ISW = 3. To calculate dynamic height NV(1) must be 18, NV(2) 19. Variables output are:

IDSTN (station identifier), XLAT, XLONG, Dynamic height (in dynamic centimeters), (NV(K), K = 3, 6).

Output occurs for ND stations, beginning with the first station in JSHP.PTN (unit 12).

SUMMARY - PEPLT:AVRGS: KBR = 4, ISW = 22

Function: Compute dynamic height relative to JREF for any range of pressure, for each of ND stations and output in map format. Four optional variables are also output, for the same range of pressure. An example command file, DYNHT.COM is found in Appendix A.

ISW options: None

Output device: Unit KTO (default 60; may be changed by the user when the branch is accessed).

ISSW options: None

PEPLT Branches (KBR), continued

- 5: Set values of elements in the ISSW array. Up to 16 inputs are allowed, each consisting of the element number followed by the element value (-1 or 0). Terminate before 16 by typing /.
- 6: Restart main program.
- 7: Exit program: a YES response to the branch query 'EXIT PROGRAM ' results in the execution of a FORTRAN stop. A NO response returns the PEPLS prompt.

Metacode Translators

The translators for the plot files (written to unit 8) created in AVRGS (branch ISW = 7) and TABLE (PEPLS branch KBR = 1) are device specific. That is, each graphics terminal has its own version. The CALCOMP high speed plotter has two versions: one with default plotting parameters, and one which allows the user to enlarge or stretch the plots, alter their distribution on the plotter paper, etc. The IMLAC and Tektronix terminals also have versions of the translator to allow plot previewing.

For all translators:

If the plot file was written to any other file than that named FOR008.DAT (via an ASSIGN statement before running PEPLT) then you must

assign that output file name to unit 8 before running the translators. For example, if your plot file is named PLOT.PPT, you must make the following assignment:

```
ASSIGN PLOT.PPT FOR008.
```

For the CALCOMP (both versions) you must also assign terminal TTA4: to FOR061:

```
ASSIGN TTA4: FOR061
```

Then

```
RUN MCTRNPLT (for MetaCode TRAnslator PLOT)
```

plots with default parameters, and

```
RUN MCTRNPLT2
```

prompts the user for changes in the plotting parameters before executing the plots. MCTRNPLT2 asks three questions: first, how many plots in the y-direction (across plotter)? The default is 1, and is retained if a / is entered. Second, what size shall the plots be? The default is 10 by 10 inches. The new dimensions are entered in inches, and need not be equal for x and y. Again a / retains the default values. Finally, the program asks for the distance between plots, in inches. The default is 2 inches in both x and y. All plots in the file assigned to unit 8 are plotted, sequentially.

For the Tektronix (or the IMLAC in Tektronix mode):

```
RUN MCTRNTEK
```

starts the plot previewer. If there is more than one plot, the program prompts for continuing to the next plot by asking 'Option ?' to which the user should respond C for continue, until all plots in the file assigned to unit 8 have been plotted.

For the IMLAC (recommended over the IMLAC in Tektronix mode, since it is simpler, and uses more of the screen):

```
RUN MCTRNDYN1
```

starts the plot previewer. This program also prompts for continuation if there is more than one plot.

This translator information is accurate as of December 1980. If you encounter difficulties you should refer to the current VAX manual.

4. Modification of POTEN to accept input CTD data in other than CTD78 disc format.

This section is intended as a guide to assist users who wish to use POTEN on CTD data with formats other than that read by the standard version. In this section the header information required by POTEN is described in detail, and the procedure for reading data is explained. The only subroutine which must be changed is DATA, providing that the input data is an even series in pressure with no gaps.

DATA requires the following header information for each station:

Description	Variable Name	Format
Ship Name	ISHP	A2
Cruise	ICRUIS	A3
Station	ISTAS	I3
Decimal Latitude (south negative)	XLAT	F
Decimal Longitude (west negative)	XLONG	F
Day	IDA	I2
Month	IMO	I2
Year (last two digits)	IYR	I2
Time (24 hour clock)	ISTME	I4
Station Label	LBBL(3)	3A4
Minimum Pressure	PMIN or IPR	F or I
Maximum Pressure	LPR	I

The CTDATA library subroutines not needed for formats different from the disc version of CTD78 are:

PVER	
CRUISE	Header Information
STATION	
DATIDX	Data Retrieval
GETDAT	

Also, the common file IDXREC.DIM should not be included in DATA -- see the statement INCLUDE 'IDXREC.DIM'. The variable LLREC is the total number of stations in the subindex directory; all statements in DATA and COMPS which refer to LLREC may be deleted. The data are stored in arrays PRESS and DATA.

Pressure is stored in PRESS,(#), temperature in DATA(1, #), salinity in DATA(2, #) with # the data scan number. Subroutine DATA must fill DATA and PRESS (all scans) when it is called for each station. Finally, DATA must return to COMPS the total number of data scans, JRMAL.

Stations are selected by the call to DATA in COMPS. The call is
CALL DATA (KST,1)

In COMPS, KST is the sequential number in the DO loop from ISW to JSW in branch 1 (or 13). If the input data is on magnetic tape, the user may wish to change the DO loop in COMPS to go from 1 to JSW: that is, start at the beginning of the tape and read through ISW stations.

The section of DATA in which the ship and cruise specification may be changed (NSW = 2) can be readily modified to accept similar information (in branch 13) pertinent to the user's input data.

The header information should be read in following statement #5, replacing the statements between #5 and #54. The data should be read in in statements which replace the calls to DATIDX and GETDAT.

Acknowledgements

Nick Fofonoff was responsible for early versions of most of the programs and subroutines documented here; his contribution to this work is gratefully acknowledged. Jerry Needell, Dan Georgi and Marie-Noelle Houssais used these programs, discovered errors, and suggested improvements. The manuscript was improved by constructive criticism from Bac-Lien Hua, and was typed by Mary Ann Lucas and Audrey Williams.

This work was supported by the Office of Naval Research under contracts N00014-76-C-197, NR083-400 and N00014-79-C-0071, NR083-004 and by NSF grant OCE-77-19403.

References

- Armi, L., and N. A. Bray, 1981. A standard analytic curve of potential temperature vs salinity for the western North Atlantic. Deep-Sea Res., submitted.
- Bray, N. A., and N. P. Fofonoff, 1981. Available potential energy for MODE eddies. J. Phys. Oceanogr., 11(1), 30-47.
- Bryden, H. L., 1973. New polynomials for thermal expansion, adiabatic temperature gradient and potential temperature of sea water. Deep-Sea Res., 20, 401-408.
- Fofonoff, N. P., 1977. Computation of potential temperature of seawater for an arbitrary reference pressure. Deep-Sea Res., 24, 489-491.
- Fofonoff, 1981. Algorithms for oceanographic computations to appear as a publication of SCOR Working Group 51.
- Fofonoff, N. P., and H. L. Bryden, 1975. Specific gravity and density of seawater at atmospheric pressure. J. Mar. Res. (Suppl.), 33, 69-82.
- Millard, R. C., Blumer, A., and N. Galbraith, 1978. A digital tape format for Woods Hole Institution CTD Data. W.H.O.I. Technical Report 78-43.
- Millero, F. J., Chen, C.-T., Bradshaw, A., and K. Schleicher, 1980. A new high pressure equation of state for seawater. Deep-Sea Res., 27A, 255-264.

Appendix A

Example command files for different routine
calculations using POTEN and PEPLT.

In this appendix documented command files which perform various routine calculations are listed. In the order in which they appear they are: ENERGY.COM, POTEN.COM, PEPLT.COM, TABLE.COM, and DYNHT.COM. Brief descriptions of these files are also found in Table 2. The files themselves contain detailed documentation. Example plots from ENERGY.COM are also included: see Figures 5 to 12. Example plots from TABLE.COM are found in Figure 4.

```

5  $! ***** ENERGY.COM *****
10  $! COMMAND FILE TO COMPUTE APE FROM CTD78 FORMAT DATA. CREATES
18  $! STANDARD PRINTOUT AND PLOTS. INTENDED AS AN AID IN LEARNING
26  $! TO USE THE VAX-11 PROGRAMS POTEN AND PEPLY.
34  $! 4 JULY 81. NAN BRAY
42  $!
50  $SET DEFAULT DBA2:<316316.LDE>
100 $ASSIGN JSHPF5.PTN FOR012 !FILE TO BE CREATED BY POTEN CONTAINING
150 $! STATION IDENTIFIERS.
200 $ASSIGN PRINT.PTN FOR004 !FILE FOR LINEPRINTER OUTPUT FROM POTEN
300 RUN/NODEFB POTEN
400 YES INITIALIZE COMMON
500 NO DO NOT INITIALIZE REGRESSION PARAMETERS
600 2.5 !PRESSURE INTERVAL FOR INPUT DATA
700 3.1,1.747 !SET PARAMETERS FOR REGRESSION
800 .....2./ !SET PDIFF TO 208--INTERVAL OVER WHICH TO,SO,DVO AVERAGED
900 / !DO NOT CHANGE DELP--PRESSURE INTERVAL FOR INPUT DATA
1000 NO DO NOT CHANGE REGRESSION PARAMETERS
1100 / !LEAVE TIME WINDOW AT DEFAULT 0-365 DAYS
1200 / !LEAVE EAST-WEST SPACE WINDOW AT DEFAULT: -180,180 DEG
1300 / !LEAVE NORTH-SOUTH SPACE WINDOW AT DEFAULT: -90,90 DEG
1400 YES CHANGE DATA LABEL
1500 LDE F55 TEST--STANDARD VERSION, NEW FDS--22 JULY 81
1600 11/ !SET ISSW ARRAY
1800 11,-17 !LIST STATION INFORMATION TO UNIT KLIST (FILE PRINT.PTN)
1850 11/
1900 12,0/ !DO NOT LIST REGRESSION SUMMARY AT EACH DEPTH FOR EACH STATION
1950 11/
2000 13,-1/ !CREATE *.REG,*.AVG FILES
2050 11/
2100 5,0/ !DO NOT LIST INDIVIDUAL INPUT DATA SCANS
2150 11/
2200 10,0/ !DO NOT LIST REGRESSION COEFFICIENTS FOR EACH LEVEL
2500 1,229,236,4/ !USING DEF CRUISE SPECS, REGRESS STA REF # ISW TO JSW
2660 13,201,208,4/ !CHANGE THE CRUISE SPECS, THEN AS IN PREVIOUS COMMAND
2675 W !SUBDIRECTORY VERSION
2690 IS001003 !SHIP, CRUISE, PROJECT NUMBER
2900 11/ !RESET ISSW ARRAY
3000 12,-17 !LIST AVERAGED REGRESSION COEFFICIENTS
3100 4,1,0,4/ !PROCEED THROUGH ENTIRE AVERAGING PROCESS. LISTS TO PRINT.PTN
3200 12/ !EXIT PROGRAM?
3300 YES
3400 $ASSIGN PRINT.PPT FOR004 !LISTING FILE FOR PEPLY OUTPUT
3500 $ASSIGN PLOT.PPT FOR008 !NCAR PLOT FILE FOR PEPLY PLOTS
3600 RUN/NODEFB PEPLY
3700 NO DO NOT READ IN PREVIOUSLY STORED COMMON
3800 YES INITIALIZE DATA SELECTION PARAMETERS
3900 5/ !SET ISSW ARRAY
4000 2,-17 !LIST OPERATIONS PERFORMED, IN PRINT:PPT
4050 5/
4100 5,-17 !PLOT CHARACTERS ON PLOTS FOR IDENTIFICATION
4200 4,2,1,6/ !ZERO C ARRAY IN AVRGs. PROGRAM CONTROL NOW IN AVRGs.
4300 4,3,17 !SET VARIABLE SELECTION PARAMETERS
4400 1000,7,.....,50/ !CHANGE 1000 TO # OF STATIONS IF NO < ALL
4500 / !DO NOT CHANGE AV THROUGH NX(1)
4600 / !DO NOT CHANGE A1 THROUGH B3
4700 / !DO NOT CHANGE NX1 THROUGH NY2
4800 4,1,1,6 !READ VARIABLES AS SELECTED INTO C ARRAY.
4900 4,4,1,5 !DIVIDE COLUMNS 1-5 BY 6(NUMBER OF STATIONS) TO AVERAGE
5000 4,6,1,4 !WRITE C ARRAY TO PRINT.PPT
5100 4,9,1,3 !INTEGRATE COLUMNS 1-3 WITH RESPECT TO PRESSURE

```

```

5200 4,10,1,3 !SUBTRACT FROM ALL LEVELS THE VALUE AT LEVEL JREF
5325 4,6,1,4 !WRITE C ARRAY TO PRINT.PPT
5400 4,11,07 !SUBTRACT FROM COL 1 COLS 2,3; ADD VALUE AT JREF FROM COL 4
5500 4,6,1,4 !WRITE C ARRAY TO PRINT.PPT
5600 4,11,17 !RESET ADDITIVE CONSTANTS
5700 1,1,2,1,3,1,4,0. !REPLACE COMPRESSIBILITY TERMS
5800 4,11,07/
5900 4,16,17/
6000 .5,5 !TAKE COLUMN 5 TO THE POWER .5
6100 4,16,07/
6200 4,6,1,4 !WRITE C ARRAY TO PRINT.PPT
6300 4,7/ !CALL PLOT BRANCH
6400 4,3 !4 PLOTS IN THIS FRAME, STARTING AT LEVEL 3 ON EACH
6500 YES INPUT NEW PLOT LABEL
6600 LDE F55 NEW EOS--22 JULY 81
6700 NO DO NO USE DEFAULT AXIS PARAMETERS
6800 -20,200,0,3000 !XMIN,XMAX,YMIN,YMAX
6900 YES CHANGE X-AXIS LABEL
7000 APE (CM/SEC)**2
7100 YES CHANGE Y-AXIS LABEL
7200 PRESSURE, DB
7300 1 !PLOT COL 1
7400 + !+ IS PLOT CHARACTER IDENTIFIER (NOT CENTERED!)
7500 2 !PLOT COL 2
7600 +
7700 3 !PLOT COL 3
7800 0
7900 4 !PLOT COL 4
8000 X
8100 4,7/ !CALL PLOT BRANCH FOR NEXT PLOT
8200 1,1/ !1 PLOT IN THIS FRAME; STARTING AT LEVEL 1
8300 NO DO NOT CHANGE PLOT LABEL
8400 NO DO NOT USE DEFAULT AXIS PARAMETERS
8500 0,100,0,3000
8600 YES CHANGE X-AXIS LABEL
8700 RMS DISPLACEMENTS, DB
8800 NO DO NOT CHANGE Y-AXIS LABEL
8900 5 !PLOT COL 5
9000 +
9100 4,2,1,5 !ZERO COLUMNS 1-5 OF C ARRAY
9200 4,3,07 !RESET SELECTED VARIABLE PARAMETERS, LEAVING OTHERS AS BEFORE
9300 ,50,61,65,64,19/
9400 /
9500 /
9600 /
9700 4,1,1,5 !READ VARIABLES INTO COLUMNS 1-5; START AT TOP OF JSHP.PTN LIS
9800 4,4,1,5 !DIVIDE COLUMNS 1-5 BY NUMBER OF STATIONS
9900 4,6,1,4 !WRITE C ARRAY TO PRINT.PPT
10000 4,12/ !RETURN CONTROL TO PEPLS
10100 5/ !SET ISSW ARRAY
10200 5,07 !NO CHARACTERS TO IDENTIFY PLOTS
10300 4,7/ !CALL PLOT BRANCH FOR NEXT FRAME; SEE EARLIER DESCRIPTION
10400 1,1/
10500 NO
10600 NO
10700 34.8,36.8,0,3000
10800 YF
10900 SALINITY, PPT
11000 NO
11100 2
11200 4,7/

```

11300	1.1/
11400	NO
11500	NO
11600	2.22,0.3000
11700	YE
11800	POTENTIAL TEMPERATURE, DEG C
11900	NO
12000	3
12100	4.7/
12200	1.1
12300	NO
12400	NO
12500	0.5,0.3000
12600	YE
12700	N, CPH
12800	NO
12900	4
13000	4.7/
13100	1.1
13200	NO
13300	NO
13400	40.220,0.3000
13500	YE
13600	DELTA-F, 1E-5 CM**3/GM
13700	NO
13800	5
13900	4.3,0/ !RESET SELECTED VARIABLE PARAMETERS
14000	/
14100	/
14200	0.,1./ !NOW GOING TO PLOT S(THETA) RATHER THAN S(P), CHANGE A1,A2
14300	,.3/
14400	4.7/
14500	1.1
14600	NO
14700	NO
14800	34.8,36.8,2.22
14900	YE
15000	SALINITY, PPT
15100	YE
15200	POTENTIAL TEMPERATURE, DEG C
15300	2
15400	4.7/ !CALL PLOT BRANCH FOR FINAL FRAME; DEEP THETA-S
15500	1.37
15600	NO
15700	NO
15800	34.93,35.03,2.6,4.8
15900	NO
16000	NO
16100	2
16200	4.12/ !RETURN CONTROL TO PEPLS
16300	7/ !EXIT PROGRAM?
16400	YES
16500	SPRINT/DEL PRINT.PTN,PRINT.PPT

LDE FSS TEST OF STANDARD VERSION

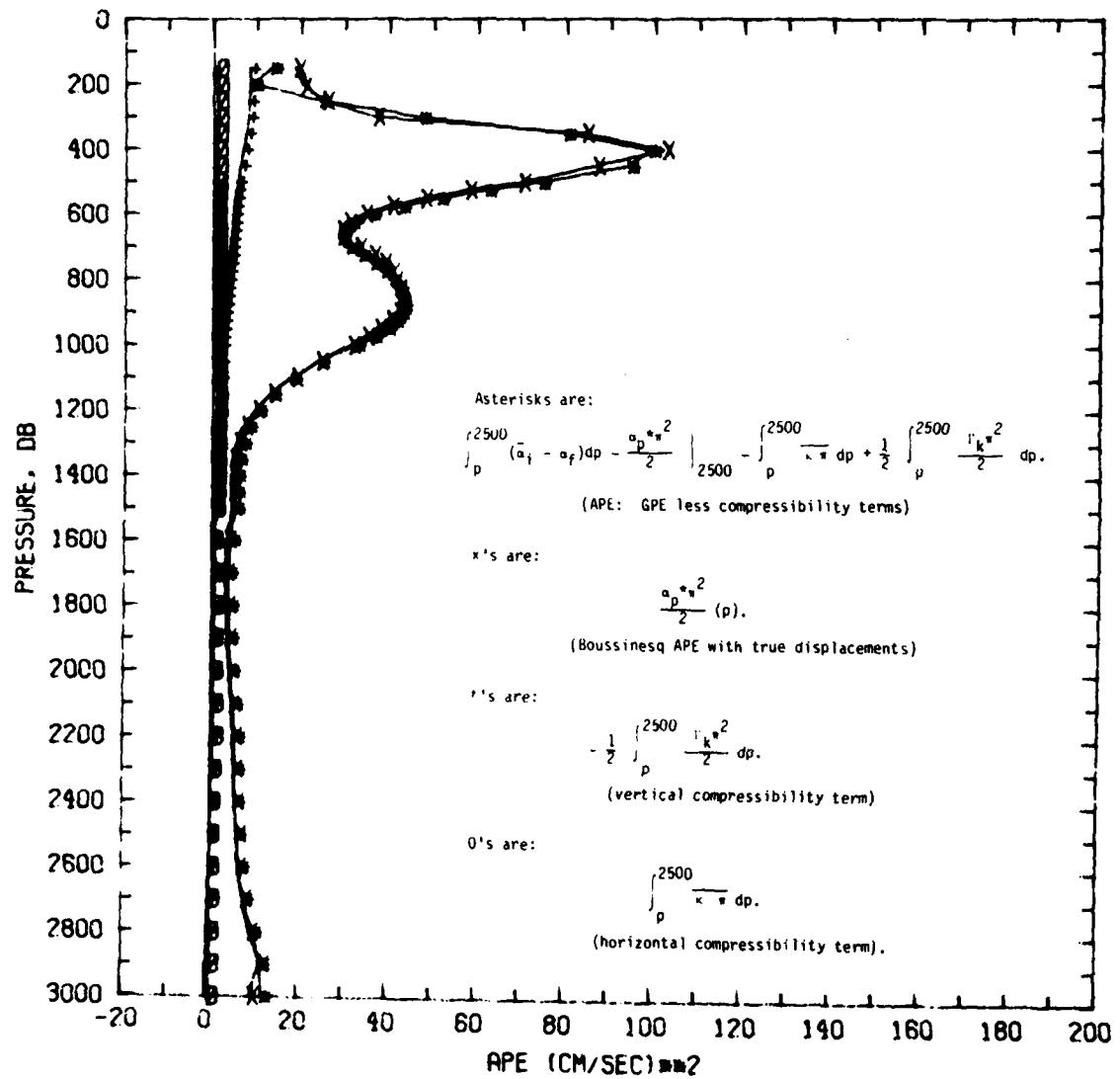


Figure 5: Example plot from ENERGY.COM: APE

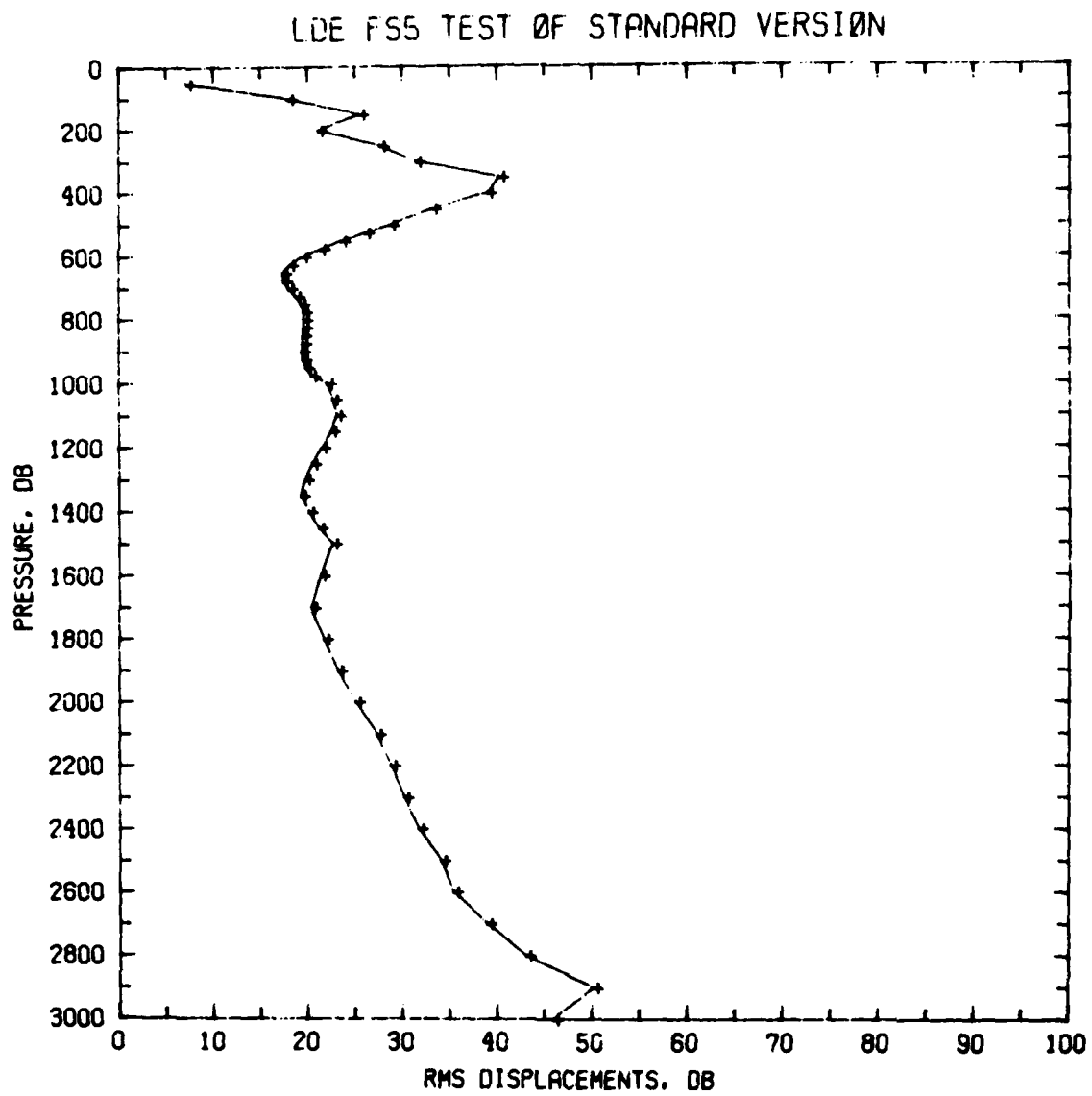


Figure 6: Example plot from ENERGY.COM.
Rms vertical displacements, π , in db.

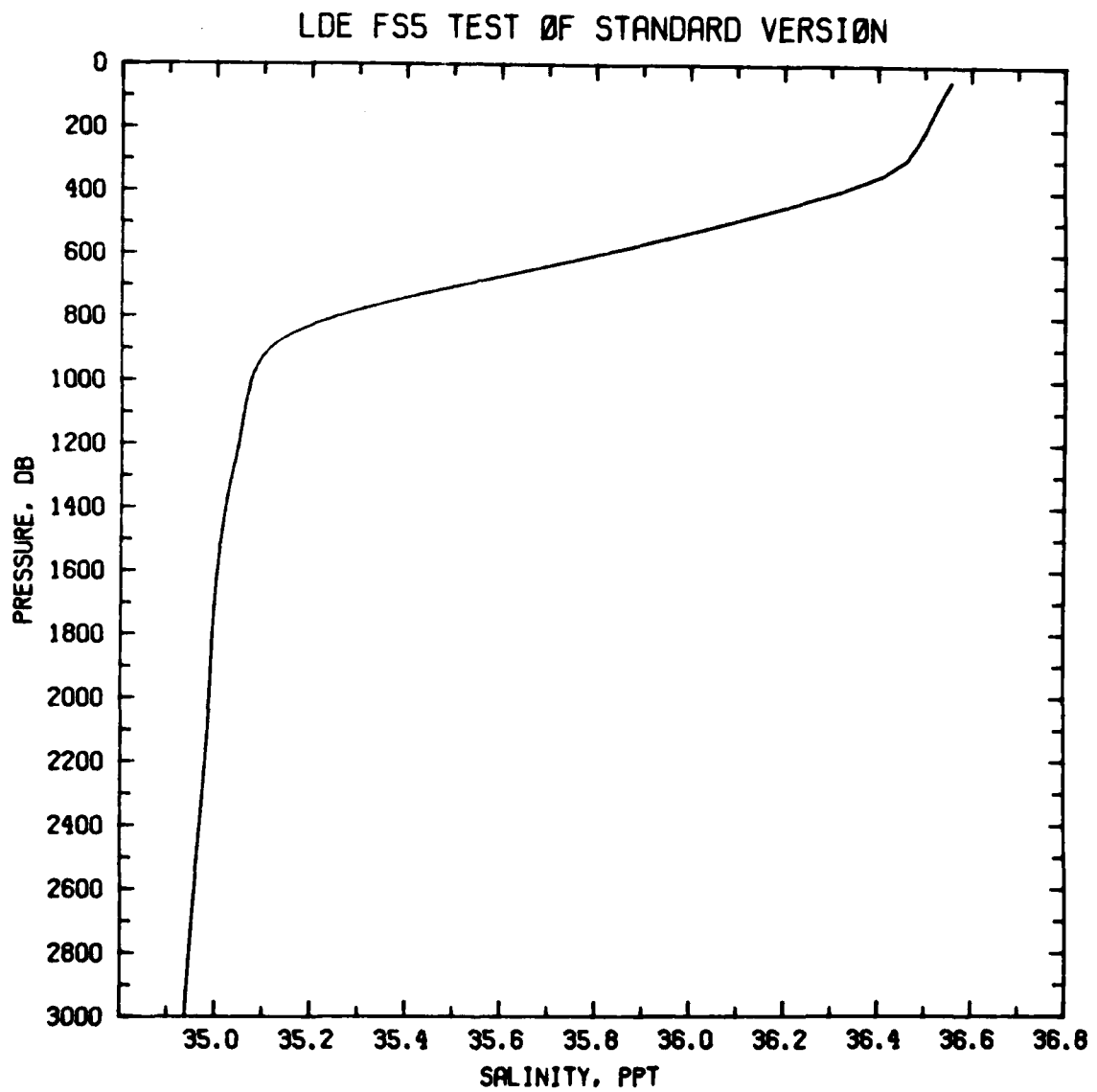


Figure 7: Example plot from ENERGY.COM
Averaged salinity in ppt along adiabatically leveled surfaces.

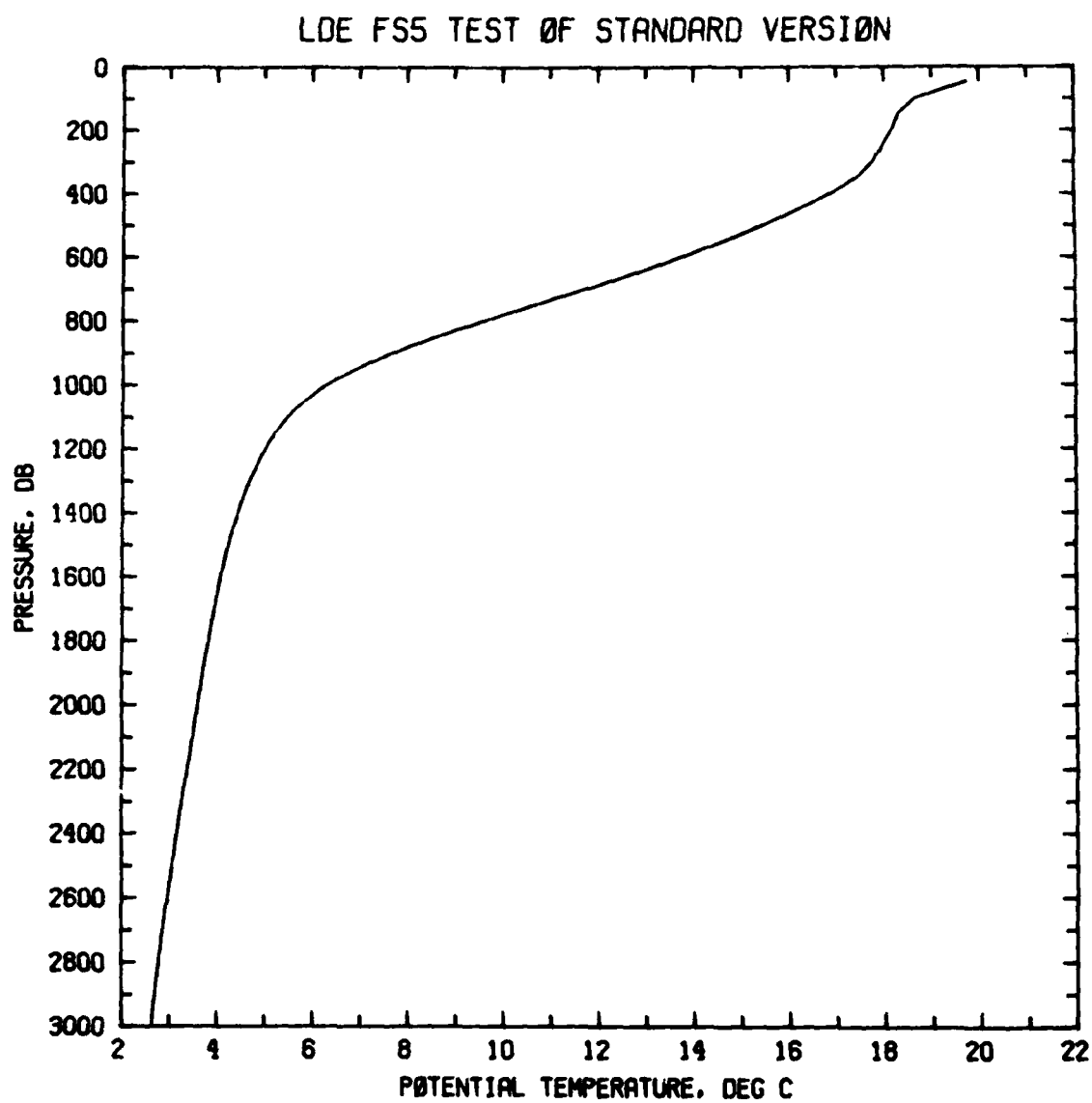


Figure 8: Example plot from ENERGY.COM
Potential temperature in °C averaged along adiabatically
leveled surfaces.

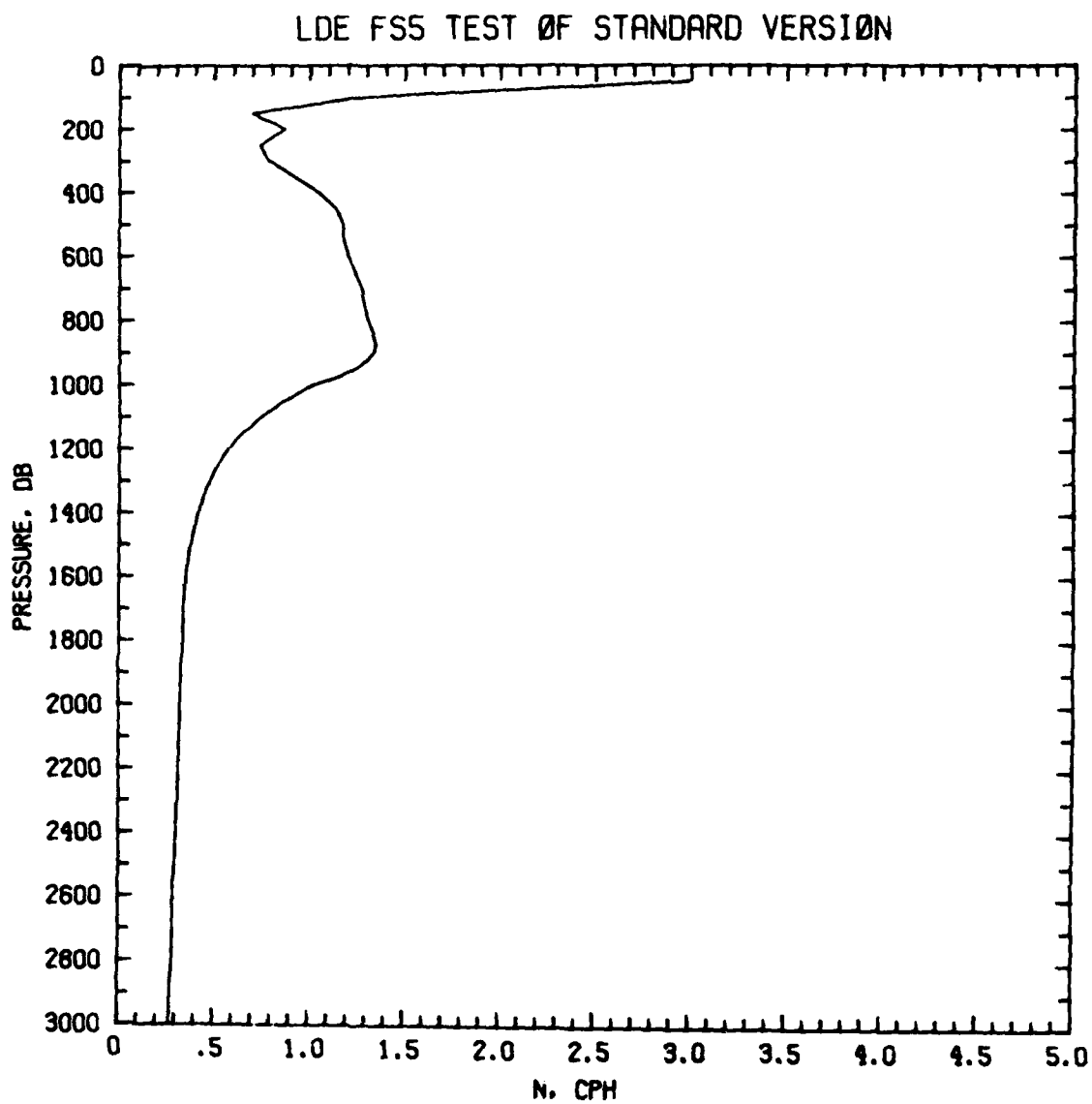


Figure 9: Example plot from ENERGY.COM
Buoyancy frequency N in cph averaged along adiabatically
leveled surfaces.

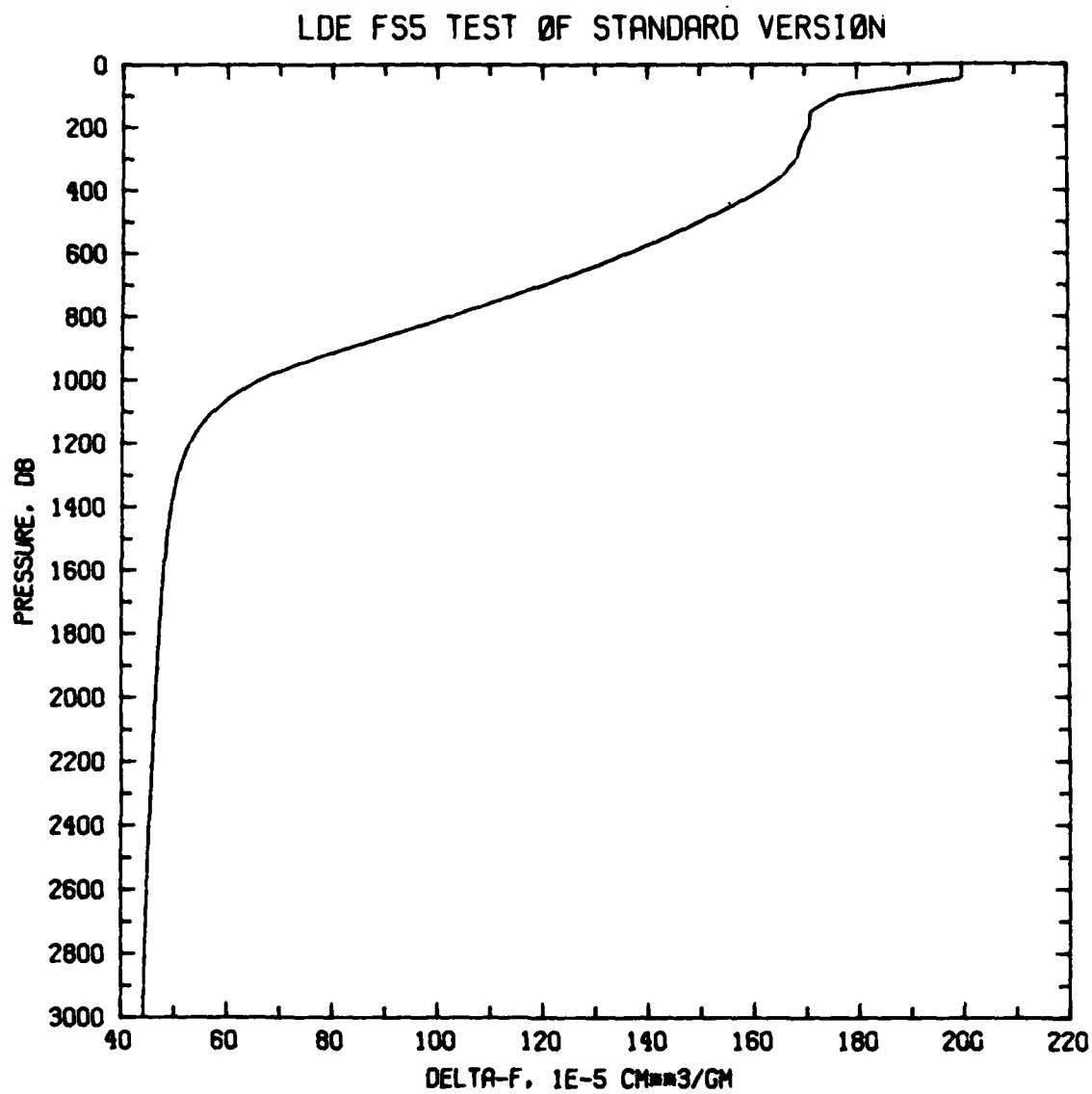


Figure 10: Example plot from ENERGY.COM
Reference (adiabatically leveled) steric anomaly in units of
 $10^{-5} \text{ cm}^3/\text{gm.}$

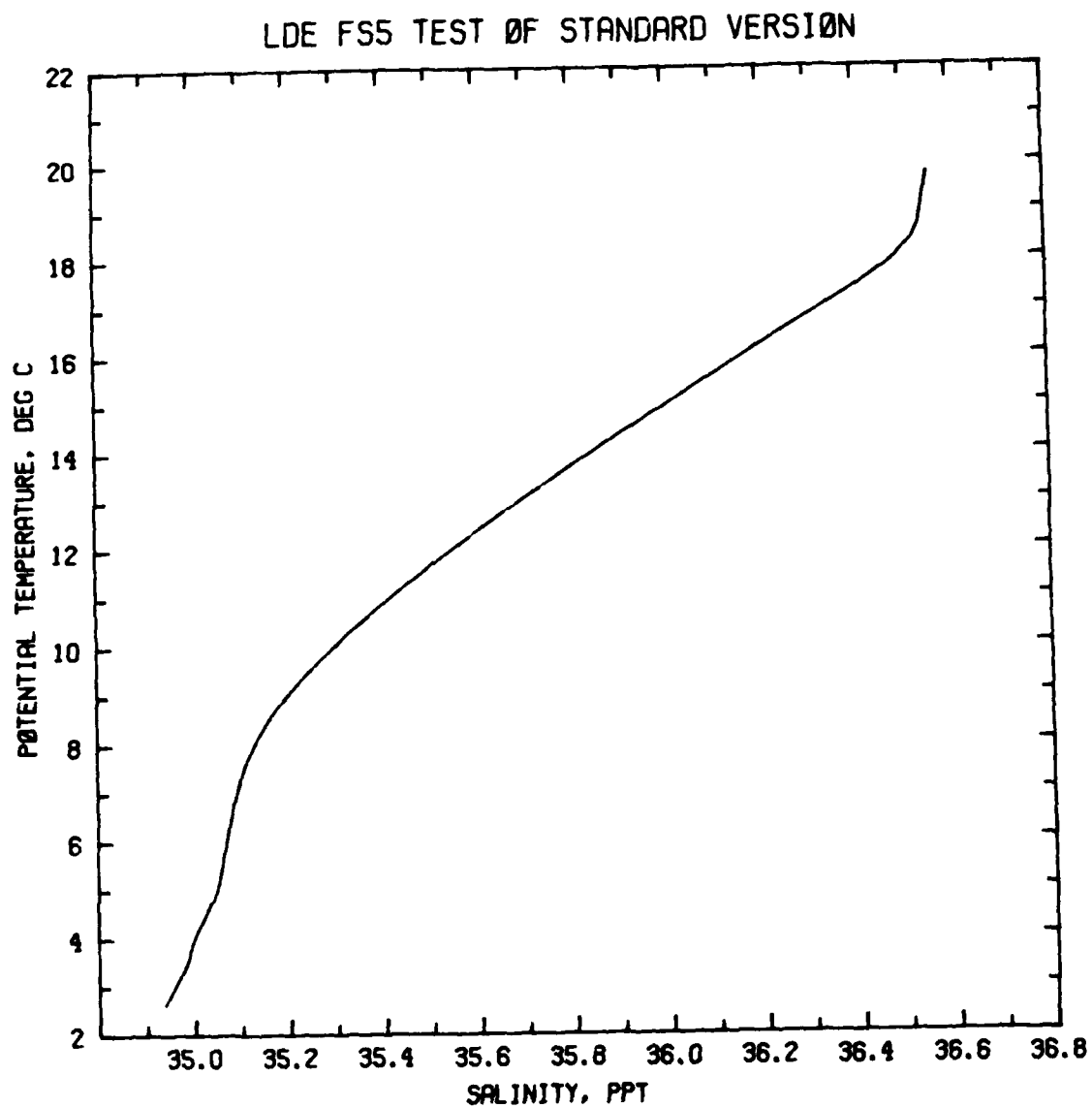


Figure 11: Example plot from ENERGY.COM
Potential temperature vs salinity computed as averages
along adiabatically leveled surfaces.

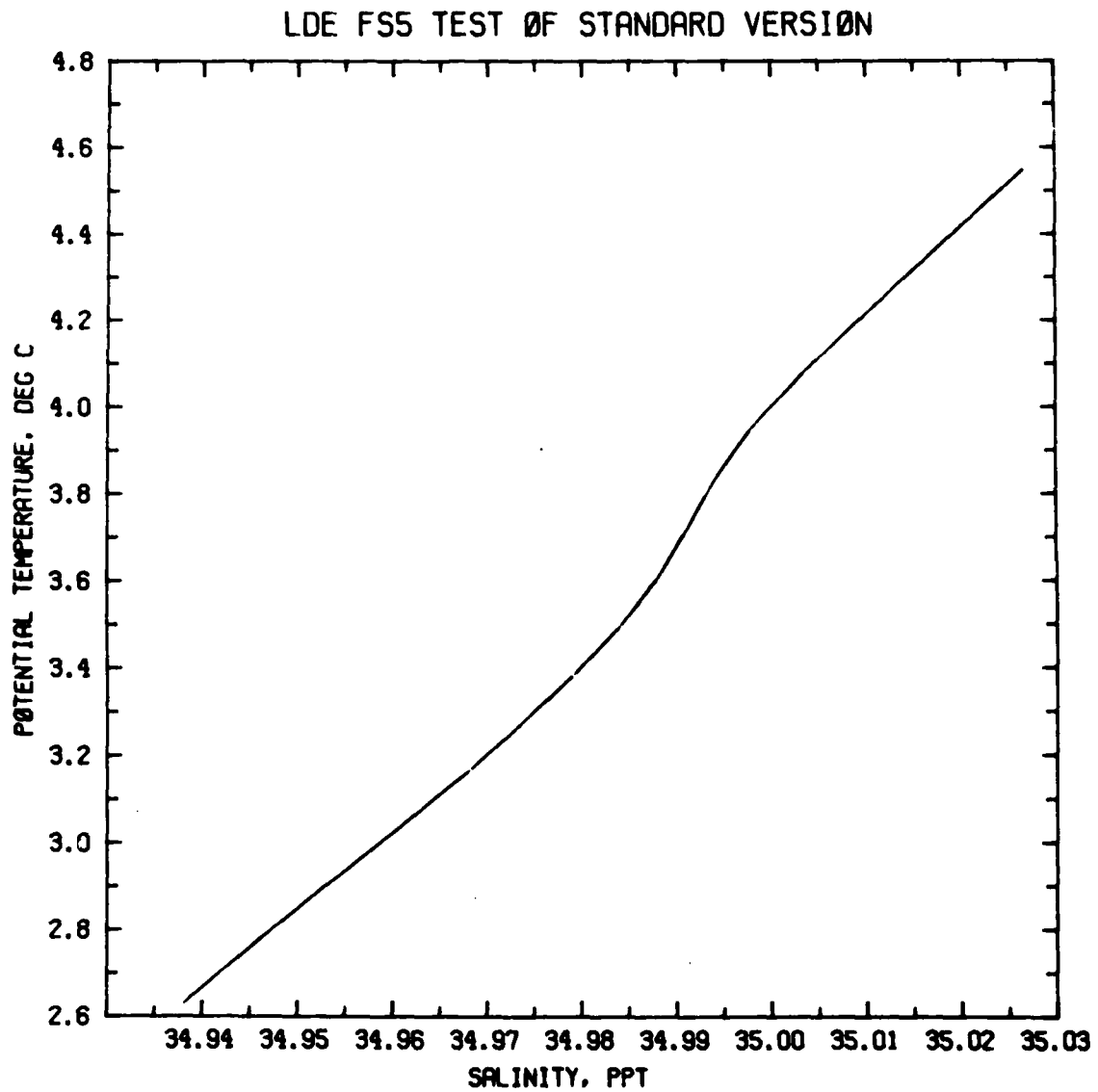


Figure 12: Example plot from ENERGY.COM
Deep potential temperature vs salinity.

```
50  $! ***** POTFN.COM *****  
100  $!TASK IS TO SET UP PARAMETERS FOR AN INTERACTIVE RUN OF POTEN  
110  $!FINAL INSTRUCTION IN THIS COM FILE CHANGES KIN TO 6, THEREBY  
120  $!RETURNING CONTROL TO THE TERMINAL.  
130  $!  
200  $ASSIGN PRINT.PTN FOR004  
300  $ASSIGN JSHPF55.PTN POR012  !CHANGE JSHPF55 TO CORRECT STATION LIST  
500  RUN/NOER POTEN  
600  YES INITIALIZE COMMON  
700  NO DO NOT INITIALIZE REGRESSION PARAMETERS  
716  2.5  !PRESSURE INTERVAL FOR INPUT DATA  
1200 0.000006/  !CONTROL NOW RETURNS TO TERMINAL FOR INTERACTIVE SESSION
```

```
5  S! ***** PEPLY.COM *****
10 S!TASK IS TO SFT UP PARAMETERS FOR AN INTERACTIVE RUN OF PEPLY
20 S!LAST STATEMENT IN THIS COM FILE CHANGES KIN TO 6, THEREBY
30 S!RETURNING CONTROL TO THE TERMINAL.
40 S!
100 SASSIGN JSHPF5.PTN FOR012 !CHANGE JSHPF5 TO CORRECT STATION LIST
125 SASSIGN PLOT.PPT FOR008 !PLOT.PPT IS METACODE FILE
150 SASSIGN PRINT.PPT FOR004 !PRINT.PPT IS LIST FILE
200 RUN/NOOER PEPLY
300 NO DO NOT READ IN PREVIOUSLY STORED COMMON
500 YES INITIALIZE DATA SELECTION PARAMETERS
600 4,2,1,6,,,6/ !CONTROL NOW RETURNS TO TERMINAL FOR INTERACTIVE SESSION
```

```

50      $! ***** TABLE.COM *****
100     $!TASK IS TO CREATE TWO PLOTS CORRESPONDING TO FIGURES
200     $!4 AND 5 IN BRAY(1981) BLUE COVER REPORT. THE FIRST
300     $!FIGURE IS BUOYANCY FREQUENCY N AS A FUNCTION OF DEPTH
400     $!AND POSITION. THE SECOND IS VERTICAL DISPLACEMENT.
500     $!BOTH ARE CREATED USING TABLE SUBROUTINE OF PEPLT;
600     $!THE FIRST FIGURE USES THE DEFAULT PLOT SPECIFICATIONS.
700     $ASSIGN PLOT.PPT FOR008
800     $ASSIGN JSHPF5.PTN FOR012
900     RUN/NODE9 PEPLT
1000    NO DO NOT READ IN PREVIOUSLY STORED COMMON
1100    YES INITIALIZE DATA SELECTION PARAMETERS
1200    1,2/  !INITIALIZE PLOT PARAMETERS IN TABLE
1300    16/   !ENTER NUMBER OF STATIONS IN PLOT
1350    3/    !3 IS FIRST LEVEL PLOTTED
1400    YES INPUT NEW PLOT LABEL
1500    LDF F55: N,CPH
1600    NO DO NOT USE DEFAULT AXIS PARAMETERS
1700    /     !USE THESE MIN AND MAX
1800    YES CHANGE X-AXIS LABEL
1900    KM EAST OF ORIGIN
2000    YES CHANGE Y-AXIS LABEL
2100    KM NORTH OF ORIGIN
2200    1,0/  !PLOT
2300    1,1/  !CHANGE PARAMETERS FOR SECOND PLOT
2400    16,,,100,3000/  !100 IS DISPLACEMENT AXIS IN DB
2450    3/
2500    YES CHANGE PLOT LABEL
2600    LDF F55: VERTICAL DISPLACEMENTS, DB
2633    NO DO NOT USE DEFAULT AXIS PARAMETERS
2666    /     !USE THESE MIN,MAX VALUES
2700    NO DO NOT CHANGE X-AXIS LABEL
2800    NO DO NOT CHANGE Y-AXIS LABEL
2816    4,3,0/  !CHANGE PLOT PARAMETERS--AVRGS
2832    /
2848    /
2864    .,05/  !RESCALE DISPLACEMENTS
2880    .,50/  !PLOT DISPLACEMENTS RATHER THAN N
2890    4,12/  !RETURN TO PEPLT
2900    1,0/  !PLOT
3000    7/     !EXIT PROGRAM
3100    YES

```

```

10  $! ***** DYNHT.COM *****
25  $! TASK IS TO CREATE GPCP COMPATIBLE OUTPUT FROM *.AVG FILES AT
31  $! SPECIFIED LEVELS PF. VARIABLES OUTPUT ARE STN ID, PF, XLAY,
37  $! XLONG, DYN HT REF TO PF AT LEVEL 50, TO, SO, REF SPECIFIC VOL, TIME
43  $! (JULIAN DAYS FROM 1 JAN+DECIMAL HOURS); FORMAT IS
46  $! 1H, A5, I4, 2F(8.2), 5F(8.3) 28 MAY 81 NAN BRAY
54  $! NUMBER OF STATIONS OUTPUT, OUTPUT VARIABLES, AND REFERENCE PRESSURE
58  $! FOR DYNAMIC HEIGHT MAY BE CHANGED IN AVRG5 BRANCH 3.
79  $! DYNAMIC HEIGHT OUTPUT IN DYNAMIC CENTIMETERS.
89  $!
100 $! ASSIGN JSHPF5.PTN FOR 12 !CHANGE JSHPF5 TO APPROPRIATE STATION LIST
200 RUN/NOPEB PEPLT
300 NO DO NOT READ IN INITIALIZED COMMON
500 YES INITIALIZE DATA SELECTION PARAMETERS
600 4,7,1,6/ !ZERO C ARRAY
700 4,3,1/ !SFT DATA VARIABLES
800 1000,18,19,13,14,19,7,50,55/
900 /
1000 /
1100 /
1200 4,22,1,6/ !CREATE FILES FOR EACH LEVEL REQUIRED
1250 YES INPUT NEW FILE NAME
1300 TEST.DAT
1400 15,17 !RANGE OF LEVELS FOR WHICH DH WILL BE OUTPUT
3400 4,12/ !RETURN CONTROL TO PEPLT
3500 7/ !EXIT PROGRAM?
3600 YES
3700 $SORT/KEY=(POSITION:7,SIZE:6) TEST.DAT TESTP.DAT !SORTS BY PRESSURE

```

Appendix B.
Program Listings for POTEN

50	POTEN:	PTENS:	SHORT DOCUMENTATION		
100	KBR	ISW	JSW	KLIST	DESCRIPTION
150	0	-	-	-	SHORT DOCUMENTATION
200	1	-	-	-	COMPUTE REGRESSIONS FOR ISW
300					SEQUENTIAL STATIONS.
400	2	-	-	-	INITIALIZE DATA SELECTION
500					PARAMETERS.
600	3	0	-	-	SET PARAMETERS: SHORT LIST.
700		1	-	-	SET PARAMETERS: FULL LYST.
800					STORES COMMON IN FILE KPTCH
900	4	1	-	LU	AVRCP AVERAGING SUBROUTINE.
1000	5	-	-	-	NOT USED
1100	6	-	-	-	LYST LABEL
1300	7	-	-	-	NOT USED
1400	8	-	-	-	LIST HEADER INFORMATION.
1500	9	-	-	-	LIST DATA RECORD.
1600	10	-	-	-	NOT USED
2000	11	-	-	-	SET ISSW (SWITCH) ARRAY.
2100	12	-	-	-	EXIT PROGRAM.

```

93  C COMPUTEN.FOR FILE: DIMENSION,COMMON AND EQUIVALENCE FOR POTEN
96  C AVAILABLE POTENTIAL ENERGY PROGRAMS. N.BRAY
98  C
100  PARAMETER KCM= 235
300  BYTE LB,PROVER
400  C
433  C DIMENSION
466  C
500  DIMENSION KHOGY(50),KBUF(46),VR(1)
600  DIMENSION KPTCM(KCM)
650  DIMENSION PRESS(3300)
700  C
716  C BLANK COMMON
732  C
750  COMMON KIN
775  C BEGINNING OF STORED COMMON
800  COMMON KTTX,KLIST,KOUT,KTP,TSW,JSW,KBR
825  C
850  C BEGINNING OF HEADER
875  C
900  COMMON LTYPE,MHDR,ICON,ISHP,KCAST,DAY,TPR,LPR
1000  COMMON XLAT,XLONG,WGT,XLTO,XLGO
1100  COMMON LBL(3),LBL(13),NSC(60),NPR(60),NSECTION
1125  C
1150  C BEGINNING OF DATA BUFFER KBUF
1175  C
1200  COMMON KTYPE,KBUF,IREC,N,NDP,KSW,L1,L2
1300  COMMON PF,TO,SO,DVO
1400  COMMON PI,THF,SF,DVF
1500  COMMON PH,THM,SH,DVM
1600  COMMON DH,PE,XPE
1700  COMMON CP(8),Z1,CT(8),Z2,F1,F2,F3
1712  C
1724  C END OF KBUF
1736  C
1750  COMMON DELP,DP
1800  COMMON A1,A2,A3,N1,N2,N3
1850  COMMON C(6),ISSW(16)
1856  COMMON ICRUIS,TPROJ,PROVER
1859  COMMON JMAX
1862  C
1874  C END OF STORED COMMON
1886  C
2000  COMMON P(3300),T(3300),S(3300),DV(3300)
2100  COMMON TH(3300),PT(3300),TT(3300)
2200  COMMON R(8),RP(8),BT(8),BA(8),CQ(36),MR(8)
2300  COMMON EX(6),JEX(6)
2400  COMMON WT(600),JSHP(600)
2425  COMMON DATA(3300,0:2)
2450  COMMON JSTN,JRMX,M1,M2
2475  COMMON LLREC,KKST
2500  C
2533  C EQUIVALENCE
2566  C
2600  EQUIVALENCE (KHOG,LTYPE),(KBUF,KTYPE)
2700  EQUIVALENCE (PDIF,A1),(VR,PF),(KTTX,KPTCM)
2750  EQUIVALENCE (PRESS,DATA)
2800  C

```

```

100  C  INDEX RECORD FIELDS DECOMPOSED
200  C  USES LABELLED COMMON : USE INCLUDE STATEMENT TO MERGE INTO PROGS.
300  C  RCH MAR 27 1980
400  C  ARRAY IDXREC CONTAINS THE INDEX RECORD
500  C  ALWAYS THE FIRST RECORD OF A DATA FILE
600  C  IDXREC HAS THE SAME STRUCTURE IN THE SUBINDEX FILE
700  C
800      COMMON/INDX/ IDXREC(256)
900  C*****
1000 C      LPGVER IS AN ASCII DESCRIPTION OF LOADING PROGRAM VERSION
1100      INTEGER LPGVER(4),LSTREC
1200 C*****
1300 C  LDATE AND LTIME CONTAIN ASCII DATE & TIME FILE CREATED
1400      INTEGER LDATE(3),LTIME(2)
1500 C  DEVICE AND FILSPEC FORM A COMPLETE FILE SPECIFIER
1600      INTEGER FILSPEC(8),DEVICE
1700 C  IFHED ARRAY HAS THE SAME STRUCTURE AS CT078 VAX FORMAT
1800      INTEGER IFHED(90),ICMNT(35)
1900 C  CNTRL HAS THE LOCATIONS OF THE BEGINNING WORD OF
2000 C  INFORMATION FIELDS OF THE INDEX RECORD
2100 C  1=CT078 HEADER 2=ABBREVIATED DATA DESCRIPTORS 3=TYPE 4=FILE SPEC
2200      INTEGER CNTRL(6)
2300 C  VARDES ARRAYS CONTAIN ABBREVIATED VARIABLE DESCRIPTORS
2400 C  MIN/MAX VALUES IVARDES CONTAINS MNEMONIC IDENTIFIERS
2500      DIMENSION VARDES(4,16),IVARDES(4,16)
2600      EQUIVALENCE(KEYWD,IDXREC(1))
2700      EQUIVALENCE(CNTRL(1),IDXREC(2))
2800 C  LSTREC IS THE NEXT AVAILABLE RECORD IN SUBINDEX FILE 1ST REC. ONLY
2900      EQUIVALENCE(LSTREC,IDXREC(9))
3000      EQUIVALENCE(IFHED(1),IDXREC(13))
3100      EQUIVALENCE (VARDES(1,1),IDXREC(115)),(IVARDES(1,1),IDXREC(115))
3200      EQUIVALENCE(KSCAN,IDXREC(105))
3300      EQUIVALENCE (RECLNG,IDXREC(107))
3400      EQUIVALENCE (MSCAN,IDXREC(106)),(MSCANS,IDXREC(112))
3500      EQUIVALENCE (PMIN,IDXREC(110)),(PRSINT,IDXREC(111))
3600      EQUIVALENCE (NTOT,IDXREC(108))
3700      EQUIVALENCE (IMPVAR,IDXREC(109)),(IPLOC,IDXREC(114))
3800      EQUIVALENCE
3900      1,(IFHED(3),YSHIP),(IFHED(4),ICRUZ),(IFHED(5),ISTAS)
4000      2,(IFHED(7),IYR),(IFHED(8),IMO),(IFHED(9),TDA)
4100      3,(IFHED(11),ILPSD),(IFHED(12),ILTSM)
4200      4,(IFHED(13),ILNSD),(IFHED(14),ILNSM)
4300      5,(IFHED(15),INWPS),(IFHED(16),IHRZ),(IFHED(17),IHRTZ)
4400      6,(IFHED(19),ILTED),(IFHED(20),ILTEM)
4500      7,(IFHED(21),ILNED),(IFHED(22),ILNEN)
4600      8,(IFHED(23),IETME),(IFHED(10),ISTME)
4700      9,(IFHED(38),ICAST),(IFHED(27),JDAY),(IFHED(28),INST)
4800      X,(IFHED(55),ICMNT)
4900 C*****
5000      EQUIVALENCE
5100      1 (DEVICE,IDXREC(193)),(FILSPEC(1),IDXREC(200))
5200      2 ,(LDATE(1),IDXREC(195)),(LTIME(1),IDXREC(196))
5300      3,(LPGVER(1),IDXREC(99))
5400 C  RMAX IS THE LAST RECORD OF A DATA FILE
5500 C  IDXLOC IS THE RECORD # OF THE INDEX FILE = 1 FOR SINGLE STATION FILE
5600      4,(RMAX,IDXREC(209)),(IDXLOC,IDXREC(208))
5700 C*****
5800 C*****
5900 C  END LABELLED COMMON FOR INDEX RECORD
6000 C*****

```

```

100 C POTEN MAIN PROG *****
200 PROGRAM POTEN
300 C *****
400 C
500 C PROGRAM TO COMPUTE REFERENCE SURFACES RELATIVE TO PF FOR
600 C CALCULATION OF AVAILABLE POTENTIAL ENERGY. REGRESSION
700 C FITS ARE MADE TO PRESSURE AND POTENTIAL TEMPERATURE AS
800 C FUNCTIONS OF SPECIFIC VOLUME ANOMALY AT PF.
900 C
1000 C JUNE 28 1976 N FOFONOFF
1100 C
1200 C INCLUDE 'COMPOTEN.FOR'
1300 C
1400 C OPEN BINARY FILE FOR STORAGE OF COMMON
1500 C
1600 C OPEN(UNIT=10,NAME='KPTCM.DAT',ACCESS='DIRECT',TYPE='OLD',
1700 C * RECORDTYPE='FIXED',RECORDSIZE=KCM,ERR=1100)
1800 C 1 CONTINUE
1900 C
2000 C KIN = 5
2100 C KTTY = 6
2200 C KLIST = 4
2300 C KOUT = 9
2400 C KTP = 1
2500 C 30 WRITE(KTTY,1000)
2600 C 1000 FORMAT(1H,'POTEN: POTENTIAL ENERGY PROGRAM')
2700 C CALL PTENS
2800 C GO TO 50
2900 C
3000 C CREATE NEW BINARY FILE FOR STORAGE OF COMMON IF NO OLD ONE EXISTS
3100 C
3200 C 1100 OPEN(UNIT=10,NAME='KPTCM.DAT',ACCESS='DIRECT',TYPE='NEW',
3300 C * RECORDTYPE='FIXED',RECORDSIZE=KCM,ERR=1100)
3400 C GO TO 1
3500 C 50 END

```

```

100  C PTENS SUBPROG POTEN ***** PTENS.FOR FILE *****
200  SUBROUTINE PTENS
300  C *****
400  C
500  C PROGRAM TO COMPUTE REFERENCE SURFACES RELATIVE TO PF FOR
600  C CALCULATION OF AVAILABLE POTENTIAL ENERGY. REGRESSION
700  C FITS ARE MADE TO PRESSURE AND POTENTIAL TEMPERATURE AS
800  C FUNCTIONS OF SPECIFIC VOLUME ANOMALY AT PF.
900  C
1000 C JUNE 28 1976 N FOFONOFF
1100 C
1200 C MODIFIED TO ACCEPT CTD78 VAX DISC DATA AS INPUT 15DEC80 N.BRAY.
1300 C
1400 C DIMENSION D(5),DOC(10)
1500 C
1600 C INCLUDE 'COMPOTEN.FOR'
1700 C
1800 C CHARACTER*8 DOC
1900 C
2000 C KIN = 5
2100 C KTTX = 6
2160 C WRITE(KTTX,40)
2220 C 40 FORMAT(1H,'INITIALIZE COMMON (YES OR NO)?')
2280 C IF(MOYES(KIN,KTTX).NE.1)GO TO 14
2340 C
2400 C INITIALIZE DATA SELECTION PARAMETERS
2480 C
2520 C CALL DATA(KTP,-1)
2580 C GO TO 30
2640 C 14 READ(10*1)KPTCM
2700 C 18 WRITE(KTTX,20)
2800 C 20 FORMAT(1H,'INITIALIZE REGRESSION PARAMETERS (YES OR NO)?')
2900 C IF(MOYES(KIN,KTTX).EQ.1)GO TO 15
3000 C READ(10*1,END=10)KPTCM
3200 C 5 WRITE(KTTX,25)
3300 C 25 FORMAT('WHAT IS THE RESOLUTION OF THE INPUT DATA, IN DB?')
3400 C READ(KIN,*)IDELP
3450 C KLIST = 6
3500 C 10 WRITE(KTTX,1005)KBR,ISW,JSW,KLIST,KOUT,KTP,KIN
3600 C 1005 FORMAT(1H,'POTEN:KBR,ISW,JSW,KLIST,KOUT,KTP,KIN',/,7I4)
3800 C READ(KIN,*)KBR,ISW,JSW,KLIST,KOUT,KTP,KIN
3900 C IF(KBR.GT.12)KBR=13
4000 C IF(KBR)1300,1300,12
4100 C 12 GO TO(100,200,300,400,500,600,703,800,900,1000,1100,1200,100,
4150 C *1300)KBR
4200 C
4300 C *****INITIALIZATION *****
4400 C 15 KTYPE = 0
4500 C MHDR = 150
4600 C MRUF = 46
4700 C NSECTION=4
4800 C NPR(1) = 4
4900 C NPR(2) = 12
5000 C NPR(3) = 17
5100 C NPR(4) = 24
5200 C NPR(5) = 50
5300 C NPR(6) = 90
5400 C NPR(7) = 100
5500 C NPR(8) = 200
5600 C NPR(9) = 500
5700 C NPR(10) = 500

```

```

5800      NPR(11) = 0
5900      NPR(12) = 2
6000      NPR(13) = 7
6100      NPR(14) = 13
6200      NPR(15) = 13
6300      NSC(1) = 0
6400      NSC(2) = 400
6500      NSC(3) = 1000
6600      NSC(4) = 1800
6700      NSC(5) = 3500
6800      NSC(6) = 6
6900      NSC(7) = 5
7000      NSC(8) = 4
7100      NSC(9) = 3
7200      NSC(10) = 3
7300      NSC(11) = 20
7400      NSC(12) = 30
7500      NSC(13) = 40
7600      NSC(14) = 50
7700      NSC(15) = 60
7710      PRNVER = 'W'
7720      ISHP = 'GY'
7730      ICRUIS = 1
7740      IPPDJ = 3
7750      GO TO 5
7800      30  DELP=2.
7900      DO 16 J=1,36
8000      16  VR(J) = 0.0
8100      PDIFF = 6.0
8200      A2 = 3.0
8300      A3 = 3.0
8400      LTYPE = 1
8500      ICMN = 0
8600      N = 2
8700      NDP = 10
8800      KSW = 1
8900      WGT = 1.0
8925      DO 17 J=1,16
8950      ISSW(J)=0
8975      17  CONTINUE
9000      GO TO 18
9100      C ***** SELECT DATA AND COMPUTE #1 *****
9200      100  CALL COMPS
9300      GO TO 10
9400      C
9500      C INITIALIZE DATA SELECTION PARAMETERS #2 *****
9600      C
9700      200  CALL DATA(KTP,-1)
9800      C *****SET PARAMETERS #3 *****
9900      300  WRITE(KLIST,3000)ICON,KSW,A2,A3,WGT,PDIFF
10000      READ(KIN,*)ICON,KSW,A2,A3,WGT,PDIFF
10100      WRITE(KLIST,3020)DELP
10200      READ(KIN,*)DELP
10300      IF(ISSW)10,10,310
10400      C
10500      C SUBROUTINE TO ACCEPT REGRESSION PARAMETERS IN ENGLISH AND
10600      C CONVERT TO POTEN PARAMETERS
10700      C
10800      310  CALL PARAM
10900      WRITE(KLIST,320)
11000      C PRINT OUT POTEN FORMAT PARAMETERS

```

		FORMAT('OSECTION LEVEL	LVEL	INDEX	START	# OF	# OF
	X	1H , 'NUMBER	NUMBER	INTERVAL	PRESSURE	TERMS	CYCLES')
11100		NSF=NSECTION					
11200		DO 330 I=1,NSF					
11300		I1=I+NSF					
11400		I2=I+2*NSF					
11500		WRITE(KLIST,335)I,NPR(I),NPR(I1),NPR(I2),NSC(I),NSC(I1),NSC(I2)					
11600		FORMAT(I3,6I8)					
11700	335	CONTINUE					
11800		C CHANGE OR LIST DATA SELECTION PARAMETERS					
11900		CALL DATA(KTP,0)					
12000	C						
12100		C CHANGE OR LIST DATA LABEL (IDENTIFIES THE DATA SOURCE)					
12200	C						
12300	C						
12400	C						
12500		WRITE(KTTY,3015)(LBL(I),I=1,13)					
12600		IF(NDYES(KIN,KTTY).EQ.1)THEN					
12700		WRITE(KLIST,3010)					
12800		READ(KIN,6005)(LBL(K),K=1,13)					
12900		ENDIF					
13000	C						
13100	C	STORE COMMON TO BINARY FILE KPTCH					
13200	C						
13250		IF(JSW.NE.2)THEN					
13300		WRITE(10*1)KPTCH					
13350		ENDIF					
13400		GO TO 10					
13500	C						
13600	3000	FORMAT(1H , 'ICON,KSW,SOP,SDT,NGT,PDIFF',/,2I4,3F6.2,F7.0)					
13700	3005	FORMAT(1H , 'MSC:P,N,NDP',/,5I5,10I3)					
13800	3006	FORMAT(1H , 'PRESSURE CONSTANTS',/,15I4)					
13900	3010	FORMAT(1H , 'INSERT LABEL <27 CHAR.')					
14000	3015	FORMAT(1H , 'INPUT NEW LABEL? OLD LABEL IS: ',/,2H ,13A4)					
14100	3020	FORMAT(1H , 'INPUT DATA RESOLUTION',/,F6.1)					
14200	C						
14300	C	*****AVERAGING SUBROUTINE #4 *****					
14400	400	CALL AVRCP					
14500		GO TO 10					
14600	C	***** #5 NOT PRESENTLY USED *****					
14700	500	GO TO 10					
14800	C	NGR = 5					
14900	C	KINP = 5					
15000	C	JMAX = 23					
15100	C	KOUT = 1					
15200	C	501 DO 505 M=9,13					
15300	C	DO 505 K = 1,100					
15400	C	505 CR(K,M) = 0.0					
15500	C	507 WRITE(KTTY,5010)KOUT,NGR,JMAX,KINP					
15600	C	5010 FORMAT(1H , 'AVDVF:KOUT,NGR,JMAX,KINP',/,4I4)					
15700	C	512 READ(KIN,*)KOUT,NGR,JMAX,KINP					
15800	C	DO 530 J=1,NGR					
15900	C	DO 520 JR=1,JMAX					
16000	C	READ(KINP,*)I,NST,KPR,(D(K),K=1,5)					
16100	C	IIPR(I) = KPR					
16200	C	DO 520 M=9,13					
16300	C	520 CR(I,M) = CR(I,M) + D(M-R)					
16400	C	530 CONTINUE					
16500	C	DO 540 J=9,13					
16600	C	DO 540 I=1,JMAX					
16700	C	540 CR(I,J) = CR(I,J)/FLOAT(NGR)					
16800	C	545 DO 550 I=1,JMAX					
16900	C	WRITE(KOUT,5000)I,NST,IIPR(I),(CR(I,K),K=9,13)					

```

17000 C 550 CONTINUE
17100 C      KOUT = 24
17200 C      READ(KIN,*)ISSW,KOUT
17300 C      GO TO (501,507,545,10)ISSW
17400 C5000  FORMAT(2I4,I5,5F10.4)
17500 C *****LIST DATA LABEL #6 *****
17600 C      600  WRITE(KLIST,6005)(LBL(K),K=1,13)
17900 C      GO TO 10
18000 C      6005  FORMAT(1H ,13A4)
18100 C ***** #7 NOT PRESENTLY USED *****
18200 C 703  MEOF = 0
18300 C 702  CALL READ(KOUT,KBUF,MBUF,IEOF)
18400 C      IF(IEOF)720,704,704
18500 C 704  MEOF = 0
18600 C 705  IF(KTYPE)900,900,710
18700 C 710  DO 715 M=1,MHDR
18800 C 715  KHDG(M) = KBUF(M)
18900 C      GO TO 800
19000 C 720  IF(MEOF)10,725,10
19100 C 725  MEOF = 1
19200 C      GO TO 10
19300 C      703  GO TO 10
19400 C *****LIST HEADER RECORD #8 *****
19500 C      800  WRITE(KLIST,8000)(LBL(K),K=1,3),XLAT,XLONG,XLTO,XLGO
19600 C      WRITE(KLIST,8005) LTYPE,MHDR,ICON,ISHP,ICAST,JDAY,IPR,LPR
19800 C      GO TO 10
20100 C      8000  FORMAT(7,3A4,4F8.3)
20200 C      8005  FORMAT(/,' TYPE MDHR ICON SHIP CAST JDAY IPR LPR',/,8I5)
20300 C *****LIST DATA RECORD #9 *****
20400 C      900  WRITE(KLIST,9000)IREC,PF,TO,SO,DVO,PM,THM,SM,DVM,Z1,Z2
20500 C      903  IF(ISSW(8))905,10,10
20600 C      905  WRITE(KLIST,9005)(CP(K),K=1,N)
20700 C      WRITE(KLIST,9010)(CY(K),K=1,N)
20800 C      GO TO 10
20900 C      9000  FORMAT(1H ,12,2(F7.1,F7.3,F7.3,F7.2),P6.2,F6.4)
21000 C      9005  FORMAT(1H ,3MCP ,6E11.4)
21100 C      9010  FORMAT(1H ,3MCT ,6E11.4)
21200 C *****MAG TAPE FUNCTIONS #10 *****
21300 C1000  CALL PTAPE(ISSW,JSW,KLIST)
21400 C      KLIST = 6
21500 C      1000  GO TO 10
21600 C5 ***** SET ISSW SWITCHES #11 *****
21700 C      1100  WRITE(KTTX,1150)(K,K=1,16),(ISSW(K),K=1,16)
21800 C      1150  FORMAT(2(1H ,X,16I4,/),' ENTER K,ISSW(K):')
21900 C      READ(KIN,*)(K,ISSW(K),M=1,16)
22000 C      GO TO 10
22100 C
22200 C ***** EXIT PROGRAM #12 *****
22300 C      1200  WRITE(KTTX,1210)
22400 C      IF(MOYES(KIN,KTTX).NE.1)GO TO 10
22500 C      STOP
22600 C      1210  FORMAT(1H , 'EXIT PROGRAM?')
22700 C** PTEN: SHORT DOCUMENTATION--BRANCH 0 *****
22800 C      1300  OPEN(UNIT=50,NAME='PTEN.DOC',TYPE='OLD',READONLY)
22900 C      DO 1350 N=1,200
23000 C      READ(50,1325,PND=1312)(DOC(I),I=1,8)
23100 C      WRITE(KTTX,1330)(DOC(I),I=1,8)
23200 C      1350 CONTINUE
23300 C      1312  CLOSE(UNIT=50)
23400 C      1325  FORMAT(8A8)
23500 C      1330  FORMAT(1H ,8A8)

```

23600
23700

GO TO 10
END

80

```

100  C COMPS SURPROG POTEN *****
200  C *****
300  SUBROUTINE COMPS
400  C *****
500  C
600  C TO COMPUTE REGRESSION COEFFICIENTS AT SPECIFIED DEPTHS.
700  C
800  C JUNE 28 1976 N FOFONOFF
900  C MODIFIED FOR CTD78 FORMAY INPUT DATA (VAX DISC VERSION) 15 DEC 80
1000 C N.BRAY
1100 C
1200      INCLUDE 'COMPOTEN.FOR'
1300  C
1400  C
1500  C IF OUTPUT IS REQUESTED THEN OUTPUT TO JSHP.PTN FILE THE
1600  C   NUMBER OF STATIONS
1700  C
1800      100 CONTINUE
2100  C
2200  C   INITIALIZE AND ACCESS INDEX FILE AND CRUISE INFORMATION
2300  C
2400      CALL DATA(KTP,2)
2500  C
2600  C   BEGIN COMPUTATION FOR ISW TOTAL STATIONS
2700  C
2800      IF (ISW.GT.LLREC) ISW=LLREC
2900      DO 106 KST=ISW,JSW
3000  C
3100  C   READ STATION HEADER FROM UNIT KTP AND CHECK IF IT MEETS
3200  C   SELECTION CRITERIA
3300  C   READ TEMPERATURE AND SALINITY DATA INTO DATA ARRAY.
3400  C
3500      101  CALL DATA(KST,1)
4000      GO TO 200
4100      106 CONTINUE
4200  C
4300  C   RETURN TO PTENS
4400  C
4500      295 RETURN
4600  C
4700  C   COMPUTE REGRESSION VERSION OF DATA
4800  C   IF ISSW(13)=-1 OUTPUT TO FILE *.REG
4900  C   MISCELLANEOUS INFORMATION MAY BE REQUESTED TO BE PRINTED
5000  C   TO FILE PRINT.PTN (KLIST=4) BY SETTING ISSW VALUES.
5100  C   SEE DETAILED WRITE UP.
5200  C
5300  C   KTYPE DISTINGUISHES BETWEEN HEADER AND DATA RECORDS:
5400  C       0=DATA, 1=HEADER.
5500  C   KF,KT,KM ARE INDICES
5600  C   N IS POLYNOMIAL ORDER
5700  C   NDP IS # OF DATA CYCLES OVER WHICH REGRESSION IS PERFORMED
5800  C   KERR COUNTS THE # OF REPLACEMENTS MADE BY SUBR EDIT
5900  C   IN EACH REGRESSION INTERVAL
6000  C   IPR AND LJP KEEP TRACK OF PRESSURE AS AN INDEX
6100  C   IREC INDEXES THE LEVELS PF
6200  C
6300      200  CONTINUE
6400      KF = 2
6500      N = NSC(NSECTION+1)
6600      NDP = NSC(2+NSECTION+1)
6700      XNDP = NDP

```

```

6A00
7000      KERR = 0
7100      IREC = 1
7400      C
7500      C  COMPUTE REGRESSIONS
7600      C
7633      C  JMAX IS THE TOTAL NUMBER OF LEVELS.
7666      C
7682      DO 2155 M = 1, JMAX
7714      P(M) = PRESS(M)
7730      T(M) = DATA(M, 1)
7746      S(M) = DATA(M, 2)
7778      2155 CONTINUE
7800      210 DO 270 J = 1, JMAX
7890      C
7900      C  SUBR JPR COMPUTES CORRECT PRESSURE PF GIVEN SECTION AND
8000      C  INTERVAL INFORMATION
8100      C
8200      IP = JPR(IREC, NPR, NSECTION)
8300      IF(IP.EQ.NSC(KF)) THEN
10500      2152      N = NSC(KF + NSECTION)
10600      NDP = NSC(KF + 2 * NSECTION)
10700      XNDP = NDP
10733      KF = KF + 1
10766      ENDIF
10800      IND = (IP - PRESS(1)) / DELP + 1
10900      M1 = IND - NDP / 2
11000      M2 = IND + NDP / 2 - 1
11020      IF(M1.LT.1) THEN
11040      M1 = 1
11060      M2 = NDP
11080      ENDIF
11100      IF(M2.GT.JMAX) GO TO 280
13400      C  IF PSSW(5) = -1 WRITE OUT SCAN #, SCALED PRESSURE, TEMP,
13500      C  SALINITY.
13600      C
13700      IF(ISSW(5)) 216, 217, 217
13800      216  WRITE(KLIST, 2160) (K, P(K), T(K), S(K), K = M1, M2)
13900      2160  FORMAT(IH, 14, 3F9.3)
14000      217 CONTINUE
14800      C
14900      C  PERFORM REGRESSIONS OVER INTERVAL CORRESPONDING TO PF
15000      C
15100      C  FIRST, FIND MEANS OF P, S.
15200      C
15300      220  PF = IP
15400      PM = 0.0
15500      SM = 0.0
15600      THM = 0.0
15700      DVM = 0.0
15800      TO = 0.0
15900      SO = 0.0
16000      DVO = 0.0
16100      XN = 0.0
16200      231 DO 230 M = M1, M2
16300      PM = PM + P(M)
16400      230  SM = SM + S(M)
16500      PM = PM / XNDP
16600      SM = SM / XNDP
16700      235 DO 250 M = M1, M2
16800      C

```

```

16900 C CALCULATE POTENTIAL TEMP AND SPECIFIC VOLUME ANOMALY
17000 C REFERRED TO PF.
17100 C
17200 2350 TH(M) = THETA(P(M),T(M),S(M),PF)
17300 DV(M) = DVA(PF,TH(M),S(M))
17400 2352 THM = THM + TH(M)
17500 DVM = DVM + DV(M)
17600 PT(M) = P(M)
17700 TT(M) = TH(M)
17800 DVX = DV(M)
17900 C
18000 C F1,F2 ARE MIN AND MAX SPECIFIC VOLUME ANOMALY WITHIN
18100 C THE REGRESSION INTERVAL.
18200 C
18300 IF(M-M1)236,236,237
18400 236 F1 = DVX
18500 F2 = DVX
18600 237 IF(DVX-F1)2372,238,238
18700 2372 F1 = DVX
18800 238 IF(F2-DVX)2382,239,239
18900 2382 F2 = DVX
19000 239 CONTINUE
19400 IF(ABS(P(M)-PF)-PDIFF)240,240,250
19500 C
19600 C AVERAGE T,S,DV OVER PF +/- PDIFF
19700 C
19800 240 T0 = T0 + T(M)
19900 S0 = S0 + S(M)
20000 DVO = DVO + DV(M)
20100 XN = XN + 1.0
20200 250 CONTINUE
20300 THM = THM/XNDP
20400 DVM = DVM/XNDP
20500 DVF = DVM
20600 T0 = T0/XN
20700 S0 = S0/XN
20800 DVO = DVO/XN
20900 C
21000 C CALL REGRESSION SUBROUTINE
21100 C
21200 2503 CALL LSFT
21300 C
21400 C IF ISSW(10)=-1 PRINT OUT REGRESSION COEFFICIENTS FOR THIS LEVEL
21500 C
21600 2507 IF(ISSW(10))251,253,253
21700 251 DO 2510 M=M1,M2
21800 DVI = DVA(P(M),T(M),S(M))
21900 PTD = PT(M) - PM
22000 TTD = TT(M) - THM
22100 2510 WRITE(KLIST,2511)M,P(M),TH(M),S(M),DVI,DV(M),PTD,TTD
22200 X *71,72
22300 2511 FORMAT(1H ,I4,F7.1,2F7.3,3F7.2,F7.3,X,F6.3,F7.4,2F3.0)
22400 WRITE(KLIST,2515)(CP(M),M=1,N),PM
22500 WRITE(KLIST,2515)(CT(M),M=1,N),DVM
22600 2515 FORMAT(1H ,6E11.5)
22700 C
22800 C IF # OF EDIT ERRORS IS LESS THAN 4, CHECK FOR ANY POINTS EXCEEDING
22900 C A2 TIMES THE STD DEV 71 (DEFAULT IS 3), AND EXCLUDE. RE-EDIT.
23000 C
23100 253 IF(KERR.GT.3)THEN
23133 WRITE(KLIST,25300)

```

```

23166 25300 FORMAT(1H,'KERR IS GREATER THAN 3--EDIT GIVES UP')
23177      GO TO 2536
23188      ENDIF
23194      IF(KERR)2536,2532,2532
23200 2532 DO 2535 M=M1,M2
23300      IF(ABS(PT(M)-PM)-A2*71)2535,2534,2534
23400 C
23500 C CALL EDITING SUBROUTINE
23600 C
23700 2534 CALL EDIT(KERR)
23750      TF(KERR)2537,220,220
23775 2537 IF(ISSW(3).EQ.-1)THEN
23800      DVI=DVA(PF,THETA(P(M),T(M),S(M),PF),S(M))
23825      WRITE(KLIST,25370)P(M),DVI
23890 25370 FORMAT(1H,'F9.1,F9.2,' FLAGGED IN COMPS, BUT NO
23875      X INTERPOLATION OF T OR S')
23883      ENDIF
23891      GO TO 220
23900 2535 CONTINUE
24000 2536 KERR = 0
24100 C
24200 C IF OUTPUT IS REQUESTED WRITE DATA BUFFER KBUF TO FILE *.REG
24300 C
24400      IF(ISSW(13))255,260,260
24500 255 WRITE(KOUT)KBUF
24600 C
24700 C IF ISSW(12)=-1 WRITE REGRESSION ESTIMATES TO UNIT KLIST
24800 C
24900 260 IF(ISSW(12))265,267,267
25000 265 WRITE(KLIST,2650)IREC,PF,TO,SO,DVO,DVM,SM,THM,71,72,N,NDP
25100 2650 FORMAT(1H,'14,F7.1,2F7.3,3F7.2,4F7.3,X,4B.3,F7.4,2I4)
25200 267 CONTINUE
25300      IREC = IREC + 1
25400 270 CONTINUE
25500 280 IF(ISSW(13))285,295,295
25600 285 CLOSE(UNIT=KOUT)
25700      GO TO 106
25800      END

```

```

100 C AVREP SUBPROG POTEN ***** AVREP *****
150 C *****
200 SUBROUTINE AVREP
300 C *****
400 C
500 C SUBROUTINE TO AVERAGE SPECIFIC VOLUME AND COEFFICIENTS.
600 C
700 C JUNE 28 1976 N FOFONOFF
750 C MODIFIED FOR VAX DISC CYD78 FORMAT 15DEC80 N. BRAY
800 C
900 C INCLUDE 'COMPOTEN.FOR'
1000 C
1100 C DIMENSION
1200 C
1300 DIMENSION CR(100,15),SWGT(100),CPM(8),SVAT(100)
1400 DIMENSION APF(100),SVI(100),EOB(100)
1500 DIMENSION VMIN(100),VMAX(100)
1600 C
1700 C CHARACTER
1800 C
1900 CHARACTER*12 FNAME(600),GNAME
2000 CHARACTER*1 IV1,IV2,IV3
2100 C
2200 C EQUIVALENCE
2300 C
2400 EQUIVALENCE (CR,IHDG),(PF,VR)
2500 EQUIVALENCE (CR(1,9),SWGT),(CR(1,10),SVAT),(CR(1,11),SVI)
2600 EQUIVALENCE (CR(1,12),APF),(VMIN,CR(1,12)),(VMAX,CR(1,13))
2800 C
2900 C READ IN STATION #'S TO BE AVERAGED. ENCODE INTO CORRESPONDING
3000 C FILE NAMES.
3100 C
3200 IF (ISW.EQ.1)THEN
3300 REWIND 12
3500 DO 61 K=1,1000
3600 READ(12,610,END=62)M,FNAME(K),WT(K)
3700 610 FORMAT(I4,A12,F5.2)
3900 61 CONTINUE
3916 62 CONTINUE
3932 JSTN=K-1
3950 GNAME(9:12)='AVG'
4000 ENDIF
4100 C
4200 C AVERAGING
4300 C
4400 100 ISW2 = ISW - 2
4500 IF (ISW2)101,113,113
4600 C
4700 C BRANCH 1--INITIAL AVERAGING--BEGINS HERE
4800 101 DO 110 J=1,100
4900 DO 110 I=1,13
5000 110 CR(J,I) = 0.0
5100 112 IRMX = 0
5200 C
5300 C BRANCHES 2 AND 3 BEGIN HERE
5400 C
5500 113 NST = 0
5600 C
5700 C OPEN APPROPRIATE FILE, READ HEADER
5800 C
5900 DO 1200 KK=1,JSTN

```

```

6000      OPEN(UNIT=KOUT,NAME=FNAME(KK),READONLY,TYPE='OLD',
6050      *   FORM='UNFORMATTED')
6100      READ(KOUT)KHDC
6200      NSW=5
6300      C
6400      C   CHECK IF DATA SELECTION PARAMETERS ARE SATISFIED. IF NOT,
6500      C   NSW IS RETURNED FROM DATA AS 4, AND STATON IS SKIPPED.
6600      C
6700      CALL DATA(KOUT,NSW)
6800      IF(NSW.NE.5)GO TO 1200
6900      NST=NST+1
7000      DELB=0.0
7100      PPR=0.0
7200      B2=0.0
7300      B3=0.0
7400      C
7500      C   IF ISSW(15)=-1 WEIGHTS ARE TAKEN FROM JSHP.PTN FILE; OTHERWISE,
7600      C   THEY ARE SET TO 1.
7700      C
7800      IF(ISSW(15))1350,1357,1357
7900      1350   WGT = WTKI
8000      GO TO 1370
8100      1357   WGT = 1.0
8200      1370   IF(ISW2)120,120,138
8300      C
8400      C   BRANCHES 1 AND 2 CONTINUE HERE FROM STATEMENT #1370
8500      C
8600      120   READ(KOUT,END=160)KBUF
8700      GO TO 140
8800      C
8900      C   BRANCH 3 (WRITE OUT AVERAGED FILES) CONTINUES HERE FROM STATEMENT #1370.
9000      C
9100      138   IF(ISSW(13))139,120,120
9200      139   CONTINUE
9300      C
9400      C   OPEN NEW FILE NAMED *.AVG CORRESPONDING TO INPUT *.REG, FOR OUTPUT
9500      C   ON UNIT 11.
9600      C
9900      GNAME(1:8)=FNAME(KK)(1:8)
10000     OPEN(UNIT=11,NAME=GNAME,TYPE='NEW',FORM='UNFORMATTED')
10100     C
10200     C   WRITE HEADER TO *.AVG
10300     C
10400     WRITE(11)KHDC
10500     GO TO 120
10600     C
10700     C   BRANCHES 1 AND 2 CONTINUE HERE FROM STATEMENT #120
10800     C
10900     140   IF(IREC-IRMX)146,142,142
11000     142   IRMX = IREC
11100     146   IF(ISW2)155,147,300
11200     C
11300     C   BRANCH 2 CONTINUES HERE FROM STATEMENT #146
11400     C
11500     147   CONTINUE
11600     1475  DEL = SVA(IREC) - DVM
11700     CALL COEFF(1.0,DEL,CP,CPH,N)
11800     CPM(1) = CPM(1) + PH - PF
11900     DO 150 J=1,N
12000     150  CR(IREC,J) = CR(IREC,J) + WGT*CPH(J)
12100     C

```

```

12200 C BRANCH 2 RETURNS TO READ NEXT DATA RECORD FROM *.REC
12300 C
12400 GO TO 120
12500 C
12600 C BRANCH 1 CONTINUES FROM STATEMENT #146
12700 C
12800 155 CONTINUE
12900 158 SVA(IREC) = SVA(IREC) + WGT*BVM
13000 VMIN(IREC) = VMIN(IREC) + WGT*F1
13100 VMAX(IREC) = VMAX(IREC) + WGT*F2
13200 SVI(IREC) = SVI(IREC) + WGT*DVZRD(PF,DVM,PH,CP,N,NDP,F1,F2;
13250 * ISHP,KCAST,ICON,DELP)
13300 SWGT(IREC) = SWGT(IREC) + WGT
13400 C
13500 C BRANCH 1 RETURNS TO READ NEXT DATA RECORD FROM *.REC
13600 C
13700 GO TO 120
13800 160 CLOSE(UNIT=KOUT)
13900 CLOSE(UNIT=11)
14000 IF(KK.LT.JSTN)GO TO 1200
14100 IF(IISW2)161,1610,420
14200 C
14300 C BRANCH 1 CONTINUES FROM PREVIOUS STATEMENT
14400 C
14500 161 DO 1605 J=1,IRMX
14600 VMIN(J) = VMIN(J)/SWGT(J)
14700 VMAX(J) = VMAX(J)/SWGT(J)
14800 SVI(J) = SVI(J)/SWGT(J)
14900 1605 SVA(J) = SVA(J)/SWGT(J)
15000 IF(IISW2)180,180,300
15100 C
15200 C BRANCH 2 CONTINUES FROM STATEMENT PRECEEDING #161
15300 C AVERAGE REGRESSION COEFFICIENTS
15400 C
15500 1610 DELB = 0.0
15600 PPR = 0.0
15700 KF = 2
15800 N = NSC(NSECTION+1)
15900 NDP = NSC(2*NSECTION+1)
16000 IF(IISW(11).EQ.-1)WRITE(KLIST,16230)
16100 DO 163 J=1,IRMX
16200 IF(JPR(J,NPR,NSECTION)-NSC(KF))1612,1611,1612
16300 1611 N = NSC(KF+NSECTION)
16400 NDP = NSC(KF+2*NSECTION)
16500 KF = KF + 1
16600 1612 DO 162 I=1,N
16700 162 CPM(I) = CR(J,I)/SWGT(J)
16800 PF = JPR(J,NPR,NSECTION)
16900 IF(IISW(11))1621,1624,1624
17000 1621 WRITE(KLIST,1623)PF,(CPM(I),I=1,N)
17100 1623 FORMAT(1H,PF,0.0,8G11.4)
17200 16230 FORMAT(1H,'AVERAGED REGRESSION COEFFICIENTS:',/,
17300 * 'PRESSURE',2X,'CP(1)',5X,'CP(2)',5X,'CP(3)',5X,'CP(4)',
17400 * 5X,'CP(5)',5X,'CP(6)')
17500 1624 DVI = SVA(J)
17600 DVF = DVZRD(PF,DVI,PF,CPH,N,NDP,VMIN(J),VMAX(J),ISHP,KCAST,IC
17700 DELA = SVI(J) - DVF
17800 SVA(J) = DVF
17900 DELPI = PF - PPR
17933 EOR(J) = DPDV(DVF,DVI,CPH,N,VMIN(J),VMAX(J))
17966 EOR(J) = 1./EOR(J)

```

```

18000      B2 = .50968E-6*(PF*DELA+PPR*DELB)*DELP1
18100      B3 = 0.5E-5*(DELA+DELB)*DELP1
18200      IF(J-1)1625,1620,1625
18300      1620  APE(1) = B2
18400      VMAX(1) = B3
18500      GO TO 1626
18600      1625  APE(J) = APE(J-1) + B2
18700      VMAX(J) = VMAX(J-1) + B3
18800      1626  DELB = DELA
18900      PPR = PF
19000      163  CONTINUE
19100      C
19200      C  BRANCHES 1 AND 2 CONTINUE HERE FROM STATEMENT #120; END QUALIFIER
19300      C  IMPLIED END OF STATION.
19400      C  IF ISSW(12)=-1 PRINT OUT AVERAGES
19500      180  IF(ISSW(12))182,400,400
19600      182  WRITE(KLIST,1832)
19700      1832  FORMAT(1H,'I REC NST PF',6X,'SUM OF',5X,'DVM',5X,'DVF',5X,
19800      * 'DV',5X,'DV',7,1H,19X,'WGT5',6X,'BAR',5X,'BAR',5X,'MIN',5X,
19900      * 'MAX')
20000      DO 183 I=1,IRMX
20100      KPR = JPR(I,NPR,NSECTION)
20200      IF(ISW.EQ.1)WRITE(KLIST,1835)I,NST,KPR,(CR(I,K),K=9,13)
20250      183  IF(ISW.EQ.2)WRITE(KLIST,1835)I,NST,KPR,(CR(I,K),K=9,11)
20300      1835  FORMAT(1H,'Z14,15,5F10.4)
20400      GO TO 400
20500      C
20600      C  BRANCH 3 CONTINUES HERE FROM STATEMENT #146
20700      C
20800      300  DVF = SVA(I REC)
20900      F3 = SVI(I REC)
21000      PI = POLY(DVF,DVM,CP,N,F1,F2) + PM
21100      IF(PI)301,302,302
21200      301  PI = 0.0
21300      302  THF = POLY(DVF,DVM,CT,N,F1,F2) + THM
21400      303  SF = DVZRO(PF,DVM,PM,CP,N,NDP,F1,F2,ISHP,KCAST,ICON)
21500      DELA = SF - DVF
21600      DELP1 = PF - PPR
21700      IF(I REC-1)305,304,305
21800      304  DELB = DELA
21900      305  B2 = B2 + 0.5E-5*(DELA+DELB)*DELP1
22000      B3 = B3 + 0.50968E-6*(PF*DELA+PPR*DELB)*DELP1
22100      DELB = DELA
22200      PPR = PF
22300      DM = EOB(I REC)
22400      PE = B3
22500      XPE = APE(I REC)
22600      C
22700      C  IF OUTPUT REQUESTED WRITE DATA TO FILE *.AVG
22800      C
22900      IF(ISSW(13))310,316,316
23000      310  WRITE(11)KBUF
23100      316  IF(JSW)317,320,3245
23200      C
23300      C  IF MAP FORMAT OUTPUT REQUESTED INITIALIZE AND REQUEST INPUT
23400      C
23500      317  IF(ISSW(14).EQ.-1)THEN
23600      3170  FORMAT(1H,'N1,N2,N3,KTO,IYR,ITH,IV1,IV2,IV3',6F5,3FX,A1))
23700      KTO = 1
23800      IYR = 73
23900      ITH = 0

```

```

24000      IV1='X'
24100      IV2='Y'
24200      IV3='Z'
24300      N1 = 1
24400      N2 = 2
24500      N3 = 3
24600      NV1 = 6
24700      NV2 = 4
24800      NV3 = 8
24900      WRITE(KTTX,3170)N1,N2,N3,KTO,IYR,ITH,IV1,IV2,IV3
25000      READ(KIN,*)N1,N2,N3,KTO,IYR,ITH
25100      READ(KIN,3175)IV1,IV2,IV3
25200      WRITE(KTTX,3176)NV1,NV2,NV3
25300      READ(KIN,*)NV1,NV2,NV3
25400      3175  FORMAT(3A1)
25500      3176  FORMAT(1H,'NV1,NV2,NV3',3I5)
25600      ENDIF
25700      JSW = 0
25800      C
25900      C  N1 TO N3 ARE EFFECTIVELY IGNORED, UNLESS ISSW(8)=-1. SEE STATEMENT #450.
26000      C
26100      320  IF(IREC-N1)321,490,321
26200      321  IF(IREC-N2)322,470,322
26300      322  IF(IREC-N3)120,324,120
26400      324  CONTINUE
26500      C
26600      C  SET VALUES FROM MAP FORMAT VARIABLES
26700      C
26800      3245  VR1 = VR(NV1)
26900          VR2 = VR(NV2)
27000          VR3 = VR(NV3)
27100      C
27200      C  IF ISSW(10)=-1 WRITE MAP VARIABLES TO UNIT KLIST (NOT IN MAP FORMAT--FOR
27300      C  CHECK PURPOSES ONLY).
27400      C
27500      325  IF(ISSW(10))326,327,327
27600      326  WRITE(KLIST,3150)ICDN,IREC,ISHP,ICAST,PF,PI,TO,VR1,VR2,VR3,
27700      X  DH,PE
27800      C
27900      C  IF MAP FORMAT NOT REQUESTED, RETURN TO STATEMENT #120 TO READ NEXT DATA
28000      C  RECORD.
28100      C
28200      C  MAP FORMAT WRITTEN TO UNIT KTO
28300      C
28400      327  IF(ISSW(14))330,120,120
28500      330  LFILE = 0
28600          IPF = PF
28700          XLG = XLONG
28800      335  WRITE(KTO,3300)(LBR(L),L=1,2),ISHP,ICAST,LFILE,IPF,
28900      X  XLAT,XLG,IYR,JDAY,ITH,IV1,VR1,IV2,VR2,IV3,VR3
29000      IF(LFILE)390,120,390
29100      3150  FORMAT(1H,'13,X,2I2,14,X,2F5.0,2F7.3,2F8.3,F6.3,F7.2)
29200      3300  FORMAT(12A4,11I4,X,11I5,F7.2,F8.2,X,12I4,12X,3(A1,F9.4))
29300      C
29400      C  ISSW(9)=-1 ALLOWS SPECIAL FUNCTIONS TO BE COMPUTED--SUBTRACTING VALUES
29500      C  AT ONE LEVEL FROM ANOTHER BEFORE OUTPUTTING IN MAP FORMAT.
29600      C
29700      450  IF(ISSW(8))455,324,324
29800      455  VR1 = VR(NV1)
29900          VR2 = VR(NV2)
30000          VR3 = VR(NV3)

```

```

30100      GO TO 120
30200      470  PF(ISSW(8))475,324,324
30300      475  VR1 = VR(NV1) - VR1
30400      VR2 = VR(NV2) - VR2
30500      VR3 = VR(NV3) - VR3
30600      GO TO 325
30700      C
30800      C  BRANCHES 1 AND 2 CONTINUE FROM 183 OR 180
30900      C
31000      C  IF ISSW(7)=0 CONTINUE THROUGH BRANCH 3 AUTOMATICALLY.
31100      C
31200      400  IF(ISSW(7))550,410,410
31300      410  ISW = ISW + 1
31400      IF(ISW-3)100,100,420
31500      C
31600      C  BRANCH 3 CONTINUES HERE FROM STATEMENT PRECEDING #161
31700      C
31800      C  IF MAP OUTPUT REQUESTED SET LFILE TO 1 TO INDICATE EOF IN MAP FORMAT.
31900      C
32000      420  IF(ISSW(14))422,560,560
32100      422  LFILE = 1
32200      GO TO 335
32300      550  CONTINUE
32400      1200 CONTINUE
32450      IF(NSW.NE.5.AND.ISW.EQ.1)GO TO 161
32475      IF(NSW.NE.5.AND.ISW.EQ.2)GO TO 1610
32487      IF(NSW.NE.5.AND.ISW.EQ.3)GO TO 420
32500      560  RETURN
32600      END

```

```

200 C DATA SUBR POTEN ***** DATA *****
500 C *****
600 SUBROUTINE DATATKUN,NSW)
700 C *****
800 C
900 C TO SELECT AND ACCESS CT078 FORMAT DATA FROM VAX DISC FORMAT
1000 C ACCESSES VARIOUS MILLARD SUBROUTINES FOUND IN CT078ZLTR
1100 C
1200 C
1300 C JAN 6 1976 N FOFONOFF
1400 C MODIFIED FOR CT078 FORMAT INPUT 15 DEC 80. N BRAY
1500 C
1550 INCLUDE 'COMPOTEN.FOR'
1600 C
1700 C INCLUDE MILLARD DIMENSION STATEMENTS
1800 C
1900 INCLUDE 'IOXREC.DIM'
2100 C
2200 C CHARACTER
2300 C
2400 CHARACTER*17 GNAME
2500 C
2600 C PROGRAM
2700 C
2800 IF(NSW.EQ.5) GO TO 80
2900 IF(NSW.EQ.2) GO TO 30
3000 IF(NSW)1,20,5
3100 C
3200 C NSW LESS THAN ZERO: INITIALIZE SELECTION PARAMETERS
3300 C
3400 I CONTINUE
3500 JDO = 0
3600 DAY1 = 0.
3700 DAY2 = 365.
3800 XEMN = -180.0
3900 XEMX = 180.0
4000 XNMN = -90.0
4100 XNMX = 90.0
4200 XLTO = 40.00
4300 XLGO = 70.00
4400 JSTN = 1
4500 RETURN
4600 C
4700 C NSW=0: LIST OR CHANGE SELECTION PARAMETERS
4800 C
4900 20 CONTINUE
5000 172 WRITE(KLIST,173)DAY1,DAY2,JDO
5100 173 FORMAT(1H,5HDAY1:F8.3,X,5HDAY2:F8.3,X,4HJDO:,14)
5200 READ(KIN,*)DAY1,DAY2,JDO
5300 174 WRITE(KLIST,175)XEMN,XEMX,XNMN,XNMX
5400 175 FORMAT(1H,7HEMN LTR,4F7.2)
5500 READ(KIN,*)XEMN,XEMX,XNMN,XNMX
5600 WRITE(KLIST,177)XLTO,XLGO
5700 177 FORMAT(1H,8HORGIN: ,2(X,F8.3))
5800 READ(KIN,*)XLTO,XLGO
5900 RETURN
6000 C
6100 C NSW = 2: READ FROM FILE STATIONS.PTN INFORMATION TO IDENTIFY
6200 C STATIONS
6300 C
6400 30 CONTINUE

```

```

6450      IF (KBR.EQ.13) THEN
6500      WRITE(KTTX,310)
6600      READ(KIN,300) PROVER
6607      WRITE(KTTX,320)
6614      READ(KIN,330) ISHP,ICRUIS,IPROJ
6650      300  FORMAT(A)
6662      310  FORMAT(1H,'ENTER SUBDIRECTORY VERSION #')
6674      320  FORMAT(1H,'ENTER SHIP CODE,CRUISE #,PROJ #')
6686      330  FORMAT(AZ,Z19)
6693      ENDIF
6700      C
6800      C MILLARD HEADER RELATED SUBROUTINES
6900      C
7000      CALL PVER(PROVER)
7100      CALL CRUISE(ISHP,ICRUIS,IPROJ)
7150      CALL STATION(0,0,KTP)
7200      CALL INDEX(11)
7300      LREC = IDXREC(9)
7325      LLREC = LREC
7337      KKST = 0
7400      M=0
7500      RETURN
7600      C
7700      C NSW = 1: READ STATION HEADER, CHECK AGAINST DATA SELECTION CRITERIA,
7800      C AND READ TEMPERATURE AND SALINITY INTO DATA ARRAY.
7900      C
8000      5  CONTINUE
8300      IF (KUN.GT.LLREC) GO TO 620
8800      CALL RECTDX(KUN)
9200      XLAT=SLAT()
9300      XLONG=SLNG()
9316      IPR = PMIN
9333      XN = NTOT-1
9366      LPR = XN+PRSINT+PMIN
9400      LBBL(1)=IFHED(3)
9500      ENCODE(4,53,LBBL(2)) IFHED(4)
9600      ENCODE(4,54,LBBL(3)) IFHED(5)
9700      53  FORMAT(I3,'-')
9800      54  FORMAT(I4)
9900      C
10000     C COMPUTE JULIAN YEAR DAY
10100     C CHECK AGAINST SELECTION PARAMETERS
10200     C
10300      IDAY=KDAY(IDA,IMO,IYR)-KDAY(31,12,IYR-1)
10350      DAY = FLOAT(IDAY) + FLOAT(ISTHE)/2400.
10400      IF (DAY-DAY1)620,602,602
10500      602 IF (DAY-DAY2)604,604,620
10600      604 CONTINUE
10700      C
10800      C CHECK LAT AND LONG AGAINST SELECTION PARAMETERS.
10900      C
11000      IF (XLONG-XEMN)620,606,606
11100      606 IF (XLONG-XEMX)608,608,620
11200      608 IF (XLAT-XNMN)620,610,610
11300      610 IF (XLAT-XNMX)616,616,620
11400      616 LTYPE = 1
11500      ICON = ICAST
11600      DAY = DAY + JDO
11700      KCAST = ISTAS
11750      JPMAX = NTOT
11800      C

```

```

11900 C IF YSSW(11)=-1 WRITE OUT HEADER INFORMATION ON UNIT KL1ST
12000 C
12100 IF(YSSW(11)/6160,620,620)
12200 6160 WRITE(KLIST,8000)(LBB(L(K),K=1,3),XLAT,XLONG,XLTO,XLGO
12300 WRITE(KLIST,8005) LTYPE,MHDR,TSHP,ISTAS,PCON,DAY,IPR
12400 C
12500 C IF YSSW(12)=-1, WRITE OUT HEADINGS FOR OUTPUT WHICH IS WRITTEN IN COMPS.
12600 C
12700 IF(YSSW(12).EQ.-1)WRITE(KLIST,8010)
12800 8010 FORMAT('OIREC PF TO SO DVO DVM SM TM'
12900 X 71 22 N NDP')
13000 8000 FORMAT(1,1H,3A4,4F8.3)
13100 8005 FORMAT(1H,3 TYPE MDHR SHIP ISTN CAST DAY IPR,1,
13150 + 2I5,2X,A2,2X,2I5,F8.3,2I5)
13200 6005 FORMAT(15,X,3A4,12,14,X,15,15,F7.2,F8.2,X,12,14,F8.5)
13300 C
13400 C MTLARD SURROUTINE TO FILL DATA ARRAY WITH TEMP AND SALINITY
13500 C DATA FOR ALL OBSERVATIONS.
13600 C
13608 WGT=1.
13616 IF(YSSW(13).EQ.-1)THEN
13617 M=M+1
13618 JICR=ICRUIS
13620 JIST=ISTAS
13622 CALL LZ(JICR)
13624 CALL LZ(JIST)
13626 ENCODE(12,52,GNAME)TSHP,JICR,JIST
13628 52 FORMAT(A2,A3,A3,'.REF')
13632 WRITE(12,8020)M,GNAME,WGT
13640 8020 FORMAT(14,A12,F5.2)
13648 OPEN(UNIT=KOUT,NAME=GNAME,TYPE='NEW',FORM='UNFORMATTED')
13664 WRITE(KOUT)KMDG
13680 ENDIF
13690 CALL DAT(IX(KUN))
13700 CALL GETDAT(KTP,DATAK,3300,2)
13800 620 RETURN
13900 C
14000 C NSW=5: CHECK ONLY LAT AND LONG OF HEADER ALREADY READ AGAINST
14100 C SELECTION PARAMETERS.
14200 C
14300 80 IF(XLONG-XEMN)85,87,87
14400 87 IF(XLONG-XEMX)89,89,85
14500 89 IF(XLAT-XNMN)85,83,83
14600 83 IF(XLAT-XNMX)82,82,85
14700 82 CONTINUE
14800 RETURN
14900 C
15000 C NSW=4 IMPLIES SELECTION CRITERIA ABOVE NOT MET.
15100 C
15200 85 NSW=4
15300 RETURN
15400 END
15403 C ***** SLAT FUNCTION *****
15407 REAL FUNCTION SLAT
15410 C *****
15414 INCLUDE 'IDXREC.DIM'
15421 C FUNCTION RETURNS DECIMAL DEGREE VALUE SIGNED - FOR SOUTH E WEST
15428 XLAT=ILTSO
15435 XLATM=ILTSM
15438 XLATM=XLATM/6000.
15442 SLAT=XLAT+SIGN(XLATM,XLAT)

```

AD-A103 892

WOODS HOLE OCEANOGRAPHIC INSTITUTION MA

F/G 8/10

VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM C--ETC(U)

AUG 81 N A BRAY

N00014-76-C-0197

UNCLASSIFIED

WHOI-81-70

NL

1-12
A
01-14-1

END
DATE
FILMED
O B
DTIC

```

15449      RETURN
15456      ENTRY SLNG
15463      XLAT=ILNSD
15470      XLATM=ILNSM
15473      XLATM=XLATM/6000.
15477      SLNG=XLAT+SIGN(XLATM,XLAT)
15484      RETURN
15491      END
15500      C ***** SUBROUTINE LZ(IA) *****
15503      SUBROUTINE LZ(IA)
15504      C *****
15506      INTEGER IA(1),IW(1)
15512      IF(IA(1).GE.100) GO TO 100
15515      IF(IA(1).GE.10) GO TO 10
15518      IF(IA(1).GE.0) GO TO 1
15521      RETURN
15524      100 CONTINUE
15527      ENCODE(3,2,IW(1)) IA(1)
15530      IA(1)=IW(1)
15533      2 FORMAT(1I3)
15536      RETURN
15539      10 CONTINUE
15542      ENCODE(3,3,IW(1)) IA(1)
15545      IA(1)=IW(1)
15548      3 FORMAT(1H0,I2)
15551      RETURN
15554      1 CONTINUE
15557      ENCODE(3,4,IW(1)) IA(1)
15560      4 FORMAT(2H00,I1)
15563      IA(1)=IW(1)
15566      RETURN
15584      END

```

```

12000 C POTENSUR.FOR FILE: SUBROUTINES FOR POTEN,PEPLT POTENTIAL ENERGY
12100 C PROGRAMS. VAX VERSION. N.BRAY.
12600 C *****
12700 SUBROUTINE SMINV(A,N,P,O,MR,IFAIL)
12800 C *****
12900 C
13000 C TO INVERT SYMMETRIC MATRIX FOR TRIANGULAR SECTION ARRANGED
13100 C IN A LINEAR ARRAY A(J).
13200 C FROM SYMINV7,,CACH #150 BY RUYISHAUSER VIA J. HALVAYS.
13300 C
13400 C APRIL 27 1975 N. FOFONOFF
13500 C
13600 DIMENSION A(I),P(I),O(1),MR(1)
13700 C
13800 IFAIL = 0
13900 DO 10 I=1,N
14000 10 MR(I) = 0
14100 C SEARCH FOR PIVOT
14200 DO 100 I=1,N
14300 BIGAJ = 0.0
14400 JJ = -N
14500 DO 20 J=1,N
14600 JJ = JJ+N-J+2
14700 B = ABS(A(JJ))
14800 IF(MR(J))20,12,20
14900 12 IF(B-BIGAJ)20,20,14
15000 14 BIGAJ = B
15100 K = J
15200 KK = JJ
15300 20 CONTINUE
15400 IF(BIGAJ)16,15,16
15500 15 IFAIL = 1
15600 RETURN
15700 C PREPARATION OF ELIMINATION
15800 16 MR(K) = 1
15900 Q(K) = 1./A(KK)
16000 P(K) = 1.0
16100 A(KK) = 0.0
16200 KMI = K-1
16300 IF(KMI)15,19,160
16400 160 JK = K - N
16500 DO 30 J=1,KMI
16600 JK = JK+N-J+1
16700 P(J) = A(JK)
16800 IF(MR(J))18,17,18
16900 17 Q(J) = -A(JK)*Q(K)
17000 DO 10 30
17100 18 Q(J) = A(JK)*Q(K)
17200 30 A(JK) = 0.0
17300 19 KPI = K+1
17400 KJ = KK
17500 IF(KPI-N)21,21,41
17600 21 DO 40 J=KPI,N
17700 KJ = KJ + 1
17800 IF(MR(J))34,32,34
17900 32 P(J) = A(KJ)
18000 GO TO 35
18100 34 P(J) = -A(KJ)
18200 35 Q(J) = -A(KJ)*Q(K)
18300 40 A(KJ) = 0.0
18400 C ELIMINATION PROPER

```

```

18500      41 JK = 0
18600      DO 50 J=1,N
18700      DO 50 K =J,N
18800      JK = JK + 1
18900      50 A(JK) = A(JK) + P(J)*O(K)
19000      100 CONTINUE
19100      150 RETURN
19200      END
19300      C SUBROUTINE TO INPUT PARAMETERS FOR REGRESSION IN POTEN
19400      C POTENTIAL ENERGY PROGRAM
19500      C
19600      C*****
19700      SUBROUTINE PARAM
19800      C*****
19900      C
20000      INCLUDE 'COMPOTEN.FOR'
20100      C
20200      INTEGER PFDELTA,PDELTA
20300      C
20400      C
20500      WRITE(KTTX,10)
20600      10  FORMAT('OINPUT NEW PARAMETERS?')
20700      IF(NOYES(KIN,KTTX).NE.1)RETURN
20800      15  WRITE(KTTX,20)
20900      20  FORMAT('OENTER THE NUMBER OF SECTIONS (GROUPS OF LEVELS WITH
21000      X THE SAME PARAMETERS)')
21100      READ(KIN,*)NSECTION
21200      IF(NSECTION.GT.19)THEN
21300      WRITE(KTTX,22)
21400      22  FORMAT('OMAXIMUM ALLOWED IS 19')
21500      GO TO 15
21600      ENDIF
21700      WRITE(KTTX,25)
21800      25  FORMAT('OENTER THE PRESSURE FOR THE FIRST LEVEL:')
21900      READ(KIN,*)INITIALP
22000      NLEVP=1
22100      NPREV=-1
22200      NSE=NSECTION+1
22300      DO 1000 I=1,NSECTION
22400      I2=I+NSE
22500      I3=I+2*NSE
22600      WRITE(KTTX,100)I
22700      100  FORMAT('OFOR SECTION',I4,' ENTER THE INTERVAL IN DB BETWEEN
22800      X LEVELED SURFACES:')
22900      READ(KIN,*)PFDELTA
23000      WRITE(KTTX,120)
23100      120  FORMAT('OENTER THE INTERVAL SIZE IN DB FOR THE REGRESSION:')
23200      READ(KIN,*)PDELTA
23300      WRITE(KTTX,140)
23400      140  FORMAT('OENTER THE FIRST PRESSURE IN THE NEXT SECTION:')
23500      READ(KIN,*)IP2
23600      NLEVEL=(IP2-INITIALP)/PFDELTA
23700      WRITE(KTTX,160)
23800      160  FORMAT('OENTER THE NUMBER OF TERMS IN THE REGRESSION:',/,
23900      X '(N=2 IMPLIES A LINEAR FIT; MAXIMUM N IS 8)')
24000      READ(KIN,*)N
24100      IF(N.GT.8)N=8
24200      C***** COMPUTE NPR(I) *****
24300      NPR(I)=NLEVEL*NLEVP
24400      C***** COMPUTE NPR(I2)
24500      NPR(I2)=PFDELTA

```

```

24600 C***** COMPUTE NSC(I) *****
24700 IF(I.EQ.1)THEN
24800     NSC(I)=INITIALP-PFDELTA
24900     IPREV=INITIALP
25000     PNDIF
25100     NSC(I+1)=IPREV+NLEVEL*PFDELTA
25200 C***** COMPUTE NPR(I3) *****
25300     NPR(I3)=NLEVP-(IPREV/PFDELTA)
25500     NPREV=NPR(I3)
25600     IPREV=NSC(I+1)
25700     NLEVP=NPR(I)
25800     INITIALP=IPREV
25900 C***** COMPUTE NSC(I2)
26000     NSC(I2)=N
26100 C***** COMPUTE NSC(I3) *****
26200     NSC(I3)=PDELTA/DELP
26300     1000 CONTINUE
26400     ITOTAL=IPREV
26450     JMAX=NLEVP
26500     WRITE(KTTX,200)NSECTION,NLEVP,ITOTAL
26600     200  FORMAT('0A TOTAL OF',I4,'SECTYONS;',I6,'LEVELS; THE DEEPEST
26700     X    LEVEL IS AT',I6,'DB. ')
26800     WRITE(KTTX,220)
26900     220  FORMAT('0CENTER MAXIMUM DEPTH OF THE DATA: ')
27000     READ(KIN,*)ITMAX
27100     I=NSECTION+1
27200     I2=2*I
27300     I3=3*I
27400     NPR(I)=NLEVP+5
27500     NPR(I2)=(ITMAX+500)/(NLEVP-NPREV)
27600     NPR(I3)=NPREV
27700     NSC(I)=IPREV
27800     NSC(I2)=N
27900     NSC(I3)=PDELTA/DELP
28000     NSECTION=NSECTION+1
28100     RETURN
28200     END
28300 C COEFF SUBR ***** PTSB1 *****
28400     SUBROUTINE COEFF(A,B,C,D,N)
28500 C *****
28600 C
28700 C COMPUTES COEFFICIENTS FOR A LINEAR TRANSFORMATION X=AX+B
28800 C FOR POLYNOMIAL OF ORDER N-1. INPUT ARRAY C. OUTPUT D.
28900 C
29000 C PCY 22 1975 N. FOFONOFF
29100 C
29200     DIMENSION C(1),D(1)
29300 C
29400     DO 25 I=1,N
29500     R = 1.0
29600     S = C(I)
29700     NMI = N - I
29800     TF(NMI)12,12,5
29900     5 DO 10 J=1,NMI
30000     TPJ = I + J
30100     R = (FLOAT(IPJ+1)/FLOAT(J))*B+R
30200     10 S = S + R*C(TPJ)
30300     12 TM1 = I - 1
30400     TF(TM1)15,15,20
30500     15 R = 1.0
30600     GO TO 25

```

```

30700      20 R = A**IM1
30800      25 D(I) = S*R
30900      RETURN
31000      END
31100      C
32300      C 9(N) FCN ***** PTSB1 *****
32400      FUNCTION R(N)
32500      C *****
32600      C
32700      INTEGER*2 IA,N
32800      PA = "77777"
32900      YF(N)1,2,2
33000      1 R = FLOAT((YFAND(N,IA)) + 32768.
33100      RETURN
33200      2 R = FLOAT(N)
33300      RETURN
33400      END
33500      C
33600      C *****
37000      SUBROUTINE EDIT(JERR)
37100      C *****
37200      C
37300      C EDIT TEMP AND SALINITY IN REGRESSION TABLE
37400      C
37500      C JAN 28 1975 N FOFONOFF
37600      C
37700      INCLUDE "COMPUTEN.FOR"
37800      EQUIVALENCE (PDIFF,A1)
37900      C
37950      IERR = 0
37975      DVPMAX = -.12
38000      DO 10 M=M1,M2
38100          DV(M) = P(M)
38200          P(M) = S(M)
38300      10 TH(M) = T(M)
38400      15 DVM = 0.0
38500          PM = 0.0
38600          THM = 0.0
38700          XNDP = NDP
38800          DO 20 M=M1,M2
38900              DVM = DVM + DV(M)
39000              PM = PM + P(M)
39100              THM = THM + TH(M)
39200              PT(M) = P(M)
39300              TTH(M) = TH(M)
39400      20 CONTINUE
39500          DVM = DVM/XNDP
39600          PM = PM/XNDP
39700          THM = THM/XNDP
39800          CALL LSFT
39900          KERR = 0
40000          DO 60 M=M1,M2
40100              IF(ABS(PT(M)-PM)-A3*Z1)40,30,30
40200      30 CORR = POLY(BV(M),DVM,CP,N,0.0,6000.0) + PM
40300              KERR = 1
40400              IF(ISSW(3))31,32,32
40450      31 DELTA = CORR-PT(M)
40451              S1 = DATAX(M-1,2)
40452              S2 = DATAX(M,2)
40453              S3 = DATAX(M+1,2)
40454              THI = THETA(DVM-1,DATAX(M-1,1),S1,PF)

```

```

40455      TH2 = THETA(DV(M),DATA(X(M,1),S2,PF)
40456      TH3 = THETA(DV(M+1),DATA(X(M+1,1),S3,PF)
40458      DV1 = DVA(PF,TH1,S1)
40470      DV2 = DVA(PF,TH2,S2)
40476      DV3 = DVA(PF,TH3,S3)
40482      DVP1 = (DV1-DV2)/DELP
40490      DVP2 = (DV2-DV3)/DELP
40500      WRITE(KLIST,3100)DV(M),P(M),CORR,DELTA,DVP1,DVP2
40600      32 P(M) = CORR
40700      40 IF(ABS(TT(M)-THM)-A3+Z2)55,50,50
40800      50 CORR = POLY(DV(M),DVH,CT,N,0.0,6000.0) + THM
40900      KERR = 1
41000      IF(ISSW(3))51,52,52
41050      51 DELTA = CORR-TH(M)
41091      S1 = DATA(X(M-1,2)
41052      S2 = DATA(X(M,2)
41053      S3 = DATA(X(M+1,2)
41054      TH1 = THETA(DV(M-1),DATA(X(M-1,1),S1,PF)
41055      TH2 = THETA(DV(M),DATA(X(M,1),S2,PF)
41056      TH3 = THETA(DV(M+1),DATA(X(M+1,1),S3,PF)
41058      DV1 = DVA(PF,TH1,S1)
41066      DV2 = DVA(PF,TH2,S2)
41074      DV3 = DVA(PF,TH3,S3)
41082      DVP1 = (DV1-DV2)/DELP
41090      DVP2 = (DV2-DV3)/DELP
41100      WRITE(KLIST,3100)DV(M),TH(M),CORR,DELTA,DVP1,DVP2
41200      52 TH(M) = CORR
41300      55 IF(KERR.EQ.0.AND.IERR.EQ.0)THEN
41312      JERR=-2
41343      ENDIF
41350      IERR = IERR + 1
41400      60 CONTINUE
41500      IF(KERR)70,70,15
41600      70 DO 75 M=M1,M2
41700      SIM) = P(M)
41800      TIM) = TH(M)
41900      75 P(M) = DV(M)
42000      JERR = JERR + 1
42100      80 RETURN
42200      3100 FORMAT(F7.1,F9.3,' REPLACED BY: ',F9.3,' CHANGE IS:',F9.3,
42250      * ' SP. VOL. GRADIENTS: ABOVE = ',F9.3,' BELOW = ',F9.3)
42300      END
42400      C
42500      C *****
42600      SUBROUTINE LSFT
42700      C *****
42800      C
42900      C LEAST SQUARES REGRESSION SUBROUTINE FOR POTEN.
43000      C
43100      C PAR 6 1976 N FOFONOFF
43200      C
43300      INCLUDE 'COMPOTEN.FOR'
43400      C
43500      1 NA = N*(N+1)/2
43600      L = 1
43700      DO 10 I=1,NA
43800      10 CORR = 0.0
43900      DO 12 I=1,N
44000      CP(I) = 0.0
44100      12 CT(I) = 0.0
44200      15 DO 20 I=1,N

```

```

44300      BP(I) = 0.0
44400      BT(I) = 0.0
44500      20 CONTINUE
44600      DO 8 I=M1,M2
44700      X = DV(I) - DVM
44800      DO 100 J=1,N
44900      IF(J-1)90,90,95
45000      90      B(J) = 1.0
45100      GO TO 100
45200      95      B(J) = X*(J-1)
45300      100 CONTINUE
45400      JK = 0
45500      X = PT(I) - PM
45600      XT = TT(I) - THM
45700      DO 8 J=1,N
45800      BP(J) = BP(J) + B(J)*X
45900      BT(J) = BT(J) + B(J)*XT
46000      IF(L-1)105,105,8
46100      105 DO 7 K=J,N
46200      JK = JK + 1
46300      7 CO(JK) = CO(JK) + B(J)*B(K)
46400      8 CONTINUE
46500      IF(L-1)173,173,174
46600      173 CALL SMINV(CO,N,B,BA,MR,IFAIL)
46700      174 DO 200 M=1,N
46800      SP = 0.0
46900      ST = 0.0
47000      JM = M-N
47100      DO 1745 J=1,N
47200      IF(J-M)1740,1740,1742
47300      1740 JM = JM + N - J + 1
47400      GO TO 1744
47500      1742 JM = JM + 1
47600      1744 SP = SP + CO(JM)*BP(J)
47700      1745 ST = ST + CO(JM)*BT(J)
47800      CP(M) = CP(M) + SP
47900      200 CT(M) = CT(M) + ST
48000      C COMPUTE RESIDUALS
48100      175 RP = 0.0
48200      RT = 0.0
48300      DO 185 I=M1,M2
48400      FP = 0.0
48500      FT = 0.0
48600      X = DV(I) - DVM
48700      DO 180 J=1,N
48800      NJ = N-J+1
48900      FP = FP*X + CP(NJ)
49000      180 FT = FT*X + CT(NJ)
49100      SP = P(I) - FP
49200      PT(I) = SP
49300      ST = TH(I) - FT
49400      TT(I) = ST
49500      IF(L-KSW)185,185,183
49600      183 RP = (SP-PM)**2 + RP
49700      RT = (ST-THM)**2 + RT
49800      185 CONTINUE
49900      L = L + 1
50000      IF(L-KSW)15,15,195
50100      195 XN = NDP - N
50200      Z1 = SORT(RP/XN)
50300      Z2 = SORT(RTYXN)

```

```

50400 IF(ISSW(6))300,350,350
50500 300 DO 310 I=1,N
50600 IT = I + ((2*N-I)*I-1)/2
50700 B(I) = ABS(CP(I))/(21*SORT(ABS(CO(I))))
50800 BP(I) = ABS(CT(I))/(22*SORT(ABS(CO(I))))
50900 310 CONTINUE
51000 WRITE(KLIST,3100)PF,N,NDP,(B(K),K=1,N)
51100 WRITE(KLIST,3110)(BP(K),K=1,N)
51200 350 RETURN
51300 3100 FORMAT(F6.0,I4,I3,8F8.3)
51400 3110 FORMAT(I3X,8F8.3)
51500 FND

```

Appendix C.
Program Listings for PEPLT

100	PEPLT/PEPLS: SHORT DOCUMENTATION				
200	KBR	ISW	JSW	KLIST	DESCRIPTION
250	0	-	-	-	SHORT DOCUMENTATION
300	1	0	-	-	CALL TABLE SUBROUTINE: LIST,
400					PLOT, OUTPUT IN MAP FORMAT
450					PLOT IS:
466		$X = A1*VRBL(NX1)+A2*VRBL(NX2)+A3*VRBL(NX3)+A4*C(IREC,1)$			
482		$Y = B1*VRBL(NY1)+B2*VRBL(NY2)+B3*VRBL(NY3)+B4*C(IREC,2)$			
488	1	1	-	-	CHANGE PARAMETERS FOR PLOT
494		2	-	-	INITIALIZE PARAMETERS FOR PLOT
500	2	-	-	-	CHANGE DATA SELECTION VARIABLES
600	3	-	-	-	CHANGE PLOT PARAMETERS
700	4	-	-	-	CALL AVRGS SUBROUTINE: HORIZONTAL
800					AVERAGES. FOR DETAILED DOCUMENTA-
900					TION, ACCESS KBR=0 AFTER ENTERING
950					AVRGS BRANCH.
1000	5	-	-	-	SET ISSW (SWITCH) ARRAY
1100	6	-	-	-	RESTART MAIN PROGRAM
1200	7	-	-	-	EXIT PROGRAM

PEPLT/PEPLS: BRANCH 3--PARAMETERS--SHORT DOCUMENTATION			
	KRR3	ISW3	DESCRIPTION
100			
300			
400	1	0	PRINT OUT PARAMETERS ON KLIST; STORE COMMON TO
500			FILE KPLCM. RETURN TO PEPLS.
600		1	INPUT VARIABLE SELECTORS NX1 TO NZ3
700		2	ENTER A1 TO A6
1100	2	2	ENTER B1 TO B6
1200	3	2	ENTER C1 TO C6
1300	4	2	ENTER D1 TO D6

100	PEPLT/AVRGS: SHORT DOCUMENTATION				
200	KRR	PSW	JSW	KLIST	DESCRIPTION
250	4	0	-	-	SHORT DOCUMENTATION
300	4	1	#	#	READ FROM DATA FILES VARIABLES IN COLUMNS
400					JSW TO KLIST
500		2	#	#	ZERO COLUMNS JSW TO KLIST
600		3	1	-	INITIALIZE AND INPUT PARAMETERS
700			0	-	INPUT PARAMETERS--NO INITIALIZATION
800		4	#	#	DIVIDE COLUMNS JSW TO KLIST BY COLUMN 6
900		5	#	-	ADD COLUMN JSW VERTICALLY FROM THE TOP
1000		6	-	LU	PRINT OUT DATA ARRAY ON UNIT KLIST
1100		7	#	-	CALL NCAR PLOT PACKAGE TO PLOT ONE FRAME.
1200					DEFAULT IS COLUMN JSW AGAINST PRESSURE.
1300					GENERAL PLOTS:
1400					$X = R1 * C(I, JSW) + R2 * C(I, NX2) + R3 * PR$
1500					$Y = R1 * PP + A2 * C(I, NY1) + A3 * C(I, NY2)$
1600					MULTIPLE PLOTS ON ONE FRAME ALLOWED
1700	KBR	ISW	JSW	KLIST	
1800		8	-	-	COMPUTE DYNAMIC HEIGHT AND POTENTIAL ENERGY:
1900					ASSUMES DVI IN COLUMN 1 (NV(1)=18) AND DVE
2000					IN COLUMN 2 (NV(2)=19).
2100		9	#	#	INTEGRATE COLUMNS JSW TO KLIST AS A FUNCTION
2200					OF PRESSURE
2300		10	#	#	SUBTRACT REFERENCE LEVEL VALUE C(JREF,8) FROM
2400					COLUMNS JSW TO KLIST
2500		11	1	-	INPUT JC1,CR1 TO JC4,CR4
2600		11	0	-	PERFORM THE FOLLOWING COLUMN ADDITION:
2700					$C(I, JC1) = CR1 * C(I, JC1) + CR2 * C(I, JC2) +$
2800					$CR3 * C(I, JC3) + CR4 * C(JREF, JC4)$
2900		12	-	-	RETURN TO PEPLS
3000		13	-	-	INPUT COLUMN #S AND CONSTANTS TO PERFORM
3100					THE FOLLOWING COLUMN MULTIPLICATION:
3200					$C(I, REC, I) = CON1 * C(I, REC, I) + CON2 * C(I, REC, J) +$
3300					$CON3 * C(I, REC, K)$. INPUT ORDER: I, J, K, CON1
3400					CON2, CON3; AN INDEX (I, J, OR K) OF VALUE -1
3500					PREVENTS THE INCLUSION OF THE ASSOCIATED
3600					AND FOLLOWING COLUMN(S).
3700	KRR	ISW	JSW	KLIST	
3800		14	-	-	OUTPUT FIRST THREE COLUMNS IN MAP FORMAT:
3900					ACCESS TO THIS BRANCH QUERIES WHAT HORIZONTAL
4000					LEVEL # IS DESIRED
4100		15	-	-	NOT USED
4200		16	1	-	INPUT X, J
4300			0	-	$C(I, REC, J) = C(I, REC, J) * X$. (SHOULD FOLLOW
4400					4.16, I IMMEDIATELY IN EXECUTION.)
4500		17	-	-	ERROR SUMMATION: VERTICAL INTEGRATION WITH
4600					(DELTA P) * 2 AS THE INCREMENT
4700		18	-	-	INPUT DELTA P INTO C(I, REC, 5)
4800		19	-	-	EXCHANGE TWO COLUMNS OF C
4900		20	-	-	CHANGE A SINGLE ELEMENT OF C
5000		21	-	-	COMPUTE STANDARD DEVIATION OF X GIVEN
5100					X-BAR IN C(I, REC, 4) AND X*X-BAR IN C(I, REC, 3).
5200					RESULT IS STORED IN C(I, REC, 1).
5300		22	1	1	CALCULATE DYNAMIC HEIGHT AT A GIVEN LEVEL
5400					RELATIVE TO PRESSURE CORRESPONDING TO JREF
5500					AND OUTPUT IN MAP FORMAT, ALONG WITH VARIABLES
5600					FROM COLUMNS 3 AND 4 AT THAT PRESSURE.
5700					NV(1) MUST BE = 18 AND NV(2)=19.

```

5 C COMPEPLY FOR FILE: DIMENSION,COMMON AND EQUIVALENCE FOR PEPLY
7 C DISPLAY PROGRAM. N.BRAY
10 PARAMETER KCM=943
32 PARAMETER JDIM=100
55 CHARACTER*8 DOC
77 CHARACTER*12 GNAME
100 DIMENSION CST(6,4),VR(35)
200 DIMENSION KHDG(150),KBUF(46)
300 DIMENSION KPLCH(KCM)
378 C
391 C COMMON
404 C
450 COMMON KIN
475 C BEGINNING OF STORED COMMON
500 COMMON KTTX,KLIST,KTP,KOUT,KBR
600 COMMON NX1,NX2,NX3,NY1,NY2,NY3,NZ1,NZ2,NZ3
700 COMMON ISW,JSW,MV,MW,YN,ND
800 C
1100 COMMON XMIN,XMAX,YMIN,YMAX
1300 COMMON A1,A2,A3,A4,A5,A6
1400 COMMON H1,H2,H3,H4,H5,H6
1500 COMMON C1,C2,C3,C4,C5,C6
1600 COMMON D1,D2,D3,D4,D5,D6
1700 COMMON X1,ZLTO,ZLGO,DAY,XPL,YPL
1800 COMMON WT
1900 COMMON NV(6),NX(6),AV,BV,CV
2000 COMMON JC1,JC2,JC3,JC4
2100 COMMON CR1,CR2,CR3,CR4
2200 COMMON JMAX,JREF
2300 COMMON NX4,NY4,NZ4
2400 COMMON IV1,IV2,IV3,JBUF,JHDR,JDO
2500 COMMON JSHP(6),DAY1,DAY2,PHIN,PMAX
2600 COMMON XEMN,XEMX,XNMN,XNMX
2633 COMMON C(100,6)
2666 COMMON ISSW(16)
2683 COMMON PLABL(10),XLABL(10),YLABL(10)
2700 C
2800 COMMON LTYPE,MHDR,ICON,ISHP,ICAST,XDAY,TPR,LPR
2900 COMMON XLAT,XLONG,WBT,XLTO,XLGO
3000 COMMON LBBL(3),LBL(13),NSC(60),NPR(60),NSECTION
3100 COMMON KTYPE,MBOF,IREF,N,NDP,KSW,L1,L2
3200 COMMON PF,TO,SO,DVO
3300 COMMON PT,THF,SF,DVF
3400 COMMON PM,THM,SM,DVM
3500 COMMON DH,PE,XPE
3600 COMMON CP(8),Z1,CT(8),Z2,F1,F2,F3
3616 C END OF STORED COMMON
3632 COMMON XDAT(100,7),YDAT(100,7)
3650 COMMON IDELP,DP
4100 C
4200 C CHAR
4300 COMMON/CHARACTER/ GNAME(200),DOC(10)
4400 C
4425 EQUIVALENCE (A1,CST),(VR,PF)
4450 EQUIVALENCE (KHDG,LTYPE),(KBUF,KTYPE)
4475 EQUIVALENCE (KTTX,KPLCH)
4500 C

```

```

100  C PEPLT PRGC ***** SEPT 24 1977 *****
200  C *****
300  PROGRAM PEPLT
400  C *****
500  C
600  C PROGRAM TO PLOT POTEN VARIABLES.
700  C JUNE 27 1976 N FOFONOFF
800  C VAX VERSION
900  C NOV 1980 N.BRAY
1000 INCLUDE 'COMPEPLT.FOR'
1100 C
1200 C
1300 OPEN(UNIT=10,NAME='KPLCH.DAT',ACCESS='DIRECT',TYPE='OLD',
1400 * RECORDTYPE='FIXED',RECORDSIZE=KCH,ERR=1100)
1500 C
1600 10 KIN = 5
1700 KTX = 6
1800 KLIST = 6
1900 KOUT = 9
2000 KTP = 11
2100 KBR = 3
2200 WRITE(KTX,1000)
2300 IF(NOTYES(KIN,KTX).EQ.1)THEN
2400 20 READ(10,1,ERR=1100)KPLCH
2500 ELSE
2508 C
2516 DO 107 I = 1,6
2524 DO 107 J = 1,9
2532 107 CST(I,J) = 0.0
2540 DO 108 I=1,100
2548 DO 108 J=1,6
2556 108 C(I,J) = 0.0
2564 DO 109 J=1,16
2572 ISSW(J)=0
2580 109 CONTINUE
2588 C
2594 ENDIF
2600 CALL PEPLS
2700 GO TO 20
2800 1000 FORMAT(1H,'PEPLT: LOAD IN PREVIOUSLY STORED COMMON?')
2900 1100 OPEN(UNIT=10,NAME='KPLCH.DAT',ACCESS='DIRECT',TYPE='NEW',
3000 * RECORDTYPE='FIXED',RECORDSIZE=KCH,ERR=1100)
3100 GO TO 10
3200 END

```

```

100 C PEPL'S SUBROUTINE ***** SEPT 24 1977 *****
200 SUBROUTINE PEPL'S
300 C *****
400 C
500 C PROGRAM TO PLOT POTEN VARIABLES.
600 C JUNE 27 1976 N FOFONOFF
700 C VAX VERSION--NOV 1980
800 C INCLUDE 'COMPEPLY.FOR'
900 C
1000 WRITE(KTTX,40)
1100 40 FORMAT(1H, 'INITIALIZE DATA, VARIABLE SELECTION PARAMETERS
1150 * (YES OR NO)?')
1200 IF(NDYES(KIN,KTTX).EQ.1)GO TO 106
1300 120 WRITE(KTTX,1200)KBR,ISW,JSW,KLIST,KTP,KOUT,KIN
1400 1200 FORMAT('PEPLT',KBR,ISW,JSW,KLIST,KTP,KOUT,KIN',/,
1500 * 6X,3I3,15,I3,14,I3)
1600 KLIST = 6
1700 KOUT = 8
1800 KTP = 11
1900 READ(KIN,*)KBR,ISW,JSW,KLIST,KTP,KOUT,KIN
2000 IF(KBR.GT.7)KBR=7
2100 IF(KBR)120,800,130
2200 130 GO TO (200,20,30,400,500,600,700)KBR
2300 C *****
2400 C INITIYALIZE
2500 C
3300 106 XMIN=-20
3400 XMAX = 100.0
3500 YMIN = 0.0
3600 YMAX = 5000.0
3700 A1 = 1.0
3800 B1 = 1.0
3900 C1 = 1.0
4000 D1 = 1.0
4100 D2 = 1.0
4200 D3 = 1.0
4300 D4 = 1.0
4400 D5 = 1.0
4500 D6 = 1.0
4600 NX1 = 12
4700 NX2 = 0
4800 NX3 = 0
4900 NY1 = 19
5000 NY2 = 0
5100 NY3 = 0
5200 NZ1 = 25
5300 NZ2 = 0
5400 NZ3 = 0
5500 MV = 3
5600 MW = 0
5700 KTP = 11
5800 CALL DATA(-1,IEOF)
5900 GO TO 120
6000 C***** BRANCH 1--CALL TABLE SUBROUTINE *****
6100 200 CALL TABLE
6200 GO TO 120
6300 C ***** BRANCH 2--CHANGE DATA SELECTION VARIABLES *****
6400 20 CALL DATA(0,IEOF)
6500 GO TO 120
6600 C ***** BRANCH 3--CHANGE OR LIST COEFFICIENTS *****
6700 30 OPENUNIT=50,NAME='PEPLS3.DDC',TYPE='OLD',REDOONLY)

```

```

6800      DO 3350 N=1,200
6900      READ(50,3325,END=3340)(DOC(I),I=1,9)
7000      WRITE(KTTY,3330)(DOC(I),I=1,9)
7100      3350 CONTINUE
7200      3325 FORMAT(9A8)
7300      3330 FORMAT(1H,9A8)
7400      3340 CLOSE (UNIT=50)
7500      300 WRITE(KTTY,3000)
7600      3000 FORMAT(1H, '*** PARAMETERS: KBR3,ISW3,KX,MV,MW')
7700      KX = 4
7800      READ(KIN,*)KBR3,ISW3,KX,MV,MW
7900      IF(KBR3)30,350,31
8000      31 IF(ISW3-1)32,32,34
8100      32 WRITE(KLIST,3200)NX1,NX2,NX3,NY1,NY2,NY3,NZ1,NZ2,NZ3
8200      3200 FORMAT(1H, 'NX1,NX2,NX3,NY1,NY2,NY3,NZ1,NZ2,NZ3',/,'914)
8300      IF(ISW3-1)34,33,30
8400      33 READ(KIN,*)NX1,NX2,NX3,NY1,NY2,NY3,NZ1,NZ2,NZ3
8500      GO TO 300
8600      C
8700      34 WRITE(KLIST,3400)KBR3,(CST(JC,KBR3),JC=1,6)
8800      3400 FORMAT(12,6(X,P10,4))
8900      IF(ISW3)30,38,35
9000      35 READ(KIN,*)(CST(JC,KBR3),JC=1,6)
9100      37 GO TO 300
9200      38 KBR3 = KBR3 + 1
9300      IF(KBR3-KX)34,34,350
9400      C
9500      350 WRITE(10,1)KPLCM
9600      GO TO 120
9700      C
9800      C *****AVERAGES #4 *****
9900      400 CALL AVRG5
10000     GO TO 120
10100     C5 ***** SET ISSW SWITCHES *****
10200     500 WRITE(KTTY,5000)(K,K=1,16),(ISSW(K),K=1,16)
10300     5000 FORMAT(2(1H, X,1614,/), ' ENTER K, ISSW(K)')
10400     READ(KIN,*)(K,ISSW(K),K=1,16)
10500     GO TO 120
10600     C ***** RETURN TO MAIN PROGRAM *****
10700     600 RETURN
10800     C ***** EXIT PROGRAM *****
10900     700 WRITE(KTTY,7000)
11000     IF(NOYES(KIN,KTTY).NE.1)GO TO 120
11100     STOP
11200     7000 FORMAT(1H, 'EXIT PROGRAM?')
11300     C** PEPLS: SHORT DOCUMENTATION--BRANCH 0 *****
11400     800 OPEN(UNIT=50,NAME='PEPLS.DOC',TYPE='OLD',READONLY)
11500     DO 850 N=1,200
11600     READ(50,825,END=812)(DOC(I),I=1,9)
11700     WRITE(KTTY,830)(DOC(I),I=1,9)
11800     850 CONTINUE
11900     812 CLOSE(UNIT=50)
12000     825 FORMAT(9A8)
12100     830 FORMAT(1H,9A8)
12200     GO TO 120
12300     END
12400

```

```

100 C AVROS SUBPROG *****
200 SUBROUTINE AVROS
300 C *****
400 C
500 C FOR HORIZONTAL AVERAGES COMPUTE LIST AND PENT.
600 C JUNE 27 1976 N FOFONOFF
700 C VAX VERSION--NOV 1980: N.BRAY
800 DIMENSION D(6)
900 DIMENSION ICHAR(6)
1000 DIMENSION XVM(4)
1050 CHARACTER*12 OUTNAME
1100 INCLUDE 'COMPELET.FOR'
1200 C
1300 EQUIVALENCE (DI,D)
1400 C
1450 CHARACTER*5 IDSTN
1500 C
1600 10 GOTO(100,200,300,400,500,600,700,800,900,1000,1100,1200,1300,
1700 * 1400,1500,1600,1700,1800,1900,2000,2100,2200,1500)ISW
1800 C
1900 C *****I READ DATA TO C-TABLE *****
2000 100 CONTINUE
2200 DO 101 K=1,ND
2300 READ(12,1011,END=1012)M,GNAME(K),WT
2400 1011 FORMAT(I4,A12,F5.2)
2500 GNAME(K)(9:12)='.AVG'
2600 101 CONTINUE
2625 GO TO 1013
2650 1012 CONTINUE
2675 ND = K-1
2687 REWIND 12
2700 1013 CONTINUE
2800 DO 170 NST=1,ND
2900 IEOF = 0
3000 OPEN(UNIT=KTP,NAME=GNAME(NST),READONLY,TYPE='OLD',FORM=
3100 * 'UNFORMATTED',PRR=168)
3200 102 CALL DATA(I,IEOF)
3300 IF(IEOF)165,105,105
3400 105 WT = 1.0
3500 IF(ISSW(15))110,115,115
3600 110 WT = WGT
3700 115 DO 160 I=JSW,KLIST
3900 120 XT = VIBL(NV(I))
4000 C(I,REC,I)=C(I,REC,I) + D(I)*WT*(AV*XT+(BV+CV*XT)*
4100 * VIBL(NX(I)))
4200 160 CONTINUE
4300 GO TO 102
4350 165 IF(ISSW.EQ.22)GO TO 800
4366 GO TO 170
4382 168 WRITE(KTTX,*)'ERROR READING',GNAME(NST)
4400 170 CONTINUE
4600 WRITE(10)1)KPLCM
4700 130 GO TO 1500
4800 C ***** #2 ZERO TABLE SET PARAMETERS *****
4900 200 DO 210 I=JSW,KLIST
5000 DO 210 J=1,100
5100 210 C(I,J) = 0.0
5200 IF(ISSW.EQ.22.AND.LFILE.EQ.0)GO TO 170
5300 GO TO 1500
5400 C ***** #3 SEP PARAMETERS *****
5500 300 IF(IJSW)320,320,310

```

```

5600      310  ND = 1
5700          NV(1) = 51
5800          NV(2) = 68
5900          NV(3) = 86
6000          NV(4) = 87
6100          NV(5) = 63
6200          NV(6) = -1
6300          DO 312 I=1,6
6400      312  NX(I)=0
6500          JC1 = 1
6600          CR1 = 1.0
6700          JC2 = 2
6800          CR2 = -1.0
6900          JC3 = 3
7000          CR3 = -1.0
7100          JC4 = 4
7200          CR4 = 1.0
7300          AV=1.
7400          BV=0.
7500          CV=0.
7600          JMAX = 55
7700          JREF = 50
7800      320  IF(JSW)340,325,325
7900      325  WRITE(KTTX,3200)ND,(NV(K),K=1,6),JREF,JMAX
8000          READ(KIN,*)ND,(NV(K),K=1,6),JREF,JMAX
8100      330  WRITE(KTTX,3300)AV,BV,CV,(NX(I),I=1,6)
8200          READ(KIN,*)AV,BV,CV,(NX(I),I=1,6)
8300      340  WRITE(KTTX,3400)A1,A2,A3,B1,B2,B3
8400          READ(KIN,*)A1,A2,A3,B1,B2,B3
8500          WRITE(KTTX,3500)NX1,NX2,NY1,NY2
8600          READ(KIN,*)NX1,NX2,NY1,NY2
8700      225  GO TO 1500
8800      3200  FORMAT(IH,'ND,NV(6),JREF,JMAX',/,9I5)
8900      3300  FORMAT(IH,'AV,BV,CV,NX(6)',/,3F6.3,6I3)
9000      3400  FORMAT(IH,'PLOT PARAMETERS:  A1      A2      A3
9100      *      B1      B2      B3',/,16X,6F9.3)
9200      3500  FORMAT(IH,'  NX1  NX2  NY1  NY2',/,4I5)
9300  C ***** #4 AVERAGE TABLE *****
9400      400  DO 410 J=1,100
9500          IF(C(J,6))405,415,405
9600      405  DO 410 I=JSW,KLIST
9700      410  C(J,I) = C(J,I)/C(J,6)
9800      415  JMAX = J - I
9900          IF(ISSW(2))420,1500,1500
10000      420  WRITE(4,425)KRR,ISW,JSW,KLIST
10100      425  FORMAT(IH,4(I3,2X))
10200          GO TO 1500
10300  C ***** #5 ADD COLUMN JSW *****
10400      500  DO 510 J=2,JMAX
10500      510  C(J,JSW) = C(J-1,JSW) + C(J,JSW)
10600          GO TO 1500
10700  C ***** #6 LIST TABLE *****
10800      600  WRITE(KLIST,6000)CLBL(K),K=1,19)
10900          WRITE(KLIST,6055)ND,JREF,JMAX,(NV(K),K=1,6)
11000          WRITE(KLIST,6056)AV,BV,CV,(NX(I),I=1,6)
11100          DO 610 J=1,JMAX
11200      605  KP = JPRIS,NPR,NSECTION)
11300          WRITE(KLIST,6050)J,KP,(C(J,K),K=1,6)
11400      610  CONTINUE
11500          GO TO 1500
11600      6000  FORMAT(IH,13X4)

```

```

11700 6050 FORMAT(I3,Y5,2X,6F10.4)
11800 6055 FORMAT(1H,3I4,6I8)
11900 6056 FORMAT(1H,3F4.2,6I8)
12000 C ***** #7 PLOT TABLE *****
12100 700 CONTINUE
12200 JMIN=1
12300 NCURV=1
12400 WRITE(KTTX,7000)
12500 READ(KIN,*)NCURV,JMIN
12600 IF(NCURV.GT.6)NCURV=6
12700 WRITE(KTTX,7010)PLABL
12800 IF(NYES(KIN,KTTX).EQ.1)THEN
12900 READ(KIN,7020)PLABL
13000 CALL STRIP(PLABL)
13100 ENDIF
13200 WRITE(KTTX,7030)
13300 IF(NYES(KIN,KTTX).EQ.-1)THEN
13400 WRITE(KTTX,7040)XMIN,XMAX,YMIN,YMAX
13500 READ(KIN,*)XMIN,XMAX,YMIN,YMAX
13600 CALL AGSETF(6HX/MIN.,XMIN)
13700 CALL AGSETF(6HX/MAX.,XMAX)
13800 CALL AGSETF(6HY/MIN.,YMIN)
13900 CALL AGSETF(6HY/MAX.,YMAX)
14000 ENDIF
14100 WRITE(KTTX,7050)XLABL
14200 IF(NYES(KIN,KTTX).EQ.1)THEN
14300 READ(KIN,7020)XLABL
14400 CALL STRIP(XLABL)
14500 ENDIF
14600 WRITE(KTTX,7060)YLABL
14700 IF(NYES(KIN,KTTX).EQ.1)THEN
14800 READ(KIN,7020)YLABL
14900 CALL STRIP(YLABL)
15000 ENDIF
15100 C SET UP PLOT LABEL
15200 C CALL AGSETF(11HLABEL/NAME.,1HT)
15300 C CALL AGSETF(12HLINE/NUMBER.,.85)
15400 C SET PARAMETERS FOR E7MXY PLOT
15500 C CALL AGSETF(1HTOP/NUMERIC/TYPE.,1.E36)
15600 CALL AGSETF(4HROW.,2.)
15700 CALL AGSETF(6HFRAME.,2)
15800 C READ DATA INTO PLOT ARRAYS
15900 DO 710 K=1,NCURV
16000 WRITE(KTTX,7070)JSW
16100 READ(KIN,*)JSW
16200 IF(ISSW(5).EQ.-1)THEN
16300 WRITE(KTTX,7080)
16400 READ(KIN,7090)ICHAR(K)
16500 ENDIF
16600 705 IJM=0
16700 DO 710 J=JMIN,JMAX
16800 IJM=IJM+1
16900 PR=FLOAT(JPRTJ,NPR,NSECTION)
17000 XDAT(IJM,K)=B1*C(J,JSW) + B2*C(J,NX2) + B3*PR
17100 YDAT(IJM,K)=A1*PR + A2*C(J,NY1) + A3*C(J,NY2)
17200 710 CONTINUE
17300 IF(TA1.NE.0.)THEN
17400 CALL AGSETF(8HY/ORDER.,1.)
17500 ELSE
17600 CALL AGSETF(8HY/ORDER.,0.)
17700 ENDIF

```

```

17800      CALL ANOTAT(XLABL,YLABL,0,0,0,0.)
17900      CALL F7HXY(XDAT,YDAT,JDIM,NCURV,IJM,PLABL)
18000      CALL AGGETP(15HSECONDARY/USER.,XYM,4)
18100      IF(0.GT.XYM(1).AND.0.LT.XYM(2))THEN
18200          CALL LINENCAR(0.,XYM(3),0.,XYM(4))
18300      ENDIF
18400      IF(15SW(5).EQ.-1)THEN
18500          DO 720 I=1,NCURV
18600              CALL POINTS(XDAT(I,I),YDAT(I,I),IJM,ICVAR(I),0)
18700          720 CONTINUE
18800      ENDIF
18900          CALL AGSETF(6HX/MIN.,1.E36)
19000          CALL AGSETF(6HX/MAX.,1.E36)
19100          CALL AGSETF(6HY/MIN.,1.E36)
19200          CALL AGSETF(6HY/MAX.,1.E36)
19300          CALL FRAME
19400          GO TO 1500
19500      C FORMATS
19600          7000  FORMAT(1H,'INPUT # OF CURVES IN THIS PLOT (MAX IS 6);
19700          *    AND INDEX OF FIRST POINT:')
19800          7010  FORMAT(1H,'CHANGE PLOT LABEL? OLD LABEL IS:V,7,4H',10A4)
19900          7020  FORMAT(10A4)
20000          7030  FORMAT(1H,'USE DEFAULT AXIS PARAMETERS?')
20100          7040  FORMAT(1H,'CURRENT VALUES OF XMIN,XMAX,YMIN,YMAX: ',4F10.3)
20200          7050  FORMAT(1H,'CHANGE X-AXIS LABEL? OLD LABEL IS: ',7,4H',10A4)
20300          7060  FORMAT(1H,'CHANGE Y-AXIS LABEL? OLD LABEL IS: ',7,4H',10A4)
20400          7070  FORMAT(1H,'INPUT COLUMN # (1 TO 6) TO BE PLOTTED',I3)
20500          7080  FORMAT(1H,'INPUT IDENTIFYING CHARACTER')
20600          7090  FORMAT(A1)
20700      C ***** #8 COMPUTE DH AND PE *****
20800          800  PPR = 0.0
20900          DELA = C(1,1)-C(1,2)
21000          DELB = DELA
21100          DO 820 J=1,JMAX
21200          805  PR = JPR(J,NPR,NSECTION)
21300          DELP = PR-PPR
21400          DELA = C(J,1)-C(J,2)
21500          DHX = 0.5*(DELA+DELB)*DELP
21600          PEX = 0.50968E-1*(PR*DELA+PPR*DELB)*DELP
21700          IF(J-1)815,810,815
21800          810  C(1,I) = DHX
21900          C(1,2) = PEX
22000          GO TO 817
22100          815  C(J,1) = C(J-1,1) + DHX
22200          C(J,2) = C(J-1,2) + PEX
22300          817  DELB = DELA
22400          820  PPR = PR
22500          IF(15W.EQ.22)THEN
22525              KLIST=2
22550              GO TO 1000
22575          ENDIF
22600          IF(15SW(2))825,1500,1500
22700          825  WRITE(4,425)KBR,15W,JSW,KLIST
22800          GO TO 1500
22900      C ***** #9 INTEGRATE OVER PRESSURE *****
23000          900  DO 950 I=JSW,KLIST
23100              PPR = 0.0
23200              CPR = C(1,I)
23300              DO 940 J=1,JMAX
23400          910  PR = JPR(J,NPR,NSECTION)
23500              IF(15W.EQ.17) GO TO 917

```

```

23600      PEX = 0.5*(C(J,I)+CPR)*(PR-PPR)
23700      GO TO 918
23800      917  PEX = .5*(C(J,I) + CPR)*(PR-PPR)**2
23900      918  IF(J-11930,920,930)
24000      920  C(I,I) = PEX
24100      GO TO 940
24200      930  CPR = C(I,I)
24300      C(J,I) = C(J-1,I) + PEX
24400      940  PPR = PR
24500      950  CONTINUE
24600      IF(ISSW(2))1975,1500,1500
24700      975  WRITE(4,425)KRR,ISW,JSW,KLTST
24800      GO TO 1500
24900      C ***** #10 SUBTRACT REFERENCE VALUE *****
25000      1000 DO 1090 I=JSW,KLTST
25100      CREF = C(JREF,I)
25200      DO 1040 J=1,JMAX
25300      1010 C(J,I) = CREF - C(J,I)
25400      1040 CONTINUE
25450      IF(ISW.EQ.22) GO TO 1425
25500      1050 CONTINUE
25700      IF(ISSW(2))1075,1500,1500
25800      1075  WRITE(4,425)KRR,ISW,JSW,KLTST
25900      GO TO 1500
26000      C ***** #11 ADD COLUMNS *****
26100      1100  IF(JSW)1110,1120,1110
26200      1110  WRITE(KTTX,1115)JC1,CR1,JC2,CR2,JC3,CR3,JC4,CR4
26300      READ(KIN,*)JC1,CR1,JC2,CR2,JC3,CR3,JC4,CR4
26400      GO TO 1500
26500      1115  FORMAT(1H,*,JC1,CR1,JC2,CR2,JC3,CR3,JC4,CR4,/,4E14,E12.4)
26600      1120 DO 1125 J=1,JMAX
26700      1125 C(J,JC1) = CR1*C(J,JC1)+CR2*C(J,JC2)+CR3*C(J,JC3)
26800      Y +CR4*C(J,JC4)
26900      IF(ISSW(2))1150,1500,1500
27000      1150  WRITE(4,425)KRR,ISW,JSW,KLTST
27100      WRITE(4,1115)JC1,CR1,JC2,CR2,JC3,CR3,JC4,CR4
27200      GO TO 1500
27300      C *****
27400      1200 RETURN
27500      C ***** #13 MULTIPLY UP TO 3 COLUMNS *****
27600      1300  I=-1
27700      J=-1
27800      K=-1
27900      CON1=1
28000      CON2=1
28100      CON3=1
28200      WRITE (KTTX,1310)
28300      1310  FORMAT(1H, 'INPUT COLUMN NUMBERS UP TO 3 VALUES, AND CORRESPONDING
28400      X MULTIPLICATIVE CONSTANTS')
28500      READ(KIN,*)I,J,K,CON1,CON2,CON3
28600      DO 1390 IREC=1,JMAX
28700      IF(I.LE.0)GO TO 1500
28800      A=C(IREC,I)*CON1
28900      IF(J.LE.0)GO TO 1380
29000      B=C(IREC,J)*CON2
29100      IF(K.LE.0)GO TO 1381
29200      C=C(IREC,K)*CON3
29300      GO TO 1385
29400      1380 CONTINUE
29500      R=1.
29600      1381 CONTINUE

```

```

29700      CO=1.
29800      1389  C(IREC,I)=A*B*CO
29900      1390  CONTINUE
30000      IF(ISSW(2))1395,1500,1500
30100      1395  WRITE(4,425)KRW,ISW,JSW,KLIST
30200      WRITE(4,1311)I,J,K,CON1,CON2,CON3
30300      1311  FORMAT(1H,3(I3,2X),3(F12.6,2X))
30400      GO TO 1500
30500      C ***** BRANCH 14--OUTPUT IN MAP FORMAT *****
30600      1400  KTO = 60
30700      JREC = 1
30800      WRITE(KTTX,1404)OUTNAME
30825      IF(INOYES(KYN,KTTYX).EQ.1)THEN
30850      READ(KIN,1402)OUTNAME
30862      ENDIF
30875      1402  FORMAT(A12)
30887      WRITE(KTTYX,1403)JREC1,JREC2
30900      READ(KIN,*)JREC1,JREC2
31000      WRITE(KTTYX,1401)OUTNAME
31050      OPEN(UNIT=KTO,NAME=OUTNAME,TYPE='NEW')
31066      JSW=1
31082      KLIST=6
31100      GO TO 100
31206      142  IF(IREC.NE.JREC1)GO TO 102
31212      1425  LFILE = 0
31218      IDSTN(1:2)=GNAME(NST)(1:2)
31224      IDSTN(3:5)=GNAME(NST)(6:8)
31236      IF(ISW.NE.22) GO TO 1430
31242      DO 145 KREC=JREC1,JREC2
31248      VR1 = C(KREC,JSW)*1.E-3
31254      VR2=C(KREC,3)
31260      VR3=C(KREC,4)
31266      VR4=C(KREC,5)
31272      VR5=C(KREC,6)
31278      KP=JPR(KREC,NPR,NSECTION)
31284      WRITE(KTO,1421)IDSTN,KP,XLAT,XLONG,VR1,VR2,VR3,VR4,VR5
31290      145  CONTINUE
32183      IF(NST.EQ.NO)LFILE=1
32200      GO TO 1490
32300      1430  VR1 = VRBL(NV(1))
32400      VR2 = VRBL(NV(2))
32500      VR3 = VRBL(NV(3))
32550      1490  CONTINUE
32552      KP=JPR(KREC,NPR,NSECTION)
32554      IDSTN(1:2)=GNAME(NST)(1:2)
32577      IDSTN(3:5)=GNAME(NST)(6:8)
32600      WRITE(KTO,1421)IDSTN,KP,XLAT,XLONG,VR1,VR2,VR3,VR4,VR5
32800      1421  FORMAT(1H,A5,I6,2(F8.2),4(F8.3),F8.3)
32900      IF((ISW.EQ.22).AND.(LFILE.EQ.0)) THEN
32912      KLIST=6
32924      GO TO 200
32936      ENDIF
32950      IF((ISW.EQ.22).AND.(LFILE.EQ.1))WRITE(KTO,1422)
32958      1422  FORMAT(/)
32979      CLOSE(UNIT=KTO)
33000      IF(LFILE)200,102,200
33100      1401  FORMAT(1H,'NEW OUTPUT FILE NAME IS ',A12)
33150      1403  FORMAT(1H,'LEVEL NUMBERS ARE ',2I3)
33175      1404  FORMAT(1H,'INPUT NEW OUTPUT FILE NAME (Y OR N) ',A12)
33200      C ***** BRANCH 0--SHORT DOCUMENTATION *****
33400      1550  GO TO 1500

```

```

34200 C *****
34300 1500 WRITE(KTIX,1505)KBR,ISW,JSW,KLIST
34400 1505 FORMAT(1H,'AVRGS:KBR,ISW,JSW,KLIST',7,5X,314,151)
34500 KLIST = 6
34600 READ(KIN,*)KBR,ISW,JSW,KLIST
34700 IF(KBR.EQ.0) GO TO 2300
34800 IF(KBR.GT.22)GO TO 1500
34900 GO TO 10
35000 C ***** #16 TAKE C(IREC,J) **X *****
35100 1600 IF(JSW)1610,1620,1610
35200 1610 WRITE(KTIX,1611)
35300 READ(KIN,*)X,J
35400 GO TO 1500
35500 1620 DO 1630 IREC=1,JMAX
35600 C(IREC,J)=ABS(C(IREC,J))**X
35700 1630 CONTINUE
35800 IF(ISSW(2))1650,1500,1500
35900 1650 WRITE(4,425)KBR,ISW,JSW,KLIST
36000 WRITE(4,1651)X,J
36100 1651 FORMAT(1H,'F12.6,2X,I2)
36200 GO TO 1500
36300 1611 FORMAT(1H,'INPUT EXPONENT,COLUMN')
36400 C ***** BRANCH 17--SUMMATION OF ERRORS OVER P *****
36500 1700 GO TO 900
36600 1755 FORMAT(1H,'2(I7,2X)')
36700 C ***** #18--DELP INTO C(IREC,5) *****
36800 1800 PPR=0.0
36900 DO 1810 J=1,JMAX
37000 PR=JPR(J,NPR,NSECTION)
37100 DELP=PR-PPR
37200 PPR=PR
37300 C(J,5)=DELP
37400 1810 CONTINUE
37500 IF(ISSW(2))1825,1500,1500
37600 1825 WRITE(4,425)KBR,ISW,JSW,KLIST
37700 GO TO 1500
37800 C ***** #19-- EXCHANGE COLUMNS *****
37900 1900 WRITE(KTIX,1910)
38000 1910 FORMAT(1H,'INPUT COLUMN NUMBERS TO BE EXCHANGED')
38100 READ(KIN,*)I,J
38200 DO 1920 IREC=1,JMAX
38300 CIREC = C(IREC,I)
38400 C(IREC,I) = C(IREC,J)
38500 C(IREC,J) = CIREC
38600 1920 CONTINUE
38700 IF(ISSW(2))1925,1500,1500
38800 1925 WRITE(4,425)KBR,ISW,JSW,KLIST
38900 WRITE(4,1755)I,J
39000 GO TO 1500
39100 C ***** CHANGE SINGLE ELEMENT OF C -- #20
39200 2000 WRITE(KTIX,2010)
39300 2010 FORMAT(1H,'INPUT COLUMN, ROW, NEW VALUE')
39400 READ(KIN,*)I,J,XCHG
39500 C(J,I) = XCHG
39600 IF(ISSW(2))2025,1500,1500
39700 2025 WRITE(4,425)KBR,ISW,JSW,KLIST
39800 WRITE(4,2030)I,J,XCHG
39900 2030 FORMAT(1H,'2(I2,2X),F12.6)
40000 GO TO 1500
40100 C ***** COMPUTE STD. DEV. (X) IN COL 1 FOR X-BAR, X*X-BAR IN COL 4,3 ***
40200 2100 IF(ISSW(2))2110,2120,2120

```

```

40300 2110 WRITE(4,425)KBR,ISW,JSW,KLIST
40400 2120 DO 2130 IREC = 1,JMAX
40500      Z = C(IREC,4)
40600      C(IREC,1) = SORT(IND*(C(IREC,3)-Z*Z)/(IND-1))
40700 2130 CONTINUE
40800      GO TO 1500
40900 C *****22--CALCULATE DH FOR EACH STATION--OUTPUT IN MAP FORMAT
41000 2200 GO TO 1400
41100 C ***** BRANCH 0--SHORT DOCUMENTATION--AVRGS *****
41200 2300 OPEN(UNIT=50,NAME='AVRGS.DOC',TYPE='OLD',READONLY)
41300      DO 2350 N=1,200
41400          READ(50,2325,END=2355)(DOC(I),I=1,9)
41500          WRITE(KYTX,2330)(DOC(I),I=1,9)
41600 2350 CONTINUE
41700 2325 FORMAT(9A8)
41800 2330 FORMAT(1H,9A8)
41900 2355 CLOSE (UNIT=50)
42000      GO TO 1500
42100      END

```

```

100 C TABLE SUBPROG PEPLT ***** OCT 27 1977 *****
200 SUBROUTINE TABLE
300 C *****
400 C
500 C TO COMPUTE AND PLOT POTEN VARIABLES.
600 C JUNE 27 1976 N FOFONOFF
700 C VAX VERSION--NOV 1980 N.BRAY
800 C
900 INCLUDE 'COMPEPLT.FOR'
950 DIMENSION VAR(100,9)
1000 C
1100 C PR00
1200 IF(IISW-11600,15,10
1300 10 NOSTN = 0
1400 ND = 1
1500 NX1=1
1600 NX2=64
1700 NY1=2
1800 NY2=12
1900 A1=1.
2000 A2=2
2100 B1=1.
2200 B2=-.003
2600 X2DIM=3.
2700 Y2DIM=3000.
2720 XMIN=-110.
2740 XMAX=-40.
2760 YMIN=-100.
2780 YMAX=-30
2800 15 WRITE(KTTX,1505)ND,PMIN,PMAX,X2DIM,Y2DIM
2900 1505 FORMAT(1H,'NO. STATIONS:',ND,PMIN,PMAX,X2DIM,Y2DIM,' ',Y4,4F7.0
3000 READ(KIN,*)ND,PMIN,PMAX,X2DIM,Y2DIM
3100 JMIN=1
3200 WRITE(KTTX,7000)
3300 READ(KIN,*)JMIN
3400 WRITE(KTTX,7010)PLABL
3500 IF(N0YES(KIN,KTTX).EQ.1)THEN
3600 READ(KIN,7020)PLABL
3700 CALL STRIP(PLABL)
3800 ENDIF
4100 WRITE(KTTX,7040)XMIN,XMAX,YMIN,YMAX
4200 READ(KIN,*)XMIN,XMAX,YMIN,YMAX
4300 CALL ACSETF(6HX/MIN.,XMIN)
4400 CALL ACSETF(6HX/MAX.,XMAX)
4500 CALL ACSETF(6HY/MIN.,YMIN)
4600 CALL ACSETF(6HY/MAX.,YMAX)
4800 WRITE(KTTX,7050)XLABL
4900 IF(N0YES(KIN,KTTX).EQ.1)THEN
5000 READ(KIN,7020)XLABL
5100 CALL STRIP(XLABL)
5200 ENDIF
5300 WRITE(KTTX,7060)YLABL
5400 IF(N0YES(KIN,KTTX).EQ.1)THEN
5500 READ(KIN,7020)YLABL
5600 CALL STRIP(YLABL)
5700 ENDIF
5800 CALL ACSETF(6HFRAME.,2.)
5900 RETURN
6000 C ***** PLOT RELATED FORMATS *****
6100 7000 FORMAT(1H,'INPUT INDEX OF FIRST POINT:')
6200 7010 FORMAT(1H,'CHANGE PLOT LABEL OLD LABEL IS:',Y4,4F7.0,10A4)

```

```

6300 7020 FORMAT(10A4)
6500 7040 FORMAT(1H , 'CURRENT VALUES OF XMIN,XMAX,YMIN,YMAX:',/,4F10.3)
6600 7050 FORMAT(1H , 'CHANGE X-AXIS LABEL? OLD LABEL IS:',/,4H ,10A4)
6700 7060 FORMAT(1H , 'CHANGE Y-AXIS LABEL? OLD LABEL IS:',/,4H ,10A4)
7100 600 CONTINUE
7200 C
7300 18 DO 180 J=4,6
7400 DO 180 I=1,100
7500 180 C(I,J) = 0.0
7600 DO 101 K=1,ND
7700 READ(12,1010,END=106)M,GNAME(K),WT
8000 1010 FORMAT(I4,A12,F5.2)
8100 GNAME(K)(9:12)=' .AVG'
8200 101 CONTINUE
8250 106 IF(ND.GT.K-1)ND=K-1
8300 REWIND 12
8400 90 DO 175 JST=1,ND
8500 IEOF=0
8600 OPEN(UNIT=KTP,NAME=GNAME(JST),READONLY,TYPE='OLD',FORM=
8700 * 'UNFORMATTED',ERR=175)
8800 95 CALL DATA(1,IEOF)
8900 IF(IEOF.EQ.-1)GO TO 111
9000 X = A1*VRBL(NX1)+A2*VRBL(NX2)+A3*VRBL(NX3)+A4*C(IREC,1)
9100 Y = B1*VRBL(NY1)+B2*VRBL(NY2)+B3*VRBL(NY3)+B4*C(IREC,2)
9200 Z = C1*VRBL(NZ1)+C2*VRBL(NZ2)+C3*VRBL(NZ3)+C4*C(IREC,3)
9300 XPR=A1*VRBL(NX1)
9400 YPR=B1*VRBL(NY1)
9500 WT = 1.0
9509 VAR(IREC,1)=PF
9518 VAR(IREC,2)=XPL
9527 VAR(IREC,3)=YPL
9536 DO 950 M=1,6
9545 VAR(IREC,M+3)=VRBL(NV(M))
9554 950 CONTINUE
9578 IF(ISSW(10).EQ.1)THEN
9581 WRITE(KOUT,1421)(VAR(IREC,K),K=1,9)
9587 ENDIF
9590 1421 FORMAT(9F8.3)
9595 1422 FORMAT(//)
9600 940 IF(ISSW(15))980,985,985
9700 980 WT = WGT
9800 985 C(IREC,4)=X
9900 C(IREC,5)=Y
10000 C(IREC,6)=Z
10100 105 IF(ISSW(12))110,113,113
10200 110 WRITE(KLIST,*)PF,X,Y,Z
10300 1100 FORMAT(2I4,A2,I4,F7.0,3F10.4)
10400 113 GO TO 95
10500 111 CONTINUE
10600 IF(ISSW(10).EQ.-1.AND.JST.EQ.ND)
10700 *WRITE(KOUT,1422)
11300 IF(ISSW(6).EQ.-1)GO TO 175
11400 171 IJM = 0
11500 DO 172 K=JMIN,IREC
11600 IJM=IJM+1
11700 XDAT(IJM,1) = C(K,4)
11800 YDAT(IJM,1) = C(K,5)
11900 172 CONTINUE
12000 IF(JST.GT.1)THEN
12100 CALL AGSETF(11HBACKGROUND.,4.)
12200 ENDIF

```

```

12300 CALL FRSTPT(XDAT(1,1),YDAT(1,1))
12400 CALL ANOTAT(XLABL,YLABL,0,0,0,0.)
12500 CALL EZMXY(XDAT,YDAT,JDIM,1,IJN,PLABL)
12550 IF(ISSW(5).EQ.-1)GO TO 175
12600 YPR2=YPR+B2*X2DIM
12700 XPR2=XPR-A2*X2DIM
12800 XPR3=XPR+A2*X2DIM
12900 CALL LINENCAR(XPR,YPR,XPR,YPR2)
13000 CALL LINENCAR(XPR2,YPR,XPR3,YPR)
13100 175 CONTINUE
13150 IF(ISSW(6).EQ.-1)GO TO 178
13200 CALL FRAME
13300 CALL AGSETF(6HY/MIN.,1.E36)
13400 CALL AGSETF(6HX/MAX.,1.E36)
13500 CALL AGSETF(6HY/MIN.,1.E36)
13600 CALL AGSETF(6HY/MAX.,1.E36)
13700 CALL AGSETF(11HBACKGROUND.,1.)
13800 178 IF(ISSW(10).EQ.-1)CLOSE(UNIT=ROUT)
13900 RETURN
14000 END

```

```

100  C DATA SURR PEPLT ***** SEPT 15 1977 *****
200      SURROUTINE DATA(NSW,IEOF)
300  C *****
400  C
500  C PROGRAM TO READ AND SELECT POTEN DATA.
600  C JUNE 27 1976 N FOFONOFF
700  C VAX VERSION--NOV 1980. N.BRAY
800      INCLUDE 'COMPEPLT.FOR'
850      REAL*4 JDO
900  C
1000 C
1100      MW = 1
1200      IF(NSW)1,20,200
1300      1 CONTINUE
1400      JBUF = 46
1500      JHDR = 150
1600      JDO = 0.
1700      PMIN = 0.0
1800      PMAX = 6000.0
1900      DAY1 = 0.
2000      DAY2 = 365.
2100      XEMN = -180.0
2200      XEMX = 180.0
2300      XNMN = -90.0
2400      XNMX = 90.0
2500      ZLTO = 31.0
2600      ZLGO = 69.50
2700      IFLAG=0
2800      RETURN
2900  C
3000      20 CONTINUE
3400      172  WRITE(KTTX,173)DAY1,DAY2
3500      173  FORMAT(1H,5NDAY1:F8.3,X,5NDAY2:F8.3)
3600      READ(KIN,*)DAY1,DAY2
3700      174  WRITE(KTTX,175)XEMN,XEMX,XNMN,XNMX
3800      175  FORMAT(1H,7HE-N LIN,4F7.2)
3900      READ(KIN,*)XEMN,XEMX,XNMN,XNMX
4000      WRITE(KTTX,177)ZLTO,ZLGO,JDO,PMIN,PMAX
4100      177  FORMAT(9H ORYGIN: ,2(X,F8.3),X,4HJDO:F8.2,10HPMIN,PMAX ,2F7.1)
4200      READ(KIN,*)ZLTO,ZLGO,JDO,PMIN,PMAX
4300      RETURN
4400  C
4600      200  IF(IFLAG.EQ.1)GO TO 212
4700      READ(KTP,END=280)KMDG
4800      IFLAG=1
4875  C
5000      251  IF(XDAY-DAY1)280,252,252
5100      252  IF(DAY2-XDAY)280,254,254
5200      254  IF(XLONG-XEMN)280,256,256
5300      256  IF(XEMX-XLONG)280,258,258
5400      258  IF(XLAT-XNMN)280,260,260
5500      260  IF(XNMX-XLAT)280,262,262
5600      262  YPL = -111.12*(XLONG-ZLGO)*COS((XLAT+ZLTO)/114.592)
5650      *   + JDO*FLOAT(ICOM)
5700      YPL = 111.12*(XLAT-ZLTO)
5800      DAY = XDAY
5890      212  READ(KTP,END=280)KBUF
5900      2615 IF(ISSW(13))2620,263,263
6000      2620 IF(IREC -1)2630,2625,2630
6100      2625 WRITE(KOUT)KMDG
6200      2630 WRITE(KOUT)KBUF

```

```
6300 263 CONTINUE
6700 270 IF(PF-PMIN)200,272,272
6800 272 IF(PMAX-PF)200,274,274
6900 274 RETURN
7000 280 IEND=-1
7100      CLOSE(UNIT=KTP)
7200      IFLAG=0
7300      RETURN
7400      END
```

```

50  C PEPLY SUBR ***** PEPLYSUB SEPT 15 1977*****
100 C VAX VERSION NOV 1980. N.BRAY.
150  FUNCTION POLY(V0,DVM,CP,N,VHN,VMX)
200 C *****
250  C
300  C TO EVALUATE POLYNOMIAL OF ORDER N-1 WITH COEFF CP(I).
350  C
400  C JAN 28 1976 N. FOFONOFF
450  C
500  DIMENSION CP(1)
550  C
600  V = V0
650  TF(V-VHN)I,2,2
700  1 V = VMN
750  2 TF(VMX-V)3,4,4
800  3 V = VMX
850  4 POLY = 0.0
900  X = V - DVM
950  DO 10 I=1,N
1000 NI = N - I + 1
1050 10 POLY = POLY*X + CP(NI)
1100  RETURN
1150  END
1200 C DPDV FCN *****
1250  FUNCTION DPDV(V0,DVM,CP,N,VHN,VMX)
1300 C *****
1350 C
1400 C TO COMPUTE DERIVATIVE OF POLYNOMIAL
1450 C
1500 C JAN 28 1976 N. FOFONOFF
1550 C
1600 DIMENSION CP(1)
1650 C
1700 V = V0
1750 TF(V-VHN)I,2,2
1800 1 V = VMN
1850 2 TF(VMX-V)3,4,4
1900 3 V = VMX
1950 4 NMI = N - 1
2000 X = V - DVM
2050 DPDV = 0.0
2100 DO 20 I = 1,NMI
2150 NMI = N - I
2200 20 DPDV = DPDV*X + FLOAT(NMI)*CP(NMI+1)
2250  RETURN
2300  END
2350 C BND FCN *****
2400  FUNCTION BND(Z,ZMIN,ZMAX)
2450 C *****
2500 C
2550 C TEST AND LIMIT VARIABLES.
2600 C
2650  BND = Z
2700  IF(Z-ZMIN)10,20,20
2750  10 BND = ZMIN
2800  RETURN
2850  20 IF(ZMAX-Z)30,40,40
2900  30 BND = ZMAX
2950  40 RETURN
3000  END
3050 C JPR FCN ***** PTSRI *****

```

```

3100      FUNCTION JPR(IREC,NPR,NS)
3150      C *****
3200      C
3250      C GENERATES PRESSURES CORRESPONDING TO IREC.
3300      C
3350      C OCT 28 1975 N FOFONOFF
3400      C
3450      DIMENSION NPR(1)
3500      C
3550      DO 100 J=1,NS
3600      IF(IREC.LT.NPR(J))THEN
3650      JPR=NPR(NS+J)*(IREC-NPR(2*NS+J))
3700      RETURN
3750      ENDIF
3800      100 CONTINUE
3850      JPR=NPR(2*NS)*(IREC-NPR(3*NS))
3900      RETURN
3950      END
4000      C*****
4050      SUBROUTINE STRIP(A)
4100      C*****
4150      C
4200      C STRIPS TRAILING BLANKS AND PUTS A $ AT THE END OF CHARACTERS
4250      C IN ARRAY A FOR CONFORMANCE WITH NCAR PLOT PACKAGE LABELS
4300      C N.BPAY 17NOV80
4350      C
4400      DIMENSION A(1)
4450      C
4500      B=' '
4550      DO 100 J=1,10
4600      K=10-J+1
4650      IF(A(K)-B)200,100,200
4700      100 CONTINUE
4750      200 NCH=K+1
4800      IF(NCH.GT.10)NCH=10
4850      A(NCH)='$ '
4900      RETURN
4950      END
5000      C PEPLY SURR =***** PESB2 4 MAY 1979 *****
5050      C D2PDV FN--SECOND DERIVATIVE OF POLYNOMIAL
5100      FUNCTION D2PDV(DV,DVN,CP,N,VHN,VHX)
5150      C *****
5200      C
5250      C
5300      C JAN 28 1976 N. FOFONOFF
5350      C
5400      DIMENSION CP(1)
5450      C
5500      V = V0
5550      IF(V-VHN)1,2,2
5600      1 V = VHN
5650      2 IF(VHX-V)3,4,4
5700      3 V = VHX
5750      4 NM1 = N - 2
5800      X = V - DVN
5850      D2PDV = 0.0
5900      DO 20 I=1,NM1
5950      NM1 = N - I
6000      NM12 = NM1 - I
6050      20 D2PDV = D2PDV*X + FLOAT(NM1+NM12)*CP(NM1+1)
6100      RETURN

```

```

6150      END
6200      C SEAWATER PROPERTIES *****
6250      C
6300      C SGO *****
6350      FUNCTION SGO(S)
6400      C *****
6450      C SIGMA-O KNUDSEN
6500      C FEB 15 1976 N. FOFONOFF
6550      C
6600      SGO = ((16.76786136E-6*S-4.8249614E-4)*S+0.814876577)*S
6650      X -0.0934458632
6700      RETURN
6750      END
6800      C SGT FCN *****
6850      FUNCTION SGT(T,S,SG)
6900      C *****
6950      C SIGMA-T KNUDSEN
7000      C FEB 15 1976 N FOFONOFF
7050      C
7100      SG = SGO(S)
7150      Z0 SGT = (((-1.43803061E-7*T-1.98248399E-3)*T-0.545939111)*T
7200      X +4.53168426)*T)/(T+67.26)+(((11.667E-8*T-8.164E-7)*T
7250      X +1.803E-5)*T)*SG+((-1.0843E-6*T+9.8185E-5)*T-4.7867E-3)*T
7300      X +1.0)*SG
7350      RETURN
7400      END
7450      C EQUATION OF STATE FOR SEAWATER EOS80
7500      C *****
7550      REAL FUNCTION EOS80(P1,T,S)
7600      C *****
7650      C EQUATION OF STATE FOR SEAWATER PROPOSED BY JPOTS 1980
7700      C REFERENCES
7750      C MILLERO ET AL 1980, DEEP-SEA RES., 27A, 255-264
7800      C JPOTS NINTH REPORT 1978, TENTH REPORT 1980
7850      C UNITS:
7900      C      PRESSURE      P      BARS
7950      C      INPUT PRESSURE PT      DECBARS
8000      C      TEMPERATURE      T      DEG CELSIUS (1PTS-68)
8050      C      SALINITY      S      NSU (1PSS-78)
8100      C      DENSITY      RHO      KG/M**3
8150      C      SPEC. VOL.      EOS80      M**3/KG
8200      C CHECK VALUE: EOS80 = 9.435561E-4 M**3/KG FOR S = 40 NSU,
8250      C T = 40 DEG C, P = 1000 BARS.
8300      C
8350      C N FOFONOFF REVISED OCT 7 1980
8400      C MODIFIED TO TAKE DB INPUT PRESSURE, AND OUTPUT IN CM**3/GM 29NOV80
8450      C N.BRAY
8500      REAL P1,P,T,S,RHO,SR,R1,R2,R3,R4
8550      REAL A,B,C,D,E,A1,B1,AW,BW,K,KO,KW
8600      C EQUIV
8650      EQUIVALENCE (E,D,B1,R4),(BW,B,R3),(C,A1,R2)
8700      EQUIVALENCE (AW,A,R1,R0),(KW,KO,K)
8750      C CONVRT PRESSURE TO BARS AND SQUARE ROOT SALINITY.
8800      P = P1*.1
8850      SR = SQRT(ABS(S))
8900      C COMPUTE DENSITY PURE WATER AT ATM PRESSURE
8950      R1 = (((16.536332E-9*T-1.120083E-6)*T+1.001685E-4)*T
9000      X -9.095290E-3)*T+8.793952E-2)*T+999.842594
9050      C SEAWATER DENSITY ATM PRESS.
9100      R2 = ((15.3875E-9*T-8.2467E-7)*T+7.6438E-5)*T-4.0899E-3)*T
9150      X+8.24493E-1

```

```

9200      R3 = (-1.6546E-6*T+1.0227E-4)*T-5.72466E-3
9250      R4 = 4.8314E-4
9300      RHO = (R4*S + R3*SR + R2I*S + R1
9350      C SPECIFIC VOLUME AT ATMOSPHERIC PRESSURE
9400      ALPHA = 1.E+3/RHO
9450      EOS80 = ALPHA
9500      IF(P.EQ.0.0)RETURN
9550      C COMPUTE COMPRESSION TERMS
9600      F = (9.1697E-10*T+2.0816E-8)*T-9.9348E-7
9650      BW = (5.2787E-8*T-6.12293E-6)*T+8.50935E-5
9700      R = RW + F+S
9750      C
9800      D = 1.91075E-4
9850      C = (-1.6078E-6*T-1.0981E-5)*T+2.2838E-3
9900      AW = ((-5.77909E-7*T+1.16092E-4)*T+1.43713E-3)*T
9950      X+3.239908
10000     A = (D*SR + C)*S + AW
10050     C
10100     RI = (-5.3009E-4*T+1.6483E-2)*T+7.944E-7
10150     AI = ((-6.1670E-5*T+1.09987E-2)*T-0.603459)*T+54.6746
10200     KW = (((-5.155288E-5*T+1.360477E-2)*T-2.3271054)*T
10250     X+149.4206)*T+19652.21
10300     KO = (BI*SR + AI)*S + KW
10350     C
10400     K = (R*P + AI)*P + KO
10450     ALPHA = ALPHA*(1.0 - P/K)
10500     EOS80 = ALPHA
10550     RETURN
10600     END
10650     C V350P FCN ***** OCT 7 1980 *****
10700     REAL FUNCTION V350P(P1)
10750     C *****
10800     C SPECIFIC VOLUME (CM**3/GM) FOR S = 35 NSU TIPSS-7814
10850     C TEMPERATURE 0 DEG CELSIUS (1PTS-68) AND PRESSURE IN DECIBARS.
10900     C EQUATION DERIVED FROM EOS80
10950     C CHECK VALUE: V350P = 9.337431E-4 M**3/KG FOR P = 1000 BARS.
11000     C MODIFIED TO ACCEPT INPUT PRESSURE IN DB AND OUTPUT SP.VOL IN
11050     C CM**3/GM 28 NOV 80. N BRAY.
11100     P = P1*.1
11150     ALPHA = 9.72662E-4*(1.0-P/(121582.27+(13.35941+5.032E-5*P)*P))
11200     ALPHA = 1.E+3*ALPHA
11250     V350P = ALPHA
11300     RETURN
11350     END
11400     C DEPTH FCN ***** OCT 7 1980 *****
11450     REAL FUNCTION DEPTH(P1,LAT)
11500     C *****
11550     C DEPTH IN METERS FROM PRESSURE IN DECIBARS USING
11600     C SAUNDERS AND FOFONOFF'S METHOD.
11650     C DEEP-SEA RES., 1976,23,109-111.
11700     C FORMULA REFITTED FOR EOS80
11750     C
11800     REAL LAT
11850     C
11900     P = P1*.1
11950     X = SIN(LAT/57.29578)
12000     X = X*X
12050     GR = 9.780318*(1.0+(5.2788E-3+2.36E-5*X)*X) + 1.092E-5*P
12100     DEPTH = (((-1.82E-11*P+2.279E-7)*P-2.2512E-3)*P+97.2659)*P
12150     DEPTH = DEPTH/GR
12200     RETURN

```

```

12250      FND
12300      C ATG FCN *****
12350      FUNCTION ATG(P,T,S)
12400      C *****
12450      C
12500      C ADIAPATIC TEMPERATURE GRADIENT. BRYDEN 1973.
12550      C
12600      DS = S - 35.0
12650      ATG = (((-2.1687E-16*T+1.8676E-14)*T-4.6206E-13)*P
12700      X+((12.7759E-12*T-1.1351E-10)*DS+((-5.4481E-14*T
12750      X+8.733E-12)*T-6.7795E-10)*T+1.8741E-8))*P
12800      X+(-4.2393E-8*T+1.8932E-6)*DS
12850      X+((16.6228E-10*T-6.836E-8)*T+8.5258E-6)*T+3.5803E-5
12900      RETURN
12950      FND
12954      C DVA FCN ***** PTSB1 *****
12958      FUNCTION DVA(P,T,S)
12962      C *****
12966      C
12970      C SPECIFIC VOLUME ANOMALY
12974      C
12978      DVA = SVAN(P,T,S,SPV)
12982      RETURN
12986      FND
12990      C
13000      C SVAN FCN *****
13050      FUNCTION SVAN(P,T,S,V)
13100      C *****
13150      C SPECIFIC VOLUME ANOMALY*1E5
13200      C FEB 15 1976 N FOFONOFF
13250      V = F0580(P,T,S)
13300      SVAN = 1.0E5*(V - V350P(P))
13450      RETURN
13500      END
13550      C THETA FCN *****
13600      FUNCTION THETA(P0,T0,S,PF)
13650      C *****
13700      C
13750      C TO COMPUTE LOCAL POTENTIAL TEMPERATURE AT PF
13800      C FOURTH-ORDER RUNGE-KUTTA INTEGRATION USING STEPS OF 100 DB
13850      C OR LF55. (RALSTON-WILF VOL 1 P115, EQ 26)
13900      C
13950      C OCT 12 1975 N. FOFONOFF
14000      C
14050      P = P0
14100      T = T0
14150      H = PF - P
14200      N = ABS(H)/1000.0 + 1.0
14250      H = H/FLD0AT(N)
14300      DO 10 I=1,N
14350      XK = H*ATG(P,T,S)
14400      T = T + 0.5*XK
14450      Q = XK
14500      P = P + 0.5*H
14550      XK = H*ATG(P,T,S)
14600      T = T + 0.29289322*(XK-Q)
14650      Q = 0.58578644*XK + 0.121320344*Q
14700      XK = H*ATG(P,T,S)
14750      T = T + 1.707106781*(XK-Q)
14800      Q = 3.414213562*XK - 4.121320344*Q
14850      P = P + 0.5*H

```

```

14900      XK = H*ATG(P,T,S)
14950      T = T + (XK-2.0*Q)/6.0
15000      TO CONTINUE
15050      THETA = T
15100      RETURN
15150      END
15200      C T68 FCN *****
15250      FUNCTION T68(T)
15300      C *****
15350      C TO CONVERT T-48 TO T-68 TEMPERATURE SCALE
15400      C FEB 15 1976 N FOFONOFF
15450      C
15500      T68 = T - 4.4E-6*T*(100.0-T)
15550      RETURN
15600      END
15650      C T48 FCN *****
15700      FUNCTION T48(T)
15750      C *****
15800      C TO CONVERT T-68 TO T-48 TEMPERATURE SCALE
15850      C FEB 15 1976 N FOFONOFF
15900      C
15950      T48 = T + 4.4E-6*T*(100.0-T)
16000      RETURN
16050      END
16100      C DVT FCN *****
16150      FUNCTION DVT(P,T,S)
16200      C *****
16250      C DERIVATIVE OF SPECIFIC VOL. WITH TEMPERATURE*1E5
16300      C FEB 20 1976 N FOFONOFF
16350      C
16400      H = 0.25
16450      DVT = (5.0E4/H)*(EOS80(P,T+H,S)-EOS80(P,T-H,S))
16500      RETURN
16550      END
16600      C DVDS FCN *****
16650      FUNCTION DVDS(P,T,S)
16700      C *****
16750      C DERIVATIVE OF SPECIFIC VOL. WITH SALINITY*1E5
16800      C FEB 20 1976 N FOFONOFF
16850      C
16900      H = 0.5
16950      DVDS = (5.0E4/H)*(EOS80(P,T,S+H)-EOS80(P,T,S-H))
17000      RETURN
17050      END
17100      C DVDP FCN *****
17150      FUNCTION DVDP(P,T,S)
17200      C *****
17250      C ADIABATIC DERIVATIVE OF SPEC. VOL. WITH PRESSURE*1E5
17300      C FEB 20 1976 N FOFONOFF
17350      C
17400      H = 6.0
17450      DVDP = (5.0E4/H)*(EOS80(P+H,T,S)-EOS80(P-H,T,S))
17500      X = ATG(P,T,S)*DVT(P,T,S)
17550      RETURN
17600      END
17650      C DKDT FCN *****
17700      FUNCTION DKDT(P,T,S)
17750      C *****
17800      C ADIABATIC COMPRESSIBILITY TEMP DERIVATIVE
17850      C FEB 20 1976 N FOFONOFF
17900      C

```

```

17950      M = 1.0
18000      DKDT = (0.5/H)*(DVDP(P,T+H,S) - DVDP(P,T-H,S))
18050      RETURN
18100      END
18150      C DKDS FCN *****
18200      FUNCTION DKDS(P,T,S)
18250      C *****
18300      C ADIABATIC COMPRESSIBILITY SALINITY DERIVATIVE.
18350      C FEB 20 1976 N FOFONOFF
18400      C
18450      M = 2.0
18500      DKDS = (0.5/H)*(DVDP(P,T,S+H) - DVDP(P,T,S-H))
18550      RETURN
18600      END
18650      C SAL FCN *****
18700      FUNCTION SAL(P,T,D)
18750      C *****
18800      C COMPUTE SALINITY GIVEN PRESSURE,TEMPERATURE AND SPECIFIC
18850      C VOLUME ANOMALY(10**5*DELTA)
18900      C FEB 16 1976 N FOFONOFF
18950      C
19000      K = 0
19050      SAL = 35.0
19100      10 S = SAL
19150      SAL = S + (D-SVAN(P,T,S,V))/DVDS(P,T,S)
19200      K = K+1
19250      IF(K-50)20,30,30
19300      20 IF(ABS(SAL-S)-0.0005)30,10,10
19350      30 RETURN
19400      END
19450      C NOYES FUNCTION **** DEC 3 1979 *****
19500      FUNCTION NOYES(KIN,KTTX)
19550      C *****
19600      C RETURNS 1 FOR YES -1 FOR NO
19650      NOYES = 0
19700      1 READ(KIN,10)LB
19750      10 FORMAT(A2)
19800      C
19850      IF(LB.EQ.2HYES)NOYES=1
19900      IF(LB.EQ.2HNO)NOYES=-1
19950      IF(NOYES)30,20,30
20000      C FRRN
20050      20 WRITE(KTTX,100)
20100      100 FORMAT('% YES OR NO? ')
20150      GO TO 1
20200      C
20250      30 RETURN
20300      END
20350      C CYSD FILE ***** JULY 15 1977 *****
20400      C THSAL FCN ***** JULY 6 1977 *****
20450      FUNCTION THSAL(KIN,T)
20500      C *****
20550      C
20600      C TAKES UP TO 25 CUBIC SPLINES TO GENERATE A SALINITY FROM
20650      C POTENTIAL TEMPERATURE REFERRED TO THE SURFACE. INPUT DATA
20700      C CONSISTS OF LOWER SPLINE BOUNDARY FOLLOWED BY FOUR COEFFICIENTS.
20750      C COEFFICIENTS ARE FROM THE FIT OF ARMI AND BRAY (1981) TO
20800      C ISELIN AND WORTHINGTON METCALF THETA-SAL DATA.
20850      C
20900      DIMENSION C(5,25)
20950      C DATA

```

```

21000 DATA C/0.00,34.738063,0.0,0.0,0.0,
21050 *0.50,34.738053,107290,.584849E-02,-.253429E-02,
21100 *1.20,34.815152,.111753,.523726E-03,.582151E-01,
21150 *1.50,34.850297,.127785,.529320E-01,-.135379,
21200 *1.75,34.883436,.128868,-.485828E-01,-.129913,
21250 *2.00,34.910587,.802174E-01,-.146093,.228920,
21300 *2.25,34.925087,.500936E-01,.255484E-01,-.267382E-01,
21350 *2.50,34.938790,.578544E-01,.552526E-02,-.359945E-01,
21400 *2.75,34.953036,.538681E-01,-.214953E-01,-.374594E-01,
21450 *3.00,34.964575,.360969E-01,-.495364E-01,.509274E-01,
21500 *3.20,34.970220,.223936E-01,-.189292E-01,.580683E-01,
21550 *3.40,34.974406,.217901E-01,.157868E-01,.479730E-02,
21600 *3.60,34.979434,.286805E-01,.185975E-01,-.294172E-01,
21650 *3.80,34.985679,.325895E-01,.102958E-02,-.279688E-01,
21700 *4.00,34.992014,.296450E-01,-.157123E-01,.843397E-02,
21750 *4.50,35.01238,.175223E-01,.357759E-02,.114377E-02,
21800 *7.00,35.07089,.455579E-01,.104386E-01,.865592E-05,
21850 *10.00,35.30174,.108423,.105172E-01,-.763343E-03,
21900 *13.00,35.70106,.150916,.364790E-02,.310805E-04,
21950 *16.00,36.18748,.173643,.392926E-02,-.689782E-02,
22000 *19.00,36.55753,.109775E-01,-.581443E-01,.696380E-01,
22050 *21.00,36.9040118,0.0,0.0,0.0,15*0.0/

```

```

DATA KNOTS/22/

```

```

22100 C
22150
22200 C
22250 250 X = 0.0
22300 DO 310 I=1,KNOTS
22350 DT = C(1,I) - T
22400 IF(DT)305,320,320
22450 305 X = -DT
22500 310 CONTINUE
22550 320 D = X
22600 TD = I-I
22650 IF(ID)325,325,330
22700 325 TD = I
22750 D = 0.0
22800 330 THSAL = ((C(5,TD)*D+C(4,TD))*D+C(3,TD))*D+C(2,TD)
22850 RETURN
22900 END

```

```

22950 C
23000 C VRBL FUNCTION: PEPLT *****
23050 FUNCTION VRBL(NVR)
23100 C *****
23150 C
23200 C PROGRAM TO SELECT POTEN VARIABLES
23250 C MAR 25 1976 N FOFONOFF
23300 C VAX VERSION--INDIAN OCEAN DATA (1976). NOV 1980. N.BRAY
23350 INCLUDE 'COMPEPLT.FOR'
23400 C
23450 C
23500 IF(NVR)10,20,30
23550 10 VRBL = 1.0
23600 RETURN
23650 20 VRBL = 0.0
23700 RETURN
23750 30 GO TO (31,32,33,34,35,36,37,38,39,40,41,42)NVR
23800 IF(NVR.GT.12)GO TO 42
23850 31 VRBL = XPL
23900 RETURN
23950 32 VRBL = YPL
24000 RETURN

```

```

24050      33 VRBL = XLAT
24100      RETURN
24150      34 VRBL = XLONG
24200      RETURN
24250      35 VRBL = ICON
24300      RETURN
24350      36 VRBL = WGT
24400      RETURN
24450      37 VRBL = DAY
24500      RETURN
24550      38 VRBL = YSHP
24600      RETURN
24650      39 VRBL = ICAST
24700      RETURN
24750      40 VRBL = N
24800      RETURN
24850      41 VRBL = NDP
24900      RETURN
24950      42 IF(NVR-48)420,43,43
25000      420 VRBL = VR(NVR-11)
25050      RETURN
25100      43 E0 = DPDV(DVF,DVN,CP,N,F1,F2)
25150      PDF = PT - PF
25200      F1 = -.050968*PDF**2/E0
25250      E2 = (F3-SF)*E0
25300      F3 = -.050968*E2*E2/E0
25350      F5 = DPDV(SF,DVN,CP,N,F1,F2)
25400      NVRX = NVR - 47
25450      DO TO(48,49,50,51,52,53,54,55,56,57,58)NVRX
25500      IF (NVRX.GT.11)GO TO 58
25550      48 VRBL = E2
25600      RETURN
25650      49 VRBL = E2*PF
25700      RETURN
25750      50 VRBL = PDF
25800      RETURN
25850      51 VRBL = SF - DVF
25900      RETURN
25950      52 VRBL = PI*SIN(XL*6/57.296)/SIN(XLAT/57.296)
26000      RETURN
26050      53 VRBL = FI
26100      RETURN
26150      54 VRBL = 0.101937*PF*DH - PE
26200      RETURN
26250      55 VRBL = E3
26300      RETURN
26350      56 VRBL = E0
26400      RETURN
26450      57 VRBL = 1.0/E0
26500      RETURN
26550      58 SHF = SAL(PF,THF,DVF)
26600      VF = EOS80(PF,THF,SHF) + 1.0
26650      THP = DPDV(DVF,DVN,CT,N,F1,F2)/E0
26700      SHP = (1.0/E0 - DVDT(PF,THF,SHF)*THP)/DVDS(PF,THF,SHF)
26750      GR = -1.981/VF**2
26800      BV1 = 100.0*GR/E0
26850      E6 = VRT13)
26900      TH1 = POLY(SF,DVN,CT,N,F1,F2) + THM
26950      SI = SAL(PF,TH1,SF)
27000      IF(NVR-67)585,582,582
27050      582 GR = (DKDT(PF,THP,SHF)*THP+DKDS(PF,THF,SHP)*SHP)

```

```

27100      E4 = -50.968*GK*PDF**2
27150      585 NVRX = NVP - 57
27200      GO TO (580,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,
27250      *76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,
27300      *97,98,20)NVRX
27350      IF(NVRX.GT.39)GO TO 20
27400      580 VRBL = BV1
27450      RETURN
27500      59 VRBL = THP
27550      RETURN
27600      60 VRBL = SHP
27650      RETURN
27700      61 VRBL = SHF
27750      RETURN
27800      62 VRBL = SHP/THP
27850      RETURN
27900      63 VRBL = PDF*PDF
27950      RETURN
28000      64 E5 = 0.572958*50RT(ABS(BV1))
28050      VRBL = SIGN(E5,BV1)
28100      RETURN
28150      65 VRBL = THETA(PF,THF,SHF,0.0)
28200      RETURN
28250      66 E5 = THETA(PF,THF,SHF,0.0)
28300      VRBL = SGT(E5,SHF,SG)
28350      RETURN
28400      67 VRBL = VF
28450      RETURN
28500      68 VRBL = -.5*GK*PDF*PDF
28550      RETURN
28600      69 VRBL = BV1 + GR*PF*GK
28650      RETURN
28700      70 VRBL = GK
28750      RETURN
28800      71 VRBL = GR*PF*GK
28850      RETURN
28900      72 Z = THETA(PF,THF,SHF,0)
28950      Z0=SHF
29000      GO TO 720
29050      73 Z=THETA(PI,TO,SO,0.0)
29100      Z0=SO
29150      720 Z1=THSAL(1,Z)
29200      VRBL=Z0-Z1
29250      RETURN
29300      74 Z = THETA(PF,THF,SHF,0.0)
29350      VRBL = Z*Z
29400      RETURN
29450      75 X = DVDP(PF,THF,SHF)
29500      Y = DVDT(PF,THF,SHF)
29550      Z = DVDS(PF,THF,SHF)
29600      VRPA = 16.*X*X + 49.0E-06*Y*Y + 25.0E-06*Z*Z
29650      VRPI = VRPA*E0*E0
29700      GO TO 770
29750      76 Z = DVDP(PF,THF,SHF)
29800      VRBL = 16.*Z*Z
29850      RETURN
29900      77 CONTINUE
29950      VRPI = Z1*Z1
30000      770 Y = VRPI + 2*PDF*PDF
30050      VRBL = .5*VRPI*E6*E6*Y
30100      RETURN

```

```

30150      78 VRPI = 3.*71*71/NDP
30200      GO TO 770
30250      79 VRPI = 16.
30300      GO TO 770
30350      80 VRPI = 71*71/NDP
30400      800 Y = .5*VRPI*(VRPI + 2*PDF)
30450      VRBL = GK**2*Y
30500      RETURN
30550      81 VRPI = 16.
30600      GO TO 800
30650      82 VRBL = -1
30700      RETURN
30750      83 VRBL = -1
30800      RETURN
30850      84 VRBL = DVDP(PF,THF,SHE)
30900      RETURN
30950      85 VRBL = DVDS(PF,THF,SHE)
31000      RETURN
31050      86 VRBL = DVDP(PF,THF,SHE)*PDF
31100      RETURN
31150      87 VRBL = -.5*PDF*PDF*E6
31200      RETURN
31250      88 VRBL = -1
31300      RETURN
31350      89 VRBL = -1
31400      RETURN
31450      90 VRBL = -1
31500      RETURN
31550      91 VRBL = (SF-F3)**2
31600      RETURN
31650      92 Z = -THP*PDF
31700      VRBL = 1/Z
31750      RETURN
31800      93 VRBL = 1/(EO*EO)
31850      RETURN
31900      94 VRBL = THI*THI
31950      RETURN
32000      95 THMM = C(IREC,4)
32050      VRBL = -(THI-THMM)/(THP*PDF)
32100      RETURN
32150      96 THMM = C(IREC,4)
32200      VRBL = -(THF-THMM)/(THP*PDF)
32250      RETURN
32300      97 VRBL = THI
32350      RETURN
32400      98 EO = 1.0/EO
32450      VRBL = -.5*E6*D2PDV(DVF,DVH,CP,W,FI,FZ)*{PDF*EO}**Z
32500      RETURN
34400      2000 END
34450      C KDAY FCN ***** KDAY5 JULY 6 1977 *****
34500      FUNCTION KDAY(10,1MO,1YR)
34550      C *****
34600      C CONVERT GREGORIAN DATE TO JULIAN DAY
34650      C USES LAST 4 DIGITS OF JULIAN DAY. ADD 2440000 TO GET
34700      C FULL JULIAN DAY.
34750      C
34800      C JULY 12 1975
34850      C
34900      TY = 1YR - 68
34950      TF(2-1MO)10,20,20
35000      10 M = 1MO - 3

```

```

35050      GO TO 30
35100      20 M = IM0 + 9
35150      IY = IY - 1
35200      30 KDAY = (1461*IY)/4 + (153*M+2)/5 + ID - 84
35250      RETURN
35300      END
35350      C KDATE ***** CYDSB JULY 7 1977. *****
35400      SUBROUTINE KDATE(KD, ID, M, IY)
35450      C *****
35500      C CONVERT JULIAN DAY TO GREGORIAN DATE
35550      C
35600      K=KD+84
35650      IY=(4*K-1)/1461
35700      ID=4*K-1-1461*IY
35750      IY = IY + 68
35800      ID=(ID+4)/4
35850      M=(5*ID-3)/153
35900      ID=5*ID-3-153*M
35950      ID=(ID+5)/5
36000      IF(M-10)20,10,10
36050      10 M=M-9
36100      IY=IY+1
36150      RETURN
36200      20 M=M+3
36250      RETURN
36300      END
36350      C VKE FCN ***** SWPR1 *****
36400      FUNCTION VKE(P,T,S)
36450      C *****
36500      C SPECIFIC VOLUME KNUDSEN/FRMAN
36550      C FEB 15 1976 N FOFONOFF
36600      C
36650      VO = 0.001*SGT(T,S,SG)
36700      VO = -VO/(1.0 + VO)
36750      20 VKE = (-4.886E-6*P/(1.0+1.83E-5*P) + ((1.5E-17*T*P
36800      X + (-6.0E-17*T+1.8E-15)*SG + (-2.0E-16*T+1.206E-14)*T
36850      X -4.248E-13)*SG + (2.14E-14*T-1.24064E-12)*T-6.68E-14)*P
36900      X + ((1.0E-12*T-4.5E-11)*SG
36950      X + (4.0E-12*T-3.28E-10)*T+1.725E-8)*SG
37000      X + ((4.0E-12*T-6.63E-10)*T+3.673E-8)*T-2.2072E-7)*P)
37050      VKE = VO + VKE*(1.0 + VO)
37100      RETURN
37150      END
37200      C DVZRO FCN *****
37250      FUNCTION DVZRO(P0,DVM,PH,CP,N,NDP,VVN,VHX,ISHP,KCAST,ICON,DELP)
37300      C *****
37350      C
37400      C TO INVERT POLYNOMIAL FOR INDEPENDENT VARIABLE.
37450      C
37500      C FEB 1 1976 N. FOFONOFF
37550      C
37600      DIMENSION CP(1)
37650      C
37700      XN = 0.0
37750      VR = 0.0
37800      PDF = P0
37850      KN = 0
37900      DV = (VMX-VVN)/FLOAT(NDP-1)
37950      DO 50 J=1,NDP
38000      V = VMX - DV*FLOAT(J-1)
38050      P = POLY(V,DVM,CP,N,VVN,VHX) + PH

```

```

38100      DP = DPDV(V,DVM,CP,N,VMN,VMX)
38150      IF (J.FO.NDP/2) THEN
38200          DP0=DP
38250          V0=V
38300      ENDIF
38350      IF(DP)5,50,50
38400      5 PD = PO - P
38450      IF(KN)7,7,9
38500      7 PPD = PD
38550      KN = KN + 1
38600      9 IF(ABS(PD)-PDF)12,10,10
38650      12 PDF = ABS(PD)
38700      VS = V
38750      10 IF(PPD*PD)15,50,50
38800      15 VR = VR + V
38850      XN = XN + 1.0
38900      PPD = PD
38950      50 CONTINUE
39000      60 IF(XN - 1.0)66,70,65
39050      65 DVZRN = VR/XN
39100      GO TO 90
39150      66 DVZRO = VS
39200      GO TO 90
39250      70 K = 0
39300      V = VR
39350      75 VP = V
39400      P = POLY(V,DVM,CP,N,VMN,VMX) + PH
39450      V = VP + (PO-P)/DPDV(V,DVM,CP,N,VMN,VMX)
39500      K = K + 1
39550      IF(K-100)80,85,85
39600      80 IF(ABS(PO-P)-0.05)85,75,75
39650      85 DVZRN = V
39700      90 IF(DVZRN-VMN)95,100,100
39750      95 CONTINUE
39800      PO=ABS((DVZRN-V0)*DP0)
39850      PPO=ABS(PO-NDP*DELP/2)
39900      WRITE(KTIX,1000)ISHP,KCAST,ICON,PO,PPD
39950      1000  FORMAT(1H,'FOR STATION ',A2,ZI3,' AT',F8.0,' DB LEVEL,
40000      * YOU SHOULD INCREASE REGRESSION INTERVAL BY ',F8.0,' DB.')
40050      DVZRN = VMN
40100      100 IF(VMX-DVZRN)105,110,110
40150      105 CONTINUE
40200      PO=ABS((DVZRN-V0)*DP0)
40250      PPO=ABS(PO-NDP*DELP/2)
40300      WRITE(KTIX,1000)ISHP,KCAST,ICON,PO,PPD
40350      DVZRO = VMX
40400      110 RETURN
40450      END

```

MANDATORY DISTRIBUTION LIST

FOR UNCLASSIFIED TECHNICAL REPORTS, REPRINTS, AND FINAL REPORTS
PUBLISHED BY OCEANOGRAPHIC CONTRACTORS
OF THE OCEAN SCIENCE AND TECHNOLOGY DIVISION
OF THE OFFICE OF NAVAL RESEARCH

(REVISED NOVEMBER 1978)

- 1 Deputy Under Secretary of Defense
(Research and Advanced Technology)
Military Assistant for Environmental Science
Room 3D129
Washington, D.C. 20301

Office of Naval Research
800 North Quincy Street
Arlington, VA 22217
- 3 ATTN: Code 483
- 1 ATTN: Code 460
- 2 ATTN: 102B
- 1 CDR Joe Spigai, (USN)
ONR Representative
Woods Hole Oceanographic Inst.
Woods Hole, MA 02543

Commanding Officer
Naval Research Laboratory
Washington, D.C. 20375
- 6 ATTN: Library, Code 2627
- 12 Defense Technical Information Center
Cameron Station
Alexandria, VA 22314
ATTN: DCA

Commander
Naval Oceanographic Office
NSTL Station
Bay St. Louis, MS 39522
- 1 ATTN: Code 8100
- 1 ATTN: Code 6000
- 1 ATTN: Code 3300
- 1 NODC/NOAA
Code D781
Wisconsin Avenue, N.W.
Washington, D.C. 20235
- 1 Mr. Michael H. Kelly
Administrative Contracting Officer
Department of the Navy
Office of Naval Research
Eastern/Central Regional Office
Building 114, Section D
666 Summer Street
Boston, MA 02210

Woods Hole Oceanographic Institution WHOI-81-70 VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM CTD DATA by Nancy Amanda Bray, 135 pages August 1981. Prepared for the Office of Naval Research under Contracts N00014-76-C-0197; NR 083-400 and N00014-79-C-0071; NR 083-004 and for the National Science Foundation under Grant OCE-77-19403 This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Foronoff (1981). The report includes examples of how the program may be used, as well as complete listings of all the required FORTRAN files.	Woods Hole Oceanographic Institution WHOI-81-70 VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM CTD DATA by Nancy Amanda Bray, 135 pages August 1981. Prepared for the Office of Naval Research under Contracts N00014-76-C-0197; NR 083-400 and N00014-79-C-0071; NR 083-004 and for the National Science Foundation under Grant OCE-77-19403 This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Foronoff (1981). The report includes examples of how the program may be used, as well as complete listings of all the required FORTRAN files.	1. Oceanic available potential energy surfaces 2. Definitions of horizontal density 3. Hydrographic data I. Bray, Nancy Amanda II. N00014-76-C-0197; NR 083-400 III. N00014-79-C-0071; NR 083-004 IV. OCE-77-19403 This card is UNCLASSIFIED	1. Oceanic available potential energy surfaces 2. Definitions of horizontal density 3. Hydrographic data I. Bray, Nancy Amanda II. N00014-76-C-0197; NR 083-400 III. N00014-79-C-0071; NR 083-004 IV. OCE-77-19403 This card is UNCLASSIFIED
Woods Hole Oceanographic Institution WHOI-81-70 VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM CTD DATA by Nancy Amanda Bray, 135 pages August 1981. Prepared for the Office of Naval Research under Contracts N00014-76-C-0197; NR 083-400 and N00014-79-C-0071; NR 083-004 and for the National Science Foundation under Grant OCE-77-19403 This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Foronoff (1981). The report includes examples of how the program may be used, as well as complete listings of all the required FORTRAN files.	Woods Hole Oceanographic Institution WHOI-81-70 VAX-11 PROGRAMS FOR COMPUTING AVAILABLE POTENTIAL ENERGY FROM CTD DATA by Nancy Amanda Bray, 135 pages August 1981. Prepared for the Office of Naval Research under Contracts N00014-76-C-0197; NR 083-400 and N00014-79-C-0071; NR 083-004 and for the National Science Foundation under Grant OCE-77-19403 This report documents the W.H.O.I. VAX-11 programs used to calculate available potential energy and related quantities from CTD data using the technique described in Bray and Foronoff (1981). The report includes examples of how the program may be used, as well as complete listings of all the required FORTRAN files.	1. Oceanic available potential energy surfaces 2. Definitions of horizontal density 3. Hydrographic data I. Bray, Nancy Amanda II. N00014-76-C-0197; NR 083-400 III. N00014-79-C-0071; NR 083-004 IV. OCE-77-19403 This card is UNCLASSIFIED	1. Oceanic available potential energy surfaces 2. Definitions of horizontal density 3. Hydrographic data I. Bray, Nancy Amanda II. N00014-76-C-0197; NR 083-400 III. N00014-79-C-0071; NR 083-004 IV. OCE-77-19403 This card is UNCLASSIFIED