VARIABLE THICKNESS TRANSIENT
GROUNDWATER FLOW
MODEL USER'S MANUAL

Battelle

Pacific Northwest Laboratories
Richland, Washington 99352

1974

Prepared for the U.S. Atomic Energy
Commission under Contract AT(45-1):1830

NOTICE

The report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

PACIFIC NORTHWEST LABORATORY
operated by
BATTELLE
for the
U.S. ATOMIC ENERGY COMMISSION
Under Contract AT(45-1)-1830

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy \$5.45; Microfiche \$1.45

3 3679 00062 2516

VARIABLE THICKNESS TRANSIENT GROUNDWATER
FLOW MODEL USER'S MANUAL

by

W. V. DeMier*
A. E. Reisenauer
K. L. Kipp

Water and Land Resources Department
*Systems Department

1974

This report is based on work sponsored by the
Atlantic Richfield Hanford Company under United
States Atomic Energy Commission Contract AT(45-1)-2130

BATTELLE
PACIFIC NORTHWEST LABORATORIES
RICHLAND, WASHINGTON 99352

SUMMARY

The Variable Thickness Transient Flow Model (VTT) is a numerical solution to the partial differential equation describing unconfined saturated transient flow through porous media. This report describes the input to the model and two support programs used during a groundwater aquifer simulation. In addition to the program listings and user and input instructions, specific features relating to modeling of the Hanford Reservation groundwater regime are described.

CONTENTS

	<u>Page</u>
SUMMARY.	i
INTRODUCTION	1
VARIABLE THICKNESS TRANSIENT MODEL PROGRAM	3
Input Data Files.	3
Data File INJ (Injections)	4
Data File PKF (River Elevation Interpolation Table)	4
Data File BIG (Main Data).	6
Calculation Type Designation.	6
Card Deck Preparation	21
Control Card Requirements	22
Support Program CLASSY.	24
ACKNOWLEDGMENTS.	25
APPENDIX A - VTT SOURCE PROGRAM LISTING AND SAMPLE INPUT DATA LIST.	A-1
APPENDIX B - SUBROUTINE RIVEL.	B-1
APPENDIX C - SUPPORT PROGRAM CLASSY.	C-1

VARIABLE THICKNESS TRANSIENT GROUNDWATER MODEL

USER'S MANUAL

INTRODUCTION

The Variable Thickness Transient Model (VTT) for saturated groundwater flow consists of a main and support programs for the CYBER-74 computer. The theory and numerical implementation of this model are described in a companion report, "Variable Thickness Transient Groundwater Flow Model - Theory and Numerical Implementation" (BNWL-1703), which covers the program details and user instructions for the successful operation of these programs. The proper relationship of this model to the overall modeling effort is shown in Figure 1.

FIGURE 1. Hydrological Modeling Program Components

The VTT groundwater model was developed under the Radionuclides in Soils Program to predict the changing height of the water table (phreatic) surface throughout the unconfined groundwater aquifer underlying the Hanford Reservation. The partial differential equations of flow and continuity used in the VTT to describe the saturated flow in the Hanford groundwater aquifer have an infinite number of solutions for any given set of the aquifer bottom surface, permeability, and storage coefficient distributions. The solution is uniquely determined only when events at the boundaries of the region being investigated for a particular set of conditions are known. These known conditions along the boundaries of the region are called the boundary conditions of the problem.

The set of initial conditions must also be known to solve the groundwater flow problem. Initial conditions consist of the values of the dependent variable (in the VTT case, the height of the phreatic surface) everywhere within the region being investigated.

When the boundary conditions, initial conditions and other input provide a unique solution to the set of partial differential equations which describe groundwater flow, then the problem is well posed.

The purpose of this document is to help the user to provide, in a systematic manner, the data necessary to use the program.

The partial differential equation used in the VTT model to simulate groundwater flow is known as the Boussinesq equation of unsteady flow (Equation 1):

$$\nabla \cdot [K (h-H^o) \nabla h] = \sigma \frac{\partial h}{\partial t} - q \quad (1)$$

where:

- $q > 0$ = infiltration
- $h (x,h,t)$ = the elevation of the phreatic surface
- ∇ = the two dimensional (x,y) gradient operator
- $h^o (x,y)$ = the aquifer bottom elevation
- $K (x,y)$ = the hydraulic conductivity
- $\sigma (x,y)$ = the storage coefficient

This equation was developed from the Dupuit or Boussinesq approximation to the generalized flow equation and was derived by averaging in the vertical direction Laplace's equation which governs incompressible, Darcian groundwater flow in three dimensions. Thus, the independent spatial variables are in the horizontal plane and the nonlinear, free-surface boundary conditions are incorporated into the governing differential equations. Furthermore, any flux resulting from vertical infiltration from the partially saturated zone above the free surface appears as a source term in the Boussinesq equation. All aquifer properties are represented by their average values over the vertical thickness of saturated flow.

Two programs are described in this report: the main VTT program and one support program, CLASSY. The CLASSY program evaluates the quality of a given simulation by comparing calculated with measured potential surfaces.

VARIABLE THICKNESS TRANSPORT MODEL PROGRAM

The main program for the VTT model was written in FORTRAN-V as implemented for the CSCX system on the UNIVAC-1108 and was later converted to the CYBER-74 System 3.4 for the Control Data CDC6600. Program execution requires approximately 50,000 words of central memory. In addition to the system OUTPUT file, five other files are required by the program. They are: PKF, INJ, BIG, POUT, PIN. Files PKF, INJ, and BIG contain input data. File POUT receives the computed potential arrays that have been specified for saving. File PIN contains the saved potential arrays from a previous run which are to be used as a starting point for the current run. Note that only control cards and update cards are physically present in the card deck. All other input is from card images in the disk (or tape) files.

Input Data Files

All input to the program comes through card image files which are linked to the program by logical file name (LFN). Logical file INJ contains the injection data, Q. Logical file PKF contains the interpolation table for use in computing river elevations. Logical file PIN contains saved potentials from a previous run. (If the current run is the start-up run then file PIN is not referenced.) Logical file BIG contains all other data required for the run. Generally, only file BIG will require updating from one run to the next in a series. Details of data file preparation are given below. The Card Types referred to are described in the section Input Card Image Description.

Data File INJ (Injections). The values of Q (injections) are input to the model from logical file INJ. The file is made up of one Card Type 24 followed by as many Type 25 cards as are required to list IQ values of Q for each month (see Card Type 4, Field 9). Note that Q(1) for each month always appears in the first field of a Card Type 24, not on the same card as Q(IQ) for the previous month.

To simplify preparation of input for a continued series of runs the file of injection values for the entire series is set up at the time of the initial run, and the program (through input variable MP) locates the proper starting position in INJ for succeeding runs. The units of Q are cubic feet per day for the current simulation month.

Data File PKF (River Elevation Interpolation Table). The VTT model requires that the potential be specified at each boundary node along the river for each month in the simulation period. These potentials are computed and provided to the model via subroutine RIVEL (described in Appendix B). Data file PKF is a list of two part packed words required by subroutine RIVEL.

Each packed word, PK, comprises an integer portion, IPK, and a decimal fraction portion, DPK. The IPK part of PK is used as a control in the subroutine to direct flow to the proper arithmetic statement functions for computing elevations at the two exact 5 mile points along the river which bracket the node of interest. (River mile 335 is taken as the base.)

IPK is calculated in integer arithmetic by the expression:

$$IPK = \frac{\text{River Mile} - 335}{5} + 1$$

Thus for nodes at river mile 370 up to 375, the value of IPK is 8. At mile 375 IPK becomes 9.

The DPK part of PK is used to interpolate between 5-mile intervals along the river. The derivation of DPK for river miles 370 through 375 is shown by the example in Figure 2. The x,y coordinates represent node centers in the 2000 foot grid system. Taking the side of one square to be one unit, the point-to-point distances are summed from node 38,79 to node 31,87 (12.9 units in the example). DPK for a given node x,y is simply the point-to-point distance from node 38,79 to node x,y divided by 12.9. This interpolation factor forms the decimal fraction portion of PK. The elevation E for a node between 38,79 and 31,87 is found by:

$$E_{(x,y)} = E_{(38,79)} + (DPK) \cdot (E_{(31,87)} - E_{(38,79)})$$

EXAMPLE OF PK CONSTRUCTION

Total distance point to point from mile
370 to mile 375 = $(3 + 7\sqrt{2})$ units = 12.9

<u>ILOC</u>	<u>NODE</u>	<u>INTERPOLATION FACTOR</u>	<u>PK (ILOC)</u>
83	38,79	0.0	8.0000
84	39,80	$\sqrt{2}/12.9 = 0.1096$	8.1096
85	39,81	$(1 + \sqrt{2})/12.9$	8.1871
86	38,82	$(1 + 2\sqrt{2})/12.9$	8.2968
87	37,83	$(1 + 3\sqrt{2})/12.9$	8.4064
88	36,84	$(1 + 4\sqrt{2})/12.9$	8.5160
89	35,85	$(1 + 5\sqrt{2})/12.9$	8.6257
90	34,86	$(1 + 6\sqrt{2})/12.9$	8.7353
91	33,86	$(2 + 6\sqrt{2})/12.9$	8.8128
92	32,87	$(2 + 7\sqrt{2})/12.9$	8.9224
93	31,87	$(3 + 7\sqrt{2})/12.9$	9.0000

FIGURE 2. PK Table Construction for VTT Model Input

Values of PK for use with 1000-foot and 2000-foot grids are listed in Table 1.

Data File BIG (Main Data). Most of the input data for the model is read from BIG. The various input/output options are specified in this file and the file configuration is dependent upon the options selected. The flow diagram in Figure 3 may be used as a guide to setting up the card images. Note that values of starting potential (PHI), aquifer bottom elevation (H), and hydraulic conductivity (K) may be entered by "matrix" or by "column-of-change." (A third option, input from saved output on File PIN, is available for PHI only.) The matrix option is useful for matrices in which all or most elements differ in value. The values are listed in order $V_{1,1}$, $V_{1,2}$, $\dots$, $V_{1,n}$, $V_{2,1}$, $V_{2,2}$, $\dots$, $V_{m,n}$, seven values per card. The column-of-change input option is useful when many adjacent nodes have identical values. Two kinds of cards are required with this option. Card Type 27 specifies the matrix rows to which values on the next card(s) apply. The second card (Type 28, 29, or 30) specifies starting columns and values to be applied to nodes in those columns. A listing of the main VTT program and a sample input data list appears in Appendix A. Table 2 contains descriptions for the input card images.

Calculation Type Designation

Figure 4 shows the Hanford VTT model grid for the unconfined aquifer, including the boundary conditions. The VTT model requires the adoption of a uniform x-y grid system. The one chosen for the Hanford model was a Cartesian x-y coordinate system with the y axis oriented north-south with a 2000 foot spacing between nodes in either direction. The origin was chosen as a point 69,000 feet south and 109,000 feet west in the Hanford Project coordinate grid. With this grid system, there are 68 nodal points on each x line and 87 nodal points on each y line. The VTT code requires that each node within this 68 x 87 matrix be marked as:

1. within the Hanford aquifer,
2. external to the Hanford aquifer,
3. a water boundary for the Hanford aquifer
[$h = H(x,y,t)$],
4. an impermeable boundary for the Hanford aquifer
 $\left(\frac{\partial h}{\partial n} = 0\right)$.

(Boundary nodes where the flux is known are treated as the mathematical equivalent, which is an impermeable node with a proper accretion term.)

Table 1 (Cont'd.)

	*****	PK	TABLE	FOR 2000 FT	GRID		
1.	0.0	1.000	1.072	1.1737	1.2754	1.3772	2PKEY001
2.	1.5807	1.6824	1.7544	1.8263	1.9281	2.0000	2PKEY002
3.	2.1446	2.2169	2.2893	2.3915	2.4638	2.5632	2PKEY003
4.	2.7831	2.8554	2.9277	3.0000	3.0769	3.1538	2PKEY004
5.	3.3977	3.3846	3.4615	3.5385	3.6154	3.6923	2PKEY005
6.	3.8462	3.9231	4.0000	4.0755	4.1823	4.2576	2PKEY006
7.	4.4401	4.5156	4.5192	4.6667	4.7422	4.8177	2PKEY007
8.	5.0000	5.0720	5.1737	5.2754	5.3474	5.4491	2PKEY008
9.	5.6520	5.7246	5.8263	5.9281	6.0000	6.0924	2PKEY009
10.	6.2500	6.3423	6.4077	6.5000	6.5923	6.6847	2PKEY010
11.	6.8420	6.9070	7.0000	7.0775	7.1872	7.2968	2PKEY011
12.	7.4939	7.5615	7.6711	7.7307	7.8903	8.0000	2PKEY012
13.	8.1872	8.2968	8.4064	8.5161	8.6257	8.7353	2PKEY013
14.	8.9225	9.0000	9.1122	9.2244	9.3037	9.4159	2PKEY014
15.	9.6402	9.7524	9.8317	9.9439	10.0462	10.1385	2PKEY015
16.	10.3232	10.4156	10.4809	10.5732	10.6385	10.7038	2PKEY016
17.	10.8885	10.9538	11.0592	11.1429	11.2265	11.3102	2PKEY017
18.	11.4776	11.5959	11.6796	11.7980	11.9163	12.0000	2PKEY018
19.							

TABLE 2. INPUT CARD IMAGE DESCRIPTION

CARD TYPE 1	Format 14I5
Field 1	<u>MINUTE</u> The maximum number of minutes to be allowed for this run before saving restart data and terminating.
Field 2	<u>MONO</u> The calendar month number at the start of this run (Jan=1, Feb=2, etc.).
Field 3	<u>MP</u> If this run is one of a continuous series of runs in which the output potential from one run is the starting potential for the next, then MP is the total number of months elapsed in the series at the beginning of this run. Otherwise MP is zero or blank.
Field 4	<u>MAXI</u> The number of rows in the potentials matrix (the number of nodes in the Y direction).
Field 5	<u>MAXK</u> The number of columns in the potentials matrix (the number of nodes in the X direction).
Field 6	<u>KRES</u> A restart signal for steady state runs. If KRES = 1 this is a restart and saved data will be read from Tape 15. Note that this restart option is available for the steady state case only. Restart data will be saved on file POUT.
CARD TYPE 2	Format 14I5
Fields 1,3,5,...,13	<u>NS(I)</u> The column number at which to begin calculations for row I.
Fields 2,4,6,...,14	<u>NE(I)</u> The column number at which to stop calculations for row I.
CARD TYPE 3	Format I3, 6I1, 10A6
Field 1	<u>NINPHI</u> Option specification for starting potentials input: <0 specifies input by matrix option 0 specifies input by column of change option

Table 2 (Continued)

	>0 specifies input from time plane number from a file written by an earlier run.
Field 2	<u>INH</u> Option specification for aquifer bottom elevation input: 0 specifies loading by matrix option 1 specifies loading by column of change option
Field 3	<u>INK</u> Option specification for permeability input: 0 specifies loading by matrix option 1 specifies loading by column of change option
Field 4	<u>NBUG</u> Option specification for printout: 1 specifies printout of title, time, potentials, time plane number, and number of iterations required for the time plane 2 specifies printout of high differences for each iteration, in addition to the information printed when NBUG=1 3 specifies printout of potentials, permeabilities, high differences at each iteration and termwise calculation information for each node; this is used for program debug printout only
Field 5	<u>NSTEP</u> Node sequence control: 1 (not currently implemented) calculation sequence is from bottom to top within each column, left to right by columns 2 calculation sequence is bottom to top, left to right 3 (not currently implemented) calculated sequence is from an input-specified row downward in each column, left to right by columns, then from this row upward and left to right
Field 6	<u>IPUN</u> Printout option for initial conditions: 0 suppresses reporting of initial conditions >0 prints initial conditions
Field 7	<u>N1</u> Transient/steady-state selector: 0 specifies operation in transient flow mode 1 specifies operation in steady-state mode
Fields 8-17	<u>CODE</u> Report title, 60 alphameric characters

Table 2 (Continued)

CARD TYPE 4	Format 14I5
Field 1	<u>MAXI</u> The number of rows in the system (nodes in y direction)
Field 2	<u>MAXK</u> The number of columns in the system (nodes in x direction)
Field 3	not currently used
Field 4	<u>N5</u> The maximum number of iterations allowed for any time plane
Field 5	<u>JP</u> The number of the first time plane for which potentials will be saved
Field 6	<u>JDELP</u> Time plane save frequency. Every <u>JDELP</u> th time plane after the <u>JP</u> th one will be saved
Field 7	<u>NP</u> The number of last time plane to be done in this run
Field 8	<u>NDUM1</u> The number of the first time plane to be printed in this run
Field 9	<u>IQ</u> The number of values of Q (injections) to be read for each time plane
Field 10	<u>IST0</u> The number of storage coefficients to be read
CARD TYPE 5	Format 7E10.4
Field 1	<u>DELZ</u> Distance between nodes, feet
CARD TYPE 6	Format 2F5.0, I5, 16F4.0
Field 1	<u>DELMAX</u> Print frequency control. Time planes will be printed at <u>DELMAX</u> intervals after time plane <u>NDUM1</u> is printed (see Type 4, Field 8)
Field 2	<u>TIME</u> The time in days at the start of this run. For the initial run of a series <u>TIME</u> =0. For continuation runs <u>TIME</u> =the time corresponding to the restart plane used

Table 2 (Continued)

Field 3	<u>NDELT</u> = 1+N where N is the number of DELT values to be read from remaining fields on this card
Field 4	<u>DELT(I)</u> The time increment list to be used. The summation of DELT(I) from I=1 to I=NDELT-1 must not exceed 27 days. The program computes DELT(NDELT), the last time step in the cycle, so as to exactly complete each current calendar month, then starts at the beginning of the DELT list again
CARD TYPE 7	Format 7E10.0
Field 1	<u>TPDIF</u> The run is complete when the absolute difference between successive time planes is less than or equal to TPDIF
Field 2	<u>THEMAX</u> Maximum allowable potential. Program takes error exit if a computed potential exceeds THEMAX
Field 3	<u>P1</u> Convergence criterion within a time plane. When the highest difference between the potentials at the N th iteration and the (n-1) th iteration is less than or equal to this value, the time plane is considered complete
CARD TYPE 8	NAMelist INITIALIZATION This card signals the beginning of NAMelist input. It always contains the characters \$MODIFY in columns 2 through 8 and blanks in all other columns. It must always be present even if Card Type 9 is omitted
CARD TYPE 9	NAMelist INPUT This card is for use in modifying the data base in continuation cases. Valid variable names are: NBUG, N5, JP, JDELP, NP, NDUM1, DELT, NDELT, DELMAX, TPDIF, THEMAX, P1, IPUN, TIME

Table 2 (Continued)

CARD TYPE 10	NAMELIST TERMINATION
	This card signals the end of NAMELIST input. It always contains \$END beginning in column 2 and blanks in all other columns. It must be present even if Card Type 9 is omitted
CARD TYPE 11	Format 7E10.0
Fields 1-7	<u>ST0(I)</u> Values of storage coefficients. Any number of cards as required to supply IST0 values of ST0 (see Field 10, Card Type 4).
CARD TYPE 12	Format 7E10.0
Fields 1-7	<u>PHI(J,I)</u> Values of initial potentials. Any number of cards as required to supply MAXI*MAXK values of PHI. This card type is omitted unless NINPHI<0 (see Type 3). The first MAXK values of PHI are for row 1, the next MAXK values for row 2, etc.
CARD TYPE 13	Format 7E10.0
Fields 1-7	<u>H(J,I)</u> Values of aquifer bottom elevation. Any number of cards as required to supply MAXI*MAXK values of H. This card type is omitted unless INH=0 (see Type 3). The first MAXK values of H are in row 1, the next MAXK values for row 2, etc.
CARD TYPE 14	Format 7E10.0
Fields 1-7	<u>SATK(J,I)</u> Values of permeability. Any number of cards as required to supply MAXI*MAXK values of SATK. This card type is omitted unless INK=0 (see Type 3). The first MAXK values of SATK are for row 1, the next MAXK values for row 2, etc.
CARD TYPE 15	Format 313
Field 1,2,3	<u>IB, IT, KKK</u> The index pairs on the next Type 17 card(s) apply to matrix rows IB

Table 2 (Continued)

through IT. There are KKK index pairs to be read from the next Card Type 17 card(s).

CARD TYPE 16 Format 10(I3, I4)

Pairs of
Fields
1-2,3-4,
...,9-10

KEY(J), KTY(J) The pairs of KEY(J), KTY(J) for J=1 to KKK are used to link a value from the array ST0 to a column or range of columns in rows IB through IT (see Type 14).

KEY₁ is the first column in the matrix and KTY₁ is the index to the value of ST0 from the table established inputting Card Type 12. KEY_{J+1} is the column number where the index number KTY changes. KKK is the number of index changes in rows IB through IT.

The effect of the list:

KEY_J,KTY_J,KEY_{J+1},KTY_{J+1},.....,KEY_{KKK},KTY_{KKK} is to place an index number in every matrix node that points to a value in the ST0 list established in Card Type 12.

Card Types 15 and 16 are repeated in sets of one Type 15 and one or more Type 16 until IT=MAXI

CARD TYPE 17 Format 313

Fields
1,2,3

IB, IT, KKK The index pairs on the next Type 19 card(s) apply to matrix rows IB through IT. There are KKK pairs of KEY, KTY to be read from the next Card Type 19 card(s)

CARD TYPE 18 Format 14(I3,02)

Pairs of
Fields
1-2,3-4,
...,13-14

KEY(J), KTY(J) The pairs of KEY(J), KTY(J) for J=1 to KKK are used to specify the calculation types for nodes in rows IB through IT

KEY₁ is the first column in the matrix and KTY₁ is the index to the type of calculation to be made at that matrix node. (See section on Boundary Type Selection.) KEY_{J+1} is the column number where the index KTY_{J+1} changes. KKK is the number of changes in rows IB through IT.

Table 2 (Continued)

The effect of the list:

KEY_J, KTY_J, KEY_{J+1}, KTY_{J+1}, ..., KEY_{KKK},
 KTY_{KKK} is to assign calculation types to
 every matrix node when Card Types 17 and 18
 are repeated in sets of one Type 17 and one
 or more Type 18 until IT=MAXI

CARD TYPE 19

Format 13A6, A2

Fields
 1-14

This card may contain up to 80 alphameric
 characters. The card is for convenience of
 the user in marking the beginning of the QLOC
 table (see Card Type 20) in the input deck.
 It may be blank but must be present.

CARD TYPE 20

Format 18I4

Fields
 1,2,3,
 ...,16,17,18

These cards form the QLOC table, showing which
 value in the list of injections (see Card Type
 24) is assigned to node I, J.

Sets of I, J, KTY, six sets per card.
 I, J = node row, node column.
 KTY = ordinal position in injection table
 (Card Type 24) at which the injection value
 for this node can be found.

CARD TYPE 21

Format 13A6, A2

Fields
 1-14

This card may contain up to 80 alphameric
 characters. It marks the beginning of the
 river flow rate array. It may be blank but
 must be present.

CARD TYPE 22

Format 7E10.0

Fields
 1-7

FLOW(N) Values of river flow rate (cubic
 feet per day). FLOW(1) is the flow rate dur-
 ing the first month of this run. The last
 Type 22 card must contain -1.1E+38 in Field 1.

Table 2 (Continued)

CARD TYPE 23	Format 12A6
Fields 1-12	Alphameric heading card for QIN file (injection or withdrawal values).
CARD TYPE 24	Format 7E10.0
Fields 1-7	<u>QIN(L)</u> List of injection values, cubic feet per day. A complete list of QIN(L) for L=1 to L=IQ (see Card Type 4, Field 9) must be present for each month, even if some values are zero for some months. The linkage of QIN to a specific node is made by the information in Card Type 20. The QIN values are put in the same order as the associated Q number on Card Type 20.
CARD TYPE 25	Format 12A6
Fields 1-12	Alphameric heading card for PK file. Convenient separator card must be present.
CARD TYPE 26	Format 7E10.0
Fields 1-7	<u>PK(I)</u> PK is a two-part packed word used by subroutine RIVEL for computing river elevations. The values of PK _I depend on the size of the mesh used in the system. Tables of PK for 2000 foot and 1000 foot mesh are given in the section describing data file PKF.
CARD TYPE 27	Format 3I3
Fields 1,2,3	<u>IB, IT, KKK</u> The values on the next Type 28 card(s) apply to rows IB through IT. There are KKK pairs of KEY and PLOP on the following Type 28 Card where PLOP is the starting or initial potential.
CARD TYPE 28	Format 7(I2, F8.0)

Table 2 (Continued)

<p>Pairs of Fields 1-2,3-4, ...,13-14</p>	<p><u>KEY(J), PLOP(J)</u> The pairs of KEY(J), PLOP(J) for J=1 to KKK are used to specify the initial potential for nodes in rows IB through IT. KEY₁ is the first column in the matrix and PLOP₁ is the value of the potential to be assigned at that column and all successive columns until column KEY_{J+1} is encountered. KEY_{J+1} is the column number where the value PLOP_{J+1} assignment begins. KKK is number of changes in the rows IB through IT.</p> <p><u>PLOP_K</u> KEY_K, PLOP_K, KEY_{K+1}, PLOP_{K+1}, ..., <u>K_{KKK}</u>, PLOP_{K_{KKK}} is to assign the starting potentials to every matrix node when Cards 28 and 29 are repeated in sets of one Card Type 28 and one or more Card Type 29 until IT=MAXI.</p>
<p>CARD TYPE 29</p>	<p>Format 7(I2, F8.0)</p> <p>This card is like the Type 28 card, except that aquifer bottom elevations are set equal to PLOP</p>
<p>CARD TYPE 30</p>	<p>Format 7(I2, F8.0)</p> <p>This card is like Type 28 card, except that permeability values are set equal to PLOP</p>
<p>CARD TYPE 31</p>	<p>Format 13A6, A2</p> <p>Alphanumeric heading for ILOC, a table used by subroutine RIVEL to compute river elevations (see Card Type 32).</p>
<p>Fields 1-14</p>	<p>Alphanumeric heading for ILOC, a table used by subroutine RIVEL to compute river elevations (see Card Type 32).</p>
<p>CARD TYPE 32</p>	<p>Format 18I4</p> <p>These cards form the ILOC table, showing which value in the list of PK values (File PKF, Table 1) is assigned to node I, J.</p> <p>Sets of I, J, ILOC, six sets per card. I, J = node row, node column. ILOC = ordinal position in File PKF (PK values) at which the interpolation factor for river elevation of this node can be found.</p>
<p>Fields 1,2,3, ...,16, 17, 18</p>	<p>Sets of I, J, ILOC, six sets per card. I, J = node row, node column. ILOC = ordinal position in File PKF (PK values) at which the interpolation factor for river elevation of this node can be found.</p>

FIGURE 4. Illustration of the VTT Model Calculational Types for the Hanford Simulation

To facilitate the marking of nodal types and to avoid right angles which cause stagnation points and singularities in the mathematical solution of Equation 1, a systematic method of representing interior, exterior and boundary nodes has been adopted. Figure 5 illustrates the different kinds of nodal types used in the VTT code.

FIGURE 5. Nodal Types Used in VTT Model

There are four basic nodal types, with variations to handle the various shapes and orientations of the impermeable boundaries:

- is a water boundary, i.e., held potential boundary node [$h = H(x,y,t)$].
- is an external node outside the aquifer.
- is an internal or nonboundary node which lies within the aquifer.
- these basic shapes with all their possible rotations are used to represent the 24 kinds and shapes of impermeable nodes. The heavy lines represent the impermeable boundaries.
-
-
-

The numerical designation of each of these symbols is shown in Figure 5 in large print and the matrix position is shown as small numbers in brackets. The impermeable surface is designated by the heavy line on the side of the block.

Card Deck Preparation

Card input preparation begins with a series of program control cards which specify basic input options, matrix size, printout frequency, saved file frequency, size of the time step to be taken, and other controls. This is followed by the loading of several values at every point in the matrix grid. These values include:

- initial potentials
- hydraulic conductivity
- aquifer bottom elevation
- calculation type
- flux index
- storage coefficient

Specific instructions for preparation of data cards are given in Table 2. The deck setup flow chart is shown in Figure 3.

Potential, aquifer bottom and hydraulic conductivity may be loaded into the program in two ways:

1. If an entire matrix is available, the data is loaded by matrix option.
2. If data are sparse, values may be entered by a column of change option.

The second option enables the user to designate the rows of a matrix which will be loaded with a series of values (to be entered in specified columns).

The column of change option is always used for the calculation types and the flux index designation.

Control Card Requirements

Control cards are needed to run the main VTT program. Following is a list of the control card deck.

JOBVTT.	NAME	ADDRESS
ATTACH(BHVTT15, ID=WVDM)		
ATTACH(TBIG, UHANFORDMAIN6, ID=WVDM)		
UPDATE(P=TBIG, C=BIG, D, F, L=123)		
ATTACH(INJ, HINJ, ID=WVDM)		
ATTACH(PKF, HPKT, ID=WVDM)		
ATTACH(PIN, OLDP, ID=WVDM)		
REWIND(BIG)		
REWIND(PKF)		
REWIND(INJ)		
REWIND(PIN)		
REQUEST(POUT, *PF)		
REDUCE.		
BHVTT15.		
CATALOG(POUT, PHI001, ID=WVDM, RP=999)		
#		

The following file names and file elements are used:

INJ	The name of a file created by CYBER-74 UPDATE processor, containing card images for Card Types 23 and 24.
PKF	The name of a file created by CYBER-74 UPDATE processor, containing card images for Card Types 25 and 26.
BIG	The name of a file created by CYBER-74 UPDATE processor, containing card images for Card Types 1 through 22 and 27 through 32

PIN The name of a file from which a potential matrix from a previous run may be read as starting potentials for the current run. Control cards for this file are present only on continuation runs.

POUT The name of a file to which potential matrices generated by the current run may be written.

The control cards permit the operator to:

1. restart a simulation from previously saved data;
2. save the necessary information to do a restart at some later time;
3. save the calculated potential (h) surfaces at certain times during the simulation for later evaluation; and
4. make modifications or complete changes to any of the three input data files.

The control specifications for these options are given in Table 2.

The amount of computer time required for a simulation run depends upon such factors as: the number of active nodes in the system; the size and number of time steps; the severity of stresses introduced (by changing injection flow rates, for example); the frequency with which results are printed out and saved; and the degree of convergence demanded. For one typical verification run on the Hanford Reservation, about 300 seconds of computer central process time were needed to run a 3 month simulation of a system with 2500 active nodes, each month being divided into five time increments and with the injection rates and river boundaries changing on a monthly basis. The convergence limit was set to concur with a maximum change of any potential between successive iterations of less than 0.0001 foot.

Normal practice has been to simulate lengthy time periods by successive short runs, each representing 2 to 4 months of simulated time. In this way the user can inspect intermediate output for errors and problem areas before large amounts of computer time have been used. Continuation is accomplished by saving, on file POUT, the last computed potential surface from the current run and attaching the resulting file as logical file PIN in the next run.

Options permit selection of potential surfaces to be saved on disk files, surfaces to be written to the line printer, and

the manner in which input data are entered. These options are described in detail in Table 2.

Support Program CLASSY

The accuracy of the simulation computed by the VTT Model is difficult to determine by visual examination of the printed output, especially for systems with a large number of nodes. Program CLASSY helps to simplify evaluation by mapping the differences between two arrays of potentials. The program computes the differences between corresponding elements in two arrays (usually, but not necessarily, one array contains measured values and the other contains computed values) and prints out maps of: 1) user specified symbols indicating the range in which the absolute differences lie, and 2) the algebraic signs of the differences. Up to 10 symbols and difference ranges may be specified. In addition to the two maps, the program prints:

- the arithmetic mean of the difference;
- the root mean square difference;
- the number of differences computed (if either matrix contains zero at some point no difference is computed and the maps will show a blank for that point);
- the fraction of total computed differences in each class;
- the values and locations of the five largest and five smallest (absolute) differences; and
- the values and locations of the largest and smallest computed and measured potentials.

Input preparation is described in the self-explanatory symbolic listing provided in Appendix C. Also listed are the two function subprograms and two subroutines required. The program as listed will handle a matrix of 87 rows by 68 columns. Conversion to a different matrix size can be accomplished by recompiling the main program with new values for the dimensions NR, NC, and NEL. The necessary changes in output format will be made automatically by the program so that output map rows will correspond to the finite difference grid rows (within the limits of the printer page width).

ACKNOWLEDGMENTS

This research was sponsored by the Physical and Life Sciences Technology Section, Research and Development Department, Atlantic Richfield Hanford Company. The authors are grateful for the support and assistance provided by D. J. Brown, Staff Geologist, and R. E. Isaacson, Manager of the Physical and Life Sciences Technology Section of Atlantic Richfield Hanford Company.

APPENDIX A
VTT SOURCE PROGRAM LISTING AND
SAMPLE INPUT DATA LIST

VTT SOURCE PROGRAM LISTING

PROGRAM VTT	EXPV1	1
* (PKF=65, INJ=65, RIG, POUT=650, PIN=650, OUTPUT, TAPE2=RIG,	EXPV1	2
* TAPE3=OUTPUT, TAPE10=POUT, TAPE12=PKF, TAPE14=INJ, TAPE15=PIN)	EXPV1	3
	VTTV1	4
	VTTV1	5
	VTTV1	6
	VTTV1	7
	VTTV1	8
	VTTV1	9
	VTTV1	10
	VTTV1	11
	VTTV1	12
	VTTV1	13
	VTTV1	14
	VTTV1	15
	VTTV1	16
	VTTV1	17
	VTTV1	18
	VTTV1	19
	VTTV1	20
	VTTV1	21
	VTTV1	22
	VTTV1	23
	VTTV1	24
	VTTV1	25
	VTTV1	26
	VTTV1	27
	VTTV1	28
	VTTV1	29
	VTTV1	30
	VTTV1	31
	VTTV1	32
	VTTV1	33
	VTTV1	34
	VTTV1	35

C... I/O FILES DESCRIPTION

C ** RIG (FOR INPUT) CONTAINS PROGRAM CONTROL VARIABLES AND THE

C VALUES OF PARAMETERS WHICH DEFINE THE SYSTEM BEING MODELLED.

C

C ** PIN (INPUT) CONTAINS THE POTENTIALS SAVED FROM PREVIOUS RUN.

C

C ** POUT (OUTPUT) RECEIVES THE POTENTIALS COMPUTED IN THIS RUN.

C

C ** INJ (INPUT) CONTAINS THE VALUES OF INJECTIONS TO THE SYSTEM.

C

C ** PKF (INPUT) CONTAINS THE INTERPOLATION TABLE FOR RIVER FLEV.

C

C... DESCRIPTION OF INPUT CARDS READ IN MAIN PROGRAM VTTV1

C ** CARD TYPE A FORMAT(1415) CARD REQUIRED FOR ALL CASES.

C

C FIELD 1 = MINUTE NUMBER OF MINUTES RUN WILL BE ALLOWED

C

C FIELD 2 = MONO THE CALENDAR MONTH NUMBER TO WHICH THE STARTING

C DATA FOR THIS RUN APPLY. (JAN=1, FEB=2, ETC)

C

C FIELD 3 = MP IF THIS RUN IS ONE OF A CONTINUOUS SERIES OF

C RUNS IN WHICH OUTPUT POTENTIALS FROM ONE RUN

C ARE USED AS STARTING POTENTIALS FOR THE NEXT,

C THEN MP IS THE TOTAL NUMBER OF ELAPSED MONTHS

C IN THE SERIES WHEN THIS RUN BEGINS. OTHERWISE

C MP IS ZERO OR BLANK.

C			VTTV1	36
C	FIELD 4	= MAXI THE NUMBER OF ROWS IN THE POTENTIAL MATRIX (THE	VTTV1	37
C		NUMBER OF NODES IN THE Y DIRECTION)	VTTV1	38
C			VTTV1	39
C	FIELD 5	= MAXK THE NUMBER OF COLUMNS IN THE POTENTIAL MATRIX	VTTV1	40
C		(THE NUMBER OF NODES IN THE X DIRECTION)	VTTV1	41
C			VTTV1	42
C	FIELD 6	= KRES A RESTART SIGNAL FOR STEADY STATE RUNS	OCT10A	2
C		IF KRES=1 THIS IS A RESTART CASE AND SAVED DATA	OCT10A	3
C		WILL BE READ FROM TAPE15.	OCT10A	4
C		NOTE THAT RESTART OPTION IS AVAILABLE ONLY FOR	SEP22A	1
C		THE STEADY STATE CASE. RESTART DATA ARE SAVED	SEP22A	2
C		ON FILE ASSIGNED TO UNIT TAPE10 .	SEP22A	3
C			OCT10A	5
C			OCT10A	6
C	** CARD TYPE B	FORMAT (14I5)	SEP22A	4
C		USE AS MANY B-CARDS AS NEEDED TO INPUT MAXI PAIRS	VTTV1	44
C		OF NS,NF	VTTV1	45
C			VTTV1	46
C	FIELDS	NS(I) THE COLUMN NUMBER AT WHICH TO BEGIN CALCULATION	VTTV1	47
C	1,3,....,13	FOR ROW I.	VTTV1	48
C			VTTV1	49
C	FIELDS	NE(I) THE COLUMN NUMBER AT WHICH TO STOP CALCULATION	VTTV1	50
C	2,4,....,14	FOR ROW I.	VTTV1	51
C	COMMON /FRIT/	AYF,DEE,DELMIN,DUM1,DUM2,E,HI,ICOUNT,	COM1	2
	* IHI, IHTOT, II, IIN, JPH, KHI, KHIDI, KIN, KK, LNG, LSW, MH1,		COM1	3
	* MH2, MV1, MV2, NAVG, NDUM2, NGRAP, NPH, NSIZE, NTYPE, N2, N3,		COM1	4
	* N4, N5H, PC3, PC4, PROTHE, PRE, RAT, SUMOMG, SUMIOL, THEM		COM1	5
	* ,HIDONE		COM1	6
C			COM1	7
	COMMON /RLK/	MINUTE,KRES	OCT10A	1
C			COM1	9
	COMMON /VRLK/	ACELL, CODE(10), DELMAX, DELT(16), DELZ, ELENG,	COM1	10
	* FLOW(7), FLOWS, H(5916), IBOT, IFLOWS, IMO, INH, INK, IPUN, IO,		IRRI6	1
	* ISTO, JDELP, JP, KTS, L, LSWP, MAXI, MAXK, MONO, MP, NRUG,		COM1	12
	* NDELT, NDUM1, NE(87), NFR, NINPHI, NO(5916), NP, NQR, NS(87),		IRRI6	2
	* NSTEP, NTOT,		COM1	14
	* N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), P1, QIN(50),		IRRI6	3
	* SATK(5916), STO(10), T(3), THEMEX, TIME, TPDF		IRRI62	1

C	COMMON /RLK2/ A(6R), R(6R), C(6R), JAW, P(6R), P(10), TEK(4),	COM1	17
	* Z(6R)	IRRI6	5
C	COMMON /TOUNIT/ TONEWP, IOOLDP, LUINJ, LUPK, WIF	IRRI6	6
C	COMMON /RLK3/ AA, RR, CC, CON1, CON2, CON3, UD	COM1	20
C	COMMON /PRLK/ ID1, ID2, ID3, ID4, ID5, ID6, ID7, ID8	COM1	21
C	COMMON /PKFY/ PK(126)	COM1	22
C	INPUT/OUTPUT DESIGNATIONS	COM1	23
C	DATA IOOLDP, TONEWP /15,10/	COM1	24
C	FILE(PIN) (POUT)	COM1	25
C	DATA WIF, LUINJ, LUPK /2,14,12/	COM1	26
C	" " "	COMPK	2
C	" " " LUPK CONTAINS INTERPOLATION TABLE FOR RIVER ELEV.	VTTV1	55
C	" " "	VTTV1	56
C	" " " LUINJ CONTAINS INJECTION VALUES (Q) FOR EACH MONTH	VTTV1	57
C	" " "	VTTV1	58
C	WIF CONTAINS ALL OTHER INPUT DATA REQUIRED FOR RUN	VTTV1	59
C	ID1= ID4*ID6 DIMENSIONS PHI,SATK,H,NO,OPHI	VTTV1	60
C	ID2 DIMENSIONS CODE	VTTV1	61
C	ID3 DIMENSIONS QIN	VTTV1	62
C	ID4 DIMENSIONS NS,NE	VTTV1	63
C	ID5 DIMENSIONS P	VTTV1	64
C	ID6 DIMENSIONS A,B,C,P,Z	VTTV1	65
C	ID7 DIMENSIONS TEK	VTTV1	66
C	ID8 DIMENSIONS STO	VTTV1	67
C		VTTV1	68
C		VTTV1	69
C		VTTV1	70
C		VTTV1	71
C		VTTV1	72
C		VTTV1	73
C		VTTV1	74
C		VTTV1	75
C		VTTV1	76
C		VTTV1	77
C		VTTV1	78
C		VTTV1	79

C	COMMON /RLK2/ A(68), H(68), C(68), JAM, P(68), P(10), TEK(4),	COM1	17
	* 7(68)	IRRI G	5
		IRRI G	6
C	COMMON /TOUNIT/ TONEWP, TOLDP, LUTNU, LUPK, MIF	COM1	20
C	COMMON /RLK3/ AA, RR, CC, CON1, CON2, CON3, UD	COM1	21
		COM1	22
C	COMMON /PRK/ ID1, ID2, ID3, ID4, ID5, ID6, ID7, ID8	COM1	23
		COM1	24
C	COMMON /PRKV/ PK(126)	COM1	25
	EQUITVALENCE (7,PI,OP), (P,KTY), (7,KEY)	COM1	26
	DATA NOPKS(126)	COMPK	2
	NAMELTST/MODIFY/NRUG,N5,JP,DELP,NP,NDUM1,DEL1,NDEL,T,DELMAX,	EXPV1	6
	1 TPDIF,THEMAX.PI,TPUN,TIME	DATAPK	2
	2 NINPHI	SRM1	12
	100 FORMAT (13,611,10A6)	SRM1	13
	110 FORMAT (14I5)	MARI7A	1
	120 FORMAT("UNABLE TO FIND TIMEPLANE",I5," AFTER 50 TRIES. LAST TIM	SRM1	14
	*EPLANE FOUND WAS",I5)	SRM1	15
	130 FORMAT (7(I2,F8.0))	SEPI2F	1
	140 FORMAT (5(I5,F10.0))	SEPI2F	2
	150 FORMAT (14(I3.02))	SRM1	16
	160 FORMAT("ADESTART DATA FOR THIS STEADY STATE CASE READ FROM UNIT 15	SRM1	17
	*"/" STARTING ITERATION NUMBER IS ",I5)	SRM1	18
	180 FORMAT(10(I3,I4))	OCT10A	9
	190 FORMAT(7F10.4)	OCT17A	1
	200 FORMAT (3I3)	SRM1	19
	210 FORMAT(7F10.0)	SRM1	20
	230 FORMAT(2F5.0,I5,14F4.0)	SRM1	21
	260 FORMAT(12A6)	SRM1	22
	270 FORMAT(" RIVER FLOWRATE=",IPE11.4)	SRM1	23
	280 FORMAT(" INJECTIONS"/(IX,J)P10F12.4))	SRM1	24
	JFLD(I,J,IM)=SHIFT(IW,J+I-60).A..N.MASK(60-J)	SRM1	25
	SHMA(I,J)=SHIFT(MASK(60-J),60-J-I)	MARI3E	1
	JFLDR(I,J,IA,N,IR)=IA.A.SHMA(I,J).OR.SHIFT(IM,N-I).A..N.SHMA(I,J)	FFRFV	5
		FFRFV	6
		FFREV	7

C	CASE IDENTIFICATION AND SWITCH SETTER	SRM1	27
	READ (2,100)NINPHT,INH,INK,NRUG,NSTEP,IPUN,N1,CODE	SRM1	28
C	CASE INPUT	SRM1	29
	READ(2,110) N5,JP,JDELP,NP,NDUM1,IQ,ISTO	SRM1	30
	JPH=JP	SRM1	31
	NPH=NP	SRM1	32
	READ (2,190)DELZ	SRM1	33
	READ(2,230)DELMAX,TIME,NDELT,(DELT(I),I=1,NDELT)	SRM1	34
	READ (2,190)TPDIF,THEMAX,P1	SRM1	35
	N3=IQ	SRM1	36
C----	SKIP HEADING CARD	SRM1	37
290	READ(2,MODIFY)	SRM1	38
	ICOUNT=DELMAX+.0001	SRM1	39
	N5H=N5	SRM1	40
	READ (LUINJ,260)	SRM1	41
C----	SKIP INJECTION VALUES USED IN EARLIER RUNS OF SERIES.	SRM1	42
	NQR=MP	SRM1	43
C TR5	INJECTION SET NT IS ONE TO START THIS RUN WITH.	SRM1	44
	NT=MP+1	SRM1	45
	DO 300 JT=1,NT	SRM1	46
	RFAD(LUINJ,210)(QIN(I),I=1,IQ)	SRM1	47
	NQR=NQR+1	SRM1	48
300	CONTINUE	SRM1	49
	WRITE(3,280)(QIN(I),I=1,IQ)	SRM1	50
	READ (2,210) (STO(I),I=1,ISTO)	SRM1	51
310	E=TPDIF	SRM1	52
C----	SKIP HEADING CARD	SRM1	53
	READ(LUPK,260)	SRM1	54
	READ (LUPK,210) PK	SRM1	55
C	SCALE SOIL PARAMETERS IN TERMS OF PROBLEM SCALING LENGTH	SRM1	56
	NTOT=MAXI*MAXK	SRM1	57
	ACFLL=DELZ**2	SRM1	58
330	NINPHY=NINPHT	SRM1	59

SRM1	60
SRM1	61
SRM1	62
SRM1	63
SRM1	64
SFPI2F	3
BINIO	1
BINIO	2
SRM1	68
SFPI2F	5
EXPVI	7
SRM1	70
SRM1	71
SRM1	72
SRM1	73
SRM1	74
SRM1	75
SRM1	76
SRM1	77
SRM1	78
SRM1	79
SRM1	80
SRM1	81
SRM1	82
SRM1	83
SRM1	84
SRM1	85
SRM1	86
SRM1	87
SRM1	88
SRM1	89
SRM1	90
SRM1	91
SRM1	92
SRM1	93

```

360 IF(JP.LT.0) JP=100000
    IF((NP.LT.0).OR.(MTNUTF.LT.0)) NP=100000
    IF(NINPHI.LE.0) GO TO 400
    NPHI=1
    NTOTPI=NTOT+1
    I=0
    370 READ(TOOLDD) J,(PHI(JN),JN=1,NTOT)
    IF(NINPHI.EQ.0) GO TO 390
    I=I+1
    IF(I.LT.50) GO TO 370
    WRITE(3,120) NINPHI,J
    375 LSW=R
    CALL MAIN2
    390 III=2
    NINPHY=J
    GO TO 410
    400 CONTINUE
    III=1
    410 GO TO (420,450,460,500,510),III
    AT III=1 READ POTENTIALS
    AT III=2 READ AQUIFER BOTTOM ELEVATIONS
    AT III=3 READ PERMEABILITIES
    AT III=4 READ STORAGE COEFF LOCATORS AND PUT INTO NO ARRAY
    AT III=5 READ CALCULATION TYPES AND PUT INTO NO ARRAY
    INITIAL PHIS FROM CARDS-TWO OPTIONS
    420 IF(NINPHI.LT.0) GO TO 440
    READ (2,200) IR,IT,KKK
    430 READ(2,130) (KEY(IJJ),PLOC(IJJ),JJJ=1,KKK)
    GO TO 520
    440 READ(2,190) (PHI(J),J=1,NTOT)
    NINPHY=0
    GO TO 620
    450 IF(INH.EQ.1) GO TO 490
    READ (2,190) (H(J), J=1,NTOT)
    GO TO 620

```

460	IF (INK.EQ.1) GO TO 470	SRM1	94
	READ(2,190) (SATK(J),J=1,N,TOT)	SRM1	95
	GO TO 620	SRM1	96
470	READ(2,200) IR,IT,KKK	SRM1	97
480	READ(2,140) (KEY(JJJ),PLOP(JJJ),JJJ=1,KKK)	SRM1	98
	GO TO 520	SRM1	99
490	CONTINUE	SRM1	100
	READ(2,200) IR,IT,KKK	SRM1	101
	READ(2,130) (KEY(JJJ),PLOP(JJJ),JJJ=1,KKK)	SRM1	102
	GO TO 520	SRM1	103
500	READ(2,200) IR,IT,KKK	SRM1	104
	READ(2,180) (KEY(JJJ),KTY(JJJ),JJJ=1,KKK)	SRM1	105
	GO TO 520	SRM1	106
510	READ(2,200) IR,IT,KKK	SRM1	107
	READ(2,150) (KEY(JJJ),KTY(JJJ),JJJ=1,KKK)	SRM1	108
520	KEY(KKK+1)=0	SRM1	109
	DO 600 I=IR,IT	SRM1	110
	JJJ=1	SRM1	111
	DO 600 K=1,MAXK	SRM1	112
	IF (K=KEY(JJJ+1)) 540,530,540	SRM1	113
530	JJJ=JJJ+1	SRM1	114
540	JAM=MAXK*(I-1)+K	SRM1	115
	NOJAM=NO(JAM)	FFREV	8
	KTYJJJ=KTY(JJJ)	FFREV	9
	GO TO (550,540,580,570,590),III	SRM1	116
550	PHI(JAM)=PLOP(JJJ)	SRM1	117
	GO TO 600	SRM1	118
560	H(JAM)=PLOP(JJJ)	SRM1	119
	GO TO 600	SRM1	120
570	NO(JAM)=JFLOR(0,12,NOJAM,4H,KTYJJJ)	FFREV	10
	GO TO 600	SRM1	122
580	SATK(JAM)=PLOP(JJJ)	SRM1	123
	GO TO 600	SRM1	124
590	NO(JAM)=JFLOR(30,6,NOJAM,54,KTYJJJ)	FFREV	11
600	CONTINUE	SRM1	126
610	IF (MAXI-IT) 620,620,410	SRM1	127

SRM1	161
SRM1	162
SRM1	163
SRM1	164
SRM1	165
SRM1	166
OCT10A	11
SRM1	168
OCT10A	12
EXPV1	8
EXPV1	9
EXPV1	10
EXPV1	11
EXPV1	12
EXPV1	13
OCT10A	15
EXPV1	14
OCT10A	17
OCT10A	18
SRM1	169
AOTCHK	1
AOTCHK	2
SRM1	170
SRM1	171
SRM1	172
SRM1	173
SRM1	174
EXPV1	15
SRM1	176
SRM1	177
SRM1	178
SRM1	179
SRM1	180
SRM1	181
SRM1	182

```

IF(IFLOWS.FO.O) IFLOWS=7
FLOWS=FLOW(IFLOWS)
WRITE(3,270) FLOWS
INITIALIZE
C 700 LSW=1
KTS=1
IF(KRES.NF.1) L=1
J=INPHY+1
IF(KRES.NE.1) GOTO 708
READ(15) L,(H(I),I=1,NTOT)
IF(FOF(15)) 706,701,706
701 READ(15) (OPHI(I),I=1,NTOT)
IF(EOF(15)) 706,702,706
702 READ(15) (PHI(I),I=1,NTOT)
IF(EOF(15)) 706,707,706
706 STOP"*** *** EOF READING TAPE 15"
707 WRITE(3,160) L
708 CONTINUE
IF(KRES.FO.1) GOTO 720
DO 710 JAM=1,NTOT
IF(PHI(JAM)-H(JAM)-1.) 712,710,710
712 H(JAM)=PHI(JAM)-1.
710 OPHI(JAM)=PHI(JAM)
C PRINT INITIAL CONDITIONS
C CONTINUE
IF(IPUN.NF.O) CALL OUT
C--- PERFORM LIGHT EDIT OF INPUT DATA
C---CALL INEDIT REMOVED TO ALLOW SMALL CORE LOADING
730 CONTINUE
LSW=2
CALL OMEG(OM)
TIME=TIME+ARS(DELT(1))
740 CALL MAIN?
RETURN
END

```

```

C
SUBROUTINE MAIN2
  MAIN2 ITERATES ON A GIVEN TIME PLANE
  COMMON /FRIT/ AYE,DEF,DELMIN,DUM1,DUM2,E,HI,ICOUNT,
  * IHI,ITHI,II,IIN,JPH,KHI,KHIDI,KIN,KK,LNG,LSW,MHI,
  * MH2,MV1,MV2,NAV,NDUM2,NGRAP,NPH,NSIZE,NTYPE,N2,N3,
  * N4,NEH,PC3,PC4,PRCTHF,PRE,RAT,SUMOMG,SUMIOL,THEM
  * ,HIDONE
C
  COMMON /RLK/ MINUTE,KRES
C
  COMMON /VHIK/ ACFL, CODE(10), DELMAX, DFLT(16), DEL7, ELENG,
  * FLOW(7), FLOWS, H(5916), IROT, IFLOWS, IMO, INH, INK, IPUN, IQ,
  * ISTO, JNFLP, JP, KTS, L, LSWP, MAXI, MAXK, MONO, MP, NRIIG,
  * NDELT, NDUM1, NF(R7), NFB, NINPHI, NO(5916), NP, NQR, NS(R7),
  * NSTEP, NTOT,
  * N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), PI, QIN(50),
  * SATK(5916), STO(10), T(3), THEMEX, TIME, TPDIF
C
  COMMON /RLK2/ A(68), P(68), C(68), JAM, P(68), R(10), TEK(4),
  * 7(68)
C
  COMMON /TOUNIT/ TONEWP, IOOLDP, LUINJ, LUPK, MIF
C
  COMMON /PLK2/ AA, BB, CC, CON1, CON2, CON3, UD
C
  COMMON /PRLK/ ID1,ID2,ID3,ID4,ID5,ID6,ID7,ID8
C
  COMMON /PRFY/ PK(126)
  DATA NOPKS/126/
  100 FORMAT('UNDESTART DATA FOR THIS STEADY STATE CASE SAVED ON UNIT 10'
  *)
  * JFLD(I,J,TW)=SHIFT(IW,J,I-60).A..N.WASK(60-J)
  IARMIN=IAPRS(MINUTE)
  HJ=0.0

```

```

SRM2
SRM2
COM1
COM1
COM1
COM1
COM1
COM1
COM1
OCT10A
COM1
COM1
IRRIG
COM1
IRRIG
COM1
IRRIG
IRRIG2
COM1
IRRIG
IRRIG
COM1
COM1
COM1
COM1
COM1
COM1
COM1
COMPK
DATAPK
OCT10A
OCT10A
FFREV
SRM2
SRM2

```

```

2
3
2
3
4
5
6
7
1
9
10
1
12
2
14
3
1
17
5
6
20
21
22
23
24
25
26
2
2
19
20
16
8
9

```

```

10  SRM2
11  SRM2
12  SPM2
13  SRM2
14  SRM2
15  SRM2
16  SRM2
17  SRM2
18  SRM2
19  FFREV
20  SKIPR
21  FFREV
22  SRM2
23  SRM2
24  SRM2
25  SRM2
26  SPM2
27  SPM2
28  SRM2
29  SRM2
30  SRM2
31  FFREV
32  SRM2
33  SRM2
34  SRM2
35  SRM2
36  SRM2
37  SRM2
38  SPM2
39  SRM2
40  SRM2
41  SRM2
42  SRM2
43  SRM2

```


270	CONTINUE	SRM2	77
C	DEBUG PRINTOUT	SRM2	78
280	CALL OUT	SRM2	79
	CALL SECOND(SEC)	SRM2	80
	IF(SEC/60..GT.IARMIN) GOTO 330	SRM2	81
C	NEED TO ITERATE AGAIN	SRM2	82
290	IF(ABS(HI)-PJ) 300,300,310	SRM2	83
C	NO MORE ITERATION ON THIS TIME PLANE	SRM2	84
300	CALL MAIN3	SRM2	85
	IF(LSW.EQ.5) GOTO 340	SRM2	86
	GO TO 120	SRM2	87
C	ITERATE AGAIN	SRM2	88
310	HI=0.	SRM2	89
	L=L+1	SRM2	90
	LSW=2	SRM2	91
	IF(L.LT.N5) GO TO 120	SRM2	92
C	GO TO SAVE ROUTINE-ITERATION LIMIT EXCEEDED	SRM2	93
320	LSW=6	SRM2	94
	GO TO 340	SRM2	95
C	GO TO SAVE ROUTINE-TIME RUN OUT	SRM2	96
330	LSW=7	SRM2	97
340	IF(MINUTE.LT.0) GO TO 370	SRM2	98
	IF(N1.EQ.0) GOTO 370	OCT10A	21
	WRITE(10) L,(H(I),I=1,NTOT)	EXPV1	18
	WRITE(10) (OPHI(I),I=1,NTOT)	EXPV1	19
	WRITE(10) (PHI(I),I=1,NTOT)	EXPV1	20
	ENDFILE 10	EXPV1	21
350	WRITE(3,100)	EXPV1	22
	CALL OUT	OCT10A	26
360	RETURN	SRM2	100
370	CALL OUT	SRM2	101
	GO TO 360	SRM2	102
	END	SRM2	103

C	TP5	CONTINUE THIS MONTH	SRM3	61
		IF (KTS.LT.NDELT) GOTO 380	SRM3	62
C	TP5	START A NEW MONTH	SRM3	63
		KTS=1	SRM3	64
C	TP5	INCREMENT CALENDAR MONTH INDEX	SRM3	65
		MONO=MONO+1	SRM3	66
		IF (MONO.GT.12) MONO=1	SRM3	67
		GOTO 380	SRM3	68
C	TP5	COMPUTE DEIT FOR INITIAL MONTH OF SERIES	SRM3	69
	350	IF (MP.EQ.0) GOTO 370	SRM3	70
C		-----DEIT CALCULATION FOR LAST STEP OF ANY MONTH EXCEPT INITIAL MONTH	SRM3	71
C			SRM3	72
		NDM1=NDELT-1	SRM3	73
		DTM=0.0	SRM3	74
		DO 360 NT=1,NDM1	SRM3	75
	360	DTM=DTM+DEIT(NT)	SRM3	76
		DEIT(KTS)=DDM(MONO)-DTM	SRM3	77
		GOTO 380	SRM3	78
	370	DEIT(KTS)=DDM(IMO)-TIME	SRM3	79
	380	TIME=TIME+DELT(KTS)	SRM3	80
		TY=1	SRM3	81
		NRV=1	SRM3	82
		DO 390 JAM=1,NTOT	SRM3	83
		OPHI(JAM)=DHI(JAM)	SRM3	84
	390	CONTINUE	SRM3	85
		J=J+1	SRM3	86
		WRITE(3,210) J,KTS,MONO	EXPV1	87
		LSW=2	SRM3	26
		HT=0.	SRM3	89
		HIDONE=0.	SRM3	90
		CALL OMEG(OM)	SRM3	91
		L=1	SRM3	92
		IF (KTS.NF.1) GOTO 450	SRM3	93
		WRITE(3,200) MONO	SRM3	94
			EXPV1	27

400	READ (LUNJ,410) (QIN(I),I=1,N3)	SRM3	96
	MP=MP+1	SRM3	97
	NQR=NQR+1	SRM3	98
	WRITE(3,180) (QIN(I),I=1,N3)	SRM3	99
410	FORMAT(7E10.0)	SRM3	100
420	IFLOWS=IFLOWS+1	SRM3	101
	IF(IFLOWS.LF.7) GO TO 430	SRM3	102
	IFLOWS=1	SRM3	103
	READ (2,410) (FLOW(I),I=1,7)	SRM3	104
	NFR=NFR+1	SRM3	105
	WRITE(3,190) (FLOW(I),I=1,7)	EXPV1	28
430	FLOW=FLOW(IFLOWS)	SRM3	107
	WRITE(3,220) FLOWS	EXPV1	29
	IF(FLOWS) 440,450,450	SRM3	109
440	IF(FLOWS+1.F38) 460,460,450	SRM3	110
450	CONTINUE	SRM3	111
	RETURN	SRM3	112
C	NORMAL END OF CASE	SRM3	113
460	LSW=5	SRM3	114
C----	WRITE PHI RINARY TAPF	SRM3	115
	IF(NP.GE.100000) GO TO 480	SRM3	116
	NTOTPI=NTOT+1	SRM3	117
	WRITE(IONFWP) J,(PHI(JN),JN=1,NTOT)	BINIO	4
479	WRITE(3,130) NTOT,IONFWP,J	EXPV1	32
C	FINAL OUTPUT	SRM3	127
480	CALL OUT	SRM3	128
490	CALL EXIT	SRM3	129
C	SET UP TO SAVE - TIME PLANE LIMIT EXCEEDED	SRM3	130
500	WRITE(3,100)	SRM3	131
	IF(NP.GE.100000) GO TO 540	SRM3	132
520	NTOTPI=NTOT+1	SRM3	133
	WRITE(IONFWP) J,(PHI(JN),JN=1,NTOT)	BINIO	5
536	WRITE(3,130) NTOT,IONFWP,J	EXPV1	35
540	CALL EXIT	SRM3	142
	END	SRM3	143

```

C----- SUBROUTINE OUT
C LSW DETERMINES THE TYPE OF PRINTOUT REQUESTED
C LSW = 1 NORMAL...PRINT INITIAL CONDITIONS
C 2 NORMAL...PRINT HI DIFFERENCE AND ITERATION NUMBER
C 3 NORMAL...END OF TIME PLANE PRINT RESULTS
C 4 ERROR...COMPUTED PHI TOO HIGH
C 5 NORMAL...END OF JOB
C 6 NORMAL...ITERATIONS EXCEEDED
C 7 NORMAL...TIME LIMIT EXCEEDED
C 8 ERROR...TIME PLANE REQUESTED NOT ON TAPE.
C 9 ERROR...TOT(68),MUM(68),DUM(68)
C DIMENSION TOT(68),MUM(68),DUM(68)
C COMMON /FRIT/ AYE,DEF,DELMIN,DUM1,DUM2,E,HI,ICOUNT,
C * THI, IHINT, II, IIN, JPH, KHI, KHIDI, KIN, KK, LNG, LSW, MHI,
C * MH2, MVI, MV2, NAVG, NDUM2, NGRAP, NPH, NSIZE, NTYPE, NZ, N3,
C * N4, N5H, PC3, PC4, PRCIME, PRF, RAT, SUMOMG, SUMIOL, THEM
C * ,HIDONE
C
C COMMON /PLK/ MINUTE,KRES
C
C COMMON /VRIK/ ACFL, CODE(10), DELMAX, DELT(16), DELZ, ELENG,
C * FLOW(7), FLOWS, H(5916), IROT, IFLOWS, IMO, INH, INK, IPIIN, IO,
C * TSTO, JDFID, JP, KTS, L, LSWP, MAXI, MAXK, MCNO, MP, NRIIG,
C * NOELT, NOUM1, NF(R7), NFB, NIMPHI, NO(5916), NP, NGR, NS(R7),
C * NSTEP, NTOT,
C * N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), PI, QIN(50),
C * SATK(5916),STO(10),T(3),THEMAX,TIME,TPOIF
C
C COMMON /RIK2/ A(68), B(68), C(68), JAM, P(68), P(10), TFK(4),
C * Z(68)
C
C COMMON /TOUNIT/ TONEWP, JOOLDP, LUINJ, LUPK, NIF
C
C COMMON /RIK3/ AA, RB, CC, CON1, CON2, CON3, UD
C
C COMMON /PRIK/ ID1, ID2, ID3, ID4, ID5, ID6, ID7, ID8

```

```

SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
SROUT
EXPV1
COM1
COM1
COM1
COM1
COM1
COM1
OCT10A
COM1
COM1
IRRI6
COM1
IRRI6
COM1
IRRI6
IRRI62
COM1
IRRI6
IRRI6
COM1
COM1
COM1
COM1
COM1
COM1

```

```

2
3
4
5
6
7
8
9
10
11
36
2
3
4
5
6
7
1
9
10
1
12
2
14
3
1
17
5
6
20
21
22
23
24
25

```

26	COMMON /PKEY/ PK(126)	COM1
2	EQUIVALENCE (A,IOT),(H,MUM),(C,DUM)	COMPK
37	DATA NOPKS/126/	EXPV1
2		DATAPK
19	FORMAT (1H0,10A6)	SROUT
20	FORMAT (12H0TIME EQUALS1PE11.4)	SROUT
21	FORMAT (1H+.1PE17.5,9E12.5/(1PE18.5,9E12.5))	SROUT
22	FORMAT (17H0PIEZOMETRIC HEAD)	SROUT
23	FORMAT (9H0HT DIFF=1PE11.4,4H I=I3,4H K=I3,10H DELTA T=FE11.4)	SROUT
24	FORMAT (14H0 AQUIFER BASE)	SROUT
25	FORMAT (1H+.6X20(I3,02)/(7X,20(I3,02)))	SROUT
26	FORMAT (11H0ITERATION=I4,10H HI DIFF=1PE11.4,4H T=I3,4H K=I3)	SROUT
27	FORMAT (19H1INITIAL CONDITIONS)	SROUT
28	FORMAT (22H1ITERATION IS COMPLETE)	SROUT
29	FORMAT (35H1ERROR EXIT-A POTENTIAL ABOVE LIMIT , 2I5)	SROUT
30	FORMAT (19H1CASE IS COMPLETE)	SROUT
31	FORMAT ("NUMBER OF ITERATIONS IN ONE TIME PLANE HAS EXCEEDED LIMIT	SROUT
32	"")	SROUT
33	FORMAT ("TIME LIMIT EXCEEDED")	SROUT
34	FORMAT (26H1ERROR EXIT-NO SUCH RECORD)	SROUT
35	FORMAT (48H0THE NUMBER OF ITERATIONS IN THIS TIME PLANE WAS(5)	SROUT
36	FORMAT ("0 INDEX AND CALCULATION TYPE MATRIX")	SROUT
37	FORMAT (13H1SOURCE FLOWS)	SROUT
38	FORMAT (7X4H0(1),7X4H0(2),7X4H0(3),7X4H0(4),7X4H0(5),7X4H0(6))	SROUT
39	FORMAT (1H0,1PE12.5/(1X,1PE12.5))	SROUT
40	FORMAT ("STORAGE COEFFICIENTS")	SROUT
41	FORMAT (7X4H0(1),7X4H0(2),7X4H0(3),7X4H0(4),7X4H0(5),7X4H0(6))	SROUT
42	FORMAT ("STORAGE COEFFICIENT INDEX AND PK INDEX")	SROUT
43	FORMAT (1H+.4X10(I6,I4)/(5X10(I6,I4)))	SROUT
44	FORMAT ("NOPK TABLE"/)	SROUT
45	FORMAT (1X,1PE12.5)	SROUT
46	FORMAT (1H1)	SROUT
47	FORMAT (27H0THIS TIME PLANE IS NUMBER I4)	SROUT
21	JELD(I,J,IW)=SHIFT(IW,J+1-60).A..M.MASK(60-J)	FFREV
48	GO TO (390,400,420,430,440,450,460,480).LSW	SROUT

49 SR0UT
50 SR0UT
51 SR0UT
52 SR0UT
53 SR0UT
54 SR0UT
55 SR0UT
56 SR0UT
57 SR0UT
58 SR0UT
59 SR0UT
60 SR0UT
61 SR0UT
62 SR0UT
63 SR0UT
64 SR0UT
65 SR0UT
66 SR0UT
67 SR0UT
68 SR0UT
69 SR0UT
70 SR0UT
71 SR0UT
72 SR0UT
73 SR0UT
74 SR0UT
75 SR0UT
76 SR0UT
77 SR0UT
78 SR0UT
79 SR0UT
80 SR0UT
81 SR0UT
82 SR0UT
83 SR0UT

```

390 WRITE (3,190)
    ICOUNT=DELMAX+.0001
    GO TO 490
400 GO TO (720,710,410), NRUG
410 WRITE (3,200)
    GO TO 490
420 WRITE (3,380),J
    WRITE (3,260),L
    WRITE (3,150),HIDONE,IHIDI,KHIDI,DELT(KTS)
    GO TO 490
430 WRITE (3,210) IH1,KH1
    GO TO 490
440 WRITE (3,220)
    GO TO 490
450 WRITE (3,230)
    GO TO 470
460 WRITE (3,240)
470 GO TO (490,490,720), NRUG
480 WRITE (3,250)
490 WRITE (3,100)CODE
    WRITE (3,110)TIME
    GO TO (530,540,500,540,540,540,540,720),LSW
500 GO TO (510,510,540),NRUG
510 IF (NDUM1=J) 520,520,690
520 NDUM1=NDUM1+ICOUNT
    GO TO 540
530 CONTINUE
540 WRITE (3,130)
    DO 570 I=1,MAXI
    IREV=MAXI+1-I
550 FORMAT(3H0T=,I3)
    WRITE (3,550) IREV
    DO 560 K=1,MAXK
    JAM=MAXK*(IREV-1)+K
560 DUM(K)= PHT(JAM)

```

84 SR0UT
85 SR0UT
86 SR0UT
87 SR0UT
88 SR0UT
89 SR0UT
90 SR0UT
91 SR0UT
92 SR0UT
93 SR0UT
94 SR0UT
95 SR0UT
96 SR0UT
97 SR0UT
98 SR0UT
99 SR0UT
100 SR0UT
101 SR0UT
102 FFREV
103 FFREV
104 FFREV
105 SR0UT
106 SR0UT
107 SR0UT
108 SR0UT
109 SR0UT
110 SR0UT
111 FFREV
112 FFREV
113 FFREV
114 SR0UT
115 SR0UT
116 SR0UT
117 SR0UT

```

570 WRITE (3,120) (DUM(K),K=1,MAXK)
    WRITE(3,370)
    GO TO (580,720,700,660,580,720,720,720), LSW
580 WRITE (3,160)
    DO 600 I=1,MAXI
    IREV=MAXI+1-I
    WRITE(3,550) IREV
    DO 590 K=1,MAXK
    JAM=MAXK*(IREV-1)+K
    DUM(K)=H(JAM)
590 WRITE(3,120) (DUM(K),K=1,MAXK)
600 IF(LSW.EQ.5) GO TO 660
610 WRITE (3,270)
    DO 630 I=1,MAXI
    IREV=MAXI+1-I
    WRITE(3,550) IREV
    DO 620 K=1,MAXK
    JAM=MAXK*(IREV-1)+K
    NOJAM=NO(JAM)
    IQT(K)=JFLN(12.9,NOJAM)
620 MUM(K)=JFLN(30.6,NOJAM)
630 WRITE (3,170) (IQT(K),MUM(K),K=1,MAXK)
    WRITE (3,330)
    DO 650 I=1,MAXI
    IREV=MAXI+1-I
    WRITE (3,550) IREV
    DO 640 K=1,MAXK
    JAM=MAXK*(IREV-1)+K
    NOJAM=NO(JAM)
    IQT(K)=JFLN(0.12,NOJAM)
640 MUM(K)=JFLN(21.9,NOJAM)
650 WRITE (3,340) (IQT(K),MUM(K),K=1,MAXK)
    WRITE (3,280)
    WRITE (3,290)
    WRITE (3,300) (QIN(K),K=1,N3)
    WRITE (3,310)

```

118 SPROUT
 119 SPROUT
 120 SPROUT
 121 SPROUT
 122 SPROUT
 123 SPROUT
 124 SPROUT
 125 SPROUT
 126 SPROUT
 127 SPROUT
 128 SPROUT
 129 SPROUT
 130 SPROUT
 2 SPTRID
 4 SPTRID
 2 COM1
 3 COM1
 4 COM1
 5 COM1
 6 COM1
 7 COM1
 1 OCT10A
 9 COM1
 10 COM1
 11 IPPIG
 12 COM1
 2 IPPIG
 14 COM1
 3 IPPIG
 1 IPPIG2
 17 COM1
 5 IPPIG
 6 IPPIG
 20 COM1
 21 COM1

```

WRITE (3,320)
WRITE (3,330) (STO(K),K=1,ISTO)
WRITE (3,340)
WRITE (3,340) PK
660 CONTINUE
670 IF (LSW-1) 720,720,680
680 WRITE (3,370)
690 CONTINUE
700 CONTINUE
GO TO 720
710 WRITE (3,190)L,HT,IHI,KHI
720 RETURN
END
      SUBROUTINE TRIDIG (ISO,ISI)
      DIMENSION W(70),G(70)
      COMMON /FRIT/ AYE,DEF,DELMIN,DUM1,DUM2,E,HI,ICOUNT,
      * IHI, IHTOT, II, IIN, JPH, KHI, KHTDI, KTN, KK, LNG, LSW, MHI,
      * MP2, MVI, MV2, NAVG, NDUM2, NGRAP, NPH, NSIZE, NTYPE, N2, N3,
      * N4, N5H, PC3, PC4, PCCTHF, PRF, PAT, SUMOMG, SUMIOL, THEM
      * ,HYOCNE
      COMMON /BLK/ MINUTE,KRFS
      COMMON /VBLK/ ACFL, CODE(10), DELMAX, DELT(16), DEL7, ELENG,
      * FLOW(7), FLOWS, H(5916), IROT, IFLOWS, IMO, INH, INK, IPIIN, IQ,
      * ISTO, JREFD, JP, KTS, L, LSWP, MAXI, MAXK, MCNO, MP, NRUG,
      * NDELT, NDIIM1, NF(87), NFO, NINPHI, NO(5916), NP, NQR, NS(87),
      * NSTEP, NTOT,
      * N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), PI, QIN(50),
      * SATK(5916),STO(10),T(3),THEMAX,TIME,TBDIF
      COMMON /BLK2/ A(48), R(48), C(48), JAM, P(68), R(10), TEK(4),
      * 7(48)
      COMMON /ICOUNT/ IONEWP, IONOLDP, LUINJ, LUPK, NIF
  
```

COM1 22
COM1 23
COM1 24
COM1 25
COM1 26
COMPK 2
SRTAID 7
SRTAID 8
SRTAID 9
SRTAID 10
SRTAID 11
SRTAID 12
SRTAID 13
SRTAID 14
SRTAID 15
SRTAID 16
SRTAID 17
SRTAID 18
SRTAID 19
SRTAID 20
SRTAID 21
SRTAID 22
SRTAID 23
SRTAID 2
SRTAID 2
SRTAID 3
SRTAID 4
SRTAID 5
SRTAID 6
SRTAID 7
SRTAID 1
SRTAID 9
SRTAID 10
SRTAID 11
SRTAID 12
SRTAID 2
SRTAID 14
SRTAID 3
SRTAID 1

```

C      COMMON /BLK3/ AA, AB, AC, CON1, CON2, CON3, UD
C
C      COMMON /PRLK/ ID1, ID2, ID3, ID4, ID5, ID6, ID7, ID8
C
C      COMMON /PKFY/ PK(126)
      G(ISO)=P(ISO)/R(ISO)
      IF(JS1.LE.ISO) GO TO 130
      IS=ISO+1
      W(ISO)=C(ISO)/R(ISO)
      DO 100 I=IS,IS1
      QUF=B(I)-A(I)*W(I-1)
      W(I)=C(I)/QUF
      G(I)=(P(I)-A(I)*G(I-1))/QUF
      Z(IS1)=G(IS1)
      MAX=IS1-ISO
      DO 110 JJ=1,MAX
      I=IS1-JJ
      Z(I)=G(I)-W(I)*Z(I+1)
      RETURN
      Z(ISO)=G(ISO)
      GO TO 120
      END
      SUBROUTINE TWO(PART1,PART2,PART3,PART4,GETEM,F)
      COMMON /FPTT/ AYE,DEF,DELMIN,DUM1,DUM2,F,HI,JCOUNT,
      * IHI, IHIPT, II, IIN, JPH, KHI, KHIDI, KTN, KK, LNG, LSW, MHI,
      * MH2, MVI, MV2, NAVG, NDUM2, NGRAP, NPH, NSIZE, NTYPE, N2, N3,
      * N4, N5H, PC3, PC4, PCTHF, PRE, RAT, SUMOMG, SUMIOL, THEM
      * ,HIDONE
      COMMON /BLK/ MINUTE,KPES
C
C      COMMON /VRLK/ ACFL, CODE(10), DELMAX, DELT(16), DELZ, ELENG,
      * FLOW(7), FLOWS, H(5916), IROT, IFLOWS, IMO, INH, INK, IPUN, IQ,
      * ISTO, JDFLP, JP, KTS, L, LSWP, MAXI, MAXK, MCNO, MP, NBUG,
      * NDELT, NDIIM1, NE(87), NFR, NINPHI, NO(5916), NP, NQR, NS(87),
      * NSTFP, NTOT,
      * N1, N5, OM, OMF(3), OPHI(5916), J, PHI(5916), PL, QIN(50),
      * SATK(5916), STO(10), T(3), THEMEX, TIME, TPDIF

```

COM1	17
IRRIG	15
IRRIG	6
COM1	20
COM1	21
COM1	22
COM1	23
COM1	24
COM1	25
COM1	26
COMPK	2
SRTWO	6
FFREV	28
SRTWO	7
SRTWO	8
SRTWO	9
SRTWO	10
SRTWO	11
SRTWO	12
SRTWO	13
SRTWO	14
SRTWO	15
SRTWO	16
SRTWO	17
SRTWO	18
SRTWO	19
SRTWO	20
SRTWO	21
SRTWO	22
SRTWO	23
SRTWO	24
SRTWO	25
SRTWO	26
SRTWO	27
SRTWO	28
SRTWO	29

```

C      COMMON /RLK2/ A(68), R(68), C(68), JAM, P(68), R(10), TEK(4),
*      7(68)

C      COMMON /TOUNIT/ IONEWP, IOLDP, LUINJ, LUPK, MIF

C      COMMON /RLK3/ AA, RB, CC, CON1, CON2, CON3, UD

C      COMMON /RLK4/ ID1, ID2, ID3, ID4, ID5, ID6, ID7, ID8

C      COMMON /PKEY/ PK(126)
EQUIVALENCE (IJ1,MH2), (IJ2,MH1), (IJ3,MV2), (IJ4,MV1)
JFLD(I,J,IW)=SHIFT(IW,J+I-60).A..N.MASK(60-J)

C----- MH2 = IJ1 = (I,K+1)
C          MH1 = IJ2 = (I,K-1)
C          MV2 = IJ3 = (I+1,K)
C          MV1 = IJ4 = (I-1,K)

C      CTEK=10000.76
TEK(1)=(SATK(JAM)*(PHI(JAM)-H(JAM))+SATK(MH2)*(PHI(MH2)-H(MH2)))
I * AA/2.0
TEK(2)=(SATK(JAM)*(PHI(JAM)-H(JAM))+SATK(MH1)*(PHI(MH1)-H(MH1)))
I * BB/2.0
TEK(3)=(SATK(JAM)*(PHI(JAM)-H(JAM))+SATK(MV2)*(PHI(MV2)-H(MV2)))
I * CC/2.0
TEK(4)=(SATK(JAM)*(PHI(JAM)-H(JAM))+SATK(MV1)*(PHI(MV1)-H(MV1)))
I * DD/2.0

100      Q1=TEK(1)+TEK(2)
          Q2=TEK(3)+TEK(4)
C----- THE FOLLOWING 4 CARDS ARE USED ONLY TO FORCE THE PROPER INDEXING.
          IJ1=IJ1
          IJ2=IJ2
          IJ3=IJ3
          IJ4=IJ4
          PART1= (TEK(1)*PHI(IJ1)+TEK(2)*PHI(IJ2))
          PART2=TEK(3)*PHI(IJ3)+TEK(4)*PHI(IJ4)
          PART3=Q2

```


30	SRTWO	
29	FFREV	
30	FFREV	
32	SRTWO	
33	SRTWO	
34	SRTWO	
35	SRTWO	
2	SRSTEP	
2	COM1	
3	COM1	
4	COM1	
5	COM1	
6	COM1	
7	COM1	
1	OCT10A	
9	COM1	
10	COM1	
1	IPRIG	
12	COM1	
2	IPRIG	
14	COM1	
3	IPRIG	
1	IPRIG2	
17	COM1	
5	IPRIG	
6	IPRIG	
20	COM1	
21	COM1	
22	COM1	
23	COM1	
24	COM1	
25	COM1	
26	COM1	
2	COMPK	
31	FFRFV	
6	SRSTEP	


```

PART4=Q1
NOJAM=NO(JAM)
IQ=JFLD(12,9,NOJAM)
QTERM=QTN(IQ)
F=(PART3+PART4)*PHI(JAM)-(PART1+PART2+QTERM)
RETURN
END
SUBROUTINE STEP (JOT,JIM,NTRIP)
COMMON /FRIT/ AYE,DEE,DELMIN,DUM1,DUM2,E,HI,ICOUNT,
* IHI, IHJ1, II, IIN, JPH, KHI, KHID1, KIN, KK, LNG, LSW, MHI,
* MH2, MV1, MV2, NAVG, NDUW2, NGRAP, NPH, NSIZE, NTYPE, NZ, N3,
* N4, NSH, PC3, PC4, PCTHF, PRE, RAT, SUMOMG, SUMIOL, THEM
* ,HIDONE
COMMON /RLK/ MINUTE,KRES
COMMON /VRLK/ ACFL, CODE(10), DELMAX, DELT(16), DELZ, FLENG,
* FLOW(7), FLOWS, H(5916), IBOT, IFLOWS, IMO, INH, INK, IPUN, IQ,
* ISTO, JDELP, JP, KTS, L, LSWP, MAXI, MAXK, MONO, MP, NRUG,
* NDELT, NDUW1, NE(87), NFO, NINPHI, NO(5916), NP, NQR, NS(87),
* NSTEP, NTOT,
* N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), PI, QIN(50),
* SATK(5916),STO(10),T(3),THEMAX,TIME,TPDIF
COMMON /RLK2/ A(68), B(68), C(68), JAM, P(68), R(10), TFK(4),
* 7(68)
COMMON /IQUNIT/ IONEWP, IOOLDP, LUINJ, LUPK, MIF
COMMON /RLK3/ AA, BB, CC, CON1, CON2, CON3, UD
COMMON /PALK/ ID1,ID2,ID3,ID4,ID5,ID6,ID7,ID8
COMMON /PKFY/ PK(126)
JFLD(1,J,IW)=SHIFT(IW,I+I-60).A..N.MASK(60-J)
JAM=JOT

```

FFREV	32
FFREV	33
FFREV	34
SRSTEP	9
SRSTEP	10
SRSTEP	11
SRSTEP	12
SRSTEP	13
CHGJR	1
SRSTEP	14
CHGJR	2
SRSTEP	15
SRSTEP	16
SRSTEP	17
SRSTEP	18
SRSTEP	19
SRSTEP	20
SRSTEP	21
SRSTEP	22
SRSTEP	23
SRSTEP	24
SRSTEP	25
SRSTEP	26
SRSTEP	27
SRSTEP	28
SRSTEP	29
SRSTEP	30
SRSTEP	31
SRSTEP	32
SRSTEP	33
SRSTEP	34
SRSTEP	35
SRSTEP	36
SRSTEP	37
SRSTEP	38
SRSTEP	39

```

715 NOJAM=NO(JAM)
   IMA=JFLD(30,3,NOJAM)
   IMAGE=JFLD(33,3,NOJAM)
   MV1=JAM-MAXK
   MV2=JAM+MAXK
   MH1=JAM-1
   MH2=JAM+1
   IF(IMA.NE.0)GO TO 50
   CON1=0.0
   IF(IMAGE.EQ.1)GO TO 43
   IF(IMAGE.EQ.2) GO TO 43
   CON1=1.0
   AA=1.0
   RR=1.0
   CC=1.0
   DD=1.0
   GO TO 44
50  IMAGE=IMAGE+1
   GO TO (40,41,42),IMA
40  CON1=0.5
   GO TO (1,3,7,12,13,15,19,24),IMAGE
   AA=0.5
   RR=0.5
   CC=0.0
   DD=1.0
   GO TO 44
3  AA=0.0
   RR=1.0
   CC=0.0
   DD=1.0
   GO TO 44
7  AA=0.0
   RR=1.0
   CC=0.5
   DD=0.5
   GO TO 44

```

40 SRSTEP
41 SRSTEP
42 SRSTEP
43 SRSTEP
44 SRSTEP
45 SRSTEP
46 SRSTEP
47 SRSTEP
48 SRSTEP
49 SRSTEP
50 SRSTEP
51 SRSTEP
52 SRSTEP
53 SRSTEP
54 SRSTEP
55 SRSTEP
56 SRSTEP
57 SRSTEP
58 SRSTEP
59 SRSTEP
60 SRSTEP
61 SRSTEP
62 SRSTEP
63 SRSTEP
64 SRSTEP
65 SRSTEP
66 SRSTEP
67 SRSTEP
68 SRSTEP
69 SRSTEP
70 SRSTEP
71 SRSTEP
72 SRSTEP
73 SRSTEP
74 SRSTEP
75 SRSTEP
76 SRSTEP

12 AA=0.0
RR=1.0
CC=1.0
DD=0.0
GO TO 44

13 AA=0.5
RR=0.5
CC=1.0
DD=0.0
GO TO 44

15 AA=1.0
RR=0.0
CC=1.0
DD=0.0
GO TO 44

19 AA=1.0
RR=0.0
CC=0.5
DD=0.5
GO TO 44

24 AA=1.0
RR=0.0
CC=0.0
DD=1.0
GO TO 44

41 CON1=0.375
GO TO (2,4,8,9,14,16,20,21),IMAGE

2 AA=0.0
RR=0.5
CC=0.0
DD=1.0
GO TO 44

4 AA=0.0
RR=1.0
CC=0.0
DD=0.5
GO TO 44

77	SRSTEP
78	SRSTEP
79	SRSTEP
80	SRSTEP
81	SRSTEP
82	SRSTEP
83	SRSTEP
84	SRSTEP
85	SRSTEP
86	SRSTEP
87	SRSTEP
88	SRSTEP
89	SRSTEP
90	SRSTEP
91	SRSTEP
92	SRSTEP
93	SRSTEP
94	SRSTEP
95	SRSTEP
96	SRSTEP
97	SRSTEP
98	SRSTEP
99	SRSTEP
100	SRSTEP
101	SRSTEP
102	SRSTEP
103	SRSTEP
104	SRSTEP
105	SRSTEP
106	SRSTEP
107	SRSTEP
108	SRSTEP
109	SRSTEP
110	SRSTEP
111	SRSTEP
112	SRSTEP
113	SRSTEP

8	AA=0.0
	RR=1.0
	CC=0.5
	DD=0.0
	GO TO 44
9	AA=0.0
	RR=0.5
	CC=1.0
	DD=0.0
	GO TO 44
14	AA=0.5
	RR=0.0
	CC=1.0
	DD=0.0
	GO TO 44
16	AA=1.0
	RR=0.0
	CC=0.5
	DD=0.0
	GO TO 44
20	AA=1.0
	RR=0.0
	CC=0.0
	DD=0.5
	GO TO 44
21	AA=0.5
	RR=0.0
	CC=0.0
	DD=1.0
	GO TO 44
42	CON1=0.625
	GO TO (18,17,23,22,6,5,11,10),IMAGE
14	AA=1.0
	RR=0.5
	CC=1.0
	DD=0.0
	GO TO 44

17 AA=1.0
 BB=0.0
 CC=1.0
 DD=0.5
 GO TO 44

23 AA=1.0
 BB=0.0
 CC=0.5
 DD=1.0
 GO TO 44

22 AA=1.0
 BB=0.5
 CC=0.0
 DD=1.0
 GO TO 44

6 AA=0.5
 BB=1.0
 CC=0.0
 DD=1.0
 GO TO 44

5 AA=0.0
 BB=1.0
 CC=0.5
 DD=1.0
 GO TO 44

11 AA=0.0
 BB=1.0
 CC=1.0
 DD=0.5
 GO TO 44

10 AA=0.5
 BB=1.0
 CC=1.0
 DD=0.0
 GO TO 44

43 NTRIP=2

44 JIM =JAM
 CON1=ACELL*CON1
 RETURN
 END

SRSTEP 114
 SRSTEP 115
 SRSTEP 116
 SRSTEP 117
 SRSTEP 118
 SRSTEP 119
 SRSTEP 120
 SRSTEP 121
 SRSTEP 122
 SRSTEP 123
 SRSTEP 124
 SRSTEP 125
 SRSTEP 126
 SRSTEP 127
 SRSTEP 128
 SRSTEP 129
 SRSTEP 130
 SRSTEP 131
 SRSTEP 132
 SRSTEP 133
 SRSTEP 134
 SRSTEP 135
 SRSTEP 136
 SRSTEP 137
 SRSTEP 138
 SRSTEP 139
 SRSTEP 140
 SRSTEP 141
 SRSTEP 142
 SRSTEP 143
 SRSTEP 144
 SRSTEP 145
 SRSTEP 146
 SRSTEP 147
 SRSTEP 148
 SRSTEP 149
 SRSTEP 150
 SRSTEP 151
 SRSTEP 152
 SRSTEP 153

SAMPLE INPUT DATA LIST

```

C-----
C      SURROUTINE OMEG(OMEGA)
C      THIS ROUTINE INTERPOLATES ON THE PAST EXPERIENCE FOR THE OPTIMUM
C      OMEGA (OVERRELAXATION FACTOR)
C-----
C      J - TIME PLANE INDEX
C      L - ITERATION INDEX WITHIN THE TIME PLANE
C      COMMON /FRIT/ AYE,DEE,DELMJN,DUM1,DUM2,F,HI,ICOUNT,
C      * IHI, IHID1, II, IIN, JPH, KHI, KHIDI, KIN, KK, LNG, LSW, MHI,
C      * MH2, MVI, MV2, NAVG, NNUM2, NGRAP, NPH, NSIZE, NTYPE, NZ, N3,
C      * N4, N5H, PC3, PC4, PRCHEF, PRE, RAT, SUMOMG, SUMIOL, THEM
C      * ,HIDONE
C
C      COMMON /BLK/ MINUTE,KRES
C
C      COMMON /VRBK/ ACFL, CODE(10), DELMAX, DFLT(16), DELZ, ELENG,
C      * FLOW(7), FLOWS, H(5916), IROT, IFLOWS, IMO, INH, INK, IPIN, IQ,
C      * ISTO, JDEFP, JP, KTS, L, LSWP, MAXI, MAXK, MCNO, MP, NBUG,
C      * NDELT, NDIW1, NE(87), NFO, NINPHI, NO(5916), NP, NQR, NS(87),
C      * NSTEP, NTOT,
C      * N1, N5, OM, OME(3), OPHI(5916), J, PHI(5916), PI, QIN(50),
C      * SATK(5916),STO(10),T(3),THEMAX,TIME,TPDIF
C
C      COMMON /BLK2/ A(68), R(68), C(68), JAM, P(68), P(10), TEK(4),
C      * Z(68)
C
C      COMMON /TOUNIT/ IONEWP, IOLLOP, LUINJ, LUPK, MIF
C
C      COMMON /BLK3/ AA, RB, CC, CON1, CON2, CON3, UD
C
C      COMMON /BLK/ ID1,ID2,ID3,ID4,ID5,ID6,ID7,ID8

```

SROMEG 2
 SROMEG 3
 SROMEG 4
 SROMEG 5
 SROMEG 6
 SROMEG 7
 COM1 8
 COM1 9
 COM1 10
 OCT10A 11
 COM1 12
 IRRIG 13
 COM1 14
 IRRIG 15
 IRRIG 16
 IRRIG 17
 IRRIG 18
 IRRIG 19
 IRRIG 20
 IRRIG 21
 IRRIG 22
 IRRIG 23
 IRRIG 24
 IRRIG 25

26 COM1
2 COMPK
10 SROMEG
1 ORXF
2 ORXF
12 SROMEG
13 SROMEG
14 SROMEG
3 ORXF
16 SROMEG
17 SROMEG
18 SROMEG
19 SROMEG
20 SROMEG
21 SROMEG
22 SROMEG
23 SROMEG
1 ORXV2
25 SROMEG
2 ORXV2
27 SROMEG
3 ORXV2
29 SROMEG
30 SROMEG
31 SROMEG
32 SROMEG
33 SROMEG
34 SROMEG
35 SROMEG
36 SROMEG
37 SROMEG
38 SROMEG
39 SROMEG
40 SROMEG
41 SROMEG

C
COMMON /PKY/ PK(126)
COMMON/OMRIK/NINPHY
FLTL=FLOAT(1)
T(1)=FLTL/DFLT(KTS)
IF(L.GT.0) GOTO 80
ONEOL=0.0
GOTO 90
80 ONEOL=1./FLTL/DELT(KTS)
90 CONTINUE
SUMIOL=SUMIOL+ONEOL
SUMOMG=SUMOMG+OMF(1)*ONEOL
IF(J.GT.NINPHY+3) GOTO 130
IF(J.LT.NINPHY+1) GOTO 100
IF(J-(NINPHY+2)) 100,110,120
100 SUMOMG=0.
SUMIOL=0.
0=1.2
GO TO 160
110 0=1.25
GO TO 160
120 0=1.3
GO TO 160
130 DEN1=(T(2)-T(3))*OME(1)
DEN2=(T(3)-T(1))*OME(2)
DEN3=(T(1)-T(2))*OME(3)
DEN0=(DEN1+DEN2+DEN3)*2.
IF(ABS(DEN0)-1.E-5) 180,140,140
140 OMA=DFEN1*OMF(1) + DEN2*OMF(2) + DEN3*OME(3)
IF(DEN0) 142,141,142
141 0=0.0
GOTO 143
142 0=OMA/DEN0
143 CONTINUE
IF(0-1.) 180,160,150

42 SPOMEG
 43 SPOMEG
 44 SPOMEG
 45 SPOMEG
 46 SPOMEG
 47 SPOMEG
 48 SPOMEG
 49 SPOMEG
 50 SPOMEG
 51 SPOMEG
 52 SPOMEG
 53 SPOMEG
 54 SPOMEG
 55 SPOMEG
 56 SPOMEG
 57 SPOMEG
 58 SPOMEG

150 IF(O=2.) 160,180,180
 160 OME(3)=OME(2)
 OME(2)=OME(1)
 OME(1)=0
 T(3)=T(2)
 T(2)=T(1)
 WRITE (3,170) 0
 170 FORMAT("OMEGA ="F10.7)
 OMEGA=0
 RETURN
 180 IF(SUM10L) 182,181,182
 181 O=0.0
 GOTO 183
 182 O=SUMOMG/SUM10L
 183 CONTINUE
 GO TO 160
 END

APPENDIX B
SUBROUTINE RIVEL

APPENDIX B

Basically, this subroutine is a set of twelve arithmetic statement functions for computing for VTT main program the Columbia River elevation along the Hanford Reservation as a function of flow rate over the range 6×10^9 to 3×10^{10} cubic feet per day. The statement functions, developed from field measurements of river elevation and flow rate, are second order to fifth order polynomials depending upon the minimum order which provides a satisfactory fit to the field data. The first function gives elevation at river mile 335 and the remaining ones are used for every fifth mile through mile 390. Potentials at nodes not exactly on five-mile marks are found by interpolation. Expressions for computing elevation are summarized in Table B-1. A symbolic listing of RIVEL appears in Appendix A.

Linkage to Main Program

The formal argument list is (F, E, ILOC). F represents the river flow rate (input) in cubic feet per day; E is the river elevation (output) in feet above mean sea level; ILOC is an index number (input) derived from nine bits of a packed word from array NO in the calling program. ILOC directs the subroutine to a work in an array PK. PK(ILOC) consists of an integer part and a fractional part. The integer part indicates which of the twelve statement functions are to be used; the fractional part is an interpolation factor. The array PK is assumed to have been read by another program and stored in labeled common /PKEY/. The dimension of PK is 256 for the 1000 foot grid; 126 for the 2000 foot grid. See description of Data File PKF for details of the PK table.

TABLE B-1

EXPRESSION FOR E, RIVER ELEVATION [FEET ABOVE MEAN SEA LEVEL]
AS A FUNCTION OF RIVER FLOW RATE, Q [10¹⁰ CU. FT. PER DAY]

<u>EXPRESSION NUMBER</u>	<u>FOR RIVER MILE</u>	
1	335	$E = 339.833 + 0.654277Q + 0.711298Q^2 - 0.0586758Q^3$
2	340	$E = 340.394 + 0.254644Q + 1.46336Q^2 - 0.182166Q^3$
3	345	$E = 338.478 + 4.79190Q + 0.0754782Q^2$
4	350	$E = 340.659 + 6.5198Q - 0.0253685Q^2 - 0.0970393Q^3$
5	355	$E = 343.394 + 11.4093Q - 2.28153Q^2 + 0.277618Q^3$
6	360	$E = 368.431 + 69.9823Q + 106.451Q^2 - 65.2764Q^3$ $+ 18.3614Q^4 - 1.9451Q^5$
7	365	$E = 348.363 + 15.2202Q - 2.79361Q^2 + 0.198399Q^3$
8	370	$E = 391.228 - 100.767Q + 147.454Q^2 - 91.8746Q^3$ $+ 26.4398Q^4 - 2.86183Q^5$
9	375	$E = 370.089 + 10.3119Q - 1.85539Q^2 + 0.13802Q^3$
10	380	$E = 374.274 + 12.5073Q - 2.85983Q^2 + 0.34901Q^3$
11	385	$E = 381.92 + 12.7138Q - 2.35154Q^2 + 0.209738Q^3$
12	390	$E = 388.623 + 2.78314Q + 10.7316Q^2 - 5.34693Q^3$ $+ 0.77552Q^4$

SUBROUTINE RIVEL

```

C..... SUBROUTINE RIVEL (FLOW,E,ILOC)
C..... THIS ROUTINE COMPUTES RIVER ELEVATION AS A FUNCTION OF FLOW RATE.
C..... APPLICABLE RANGE OF FLOW IS 0.1E+10 TO 6.9E+10 CU.FT./DAY
C      THIS VERSION IS FOR ELEVATIONS AFTER MCNARY POOL WAS ESTABLISHED.
C..... FLOW (INPUT) IS EXPECTED IN CURIC FEET PFR DAY
C..... E (OUTPUT) IS RIVER ELEVATION IN FEET ABOVE MEAN SEA LEVEL
C..... ILOC (INPUT) IS SUBSCRIPT OF THE PK VALUE TO BE USED IN CALCULA-
C      TING ELEVATION AT THE PRESENT POSITION ON THE BOUNDARY.
C      LOGICAL IDPRNT,INRNGE
C      COMMON /PKFY/ PK(126)
C      DATA NPKS/126/
C      DATA IDPRNT/.TRUE./
100  IF (FLOW.EQ.0) THEN
110  IF (ILOC.EQ.1) THEN
120  IF (FLOW.EQ.0) THEN
130  IF (FLOW.EQ.0) THEN
140  IF (FLOW.EQ.0) THEN
150  IF (FLOW.EQ.0) THEN
160  IF (FLOW.EQ.0) THEN
170  IF (FLOW.EQ.0) THEN
180  IF (FLOW.EQ.0) THEN
190  IF (FLOW.EQ.0) THEN
200  IF (FLOW.EQ.0) THEN
210  IF (FLOW.EQ.0) THEN
220  IF (FLOW.EQ.0) THEN
230  IF (FLOW.EQ.0) THEN
240  IF (FLOW.EQ.0) THEN
250  IF (FLOW.EQ.0) THEN
260  IF (FLOW.EQ.0) THEN
270  IF (FLOW.EQ.0) THEN
280  IF (FLOW.EQ.0) THEN
290  IF (FLOW.EQ.0) THEN
300  IF (FLOW.EQ.0) THEN
310  IF (FLOW.EQ.0) THEN
320  IF (FLOW.EQ.0) THEN
330  IF (FLOW.EQ.0) THEN
340  IF (FLOW.EQ.0) THEN
350  IF (FLOW.EQ.0) THEN

```


74 SP1VEL
 75 SP1VEL
 76 SP1VEL
 77 SP1VEL
 78 SP1VEL
 79 SP1VEL
 80 SP1VEL
 81 SP1VEL
 82 SP1VEL
 83 SP1VEL
 84 SP1VEL
 85 SP1VEL
 86 SP1VEL
 87 SP1VEL
 88 SP1VEL
 89 SP1VEL
 90 SP1VEL
 91 SP1VEL
 92 SP1VEL
 93 SP1VEL
 94 SP1VEL
 95 SP1VEL
 96 SP1VEL
 97 SP1VEL
 98 SP1VEL
 40 EXPV1
 100 SP1VEL
 101 SP1VEL

200 F=F6(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(E7(F)-F)*FF
 GOTO 270
 210 F=F7(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(F8(F)-F)*FF
 GOTO 270
 220 F=F8(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(F9(F)-F)*FF
 GOTO 270
 230 F=F9(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(F10(F)-F)*FF
 GOTO 270
 240 F=E10(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(F11(F)-F)*FF
 GOTO 270
 250 F=F11(F)
 IF(FF.LT..00001) GOTO 270
 F=E+(F12(F)-F)*FF
 GOTO 270
 260 F=E12(F)
 270 IF(.NOT.INPNGF) WP1TF(3,100) FLOW=E*1LOC
 RETURN
 END

APPENDIX C
SUPPORT PROGRAM CLASSY

SUPPORT PROGRAM CLASSY

```

C.....
C      FUNCTION FLDR(I,J,IA,N,IR)
C      DEAL IA,IR,M
C
C      THIS FUNCTION WHEN REFERENCED BY A FORTRAN STATEMENT OF THE FORM-
C      IA= FLDR(I,J,IA,N,IR)
C
C      HAS THE SAME EFFECT AS THE FOLLOWING CSCX FORTRAN V SEQUENCE
C
C      FLDR(I,J,IA)=FLP(N,J,IR)
C
C      THAT IS, J BITS OF IA, STARTING AT BIT POSITION I ARE REPLACED
C      BY J BITS FROM IR, STARTING AT BIT POSITION N.
C
C      (THE LEFTMOST BIT IN A WORD IS CONSIDERED TO BE IN BIT POSITION 0)
C
C      WHEN REFERENCED BY A STATEMENT OF THE FORM
C
C      IX= FLDR(I,J,IA,N,IR)
C
C      THE VALUE OF IA REMAINS UNCHANGED AND
C      THE EFFECT IS LIKE THE CSCX FORTRAN V SEQUENCE
C
C      IX=IA
C      FLDR(I,J,IX)=FLD(N,J,IR)
C
C      M=SHIFT(MASK(60-J),60-J-1)
C      FLDR=IA.AND.M.OR.SHIFT(IR,N-1).AND..NOT.M
C      RETURN
C      END

```



```

C
C      FUNCTION FLD(I,J,IW)
C      REAL IW
C
C      FORTRAN FUNCTION SUBPROGRAM FOR THE CDC6600 SYSTEM 3.4
C      WV OF MIER  BNW MATH BLOG  ROOM 1313  946-2370  5 APR 1973
C
C      C.... THIS FUNCTION WHEN REFERENCED BY A FORTRAN STATEMENT OF THE FORM
C
C      NWRD = FLD(I,J,IW)
C
C      HAS THE SAME EFFECT AS THE CSOX FORTRAN V STATEMENT
C
C      NWRD = FLD(I,J,IW)
C
C      THAT IS, THE LAST J BITS OF NWRD ARE REPLACED BY J BITS OF IW,
C      STARTING WITH THE ITH BIT OF IW.
C      (THE LEFTMOST BIT POSITION IS POSITION ZERO)
C
C      IF I IS GREATER THAN 0 THEN BITS IN POSITIONS ZERO THRU I-1 ARE
C
C      SET TO 0
C
C      FLD=SHIFT(IW,J+1-60).AND..NOT.MASK(60-J)
C      RETURN
C      END
C      PROGRAM CLASSY(INPUT,OUTPUT,LFA,LFP,TAPF1=LFA,TAPF2=LFP,
C      * TAPF5=INPUT,TAPF6=OUTPUT)
C
C      C***** PROGRAM CLASSY
C
C      C----- THIS PROGRAM COMPUTES:  ARS( A(I,J)-P(I,J) ) AND PRINTS A MAP
C      C      OF SYMBOLS INDICATING THE RANGES IN WHICH THE DIFFERENCES LIF.
C
C      C----- ADDITIONAL INFORMATION SUCH AS THE RMS DIFFERENCE AND THE VALUES
C      C      AND LOCATIONS OF HIGHEST AND LOWEST DIFFERENCES IS ALSO PRINTED.
C

```

C***** I N P U T D E S C R I P T I O N

ARRAY A WILL BE READ FROM LOGICAL FILE LFA ON UNIT 1.
 ARRAY P WILL BE READ FROM LOGICAL FILE LFP ON UNIT 2.

IT IS ASSUMED THAT THE ARRAYS WERE WRITTEN BY BUFFEROUT
 USING A SEQUENCE OF THE FORM

```

•   LANG=NR*NC
•   COMMON J,X(NR*NC)
•   BUFFEROUT(LFX,1)(J,X(LNG))
•
•

```

WHERE X STANDS FOR A OR P,
 NC STANDS FOR THE NUMBER OF COLUMNS IN X
 NR STANDS FOR THE NUMBER OF ROWS IN X.
 NHL STANDS FOR THE NUMBER OF HI AND LO DIF

THUS WORD 1 OF THE RECORD CONTAINS THE
 TIMEPLANE NUMBER, AND THE NEXT NC WORDS CONTAIN
 THE VALUES FOR ROW 1 OF THE ARRAY.

C**** DATA CARD 1 (13A6,A2)
 C TITLE CARD

C**** DATA CARD 2 (16I5)
 C FIELD 1 THE NUMBER OF THE TIMEPLANE SOUGHT ON LFA
 C FIELD 2 THE NUMBER OF THE TIMEPLANE SOUGHT ON LFP
 C FIELD 3 THE NUMBER OF CLASSIFICATION RANGES TO BE SPECIFIED.
 C FIELD 4 CONTROL VARIABLE, NFLG.
 C FOR NFLG=0 DIFFERENCES WILL BE A(J)-P(J)
 C FOR NFLG=1 DIFFERENCES WILL BE (A-P)*100./A

```

C**** DATA CARD 3 (10F7.0)
C      LOW END VALUES OF CLASSIFICATION RANGES (NS MAXIMUM)
C
C**** DATA CARD 4 (80A1)
C      LIST OF SYMBOLS CORRESPONDING TO CLASS RANGES (NS MAX)
C
C      NOTE THAT IF A(I,J)=0.0 OR P(I,J)=0.0 THEN THE CLASS SYMBOL IS
C      BLANK AND NO CALCULATIONS ARE DONE FOR THE I,JTH ELEMENT. THIS
C      FEATURE MAY BE USED TO SPECIFY SUPPRESSION OF CALCULATIONS AT ANY
C      POINT IN THE ARRAY BY DEFINING ELEMENTS TO BE ZERO.
C
C**** DATA CARDS 5 (16I5)
C      MAPPING SUPPRESSION SPECIFICATIONS
C
C      FIELD 1 IS A ROW NUMBER NRB TO WHICH THE REMAINING
C      COLUMN RANGES ON THIS CARD APPLY
C
C      FIELDS 2 THRU 15 CONTAIN PAIRS OF LS,LF WHICH
C      SPECIFY THAT COLUMNS LS THRU LE IN ROW NRB ARE NOT
C      TO BE MAPPED. IF THE LAST VALUE OF LE IS OMITTED
C      THEN A VALUE OF NC, THE RIGHTMOST COLUMN OF THE
C      MATRIX IS ASSUMED FOR LE.
C
C      FIELD 16 MAY BE USED TO SPECIFY A VALUE OF LS IF
C      FIELDS 1 THRU 15 ARE ALREADY FILLED. A VALUE OF NC
C      WILL BE ASSUMED FOR LE IN THIS CASE
C
C      THE FOLLOWING FORTRAN-V PARAMETER STATEMENTS HAVE BEEN REWRITTEN AS A
C      DATA STATEMENT TO CONFORM TO THE MORE LIMITED FORTRAN-IV EXTENDED VER-
C      SION OF FORTRAN USED BY THE CYBER SYSTEM.
C      PARAMETER NR=87, NC=68, NHL=5, MAXIP=50, NSMAX=10
C      PARAMETER NFL=NR*NC, NSP1=NSMAX+1
C
C      THE FOLLOWING DIMENSION SPECIFICATIONS FROM FORTRAN-V HAVE BEEN
C      SIMPLIFIED FOR THE CYBER, USING INTEGERS INSTEAD OF PARAMETERS.
C

```



```

70 FORMAT(' SUPPRESSION CARDS READ')
80 FORMAT(' TIMEPLANE NUMBER',I5,' READ FROM TAPE',I2)
90 FORMAT(1H1,13A6,A2)
100 FORMAT(13A6,A2)
120 FORMAT(16I5)
140 FORMAT(' REQUESTED TIME PLANE NOT FOUND AFTER',I4,' TRIES ON TAP
      *E',I2)
150 FORMAT(80A1)
160 FORMAT(10F7.0)
170 FORMAT(1H1)
180 FORMAT(' CODE FOR DIFFERENCE CLASSES-' )
190 FORMAT(31X,A1,' ARS(DIFF) LESS THAN OR EQUAL TO',1PE11.4,
      1, AND GREATER THAN',F10.4)
200 FORMAT(1H ,90X,4I10)
210 FORMAT(1H ,13I10)
211 FORMAT(' ***** NOTE- THE MAPS IN THIS RUN ARE FOR PERCENT DIFFER
      *NCES (A(J)-P(J))*100/A(J)')
220 FORMAT(1H ,13I11)
230 FORMAT(' ')
260 FORMAT(' SMALLEST VALUE IN ARRAY A IS',1PE11.4,' AT ROW',I4,
      * , COLUMN',I4)
270 FORMAT(' LARGEST ABS DIFFERENCES',10X,' SMALLEST ABS DIFFERENCES' /
      1 , ROW COL VALUE ,10X,' ROW COL VALUE')
280 FORMAT(1H ,I5,I6,1PE12.4,10X,I5,I6,F12.4)
290 FORMAT(31X,A1,' ARS(DIFF) LESS THAN OR EQUAL TO',1PE11.4)

300 FORMAT(31X,A1,' ARS(DIFF) GREATER THAN',1PE11.4)
310 FORMAT(' ALGEBRAIC SUM OF (A-R) IS',1PE12.5)
320 FORMAT(' ALGEBRAIC SUM OF ELEMENTS IN ARRAY A IS',1PE12.5)
330 FORMAT(' ALGEBRAIC SUM OF ELEMENTS IN ARRAY P IS',1PE12.5)
340 FORMAT(1H0,' RMS DIFF=',1PE11.4,' NUMBER OF DIFFERENCES=',I5)
350 FORMAT(' LARGEST VALUE IN ARRAY A IS', 1PE11.4,' AT ROW',I4,
      * , COLUMN',I4)
360 FORMAT(' AVERAGE VALUE IN ARRAY P IS',1PE12.5)
370 FORMAT(' AVERAGE VALUE IN ARRAY A IS',1PE12.5)
380 FORMAT(1X,F8.2,' PERCENT OF THE DIFFERENCES ARE IN CLASS ',A1)
390 FORMAT(' ARITHMETIC MEAN DIFFERENCE IS',1PE12.4)

```

```

400 FORMAT('0LARGEST VALUE IN ARRAY P IS',1PE11.4,' AT ROW',I4,
* , COLUMN',I4)
410 FORMAT('0SMALLEST VALUE IN ARRAY P IS',1PE11.4,' AT ROW',I4,
* , COLUMN',I4)
C
C**** FORMAT 420 IS SPECIAL FOR 68 COL MATRIX. THE 68 SHOULD BE REPLACED
C BY THE ACTUAL VALUE OF NC. IF NC IS GREATER THAN 113 THE 10X SPEC-
C IFICATION MUST BE SHORTENED. IF NC IS GREATER THAN 128 REMOVE THE
C 'ROW' SPEC, ETC. TO FIT 131 MAX CHARACTERS FOR ENTIRE FORMAT.
C
420 FORMAT('1H ,68A1,10X',ROW',I5)
C
430 FORMAT('0NUMBER OF COLUMNS TOO HIGH. 258 IS MAXIMUM.')
440 FORMAT('0FORMAT FOR SYMBOL MAP IS ',5A6)
445 FORMAT('0PARITY ERROR READING TAPE',I1)
C
LE(8) = 0
NFE=NR*NC
NSD1=NSMAX+1
C
C----- MODIFY OUTPUT FORMAT FOR CLASS OUTPUT
C
IF(NC.LT.259) GOTO 450
WRITE(6,430)
STOP
450 IF(NC-136) 470,510,460
C
C----- RESTRUCTURE FMT(4) FOR NC.GT.258
460 KADD=NC-131
IH=IHF(KADD)
FMT(4)=FLDR(6,6,FMT(4),0,HD(IH))
IT=ITF(KADD)
FMT(4)=FLDR(12,6,FMT(4),0,HO(IT))
IU=IUF(KADD)
FMT(4)=FLDR(18,6,FMT(4),0,HD(IU))
GOTO 510

```

```

C 470 IF(NC-127) 500,480,490

480 FMT(2)=C127
    FMT(3)=HR10
    FMT(4)=HR10
    GOTO 510

C
C----- RESTRUCTURE FMT(2) AND FMT(4) FOR NC.GT.127
490 FMT(2)=FLDR(12,6,FMT(2),0,HD(3))
    FMT(2)=FLDR(18,6,FMT(2),0,HD(8))
    KADD=NC-127
    IH=IHF(KADD)
    FMT(4)=FLDR(6,6,FMT(4),0,HD(IH))
    IT=ITF(KADD)
    FMT(4)=FLDR(12,6,FMT(4),0,HD(IT))
    IU=IUF(KADD)
    FMT(4)=FLDR(18,6,FMT(4),0,HD(IU))
    GOTO 510

C
C----- RESTRUCTURE FMT(2),FMT(3),FMT(4),FOR NC.LT.127
500 FMT(3)=HB10
    FMT(4)=HR10
    IH=IHF(NC)
    FMT(2)=FLDR(6,6,FMT(2),0,HD(IH))
    IT=ITF(NC)
    FMT(2)=FLDR(12,6,FMT(2),0,HD(IT))
    IU=IUF(NC)
    FMT(2)=FLDR(18,6,FMT(2),0,HD(IU))

C
CSDS=.FALSE.
510 WRITE(6,440) FMT
    READ(5,100) HDG
    WRITE(6,90) HDG
C

```

```

C----- NTA IS THE NUMBER OF THE TIMEPLANE SOUGHT ON TAPE 1.
C NTP IS THE NUMBER OF THE TIMEPLANE SOUGHT ON TAPE 2.
C NS IS HOW MANY CLASSES TO ASSIGN (10 MAXIMUM)
C NELG IS SWITCH FOR DIF OR PERCENT DIF
C READ(5,120) NTA,NTP,NS,NELC
C
C----- READ LOW ENDS OF DIF CLASS RANGES
C READ (5,160) (C(I),I=1,NS)
C
C----- READ CLASS SYMBOLS
C READ (5,150) (S(I),I=2,NSP1)
C
C NTRY=0
C----- READ ARRAY A
C 520 LUR=1
C RUFFERIN(1,1)(J,A(NFL))
C IF(UNIT(1)) 540,540,525
C
C 525 WRITE(6,445) LUR
C STOP
C
C 540 NTRY=NTRY+1
C IF(NTRY.LE.MAXTP) GOTO 550
C----- REQUESTED TIMEPLANE NOT FOUND AFTER MAXTP TRIES.
C 545 NTRY=NTRY-1
C WRITE(6,140) NTRY,LUR
C STOP
C 550 IF(J.NE.NTA)GOTO 520
C
C C..... HERE THE DESIRED TIMEPLANE HAS BEEN READ FROM TAPE 1.
C WRITE(6,80) J,LUR
C REWIND LUR
C

```



```

WRITE(6,50) J
DO 1575 IR=1,NR
  KCR=(IR-1)*NC+1
  KCF=KCR+NC-1
  WRITE(6,49) IR
  WRITE(6,51) (A(KC),KC=KCR,KCF)
1575 CONTINUE
C INITIALIZE COUNTERS
DO 560 I=1,NS
560 NNC(I)=0
NSW=0
SMALC=1.E+38
SMALM=1.E+38
BIGC=0.0
BIGM=0.0
ASUM=0.0
SUM=0.0
SCS=0.0
DC=0.0
PMS=0.0
NOP=0
C
NFM1=NFL-1
NSM1=NS-1
C
C..... READ ARRAY P
NTRY=0
LUR=2
565 BUFFERIN(2,1)(JP,P(NFL))
  IF(UNIT(2)) 570,570,525
570 NTRY=NTRY+1
  IF(NTRY.LE.MAXTP) GOTO 575
  GOTO 545

```

```

575 IF(JP.NF.NTP) GOTO 565
C.... THE DESIRED TIMEPLANE HAS BEEN READ FROM TAPE 2. START ANALYSIS.
WRITE(6,80) JP,LUR
WRITE(6,50) JP
REWIND LUR
DO 555 IR=1,NR
KCR=(IR-1)*NC+1
KCF=KCR+NC-1
WRITE(6,49) IR
WRITE(6,51) (P(KC),KC=KCR,KCF)
555 CONTINUE
C
GOTO 750
576 CONTINUE
IF(.NOT.SCRDS) WRITE(6,60)
IF(SCRDS) WRITE(6,70)
C
DO 680 I=1,NR
KS=I*NC-NC + 1
KF=KS + NC - 1
C
PROCESS A ROW
C
580 DO 650 J= KS,KE
JC=J-NC*(I-1)
DCS(JC)=S(I)
IF(A(J).LT.1.E-40.OR.P(J).LT.1.E-40) GOTO 650
C
C----- ALGEBRAIC SUMS
PM=PM+P(J)
PC=PC+A(J)
DTF= A(J)-P(J)
TF(NFLG.EQ.1) DTF=DTF*100./A(J)
CMM(I,JC)=PLUS
TF(DTF.LT.0.0) CMM(I,JC)=MINJS
ASUM=ASUM+DTF
TH=I
JH=JC
IL=I
JL=JC
C

```

C----- SAVE HIGH AND LOW DIFFERENCES AND THEIR LOCATIONS.

CALL LOLO(DIF,IL,JL,KSIG)
CALL HITOP(DIF,JH,JH,KSIG)

C.....ASSIGN CLASS SYMBOLS
C

DO 590 NR=1,NS
NT=NS-NR+1

IF(ABS(DIF).GT. C(NT)) GOTO 600
DCS(JC)=S(1)
590 CONTINUE
600 DCS(JC)=S(NT+1)
NMC(NT)=NMC(NT)+1

C..... SAVE LARGEST VALUE IN ARRAY A
C

IF(A(J).LT.RIGC) GOTO 610
RIGC=A(J)
JRIGC=JC
IRIGC=I

C..... SAVE LARGEST VALUE IN ARRAY P
C

610 IF(P(J).LT.RIGM) GOTO 620
RIGM=P(J)
JRIGM=JC
IRIGM=I

C.....SAVE SMALLEST VALUE IN ARRAY A
C

620 IF(A(J).GT.SMALC) GOTO 630
SMALC=A(J)
JLILC=JC
ILILC=I

C

```

C..... SAVE SMALLEST VALUE IN ARRAY P
C
630 IF(P(J).GT.SMALM) GOTO 640
    JLILM=JC
    SMALM=P(J)
    ILILM=I
C     SAVE SUM OF SQUARES OF DIFFERENCES
640 SQS=SQS+DIF**2
C     INCREMENT DIFFERENCE COUNTER
    NOD=NOD+1
650 CONTINUE
    DO 660 L=1,NC
660 STAB(L,L)=DCS(L)
    DO 670 II=1,NC
670 DCS(II)=S(1)
680 CONTINUE
    ACD=PC/NOD
    AMP=PM/NOD
    AMEAN=ASUM/NOD
    PMS=SQRT(SQS/NOD)
    WRITE(6,270)
    WRITE(6,280) (TRH(I),ICH(I),HIV(I),IRL(I),ICL(I),VLO(I),I=1,NHL)
    WRITE(6,290) AMEAN

    WRITE(6,340) RMS,NOD
    WRITE(6,350) RIGC,IRIGC,JRIGC
    WRITE(6,260) SMALC,JLILC,JLILC
    WRITE(6,400) RIGM,IRIGM,JRIGM
    WRITE(6,410) SMALM,ILILM,JLILM
    ENOD=NOD
    DO 690 I=1,NS
690 PNC(I)=100.*NNC(I)/ENOD
    WRITE(6,230)
    WRITE(6,380) (PNC(I),S(I+1),I=1,NS)
    WRITE(6,310) ASUM
    WRITE(6,320) PC

```

```

WRITE(6,330) PM
WRITE(6,370) ACP
WRITE(6,360) AMP
C.....PRINT CLASS SYMBOLS AND RANGES
WRITE(6,170)
WRITE(6,180)
WRITE(6,290) S(2),C(2)
WRITE(6,190) S(I+1),C(I+1),C(I),I=2,NSM1)
WRITE(6,300) S(NS+1),C(NS)
C
C.....PRINT CLASS TABLE HEADING
IF(NFLG.EQ.1) WRITE(6,211)
C
700 WRITE(6,200) KNO(1),KNO(1),KNO(1),KNO(1)
WRITE(6,210) KNO,(KNO(K),K=1,3)
WRITE(6,220) KNO,KNO,KNO,KNO,KNO,KNO,KNO,KNO,KNO,KNO,KNO,
1 KNO,KNO(1)
IF(NSW.EQ.2) GOTO 730
NR0W = NR
00 720 IR=1,NROW
I=NR0W-IR+1
710 WRITE(6,FMT)(STAB(I,J),J=1,NC),I
720 CONTINUE
NSW=2
WRITE(6,170)
GOTO 700
730 00 740 IR=1,NROW
I=NR0W-IR+1
WRITE(6,FMT)(CMM(I,J),J=1,NC),I
740 CONTINUE
STOP
C
C----- READ BLANKOUT SPECS (SET ZERO ELEMENTS) AND GO BACK TO 576
C
750 READ(5,120) NRR,(LS(I),LF(I),I=1,7),LS(8)
IF(EOF(5)) 576,755,576

```

```

755 IF(NRR.LT.1) GOTO 576
   IF(NRR.GT.NR) STOP 'SPECIFIED PLANKOUT ROW NUMBER IS TOO BIG'
760 JFL=NRR*NC-NC
   SCRDS=.TRUE.
   DO 800 NK=1,8
     IF(LS(NK).EQ.0) GOTO 750
     IF(LF(NK).EQ.0) LF(NK)=NC
     IF(LS(NK).GT.NC) STOP 'PLANKOUT COLUMN SPEC LS IS TOO BIG'
     IF(LE(NK).GT.NC) STOP 'PLANKOUT COLUMN SPEC LE IS TOO BIG'
     K1=JFL+LS(NK)
     K2=JFL+LE(NK)
     DO 770 NZ=K1,K2
       D(NZ)=0.0
770 CONTINUE
     IF(LF(NK).EQ.NC) GOTO 750
800 CONTINUE
   GOTO 576
END

SUBROUTINE LOLO(V,IR,IC,KSIG)
C.....THIS ROUTINE MAINTAINS AN ARRAY, VLO, (DIMENSIONED NVAL) WHICH
C CONTAINS THE NVAL LOWEST ABSOLUTE DIFFERENCES FOUND SO FAR.
C.....IF THE ABSOLUTE VALUE OF V IS GREATER THAN THE ABSOLUTE VALUE OF
C VLO(NVAL) THEN KSIG IS SET TO ZERO AND CONTROL IS RETURNED TO THE
C CALLING PROGRAM
C.....IF THE ABSOLUTE VALUE OF V IS NOT GREATER THAN THE ABSOLUTE
C VALUE OF VLO(NVAL) THEN V REPLACES VLO(NVAL) AND THE ARRAY VLO IS
C REORDERED
C
COMMON/LO5/IRL(5),ICL(5),VLO(5)
LOGICAL FIRST
DATA NVAL/5/
DATA FIRST/.TRUE./
IF(.NOT.FIRST) GOTO 110
FIRST=.FALSE.
DO 100 I=1,NVAL
  VLO(I)=1.E+30
100 CONTINUE
110 CONTINUE
IF(ABS(V).LE.ABS(VLO(NVAL))) GOTO 120
KSIG=0
RETURN

```

```

120 VLO(NVAL)=V
   IRL(NVAL)=IR
   ICL(NVAL)=IC
   DO 130 IB=2,NVAL
   I=NVAL-IB+2
   IF(ABS(VLO(I)).GE.ARS(VLO(I-1)))GOTO 130
   VK=VLO(I)
   IRK=IRL(I)
   ICK=ICL(I)

   IRL(I)=IRL(I-1)
   ICL(I)=ICL(I-1)
   VLO(I)=VLO(I-1)
   IRL(I-1)=IRK
   ICL(I-1)=ICK
   VLO(I-1)=VK
130 CONTINUE
   KSIG=1
   RETURN
   END

SUBROUTINE HITOP(V,IR,IC,KSIG)
C.....THIS ROUTINE MAINTAINS AN ARRAY, HIV, (DIMENSIONED NVAL) WHICH
C   CONTAINS THE NVAL HIGHEST ABSOLUTE DIFFERENCES FOUND SO FAR.
C.....IF THE INPUT VALUE OF ABS(V) IS LESS THAN HIV(1) THEN KSIG IS SET
C   TO ZERO AND CONTROL IS RETURNED TO CALLING PROGRAM WITH NO FURTHER
C   ACTION BY SUBROUTINE HITOP.
C.....IF THE ABSOLUTE VALUE OF V IS GREATER THAN OR EQUAL TO THE
C   ABSOLUTE VALUE OF HIV(1) THEN V REPLACES HIV(1) AND THE ARRAY HIV
C   IS ORDERED SO THAT THE LOWEST ABSOLUTE VALUE IN THE ARRAY OCCUPIES
C   POSITION 1
C
COMMON/HI5/IRH(5),ICH(5),HIV(5)
DATA NVAL/5/
IF(ABS(V).GE.ARS(HIV(1)))GOTO 100
KSIG=0
RETURN

```

```

100 HIV(1)=V
   IRH(1)=IR
   ICH(1)=IC
   DO 110 IB=2,NVAL
      I=IVAL-IR+2
      IF(ABS(HIV(1)).GE.ABS(HIV(I-1))) GOTO 110
      IRK=IRH(I)
      ICK=ICH(I)
      VK=HIV(I)
      IRH(I)=IRH(I-1)
      ICH(I)=ICH(I-1)
      HIV(I)=HIV(I-1)
      IRH(I-1)=IRK
      ICH(I-1)=ICK
      HIV(I-1)=VK
110 CONTINUE
   KSIG=1
   RETURN
   END

```


DISTRIBUTION

<u>No. of Copies</u>	<u>Off-Site</u>
1	<u>Chicago Patent Group</u> A. A. Churm
215	<u>Energy Research and Development Administration Technical Information Center</u>
2	<u>Energy Research and Development Administration Headquarters</u> Waste Management and Transportation Division Germantown, Maryland 20767 Owen P. Gornley Chief, Waste Facilities Branch Robert W. Ramsey, Jr. Chief, Development Branch
2	<u>Nuclear Regulatory Commission Directorate of Regulation</u> 7920 Norfolk Avenue Bethesda, Maryland 20014 Ronald L. Bullard Chief, Environmental Specialist Branch W. Gamill Chief, Site Analysis Branch
2	<u>Energy Research and Development Administration Division of Production & Material Management</u> Washington, D. C. 20545 William L. Lennemann Chief, Chemical Process James W. Pollock
2	<u>Energy Research and Development Administration</u> Idaho Falls, Idaho 83401 Dr. Adrian H. Dahl Chief, Environmental Science Branch B. L. Schmalz

<u>No. of Copies</u>	<u>Off-Site</u>
2	<u>U. S. Geological Survey</u> 2100 M. Street Washington, D. C. 20037 George Debuchananne, Director M. K. Hubbert Research Geophysicist
1	<u>U. S. Geological Survey</u> Menlo Park, California 94025 Jacob Rubin
1	<u>U. S. Geological Survey</u> Groundwater Branch Denver, Colorado 80225 R. W. Stallman
1	<u>U. S. Geological Survey</u> Water Resources Division Bldg. 53, Room H-2910 Lakewood, Colorado 80225 J. D. Bredehoeft
1	<u>U. S. Geological Survey</u> Water Resources Division P.O. Box 917 Council Bluffs, Iowa 51501 O. J. Ramsvick Supervisory Hydrologist
1	<u>U. S. Geological Survey</u> Water Resources Divison P.O. Box 2857 Raleigh, North Carolina 27202 Ralph C. Heath District Chief
1	<u>U. S. Bureau of Mines</u> 4800 Forbes Avenue Pittsburgh, Pennsylvania 15213 George Evans

No. of
Copies

Off-Site

2

U. S. Bureau of Mines
Spokane Mining Research Laboratory
N. 1430 Washington St.
Spokane, Washington 99201

Robert C. Bates

Michael M. McDonald

1

Environmental Protection Agency
Crystal Mall
Washington, D. C. 20460

Arnold Joseph

1

Environmental Protection Agency
Office of Research & Monitoring
Edison Water Quality Office
Edison, New Jersey 08817

Richard Field

1

National Water Commission
800 N. Quincy
Arlington, Virginia 22209

Dr. John S. Gladwell

1

USDA-ARS
Engineering Research Center
CSU Foothills Campus
Fort Collins, Colorado 80521

Dr. David A. Woolhiser
Hydraulic Engineer

1

USDA-ARS
P.O. Box 1096
Boise, Idaho 83701

J. L. Robins

1

USDA-ARS
Snake River Conservation Research Center
Route 1, Box 186
Kimberly, Idaho 83341

Dr. Marvin E. Jensen, Director

No. of
Copies

Off-Site

- 1 USDA Hydrograph Laboratory
Plant Industry Station
Soils Building
Beltsville, Maryland 20705
- H. N. Holton, Director
- 1 USDA Sedimentation Laboratory
P.O. Box 30
Oxford, Mississippi 38655
- A. R. Robinson, Director
- 1 USDA-ARS
North Central Watershed Research Center
P.O. Box 916
Columbia, Missouri 65201
- Dr. C. R. Amerman
Hydraulic Engineer
- 1 USDA-ARS
Northeast Watershed Research Center
Pennsylvania State University
111 Research Building 3
University Park, Pennsylvania 16802
- Dr. A. S. Rogowski
Soil Scientist
- 1 National Academy of Science
U. S. National Committee for the IHD
2101 Constitution Avenue
Washington, D. C. 20418
- Dr. Leo Heindl
Executive Secretary
- 1 National Academy of Science
Committee on Radioactive Waste Management
National Research Council
2101 Constitution Avenue
Washington, D. C. 20418
- Dr. Cyrus Klingsberg
Technical Secretary

<u>No. of Copies</u>	<u>Off-Site</u>
1	<u>National Science Foundation</u> Division of Advanced Technology Application 1800 Sixth Street N.W. Washington, D. C. 20530 Ray Zahradnik
1	<u>Robert S. Kerr Water Research Center</u> P.O. Box 1198 Ada, Oklahoma 74820 Dr. J. W. Keeley
1	<u>Alabama Geologic Survey</u> P.O. Box Drawer 0 University, Alabama 35486 Phil Lammereaux State Geologist
1	<u>Idaho Bureau of Mines & Geology</u> University of Idaho Moscow, Idaho 83843 Dr. Roy E. Williams
1	<u>Idaho Department of Water Administration</u> State House, Annex 2 Boise, Idaho 83707 Keith R. Higginson, Director
1	<u>Idaho Water Resources Board</u> Boise, Idaho R. R. Lee
1	<u>Illinois State Water Survey</u> P.O. Box 232 Urbana, Illinois 61801 Keros Cartwright
1	<u>State Geological Survey of Kansas</u> University of Kansas Lawrence, Kansas 66044 Dr. John C. Halepaska Chief, Water Resources Section

<u>No. of Copies</u>	<u>Off-Site</u>
1	<u>State of Maine</u> <u>Environmental Improvement Commission</u> Augusta, Maine 04330 Steven D. Freedman
1	<u>St. Anthony Falls Hydraulic Laboratory</u> <u>Mississippi River at Third Avenue East</u> Minneapolis, Minnesota 55414 Professor C. Edward Bowers
1	<u>Desert Research Institute</u> <u>Center for Water Resources Research</u> Reno, Nevada Dr. Clinton Case
1	<u>State of North Carolina</u> <u>Groundwater Division</u> Water and Air Resources Raleigh, North Carolina 27611 Harry M. Peek, Chief
1	<u>Water Quality Management Laboratory</u> <u>Rt. 2, Box 322A</u> Durant, Oklahoma 74701 V. L. Hauser
1	<u>State of Oregon</u> <u>516 Public Service Bldg.</u> Salem, Oregon 97310 Chris Wheeler State Engineer
1	<u>Agencies for Environmental Conservation</u> <u>Montpelier, Vermont 05602</u> Dr. Martin L. Johnson Secretary

No. of
Copies

Off-Site

1	<u>State of Washington</u> Department of Ecology P.O. Box 829 Olympia, Washington 98504 John R. Raymond
1	<u>Washington State Water Research Center</u> Pullman, Washington 99163 James Crosby, III Geologist
1	<u>Atomic Industrial Forum, Inc.</u> 475 Park Avenue South New York, New York 10016 Gerald V. Halvorsen Environmental Projects Manager
1	<u>Computer Sciences Corporation</u> Federal Building Richland, Washington 99352 R. W. Nelson
1	<u>Sandia Laboratories</u> Division 5166 Albuquerque, New Mexico 87115
1	<u>University of Arizona</u> Office of Arid Lands Studies 1201 East Speedway Blvd. Tucson, Arizona 95719 Dr. Ken Faster
1	<u>University of Arizona</u> Department of Hydrology and Water Resources 200-E Old Psychology Tucson, Arizona 95721 Dr. A. K. Tyagi

<u>No. of Copies</u>	<u>Off-Site</u>
1	<u>University of Arizona</u> Tucson, Arizona 95721 E. S. Simpson
2	<u>University of California</u> Berkeley, California 94720 Dr. Warren Kauffman P. A. Witherspoon Department of Civil Engineering
1	<u>University of California at Davis</u> Department of Water Sciences and Engineering Davis, California 95616 J. N. Luthin
1	<u>University of California at Riverside</u> College of Biological & Agricultural Sciences Riverside, California 92507 Dr. N. T. Coleman
1	<u>Stanford University</u> Department of Civil Engineering Stanford, California 94305 Paul Kruger
1	<u>Stanford University</u> School of Earth Sciences Department of Ecology Stanford, California 94305 E. Aguado
1	<u>Georgia Institute of Technology</u> School of Civil Engineering Atlanta, Georgia 30332 Dr. J. R. Wallace

<u>No. of Copies</u>	<u>Off-Site</u>
2	<u>University of Illinois</u> <u>Urbana, Illinois 61801</u> A. Klute Professor Ven Te Chow Hydrosystems Laboratories
1	<u>Purdue University</u> <u>Dept. of Agricultural Engineering</u> <u>West LaFayette, Indiana 47907</u> Emru D. Millette
1	<u>Iowa State University</u> <u>Ames, Iowa</u> D. Kirkham
1	<u>University of Kentucky</u> <u>Department of Mechanical Engineering</u> <u>Lexington, Kentucky 40506</u> C. T. Hahn
1	<u>Harvard University</u> <u>School of Public Health</u> <u>Boston, Massachusttts 02138</u> Dr. D. W. Moeller
1	<u>Massachusetts Institute of Technology</u> <u>77 Massachusetts Avenue</u> <u>Room 48-263</u> <u>Cambridge, Massachusetts 02139</u> Dr. Donald Harleman
1	<u>University of Minnesota</u> <u>Department of Civil Engineering</u> <u>Minneapolis, Minnesota 55418</u> H. Steform

No. of
Copies

Off-Site

2	<u>University of Nebraska</u> School of Civil Engineering Lincoln, Nebraska 68503 Mark J. Hammer Warren Viessman, Jr.
1	<u>Princeton University</u> School of Engineering/Applied Science Department of Civil and Geological Engineering Princeton, New Jersey 08540 Dr. Robert W. Cleary
2	<u>Princeton University</u> Department of Civil and Geological Engineering Princeton, New Jersey 08540 R. J. M. DeWiest G. F. Pinder
1	<u>New Mexico State University</u> Department of Agronomy P. O. Box 3Q Las Cruces, New Mexico 88001 P. J. Wierenga
1	<u>New Mexico Institute of Mining and Technology</u> Department of Geosciences Socorro, New Mexico 87801 Lynn W. Gelhar
1	<u>State University of New York at Buffalo</u> Department of Mechanical Engineering Buffalo, New York 14214 Ralph T. Cheng

No. of
Copies

Off-Site

1	<u>State University of New York at Buffalo</u> <u>Department of Civil Engineering</u> Buffalo, New York 14214 Dr. Ralph Rumer
1	<u>Cornell University</u> <u>School of Civil Engineering</u> B22 Bailey Hall Ithaca, New York 14850 W. H. Brutsaert
1	<u>Oregon State University</u> <u>Corvallis, Oregon 97331</u> Dr. Larry Boersma Professor of Soils
1	<u>Drexel Institute of Technology</u> <u>Philadelphia, Pennsylvania 19104</u> Irwin Remson
1	<u>Vanderbilt University</u> <u>Nashville, Tennessee 37203</u> Dr. Frank L. Parker
1	<u>Utah State University</u> <u>Utah Water Research Laboratory</u> <u>College of Engineering</u> Logan, Utah 84321 J. Paul Riley
1	<u>University of Wyoming</u> <u>Laramie, Wyoming 92070</u> Dr. Paul A. Rechard Director, Water Resources Research
1	<u>Washington State University</u> <u>Department of Civil Engineering</u> 148 Sloan Hall Pullman, Washington 99163 Dr. Donald L. Bender

No. of
Copies

Off-Site

- 2 Washington State University
 Department of Agronomy and Soils
 Pullman, Washington 99163
- Dr. Walter H. Gardner
- Dr. Gaylon S. Campbell
- 1 Washington State University
 Albrook Hydraulic Laboratory
 Pullman, Washington 99163
- Dr. John F. Orsborn, Head
- 1 Washington State University
 Agricultural Engineering Department
 Pullman, Washington 99163
- Dr. L. G. King
- 1 University of Wisconsin
 Department of Soil Science
 Madison, Wisconsin 53706
- W. R. Gardner
- 1 Water Conservation and Irrigation
 Commission
 P. O. Box 220R Royal Exchange
 Sydney, 2000, Australia
- Frans R. Kalf
- 1 University of New South Wales
 School of Civil Engineering
 Sydney, Australia
- Dr. K. K. Watson
- 1 University of Waterloo
 Department of Civil Engineering
 Faculty of Engineering
 Waterloo, Ontario, Canada
- John Pickens

No. of
Copies

Off-Site

1	<u>McMaster University</u> Department of Civil Engineering Hamilton, Ontario, Canada Dr. William James
1	<u>Institut de Mecanique (I.M.G.)</u> Domaine Universitaire B. P. 53 Centre de Tri 38041 Grenoble-Cedex, France Georges Vachaud
1	<u>Technion-Israel Institute of Technology</u> Haifa, Israel J. Bear
1	<u>Agricultural Research Organization</u> The Volcani Center The Institute of Soils and Water P. O. B. 6 Bet Dagan Shlomo P. Neuman
1	<u>Kyoto University</u> Kyoto, Japan Yoriteru Inoue

On-Site Hanford

8	<u>AEC Richland Operations Office</u> O. J. Elgert J. L. Rhoades P. J. Holsted (2) E. B. Jackson P. G. Rhoades R. B. Goranson B. J. Melton
---	---

No. of
Copies

On-Site Hanford

Atlantic Richfield Hanford Co.

16
G. E. Backman
G. L. Borsheim
D. J. Brown
L. E. Bruns
R. A. Deju
R. E. Gephart
W. M. Harty
R. E. Isaacson (2)
L. M. Knights
C. W. Malody
B. J. McMurray
G. A. Nicholson
H. P. Shaw
P. W. Smith
ARHCO File

1 Westinghouse Hanford Co. HEDL

R. B. Hall

18 Battelle-Northwest

D. B. Cearlock (3)
J. R. Eliason
K. L. Kipp
D. E. Olesen
A. E. Reisenauer
R. C. Routson
R. J. Serne
D. R. Friedrichs
S. W. Ahlstrom
R. G. Baca
A. Brandstetter
R. W. Wallace
J. J. Fuquay
A. J. Shuckrow
Tech. Information
Tech. Pubs.

3 United Nuclear, Inc.

C. D. Corbit (2)
F. E. Owen