TEXAS DEPARTMENT OF WATER RESOURCES

UM-

GWSIM-IV

GROUND-WATER SIMULATION PROGRAM

Program Documentation and User's Manual

by

Tommy Knowles

Data Collection and Evaluation Section

Data and Engineering Services Division

1983

Table of Contents

List of Figures

Forward

GWSIM-IV, documented herein, is a digital modeling technique which is capable of simulating ground-water flow. and conservative mass transport. . The solution procedure was developed by T. A. Prickett and C. G. Lonnquist, Illinois State Water Survey, and was later modified by personnel of the Texas Department of Water Resources.

The purpose of the program is to determine water levels at the end of a given time period. The technique is based on the differential equation describing non-steady, two dimensional flow of ground water in a nonhomogeneous, anisotropic, water-table and/or artesian aquifer.

The purpose of the mass transport portion of this modeling technique is to determine concentrations of a conservative constitutuent at the end of a given time period. The technique is based on the differential equation governing non-steady state, two-dimensional mass transport in a non-homogeneous, anisotropic aquifer. An iterative alternating direction implicit procedure is used to solve the finite difference approximations to the governing differential equations.

GROUND-WATER SIMULATION PROGRAM

GWSIM-IV

INTRODUCTION

Hydraulic simulation was based on work of T. A. Prickett and C. G. Lonnquist, Illinois State Water Survey (Prickett, 1971). Modifications were made to the program to allow additional types of input and output and to improve the program's ability to simulate different aquifer configurations.

The program is structured to simulate water-table elevations, usually referred to as heads, for a given period of time. It advances through time by major time steps, which are further divided into one or more minor time steps.

Operation of the program is controlled by options. Proper selection of options allows the user to tailor model input, operation, and output to an individual problem.

HYDRAULIC SIMULATION

Development of Finite Difference Equation

The partial differential equation describing non-steady flow in a non-homogeneous aquifer may be written as follows:

$$\frac{\partial}{\partial x}\left(T\frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(T\frac{\partial h}{\partial y}\right) = S\frac{\partial h}{\partial t} + W \qquad (1)$$

where

T = aquifer transmissivity (L^2t^{-1})

h = head (L)

S = storage coefficient

t = time (t)

W = net ground-water flux per unit area (Lt^{-1})

x,y = rectangular coordinates (L)
(Prickett, 1971).

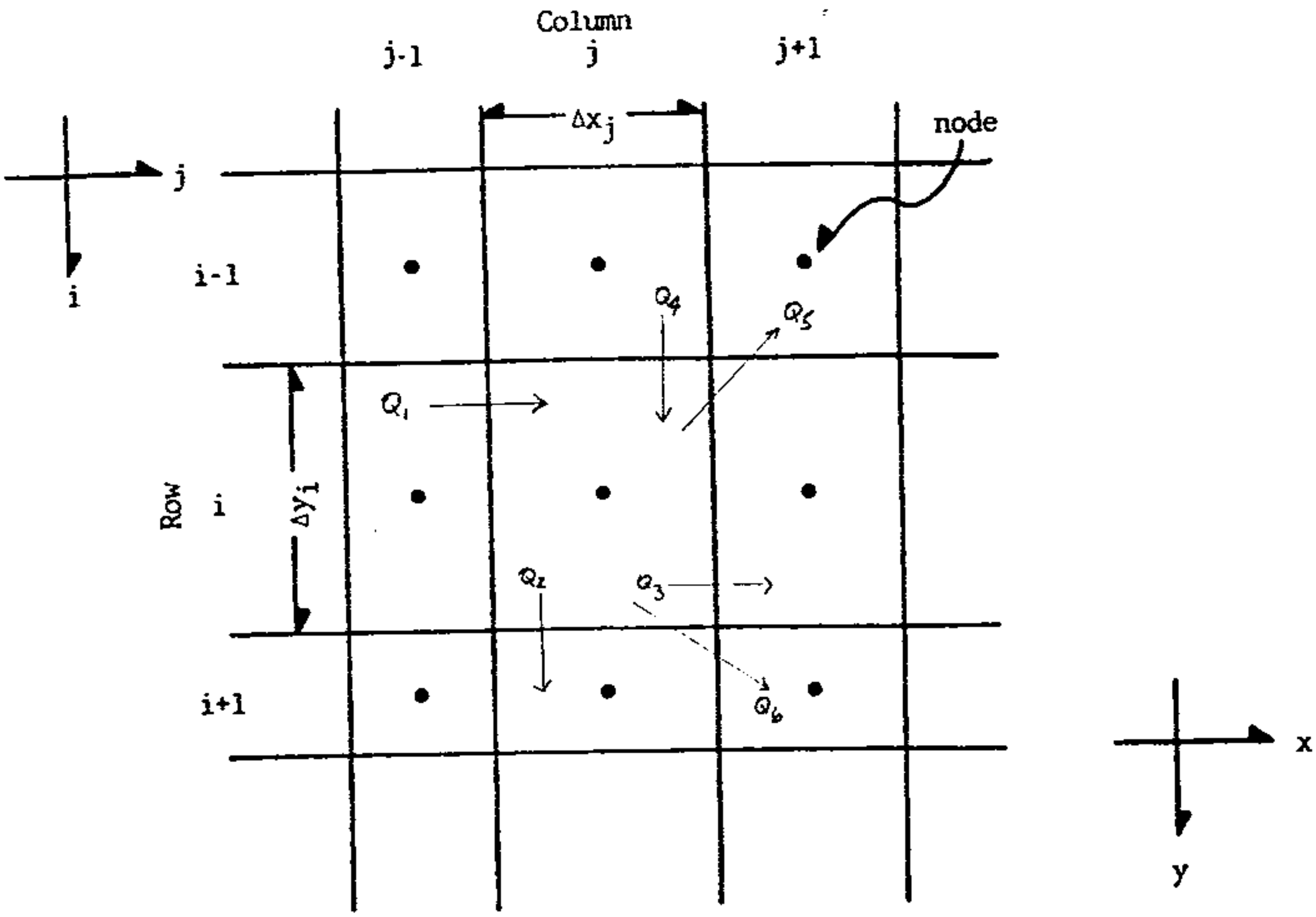
The numerical solution for Equation 1 can be obtained by applying the finite difference approach. The steps in applying the approach are as follows:

(a) a finite difference grid is superimposed upon a map showing the extent of the aquifer, thus allowing the finite difference grid to replace the continuous aquifer with an equivalent set of discrete elements;

(b) the governing partial differential equation is written in finite difference form for each of the discrete elements; and

(c) the resulting set of linear finite difference equations is then solved numerically for the head with the aid of a digital computer.

An example of a portion of a finite difference grid is shown in Figure 1. Each of the discrete elements is a cell, and the center of each cell is a node. Each cell is referenced by its row (i) and column (j) number.



Finite Difference Grid

Figure 1

Equation 1 may be approximated as

$$\frac{1}{\Delta x_j}\left(\left\{T_{i,j+\frac{1}{2}}\left(\frac{h_{i,j+1}-h_{i,j}}{\Delta x_{j+\frac{1}{2}}}\right)\right\}-\left\{T_{i,j-\frac{1}{2}}\left(\frac{h_{i,j}-h_{i,j-1}}{\Delta x_{j-\frac{1}{2}}}\right)\right\}\right)$$

$$+\frac{1}{\Delta y_i}\left(\left\{T_{i+\frac{1}{2},j}\left(\frac{h_{i+1,j}-h_{i,j}}{\Delta y_{i+\frac{1}{2}}}\right)\right\}-\left\{T_{i-\frac{1}{2},j}\left(\frac{h_{i,j}-h_{i-1,j}}{\Delta y_{i-\frac{1}{2}}}\right)\right\}\right)$$

$$=\frac{S_{i,j}}{\Delta t}\left(h_{i,j}-H_{i,j}\right)+W_{i,j} \qquad (2)$$

where

Δx_j = grid spacing in the x-direction for column j,

Δy_i = grid spacing in the y-direction for row i,

$T_{i,j+\frac{1}{2}}$ = transmissivity between node i,j and i,j+1,

$h_{i,j}$ = head at node i,j at end of time step,

$S_{i,j}$ = storage coefficient for cell i,j,

Δt = time step increment,

$H_{i,j}$ = head at node i,j at beginning of time step,

$W_{i,j}$ = net withdrawal per unit surface area for cell i,j, and

$\Delta x_{j+\frac{1}{2}}$ = distance between node i,j and node i,j+1

Multiplying Equation 2 by the area of cell i,j, $\Delta x_j \, \Delta y_i$, and rearranging terms results in

$$A_{i,j}\left(h_{i,j-1}-h_{i,j}\right)+B_{i,j}\left(h_{i,j+1}-h_{i,j}\right)$$

$$+C_{i,j}\left(h_{i-1,j}-h_{i,j}\right)+D_{i,j}\left(h_{i+1,j}-h_{i,j}\right)$$

$$=E_{i,j}\left(h_{i,j}-H_{i,j}\right)+Q_{i,j} \qquad (3)$$

where

$$A_{i,j}=\frac{TK_{i,j}\Delta x_{j-1}+TK_{i,j-1}\Delta x_j}{\Delta x_{j-1}+\Delta x_j}\;P_{i,j-1,1}\;\Delta y_i\;\frac{2}{\Delta x_{j-1}+\Delta x_j}$$

$$B_{i,j} = \frac{TK_{i,j+1}\Delta x_j + TK_{i,j}\Delta x_{j+1}}{\Delta x_j + \Delta x_{j+1}} \; P_{i,j,1} \; \Delta y_i \; \frac{2}{\Delta x_j + \Delta x_{j+1}} ,$$

$$C_{i,j} = \frac{TK_{i,j}\Delta y_{i-1} + TK_{i-1,j}\Delta y_i}{\Delta y_{i-1} + \Delta y_i} \; P_{i-1,j,2} \; \Delta x_j \; \frac{2}{\Delta y_{i-1} + \Delta y_i} ,$$

$$D_{i,j} = \frac{TK_{i+1,j}\Delta y_i + TK_{i,j}\Delta y_{i+1}}{\Delta y_i + \Delta y_{i+1}} \; P_{i,j,2} \; \Delta x_j \; \frac{2}{\Delta y_i + \Delta y_{i+1}} ,$$

$$E_{i,j} = \frac{S_{i,j}\Delta x_j \Delta y_i}{\Delta t} \text{ , and}$$

$$Q_{i,j} = W_{i,j}\Delta x_j \Delta y_i + \delta R_{i,j}(h_{i,j} - RD_{i,j})$$

where

$TK_{i,j}$ = saturated thickness for cell i,j.

δ = 1 for river cells, cells with leakage, and spring cells with $H_{i,j} > RD_{i,j}$

$= 0$ for spring cells with $H_{i,j} < RD_{i,j}$

$P_{i,j,1}$ = hydraulic conductivity between cells i,j and i,j+1

$P_{i,j,2}$ = hydraulic conductivity between cells i,j and i+1,j.

The first part of terms A,B,C, and D is the saturated thickness at the cell face. The last part is the distance separating two adjoining nodes.

The last part of term Q represents flow that is a function of the head for the cell. The function is linear with the term R equal to change in flow rate per unit change in head (slope). If head is greater than the reference head, water moves out of the cell and the opposite is true if head is less than the reference head. If the flow is to be from a spring, flow is allowed to only leave the cell. If flow from a river or lake is to be simulated, the reference head would be the water surface elevation. If spring flow is to be simulated, the reference head would be the elevation of the spring opening. For leakage from another aquifer, the reference head would be the head in that aquifer. The determination of the slope term (R) value depends upon the process being modeled. For leakage or flow from a river, the slope could be determined by assuming that the source of water (river or other aquifer) is separated from the modeled aquifer by a confining bed. The slope term could be evaluated as

$$R = PA_c/m$$

where

P = hydraulic conductivity of confining bed,

A_c = area of confining bed through which leakage takes place, and

m = thickness of confining bed

An equation of the same form as Equation 3 is written for each cell in the finite difference grid. This results in a large set of linear equations with the water levels (heads) at the end of the time step, h, as unknowns. An iterative alternating direction implicit procedure is used to solve the set of equations.

Solution Technique

The iterative alternating direction implicit (IADI) procedure involves reducing a large set of equations to several smaller sets of equations. One such small set of equations is generated by writing Equation 3 for each node in a column, assuming that the heads for the nodes on the adjacent columns are known. The unknowns in this set of equations are the heads for the nodes along the column. The heads for the nodes along adjoining columns are not considered unknowns. This set of equations is solved by Gauss elimination and the process repeats until each column is treated. The next step is to develop a set of equations along each row, assuming the heads for the nodes along adjoining rows are known. The set of equations for each row is solved, and the process repeats for each row in the finite difference grid.

Once the sets of equations for the columns and the sets of equations for the rows have been solved, one "iteration" has been completed. The iteration process is repeated until the procedure has converged. Upon convergence, the terms $h_{i,j}$ represent the heads at the end of the time step. These heads are used as the beginning heads for the following time step. For a more detailed discussion of the iterative alternating direction implicit procedure, see Peaceman and Rachford (1955) and Prickett and Lonnquist (1971).

The program uses a head predictor algorithm and iteration parameters to speed convergence. The original Prickett program incorporated a head predictor algorithm. This allows the solution procedure to have an improved estimate of the solution before the iterative procedure is started. Without a predictor algorithm, the beginning head values would be the estimates of the ending head values. During each major time step, the external stimuli, pumpage and recharge rate, are constant. It is only natural to assume that, based on constant external stimuli, water levels will continue to change in a consistent manner throughout the time period. The predictor algorithm uses this assumption in that as soon as a consistent pattern of change is established, this pattern is used to improve the initial estimates of the unknown heads. If an inconsistent pattern is established, no prediction is made.

The head prediction algorithm is used during each major time step. After two small time steps have been completed, the algorithm attempts to predict the heads at the end of the next time step. If the direction of the change in head for any one node is different from the direction of the change in head during a previous time step, the head change pattern is assumed to be inconsistent and no prediction is made for that node.

Iteration parameters are also utilized to aid convergence. Development of those parameters is beyond the scope of this manual, but a brief discussion may be in order. Equation 4 is a rearrangement of Equation 3 incorporating the normalized iteration parameter, G_p; and, as required in the first phase of the IADI procedure, assumes that the heads along the columns are unknowns.

$$Ah^{n+1}_{i,j-1} + (B+G_p)\, h^{n+1}_{i,j} + Ch^{n+1}_{i,j+1} = (D+G_p)h^{n}_{i,j} + Eh^{n}_{i-1,j} + Fh^{n}_{i+1,j} + G \qquad (4)$$

where

A,B,C,D,E,F,G = collection of terms in Equation 3.

$h^{n+1}_{i,j}$ = head of cell i,j at iteration number n+1, and

G_p = normalized iteration parameter.

Please note that when the solution converges, h^{n+1} becomes approximately equal to h^n, and the product of head times the normalized iteration parameter appears on each side of the equal sign therefore cancelling out.

The calculation of G_p was adopted from Trescott, 1976, and may be expressed as

$$G_p = \rho_p(A_{i,j} + B_{i,j} + C_{i,j} + D_{i,j}) \quad ,$$

$$\rho_p = \rho_{p-1}\xi \ , \quad p=2,3,\ldots,P \quad ,$$

$$\xi = \exp\ (\ln(1./\rho_1)/P-1) \quad ,$$

$$\rho_1 = \min\ (2,\ fx,\ fy) \quad ,$$

$$fx = \pi^2/(2m^2(1+(\frac{\Delta y}{\Delta x})^2)) \quad , \text{ and}$$

$$fy = \pi^2/(2n^2(1+(\frac{\Delta x}{\Delta y})^2))$$

where

ρ = iteration parameter,

p = iteration index which cycles from one to the number of iteration parameters to be used,

m = number of rows,

n = number of columns, and

P = number of iteration parameters

The value of p changes for each iteration and cycles from one to the number of parameters used. The iteration parameters are printed at the beginning of simulation.

For program GWSIM-IV, the solution procedure is assumed to have converged if the sum of the changes in head during an iteration is less than a specified input value, ERROR. As the convergence criterion is reduced (i.e., smaller values for ERROR) the finite difference solution will more closely approximate the theoretical solution. However, as the error criterion is reduced, the number of iterations required for convergence will increase. There is a point of diminishing returns where the increase in accuracy does not justify the increase in the number of iterations and the resulting increase of computer execution time. A few tests should be made with difference values for ERROR to determine the value that yields good results with few iterations. The program performs at least four iterations.

MASS TRANSPORT SIMULATION

Development of Finite Difference Equation

The partial differential equation describing mass transport and dispersion of dissolved, conservative solutes in a saturated porous medium may be written

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial C}{\partial x_i}\right) - \frac{\partial (V_i C)}{\partial x_i} - W_i \qquad (5)$$

where

C	is the mass concentration,	$(\frac{M}{L^3})$
D_{ij}	is the coefficient of hydrodynamic dispersion	(L^2/T)
V_i	is the velocity of flow,	(L/T)
W_i	is the mass flux from the aquifer	$(\frac{M}{L^3 T})$

$q_{i,j,2}$ = flow rate of solution from cell i,j to cell i+1,j , and

$C^i_{j-1,j}$ = concentration of solute in the flow from cell i,j-1 to cell i,j.

The flow rate representing the amount of mass taken from storage may be expressed as:

$$Q_5 = (V\emptyset_{i,j}\ C\emptyset_{i,j} - V_{i,j}\ C_{i,j})\ /\Delta t$$

where

$V\emptyset_{i,j}$ = volume of solution in cell i,j at the beginning of the time step,

$C\emptyset_{i,j}$ = concentration of solute in cell i,j at the beginning of the time step, and

$V_{i,j}$ = volume of solution in cell i,j at the end of the time increment.

The term Q_6 represents the mass flux leaving the cell thru the processes of pumpage, recharge, springflow, evapotranspiration, interformational leakage, or mixing with surface water. For the remainder of development, mass transfer out of a cell will be assumed to be positive and transfer into a cell will be assumed to be negative. The term Q_6 may be expressed as

$$Q_6 = C_{i,j}\ (Qp_{i,j} + Qo_{i,j}) - Cr_{i,j}\ (Qr_{i,j} + Qi_{i,j})$$

where

$Qp_{i,j}$ = rate of pumpage for cell i,j ,

$Qo_{i,j}$ = rate of water leaving cell i,j thru a process where flow is a function of head,

$Cr_{i,j}$ = concentration of solute for water entiring cell i,j ,

$Qr_{i,j}$ = rate of recharge for cell i,j , and

$Qi_{i,j}$ = rate of water entering cell i,j thru a process where flow is a function of head.

For each cell only one of the terms Qo and Qi will be non-zero. The term which is zero is determined by the relative value of h and RD. If h is greater than RD, Qi is zero and if h is less than RD, Qo is zero.

t	is time, and	(T)
x_i	is the i^{th} dimension (Konikow and Bredehoeft, 1974)	(L)

The term on the left represents the change in concentration over time. The first term on the right represents the mass transport by hydrodynamic dispersion and the second term represents mass transport by convection. Equation 5 is solved by applying the finite difference approach.

Figure 1 can be used to explain the development of the finite difference equation. The mass flow rate terms (Q_1, ..., Q_6) are arbitrarily assigned flow directions in Figure 1. Flows rate terms Q_1, Q_2, Q_3, and Q_4 represent cell-to-cell mass transfer. Q_5 represents the mass change in the cell over a unit interval of time. Q_6 is assumed to represent the mass leaving the cell.

The continuity of mass requires that the mass transport rate be equaled as follows

$$Q_1 + Q_4 = Q_2 + Q_3 + Q_5 + Q_6. \qquad (6)$$

The cell-to-cell mass transport includes mass transport by hydrodynamic dispersion and by convection. For the following equation development, it is assumed that both processes move mass in the same direction.

The terms Q_1, Q_2, Q_3, and Q_4 may be expressed as

$$Q_1 = THIK^{i}_{j-1,j}\ \Delta y_i\ D_{i,j-1,1}(C_{i,j-1} - C_{i,j})/\Delta x_{j-1/2} + |q_{i,j-1,1}|\,C^{i}_{j-1,j}\ ,$$

$$Q_2 = THIK^{j}_{i,i+1}\ \Delta x_j\ D_{i,j,2}(C_{i,j} - C_{i+1,j})/\Delta y_{i+1/2} + |q_{i,j,2}|\,C^{j}_{i-1,i}\ ,$$

$$Q_3 = THIK^{i}_{j,j+1}\ \Delta y_i D_{i,j,1}(C_{i,j} - C_{i,j+1})/\Delta x_{j+1/2} + |q_{i,j,1}|\,C^{i}_{j,j+1}\ , \text{ and}$$

$$Q_4 = THIK^{j}_{i-1,i}\ \Delta x_j\ D_{i-1,j,2}(C_{i-1,j} - C_{i,j})/\Delta y_{i-1/2} + |q_{i-1,j,2}|\,C^{j}_{i-1,i}$$

where

$THIK^{i}_{j-1,j}$ = aquifer thickness for the cell face between cell i,j-1 and cell i,j,

$D_{i,j,1}$ = coefficient of hydrodynamic dispersion between cell i,j and cell i,j+1,

$D_{i,j,2}$ = coefficient of hydrodynamic dispersion between cell i,j and cell i+1,j,

$C_{i,j}$ = concentration of solute in cell i,j ,

$q_{i,j,1}$ = flow rate of solution from cell i,j to cell i,j+1 , Substituting and rearranges equations 6 ~~10, 11, and 12 into Equation 9, and rearranging~~ we have:

$$THIK^{i}_{j-1,j} \Delta y_i \; D_{i,j-1,1}(C_{i,j-1} - C_{i,j})/\Delta x_{j-½} \quad + |q_{i,j-1,1}| \, C^{i}_{j-1,j}$$

$$+ \; THIK^{i}_{j,j+1} \; \Delta y_i \; D_{i,j,1}(C_{i,j+1} - C_{i,j})/\Delta x_{j+½} \quad + |q_{i,j,1}| \, C^{i}_{j,j+1}$$

$$+ \; THIK^{j}_{i-1,i} \; \Delta x_j \; D_{i-1,j,2}(C_{i-1,j} - C_{i,j})/\Delta y_{i-½} \quad + |q_{i-1,j,2}| \, C^{j}_{i-1,i}$$

$$+ \; THIK^{j}_{i,i+1} \; \Delta x_j \; D_{i,j,2}(C_{i+1,j} - C_{i,j})/\Delta y_{i+½} \quad + |q_{i,i,2}| \, C^{j}_{i,i+1}$$

$$= (V_{i,j} \; C_{i,j} - VØ_{i,j} \; CØ_{i,j})/\Delta t + Cr(Qr_{i,j} + Qi_{i,j}) - C_{i,j}(Qp_{i,j} + Qo_{i,j}) \qquad (7)$$

The method of determining the concentration of solute moving between cells is an important consideration. The basic assumption of finite difference approximations in that the value at the center of a cell is representative of the entire cell. The use of concentration $C_{i,j}$ for $C^{i}_{j,j+1}$ when the flow is from cell i,j to cell i,j+1 can result in too rapid a movement of mass. To help alleviate this potential problem, $C^{i}_{j,j+1}$ is formulated as

$$C^{i}_{j,j+1} = \beta(\alpha C_{i,j} + (1-\alpha)C_{i,j+1} + (1-\beta)(\alpha C_{i,j+1} + (1-\alpha)C_{i,j})$$

where

β = 1 if $q_{i,j,1}$ is positive (from cell i,j to cell i,j+1)

= 0 if $q_{i,j,1}$ is negative, and

α = averaging coefficient

The range of α is from 0.5 to 1.0. A value of 0.5 indicates that a numerical average of the two concentration is to be used. A value of 1.0 indicates that no averaging is to be used.

An equation of the same form as Equation 7, ~~with the substitution of Equation 14,~~ is written for each cell in the finite difference grid. This results in large set of linear equations with the concentrations at the end of the time period, $C_{i,j}$, as unknowns. An iterative alternating direction implicit procedure is used to solve the set of equations.

Solution Technique

The type of solution procedure used to solve the hydrologic finite difference equations (Equation 3) is used to solve the mass transport finite difference equations. The iterative alternation direction implicit (IADI) procedure involves reducing a large set of equations to several smaller sets of equations. One such small set of equations is generated by writing Equation (7) for each node in a column but assuming that the concentration for the nodes on the adjacent columns are known. The unknowns in this set of equations are the concentration for the nodes along the column. The concentration for the nodes along adjoining columns are not considered unknowns. This set of equations is solved by Gauss elimination and the process repeats until each column is treated. The next step is to develop a set of equations along each row, assuming the concentration for the nodes along adjoining rows are known. The set of equations for each row is solved and the process repeats for each row in the finite difference grid.

Once the sets of equations for the columns and the sets of equations for the rows have been solved, one "iteration" has been completed. The iteration process is repeated until the procedure has converged. Once convergence is accomplished, the terms $C_{i,j}$ represents the concentrations at the end of the time step. These concentrations are used as the beginning concentrations for the following time step.

For program GWSIM-IV, the solution procedure is assumed to have converged if the sum of the changes in concentration during an iteration is less than a specified input value, QERROR. As the convergence criterion is tightened (smaller values for QERROR) the finite difference solution will become a closer approximation to the theoretical solution. However, as the error criterion is tightened, the number of iterations required for convergence will increase. There is a point of diminishing returns where the increase in accuracy does not justify the increase in the number of iterations and the resulting increase of computer execution time. A few tests should be made with difference values for QERROR to determine the value which yields good results with few iterations.

FEATURES OF THE PROGRAM

In each of the following sections, an important phase of the GWSIM-IV program is explained.

Time Steps

Program GWSIM-IV was written to simulate water levels in an aquifer after uniform steps in time. For most regional modeling problems, these uniform time steps would represent yearly time steps. At the beginning of each of these major time steps, the program is designed to accept new values for pumpage and recharge rates. This provides the ability to change the external factors during a long term simulation period. Normally, only yearly values of these parameters are available.

Each of the major time steps is accomplished by completing a number of smaller time steps, called minor time steps. This allows better simulation of the aquifer's response to pumpage and recharge by reducing the shock of sudden changes in these external stimuli. The minor time steps may be non-uniform in size with the first steps small and the later steps large. The size of each time step may be increased over the previous time step, allowing an acceleration of the length of time steps. The program determines the length of the initial minor time step based on the length of the major time step, number of minor steps, and time step acceleration factor.

Boundary Conditions

The program GWSIM-IV is capable of simulating two types of boundary. The first type could be called a barrier or non-flow condition. No ground-water flow is allowed to cross this type of boundary. The exterior sides of the finite difference grid represent no-flow boundaries. Other barriers may be represented by the appropriate location of exterior cells (Flag = 3) which allow no ground-water flow.

The second type of boundary which may be simulated is a constant-head boundary. This type of boundary may be required to simulate a stream or lake. This may be accomplished by designating a cell to be type 0, constant head. Ground-water flow can occur with this type of cell but the water level for the node will not be calculated.

Type of Cell Declaration

Each cell in the finite difference grid must be assigned a type declaration. A cell's type declaration is based on the conditions existing at the node at the beginning of the simulation period. The entire cell is assumed to exhibit the same characteristics as does the node. These declarations are

1) FLAG = 0 for constant head/concentration,
2) FLAG = 1 for water table conditions,
3) FLAG = 2 for artesian conditions, and
4) FLAG = 3 for exterior nodes.

These declarations are used to indicate whether the node is active or if it is outside the ground-water system. A FLAG = 0 cell is assumed to be a constant-head cell. Ground water may enter or leave this type of cell, but a new water level will not be simulated. A FLAG = 1 cell is assumed to exhibit water table characteristics. That is, the flow area for ground-water movement

-and concentration

will vary as water table elevations change.

FLAG = 2 nodal declaration is used to identify cells for which flow area does not vary as water levels change. The FLAG = 3 nodal declaration is for cells which are to be considered exterior to the ground-water system and are therefore considered exterior cells.

The solution procedure of GWSIM-IV is programmed to ignore any exterior nodes. Since many ground-water formations are not rectangular in shape, a superimposed rectangular grid would contain cells which are not a portion of the ground-water system. It would be wasteful of computer time to compute a ground-water head elevation for these vacant cells. For this reason, the program only simulates heads for cells which are in the ground-water system and flagged FLAG = 1 or 2.

Program Options

GWSIM-IV was constructed to allow the user a large amount of versatility. The input, operation, and output are controlled by a series of options. The General Program Options are set at the beginning of the run, and the Time Step Options are set at the beginning of each major time step. The mass transport procedure also has a set of options. These Quality Options are read for each time step if mass transport is simulated.

An option is enabled if it is assigned a value greater than zero.

General Program Options

The following options may be set at the beginning of the simulation. See Data Set 3.

1 PRINT HYDROGRAPHS

Enabling this option causes the reading of Data Set 4 and results in the printing of a hydrograph for specified cells. See description of subroutine HYDRO.

2 PRINT CROSS SECTIONS

Enabling this option causes the reading of Data Set 5 and allows printing a profile view of water levels along columns and/or rows. See descriptions of Time Step Option 23 and subroutine XSECT.

3 READ CONSTANT GRID SPACINGS

Enabling this option causes the program to read a constant grid spacing for each direction; thus individual grid spacings will not be read. See Data Set 6.

4 WRITE GRID SPACINGS

Enabling this option results in a listing of the grid spacings.

5 READ DEFAULT PHYSICAL DATA

Enabling this option causes the program to read a set of physical data which will be assigned to each cell in the system. Values for each cell will not be read. See Data Set 7.

6 PHYSICAL DATA CORRECTIONS

Enabling this option allows replacement of physical data values for specific nodes. Data Set 8 is read.

7 ADJUST PARAMETERS

Enabling this option allows modification of storage coefficient and hydraulic conductivity values. Maps of the parameters may also be printed.

If the option equals 1, no maps are printed; if it equals 2, hydraulic and transmissivity conductivity maps in both directions are printed; if it equals 3, a stora coefficient map is printed; and if it equals 4, all maps are printed. Data Set 9 is read. See description of subroutine CALIB.

8 WRITE PHYSICAL DATA

Enabling this option results in a cell-by-cell listing of physical data.

9 PLOT INITIAL WATER LEVELS

Enabling this option results in the printing of a map indicating the initial water levels.

10 LIST AND PLOT INITIAL SATURATED THICKNESS

If this option is set equal to 1 or 2, a listing of the initial saturated thickness for each cell is printed. If this option equals 2 or 3, a map of initial saturated thicknesses is produced. Also listed with the map are the volumes of water in storage and corresponding surface areas by range of saturated thickness.

11 READ LEAKAGE TERMS ASSIGNMENT

Enabling this option causes the program to read leakage terms which are to be assigned to some or all cells in the grid. Data Set 10 is read.

12 READ LEAKAGE TERMS ADJUSTMENT

Enabling this option causes the program to read adjustment factors which are applied to leakage terms for some or all cells in the grid. Data Set 12 is read.

13 WRITE LEAKAGE TERMS

Enabling this option results in a cell-by-cell listing of the leakage terms (reference heads and slopes).

14 CALCULATE STEADY-STATE HEADS

Enabling this option results in the calculation of steady-state heads. The head-predictor algorithm is not used.

15 COMPUTE MASS TRANSPORT

Enabling this option causes the program to perform mass-transport modeling.

Time Step Options

The following options may be set for each major time step. See Data Set 13.

1 CHANGE TIME STEP PARAMETERS

Enabling this option changes the parameters controlling time step lengths for this major time step. The length of this major time step, number of minor time steps, and time step acceleration factor are changed. The original values of these parameters are reset at the end of this major time step. See Data Set 13.

2 READ PUMPAGE FOR EACH CELL

Enabling this option causes the program to read a pumpage rate for each cell in the system. Data Set 14 is read.

3 READ PUMPAGE BY BLOCK

Enabling this option causes the program to read a pumpage rate which is to be assigned to all cells in a specified block or region of the model grid. Data Set 15 is read.

4 PUMPAGE ADJUSTMENTS

Enabling this option causes the program to read an adjustment factor which multiplies the pumpage rate for all cells in a specified block of the model grid. Data Set 16 is read.

5 READ RECHARGE FOR EACH CELL

Enabling this option causes the program to read a recharge rate for each cell in the system. Data Set 17 is read.

6 READ RECHARGE BY BLOCK

Enabling this option causes the program to read a recharge rate which is to be assigned to all cells in a specified block of the grid. Data Set 18 is read.

7 RECHARGE ADJUSTMENTS

Enabling this option causes the program to read an adjustment factor which multiplies the recharge rate for all cells in a specified block of the grid. Data Set 19 is read.

8 NOT USED

9 STORE PUMPAGE AND RECHARGE RATES

Enabling this option causes the pumpage and recharge rates to be written on unit IN2. These values may be read for a later time step by then enabling Time Step Option 10.

10 RETRIEVE PUMPAGE AND RECHARGE RATES

Enabling this option causes the pumpage and recharge rates to be read from unit IN2. These values must have been stored for a previous time step. This option should not be enabled for the first time step.

11 LIST PUMPAGE AND RECHARGE RATES

Enabling this option causes the printing of the pumpage rates and the recharge rates for each cell.

12 PLOT FLOWS - MINOR

Enabling this option causes the printing of maps indicating groundwater flows at the end of each minor time step during this major step. See discussion of subroutine FLUX.

13 LIST HEADS - MINOR

Enabling this option causes the listing of heads at the end of each minor time step during this major time step.

14 SAVE HEADS

Enabling this option causes the heads at the end of this major time step to be written on unit OUT1.

15 SAVE PHYSICAL DATA

Enabling this option causes the nodal type, bottom of aquifer elevation, head at end of this major time step, hydraulic conductivity, and storage coefficient to be written on unit OUT1. A format card is also written so that these data could be used to re-start the model. A new Data Set 7 is produced.

16 LIST HEADS - MAJOR

Enabling this option causes the listing of heads at the end of this major time step.

17 PLOT FLOWS - MAJOR

Enabling this option causes the printing of maps indicating groundwater flows at the end of this major time step. See discussion of subroutine FLUX.

18 LIST HEAD CHANGES DURING THIS STEP

Enabling this option causes the listing of head changes occurring during this major time step.

19 PLOT HEAD CHANGES DURING THIS STEP

Enabling this option causes the printing of a map indicating head changes occurring during this major time step. See discussion of subroutine PLOTS.

20 LIST HEAD CHANGES THROUGH THIS STEP

Enabling this option causes the listing of head changes occurring from beginning of simulation through this major time step.

21 PLOT HEAD CHANGES THROUGH THIS STEP

Enabling this option causes the printing of a map indicating head changes occurring from beginning of simulation through this major time step.

22 COMPARE MEASURED HEADS

Enabling this option causes comparison of simulated heads with measured heads. A listing of the simulated head, measured head, and simulation error (simulated minus measured) for all cells is printed. A map of simulation errors is also printed. See discussion of subroutine PLOTS. Data Set 22 is read.

23 PLOT CROSS SECTIONS

Enabling this option causes the printing of water-level profiles. General Program Option 2 must have been enabled and Data Set 5 read. See discussion of subroutine XSECT.

24 READ CONSTANT HEADS

Enabling this option allows input of head for constant-head cells (FLAG = 0) at the end of this major time step. Also, a change in head may be read. Heads at the end of minor time steps are interpreted, always maintaining a minimum saturated thickness of 0.1. The option value causes the following to be read:

Option Value	Action
0	Nothing
1	Heads for all cells
2	Head changes for all cells
3	Heads for block of cells
4	Head changes for block of cells

Data Set 20 is read.

25 LIST AND PLOT SATURATED THICKNESS

If this option is set equal to 1 or 2, a listing of saturated thicknesses at the end of this major time step is printed. If this option equals 2 or 3, a printer map of saturated thickness is produced. See discussion of subroutine PLOTH.

26 PLOT HEADS

Enabling this option causes the printing of a map indicting the elevation of water levels at the end of this major time step. See discussion of subroutine PLOTH.

27 READ LIMITS FOR STATISTICAL BLOCKS

Enabling this option causes the program to compute statistical data for blocks of the grid. The data are printed if plots of water-level change or simulation error are printed, see Time Step Options 19, 21, and 22. Data Set 21 is read. See discussion of Subroutine PLOTS.

Mass Transport Options

The following options may be set for each time step. See Data Set 24.

Option 1 READ DISPERSIVITY COEFFICIENTS FOR EACH CELL FACE

The enabling of this option allows input of dispersivity coefficients for each cell face. Data Set 25 is read

2 WRITE DISPERSIVITY COEFFICIENTS FOR EACH CELL FACE

The enabling of this option causes the listing of the dispersivity coefficients by cell. Values are listed for dispersion coefficient in the X-direction followed by a listing of coefficients in the y-direction.

3 READ DISPERSIVITY COEFFICIENTS BY BLOCK

Enabling this option causes the program to read coefficients which are to be assigned to all cells in a specified block of the grid. Data Set 26 is read.

4 READ DISPERSIVITY COEFFICIENT ADJUSTMENTS

Enabling this option causes the program to read adjustment factors which multiplies the coefficients for all cells in a specified block of the grid. Data Set 27 is read.

5 READ RECHARGE QUALITIES FOR EACH CELL

The enabling of this option causes the input of a recharge concentration for each cell in the system. Data Set 28 is read

6 WRITE RECHARGE QUALITIES FOR EACH CELL

The enabling of this option causes the echo printing of input values read by option 5.

7 READ RECHARGE QUALITIES BY BLOCK

Enabling this option causes the program to read quality values which are to be assigned to all cells in a specified region of the grid. Data Set 29 is read.

8 READ RECHARGE QUALITIES ADJUSTMENTS

Enabling this option causes the program to read adjustment factors which multiplies the quality values for all cells in a specified region of the grid. Data Set 30 is read.

9 LIST CONCENTRATIONS AT END OF STEP

The enabling of the option causes a listing of the cell concentrations at the end of the major time step.

10 PLOT CONCENTRATIONS AT END OF STEP

The enabling of this option results in a printer map of concentrations at end of the major time step. See discussion of subroutine QPLOTS.

11 LIST CHANGES IN CONCENTRATIONS DURING THIS STEP

The enabling of this option results in a listing of the changes in concentration during the major time step.

12 PLOT CONCENTRATIONS CHANGES DURING THIS STEP

The enabling of this option causes a printer map of concentration changes during this major time step. See discussion of subroutine QPLOTS.

13 PLOT CONCENTRATION CHANGES THROUGH THIS STEP.

The enabling of this option causes a printer map of the concentration changes which occurred from the beginning of simulation through the completion of this major time step. See discussion of subroutine QPLOTS.

14 READ MEASURED VALUES OF CONCENTRATIONS AND PLOT ERROR MAP

The enabling of this option causes the reading of measured concentrations for each cell. See Data Set 37. A listing of simulated concentration, measured concentration, and simulation error (simulated minus measured) for all cells is printed. A printer plot of simulation errors in concentration is produced. See discussion of subroutine QPLOTS.

15 READ IN INITIAL CONCENTRATIONS FOR EACH CELL

The enabling of this option allows input of concentration at the beginning of the simulation for each cell. See Data Set 31.

16 WRITE INITIAL CONCENTRATIONS FOR EACH CELL

The enabling of this option causes an echo print of the concentrations read by option 15.

17 READ INITIAL CONCENTRATIONS BY BLOCK

Enabling this option causes the program to read quality values which are to be assigned to all cells in a specified region of the grid. Data Set 32 is read.

18 INITIAL CONCENTRATION ADJUSTMENTS

Enabling this option causes the program to read adjustment factors which multiplies the initial concentrations for all cells in a specified region of the grid. Data Set 33 is read.

19 LIST CONCENTRATIONS AT END OF SMALL TIME STEP

The enabling of this option causes the printing of a listing of the concentrations for each cell to be listed at the completion of each minor time step.

20 LIST CHANGES IN CONCENTRATION THROUGH THIS STEP

The enabling of this option causes a printing of a listing of concentration changes which occurred from the beginning of simulation through the completion of this major time step.

21 READ POROSITY FOR EACH CELL

The enabling of this option allows input of porosity values for each cell. Data Set 34 is read.

22 WRITE POROSITY FOR EACH CELL

The enabling of this option causes an echo print of the porosity values read by Option 21.

23 READ POROSITY BY BLOCK

Enabling this option causes the program to read porosity values which are to be assigned to all cells in a specified region of the grid. Data Set 35 is read.

24 READ POROSITY ADJUSTMENTS

Enabling this option causes the program to read adjustment factors which multiplies the porosity values for all cells in a specified region of the grid. Data Set 36 is read.

Water-Table Condition Adjustment

The basic solution technique is designed to solve a set of equations based on Equation 3. For water-table conditions, transmissivity is a function of the water-table elevation and is not a constant as assumed in the development of Equation 3. However, Equation 3 may be used if the changes in head and the size of the time step are such that the transmissivity may be assumed to be constant during the time step. For GWSIM-IV, it is assumed that during each minor time step, the values for transmissivity are constant. The transmissivity for each cell equals the hydraulic conductivity times the saturated thickness. Saturated thickness equals the distance separating the water table and the base of aquifer for water table cells. For cells declared to be constant head or artesian, the input value of saturated thickness is used. The minimum saturated thickness is 0.1.

Units

The program operates using units of length and time. The unit of measure of length may be from either the English or metric system, as long as a consistent set of units is used. The pumpage, recharge, and hydraulic conductivity values are multiplied by conversion factors so that internally, all items measured in length units will be expressed in the same units.

The internal unit of time is days. The pumpage, recharge, and hydraulic conductivity conversion factors may be used to convert input rates so they are expressed in days.

The pumpage and recharge rates are input as volumes per major time step. For example, a cell from which 100 acre-feet of water were pumped annually could be assigned a pumpage value of 100 if the major time step was 365 days long. If the major step was two years in length, the input value would be 200. Assuming a major time step length of one year and the program interval length unit in feet, the factor to convert acre-feet per year to cubic feet per day is 119.34 (43,560 cubic feet per acre-foot/365 days per year).

The conversion factor for hydraulic conductivity converts from the external units into units of length per day. The factor to convert gallons per day per square foot to feet per day is 0.13369 (1/7.48 gallons per cubic foot).

The program is designed to allow any units to be used for quality concentration so long as they are consistent during a model application. The units could be milligrams per liter, parts per thousand, or any other measure. This allows the user to select the units most advantageous to the particular model application.

Temporary Storage of Data

To reduce storage requirements, GWSIM-IV only keeps in core the data

necessary for each phase of the simulation. [illegible] at the beginning of simula-tion and at the beginning and end of each major time step are stored on tape until needed for output routines.
If mass transport is simulated, some hydraulic data are stored and the arrays are used during the mass transport simulations. The hydraulic data are retrieved once the transport simulation is completed. The mass transport parameters are likewise stored on tape and read as needed.

Change of Cell Type

The program is structured so that a node can change its type declaration after each minor time step based on its relative value of head. If the head for an artesian node drops below the elevation of the top of aquifer, the node declaration changes to water table (FLAG = 1). The storage coefficient term is multiplied by STRFCT, the ratio of water table storage coefficient to artesian storage coefficient. (See Data Set 2.) If the head for a water table cell is larger than the top of aquifer, the cell's type declaration is changed to artesian and storage coefficient is divided by STRFCT.

STORAGE ADJUSTMENT

The storage term requires special treatment for nodes that convert from water-table to artesian conditions, or vice versa, during a time step. The storage adjustment is made during the solution procedure. If the head calculated during the last iteration indicates that a node has changed condition, the storage term is expressed as follows (subscripts are omitted)

Water-table to artesian

$$\left(S_a\ (h^n - TOP) + S_w\ (TOP - H)\right)\frac{\Delta x \Delta y}{\Delta t}$$

Artesian to water-table

$$\left(S_w\ (h^n - TOP) + S_a\ (TOP - H)\right)\frac{\Delta x \Delta y}{\Delta t}$$

where

S_a = storage coefficient for artesian condition,

S_w = specific yield for water-table condition, and

TOP = elevation of top of aquifer.

In the program, the storage term is re-arranged using the input value that equals the ratio of specific yield to storage coefficient.

Calculation

The calculation of dispersion coefficients is based on a procedure similar to that discussed by Konikow and Bredehoeft (1978). Hydrodynamic dispersion may be considered as two separate processes. One is mechanical dispersion which depends on the direction and rate of fluid flow and on the nature of the porous material through which the flow occurs. The second is diffusion that depends on movement of molecules and atoms. In developing this program, it was assumed that diffusion has negligible input to hydrodynamic dispersion.

It was further assumed that the coefficient of hydrodynamic dispersion can be calculated as a function of flow velocities and the dispersivity of the aquifer

Aquifer dispersivity is characterized by two constants; longitudinal and transverse dispersivities. Longitudinal dispersivity applies in the direction of flow and transverse, at right angle to flow. For this program, the coefficients of hydrodynamic dispersion may be expressed as:

$$D_{i,j,1} = \frac{DL_{i,j}(q_{i,j,1})^2 + DT_{i,j}(q_{i,j,2})^2}{((q_{i,j,1})^2 + (q_{i,j,2})^2)^{\frac{1}{2}}}$$

$$D_{i,j,2} = \frac{DL_{i,j}(q_{i,j,2})^2 + DT_{i,j}(q_{i,j,1})^2}{((q_{i,j,1})^2 + (q_{i,j,2})^2)^{\frac{1}{2}}}$$

where

$DL_{i,j}$ = Longitudinal dispersivity coefficient for cell i,j, ~~and~~

$DT_{i,j}$ = Transverse dispersivity coefficient for cell i,j,

$q_{i,j,1}$ = Rate of flow thru pores between cells i,j and i,j+1, and

$q_{i,j,2}$ = Rate of flow thru pores between cells i,j and i+1,j.

INPUT

Program GWSIM-IV was written to allow the user great flexibility in the construction of a data deck. The user has the option of choosing formats, the method of assigning the physical parameters of the system, and the form of external stimuli. The input and output may be tailored to fit the user's needs.

The items required for input to hydrologic modeling are as follows:

1. finite difference grid spacings,
2. nodal type,
3. land surface elevation,
4. top of aquifer elevation,
5. base of aquifer elevation,
6. saturated thickness,
7. initial head (water-level elevation),
8. hydraulic conductivity,
9. storage coefficient,
10. leakage terms, and
11. pumpage and recharge rates

Additional input items required for mass transport modeling are as follows

1) dispersivity coefficient,

2) porosity,

3) inflow (recharge, inleakage, injection) concentration, and

4) initial concentrations.

Input Unit Numbers

GWSIM-I V uses one unit number variable for data input and six unit number variables for internal storage of data. Unit variable 'IN' is used to read all user supplied data and is set equal to 5. Unit Variable 'IN1' is used to store the initial water table elevation and is set equal to 11. Variable 'IN2' is used to store pumpage and recharge rates and is set equal to 12. Unit 'IN3' is used to store water table elevations for the hydrograph routine and is set equal to 13. Unit variable 'IN4' is used to store water table elevations at the beginning of a major time step and is set equal to 14.

General Description

All user supplied data are read using formatted read statement. Most of the data sets are read using variable, or object time formats. These formats are set equal to the default format shown in this manual. If the user desires, the program can be instructed to read a format that will override the default format. This override is accomplished by adding 5 to the value of the option that controls the reading of the data set. If this override is desired, a card containing the new format becomes the first card of the data set.

As an example of how this could occur, assume that the user desires to read default grid spacings by the format (2F10.0) instead of by the default format (2F5.0). To accomplish this, General Program Option 3 is set equal to 6 (1-read default spacings plus 5-read another format) and the format card with (2F10.0) shown is placed immediately before the card containing the grid spacings. The spacing in the X-direction is punched into the first ten columns and the Y-direction, punched into the second ten columns.

Integer input parameters must be punched, right justified, and without a decimal point.

DATA SET SEQUENCE

Figure 2

Data Set Number	Title	Is Data Set Read?*
1	Title	Yes
2	Parameters	Yes
3	General program options	Yes
4	Hydrograph specifications	If GP Option 1 GTO
5	Cross section specifications	If GP Option 2 GTO
6	Grid spacings	Yes
7	Physical data	Yes
8	Physical data corrections	If GP Option 6 GTO
9	Physical data adjustments	If GP Option 7 GTO
10	Leakage term assignment	IF GP Option 11 GTO
11	Leakage term adjustment	IF GP Option 12 GTO
12	Spring/River data	If [illegible] cells used
	The following data sets may be read for each major time step	
13	Time step options	Yes
14	Pumpage for all cells	If TS Option 2 GTO
15	Pumpage by block	If TS Option 3 GTO
16	Pumpage adjustments	If TS Option 4 GTO
17	Recharge for all cells	If TS Option 5 GTO
18	Recharge by block	If TS Option 6 GTO
19	Recharge adjustments	If TS Option 7 GTO
20	Heads for constant head cells	If TS Option 24 GTO
21	Limits for statistical blocks	If TS Option 27 GTO
22	Measured heads	If TS Option 22 GTO
23	Mass transport title	If GP Option 15 GTO
24	Mass transport options	If GP Option 15 GTO
25	Dispersion coefficients for all cells	If GP Option 15 GTO and MT Option 1 GTO
26	Dispersion coefficients by block	If GP Option 15 GTO and MT Option 3 GTO
27	Dispersion coefficients adjustments	If GP Option 15 GTO and MT Option 4 GTO
28	Recharge quality for all cells	If GP Option 15 GTO and MT Option 5 GTO
29	Recharge quality by block	If GP Option 15 GTO and MT Option 7 GTO
30	Recharge quality adjustments	If GP Option 15 GTO and MT Option 8 GTO
31	Initial concentrations for all cells	If GP Option 15 GTO and MT Option 14 GTO
32	Initial concentrations by block	If GP Option 15 GTO and MT Option 17 GTO
33	Initial concentration adjustments	If GP Option 15 GTO and MT Option 18 GTO
34	Porosity for all cells	IF GP Option 15 GTO and MT Option 21 GTO
35	Porosity by block	IF GP Option 15 GTO and MT Option 23 GTO
36	Porosity adjustments	IF GP Option 15 GTO and MT Option 24 GTO
37	Measured concentrations	IF GP Option 15 GTO and MT Option 14 GTO

*

GP Option = General Program Option (Data Set 3)
TS Option = Time Step Option (Data Set 13)
MT Option = Mass Transport Option (Data Set 24)
GTO = greater than zero

Data Set Descriptions

The sequence of the data sets is shown in Figure 2. Data Sets 1 through 12 may be read only once, whereas the remainder of the sets may be read for each major time step. Many of the data sets are read only if certain option is enabled.

Data Sets 23 through 34 may be read only if mass transport is modeled (General Program Option 15 enabled).

Data Set 1 - Title

This data set contains one card for input of a title statement. The title should be centered on the card.

Data Set 2 - Parameters

Columns	Format	Description
Card One		
1-5	I5	Number of major time steps
6-10	I5	Number of minor time steps
11-15	I5	Number of rows in grid
16-20	I5	Number of columns in grid
21-25	I5	Number of iteration parameters
26-30	I5	Number of spring or river cells

Card Two		
1-10	F10.0	Length of major time step (days)
11-20	F10.0	Convergence criterion
21-30	F10.0	Time acceleration factor
31-40	F10.0	Units conversion factor for pumpage and recharge
43-46	A6	Label to indicate pumpage and recharge units
49-52	A6	Label to indicate length units
53-64	2A6	Label to indicate units for ground-water flow maps
Card Three		
1-10	F10.0	Units conversion factor for hydraulic conductivity
11-20	F10.0	Units conversion factors for ground-water flow maps
21-30	F10.0	Ratio of water table to artesian storage coefficient (STRFCT)
31-40	F10.0	Scaling factor for plotting head changes, heads, saturated thicknesses, and cross-sections.

The number of iteration parameters should be between 3 and 7. A suggested value is 4. The convergence criterion is a function of the problem, and various values should be used until an appropriate value is determined; a suggested value is 1.0. A suggested time acceleration factor is 1.20, which allows for 20 percent growth in minor time step length.

The units conversion factor for pumpage and recharge converts the input rates into the internal units, cubic length per day. For example, the factor to convert acre-feet per year to cubic feet per day is 119.34, and the corresponding label could be 'AC-FT.' If the input water table units are in feet, the corresponding length label could be 'FEET.' The units conversion factor for hydraulic conductivity converts the input units into the internal units of length per day. For example, the factor to convert gallons per day per square foot to feet per day is 0.13369. If Time Step Options 12 or 17 are enabled, maps will be produced indicating ground-water flow. The internal units for flow are cubic length per day and the ground-water flow maps units conversion factor converts this rate to a more meaningful value, remembering that only 3 digits may be printed. If the length unit is feet, a factor value of 0.00008379 will result in rates printed in 100's of acre-feet per year. An appropriate label would be '100's AF/YR.'

The scaling factor for plotting has units of length per inch. For example, if the desired output is to be plotted so that each inch on the plot equals 1,000 feet, the input value should be 1000. If zero is read, the program does no scaling. If any negative number is read, the program calculates the scale factor to provide the most detail possible. See discussion of subroutine PLOTH.

Data Set 3 - General program options

This one-card data set contains the General Program Options. The value of the option is punched into the column number corresponding to the option number.

Data Set 4 - Hydrograph Specifications

This data set is required if General Program Option 1 is enabled. Up to 25 cells may be so identified.

Columns	Format	Description
Card One		
1-3	I3	Number of cells for which hydrographs are to be printed
4-6	I3	Row number of first identified cell
7-9	I3	Column number of first identified cell
10-12	I3	Row number of second identified cell

The sequence continues through the twelfth identified cell.

Columns	Format	Description
76-78	I3	Row number of thirteenth identified cell
Card Two		
1-3	I3	Column number of thirteenth identified cell
4-6	I3	Row number of fourteenth identified cell

The sequence continues until

Columns	Format	Description
72-75	I3	Column number of twenty-fifth identified cell.

The second card is required if more than twelve hydrographs are requested.

Data Set 5 - Cross section specifications

This two card data set is requried if General Program Option 2 is enabled. The first card indicates the number of and the corresponding column numbers for the columns for which profiles are requested. The second card contains similar data for rows. Up to 25 rows and 25 columns may be used.

Column	Format	Description
Card One		
1-3	I3	Number of columns for which cross sections are requested
4-6	I3	First column to be printed
7-9	I3	Second column to be printed

The sequence continues through the last column

75-78	I3	Twenty-fifth column to be printed.
Card Two		
1-3	I3	Number of rows for which cross sections are requested
4-6	I3	First row to be printed
7-9	I3	Second row to be printed

The sequence continues through the last row

75-78	I3	Twenty-fifth row to be printed.

Data Set 6 - Grid spacings

This data set contains the grid spacings. If General Program Option 3 is enabled (equal to 1 or 6 if format is to be read), constant spacings are read as follows:

Columns	Format	Description
1-5	F5.0	Grid spacing in X-direction (between columns)
6-10	F5.0	Grid spacing in Y-direction (between rows)

The grid spacings in the X-direction are read 15 values per card, with 5 spaces per value. The grid spacings in the Y-direction are read similarly.

The format may be changed by adding 5 to General Program Option 3.

The unit for grid spacing is length.

Data Set 7 - Physical data

This data set contains the data necessary to describe the system. tem.

If data are to be read for each cell, a card for each cell is read. Data are read by row, with data for all cells (beginning with column 1) in row 1 read first.

If a set default values are to be assigned to all cells in the grid, General Program Option 5 is enabled and only one card is read.

Columns*	Format*	Description
5	I1	Nodal type declaration
6-10	F5.0	Land surface elevation
11-15	F5.0	Top of aquifer elevation
16-20	F5.0	Base of aquifer elevation
21-25	F5.0	Saturated thickness
26-30	F5.0	Initial head (water level)
31-35	F5.0	Hydraulic conductivity in X-direction
36-40	F5.0	Hydraulic conductivity in Y-direction
41-45	F5.0	Storage coefficient**

*Another format may be used by adding 5 to General Program Option 5 and placing a format card at the first of this data set.

**If the nodal type declaration is for artesian condition (FLAG=2), the storage coefficient must be multiplied by 1,000,000 prior to coding.

The unit for the base of aquifer elevation and initial water level is length and it must be the same as that used for grid spacings.

Data Set 8 - Physical data corrections

This data set contains corrections to the physical data and is read if General Program Option 6 is enabled. The corrections are applied to all cells in a specified region of the grid.

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
10-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-25	I5	Nodal type declaration
26-30	F5.0	Land surface elevation
31-35	F5.0	Top of aquifer elevation
36-40	F5.0	Base of aquifer elevation
41-45	F5.0	Saturated thickness
46-50	F5.0	Initial head (water level)
51-55	F5.0	Hydraulic conductivity in X-direction
56-60	F5.0	Hydraulic conductivity in Y-direction
61-65	F5.0	Storage coefficient**

*Another format may be used by adding 5 to General Program Option 6 and placing a format card at the first of this data set.

**If the nodal declaration is for artesian condition (FLAG = 2), the storage coefficient must be multiplied by 1,000,000 prior to coding.

The last card must be blank.

Data Set 9 - Physical data adjustments

This data set contains factors to adjust the initial values of hydraulic conductivity and storage coefficient and is read if General Program Option 7 is enabled. One data card is required for each adjustment, and each adjustment is applied to a specified section of the grid. If the adjustment factor is non-negative, the present value of the parameter is multiplied by the value and adjustments are cumulative. If the value is negative, the absolute value of the adjustment factor is assigned to all cells in the grid section.

If the parameter identifier is a negative one or two, new hydraulic conductivities are calculated by dividing the adjustment value by saturated thickness. Thus, the adjustment value becomes a transmissivity value.

Columns	Format	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-25	I5	Parameter identifier (1 or -1 for hydraulic conductivity i x-direction, 2 or -2 for hydraulic conductivity in y-direction, anc 3 for storage coefficient).
26-35	F10.0	Adjustment value

The last card must be blank.

Data Set 10 - Leakage term assignment

this data set contains leakage terms to be assigned to some or all cells in the grid and is read if General Program Option 11 is enabled. Terms may be read only for all cells (Option 11 equal 1), only for block of cells (Option 11 equal 3), or all cells followed by replacement by blocks (Option 11 equal 2).

For all cells (group one) (Option 11 = 1 or 2):

These cards are read only if values for all cells are to be read. The values are read as a pair of values for each cell, with the reference head (RD) first and slope (R) second.

The values are read a row at a time with 5 pairs of values on each card. The first 10 columns are skipped, followed by 10 fields, each 7 columns wide. The first pair of values for each row would be punched into columns 11 through 24.

Another format may be used by adding 5 to General Program Option 11 and placing a format card at the first of this card group.

The units of reference head (RD) are length and should agree with units of Data

Set 7. The units of slope (R) are volume per major time step per unit of length and they are converted to cubic length per day per length by the conversion factor in Data Set 2.

The slope values may be read on a per unit area (per day) basis; i.e., feet per day per foot instead of acre-feet per year per foot. This is accomplished by placing a negative sign before the slope values, R, read in this data set.

Block of cells (group two) (Option 11 = 2 or 3):

These cards are read only if values for cells in a specified region of the grid are to be assigned.

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-30	F10.0	Reference head
31-40	F10.0	Slope

*Another format may be used by adding 5 to General Program Option 11 and placing a format card at the first of this card group.

The last card must be blank.

Only cards in group one are read if Option 11 equal 1 and only cards in group two are read if the option equals 3. If the option equals 2, group one cards are followed by group two cards.

Data Set 11 - Leakage terms adjustment

This data set contains adjustment factors which will multiply the leakage terms for all or some cells in the grid and is read if Time Step Option 12 is enabled. These data are read the same way as data in Data Set 10 are read except that instead of the input values being assigned to the cells, the input values multiply the leakage terms. Cards for group one are read if Option 12=1, group two if Option 12=3, and group one are followed by group two if the option equals 2.

Data Set 12 - Spring/river cell data

This data set contains row and column numbers, reference head, and slope terms for cells declared to be springs or river cells. The data set is read if spring or river cells are to simulated as indicated on card one of Data Set 2. For a spring cell, flow will be from a cell as long as the head for the cell is larger than the reference head. There is no flow if the head is less than the reference head. For a river cell, flow will be out of the cell is head for the cell is greater than the reference head and will be into the cell if the head is less than the reference head. At the end of each major time step, total flow volume is printed for each spring or river cell.

A river cell is designated by coding the slope term as a negative number.

Columns	Format	Description
1-5	I5	Row number for cell
6-10	I5	Column number for cell
11-20	F10.0	Reference head
21-30	F10.0	Slope

Data Set 13 - Time step options

This one-card data set contains the Time Step Options plus the parameters needed to adjust time step size (see Time Step Option 1) and a comment field.

Columns	Format	Description
1	I1	Value for option 1
2	I1	Value for option 2

Sequence continues through option 27.

Columns	Format	Description
27	I1	Value for option 27
31-35	I5	Number of minor time steps for this step if Time Step Option 1 is enabled.
36-45	F10.0	Length of this major time step, in days, if Time Step Option 1 is enabled.
46-55	F10.0	Time step acceleration factor for this major time step if Time Step Option 1 is enabled.
56-79	6A4	Comment to describe time step.

Data Set 14 - Pumpage for all cells
This data set contains a pumpage value for each cell in the system and is read if Time Step Option 2 is enabled.

The data are read a *row* at a time, with 10 values per card.

Columns*	Format*	Description
11-17	F7.0	Value for column 1 (11, 21, etc.)
18-24	F7.0	Value for column 2 (12, 22, etc.)
25-31	F7.0	Value for column 3 (13, 23, etc.)
32-38	F7.0	Value for column 4 (14, 24, etc.)
39-45	F7.0	Value for column 5 (15, 25, etc.)
46-52	F7.0	Value for column 6 (16, 26, etc.)
53-59	F7.0	Value for column 7 (17, 27, etc.)
60-66	F7.0	Value for column 8 (18, 28, etc.)
67-73	F7.0	Value for column 9 (19, 29, etc.)
74-80	F7.0	Value for column 10 (20, 30, etc.)

*Another format may be used by adding 5 to the controlling option and placing a format card at the first of the data set.

The units are volume per major time step, i.e., acre-feet per year. They are converted to cubic length per day by the conversion factor in Data Set 2

Data Set 15 - Pumpage by block
This data set contains pumpage rates to be assigned to all cells in a specified region of the grid and is read if Time Step Option 3 is enabled.

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-30	F10.0	Pumpage rate

*Another format may be used by adding 5 to the controlling option and placing a format card at the first of the data set.

The units are the same as those for Data Set 14.

The last card must be blank.

Data Set 16 - Pumpage adjustments
This data set contains pumpage adjustment factors which will multiply the pumpage rates for all cells in a specified region of the grid and is read if Time Step Option 4 is enabled.

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-30	F10.0	Pumpage adjustment factor.

*Another format may be used by adding 5 to the controlling option and placing a format card at the first of the data set.

The last card must be blank.

Data Set 17 - Recharge for all cells

This data set contains a recharge value for each cell in the system and is read if Time Step Option 5 is enabled. ~~The first card is the variable format card~~. The recharge rates are read a row at a time as is Data Set 14. The units are the same as those for Data Set 14.

Data Set 18 - Recharge by block

This data set contains recharge rates to be assigned to all cells in a specified region of the grid and is read if Time Step Option 6 is enabled. Data are read in the same manner as for Data Set 15 except recharge rate is read instead of pumpage rate. The units are the same as those for Data Set 14.

The last card must be blank.

Data Set 19 - Recharge adjustments

This data set contains recharge adjustment factors which will multiply the recharge rates for all cells in a specified region of the grid and is read if Time Step Option 7 is enabled. Data are read in the same manner as for Data Set 16 except the factor is applied to recharge.

The last card must be blank.

Data Set 20 - Heads for constant-head cells

This data set contains the end-of-major-time heads or changes in head during the major time step for constant-head cells, FLAG=0, and is read if Time Step Option 24 is enabled.

If data are to be read for all cells, option value of 1 or 2, Data are read in the same manner as for Data Set 14.

If values are to be read for a specified region of the grid, option value of 3 or 4~~, and the format used was (4I5,F10.0)~~, the data would be read as follows:

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment
21-30	F10.0	Head or change in head

*Another format may be used by adding 5 to the controlling option and placing a format card at the first of the data set.

If the option value is 3 or 4, the last card must be blank.

Data Set 21 - Limits of statistical blocks

This data set contains the row and column numbers which delineate a section of the grid for which the statistical data are to be calculated and is read if Time Step Option 27 is enabled. Up to 60 such blocks may be identified.

Columns*	Format*	Description
1-5	I5	First row of grid segment
6-10	I5	Last row of grid segment
11-15	I5	First column of grid segment
16-20	I5	Last column of grid segment

*Another format may be used by adding 5 to the controlling option and placing a format card at the first of the data set.

The last card must be blank.

Data Set 22 - Measured heads

This data set contains measured (observed) heads at the end of the major time step and is read if Time Step Option 22 is enabled. These heads are compared to the simulated heads.

Data are read in the same manner as for Data Set 14.

Data Set 23 - Mass transport title

This data set contains one card for input of title sentence. During output from the mass transport portion of the program, this sentence is printed following the title sentence contained in Data Set 1. This sentence should be centered on the card. This data set is read only during the first major time step.

Data Set 24 - Mass transport options

This one-card data set contains the Mass Transport Options plus other parameters and is read if General Program Option 15 is enabled.

Columns	Format	Description
1	I1	Value for option 1
2	I2	Value for option 2

Sequence contiue through column 20

Columns	Format	Description
31-40 ~~26-35~~	F10.0	Convergence criterion
~~36-45~~	~~F10.0~~	~~Ratio of porosity to specific~~ yield ~~(water table~~)
~~46-55~~	~~F10.0~~	Ratio ~~of porosity to storage~~ coefficient ~~(artesian)~~
41-50 ~~56-65~~	F10.0	Averaging coefficient for flows. Read only if a value other than 1 is to be used.

After the first time step, only the option values are read.

Data Set ~~26~~ 25 Dispersivity coefficients for all cells.
This data set contains the dispersivity coefficients for each cell and is read if Mass Transport Option 1 is enabled. The programs read all the longitudinal coefficients and then reads all the transverse coefficients. Each group of data is read in the same manner as Data Set 14. The unit is length.

Data Set ~~27~~ 26 Dispersivity coefficients by block.
This data set contains dispersivity coefficients to be assigned to all cells in a specified region of the grid and is read if Mass Transport Option 3 is enabled. The programs read the longitudinal dispersivity coefficients followed by the transverse coefficients; each read in the same manner as Data Set 15. The unit is length.

Data Set ~~28~~ 27 Dispersivity coefficient adjustments.
This data set contains dispersivity coefficient adjustment factors which will multiply the coefficients for all cells in a specified region of the grid and is read if Mass Transport Option 4 is enabled. The program reads the longitudinal adjustment factors followed by the transverse factors; each read in the same manner as Data Set 16.

Data Set 28 - Recharge quality for all cells
This data set contains the recharge quality for each cell and is read if Mass Transport Option 5 is enabled. The data are read in the same manner as Data Set 14. The units for recharge quality is used for initial concentrations. Recharge quality is used for any flows moving vertically into a cell.

Data Set 29 - Recharge quality by block
This data set contains recharge quality values to be assigned to all cells in a specified region of the grid and is read if Mass Transport Option 7 is enabled. The data are read in the same manner as Data Set 15. The units for recharge quality are the same as for initial concentrations. Recharge quality is used for any flows moving vertically into a cell.

Data Set 30 - Recharge quality adjustments
This data set contains recharge quality adjustment factors which will multiply the quality values for all cells in a specified region of the grid and is read if Mass Transport Option 8 is enabled. The data are read in the same manner as Data Set 16. The units for recharge quality are the same as for initial concentration. Recharge quality is used for any flows moving vertically into a cell.

Data Set 31 - Initial concentrations for all cells
This data set contains initial concentration for each cell and is read if Mass Transport Option 15 is enabled. Data are read in the same manner as Data Set 14. The units of initial concentration should match units used for recharge quality.

Data Set 32 - Initial concentrations by block
This data set contains initial concentrations to be assigned to all cells in a specified region of the grid and is read if Mass Transport Option 17 is enabled. Data are read in the same manner as Data Set 15. The units for initial concentration should match units used for recharge quality.

Data Set 33 - Initial concentration adjustments
This data set contains initial concentration adjustment factors which will multiply the concentration for all cells in a specified region of the grid and is read if Mass Transport Option 18 is enabled. Data are read in the same manner as Data Set 16.

Data Set 34 - Porosity for all cells.
This data set contains the porosity values for each cell and are read if Mass Transport Option 21 is enabled. They are read in the same manner as Data Set 14. The data are dimensionless.

Data Set 35 - Porosity by block.
This data set contains porosity values to be assigned to all cells in a specified region of the grid and are read if Mass Transport Option 23 is enabled. The data are read in the same manner as Data Set 15. The data are dimensionless.

Data Set 36 - Porosity adjustments.
This data set contains porosity adjustment factors which multiply the porosity values in a specified region of the grid and is read if Mass Transport Option 24 is enabled. The data are read in the same manner as Data Set 16. The data are dimensionless.

Data Set 37 - Measured concentrations
This data set contains measured value of concentration for the end of the major time step and are read if Mass Transport Option 14 is enabled. Data are read in the same manner as Data Set 14. The units should match the units used for initial concentration.

OUTPUT

Program GWSIM-IV was written to allow the user the ability to determine the types of output desired. The user selects the types of output produced by the program by the appropriate enabling of certain options. By proper planning, the user can eliminate the printing of unneeded information.

Output Unit Numbers

Two unit number variables are used for output of information. The unit number associated with the variable "OUT" should be set to the printer's logical unit number. The unit number associated with variable 'OUT1' could be any device for storage of simulated heads and/or physical data. The data may be punched or placed on a mass storage device. The variable 'OUT' is set equal to 6 and 'OUT1' equals 10.

General Description

The output may be tailored to the user's needs. The program automatically prints the values of many parameters, and through enabling options, almost all data may be printed. Generally, the enabling of an option is required to print any data that require a significant amount of printing. For example, the enabling of Time Step Option 11 is required to list the pumping rate for each cell.

Hydrologic

At the end of each minor time step, a message is printed which indicates the number of days simulated and the equivalent number of major time steps completed. The sum of the changes in head during the last iteration of the IADI procedure is printed. The number of iterations needed to complete the minor time step also is printed. If the number of iterations is equal to 51, the IADI procedure may not have converged. This could occur with an exceedingly small error criterion or an error in the physical data.

Upon the completion of each major time step, a listing of any springs / river cells is printed. The node will be identified by its row and column number. The flow rate at the end of the time step will also be printed along with the calculated water level in the cell. The total volume of spring flow during the time step is also printed. This number is calculated by summing the amount of flow for each of the minor time steps. The flow for each minor time step is equal to the flow rate at the end of the time step multiplied by the length of the time step.

A listing of the mass balances is printed upon completion of each major time step. Values for this time step and cumulative totals are printed. Values are expressed as rates per day and as total volume. Pumpage and recharge show values titled positive, negative, and net. A positive pumpage represents an outflow, and a positive recharge represents an inflow. The opposite is true for the negative values. Net equals positive minus negative. The reduction in pumpage values represents that amount of water that was not pumped because a cell dewatered, expressed as an average daily rate.

The reduction in recharge values represents that amount of water that could not be recharged because the water level for a water table cell was above the top of aquifer and land surface, expressed as an average daily rate. The mass balance terms give an indication of the accuracy of the simulation. Flow out of the aquifer are considered positive.

Mass Transport

The program automatically prints the information contained in Data Set 24. Other information printed includes default values for dispersion coefficients, recharge quality, and initial concentrations. Corrections to default values are also printed.

At the end of each minor time step, a message is printed which indicates the number of iterations required to complete the time step and the length of the time step in days. If the number of iterations is equal to 51, the IADI procedure may not have converged. Included in the printed message is a mass balance print-out. Terms shown include mass transferred into the aquifer, out of the aquifer, change in mass in the aquifer, and error in mass balance. The units of the mass balance terms are ~~gallons~~ cubic length times the units used for concentration. For example, if the units used were milligrams per liter, the units of the mass balance terms would be milligrams per liter ~~gallons~~ cubic length.

A cumulative balance of the mass balance terms are printed at the end of each major step.

APPLICATION TO EXAMPLE PROBLEM

As a demonstration of how the program GWSIM-II could be applied, an example problem was constructed. The problem involves simulating a 2 mile by 1 mile section of a water table aquifer. No flow is allowed to cross the boundary. One well is located approximately one-half mile from the left edge of the section and is pumping at the rate of 1,500 acre-feet per year. A second well located one-half mile from the right edge of the section is recharging water at the rate of 150 acre-feet per year. The initial head equals 50 feet, storage coefficient equals 10 percent, porosity equals 12.5 percent, permeability for both directions equals 400 gallons per square foot per day, the bottom of aquifer elevation equals -50 feet, the dispersion coefficient equals 0.05 square feet per day, initial concentration equals 100 milligrams per liter (mg/l) of total dissolved solids (TDS), and the recharge quality equals 200 mg/l of TDS. The aquifer is to be simulated for two years.

A uniform finite difference grid with ten columns and five rows was superimposed over the aquifer. A major time step length of one year was selected and the number of minor time steps was set at twelve. Error criterion of 0.05 feet and 0.01 mg/l were used for hydrologic and mass transport simulations; respectively. Figure 3 shows the data cards needed to simulate the aquifer problem. The user should study each card to determine the significance of each number. Figure 4 illustrates a portion of the output for the example problem.

APPENDIX

A. Program Description
B. Flow Chart of Main Program
C. Flow Chart of ~~Pumpage Prediction Program~~ Mass Transport Simulation Subroutine
D. Glossary of Selected Program Variables
E. Listing of Computer Program

APPENDIX A

PROGRAM DESCRIPTION

A brief discussion of each segment of GWSIM-IV is included in this appendix.

PROGRAM DESCRIPTION

EXEC
~~MAIN~~ PROGRAM

The exec ~~main~~ program reads basic data and calls various subroutines. All variables are modified and corrected as required, during each time step, in the exec ~~main~~ program. The majority of the arrays are dimensioned in the exec ~~main~~ program. If the finite difference grid contains more than 31 rows or 31 columns, the array declaration will have to be changed only in this segment.

SUBROUTINE - CALIB

This subroutine adjusts the values of hydraulic conductivity and storage coefficient. Such changes may be necessary during the calibration phase of model construction. The routine may also produce printer maps illustrating the values of hydraulic conductivity, transmissivity and storage coefficient. Hydraulic conductivity values are divided by 10 prior to printing, transmissivity values are divided by 100, and the specific yield for water table cells is multiplied by 1000 and storage coefficient for artesian cells is mutiplied by 1,000,000. Data Set 9 is read by this routine. The value of General Program Option 7 determines what maps will be printed. If the option is: equal to 1, no maps are printed; equal to 2, a hydraulic conductivity map and a transmissivity map for each direction are produced; equal 3, the storage coefficient map is printed; and equal to 4, all maps are printed. Transmissivity values equal the appropriate hydraulic conductivity value times the saturated thickness at the corresponding cell face.

SUBROUTINE - FLUX

This subroutine prints a map indicating the ground-water flows between nodes at the end of a time step. The maps are printed if either Time Step Options 12 or 17 is enabled. Both should not be enabled for the same time step. If maps are to be produced, the appropriate units conversion factor and label must be read in Data Set 2.

Two maps are produced. The first map shows flow between columns and is labeled 'Direction 1.' For cell i,j, the value printed is for flow from cell i,j to cell i,j+1. The second map, labeled 'Direction 2,' shows flow between rows. For cell i,j the flow is from cell i,j to cell i+1,j. A negative number represents a reversal of flow, i.e., from cell i,j+1 to cell i,j.

An example of a map produced by FLUX is shown in Figure 4.

SUBROUTINE - GETPMP

This subroutine is called for each major time step, and it reads the pumpage and recharge data. ~~The routine to calculate pumpage for the High Plains Model, HGHPMP, is called by this subroutine.~~ The net withdrawal rate in Equation 3, $Q_{i,j}$, is calculated, and the units are cubic length per day.

SUBROUTINE - HYDRO

This subroutine produces a hydrograph of water levels for specified cells. The program plots water levels at the end of major time steps and measured water levels if available. There is no limit to the number of major time steps. The head at the end of twenty time steps will be plotted per page.

SUBROUTINE - OUTPUT

This subroutine prints most of the model results. The mass balances are also computed in this routine. Many of the plotting routines are called from OUTPUT. Example output is shown in Figure 4.

SUBROUTINE - PHYSDT

This routine reads the physical data describing the aquifer. Subroutine CALIB is called to adjust hydraulic conductivity and storage coefficient. The units of hydraulic conductivity are converted to length per day units, and storage coefficient is multiplied by the cell's dimensions.

SUBROUTINE - PLOTH

The routine produces print plots of head or saturated thickness. A letter will be printed for each active cell in the system to indicate that cell's value of the parameter. The range for each letter is printed with statistics to indicate the distribution of the parameter. An example of such a map is shown in Figure 4.

The program will scale the plot in one of two ways. First, if the plotting scale factor read in Data Set 2 is zero, the maps will be printed without regard to cell spacings. No lines or spaces are skipped during the printing, resulting in a compact map.

If the scale factor is non-zero, grid spacings will be considered. If the factor greater than zero, the program attempts to print the information based on that scale. For example, if the factor equals 1000, the maps will be printed with 1000 length units per inch. If the grid spacings are such that more than one row (or column) occurs at a printing position, only the highest numbered row (or column) is shown. The plot will be segmented if necessary to produce a plot at the desired scale. As safety features, the plot will not be completed if the distance separating the first and last columns or first and last rows is more than 50 times the scale factor. Stated another way, the resulting plot may be no wider or longer than 50 inches.

If the scale factor is negative, the program computes the smallest scale factor that allows all data to be plotted. The 50 inch maximum size still applies, however.

Water quality values are also plotted by this routine

SUBROUTINE - PLOTS

This routine produces plots similar to those produced by Subroutine PLOTH. A map of simulated errors or head changes may be produced. Simulated error or difference is equal to the simulated head level minus observed head level. Statistics are printed which may be used to compare the head differences. The mean, standard deviation, maximum, and minimum values for the simulated head, observed head (if error map is produced) or beginning head (if head change map is produced), and difference in head are printed. The nodes with the maximum and minimum values are identified by row and column numbers. The mean and standard deviation of the absolute value of the head value is also printed. The covariance and regression coefficient are also printed, but these values have meaning only when an error map is produced. These two values are used to indicate the goodness-of-fit between the simulated and observed water level.

The subroutine only considers cells for which the observed head level is not zero. This allows the possibility of reading a set of observed head levels (Data Set 22) which contains known values only for cells that contain a measured well. Normally, Data Set 22 contains a measured value for all active cells, with most values obtained from a contour map.

Water quality values are also printed by this routine.

SUBROUTINE - QSOLVE

This routine solves the system of equations for the concentrations using the iterative alternating direction implicit procedure. A user supplied error criterion terminates the iterative sequence for each time step.

SUBROUTINE - QUAL

This subroutine reads data related to mass transport and calls mass transport related subroutines. The majority of the mass transport modeling is performed by this subroutine.

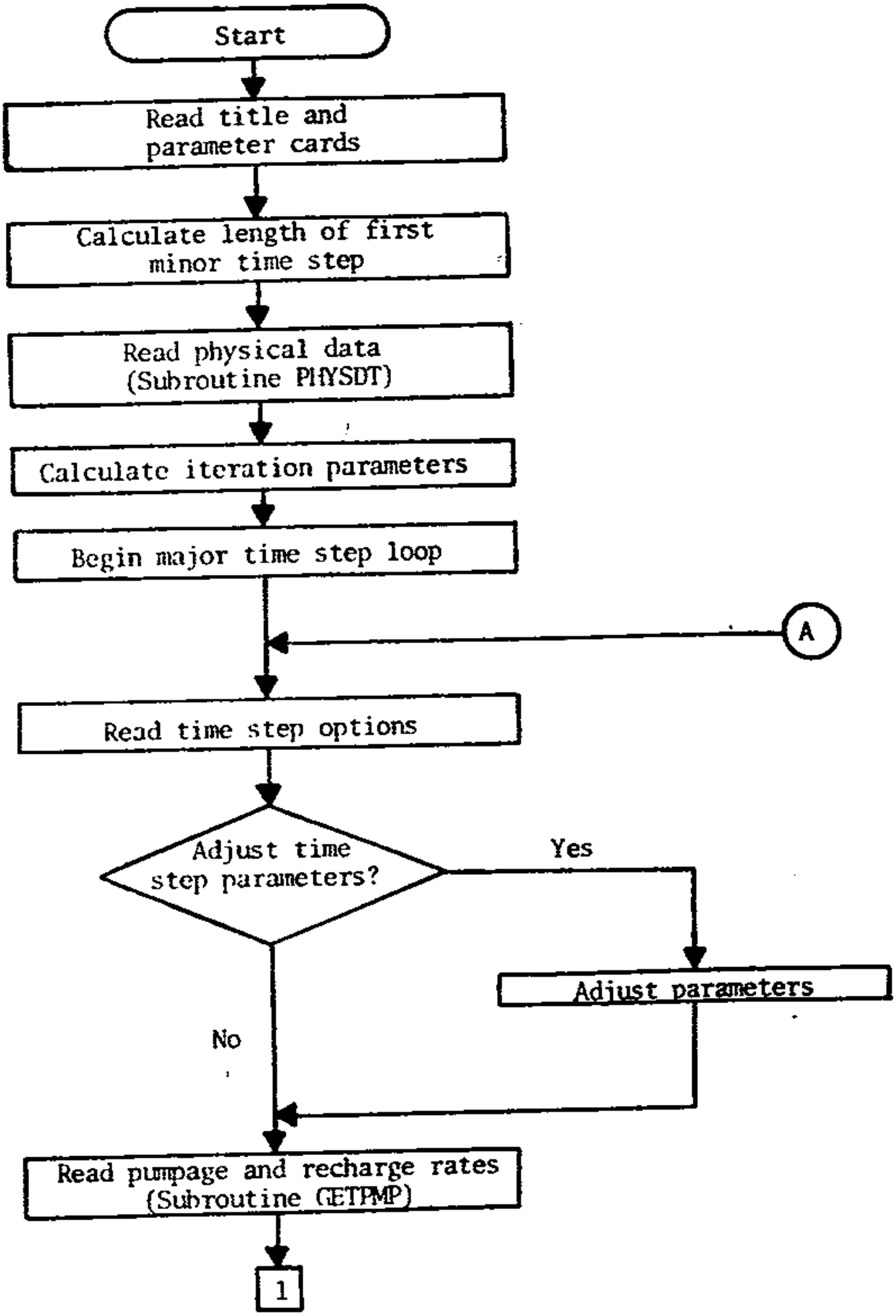
SUBROUTINE - SOLVE

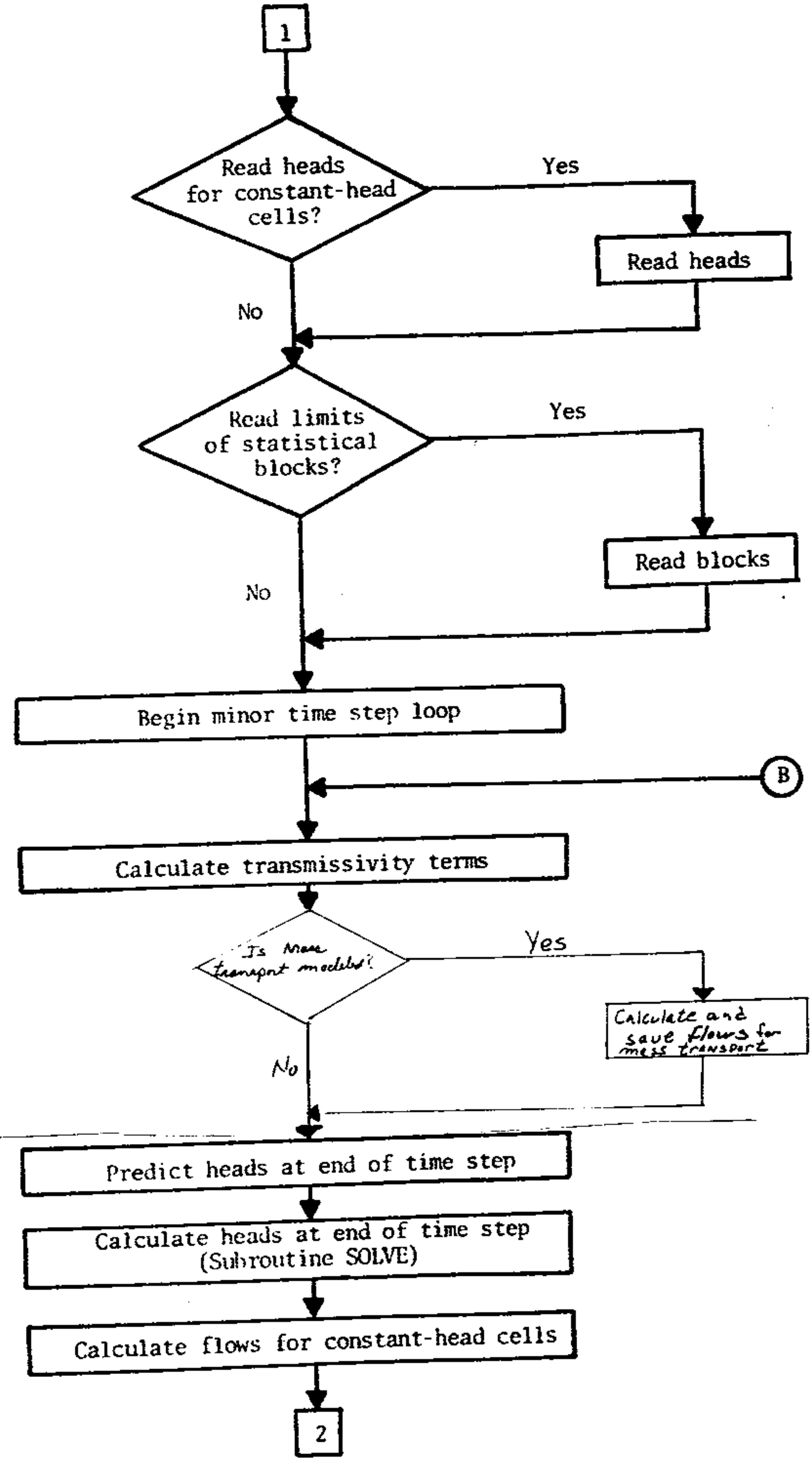
This routine solves the system of equations for the non-study state head using the iterative alternating direction implicit procedure. A user supplied error criterion terminates the iteration sequence for each time step. At least four iterations are completed to insure stability.

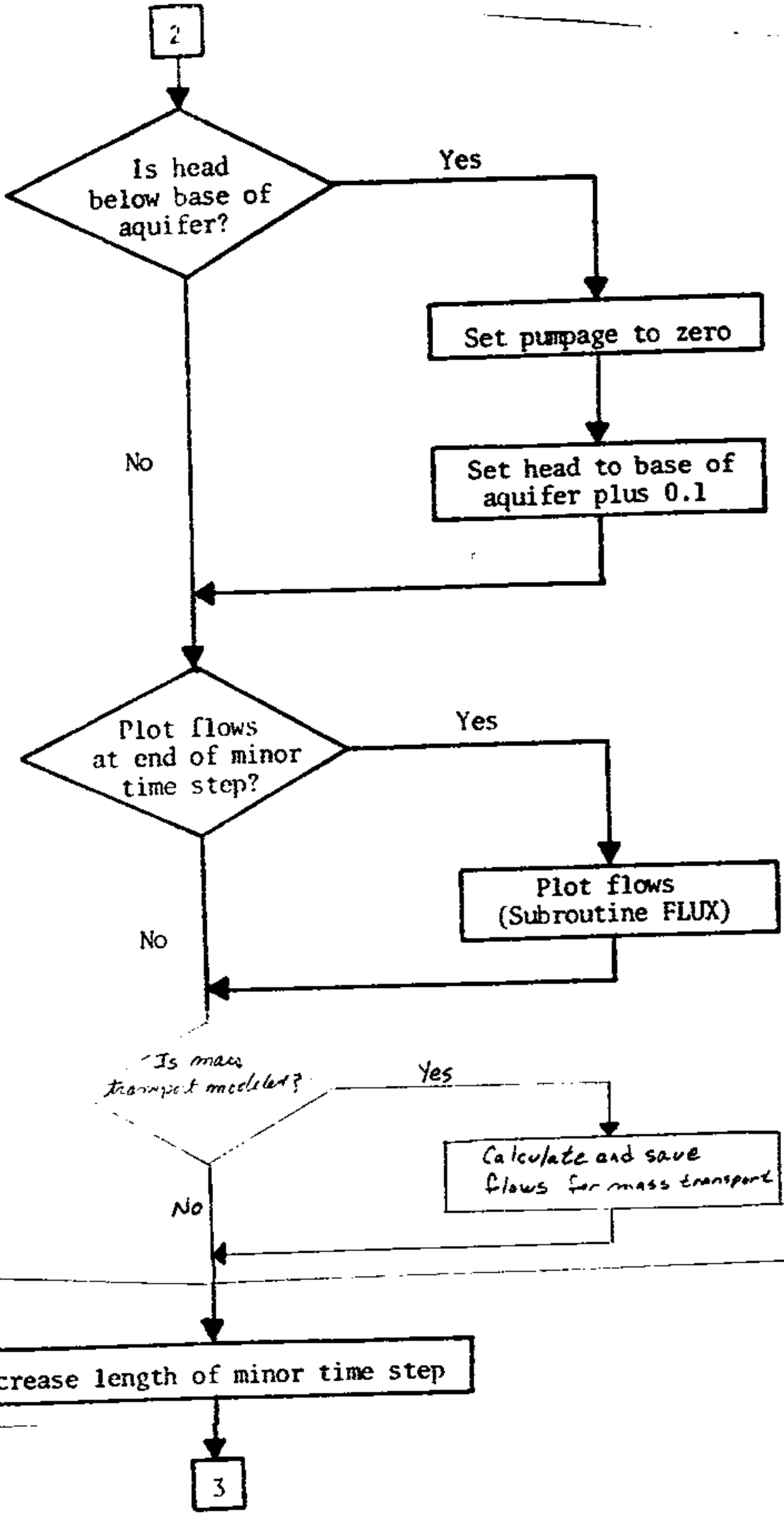
APPENDIX B

FLOW CHART OF MAIN PROGRAM

An abbreviated flow chart of the main program of GWSIM-IV is included in this appendix.



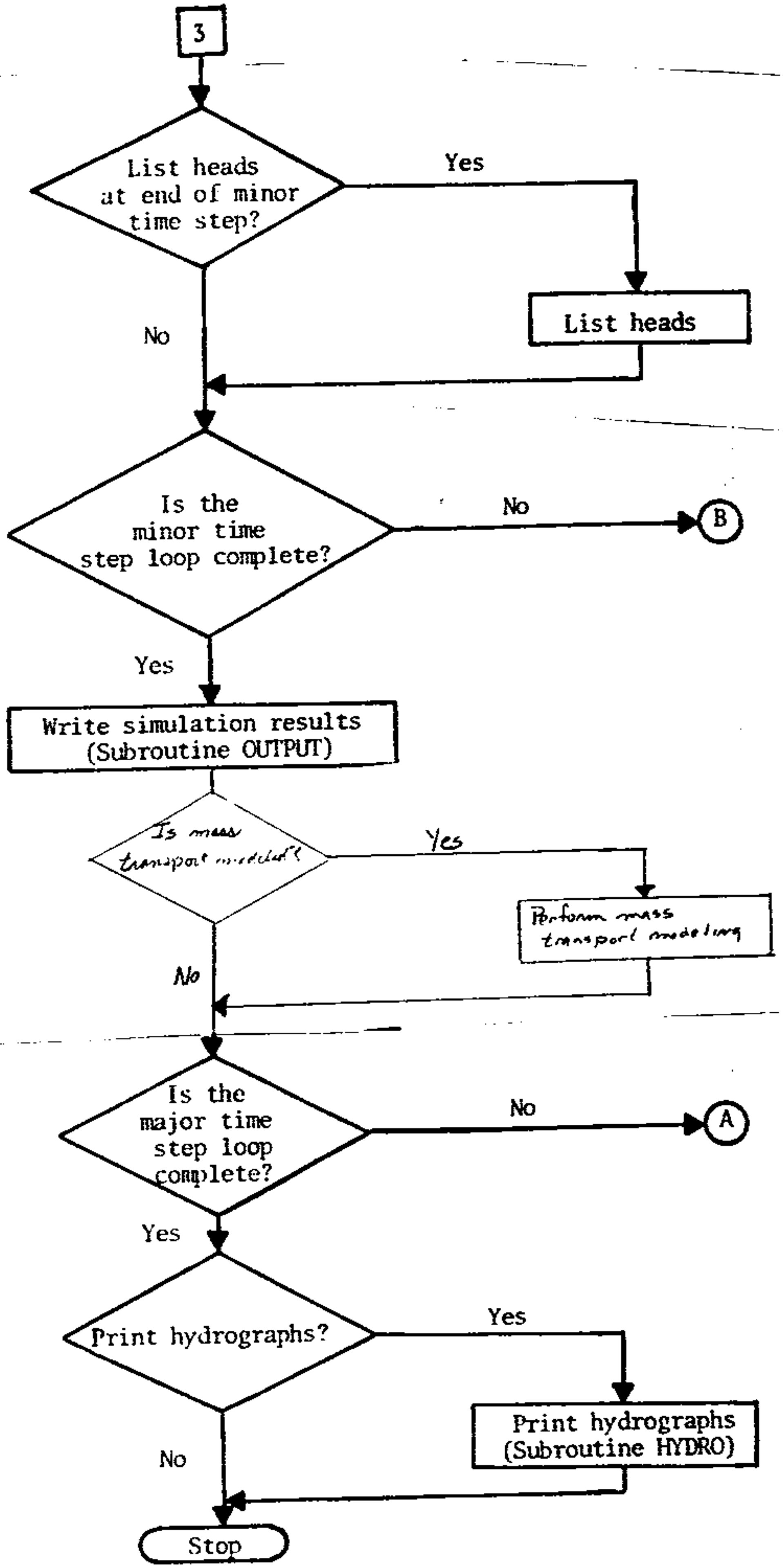

1
Read heads for constant-head cells?
Yes
Read heads
No
Read limits of statistical blocks?
Yes
Read blocks
No
Begin minor time step loop
B
Calculate transmissivity terms
Is mass transport modeled?
Yes
Calculate and save flows for mass transport
No
Predict heads at end of time step
Calculate heads at end of time step (Subroutine SOLVE)
Calculate flows for constant-head cells
2

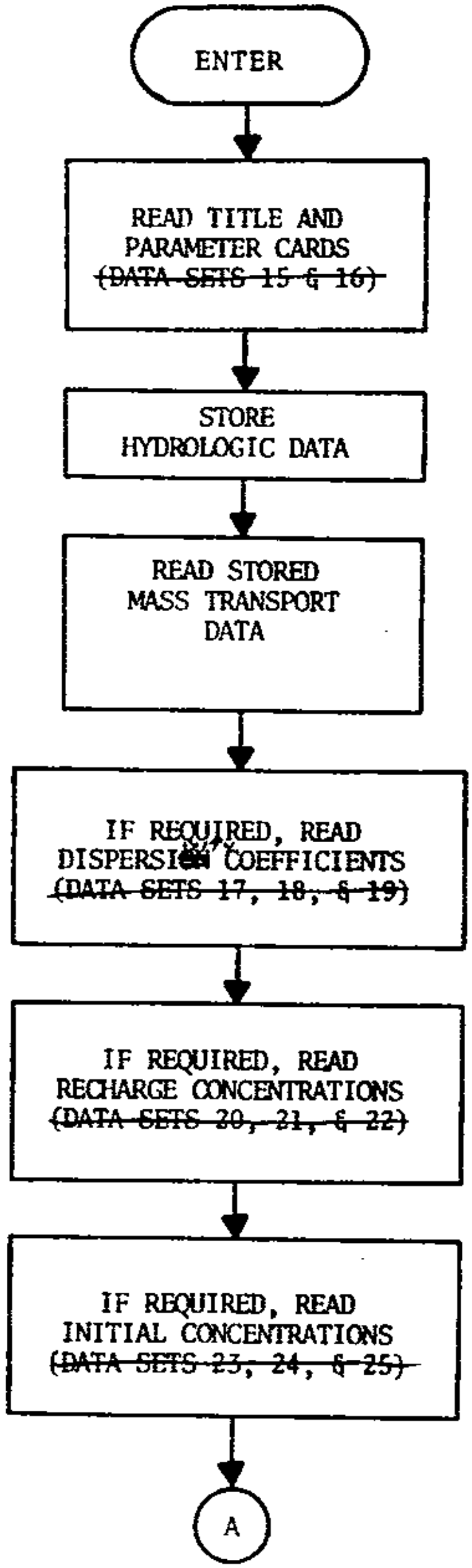

2
Is head below base of aquifer?
Yes
Set pumpage to zero
Set head to base of aquifer plus 0.1
No
Plot flows at end of minor time step?
Yes
Plot flows (Subroutine FLUX)
No
Is mass transport modeled?
Yes
Calculate and save flows for mass transport
No
Increase length of minor time step
3

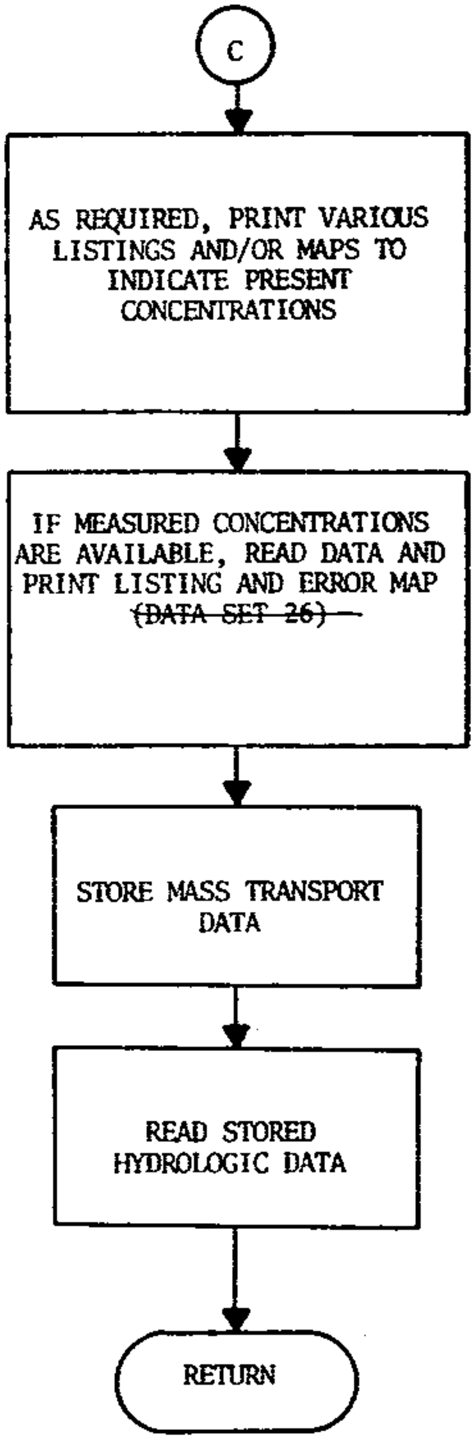
APPENDIX C

FLOW CHART OF SUBROUTINE QUAL

An abbreviated flow chart of subroutine QUAL is included in this appendix.


3
List heads at end of minor time step?
Yes
List heads
No
Is the minor time step loop complete?
No
B
Yes
Write simulation results (Subroutine OUTPUT)
Is mass transport modeled?
Yes
Perform mass transport modeling
No
Is the major time step loop complete?
No
A
Yes
Print hydrographs?
Yes
Print hydrographs (Subroutine HYDRO)
No
Stop


ENTER
READ TITLE AND
PARAMETER CARDS
(DATA SETS 15 & 16)
STORE
HYDROLOGIC DATA
READ STORED
MASS TRANSPORT
DATA
IF REQUIRED, READ
DISPERSION COEFFICIENTS
(DATA SETS 17, 18, & 19)
IF REQUIRED, READ
RECHARGE CONCENTRATIONS
(DATA SETS 20, 21, & 22)
IF REQUIRED, READ
INITIAL CONCENTRATIONS
(DATA SETS 23, 24, & 25)
A


C
AS REQUIRED, PRINT VARIOUS LISTINGS AND/OR MAPS TO INDICATE PRESENT CONCENTRATIONS
IF MEASURED CONCENTRATIONS ARE AVAILABLE, READ DATA AND PRINT LISTING AND ERROR MAP (DATA SET 26)
STORE MASS TRANSPORT DATA
READ STORED HYDROLOGIC DATA
RETURN

APPENDIX D

GLOSSARY OF SELECTED PROGRAM VARIABLES

A glossary of selected program variables used in GWSIM-IV is included in this appendix.

GLOSSARY OF SELECTED PROGRAM VARIABLES

Variable Name	Definition
ALPHA	Weighting factor for determining concentration of flux between cells
BOTLEL(I,J)	Elevation of bottom of aquifer for cell i,j (L)
DELMAJ	Length of major time step in days (T)
DELTA	Length of minor time step in days (T)
DELX(J)	Grid spacings in x-direction (L)
DELY(I)	Grid spacings in y-direction (L)
DL(I,J)	Hydrologics-Change in head during preceding time step (L) Mass Transport-Flux between cells I,J and I+1,J(L**3/T)
ERROR	Minimum head change allowed for convergence of solution procedure (L)
FLAG(I,J)	Type declaration for cell i,j Equal Zero -- Constant head Equal One — water table Equal Three -- Boundary
FLXFCT	Factor to convert ground-water flows prior to printing
FLXNAM	Title to indicate units of printed ground-water flows
FMT	Variable format array
H(I,J)	Hydrologics - Head at end of time step (L) Mass Transport - Volume of solution in cell at end of time step (L**3)
HO(I,J)	Hydrologics - Head at beginning of time step (L) Mass Transport - Volume of solution in cell at beginning of time step (L**3)
I	Model row number

IN,IN1,IN2,IN3, IN4,IN5,IN6	Input unit numbers
IOPT	
ISAVE	Storage array for row numbers of hydrograph cells
ISTEP	Major time step number
ISPRNG	Storage array for row number of spring/river cells
ITER	Number of iterations by IADI procedure
J	Model column number
JSAVE	Storage array for column number of hydrograph cells
JSPRNG	Storage array for column number of spring/river cells
KHYD	Switch variable to cause printing of hydrographs
KQUAL	Switch variable to cause mass transport simulation
MCOLS	Storage array for column numbers for cross-section procedure
MINOR	Minor time step number
MROWS	Storage array for row numbers for cross-section procedure
NBLK	Number of statistical blocks
NC	Number of columns in model
NCOLS	Number of columns for which cross sections are desired
NPARM	Number of iteration parameters
NR	Number of rows in model
NROWS	Number of rows for which cross sections are desired
NSAVE	Number of nodes for which hydrographs are desired
NSPRG	Number of springs or river cells

NSP	Number of minor time steps per major time step
NSTEPS	Number of major time steps
OPT	General program and time step options array
OUT,OUT1	Output unit numbers
P(I,J,1)	Hydrologic - Aquifer permeability between cells I,J and I,J+1 (L/T) . Mass transport - Dispersion coefficient between cells I,J and I,J+1 (L*L/T) .
P(I,J,2)	Hydrologic - Aquifer permeability between cells I,J and I+1,J(L/T) . Mass transport - Dispersion coefficient between cells I,J and I+1,J(L*L/T)
PERFCT	Factor to convert input values of hydraulic conductivity to interal units of length per day
PMPFCT	Factor to convert input values of pumpage and recharge rates to interal units of cubic length per day
PMPNAM	Title to indicate units on pumpage and recharge input rates
PRMITR(10)	Iteration parameters
Q(I,J)	Pumpage rate for cell i,j (L**3/T)
QERROR	Minimum concentration change for convergence
R(I,J)	Hydrologic - Slope of flow response line. (L*L/T) Mass transport - Concentration at beginning of time step

RD(I,J)	✗	Hydrologic - Minimum head for springflow or reference head in source (sink) (L) Mass transport - Concentration at end of time step
RHC(I,J)		Recharge rate for cell i,j (L**3/T)
SCALE		Plotting scale factor (L/inch)
SF1(I,J)		Storage coefficient for cell i,j
STORFT(2)	✗	Ratio of porosity to storage coefficient 1. Water table 2. Artesian
SURFAC(I,J)	✗	Hydraulics - Land surface elevation (L) Mass transport - Recharge concentration
T(I,J,1)	✗	Hydraulics - Transmissibility term between cells I,J and I,J+1(L*L/T) Mass transport - Dispersion term between cells I,J and I,J+1 (L**3/T)
T(I,J,2)	✗	Hydraulics - Transmissibility term between cells I,J and I+1,J(L*L/T) Mass transport - Dispersion term between cells I,J and I+1,J(L**3/T)
TIMACL		Time step acceleration factor
THIK(I,J)		Saturated thickness for cell i,j (L)
TOPAQ(I,J)	✗	Hydraulics - Top of aquifer elevation (FT) Mass transport - Flux between cells I,J and I,J+1(L*3/T)
XLGTNM		Title to indicate length unit

TIMACL Time acceleration factor
STRFCT Ratio of water table to artesian storage coeff.

APPENDIX E

LISTING OF COMPUTER PROGRAM

A listing of the computer program for GWSIM-I V is included in this appendix.

```
      NP(2)=1                                                   CALIB
      NP(3)=1                                                   CALIB
   50 CONTINUE                                                  CALIB
      NP(4)=NP(1)                                               CALIB
      NP(5)=NP(2)                                               CALIB
      WRITE (OUT,330)                                           CALIB
   60 READ (IN,340) II,III,JJ,JJJ,K,HA                          CALIB
      IF (II.LT.1) GO TO 110                                    CALIB
      WRITE (OUT,350) II,III,JJ,JJJ,K,HA                        CALIB
      DO 100 I=II,III                                           CALIB
      DO 100 J=JJ,JJJ                                           CALIB
      IF(K.GT.0) GO TO 80                                       CALIB
      T1=THIK(I,J)                                              CALIB
      IF(FLAG(I,J).EQ.1) T1=H(I,J)-BOTLEL(I,J)                  CALIB
      IF(K.EQ.-2) GO TO 70                                      CALIB
      P(I,J,1)=0                                                CALIB
      IF(J.EQ.NC) GO TO 100                                     CALIB
      T2=THIK(I,J+1)                                            CALIB
      IF(FLAG(I,J+1).EQ.1) T2=H(I,J+1)-BOTLEL(I,J+1)            CALIB
      P(I,J,1)=HA/((T1*DELX(J+1)+T2*DELX(J))/(DELX(J)+DELX(J+1)))   CALIB
      GO TO 100                                                 CALIB
   70 P(I,J,2)=0.                                               CALIB
      IF(I.EQ.NR) GO TO 100                                     CALIB
      T2=THIK(I+1,J)                                            CALIB
      IF(FLAG(I+1,J).EQ.1) T2=H(I+1,J)-BOTLEL(I+1,J)            CALIB
      P(I,J,2)=HA/((T1*DELY(I+1)+T2*DELY(I))/(DELY(I)+DELY(I+1)))   CALIB
   80 CONTINUE                                                  CALIB
      IF(K.EQ.3) GO TO 90                                       CALIB
      P(I,J,K)=P(I,J,K)*HA                                      CALIB
      IF(HA.LT.0.0) P(I,J,K)=-HA                                CALIB
      GO TO 100                                                 CALIB
   90 S(I,J)=S(I,J)*HA                                          CALIB
      IF(HA.LT.0.)S(I,J)=-HA                                    CALIB
  100 CONTINUE                                                  CALIB
      GO TO 60                                                  CALIB
  110 DO 320 K=1,5                                              CALIB
      IF (NP(K).LT.1) GO TO 320                                 CALIB
      IST=1                                                     CALIB
  120 IEND=IST+31                                               CALIB
      GO TO (130,140,150,150,170),K                             CALIB
  130 WRITE (OUT,360)                                           CALIB
      WRITE (OUT,420)                                           CALIB
      GO TO 180                                                 CALIB
  140 WRITE (OUT,370)                                           CALIB
      WRITE (OUT,420)                                           CALIB
      GO TO 180                                                 CALIB
  150 WRITE (OUT,380)                                           CALIB
      WRITE (OUT,420)                                           CALIB
      WRITE (OUT,450)                                           CALIB
      GO TO 180                                                 CALIB
```



```
  160 WRITE (OUT,430)                                           CALIB
      WRITE (OUT,420)                                           CALIB
      GO TO 180                                                 CALIB
  170 WRITE(OUT,440)                                            CALIB
      WRITE (OUT,420)                                           CALIB
  180 CONTINUE                                                  CALIB
      IF(NR.LT.IEND)IEND=NR                                     CALIB
      WRITE (OUT,390)                                           CALIB
      DO 190 I=IST,IEND                                         CALIB
  190 NB(I)=IEND-I+IST                                          CALIB
      WRITE (OUT,410) (NB(I),I=IST,IEND)                        CALIB
      DO 310 J=1,NC                                             CALIB
      DO 300 I=IST,IEND                                         CALIB
      L=IEND-I+IST                                              CALIB
      NB(I)=0                                                   CALIB
      T1=THIK(L,J)                                              CALIB
      IF(FLAG(L,J).EQ.1) T1=H(L,J)-BOTLEL(L,J)                  CALIB
      IFL=FLAG(L,J)+1                                           CALIB
      M=K                                                       CALIB
      GO TO (260,280,200,270,280),M                             CALIB
  200 GO TO (210,220,230,240),IFL                               CALIB
  210 NB(I)=999                                                 CALIB
      GO TO 250                                                 CALIB
  220 NB(I)=S(L,J)*1000.+0.5                                    CALIB
      GO TO 250                                                 CALIB
  230 NB(I)=S(L,J)*1.E8+0.5                                     CALIB
      GO TO 250                                                 CALIB
  240 NB(I)=999999                                              CALIB
  250 CONTINUE                                                  CALIB
      GO TO 300                                                 CALIB
  260 NB(I)=P(L,J,M)*0.1+0.5                                    CALIB
      GO TO 290                                                 CALIB
  270 IF(J.EQ.NC) GO TO 290                                     CALIB
      T2=THIK(L,J+1)                                            CALIB
      IF(FLAG(L,J+1).EQ.1) T2=H(L,J+1)-BOTLEL(L,J+1)            CALIB
      NB(I)=(T1*DELX(J+1)+T2*DELX(J))/(DELX(J)+DELX(J+1))       CALIB
     1*P(L,J,1)*0.01+0.5                                        CALIB
      GO TO 290                                                 CALIB
  280 IF(L.EQ.IEND) GO TO 290                                   CALIB
      T2=THIK(L+1,J)                                            CALIB
      IF(FLAG(L+1,J).EQ.1) T2=H(L+1,J)-BOTLEL(L+1,J)            CALIB
      NB(I)=(T1*DELY(L+1)+T2*DELY(L))/(DELY(L)+DELY(L+1))       CALIB
     1*P(L,J,2)*0.01+0.5                                        CALIB
  290 IF(IFL.EQ.4) NB(I)=99999                                  CALIB
  300 CONTINUE                                                  CALIB
  310 WRITE (OUT,410) (NB(I),I=IST,IEND),J                      CALIB
      IST=IEND+1                                                CALIB
      IF (IST.LT.NR) GO TO 120                                  CALIB
  320 CONTINUE                                                  CALIB
      RETURN                                                    CALIB
```

```
C*******************************************************************
      DELTA=1.0
      N=NSP
   10 N=N-1
      IF (N) 30,30,20
   20 DELTA=DELTA+TIMACL**N
      GO TO 10
   30 DELTA=DELMAJ/DELTA
      WRITE (OUT,560) NSTEPS,NSP,DELMAJ,DELTA,ERROR,NC,NR,NPARM,NSPRG,S
     1TRFCT,TIMACL,PMPFCT,PMPNAM,PERFCT,FLXFCT,FLXNAM,XLGTHM
C*******************************************************************
C        READ PHYSICAL DATA
C*******************************************************************
      CALL PHYSDT(NROW,NCOL,FLAG,BOTLEL,H,HO,P,SF1,T,THIK,SURF,
     1TOPAO,R,RD)
   40 TIME=0.0
      NBLK=1
      IRWC(1,1)=1
      IRWC(2,1)=NR
      IRWC(3,1)=1
      IRWC(4,1)=NC
      TIMAC2=TIMACL
      DELMJ2=DELMAJ
      DEL=DELTA
C*******************************************************************
C        CALCULATE ITERATION PARAMETERS
C*******************************************************************
      HA=2.
      HC=3.14159*3.14159/(2*NR*NR)
      HB=3.14159*3.14159/(2*NC*NC)
      DO 50 I=1,NR
      DO 50 J=1,NC
      IF (FLAG(I,J).GT.2) GO TO 50
      HF=DELX(J)*DELX(J)/(DELY(I)*DELY(I))
      HD=HB/(1.+HF)
      F=HC/(1.+1./HF)
      HA=AMIN1(HA,HD,F)
   50 CONTINUE
      F=EXP(ALOG(1./HA)/(NPARM-1))
      PRMITR(1)=HA
      DO 60 I=2,NPARM
   60 PRMITR(I)=PRMITR(I-1)*F
      WRITE (OUT,810) (PRMITR(I),I=1,NPARM)
      TIME1=0.
C*******************************************************************
C        BEGIN MAJOR TIME STEP LOOP
C*******************************************************************
      DO 480 ISTEP=1,NSTEPS
      WRITE(OUT,710)TITMOD,T          AAA=1.
      WRITE (OUT,720) ISTEP
```



```
      DELTA=DEL
      NSP1=NSP
      DELMAJ=DELMJ2
      TIMACL=TIMAC2
C*******************************************************************
C        READ OPTIONS FOR THIS TIME STEP
C*******************************************************************
      READ (IN,670) OPT,NSP2,DELMJ1,TIMAC1
     1,(B(J),J=1,8)
      WRITE (OUT,880) (B(J),J=1,8)
      IF(OPT(14).GT.0.OR.OPT(15).GT.0) KOUT=1
      DO 70 I=1,30
      IF (OPT(I).GT.0) WRITE (OUT,550) I,OPT(I)
      IF (OPT(1).GT.0.AND.I.EQ.1) WRITE (OUT,690) NSP2,DELMJ1,TIMAC1
   70 CONTINUE
   80 CONTINUE
      NSP1=NSP
      IF (OPT(1).LT.1) GO TO 120
C*******************************************************************
C        ADJUST TIME STEP PARAMETERS
C*******************************************************************
      TIMACL=TIMAC1
      DELTA=1.
      N=NSP2
   90 N=N-1
      IF (N) 110,110,100
  100 DELTA=DELTA+TIMACL**N
      GO TO 90
  110 DELTA=DELMJ1/DELTA
      NSP1=NSP2
      DELMAJ=DELMJ1
  120 CONTINUE
      DO 130 I=1,18
      SUMS(I,1)=0.
      IF(ISTEP.EQ.1)SUMS(I,2)=0.
  130 CONTINUE
      DO 140 I=1,NR
      DO 140 J=1,NC
  140 QSUM(I,J)=0.0
C*******************************************************************
C        READ EXTERNAL FLUX (PUMPAGE AND RECHARGE)
C*******************************************************************
      CALL GETPMP(NROW,NCOL,FLAG,Q,RNG)AAA)
C*******************************************************************
C        READ ENDING HEADS FOR CONSTANT-HEAD CELLS
C*******************************************************************
      IF (OPT(24).LT.1) GO TO 210
      WRITE(OUT,750)
      IFL=OPT(24)
      IF(OPT(24).GT.5) OPT(24)=OPT(24)-5
```

```
      1W,NCOL), RO(NROW,NCOL)
       GO TO (10,20,80,100,110), N
C      N=4  READ HEADS
10     REWIND IN6
       RETURN
20     REWIND IN7
       WRITE (IN7) ((R(I,J),RO(I,J),I=1,NR),J=1,NC)
       DO 70 J=1,NC
       DO 60 I=1,NR
       R(I,J)=0.0
       RO(I,J)=0.0
       IF (I-NR) 30,40,40
30     RO(I,J)=T(I,J,2)*(H(I,J)+HO(I,J)-H(I+1,J)-HO(I+1,J))*0.5
40     IF (J-NC) 50,60,60
50     R(I,J)=T(I,J,1)*(H(I,J)+HO(I,J)-H(I,J+1)-HO(I,J+1))*0.5
60     CONTINUE
       WRITE (IN8) (R(I,J),I=1,NR)
70     WRITE (IN8) (RO(I,J),I=1,NR)
       REWIND IN7
       READ (IN7) ((R(I,J),RO(I,J),I=1,NR),J=1,NC)
       RETURN
80     DO 90 J=1,NC
       READ (IN6) (H(I,J),I=1,NR)
       READ (IN6) (HO(I,J),I=1,NR)
90     CONTINUE
       RETURN
100    CONTINUE
       RETURN
110    REWIND IN8
       RETURN
       END
```



```
      SUBROUTINE XSECT (NROW,NCOL,FLAG,NSIM,NOBS,BOTLEL)              XSECT
C*********************************************************************XSECT
C     THIS SUBROUTINE PRODUCES A PRINTER PLOT OF CROSS-SECTIONS ALONG XSECT
C     ROWS OR COLUMNS.  MAXIMUM NUMBER OF ROWS OR COLUMNS IS 100.     XSECT
C*********************************************************************XSECT
      INTEGER IDUM(10)                                                XSECT
      DATA IDUM/1,2,3,4,5,6,7,8,9,0/                                  XSECT
      COMMON /ITCOM/ NR,NC,ISTEP,NPARM,IN,OUT,OUT1,OPT(30),ITER,NSAVE,XSECT
     1ISAVE(25),JSAVE(25),KHYD,NCOLS,NCOLS(25),NROWS,NROWS(25)        XSECT
     2,IN1,IN2,IN3,IN4,IN5,IN6                                        XSECT
     1,NSTEPS,NBLK,IRWC(4,80),NSPRG,ISPRG(25),JSPRG(25)               XSECT
      COMMON /RLCOM/ FMT(20),TITLE(20),DELX(100),DELY(100),PRMITR(10),E XSECT
     1(100),G(100),SUMS(18,2),                                        XSECT
     2                          ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,     XSECT
     3DELMAJ,E,XLGTNM,FLXNAM(2),FLXFCT                                XSECT
```



```
     1,DELMJ2,TIME,STRFCT,SCALE                                       XSECT
     1,TITMOD(20),VFMT(20,8)
      INTEGER OPT,FLAG,OUT,OUT1                                       XSECT
      DIMENSION NSIM(NROW,NCOL),NOBS(NROW,NCOL),PLOT(100),            XSECT
     1FLAG(NROW,NCOL)                                                 XSECT
     2,BOTLEL(NROW,NCOL)                                              XSECT
     1,XINTER(10)                                                     XSECT
      DATA XINTGR/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0/            XSECT
      EQUIVALENCE (PLOT(1),E(1))                                      XSECT
      DATA SIM,OBS,BLANK,BOTH/1HS,1HO,1H ,1HB/                        XSECT
      DIMENSION XCOL(2)                                               XSECT
      DATA XROW/3HROW/,XCOL/4HCOLU,2HMN/                              XSECT
      IOPT=OPT(22)                                                    XSECT
      DO 220 L=1,NCOLS                                                XSECT
      IF (NCOLS.LT.1) GO TO 220                                       XSECT
      ORIGX=0                                                         XSECT
      IF(SCALE.GT.0.) GO TO 20                                        XSECT
      DO 10 I=1,NR                                                    XSECT
   10 G(I)=I                                                          XSECT
      JST=1                                                           XSECT
      J=NR+1                                                          XSECT
      GO TO 60                                                        XSECT
   20 G(1)=0                                                          XSECT
      DO 30 I=2,NR                                                    XSECT
   30 G(I)=G(I-1)+(DELY(I)+DELY(I-1))/2.                              XSECT
      JST=1                                                           XSECT
      IF(G(NR)/SCALE.GT.50.) GO TO 450                                XSECT
   40 DO 50 J=JST,NR                                                  XSECT
      IF((G(J)-ORIGX).GT.9.9*SCALE) GO TO 60                          XSECT
   50 CONTINUE                                                        XSECT
      J=NR+1                                                          XSECT
   60 JEND=J-1                                                        XSECT
      GSAVE=G(JEND)                                                   XSECT
      IF(JST.EQ.JEND.AND.JST.NE.NR) GO TO 230                         XSECT
      WRITE(OUT,540)TITMOD,TITLE,ISTEP
      J=NCOLS(L)                                                      XSECT
      WRITE (OUT,470) J                                               XSECT
      WRITE (OUT,550)                                                 XSECT
      IF(SCALE.LE.1.E-3) GO TO 80                                     XSECT
      DO 70 I=JST,JEND                                                XSECT
      K=(G(I)+SCALE/20.-ORIGX) /(SCALE/10.)+1                         XSECT
   70 G(I)=K+0.5                                                      XSECT
   80 WRITE (OUT,570) XROW                                            XSECT
      DO 110 LL=1,2                                                   XSECT
      DO 90 K=1,100                                                   XSECT
   90 PLOT(K)=BLANK                                                   XSECT
      DO 100 I=JST,JEND                                               XSECT
      KG=G(I)                                                         XSECT
      KL=I/10                                                         XSECT
      PLOT(KG)=XINTGR(KL)                                             XSECT
```

```
            PLOT(KG)=XINTGR(KL)
            IF(KL.EQ.0) PLOT(KG)=BLANK                                  XSECT
            IF(LL.EQ.1) GO TO 330                                       XSECT
            KL=MOD(J,10)                                                XSECT
            IF(KL.EQ.0) KL=10                                           XSECT
            PLOT(KG)=XINTGR(KL)                                         XSECT
        330 CONTINUE                                                    XSECT
        340 WRITE(OUT,520) PLOT                                         XSECT
            HMAX=-1.E5                                                  XSECT
            HMIN=1.E5                                                   XSECT
            DO 350 J=JST,JEND                                           XSECT
            IF (FLAG(I,J).GT.2) GO TO 350                               XSECT
            HA=HSIM(I,J)                                                XSECT
            IF(IOPT.GT.0.AND.ABS(HOBS(I,J)).GE.1.E-2) HA=HOBS(I,J)      XSECT
            HB=HA                                                       XSECT
            IF(FLAG(I,J).EQ.1) HB=BOTLEL(I,J)                           XSECT
            HMAX=AMAX1(HMAX,HSIM(I,J),HA,HB)                            XSECT
            HMIN=AMIN1(HMIN,HSIM(I,J),HA,HB)                            XSECT
        350 CONTINUE                                                    XSECT
            HMAX=IFIX(HMAX+1.)                                          XSECT
            HMIN=IFIX(HMIN-1.)                                          XSECT
            XINC=(HMAX-HMIN)/40.                                        XSECT
            XINC=FLOAT(IFIX((XINC+0.5)*2.))/2.0                         XSECT
            XXINC=XINC*0.5                                              XSECT
            DO 400 LL=1,41                                              XSECT
            DO 360 J=1,100                                              XSECT
        360 PLOT(J)=BLANK                                               XSECT
            DO 390 J=JST,JEND                                           XSECT
            KG=G(J)                                                     XSECT
            IF (FLAG(I,J).GT.2) GO TO 390                               XSECT
            IF(ABS(BOTLEL(I,J)-HMAX).LE.XXINC) PLOT(KG)=1N+             XSECT
            IF(ABS(HSIM(I,J)-HMAX).LE.XXINC) PLOT(KG)=SIM               XSECT
            IF (IOPT.LT.1) GO TO 390                                    XSECT
            IF (ABS(HOBS(I,J)-HMAX).LE.XXINC) GO TO 370                 XSECT
            GO TO 390                                                   XSECT
        370 IF(PLOT(KG).EQ.SIM) GO T O 380                              XSECT
            PLOT(KG)=OBS                                                XSECT
            GO TO 390                                                   XSECT
        380 PLOT(KG)=BOTH                                               XSECT
        390 CONTINUE                                                    XSECT
            WRITE (OUT,510) HMAX,PLOT                                   XSECT
            HMAX=HMAX-XINC                                              XSECT
        400 CONTINUE                                                    XSECT
            WRITE (OUT,570) XCOL                                        XSECT
            DO 430 LL=1,2                                               XSECT
            DO 410 K=1,100                                              XSECT
        410 PLOT(K)=BLANK                                               XSECT
            DO 420 J=JST,JEND                                           XSECT
            KG=G(J)                                                     XSECT
            KL=J/10                                                     XSECT
```



```
            PLOT(KG)=XINTGR(KL)
            IF(KL.EQ.0) PLOT(KG)=BLANK                                  XSECT
            IF(LL.EQ.1) GO TO  420                                      XSECT
            KL=MOD(J,10)                                                XSECT
            IF(KL.EQ.0) KL=10                                           XSECT
            PLOT(KG)=XINTGR(KL)                                         XSECT
        420 CONTINUE                                                    XSECT
        430 WRITE (OUT,520) PLOT                                        XSECT
            JST=JEND                                                    XSECT
            ORIGX=GSAVE                                                 XSECT
            G(JST)=GSAVE                                                XSECT
            IF(JST.LT.NC) GO TO 270                                     XSECT
        440 CONTINUE                                                    XSECT
            RETURN                                                      XSECT
        450 WRITE (OUT,580)                                             XSECT
        460 RETURN                                                      XSECT
      C*******************************************************************  XSECT
      C                                                                 XSECT
      C                                                                 XSECT
      C                                                                 XSECT
      C                                                                 XSECT
        470 FORMAT (25HOCROSS-SECTION FOR COLUMN,I3)                    XSECT
        480 FORMAT (1HO,24X,100I1)                                      XSECT
        490 FORMAT (15X,100I1)                                          XSECT
        500 FORMAT (7X,'HEAD',5X,(100I1)                                XSECT
        510 FORMAT (1X,F10.2,5X,100A1)                                  XSECT
        520 FORMAT (T17,100A1)                                          XSECT
        530 FORMAT (22HOCROSS-SECTION FOR ROW,I3)                       XSECT
        540 FORMAT(1H1,T26,20A4//T26,20A4/T64,'FOR TIME STEP',I5//)
        550 FORMAT (1H+,T40,'S=SIMULATED',5X,'O=OBSERVED',5X,'B=OBSERVED+SIMUL  XSECT
           1ATED',5X,'+=BASE')                                          XSECT
        560 FORMAT (1H+,15X,2A4)                                        XSECT
        570 FORMAT (1HO,T17,2A4)                                        XSECT
        580 FORMAT (' SCALE INCORRECT  PLOT TERMINATED'//)              XSECT
            END                                                         XSECT
```

```
      COMMON /XLCOM/ FMT(20),TITLE(20),DELX(100),DELY(100),   PARM(10),B
     1(100),G(100),SUMS(16,2),
     2                                  ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,
     3DELMAJ,E,XLGTHM,FLXNAM(2),FLXFCT
     1,DELMJ2,TIME,STRFCT,SCALE
     1,TITMOD(20),VFMT(20,8)
       INTEGER OPT,FLAG,OUT,OUT1
       DIMENSION H(NROW,NCOL),HO(NROW,NCOL),T(NROW,NCOL,2),SF1(NROW,NCOL
     1),Q(NROW,NCOL) ,FLAG(NROW,NCOL)
     1,RO(NROW,NCOL),R(NROW,NCOL)
     1,TOPAQ(NROW,NCOL)
        DOUBLE PRECISION BB,CC,W
      ITER=0
   10 CONTINUE
      ITER=ITER+1
      IF (ITER.GT.50) GO TO 280
      K=MOD(ITER,NPARM)+1
      PK=PARM(K)
      E=0.0
C*****************************************************************SOLV1610
C     ROW CALCULATIONS                                            SOLV1610
C*****************************************************************SOLV1610
      DO 180 II=1,NR
      I=II
      IF(MOD(ISTEP+ITER,2).EQ.1)I=NR-I+1
      EE=1.
      IF(I.EQ.1)EE=0.0
      JSTRT=1
   20 DO 30 JJ=JSTRT,NC
      IFL=FLAG(I,JJ)+1
      GO TO (30,40,40,30), IFL
   30 CONTINUE
      GO TO 180
   40 CONTINUE
      JJP1=JJ+1
   50 DO 60 JJJ=JJP1,NC
      IFL=FLAG(I,JJJ)+1
      GO TO (70,60,60,70), IFL
   60 CONTINUE
      JJJ=NC
      JSTRT=NC
      GO TO 80
   70 JJJ=JJJ-1
      JSTRT=JJJ+1
   80 CONTINUE
      AA=0.
      DD=0.
      BB=0.
      IF (JJ.EQ.1) GO TO 90
      IF (FLAG(I,JJ-1).GT.0) GO TO 90
```



```
      BB=T(I,JJ-1,1)
      DD=BB*H(I,JJ-1)
   90 CONTINUE
      DO 150 J=JJ,JJJ
      TT=-AA
      BB=BB+TT
  100 TT=T(I,J,1)
      CC=-TT
      BB=BB+TT
      IF(FLAG(I,J+1).EQ.0)DD=DD+TT*H(I,J+1)
  110 TT=T(I-1,J,2)*EE
      BB=BB+TT
      DD=DD+TT*H(I-1,J)
  120 TT=T(I,J,2)
      BB=BB+TT
      DD=DD+TT*H(I+1,J)
      DD=DD+BB*PK*H(I,J)
      BB=BB*(1.+PK)
  130  RAT=1.
       IFL=FLAG(I,J)+1
       IF(IFL.EQ.2.AND.H(I,J).GT.TOPAQ(I,J)) RAT=1./STRFCT
       IF(IFL.EQ.3.AND.H(I,J).LT.TOPAQ(I,J)) RAT=STRFCT
       DD=DD-Q(I,J)+SF1(I,J)/DELTA*(HO(I,J)-TOPAQ(I,J)*(1.-RAT))
       BB=BB+SF1(I,J)/DELTA*RAT
       HA=R(I,J)
       IF(HA) 135,140,131
  131  IF(H(I,J)-RD(I,J))140,140,132
  132  BB=BB+HA
       DD=DD+HA*RD(I,J)
       GO TO 140
  135  BB=BB-HA
       DD=DD-HA*RD(I,J)
  140 W=BB-AA*B(J-1)
      B(J)=CC/W
      G(J)=(DD-AA*G(J-1))/W
      DD=0.
      BB=0.
  150 AA=-T(I,J,1)
C*****************************************************************SOLV8710
C     RE-ESTIMATE HEADS                                           SOLV8810
C*****************************************************************SOLV8810
      E=E+ABS(H(I,JJJ)-G(JJJ))
      H(I,JJJ)=G(JJJ)
      N=JJJ
  160 N=N-1
      IF (N-JJ+1) 180,180,170
  170 HA=G(N)-B(N)*H(I,N+1)
      E=E+ABS(HA-H(I,N))
      H(I,N)=HA
      GO TO 160
```

```
     2                                       ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,
     3DELMAJ,E,XLGTNM,FLXNAM(2),FLXFCT
     1,DELMJ2,TIME,STRFCT
     1,SCALE,TITMOD(20),VFMT(20,5)
      INTEGER OPT,FLAG,OUT,OUT1
      DIMENSION H(NROW,NCOL),T(NROW,NCOL,2)
      DIMENSION NB(100)
      EQUIVALENCE (E(1),NB(1))
     DO 80 M=1,2
     WRITE(OUT,130)TITMOD,TITLE,ISTEP
     WRITE (OUT,120) FLXNAM,M
     WRITE (OUT,90)
     IST=1
10   IEND=IST+31
     IF(NR.LT.IEND)IEND=NR
     DO 20 I=IST,IEND
20   NB(I)=IEND-I+IST
     WRITE (OUT,110) (NB(I),I=IST,IEND)
     WRITE (OUT,110)
     DO 70 J=1,NC
     DO 60 I=IST,IEND
     L=IEND-I+IST
     NA=0.
     IF (L.EQ.NR.AND.M.EQ.2) GO TO 60
     IF (J.EQ.NC.AND.M.EQ.1) GO TO 60
     GO TO (40,30), M
30   NA=T(L,J,M)=(H(L,J)-H(L+1,J))*FLXFCT+0.5
     GO TO 50
40   NA=T(L,J,M)=(H(L,J)-H(L,J+1))*FLXFCT+0.5
50   IF(NA.LT.0.)NA=NA-1.
60   NB(I)=NA
70   WRITE (OUT,110) (NB(I),I=IST,IEND),J
     IST=IEND+1
     IF (IST.GT.NR) GO TO 80
     WRITE (OUT,100)
     GO TO 10
80   CONTINUE
     RETURN
C
90   FORMAT (5X,'ROWS')
100  FORMAT (1H1)
110   FORMAT (1X,32I4,I3)
120  FORMAT (/' FLOWS (',2A8,' ) IN DIRECTION ',I5/)
  130 FORMAT(1H1,T25,20A4//T25,20A4/T5,'FOR TIME STEP',I5//)
     END
```



```
      SUBROUTINE GETPMP (NROW,NCOL,FLAG,Q,RHG)
```



```
      COMMON /ITCOM/ NR,NC,ISTEP,NPARM,IN,OUT,OUT1,OPT(30),ITER,NSAVE,
     1ISAVE(25),JSAVE(25),KHYD,NCOLS,NCOLS(25),NROWS,NROWS(25)
     2,IN1,IN2,IN3,IN4,IN5,IN6
     1,NSTEPS,NBLK,IRWC(4,80),NSPRG,ISPRG(25),JSPRG(25)
      COMMON /RLCOM/ FMT(20),TITLE(20),DELX(100),DELY(100),PRMITR(10),B
     1(100),G(100),SUMS(15,2),
     2                                       ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,
     3DELMAJ,E,XLGTNM,FLXNAM(2),FLXFCT
     1,DELMJ2,TIME,STRFCT,SCALE
     1,TITMOD(20),VFMT(20,6)
      INTEGER OPT,FLAG,OUT,OUT1
      DIMENSION FLAG(NROW,NCOL),Q(NROW,NCOL),RHG(NROW,NCOL)
     DO 10 I=1,NR
     DO 10 J=1,NC
     RHG(I,J)=0.
10   Q(I,J)=0.
      IF(OPT(10).LT.1) GO TO 11
      REWIND IN2
      READ (IN2) Q,RHG
11    CONTINUE
C*****************************************************************
C     READ PUMPAGE FOR ALL CELLS
C*****************************************************************
      IF (OPT(2).LT.1) GO TO 30
      DO 15 I=1,20
15    FMT(I)=VFMT(I,4)
      IF(OPT(2).LT.5) GO TO 19
     READ (IN,450) FMT
      WRITE (OUT,470) (FMT(I),I=1,10)
19    DO 20 I=1,NR
20   READ (IN,FMT) (Q(I,J),J=1,NC)
C*****************************************************************
C     READ PUMPAGE BY BLOCK
C*****************************************************************
30    IF (OPT(3).LT.1) GO TO 60
     WRITE (OUT,350)
     WRITE (OUT,380)
      DO 38 I=1,20
38    FMT(I)=VFMT(I,5)
      IF(OPT(3).LT.5) GO TO 40
      READ (IN,450) FMT
      WRITE (OUT,470) (FMT(I),I=1,10)
40    READ(IN,FMT) II,III,JJ,JJJ,NA
     IF (II.LT 1) GO TO 60
     WRITE (OUT,380) II,III,JJ,JJJ,NA
     DO 50 I=II,III
     DO 50 J=JJ,JJJ
     IF (FLAG(I,J) GT.2) GO TO 50
     Q(I,J)=NA
50   CONTINUE
```

```
C**********************************************************************   EXEC
      CALL SOLVE(NROW,NCOL,FLAG,H,HO,T,SF1,Q,R,RO,TOPAQ)                  EXEC
  320 CONTINUE                                                            EXEC
      TIM=TIME/DELMJ2                                                     EXEC
  330 CONTINUE                                                            EXEC
      WRITE (OUT,860) TIME,TIM,E,ITER                                     EXEC
      DO 450 J=1,NC                                                       EXEC
      DO 450 I=1,NR                                                       EXEC
      SUMCHD=0.                                                           EXEC
      IFL=FLAG(I,J)+1                                                     EXEC
      GO TO (340,350,380,450), IFL                                        EXEC
C**********************************************************************   EXEC
C     DETERMINE FLOWS WITH CONSTANT HEAD CELLS                            EXEC
C**********************************************************************   EXEC
  340 IF(I.GT.1)SUMCHD=SUMCHD-T(I-1,J,2)*(H(I-1,J)-H(I,J))*DELTA          EXEC
      IF(I.LT.NR)SUMCHD=SUMCHD+T(I,J,2)*(H(I,J)-H(I+1,J))*DELTA           EXEC
      IF(J.GT.1)SUMCHD=SUMCHD-T(I,J-1,1)*(H(I,J-1)-H(I,J))*DELTA          EXEC
      IF(J.LT.NC)SUMCHD=SUMCHD+T(I,J,1)*(H(I,J)-H(I,J+1))*DELTA           EXEC
      IF(SUMCHD.GT.0.)SUMS(6,1)=SUMS(6,1)+SUMCHD                          EXEC
      IF(SUMCHD.LT.0.)SUMS(5,1)=SUMS(5,1)-SUMCHD                          EXEC
      GO TO 450                                                           EXEC
  350 CONTINUE                                                            EXEC
      IF(KSSH.GT.0) HO(I,J)=H(I,J)
       GO TO (450,351,352,450),IFL
  351  HC=AMIN1(H(I,J),TOPAQ(I,J))
       HA=SF1(I,J)*(HC-HO(I,J))
       HB=SF1(I,J )/STRFCT*(H(I,J)-HC)
       GO TO 353
  352  HC=AMAX1(H(I,J),TOPAQ(I,J))
       HA=SF1(I,J)*STRFCT*(H(I,J)-HC)
       HB=SF1(I,J) *(HC-HO(I,J))
  353  IF(HA.GT.0.) SUMS(7,1)=SUMS(7,1)+ HA
       IF(HB.GT.0.) SUMS(17,1)=SUMS(17,1) + HB
       IF(HA.LT.0.) SUMS(8,1)=SUMS(8,1) - HA
       IF(HB.LT.0.) SUMS(18,1) = SUMS(18,1) - HB
      HA=R(I,J)*(H(I,J)-RO(I,J))*DELTA                                    EXEC
      IF(R(I,J)) 380,380,370                                              EXEC
  360 QSUM(I,J)=QSUM(I,J)-HA                                              EXEC
      GO TO 380                                                           EXEC
  370 IF(H(I,J).LT.RO(I,J)) GO TO 360                                     EXEC
      QSUM(I,J)=QSUM(I,J)+HA                                              EXEC
  380 CONTINUE                                                            EXEC
C**********************************************************************   EXEC
C      CHECK FOR CHANGE OF NODE TYPE                                      EXEC
C**********************************************************************   EXEC
      ITYPE=FLAG(I,J)+1                                                   EXEC
      GO TO (420,390,410,430),ITYPE                                       EXEC
  390 IF(H(I,J).LE.TOPAQ(I,J)) GO TO 430                                  EXEC
      IF(H(I,J).LE.SURF(I,J)) GO TO 400                                   EXEC
      HA=RHG(I,J)*PMPFCT                                                  EXEC
```



```
      RHG(I,J)=0.0                                                        EXEC
      Q(I,J)=Q(I,J)+HA                                                    EXEC
      SUMS(10,1)=SUMS(10,1)+HA*(DELMAJ-TIME+TIME1)                        EXEC
      WRITE(OUT,770) I,J,HA,XLGTHM                                        EXEC
C**********************************************************************   EXEC
C      NODE CHANGED FROM WATER TABLE TO ARTESIAN                          EXEC
C**********************************************************************   EXEC
  400 SF1(I,J)=SF1(I,J)/STRFCT                                            EXEC
      WRITE (OUT,750) I,J,H(I,J)                                          EXEC
      FLAG(I,J)=2                                                         EXEC
      GO TO 430                                                           EXEC
  410 IF(H(I,J).GT.TOPAQ(I,J)) GO TO 430                                  EXEC
C**********************************************************************   EXEC
C      NODE CHANGED FROM ARTESIAN TO WATER TABLE                          EXEC
C**********************************************************************   EXEC
      SF1(I,J)=SF1(I,J)*STRFCT                                            EXEC
      WRITE (OUT,760) I,J,H(I,J)                                          EXEC
      FLAG(I,J)=1                                                         EXEC
      GO TO 430                                                           EXEC
  420 IF(H(I,J).GT.TOPAQ(I,J)) GO TO  430                                 EXEC
      THIK(I,J)=H(I,J)-BOTLEL(I,J)                                        EXEC
  430 CONTINUE                                                            EXEC
C**********************************************************************   EXEC
C      IF H IS BELOW BOTTOM ELEVATION, REDUCE PUMPAGE, IF POSSIBLE        EXEC
C**********************************************************************   EXEC
      IF (H(I,J).GT.BOTLEL(I,J)) GO TO 450                                EXEC
      HA=Q(I,J)+RHG(I,J)*PMPFCT                                           EXEC
      Q(I,J)=-RHG(I,J)*PMPFCT                                             EXEC
      IF(HA.LT.1.)HA=0.                                                   EXEC
      SUMS(9,1)=SUMS(9,1)+HA*(DELMAJ-TIME+TIME1)                          EXEC
C**********************************************************************   EXEC
C      SET MINIMUM THICKNESS TO 0.1                                       EXEC
C**********************************************************************   EXEC
      H(I,J)=BOTLEL(I,J)+0.1                                              EXEC
      WRITE (OUT,700) I,J,HA,XLGTHM                                       EXEC
  440 CONTINUE                                                            EXEC
  450 CONTINUE                                                            EXEC
C**********************************************************************   EXEC
C      SUM FLOW FOR THIS TIME STEP                                        EXEC
C**********************************************************************   EXEC
      IF(KQUAL.GT.0) CALL SUMFLO(H,HO,T,R,RO,B,DELTA,DELMAJ,NC,NR,NROW,   EXEC
     1NCOL,2,INB,INE)                                                     EXEC
C**********************************************************************   EXEC
C      PRINT MAP OF FLOWS - MINOR TIME STEP                               EXEC
C**********************************************************************   EXEC
      IF (OPT(12).GT.0) CALL FLUX (NROW,NCOL,H,T)                         EXEC
C**********************************************************************   EXEC
C      INCREASE SIZE OF TIME STEP                                         EXEC
C**********************************************************************   EXEC
      DELTA=DELTA*TIMACL                                                  EXEC
```

```
C       CONVERT PUMPAGE AND RECHARGE UNITS
C***************************************************************************
        DO 340 I=1,NR
        DO 340 J=1,NC
        IF (FLAG(I,J).LT.1.OR.FLAG(I,J).GT.2) GO TO 340
        IF (Q(I,J)) 240,250,250
  240   SUMS(2,1)=SUMS(2,1)-Q(I,J)
        GO TO 260
  250   SUMS(1,1)=SUMS(1,1)+Q(I,J)
  260   IF (RHG(I,J)) 270,280,280
  270   SUMS(4,1)=SUMS(4,1)-RHG(I,J)
        GO TO 290
  280   SUMS(3,1)=SUMS(3,1)+RHG(I,J)
  290   HA=0
        HB=0
        IF (Q(I,J)) 300,310,310
  300   HA=-Q(I,J)
        Q(I,J)=0.
  310   IF (RHG(I,J)) 320,330,330
  320   HB=-RHG(I,J)
        RHG(I,J)=0.
  330   RHG(I,J)=RHG(I,J)+HA
        Q(I,J)=(Q(I,J)+HB-RHG(I,J))*PMPFCT
  340   CONTINUE
C***************************************************************************
C       RETURN
C***************************************************************************
        RETURN
C
  350   FORMAT (//T30,'BLOCK  PUMPAGE  ASSIGNMENT'/)
  360   FORMAT (/T21,'ROW     ROW  COLUMN  COLUMN'T56,'VALUE'/T21,'START  E
       1ND  START     END'//)
  370   FORMAT (//T30,'BLOCK  RECHARGE  ASSIGNMENT')
  371    FORMAT(//T30,'PER UNIT AREA  -  (LENGTH PER TIME STEP)')
  380    FORMAT (T21,I3,4X,I3,3X,I3,5X,I3,G19.4)
  390   FORMAT (//T30,'BLOCK PUMPAGE ADJUSTMENTS')
  400   FORMAT (//T30,'BLOCK RECHARGE ADJUSTMENTS')
    410 FORMAT(1H1,T25,20A4//T25,20A4/)
  420    FORMAT (1H0,T21,'PUMPAGE FOR TIME STEP',I5/)
  430   FORMAT (/I5,10F10.3/(5X,10F10.3))
  440    FORMAT (1H0,T21,'RECHARGE FOR TIME STEP',I5/)
  450   FORMAT (20A4)
  460   FORMAT (4I5,F10.0)
  470    FORMAT (T70,'FORMAT IS',T80,10A4/)
        END
```



```
        SUBROUTINE HYDRO (NROW,NCOL,FLAG,HSIM,H,H1)
```



```
         COMMON /ITCOM/ NR,NC,ISTEP,NPARM,IN,OUT,OUT1,OPT(30),ITER,NSAVE,
       1ISAVE(25),JSAVE(25),KHYD,NCOLS,MCOLS(25),NROWS,MROWS(25)
       2,IN1,IN2,IN3,IN4,IN5,IN6
       1,NSTEPS,NBLK,IRWC(4,60),NSPRG,ISPRG(25),JSPRG(25)
         COMMON /RLCOM/ FMT(20),TITLE(20),DELX(100),DELY(100),PRMITR(10),B
       1(100),C(100),SUMS(15,2),
       2                                ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,
       3DELMAJ,E,XLGTHM,FLXNAM(2),FLXFCT
       1,DELMJ2,TIME,STRFCT,SCALE
       1,TITMOD(20),VFMT(20,6)
         INTEGER OPT,FLAG,OUT,OUT1
         DIMENSION FLAG(NROW,NCOL),HSIM(NROW,NCOL),H(NROW,NCOL),H1(NROW,NC
       1OL),IX(21)
C***************************************************************************
C       THIS SUBROUTINE PRODUCES A HYDROGRAPH FOR SPECIFIED NODES.
C       FOR ARRAYS H AND H1, FIRST SUBSCRIPT INDICATES THE NUMBER OF THE
C       SPECIFIED NODE AND THE SECOND SUBSCRIPT INDICATES THE TIME PERIOD
C***************************************************************************
        REAL MAP(21)
        DATA KAT,IX(1),IXX/0,0,1/
        NYR=NSTEPS
        REWIND IN3
        DATA BLANK,SIM,OBS,XINC/1H ,1HS,1HO,5.0/,BOTH/1HB/
        READ (IN3) HSIM
        IX(1)=1
        DO 10 K=1,NSAVE
        I=ISAVE(K)
        J=JSAVE(K)
        H(K,1)=-1.E5
   10   H1(K,1)=HSIM(I,J)
   20   NYRS=NYR-20
C***************************************************************************
        IF (KAT.GT.0.AND.NYRS.LE.-20) RETURN
C***************************************************************************
        IF(NYRS.LE.0)NYRS=NYR
        NSTOP=0
        IF(KAT.LT.1)NSTOP=1
        IF (NYRS.EQ.NYR) GO TO 30
        NYRS=20
   30   CONTINUE
        NYR=NYR-NYRS
        DO 60 N=1,NYRS
        IK=IXX+N
        READ (IN3) HSIM
        NSTOP=NSTOP+1
        READ (IN3) IOPT
        IX(IK)=IOPT
        DO 40 K=1,NSAVE
        I=ISAVE(K)
        J=JSAVE(K)
```

```
     2                                          ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,     OUTPUT
     3DELMAJ,E,XLGTNM,FLXNAM(2),FLXFCT                                                OUTPUT
     1,DELMJ2,TIME,STAPCT,SCALE                                                       OUTPUT
     1,TITMOD(20),YFMT(20,8)
      INTEGER OPT,FLAG,OUT,OUT1                                                       OUTPUT
      DIMENSION FLAG(NROW,NCOL),H(NROW,NCOL),HO(NROW,NCOL)                            OUTPUT
     1,P(NROW,NCOL,2),BOTLEL(NROW,NCOL),SF1(NROW,NCOL),                               OUTPUT
     2THIK(NROW,NCOL),TOPAQ(NROW,NCOL),QSUM(NROW,NCOL)                                OUTPUT
     2,SURF(NROW,NCOL)                                                                OUTPUT
     2,T(NROW,NCOL,2)                                                                 OUTPUT
     1,R(NROW,NCOL)                                                                   OUTPUT
C*********************************************************************                OUTPUT
C      SAVE HEADS FOR HYDROGRAPH ROUTINE                                              OUTPUT
C*********************************************************************                OUTPUT
      IF (KHYD.LT.1) GO TO 10                                                         OUTPUT
C*********************************************************************                OUTPUT
C     SAVE SIMULATED HEAD FOR HYDROGRAPH ROUTINE                                      OUTPUT
C*********************************************************************                OUTPUT
      WRITE (IN3) H                                                                   OUTPUT
      WRITE (IN3) OPT(22)                                                             OUTPUT
   10 CONTINUE                                                                        OUTPUT
C*********************************************************************                OUTPUT
C     PREFORM MASS BALANCE COMPUTATIONS                                               OUTPUT
C*********************************************************************                OUTPUT
      DLT2J=DELMAJ/DELMJ2                                                             OUTPUT
      DO 40 I=1,NR                                                                    OUTPUT
      DO 40 J=1,NC                                                                    OUTPUT
      HA=QSUM(I,J)                                                                    OUTPUT
      IF(R(I,J)) 20,40,30                                                             OUTPUT
   20 IF(HA.LT.0.0) SUMS(15,1)=SUMS(15,1) - HA                                        OUTPUT
      IF(HA.GT.0.0) SUMS(14,1)=SUMS(14,1) + HA                                        OUTPUT
      GO TO 40                                                                        OUTPUT
   30 SUMS(11,1)= SUMS(11,1) + HA                                                     OUTPUT
   40 CONTINUE                                                                        OUTPUT
      DO 90 K=1,NSPRG                                                                 OUTPUT
      IF(NSPRG.LT.1) GO TO 90                                                         OUTPUT
      I=ISPRG(K)                                                                      OUTPUT
      J=JSPRG(K)                                                                      OUTPUT
      HA=QSUM(I,J)                                                                    OUTPUT
      HB=HA/PMPFCT*DLT2J                                                              OUTPUT
     1/DELMAJ                                                                         OUTPUT
      IF(R(I,J)) 50,90,50                                                             OUTPUT
   50 IF(HA) 70,80,60                                                                 OUTPUT
   60 SUMS(12,1)=SUMS(12,1)+ HA                                                       OUTPUT
      SUMS(14,1)=SUMS(14,1) - HA                                                      OUTPUT
      WRITE (OUT,580) I,J,H(I,J),HB,PMPNAM                                            OUTPUT
      GO TO 90                                                                        OUTPUT
   70 SUMS(13,1)=SUMS(13,1) - HA                                                      OUTPUT
      SUMS(15,1)=SUMS(15,1) + HA                                                      OUTPUT
      HB=-HB                                                                          OUTPUT
                                                                                      OUTPUT
```



```
      WRITE (OUT,600) I,J,H(I,J),HB,PMPNAM
      GO TO 90                                                                        OUTPUT
   80 WRITE (OUT,590) I,J,H(I,J),HB,PMPNAM                                            OUTPUT
   90 CONTINUE                                                                        OUTPUT
      WRITE(OUT,830) TITMOD,TITLE                                                     OUTPUT
      WRITE (OUT,450) ISTEP,ISTEP,XLGTNM,PMPNAM,XLGTNM,PMPNAM
      SUMS(16,1)=SUMS(6,1)-SUMS(7,1)+ PMPFCT*DELMAJ*(SUMS(2,1)-SUMS(1,1               OUTPUT
     1)+SUMS(3,1)-SUMS(4,1))+SUMS(6,1)-SUMS(5,1)+SUMS(9,1)-SUMS(10,1)                 OUTPUT
     2-SUMS(11,1)-SUMS(12,1)+SUMS(13,1)-SUMS(14,1)+SUMS(15,1)                         OUTPUT
     3+SUMS(16,1)-SUMS(17,1)                                                          OUTPUT
      DO 100 K=1,4
  100 SUMS(K,1)=SUMS(K,1)*DLT2J                                                       OUTPUT
      DO 110 K=1,16                                                                   OUTPUT
  110 SUMS(K,2)=SUMS(K,2)+SUMS(K,1)                                                   OUTPUT
      DO 120 K=5,18                                                                   OUTPUT
  120 SUMS(K,1)=SUMS(K,1)/DELMAJ                                                      OUTPUT
      DO 130 K=1,3,2                                                                  OUTPUT
      B(1)=SUMS(K,1)*PMPFCT/DLT2J                                                     OUTPUT
      B(2)=SUMS(K,2)*PMPFCT*DELMJ2/TIME                                               OUTPUT
      B(3)=SUMS(K+1,1)*PMPFCT/DLT2J                                                   OUTPUT
      B(4)=SUMS(K+1,2)*PMPFCT*DELMJ2/TIME                                             OUTPUT
      B(8)=SUMS(K,1)-SUMS(K+1,1)                                                      OUTPUT
      B(5)=B(1)-B(3)                                                                  OUTPUT
      B(6)=SUMS(K,2)-SUMS(K+1,2)                                                      OUTPUT
      B(7)=B(2)-B(4)                                                                  OUTPUT
      IF (K.EQ.1) WRITE (OUT,490)                                                     OUTPUT
      IF (K.EQ.3) WRITE (OUT,500)                                                     OUTPUT
      WRITE (OUT,510) (B(L),SUMS(K,L),L=1,2),(B(L+2),SUMS(K+1,L),L=1,2),              OUTPUT
     1(B(L),L=5,8)                                                                    OUTPUT
  130 CONTINUE                                                                        OUTPUT
      DO 140 K=5,7,2                                                                  OUTPUT
      B(1)=SUMS(K,1)/PMPFCT*DLT2J                                                     OUTPUT
      HA=SUMS(K,2)/TIME                                                               OUTPUT
      B(2)=HA/PMPFCT*TIME/DELMJ2                                                      OUTPUT
      B(3)=SUMS(K+1,1)/PMPFCT*DLT2J                                                   OUTPUT
      HB=SUMS(K+1,2)/TIME                                                             OUTPUT
      B(4)=HB/PMPFCT*TIME/DELMJ2                                                      OUTPUT
      B(5)=SUMS(K,1)-SUMS(K+1,1)                                                      OUTPUT
      B(6)=B(1)-B(3)                                                                  OUTPUT
      B(7)=HA-HB                                                                      OUTPUT
      B(8)=B(2)-B(4)                                                                  OUTPUT
      IF (K.EQ.5) WRITE (OUT,520) SUMS(K,1),B(1),HA,B(2),SUMS(K+1,1),B(3              OUTPUT
     1),HB,B(4),(B(L),L=5,8)                                                          OUTPUT
      IF (K.EQ.7) WRITE (OUT,530) SUMS(K,1),B(1),HA,B(2),SUMS(K+1,1),B(3              OUTPUT
     1),HB,B(4),(B(L),L=5,8)                                                          OUTPUT
  140 CONTINUE                                                                        OUTPUT
      B(1)=SUMS(17,1)/PMPFCT*DLT2J                                                    OUTPUT
      HA=SUMS(17,2)/TIME
      B(2)=HA/PMPFCT*TIME/DELMJ2
      B(3)=SUMS(18,1)/PMPFCT*DLT2J
```

```
      IF (OPT(20).LT.1.AND.OPT(21).LT.1) GO TO 310                     OUTPUT
C*****************************************************************    OUTPUT
C      READ INITIAL WATER LEVELS                                       OUTPUT
C*****************************************************************    OUTPUT
      REWIND IN1                                                       OUTPUT
      READ (IN1) HO                                                    OUTPUT
      IF (OPT(20).LT.1) GO TO 300                                      OUTPUT
C*****************************************************************    OUTPUT
C     PRINT HEAD CHANGES THROUGH THIS TIME STEP                        OUTPUT
C*****************************************************************    OUTPUT
      WRITE(OUT,830) TITMOD,TITLE                                      OUTPUT
      WRITE (OUT,710) ISTEP                                            OUTPUT
      DO 290 I=1,NR                                                    OUTPUT
      DO 280 J=1,NC                                                    OUTPUT
  280 B(J)=H(I,J)-HO(I,J)                                              OUTPUT
  290 WRITE (OUT,650) I,(B(J),J=1,NC)                                  OUTPUT
  300 CONTINUE                                                         OUTPUT
C*****************************************************************    OUTPUT
C       PLOT WATER LEVELS CHANGES THROUGH THIS TIME STEP               OUTPUT
C*****************************************************************    OUTPUT
      IF (OPT(21).GT.0) CALL PLOTS (NROW,NCOL,FLAG,H,HO,3)             OUTPUT
  310 CONTINUE                                                         OUTPUT
      IF (OPT(22).LT.1) GO TO 400                                      OUTPUT
  320 CONTINUE                                                         OUTPUT
C*****************************************************************    OUTPUT
C      READ MEASURED WATER LEVEL DATA                                  OUTPUT
C*****************************************************************    OUTPUT
      DO 330 I=1,20                                                    OUTPUT
  330 FMT(I)=YFMT(I,4)                                                 OUTPUT
      IF(OPT(22).LT.5) GO TO 340                                       OUTPUT
      READ (IN,890) FMT                                                OUTPUT
      WRITE (OUT,900) (FMT(I),I=1,10)                                  OUTPUT
  340 DO 350 I=1,NR                                                    OUTPUT
      READ (IN,FMT) (HO(I,J),J=1,NC)                                   OUTPUT
C*****************************************************************    OUTPUT
C      LIST SIMULATED AND MEASURED WATER LEVELS                        OUTPUT
C      AND SIMULATION ERRORS                                           OUTPUT
C*****************************************************************    OUTPUT
  350 CONTINUE                                                         OUTPUT
      WRITE(OUT,830) TITMOD,TITLE                                      OUTPUT
      WRITE (OUT,840) ISTEP                                            OUTPUT
      DO 390 I=1,NR                                                    OUTPUT
      DO 370 J=1,NC                                                    OUTPUT
      B(J)=0.                                                          OUTPUT
      IFL=FLAG(I,J)+1                                                  OUTPUT
      GO TO (370,360,360,370), IFL                                     OUTPUT
  360 B(J)=H(I,J)-HO(I,J)                                              OUTPUT
  370 CONTINUE                                                         OUTPUT
      JST=1                                                            OUTPUT
  380 JEND=JST+9
```



```
      JEND=MIN0(JEND,NC)                                               OUTPUT
      WRITE (OUT,850)                                                  OUTPUT
      WRITE (OUT,860) I,(H(I,J),J=JST,JEND)                            OUTPUT
      WRITE (OUT,870) (HO(I,J),J=JST,JEND)                             OUTPUT
      WRITE (OUT,870) (B(J),J=JST,JEND)                                OUTPUT
      JST=JEND+1                                                       OUTPUT
      IF (JST.GT.NC) GO TO 390                                         OUTPUT
      GO TO 380                                                        OUTPUT
  390 CONTINUE                                                         OUTPUT
C*****************************************************************    OUTPUT
C     PRINT MAP OF SIMULATION ERRORS                                   OUTPUT
C*****************************************************************    OUTPUT
      CALL PLOTS (NROW,NCOL,FLAG,H,HO,1)                               OUTPUT
C*****************************************************************    OUTPUT
C     SAVE HEADS FOR HYDROGRAPH ROUTINE                                OUTPUT
C*****************************************************************    OUTPUT
      WRITE (IN3) HO                                                   OUTPUT
  400 CONTINUE                                                         OUTPUT
C*****************************************************************    OUTPUT
C     PRINT CROSS-SECTIONS                                             OUTPUT
C*****************************************************************    OUTPUT
      IF (OPT(23).LT.1) GO TO 430                                      OUTPUT
      IF(OPT(23).NE.2) GO TO 410                                       OUTPUT
      OPT(22)=1                                                        OUTPUT
      REWIND IN1                                                       OUTPUT
      READ (IN1) HO                                                    OUTPUT
      GO TO 420                                                        OUTPUT
  410 IF(OPT(23).NE.3) GO TO 430                                       OUTPUT
      OPT(22)=1                                                        OUTPUT
      REWIND IN4                                                       OUTPUT
      READ (IN4) HO                                                    OUTPUT
  420 CONTINUE                                                         OUTPUT
      CALL XSECT (NROW,NCOL,FLAG,H,HO,BOTLEL)                          OUTPUT
  430 CONTINUE                                                         OUTPUT
C*****************************************************************    OUTPUT
C      LIST AND PLOT SATURATED THICKNESS                               OUTPUT
C*****************************************************************    OUTPUT
      IF (OPT(25).LT.1) GO TO 470                                      OUTPUT
      IF(OPT(25).EQ.3) GO TO 460
      WRITE(OUT,830) TITMOD,TITLE                                      OUTPUT
      WRITE (OUT,850) ISTEP                                            OUTPUT
      DO 450 I=1,NR                                                    OUTPUT
      DO 440 J=1,NC                                                    OUTPUT
      HO(I,J)=THIK(I,J)                                                OUTPUT
      IF(FLAG(I,J).EQ.3) HO(I,J)=0.0                                   OUTPUT
  440 IF(FLAG(I,J).EQ.1) HO(I,J)=H(I,J)-BOTLEL(I,J)                    OUTPUT
  450 WRITE (OUT,650) I,(HO(I,J),J=1,NC)                               OUTPUT
      IF(OPT(25).LT.2) GO TO 470                                       OUTPUT
  460 CALL PLOTH(NROW,NCOL,FLAG,H,THIK,TOPAQ,BOTLEL,SF1,2)             OUTPUT
  470 CONTINUE
```

```
      IF (OPT(I).GT.0) WRITE (OUT,690) I,OPT(I)                          PHYSDT
   10 CONTINUE                                                           PHYSDT
      KSSH=OPT(14)                                                       PHYSDT
      KQUAL=OPT(15)                                                      PHYSDT
C*********************************************************************** PHYSDT
C     IF HYDROGRAPHS ARE REQUIRED, READ NUMBER OF AND COORDINATES OF     PHYSDT
C     THE SPECIFIC NODES.                                                PHYSDT
C*********************************************************************** PHYSDT
      IF (OPT(1).LT.1) GO TO 20                                          PHYSDT
      KHYD=OPT(1)                                                        PHYSDT
      READ (IN,860) NSAVE,(ISAVE(I),JSAVE(I),I=1,NSAVE)                  PHYSDT
   20 CONTINUE                                                           PHYSDT
C*********************************************************************** PHYSDT
C     IF CROSS-SECTIONS ARE REQUESTED, READ NUMBER OF AND INDEX FOR THE  PHYSDT
C     REQUESTED COLUMNS AND ROWS, RESPECTIVELY.                          PHYSDT
C*********************************************************************** PHYSDT
      IF (OPT(2).LT.1) GO TO 30                                          PHYSDT
      READ (IN,860) NCOLS,(MCOLS(I),I=1,NCOLS)                           PHYSDT
      READ (IN,860) NROWS,(MROWS(I),I=1,NROWS)                           PHYSDT
   30 CONTINUE                                                           PHYSDT
C*********************************************************************** PHYSDT
C     READ GRID SPACINGS IN THE X AND Y DIMENSIONS, RESPECTIVELY.        PHYSDT
C*********************************************************************** PHYSDT
      DO 40 K=1,20                                                       PHYSDT
   40 FMT(K)=VFMT(K,1)                                                   PHYSDT
      IF(OPT(3).LT.5) GO TO 50                                           PHYSDT
      OPT(3)=OPT(3)-5                                                    PHYSDT
      READ (IN,870) FMT                                                  PHYSDT
      WRITE (OUT,740) (FMT(K),K=1,10)                                    PHYSDT
   50 IF(OPT(3).LT.1) GO TO 80                                           PHYSDT
C*********************************************************************** PHYSDT
C     READ AND WRITE CONSTANT GRID SPACINGS                              PHYSDT
C*********************************************************************** PHYSDT
      READ (IN,FMT) HA,HB                                                PHYSDT
      DO 60 I=1,NR                                                       PHYSDT
   60 DELY(I)=HB                                                         PHYSDT
      DO 70 J=1,NC                                                       PHYSDT
   70 DELX(J)=HA                                                         PHYSDT
      WRITE (OUT,730) HA,HB                                              PHYSDT
      GO TO 90                                                           PHYSDT
   80 CONTINUE                                                           PHYSDT
      READ (IN,FMT) (DELX(J),J=1,NC)                                     PHYSDT
      READ (IN,FMT) (DELY(I),I=1,NR)                                     PHYSDT
      IF (OPT(4).LT.1) GO TO 90                                          PHYSDT
      WRITE (OUT,690)                                                    PHYSDT
      WRITE (OUT,800) (DELX(J),J=1,NC)                                   PHYSDT
      WRITE (OUT,700)                                                    PHYSDT
      WRITE (OUT,800) (DELY(I),I=1,NR)                                   PHYSDT
   90 CONTINUE                                                           PHYSDT
      IF(SCALE.GT.-1.E-3) GO TO 120                                      PHYSDT
```



```
      SCALE=1.E6                                                         PHYSDT
      DO 100 I=1,NR                                                      PHYSDT
  100 SCALE=AMIN1(DELY(I)*5.,SCALE)                                      PHYSDT
      DO 110 J=1,NC                                                      PHYSDT
  110 SCALE=AMIN1(DELX(J)*10.,SCALE)                                     PHYSDT
  120 WRITE (OUT,810) SCALE,XLGTHM                                       PHYSDT
C*********************************************************************** PHYSDT
C     READ PHYSICAL DATA FORMAT CARD                                     PHYSDT
C*********************************************************************** PHYSDT
      DO 130 K=1,20                                                      PHYSDT
  130 FMT(K)=VFMT(K,2)                                                   PHYSDT
      IF(OPT(5).LT.5) GO TO 140                                          PHYSDT
      OPT(5)=OPT(5)-5                                                    PHYSDT
      READ (IN,870) FMT                                                  PHYSDT
      WRITE (OUT,740) (FMT(K),K=1,10)                                    PHYSDT
  140 IF(OPT(5).LT.1) GO TO 160                                          PHYSDT
C*********************************************************************** PHYSDT
C     READ AND WRITE DEFAULT VALUES TO BE ASSIGNED TO ALL NODES          PHYSDT
C*********************************************************************** PHYSDT
      READ (IN,FMT) K,(B(N),N=1,8)                                       PHYSDT
      IF(K.EQ.2) B(8)=B(8)/1.E6                                          PHYSDT
      WRITE (OUT,720) K,(B(N),N=1,8)                                     PHYSDT
      DO 150 J=1,NC                                                      PHYSDT
      DO 150 I=1,NR                                                      PHYSDT
      FLAG(I,J)=K                                                        PHYSDT
      SURF(I,J)=B(1)                                                     PHYSDT
      TOPAQ(I,J)=B(2)                                                    PHYSDT
      BOTLEL(I,J)=B(3)                                                   PHYSDT
      THIK(I,J)=B(4)                                                     PHYSDT
      H(I,J)=B(5)                                                        PHYSDT
      P(I,J,1)=B(6)                                                      PHYSDT
      P(I,J,2)=B(7)                                                      PHYSDT
  150 SF1(I,J)=B(8)                                                      PHYSDT
      GO TO 180                                                          PHYSDT
  160 CONTINUE                                                           PHYSDT
C*********************************************************************** PHYSDT
C     READ NODE CARDS                                                    PHYSDT
C*********************************************************************** PHYSDT
      DO 170 I=1,NR                                                      PHYSDT
      DO 170 J=1,NC                                                      PHYSDT
      READ (IN,FMT) FLAG(I,J),SURF(I,J),TOPAQ(I,J),BOTLEL(I,J),THIK(I,J  PHYSDT
     1),H(I,J),P(I,J,1),P(I,J,2),SF1(I,J)                                PHYSDT
      IF(FLAG(I,J).EQ.2) SF1(I,J)=SF1(I,J)/1.E6                          PHYSDT
  170 CONTINUE                                                           PHYSDT
  180 CONTINUE                                                           PHYSDT
      IF (OPT(6).LT.1) GO TO 230                                         PHYSDT
      DO 190 K=1,20                                                      PHYSDT
  190 FMT(K)=VFMT(K,3)                                                   PHYSDT
      IF(OPT(6).LT.5) GO TO 200                                          PHYSDT
      READ (IN,870) FMT                                                  PHYSDT
```

PHYSDT

```
  370 FMT(N)=VFMT(N,9)
      WRITE(OUT,765)
      WRITE(OUT,740)(FMT(K),K=1,10)
      DO 400 N=1,NSPRG
      IF(NSPRG.LT.1) GO TO 400
      READ (IN,FMT) I,J,RO(I,J),R(I,J)
      HA=-R(I,J)
      IF(HA.GT.0.0) GO TO 380
      WRITE (OUT,850) I,J,RO(I,J),R(I,J)
     1,PMPNAM,XLGTNM
      GO TO 390
  380 WRITE (OUT,880) I,J,RO(I,J),HA
     1,PMPNAM,XLGTNM
  390 R(I,J)=R(I,J)*PMPFCT
      ISPRG(N)=I
  400 JSPRG(N)=J
  410 CONTINUE
      IF (KHYD.LT.1) GO TO 420
C*********************************************************************
C      SAVE ORIGINAL HEADS FOR HYDROGRAPH ROUTINE
C*********************************************************************
      REWIND IN3
      WRITE (IN3) H
  420 CONTINUE
C*********************************************************************
C       SAVE INITIAL WATER LEVELS
C*********************************************************************
      REWIND IN1
      WRITE (IN1) H
C*********************************************************************
C       CHECK INPUT DATA
C*********************************************************************
      DO 470 I=1,NR
      DO 470 J=1,NC
      IF(FLAG(I,J).GT.2) GO TO 470
      IF(H(I,J).GT.BOTLEL(I,J)) GO TO 430
      H(I,J)=BOTLEL(I,J)+0.1
      WRITE( OUT,810) I,J,H(I,J)
  430 IF(TOPAQ(I,J).LE.SURF(I,J)) GO TO 440
      TOPAQ(I,J)=SURF(I,J)
      WRITE (OUT,820) I,J,TOPAQ(I,J)
  440 IF(FLAG(I,J).EQ.1) GO TO 450
      IF(FLAG(I,J).LT.1) GO TO 460
      IF(H(I,J).GT.TOPAQ(I,J)) GO TO 460
      FLAG(I,J)=1
        SF1(I,J)=SF1(I,J)*STRFCT
      WRITE (OUT,830) I,J
      GO TO 460
  450 IF(H(I,J).LE.TOPAQ(I,J)) GO TO 460
      IF(FLAG(I,J).LT.1) GO TO 460
```


PHYSDT

```
      FLAG(I,J)=2
        SF1(I,J)=SF1(I,J)/STRFCT
      WRITE (OUT,840) I,J
  460 CONTINUE
  470 CONTINUE
C*********************************************************************
C       ZERO PERMEABILITIES FOR BOUNDARY CELLS
C*********************************************************************
      DO 490 I=1,NR
      DO 490 J=1,NC
      IF(FLAG(I,J).EQ.3) P(I,J,1)=0.0
      IF(FLAG(I,J).EQ.3) P(I,J,2)=0.0
      IF(I.EQ.NR) GO TO 480
      IF(FLAG(I+1,J).EQ.3) P(I,J,2)=0.0
      IF(FLAG(I,J).EQ.0.AND.FLAG(I+1,J).EQ.0) P(I,J,2)=0.0
  480 IF(J.EQ.NC) GO TO 490
      IF(FLAG(I,J+1).EQ.3) P(I,J,1)=0.0
      IF(FLAG(I,J).EQ.0.AND.FLAG(I,J+1).EQ.0) P(I,J,1)=0.0
  490 CONTINUE
C*********************************************************************
C      WRITE PHYSICAL PARAMETERS
C*********************************************************************
      IF (OPT(8).LT.1) GO TO 510
      DO 500 I=1,NR
      WRITE(OUT,680)TITMOD,TITLE
      WRITE (OUT,640)
      DO 500 J=1,NC
      IFLG=FLAG(I,J)
      IF(IFLG.EQ.0)IFLG=4
      WRITE (OUT,650) TYPE(1,IFLG),TYPE(2,IFLG),I,J,FLAG(I,J),
     1SURF(I,J),TOPAQ(I,J),BOTLEL(I,J),THIK(I,J),H(I,J),P(I,J,1),
     2P(I,J,2),SF1(I,J)
  500 CONTINUE
  510 CONTINUE
C*********************************************************************
C       CONVERT UNITS
C*********************************************************************
      DO 520 I=1,NR
      DO 520 J=1,NC
      P(I,J,1)=P(I,J,1)*PERFCT
      P(I,J,2)=P(I,J,2)*PERFCT
      IF(KSSH.GT.0) SF1(I,J)=0.0
  520 SF1(I,J)=SF1(I,J)*DELX(J)*DELY(I)
C*********************************************************************
C       PLOT INITIAL WATER LEVELS
C*********************************************************************
      IF(OPT(9).GT.0) CALL PLOTH(NROW,NCOL,FLAG,H,THIK,TOPAQ,BOTLEL,SF1
     1,1)
C*********************************************************************
C       LIST AND PLOT INITIAL SATURATED THICKNESS
```

```
C       XINCRE EQUALS THE RANGE FOR EACH PRINTER SYMBOL                  PLOTH
C*****************************************************************       PLOTH
        DATA XINCRE,BLANK,IFIRST/10.0,4H    ,0/                          PLOTH
        DATA SYMBOL/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL,     PLOTH
       11HM,1HN,1HO,1HP,1HQ,1HR,1HS,1HT,1HU,1HV,1HW,1HX,1HY,1HZ/         PLOTH
        DIMENSION IDUM(10)                                               PLOTH
        DATA IDUM/1,2,3,4,5,6,7,8,9,0/                                   PLOTH
        DIMENSION XROW(3), XCOL(2)                                       PLOTH
        DATA XROW/1HR,1HO,1HW/,XCOL/4HCOLU,4HMN  /                       PLOTH
        ENTRY OPLOTH (NROW,NCOL,FLAG,H,OTITLE,ICO)                       PLOTH
        DIMENSION OTITLE(20)                                             PLOTH
        HA=0                                                             PLOTH
        HB=-1.E-6                                                        PLOTH
        HC=1.E6                                                          PLOTH
        M=0                                                              PLOTH
        DO 20 I=1,NR                                                     PLOTH
        DO 20 J=1,NC                                                     PLOTH
        IF(FLAG(I,J).GT.2) GO TO 20                                      PLOTH
        IF(FLAG(I,J).EQ.0) GO TO 20                                      PLOTH
        M=M+1                                                            PLOTH
        HD=H(I,J)                                                        PLOTH
        IF(ICO.NE.2) GO TO 10                                            PLOTH
        HD=THIK(I,J)                                                     PLOTH
        IF(FLAG(I,J).EQ.1) HD=H(I,J)-BOTLEL(I,J)                         PLOTH
     10 HA=HA+HD                                                         PLOTH
        HB=AMAX1(HB,HD)                                                  PLOTH
        HC=AMIN1(HC,HD)                                                  PLOTH
     20 CONTINUE                                                         PLOTH
        HA=HA/M                                                          PLOTH
        ERR=HC                                                           PLOTH
        HB=(HB-HC)/24                                                    PLOTH
        HC=0.1                                                           PLOTH
        IF(HB.GT.0.1) HC=1.                                              PLOTH
        IF(HB.GT.1.) HC=5.                                               PLOTH
        IF(HB.GT.5.) HC=10.                                              PLOTH
        IF(HB.GT.10.) HC=25.                                             PLOTH
        IF(HB.GT.25.) HC=50.                                             PLOTH
        IF(HB.GT.50.) HC=100.                                            PLOTH
        HA=IFIX(HA/HC)*HC                                                PLOTH
        HB=IFIX(ERR/HC)*HC                                               PLOTH
        XRANGE(1)=-1.E6                                                  PLOTH
        XRANGE(27)=1.E6                                                  PLOTH
        ERR=HA-HC*12                                                     PLOTH
        XRANGE(2)=AMAX1(ERR,HB)                                          PLOTH
        DO 30 I=3,26                                                     PLOTH
     30 XRANGE(I)=XRANGE(2)+HC*(I-2)                                     PLOTH
        ORIGX=0.                                                         PLOTH
        IF(SCALE.GT.0.) GO TO 50                                         PLOTH
        DO 40 J=1,NC                                                     PLOTH
     40 G(J)=J                                                           PLOTH
```



```
        JST=1                                                            PLOTH
        J=NC+1                                                           PLOTH
        GO TO 100                                                        PLOTH
     50 CONTINUE                                                         PLOTH
        B(1)=0                                                           PLOTH
        DO 60 I=2,NR                                                     PLOTH
     60 B(I)=B(I-1)+(DELY(I)+DELY(I-1))/2.                               PLOTH
        G(1)=0.                                                          PLOTH
        DO 70 J=2,NC                                                     PLOTH
     70 G(J)=G(J-1)+(DELX(J)+DELX(J-1))/2.                               PLOTH
        JST=1                                                            PLOTH
        IF(G(NC)/SCALE.GT.50.) GO TO 310                                 PLOTH
        IF(B(NR)/SCALE.GT.50.) GO TO 310                                 PLOTH
     80 DO 90 J=JST,NC                                                   PLOTH
        IF((G(J)-ORIGX).GT.9.9*SCALE) GO TO 100                          PLOTH
     90 CONTINUE                                                         PLOTH
        J=NC+1                                                           PLOTH
    100 JEND=J-1                                                         PLOTH
        GSAVE=G(JEND)                                                    PLOTH
        IF(JST.EQ.JEND.AND.JST.NE.NC) GO TO 310                          PLOTH
        WRITE(OUT,540)TITMOD,TITLE
        IF(ICO.EQ.3) WRITE (OUT,550) OTITLE                              PLOTH
        IF(ICO.EQ.1) WRITE (OUT,460) ISTEP                               PLOTH
        IF(ICO.EQ.2) WRITE(OUT,470) ISTEP                                PLOTH
        IF(ICO.EQ.3) WRITE (OUT,480) ISTEP                               PLOTH
        WRITE (OUT,660)                                                  PLOTH
        IF(SCALE.LE.1.E-3) GO TO 120                                     PLOTH
        DO 110 J=JST,JEND                                                PLOTH
        K=(G(J)+SCALE/20.-ORIGX)/(SCALE/10.)+1                           PLOTH
    110 G(J)=K+0.5                                                       PLOTH
    120 WRITE (OUT,670)  XCOL                                            PLOTH
        DO 150 L=1,2                                                     PLOTH
        DO 130 K=1,100                                                   PLOTH
    130 PLOTS(K)=BLANK                                                   PLOTH
        DO 140 J=JST,JEND                                                PLOTH
        KG=G(J)                                                          PLOTH
        KL=J/10                                                          PLOTH
        PLOTS(KG)=XINTGR(KL)                                             PLOTH
        IF(KL.EQ.0) PLOTS(KG)=BLANK                                      PLOTH
        IF(L.EQ.1) GO TO 140                                             PLOTH
        KL=MOD(J,10)                                                     PLOTH
        IF(KL.EQ.0) KL=10                                                PLOTH
        PLOTS(KG)=XINTGR(KL)                                             PLOTH
    140 CONTINUE                                                         PLOTH
    150 WRITE (OUT,560) PLOTS                                            PLOTH
        WRITE (OUT,530)                                                  PLOTH
        WRITE (OUT,580)                                                  PLOTH
        YOTS=B(1)                                                        PLOTH
        DO 270 I=1,NR                                                    PLOTH
        DO 160 K=1,100                                                   PLOTH
```

```
      WRITE (OUT,510) SYMBOL(I),XRANGE(I),XRANGE(I+1),IFREQ(I),PERCT(I)      PLOTH
      IF(ICO.EQ.2) WRITE (OUT,580) TOT1(I),TOT(I),TOT2(I)                    PLOTH
      TOT4=TOT4+TOT2(I)                                                      PLOTH
      TOT3=TOT3+TOT1(I)                                                      PLOTH
  640 TOT5=TOT5+TOT(I)                                                       PLOTH
      WRITE (OUT,450)                                                        PLOTH
      IF(ICO.EQ.2) WRITE (OUT,590) TOT3,TOT5,TOT4                            PLOTH
C*************************************************************************  PLOTH
      RETURN                                                                 PLOTH
C                                                                            PLOTH
C                                                                            PLOTH
C                                                                            PLOTH
C                                                                            PLOTH
  450 FORMAT (4X,'*',3X,'CONSTANT HEAD'/)                                    PLOTH
  460 FORMAT (T5,'HEADS AT END OF TIME STEP',I5)                             PLOTH
  470 FORMAT (T5,'SATURATED THICKNESS AT END OF TIME STEP',I5)               PLOTH
  480 FORMAT (T5,'QUALITY VALUES AT END OF TIME STEP',I5/)                   PLOTH
  490 FORMAT (1H0,T25,'FREQUENCY DISTRIBUTION',//,' SYMBOL',12X,5HRANGE(     PLOTH
     1FT),12X,'FREQUENCY',5X,' PER CENT .LE.')                               PLOTH
  500 FORMAT (6H MEAN,19X,F10.5,/19H STANDARD DEVIATION,5X,F10.5)            PLOTH
  510 FORMAT (1H ,3X,A1,6X,F8.3,4H TO ,F8.3,9X,I4,6X,F10.1)                  PLOTH
  520 FORMAT (1H0,19X,100I1)                                                 PLOTH
  530 FORMAT (11X,100I1)                                                     PLOTH
  540 FORMAT(1H1,T26,20A4//T26,20A4/)
  550 FORMAT (25X,20A4/)                                                     PLOTH
  560 FORMAT (1H+,A1)                                                        PLOTH
  570 FORMAT (1H+,10X,2A4)                                                   PLOTH
  580 FORMAT (1H+,T65,3E20.8)                                                PLOTH
  590 FORMAT (/T60,'TOTAL',3E20.8)                                           PLOTH
  600 FORMAT (//)                                                            PLOTH
  610 FORMAT (1H+,T70,'AREA',1X,A8,'**2',T89,'VOLUME',1X,A6,'**3',           PLOTH
     1T109,'VOLUME',1X,A6,'**3'/T89,'SATURATED',T109,'PRESSURE')             PLOTH
  620 FORMAT (1H ,I5,4X,1H.,100A1)                                           PLOTH
  630 FORMAT (1H ,3X,A1,8X,F8.3,4H TO ,F8.3,9X,I4,6X,F10.1)                  PLOTH
  640 FORMAT (1H0,19X,100I1)                                                 PLOTH
  650 FORMAT (10X,1H.,100I1)                                                 PLOTH
  660 FORMAT (11X,100A1)                                                     PLOTH
  670 FORMAT (10X,2A4)                                                       PLOTH
  680 FORMAT (T5,'ROW')                                                      PLOTH
  690 FORMAT (' SCALE INCORRECT  PLOT TERMINATED'//)                         PLOTH
      END                                                                    PLOTH
```



```
      SUBROUTINE PLOTS (NROW,NCOL,FLAG,HSIM,HOBS,N)                          PLOTS
C*************************************************************************  PLOTS
C     THIS SUBROUTINE PRODUCES A SYMBOLIC MAP OF VARIOUS PARAMETERS.         PLOTS
C     MAXIMUM NUMBER OF COLUMNS IS 100.                                      PLOTS
C     N=1  SIMULATION ERROR MAP                                              PLOTS
```



```
C     N=2 HEAD CHANGE DURING THIS TIME STEP                                  PLOTS
C     N=3 HEAD CHANGE THROUGH THIS TIME STEP                                 PLOTS
C*************************************************************************  PLOTS
      COMMON /ITCOM/ NR,NC,ISTEP,NPARM,IN,OUT,OUT1,OPT(30),ITER,NSAVE,       PLOTS
     1ISAVE(25),JSAVE(25),KNYD,NCOLS,MCOLS(25),NROWS,MROWS(25)              PLOTS
     2,IN1,IN2,IN3,IN4,IN5,IN6                                               PLOTS
     1,NSTEPS,NBLK,IRWC(4,60),NSPRG,ISPRG(25),JSPRG(25)                      PLOTS
      COMMON /RLCOM/ FMT(20),TITLE(20),DELX(100),DELY(100),PRMITR(10),S      PLOTS
     1(100),C(100),SUMSS(15,2),                                              PLOTS
     2                            ERROR,PMPFCT,PMPNAM,PERFCT,DELTA,          PLOTS
     3DELMAJ,E,XLGTHM,FLXNAM(2),FLXFCT                                       PLOTS
     1,DELMJ2,TIME,STRFCT,SCALE                                              PLOTS
     1,TITMOD(20),VFMT(20,8)
      INTEGER OPT,FLAG,OUT,OUT1                                              PLOTS
      DIMENSION IFREQ(26),SYMBOL(26),XRANGE(27),PLOT(100),                   PLOTS
     1PERCT(26),HSIM(NROW,NCOL),HOBS(NROW,NCOL),FLAG(NROW,NCOL)             PLOTS
     1,XINTGR(10)                                                            PLOTS
      DATA XINTGR/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0/                   PLOTS
C*************************************************************************  PLOTS
C     XINCRE EQUALS THE RANGE FOR EACH PRINTER SYMBOL                        PLOTS
C*************************************************************************  PLOTS
      DATA XINCRE,BLANK,IFIRST/10.0,4H    ,0/                                PLOTS
      DATA SYMBOL/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL,           PLOTS
     11HM,1HN,1HO,1HP,1HQ,1HR,1HS,1HT,1HU,1HV,1HW,1HX,1HY,1HZ/               PLOTS
      DIMENSION IDUM(10)                                                     PLOTS
      DATA IDUM/1,2,3,4,5,6,7,8,9,0/                                         PLOTS
      DIMENSION XROW(3), XCOL(2)                                             PLOTS
      DATA XROW/1HR,1HO,1HW/,XCOL/4HCOLU,4HMN  /                             PLOTS
      DIMENSION SUMS(10),IMAXS(4,3),HMAXS(2,3)                               PLOTS
      ENTRY OPLOTS(NROW,NCOL,FLAG,HSIM,HOBS,OTITLE,N)                        PLOTS
      DIMENSION OTITLE(20)                                                   PLOTS
      HA=0                                                                   PLOTS
      HB=-1.E+6                                                              PLOTS
      HC=1.E6                                                                PLOTS
      M=0                                                                    PLOTS
      DO 10 I=1,NR                                                           PLOTS
      DO 10 J=1,NC                                                           PLOTS
      IF(FLAG(I,J).GT.2) GO TO 10                                            PLOTS
      IF(N.EQ.1.AND. ABS(HOBS(I,J)).LT.1.E-3) GO TO 10                       PLOTS
      M=M+1                                                                  PLOTS
      HD=HSIM(I,J)-HOBS(I,J)                                                 PLOTS
      HA=HA+HD                                                               PLOTS
      HB=AMAX1(HB,HD)                                                        PLOTS
      HC=AMIN1(HC,HD)                                                        PLOTS
   10 CONTINUE                                                               PLOTS
      HA=HA/M                                                                PLOTS
      ERR=HC                                                                 PLOTS
      HB=(HB-HC)/26                                                          PLOTS
      HC=0.1                                                                 PLOTS
      IF(HB.GT.0.1) HC=1                                                     PLOTS
```

FORM SUPLM

```
      DO 300 KL=1,KK
      IF(KK.LT.1) GO TO 300
      WRITE (OUT,780)
  300 CONTINUE
  310 WRITE (OUT,670) I,PLOT
      YDIS=YDIS+SCALE/8.*(KK+1)
  320 CONTINUE
      WRITE (OUT,780)
      WRITE (OUT,830) XCOL
      DO 350 L=1,2
      DO 330 K=1,100
  330 PLOT(K)=BLANK
      DO 340 J=JST,JEND
      KG=G(J)
      KL=J/10
      PLOT(KG)=XINTGR(KL)
      IF(KL.EQ.0) PLOT(KG)=BLANK
      IF(L.EQ.1) GO TO 340
      KL=MOD(J,10)
      IF(KL.EQ.0) KL=10
      PLOT(KG)=XINTGR(KL)
  340 CONTINUE
  350 WRITE (OUT,790) PLOT
      JST=JEND
      ORIGK=GSAVE
      G(JST)=GSAVE
      IF(JST.LT.NC) GO TO 70
      GO TO 370
  360 WRITE (OUT,650)
  370 CONTINUE
      DO 670 NBLKN=1,NBLK
      HMAX=-1.E8
      HMIN=1.E8
      DO 380 K=1,10
  380 SUMS(K)=0.
      DO 390 K=1,3
      HMAXS(1,K)=-1.E6
  390 HMAXS(2,K)=1.E6
      DO 400 I=1,26
  400 IFREQ(I)=0.
      XNODE=0.0
      IST=IRWC(1,NBLKN)
      IEND=IRWC(2,NBLKN)
      JST=IRWC(3,NBLKN)
      JEND=IRWC(4,NBLKN)
      DO 520 I=IST,IEND
      DO 510 J=JST,JEND
      IFL=FLAG(I,J)+1
      GO TO(510,410,410,510),IFL
  410 ERR=HSIM(I,J)-HOBS(I,J)
```



```
      IF(N.EQ.1.AND. ABS(HOBS(I,J)).LT.1.E-3) GO TO 510
      XNODE=XNODE+1.
      DO 420 KK=2,26
      IF(ERR.LT.XRANGE(KK)) GO TO 430
  420 CONTINUE
      IERR=25
      GO TO 440
  430 IERR=KK-1
  440 IFREQ(IERR)=IFREQ(IERR)+1
      DO 500 K=1,3
      GO TO (450,460,470), K
  450 HA=HSIM(I,J)
      GO TO 480
  460 HA=HOBS(I,J)
      GO TO 480
  470 HA=ERR
  480 HOUM=HMAXS(1,K)
      HMAX=AMAX1(HA,HMAXS(1,K))
      IF (HOUM.GE.HMAX) GO TO 490
      HMAXS(1,K)=HMAX
      IMAXS(1,K)=I
      IMAXS(2,K)=J
  490 HOUM=HMAXS(2,K)
      HMIN=AMIN1(HA,HMAXS(2,K))
      IF (HOUM.LE.HMIN) GO TO 500
      HMAXS(2,K)=HMIN
      IMAXS(3,K)=I
      IMAXS(4,K)=J
  500 CONTINUE
      SUMS(1)=SUMS(1)+1.
      SUMS(2)=SUMS(2)+HSIM(I,J)*HOBS(I,J)
      SUMS(3)=SUMS(3)+HSIM(I,J)*HSIM(I,J)
      SUMS(4)=SUMS(4)+HSIM(I,J)
      SUMS(5)=SUMS(5)+HOBS(I,J)*HOBS(I,J)
      SUMS(6)=SUMS(6)+HOBS(I,J)
      SUMS(7)=SUMS(7)+ERR*ERR
      SUMS(8)=SUMS(8)+ERR
      ERR=ABS(ERR)
      SUMS(9)=SUMS(9)+ERR*ERR
      SUMS(10)=SUMS(10)+ERR
  510 CONTINUE
  520 CONTINUE
      IF (NBLKN.GT.1) GO TO 540
      PERCT(1)=FLOAT(IFREQ(1))/XNODE*100.
      DO 530 I=2,26
  530 PERCT(I)=FLOAT(IFREQ(I))/XNODE*100.+PERCT(I-1)
      WRITE (OUT,780)
      WRITE (OUT,760) (SYMBOL(I),XRANGE(I),XRANGE(I+1),IFREQ(I),PERCT(I)
     1,I=1,25)
      WRITE (OUT,650)
```

13 TOWRGW+TK-GWSIM-IV QREAD 27 OCT 89 PAGE 2

```
      READ(IN,1160) FMT
      WRITE(OUT,1400)(FMT(I),I=1,10)
  160 WRITE(OUT,1230)
  170 READ(IN,FMT) II,III,JJ,JJJ,NA
      IF(II.LT.1) GO TO 190
      WRITE(OUT,1240)II,III,JJ,JJJ,NA
      DO 180 I=II,III
      DO 180 J=JJ,JJJ
  180 A(I,J)=NA
      GO TO 170
  190 IF(IOPTQ(4).LT.1) GO TO 250
C*****************************************************************************
C     READ VALUE ADJUSTMENTS
C*****************************************************************************
  200 WRITE(OUT,1250)(HEADNG(I,NTYPE),I=1,8)
      DO 210 I=1,20
  210 FMT(I)=VFMT(I,5)
      IF(IOPTQ(4).LT.5) GO TO 220
      READ(IN,1160) FMT
      WRITE(OUT,1400)(FMT(I),I=1,10)
  220 WRITE(OUT,1230)
  230 READ(IN,FMT) II,III,JJ,JJJ,NA
      IF(II.LT.1) GO TO 250
      WRITE(OUT,1240) II,III,JJ,JJJ,NA
      DO 240 I=II,III
      DO 240J=JJ,JJJ
  240 A(I,J)=A(I,J)+NA
      GO TO 230
  250 RETURN
 1160 FORMAT(20A4)
 1120 FORMAT(1H1,T25,20A4,//T25,20A4/,T25,20A4/)
 1170 FORMAT(1H0,T20,8A4,' FOR EACH CELL'//)
 1210 FORMAT(1H0,I5,10G10.2/(6X,10G10.2))
 1220 FORMAT(1H0,T20,8A4,' BY BLOCK'//)
 1250 FORMAT(1H0,T20,8A4,' ADJUSTMENTS'//)
 1230 FORMAT(/T21,'ROW     ROW  COLUMN  COLUMN',T56,'VALUE'/T21,
     1'START  END  START     END'//)
 1240 FORMAT(T21,I3,4X,I3,3X,I3,5X,I3,2G19.4)
 1400 FORMAT(T70,'FORMAT IS',T80,10A4/)
      END
```

14 TOWRGW+TK-GWSIM-IV OSOLVE 27 OCT 89 PAGE 1

```
       SUBROUTINE OSOLVE(H,HO,T,SF1,R,Q1,Q2,Q,BOTLEL,RO,Q3,RHG,THIK,
     1B,G,FLAG,ERROR,DELMAJ,NROW,NCOL,NR,NC,ITER,ALPHA)
      DIMENSION Q1(NROW,NCOL),Q2(NROW,NCOL),THIK(NROW,NCOL),
     1RO(NROW,NCOL),RHG(NROW,NCOL),
     2 BOTLEL(NROW,NCOL),Q3(NROW,NCOL)
C**********************************************************************
```

14 TOWRGW+TK-GWSIM-IV OSOLVE 27 OCT 89 PAGE 2

```
C       THIS ROUTINE SOLVES FOR CONCENTRATIONS BY THE IDAI PROCEDURE
C**********************************************************************
        DIMENSION H(NROW,NCOL),HO(NROW,NCOL),T(NROW,NCOL,2),
       1 B(1),G(1),SF1(NROW,NCOL),Q(NROW,NCOL),R(NROW,NCOL)
C**********************************************************************
C        ALPHA IS A CALIBRATION VALUE WHICH WEIGHTS THE CONCENTRATION
C         OF THE FLUX BETWEEN CELL.  RANGE OF VALUES IS FROM 0.5 TO 1.0
C         A VALUE OF ONE MEANS NO AVERAGING OF CONCENTRATIONS
C         C(I->J)=ALPHA*C(I)+(1-ALPHA)*C(J)
C**********************************************************************
              DOUBLE PRECISION AA,BB,CC,DD,W
        INTEGER FLAG(NROW,NCOL)
        ISTEP=1
        ITER=0
 10     CONTINUE
        ITER=ITER+1
        IF (ITER.GT.50) GO TO 610
        E=0.0
C**********************************************************************
C       ROW CALCULATIONS
C**********************************************************************
        DO 410 II=1,NR
        I=II
        IF(MOD(ISTEP+ITER,2).EQ.1)I=NR-I+1
        JSTRT=1
 20     DO 30 JJ=JSTRT,NC
        IFL=FLAG(I,JJ)+1
        GO TO (30,40,40,30), IFL
 30     CONTINUE
        GO TO 410
 40     CONTINUE
        JJP1=JJ+1
 50     DO 60 JJJ=JJP1,NC
        IFL=FLAG(I,JJJ)+1
        GO TO (70,60,60,70), IFL
 60     CONTINUE
        JJJ=NC
        JSTRT=NC
        GO TO 80
 70     JJJ=JJJ-1
        JSTRT=JJJ+1
 80     CONTINUE
        DO 370 J=JJ,JJJ
        BB=R(I,J)/DELMAJ
        DD=RO(I,J)/DELMAJ*HO(I,J)
 90     CONTINUE
        BB=BB+Q3(I,J)
        DD=DD+RHG(I,J)*Q(I,J)
        AA=0.0
        CC=0.0
```

```
  460 CONTINUE
      III=NR
      ISTRT=NR
      GO TO 480
  470 III=II-1
      ISTRT=III+1
  480 CONTINUE
      DO 760 I=II,III
      BB=R(I,J)/DELMAJ
      DD=RO(I,J)/DELMAJ*HO(I,J)
      DD=DD+RHG(I,J)*Q(I,J)
      BB=BB+O3(I,J)
      AA=0.0
      CC=0.0
      IF (J-1) 490,530,490
  490 TT=T(I,J-1,1)
      HA=Q1(I,J-1)
      BB=BB+TT
      DD=DD+TT*H(I,J-1)
      HB=HO(I,J-1)
      IF (HA) 500,530,510
  500 FACT=ALPHA
      FACT1=1.-FACT
      GO TO 520
  510 FACT1=ALPHA
      FACT=1.-FACT1
  520 BB=BB-HA*FACT
      DD=DD+HA*FACT1*H(I,J-1)
  530 IF (J-NC) 540,580,540
  540 TT=T(I,J,1)
      HA=Q1(I,J)
      BB=BB+TT
      DD=DD+TT*H(I,J+1)
      HB=HO(I,J+1)
      IF (HA) 550,580,560
  550 FACT1=ALPHA
      FACT=1.-FACT1
      GO TO 570
  560 FACT=ALPHA
      FACT1=1.-FACT
  570 BB=BB+HA*FACT
      DD=DD-HA*FACT1*H(I,J+1)
  580 IF (I-II) 590,590,600
  590 IF (I-1) 660,660,600
  600 TT=T(I-1,J,2)
      HA=Q2(I-1,J)
      BB=BB+TT
      AA=-TT
      HB=HO(I-1,J)
      IF (HA) 610,660,620
```



```
  610 FACT=ALPHA
      FACT1=1.-FACT
      GO TO 630
  620 FACT1=ALPHA
      FACT=1.-FACT1
  630 BB=BB-HA*FACT
      IF (FLAG(I-1,J)) 650,650,640
  640 AA=AA-HA*FACT1
      GO TO 660
  650 DD=DD+H(I-1,J)*(HA*FACT1+TT)
      AA=0.
  660 IF (I-III) 680,670,680
  670 IF (I-NR) 680,750,750
  680 TT=T(I,J,2)
      HA=Q2(I,J)
      BB=BB+TT
      CC=-TT
      HB=HO(I+1,J)
      IF (HA) 690,750,700
  690 FACT1=ALPHA
      FACT=1.-ALPHA
      GO TO 710
  700 FACT=ALPHA
      FACT1=1.-FACT
  710 BB=BB+HA*FACT
  720 IF (FLAG(I+1,J)) 740,740,730
  730 CC=CC+HA*FACT1
      GO TO 750
  740 DD=DD+H(I+1,J)*(TT-HA*FACT1)
      CC=0.
  750 W=BB-AA*B(I-1)
      B(I)=CC/W
  760 G(I)=(DD-AA*G(I-1))/W
C*****************************************************************
C     RE-ESTIMATE CONCENTRATIONS
C*****************************************************************
      E=E+ABS(H(III,J)-G(III))
      H(III,J)=G(III)
      N=III
  770 N=N-1
      IF (N-II+1) 790,790,780
  780 HA=G(N)-B(N)*H(N+1,J)
      E=E+ABS(H(N,J)-HA)
      H(N,J)=HA
      GO TO 770
  790 IF (ISTRT-NR) 420,800,800
  800 CONTINUE
      IF (E.GT.ERROR) GO TO 10
  810 RETURN
C*****************************************************************
```

```
      WRITE(IN7) R,RO,P,T,SURFAC,NSEGN,OL,TOPAQ,SF1
      WRITE (IN7) H,HO
      IF (KOT.LT.1) GO TO 70                                            QUAL
C*********************************************************************  QUAL
C     FOR SECOND OR LATER TIME STEP, READ MASS TRANSPORT                QUAL
C     COEFFICIENTS FROM 'IN8'                                           QUAL
C*********************************************************************  QUAL
      REWIND IN8                                                        QUAL
      READ (IN8) ((R(I,J),RO(I,J),P(I,J,1),P(I,J,2),SURFAC(I,J),NSEGN(I, QUAL
     1J),SF1(I,J),I=1,NR),J=1,NC)                                       QUAL
      READ (IN8) H,HO
   70 CONTINUE                                                          QUAL
      I=IOPT(1)+IOPT(3)+IOPT(4)                                         QUAL
      IF(I.LT.1) GO TO 110
C************************************************************************
C     READ DISPERSIVITY COEFFICIENTS
C************************************************************************
      DO 104 I=1,4
  104 IDCF(I)=0
      IF(IOPT(1).LT.1) GO TO 105
      IDCF(1)=IOPT(1)
      IDCF(2)=IOPT(2)
      CALL QREAD(QTITLE,P(1,1,1),IDCF,NROW,NCOL,1)
      CALL QREAD(QTITLE,P(1,1,2),IDCF,NROW,NCOL,5)
      IDCF(1)=0
  105 IF(IOPT(3).LT.1) GO TO 107
      IDCF(3)=IOPT(3)
      CALL QREAD(QTITLE,P(1,1,1),IDCF,NROW,NCOL,1)
      CALL QREAD(QTITLE,P(1,1,2),IDCF,NROW,NCOL,5)
      IDCF(3)=0
  107 IF(IOPT(4).LT.1) GO TO 110
      IDCF(4)=IOPT(4)
      CALL QREAD(QTITLE,P(1,1,1),IDCF,NROW,NCOL,1)
      CALL QREAD(QTITLE,P(1,1,2),IDCF,NROW,NCOL,5)
  110 I=IOPT(6)+IOPT(7)+IOPT(8)
      IF(I.LT.1) GO TO 120
C************************************************************************
C     READ RECHARGE QUALITIES
C************************************************************************
      CALL QREAD(QTITLE,SURFAC,IOPT(6),NROW,NCOL,2)
  120 I=IOPT(15)+IOPT(17)+IOPT(18)
      IF(I.LT.1) GO TO 130
C************************************************************************
C     READ INITIAL CONCENTRATIONS
C************************************************************************
      CALL QREAD(QTITLE,R,IOPT(15),NROW,NCOL,3)
  130 I=IOPT(21)+IOPT(23)+IOPT(24)
      IF(I.LT.1) GO TO 140
C************************************************************************
C     READ POROSITY
```



```
C************************************************************************
      CALL QREAD(QTITLE,SF1,IOPT(21),NROW,NCOL,4)
  140 CONTINUE
      KOT=KOT+1
      IF (KOT.GT.1) GO TO 90                                            QUAL
C*********************************************************************  QUAL
C     CALCULATE INITIAL VOLUME IN STORAGE                               QUAL
C*********************************************************************  QUAL
      REWIND IN4                                                        QUAL
      READ(IN4) HO
      DO 80 J=1,NC
      DO 80 I=1,NR                                                      QUAL
      HO(I,J)=THIK(I,J)                                                 QUAL
      IF(FLAG(I,J).EQ.1)HO(I,J)=(HO(I,J)-BOTLEL(I,J))
      HO(I,J)=HO(I,J)*SF1(I,J)*DELX(J)*DELY(I)
   80 CONTINUE
   90 CONTINUE                                                          QUAL
      IF (ISTEP.GT.1) GO TO 550                                         QUAL
      DO 540 J=1,NC                                                     QUAL
      DO 540 I=1,NR                                                     QUAL
  540 NSEGN(I,J)=R(I,J)                                                 QUAL
  550 CONTINUE                                                          QUAL
C*********************************************************************  QUAL
C     ADD FLOWS FROM AQUIFER TO PUMPAGE                                 QUAL
C*********************************************************************  QUAL
      DO 560 J=1,NC                                                     QUAL
      DO 560 I=1,NR                                                     QUAL
      IF(QSUM(I,J).GT.0.) Q(I,J)=Q(I,J)+QSUM(I,J)/DELMAJ                QUAL
  560 CONTINUE                                                          QUAL
C*********************************************************************  QUAL
C      REWIND STORAGE DEVICE                                            QUAL
C*********************************************************************  QUAL
      CALL SUMFLO (TOPAQ,OL,T,R,RO,S,DELTA,DELMAJ,NC,NR,NROW,NCOL,5,IN8, QUAL
     1IN7)                                                              QUAL
      DO 600 I=1,NR                                                     QUAL
      DO 600 J=1,NC                                                     QUAL
      RO(I,J)=R(I,J)                                                    QUAL
      IF(FLAG(I,J).EQ.1) THIK(I,J)=H(I,J)-BOTLEL(I,J)                   QUAL
C*********************************************************************
C      CONVERT Q FROM NET WITHDRAWAL TO ASSIGNED PUMPAGE                QUAL
C      PLUS SPRING FLOWS                                                QUAL
C     PLUS FLOWS FROM AQUIFER                                           QUAL
C*********************************************************************  QUAL
      Q(I,J)=Q(I,J)+RHG(I,J)*PMPFCT                                     QUAL
C*********************************************************************  QUAL
C     RHG EQUALS RECHARGE PLUS FLOWS TO AQUIFER                         QUAL
C*********************************************************************  QUAL
      RHG(I,J)=RHG(I,J)*PMPFCT                                          QUAL
      IF(QSUM(I,J).LT.1.) RHG(I,J)=RHG(I,J)-QSUM(I,J)/DELMAJ            QUAL
  600 CONTINUE                                                          QUAL
                                                                        QUAL
```

```
      HC=T(I,J-1,1)*(RD(I,J-1)-RD(I,J))
      IF(HC.GT.0.) HA=HA+HC
      IF(HC.LT.0.) HB=HB-HC
      IF(TOPAQ(I,J-1)) 790,810,800
  790 HB=HB-TOPAQ(I,J-1)*(ALPHA*RD(I,J)+ALPHA1*RD(I,J-1))
      GO TO 810
  800 HA=HA+TOPAQ(I,J-1)*(ALPHA*RD(I,J-1)+ALPHA1*RD(I,J))
  810 IF(J-NC) 820,850,850
  820 IF(FLAG(I,J+1).EQ.0) GO TO 850
      HC=T(I,J,1)*(RD(I,J+1)-RD(I,J))
      IF(HC.GT.0.) HA=HA+HC
      IF(HC.LT.0.) HB=HB-HC
      IF(TOPAQ(I,J)) 830,850,840
  830 HA=HA - TOPAQ(I,J)*(ALPHA*RD(I,J+1)+ALPHA1*RD(I,J))
      GO TO 850
  840 HB=HB + TOPAQ(I,J)*(ALPHA*RD(I,J)+ALPHA1*RD(I,J+1))
  850 SUMP=SUMP + HA*DELTA
      SUMR=SUMR+ HB * DELTA
  860 CONTINUE
      SUMST=SUMST+RD(I,J)*R(I,J)
      SUMEN=SUMEN+H(I,J)*RD(I,J)
      SUMP=SUMP+Q(I,J)*DELTA*RD(I,J)
      SUMR=SUMR+RNC(I,J)*DELTA*SURFAC(I,J)
  870 CONTINUE
      DMAS=SUMEN-SUMST
      SUMD=SUMD+DMAS
      SUMI=SUMI+SUMR
      SUMO=SUMO+SUMP
      SUMER=DMAS+SUMP-SUMR
      SUME=SUME+SUMER
      IF (NSMAL.EQ.1.OR.IOPT(19).GT.0) WRITE (OUT,1360)
      WRITE (OUT,1370) NSMAL,ITER,DELTA,SUMR,SUMP,DMAS,SUMER
      IF (IOPT(19).GT.0) GO TO 880
      GO TO 900
  880 WRITE(OUT,1120) TITMOD,TITLE,OTITLE
      WRITE (OUT,1380) NSMAL
C*********************************************************************
C      WRITE CONCENTRATIONS AT END OF TIME STEP
C*********************************************************************
      DO 890 I=1,NR
  890 WRITE (OUT,1210) I,(RD(I,J),J=1,NC)
  900 CONTINUE
      DO 910 I=1,NR
      DO 910 J=1,NC
      HD(I,J)=H(I,J)
  910 R(I,J)=RD(I,J)
      DELTA=DELTA*TIMACL
C*********************************************************************
C      END OF SMALL TIME STEP
C*********************************************************************
```



```
  920 CONTINUE
      DELTA=DEL
      IF (IOPT(19).GT.0) WRITE (OUT,1350)
      WRITE (OUT,1390) ISTEP,SUMI,SUMO,SUMD,SUME
C*********************************************************************
C      READ CONCENTRATIONS AT BEGINNING OF TIME STEP
C*********************************************************************
      REWIND INS
      READ (INS) R
      IF (IOPT(19).GT.0) GO TO 940
      IF (IOPT(9).LT.1) GO TO 940
C*********************************************************************
C      WRITE ENDING CONCENTRATIONS
C*********************************************************************
      WRITE(OUT,1120) TITMOD,TITLE,OTITLE
      WRITE (OUT,1280) ISTEP
      DO 930 I=1,NR
  930 WRITE (OUT,1210) I,(RD(I,J),J=1,NC)
C*********************************************************************
C      PLOT CONTOUR MAP OF CONCENTRATIONS
C*********************************************************************
  940 IF(IOPT(10).GT.0) CALL OPLOTH(NROW,NCOL,FLAG,RD,OTITLE,3)
      IF (IOPT(11).LT.1) GO TO 970
C*********************************************************************
C      LIST QUALITY CHANGES DURING THIS TIME STEP
C*********************************************************************
      WRITE(OUT,1120) TITMOD,TITLE,OTITLE
      WRITE (OUT,1300) ISTEP
      DO 960 I=1,NR
      DO 950 J=1,NC
  950 B(J)=RD(I,J)-R(I,J)
  960 WRITE (OUT,1210) I,(B(J),J=1,NC)
  970 IF (IOPT(20).LT.1) GO TO 1000
C*********************************************************************
C      LIST CHANGES IN CONCENTRATIONS THROUGH THIS STEP
C*********************************************************************
      WRITE(OUT,1120) TITMOD,TITLE,OTITLE
      WRITE (OUT,1310) ISTEP
      DO 990 I=1,NR
      DO 980 J=1,NC
  850 B(J)=RD(I,J)-HBEGN(I,J)
  990 WRITE (OUT,1210) I,(B(J),J=1,NC)
C*********************************************************************
C      PLOT QUALITY CHANGES DURING THIS TIME STEP
C*********************************************************************
 1000 IF(IOPT(12).GT.0) CALL OPLOTS(NROW,NCOL,FLAG,RD,R,OTITLE,5)
C*********************************************************************
C      PLOT QUALITY CHANGES THROUGH THIS TIME STEP
C*********************************************************************
      IF(IOPT(13).GT.0) CALL OPLOTS(NROW,NCOL ,FLAG,RD,HBEGN,OTITLE,6)
```