United States
Environmental Protection
Agency

EPA-600/7-82-037b

May 1982

EPA

Research and Development

VERIFICATION AND TRANSFER OF
THERMAL POLLUTION MODEL
Volume II. User's Manual for
Three-dimensional Free-surface Model

Prepared for

Office of Water and Waste Management
EPA Regions 1-10

Prepared by

Industrial Environmental Research
Laboratory
Research Triangle Park NC 27711

RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into nine series. These nine broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and a maximum interface in related fields. The nine series are:

1. Environmental Health Effects Research
2. Environmental Protection Technology
3. Ecological Research
4. Environmental Monitoring
5. Socioeconomic Environmental Studies
6. Scientific and Technical Assessment Reports (STAR)
7. Interagency Energy-Environment Research and Development
8. "Special" Reports
9. Miscellaneous Reports

This report has been assigned to the INTERAGENCY ENERGY-ENVIRONMENT RESEARCH AND DEVELOPMENT series. Reports in this series result from the effort funded under the 17-agency Federal Energy/Environment Research and Development Program. These studies relate to EPA's mission to protect the public health and welfare from adverse effects of pollutants associated with energy systems. The goal of the Program is to assure the rapid development of domestic energy supplies in an environmentally-compatible manner by providing the necessary environmental data and control technology. Investigations include analyses of the transport of energy-related pollutants and their health and ecological effects; assessments of, and development of, control technologies for energy systems; and integrated assessments of a wide range of energy-related environmental issues.

EPA REVIEW NOTICE

EPA-600/7-82-037b
May 1982

VERIFICATION AND TRANSFER OF THERMAL POLLUTION MODEL

VOLUME II: USER'S MANUAL FOR THREE-DIMENSIONAL FREE-SURFACE MODEL

By

Samuel S. Lee, Subrata Sengupta,
S. Y. Tuann and C. R. Lee
Department of Mechanical Engineering
University of Miami
Coral Gables, Florida 33124

NASA Contract No. NAS 10-9410

NASA Project Manager: Roy A. Bland

National Aeronautics and Space Administration
Kennedy Space Center
Kennedy Space Center, Florida 32899

EPA Interagency Agreement No. 78-DX-0166

EPA Project Officer: Theodore G. Brna

Industrial Environmental Research Laboratory
Office of Environmental Engineering and Technology
Research Triangle Park, North Carolina 27711

Prepared for:

U. S. Environmental Protection Agency
Office of Research and Development
Washington, D. C. 20460

PREFACE

This volume presents the description and program documentation of the three-dimensional, free-surface mathematical model for thermal pollution analysis and prediction for shallow water bodies, for example, lakes and coastal waters. The program was developed by the Thermal Pollution Group at the University of Miami, and was successfully verified through application to several sites. This success was made possible by funding and technical assistance provided by the National Aeronautics and Space Administration (NASA) and the Environmental Protection Agency (EPA).

The model is time dependent, and the leap-frog and DuFort-Frankel schemes are adopted for solving the predictive equations based on the conservation principles of mass, momentum and energy. The model has been developed with minimal physical and site restrictive assumptions, and its algorithm has sufficient generality to allow for different boundary conditions specified at open boundaries. The program shows both the temporal and spatial variations of the surface water height. It computes three-dimensional velocity and temperature fields. The model can serve as an effective means for hydrothermal analysis and prediction. Plotting programs employed for representing the numerous results are also included.

The volume is intended as a user's manual and, as such, presents specific instructions regarding data preparation for program execution. To illustrate further, an example case is included here with its input data, hard copy printout and plots. The complete listing of the program and its accessories is also included.

ABSTRACT

A mathematical model that can be used for the analysis of thermal discharge from power plants into tidal estuaries and coastal waters is described. This transient, free-surface, three-dimensional model can be applied to predict the water temperature as a function of time and position in a specified region.

In situations of practical relevance, the specified coastal or offshore region will be a water body of irregular bottom topography with possible islands or keys. The user specified the boundary and boundary conditions, as well as the water depth distribution. Semi-diurnal tide is considered in the model. Hourly weather data is needed for wind stress calculation and heat exchange between water and the atmosphere. The ambient temperature is assumed of a sinusoidal form of 24-hour period. The ambient turbulence is included by an eddy viscosity and diffusivity formulation. The appropriate values are to be calibrated against measured currents.

CONTENTS

FIGURES

TABLES

SYMBOLS

A_v	Vertical eddy viscosity, $cm^2\ sec^{-1}$
B_h	Horizontal eddy diffusivity, $cm^2\ sec^{-1}$
B_v	Vertical eddy diffusivity, $cm^2\ sec^{-1}$
f	Coriolis factor, sec^{-1}
	Relative humidity in fraction of unit
g	Acceleration of gravity, $cm\ sec^{-2}$
h	Local water depth with respect to mean sea level, cm
H	Total water depth, cm
H_s	Gross solar radiation, BTU $ft^{-2}\ day^{-1}$
I	Node index in the direction of the x-axis
J	Node index in the direction of the y-axis
K	Node index in the direction of the z-axis
K_s	Surface heat exchange coefficient, BTU $ft^{-2}\ day^{-1}$ deg F^{-1}
L	Reference length, cm
P	Pressure, dynes cm^{-2}
t	Time, sec
T	Water temperature, deg C
T_a	Air temperature, deg F
T_{ave}	Average of air and dewpoint temperatures, deg F
T_d	Dewpoint temperature, deg F
T_e	Equilibrium temperature, deg F
T_s	Ambient surface temperature, deg F
u	Component of water velocity along x-axis, cm sec^{-1}
	Wind speed, mph
v	Component of water velocity along the y-axis, cm sec^{-1}
w	Component of water velocity along the z-axis, cm sec^{-1}
η	Displacement of the free surface with respect to the mean water level, cm
ρ	Water density, gm cm^{-3}
Ω	Nondimensional vertical fluid velocity
σ	Nondimensional vertical coordinate

ACKNOWLEDGMENTS

This work was supported by a contract from the National Aeronautics and Space Administration (NASA-KSC) and the Environmental Protection Agency (EPA-RTP).

The authors express their sincere gratitude for the technical and managerial support of Mr. Roy A. Bland, the NASA-KSC project manager of this contract, and the NASA-KSC remote sensing group. Special thanks are also due to Dr. Theodore G. Brna, the EPA-RTP project manager, for his guidance and support of the experiments, and to Mr. Albert W. Morneault from Florida Power Company (FPC), Tarpon Springs, and his data collection group for data acquisition. The support of Mr. Charles H. Kaplan of EPA was extremely helpful in the planning and reviewing of this project.

SECTION 1

INTRODUCTION

The analysis of thermal discharges is important in order to minimize the environmental impact and to manage efficiently and safely the waste heat problems. The study of technological solutions to the problems of heated water disposal involves complicated relationships, such as the location, geometry and types of the discharge outlet, the flow condition and temperature of the receiving water body, the meteorological conditions of the site, the waste heat output of the power plant, etc. A thorough understanding of the thermal effects and physical processes of heated water dispersion to the environment is an essential part of serving the rapidly growing demand for electrical energy, while reducing the possible impact on the receiving ecosystem.

The thermal effluent from a power plant will have variable consequences on the aquatic life of a receiving water body and the adjacent environment depending on the temperature rise. Therefore, the prime objective of the heated water discharge system is to bring the discharged water into thermal equilibrium with the surrounding water by bringing thermal outfall to the mainstream of the water body, whereby the mixing and convective processes will increase the surface heat transfer to the atmosphere. Thus, the temperature rise within the tolerance of natural environmental conditions is very important on the disposal system design and the standards for regulating thermal effluent.

Under limited circumstances, in-situ measurements can serve for diagnostic and monitoring purposes for meeting the need of analyzing thermal impact on receiving water body. However, to provide a priori information about the nature and extent of thermal impact for site selection and discharge system design, numerical modeling for simulating hydrothermal behavior of the water body is imperative.

During the past years, many numerical models have been developed for hydrothermal studies. Dunn et al. (1975) gave a presentation of different models developed up to then. They applied those models to the Point Beach Nuclear Power Plant and compared the performance of various models in predicting a standard data base. A general conclusion that can be made from their analysis is that though some models may perform well under certain conditions, a generalized model which accounts for wind, current, tide, bottom topography and diverse meteorological conditions is yet to be developed.

Since 1974 the Thermal Pollution Group at the University of Miami has endeavored to develop a mathematical package for hydrothermal studies. The primary motivation behind the effort was to develop a series of models which make minimal site restrictive assumptions enabling application to diverse basin and discharge configurations. Two separate formulations were made, one with the rigid-lid approximation and the other with the free-surface included. The details of the package and formulation are presented in a number of reports by Lee et al. (1978).

The present report concerns the UM's free-surface model and its application to the Anclote Anchorage in Florida for waste heat discharge from a power plant. The features of the model are: a) three-dimensional, b) nonlinear, c) time-dependent, d) irregular topography, e) driving forces including wind, tide, heat and mass flux, f) graphical representation of results of velocity and temperature fields, g) prediction of temporal and spatial variation of water surface.

The descriptions of the main program and its subroutines, main algorithm and flow chart, program symbols, input data and logic parameters, as well as the description of associated plotting programs, are contained herein for the ready access of the computer program package to the user. A preliminary review on existing three-dimensional free-surface models, basic concepts of the present model, assumptions, approximations, governing equations, initial and boundary conditions, finite difference implementation and numerical solution methods is presented in Lee et al. (1979) and Carter (1977).

The report also contains a plotting program which is used to analyze the results of the main calculation. A subroutine to compare the calculated temperature field with that obtained by IR scanning is presented in the program. Note that the IR temperature is interpreted by hand from the mosaic film and then read in for isotherm plotting and comparison.

The model has been tested for its adaptability. That is, the model allows for program modifications so that different initial and boundary conditions could be considered. The main program has several flag statements which make different usage of same program possible. Any program modification for the purpose of model transition should be made with care, and the new program should be validated by sample runs to assure that the effect of the modification is as desired. The same is also applicable to the plotting program.

The program therefore contains two parts. Part 1 is primary and performs calculations; Part 2 is secondary and is for analyzing and plotting the results of Part 1. ANCMN is the driving program of Part 1. The input contains parameters, geometrical and initial data, or tape which stores intermediate results; output contains printout of results at preset time intervals, in both hard copy and tape form. The hard copy printout provides the base for analysis, upon which decisions and choice of the plots needed for further analysis and detailed comparison with measured results can be made. PLOTMN is the driving program of Part 2. The tape output from Part 1 is the main input. In addition, control cards assigning choice of plot, plot size, simulation hour and measured result for comparison are also required. Output is in printout and plot tape. The latter is used for plotting by CALCOMP plotter.

SECTION 2

RECOMMENDATIONS

The model can be enlarged to handle any passive constituent, dissolved or suspended, possessing arbitrary decay characteristics. The formulation of the constituent transport is based on the convection-diffusion equation, which is analogous to the thermal transport equation in the present model. The enlarged model would then be an ideal tool to study the ecological response of aquatic biota to the thermal effluents of the power plant.

The model can be modified to include bouyancy effects caused by fresh water/salt water sensing by including a salinity-dispersion equation. This equation will be of similar form to the energy equation.

The code is written for a constant grid size. Modifications can be made to incorporate a coarse grid for the complete field in comparison with a fine grid near the discharge location. This will allow a more accurate prediction of plume behavior in the near field. A penalty in computational cost will be incurred.

SECTION 3

PROGRAM DESCRIPTION AND FLOW CHART FOR MAIN PROGRAM (ANCMN)

DESCRIPTION OF PROGRAM ALGORITHM

The governing partial differential equations are given in Table 1; the symbols and definitions are referenced in Nomenclature. The problem is set up as an initial-boundary value problem, so values of dependent variables are assumed known initially and prescribed on boundaries. Values at successive time steps are obtained by using a true explicit scheme. The leap-frog finite difference format is used to calculate surface elevation η and two horizontal velocity components, u and v, at time n+1, where n is the present time step. The variables at times n-1 and n are all known. The sequence in which calculations are performed is as follows:

1. Integrate the surface elevation equation using central-time central-space (CTCS) differencing. That is, η^{n-1}, η^{n} and u^{n}, v^{n} and h. Note that h is independent of time while the present water depth $H^{n} = h+\eta^{n}$ is needed in this calculation. In the subroutine BETA, not only η^{n+1} but also Ω^{n+1} (i, j, k), the modified vertical velocity in transform (σ) plane, is accomplished immediately after the η computation.

2. The next task in the sequence is to calculate the nonlinear inertia terms that appear in the horizontal momentum equations. Here, two subroutines are involved: BNRTIA is

TABLE 1. Governing Equations

Continuity Equation*:

$$\frac{\partial Hu}{\partial x} + \frac{\partial Hv}{\partial y} + H\frac{\partial \Omega}{\partial \sigma} + \frac{\partial \eta}{\partial t} = 0$$

u Momentum Equation:

$$\frac{\partial Hu}{\partial t} + \frac{\partial Huu}{\partial x} + \frac{\partial Hvu}{\partial y} + H\frac{\partial \Omega u}{\partial \sigma} + (1+\sigma)\frac{\partial u}{\partial \sigma}\frac{\partial \eta}{\partial t} - fvH$$

$$= -\frac{H}{\rho}\frac{\partial p}{\partial x} - gH\left(\sigma\frac{\partial H}{\partial x} + \frac{\partial \eta}{\partial x}\right) + \frac{Av}{H}\frac{\partial^2 u}{\partial \sigma^2}$$

v Momentum Equation:

$$\frac{\partial Hv}{\partial t} + \frac{\partial Huv}{\partial x} + \frac{\partial Hvv}{\partial y} + H\frac{\partial \Omega v}{\partial \sigma} + (1+\sigma)\frac{\partial v}{\partial \sigma}\frac{\partial \eta}{\partial t} + fuH$$

$$= -\frac{H}{\rho}\frac{\partial p}{\partial y} - gH\left(\sigma\frac{\partial H}{\partial y} + \frac{\partial \eta}{\partial y}\right) + \frac{Av}{H}\frac{\partial^2 v}{\partial \sigma^2}$$

Energy Equation:

$$\frac{\partial HT}{\partial t} + \frac{\partial HuT}{\partial x} + \frac{\partial HvT}{\partial y} + H\frac{\partial \Omega v}{\partial \sigma} + (1+\sigma)\frac{\partial T}{\partial \sigma}\frac{\partial \eta}{\partial t}$$

$$= \frac{\ddot{B}_v}{H}\frac{\partial^2 T}{\partial \sigma^2} + B_h\left[\frac{\partial}{\partial x}\left(H\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(H\frac{\partial T}{\partial y}\right)\right]$$

---

* This equation is vertically integrated to yield a) prognostic equation for y, and b) synoptic equation for Ω; they are

$$\frac{\partial \eta}{\partial t} = -\int_{-1}^{0}\left(\frac{\partial Hu}{\partial x} + \frac{\partial Hv}{\partial y}\right)d\sigma$$

$$\Omega = \frac{\sigma}{H}\frac{\partial \eta}{\partial t} + \frac{1}{H}\int_{\sigma}^{0}\left(\frac{\partial Hu}{\partial x} + \frac{\partial Hv}{\partial y}\right)d\sigma.$$

The latter, upon transformation, yields the actual vertical velocity

$$w = \Omega H + (1+\sigma)\frac{\partial \eta}{\partial t} + u\frac{\partial H}{\partial x} + v\frac{\partial H}{\partial y} + u\frac{\partial \eta}{\partial x} + v\frac{\partial \eta}{\partial y}$$

for interior points, while ABNR3 is for open boundary points. Note, for the Anclote Anchorage sample problem, the open boundaries are at j = 1 and j = 14, and the imposing tides are applied at points immediately outside these open boundaries. Therefore, program modification is needed if different open boundary conditions are employed.

3. Following the inertia terms computations, which may be skipped if the Rossby Number is very close to zero, new values of u and v at n+1 are computed for all points in the grid. Again, the leap-frog and central-space scheme is used, but DuFort-Frankel differencing is applied to the vertical momentum diffusion terms. Two subroutines are called here, BVEL for interior points and ASAF3 for open boundary points. Since velocity at all points is calculated without distinction, a subroutine GIVENU is needed to specify the given discharge or flowrate at particular points, that is, to replace

calculated velocities at those points with known values.

Steps 1, 2 and 3 are calculations for surface elevation, modified vertical velocity and horizontal velocity components; they constitute the V-calculation. Whether the V-calculation is to be carried out or not depends on flagged statement KVEL = 1 or 0. The next group of calculations is for thermal transport, or T-calculation, and it involves the energy equation only. Similarly, this group is to be flagged by statement whether KTEMP is unity or zero.

4. The convective term in the energy equation is calculated next, using either a given velocity field, in the case of KVEL = 0, or the presently known velocity field at n+1, in the case of KVEL = 1. The subroutine for this purpose is CONV.

5. The energy equation itself is then integrated over time to obtain T at n+1. The forward-time, central-space (FTCS) and DuFort-Frankel differencing for the vertical diffusion term are used in the calculation. The subroutine involved is TCOMPT. Since temperatures at all points are computed without distinction as to whether the points are with given temperatures or not, subroutine GIVENT is needed to respecify the temperature at the given points.

Clearly, Steps 4 and 5 make up the T-calculation. In either of V- or T-calculations, vertical velocity w is involved. Instead of w, the rate of change of surface elevation, $\frac{\partial \eta}{\partial t}$, is used for convection in vertical direction.

6. The actual vertical velocity, w, is computed when it is needed for printout. The subroutine is WCAL, and instead of u^{n+1} and v^{n+1} which are defined at half J and half I respectively, the interpreted velocities at center of grid cell are used in this calculation. Since a space-staggered scheme is used, the water level and vertical velocity are described at the center of grid cell, while the horizontal velocities are described at the edges of cells.

7. The real time (or simulation time) is checked and Steps 1 through 6 are repeated; that is, the above procedure is repeated for n+2 using values at n+1 and n.

Reference to the flow chart presented in Figure 1 will clarify the description of program algorithm.

FLOW CHART

Figure 1 shows the main flow chart of the three-dimensional, free-surface program applied to the Anclote Anchorage. In the flow chart, the subroutines and their functions are described briefly. Table 2 lists the subroutines called in the main-program, ANCMN.

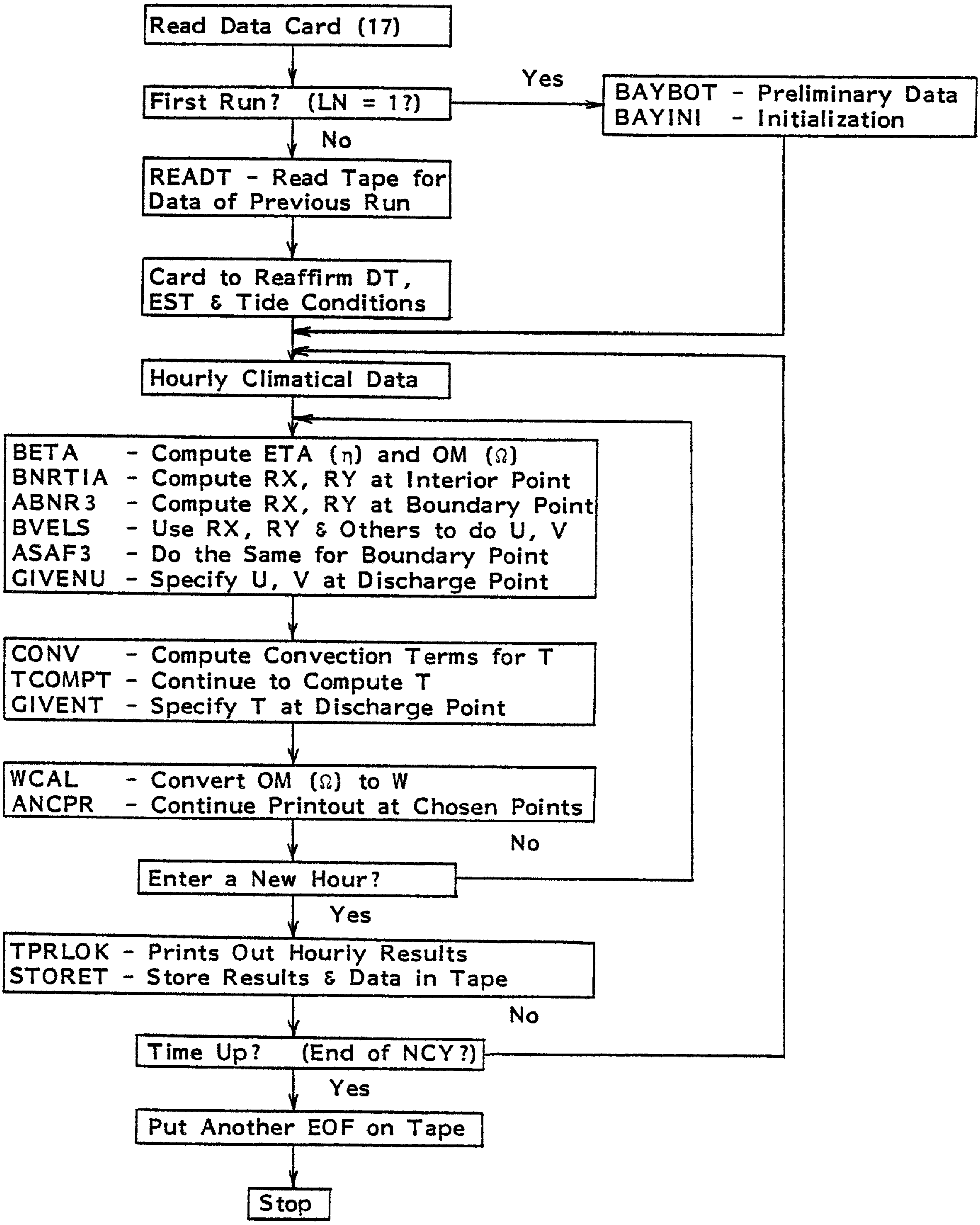


Figure 1. Flow chart for calculating program ANCMN and its subroutines

TABLE 2. Subroutines Required in Main Calculating Program ANCMN

No.	Name	Description	Remark
1	BAYBOT	Reads in bay bottom sounding (H) and various grid book-keeping matrices, calculates depth matrices accordingly.	Skip if LN=1, depends on problem and grid work, data file AMATN preferred.
2	BAYINI	Initializes the dependent variables such as U, V, W, T, ETA, RX, RY also set 10^9 to values outside the calculation region.	Also depends on probelm and run conditions. RX RY are inertia terms.
3	READT	Reads in stored data from tape for continuing run.	Skip if LN=1.
4	IRREAD	If required, it reads in IR data as initial temperature; thus it replaces T input from READT.	No need if T-calculation began with ambient temperature.
5	EQTEMP	Calculates the equilibrium temperature over that hour.	Number of hour is NCY.
6	BETA	Calculates surface elevation ETA and vertical velocity OM (Ω).	Mandatory in V-calculation.
7	BNRTIA	Calculates inertia terms of momentum eq. for boundary pts.	Skip if ROSSBY = 0.
8	ABNR3	Calculates inertia terms of momentum eq. for boundary pts. on north and south exit of the anchorage.	Skip if ROSSBY = 0.
9	BVELS	Calculates velocity at interior pts. - main part of velocity calculation.	
10	ASAF3	Calculates velocity at boundary pts. along north and south boundaries.	
11	GIVENU	Specifies velocity at control pts., such as discharge, intake and river head.	

TABLE 2. Subroutines Required in Main Calculating Program ANCMN (Continued)

No.	Name	Description	Remark
12	CONV	Calculates convective terms of energy equilibrium.	
13	TCOMPT	Calculates temperature.	
14	GIVENT	Specifies temperature at discharge outlet.	Depends on run condition.
15	WCAL	Converts the OM (Ω) vertical velocity to physical vertical velocity W.	
16	ANCPR	Prints surface elevation, velocity (mag. with dir.) and temperature at 4 chosen locations after completion of marching of DT, i.e. printout after each time step.	Locations are to be chosen by user.
17	TPRLOK	Prints out the velocity field, water depth, surface elevation and temperature field after one hour of simulation time.	
18	STORET	After TPRLOK for printout, the same data and relevant parameters are stored onto tape, either for later plotting or for continued calculation.	Skip if KSTORE = 0.
19	ZZ1	A subroutine called by ANCPR and TPRLOK; it is for finding velocity dir. based on U and V components.	
20	AMATN	Is a data file containing all grid matrices and bottom depth matrix to be read in BAYBOT.	Same data could be in card form, inserted after TINIT card.
21	C2007	Climatical data used in the run; the data is on hourly basis.	Could also be in card form.

SUBROUTINE DESCRIPTIONS

This section describes the subroutines used in ANCMN, in order of their appearance.

BAYBOT

It reads the marker matrix MAR; the elevation matrix ELEV which is changed to depth matrix H by adding a constant STAGE; then four more marker matrices, MEX, MEY, MX and MY. It interpolates H according to marker MEX, MX and MY to find additional depth matrices, called HB, HU and HV, respectively.

BAYINI

It initializes most of the variable matrices. Since the initial condition for velocity is a quiescent condition, U1, U2, U3, V1, V2, V3, RX, RY, ETA1, ETA, ETA3, UB, VB, OM, W and TC are set to zero. The quiet bay is assumed to have a constant temperature TINIT to begin with, so TI, T2, T3 and TB are set to TINIT.

READT

This subroutine reads in input parameters, physical quantities and intermediate results stored on tape.

IRREAD

It reads in in-situ measured or IR scanned temperature as an alternative initial condition to temperature calculation. This temperature is interpolated by hand and stored in the form of matrix T1.

EQTEMP

It calculates the equilibrium temperature T_e and surface heat exchange coefficient K_s of a natural water surface. The procedures for these calculations are as follows:

1. $T_d = T_a - (14.55 + 0.114\, T_a)(1-f) - [(2.5 + 0.007\, T_a)(1-f)]^3$

   where T_d = dewpoint temperature in °F
   T_a = air temperature in °F
   f = relative humidity in fraction of unit

2. $\beta = 0.255 - 0.0085\, T_{ave} + 0.000204\, T_{ave}^2$

   where $T_{ave} = \frac{1}{2}(T_s + T_d)$, and β is an intermediate step
   T_s = ambient surface temperature in °F

3. $f(u) = 70 + 0.7\, u^2$, and u is wind speed in mph

4. $K_s = 15.7 + (\beta + 0.26)\ f(u)$

where K_s = surface heat exchange coefficient in BTU/(ft² day °F)

5. $T_e = T_d + \frac{H_s}{K_s}$

where T_e = equilibrium temperature in °F
H_s = gross solar radiation in BTU/(ft² day)

Note: The T_a, f, u, H_s and T_s are climatological data; however, care must be taken for the hourly data TAIR, HUMID, WIND, SRAD and TSURF are in metric units. Therefore, in the above calculations, the basic data must first be transformed into English units, then the final results, T_e and K_s, must again be transformed back to metric units. T_e (TEQ) and K_s (SK) are used in subroutine CONV for T-calculation.

BETA

Computes η^{n+1} (ETA3) and Ω^{n+1} (OM) by using central differencing from the continuity equation. The vertical integration is done using Simpson's rule. The following symbols are used:

DR = depth at half-integer j point on right edge, i + 1, of the cell.

DL = depth at half-integer j point on left edge, i, of the cell.

D2 = depth at half-integer i point on upper edge, j + 1, of the cell.

D1 = depth at half-integer i point on lower edge, j, of the cell.

DHUX = $\frac{\partial Hu}{\partial x}$

DHVY = $\frac{\partial Hv}{\partial y}$

AH = total depth at the center of the cell, a half-grid point.

BNRTIA

It computes the sums of the nonlinear inertia terms, RX and RY, in the x and y momentum equation at each interior point of the domain. Note that RX (i, j, k) = RY (i, j, k) = 0 for k = KN. The following symbols are used, in their order of appearance:

AH = depth either at u-point or at v-point

DET = $\frac{\partial \eta}{\partial t}$

DHUUX = $\frac{\partial Huu}{\partial x}$ DHUVX = $\frac{\partial Huv}{\partial x}$

$DHUVY = \frac{\partial Huv}{\partial y}$ $DHVVY = \frac{\partial Hvv}{\partial y}$

D2 = depth at forward half-grid point in either x or y direction.

D1 = depth at backward half-grid point in either x or y direction.

UBAR2 = average u at forward half-grid point.

UBAR1 = average u at backward half-grid point.

VBAR2 = average v at forward half-grid point.

VBAR1 = average v at backward half-grid point.

E2 = averaged η at (i, j + 1) or (i + 1, j)

E1 = averaged η at (i, j)

D2 = depth at (i, j + 1) or (i + 1, j)

D1 = depth at (i, j)

$DUOMS = \frac{\partial u\Omega}{\partial \sigma}$ $DVOMS = \frac{\partial v\Omega}{\partial \sigma}$

$DUS = \frac{\partial u}{\partial \sigma}$ $DVS = \frac{\partial v}{\partial \sigma}$

$$RX = \frac{\partial Huu}{\partial x} + \frac{\partial Huv}{\partial y} + H \frac{\partial u\Omega}{\partial \sigma} + (1 + \sigma) \frac{\partial u}{\partial \sigma}\frac{\partial \eta}{\partial t}$$

$$RY = \frac{\partial Huv}{\partial x} + \frac{\partial Hvv}{\partial y} + H \frac{\partial v\Omega}{\partial \sigma} + (1 + \sigma) \frac{\partial v}{\partial \sigma}\frac{\partial \eta}{\partial t}$$

ABNR3

It computes RX and RY for points on boundary. The tide heights at half-grid points outside the north and south boundary are computed first. Since the open boundary is in x-direction, that is, only the v-point appears, RX = 0 and RY is given by

$$RY = \frac{\partial Huv}{\partial x} + \frac{\partial Hvv}{\partial y} + H \frac{\partial u\Omega}{\partial \sigma} + (1 + \sigma) \frac{\partial v}{\partial \sigma}\frac{\partial \eta}{\partial t}$$

In the computation, it is assumed:

D2 = depth at half-grid point just outside the boundary
= HV (i, JN) + North Tide Height, if D2 is HB (i, JN)
= HV (i, 1) + South Tide Height, if D2 is HB (i, 0)

VBAR2 = v at half-grid point just outside the boundary
= V2 (i, JN, K) if it is north boundary
= V2 (i, 1, k) if it is south boundary

BVELS

It calculates u^{n+1} (U3) and v^{n+1} (V3) at interior points by central time differencing and DuFort-Frankel scheme. Note that the horizontal diffusion of momentum is neglected in the model.

The equations for u and v are

$$\frac{1}{H}\frac{\partial Hu}{\partial t} = fv - g\frac{\partial \eta}{\partial x} + \frac{A_v}{H^2}\frac{\partial^2 u}{\partial \sigma^2} - \frac{1}{H}RX$$

$$\frac{1}{H}\frac{\partial Hv}{\partial t} = -fu - g\frac{\partial \eta}{\partial y} + \frac{A_v}{H^2}\frac{\partial^2 v}{\partial \sigma^2} - \frac{1}{H}RY$$

where RX and RY are the nonlinear inertia terms.

The following symbols are introduced for briefness.

A2 = Coriolis term

A4 = $g\frac{\partial \eta}{\partial x}$ or $g\frac{\partial \eta}{\partial y}$

A5 = $\frac{1}{H}$ RX or $\frac{1}{H}$ RY

A6 = the rest of vertical diffusion term.

ASAF3

This calculates v^{n+1} (V3) at the v-points on the south (j = 1) and north (j = JN) boundaries. It is similar to ABNR3. The tide heights at imaginary half-grid points just outside the south and north boundary are computed first. The term $\frac{\partial \eta}{\partial y}$ is calculated at v-point at both j = 1 and JN, with tide height at outside half-grid point and η at inside half-grid point. Since the u velocity on these boundaries are assumed zero, the Coriolis force term B2 is set to zero. Symbols B4, B5 and B6 stand for pressure, convection and diffusion terms respectively.

GIVENU

It specifies velocities at cooling system outlet and intake. These velocities are determined from power plant flowrate. The river flowrate is also simulated by imposing velocities at river entry point.

CONV

It computes the sum of the convective terms in the energy equation at each point. Note that T is designated at half-grid points. The following symbols are used in this subroutine:

AH = depth at point (i, j) where TC (i, j, k) is to be calculated.

DR = depth at forward u-point

DL = depth at backward u-point

D2 = depth at forward v-point

D1 = depth at backward v-point

DTZ (i, j) = temperature slope $\frac{\partial T}{\partial \sigma}$ at the surface

UR = forward u but at T-level

UL = backward u but at T-level

DHUTX = $\frac{\partial HuT}{\partial x}$

VR = forward v but at T-level

VL = backward v but at T-level

DHVTY = $\frac{\partial HvT}{\partial y}$

The convection term TC is written for

$$TC = \frac{\partial HuT}{\partial x} + \frac{\partial HvT}{\partial y} + H \frac{\partial \Omega T}{\partial \sigma} + (1 + \sigma) \frac{\partial T}{\partial \sigma} \frac{\partial \eta}{\partial t}$$

TCOMPT

This subroutine computes temperature T^{n+1} (T3) for each point in the domain. The equation is

$$\frac{1}{H} \frac{\partial HT}{\partial t} = \frac{B_v}{H^2} \frac{\partial^2 T}{\partial \sigma^2} + B_h\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) - \frac{1}{H} TC$$

where TC is the convection term obtained from subroutine CONV. The boundary condition on solid boundary is adiabatic, but on open boundary, T is assumed known. There are three different formulas for computing the $\frac{\partial^2 T}{\partial \sigma^2}$ although the same DuFort-Frankel format is used throughout. This is because the surface temperature slope is DTZ (i, j) and the bottom is adiabatic. The symbols here are

$$D2TX = \frac{\partial^2 T}{\partial x^2} \qquad D2TY = \frac{\partial^2 T}{\partial y^2}$$

GIVENT

It specifies the temperature of cooling system discharge water and river delivery.

WCAL

This subroutine calculates vertical velocity w (i, j, k) from modified vertical velocity Ω (i, j, k) by using σ-transformation formula. The following symbols are used:

$$DEX = \frac{\partial \eta}{\partial x} \qquad DEY = \frac{\partial \eta}{\partial y}$$

$$DAHX = \frac{\partial H}{\partial x} \qquad DAHY = \frac{\partial H}{\partial y}$$

ANCPR

It prints continuous records of elevation, velocity and temperature at certain particular points.

All data are printed on a single line, with the first item on the line being the total simulation time TTOT in sec. Items for each point are: surface elevation, resultant velocity, direction in which the velocity vector points (deg positive clockwise from North), and temperature, in that order.

TPRLOK

This subroutine performs the major printing tasks. The following variables are printed hourly, or controlled by printout interval TPRT:

AVR = resultant velocity, cm/sec

ANG = direction in which the velocity points

W = vertical velocity, cm/sec

ETA = surface elevation, cm

T2 = temperature, deg C

After TPRLOK is executed, the main program ANCMN prints out the total depth, $H = h + \eta$, at each point. Thus, the hourly printout of the relevant variables is completed. In this subroutine, there are two flags, KUV and KPROF. If both are zero, it skips the printing of velocity components u and v either presented in layers (k = 1, KZ) or in cross section along x-direction (j = 1, JM). These two flags are flipped only by interchanging the statements in the subroutine.

STORET

It records all the relevant data and results of the preceding simulation hour. STORET puts one EOF on tape after each block, while the main program ANCMN puts another EOF after the last block of data is recorded.

ZZ1

This subroutine finds the direction of the resultant horizontal velocity.

SECTION 4

LIST OF PROGRAM SYMBOLS OF MAIN PROGRAM

This section presents the program symbols and their definition in alphabetical order. In many cases, the symbols are described with the aid of diagram to show the definition.

DESCRIPTION OF MAIN VARIABLES

The relative position and designation of variables are shown in Figure 2. The water depth, h, is described in integer values of i and j; the u-component is described at half-integer value of j and integer values of i and k; the v-component at half-integer value of i and integer values of j and k; the w-component at integer value of k and half-integer values of i and j; the surface elevation, η, is described at half-integer values of i and j; and the temperature, T, is described at half-integer values of i, j and k. The modified vertical velocity, Ω, is described at the same place as w-component. Figure 3 shows the space-staggered grid system in horizontal projection.

Table 3 lists all the symbols used for dependent variables appearing in the program. Since three levels of time step are used, same variable at different level is assigned with different symbols. The rule for symbolizing the dependent variables is: for variable $F(i\Delta x, j\Delta y, k\Delta \sigma, n\Delta t)$, F1(i, j, k) is used to denote the value of variable at n-1; F2(i, j, k) is the present value; while F3(i, j, k) is the value at n+1 thus to be computed; and FB(i, j, k) is the interpreted value of F2(i, j, k) at a set of grid points differing from where it is designated.

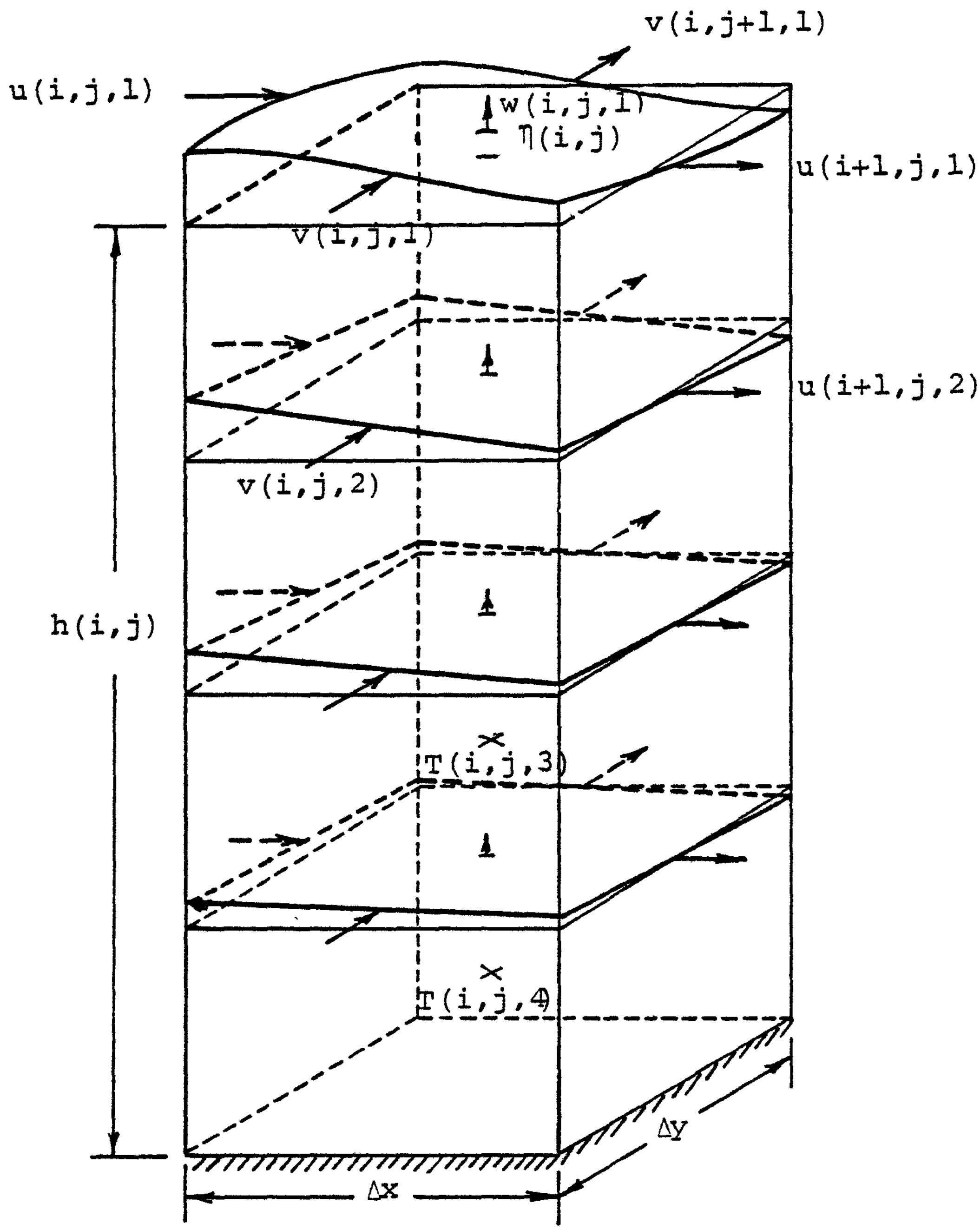


Figure 2. Relative position and designation of the variables

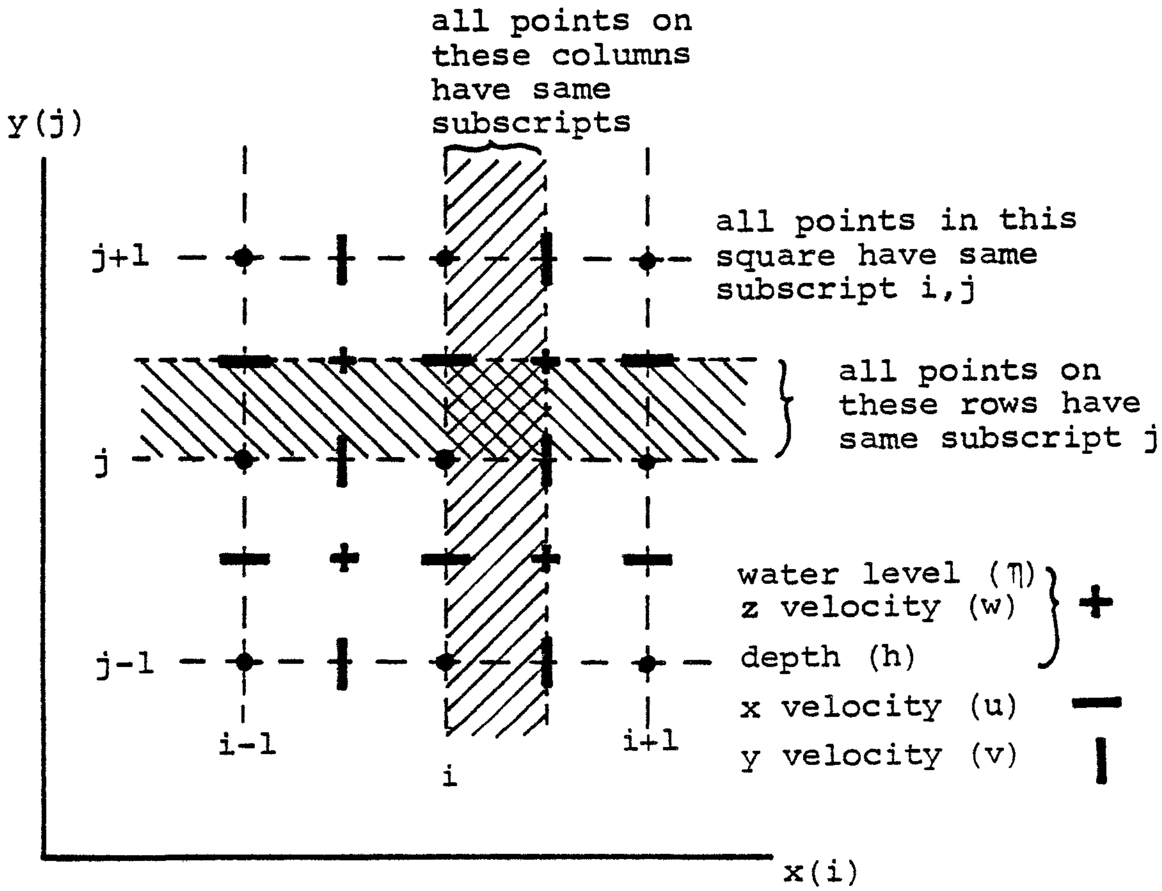


Figure 3. Space-staggered grid system - plan

TABLE 3. Symbols used in the program

Symbols	Argument	Description
U1, U2, U3, UB	(i, j, k)	u-component
V1, V1, V3, VB	(i, j, k)	v-component
T1, T2, T3, TB	(i, j, k)	Temperature T
ETA1, ETA2, ETA3	(i, j)	Surface Elevation η
TIDE1N, TIDE2N, TIDE3N	(i)	Tide Outside North Boundary
TIDE1S, TIDE2S, TIDE3S	(i)	Tide Outside South Boundary
DTZ	(i, j)	Heat Exchange at Water Surface
RX; RY; TC	(i, j, k)	Convection Terms in u; v; T equation
W; OM	(i, j, k)	w-component; Non-dimensional Ω
WAT	(i, j)	Total Water Depth or $h + \eta$

Note: In the program ANCMN, ETA2 is labeled as ETA and ETX; the former is used for calculation while the latter for printout.

MARKER MATRICES

The following integer-valued matrices are introduced to describe the grid system and to distinguish boundary from interior.

MAR (i, j)

MAR (i, j) identifies nodes in the full-grid system, i.e.

MAR = 0, (i, j) outside of boundary, hence no calculation

MAR = 1, (i, j) inside or on a boundary, as shown in Figure 4.

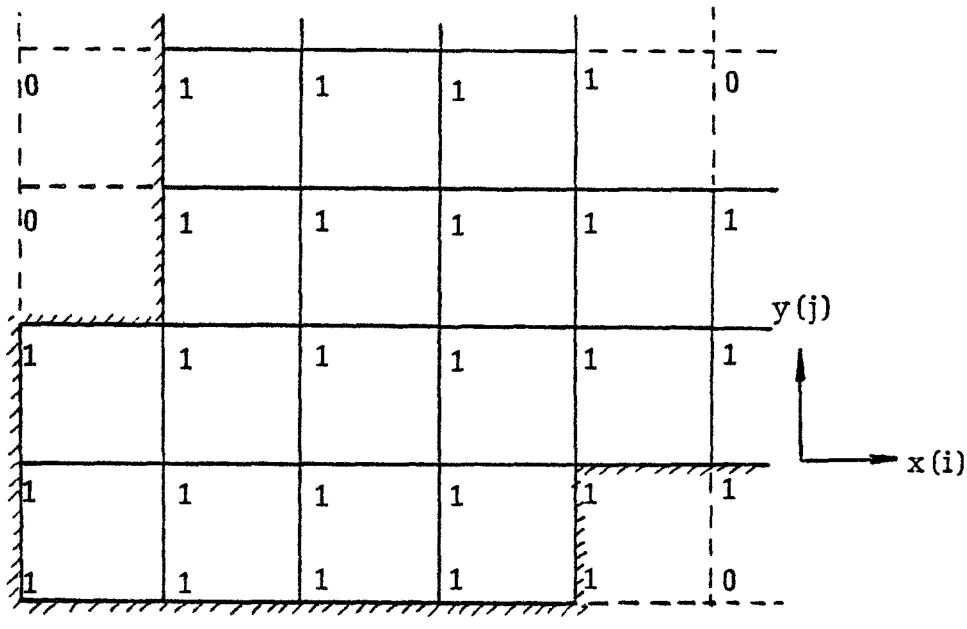


Figure 4. MAR (i, j) matrix

MEX (i, j)

MEX (i, j) provides marker to the half-grid system with reference to the y-direction boundaries, i.e.

MEX = 0, (i, j) outside of y-boundary, or exterior

MEX = 1, (i, j) just inside an east boundary

MEX = 2, (i, j) just inside a west boundary

MEX = 3, (i, j) nowhere near to y-boundary, or interior, as shown in Figure 5.

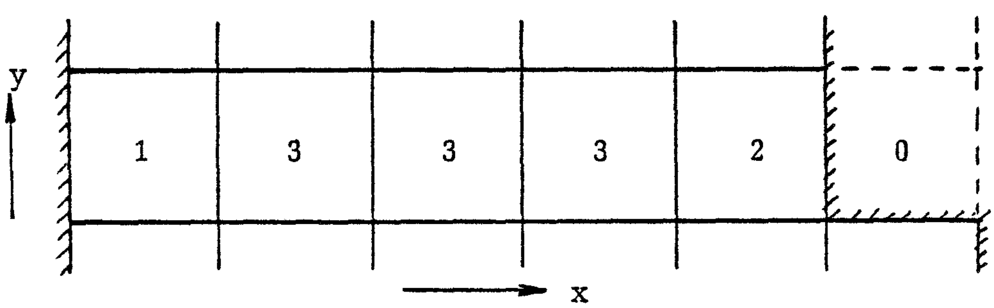


Figure 5. MEX (i, j) matrix

MEY (i, j)

MEY (i, j) provides marker to the half-grid system with reference to the x-direction boundaries, as shown in Figure 6.

MEY = 0, (i, j) outside of x-boundary, or exterior

MEY = 1, (i, j) just inside a south boundary

MEY = 2, (i, j) just inside a north boundary

MEY = 3, (k, j) nowhere near to x-boundary, or interior

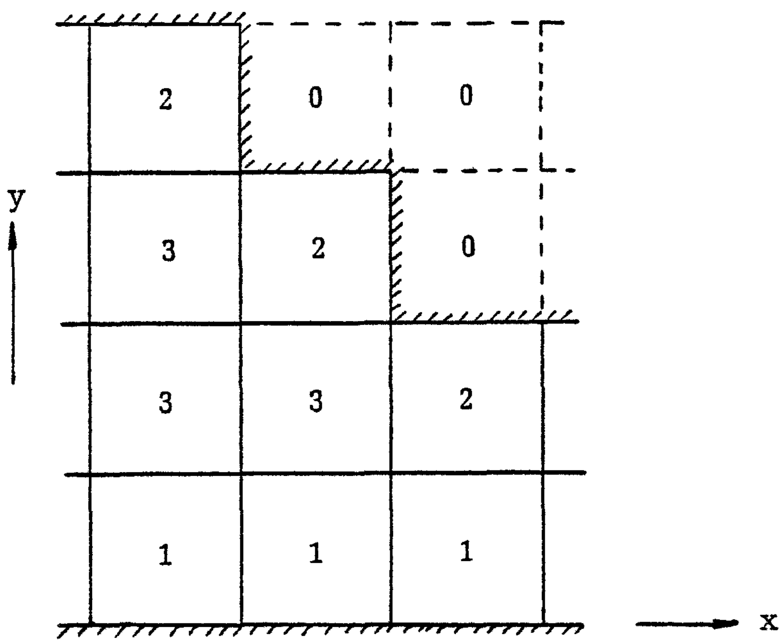


Figure 6. MEY (i, j) matrix

MX (i, j)

MX (i, j) provides marker to u-points, as shown in Figure 6.

MX = 0, (i, j) outside of y-boundary, or exterior

MX = 1, (i, j) on east boundary

MX = 2, (i, j) on west boundary

MX = 3, (i, j) nowhere near to y-boundary

MY (i, j)

MY (i, j) provides marker to v-points, as also shown in Figure 6.

MY = 0, (i, j) outside of x-boundary, or exterior

MY = 1, (i, j) on south boundary

MY = 2, (i, j) on north boundary

MY = 3, (i, j) nowhere near to x-boundary

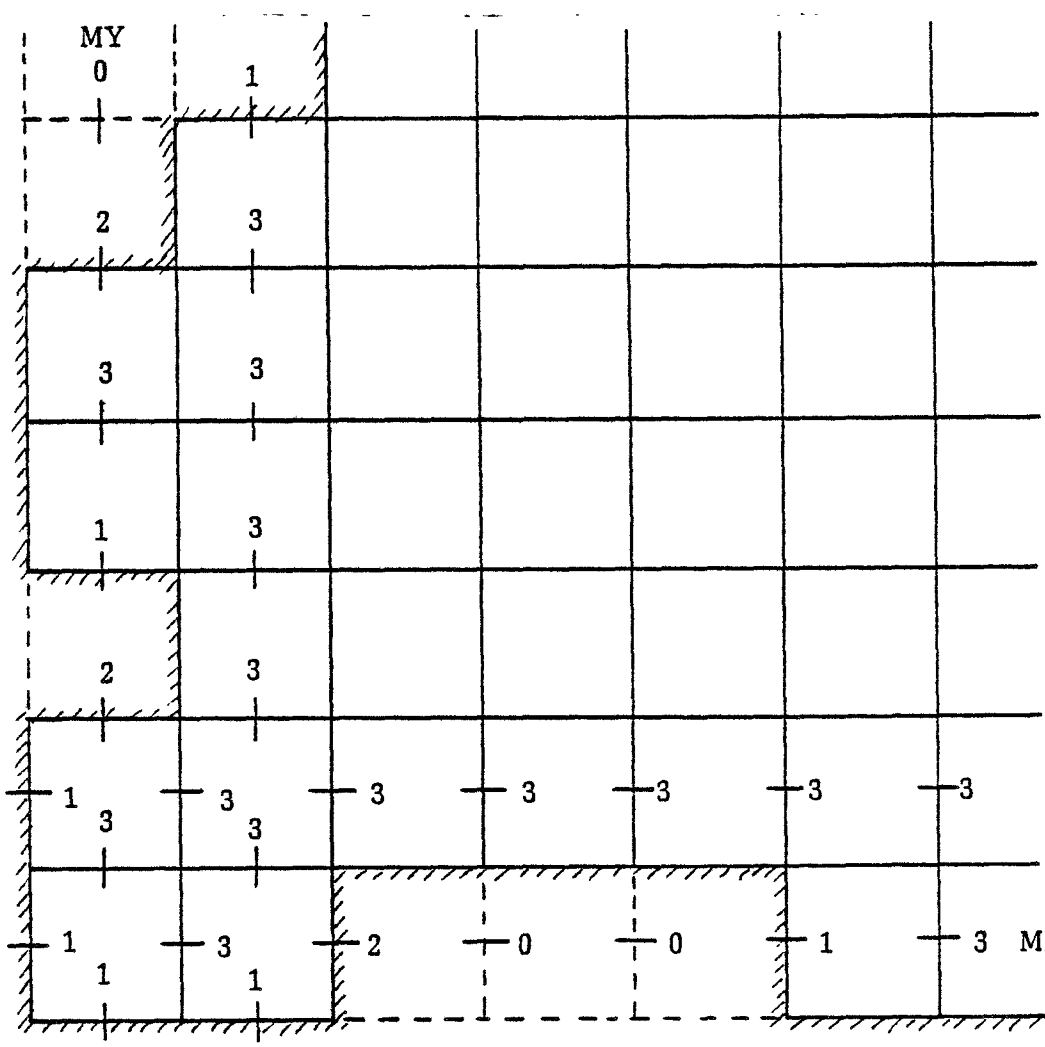


Figure 7. MX (i, j) and MY (i, j) matrices

DEPTH MATRIX AND ITS DERIVATIVE

The bathymetry of the area of interest is given by the matrix ELEV (i, j) designated at full-grid points. The values in feet are positive for lake or inland waters as elevation above MSL. However, for coastal water, the depths are read from the survey chart and are designated by positive values. For certain periods of the year, the water level may differ from MSL; a stage (STAGE1) is added to the ELEV (i, j) to obtain the actual depth matrix H (i, j). Note that the values of H (i, j) are in cm. To facilitate the calculation, the matrices HB (i, j), HU (i, j) and HV (i, j) are derived from H (i, j), and are for depths at half-grid points, u-points and v-points, respectively. Thus, the following real-valued matrices are introduced.

ELEV (i, j) elevations of the bottom with respect to MSL

HB (i, j) depth at half-grid points, in accordance with MEX/MEY

HU (i, j) depth at u-points, in accordance with MX

HV (i, j) depth at v-points, in accordance with MY

SIZE OF THE MATRICES, OR DIMENSIONS OF SUBSCRIPTED QUANTITIES

Let the grid work consist of IN x JN x KN nodes, i.e. IN nodes in x(i) direction, JN nodes in y(j) direction, and KN levels in σ(k) direction. Then there are IM = IN - 1 half-grid points in x-direction, JM = JN - 1 half-grid points in y-direction, and KZ = KN - 1 layers in σ- direction.

The values IN, JN, KN, IM and JM are the parameters to be specified at the beginning of the program and are determined by the grid used. Therefore, the dimensions of the matrices are given in terms of these parameters. Table 4 shows the size of the matrices already defined.

TABLE 4. Size of the Matrices

Symbol	Least Size	Given Size
UI, U2, U3	(IN, JM, KN)	(IN, JN, KN)
V1, V2, V3	(IM, JN, KN)	(IN, JN, KN)
T1, T2, T3, TC, TB	(IM, JM, KZ)	(IM, JM, KN)
UB, VB	(IM, JM, KZ)	(IM, JM, KN)
W, OM	(IM, JM, KN)	(IM, JM, KN)

TABLE 4. Size of the Matrices
(Continued)

Symbol	Least Size	Given Size
RX, RY	(IM, JM, KZ)	(IN, JN, KN)
ETA1, ETA, ETX, ETA3	(IM, JM)	(IM, JM)
WAT, DTZ	(IM, JM)	(IM, JM)
ELEV, H	(IN, JN)	(IN, JN)
HB	(IM, JM)	(IM, JM)
HU	(IN, JM)	(IN, JN)
HV	(IM, JN)	(IN, JN)
TIDE1N ... TIDE3S	(IM)	(IM)
AVR, ANG	(JM)	(JM)
MAR	(IN, JN)	(IN, JN)
MEX, MEY	(IM, JM)	(IM, JM)
MX	(IN, JM)	(IN, JN)
MY	(IM, JN)	(IN, JN)

AVR and ANG are used to facilitate printout of velocity and angle respectively.

OTHER SYMBOLS OCCURRING IN PROGRAM <u>ANCMN</u>

ALREF: Reference horizontal length L in cm.

AV: Vertical eddy viscosity, estimated by means of

$$A_v = 0.0018\ H^{4/3},\ H \text{ in cm},\ A_v \text{ in cm}^2\ \text{sec}^{-1}.$$

BH: Horizontal eddy diffusivity, estimated by means of

$$B_h = 0.0018\ L^{4/3},\ L \text{ in cm},\ B_h \text{ in cm}^2\ \text{sec}^{-1}.$$

BV: Vertical eddy diffusivity, estimated by the same formula as A_v, thus turbulent Prandtl No. is one.

DHR: Time increment in hour as simulation continues.

DTX: As a check on when to printout.

DS: Increment in σ-direction, in fraction of unit.

DT: Time step in second.

DX: Increment in x-direction, in cm.

DY: Increment in y-direction, in cm.

DUMS: 2DS.

DUMX: 2DX.

DUMY: 2DY.

EST: Eastern standard time in the day of simulation.

FCOR: Coriolis factor = $2W_e$ sin(latitude), sec^{-1}.

W_e = earth's angular rate of rotation.

G: Earth's gravitation = 980 cm sec^{-2}.

I: Index for x-axis.

IGO: Flag. Set as 1 initially; it changes to 0 when the calculation becomes unstable.

IM: Maximum number of half-grid point in x-direction.

IN: Maximum number of full-grid point in x-direction.

J: Index for y-axis.

JCTR: Index for simulation hour.

JM: Maximum number of half-grid point in y-direction.

JN: Maximum number of full-grid point in x-direction.

K: Index for σ-axis.

KN: Maximum number of full-grid point in σ-direction.

KZ: Maximum number of half-grid point in σ-direction.

KSTORE: Flag. Set as 1 to store hourly result on tape.

Set as 0 if no store is needed.

KVEL: Flag. Set as 1 if velocities are to be calculated, otherwise set as 0.

KTEMP: Flag. Set as 1 if temperature is to be calculated, otherwise set as 0.

LN: Set as 1 for 1st run of present case; set as n for subsequent n^{th} run.

MBLOK: Data block number which is to compare with data block NBLOK which is to be read in. Used only when LN > 1.

NBLOK: Index for data block.

NCASE: Case number.

NCY: Number of hours to be simulated in this run.

QQ: 57.3, used for changing from deg to rad.

ROSSBY: Rossby number.

RR: Water density, = 1.

RWEX: Number of hours between climatological input data, = 1.

TABN: Ambient water temperature outside of north entrance.

TABS: Ambient water temperature outside of south entrance.

THETA: Angle between north and y-axis, clockwise positive.

TINIT: Water temperature at initial instant before the waste heat discharge start.

TPRT: Time between printouts, in sec.

TTOT: Total simulation time, in sec.

TZ: Record of time for hourly printout.

TZERO: EST hour at the beginning of present simulation run.

The following symbols are used to specify tidal condition.

AMPLIT: Tide amplitude, in cm.

DPHASE: Phase lag per Δx, in hour.

PERIOD: Tide period, in hour.

PHASE: Phase difference between tides at north and south entrance.

STAGE: Difference in cm between daily mean level and short-term (weekly) average sea level.

STAGE1: Difference in cm between short-term (weekly) average sea level and long term average level (MSL).

TSHIFT: Time shift for adjusting tide with EST, in hour.

The following symbols are used to specify the hourly climatological conditions.

TAIR: Ambient air temperature, deg C.

HUMID: Relative humidity, fraction.

WIND: Wind speed, cm sec^{-1}.

WDIR: Direction from which wind is coming, deg measured clockwise from North.

SRAD: Gross solar radiation, in BTU/(ft^2 day).

TSURF: Surface water temperature, deg C.

The following symbols are related to climatological data and appear in the calculation of wind stress, equilibrium ambient temperature and heat exchange at surface.

EPSLON: Direction to which wind blows, in rad.

WPR: Wind speed in m sec^{-1}.

CTEN: Empirical constant appears in wind stress formula.

TAU: Wind stress τ.

TAUX: x-component of wind stress τ_x.

TAUY: y-component of wind stress τ_y.

TDEW: Dewpoint temperature, deg C.

TEQ: Equilibrium temperature, deg C.

SK: Surface heat exchange coefficient in cal/(cm^2 sec °C).

SECTION 5

PREPARATION OF SIMULATION RUN

This section describes the preparation work needed for ANCMN run. The flow chart and the associated subroutines in Figure 1 and Table 2 are referred to in the following description.

1. Specify number of full-grid points, IN, JN, KN and number of half-grid points, IM, JM, in PARAMETER statement. Although the domain of solution under consideration is usually smaller than the rectangular space of IN x JN x KN, the marker matrices will assure that the grid points outside of domain skip the calculation. To have a clear printout, the variables at off domain point have been set to 10^9. This value is beyond the capacity of the computer printout in printing real numbers (F format) so that stars will be printed and show the off domain area.

2. Specify run number by input data LN, card #2:

   For LN = 1, i.e. first run, data file or card deck of AMATN is needed.

   For LN > 1, i.e. subsequent run, tape with previous result is needed.

   Specify flag for storage by KSTORE, card #3:

   For KSTORE = 0, desire no storage.

   For KSTORE = 1, tape must be provided for storing results.

   Specify flag for velocity calculation by KVEL, card #4:

   For KVEL = 0, no V-calculation, thus thermal dispersion only.

   For KVEL = 1, do V-calculation, thus circulation included.

   Specify flag for temperature calculation by KTEMP, card #5:

   For KTEMP = 0, no T-calculation, thus only a hydrodynamic model.

   For KTEMP = 1, do T-calculation, a complete hydrothermal model.

Specify data block number MBLOK, to make sure the data read in from tape is correct, card #6.

Specify number of hours to be simulated in this run by NCY, card #7.

Specify the time between successive printouts by TPRT, card #8.

Specify grid size by input data DX, DY, DS, card #9.

Specify time step DT and tide data STAGE, AMPLIT, PHASE, DPHASE, PERIOD, TSHIFT by input data, card #10.

Specify Coriolis factor FCOR and stage STAGE1 by data, card #11.

Specify the angle between North and the y-axis of grid system by THETA, card #12.

Specify reference length ALREF and Rossby No. ROSSBY, card #13.

Specify number of hours between weather observations RWEX, card #14.

Specify TZERO, the Eastern Standard Time when the simulation starts, card #15.

Specify water density, vertical eddy viscosity, vertical eddy and horizontal eddy diffusivity, RR, AV, BV, BH, card #16.

Specify initial temperature TINIT, a constant for whole domain, card #17.

3. In general, the first run of present case has:

   LN = 1, KSTORE = 1, KVEL = 1, KTEMP = 1, MBLOK = 0.

   Then the subroutines BAYBOT and BAYINI are used to initialize the calculation. This includes reading matrices, MAR, ELEV, MEX, MEY, MX, MY, by BAYBOT from data file AMATN. The same subroutine calculates the derivative height matrices, HB, HU and HV. The initialization of various variable matrices is done in subroutine BAYINI and in the main program itself.

4. In general, the continued n^{th} run has the same NCASE with:

   LN = n, KSTORE = 1, KVEL = 1, KTEMP = 1.

   MBLOK = index number of the data block which is to be read in; the calculation will continue thereafter. In fact, the data being read contains all the information needed to continue the run. However, to allow for the freedom of matching tide of different amplitude, period and phase shift, an additional card (#18) specifying NBLOK, TTOT, DT, EST, AMPLIT, PHASE, DPHASE, PERIOD and TSHIFT is needed.

The data may be same as those contained in the tape or different from them so that the calculation goes on to follow another tide format. In accordance with this change of tide, the NBLOK, TTOT, DT, EST may be reset.

5. The main loop in the main program ANCMN is the hourly simulation loop, which is started with hourly climatological data card containing TAIR, HUMID, WIND, WDIR, SRAD and TSURF. The wind stress and equilibrium temperature are then computed and held thereafter as constants throughout that hour.

   The main part of the hourly loop is an internal loop for Δt increment, in which the main calculation is done in the order of (η, Ω), (u, v), T, W, then a printout of elevation, surface velocity and surface temperature at certain chosen half-grid points.

6. The V-calculation controlled by flag KVEL consists of subroutines BETA, BNRTIA, ABNR3, BVEL, ASAF3 and GIVENU. BETA computes η and Ω. BNRTIA and ABNR3 compute the convection terms, RX and RY, for the momentum equations; this computation is decided by whether ROSSBY is zero or not. The (u, v) calculations are done by BVELS and ASAF3. The given velocities at control points are re-specified by GIVENU. The BNRTIA and BVELS are for interior points while ABNR3 and ASAF3 perform the same purpose except for normal velocity points along open boundaries, where the water elevation is specified as a function of time.

   The T-calculation controlled by flag KTEMP consists of CONV, TCOMPT and GIVENT. CONV computes the convective term TC, then TCOMPT computes T, and GIVENT respecifies T at discharge points.

   After the completion of marching forward to $(n+1)\Delta t$, the variables are relabeled and UB and VB are computed as the horizontal components of velocity at centers of (I, J) blocks. Finally, before the printout of newly obtained variables at fixed locations to serve as flow development at fixed point, the surface velocity and temperature at a critical point are compared with preset values to see whether an instability has developed. If instability does occur, the program terminates after producing a hard copy of the latest result.

7. The subroutine ANCPR produces step-by-step records of surface elevation, surface velocity and surface temperature at certain chosen points. These points are selected because of the variables that are believed to undergo the most change, as they are close to open boundaries, river exit and discharge outlet.

   The hourly loop included subroutines SOTRET and TPRLOK too; the former stores the hourly results as well as all pertinent data onto tape for later uses, and the latter produces a printout of resultant horizontal velocities, vertical velocities, temperatures at four levels and elevation of free surface. In addition, the main program ANCMN itself does the calculation and printout of total water depth, $H = h + \eta$, before starting next hourly loop.

8. ANCMN performs the hourly loop NCY a number of times. Therefore, NCY number of climatological data cards are needed to provide the necessary data.

SECTION 6

INPUT DATA

The input cards for running ANCMN are given in Table 5 below. Note that the data symbols have already been defined in the previous section; however, the following remarks should be considered.

* Free format is used for all data input.

* Distinction must be made for integer and real number.

* The order of these cards must be followed.

TABLE 5. Input Data for ANCMN

Input	Card Content	Symbol	Definition/Value
#1	1	NCASE	= Case No.
#2	1	LN	= 1, if it is first run then data file or card deck AMATN is needed in #18 = n, if it is n^{th} run
#3	1	KSTORE	= 0 no store, then there are no continued runs = 1 store intermediate results on tape for plotting or next run
#4	1	KVEL	= 0 no V-calculation, i.e. dispersion of T by given (u, v) field = 1 do V-calculation, so momentum is under dispersion
#5	1	KTEMP	= 0 no T-calculation, i.e. hydrodynamic only = 1 i.e. hydrothermal model
#6	1	MBLOK	= No. of latest hour of last run

TABLE 5. Input Data for ANCMN
(Continued)

Input	Card Content	Symbol	Definition/Value
#7	1	NCY	= Number of hours intended for simulation in this run
#8	1	TPRT	= 3600 seconds, hourly loop
#9	3	DX	= x-direction grid size in cm
		DY	= y-direction grid size in cm
		DS	= σ-direction grid size in nondimensional unit
#10	7	DT	= Time step in sec, follows $\Delta T < \frac{\text{Min}(\Delta X, \Delta Y)}{\sqrt{2gh_{max}}}$
		STAGE	= Average level of tide-MWL, in cm
		AMPLIT	= Amplitude of tide, in cm
		PHASE	= Phase lag of the north tide behind the south tide, in hr
		DPHASE	= Phase lag in east-west direction, in hour per Δx
		PERIOD	= Period of tide at entrances, in hr
		TSHIFT	= Time shift (in hr) for tide to agree with EST time
#11	2	FCOR	= Coriolis factor = $2W_e$ sin(lat), in sec^{-1}
		STAGE1	= MWL-MSL (datum for sounding), if MWL $\neq$ MSL, in cm
#12	1	THETA	= Clockwise angle from North to the y-axis of grid work, in deg
#13	2	ALREF	= Horizontal reference length, in cm

TABLE 5. Input Data for ANCMN
(Continued)

Input	Card Content	Symbol	Definition/Value
		ROSSBY	= Rossby No. which controls whether advection is needed to account for in the equations of motion, zero or nonzero
#14	1	RWEX	= Number of hours between climatical data, generally it agrees with hourly loop
#15	1	TZERO	= EST when the simulation run starts
#16	4	RR	= Density of water = 1.0
		AV	= $0.002\ (AHREF)^{4/3}$, AHREF = reference depth in cm
		BV	= Same value as A_v
		BH	= $0.002\ (ALREF)^{4/3}$
#17	1	TINIT	= Initial temperature, a constant for the whole domain
#18	A deck of cards or data file, called AMATN, to specify matrices MAR, ELEV, MEX, MEY, MX and MY. It is required only if LN = 1.		
#19	9	NBLOK	= To reset data block number
		TOTT	= To reset total time in sec if necessary
		DT	= To reset time step ΔT if necessary
		EST	= To reset EST
		AMPLIT	= If tide changes
		PHASE	= New phase lag
		DPHASE	= New phase lag per ΔX
		PERIOD	= If tide has different period

TABLE 5. Input Data for ANCMN
(Continued)

Input	Card Content	Symbol	Definition /Value
		TSHIFT	= Time shift of the new tide
#20	A deck of NCY cards, each card contains six hourly weather data:		
		TAIR	= Air temperature, deg C
		HUMID	= Relative humidity in fraction of unit
		WIND	= Surface wind speed, cm/sec
		WDIR	= Wind direction, from which direction wind is blowing
		SRAD	= Gross solar radiation, BTU/(ft^2 day)
		TSURF	= Ambient surface water temperature, deg C

SECTION 7

PLOTTING PROGRAM

This section presents the descriptions of main plotting program PLOTMN and its subroutines. As mentioned earlier, the plotting and analyzing of the results constitute Part 2 of the three-dimensional, free-surface model. Here, the tape containing the hourly results of simulation is the main input. The control data cards help the user to choose the hour, the plot and the comparison. The output is in a plot tape which is used by a CALCOMP plotter to generate plots.

DESCRIPTION AND FLOW CHART OF PLOTMN

The purpose of PLOTMN is to read in the measured and/or calculated temperature fields and to plot the isotherms. In addition, the calculated velocity field is plotted in σ-planes (σ-axis), in certain x-cross sections (y-axis) and in y-cross sections (x-axis), and the surface elevation field is plotted in contour plots.

Since the main input is the data block from Part 1, the symbols and their dimensions agree with those that appear in ANCMN. In order to store the IR scanned surface temperature at four tidal stages for later comparison with calculated results, a TIR (IM, JM, 4) matrix is added. It is to be noted that temperature fields are interpolated at half-grid points from the mosaic IR images by hand. Several data cards containing the quantities to be used in plot caption are read in as well as control cards which assign the data block to be used (NPLOUT) and the plot to be done (IPLOT). A flag NSTAND is to assign which measured temperature field is to be compared with the calculated. The algorithm for PLOTMN is simple and straightforward since no complicated calculation is involved. The only calculation is to compute average deviation of calculated temperature field from measured temperature field at the same tide stage. The average deviation is given by

$$\delta^2 = \frac{\sum_{i,j} (TB(i, j, 1) - TIR(i, j))^2}{\sum_{i,j} (i, j)}$$

where TB is the calculated temperature while TIR is the measured temperature by infrared, and $\sum_{i,j}(i, j)$ is the number of surface half-grid points in the domain.

The isotherms of TIR and TB are the plots of main concern, as one is to be compared with the other in order to assess the accuracy of the model in predicting the hydrothermal dispersion of waste heat. Occasionally, the water surface contour, the surface current and the velocity profile are also of interest, as they depict the circulation set up by the tide and the wind in conformity with the configuration and bathymetry of the waters. The surface elevation contour is done by the subroutine ECHKON which is also used for plotting isotherms, but care must be taken to assign the contour values, since the surface elevation changes with tide; thus, the hard copy printout of surface elevation ETA must be consulted in order to choose the right contour values. The surface current is done by PLOTUV, while the velocity profiles in j(y)-cross sections and i(x)-cross sections are plotted by PLOTUW and PLOTVW, respectively. It is noted that the velocity scale for horizontal components is different from that for vertical component. The ratio is to remain the same as the ratio of horizontal to vertical length scale. Therefore, the velocity profiles are exaggerated in vertical direction; however, since the horizontal velocities of the top level ($\sigma = 0$) are plotted right on the water surface, they show free-surface profiles too. Note that the j-sections and i-sections are fixed by given I- J- values. They are chosen by the user's concern about the effect of plan-form configuration and bathymetry on currents.

The flow chart for the main program PLOTMN is presented in Figure 4, and the subroutines are listed in Table 4 for quick reference. It is to be noted that several CALCOMP subroutines are also listed.

SUBROUTINES

ECHKON

This subroutine calls subroutines CONLIN and ENDER. This program was developed by the National Hurricane center for map contouring using CALCOMP or MILGO-type plotter. ECHKON is the entry point for the package. It scans the rectangular gridded scalar field, such as surface temperature or surface elevation, to determine where to start a new contour. Each contour is done in a loop. Inside the loop the subroutine CONLIN is called to do the interpolation and drawing, and ENDER is called by CONLIN to label each contour of the same contour value. The exit of contour loop in ECHKON is made when the final contour value increased by increment has reached the specified maximum. Here, ECHKON is used for contouring TIR(i, j, N = 1, 4), TB(i, j, 1) and ETA(i, j).

PLOTUV

It computes the horizontal resultant velocity from components u and v at each level. In general, there are four levels corresponding to k = 1 to 4. However, for the present problem, only the surface current is of interest; therefore, KPLOT is set to 1.

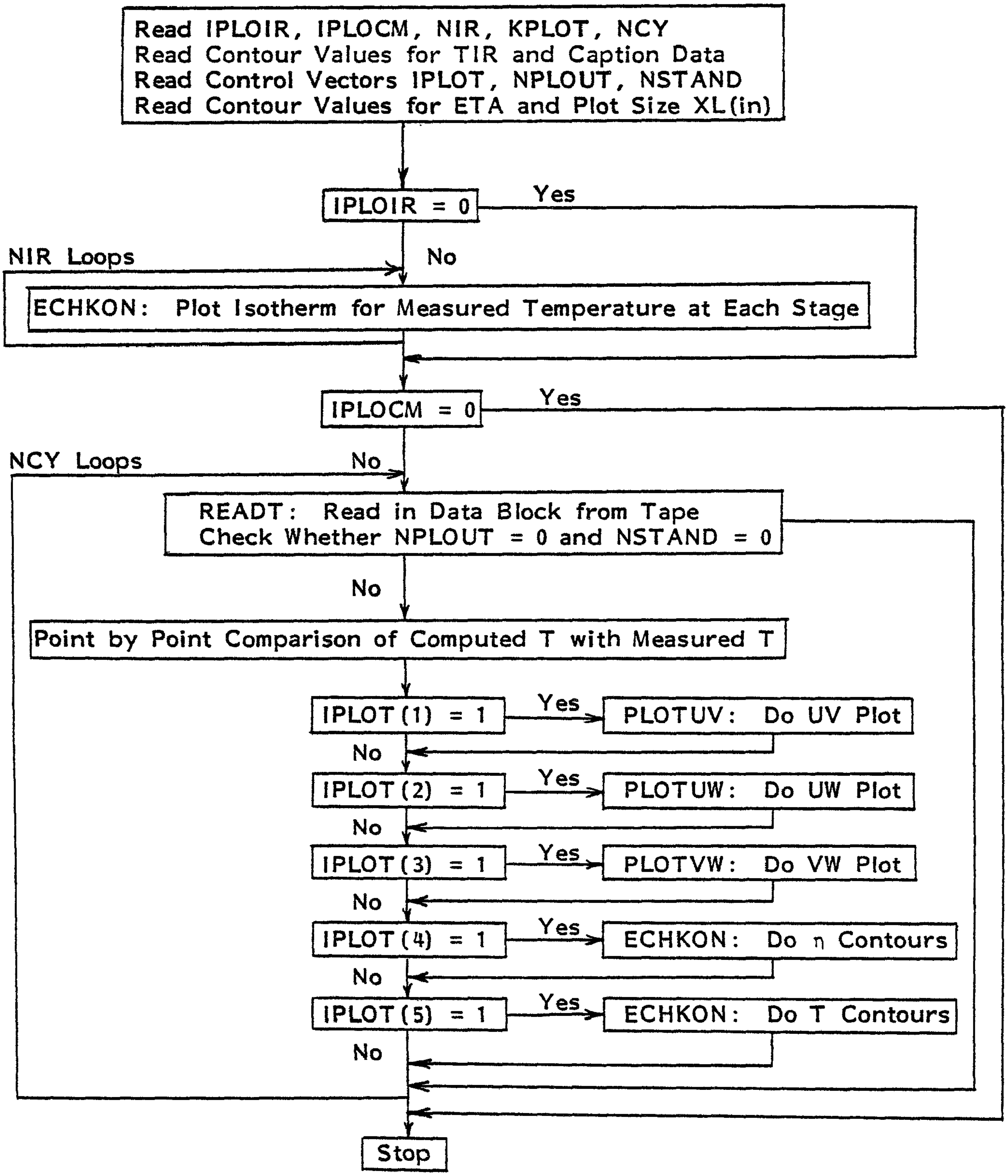


Figure 8. Flow chart for plotting program PLOTMN and its subroutines

TABLE 6. Subroutines Required in Main Plotting Program PLOTMN

No.	Name	Description	Remark
1	FACTOR		
2	PLOTS	Are CALCOMP Subroutines	In UCS * ACALCOMP of UNIVAC 1100
3	PLOT		
4	ECHKON	Subroutine for plotting isotherms and contour of surface elevation; also draws the domain	Calls subroutines ENDER, CONLIN, OUTLIN
5	READT	Same as READT in ANCMN, for read in-stored data from tape	
6	PLOTUV	Plot U, V on different layer, to select the layer one can choose KPLOT value; normally it is the surface layer	Calls subroutine OUTLIN
7	PLOTUW	Plot U, W on chosen J sections west-east across the bay	J section once chosen is fixed
8	PLOTVW	Plot V, W on chosen I sections south-north across the bay	I section once chosen is fixed
9	CAPTN1	Write common heading on each diagram	
10	CAPTN2	Write title for UV Plot	
11	CAPTN3	Write title for IR-T isotherms	
12	CAPTN4	Write deviation of calculated temperature from IR-T	
13	CAPTN5	Write tidal stage on the diagram	
14	CAPTN6	Write title for UW plot	
15	CAPTN7	Write title for VW plot	
16	CAPTN8	Write title for surface elevation contours	

TABLE 6. Subroutines Required in Main Plotting Program PLOTMN (Continued)

No.	Name	Description	Remark
17	CAPTN9	Write title for calculated surface isotherm	
18	ENDER	Write contour value to label contour	Subroutine of CONLIN
19	CONLIN	Called upon to draw individual contour	Subroutine of ECHKON
20	OUTLIN	Called upon to draw outline of the computational domain	Subroutine of ECHKON and PLOTUV
21	FIT	Fit a parabola to three point used in PLOTUV to interpolate water depth	Subroutine of PLOTUV
22	VECT	Calculates the velocity and calls AROHD to draw the vector	Subroutine of PLOTUV
23	AROHD	Are CALCOMP subroutines, called upon in various sub-routines used in PLOTMN	In library file UCS * ACALCOMP of UNIVAC 1100
24	NUMBER		
25	SYMBOL		

PLOTUW

It computes the resultant velocity based on components u and w. As mentioned, the vertical and horizontal components cannot be made to the same scale; therefore, the velocity profiles are distorted. The j-cross sections on which the velocity is computed and plotted are preassigned in the subroutine itself. For the present problem, plots are drawn for j = 4, 8, 12. All three plots are done on the same sheet.

PLOTVW

It computes the resultant velocity based on components v and w, and plots the velocity vectors on the vertical cross section along y-direction. The i values are likewise preassigned in the subroutine. Here plots are drawn for i = 4, 8, 12 and are on one sheet.

INPUT DATA

Table 7 lists the data to for PLOTMN. Free format is used generally. The total input consists of data cards and data files, but in the list, the input data have been numbered in the order of their appearance, regardless of whether card form or file form is used.

TABLE 7. Input Data for PLOTMN

Input	Card Content	Symbol	Definition/Value
#1	5	IPLOIR	= 0, if no isotherm of IR obtained temperature is desired = 1, otherwise
		IPLOCM	= 0, if no isotherm of computed temperature is to be plotted = 1, otherwise
		NIR	= Number of IR obtained temperature fields
		KPLOT	= Number of the σ-level to be plotted
		NCY	= Number of simulation hours
#2	NIR	TL(I)	= Array of min. contour values for TIR
#3	NIR	TH(I)	= Array of max. contour values for TIR
#4	NIR	TI(I)	= Array of increment values for TIR contouring
#5	NIR	TA1(I)	= Array of ambient temperature, assigned to out-domain points
#6	NIR	Q1(I)	= Array of EST time, for caption
#7	NIR	Q2(I)	= Array of wind speed values, for caption on TIR plot
#8	NIR	Q3(I)	= Array of wind direction values, for caption on TIR Plot
#9	NIR	Q4(I)	= Array of air temperature for caption on TIR plot
#10	NIR	Q5(I)	= Array of discharge temperature for caption on TIR plot
#11	1	Q6	= Discharge flowrate, for caption in general
#12	5	IPLOT(I)	= Array of integers to assign the plot desired

TABLE 7. Input Data for PLOTMN (Continued)

Input	Card Content	Symbol	Definition/Value
#13	NCY	NPLOUT(I)	= Array of integers to assign the hour desired to be plotted
#14	NCY	NSTAND(I)	= Array of intergers to assign the TIR to be compared with
#15	NIR	ETAL(I)	= Array of min. contour values for ETA
#16	NIR	ETAH(I)	= Array of max. contour values for ETA
#17	NIR	ETAINT(I)	= Array of increment values for ETA contouring
#18	1	XL	= Plot size in x-direction (in)
#19	2	DX, DY	= Spacing in x- and y-direction (cm)
#20	Data File	I, J	= The (i, j) value of boundary nodes, used for drawing domain's boundary
#21	Data File	TIR(I,J,N)	= To read in the IR measured temperature fields of different tidal stage, files like HDATA, EDATA, LDATA, FDATA

REFERENCES

Carter, C. V. The Hydrothermal Characteristics of Shallow Lakes. Ph.D. Thesis, Department of Mechanical Engineering, University of Miami, Coral Gables, Florida. December 1977.

Dunn, W. E., Policastro, A. J. and R. A. Paddock. Surface Thermal Plumes: Evaluation of Mathematical Models for the Near and Complete Field. Part One and Two. Energy and Environmental Systems Division, Great Lakes Project, Argonne National Lab., May 1975.

Lee, S. S. and S. Sengupta. Three-dimensional Thermal Pollution Models Volume I = Review. Department of Mechanical Engineering, SEA, University of Miami, Coral Gables, Florida, 1978.

Tuann, S. Y., Lee, S. S., Sengupta, S. and C. R. Lee. Application of Three-dimensional, Free-surface Model to Shallow Tidal Waters. Proceedings of the Third International Symposium on Computer Methods for Partial Differential Equations, Bethlehem. June 1979.

APPENDIX A

EXAMPLE CASE

INTRODUCTION

The present model has been successfully applied to thermal dispersion study at Anclote Anchorage. The Anchorage is located on Florida's Gulf coast and north of St. Petersburg (Figure 9). It is a relatively shallow passage between the mainland and the Anclote Key. A series of barrier islands separates the anchorage from the Gulf of Mexico. Through natural channels to the north and south of the Keys, the Anchorage has an unrestricted exchange of water with the Gulf.

The Anclote power plant operated by the Florida Power Corporation has two 515 MW, oil-fired, electrical generating units. Once-through cooling water is drawn from the Anclote River through a man-made canal. The six pumps delivering a total of 1,990,000 gpm (125.6 m^3/sec) are designed to raise the water temperature 2.8°C above the ambient. The heated water is discharged back into the Anchorage through the discharge canal with a dredged submarine extension. The designed total flowrate is approximately 53 times the long-term average flowrate of the Anclote River. At present, only Unit 1 is operative while Unit 2 is still pending permission. That is, the present flowrate is 62.8 m^3/sec (995,000 gpm).

The principal driving mechanism for current circulation is tidal flux at the north and south entrances of the Anchorage. The tide is predominantly semidiurnal with mean range of 2 feet. Earlier measurements of temperature and salinity indicated the currents flow in and out through both entrances; however, the exchange appears to be stronger in the south than in the north, or the currents generally flow north during flood tide and south during ebb tide. Moreover, the wind plays an important part too. The surface current direction depends on wind blowing at wind speeds exceeding 15 mph.

The model as applied to the Anclote Anchorage shows its capacity of considering the effects of geometry and bathymetry, spatio-temporal variation of the free surface, various boundary conditions, including tides of different phase and range, surface heat transfer based on equilibrium temperature concept, and changing meteorological conditions. In addition, turbulence has been considered by using the eddy transport concept, and the effects of baroclinicity have been included. Again, the user should refer to Tuann et al. (1979) for the general review, mathe-

Figure 9. Anclote Anchorage location in the state of Florida

matical formulation, finite difference implementation and numerical method of solution.

The finite difference grid work is three dimensional and is designed to cover the area of interest. The grid size $\Delta x \times \Delta y \times \Delta \sigma$ is that the least number of vertical layers is four. That is, a 4-layer, 5-level, vertical partition is a reasonable choice for the present generation of computer. The grid work is allowed to orient away from north-south, east-west system, but in general, the x-axis of the grid system aligns with west-east, and y-axis with south-north. Thus, the subscript i increases eastward, while j increases northward. The z-axis is chosen upward from mean water surface, while the subscript k increases downward from the water surface. That is, the $k = 1$ level is the free surface which is continuously changing, while $k = 5$ is always the bottom.

For the study of Anclote Anchorage, the grid is 16 x 14 x 5 with five levels, each with 224 nodes for a total of 1120 nodes. The grid size used is $\Delta x = \Delta y = 417$ m. Depths off the natural coastal line are read from the Coast and Geodetic Survey chart. The maximum depth is 4 m at the south end of Anclote Key. It was found that gravity waves were the dominating consideration with regard to the maximum allowable time step Δt. A 15 second magnitude of Δt was found to work well for the present grid system.

Numerical results were obtained with the University of Miami Computing Center UNIVAC 1100 computer. The time histories of the three velocity components, (u, v, w), the surface elevation, η, and the temperature, T, for a 24-hour simulation period were obtained with about 90 minutes of computer time in most cases. This is a time ratio of about 16:1 (the ratio of real time to computer time).

PROBLEM STATEMENT

Florida Power Corporation has a fossil fuel power plant situated at Tarpon Springs on the Anclote Anchorage. The discharge rate is 62.8 m^3/sec of water at temperatures, in general, 2.8°C above the ambient water. On June 19-20, 1978, a team carried out an in-situ data acquisition mission to gather field data on temperature and current. At the same time, four flights by NASA/KSC were undertaken to obtain temperature by remote sensing method. These four flights were intended to cover four different tidal stages in the Anchorage. The remotely sensed data were processed into digicolor film. The in-situ measurement of surface water temperature at the time when the airborne IR data was undertaken provides a reference for IR temperature. With this reference, the isotherms were drawn from the digicolor film. The in-situ current measurement data were used in plotting current of different depth. The ground measured temperatures were used to draw surface and subsurface isotherms.

Once the model has been verified for its versatility and its capacity,

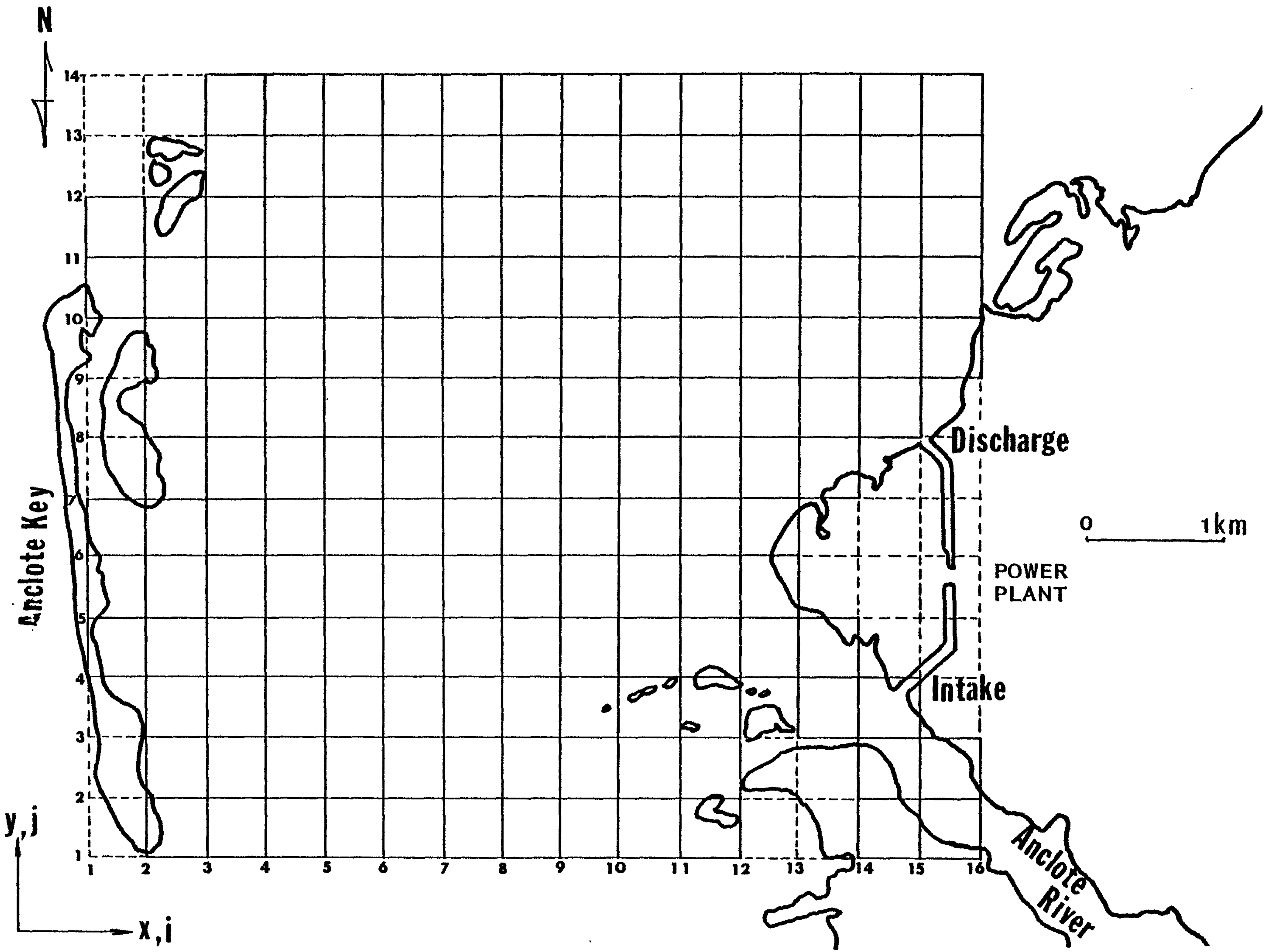


Figure 10. Grid work for the Anclote Anchorage

and in particular, to the prediction of hydrothermal development in well-mixed shallow coastal waters, the model was run with actual tidal and meteorological data as input, but the initial temperature condition either could be that of uniform state, that is to assume the power plant start impulsively, or could be that of an IR temperature field, that is using IR data as the initial temperature field. The demonstrative runs were carried out to simulate the hydrothermal situation for several days. The predicted current fields are verified against the in-situ measured currents, and the predicted isotherms are verified against the IR-obtained surface isotherms and the in-situ measured subsurface isotherms.

CALCULATION OF PARAMETERS AND INPUT DATA

In this section, the specification grid system, reference and physical parameters, tidal and meteorological data, discharge and intake velocities, ambient and discharge temperatures will be presented. The actual calculation of some input data quantities is carried out in detail for the purpose of demonstration.

Grid System

The map indicating the exact locations of power plant, intake and discharge outlets, and the sounding of the Anchorage was used to determine the size of the domain, the grid system to cover it, and the arrangement of intake and discharge points in the system. So, a domain of about 6 km x 5 km covering most of the Anchorage was used. A grid system of 16 x 14 was selected in the horizontal plane. The size of the grid cell is $\Delta x = \Delta y = 416.7$ m; this size and the grid orientation has made the intake and discharge outlets to the open water fall in with nodes respectively, and the intake and discharge channels have 45° and 315° orientation respectively. The depth was specified according to sounding chart. There are five nodes in the vertical direction. This gave a total of 16 x 14 x 5 nodes. The coordinate system and grid work are shown in Figure 10. The MAR matrix, bottom elevation matrix and four additional marker matrices are stored in data file AMATN.

Reference Quantities

L: Reference length = ALREF = Maximum Length = 6 km

BH: Horizontal eddy diffusivity = $0.002L^{4/3}$ = $0.002 \times (600{,}000)^{4/3} = 100{,}000$ cm²/sec

BV: Vertical eddy diffusivity = $0.002H^{4/3}$ (H = maximum depth) = $0.002 \times (360)^{4/3} = 6$ cm² sec

For shallow well-mixed tidal water about three times the calculated value was found suitable. Here, we use BV = AV = 20 cm²/sec, i.e. Turbulent Prandtl No. = 1

RR: Density of water = 1.0

THETA: 0, since the grid system orients along North-South

RWEX: 1.0

ROSSBY: 0.0, that is, based on test run, it was shown that the nonlinear inertia terms can be safely neglected to save computation

TINIT: Initial uniform temperature or reference temperature = 20 deg C

Calculation of Time Step, DT

In order to determine the time step, DT, the stability criterion has to be followed, which is done as follows.

$$DT < \frac{DX}{\sqrt{2gH}} = \frac{41760}{\sqrt{2 \times 980 \times 360}} = 50 \text{ sec}$$

About 1/3 of this value is reasonably safe to use. Here we use DT = 15 sec.

Calculation of Intake and Discharge Velocities

1. Flowrate = 995,000 gpm (from power plant physical data)
   $= 62.8 \ m^3/sec$
   $= 62.8 \times 10^6 \ cm^3/sec$

2. Both intake and discharge canals are at 45° from N, therefore

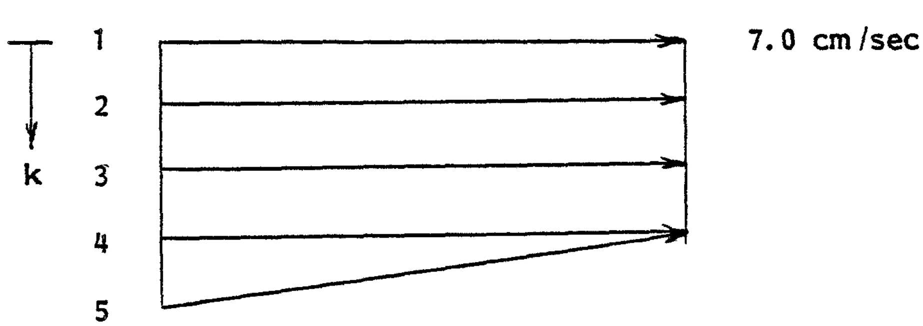
   $31.4 \times 10^6 \ cm^3/sec$ is crossing the Δx and Δy at the point of intake and discharge.

3. The average depth at intake and discharge is approximately 4' or 122 cm, and the width is $\Delta x = \Delta y = 41760$ cm, so the cross-sectional area is $41760 \times 122 \ cm^2$.

4. The average velocity is:

$$U_{ave} = V_{ave} = \frac{31.4 \times 10^6}{41760 \times 122} = 6.163 \text{ cm/sec}$$

5. The velocity profiles are assumed as shown.



6. To allow for canal storage during tide change we assume the intake and discharge velocities to be sinusoidal, i.e.

Intake: $V3(14,\ 4,\ k) = 7 - 3 \times \cos[\frac{2\pi}{12.5}(EST - 7.625)]$

$U3(15,\ 3,\ k) = V3(14,\ 4,\ k)$ for $k = 1, 2, 3, 4$

Discharge: $V3(14,\ 8,\ k) = 7 - 3 \times \cos[\frac{2\pi}{12.5}(EST - 7.5)]$

$U3(15,\ 8,\ k) = -V3(14,\ 8,\ k)$ for $k = 1, 2, 3, 4$

where 7.625 and 7.5 are taken to be the phase shift which takes into account the time to travel from the south end of Anclote Key to the concerned point.

Calculation of Tide on June 20, 1978

Simulated diurnal tide is shown in Figure 11, where

1. Period = 12.5 hr
2. Stage = short term average sea level - MSL = 48 cm
3. Amplitude = ½ short term average tide range = 65 cm
4. Time shift = 7.125 hr

   i.e. at 7.125 a.m., June 19, 1978, the tide at the south end of Anclote Key was zero.

5. W - E lapse = 0.014 hr/DX

   Wave propagation speed $C = \sqrt{2gh} = \sqrt{2 \times 980 \times 360} = 850$ cm/sec (H = 360 cm is the maximum depth of the Anchorage.)

   The time needed to travel one grid distance is

$$\frac{DX}{C} = \frac{41760}{850} = 50 \text{ sec} = 0.014 \text{ hr.}$$

   We use 0.014 hr per DX for phase shift in W - E direction and the imposing tide at the south entrance is

$$\eta_s = 48 + 65 \sin[\frac{2\pi}{12.5}(EST - 7.125 - 0.014(I - 1))]$$

   I = grid no. in W - E direction.

6. S - N lapse = 0.15 hr.

   Distance from south entrance to north entrance is about 543,000 cm.

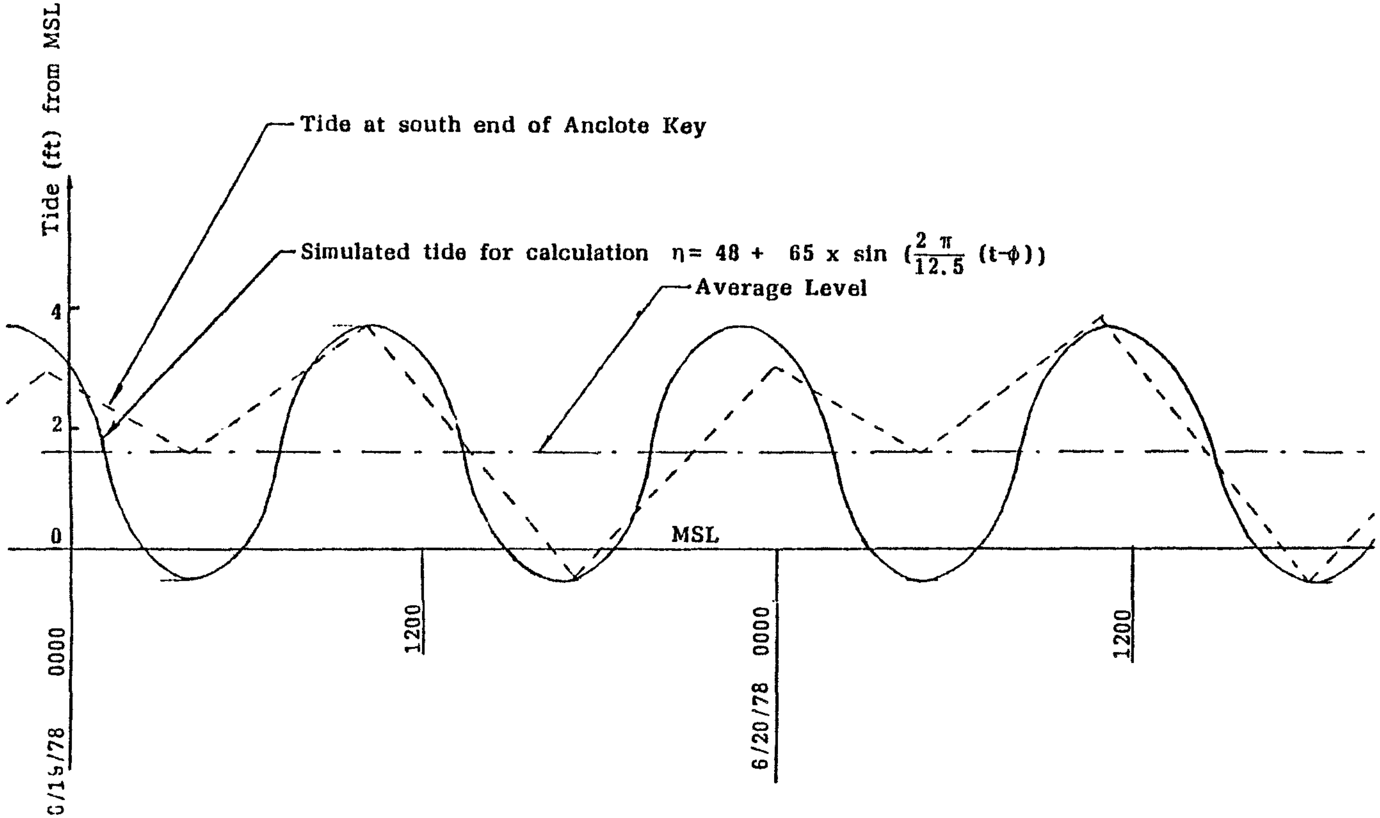


Figure 11. Semidiurnal tide for June 19-20, 1978 at south end of Anclote Key

Time for wave to travel this distance is $\frac{543000}{850}$ = 0.18 hr. We take 0.15 hr as phase difference between the south and the north boundaries; there, the imposing tide at the north entrance is

$$\eta_n = 48 + 65 \sin[\frac{2\pi}{12.5}(EST - 7.125 - 0.15 - 0.014(I - 1)]$$

Calculation of Anclote River Flowrate and Temperature

1. The distance traveled from South Anclote Key to Tarpon Springs is 20 DX. We estimate a time lapse of 0.5 hr to account for the retardation due to buffering effect of river storage and Anclote River's natural outflow.

2. The average current is estimated to be 20 cm/sec, therefore, we take

$$U3(16, 1, k) = 20 \cos[\frac{2\pi}{12.5}(EST - 7.625)]$$

$$V3(15, 1, k) = -20 \cos[\frac{2\pi}{12.5}(EST - 7.625)]$$

for k = 1, 2, 3, 4.

3. The surface elevation at Tarpon Springs is to be calculated.

4. To be in accordance with given velocities at Tarpon Springs, the temperature there is also assigned and its value has a 24 hr period instead of 12.5 hr. This temperature is

$$T3(15, 1, k) = 26.9 + 0.5 \sin[\frac{2\pi}{24}(EST - 12)]$$

where the 12 hr shift is to make the peak temperature occur at 1800. Thus, the water in and out at Tarpon Springs has a temperature ranging from 26.4 (before dawn) to 27.4 (late afternoon).

Discharge Temperature and Gulf Water Temperature

1. On June 19-20, 1978, the recorded discharge temperature at daytime is in the range of 29.3-30.3. To account for the further drop of discharge temperature due to cooler ambient temperature at nighttime, we assume a sinusoidal variation of discharge temperature with diurnal period.

2. Discharge temperature is estimated

$$T3(14, 8, k) = 29.4 + 0.4 \sin[\frac{2\pi}{24}(EST - 12)]$$

therefore, the highest discharge temperature of 30.3°C occurs at

6 p.m. and the lowest (29.3°C) at 6 a.m.

3. The Gulf water outside the Anclote Anchorage as well as the atmosphere is sink to the heat disposal from the power plant; therefore, the boundary condition on temperature at the north and south entrance is not considered as adiabatic as in normal case of far-field thermal pollution problem. Instead, we specify the outside-anchorage ambient temperatures. Again, they are 24 hr periodic and their values should be in accordance with the measured temperature in the same neighborhood. Here in compliance with measured data, we use

$$T_{ab} = 27.0 + 0.2 \sin[\frac{2\pi}{24}(EST - 12)]$$

for both ambient temperature outside the south and the north boundaries.

EXECUTION DECKS FOR CALCULATION AND PLOTTING RUNS

The following execution decks are for use in UNIVAC 1100 computer at the University of Miami. These may have to be modified if a different computer is used. The programs and subroutines used in these runs are all compiled and stored in the file.

Calculation Run

First Run--

1. @ ASG, A FILENAME.
   The file 'FILENAME' is assigned for the run.

2. @ ASG, T 8., 16N, TAPENAME1
   A tape file names '8' is being assigned. The tape is 9-track, and the reel number is 'TAPENAME1.'

3. @ PRT, S FILENAME.ANCMN
   The main program 'ANCMN' is printed.

4. @ PACK FILENAME.
   'FILENAME' is packed together, eliminating the space left by deleted elements, and thus, condensing the file.

5. @ PREP FILENAME.
   Prepare an entry point table for the 'FILENAME.'

6. @ MAP, S
   Combines relocatable elements to form an executable absolute element.

7. IN FILENAME.ANCMN

8. LIB FILENAME.

9. END

10. @ XQT

11. 3
Case number (NCASE).

12. 1
First run (LN).

13. 1
Store the calculation results on to tape 'TAPENAME1' (KSTORE).

14. 1
Calculate velocities (KVEL).

15. 1
Calculate temperatures (KTEMP).

16. 0
Specify numbers of the latest hour of the last run (MBLOCK).

17. 13
Number of hours to be simulated (NCY).

18. 3600.
Print the results at each 3600 seconds (TPRT).

19. 41760., 41760., 0.25
Grid sizes in x-, y- and σ-direction (DX, DY, DS).

20. 15.0, 0., 60., 0.08, 0.005, 12.5, 7.125
Specify time step, the difference of average tidal level and mean water level, amplitude of tide, north-south phase lag, east-west phase lag per DX, period of tide, and time shift (DT, STAGE, AMPLIT, PHASE, DPHASE, PERIOD, TSHIFT).

21. 0.66E-4, 48.8
Coriolis factor and the difference of mean water level and mean sea level (FCOR, STAGE1).

22. 0.
The y-axis coincides with North (THETA).

23. 8.E5, 0.
Horizontal reference length and Rossby number (ALREF, ROSSBY).

24. 1.
Number of hours between climatic data (RWEX).

25. 7.
Simulation run starts at EST 0700 (TZERO).

26. 1.026, 20., 20., 50000.
Specify water density, vertical eddy viscosity, vertical eddy diffusivity, horizontal eddy diffusivity (RR, AV, BV, BH).

27. 27.0
Initial water temperature (TINT).

28. @ ADD FILENAME.AMATN
Input data file 'AMATN' for specifying grid matrices and initial water depth.

29. @ ADD FILENAME.FDATA
Input data file 'FDATA' for initializing temperature distribution.

30. @ ADD FILENAME.C2007
Input data file 'C2007' of climatic data.

31. @ FIN
Terminate this calculation run.

Sebsequent Run--

1. @ ASG, A FILENAME.

2. @ ASG, T 7., 16N, TAPENAME1
A tape file named '7' is being assigned; the reel number is 'TAPENAME1.' This tape was used in the first run for storing the hourly calculation results for 13 hours in 13 blocks.

3. @ MOVE 7., 12
Move TAPENAME1 to the 13th block which is the last hour result of the first run and is going to be used as input data for this subsequent run.

4. @ ASG, T 8., 16N, TAPENAME2
A new tape named 'TAPENAME2' is assigned to store the calculation results of this run.

5-13. Same as the cards 3-11 of the first run.

14. 2
Continuing run (LN).

15-17. Same as the cards 13-15 of the first run.

18. 13
The last hour of the first run is 13.

19-29. Same as the cards 17-27 of the first run.

30. Same as the card 20 of the first run. If new tidal data is needed, this card has to be changed.

31. @ ADD FILENAME.FDATA
If different IR temperature distribution is needed, FDATA has to be changed.

32. @ ADD FILENAME.C2007
The data file C2007 has to be changed since the weather condition will be different from the first run.

33. @ FIN

Plotting Run

1. @ ASG, A FILENAME.

2. @ ASG, T 7., 16N, TAPENAME1
A tape file named '7' is being assigned. 'TAPENAME1' stored the results of the calculation run.

3. @ ASG, T 11., 16, TAPENAME2
A tape file named '11' is being assigned. The tape is 7-track, and the reel number is 'TAPENAME2.' This is used for plotting tape.

4. @ PRT, S FILENAME.PLOTMN

5. @ PACK FILENAME.

6. @ PREP FILENAME.

7. @ MAP, S

8. IN FILENAME.PLOTMN

9. LIB FILENAME.

10. LIB UCS*ACALCOMP.
Call 'CALCOMP' plotter library.

11. END

12. @ XQT

13. 1, 1, 1, 1, 6
Plot IR isotherms and computed isotherms; only one IR temperature field is to be plotted. Plot isotherms on the surface level only, and run for 6 simulation hours (IPLOIR, IPLOCM, NIR, KPLOT, NCY).

14. 27.5

Minimum contour value for IR plot (TL).

15. 30.0
Maximum contour value for IR Plot (TH).

16. 0.75
Increment of contour value for IR plot (TI).

17. 27.0
Ambient temperature (TA1).

18. 13.
EST for caption (Q1).

19. 358.0
Wind speed in cm/sec for caption (Q2).

20. 110.
Wind direction for caption (Q3).

21. 29.4
Air temperature for caption (Q4).

22. 29.5
Discharge temperature for caption (Q5).

23. 62.7
Discharge flowrate in cm^3/sec for caption (Q6).

24. 1, 1, 1, 0, 1
Plot UV, UW, VW velocities and isotherms (IPLOT).

25. 0, 0, 0, 0, 0, 1
Plot the results at the 6th hour (NPLOUT).

26. 0, 0, 0, 0, 1, 1
Compare the deviation of computed temperature from IR tempterature (NSTAND).

27. 0.0
Minimum contour value for surface height (ETAL).

28. 0.0
Maximum contour value for surface height (ETAH).

29. 0.0
Increment of contour value for surface height (ETAINT).

30. 6.
6" plot size in x-direction (XL).

31. 42000., 42000.
    Grid size (cm) in x- and y-direction (DX, DY).

32. @ ADD FILENAME.APER1
    Specify boundary nodes for plotting the boundary.

33. @ ADD FILENAME.EDATA
    Input data file 'EDATA;' specify IR temperature distribution at ebb tidal stage.

34. @ FIN
    Terminate this plotting run.

The input data file AMATN, FDATA, C2007, APER1 and EDATA are listed in the Appendix B. If these data are not stored in the 'FILENAME,' card decks have to be substituted.

SAMPLE OUTPUT (EST 1300, June 20, 1978)

TAIR,HUMID,WIND,WDIR,SRAD,TSURF						29.40		.62	357.60		110.00		.40	27.00		
18015.	39.0	14.7	218.	26.8	40.7	1.1	219.	27.1	40.8	10.7	292.	26.9	41.7	7.7	323.	28.7
18030.	38.9	14.7	218.	26.8	40.6	1.1	221.	27.1	40.7	10.7	292.	26.9	41.7	7.7	323.	28.7
18045.	38.8	14.7	217.	26.8	40.5	1.1	219.	27.1	40.6	10.7	292.	26.9	41.6	7.6	323.	28.7
18060.	38.7	14.8	217.	26.8	40.4	1.1	221.	27.1	40.5	10.7	292.	26.9	41.5	7.6	323.	28.7
18075.	38.6	14.8	217.	26.8	40.4	1.1	220.	27.1	40.4	10.7	292.	26.9	41.4	7.6	323.	28.7
18090.	38.5	14.8	217.	26.8	40.3	1.2	222.	27.1	40.3	10.7	292.	26.9	41.3	7.6	323.	28.7
18105.	38.4	14.8	217.	26.8	40.2	1.2	221.	27.1	40.2	10.7	292.	26.9	41.2	7.5	323.	28.7
18120.	38.3	14.9	217.	26.8	40.1	1.2	223.	27.1	40.1	10.7	292.	26.9	41.1	7.6	323.	28.7
18135.	38.2	14.9	217.	26.8	40.0	1.2	223.	27.1	40.0	10.6	292.	26.9	41.0	7.5	323.	28.7
18150.	38.1	14.9	217.	26.8	39.9	1.2	224.	27.1	39.9	10.7	292.	26.9	40.9	7.5	323.	28.7
18165.	38.0	15.0	217.	26.8	39.8	1.2	224.	27.1	39.9	10.6	292.	26.9	40.9	7.5	324.	28.7
18180.	37.9	15.0	217.	26.8	39.7	1.2	226.	27.1	39.8	10.7	292.	26.9	40.8	7.5	323.	28.7
18195.	37.8	15.0	216.	26.8	39.7	1.2	226.	27.1	39.7	10.6	292.	26.9	40.7	7.5	324.	28.7
18210.	37.7	15.1	216.	26.8	39.6	1.3	227.	27.1	39.6	10.7	292.	26.9	40.6	7.5	324.	28.7
18225.	37.6	15.1	216.	26.8	39.5	1.3	228.	27.1	39.5	10.6	292.	26.9	40.5	7.5	324.	28.7
18240.	37.5	15.1	216.	26.8	39.4	1.3	229.	27.1	39.4	10.7	292.	26.9	40.4	7.6	324.	28.7
18255.	37.4	15.2	216.	26.8	39.3	1.3	229.	27.1	39.3	10.6	292.	26.9	40.3	7.5	324.	28.7
18270.	37.3	15.2	216.	26.8	39.2	1.3	230.	27.1	39.2	10.7	292.	26.9	40.2	7.6	324.	28.7
18285.	37.2	15.2	216.	26.8	39.1	1.3	231.	27.1	39.1	10.7	292.	26.9	40.2	7.6	324.	28.7
18300.	37.1	15.3	216.	26.8	39.0	1.3	232.	27.1	39.0	10.7	292.	26.9	40.1	7.6	324.	28.7
18315.	37.0	15.3	216.	26.8	39.0	1.3	232.	27.1	38.9	10.7	292.	26.9	40.0	7.6	324.	28.7
18330.	36.9	15.3	216.	26.8	38.9	1.4	233.	27.1	38.8	10.7	291.	26.9	39.9	7.6	323.	28.7
18345.	36.8	15.4	216.	26.8	38.8	1.4	233.	27.1	38.7	10.7	291.	26.9	39.8	7.6	323.	28.7
18360.	36.7	15.4	216.	26.8	38.7	1.4	234.	27.1	38.6	10.7	291.	26.9	39.7	7.6	323.	28.7
18375.	36.6	15.4	216.	26.8	38.6	1.4	235.	27.1	38.5	10.7	291.	26.9	39.6	7.6	323.	28.7
18390.	36.5	15.5	216.	26.8	38.5	1.4	236.	27.1	38.4	10.7	291.	26.9	39.6	7.6	323.	28.7
18405.	36.4	15.5	215.	26.8	38.4	1.4	236.	27.1	38.3	10.7	291.	26.9	39.5	7.6	323.	28.7
18420.	36.4	15.5	215.	26.8	38.3	1.5	237.	27.1	38.2	10.7	291.	26.9	39.4	7.6	323.	28.7
18435.	36.3	15.6	215.	26.8	38.2	1.5	237.	27.1	38.1	10.7	291.	26.9	39.3	7.6	323.	28.7
18450.	36.2	15.6	215.	26.8	38.1	1.5	238.	27.1	38.0	10.7	291.	26.9	39.2	7.6	323.	28.7
18465.	36.1	15.6	215.	26.9	38.1	1.5	239.	27.1	37.9	10.7	291.	26.9	39.1	7.6	323.	28.7
18480.	36.0	15.7	215.	26.8	37.9	1.6	239.	27.1	37.9	10.7	291.	26.9	39.0	7.6	323.	28.7
18495.	35.8	15.7	215.	26.8	37.9	1.6	240.	27.1	37.8	10.7	291.	26.9	38.9	7.6	323.	28.7
18510.	35.7	15.7	215.	26.8	37.8	1.6	241.	27.1	37.7	10.8	291.	26.9	38.8	7.7	323.	28.7
18525.	35.6	15.8	215.	26.8	37.7	1.6	241.	27.1	37.6	10.8	291.	26.9	38.8	7.7	323.	28.7
18540.	35.5	15.8	215.	26.8	37.6	1.6	242.	27.1	37.5	10.8	291.	26.9	38.7	7.7	323.	28.7
18555.	35.4	15.8	215.	26.8	37.5	1.7	242.	27.1	37.4	10.8	291.	26.9	38.6	7.7	323.	28.7
18570.	35.3	15.9	215.	26.8	37.4	1.7	243.	27.1	37.3	10.8	291.	26.9	38.5	7.7	323.	28.7
18585.	35.2	15.9	215.	26.8	37.3	1.7	243.	27.1	37.2	10.8	291.	26.9	38.4	7.7	323.	28.7
18600.	35.1	15.9	215.	26.8	37.2	1.7	244.	27.1	37.1	10.8	291.	26.9	38.3	7.7	323.	28.7
18615.	35.0	16.0	215.	26.8	37.1	1.7	244.	27.1	37.0	10.8	291.	26.9	38.2	7.7	323.	28.7
18630.	34.9	16.0	215.	26.8	37.0	1.8	245.	27.1	36.9	10.8	291.	26.9	38.1	7.7	323.	28.7
18645.	34.8	16.0	215.	26.8	37.0	1.8	245.	27.1	36.8	10.8	291.	26.9	38.0	7.7	323.	28.7
18660.	34.7	16.1	215.	26.8	36.8	1.8	246.	27.1	36.7	10.9	291.	26.9	37.9	7.8	323.	28.7
18675.	34.6	16.1	215.	26.8	36.8	1.8	247.	27.1	36.6	10.9	291.	26.9	37.9	7.8	323.	28.7
18690.	34.5	16.2	215.	26.8	36.7	1.9	247.	27.1	36.5	10.9	291.	26.9	37.8	7.8	323.	28.7
18705.	34.4	16.2	215.	26.8	36.6	1.9	248.	27.1	36.4	10.9	291.	26.9	37.7	7.8	323.	28.7
18720.	34.3	16.2	215.	26.8	36.5	1.9	248.	27.1	36.3	10.9	291.	26.9	37.6	7.8	323.	28.7
18735.	34.2	16.3	215.	26.8	36.4	1.9	249.	27.1	36.2	10.9	291.	26.9	37.5	7.8	323.	28.7
18750.	34.1	16.3	215.	26.8	36.3	2.0	249.	27.1	36.1	10.9	291.	26.9	37.4	7.8	323.	28.7
18765.	34.0	16.3	214.	26.8	36.2	2.0	249.	27.1	36.0	10.9	291.	26.9	37.3	7.8	323.	28.7
18780.	33.9	16.4	215.	26.8	36.1	2.0	250.	27.1	35.9	11.0	291.	26.9	37.2	7.8	323.	28.8
18795.	33.8	16.4	214.	26.8	36.0	2.1	250.	27.1	35.8	11.0	291.	26.9	37.1	7.8	323.	28.8
18810.	33.7	16.4	214.	26.8	35.9	2.1	251.	27.1	35.7	11.0	291.	26.9	37.0	7.9	323.	28.8
18825.	33.6	16.5	214.	26.8	35.8	2.1	251.	27.1	35.6	11.0	291.	26.9	36.9	7.9	323.	28.8
18840.	33.5	16.5	214.	26.8	35.7	2.2	252.	27.1	35.5	11.0	291.	26.9	36.8	7.9	323.	28.8
18855.	33.4	16.5	214.	26.8	35.6	2.2	252.	27.1	35.4	11.0	291.	26.9	36.7	7.9	323.	28.8
18870.	33.3	16.6	214.	26.8	35.5	2.2	253.	27.1	35.3	11.1	291.	26.9	36.6	7.9	323.	28.8

18885.	33.2	16.6	214.	26.8	35.5	2.3	253.	27.1	35.2	11.1	291.	26.9	36.5	7.9	323.	28.8		
18900.	33.1	16.6	214.	26.8	35.4	2.3	253.	27.1	35.1	11.1	290.	26.9	36.4	7.9	323.	28.8		
18915.	33.0	16.7	214.	26.8	35.3	2.3	254.	27.1	35.0	11.1	290.	26.9	36.3	7.9	323.	28.8		
18930.	32.9	16.7	214.	26.8	35.2	2.4	254.	27.1	34.9	11.1	290.	26.9	36.2	7.9	322.	28.8		
18945.	32.8	16.7	214.	26.8	35.1	2.4	255.	27.1	34.8	11.1	290.	26.9	36.1	7.9	322.	28.8		
18960.	32.7	16.8	214.	26.8	35.0	2.4	255.	27.1	34.7	11.2	290.	26.9	36.0	8.0	322.	28.8		
18975.	32.5	16.8	214.	26.8	34.9	2.5	255.	27.1	34.6	11.2	290.	26.9	35.9	8.0	322.	28.8		
18990.	32.4	16.8	214.	26.8	34.8	2.5	256.	27.1	34.5	11.2	290.	26.9	35.8	8.0	322.	28.8		
19005.	32.3	16.8	214.	26.8	34.7	2.5	256.	27.1	34.4	11.2	290.	26.9	35.7	8.0	322.	28.8		
19020.	32.2	16.9	214.	26.8	34.6	2.6	256.	27.1	34.3	11.2	290.	26.9	35.6	8.0	322.	28.8		
19035.	32.1	16.9	214.	26.8	34.5	2.6	257.	27.1	34.2	11.2	290.	26.9	35.5	8.0	322.	28.8		
19050.	32.0	16.9	214.	26.8	34.4	2.6	257.	27.1	34.1	11.2	290.	26.9	35.4	8.0	322.	28.8		
19065.	31.9	17.0	214.	26.8	34.3	2.7	257.	27.1	33.9	11.2	290.	26.9	35.3	8.0	322.	28.8		
19080.	31.8	17.0	214.	26.8	34.2	2.7	257.	27.1	33.8	11.3	290.	26.9	35.2	8.0	322.	28.8		
19095.	31.7	17.0	214.	26.8	34.1	2.7	258.	27.1	33.7	11.3	290.	26.9	35.1	8.1	322.	28.8		
19110.	31.6	17.1	214.	26.8	34.0	2.8	258.	27.1	33.6	11.3	290.	26.9	35.0	8.1	322.	28.8		
19125.	31.5	17.1	214.	26.8	33.9	2.8	258.	27.1	33.5	11.3	290.	26.9	34.9	8.1	322.	28.8		
19140.	31.4	17.1	214.	26.8	33.8	2.8	259.	27.1	33.4	11.3	290.	26.9	34.8	8.1	322.	28.8		
19155.	31.3	17.2	214.	26.8	33.7	2.9	259.	27.1	33.3	11.3	290.	26.9	34.7	8.1	322.	28.8		
19170.	31.2	17.2	214.	26.8	33.6	2.9	259.	27.1	33.2	11.4	290.	26.9	34.6	8.1	322.	28.8		
19185.	31.1	17.2	214.	26.8	33.5	2.9	259.	27.1	33.1	11.4	290.	26.9	34.5	8.1	322.	28.8		
19200.	30.9	17.3	214.	26.8	33.4	3.0	260.	27.1	33.0	11.4	290.	26.9	34.4	8.1	322.	28.8		
19215.	30.8	17.3	214.	26.8	33.3	3.0	260.	27.1	32.9	11.4	290.	26.9	34.3	8.2	322.	28.8		
19230.	30.7	17.3	214.	26.8	33.2	3.1	260.	27.1	32.8	11.4	290.	26.9	34.2	8.2	322.	28.8		
19245.	30.6	17.3	214.	26.8	33.1	3.1	260.	27.1	32.7	11.4	290.	27.0	34.1	8.2	322.	28.8		
19260.	30.5	17.4	214.	26.8	33.0	3.1	261.	27.1	32.6	11.4	290.	27.0	34.0	8.2	322.	28.8		
19275.	30.4	17.4	214.	26.8	32.9	3.2	261.	27.1	32.5	11.4	290.	27.0	33.9	8.2	322.	28.8		
19290.	30.3	17.4	214.	26.8	32.8	3.2	261.	27.1	32.4	11.5	290.	27.0	33.8	8.2	322.	28.8		
19305.	30.2	17.5	214.	26.8	32.7	3.3	261.	27.1	32.3	11.5	290.	27.0	33.7	8.2	322.	28.8		
19320.	30.1	17.5	214.	26.8	32.6	3.3	262.	27.1	32.2	11.5	290.	27.0	33.6	8.2	322.	28.8		
19335.	30.0	17.5	214.	26.8	32.5	3.3	262.	27.1	32.1	11.5	290.	27.0	33.5	8.2	322.	28.8		
19350.	29.9	17.6	214.	26.8	32.4	3.4	262.	27.1	32.0	11.5	290.	27.0	33.4	8.3	322.	28.8		
19365.	29.8	17.6	214.	26.8	32.3	3.4	262.	27.1	31.9	11.5	290.	27.0	33.3	8.3	322.	28.8		
19380.	29.6	17.6	214.	26.8	32.2	3.5	263.	27.1	31.7	11.5	290.	27.0	33.1	8.3	322.	28.8		
19395.	29.5	17.7	214.	26.8	32.1	3.5	263.	27.1	31.6	11.5	290.	27.0	33.0	8.3	322.	28.8		
19410.	29.4	17.7	213.	26.8	32.0	3.5	263.	27.1	31.5	11.6	290.	27.0	32.9	8.3	322.	28.8		
19425.	29.3	17.7	213.	26.8	31.9	3.6	263.	27.1	31.4	11.6	290.	27.0	32.8	8.3	322.	28.8		
19440.	29.2	17.8	213.	26.8	31.8	3.6	264.	27.1	31.3	11.6	290.	27.0	32.7	8.3	322.	28.8		
19455.	29.1	17.8	213.	26.8	31.7	3.7	264.	27.1	31.2	11.6	290.	27.0	32.6	8.3	322.	28.8		
19470.	29.0	17.8	213.	26.8	31.6	3.7	264.	27.1	31.1	11.6	290.	27.0	32.5	8.3	322.	28.8		
19485.	28.9	17.9	213.	26.8	31.5	3.7	264.	27.1	31.0	11.6	290.	27.0	32.4	8.3	322.	28.8		
19500.	28.8	17.9	213.	26.8	31.3	3.8	264.	27.1	30.9	11.6	290.	27.0	32.3	8.4	322.	28.8		
19515.	28.7	17.9	213.	26.8	31.3	3.8	265.	27.1	30.8	11.6	290.	27.0	32.2	8.4	322.	28.8		
19530.	28.6	18.0	213.	26.8	31.1	3.9	265.	27.1	30.7	11.7	290.	27.0	32.1	8.4	322.	28.8		
19545.	28.4	18.0	213.	26.8	31.1	3.9	265.	27.1	30.6	11.7	290.	27.0	32.0	8.4	322.	28.8		
19560.	28.3	18.0	213.	26.8	30.9	3.9	265.	27.1	30.5	11.7	290.	27.0	31.9	8.4	322.	28.8		
19575.	28.2	18.1	213.	26.8	30.8	4.0	265.	27.1	30.4	11.7	290.	27.0	31.8	8.4	322.	28.8		
19590.	28.1	18.1	213.	26.8	30.7	4.0	266.	27.1	30.2	11.7	290.	27.0	31.6	8.4	322.	28.8		
19605.	28.0	18.1	213.	26.8	30.6	4.1	266.	27.1	30.1	11.7	290.	27.0	31.5	8.4	322.	28.8		
19620.	27.9	18.2	213.	26.8	30.5	4.1	266.	27.1	30.0	11.7	290.	27.0	31.4	8.4	321.	28.8		
19635.	27.8	18.2	213.	26.8	30.4	4.2	266.	27.1	29.9	11.7	290.	27.0	31.3	8.4	321.	28.8		
19650.	27.7	18.2	213.	26.8	30.3	4.2	266.	27.1	29.8	11.7	290.	27.0	31.2	8.4	321.	28.8		
19665.	27.6	18.3	213.	26.8	30.2	4.2	266.	27.1	29.7	11.7	290.	27.0	31.1	8.4	321.	28.8		
19680.	27.4	18.3	213.	26.8	30.1	4.3	267.	27.1	29.6	11.8	290.	27.0	31.0	8.4	321.	28.8		
19695.	27.3	18.3	213.	26.8	30.0	4.3	267.	27.1	29.5	11.8	290.	27.0	30.9	8.4	321.	28.8		
19710.	27.2	18.4	213.	26.8	29.9	4.4	267.	27.1	29.4	11.8	290.	27.0	30.8	8.4	321.	28.8		
19725.	27.1	18.4	213.	26.8	29.8	4.4	267.	27.1	29.3	11.8	290.	27.0	30.7	8.4	321.	28.8		
19740.	27.0	18.4	213.	26.8	29.7	4.4	267.	27.1	29.2	11.8	290.	27.0	30.6	8.4	321.	28.8		
19755.	26.9	18.5	213.	26.8	29.6	4.5	267.	27.1	29.0	11.8	290.	27.0	30.5	8.4	321.	28.8		
19770.	26.8	18.5	213.	26.8	29.5	4.5	268.	27.1	28.9	11.8	290.	27.0	30.3	8.4	321.	28.8		
19785.	26.7	18.5	213.	26.8	29.4	4.6	268.	27.1	28.8	11.8	290.	27.0	30.2	8.4	321.	28.8		
19800.	26.6	18.6	213.	26.8	29.3	4.6	268.	27.1	28.7	11.9	290.	27.0	30.1	8.5	321.	28.8		
19815.	26.4	18.6	213.	26.8	29.2	4.6	268.	27.1	28.6	11.9	290.	27.0	30.0	8.5	321.	28.8		

19830.	26.3	18.6	213.	26.8	29.1	4.7	268.	27.1	28.5	11.9	290.	27.0	29.9	8.5	321.	28.9
19845.	26.2	18.6	213.	26.8	29.0	4.7	268.	27.1	28.4	11.9	290.	27.0	29.8	8.5	321.	28.9
19860.	26.1	18.7	213.	26.8	28.9	4.8	268.	27.1	28.3	11.9	290.	27.0	29.7	8.5	321.	28.8
19875.	26.0	18.7	213.	26.8	28.8	4.8	269.	27.1	28.2	11.9	290.	27.0	29.6	8.5	321.	28.8
19890.	25.9	18.7	213.	26.8	28.6	4.8	269.	27.1	28.1	11.9	290.	27.0	29.5	8.5	321.	28.9
19905.	25.8	18.8	213.	26.8	28.5	4.9	269.	27.1	27.9	11.9	290.	27.0	29.4	8.5	321.	28.9
19920.	25.6	18.8	213.	26.8	28.4	4.9	269.	27.1	27.8	11.9	290.	27.0	29.3	8.5	321.	28.8
19935.	25.5	18.8	213.	26.8	28.3	5.0	269.	27.1	27.7	11.9	290.	27.0	29.2	8.5	321.	28.8
19950.	25.4	18.9	213.	26.8	28.2	5.0	269.	27.1	27.6	12.0	290.	27.0	29.0	8.5	321.	28.8
19965.	25.3	18.9	213.	26.8	28.1	5.0	269.	27.1	27.5	12.0	290.	27.0	28.9	8.5	321.	28.8
19980.	25.2	18.9	213.	26.8	28.0	5.1	269.	27.1	27.4	12.0	290.	27.0	28.8	8.5	321.	28.8
19995.	25.1	19.0	213.	26.8	27.9	5.1	269.	27.1	27.3	12.0	290.	27.0	28.7	8.5	321.	28.8
20010.	25.0	19.0	213.	26.8	27.8	5.2	270.	27.1	27.2	12.0	290.	27.0	28.6	8.5	321.	28.8
20025.	24.9	19.0	213.	26.8	27.7	5.2	270.	27.1	27.1	12.0	290.	27.0	28.5	8.5	321.	28.8
20040.	24.7	19.1	213.	26.8	27.6	5.2	270.	27.1	26.9	12.0	290.	27.0	28.4	8.5	321.	28.8
20055.	24.6	19.1	212.	26.8	27.5	5.3	270.	27.1	26.8	12.0	290.	27.0	28.3	8.5	321.	28.8
20070.	24.5	19.1	212.	26.8	27.4	5.3	270.	27.1	26.7	12.0	290.	27.0	28.2	8.5	321.	28.8
20085.	24.4	19.1	212.	26.8	27.3	5.3	270.	27.1	26.6	12.0	290.	27.0	28.1	8.5	321.	28.8
20100.	24.3	19.2	212.	26.8	27.2	5.4	270.	27.1	26.5	12.0	290.	27.0	28.0	8.6	321.	28.8
20115.	24.2	19.2	212.	26.8	27.1	5.4	270.	27.1	26.4	12.1	290.	27.0	27.9	8.6	321.	28.8
20130.	24.0	19.2	212.	26.8	26.9	5.5	270.	27.1	26.3	12.1	290.	27.0	27.7	8.6	321.	28.8
20145.	23.9	19.3	212.	26.8	26.8	5.5	270.	27.1	26.2	12.1	291.	27.0	27.6	8.6	321.	28.8
20160.	23.8	19.3	212.	26.8	26.7	5.5	270.	27.1	26.0	12.1	291.	27.0	27.5	8.6	321.	28.8
20175.	23.7	19.3	212.	26.8	26.6	5.6	271.	27.1	25.9	12.1	291.	27.0	27.4	8.6	321.	28.8
20190.	23.6	19.4	212.	26.8	26.5	5.6	271.	27.1	25.8	12.1	291.	27.0	27.3	8.6	321.	28.8
20205.	23.5	19.4	212.	26.8	26.4	5.7	271.	27.1	25.7	12.1	291.	27.0	27.2	8.6	321.	28.8
20220.	23.4	19.4	212.	26.8	26.3	5.7	271.	27.1	25.6	12.1	291.	27.0	27.1	8.6	321.	28.8
20235.	23.2	19.4	212.	26.8	26.2	5.7	271.	27.1	25.5	12.1	291.	27.0	27.0	8.6	321.	28.9
20250.	23.1	19.5	212.	26.8	26.1	5.8	271.	27.1	25.4	12.1	291.	27.0	26.9	8.6	321.	28.8
20265.	23.0	19.5	212.	26.8	26.0	5.8	271.	27.1	25.2	12.2	291.	27.0	26.8	8.6	321.	28.9
20280.	22.9	19.5	212.	26.8	25.9	5.8	271.	27.1	25.1	12.2	291.	27.0	26.7	8.6	321.	28.8
20295.	22.8	19.6	212.	26.8	25.8	5.9	271.	27.1	25.0	12.2	291.	27.0	26.6	8.6	321.	28.8
20310.	22.7	19.6	212.	26.8	25.7	5.9	271.	27.1	24.9	12.2	291.	27.0	26.4	8.6	321.	28.8
20325.	22.5	19.6	212.	26.8	25.6	6.0	271.	27.1	24.8	12.2	291.	27.0	26.3	8.6	321.	28.8
20340.	22.4	19.7	212.	26.8	25.4	6.0	271.	27.1	24.7	12.2	291.	27.0	26.2	8.7	321.	28.8
20355.	22.3	19.7	212.	26.8	25.3	6.0	271.	27.1	24.6	12.2	291.	27.0	26.1	8.7	321.	28.9
20370.	22.2	19.7	212.	26.8	25.2	6.1	271.	27.1	24.4	12.2	291.	27.0	26.0	8.7	321.	28.8
20385.	22.1	19.7	212.	26.8	25.1	6.1	272.	27.1	24.3	12.2	291.	27.0	25.9	8.7	321.	28.8
20400.	22.0	19.8	212.	26.8	25.0	6.1	272.	27.1	24.2	12.3	291.	27.0	25.8	8.7	321.	28.8
20415.	21.8	19.8	212.	26.8	24.9	6.2	272.	27.1	24.1	12.3	291.	27.0	25.7	8.7	321.	28.8
20430.	21.7	19.8	212.	26.8	24.8	6.2	272.	27.1	24.0	12.3	291.	27.0	25.6	8.7	321.	28.8
20445.	21.6	19.8	212.	26.8	24.7	6.2	272.	27.1	23.9	12.3	291.	27.0	25.5	8.7	321.	28.8
20460.	21.5	19.9	212.	26.8	24.6	6.3	272.	27.1	23.8	12.3	291.	27.0	25.3	8.7	321.	28.8
20475.	21.4	19.9	212.	26.8	24.5	6.3	272.	27.1	23.7	12.3	291.	27.0	25.2	8.7	321.	28.8
20490.	21.2	19.9	212.	26.8	24.4	6.3	272.	27.1	23.5	12.3	291.	27.0	25.1	8.7	321.	28.9
20505.	21.1	20.0	212.	26.8	24.3	6.4	272.	27.1	23.4	12.3	291.	27.0	25.0	8.7	321.	28.8
20520.	21.0	20.0	212.	26.8	24.1	6.4	272.	27.1	23.3	12.3	291.	27.0	24.9	8.7	321.	28.9
20535.	20.9	20.0	212.	26.8	24.0	6.5	272.	27.1	23.2	12.3	291.	27.0	24.8	8.7	321.	28.8
20550.	20.8	20.0	212.	26.8	23.9	6.5	272.	27.1	23.1	12.4	291.	27.0	24.7	8.8	321.	28.8
20565.	20.7	20.1	212.	26.8	23.8	6.5	272.	27.1	23.1	12.4	291.	27.0	24.6	8.8	321.	28.8
20580.	20.5	20.1	212.	26.8	23.7	6.6	272.	27.1	22.8	12.4	291.	27.0	24.5	8.8	321.	28.8
20595.	20.4	20.1	212.	26.8	23.6	6.6	272.	27.1	22.7	12.4	291.	27.0	24.3	8.8	321.	28.8
20610.	20.3	20.1	212.	26.8	23.5	6.6	272.	27.1	22.6	12.4	291.	27.0	24.2	8.8	321.	28.8
20625.	20.2	20.2	212.	26.8	23.4	6.7	272.	27.1	22.5	12.4	291.	27.0	24.1	8.8	321.	28.8
20640.	20.1	20.2	212.	26.8	23.3	6.7	272.	27.1	22.4	12.4	291.	27.0	24.0	8.8	321.	28.8
20655.	20.0	20.2	212.	26.8	23.2	6.7	273.	27.1	22.3	12.4	291.	27.0	23.9	8.8	321.	28.8
20670.	19.8	20.2	212.	26.8	23.0	6.8	273.	27.1	22.1	12.4	291.	27.0	23.8	8.8	321.	28.8
20685.	19.7	20.3	212.	26.8	22.9	6.8	273.	27.1	22.0	12.5	291.	27.0	23.7	8.8	321.	28.8
20700.	19.6	20.3	212.	26.8	22.8	6.8	273.	27.1	21.9	12.5	291.	27.0	23.6	8.8	321.	28.9
20715.	19.5	20.3	212.	26.8	22.7	6.9	273.	27.1	21.8	12.5	291.	27.0	23.5	8.8	321.	28.8
20730.	19.4	20.3	212.	26.8	22.6	6.9	273.	27.1	21.7	12.5	291.	27.0	23.3	8.8	321.	28.8
20745.	19.2	20.4	212.	26.8	22.5	6.9	273.	27.1	21.6	12.5	291.	27.0	23.2	8.9	321.	28.8
20760.	19.1	20.4	212.	26.8	22.4	7.0	273.	27.1	21.4	12.5	291.	27.0	23.1	8.9	321.	28.8

20775.	19.0	20.4	212.	26.8	22.3	7.0	273.	27.1	21.3	12.5	291.	27.0	23.0	8.9	321.	28.8
20790.	18.9	20.4	212.	26.8	22.1	7.0	273.	27.1	21.2	12.5	291.	27.0	22.9	8.9	321.	28.8
20805.	18.8	20.5	212.	26.8	22.0	7.1	273.	27.1	21.1	12.5	291.	27.0	22.8	8.9	321.	28.8
20820.	18.6	20.5	212.	26.8	21.9	7.1	273.	27.1	21.0	12.5	291.	27.0	22.7	8.9	321.	28.8
20835.	18.5	20.5	212.	26.8	21.8	7.1	273.	27.1	20.9	12.6	291.	27.0	22.5	8.9	320.	28.8
20850.	18.4	20.5	212.	26.8	21.7	7.2	273.	27.1	20.7	12.6	291.	27.0	22.4	8.9	320.	28.8
20865.	18.3	20.6	212.	26.8	21.6	7.2	273.	27.1	20.6	12.6	291.	27.0	22.3	8.9	320.	28.8
20880.	18.2	20.6	212.	26.8	21.5	7.2	273.	27.1	20.5	12.6	291.	27.0	22.2	8.9	320.	28.8
20895.	18.0	20.6	212.	26.8	21.4	7.3	273.	27.1	20.4	12.6	291.	27.0	22.1	8.9	320.	28.8
20910.	17.9	20.6	212.	26.8	21.2	7.3	273.	27.1	20.3	12.6	291.	27.0	22.0	8.9	320.	28.8
20925.	17.8	20.7	212.	26.8	21.1	7.3	273.	27.1	20.2	12.6	291.	27.0	21.9	9.0	320.	28.8
20940.	17.7	20.7	212.	26.8	21.0	7.3	273.	27.1	20.0	12.6	291.	27.0	21.7	9.0	320.	28.8
20955.	17.6	20.7	212.	26.8	20.9	7.4	273.	27.1	19.9	12.6	291.	27.0	21.6	9.0	320.	28.8
20970.	17.4	20.7	212.	26.8	20.8	7.4	273.	27.2	19.8	12.6	291.	27.0	21.5	9.0	320.	28.8
20985.	17.3	20.7	212.	26.8	20.7	7.4	273.	27.1	19.7	12.7	291.	27.0	21.4	9.0	320.	28.8
21000.	17.2	20.8	212.	26.8	20.6	7.5	273.	27.2	19.6	12.7	291.	27.0	21.3	9.0	320.	28.8
21015.	17.1	20.8	212.	26.8	20.5	7.5	273.	27.2	19.4	12.7	292.	27.0	21.2	9.0	320.	28.3
21030.	17.0	20.8	212.	26.8	20.3	7.5	274.	27.2	19.3	12.7	292.	27.0	21.0	9.0	320.	28.8
21045.	16.8	20.8	212.	26.8	20.2	7.6	274.	27.2	19.2	12.7	292.	27.0	20.9	9.0	320.	28.8
21060.	16.7	20.9	212.	26.8	20.1	7.6	274.	27.2	19.1	12.7	292.	27.0	20.8	9.0	320.	28.8
21075.	16.6	20.9	212.	26.8	20.0	7.6	274.	27.2	19.0	12.7	292.	27.0	20.7	9.0	320.	28.8
21090.	16.5	20.9	212.	26.8	19.9	7.7	274.	27.2	18.3	12.7	292.	27.0	20.6	9.0	320.	28.8
21105.	16.4	20.9	212.	26.8	19.8	7.7	274.	27.2	18.7	12.7	292.	27.0	20.5	9.1	320.	28.8
21120.	16.2	21.0	212.	26.8	19.7	7.7	274.	27.2	18.6	12.7	292.	27.0	20.3	9.1	320.	28.8
21135.	15.1	21.0	212.	26.8	19.6	7.7	274.	27.2	18.5	12.7	292.	27.0	20.2	9.1	320.	28.8
21150.	16.0	21.0	212.	26.8	19.4	7.8	274.	27.2	18.4	12.8	292.	27.0	20.1	9.1	320.	28.8
21165.	15.9	21.0	212.	26.8	19.3	7.8	274.	27.2	18.3	12.8	292.	27.0	20.0	9.1	320.	28.8
21180.	15.8	21.0	211.	26.8	19.2	7.8	274.	27.2	18.1	12.8	292.	27.0	19.9	9.1	320.	28.8
21195.	15.6	21.1	211.	26.8	19.1	7.9	274.	27.2	18.0	12.8	292.	27.0	19.9	9.1	320.	28.8
21210.	15.5	21.1	211.	26.8	19.0	7.9	274.	27.2	17.9	12.8	292.	27.0	19.6	9.1	320.	28.8
21225.	15.4	21.1	211.	26.8	18.9	7.9	274.	27.2	17.8	12.8	292.	27.0	19.5	9.1	320.	28.8
21240.	15.3	21.1	211.	26.8	18.8	8.0	274.	27.2	17.7	12.8	292.	27.0	19.4	9.1	320.	28.8
21255.	15.2	21.1	211.	26.8	18.7	8.0	274.	27.2	17.5	12.8	292.	27.0	19.3	9.1	320.	28.8
21270.	15.0	21.2	211.	26.8	18.5	8.0	274.	27.2	17.4	12.8	292.	27.0	19.2	9.1	320.	28.8
21285.	14.9	21.2	211.	26.9	18.4	8.0	274.	27.2	17.3	12.8	292.	27.0	19.1	9.1	320.	26.8
21300.	14.8	21.2	211.	26.8	18.3	8.1	274.	27.2	17.2	12.8	292.	27.0	18.9	9.1	320.	28.8
21315.	14.7	21.2	211.	26.8	18.2	8.1	274.	27.2	17.1	12.8	292.	27.0	18.8	9.2	320.	28.8
21330.	14.5	21.3	211.	26.8	18.1	8.1	274.	27.2	16.9	12.9	292.	27.0	18.7	9.2	320.	28.8
21345.	14.4	21.3	211.	26.8	18.0	8.2	274.	27.2	16.8	12.9	292.	27.0	18.6	9.2	320.	28.9
21360.	14.3	21.3	211.	26.8	17.9	8.2	274.	27.2	16.7	12.9	292.	27.0	18.5	9.2	320.	28.8
21375.	14.2	21.3	211.	26.6	17.7	8.2	274.	27.2	16.6	12.9	292.	27.0	18.3	9.2	320.	28.8
21390.	14.1	21.3	211.	26.8	17.6	8.3	274.	27.2	16.5	12.9	292.	27.0	18.2	9.2	320.	28.8
21405.	13.9	21.4	211.	26.8	17.5	8.3	274.	27.2	16.3	12.9	292.	27.0	18.1	9.2	320.	28.8
21420.	13.8	21.4	211.	26.8	17.4	8.3	274.	27.2	16.2	12.9	292.	27.0	18.0	9.2	320.	28.8
21435.	13.7	21.4	211.	26.8	17.3	8.3	274.	27.2	16.1	12.9	292.	27.0	17.9	9.2	320.	28.8
21450.	13.6	21.4	211.	26.8	17.2	9.4	274.	27.2	16.0	12.9	292.	27.0	17.8	9.2	320.	28.8
21465.	13.4	21.4	211.	26.8	17.1	8.4	274.	27.2	15.9	12.9	292.	27.0	17.6	9.2	320.	28.8
21480.	13.3	21.5	211.	26.8	16.9	8.4	274.	27.2	15.7	12.9	292.	27.0	17.5	9.2	320.	28.8
21495.	13.2	21.5	211.	26.9	16.8	8.5	275.	27.2	15.6	12.9	292.	27.0	17.4	9.2	320.	28.8
21510.	13.1	21.5	211.	26.9	16.7	8.5	275.	27.2	15.5	12.9	292.	27.0	17.3	9.2	320.	28.8
21525.	13.0	21.5	211.	26.9	16.6	8.5	275.	27.2	15.4	13.0	292.	27.0	17.2	9.2	320.	28.8
21540.	12.8	21.5	211.	26.9	16.5	8.5	275.	27.2	15.2	13.0	292.	27.0	17.0	9.2	320.	28.8
21555.	12.7	21.6	211.	26.9	16.4	8.6	275.	27.2	15.1	13.0	292.	27.0	16.9	9.2	320.	28.8
21570.	12.6	21.6	211.	26.9	16.2	8.6	275.	27.2	15.0	13.0	292.	27.0	16.8	9.3	320.	28.8
21585.	12.5	21.6	211.	26.9	16.1	8.6	275.	27.2	14.9	13.0	292.	27.0	16.7	9.3	320.	28.8
21600.	12.3	21.6	211.	26.9	16.0	8.7	275.	27.2	14.8	13.0	292.	27.0	16.6	9.3	320.	28.8

```
EST,TAIR,TEQ,TAN,TAS,TAUX,TAUY= 13.00 29.40 25.68 27.05 27.05     -.139770      .050813
DATA RECORDED ON TAPE,   TTOT=    21600.   CASE NBR=    3   BLOK NBR=    6
RESULTANT VELOCITIES AND DIRS,     K=   1

I=   1  VRES=*********************   3.1    2.5    2.2*********************   2.3    2.5***************
             *********************  100.   112.   100.*********************   86.    87.**************
I=   2  VRES=  22.8   15.9   11.2    7.7    5.9    4.5    2.0    2.4    2.5    4.7    6.7**************
              188.   188.   184.   160.   134.   116.   141.   152.   147.    96.    86.**************
I=   3  VRES=  34.4   31.4   25.7   18.8   13.6   10.2    6.4    3.9    2.4    4.5   10.6   14.0   19.4
              194.   193.   188.   191.   177.   167.   166.   160.   122.    49.    32.   356.   348.
I=   4  VRES=  27.9   26.2   24.3   21.3   18.6   14.7    9.0    4.6     .2    5.6   12.4   22.0   34.2
              203.   199.   194.   197.   194.   186.   187.   194.   238.     3.   356.   343.   336.
I=   5  VRES=  26.7   24.2   21.5   18.4   17.1   14.4   10.0    6.2    3.4    7.7   16.0   27.8   54.0
              206.   199.   197.   198.   204.   201.   237.   225.   281.   336.   332.   326.   322.
I=   6  VRES=  21.5   21.3   19.3   16.5   15.9   14.2   11.7    8.4    6.6    8.9   15.1   21.2   35.0
              211.   205.   203.   199.   206.   210.   223.   241.   276.   314.   314.   299.   265.
I=   7  VRES=  15.6   16.0   15.3   11.5   11.0   11.9   12.6   11.4    9.1    9.8   13.0   16.2   18.4
              218.   215.   221.   217.   219.   221.   235.   251.   273.   295.   302.   293.   274.
I=   8  VRES=  12.1   12.5   14.8   11.4    7.4    7.4   10.5   11.9   10.8   10.5   11.6   13.0   13.3
              224.   222.   236.   242.   246.   252.   255.   260.   271.   286.   294.   292.   286.
I=   9  VRES=   9.4    9.9   13.3   13.0    7.1    5.5    8.3   11.3   11.3   10.5   10.6   11.2   11.6
              222.   224.   237.   243.   243.   270.   280.   275.   275.   284.   292.   295.   295.
I=  10  VRES=   7.6    7.5    9.3   11.2    6.6    4.6    6.7    9.7   11.0   10.4   10.0   10.2   10.8
              212.   221.   236.   248.   249.   277.   285.   277.   277.   283.   291.   298.   300.
I=  11  VRES=   5.7    4.9    5.1    9.6    5.9    4.1    5.9    8.7   10.9   10.5    9.6    9.3    9.6
              195.   203.   214.   252.   256.   275.   289.   280.   278.   286.   295.   300.   301.
I=  12  VRES=*********************   8.7    2.6    1.8    4.9    9.0   12.2   10.6    8.7    8.5    8.2
             *********************  275.   281.   264.   286.   288.   286.   291.   297.   299.   301.
I=  13  VRES=**************    2.8    4.8**************    2.1   11.3   13.6    9.6    7.3    7.4    7.1
             **************   336.   302.**************   277.   291.   303.   305.   298.   297.   303.
I=  14  VRES=   6.1   16.2   14.4*****************************   12.6    9.3    5.8    5.7    5.8    5.8
              321.   343.     9.*****************************   311.   320.   313.   308.   307.   312.
I=  15  VRES=   8.3    6.0*****************************************    1.6    3.1    3.4    3.1    3.0
              314.   308.*****************************************   321.   335.   331.   327.   327.
RESULTANT VELOCITIES AND DIRS,     K=   2

I=   1  VRES=*********************   2.9    2.4    2.1*********************   2.2    2.4***************
             *********************  101.   112.   100.*********************   87.    87.**************
I=   2  VRES=  21.4   14.9   10.6    7.4    5.7    4.4    2.0    2.4    2.5    4.6    6.5**************
              187.   167.   183.   159.   133.   115.   139.   150.   146.    97.    87.**************
I=   3  VRES=  32.4   29.6   24.2   17.7   13.0    9.8    6.2    3.9    2.5    4.3    9.9   13.0   18.1
              193.   192.   187.   190.   176.   165.   163.   156.   121.    51.    33.   357.   348.
I=   4  VRES=  26.1   24.6   22.9   20.1   17.6   13.9    8.6    4.4     .3    5.2   11.5   20.4   31.9
              202.   198.   193.   195.   192.   184.   184.   188.   136.     6.   358.   344.   336.
I=   5  VRES=  25.1   22.7   20.2   17.3   16.1   13.5    9.4    5.7    2.6    7.0   14.7   25.8   50.3
              205.   198.   196.   196.   202.   199.   204.   221.   279.   338.   333.   326.   322.
I=   6  VRES=  20.2   20.0   18.1   15.5   14.9   13.2   10.8    7.6    5.7    8.0   13.8   19.4   32.4
              210.   204.   202.   198.   204.   209.   221.   238.   275.   315.   315.   299.   265.
I=   7  VRES=  14.5   14.9   14.3   10.7   10.3   11.1   11.6    9.4    8.1    8.8   11.8   14.8   16.8
              217.   214.   220.   216.   217.   219.   233.   249.   271.   295.   303.   294.   273.
I=   8  VRES=  11.3   11.6   13.7   10.5    6.8    6.7    9.6   10.8    9.7    9.4   10.4   11.8   12.1
              223.   221.   235.   241.   244.   251.   254.   259.   270.   286.   294.   293.   286.
I=   9  VRES=   8.7    9.2   12.3   12.0    6.5    4.9    7.5   10.2   10.1    9.5    9.6   10.1   10.5
              221.   223.   236.   242.   242.   269.   280.   274.   275.   283.   292.   295.   296.
I=  10  VRES=   7.1    7.0    8.6   10.3    6.0    4.1    6.0    8.7    9.9    9.3    9.0    9.2    9.7
              210.   219.   235.   247.   247.   275.   284.   277.   276.   282.   291.   299.   300.
I=  11  VRES=   5.3    4.6    4.7    8.8    5.3    3.6    5.3    7.6    9.9    9.4    8.6    8.4    8.6
              194.   202.   213.   251.   255.   274.   289.   280.   277.   286.   296.   300.   302.
I=  12  VRES=*********************   7.9    2.4    1.6    4.4    8.2   11.2    9.5    7.8    7.6    7.3
             *********************  275.   280.   262.   285.   283.   286.   291.   298.   299.   301.
I=  13  VRES=**************    2.5    4.4**************    1.9   10.4   12.5    8.7    6.6    6.6    6.4
             **************   337.   302.**************   276.   291.   304.   306.   298.   298.   304.
I=  14  VRES=   5.6   15.1   13.7*****************************   12.0    8.5    5.2    5.2    5.2    5.2
              322.   343.    10.*****************************   312.   320.   314.   309.   308.   313.
I=  15  VRES=   7.9    5.5*****************************************    1.4    2.8    3.0    2.7    2.7
              315.   308.*****************************************   323.   336.   332.   328.   329.
```

```
RESULTANT VELOCITIES AND DIRS,    K=   3

I=  1  VRES=******************************       2.4       2.0       1.7******************************       1.8       2.0********************
            ******************************      101.      112.      100.******************************       88.       88.********************
I=  2  VRES=      17.2      12.0       8.5       6.0       4.6       3.6       1.7       2.0       2.0       3.8       5.2********************
                  186.      186.      182.      156.      133.      115.      138.      149.      145.       97.       87.********************
I=  3  VRES=      26.1      23.8      19.5      14.3      10.5       9.0       5.1       3.3       2.1       3.5       8.0      10.3      14.4
                  192.      191.      186.      189.      175.      164.      161.      154.      120.       52.       34.      357.      349.
I=  4  VRES=      20.9      19.8      18.4      16.2      14.3      11.3       7.0       3.6        .4       4.1       9.2      15.2      25.5
                  201.      197.      191.      194.      191.      183.      182.      185.      125.        8.      358.      344.      337.
I=  5  VRES=      20.1      18.3      16.2      13.9      12.9      10.9       7.5       4.5       2.0       5.5      11.7      20.4      40.3
                  204.      197.      195.      195.      201.      198.      202.      218.      277.      339.      334.      326.      322.
I=  6  VRES=      16.1      16.0      14.5      12.4      11.9      10.6       9.6       5.0       4.4       6.2      10.9      15.4      25.8
                  209.      203.      201.      196.      203.      207.      219.      236.      274.      316.      315.      299.      264.
I=  7  VRES=      11.6      11.9      11.4       8.5       8.2       8.8       9.2       7.4       6.3       6.8       9.3      11.6      13.2
                  216.      213.      219.      214.      216.      218.      232.      248.      270.      295.      303.      293.      273.
I=  8  VRES=       9.0       9.3      10.9       8.3       5.4       5.3       7.5       8.5       7.6       7.3       8.2       9.2       9.4
                  222.      220.      234.      240.      244.      250.      253.      258.      269.      285.      294.      292.      286.
I=  9  VRES=       6.9       7.3       9.8       9.5       5.1       3.8       5.9       8.0       7.9       7.4       7.5       7.9       8.2
                  220.      222.      235.      241.      241.      269.      280.      274.      274.      283.      292.      295.      296.
I= 10  VRES=       5.6       5.5       6.8       8.1       4.7       3.2       4.7       6.8       7.7       7.3       7.0       7.2       7.6
                  209.      219.      234.      246.      246.      276.      284.      277.      276.      282.      291.      299.      300.
I= 11  VRES=       4.3       3.7       3.8       7.0       4.2       2.8       4.1       6.1       7.7       7.3       6.7       6.5       6.7
                  193.      201.      212.      251.      254.      274.      289.      280.      277.      286.      296.      300.      302.
I= 12  VRES=******************************       6.2       1.8       1.3       3.4       6.5       8.8       7.5       6.1       5.9       5.7
            ******************************      274.      279.      261.      285.      288.      286.      291.      298.      299.      301.
I= 13  VRES=********************       2.0       3.5********************       1.5       8.2       9.9       6.8       5.2       5.2       5.0
            ********************      338.      302.********************      275.      291.      304.      306.      298.      298.      304.
I= 14  VRES=       4.4      12.0      11.8****************************************      10.5       6.7       4.1       4.0       4.0       4.0
                  322.      343.       13.****************************************      312.      321.      314.      309.      309.      314.
I= 15  VRES=       6.9       4.4************************************************************       1.1       2.2       2.4       2.1       2.1
                  315.      308.************************************************************      325.      337.      333.      329.      329.
RESULTANT VELOCITIES AND DIRS,    K=   4

I=  1  VRES=******************************       1.4       1.2       1.0******************************       1.1       1.2********************
            ******************************      101.      112.      100.******************************       98.       88.********************
I=  2  VRES=      10.0       7.0       5.0       3.5       2.7       2.2       1.0       1.2       1.2       2.2       3.1********************
                  185.      185.      181.      157.      132.      115.      137.      148.      145.       97.       87.********************
I=  3  VRES=      15.4      14.0      11.5       8.4       6.2       4.7       3.0       2.0       1.3       2.1       4.7       6.0       8.4
                  191.      190.      185.      188.      173.      163.      160.      152.      119.       53.       35.      357.      349.
I=  4  VRES=      12.3      11.6      10.9       9.6       8.5       6.7       4.2       2.2        .3       2.4       5.3       9.4      14.9
                  200.      196.      190.      193.      190.      181.      180.      182.      123.        9.      359.      344.      337.
I=  5  VRES=      11.8      10.7       9.5       8.2       7.6       6.4       4.4       2.6       1.1       3.1       6.7      11.9      23.7
                  203.      196.      194.      194.      199.      196.      200.      215.      275.      340.      334.      326.      322.
I=  6  VRES=       9.4       9.4       8.5       7.3       7.0       6.2       5.0       3.5       2.5       3.6       6.3       8.9      15.1
                  208.      202.      199.      195.      202.      205.      218.      234.      272.      316.      315.      299.      264.
I=  7  VRES=       6.7       6.9       6.6       4.9       4.7       5.1       5.4       4.3       3.6       3.9       5.3       6.7       7.6
                  215.      212.      218.      213.      215.      216.      230.      245.      269.      295.      303.      293.      272.
I=  8  VRES=       5.2       5.4       6.3       4.8       3.1       3.0       4.3       4.9       4.4       4.2       4.7       5.3       5.4
                  221.      219.      233.      239.      243.      259.      252.      257.      268.      285.      294.      292.      286.
I=  9  VRES=       4.0       4.2       5.7       5.5       3.0       2.2       3.4       4.6       4.5       4.2       4.3       4.5       4.7
                  219.      221.      235.      241.      240.      269.      280.      273.      273.      283.      291.      295.      296.
I= 10  VRES=       3.3       3.2       3.9       4.7       2.7       1.8       2.7       3.9       4.4       4.2       4.0       4.1       4.4
                  208.      218.      233.      246.      245.      276.      284.      276.      275.      281.      291.      299.      301.
I= 11  VRES=       2.5       2.2       2.2       4.0       2.4       1.6       2.4       3.5       4.4       4.2       3.8       3.7       3.9
                  193.      200.      211.      259.      254.      274.      289.      277.      277.      286.      296.      300.      302.
I= 12  VRES=******************************       3.6       1.0        .7       2.0       3.7       5.0       4.3       3.5       3.4       3.3
            ******************************      274.      279.      260.      285.      288.      285.      291.      298.      299.      302.
I= 13  VRES=********************       1.2       2.0********************        .8       4.7       5.7       3.9       2.9       3.0       2.8
            ********************      339.      302.********************      275.      291.      304.      306.      298.      298.      305.
I= 14  VRES=       2.6       7.0       8.6****************************************       8.1       3.9       2.3       2.3       2.3       2.3
                  322.      343.       20.****************************************      313.      321.      315.      310.      309.      314.
I= 15  VRES=       5.2       2.5************************************************************        .6       1.3       1.4       1.2       1.2
                  315.      308.************************************************************      326.      337.      333.      330.      330.
```

W VELOCITIES, CM/SEC, K= 1

I													
1	*******	*******	*******	-.0039	-.0038	-.0038	*******	*******	*******	-.0037	-.0037	*******	*******
2	-.0044	-.0043	-.0042	-.0041	-.0041	-.0040	-.0040	-.0040	-.0039	-.0039	-.0040	*******	*******
3	-.0045	-.0045	-.0043	-.0042	-.0042	-.0041	-.0040	-.0040	-.0040	-.0040	-.0042	-.0043	-.0045
4	-.0044	-.0044	-.0043	-.0043	-.0041	-.0041	-.0041	-.0040	-.0040	-.0040	-.0042	-.0044	-.0047
5	-.0044	-.0043	-.0043	-.0041	-.0042	-.0041	-.0040	-.0041	-.0040	-.0041	-.0042	-.0044	-.0051
6	-.0043	-.0043	-.0042	-.0043	-.0041	-.0041	-.0041	-.0040	-.0041	-.0040	-.0042	-.0043	-.0045
7	-.0043	-.0043	-.0043	-.0041	-.0042	-.0041	-.0041	-.0041	-.0040	-.0041	-.0041	-.0042	-.0042
8	-.0042	-.0043	-.0043	-.0042	-.0042	-.0041	-.0041	-.0040	-.0042	-.0040	-.0041	-.0041	-.0041
9	-.0042	-.0042	-.0043	-.0043	-.0041	-.0041	-.0041	-.0042	-.0040	-.0042	-.0041	-.0041	-.0041
10	-.0042	-.0042	-.0042	-.0042	-.0041	-.0041	-.0041	-.0041	-.0042	-.0041	-.0041	-.0041	-.0041
11	-.0041	-.0041	-.0041	-.0042	-.0041	-.0041	-.0041	-.0041	-.0041	-.0041	-.0041	-.0041	-.0041
12	*******	*******	*******	-.0042	-.0041	-.0041	-.0041	-.0042	-.0042	-.0041	-.0041	-.0041	-.0041
13	*******	*******	-.0044	-.0043	*******	*******	-.0041	-.0042	-.0042	-.0042	-.0041	-.0041	-.0041
14	-.0045	-.0049	-.0049	*******	*******	*******	*******	-.0042	-.0042	-.0041	-.0041	-.0041	-.0041
15	-.0045	-.0045	*******	*******	*******	*******	*******	*******	-.0041	-.0041	-.0041	-.0041	-.0041

W VELOCITIES, CM/SEC, K= 2

I													
1	*******	*******	*******	-.0026	-.0025	-.0024	*******	*******	*******	-.0020	-.0020	*******	*******
2	-.0010	-.0016	-.0034	-.0053	-.0044	-.0042	-.0034	-.0031	-.0026	-.0027	-.0037	*******	*******
3	-.0031	-.0015	-.0020	-.0042	-.0049	-.0036	-.0039	-.0039	-.0032	-.0030	-.0041	-.0037	-.0024
4	-.0076	-.0017	.0007	-.0018	-.0040	-.0021	-.0028	-.0030	-.0025	-.0025	-.0035	-.0027	-.0043
5	-.0070	-.0033	-.0016	-.0027	-.0036	-.0022	-.0026	-.0024	-.0016	-.0016	-.0023	-.0017	-.0062
6	-.0046	-.0042	-.0029	-.0031	-.0038	-.0036	-.0030	-.0021	-.0018	-.0012	-.0019	-.0027	-.0022
7	-.0050	-.0047	-.0050	-.0044	-.0043	-.0039	-.0034	-.0025	-.0021	-.0017	-.0021	-.0032	-.0038
8	-.0034	-.0041	-.0053	-.0044	-.0036	-.0035	-.0035	-.0032	-.0024	-.0021	-.0022	-.0026	-.0028
9	-.0020	-.0032	-.0037	-.0028	-.0019	-.0023	-.0034	-.0035	-.0025	-.0024	-.0021	-.0024	-.0024
10	-.0021	-.0026	-.0024	-.0025	-.0020	-.0023	-.0034	-.0038	-.0032	-.0025	-.0024	-.0024	-.0027
11	-.0022	-.0021	-.0018	-.0029	-.0027	-.0026	-.0036	-.0044	-.0039	-.0030	-.0030	-.0029	-.0036
12	*******	*******	*******	-.0028	-.0023	-.0022	-.0028	-.0038	-.0044	-.0042	-.0037	-.0034	-.0037
13	*******	*******	-.0025	-.0026	*******	*******	-.0023	-.0020	-.0042	-.0043	-.0033	-.0030	-.0031
14	-.0036	-.0037	-.0079	*******	*******	*******	*******	.0056	-.0033	-.0032	-.0028	-.0026	-.0027
15	.0005	-.0025	*******	*******	*******	*******	*******	*******	-.0024	-.0023	-.0020	-.0019	-.0021

W VELOCITIES, CM/SEC, K= 3

I													
1	*******	*******	*******	-.0014	-.0013	-.0012	*******	*******	*******	-.0005	-.0006	*******	*******
2	.0014	.0005	-.0025	-.0056	-.0043	-.0039	-.0026	-.0021	-.0014	-.0016	-.0023	*******	*******
3	-.0024	.0006	-.0003	-.0036	-.0047	-.0028	-.0034	-.0034	-.0023	-.0020	-.0036	-.0028	-.0005
4	-.0090	.0004	.0041	-.0004	-.0036	-.0006	-.0016	-.0020	-.0012	-.0012	-.0026	-.0013	-.0034
5	-.0081	-.0021	.0005	-.0014	-.0029	-.0006	-.0013	-.0010	.0002	.0003	-.0007	.0004	-.0060
6	-.0045	-.0037	-.0017	-.0019	-.0031	-.0029	-.0018	-.0005	.0001	.0009	.0001	-.0013	-.0003
7	-.0050	-.0046	-.0051	-.0042	-.0040	-.0034	-.0026	-.0011	-.0005	.0001	-.0005	-.0022	-.0031
8	-.0025	-.0035	-.0055	-.0040	-.0028	-.0028	-.0029	-.0024	-.0010	-.0005	-.0006	-.0013	-.0015
9	-.0002	-.0021	-.0029	-.0014	-.0001	-.0008	-.0026	-.0027	-.0012	-.0009	-.0006	-.0009	-.0010
10	-.0004	-.0011	-.0009	-.0011	-.0003	-.0008	-.0026	-.0033	-.0023	-.0012	-.0009	-.0010	-.0015
11	-.0007	-.0004	.0000	-.0018	-.0014	-.0013	-.0028	-.0043	-.0034	-.0018	-.0020	-.0018	-.0028
12	*******	*******	*******	-.0016	-.0008	-.0009	-.0016	-.0033	-.0042	-.0038	-.0030	-.0026	-.0031
13	*******	*******	-.0009	-.0013	*******	*******	-.0008	-.0003	-.0038	-.0040	-.0025	-.0020	-.0020
14	-.0025	-.0024	-.0086	*******	*******	*******	*******	.0115	-.0024	-.0022	-.0017	-.0014	-.0015
15	.0036	-.0009	*******	*******	*******	*******	*******	*******	-.0011	-.0009	-.0005	-.0003	-.0006

W VELOCITIES, CM/SEC, K= 4

I													
1	*******	*******	*******	-.0006	-.0004	-.0004	*******	*******	*******	.0002	.0002	*******	*******
2	.0017	.0011	-.0015	-.0041	-.0031	-.0027	-.0016	-.0011	-.0005	-.0007	-.0013	*******	*******
3	-.0018	.0012	.0004	-.0023	-.0032	-.0017	-.0023	-.0022	-.0013	-.0010	-.0024	-.0016	.0005
4	-.0071	.0011	.0043	.0001	-.0025	.0002	-.0007	-.0010	-.0003	-.0003	-.0016	-.0004	-.0020
5	-.0064	-.0010	.0013	-.0005	-.0017	.0001	-.0005	-.0002	.0009	.0010	.0001	.0011	-.0039
6	-.0032	-.0025	-.0009	-.0009	-.0019	-.0017	-.0008	.0003	.0008	.0015	.0008	-.0004	.0008
7	-.0036	-.0032	-.0037	-.0029	-.0028	-.0023	-.0015	-.0002	.0002	.0009	.0003	-.0011	-.0020
8	-.0014	-.0023	-.0040	-.0027	-.0017	-.0017	-.0019	-.0014	-.0001	.0003	.0002	-.0004	-.0005
9	.0005	-.0011	-.0018	-.0005	.0007	.0001	-.0015	-.0017	-.0004	.0000	.0002	-.0001	-.0001
10	.0004	-.0002	-.0001	-.0002	.0005	.0000	-.0015	-.0022	-.0012	-.0003	-.0001	-.0002	-.0006
11	.0001	.0003	.0007	-.0008	-.0005	-.0004	-.0018	-.0030	-.0022	-.0008	-.0010	-.0009	-.0017
12	*******	*******	*******	-.0006	-.0000	.0000	-.0007	-.0021	-.0030	-.0026	-.0019	-.0015	-.0019
13	*******	*******	-.0000	-.0004	*******	*******	-.0000	.0005	-.0025	-.0028	-.0014	-.0010	-.0010
14	-.0013	-.0011	-.0048	*******	*******	*******	*******	.0100	-.0014	-.0012	-.0007	-.0004	-.0006
15	.0033	-.0000	*******	*******	*******	*******	*******	*******	-.0002	-.0001	.0002	.0004	.0001

SURFACE ELEVATIONS, ETA, CM

I= 1	ETA=	****	****	****	14.9	15.3	15.5	****	****	****	16.4	16.4	****	****
I= 2	ETA=	12.1	12.6	13.2	13.6	14.2	14.5	14.6	14.8	15.1	15.4	15.3	****	****
I= 3	ETA=	12.1	12.6	13.2	13.6	13.9	14.3	14.5	14.6	14.8	14.8	14.3	13.5	12.6
I= 4	ETA=	12.1	12.7	13.2	13.6	13.8	14.2	14.4	14.5	14.6	14.6	14.1	13.5	12.8
I= 5	ETA=	12.2	12.7	13.2	13.6	13.9	14.2	14.4	14.5	14.6	14.5	14.2	13.8	13.3
I= 6	ETA=	12.3	12.8	13.2	13.6	13.9	14.2	14.4	14.6	14.7	14.6	14.4	14.2	14.1
I= 7	ETA=	12.5	12.9	13.3	13.6	14.0	14.3	14.6	14.7	14.8	14.8	14.6	14.5	14.4
I= 8	ETA=	12.8	13.2	13.7	14.0	14.4	14.6	14.8	14.9	14.9	14.9	14.9	14.8	14.7
I= 9	ETA=	13.1	13.5	14.3	14.6	15.0	15.2	15.1	15.1	15.1	15.1	15.0	15.0	14.9
I= 10	ETA=	13.5	14.0	14.9	15.1	15.3	15.4	15.3	15.3	15.3	15.3	15.2	15.1	15.0
I= 11	ETA=	13.7	14.4	15.3	15.5	15.6	15.7	15.6	15.5	15.5	15.4	15.4	15.3	15.2
I= 12	ETA=	****	****	****	16.0	15.9	16.0	16.0	15.8	15.7	15.6	15.6	15.4	15.4
I= 13	ETA=	****	****	17.0	16.6	****	****	16.5	16.5	16.2	15.9	15.8	15.7	15.5
I= 14	ETA=	19.2	18.6	17.1	****	****	****	****	17.0	16.6	16.3	16.0	15.9	15.7
I= 15	ETA=	19.8	19.2	****	****	****	****	****	****	16.7	16.4	16.2	16.0	15.9

TEMPERATURES, DEG C. K= 1

I= 1	TEMP=	****	****	****	27.14	27.17	27.18	****	****	****	27.08	27.13	****	****
I= 2	TEMP=	27.40	27.46	27.37	27.24	27.19	27.22	27.34	27.32	27.22	27.09	27.11	****	****
I= 3	TEMP=	26.99	27.12	27.20	27.29	27.30	27.19	27.22	27.23	27.19	27.11	27.06	27.15	27.19
I= 4	TEMP=	26.94	26.85	26.89	26.94	27.09	27.05	27.08	27.12	27.16	27.15	27.09	27.08	27.07
I= 5	TEMP=	26.93	26.93	26.90	26.90	26.93	26.94	26.98	27.00	27.05	27.09	27.07	27.03	26.99
I= 6	TEMP=	26.95	26.86	26.86	26.85	26.85	26.88	26.93	26.98	27.02	27.04	27.02	26.97	26.93
I= 7	TEMP=	26.86	26.84	26.85	26.82	26.75	26.78	26.86	26.95	26.97	26.99	26.99	26.97	26.96
I= 8	TEMP=	26.89	26.88	26.91	27.00	26.97	26.92	26.89	26.98	27.01	27.02	27.00	26.99	26.97
I= 9	TEMP=	26.90	26.89	26.89	27.04	27.14	27.18	27.19	27.20	27.18	27.14	27.07	27.01	26.97
I= 10	TEMP=	26.89	26.86	26.86	27.03	27.27	27.42	27.52	27.62	27.58	27.42	27.24	27.08	26.98
I= 11	TEMP=	26.82	26.77	26.76	27.07	27.48	27.83	28.05	28.24	28.16	27.81	27.45	27.15	27.00
I= 12	TEMP=	****	****	****	27.17	27.64	28.07	28.41	28.87	28.68	28.10	27.57	27.18	26.99
I= 13	TEMP=	****	****	27.02	27.11	****	****	28.26	29.33	29.04	28.26	27.55	27.16	26.99
I= 14	TEMP=	27.26	27.16	28.11	****	****	****	****	29.51	28.84	27.99	27.37	27.04	26.93
I= 15	TEMP=	27.03	27.12	****	****	****	****	****	****	27.55	27.30	26.97	26.78	26.78

TEMPERATURES, DEG C. K= 2

I= 1	TEMP=	****	****	****	27.14	27.17	27.18	****	****	****	27.08	27.13	****	****
I= 2	TEMP=	27.40	27.47	27.37	27.24	27.19	27.22	27.34	27.32	27.22	27.09	27.11	****	****
I= 3	TEMP=	26.99	27.12	27.20	27.29	27.31	27.19	27.22	27.23	27.19	27.10	27.06	27.15	27.19
I= 4	TEMP=	26.94	26.85	26.89	26.94	27.09	27.05	27.08	27.12	27.16	27.15	27.09	27.08	27.07
I= 5	TEMP=	26.93	26.93	26.90	26.90	26.93	26.94	26.97	27.00	27.05	27.09	27.07	27.03	26.99
I= 6	TEMP=	26.95	26.86	26.86	26.85	26.85	26.88	26.93	26.98	27.02	27.04	27.01	26.97	26.93
I= 7	TEMP=	26.86	26.84	26.84	26.82	26.74	26.78	26.86	26.95	26.97	26.99	26.99	26.97	26.96
I= 8	TEMP=	26.89	26.88	26.91	27.00	26.97	26.92	26.89	26.98	27.01	27.02	27.00	26.99	26.97
I= 9	TEMP=	26.90	26.89	26.89	27.04	27.14	27.18	27.19	27.19	27.18	27.13	27.07	27.01	26.97
I= 10	TEMP=	26.89	26.86	26.86	27.03	27.27	27.42	27.52	27.62	27.57	27.41	27.23	27.07	26.98
I= 11	TEMP=	26.82	26.77	26.76	27.07	27.48	27.83	28.05	28.23	28.15	27.81	27.45	27.14	27.00
I= 12	TEMP=	****	****	****	27.16	27.64	28.07	28.41	28.86	28.67	28.10	27.57	27.17	26.99
I= 13	TEMP=	****	****	27.02	27.11	****	****	28.26	29.32	29.03	28.25	27.55	27.16	26.99
I= 14	TEMP=	27.26	27.17	28.12	****	****	****	****	29.50	28.84	27.99	27.37	27.04	26.93
I= 15	TEMP=	27.03	27.12	****	****	****	****	****	****	27.55	27.30	26.97	26.78	26.78

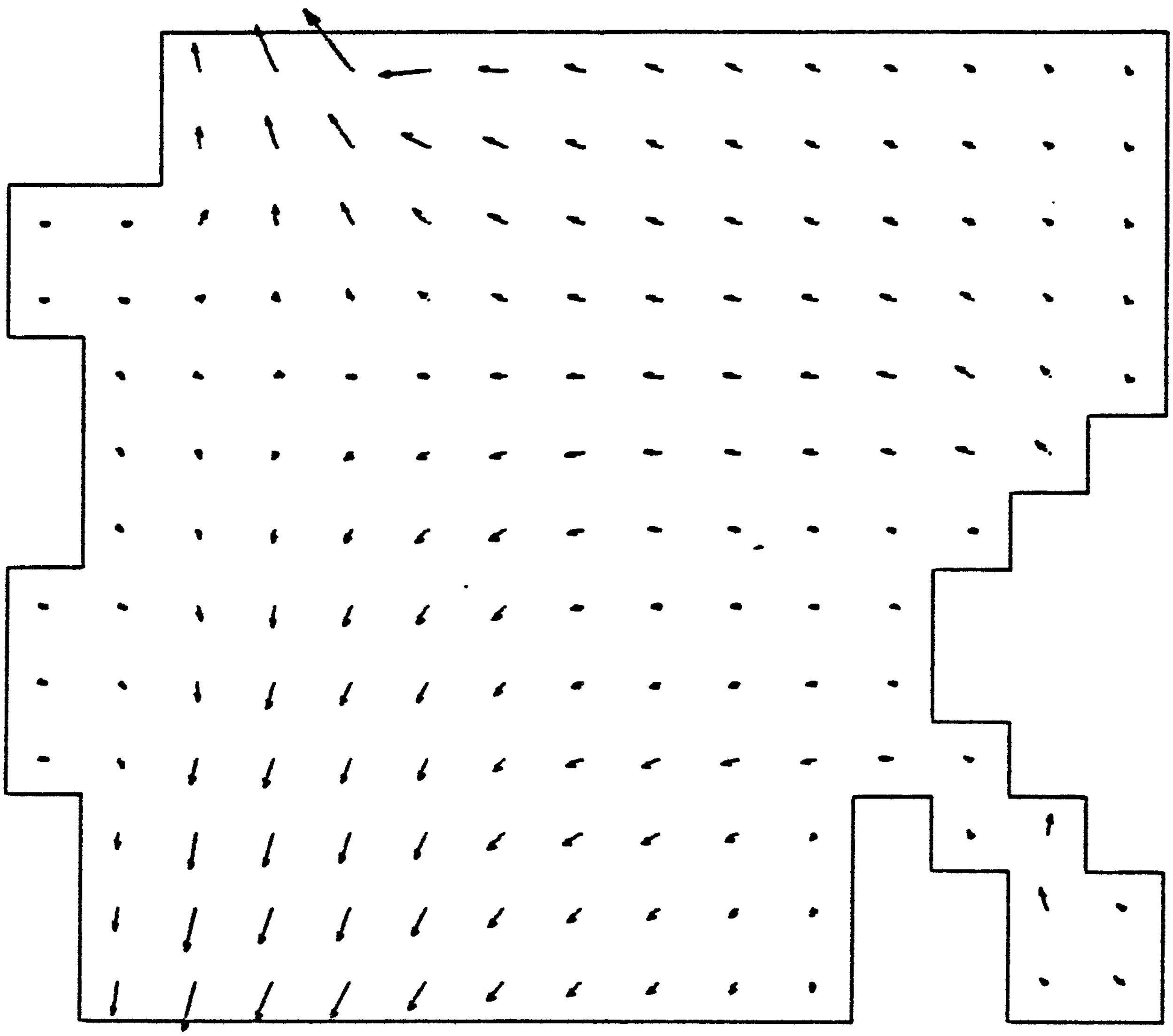
TEMPERATURES, DEG C. K= 3

I= 1	TEMP=	****	****	****	27.14	27.17	27.19	****	****	****	27.08	27.13	****	****
I= 2	TEMP=	27.42	27.48	27.38	27.25	27.19	27.22	27.34	27.32	27.22	27.09	27.11	****	****
I= 3	TEMP=	26.99	27.12	27.21	27.31	27.32	27.20	27.23	27.24	27.19	27.11	27.07	27.15	27.19
I= 4	TEMP=	26.95	26.86	26.89	26.93	27.10	27.05	27.09	27.13	27.17	27.15	27.09	27.09	27.08
I= 5	TEMP=	26.94	26.94	26.91	26.90	26.94	26.94	26.98	27.00	27.06	27.10	27.07	27.03	27.01
I= 6	TEMP=	26.95	26.87	26.87	26.85	26.86	26.89	26.94	26.99	27.03	27.05	27.02	26.97	26.94
I= 7	TEMP=	26.86	26.84	26.85	26.82	26.75	26.78	26.86	26.95	26.97	27.00	26.99	26.97	26.96
I= 8	TEMP=	26.89	26.89	26.91	27.00	26.97	26.92	26.89	26.98	27.01	27.03	27.01	26.99	26.98
I= 9	TEMP=	26.90	26.89	26.89	27.04	27.14	27.18	27.19	27.19	27.17	27.13	27.07	27.01	26.98
I= 10	TEMP=	26.89	26.86	26.86	27.04	27.27	27.42	27.52	27.61	27.56	27.41	27.23	27.08	26.99
I= 11	TEMP=	26.82	26.77	26.76	27.07	27.48	27.83	28.05	28.23	28.14	27.80	27.45	27.15	27.00
I= 12	TEMP=	****	****	****	27.17	27.64	28.07	28.41	28.86	28.67	28.09	27.57	27.18	26.99
I= 13	TEMP=	****	****	27.02	27.11	****	****	28.26	29.32	29.03	28.25	27.55	27.16	26.99
I= 14	TEMP=	27.26	27.18	28.17	****	****	****	****	29.50	28.84	27.99	27.38	27.04	26.94
I= 15	TEMP=	27.03	27.12	****	****	****	****	****	****	27.55	27.30	26.97	26.78	26.79

```
TEMPERATURES, DEG C.     K=    4
I=   1    TEMP=*********************  27.14  27.17  27.19*********************  27.08  27.13**************
I=   2    TEMP=  27.45  27.50  27.39  27.26  27.19  27.23  27.35  27.33  27.22  27.09  27.11*************
I=   3    TEMP=  26.99  27.13  27.22  27.33  27.34  27.21  27.24  27.25  27.20  27.11  27.07  27.16  27.20
I=   4    TEMP=  26.96  26.87  26.89  26.93  27.11  27.07  27.10  27.14  27.18  27.16  27.10  27.10  27.09
I=   5    TEMP=  26.95  26.95  26.92  26.91  26.95  26.95  26.99  27.01  27.07  27.11  27.08  27.04  27.03
I=   6    TEMP=  26.86  26.87  26.87  26.86  26.87  26.89  26.95  27.00  27.04  27.06  27.03  26.98  26.95
I=   7    TEMP=  26.87  26.84  26.85  26.83  26.75  26.78  26.86  26.96  26.98  27.01  27.00  26.98  26.97
I=   8    TEMP=  26.89  26.89  26.91  27.01  26.97  26.93  26.88  26.98  27.02  27.03  27.01  27.00  26.98
I=   9    TEMP=  26.91  26.89  26.90  27.04  27.14  27.19  27.19  27.18  27.17  27.14  27.08  27.02  26.98
I=  10    TEMP=  26.90  26.86  26.86  27.04  27.28  27.43  27.52  27.60  27.55  27.41  27.24  27.08  26.99
I=  11    TEMP=  26.82  26.78  26.77  27.07  27.48  27.84  28.05  28.22  28.13  27.80  27.45  27.15  27.01
I=  12    TEMP=*********************  27.17  27.64  28.07  28.41  28.66  28.66  28.09  27.57  27.18  26.99
I=  13    TEMP=**************  27.03  27.11**************  28.27  29.32  29.03  28.25  27.55  27.17  26.99
I=  14    TEMP=  27.27  27.19  28.25****************************  29.50  28.84  27.99  27.38  27.05  26.94
I=  15    TEMP=  27.03  27.13*************************************  27.56  27.30  26.97  26.78  26.79
TEMPERATURES, DEG C.     K=    5
I=   1    TEMP=*********************  27.15  27.17  27.19*********************  27.09  27.14**************
I=   2    TEMP=  27.46  27.51  27.40  27.26  27.20  27.23  27.35  27.33  27.23  27.09  27.12*************
I=   3    TEMP=  26.98  27.13  27.22  27.35  27.36  27.22  27.25  27.25  27.20  27.11  27.07  27.16  27.21
I=   4    TEMP=  26.97  26.88  26.89  26.93  27.12  27.07  27.10  27.15  27.19  27.16  27.10  27.10  27.10
I=   5    TEMP=  26.96  26.96  26.92  26.91  26.96  26.96  26.99  27.02  27.08  27.11  27.08  27.04  27.05
I=   6    TEMP=  26.86  26.88  26.88  26.86  26.87  26.90  26.95  27.00  27.05  27.06  27.03  26.99  26.96
I=   7    TEMP=  26.87  26.85  26.85  26.83  26.75  26.78  26.86  26.97  26.99  27.01  27.00  26.98  26.97
I=   8    TEMP=  26.90  26.89  26.92  27.01  26.97  26.93  26.88  26.98  27.02  27.04  27.02  27.00  26.99
I=   9    TEMP=  26.91  26.90  26.90  27.05  27.14  27.19  27.19  27.18  27.17  27.14  27.08  27.02  26.99
I=  10    TEMP=  26.90  26.86  26.86  27.04  27.28  27.43  27.52  27.60  27.55  27.41  27.24  27.08  27.00
I=  11    TEMP=  26.92  26.78  26.77  27.08  27.48  27.84  28.06  28.22  28.13  27.80  27.45  27.15  27.01
I=  12    TEMP=*********************  27.17  27.65  28.07  28.41  28.86  28.65  28.09  27.57  27.18  27.00
I=  13    TEMP=**************  27.03  27.11**************  28.27  29.32  29.03  28.25  27.55  27.17  27.00
I=  14    TEMP=  27.27  27.20  28.29****************************  29.50  28.84  27.99  27.38  27.05  26.94
I=  15    TEMP=  27.33  27.13*************************************  27.56  27.30  26.98  26.78  26.79
WATER DEPTHS,CM
I=     1*********************   79.0   79.3   79.6*********************   95.7   95.7**************
I=     2  251.4  221.5  199.2  161.5  124.0  139.5  154.9  139.8  117.2  109.9  109.8*************
I=     3  158.1  335.7  328.7  321.4  268.4  245.9  261.4  231.1  179.2  139.8  146.9  161.3  198.5
I=     4  320.0  312.9  328.7  374.8  367.4  322.1  322.3  299.6  230.7  185.3  192.4  230.0  267.4
I=     5  335.4  313.0  305.8  329.1  337.0  329.7  329.9  314.8  284.4  238.6  238.3  275.9  305.9
I=     6  305.0  297.8  290.6  298.6  306.5  314.5  314.7  314.9  307.3  276.8  269.0  291.6  322.0
I=     7  244.2  244.6  245.0  230.1  222.4  246.0  292.0  322.6  315.0  292.2  276.8  284.3  307.1
I=     8  191.1  191.5  199.7  177.1  139.4  154.8  238.8  307.5  307.6  292.3  277.0  276.9  292.1
I=     9  153.4  153.9  169.7  177.7  140.0  140.2  208.7  284.9  307.7  292.5  277.2  277.1  292.3
I=    10  130.3  123.7  147.5  185.8  163.2  148.0  193.6  262.2  292.7  285.0  277.4  277.3  284.8
I=    11  109.2  101.3  132.6  178.6  163.5  133.1  149.2  216.7  270.0  277.6  269.9  269.8  269.7
I=    12*********************  163.9  140.9  110.5  118.2  163.7  232.1  254.9  239.6  239.5  239.4
I=    13**************  157.2  156.9**************  111.0  164.3  202.1  194.3  194.1  201.6  201.5
I=    14  136.6  158.9  172.6****************************  187.7  164.4  148.9  171.5  186.6  178.8
I=    15  144.8  136.6***********************************  111.2  126.2  148.8  156.2  148.5
```

SAMPLE PLOTTING

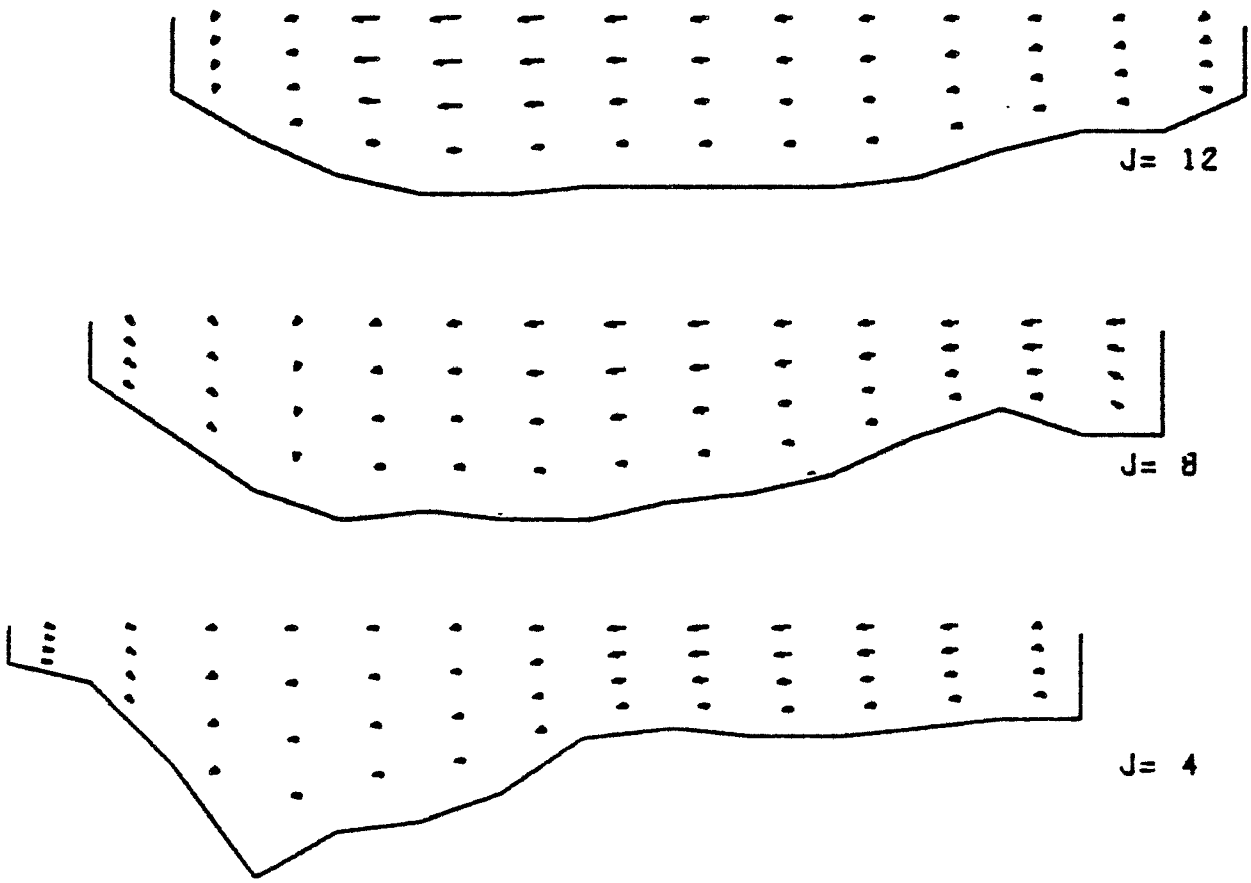
TIME(JUNE 20, 1978):	13.0
WIND SPEED(CM/SEC):	358.0
WIND DIRECTION(DEG/N):	110.
AIR TEMPERATURE(DEG-C):	29.4
DISCHARGE TEMP(DEG-C):	29.5
DISCH FLOWRATE(CUM/SEC):	62.7
LENGTH SCALE(1CM= X CM):	41339.
VELOCITY SCALE(CM/SEC):	52.49



EBB TIDE

Figure 12. Surface velocity, Anclote Anchorage by modeling

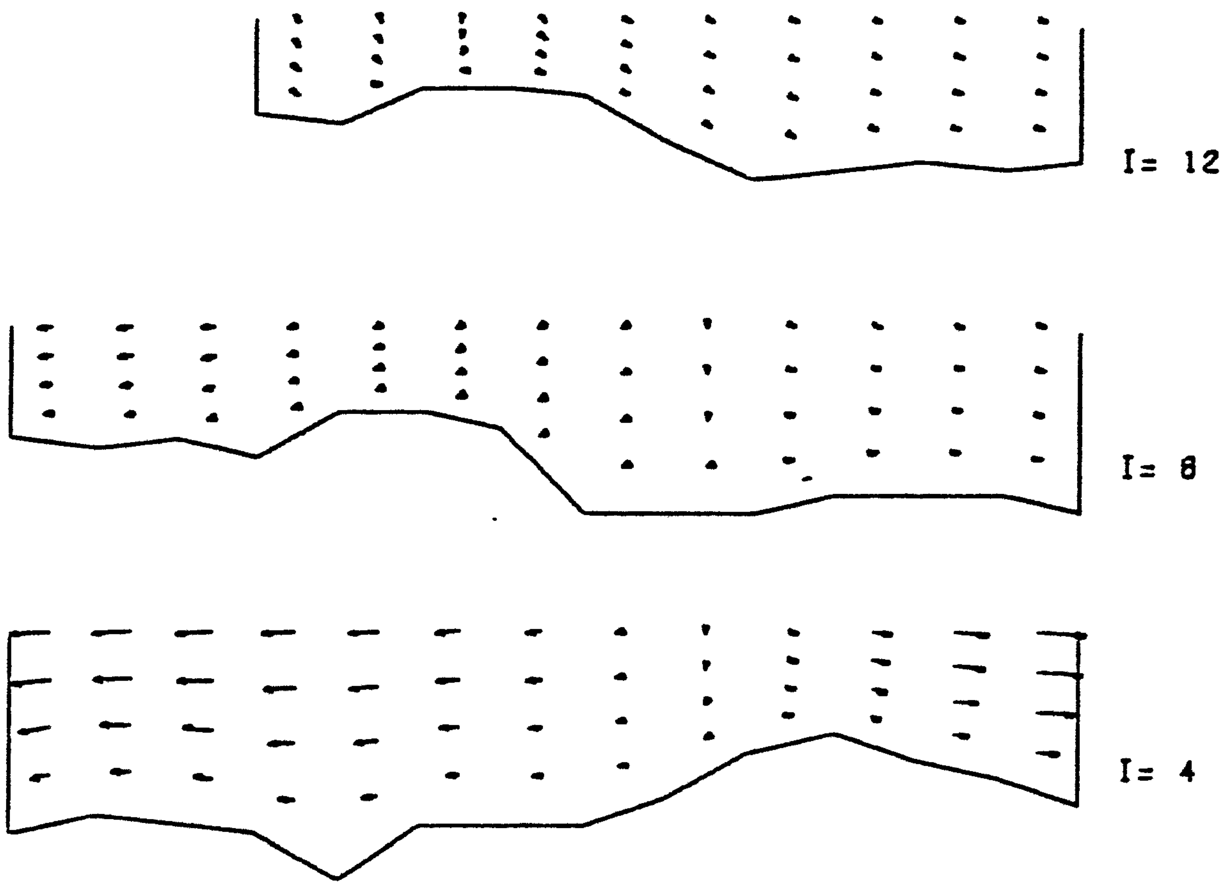
TIME(JUNE 20, 1978):	13.0
WIND SPEED(CM/SEC):	358.0
WIND DIRECTION(DEG/N):	110.
AIR TEMPERATURE(DEG-C):	29.4
DISCHARGE TEMP(DEG-C):	29.5
DISCH FLOWRATE(CUM/SEC):	62.7
LENGTH SCALE(1CM= X CM):	41339.
VELOCITY SCALE(CM/SEC):	52.49



EBB TIDE

Figure 13. UW velocity, Anclote Anchorage by modeling

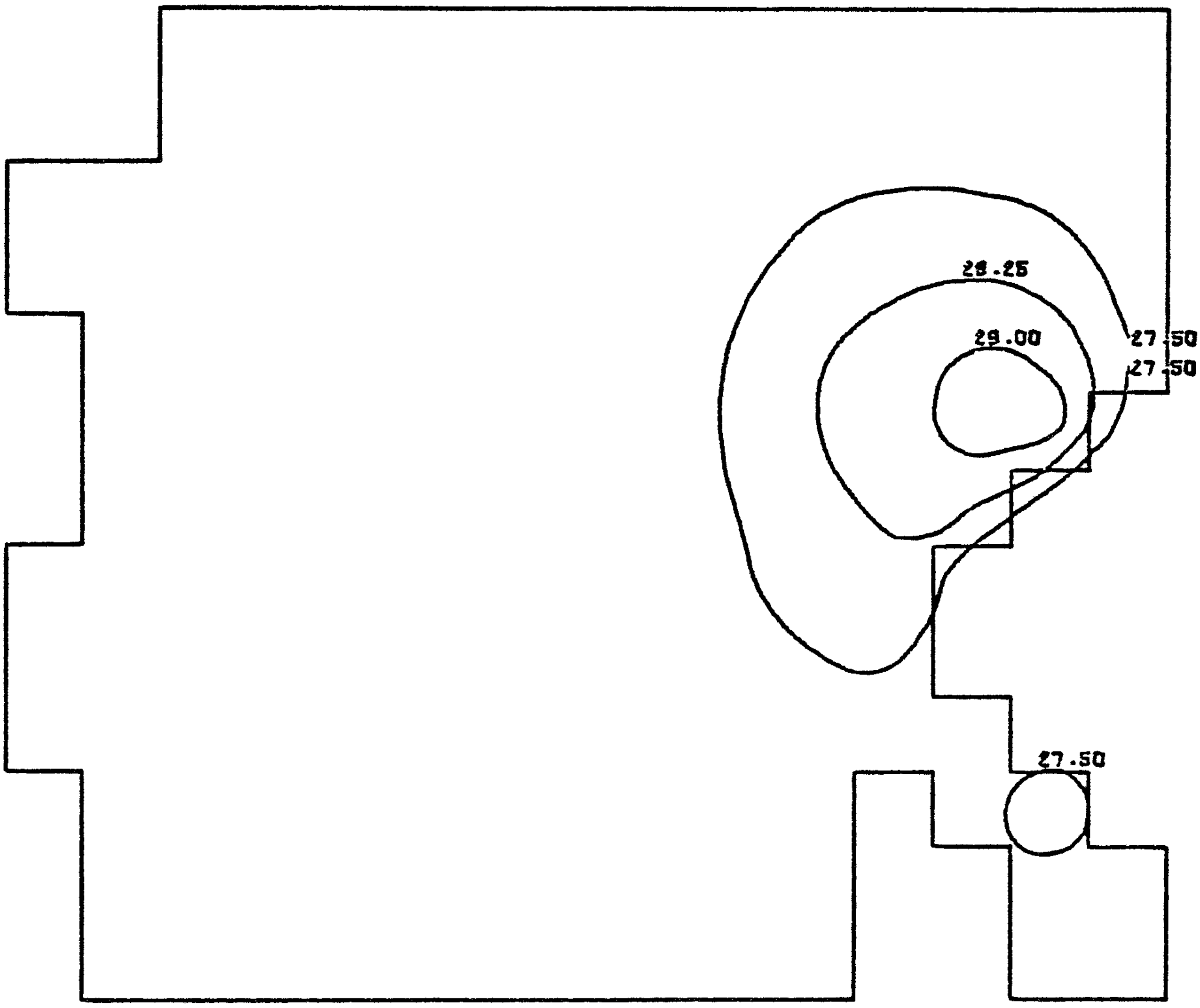
TIME(JUNE 20, 1978):	13.0
WIND SPEED(CM/SEC):	358.0
WIND DIRECTION(DEG/N):	110.
AIR TEMPERATURE(DEG-C):	29.4
DISCHARGE TEMP(DEG-C):	29.5
DISCH FLOWRATE(CUM/SEC):	62.7
LENGTH SCALE(1CM= X CM):	41339.
VELOCITY SCALE(CM/SEC):	52.49



EBB TIDE

Figure 14. VW velocity, Anclote Anchorage by modeling

TIME(JUNE 20, 1978): 13.0
WIND SPEED(CM/SEC): 358.0
WIND DIRECTION(DEG/N): 110.
AIR TEMPERATURE(DEG-C): 29.4
DISCHARGE TEMP(DEG-C): 29.5
DISCH FLOWRATE(CUM/SEC): 62.7
LENGTH SCALE(1CM= X CM): 41339.
VELOCITY SCALE(CM/SEC): 52.49



DEVIATION FROM IR TEMP: 0.410
EBB TIDE

Figure 15. Surface temperature, Anclote Anchorage by modeling

APPENDIX B

FORTRAN SOURCE PROGRAM LISTING

LIST OF SUBROUTINES OF THE MODEL

Calculating Part

1. Main Program

   ANCMN

2. Subroutines Called (in order)

   BAYBOT: Reads grid matrices and bottom topography

   BAYINI: Specifies initial conditions

   READT: Reads data from tape for continuing run

   IRREAD: Reads IR data as initial temperature distribution

   EQTEMP: Calculates equilibrium temperature

   BETA: Calculates surface elevation and vertical velocity in $xy\sigma$ coordinates

   BNRTIA: Calculates inertia terms in momentum eq. at interior points

   ABNR3: Calculates inertia terms in momentum eq. on the north and south boundaries

   BVELS: Calculates interior velocities

   ASAF3: Calculates north and south boundary velocities

   GIVENU: Specifies velocities at discharge point and river mouth

   CONV: Calculates convective terms in energy eq.

   TCOMPT: Calculates interior temperatures

   GIVENT: Specifies temperature at discharge point

WCAL: Converts vertical velocity in xyσ coordinates into xyz coordinates

ANCPR: Prints surface height, velocity and temperature at four locations at each time step

TPRLOK: Main printing program

STORET: Stores calculating results onto the tape

ZZI: Finds the current direction

3. Data Files

AMATN: Specifies marker matrices and elevations

APER1: Specifies outline of interest area

C2007: Climates data on June 20, 1978, start at 0700

HDATA: High tide data from IR

EDATA: Ebb tide data from IR

LDATA: Low tide data from IR

FDATA: Flood tide data from IR

<u>Plotting Part</u>

1. Main Program

PLOTMN

2. Subroutines Called (in order)

PLOTUV: Plots U, V velocities on different levels

PLOTUW: Plots U, W velocities at different j sections

PLOTVW: Plots V, W velocities at different i sections

ECHKON: Plots surface isotherms and surface height

ENDER: Subroutine in ECHKON, for labeling

CONLIN: Subroutine in ECHKON, for contouring

CAPTN1: Writes captions on the plot

CAPTN2: Writes captions on the plot

CAPTN3: Writes captions on the plot

CAPTN4: Writes captions on the plot

CAPTN5: Writes captions on the plot

CAPTN6: Writes captions on the plot

CAPTN7: Writes captions on the plot

CAPTN8: Writes captions on the plot

CAPTN9: Writes captions on the plot

FIT: Fits a parabolar to three points

VECT: Establishes the components of a vector

OUTLIN: Draws the outline of interest area

* The plotting subroutines PLOTS, PLOT, AROHD, NUMBER, SYMBOL are existing in UNIVAC 1100, University of Miami, CALCOMP file.

SUBROUTINE LISTINGS

```
*FLOW(1).ANCMN FOR CREATED ON 14 DEC 79 AT 10:41:39
      C*******************************************************************
      C  MAIN PROGRAM OF 3-D, FREE-SURFACE, HYDROTHERMAL MODEL.  FEATURES:
      C  CTCS & DUFORT-FRANKEL DIFFERENCING; DIFFUSION OF T CONSIDERED;
      C  NONLINEAR TERMS CAN BE INCLUDED; ASSOCIATED WITH PLOT PROGRAM-PLOTMN
      C*******************************************************************
            PARAMETER IN=16,JN=14,KN=5,IM=15,JM=13
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),PX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            DIMENSION T1(IM,JM,KN),T2(IM,JM,KN),T3(IM,JM,KN),TC(IM,JM,KN)
            DIMENSION AVR(JM),ANG(JM),DTZ(IM,JM),TE(IM,JM,KN)
            DIMENSION ETX(IM,JM),TIDE3N(IM),TIDE2N(IM),TIDE1N(IM)
            DIMENSION WAT(IM,JM),TIDE3S(IM),TIDE2S(IM),TIDE1S(IM)
            WRITE(6,267)
      C
      C**1**CASE NO UNDERTAKING, FOR LABELLING PURPOSE
            READ 2, NCASE
      C
      C**2**LN=1 FIRST RUN OF PRESENT CASE; >1 SUBSEQUENT RUN
            READ 2, LN
      C
      C**3**KSTORE=0 NO STORE(TEST RUN), =1 STORE ON TAPE
            READ 2, KSTORE
      C
      C**4**KVEL=0 NO V-CALCULATION, =1 DO V-CALCULATION
            READ 2, KVEL
      C
      C**5**KTEMP=0 NO T-CALCULATION, =1 DO T-CALCULATION
            READ 2, KTEMP
      C
      C**6**TO ASSURE MBLOK CONTINUE, THUS AS COUNTER OF SIMULATION HOUR
            READ 2, MBLOK
      C
      C**7**THIS RUN WILL DO NCY*TPRT/3600 HOURS OF SIMULATION
            READ 2, NCY
      C
      C**8**FOR HOURLY CYCLE TPRT=3600; OTHERWISE CLIMDATA DO LIKEWISE
            READ 2, TPRT
      C
      C**9**DX,DY GRID SIZE IN CM; DS SPACING IN SEGMA DIRECTION(.25)
            READ 2, DX,DY,DS
      C
      C**10**TIMESTEP,MWL-TEMPORAL MWL,AMPLIT,N-S PHASE DIFFERENCE,
      C**10**E-W PHASE DIFF PER DX, TIDEPERIOD, TIDE SHIFT IN HOUR
            READ 2, DT,STAGE,AMPLIT,PHASE,DPHASE,PERIOD,TSHIFT
      C
      C**11**FCOR=CORIOLIS FACTOR,TEMPORAL MWL-ANNUAL MWL(REF. FOR SOUNDING)
            READ 2, FCOR,STAGE1
      C
      C**12**THETA=ANGLE BETWEEN NORTH AND GRID Y-AXIS, POSITIVE CLOCKWISE
            READ 2, THETA
      C
      C**13**ALREF=REFERENCE HORIZONTAL LENGTH IN CM, ROSSBY NO(ESTIMATED)
            READ 2, ALREF,ROSSBY
      C
      C**14**RWEX=NO OF HOURS BETWEEN WEATHER OBSERVATIONS(IN GENERAL HOURLY)
            READ 2, RWEX
      C
      C**15**TZERO=EST OF THE DAY WHEN THE SIMULATION RUN STARTS
            READ 2, TZERO
      C
      C**16**DENSITY,VERT EDDY VISCOSITY,VERT & HORI EDDY DIFFUSIVITY
            READ 2, RR,AV,BV,BH
      C
      C**17**INITIAL TEMP FOR THE WHOLE COMPUTATIONAL DOMAIN
            READ 2, TINIT
      C
            WRITE(6,32) NCASE
            WRITE(6,3) LN
            IF(KSTORE.EQ.0) WRITE(6,130)
            IF(KSTORE.GT.0) WRITE(6,131)
        130 FORMAT(1X,'DATA NOT RECORDED ON TAPE')
        131 FORMAT(1X,'DATA RECORDED ON TAPE')
            WRITE(6,4) NCY
            WRITE(6,5) DX,DY,DS,DT
```

```
      WRITE(6,34) STAGE,AMPLIT,PHASE,DPHASE,PERIOD,TSHIFT
      WRITE(6,7) FCOR
      WRITE(6,1011) THETA
      WRITE(6,15) ALREF
   15 FORMAT(1X,'ALREF=',F12.0,'    CM')
      WRITE(6,72) TZERO
      WRITE(6,73) RR,AV,BV,BH
      WRITE(6,806) RWEX
  806 FORMAT(1X,'RWEX=',F10.2)
      WRITE(6,384) TINIT
  384 FORMAT(1X,'TINIT=',F10.2)
   72 FORMAT(1X,'TZERO=',F10.2)
   73 FORMAT(1X,'RR=',F10.2,'   AV=',F10.2,'   BV=',F10.2,'  BH=',F10.2)
      G=980.
      QQ=57.3
      KZ=KN-1
      IGO=1
      THETA=THETA/QQ
      DUMX=2.*DX
      DUMY=2.*DY
      DUMS=2.*DS
      IF(LN.GE.2) GO TO 1
      EST=TZERO
      TAUX=0.
      TAUY=0.
      TTOT=0.
      NBLOK=0
      CALL BAYBOT(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETA,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CSTAGE1)
C
C**18**A DECK OF AMATN CARDS OR FILE AMATN IS NEEDED HERE,(ONLY IF LN=1)***
C
      CALL BAYINI(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETA,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CT1,T2,T3,TC,TB,TINIT)
      DO 501 I=1,IM
      TIDE1S(I)=0.
      TIDE2S(I)=0.
501   TIDE3S(I)=0.
      DO 502 I=1,IM
      TIDE1N(I)=0.
      TIDE2N(I)=0.
502   TIDE3N(I)=0.
      GO TO 222
C * * * * * * * * * * * * * * * * * * * * * * * * * *
    1 CONTINUE
      CALL READT(IN,JN,KN,IM,JM,U1,U2,V1,V2,W,ETA1,T1,T2,TB,
     CETA,MX,MY,MAR,H,HB,UB,VB,HU,HV,MEX,MEY,NCASE,NBLOK,TTOT,DT,
     CTIDE1N,TIDE2N,TIDE3N,TIDE1S,TIDE2S,TIDE3S,STAGE,EST,
     CAMPLIT,PHASE,DPHASE,PERIOD)
      IF(NBLOK.NE.MBLOK) GO TO 1000
C
C*****IF ANY OF FOLLOWING PARAMETERS NEED TO BE CHANGED, HERE IS CHANCE
C**19**CONT BLOCK NO, CONT TOTAL TIME,DT,EST,TIDE DATA..........FOLLOW
      READ 2,NBLOK,TTOT,DT,EST,AMPLIT,PHASE,DPHASE,PERIOD,TSHIFT
C
      WRITE(6,34) STAGE,AMPLIT,PHASE,DPHASE,PERIOD,TSHIFT,DT
      WRITE(6,33) EST,NCASE,NBLOK
C
C*****IF IR-T IS USED AS INITIAL TEMP, THEN RELIEVE FOLLOWING CALL
C*****THEN A DECK OF IR-DATA CARDS IS NEEDED IN IRREAD SUBROUTINE
  222 CALL IRREAD(IM,JM,KN,T1,T2)
C
C 222 CONTINUE
C*********************************************
C*********MAIN LOOP IN HOURLY-STEP, WITH HOURLY CLIMATICAL DATA
C*********************************************
      DO 100 JCTR=1,NCY
C
C**20**A DECK OF NCY CLIMATICAL DATA CARDS FOLLOWS
C**20**EACH CARD RECORDS AIR TEMP,HUMIDITY,WSPEED,WDIR,SORAD,SURFACE TEMP
      READ 2, TAIR,HUMID,WIND,WDIR,SRAD,TSURF
C
      WRITE(6,328) TAIR,HUMID,WIND,WDIR,SRAD,TSURF
  328 FORMAT(1X,'TAIR,HUMID,WIND,WDIR,SRAD,TSURF',6F10.2)
      CB=2.
```

```
      EPSLON=(WDIR+180.)/57.3
      WPR=WIND/100.
      IF(WIND.LT.100.)CTEN=.00125*(WPR**.2)
      IF((WIND.GE.100.).AND.(WIND.LT.1500.))CTEN=.0005*SQRT(WPR)
      IF(WIND.GE.1500.)CTEN=.0026
      TAU=CTEN*.00123*(WIND**2)
      TAUX=TAU*SIN(EPSLON-THETA)
      TAUY=TAU*COS(EPSLON-THETA)
      CALL EQTEMP(TAIR,HUMID,WIND,WDIR,SRAD,TSURF,TDEW,SK,TEQ)
      DTX=TPRT
      TZ=0.
C * * * * * * * * * * * * * * * * * * * * *
  702 CONTINUE
      TTOT=TTOT+DT
      TZ=TZ+DT
      DHR=DT/3600.
      EST=EST+DHR
C
C*****NORTH & SOUTH AMBIENT TEMP IS SINE OF 24 HR PERIOD, OR SPECIFIED
      TABS=27.0+.2*SIN(.2618*(EST-12.0))
      TABN=27.0+.2*SIN(.2618*(EST-12.0))
C
      IF(KVEL.EQ.0) GO TO 61
C
      CALL BETA(IN,JN,KN,IM,JM,U2,V2,OM,ETA1,ETA,ETA3,
     CHB,HL,HV,MEX,MEY,CB,DX,DY,DS,DT)
      IF(RCSSBY.LE.0.) GO TO 41
      CALL BNRTIA(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETA,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,CB)
      CALL ABNR3(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,EST,
     CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,CB,
     CTIDE1S,TIDE2S,TIDE3S,TIDE1N,TIDE2N,TIDE3N,STAGE,
     CAMPLIT,PHASE,OPHASE,PERIOD,TSHIFT)
   41 CONTINUE
      CALL BVELS(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETA,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,CB,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CAV)
      CALL ASAF3(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
     CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,CB,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CTIDE1S,TIDE2S,TIDE3S,EST,STAGE,AV,TIDE1N,TIDE2N,TIDE3N,
     CAMPLIT,PHASE,OPHASE,PERIOD,TSHIFT)
      CALL GIVENU(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,T1,T2,T3,EST)
   61 CONTINUE
C
      IF(KTEMP.EQ.0) GO TO 63
C
      CALL CONV(IN,JN,KN,IM,JM,DX,DY,DS,DT,DUMX,DUMY,DUMS,SK,RR,BV,
     CU2,UB,V2,VB,T2,TC,HB,HU,HV,OM,ETA1,ETA,ETA3,CP,MEX,MEY,W,DTZ,
     CTEQ)
      CALL TCOMPT(IN,JN,KN,IM,JM,DX,DY,DS,DT,DUMX,DUMY,DUMS,
     CHB,TC,T1,T2,T3,MEX,MEY,ETA,CP,TAIR,ETA1,ETA3,DTZ,BV,RH,TABN,TABS)
      CALL GIVENT(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,T1,T2,T3,EST)
   63 CONTINUE
C
      DO 200 I=1,IM
      DO 200 J=1,JM
      ETX(I,J)=ETA3(I,J)
      ETA1(I,J)=ETA(I,J)
  200 ETA(I,J)=ETA3(I,J)
      DO 201 I=1,IN
      DO 201 J=1,JN
      DO 202 K=1,KZ
      U1(I,J,K)=U2(I,J,K)
      U2(I,J,K)=U3(I,J,K)
      V1(I,J,K)=V2(I,J,K)
      V2(I,J,K)=V3(I,J,K)
  202 CONTINUE
  201 CONTINUE
      DO 203 I=1,IM
      DO 203 J=1,JM
      IF(MEX(I,J).EQ.0) GO TO 203
      DO 204 K=1,KZ
      T1(I,J,K)=T2(I,J,K)
      T2(I,J,K)=T3(I,J,K)
      UB(I,J,K)=(U2(I,J,K)+U2(I+1,J,K))/2.
```

```
      VB(I,J,K)=(V2(I,J,K)+V2(I,J+1,K))/2.
  204 CONTINUE
  203 CONTINUE
C
      DTX=TPRT-T2
      CALL WCAL(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETA,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT)
      IF(VB(3,1,1).GE.200.) IGO=0
      IF(ABS(TB(14,2,1)).GE.50.) IGO=0
      CALL ANCPR(IM,JM,KN,UB,VB,ETX,TTOT,ROSSBY,THETA,T2)
  700 CONTINUE
      IF(ICO.NE.0) GO TO 861
      WRITE(6,703)
      CALL TPRLOK(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETX,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,T2,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CKVEL,KTEMP,AVR,ANG,THETA)
      GO TO 1000
  861 CONTINUE
      IF(T2.LT.TPRT) GO TO 702
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  600 CONTINUE
      WRITE(6,607) EST,TAIR,TEQ,TABN,TABS,TAUX,TAUY
  607 FORMAT(1X,'EST,TAIR,TEQ,TAN,TAS,TAUX,TAUY=',5F6.2,2F12.6)
      DO 343 I=1,IM
      DO 343 J=1,JM
      IF(MEX(I,J).EQ.0) GO TO 343
      DO 344 K=1,KN
      IF(K.EQ.1) TB(I,J,K)=T2(I,J,K)+((HB(I,J)+ETA(I,J))*DS/2.)*DTZ(I,J)
      IF(K.EQ.KN) TB(I,J,K)=T2(I,J,K-1)
      IF((K.NE.1).AND.(K.NE.KN)) TB(I,J,K)=(T2(I,J,K)+T2(I,J,K-1))/2.
344   CONTINUE
343   CONTINUE
      IF(KSTORE.EQ.0) GO TO 132
      CALL STORET(IN,JN,KN,IM,JM,U1,U2,V1,V2,W,ETA1,T1,T2,TB,
     CETA,MX,MY,MAR,H,HB,UB,VB,HU,HV,MEX,MEY,NCASE,NBLOK,TTOT,DT,
     CTIDE1N,TIDE2N,TIDE3N,TIDE1S,TIDE2S,TIDE3S,STAGE,EST,
     CAMPLIT,PHASE,DPHASE,PERIOD)
  132 CONTINUE
      CALL TPRLOK(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,ETX,ETA3,
     CRX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,TB,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CKVEL,KTEMP,AVR,ANG,THETA)
      WRITE(6,812)
  812 FORMAT(1X,'WATER DEPTHS,CM')
      DO 813 I=1,IM
      DO 813 J=1,JM
      WAT(I,J)=10.**9
      IF(MEX(I,J).NE.0) WAT(I,J)=HB(I,J)+ETA(I,J)
  813 CONTINUE
      DO 814 I=1,IM
  814 WRITE(6,815) I,(WAT(I,J),J=1,JM)
  815 FORMAT(1X,'I=',I4,14F7.1/)
      WRITE(6,267)
  100 CONTINUE
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(KSTORE.EQ.0) GO TO 134
      END FILE 8
      WRITE(6,308)
  308 FORMAT(1X,'DOUBLE EOF PLACED ON TAPE')
  134 CONTINUE
    2 FORMAT( )
    3 FORMAT(1X,'LN=',I6)
    4 FORMAT(1X,'NCY=',I10)
    5 FORMAT(1X,'DX=',F10.0,'  CM      DY=',F10.0,' CM       DS=',F10.2,
     C'       TIME STEP=',F10.2,'SEC')
    7 FORMAT(1X,'FCOR=',E15.7,'  PER SEC')
   32 FORMAT(1X,'NCASE=',I4)
   33 FORMAT(1X,'START AT EST=',F6.2,'   CASE NO.=',I4,'    NBLOK=',I4)
   34 FORMAT(1X,'STAGE,AMP,N-S PHASE,E-W PHASE,PER,TSHIFT',6F7.3)
  267 FORMAT('1')
  703 FORMAT(1X,'COMPUTATIONS BEING STOPPED BECAUSE OF INSTABILITY')
 1011 FORMAT(1X,'THETA',F10.1,'  DEG')
 1000 STOP
      END
```

```
*FLOW(1).BAYBOT FOR CREATED ON 4 DEC 79 AT 09:44:24
      C****************************************************************
      C  READS MARKER MATRICES & ELEV MATRIX, DETERMINES MATRICES HB, HU, HV.
      C****************************************************************
            SUBROUTINE BAYBOT(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
           CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
           CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
           CSTAGE)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            AB=10.**9
            WRITE(6,51)
            DO 99 I=1,IN
            READ 2, (MAR(I,J),J=1,JN)
            WRITE(6,60) I,(MAR(I,J),J=1,JN)
         99 CONTINUE
            WRITE(6,71)
         71 FORMAT(1X,'BOTTOM ELEVATIONS BELOW MSL, FT'/)
            DO 37 I=1,IN
            READ 2, (ELEV(I,J),J=1,JN)
            WRITE(6,30) I,(ELEV(I,J),J=1,JN)
         37 CONTINUE
            DO 100 I=1,IN
            DO 100 J=1,JN
            IF(MAR(I,J).EQ.0) GO TO 100
            ELEV(I,J)=-ELEV(I,J)
            ELEV(I,J)=30.48*ELEV(I,J)
            H(I,J)=STAGE-ELEV(I,J)
            H(I,J)=AMAX1(0.,H(I,J))
        100 CONTINUE
            WRITE(6,31)
            DO 29 I=1,IN
            WRITE(6,30) I,(H(I,J),J=1,JN)
         29 CONTINUE
            WRITE(6,52)
         52 FORMAT(1X,'MEX MATRIX')
            DO 53 I=1,IM
            READ 2, (MEX(I,J),J=1,JM)
            WRITE(6,60) I,(MEX(I,J),J=1,JM)
         53 CONTINUE
            WRITE(6,54)
         54 FORMAT(1X,'MEY MATRIX')
            DO 55 I=1,IM
            READ 2, (MEY(I,J),J=1,JM)
            WRITE(6,60) I,(MEY(I,J),J=1,JM)
         55 CONTINUE
            WRITE(6,1)
          1 FORMAT(1X,'MX MATRIX')
            DO 826 I=1,IN
            READ 2, (MX(I,J),J=1,JM)
            WRITE(6,60) I,(MX(I,J),J=1,JM)
        826 CONTINUE
            WRITE(6,3)
          3 FORMAT(1X,'MY MATRIX')
            DO 402 I=1,IM
            READ 2, (MY(I,J),J=1,JN)
            WRITE(6,60) I,(MY(I,J),J=1,JN)
        402 CONTINUE
            DO 101 I=1,IM
            DO 101 J=1,JM
            IF(MEX(I,J).EQ.0) GO TO 101
            HB(I,J)=(H(I,J)+H(I+1,J)+H(I+1,J+1)+H(I,J+1))/4.
        101 CONTINUE
            DO 102 I=1,IN
            DO 102 J=1,JM
            IF(MX(I,J).EQ.0) GO TO 102
            HU(I,J)=(H(I,J)+H(I,J+1))/2.
        102 CONTINUE
            DO 103 I=1,IM
            DO 103 J=1,JN
            IF(MY(I,J).EQ.0) GO TO 103
            HV(I,J)=(H(I,J)+H(I+1,J))/2.
        103 CONTINUE
            WRITE(6,109)
        109 FORMAT(1X,'HB MATRIX')
```

```
      DO 110 I=1,IM
  110 WRITE(6,30) I,(HB(I,J),J=1,JM)
      WRITE(6,111)
  111 FORMAT(1X,'HU MATRIX')
      DO 112 I=1,IN
  112 WRITE(6,30) I,(HU(I,J),J=1,JM)
      WRITE(6,113)
  113 FORMAT(1X,'HV MATRIX')
      DO 114 I=1,IM
  114 WRITE(6,30) I,(HV(I,J),J=1,JN)
    2 FORMAT ( )
   30 FORMAT(1X,'I=',I4,15F8.1)
   31 FORMAT(1X,'WATER DEPTHS, CM'/)
   40 FORMAT(1X,'I=',I4,15E8.2)
   41 FORMAT(1X,'BOTTOM SLOPES, CM/CM'/)
   51 FORMAT(1X,'MAR MATRIX'/)
   60 FORMAT(1X,'I=',I4,10X,15I4)
   61 FORMAT(1X,'STAGE=',F10.2)
  250 FORMAT('1')
      RETURN
      END
```

```
*FLOW(1).BAYINI FOR CREATED ON 4 DEC 79 AT 09:45:24
C**********************************************************************
C  INITIALIZES MOST OF THE DEPENDANT VARIABLES
C**********************************************************************
      SUBROUTINE BAYINI(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
     CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CT1,T2,T3,TC,TB,TINIT)
      DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
     CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
     CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
     CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
     CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
     CMEY(IM,JM),TB(IM,JM,KN)
      DIMENSION T1(IM,JM,KN),T2(IM,JM,KN),T3(IM,JM,KN),TC(IM,JM,KN)
      KZ=KN-1
      DO 100 I=1,IN
      DO 100 J=1,JN
      DO 100 K=1,KN
      U1(I,J,K)=0.
      U2(I,J,K)=0.
      U3(I,J,K)=0.
      V1(I,J,K)=0.
      V2(I,J,K)=0.
      V3(I,J,K)=0.
      RX(I,J,K)=0.
      RY(I,J,K)=0.
  100 CONTINUE
      DO 200 I=1,IM
      DO 200 J=1,JM
      ETA1(I,J)=0.
      ETA(I,J)=0.
      ETA3(I,J)=0.
      DO 300 K=1,KN
      UB(I,J,K)=0.
      VB(I,J,K)=0.
      W(I,J,K)=0.
      OM(I,J,K)=0.
      T1(I,J,K)=TINIT
      T2(I,J,K)=TINIT
      T3(I,J,K)=TINIT
      TC(I,J,K)=0.
      TB(I,J,K)=TINIT
  300 CONTINUE
  200 CONTINUE
      ABJ=10.**9
      DO 400 I=1,IM
      DO 400 J=1,JM
      IF(MEX(I,J).NE.0) GO TO 400
      ETA1(I,J)=ABJ
      ETA(I,J)=ABJ
      ETA3(I,J)=ABJ
      DO 500 K=1,KN
      TB(I,J,K)=ABJ
      W(I,J,K)=ABJ
  500 CONTINUE
  400 CONTINUE
      RETURN
      END
```

```
*FLOW(1).READT FOR CREATED ON 7 DEC 79 AT 10:08:01
      C*********************************************************************
      C  READS IN HOURLY RESULT STORED IN TAPE FOR CONTINUE RUN OR PLOTTING
      C*********************************************************************
            SUBROUTINE READT(IN,JN,KN,IM,JM,U1,U2,V1,V2,W,ETA1,T1,T2,TB,
           CETA,MX,MY,MAR,H,HB,UB,VB,HU,HV,MEX,MEY,NCASE,NBLOK,TTOT,DT,
           CTIDE1N,TIDE2N,TIDE3N,TIDE1S,TIDE2S,TIDE3S,STAGE,EST,
           CAMPLIT,PHASE,DPHASE,PERIOD)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),V1(IN,JN,KN),V2(IN,JN,KN),
           CW(IM,JM,KN),ETA1(IM,JM),ETA(IM,JM),MX(IN,JN),MY(IN,JN),MAR(IN,JN),
           CH(IN,JN),HB(IM,JM),UB(IM,JM,KN),VB(IM,JM,KN),
           CHU(IN,JN),HV(IN,JN),MEX(IM,JM),MEY(IM,JM),TB(IM,JM,KN)
            DIMENSION T1(IM,JM,KN),TIDE1N(IM),TIDE2N(IM),TIDE3N(IM)
            DIMENSION T2(IM,JM,KN),TIDE1S(IM),TIDE2S(IM),TIDE3S(IM)
          1 CONTINUE
            READ (7,END=1) NBLOK
            READ  (7) (((U1(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((U2(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((V1(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((V2(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C((MAR(I,J),J=1,JN),I=1,IN),
           C((MX (I,J),J=1,JN),I=1,IN),
           C((MY (I,J),J=1,JN),I=1,IN),
           C((HU (I,J),J=1,JN),I=1,IN),
           C((HV (I,J),J=1,JN),I=1,IN),
           C((H(I,J),J=1,JN),I=1,IN)
            READ  (7) ((( W(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((UB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((VB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((T1(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((T2(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((TB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C((ETA1(I,J),J=1,JM),I=1,IM),
           C((ETA(I,J),J=1,JM),I=1,IM),
           C((MEX(I,J),J=1,JM),I=1,IM),
           C((MEY(I,J),J=1,JM),I=1,IM),
           C((HB (I,J),J=1,JM),I=1,IM),
           C(TIDE1N(I),I=1,IM),
           C(TIDE2N(I),I=1,IM),
           C(TIDE3N(I),I=1,IM),
           C(TIDE1S(I),I=1,IM),
           C(TIDE2S(I),I=1,IM),
           C(TIDE3S(I),I=1,IM)
            READ (7) TTOT,NCASE,DT,STAGE,EST,AMPLIT,PHASE,DPHASE,
           CPERIOD
            READ (7,END=500) AAA
            WRITE(6,2)
          2 FORMAT(1X,'NO EOF AT END OF DATA')
            NBLOK=-100
            GO TO 1000
        500 CONTINUE
            WRITE(6,112) TTOT,NCASE,NBLOK
        112 FORMAT(1X,'DATA READ FROM TAPE,    TTOT=',F10.0,
           C'    CASE NBR=',I5,'    BLOK NBR=',I5)
       1000 RETURN
            END
```

```
*FLOW(1).IRREAD SYM CREATED ON 6 DEC 79 AT 09:47:52
      C*********************************************************************
      C  IN THE CASE OF STARTING FROM GIVEN T-FIELD, READS IN THE IR-TEMP
      C*********************************************************************
            SUBROUTINE IRREAD(IM,JM,KN,T1,T2)
            DIMENSION T1(IM,JM,KN),T2(IM,JM,KN)
      C*****DATA READ FROM ONE OF THE (F,H,E,L)DATA BY IR****************
            DO 550 J=1,JM
      550   READ 2, (T1(I,J,1),I=1,IM)
      2     FORMAT( )
            DO 555 J=1,JM
            DO 555 I=1,IM
            DO 553 K=2,KN
      553   T1(I,J,K)=T1(I,J,1)
            DO 554 K=1,KN
      554   T2(I,J,K)=T1(I,J,1)
      555   CONTINUE
            RETURN
            END
```

```
*FLOW(1).EQTEMP SYM CREATED ON 12 DEC 79 AT 20:36:54
      C*******************************************************************
      C  COMPUTES EQUILIBRIUM T AND SURFACE HEAT EXCHANGE COEFFICIENT SK
      C*******************************************************************
            SUBROUTINE EQTEMP(TAIR,HUMID,WIND,WDIR,SRAD,TSURF,TDEW,SK,TEQ)
            TAIR1=TAIR*9./5.+32.
            TSURF1=TSURF*9./5.+32.
            WIND1=WIND/44.7
            SRAD1=SRAD*1440.
            TDEW1=(14.55+.114*TAIR1)*(1.-HUMID)
            TDEW2=((2.5+.007*TAIR1)*(1.-HUMID))**3
            TDEW=TAIR1-TDEW1-TDEW2
            TFILM=(TSURF1+TDEW)/2.
            BETA=.255-.0085*TFILM+.000204*TFILM**2
            FCNU=70.+.7*WIND1**2
            SK=15.7+(BETA+.26)*FCNU
            TEQ=TDEW+SRAD1/SK
            SK=SK*.00000564
            TEQ=(TEQ-32.)*5./9.
            RETURN
            END
```

```
*FLOW(1).BETA FOR CREATED ON 4 DEC 79 AT 09:46:12
      C*****************************************************************
      C  COMPUTES ETA AND OMEGA OF ADVANCE TIME, USES SIMPSON'S RULE FOR INTG
      C*****************************************************************
            SUBROUTINE BETA(IN,JN,KN,IM,JM,U2,V2,OM,ETA1,ETA,ETA3,
           CHB,HU,HV,MEX,MEY,CB,DX,DY,DS,DT)
            DIMENSION U2(IN,JN,KN),V2(IN,JN,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),HB(IM,JM),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            DIMENSION F(5)
            S25=-.25
            S50=-.50
            S75=-.75
            B=2.
            ABC=CB*DT
            KZ=KN-1
            DO 900 I=1,IM
            DO 900 J=1,JM
            IF(MEX(I,J).EQ.0) GO TO 900
            DR=HU(I+1,J)+(ETA(I+1,J)+ETA(I,J))/2.
            IF(MEX(I,J).EQ.2) DR=HU(I+1,J)+ETA(I,J)
            DL=HU(I,J)+(ETA(I-1,J)+ETA(I,J))/2.
            IF(MEX(I,J).EQ.1) DL=HU(I,J)+ETA(I,J)
            D2=HV(I,J+1)+(ETA(I,J+1)+ETA(I,J))/2.
            IF(MEY(I,J).EQ.2) D2=HV(I,J+1)+ETA(I,J)
            D1=HV(I,J)+(ETA(I,J-1)+ETA(I,J))/2.
            IF(MEY(I,J).EQ.1) D1=HV(I,J)+ETA(I,J)
            DO 100 K=1,KZ
            DHUX=(DR*U2(I+1,J,K)-DL*U2(I,J,K))/DX
            DHVY=(D2*V2(I,J+1,K)-D1*V2(I,J,K))/DY
            F(K)=DHUX+DHVY
        100 CONTINUE
            SUM=(F(1)+4.*F(2)+2.*F(3)+4.*F(4))/3.
            DET=-DS*SUM
            ETA3(I,J)=ETA1(I,J)+ABC*DET
            AH3=HB(I,J)+ETA3(I,J)
            IF(AH3.GT.0.) GO TO 200
            ETA3(I,J)=(10.**(-6))-HB(I,J)
            DET=(ETA3(I,J)-ETA1(I,J))/ABC
        200 CONTINUE
            AH=HB(I,J)+ETA(I,J)
            OM(I,J,2)=(-S25*DET+DS*(F(1)+F(2))/B)/AH
            OM(I,J,3)=(-S50*DET+DS*(F(1)/B+F(2)+F(3)/B))/AH
            OM(I,J,4)=(-S75*DET+DS*(F(1)/B+F(2)+F(3)+F(4)/B))/AH
        900 CONTINUE
            RETURN
            END
```

```
*FLOW(1).BNRTIA FOR CREATED ON 4 DEC 79 AT 09:48:40
      C*****************************************************************
      C  COMPUTES NONLINEAR TERMS RX/RY AT INTERIOR HALF-GRID U-/V- POINTS
      C*****************************************************************
            SUBROUTINE BNRTIA(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
           CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
           CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,CB)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            KZ=KN-1
            ABC=CB*DT
            ACE=10.**(-6)
      C*****COMPUTES RX AT INTERIOR U-POINTS*****
            DO 100 I=2,IM
            DO 100 J=1,JM
            IF(MX(I,J).NE.3) GO TO 100
            AH=HU(I,J)+(ETA(I,J)+ETA(I-1,J))/2.
            AH=AMAX1(AH,ACE)
            DET=(ETA3(I,J)-ETA1(I,J)+ETA3(I-1,J)-ETA1(I-1,J))/ABC/2.
            DO 110 K=1,KZ
            D2=HB(I,J)+ETA(I,J)
            D1=HB(I-1,J)+ETA(I-1,J)
            UBAR2=(U2(I,J,K)+U2(I+1,J,K))/2.
            UBAR1=(U2(I,J,K)+U2(I-1,J,K))/2.
            DHUUX=(D2*(UBAR2**2)-D1*(UBAR1**2))/DX
            IF(MY(I,J).EQ.1) GO TO 104
            E2=(ETA(I-1,J+1)+ETA(I,J+1)+ETA(I-1,J)+ETA(I,J))/4.
            E1=(ETA(I,J)+ETA(I,J-1)+ETA(I-1,J-1)+ETA(I-1,J))/4.
            D2=H(I,J+1)+E2
            D1=H(I,J)+E1
            D1=AMAX1(D1,1.)
            D2=AMAX1(D2,1.)
            UBAR2=(U2(I,J,K)+U2(I,J+1,K))/2.
            UBAR1=(U2(I,J,K)+U2(I,J-1,K))/2.
            VBAR2=(V2(I-1,J+1,K)+V2(I,J+1,K))/2.
            VBAR1=(V2(I,J,K)+V2(I-1,J,K))/2.
            DHUVY=(D2*UBAR2*VBAR2-D1*UBAR1*VBAR1)/DY
            GO TO 106
        104 CONTINUE
            E2=(ETA(I-1,J+1)+ETA(I,J+1)+ETA(I-1,J)+ETA(I,J))/4.
            D2=H(I,J+1)+E2
            D2=AMAX1(D2,1.)
            UBAR2=(U2(I,J,K)+U2(I,J+1,K))/2.
            VBAR2=(V2(I-1,J+1,K)+V2(I,J+1,K))/2.
            DHUVY=D2*UBAR2*VBAR2/DY
        106 CONTINUE
            IF(K.EQ.1) GO TO 107
            A3=U2(I,J,K-1)*(OM(I,J,K-1)+OM(I-1,J,K-1))/2.
            A1=U2(I,J,K+1)*(OM(I,J,K+1)+OM(I-1,J,K+1))/2.
            DUOMS=(A3-A1)/DUMS
            DUS=(U2(I,J,K-1)-U2(I,J,K+1))/DUMS
            GO TO 108
        107 A3=0.
            A2=U2(I,J,K+1)*(OM(I,J,K+1)+OM(I-1,J,K+1))/2.
            A1=U2(I,J,K+2)*(OM(I,J,K+2)+OM(I-1,J,K+2))/2.
            DUOMS=(3.*A3-4.*A2+A1)/DUMS
            DUS=(3.*U2(I,J,1)-4.*U2(I,J,2)+U2(I,J,3))/DUMS
        108 SIG=-DS*FLOAT(K-1)
            RX(I,J,K)=DHUUX+DHUVY+AH*DUOMS+(1.+SIG)*DUS*DET
        110 CONTINUE
        100 CONTINUE
      C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      C*****COMPUTES RY AT INTERIOR V-POINTS*****
            DO 200 I=1,IM
            DO 200 J=2,JM
            IF(MY(I,J).NE.3) GO TO 200
            AH=HV(I,J)+(ETA(I,J)+ETA(I,J-1))/2.
             AH=AMAX1(AH,ACE)
            DET=(ETA3(I,J)-ETA1(I,J)+ETA3(I,J-1)-ETA1(I,J-1))/ABC/2.
            DO 210 K=1,KZ
            D2=HB(I,J)+ETA(I,J)
            D1=HB(I,J-1)+ETA(I,J-1)
            VBAR2=(V2(I,J,K)+V2(I,J+1,K))/2.
            VBAR1=(V2(I,J,K)+V2(I,J-1,K))/2.
            DHVVY=(D2*(VBAR2**2)-D1*(VBAR1**2))/DY
```

```
      IF(MX(I,J).EQ.1) GO TO 204
      E2=(ETA(I,J)+ETA(I+1,J)+ETA(I+1,J-1)+ETA(I,J-1))/4.
      E1=(ETA(I,J)+ETA(I,J-1)+ETA(I-1,J-1)+ETA(I-1,J))/4.
      D2=H(I+1,J)+E2
      D1=H(I,J)+E1
      D1=AMAX1(D1,1.)
      D2=AMAX1(D2,1.)
      UBAR2=(U2(I+1,J,K)+U2(I+1,J-1,K))/2.
      UBAR1=(U2(I,J,K)+U2(I,J-1,K))/2.
      VBAR2=(V2(I+1,J,K)+V2(I,J,K))/2.
      VBAR1=(V2(I-1,J,K)+V2(I,J,K))/2.
      DHUVX=(D2*UBAR2*VBAR2-D1*UBAR1*VBAR1)/DX
      GO TO 206
  204 CONTINUE
      E2=(ETA(I,J)+ETA(I+1,J)+ETA(I+1,J-1)+ETA(I,J-1))/4.
      D2=H(I+1,J)+E2
      D2=AMAX1(D2,1.)
      UBAR2=(U2(I+1,J,K)+U2(I+1,J-1,K))/2.
      VBAR2=(V2(I+1,J,K)+V2(I,J,K))/2.
      DHUVX=D2*UBAR2*VBAR2/DX
  206 CONTINUE
      IF(K.EQ.1) GO TO 207
      A3=V2(I,J,K-1)*(OM(I,J,K-1)+OM(I,J-1,K-1))/2.
      A1=V2(I,J,K+1)*(OM(I,J,K+1)+OM(I,J-1,K+1))/2.
      DVOMS=(A3-A1)/DUMS
      DVS=(V2(I,J,K-1)-V2(I,J,K+1))/DUMS
      GO TO 208
  207 A3=0.
      A2=V2(I,J,K+1)*(OM(I,J,K+1)+OM(I,J-1,K+1))/2.
      A1=V2(I,J,K+2)*(OM(I,J,K+2)+OM(I,J-1,K+2))/2.
      DVOMS=(3.*A3-4.*A2+A1)/DUMS
      DVS=(3.*V2(I,J,1)-4.*V2(I,J,2)+V2(I,J,3))/DUMS
  208 SIG=-DS*FLOAT(K-1)
      RY(I,J,K)=DHUVX+DHVVY+AH*DVOMS+(1.+SIG)*DVS*DET
  210 CONTINUE
  200 CONTINUE
C * * * * * * * * * * * * * * * ** * * * * * * * * * * * * * *
      RETURN
      END
```

```
*FLOW(1).ABNR3 FOR CREATED ON 6 DEC 79 AT 10:20:22
C*************************************************************************
C  AT VELOCITY POINTS ON THE OPEN BOUNDARIES, COMPUTES RX ON Y-
C  BOUNDARY, COMPUTE RY ON X-BOUNDARY; RX & RY ARE NONLINEAR TERMS.
C  TIDE HEIGHT JUST OUTSIDE THE OPEN BOUNDARY MUST BE GIVEN
C*************************************************************************
       SUBROUTINE ABNR3(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,EST,
      CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
      CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,CB,
      CTIDE1S,TIDE2S,TIDE3S,TIDE1N,TIDE2N,TIDE3N,STAGE,
      CAMPLIT,PHASE,DPHASE,PERIOD,TSHIFT)
       DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
      CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
      CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
      CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
      CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
      CMEY(IM,JM),
      CTIDE1N(IM),TIDE2N(IM),TIDE3N(IM),
      CTIDE1S(IM),TIDE2S(IM),TIDE3S(IM)
      C,SOM(5)
       KZ=KN-1
       ABC=CB*DT
       ACE=10.**(-6)
       DO 11 I=3,15
       TIDE3N(I)=STAGE+AMPLIT*SIN(6.283/PERIOD*((EST-TSHIFT)-PHASE
      C-DPHASE*(I-1)))
 11    CONTINUE
       DO 12 I=2,11
       TIDE3S(I)=STAGE+AMPLIT*SIN(6.283/PERIOD*((EST-TSHIFT)
      C-DPHASE*(I-1)))
 12    CONTINUE
C*****COMPLTES RY ON NORTH BOUNDARY OF THE ANCHORAGE*****
       J=14
       DO 100 I=3,15
       AH=HV(I,J)+(TIDE2N(I)+ETA(I,J-1))/2.
         AH=AMAX1(AH,ACE)
       DET=(TIDE3N(I)-TIDE1N(I)+ETA3(I,J-1)-ETA1(I,J-1))/ABC/2.
       DO 500 K=1,KN
       SIG=-DS*FLOAT(K-1)
       DET1=(TIDE3N(I)-TIDE1N(I))/ABC
       AH1=HV(I,J)+TIDE2N(I)
       SOM(K)=(SIG/AH1)*DET1
 500   CONTINUE
       DO 210 K=1,KZ
       D2=HV(I,J)+TIDE2N(I)
       D1=HB(I,J-1)+ETA(I,J-1)
       VBAR2=V2(I,J,K)
       VBAR1=(V2(I,J,K)+V2(I,J-1,K))/2.
       DHVVY=(D2*(VBAR2**2)-D1*(VBAR1**2))/DY
       DHUVX=0.
       IF(K.EQ.1) GO TO 207
       A3=V2(I,J,K-1)*(SOM(K-1)+OM(I,J-1,K-1))/2.
       A1=V2(I,J,K+1)*(SOM(K+1)+OM(I,J-1,K+1))/2.
       DVOMS=(A3-A1)/DUMS
       DVS=(V2(I,J,K-1)-V2(I,J,K+1))/DUMS
       GO TO 208
 207   CONTINUE
       A3=0.
       A2=V2(I,J,K+1)*(SOM(K+1)+OM(I,J-1,K+1))/2.
       A1=V2(I,J,K+2)*(SOM(K+2)+OM(I,J-1,K+2))/2.
       DVOMS=(3.*A3-4.*A2+A1)/DUMS
       DVS=(3.*V2(I,J,1)-4.*V2(I,J,2)+V2(I,J,3))/DUMS
   208 SIG=-DS*FLOAT(K-1)
       RY(I,J,K)=DHUVX+DHVVY+AH*DVOMS+(1.+SIG)*DVS*DET
   210 CONTINUE
 100   CONTINUE
C*****COMPLTES RY ON SOUTH BOUNDARY OF THE ANCHORAGE*****
       J=1
       DO 300 I=2,11
       AH=HV(I,J)+(TIDE2S(I)+ETA(I,J))/2.
         AH=AMAX1(AH,ACE)
       DET=(TIDE3S(I)-TIDE1S(I)+ETA3(I,J)-ETA1(I,J))/ABC/2.
       DO 600 K=1,KN
       SIG=-DS*FLOAT(K-1)
       DET1=(TIDE3S(I)-TIDE1S(I))/ABC
       AH1=HV(I,J)+TIDE2S(I)
       SOM(K)=(SIG/AH1)*DET1
 600   CONTINUE
       DO 310 K=1,KZ
```

```
      D2=HV(I,J)+TIDE2S(I)
      D1=HE(I,J)+ETA(I,J)
      VBAR2=V2(I,J,K)
      VBAR1=(V2(I,J,K)+V2(I,J+1,K))/2.
      DHVVY=(D1*(VBAR1**2)-D2*(VBAR2**2))/DY
      DHUVX=0.
      IF(K.EQ.1) GO TO 307
      A3=V2(I,J,K-1)*(SOM(K-1)+OM(I,J,K-1))/2.
      A1=V2(I,J,K+1)*(SOM(K+1)+OM(I,J,K+1))/2.
      DVOMS=(A3-A1)/DUMS
      DVS=(V2(I,J,K-1)-V2(I,J,K+1))/DUMS
      GO TO 308
307   CONTINUE
      A3=0.
      A2=V2(I,J,K+1)*(SOM(K+1)+OM(I,J,K+1))/2.
      A1=V2(I,J,K+2)*(SOM(K+2)+OM(I,J,K+2))/2.
      DVOMS=(3.*A3-4.*A2+A1)/DUMS
      DVS=(3.*V2(I,J,1)-4.*V2(I,J,2)+V2(I,J,3))/DUMS
308   SIG=-DS*FLOAT(K-1)
      RY(I,J,K)=DHUVX+DHVVY+AH*DVOMS+(1.+SIG)*DVS*DET
310   CONTINUE
300   CONTINUE
      RETURN
      END
```

```
*FLOW(1).BVELS FOR CREATED ON 4 DEC 79 AT 09:51:16
C**********************************************************************
C  COMPUTES U3/V3 AT INTERIOR HALF-GRID U-/V- POINTS
C**********************************************************************
      SUBROUTINE BVELS(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
     CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,CB,
     CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
     CAV)
      DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
     CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
     CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
     CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
     CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
     CMEY(IM,JM)
      KZ=KN-1
      ABC=CB*DT
      ACE=10.**(-6)
C*****COMPLTES U3 AT INTERIOR U-POINTS*****
      DO 100 I=2,IM
      DO 100 J=1,JM
      IF(MX(I,J).NE.3) GO TO 100
      AH=HU(I,J)+(ETA(I,J)+ETA(I-1,J))/2.
      AH=AMAX1(AH,ACE)
      AH1=HU(I,J)+(ETA1(I,J)+ETA1(I-1,J))/2.
      AH3=HU(I,J)+(ETA3(I,J)+ETA3(I-1,J))/2.
      G1=AH1/AH3
      G2=AH/AH3
      DEF=ABC*AV/(DS**2)/AH3/AH
      A1=1./(1.+DEF)
      DEX=(ETA(I,J)-ETA(I-1,J))/DX
      A4=G*DEX
      DO 101 K=1,KZ
      A2=FCOR*(V2(I-1,J+1,K)+V2(I,J+1,K)+V2(I,J,K)+V2(I-1,J,K))/4.
      A5=RX(I,J,K)/AH
      IF(K.EQ.1) GO TO 102
      A6=AV*(U2(I,J,K+1)-U1(I,J,K)+U2(I,J,K-1))/(AH*DS)**2
      U3(I,J,K)=A1*(G1*U1(I,J,K)+G2*ABC*(A2-A4-A5+A6))
      GO TO 101
  102 CONTINUE
      A6=2.*AV*(U2(I,J,K+1)-U1(I,J,K)/2.+AH*TAUX*DS/AV)/(AH*DS)**2
      U3(I,J,K)=A1*(G1*U1(I,J,K)+G2*ABC*(A2-A4-A5+A6))
  101 CONTINUE
  100 CONTINUE
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C*****COMPLTES V3 AT INTERIOR V-POINTS*****
      DO 200 I=1,IM
      DO 200 J=2,JM
      IF(MY(I,J).NE.3) GO TO 200
      AH=HV(I,J)+(ETA(I,J)+ETA(I,J-1))/2.
      AH=AMAX1(AH,ACE)
      AH1=HV(I,J)+(ETA1(I,J)+ETA1(I,J-1))/2.
      AH3=HV(I,J)+(ETA3(I,J)+ETA3(I,J-1))/2.
      G1=AH1/AH3
      G2=AH/AH3
      DEF=ABC*AV/(DS**2)/AH3/AH
      A1=1./(1.+DEF)
      DEY=(ETA(I,J)-ETA(I,J-1))/DY
      B4=G*DEY
      DO 201 K=1,KZ
      B2=-FCOR*(U2(I,J,K)+U2(I+1,J,K)+U2(I+1,J-1,K)+U2(I,J-1,K))/4.
      B5=RY(I,J,K)/AH
      IF(K.EQ.1) GO TO 202
      B6=AV*(V2(I,J,K+1)-V1(I,J,K)+V2(I,J,K-1))/(AH*DS)**2
      V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
      GO TO 201
  202 CONTINUE
      B6=2.*AV*(V2(I,J,K+1)-V1(I,J,K)/2.+AH*TAUY*DS/AV)/(AH*DS)**2
      V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
  201 CONTINUE
  200 CONTINUE
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      RETURN
      END
```

```
*FLOW(1).ASAF3 FOR CREATED ON 12 DEC 79 AT 13:47:21
      C*********************************************************************
      C   COMPUTES TIDE HEIGHT AT POINTS JUST OUTSIDE OF OPEN BOUNDARIES, THEN
      C   COMPUTES THE NORMAL VELOCITIES AT BOUNDARY POINTS.
      C*********************************************************************
            SUBROUTINE ASAF3(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
           CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,CB,
           CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
           CTIDE1S,TIDE2S,TIDE3S,EST,STAGE,AV,
           CTIDE1N,TIDE2N,TIDE3N,AMPLIT,PHASE,DPHASE,PERIOD,TSHIFT)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM),
           CTIDE1N(IM),TIDE2N(IM),TIDE3N(IM),
           CTIDE1S(IM),TIDE2S(IM),TIDE3S(IM)
            KZ=KN-1
            ABC=CB*DT
            DO 11 I=3,15
            TIDE3N(I)=STAGE+AMPLIT*SIN(6.283/PERIOD*((EST-TSHIFT)-PHASE
           C-DPHASE*(I-1)))
      11    CONTINUE
            DO 12 I=2,11
            TIDE3S(I)=STAGE+AMPLIT*SIN(6.283/PERIOD*((EST-TSHIFT)
           C-DPHASE*(I-1)))
      12    CONTINUE
      C*****COMPUTES V ON NORTH BOUNDARY*****
            J=14
            DO 300 I=3,15
            AH1=HV(I,J)+(TIDE1N(I)+ETA1(I,J-1))/2.
            AH=HV(I,J)+(TIDE2N(I)+ETA(I,J-1))/2.
            AH=AMAX1(AH,10.**(-6))
            AH3=HV(I,J)+(TIDE3N(I)+ETA3(I,J-1))/2.
            G1=AH1/AH3
            G2=AH/AH3
            DEY=(TIDE2N(I)-ETA(I,J-1))/DY
            DEF=ABC*AV/(DS**2)/AH3/AH
            A1=1./(1.+DEF)
            B4=G*DEY
            DO 301 K=1,KZ
            B2=0.
            B5=RY(I,J,K)/AH
            IF(K.EQ.1) GO TO 302
            B6=AV*(V2(I,J,K+1)-V1(I,J,K)+V2(I,J,K-1))/(AH*DS)**2
            V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
            GO TO 301
        302 CONTINUE
            B6=2.*AV*(V2(I,J,K+1)-V1(I,J,K)/2.+AH*TAUY*DS/AV)/(AH*DS)**2
            V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
        301 CONTINUE
        300 CONTINUE
      C*****UPDATES TIDE HEIGHT*****
            DO 13 I=3,20
            TIDE1N(I)=TIDE2N(I)
      13    TIDE2N(I)=TIDE3N(I)
      C*****COMPUTES V ON SOUTH BOUNDARY*****
            J=1
            DO 100 I=2,11
            AH1=HV(I,J)+(TIDE1S(I)+ETA1(I,J))/2.
            AH=HV(I,J)+(TIDE2S(I)+ETA(I,J))/2.
            AH=AMAX1(AH,10.**(-6))
            AH3=HV(I,J)+(TIDE3S(I)+ETA3(I,J))/2.
            G1=AH1/AH3
            G2=AH/AH3
            DEY=(ETA(I,J)-TIDE2S(I))/DY
            DEF=ABC*AV/(DS**2)/AH3/AH
            A1=1./(1.+DEF)
            B4=G*DEY
            DO 401 K=1,KZ
            B2=0.
            B5=RY(I,J,K)/AH
            IF(K.EQ.1) GO TO 402
            B6=AV*(V2(I,J,K+1)-V1(I,J,K)+V2(I,J,K-1))/(AH*DS)**2
            V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
            GO TO 401
      402   CONTINUE
            B6=2.*AV*(V2(I,J,K+1)-V1(I,J,K)/2.+AH*TAUY*DS/AV)/(AH*DS)**2
```

```
      V3(I,J,K)=A1*(G1*V1(I,J,K)+G2*ABC*(B2-B4-B5+B6))
  401 CONTINUE
  100 CONTINUE
C*****UPDATES TIDE HEIGHT*****
      DO 14 I=2,11
      TIDE1S(I)=TIDE2S(I)
   14 TIDE2S(I)=TIDE3S(I)
      RETURN
      END
```

```
*FLOW(1).GIVENU SYM CREATED ON 14 SEP 79 AT 17:47:43
      C*****SPECIFY VELOCITY AT DISCHARGE,INTAKE,AND RIVER HEAD*********
            SUBROUTINE GIVENU(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,T1,T2,T3,
           CEST)
            DIMENSION V1(IN,JN,KN),V2(IN,JN,KN),V3(IN,JN,KN),
           CT1(IM,JM,KN),T2(IM,JM,KN),T3(IM,JM,KN)
           C,U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN)
            KZ=KN-1
            DO 100 K=1,KZ
            V3(15,1,K)=-10.*COS(6.2832/25.0*(EST-5.0))
            U3(16,1,K)=10.*COS(6.2832/25.0*(EST-5.0))
            V3(14,4,K)=5.-2.*COS(6.2832/12.5*(EST-7.625))
            U3(15,3,K)=5.-2.*COS(6.2832/12.5*(EST-7.625))
            V3(14,8,K)=5.-2.*COS(6.2832/12.5*(EST-7.5))
            U3(15,8,K)=-V3(14,8,K)
        100 CONTINUE
            RETURN
            END
```

```
*FLOW(1).CONV FOR CREATED ON 14 DEC 79 AT 10:43:57
      C**********************************************************************
      C  COMPUTES TC-THE CONVECTIVE TERMS OF T-EQN, AT T-POINTS.
      C**********************************************************************
            SUBROUTINE CONV(IN,JN,KN,IM,JM,DX,DY,DS,DT,DUMX,DUMY,DUMS,
           CSK,RR,BV,U2,UB,V2,VB,T2,TC,HB,HU,HV,OM,ETA1,ETA,ETA3,CB,MEX,
           CMEY,W,DTZ,TEQ)
            DIMENSION U2(IN,JN,KN),V2(IN,JN,KN),UB(IM,JM,KN),VB(IM,JM,KN),
           CT2(IM,JM,KN),HB(IM,JM),HU(IN,JN),HV(IN,JN),OM(IM,JM,KN),
           CETA1(IM,JM),ETA(IM,JM),ETA3(IM,JM),TC(IM,JM,KN),W(IM,JM,KN)
            DIMENSION MEX(IM,JM),MEY(IM,JM),DTZ(IM,JM)
            ABC=CB*DT
            KZ=KN-1
            HK=SK/(RR*BV)
            DO 100 I=1,IM
            DO 100 J=1,JM
            IF(MEX(I,J).EQ.0) GO TO 100
            AH=HB(I,J)+ETA(I,J)
            DET=(ETA3(I,J)-ETA1(I,J))/ABC
            DR=HU(I+1,J)+(ETA(I+1,J)+ETA(I,J))/2.
            IF(MEX(I,J).EQ.2) DR=HU(I+1,J)+ETA(I,J)
            DL=HU(I,J)+(ETA(I-1,J)+ETA(I,J))/2.
            IF(MEX(I,J).EQ.1) DL=HU(I,J)+ETA(I,J)
            D2=HV(I,J+1)+(ETA(I,J+1)+ETA(I,J))/2.
            IF(MEY(I,J).EQ.2) D2=HV(I,J+1)+ETA(I,J)
            D1=HV(I,J)+(ETA(I,J-1)+ETA(I,J))/2.
            IF(MEY(I,J).EQ.1) D1=HV(I,J)+ETA(I,J)
            DTZ(I,J)=-HK*(TEQ-T2(I,J,1))
            DO 110 K=1,KZ
            UR=(U2(I+1,J,K)+U2(I+1,J,K+1))/2.
            UL=(U2(I,J,K)+U2(I,J,K+1))/2.
            TR=T2(I,J,K)
            IF(MEX(I,J).NE.2) TR=(T2(I,J,K)+T2(I+1,J,K))/2.
            TL=T2(I,J,K)
            IF(MEX(I,J).NE.1) TL=(T2(I,J,K)+T2(I-1,J,K))/2.
            DHUTX=(DR*UR*TR-DL*UL*TL)/DX
            VR=(V2(I,J+1,K)+V2(I,J+1,K+1))/2.
            VL=(V2(I,J,K)+V2(I,J,K+1))/2.
            TR=T2(I,J,K)
            IF(MEY(I,J).NE.2) TR=(T2(I,J,K)+T2(I,J+1,K))/2.
            TL=T2(I,J,K)
            IF(MEY(I,J).NE.1) TL=(T2(I,J,K)+T2(I,J-1,K))/2.
            DHVTY=(D2*VR*TR-D1*VL*TL)/DY
            OMR=OM(I,J,K)
            OML=OM(I,J,K+1)
            IF(K.NE.1) TR=(T2(I,J,K)+T2(I,J,K-1))/2.
            IF(K.EQ.1) TR=T2(I,J,K)+(DS/2.)*DTZ(I,J)
            IF(K.NE.KZ) TL=(T2(I,J,K)+T2(I,J,K+1))/2.
            IF(K.EQ.KZ) TL=T2(I,J,K)
            DTOMS=(OMR*TR-OML*TL)/DS
            DTS=(TR-TL)/DS
            SIG=-FLOAT(K-1)*DS-DS/2.
            TC(I,J,K)=DHUTX+DHVTY+AH*DTOMS+DET*DTS*(1.+SIG)
        110 CONTINUE
        100 CONTINUE
            RETURN
            END
```

```
*FLOW(1).TCOMPT FOR CREATED ON 1 APR 80 AT 14:29:38
      C*************************************************************************
      C   COMPUTES T3 BY CTCS+DUFORT-FRANKEL SCHEME TO INTEGRATE T-EQN
      C*************************************************************************
            SUBROUTINE TCOMPT(IN,JN,KN,IM,JM,DX,DY,DS,DT,DUMX,DUMY,DUMS,
           CHB,TC,T1,T2,T3,MEX,MEY,ETA,CB,TAIR,ETA1,ETA3,DTZ,BV,BH,TABN,TABS)
            DIMENSION TC(IM,JM,KN),T1(IM,JM,KN),T2(IM,JM,KN),T3(IM,JM,KN),
           CHB(IM,JM),MEX(IM,JM),ETA(IM,JM),ETA1(IM,JM),ETA3(IM,JM)
            DIMENSION DTZ(IM,JM),MEY(IM,JM)
            KZ=KN-1
            ABC=DT
            DO 100 I=1,IM
            DO 100 J=1,JM
            IF(MEY(I,J).EQ.0) GO TO 100
            AH=HB(I,J)+ETA(I,J)
            AH1=HB(I,J)+ETA1(I,J)
            AH3=HB(I,J)+ETA3(I,J)
            AAC=BV/AH/(DS**2)
            ACC=BH*AH
            DEF=1./(1.+ABC*BV/AH3/AH/(DS**2))
            DEG=1./(1.+ABC*BV/2./AH3/AH/(DS**2))
            DO 200 K=1,KZ
            IF(MEX(I,J).EQ.3)D2TX=(T2(I+1,J,K)+T2(I-1,J,K)-2.*T2(I,J,K))
           C/(DX**2)
            IF(MEX(I,J).EQ.1)D2TX=(T2(I+1,J,K)-T2(I,J,K))/(DX**2)
            IF(MEX(I,J).EQ.2)D2TX=(T2(I-1,J,K)-T2(I,J,K))/(DX**2)
            IF(MEY(I,J).EQ.3)D2TY=(T2(I,J+1,K)+T2(I,J-1,K)-2.*T2(I,J,K))
           C/(DY**2)
            IF(MEY(I,J).EQ.1)D2TY=(T2(I,J+1,K)+TABS-2.*T2(I,J,K))/(DY**2)
            IF(MEY(I,J).EQ.2)D2TY=(T2(I,J-1,K)+TABN-2.*T2(I,J,K))/(DY**2)
            IF(K.EQ.1) GO TO 50
            IF(K.EQ.KZ) GO TO 51
            BAR1=T2(I,J,K+1)-T1(I,J,K)+T2(I,J,K-1)
            BAR2=-TC(I,J,K)+AAC*BAR1+ACC*(D2TX+D2TY)
            T3(I,J,K)=DEF*(AH*T2(I,J,K)+ABC*BAR2)/AH3
            GO TO 200
         50 CONTINUE
            BAR1=T2(I,J,K+1)-T1(I,J,K)/2.+DS*DTZ(I,J)
            BAR2=-TC(I,J,K)+AAC*BAR1+ACC*(D2TX+D2TY)
            T3(I,J,K)=DEG*(AH*T2(I,J,K)+ABC*BAR2)/AH3
            GO TO 200
         51 CONTINUE
            BAR1=-T1(I,J,K)/2.+T2(I,J,K-1)
            BAR2=-TC(I,J,K)+AAC*BAR1+ACC*(D2TX+D2TY)
            T3(I,J,K)=DEG*(AH*T2(I,J,K)+ABC*BAR2)/AH3
        200 CONTINUE
        100 CONTINUE
            RETURN
            END
```

```
*FLOW(1).GIVENT SYM CREATED ON 6 DEC 79 AT 10:22:11
      C*****SPECIFY TEMPERATURE AT DISCHARGE AND RIVER HEAD*****************
            SUBROUTINE GIVENT(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,T1,T2,T3,
           CEST)
            DIMENSION V1(IN,JN,KN),V2(IN,JN,KN),V3(IN,JN,KN),
           CT1(IM,JM,KN),T2(IM,JM,KN),T3(IM,JM,KN)
           C,U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN)
            DO 200 K=1,KN
            T3(14,8,K)=29.4+.4*SIN(.2618*(EST-12.))
            T3(15,1,K)=26.9+.5*SIN(.2618*(EST-12.))
      200   CONTINUE
            RETURN
            END
```

```
*FLOW(1).WCAL FOR CREATED ON 4 DEC 79 AT 11:35:41
      C*****************************************************************
      C  CALCULATES ACTUAL VERTICAL VELOCITY W
      C*****************************************************************
            SUBROUTINE WCAL(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
           CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,
           CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NRLOK,TTOT)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAR(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            KZ=KN-1
            DO 900 I=1,IM
            DO 900 J=1,JM
            IF(MEX(I,J).EQ.0) GO TO 900
            DET=(ETA3(I,J)-ETA1(I,J))/2./DT
            AH=HB(I,J)+ETA(I,J)
            IF(MEX(I,J).EQ.3) GO TO 15
            IF(MEX(I,J).EQ.1) GO TO 1
            IF(MEX(I,J).EQ.2) GO TO 2
         15 CONTINUE
            DEX=(ETA(I+1,J)-ETA(I-1,J))/DUMX
            DAHX=(HB(I+1,J)+ETA(I+1,J)-HB(I-1,J)-ETA(I-1,J))/DUMX
            GO TO 3
          1 DEX=(ETA(I+1,J)-ETA(I,J))/DX
            E2=(ETA(I+1,J)+ETA(I,J))/2.
            E1=ETA(I+1,J)-1.5*(ETA(I+1,J)-ETA(I,J))
            DAHX=(HU(I+1,J)+E2-HU(I,J)-E1)/DX
            GO TO 3
          2 DEX=(ETA(I,J)-ETA(I-1,J))/DX
            E2=ETA(I-1,J)+1.5*(ETA(I,J)-ETA(I-1,J))
            E1=(ETA(I-1,J)+ETA(I,J))/2.
            DAHX=(HU(I,J)+E2-HU(I-1,J)-E1)/DX
            GO TO 3
          3 CONTINUE
            IF(MEY(I,J).EQ.3) GO TO 16
            IF(MEY(I,J).EQ.1) GO TO 4
            IF(MEY(I,J).EQ.2) GO TO 5
         16 CONTINUE
            DEY=(ETA(I,J+1)-ETA(I,J-1))/DUMY
            DAHY=(HB(I,J+1)+ETA(I,J+1)-HB(I,J)-ETA(I,J))/DUMY
            GO TO 6
          4 DEY=(ETA(I,J+1)-ETA(I,J))/DY
            E2=(ETA(I,J+1)+ETA(I,J))/2.
            E1=ETA(I,J+1)-1.5*(ETA(I,J+1)-ETA(I,J))
            DAHY=(HV(I,J+1)+E2-HV(I,J)-E1)/DY
            GO TO 6
          5 DEY=(ETA(I,J)-ETA(I,J-1))/DY
            E2=ETA(I,J-1)+1.5*(ETA(I,J)-ETA(I,J-1))
            E1=(ETA(I,J-1)+ETA(I,J))/2.
            DAHY=(HV(I,J)+E2-HV(I,J-1)-E1)/DY
            GO TO 6
          6 CONTINUE
            DO 444 K=1,KZ
            W(I,J,K)=OM(I,J,K)*AH-FLOAT(K-1)*DS*(DET+UB(I,J,K)*DAHX+VB(I,J,K)*
           CDAHY)+DET+UB(I,J,K)*DEX+VB(I,J,K)*DEY
        444 CONTINUE
        900 CONTINUE
            RETURN
            END
```

```
*FLOW(1).ANCPR FOR CREATED ON 3 DEC 79 AT 20:16:46
        C*****************************************************************
        C  PRINTS ETA, RESULTANT VELOCITY, AND T AT 4 FIXED LOCATIONS CONTINU-
        C  OUSLY, THUS SHOWS THE FLOW AND TEMPERATURE DEVELOPMENT AT THESE PTS.
        C*****************************************************************
              SUBROUTINE ANCPR(IM,JM,KN,UB,VB,E,TTOT,R,TH,T2)
              DIMENSION UB(IM,JM,KN),VB(IM,JM,KN),E(IM,JM),T2(IM,JM,KN)
              Q=57.3
              U=UB(6,1,1)
              V=VB(6,1,1)
              X1=SQRT(U**2+V**2)
              CALL ZZ1(U,V,ZED)
              Z1=Q*(TH+ZED)
              IF(Z1.LT.0.) Z1=Z1+360.
              S1=E(6,1)
              TM1=T2(6,1,1)
              U=UB(12,4,1)
              V=VB(12,4,1)
              X2=SQRT(U**2+V**2)
              CALL ZZ1(U,V,ZED)
              Z2=Q*(TH+ZED)
              IF(Z2.LT.0.) Z2=Z2+360.
              S2=E(12,4)
              TM2=T2(12,4,1)
              U=UB(8,12,1)
              V=VB(8,12,1)
              X3=SQRT(U**2+V**2)
              CALL ZZ1(U,V,ZED)
              Z3=Q*(TH+ZED)
              IF(Z3.LT.0.) Z3=Z3+360.
              S3=E(8,12)
              TM3=T2(8,12,1)
              U=UB(14,9,1)
              V=VB(14,9,1)
              X4=SQRT(U**2+V**2)
              CALL ZZ1(U,V,ZED)
              Z4=Q*(TH+ZED)
              IF(Z4.LT.0.) Z4=Z4+360.
              S4=E(14,9)
              TM4=T2(14,9,1)
              WRITE(6,1) TTOT,S1,X1,Z1,TM1,S2,X2,Z2,TM2,S3,X3,Z3,TM3,
             CS4,X4,Z4,TM4
            1 FORMAT(1X,F7.0,2F6.1,F5.0,3F6.1,F5.0,3F6.1,F5.0,F6.1,
             C2F6.1,F5.0,F6.1)
              RETURN
              END
```

```
*FLOW(1).TPRLOK FOR CREATED ON 4 DEC 79 AT 11:31:18
      C*****************************************************************
      C  PRINTS CUT HOURLY RESULTS OF HORIZONTAL RESULTANT VEL, W-VEL, AND T
      C  AT 4 LEVELS, AND THE SURFACE ELEVATION ETA
      C*****************************************************************
            SUBROUTINE TPRLOK(IN,JN,KN,IM,JM,U1,U2,U3,V1,V2,V3,W,OM,ETA1,
           CETA,ETA3,RX,RY,MX,MY,MAR,H,HB,ELEV,UB,VB,HU,HV,MEX,MEY,T2,
           CDUMX,DUMY,DUMS,DX,DY,DS,DT,FCOR,TAUX,TAUY,G,NCASE,NBLOK,TTOT,
           CKVEL,KTEMP,VR,ANG,THETA)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),U3(IN,JN,KN),V1(IN,JN,KN),
           CV2(IN,JN,KN),V3(IN,JN,KN),W(IM,JM,KN),OM(IM,JM,KN),ETA1(IM,JM),
           CETA(IM,JM),ETA3(IM,JM),RX(IN,JN,KN),RY(IN,JN,KN),MX(IN,JN),
           CMY(IN,JN),MAP(IN,JN),H(IN,JN),HB(IM,JM),ELEV(IN,JN),
           CUB(IM,JM,KN),VB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),
           CMEY(IM,JM)
            DIMENSION VR(JM),ANG(JM),T2(IM,JM,KN)
            KUV=0
            KPROF=0
            QQ=57.3
            KZ=KN-1
            IF(KVEL.EQ.0) GO TO 88
            IF(KLV.EQ.0) GO TO 200
            DO 2 K=1,KZ
            WRITE(6,500) K
            DO 3 I=1,IM
            WRITE(6,501) I,(UB(I,J,K),J=1,JM)
          3 WRITE(6,502) (VB(I,J,K),J=1,JM)
          2 CONTINUE
        200 CONTINUE
            DO 11 K=1,KZ
            WRITE(6,700) K
            DO 12 I=1,IM
            DO 1 J=1,JM
            VR(J)=10.**9
          1 ANG(J)=10.**9
            DO 13 J=1,JM
            IF(MEY(I,J).EQ.0) GO TO 13
            U=UB(I,J,K)
            V=VB(I,J,K)
            VR(J)=SQRT(U**2+V**2)
            CALL ZZ1(U,V,ZED)
            ANG(J)=QQ*(THETA+ZED)
            IF(ANG(J).LT.0.) ANG(J)=360.+ANG(J)
         13 CONTINUE
            WRITE(6,701) I,(VR(J),J=1,JM)
            WRITE(6,702) (ANG(J),J=1,JM)
         12 CONTINUE
         11 CONTINUE
            DO 4 K=1,KZ
            WRITE(6,503) K
            DO 5 I=1,IM
          5 WRITE(6,504) I,(W(I,J,K),J=1,JM)
          4 CONTINUE
            WRITE(6,505)
            DO 6 I=1,IM
          6 WRITE(6,506) I,(ETA(I,J),J=1,JM)
            IF(KPROF.EQ.0) GO TO 100
            WRITE(6,508)
            DO 10 J=1,JM
            WRITE(6,609) J
            DO 10 K=1,KZ
            WRITE(6,507) K,(UB(I,J,K),I=1,IM)
         10 CONTINUE
            WRITE(6,509)
            DO 20 I=1,IM
            WRITE(6,601) I
            DO 20 K=1,KZ
            WRITE(6,507) K,(VB(I,J,K),J=1,JM)
         20 CONTINUE
        100 CONTINUE
         88 CONTINUE
            IF(KTEMP.EQ.0) GO TO 89
            DO 31 K=1,KN
            WRITE(6,550) K
        550 FORMAT(1X,'TEMPERATURES, DEG C.     K=',I4)
            DO 32 I=1,IM
            IF(K.EQ.1) WRITE(6,552) I,(T2(I,J,K),J=1,JM)
            IF(K.GT.1) WRITE(6,551) I,(T2(I,J,K),J=1,JM)
         32 CONTINUE
```

```
552 FORMAT(1X,'I=',I4,'    TEMP=',14F7.2//)
551 FORMAT(1X,'I=',I4,'    TEMP=',15F7.2)
 31 CONTINUE
 89 CONTINUE
500 FORMAT(1X,'U,V VELOCITIES, CM/SEC,     K=',I4/)
501 FORMAT(1X,'I=',I4,'  U=',15F7.1)
502 FORMAT(1X,8X,'V=',15F7.1/)
503 FORMAT(1H0,'W VELOCITIES, CM/SEC,     K=',I4/)
504 FORMAT(1X,'I=',I4,'  W=',15F7.4)
505 FORMAT(1X,'SURFACE ELEVATIONS, ETA, CM'/)
506 FORMAT(1X,'I=',I4,'  ETA=',14F7.1//)
507 FORMAT(1X,'K=',I4,15F8.1)
508 FORMAT(1X,'VERTICAL PROFILES OF U VELOCITIES')
509 FORMAT(1X,'VERTICAL PROFILES OF V VELOCITIES')
600 FORMAT('1')
601 FORMAT(1H0,'I=',I4//)
609 FORMAT(1H0,'J=',I4//)
700 FORMAT(1X,'RESULTANT VELOCITIES AND DIRS,     K=',I4/)
701 FORMAT(1X,'I=',I4,'  VRES=',15F7.1)
702 FORMAT(1X,12X,15F7.0/)
    RETURN
    END
```

```
*FLOW(1).STORET FOR CREATED ON 4 DEC 79 AT 11:22:36
      C*******************************************************************
      C  STORES HOURLY RESULT ONTO TAPE FOR LATER RUN OR PLOTTING
      C*******************************************************************
            SUBROUTINE STORET(IN,JN,KN,IM,JM,U1,U2,V1,V2,W,ETA1,T1,T2,TB,
           CETA,MX,MY,MAR,H,HB,UB,VB,HU,HV,MEX,MEY,NCASE,NBLOK,TTOT,DT,
           CTIDE1N,TIDE2N,TIDE3N,TIDE1S,TIDE2S,TIDE3S,STAGE,EST,
           CAMPLIT,PHASE,DPHASE,PERIOD)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),V1(IN,JN,KN),V2(IN,JN,KN),
           CW(IM,JM,KN),ETA1(IM,JM),ETA(IM,JM),MX(IN,JN),MY(IN,JN),MAR(IN,JN),
           CH(IN,JN),HB(IM,JM),UB(IM,JM,KN),VB(IM,JM,KN),
           CHU(IN,JN),HV(IN,JN),MEX(IM,JM),MEY(IM,JM),TB(IM,JM,KN)
            DIMENSION T1(IM,JM,KN),TIDE1N(IM),TIDE2N(IM),TIDE3N(IM)
            DIMENSION T2(IM,JM,KN),TIDE1S(IM),TIDE2S(IM),TIDE3S(IM)
            NBLOK=NBLOK+1
            WRITE (8) NBLOK
            WRITE (8) (((U1(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((U2(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((V1(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C(((V2(I,J,K),K=1,KN),J=1,JN),I=1,IN),
           C((MAR(I,J),J=1,JN),I=1,IN),
           C((MX (I,J),J=1,JN),I=1,IN),
           C((MY (I,J),J=1,JN),I=1,IN),
           C((HU (I,J),J=1,JN),I=1,IN),
           C((HV (I,J),J=1,JN),I=1,IN),
           C((H(I,J),J=1,JN),I=1,IN)
            WRITE (8) ((( W(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((UB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((VB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((T1(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((T2(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C(((TB(I,J,K),K=1,KN),J=1,JM),I=1,IM),
           C((ETA1(I,J),J=1,JM),I=1,IM),
           C((ETA(I,J),J=1,JM),I=1,IM),
           C((MEX(I,J),J=1,JM),I=1,IM),
           C((MEY(I,J),J=1,JM),I=1,IM),
           C((HB (I,J),J=1,JM),I=1,IM),
           C(TIDE1N(I),I=1,IM),
           C(TIDE2N(I),I=1,IM),
           C(TIDE3N(I),I=1,IM),
           C(TIDE1S(I),I=1,IM),
           C(TIDE2S(I),I=1,IM),
           C(TIDE3S(I),I=1,IM)
            WRITE (8) TTOT,NCASE,DT,STAGE,EST,AMPLIT,PHASE,DPHASE,
           CPERIOD
            END FILE 8
            WRITE(6,112) TTOT,NCASE,NBLOK
        112 FORMAT(1X,'DATA RECORDED ON TAPE,    TTOT=',F10.0,
           C'    CASE NBR=',I5,'    BLOK NBR=',I5)
            RETURN
            END
```

```
*FLOW(1).ZZ1 FOR CREATED ON 4 DEC 79 AT 11:39:05
      C*****DETERMINES THE ANGLE OF RESULTANT VELOCITY ******
            SUBROUTINE ZZ1(U,V,ZED)
            IF(ABS(V).LT..0001) GO TO 10
            ZED=ATAN2(U,V)
            GO TO 100
         10 CONTINUE
            IF(U.GT.0.) ZED=1.571
            IF(U.LE.0.) ZED=4.713
        100 RETURN
            END
```

```
*FLOW(1).AMAIN SYM CREATED ON 25 APR 79 AT 10:28:45
0,0,0,1,1,1,1,0,0,1,1,1,0,0
1,1,1,1,1,1,1,1,1,1,1,1,0,0
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1
0,0,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,0,1,1,1,1,1,1,1,1
1,1,1,1,0,0,0,1,1,1,1,1,1,1
1,1,1,0,0,0,0,0,1,1,1,1,1,1
0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0,1.0,1.0,0.0,0.0
1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,1.0,0.0,0.0
12.,11.,8.0,8.0,3.0,3.0,5.0,5.0,3.0,2.0,2.0,2.0,2.0,4.0
8.0,8.0,9.0,10.,13.,8.0,8.0,8.0,6.0,3.0,3.0,4.0,5.0,7.0
10.,8.0,8.0,8.0,10.,9.0,9.0,9.0,8.0,6.0,4.0,6.0,7.0,8.0
9.0,9.0,8.0,8.0,9.0,8.0,9.0,8.0,8.0,7.0,6.0,7.0,8.0,9.0
7.0,7.0,7.0,7.0,7.0,8.0,8.0,8.0,9.0,8.0,7.0,7.0,8.0,9.0
5.0,5.0,5.0,5.0,3.0,3.0,5.0,9.0,8.0,8.0,7.0,7.0,7.0,8.0
3.0,4.0,3.0,5.0,2.0,2.0,2.0,7.0,8.0,8.0,7.0,7.0,7.0,8.0
2.0,3.0,2.0,4.0,4.0,2.0,4.0,6.0,8.0,8.0,7.0,7.0,7.0,6.0
2.0,2.0,1.0,4.0,4.0,3.0,2.0,5.0,7.0,7.0,7.0,7.0,7.0,7.0
1.0,1.0,1.0,3.0,4.0,2.0,2.0,2.0,6.0,7.0,7.0,6.0,7.0,6.0
0.0,0.0,3.0,3.0,3.0,1.0,1.0,2.0,3.0,6.0,5.0,5.0,5.0,5.0
1.0,3.0,3.0,3.0,3.0,0.0,1.0,2.0,6.0,3.0,3.0,4.0,4.0,4.0
2.0,3.0,3.0,5.0,0.0,0.0,0.0,6.0,2.0,2.0,3.0,4.0,4.0,3.0
3.0,2.0,1.0,0.0,0.0,0.0,0.0,0.0,1.0,1.0,2.0,2.0,2.0,2.0
0,0,0,1,1,1,0,0,0,1,1,0,0
1,1,1,3,3,3,1,1,1,3,3,0,0
3,3,3,3,3,3,3,3,3,3,3,1,1
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
2,2,2,3,3,3,3,3,3,3,3,3,3
0,0,0,3,2,2,3,3,3,3,3,3,3
0,0,1,2,0,0,2,3,3,3,3,3,3
1,1,2,0,0,0,0,2,3,3,3,3,3
2,2,0,0,0,0,0,0,2,2,2,2,2
0,0,0,1,3,2,0,0,0,1,2,0,0
1,3,3,3,3,3,3,3,3,3,2,0,0
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
1,3,3,3,3,3,3,3,3,3,3,3,2
0,0,0,1,3,3,3,3,3,3,3,3,2
0,0,1,2,0,0,1,3,3,3,3,3,2
1,3,2,0,0,0,0,1,3,3,3,3,2
1,2,0,0.0,0,0,0,1,3,3,3,2
0,0,0,1,1,1,0,0,0,1,1,0,0
1,1,1,3,3,3,1,1,1,3,3,0,0
3,3,3,3,3,3,3,3,3,3,3,1,1
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
3,3,3,3,3,3,3,3,3,3,3,3,3
2,2,2,3,3,3,3,3,3,3,3,3,3
0,0,1,3,2,2,3,3,3,3,3,3,3
1,1,3,2,0,0,2,3,3,3,3,3,3
3,3,2,0,0,0,0,2,3,3,3,3,3
2,2,0,0,0,0,0,0,2,2,2,2,2
```

```
79        0,0,0,1,3,3,2,0,0,1,3,2,0,0
80        1,3,3,3,3,3,3,3,3,3,3,2,0,0
81        1,3,3,3,3,3,3,3,3,3,3,3,3,2
82        1,3,3,3,3,3,3,3,3,3,3,3,3,2
83        1,3,3,3,3,3,3,3,3,3,3,3,3,2
84        1,3,3,3,3,3,3,3,3,3,3,3,3,2
85        1,3,3,3,3,3,3,3,3,3,3,3,3,2
86        1,3,3,3,3,3,3,3,3,3,3,3,3,2
87        1,3,3,3,3,3,3,3,3,3,3,3,3,2
88        1,3,3,3,3,3,3,3,3,3,3,3,3,2
89        1,3,3,3,3,3,3,3,3,3,3,3,3,2
90        0,0,0,1,3,3,3,3,3,3,3,3,3,2
91        0,0,1,3,2,0,1,3,3,3,3,3,3,2
92        1,3,3,2,0,0,0,1,3,3,3,3,3,2
93        1,3,2,0,0,0,0,0,1,3,3,3,3,2
```

```
*FLO*(1).APER1 SYM CREATED ON 19 JUL 79 AT 11:16:24
1        2,1
2        12,1
3        12,4
4        13,4
5        13,3
6        14,3
7        14,1
8        16,1
9        16,3
10       15,3
11       15,4
12       14,4
13       14,5
14       13,5
15       13,7
16       14,7
17       14,8
18       15,8
19       15,9
20       16,9
21       16,14
22       3,14
23       3,12
24       1,12
25       1,10
26       2,10
27       2,7
28       1,7
29       1,4
30       2,4
31       2,1
```

```
*FLOW(1).C2007 SYM CREATED ON 20 JUN 80 AT 11:13:26
 1        25.0,.82,357.6,90.0,.05,26.6
 2        26.1,.78,357.6,90.0,.20,26.7
 3        27.2,.74,491.7,110.,.30,26.8
 4        28.8,.70,581.1,110.,.30,26.9
 5        29.8,.66,447.0,110.,.40,27.0
 6        29.4,.62,357.6,110.,.40,27.0
 7        30.0,.59,357.6,90.0,.60,27.0
 8        30.6,.58,402.3,90.0,.50,27.0
 9        30.6,.57,402.3,110.,.55,27.0
10        30.0,.57,402.3,100.,.40,27.1
11        30.0,.56,402.3,100.,.30,27.2
12        29.4,.55,312.9,110.,.20,27.1
13        28.8,.53,312.9,90.0,.15,27.0
14        28.0,.50,223.5,90.0,.05,27.0
15        27.2,.49,223.5,90.0,.00,27.0
16        26.8,.48,223.5,90.0,.00,27.0
```

```
*FLOW(I).HDATA SYM CREATED ON 17 MAY 79 AT 15:38:59
 1        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
 2        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
 3        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.5,27.0
 4        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.5,27.6,27.0,27.0
 5        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,27.8,27.0,27.0,27.0
 6        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,28.0,27.0,27.0,27.0
 7        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,28.2,29.0,29.6,29.6
 8        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.6,28.2,29.6,29.6
 9        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.7,28.4,28.5
10        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.6,28.3,28.2
11        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,26.2,27.8
12        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.5,27.3
13        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
```

```
I*FLOW(1).EDATA SYM CREATED ON 18 JAN 80 AT 13:10:50
  1        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
  2        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
  3        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.1,27.0,27.0,27.2,27.0
  4        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.5,27.6,27.5,27.0,27.0
  5        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.9,28.3,27.9,27.0,27.0,27.0
  6        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.8,28.7,28.7,27.0,27.0,27.0
  7        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.7,28.7,28.9,29.0,29.6,29.6
  8        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.6,28.6,29.1,29.3,29.6,29.6
  9        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.5,28.5,29.2,29.4,29.0,28.7
 10        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.8,28.7,28.8,28.7,27.7
 11        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,27.8,28.3,27.7,27.3
 12        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.4,27.3,27.0
 13        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.0,27.0
```

```
*FLOW(I).LDATA SYM CREATED ON 18 MAY 79 AT 13:58:37
 1        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
 2        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
 3        27.0,27.0,27.0,27.0,27.2,27.2,27.2,27.2,27.0,27.0,27.0,27.0,27.6,27.6,27.0
 4        27.0,27.0,27.0,27.0,27.3,27.7,27.6,27.5,27.[illegible],27.4,27.4,27.5,27.7,27.0,27.0
 5        27.0,27.0,27.0,27.0,27.3,27.6,27.7,27.9,27.8,27.7,27.8,27.8,27.0,27.0,27.0
 6        27.0,27.0,27.0,27.0,27.2,27.6,27.9,28.0,28.[illegible],28.2,29.4,28.6,27.0,27.0,27.0
 7        27.0,27.0,27.0,27.0,27.0,27.5,28.1,28.4,28.[illegible],28.7,29.1,29.5,29.0,30.2,30.2
 8        27.0,27.0,27.0,27.0,27.0,27.3,28.0,28.5,29.0,29.4,29.9,30.0,30.1,30.2,30.2
 9        27.0,27.0,27.0,27.0,27.0,27.0,27.3,28.1,28.6,29.3,29.5,29.7,29.2,28.5,28.0
10        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,28.0,28.5,28.7,28.4,28.2,28.0,27.8
11        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.[illegible],27.6,27.8,27.6,27.5,27.4,27.3
12        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.[illegible],27.4,27.3,27.2,27.2,27.1
13        27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
```

```
*FLOW(1).FDATA SYM CREATED ON 20 JUN 80 AT 10:47:16
 1         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2
 2         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.2
 3         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,27.3,27.0
 4         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.3,27.4,27.4,27.0,27.0
 5         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.3,27.4,27.4,27.0,27.0,27.0
 6         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.5,27.9,27.8,27.0,27.0,27.0
 7         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.3,27.7,28.3,28.7,28.7,29.5,29.5
 8         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.2,27.8,28.5,29.1,29.3,29.5,29.5
 9         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.6,28.3,28.9,29.1,28.9,27.3
10         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.4,27.8,28.2,29.0,27.6,27.1
11         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.5,27.7,27.6,27.3,27.0
12         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.1,27.2,27.2,27.0,27.0
13         27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0,27.0
```

```
*FLOW(1).PLOTMN FOR CREATED ON 11 DEC 79 AT 18:30:09
      C*****************************************************************
      C  MAIN PROGRAM FOR ANALYSING AND PLOTTING THE RESULTS OBTAINED BY ANCMN
      C*****************************************************************
            PARAMETER IN=16,JN=14,KN=5,IM=15,JM=13,NCN=31,MCY=15
            INTEGER RGRID
            DIMENSION IBUF(1000),TL(4),TH(4),TI(4),Q1(MCY),Q2(MCY),Q3(MCY),
           CQ4(MCY),Q5(MCY),TA1(4)
            DIMENSION IPLOT(5),NPLOUT(MCY),NSTAND(MCY)
            DIMENSION U1(IN,JN,KN),U2(IN,JN,KN),V1(IN,JN,KN),V2(IN,JN,KN),
           CW(IM,JM,KN),ETA1(IM,JM),ETA(IM,JM),MX(IN,JN),MY(IN,JN),
           CMAR(IN,JN),H(IN,JN),HB(IM,JM),UP(IM,JM,KN),MIDEW(4),
           CVB(IM,JM,KN),HU(IN,JN),HV(IN,JN),MEX(IM,JM),MEY(IM,JM)
            DIMENSION T1(IM,JM,KN),T2(IM,JM,KN),TB(IM,JM,KN)
            DIMENSION TIDE1N(IM),TIDE2N(IM),TIDE3N(IM),TIR(IM,JM,4)
            DIMENSION TIDE1S(IM),TIDE2S(IM),TIDE3S(IM),RGRID(IM,JM)
            DIMENSION XX(NCN),YY(NCN),ETAL(MCY),ETAH(MCY),ETAINT(MCY)
            DATA (MIDEW(I),I=1,4)/'EBB','LOW','FLOD','HIGH'/
            READ 2,IPLOIR,IPLOCM,NIR,KPLOT,NCY
            READ 2,(TL(I),I=1,NIR)
            READ 2,(TH(I),I=1,NIR)
            READ 2,(TI(I),I=1,NIR)
            READ 2,(TA1(I),I=1,NIR)
            READ 2,(Q1(I),I=1,NIR)
            READ 2,(Q2(I),I=1,NIR)
            READ 2,(Q3(I),I=1,NIR)
            READ 2,(Q4(I),I=1,NIR)
            READ 2,(Q5(I),I=1,NIR)
            READ 2,Q6
            READ 2,(IPLOT(I),I=1,5)
            READ 2,(NPLOUT(I),I=1,NCY)
            READ 2,(NSTAND(I),I=1,NCY)
            READ 2,(ETAL(I),I=1,NIR)
            READ 2,(ETAH(I),I=1,NIR)
            READ 2,(ETAINT(I),I=1,NIR)
            READ 2, XL
            READ 2, DX,DY
            DO 9 NC=1,NCN
            READ 2, I,J
            XX(NC)=FLOAT(I-1)
            YY(NC)=FLOAT(J-1)
          9 CONTINUE
          2 FORMAT ( )
            KZ=KN-1
            FACT=1.
            CALL FACTOR(FACT)
            AKN=FLOAT(KN-1)
            DELZ=100.
            ARMIN=.04
            ARMAX=.15
            DS=.25
            ALREF=800000.
            PHI=0.
            COSF=COS(PHI)
            SINF=SIN(PHI)
            XSCALE=XL/DX/FLOAT(IN-1)
            YSCALE=XSCALE
            ZSCALE=.003
            USCALE=.0075
            VSCALE=USCALE
            WSCALE=3.
            XSC=1./2.54/XSCALE
            YSC=1./2.54/YSCALE
            ZSC=1./2.54/ZSCALE
            USC=1./2.54/USCALE
            VSC=1./2.54/VSCALE
            WSC=1./2.54/WSCALE
            WRITE(6,100) XSC,YSC,ZSC
        100 FORMAT(1X,'XSC=',F12.2,'  YSC=',F12.2,'  ZSC=',F12.2)
            WRITE(6,200) USC,VSC,WSC
        200 FORMAT(1X,'USC=',F12.2,'  VSC=',F12.2,'  WSC=',F12.2)
            CALL PLOTS(IBUF,1000,11)
            NPLT=0
            CALL PLOT(3.,3.,-3)
      C*****************************************************************
      C*****PLOTS ISOTHERMS FOR THE IR OBTAINED & MAN INTERPOLATED T-FIELD
            DO 8 N=1,NIR
            DO 8 J=1,JM
            READ 2,(TIR(I,J,N),I=1,IM)
```

```
8       CONTINUE
        IF(IFLOIR.EQ.0) GO TO 7
        DO 10 N=1,NIR
        P1=Q1(N)
        P2=Q2(N)
        P3=Q3(N)
        P4=Q4(N)
        P5=Q5(N)
        ZLIT=TL(N)
        ZBIG=TH(N)
        SAMCON=ZLIT
        TINT=TI(N)
        CALL PLOT(10.,0.,-3)
        DO 11 J=1,JM
        DO 11 I=1,IM
11      ETA(I,J)=TIR(I,J,N)
        CALL ECHKON(ETA,IM,JM,1,IM,1,JM,4.8,5.6,.04,SAMCON,TINT,
       CRGRID,IM,JM,ZLIT,ZBIG,0.,0.,0.,0.,0,0,.07,1.,NPLT,IN,JN,KN,
       CNCN,DX,DY,XSCALE,YSCALE,COSF,SINF,XX,YY)
        CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
        CALL CAPTN5(N,MIDEW)
12      CALL CAPTN3(N)
        CALL PLOT(0.,0.,3)
10      CONTINUE
7       IF(IFLOCM.EQ.0) GO TO 14
C*****************************************************************
C*****ANALYZES AND PLOTS THE HOURLY RESULTS OBTAINED FROM CALCULATION
C *** READS IN HOURLY RESULT AND MAKES THE CHOICE OF WHICH TO BE PLOTTED
        DO 1000 NCTR=1,NCY
        CALL READT(IN,JN,KN,IM,JM,U1,U2,V1,V2,W,ETA1,T1,T2,TB,
       CETA,MX,MY,MAR,H,HB,UB,VB,HU,HV,MEX,MEY,NCASE,NBLOK,TTOT,DT,
       CTIDE1N,TIDE2N,TIDE3N,TIDE1S,TIDE2S,TIDE3S,STAGE,EST,
       CAMPLIT,PHASE,OPHASE,PERIOD)
        IF((NPLOUT(NCTR).EQ.0).AND.(NSTAND(NCTR).EQ.0)) GO TO 1000
        N=NSTAND(NCTR)
        NG=NFLOUT(NCTR)
        DO 3006 I=1,IM
        DO 3006 J=1,JM
        IF(MEX(I,J).EQ.0) TB(I,J,1)=TA1(N)
 3006   CONTINUE
        P1=EST
        P2=Q2(NG)
        P3=Q3(NG)
        P4=Q4(NG)
        P5=Q5(NG)
        SUM=0.
        K=KPLOT
        DO 995 J=1,JM
        DO 995 I=1,IM
        SUM=SUM+(TB(I,J,K)-TIR(I,J,N))**2
995     CONTINUE
        SUM=SUM/(IM*JM)
        SUM=SQRT(SUM)
        WRITE(6,992) EST,N,SUM
        IF(NFLOUT(NCTR).EQ.0) GO TO 1000
        CALL PLOT(10.,0.,-3)
        HMAX=0.
        DO 5 I=1,IM
        DO 5 J=1,JM
        IF(MEX(I,J).EQ.0) GO TO 5
        AH=HE(I,J)+ETA(I,J)
        HMAX=AMAX1(AH,HMAX)
    5   CONTINUE
        KQ=1+INT(HMAX/DELZ)
C*****PLOTS RESULTANT VEL OF U && V ON NEARLY HORIZONTAL SIGMA PLANE
        IF(IFLOT(1).EQ.0) GO TO 1001
        CALL PLOTUV(IN,JN,KN,IM,JM,UR,VR,MEX,MEY,NCN,
       CXX,YY,AKN,DELZ,ZPRR,ARMIN,ARMAX,COSF,SINF,KQ,HB,UDUM1,
       CUDUM,XL,UMAX,DX,DY,HMAX,NPLT,KPLOT,
       CXSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA)
        CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
        CALL CAPTN5(N,MIDEW)
        CALL CAPTN2(N)
        CALL PLOT(0.,0.,3)
        CALL PLOT(10.,0.,-3)
C*****PLOTS RESULTANT VEL OF U && W ON E-W VERTICAL SECTIONS
1001    IF(IFLOT(2).EQ.0) GO TO 1002
        CALL PLOTUW(IN,JN,KN,IM,JM,UR,W,KZ,DX,DY,DS,
       CPLOTHT,HMAX,UMAX,WMAX,MEX,HU,ARMIN,ARMAX,XL,COSF,SINF,HB,MAR,
```

```
     CNPLT,XSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA,MX)
      CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
      CALL CAPTN5(N,MIDEW)
      CALL CAPTN6(N)
      CALL PLOT(0.,0.,3)
      CALL PLOT(10.,0.,-3)
C*****PLOTS RESULTANT VEL OF V & W ON N-S VERTICAL SECTIONS
1002  IF(IFLOT(3).EQ.0) GO TO 1003
      CALL PLOTVW(IN,JN,KN,IM,JM,VB,W,KZ,DX,DY,DS,
     CPLOTHT,HMAX,UMAX,WMAX,MEX,HV,ARMIN,ARMAX,XL,COSF,SINF,HB,MAR,
     CNPLT,XSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA,MY)
      CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
      CALL CAPTN5(N,MIDEW)
      CALL CAPTN7(N)
      CALL PLOT(0.,0.,3)
      CALL PLOT(10.,0.,-3)
C*****PLOTS SURFACE ELEVATION CONTOUR, CONTOUR VALUES MUST BE GIVEN INDIV
 1003 IF(IFLOT(4).EQ.0) GO TO 997
      ZLIT=ETAL(NG)
      ZBIG=ETAH(NG)
      SAMCON=ZLIT
      EINT=ETAINT(NG)
      CALL ECHKON(ETA,IM,JM,1,IM,1,JM.4.8,5.6,.04,SAMCON,EINT,
     CRGRID,IM,JM,ZLIT,ZBIG,0.,0.,0.,0.,0,0,.07,1.,NPLT,IN,JN,KN,
     CNCN,DX,DY,XSCALE,YSCALE,COSF,SINF,XX,YY)
      CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
      CALL CAPTN5(N,MIDEW)
      CALL CAPTN8(N)
      CALL PLOT(0.,0.,3)
      CALL PLOT(10.,0.,-3)
C*****CONTOURS THE ISOTHERMS OF CALCULATED T-FIELDS, VALUES ARE ASSIGNED
 997  IF(IFLOT(5).EQ.0) GO TO 1000
      K=KPLOT
      DO 998 I=1,IM
      DO 998 J=1,JM
998   ETA(I,J)=TB(I,J,K)
      ZLIT=TL(NG)
      ZBIG=TH(NG)
      SAMCON=ZLIT
      TINT=TI(NG)
      CALL ECHKON(ETA,IM,JM,1,IM,1,JM,4.8,5.6,.04,SAMCON,TINT,
     CRGRID,IM,JM,ZLIT,ZBIG,0.,0.,0.,0.,0,0,.07,1.,NPLT,IN,JN,KN,
     CNCN,DX,DY,XSCALE,YSCALE,COSF,SINF,XX,YY)
      CALL CAPTN1(P1,P2,P3,P4,P5,Q6,XSC,USC)
      CALL CAPTN5(N,MIDEW)
      CALL CAPTN4(SUM)
      CALL CAPTN9(N)
      CALL PLOT(0.,0.,3)
      CALL PLOT(10.,0.,-3)
991   FORMAT(1X,15F8.2)
992   FORMAT(F8.2,'  DEVIATION W.R.T. IR TEMP AT',I5,' IS(DEG-C)',F12.5)
1000  CONTINUE
14    CALL PLOT(0.,0.,999)
101   STOP
      END
```

```
*FLOW(1).PLOTUV FOR CREATED ON 7 DEC 79 AT 10:05:19
C*****************************************************************
C   TO PLOT RESULTANT OF U & V ON SIGMA PLANE K=KPLOT
C*****************************************************************
      SUBROUTINE PLOTUV(IN,JN,KN,IM,JM,UB,VB,MEX,MEY,NCN,XX,YY,AKN,
     CDELZ,ZPPR,ARMIN,ARMAX,COSF,SINF,KC,HB,UDUM1,UDUM,XL,UMAX,DX,DY,
     CHMAX,NPLT,KPLOT,
     CXSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA)
      DIMENSION UB(IM,JM,KN),VB(IM,JM,KN),MEX(IM,JM),MEY(IM,JM),
     CXX(NCN),YY(NCN),HB(IM,JM),ETA(IM,JM)
      WRITE(6,32)
   32 FORMAT(1X,'UV PLOTS')
      A1=DX*XSCALE/2.
      M=KPLOT
      IF(M.GT.1) GO TO 20
      DEPTH=0.
      DO 30 I=1,IM
      DO 30 J=1,JM
      IF(MEX(I,J).EQ.0) GO TO 30
      AI=FLOAT(I-1)*DX*XSCALE
      AJ=FLOAT(J-1)*DY*YSCALE
      AAI=AI+UB(I,J,M)*USCALE
      AAJ=AJ+VB(I,J,M)*VSCALE
      CALL VECT(AI,AAI,AJ,AAJ,ARMIN,ARMAX,COSF,SINF)
   30 CONTINUE
      GO TO 8
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   20 CONTINUE
      DEPTH=DELZ*FLOAT(M-1)
      DO 40 I=1,IM
      DO 40 J=1,JM
      IF(MEX(I,J).EQ.0) GO TO 40
      AH=HB(I,J)+ETA(I,J)
      IF(DEPTH.GE.AH) GO TO 40
      DDZ=AH/AKN
      L1=1+INT(DEPTH/DDZ)
  298 L2=L1+1
      L3=L2+1
      IF(L3.LE.KN) GO TO 299
      L1=L1-1
      GO TO 298
  299 CONTINUE
      Z1=DDZ*FLOAT(L1-1)
      Z2=Z1+DDZ
      Z3=Z2+DDZ
      E1=UB(I,J,L1)
      E2=UB(I,J,L2)
      E3=UB(I,J,L3)
      CALL FIT(Z1,Z2,Z3,E1,E2,E3,A,B,C)
      UDEPTH=A*DEPTH**2+B*DEPTH+C
      E1=VB(I,J,L1)
      E2=VB(I,J,L2)
      E3=VB(I,J,L3)
      CALL FIT(Z1,Z2,Z3,E1,E2,E3,A,B,C)
      VDEPTH=A*DEPTH**2+B*DEPTH+C
      AI=FLOAT(I-1)*DX*XSCALE
      AJ=FLOAT(J-1)*DY*YSCALE
      AAI=AI+UDEPTH*USCALE
      AAJ=AJ+VDEPTH*VSCALE
      CALL VECT(AI,AAI,AJ,AAJ,ARMIN,ARMAX,COSF,SINF)
   40 CONTINUE
    8 CONTINUE
C*****DRAWS OUTLINE OR BOUNDARY OF THE INTEREST AREA
      CALL PLOT(-A1,-A1,-3)
      CALL OUTLIN(IN,JN,KN,NCN,DX,DY,XSCALE,YSCALE,COSF,SINF,XX,YY)
      CALL PLOT(A1,A1,-3)
      NPLT=NPLT+1
      WRITE(6,19) NPLT,DEPTH
   19 FORMAT(1X,'PLOT NOR=',I6,'  COMPLETED.   DEPTH=',F10.0,'  CM')
   10 CONTINUE
  500 CONTINUE
    2 FORMAT( )
      RETURN
      END
```

```
*FLOW(1).PLOTUW FOR CREATED ON 19 DEC 79 AT 12:12:46
      C*******************************************************************
      C  PLOTS RESULTANT OF U & W ON SELECTED E-W CROSS SECTIONS
      C*******************************************************************
            SUBROUTINE PLOTUW(IN,JN,KN,IM,JM,UB,W,KZ,DX,DY,DS,
           CPLOTHT,HMAX,UMAX,WMAX,MEX,HU,ARMIN,ARMAX,XL,COSF,SINF,HB,MAR,
           CNPLT,XSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA,MX)
            DIMENSION MAR(IN,JN),HB(IM,JM),ETA(IM,JM),MX(IN,JN)
            DIMENSION UB(IM,JM,KN),W(IM,JM,KN),MEX(IM,JM),HU(IN,JN)
            WRITE(6,32)
         32 FORMAT(1X,'UW PLOTS')
            KZ=KN-1
            AI=DX*XSCALE/2.
            DO 1000 IJ=1,3
            IF(IJ.EQ.1) J=4
            IF(IJ.EQ.2) J=8
            IF(IJ.EQ.3) J=12
            CALL PLOT(0.,+1.5,-3)
            DO 705 I=1,IM
            IF(MEX(I,J).EQ.0) GO TO 705
            AI=FLOAT(I-1)*DX*XSCALE
            AH=HB(I,J)+ETA(I,J)
            DO 706 K=1,KZ
            AK=(STAGE+ETA(I,J)-FLOAT(K-1)*DS*AH)*ZSCALE
            AAI=AI+UB(I,J,K)*USCALE
            AAK=AK+W(I,J,K)*WSCALE
            YW=.2*SQRT((AAI-AI)**2+(AAK-AK)**2)
            YW=AMAX1(ARMIN,AMIN1(YW,ARMAX))
            CALL AROHD(AI,AK,AAI,AAK,YW,0.0,12)
        706 CONTINUE
        705 CONTINUE
      C*****DRAWS BOUNDARY OR BOTTOM OF THE CROSS SECTION
            CALL PLOT(-AI,0.,-3)
            NN=0
            DO 710 I=1,IN
            IF(MX(I,J).EQ.0) GO TO 710
            NN=NN+1
            IF(NN.GT.1) GO TO 711
            AI=FLOAT(I-1)*DX*XSCALE
            AK=(STAGE+ETA(I,J))*ZSCALE
            CALL PLOT(AI,AK,3)
            AK=-ZSCALE*(HU(I,J)-STAGE)
            CALL PLOT(AI,AK,2)
            GO TO 712
        711 CONTINUE
            AI=FLOAT(I-1)*DX*XSCALE
            AK=-ZSCALE*(HU(I,J)-STAGE)
            CALL PLOT(AI,AK,2)
            ID=I
            AID=AI
        712 CONTINUE
        710 CONTINUE
             IF(I.LT.IN) GO TO 707
             AK=(STAGE+0.0)*ZSCALE
             GO TO 708
        707  AK=(STAGE+ETA(I,J))*ZSCALE
        708  CALL PLOT(AID,AK,2)
            WRITE(6,77) J
         77 FORMAT(1X,'J=',I4)
            FJ=FLOAT(J)
            CALL SYMBOL(5.4,-0.7,.1,2HJ=,0.,2)
            CALL NUMBER(5.7,-0.7,.1,FJ,0.,-1)
            CALL PLOT(AI,0.,-3)
       1000 CONTINUE
            CALL PLOT(0.,-4.5,-3)
            NPLT=NPLT+1
            WRITE(6,18) NPLT
         18 FORMAT(1X,'PLOT NBR',I4,'   COMPLETED')
            RETURN
            END
```

```
*FLOW(1).PLOTVW FOR CREATED ON 19 DEC 79 AT 12:10:41
C*****************************************************************
C  PLOTS RESULTANT OF V & W ON SELECTED N-S CROSS SECTIONS
C*****************************************************************
      SUBROUTINE PLOTVW(IN,JN,KN,IM,JM,VB,W,KZ,DX,DY,DS,
     CPLOTHT,HMAX,UMAX,WMAX,MEX,HV,ARMIN,ARMAX,XL,COSF,SINF,HB,MAR,
     CNPLT,XSCALE,YSCALE,ZSCALE,USCALE,VSCALE,WSCALE,STAGE,ETA,MY)
      DIMENSION MAR(IN,JN),HB(IM,JM),ETA(IM,JM),MY(IN,JN)
      DIMENSION VB(IM,JM,KN),W(IM,JM,KN),MEX(IM,JM),HV(IN,JN)
      WRITE(6,32)
   32 FORMAT(1X,'VW PLOTS')
      KZ=KN-1
      A1=DX*XSCALE/2.
      DO 1000 M=1,3
      IF(M.EQ.1) I=4
      IF(M.EQ.2) I=8
      IF(M.EQ.3) I=12
      CALL PLOT(0.,+1.5,-3)
      DO 802 J=1,JM
      IF(MEX(I,J).EQ.0) GO TO 802
      AJ=FLOAT(J-1)*DY*YSCALE
      AH=HB(I,J)+ETA(I,J)
      DO 803 K=1,KZ
      AK=(STAGE+ETA(I,J)-FLOAT(K-1)*DS*AH)*ZSCALE
      AAJ=AJ+VB(I,J,K)*VSCALE
      AAK=AK+W(I,J,K)*WSCALE
      YW=.2*SQRT((AAJ-AJ)**2+(AAK-AK)**2)
      YW=AMAX1(ARMIN,AMIN1(YW,ARMAX))
      CALL ARCHD(AJ,AK,AAJ,AAK,YW,0.0,12)
  803 CONTINUE
  802 CONTINUE
C*****DRAWS BOUNDARY OR BOTTOM OF THE CROSS SECTION
      CALL PLOT(-A1,0.,-3)
      NN=0
      DO 810 J=1,JN
      IF(MY(I,J).EQ.0) GO TO 810
      NN=NN+1
      IF(NN.GT.1) GO TO 811
      AJ=FLOAT(J-1)*DY*YSCALE
      AK=(STAGE+ETA(I,J))*ZSCALE
      CALL PLOT(AJ,AK,3)
      AK=-ZSCALE*(HV(I,J)-STAGE)
      CALL PLOT(AJ,AK,2)
      GO TO 812
  811 CONTINUE
      AJ=FLOAT(J-1)*DY*YSCALE
      AK=-ZSCALE*(HV(I,J)-STAGE)
      CALL PLOT(AJ,AK,2)
      JD=J
      AJD=AJ
  812 CONTINUE
  810 CONTINUE
      IF(J.LT.JN) GO TO 807
      AK=(STAGE+0.0)*ZSCALE
      GO TO 808
 807  AK=(STAGE+ETA(I,J))*ZSCALE
 808  CALL PLOT(AJD,AK,2)
      WRITE(6,77) I
   77 FORMAT(1X,'I=',I4)
      FI=FLOAT(I)
      CALL SYMBOL(5.4,-0.7,.1,2HI=,0.,2)
      CALL NUMBER(5.7,-0.7,.1,FI,0.,-1)
      CALL PLOT(A1,0.,-3)
 1000 CONTINUE
      CALL PLOT(0.,-4.5,-3)
      NPLT=NPLT+1
      WRITE(6,18) NPLT
   18 FORMAT(1X,'PLOT NBR',I4,'   COMPLETED')
      RETURN
      END
```

```
*FLOW(1).OUTLIN FOR CREATED ON 10 DEC 79 AT 11:23:23
      C**********************************************************************
      C  DRAWS OUTLINE/BOUNDARY OF THE INTEREST AREA
      C**********************************************************************
            SUBROUTINE OUTLIN(IN,JN,KN,NCN,DX,DY,XSCALE,YSCALE,COSF,
           CSINF,XX,YY)
            DIMENSION XX(NCN),YY(NCN)
            DO 15 NC=1,NCN
            A=XX(NC)*DX*XSCALE
            B=YY(NC)*DY*YSCALE
            KV=2
            IF(NC.EQ.1) KV=3
            X=A*COSF+B*SINF
            Y=-A*SINF+B*COSF
         15 CALL PLOT(X,Y,KV)
            RETURN
            END
```

```
*FLOW(1).ECHKON ELT CREATED ON 4 DEC 79 AT 09:59:18
C                                 SUBROUTINE ECHKON
C
C            THIS IS ENTRY SUBROUTINE FOR NHC CONTOURING PROGRAM
C                   [CALCOMP OR MILGO TYPE PLOTTER]
C
C              ELBERT HILL, NHC, MIAMI, FLA.       FALL, 1970
C
C        THE COMPLETE PACKAGE CONSISTS OF 3 SUBROUTINES, ECHKON, CONLIN, AND ENDER
C        ALL 3 ARE CATALOGUED TOGETHER IN THE UM 360/65 UNDER MODULE NAME ECHKON,
C           AND DECKS ARE NOT NEEDED.
C
C        ANY RECTANGULAR GRIDDED SCALAR FIELD CAN BE CONTOURED ON MILGO
C        OR CALCOMP TYPE PLOTTER BY SETTING UP PROPER CALLING ARGUMENTS AND
C           PROCEDURES AS INDICATED BELOW AND THEN CALLING ECHKON.
C
C              ------CALLING STATEMENT IS AS FOLLOWS------
C
C        CALL ECHKON[HH,IN1,IN2,NEX1,NEX2,NEY1,NEY2,HI,WID,PLTINC,SAMCON,
C       2CONINT,RGRID,IN3,IN4,ZLIT,ZBIG,ANORTH,ASOUTH,AEAST,AWEST,NDASHD,
C       3NDASHU,XLABEL,SMOOTH,IRECCY]
C
C                   ---DESCRIPTION OF CALLING ARGUMENTS---
C
C        HH IS ARRAY CONTAINING GRID DATA TO BE CONTOURED. ITS DIMENSIONS
C           ARE IN1 AND IN2. DIMENSION HH[IN1,IN2]. POINT 1,1 IS LOWER LEFT
C           CORNER OF GRID. IN1 IS DIMENSION IN X DIRECTION AND IN2 IS
C           DIMENSION IN Y DIRECTION.
C       [X INCREASES FROM WEST TO EAST AND Y INCREASES FROM SOUTH TO NORTH]
C
C        NEX1, NEX2, NEY1, AND NEY2 DETERMINE THE PORTION OF HH GRID TO
C           BE USED. NEX1 AND NEX2 ARE THE FIRST[LEFTMOST] AND LAST[RIGHTMOST]
C           COLUMNS TO BE USED. NEY1 AND NEY2 ARE THE FIRST[BOTTOM] AND LAST[TOP]
C           ROWS TO BE USED.  [THUS ANY SECTION OF HH CAN BE USED]
C              FOR FULL GRID---
C                             NEX1 > 1
C                             NEX2 > IN1
C                             NEY1 > 1
C                             NEY2 > IN2
C
C        HI IS HEIGHT IN INCHES OF CONTOUR MAP BETWEEN LIMITS NEY1 AND NEY2
C        WID IS WIDTH IN INCHES OF CONTOUR MAP BETWEEN LIMITS NEX1 AND NEX2
C
C        PLTINC IS STRAIGHT LINE PLOT INCREMENT IN INCHES TO BE USED
C           ALONG CONTOUR. GOOD VALUE IS .04, BUT CAN BE VARIED UP OR DOWN.
C         SINCE LARGER VALUES CAUSE PROGRAM TO RUN A LITTLE FASTER, IDEAL VALUE
C           IS LARGEST THAT WILL STILL GIVE SMOOTH LOOKING CURVES.
C            DO SOME EXPERIMENTING WITH IT. START WITH .03 OR .04 AND INCREASE.
C
C        SAMCON IS ANY SAMPLE CONTOUR VALUE. IT IS USED AS A STARTING POINT
C           FOR COUNTING UP AND DOWN TO GET OTHER CONTOUR VALUES.
C
C        CONINT IS CONTOUR INTERVAL TO BE USED.
C
C        RGRID IS AN INTEGER*2 STORAGE ARRAY USED INTERNALLY IN PROGRAM
C           AND NEED NOT BE INITIALIZED. IT IS INCLUDED AS ARGUMENT IN ORDER
C           TO TAKE ADVANTAGE OF VARIABLE DIMENSIONS. DECLARE AS INTEGER*2
C           BEFORE CALLING.
C
C        IN3 AND IN4 ARE X AND Y DIMENSIONS OF RGRID.   DIMENSION RGRID[IN3,IN4]
C           IN3 MUST BE AT LEAST AS LARGE AS NEX2-NEX1<1
C           IN4 MUST BE AT LEAST AS LARGE AS NEY2-NEY1<1
C            [THUS RGRID MUST BE AS LARGE AS PORTION OF DATA ARRAY HH BEING USED]
C
C        ZLIT AND ZBIG ARE LOWER AND UPPER CONTOUR CHECK LIMITS. NO CONTOUR
C           WILL BE DRAWN BELOW VALUE OF ZLIT OR ABOVE VALUE OF ZBIG.
C            [USEFUL TO PREVENT DRAWING FOR ANY COMPLETELY WILD DATA]
C
C        ANORTH, ASOUTH, AEAST, AND AWEST CAN BE USED TO ELIMINATE ANY
C           NUMBER OF INCHES FROM ANY SIDE OF FINAL DRAWING.
C
C           FOR FULL DRAWING WITH HEIGHT > HI AND WIDTH > WID,
C             INITIALIZE ALL 4 OF ABOVE ARGUMENTS TO ZERO.
C
C           FOR EACH OF THE ABOVE WITH POSITIVE VALUE, THIS MANY INCHES
C            WILL BE ELIMINATED ON SIDE TO WHICH IT APPLIES.
C            THIS ALLOWS US TO FIT ANY RECTANGULAR GRID TO ANY MERCATOR
C             OR OTHER MAP LIMITS WITHOUT ACTUALLY ADJUSTING THE GRID.
C
```

```
C      NDASHD AND NDASHU CONTROL TYPE OF CONTOURS [SOLID OR DASHED LINES]
C         IF EITHER OR BOTH ARE ZERO OR LESS, CONTOURS ARE SOLID LINES.
C
C       IF BOTH ARE POSITIVE, CONTOURS WILL BE DASHED AS FOLLOWS----
C
C          PEN DOWN SECTION LENGTH > NDASHD*PLTINC   [PLTINC IS INCREMENT LENGTH]
C            PEN UP SECTION LENGTH > NDASHU*PLTINC
C              [THUS LENGTH OF DASHES AND SKIPS IS FULLY VARIABLE]
C
C          XLABEL CONTROLS LABELING OF CONTOURS. LINES ARE LABELED
C              ONLY IF XLABEL GREATER THAN ZERO. VALUE OF XLABEL
C                IS HEIGHT IN INCHES OF LABEL NUMBERS. LINES ARE LABELED
C                WITH NEAREST WHOLE NUMBER VALUE OF CONTOUR. IF SPECIAL
C                LABELING TO INCLUDE ONLY PART OF NUMBER OR TO INCLUDE
C                DECIMALS IS DESIRED, SUBROUTINE ENDER MUST BE CHANGED.
C
C          SMOOTH IS A CONTROL FOR VARYING CONTOUR SMOOTHING.
C              INITIALIZE SMOOTH TO SOME VALUE BETWEEN 0.25 AND 7.5
C           [ANY VALUE OUTSIDE THIS RANGE IS SET INTERNALLY TO 1.0]
C               LARGER VALUES GIVE SMOOTHER CHART WITH LESS DETAIL, WHILE
C                 SMALLER VALUES GIVE LESS SMOOTHING AND MORE DETAIL.
C            [NORMAL VALUE FOR MOST RUNS SHUD BE ABOUT 1.5]
C            ANYTHING LESS THAN ABOUT 0.40 OR LARGER THAN ABOUT 3. IS
C              PROBABLY NO GOOD. BEGIN WITH 1.5 AND EXPERIMENT UP OR DOWN
C            TO DETERMINE MOST DESIRABLE VALUE FOR YOUR NEEDS.
C          [INPUT GRID DATA VALUES ARE NOT ALTERED IN THIS SMOOTHING]
C
C          IRECCY IS PLOT TAPE RECORD COUNTER. INITIALIZE TO NUMBER
C              OF PLOT RECORDS WRITTEN BEFORE FIRST CALL TO CONTOUR SUBROUTINE.
C
C       ALL OF THE ABOVE ARGUMENTS EXCEPT ARRAY RGRID MUST BE DEFINED.
C          ARGUMENTS ARE NOT ALTERED WITHIN PROGRAM, AND RETURN INTACT.
C
C            PLOTTER BUFFER SPACE MUST BE SET UP AND CALL TO PLOTS
C                MADE BEFORE FIRST CALL TO THIS SUBROUTINE.
C
C              PLOT TAPE MUST BE CLOSED OUT AFTER FINAL CALL.
C
C            ANY NUMBER OF SUCCESSIVE CALLS CAN BE MADE TO CONTOUR
C            SUBROUTINE ECHKON. EACH MAP BECOMES A SEPARATE PLOT RECORD.
C            NO INTERNAL MAP SPACING IS PROVIDED, WITH PEN RETURNING TO
C            ORIGINAL ORIGIN[LOWER LEFT CORNER] AT COMPLETION OF MAP.
C      [THUS IT IS SIMPLE TO PUT MORE THAN ONE SET OF CONTOURS ON SAME MAP]
C          ANY SPECIAL MARKINGS OR LABELS THAT ARE DESIRED MUST BE DONE
C            OUTSIDE THIS SUBROUTINE. THIS SUBROUTINE DRAWS CONTOURS ONLY
C          WITH INCOMING ORIGIN BEING LOWER LEFT CORNER OF CONTOUR CHART.
C
C-------------------------------------------------------------------------
C-------------------------------------------------------------------------
C
      SUBROUTINE ECHKON(HH,IN1,IN2,NEX1,NEX2,NEY1,NEY2,HI,WID,PLTINC,
     2SAMCON,CONINT,RGRID,IN3,IN4,ZLIT,ZBIG,ANORTH,ASOUTH,AEAST,AWEST,
     3NDASHD,NDASHU,XLABEL,SMOOTH,IRECCY,IN,JN,KN,
     CNCN,DX,DY,XSCALE,YSCALE,COSF,SINF,XX,YY)
C
C          SEE ABOVE COMMENTS FOR DESCRIPTION AND USE OF ABOVE ARGUMENTS
C
      COMMON /STRCON/SMHI,SMWI,X,Y,XGRID,YGRID,CUTOF,SCHI,SCWI,TMAX,XPP,
     2YPP,CGIG,U,V,NXUX,JDOO,NUVX,NUVY,YCPTH,SOUTH,EAST,WEST,CLIT,CBIG,
     3LCLX,LCLY,INCRCS,OINC,CLOSIT,PVAL,PVOL,NENTER,HINUM,NMX1,NMY1,
     4NMX11,NMY11,MOSINC,VALLIN,HINC,MAXCRD,WHAT,LDASH1,LDASH2,DASHER,
     5DOLARS,OUTS
      COMMON /OENDEO/HIXEN,WCDE,HOGH,XXLAST,YYLAST
      LOGICAL DASHER,DOLARS,OUTS
      DIMENSION HH(IN1,IN2),RGRID(IN3,IN4),HEN(4),XX(NCN),YY(NCN)
      INTEGER RGRID
      DATA IMAP/0/
C
C      NHC SUBROUTINE FOR CONTOURING SCALAR FIELD ON CALCOMP OR MILGO TYPE PLOTTE
C                  ELBERT HILL       ---FALL, 1970--
C
      I=IMAP+1
      WRITE(6,10)I
   10 FORMAT(///,2X,11HCONTOUR MAP,I4,1X,26HINCOMING ARGUMENTS FOLLOW.)
      WRITE(6,12)IN1,IN2,NEX1,NEX2,NEY1,NEY2,HI,WID,PLTINC,SAMCON,CONINT
     2,IN3,IN4
   12 FORMAT(/,5X,63HIN1 IN2 NEX1 NEX2 NEY1 NEY2 HI WID PLTINC SAMCON CO
     2NINT IN3 IN4,//,2X,6I5,2F10.4,3F12.3,2I10)
      WRITE(6,14)ZLIT,ZBIG,ANORTH,ASOUTH,AEAST,AWEST,NDASHD,NDASHU,XLABE
```

```
     2L,SMOOTH,IRECCY
  14 FORMAT(/,5X,70HZLIT ZBIG ANORTH ASOUTH AEAST AWEST NDASHO NDASHU X
     2LABEL SMOOTH IRECCY,//,2X,2F12.3,4F10.4,2I6,2F10.3,I6,/)
      MAXCRD=0
      WHAT=-99.
      CONINC=CONINT
      IF(CONINC.NE.0.)GO TO 3
      WRITE(6,2)I
   2 FORMAT(/,2X,3HMAP,I3,14H ZERO INTERVAL)
      GO TO 120
   3 IRECI=IRECCY+1
      LDASH1=NDASHO
      LDASH2=NDASHU
      DASHER=.FALSE.
      IF(LDASH2.GT.0.AND.LDASH1.GT.0)DASHER=.TRUE.
      HINUM=XLABEL
      DOLABS=.FALSE.
      IF(HINUM.GT.0.)DOLABS=.TRUE.
      IF(CONINC.LT.0.)CONINC=-CONINC
      PVOL=.005*CONINC
      MOSIAC=0
      VALLIN=-989898.989
      NUVX=NEX2-NEX1+1
      NUVY=NEY2-NEY1+1
      IF(NUVX.GT.3.AND.NUVX.LE.IN3.AND.NUVY.GT.3.AND.NUVY.LE.IN4)GO TO 8
      WRITE(6,7)NEX1,NEX2,NUVX,NEY1,NEY2,NUVY
   7 FORMAT(/,10X,23HBAD ARRAY LIMITS. SKIP./10X,3I10/10X,3I10)
      GO TO 120
C        SKIP IF NUVX OR NUVY LESS THAN 4
   8 YORTH=HI-ANORTH
      SOUTH=ASOUTH
      EAST=WID-AEAST
      WEST=AWEST
      IF(WEST.LT.0.)WEST=0.
      IF(EAST.GT.WID)EAST=WID
      IF(SOUTH.LT.0.)SOUTH=0.
      IF(YORTH.GT.HI)YORTH=HI
      HIXEN=HINUM
      WODE=EAST-WEST
      HOGH=YORTH-SOUTH
      XXLAST=99.
      YYLAST=99.
      QINC=PLTINC
      CLIT=QINC/1.99
      CBIG=QINC/2.
      TMAX=4.0*(YORTH-SOUTH+EAST-WEST)
      XGRID=WID/FLOAT(NUVX-1)
      YGRID=HI/FLOAT(NUVY-1)
      HINC=XGRID
      IF(YGRID.LT.XGRID)HINC=YGRID
      X=SMOOTH
      IF(X.LT..25.OR.X.GT.7.5)X=1.0
      HINC=X*HINC
      CUTOF=SQRT(XGRID*XGRID+YGRID*YGRID)+.01
      CLOSIT=.04
C     CLOSIT IS VALUE FOR CLOSED CONTOUR CHECK
      NMX1=NEX1
      NMY1=NEY1
      NMX11=NMX1-1
      NMY11=NMY1-1
      NEX4=NEX1-1
      NEY4=NEY1-1
C
C        NEXT DETERMINE MAX.AND MIN VALUES IN SCALAR FIELD
      ZMAX=HH(NEX1,NEY1)
      ZMIN=ZMAX
      DO 30 I=NEX1,NEX2
      DO 30 J=NEY1,NEY2
      IF(HH(I,J).GT.ZMAX)ZMAX=HH(I,J)
      IF(HH(I,J).LT.ZMIN)ZMIN=HH(I,J)
  30 CONTINUE
      IF(ZMAX.GT.ZBIG)ZMAX=ZBIG
      IF(ZMIN.LT.ZLIT)ZMIN=ZLIT
C        NEXT DETERMINE BOTTOM STARTING VALUE FOR CONTOUR LOOP
      PVAL=SAMCON
  32 IF(PVAL.GT.ZMIN)GO TO 34
      PVAL=PVAL+CONINC
      GO TO 32
  34 IF(PVAL-CONINC.LT.ZMIN)GO TO 35
```

```
      PVAL=PVAL-CONINC
      GO TO 34
C
   35 XPP=0.
      YPP=0.
      N1=NCVX-1
      N2=NCVY-1
C
C     CONTOUR LOOP STARTS BELOW AT STATEMENT 36
C      THIS LOOP DETERMINES WHERE TO START A NEW CONTOUR, THEN CALLS
C        SUBROUTINE CONLIN TO DRAW EACH CONTOUR.  EXIT IS MADE WHEN
C      ALL CONTOURS COMPLETED.
C
C     THERE ARE 2 SCANS FOR EACH CONTOUR VALUE. FIRST WITH VARIABLE OUTS AS
C      FALSE SELECTS ONLY CONTOURS ENTERING GRID FROM OUTSIDE EDGES.
C       SECOND SCAN WITH OUTS TRUE SELECTS REMAINING INNER CONTOURS.
C      STARTING POINT CLOSEST TO PLOT PEN POSITION IS SELECTED IN EACH CASE.
C
   36 IF(PVAL.GE.ZMAX)GO TO 110
      OUTS=.FALSE.
      DO 37 I=1,N1
      DO 37 J=1,N2
   37 RGRID(I,J)=0
   38 DZ=999999.
      DO 100 I=1,N1
      DO 100 J=1,N2
      IF(OUTS)GO TO 600
      IF(I.EQ.1.OR.J.EQ.1.OR.I.EQ.N1.OR.J.EQ.N2)GO TO 600
      GO TO 100
  600 IF(RGRID(I,J).EQ.1)GO TO 100
      IF(RGRID(I,J).GT.1.AND.OUTS)GO TO 100
      II=NEX4+I
      JJ=NEY4+J
      HEN(1)=HH(II,JJ)
      HEN(2)=HH(II,JJ+1)
      HEN(3)=HH(II+1,JJ+1)
      HEN(4)=HH(II+1,JJ)
      DO 400 K=1,4
      IF(ABS(HEN(K)-PVAL).GE.PVOL)GO TO 400
      IF(HEN(K).GE.PVAL)HEN(K)=PVAL+PVOL
      IF(HEN(K).LT.PVAL)HEN(K)=PVAL-PVOL
  400 CONTINUE
      IF(OUTS)GO TO 250
      NENN=1
      IF(I.EQ.1.AND.HEN(1).GT.PVAL.AND.HEN(2).LT.PVAL)GO TO 601
      NENN=3
      IF(I.EQ.N1.AND.HEN(3).GT.PVAL.AND.HEN(4).LT.PVAL)GO TO 601
      NENN=4
      IF(J.EQ.1.AND.HEN(4).GT.PVAL.AND.HEN(1).LT.PVAL)GO TO 601
      NENN=2
      IF(J.EQ.N2.AND.HEN(2).GT.PVAL.AND.HEN(3).LT.PVAL)GO TO 601
      GO TO 602
  250 DO 410 K=1,4
      I1=K
      I2=K+1
      IF(K.EQ.4)I2=1
      IF(HEN(I1).GT.PVAL.AND.HEN(I2).LT.PVAL)GO TO 408
  410 CONTINUE
      GO TO 602
  408 NENN=K
  601 IF(RGRID(I,J).EQ.0.OR.OUTS)GO TO 640
      I1=RGRID(I,J)/10
      I2=RGRID(I,J)-10*I1
      IF(I1.EQ.NENN.OR.I2.EQ.NENN)GO TO 100
  640 GO TO(340,342,344,346),NENN
  602 IF(RGRID(I,J).EQ.0)RGRID(I,J)=1
      GO TO 100
  340 Y=YGRID*(FLOAT(J-1)+(PVAL-HEN(1))/(HEN(2)-HEN(1)))
      X=XGRID*FLOAT(I-1)
      GO TO 45
  342 X=XGRID*(FLOAT(I-1)+(PVAL-HEN(2))/(HEN(3)-HEN(2)))
      Y=YGRID*FLOAT(J)
      GO TO 45
  344 Y=YGRID*(FLOAT(J-1)+(PVAL-HEN(4))/(HEN(3)-HEN(4)))
      X=XGRID*FLOAT(I)
      GO TO 45
  346 X=XGRID*(FLOAT(I-1)+(PVAL-HEN(1))/(HEN(4)-HEN(1)))
      Y=YGRID*FLOAT(J-1)
```

```
   45 D=(X-XPP)*(X-XPP)+(Y-YPP)*(Y-YPP)
      IF(D.GE.DZ)GO TO 100
      DZ=D
      NENTER=NENN
      LCLX=I
      LCLY=J
      XTT=X
      YTT=Y
  100 CONTINUE
      IF(DZ.GE.999990.)GO TO 105
      IF(RGRID(LCLX,LCLY).EQ.0)RGRID(LCLX,LCLY)=1
      X=XTT
      Y=YTT
C
      WRITE(6,101)PVAL,DOLABS,OUTS
  101 FORMAT(1X,E13.5,2L13)
C         NEXT CALL SUBROUTINE CONLIN TO ACTUALLY DRAW CONTOUR WITH VALUE PVAL
C
      CALL CONLIN(HH,IN1,IN2,RGRID,IN3,IN4)
C
C       NOW GO BACK TO INNER LOOP TO SEE IF THERE ARE OTHER PVAL CONTOURS
C       TO BE DRAWN.
C
      GO TO 38
  105 IF(OUTS)GO TO 612
      OUTS=.TRUE.
      GO TO 38
  612 PVAL=PVAL+CONINC
C        INCREMENT CONTOUR AND GO TO TOP OF LOOP FOR NEXT CONTOUR
      GO TO 36
  110 CALL PLOT(0.,0.,-3)
C
C        NOW TO DRAW THE OUTLINE OF INTEREST AREA BY CALLING OUTLIN
C
      A1=DX*XSCALE/2.
      CALL PLOT(-A1,-A1,-3)
      CALL OUTLIN(IN,JN,KN,NCN,DX,DY,XSCALE,YSCALE,
     CCOSF,SINF,XX,YY)
      CALL PLOT(A1,A1,-3)
      IMAP=IMAP+1
      IRECCY=IRECCY+1
      WRITE(6,115)IMAP,IREC1,IRECCY
  115 FORMAT(/,10X,11HCONTOUR MAP,I3,24H BEGINS WITH PLOT RECORD,I3,14H
     2AND ENDS WITH,I3)
      WRITE(6,116)MOSINC,VALLIN,MAXCRD,WHAT
  116 FORMAT(12X,21HMOST LINE INCREMENTS ,I5,12H ON CONTOUR ,F10.2,/,12X
     2,12HMOST SQUARES,I4,12H ON CONTOUR ,F10.2,/)
  120 RETURN
      END
```

```
*FLOW(1).CONLIN ELT CREATED ON 27 AUG 79 AT 21:35:06
      SUBROUTINE CONLIN(HH,IN1,IN2,RGRID,IN3,IN4)
      COMMON /STRCON/SMHI,SMWI,X,Y,XGRID,YGRID,CUTOF,SQHI,SQWI,THAX,XPP,
     2YPP,CBIG,U,V,NXUX,JDOO,NUVX,NUVY,YORTH,SOUTH,EAST,WEST,CLIT,CBIG,
     3LCLX,LCLY,INCROS,QINC,CLOSIT,PVAL,PVOL,NENTER,HINUM,NMX1,NMY1,
     4NMX11,NMY11,MOSINC,VALLIN,HINC,MAXCRD,WHAT,LDASH1,LDASH2,DASHER,
     5DOLARS,OUTS
      DIMENSION HH(IN1,IN2),CIDE(4,2),XXPLOT(275,2),HAX(4),LEXE(4),
     2CORD(400,2),HIPPS(400)
      INTEGER RGRID(IN3,IN4)
      LOGICAL INCS,DOLARS,DASHER,CLOS,OUTS,DASHIX
C
C           THIS SUBROUTINE IS CALLED TO DRAW EACH INDIVIDUAL CONTOUR
C
C        IF DOLARS ENTERS AS TRUE, LABEL CONTOURS WITH HEIGHT HINUM
      DASHIX=DASHER
      LABLIT=9
      IF(DOLARS)LABLIT=0
      INCS=.FALSE.
      YMAX=-9.
      XMAX=-9.
      NENST=NENTER
      IOPLOT=2
      NHARD=LDASH1
      NSOFT=LDASH2
      NUGG=0
      XX=X
      YY=Y
      XBIG=XX
      YBIG=YY
      LZX=LCLX
      LZY=LCLY
      IPER=2
      XD=0.
      YD=0.
      TOT=0.
      HYPTOT=0.
      NCORD=0
      CLOS=.FALSE.
      GO TO 400
C
C     END SETUP. BEGIN LOOP THAT PICKS EXACT STRAIGHT LINE SEGMENTED TRAVERSE
C
  250 IF(NCORD.LT.400)GO TO 252
      WRITE(6,251)NCORD,PVAL
  251 FORMAT(/,2X,I5,14H SQUARES LINE ,F10.5,2X,7HSHUTOFF)
C     SHUTOFF MESSAGE HERE INDICATES THAT THIS CONTOUR CROSSES MORE
C        THAN 400 GRID SQUARES. ARRAYS CORD AND HIPPS ARE TOO SMALL. CONTOUR
C         WILL BE CUT OFF AT SQUARE 400. CURE IS TO ENLARGE ARRAYS
C            AND ASSOCIATED CUTOFF CHECK STATEMENT ABOVE.
      GO TO 730
  252 NCORD=NCORD+1
      HYPTOT=HYPTOT+HYPE
      HIPPS(NCORD)=HYPE
      CORD(NCORD,1)=XXSQ
      CORD(NCORD,2)=YYSQ
      XD=XD+XXSQ
      YD=YD+YYSQ
      IF(NEXET.EQ.1)LZX=LZX-1
      IF(NEXET.EQ.3)LZX=LZX+1
      IF(NEXET.EQ.2)LZY=LZY+1
      IF(NEXET.EQ.4)LZY=LZY-1
      IF(LZX.LT.1.OR.LZX.GE.NUVX)GO TO 730
      IF(LZY.LT.1.OR.LZY.GE.NUVY)GO TO 730
      IF(LZX.EQ.LCLX.AND.LZY.EQ.LCLY.AND.SQRT(XD*XD+YD*YD).LE.CLOSIT.AND
     2.OUTS.AND.NCORD.GT.3)GO TO 701
      GO TO 700
  701 CLOS=.TRUE.
      IF(LABLIT.EQ.0)LABLIT=-1
      GO TO 730
  700 NENTER=NEXET+2
      IF(NEXET.GT.2)NENTER=NEXET-2
      XBIG=XIND
      YBIG=YIND
  400 NUMOLT=0
      OX1=XBIG-XGRID*FLOAT(LZX-1)
      IF(OX1.LT.0.)OX1=0.
      IF(OX1.GT.XGRID)OX1=XGRID
      OY1=YBIG-YGRID*FLOAT(LZY-1)
```

```
      IF(OY1.LT.0.)OY1=0.
      IF(OY1.GT.YGRID)OY1=YGRID
      I=LZX+NMX11
      J=LZY+NMY11
C
C        START EXIT POINT LOOP
C
      HAX(1)=HH(I,J)
      HAX(2)=HH(I,J+1)
      HAX(3)=HH(I+1,J+1)
      HAX(4)=HH(I+1,J)
      DO 401 II=1,4
      IF(ABS(HAX(II)-PVAL).GE.PVOL)GO TO 401
      IF(HAX(II).GE.PVAL)HAX(II)=PVAL+PVOL
      IF(HAX(II).LT.PVAL)HAX(II)=PVAL-PVOL
  401 CONTINUE
      NEXET=0
      DO 435 III=1,4
      CIDE(III,1)=-1.
      CIDE(III,2)=-1.
      I1=III
      I2=III+1
      IF(III.EQ.4)I2=1
C
C       STATEMENT BELOW SELECTS SIDES THAT HAVE EXIT POINTS
C
      IF(HAX(I1).LT.PVAL.AND.HAX(I2).GT.PVAL)GO TO 420
      GO TO 435
  420 NUMOUT=NUMOUT+1
      IF(NUMOUT.EQ.1)NN1=III
      IF(NUMOUT.EQ.2)NN2=III
      GO TO(422,424,426,428),III
  422 OY2=((PVAL-HAX(1))/(HAX(2)-HAX(1)))*YGRID
      OX2=0.
      GO TO 430
  424 OX2=((PVAL-HAX(2))/(HAX(3)-HAX(2)))*XGRID
      OY2=YGRID
      GO TO 430
  426 OY2=((PVAL-HAX(4))/(HAX(3)-HAX(4)))*YGRID
      OX2=XGRID
      GO TO 430
  428 OX2=((PVAL-HAX(1))/(HAX(4)-HAX(1)))*XGRID
      OY2=0.
  430 CIDE(III,1)=OX2
      CIDE(III,2)=OY2
  435 CONTINUE
C     UNLESS WE HAVE NULL POINT SQUARE, NUMOUT SHUD BE 1 WITH OUT AT OX2, OY2
      IF(NUMOUT.NE.1)GO TO 432
      NEXET=NN1
      GO TO 445
  432 IF(NUMOUT.EQ.2)GO TO 438
  431 WRITE(6,436)LZX,LZY,NUMOUT,PVAL,XBIG,YBIG
  436 FORMAT(/,2X,10HNO WAY OUT,5X,3I10,3F10.2,/)
      GO TO 500
C
C     BEGIN SECTION THAT DETERMINES PROPER PATH THRU GRID SQUARE
C       CONTAINING HYPERBOLIC CONFIGURATION. (2 ENTRY AND 2 EXIT SIDES)
C
  438 IF(RGRID(LZX,LZY).GT.1)GO TO 442
      XID=CIDE(NN1,1)-OX1
      YID=CIDE(NN1,2)-OY1
      DAA=SQRT(XID*XID+YID*YID)
      XID=CIDE(NN2,1)-OX1
      YID=CIDE(NN2,2)-OY1
      DBB=SQRT(XID*XID+YID*YID)
      IF(DBB.LT.DAA)GO TO 440
  439 OX2=CIDE(NN1,1)
      OY2=CIDE(NN1,2)
      NEXET=NN1
      GO TO 414
  440 OX2=CIDE(NN2,1)
      OY2=CIDE(NN2,2)
      NEXET=NN2
  414 RGRID(LZX,LZY)=10*NENTER+NEXET
      GO TO 445
  442 I1=RGRID(LZX,LZY)/10
      I2=RGRID(LZX,LZY)-10*I1
      RGRID(LZX,LZY)=1
      IF(I1.GT.0.AND.I2.GT.0.AND.I1.NE.I2.AND.NENTER.GT.0)GO TO 417
```

```
  415 WRITE(6,416)LZX,LZY,I1,I2,NENTER
  416 FORMAT(/,2X,12HBAD COL EXIT,2X,5I12)
      GO TO 500
  417 LEXE(1)=0
      LEXE(2)=0
      LEXE(3)=0
      LEXE(4)=0
      LEXE(I1)=1
      LEXE(I2)=1
      DO 418 II=1,4
      IF(LEXE(II).EQ.0.AND.II.NE.NENTER.AND.CIDE(II,1).GT.(-.5).AND.CIDE
     2(II,2).GT.(-.5))GO TO 419
  418 CONTINUE
      GO TO 415
  419 NEXET=II
      OX2=CIDE(II,1)
      OY2=CIDE(II,2)
C                          END HYPERBOLIC GRID SQUARE SECTION
  445 XIND=OX2+XGRID*FLOAT(LZX-1)
      YIND=OY2+YGRID*FLOAT(LZY-1)
      IF(RGRID(LZX,LZY).EQ.0)RGRID(LZX,LZY)=1
      XXSQ=XIND-XBIG
      YYSQ=YIND-YBIG
      HYPE=SQRT(XXSQ*XXSQ+YYSQ*YYSQ)
      IF(HYPE.GE..0001.AND.HYPE.LE.CUTOF)GO TO 396
      WRITE(6,397)LZX,LZY,NENTER,NEXET,XBIG,YBIG,XIND,YIND,OX1,OY1,OX2,
     2OY2,XXSQ,YYSQ,HYPE,NENST,LCLX,LCLY,NCORD,XX,YY
  397 FORMAT(/,2X,4HHERE,4I10,/,2X,11F10.5,//,2X,4I10,5X,2F12.6,//)
      GO TO 500
  396 GO TO 250
C
C     LINE SEGMENTED CONTOUR TRAVERSE NOW COMPLETE. NEXT, DIVIDE THIS
C       TRAVERSE INTO NRINC EQUAL SEGMENTS. THIS NUMBER IS FUNCTION NOT
C         ONLY OF LENGTH OF TRAVERSE, BUT OF INCOMING ARGUMENT SMOOTH, WHICH
C           CONTROLS DEGREE OF SMOOTHING DESIRED.
C
  730 IF(NCORD.LE.MAXCRD)GO TO 732
      MAXCRD=NCORD
      WHAT=PVAL
  732 NRINC=HYPTOT/HINC+1.
      HANC=HYPTOT/FLOAT(NRINC)
      IF(.NOT.CLOS.AND.NCORD.GT.1.AND.NRINC.GT.1)GO TO 734
      IF(CLOS.AND.NCORD.GT.3.AND.NRINC.GT.2)GO TO 734
      GO TO 502
C
C     NEXT, SET UP ENTRY AND EXIT SLOPE DATA FOR FIRST SEGMENT BEFORE
C         ENTERING MAIN CURVILINEAR INTERPOLATE AND PLOT LOOP.
C
  734 XBEG=XX
      YBEG=YY
      IF(NRINC*HANC.LT..75)DASHIX=.FALSE.
      H=0.
      XEND=XX
      YEND=YY
      DO 740 I=1,NCORD
      IF(H+HIPPS(I).GE.HANC)GO TO 742
      H=H+HIPPS(I)
      XEND=XEND+CORD(I,1)
  740 YEND=YEND+CORD(I,2)
      GO TO 745
  742 X=(HANC-H)/HIPPS(I)
      XEND=XEND+X*CORD(I,1)
      YEND=YEND+X*CORD(I,2)
  745 XXSQ=XEND-XBEG
      YYSQ=YEND-YBEG
      HYPE=SQRT(XXSQ*XXSQ+YYSQ*YYSQ)
      IF(HYPE.GE..0001)GO TO 750
  747 WRITE(6,748)XBEG,YBEG,PVAL,HYPE
  748 FORMAT(/,2X,14HHYPE TOO SMALL,2X,4F12.6)
      GO TO 500
  750 CANG=XXSQ/HYPE
      SANG=YYSQ/HYPE
      HYPM=HYPE
      IF(CLOS)GO TO 751
      COSBAC=CANG
      SINBAC=SANG
      GO TO 759
  751 XB=XX
      YB=YY
```

```
      H=0.
      DO 753 I=1,NCORD
      J=NCORD-I+1
      IF(H+HIPPS(J).GE.HANC)GO TO 755
      H=H+HIPPS(J)
      XB=XB-CORD(J,1)
  753 YB=YB-CORD(J,2)
      GO TO 757
  755 X=(HANC-H)/HIPPS(J)
      XB=XB-X*CORD(J,1)
      YB=YB-X*CORD(J,2)
  757 XXSQ=XBEG-XB
      YYSQ=YBEG-YB
      HYPE=SQRT(XXSQ*XXSQ+YYSQ*YYSQ)
      IF(HYPE.LT..0001)GO TO 747
      SINBAC=YYSQ/HYPE
      COSBAC=XXSQ/HYPE
  759 A=.5*ATAN2(SANG*COSBAC-CANG*SINBAC,CANG*COSBAC+SANG*SINBAC)
      SA=SIN(A)
      CA=COS(A)
      SENT=SINBAC*CA+COSBAC*SA
      CENT=COSBAC*CA-SINBAC*SA
      ENTIR=ATAN2(SENT,CENT)
      SSENT=SENT
      CCENT=CENT
C
C           ENTER MAIN CURVILINEAR INTERPOLATE AND PLOT LOOP
C
      DO 800 LUPE=1,NRINC
      IF(LUPE.NE.NRINC)GO TO 762
      IF(CLOS)GO TO 760
      SOUT=SANG
      COUT=CANG
      GO TO 200
  760 SOUT=SSENT
      COUT=CCENT
      GO TO 200
  762 XIND=XX
      YIND=YY
      ZANC=HANC*FLOAT(LUPE+1)
      H=0.
      DO 764 I=1,NCORD
      IF(H+HIPPS(I).GE.ZANC)GO TO 766
      H=H+HIPPS(I)
      XIND=XIND+CORD(I,1)
  764 YIND=YIND+CORD(I,2)
      GO TO 768
  766 X=(ZANC-H)/HIPPS(I)
      XIND=XIND+X*CORD(I,1)
      YIND=YIND+X*CORD(I,2)
  768 XXSQ=XIND-XEND
      YYSQ=YIND-YEND
      HYPE=SQRT(XXSQ*XXSQ+YYSQ*YYSQ)
      IF(HYPE.LT..0001)GO TO 747
      SINFOR=YYSQ/HYPE
      COSFOR=XXSQ/HYPE
      HYPFOR=HYPE
      A=.5*ATAN2(SINFOR*CANG-COSFOR*SANG,COSFOR*CANG+SINFOR*SANG)
      SA=SIN(A)
      CA=COS(A)
      SOUT=SANG*CA+CANG*SA
      COUT=CANG*CA-SANG*SA
      EXET=ATAN2(SOUT,COUT)
  200 HYP=HYPM
      IF(HYP.GT.OINC)GO TO 449
      IF(XEND.LT.WEST.OR.XEND.GT.EAST.OR.YEND.LT.SOUTH.OR.YEND.GT.YORTH)
     2GO TO 790
      IF(INCS)GO TO 446
      INCS=.TRUE.
      CALL PLOT(XBEG-WEST,YBEG-SOUTH,3)
  446 I=IPER
      IF(DASHIX.AND.I.EQ.2)I=IDPLOT
C     CALL PLOT(XEND-WEST,YEND-SOUTH,I)
      NUGG=NUGG+1
      TOT=TOT+HYP
      GO TO 790
C
C     BEGIN SNAKE INTERPOLATION FOR SEGMENT
C
```

```
C      ENTIR AND EXET ARE TRUE ENTRY AND EXIT DIRECTIONS AT ENDS OF SEGMENT
C         [SNAKELIKE INTERPOLATION IS MADE FOR FIT BETWEEN 2 END DIRECTIONS]
C       SENT AND CENT ARE SINE AND COSINE OF TRUE ENTRY ANGLE
C       SOUT AND COUT ARE SINE AND COSINE OF TRUE EXIT ANGLE
C
C       SANG AND CANG ARE SINE AND COSINE OF STRAIGHT LINE ENTRY/EXIT CONNECTOR
C
C       BEGIN SECTION THAT INTERPOLATES AND PLOTS THRU SEGMENT
C
  449 S=SENT*CANG-CENT*SANG
      C=CENT*CANG+SENT*SANG
      C1=(2.*ATAN2(S,C))/HYP
      SING=SANG*C-CANG*S
      CING=CANG*C+SANG*S
      S=SOUT*CING-COUT*SING
      C=COUT*CING+SOUT*SING
      C2=(2.*ATAN2(S,C))/HYP
      TX=XBEG
      TY=YBEG
      NINC=0
      TYP=-CBIG
      HYPMAX=HYP-CLIT
      H25=.25*HYP
  450 TYP=TYP+QINC
      D1=TYP
      IF(D1.GT.H25)D1=H25
      D2=TYP-H25
      IF(D2.LT.0.)D2=0.
      SING=ENTIR-C1*TYP+C2*(D2-D1)
      TX=TX+QINC*COS(SING)
      TY=TY+QINC*SIN(SING)
C
C          END SNAKE INTERPOLATION SECTION---TRY AND FIGURE IT OUT AND GO NUTS-
C      NEXT STORE POINTS THRU THIS SEGMENT FOR FINAL ADJUST AND PLOT
C
      IF(NINC.LT.275)GO TO 453
      WRITE(6,454)PVAL,XBEG,YBEG,XEND,YEND
  454 FORMAT(/,2X,12HNINC SHUTOFF,2X,5F12.3)
C     IF SHUTOFF MESSAGE RECEIVED HERE, ARRAY XXPLOT IS TOO SMALL.
C     FOR INFREQUENT MESSAGES, DONT WORRY ABOUT IT, SINCE LACK OF
C        CLOSURE IS ADJUSTED OUT.  IF MESSAGE PERSISTS, EITHER INCREASE
C          PLOT INCREMENT LENGTH OR SIZE OF XXPLOT.
      GO TO 455
  453 NINC=NINC+1
      XXPLOT(NINC,1)=TX
      XXPLOT(NINC,2)=TY
      IF(TYP.LE.HYPMAX)GO TO 450
C
C          ADJUST FOR CLOSURE ERROR, THEN PLOT CURVE ALONG THIS SEGMENT.
C
  455 XER=(XEND-XXPLOT(NINC,1))/FLOAT(NINC)
      YER=(YEND-XXPLOT(NINC,2))/FLOAT(NINC)
      U=0.
      V=0.
      NUNC=0
C
C          BEGIN SEGMENT PLOT LOOP----DASHED OR SOLID CURVES-----
C          SUBROUTINE ENDER IS CALLED TO LABEL LINES
C
      DO 610 I=1,NINC
      U=U+XER
      V=V+YER
      X=XXPLOT(I,1)+U
      Y=XXPLOT(I,2)+V
      IF(X.LT.WEST.OR.X.GT.EAST.OR.Y.LT.SOUTH.OR.Y.GT.YORTH)GO TO 608
      IF(INCS)GO TO 603
      INCS=.TRUE.
      IF(IPER.EQ.2)CALL PLOT(XBEG-WEST,YBEG-SOUTH,3)
  603 XPP=X
      YPP=Y
      XUX=XPP-WEST
      YUY=YPP-SOUTH
      IF(LABLIT.GT.0)GO TO 604
      IF(LABLIT.EQ.0)CALL ENDER(XUX,YUY,PVAL,1)
      LABLIT=1
  604 IF(DASHIX.AND.IPER.EQ.2)GO TO 615
      CALL PLOT(XUX,YUY,IPER)
      GO TO 609
  615 CALL PLOT(XUX,YUY,IOPLOT)
```

```
      IF(ICPLOT.EQ.3)GO TO 630
      NHARD=NHARD-1
      IF(NHARD.GT.0)GO TO 609
      ICPLOT=3
      NHARD=LDASH1
      GO TO 609
  630 NSOFT=NSOFT-1
      IF(NSOFT.GT.0)GO TO 609
      ICPLOT=2
      NSOFT=LDASH2
  609 IPER=2
      NUNC=NUNC+1
      IF(.NOT.DOLABS.OR.YUY.LE.YMAX)GO TO 610
      YMAX=YUY
      XMAX=XUX
      GO TO 610
  608 IPER=3
      IF(LABLIT.NE.1)GO TO 610
      CALL ENDER(X-WEST,Y-SOUTH,PVAL,1)
      LABLIT=0
  610 CONTINUE
C
C         END ADJUST AND PLOT SECTION
C
      NUGG=NUGG+NUNC
      TOT=TOT+QINC*FLOAT(NUNC)
      IF(TOT.LT.TMAX)GO TO 790
      WRITE(6,462)PVAL,TMAX
  462 FORMAT(/,2X,16HREACHED TMAX ON ,2F12.4)
      GO TO 500
  790 IF(LLPE.EQ.NRINC)GO TO 800
      HYPM=HYPFOR
      SENT=SOUT
      CENT=COUT
      ENTIR=EXET
      SANG=SINFOR
      CANG=COSFOR
      XBEG=XEND
      YBEG=YEND
      XEND=XIND
      YEND=YIND
  800 CONTINUE
C
C         END MAIN CURVILINEAR INTERPOLATE AND PLOT LOOP
C
  500 IF(.NOT.CLOS.OR..NOT.DOLABS.OR.YMAX.LT..01)GO TO 501
      XPP=XMAX+WEST
      YPP=YMAX+SOUTH
      CALL ENDER(XMAX,YMAX,PVAL,2)
  501 IF(.NOT.CLOS.AND.LABLIT.EQ.1)CALL ENDER(XUX,YUY,PVAL,1)
      IF(NUGG.LE.MOSINC)GO TO 502
      MOSINC=NUGG
      VALLIN=PVAL
  502 RETURN
      END
```

```
*FLOW(1).ENDER ELT CREATED ON 8 MAY 79 AT 10:17:51
            SUBROUTINE ENDER(X,Y,PVAL,ICO)
            COMMON /CENDEO/HONUM,WODE,HOGH,XLAS,YLAS
            DIMENSION D(3)
      C
      C        THIS SUBROUTINE IS CALLED TO LABEL CONTOURS
      C
            DM=SQRT((X-XLAS)*(X-XLAS)+(Y-YLAS)*(Y-YLAS))
            IF(DM.LT.2.*HONUM.OR.HONUM.LE.0.)GO TO 25
            IF(ICO.EQ.2)GO TO 14
            D(1)=ABS(Y-HOGH)
            D(2)=ABS(X-WODE)
            D(3)=ABS(Y)
            K=1
            DM=ABS(X)
            DO 10 I=1,3
            IF(D(I).GE.DM)GO TO 10
            DM=D(I)
            K=I+1
         10 CONTINUE
            GO TO(12,14,16,18),K
         12 YAD=-HONUM/2.
            I=1
            GO TO 100
         14 YAD=.02
            I=2
            GO TO 100
         16 XAD=.02
            YAD=-HONUM/2.
            GO TO 20
         18 YAD=-HONUM-.02
            I=2
            GO TO 100
         20 XAD=X+XAD
            YAD=Y+YAD
            CALL NUMBER(XAD,YAD,HONUM,PVAL,0.,+2)
            CALL PLOT(X,Y,3)
            XLAS=X
            YLAS=Y
         25 RETURN
        100 XAD=-.75*HONUM
            IF(PVAL.GE.9.5.OR.PVAL.LE.(-.5))XAD=XAD-HONUM
            IF(PVAL.GE.99.5.OR.PVAL.LE.(-9.5))XAD=XAD-HONUM
            IF(PVAL.GE.999.5.OR.PVAL.LE.(-99.5))XAD=XAD-HONUM
            IF(I.EQ.2)XAD=.5*XAD
            GO TO 20
            END
```

```
*FLOW(1).FIT FOR CREATED ON 4 DEC 79 AT 10:02:34
      C*****************************************************************
      C  FITS A FARABOLA TO THREE POINTS, USED IN MAKING PLOT.
      C*****************************************************************
            SUBROUTINE FIT(Z1,Z2,Z3,E1,E2,E3,A,B,C)
            D=(Z1**2)*(Z2-Z3)-Z1*(Z2**2-Z3**2)+Z3*(Z2**2)-Z2*(Z3**2)
            A=(E1*(Z2-Z3)-Z1*(E2-E3)+E2*Z3-E3*Z2)/D
            B=((Z1**2)*(E2-E3)-E1*(Z2**2-Z3**2)+E3*(Z2**2)-E2*(Z3**2))/D
            C=((Z1**2)*(Z2*E3-Z3*E2)-Z1*(E3*(Z2**2)-E2*(Z3**2))+E1*(Z3*(Z2**2)
           C-Z2*(Z3**2)))/D
            RETURN
            END
```

```
*FLOW(1).VECT FOR CREATED ON 4 DEC 79 AT 11:33:19
      C*****TO DRAW VELOCITY VECTOR BY CALLING CALCOMP SUBROUTINE AROHD*****
            SUBROUTINE VECT(AI,AAI,AJ,AAJ,ARMIN,ARMAX,COSF,SINF)
            YW=.2*SQRT((AAI-AI)**2+(AAJ-AJ)**2)
            YW=AMAX1(ARMIN,AMIN1(ARMAX,YW))
            BI=AI*COSF+AJ*SINF
            BJ=-AI*SINF+AJ*COSF
            BBI=AAI*COSF+AAJ*SINF
            BBJ=-AAI*SINF+AAJ*COSF
            CALL AROHD(BI,BJ,BBI,BBJ,YW,0.,12)
            RETURN
            END
```

```
*FLOW(1).CAPTN1 ELT CREATED ON 14 SEP 79 AT 10:29:24
            SUBROUTINE CAPTN1(P1,P2,P3,P4,P5,P6,Q7,Q8)
      C******WRITE COMMON HEADING FOR EACH PLOT WHETHER TEMP OR VELOCITY******
      C*****[(MONTH DAY, YEAR)] NEEDS RESET*****
            CALL SYMBOL(0.8,6.70,0.10,25HTIME(JUNE 20, 1978):        ,0.0,25)
            CALL NUMBER(4.0,6.70,0.10,P1,0.0,+1)
            CALL SYMBOL(0.8,6.50,0.10,25HWIND SPEED(CM/SEC):        ,0.0,25)
            CALL NUMBER(4.0,6.50,0.10,P2,0.0,+1)
            CALL SYMBOL(0.8,6.30,0.10,25HWIND DIRECTION(DEG/N):     ,0.0,25)
            CALL NUMBER(4.0,6.30,0.10,P3,0.0,+0)
            CALL SYMBOL(0.8,6.10,0.10,25HAIR TEMPERATURE(DEG-C):    ,0.0,25)
            CALL NUMBER(4.0,6.10,0.10,P4,0.0,+1)
            CALL SYMBOL(0.8,5.90,0.10,25HDISCHARGE TEMP(DEG-C):     ,0.0,25)
            CALL NUMBER(4.0,5.90,0.10,P5,0.0,+1)
            CALL SYMBOL(0.8,5.70,0.10,25HDISCH FLOWRATE(CUM/SEC):   ,0.0,25)
            CALL NUMBER(4.0,5.70,0.10,P6,0.0,+1)
            CALL SYMBOL(0.8,5.50,0.10,25HLENGTH SCALE(1CM= X CM):   ,0.0,25)
            CALL NUMBER(4.0,5.50,0.10,Q7,0.0,+0)
            CALL SYMBOL(0.8,5.30,0.10,25HVELOCITY SCALE(CM/SEC):    ,0.0,25)
            CALL NUMBER(4.0,5.30,0.10,Q8,0.0,+2)
            RETURN
            END
```

```
*FLOW(1).CAPTN2 SYM CREATED ON 11 SEP 79 AT 22:29:52
          SUBROUTINE CAPTN2(N)
C*****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM FOR CALCULATED UV****
          CALL SYMBOL(1.0,-1.8,.14,24HFIG      SURFACE VELOCITY,.0,24)
          CALL SYMBOL(1.0,-2.2,.14,29HANCLOTE ANCHORAGE BY MODELING,.0,29)
          CALL PLOT(-1.0,-2.5,3)
          CALL PLOT(-1.0,+7.5,2)
          CALL PLOT(+6.5,+7.5,2)
          CALL PLOT(+6.5,-2.5,2)
          CALL PLOT(-1.0,-2.5,2)
          RETURN
          END
```

```
*FLOW(1).CAPTN3 SYM CREATED ON 11 SEP 79 AT 22:31:55
          SUBROUTINE CAPTN3(N)
     C****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM AND BLOCK IT****
          CALL SYMBOL(1.0,-2.0,.14,29HFIG        TEMPERATURE FROM IR,0.0,29)
          CALL PLOT(-1.0,-2.5,3)
          CALL PLOT(-1.0,+7.5,2)
          CALL PLOT(+6.5,+7.5,2)
          CALL PLOT(+6.5,-2.5,2)
          CALL PLOT(-1.0,-2.5,2)
          RETURN
          END
```

```
*FLOW(1).CAPTN4 SYM CREATED ON 11 SEP 79 AT 22:36:39
              SUBROUTINE CAPTN4(SUM)
        C**** WRITE DEVIATION VALUE(FROM IR) ON ISOTHERM PLOT *******
              CALL SYMBOL (0.8,-0.5,.10,23HDEVIATION FROM IR TEMP:,0.,23)
              CALL NUMBER(4.C,-0.5,.10,SUM,0.,+3)
              RETURN
              END
```

```
*FLOW(1).CAPTNS SYM CREATED ON 11 SEP 79 AT 22:39:49
        SUBROUTINE CAPTNS(N,MIDEW)
C*****WRITE TIDAL STAGE AT THAT TIME*****
        DIMENSION MIDEW(4)
        IBCD=MIDEW(N)
        CALL SYMBOL(0.8,-.7,0.10,IBCD,0.,4)
        CALL SYMBOL(1.3,-.7,0.10,4HTIDE,0.,4)
        RETURN
        END
```

```
*FLOW(1).CAPTN6 SYM CREATED ON 11 SEP 79 AT 22:41:51
            SUBROUTINE CAPTN6(N)
      C*****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM FOR CALCULATED UW****
            CALL SYMBOL(1.0,-1.8,.14,19HFIG      UW VELOCITY,.0,19)
            CALL SYMBOL(1.0,-2.2,.14,29HANCLOTE ANCHORAGE BY MODELING,.0,29)
            CALL PLOT(-1.0,-2.5,3)
            CALL PLOT(-1.0,+7.5,2)
            CALL PLOT(+6.5,+7.5,2)
            CALL PLOT(+6.5,-2.5,2)
            CALL PLOT(-1.0,-2.5,2)
            RETURN
            END
```

```
*FLOW(1).CAPTN7 SYM CREATED ON 11 SEP 79 AT 22:43:36
      SUBROUTINE CAPTN7(N)
C*****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM FOR CALCULATED VW****
      CALL SYMBOL(1.0,-1.8,.14,19HFIG      VW VELOCITY,.0,19)
      CALL SYMBOL(1.0,-2.2,.14,29HANCLOTE ANCHORAGE BY MODELING,.0,29)
      CALL PLOT(-1.0,-2.5,3)
      CALL PLOT(-1.0,+7.5,2)
      CALL PLOT(+6.5,+7.5,2)
      CALL PLOT(+6.5,-2.5,2)
      CALL PLOT(-1.0,-2.5,2)
      RETURN
      END
```

```
*FLOW(1).CAPTN8 SYM CREATED ON 11 SEP 79 AT 22:45:43
            SUBROUTINE CAPTN8(N)
      C*****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM FOR CALCULATED ELEVATION****
            CALL SYMBOL(1.0,-1.8,.14,25HFIG      SURFACE ELEVATION,.0,25)
            CALL SYMBOL(1.0,-2.2,.14,29HANCLOTE ANCHORAGE BY MODELING,.0,29)
            CALL PLOT(-1.0,-2.5,3)
            CALL PLOT(-1.0,+7.5,2)
            CALL PLOT(+6.5,+7.5,2)
            CALL PLOT(+6.5,-2.5,2)
            CALL PLOT(-1.0,-2.5,2)
            RETURN
            END
```

```
*FLOW(1).CAPTN9 SYM CREATED ON 11 SEP 79 AT 22:52:41
            SUBROUTINE CAPTN9(N)
      C*****WRITE PLOT TITLE ON BOTTOM OF DIAGRAM FOR CALCULATED TEMPERATURE****
            CALL SYMBOL(1.0,-1.8,.14,27HFIG      SURFACE TEMPERATURE,.0,27)
            CALL SYMBOL(1.0,-2.2,.14,29HANCLOTE ANCHORAGE BY MODELING,.0,29)
            CALL PLOT(-1.0,-2.5,3)
            CALL PLOT(-1.0,+7.5,2)
            CALL PLOT(+6.5,+7.5,2)
            CALL PLOT(+6.5,-2.5,2)
            CALL PLOT(-1.0,-2.5,2)
            RETURN
            END
```

TECHNICAL REPORT DATA *(Please read Instructions on the reverse before completing)*		
1. REPORT NO. EPA-600/7-82-037b	2.	3. RECIPIENT'S ACCESSION NO.
4. TITLE AND SUBTITLE Verification and Transfer of Thermal Pollution Model; Volume II. User's Manual for Three-dimensional Free-surface Model		5. REPORT DATE May 1982 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) S.S.Lee, S.Sengupta, S.Y.Tuann, and C.R.Lee		8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS The University of Miami Department of Mechanical Engineering P.O. Box 248294 Coral Gables, Florida 33124		10. PROGRAM ELEMENT NO. 11. CONTRACT/GRANT NO. EPA IAG-78-DX-0166*
12. SPONSORING AGENCY NAME AND ADDRESS EPA, Office of Research and Development Industrial Environmental Research Laboratory Research Triangle Park, NC 27711		13. TYPE OF REPORT AND PERIOD COVERED Final; 3/78-9/80 14. SPONSORING AGENCY CODE EPA/600/13
15. SUPPLEMENTARY NOTES IERL-RTP project officer is Theodore G.Brna, Mail Drop 61, 919/541-2683. (*) IAG with NASA, Kennedy Space Center, FL 32899, subcontracted to U. of Miami under NASA Contract NAS 10-9410.		
16. ABSTRACT The six-volume report: describes the theory of a three-dimensional (3-D) mathematical thermal discharge model and a related one-dimensional (1-D) model, includes model verification at two sites, and provides a separate user's manual for each model. The 3-D model has two forms: free surface and rigid lid. The former, verified at Anclote Anchorage (FL), allows a free air/water interface and is suited for significant surface wave heights compared to mean water depth; e.g., estuaries and coastal regions. The latter, verified at Lake Keowee (SC), is suited for small surface wave heights compared to depth (e.g., natural or man-made inland lakes) because surface elevation has been removed as a parameter. These models allow computation of time-dependent velocity and temperature fields for given initial conditions and time-varying boundary conditions. The free-surface model also provides surface height variations with time. The 1-D model is considerably more economical to run but does not provide the detailed prediction of thermal plume behavior of the 3-D models. The 1-D model assumes horizontal homogeneity, but includes area-change and several surface-mechanism effects.		
17. KEY WORDS AND DOCUMENT ANALYSIS		
a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
Pollution Thermal Diffusivity Mathematical Models Estuaries Lakes Plumes	Pollution Control Stationary Sources	13B 20M 12A 08H,08J 21B
18. DISTRIBUTION STATEMENT Release to Public	19. SECURITY CLASS *(This Report)* Unclassified 20. SECURITY CLASS *(This page)* Unclassified	21. NO. OF PAGES 152 22. PRICE

EPA Form 2220-1 (9-73)