A MODIFICATION OF THE FINITE-DIFFERENCE MODEL FOR SIMULATION OF TWO DIMENSIONAL GROUND-WATER FLOW TO INCLUDE SURFACE-GROUND WATER RELATIONSHIPS

By Melih M. Ozbilgin and David C. Dickerman

U.S. GEOLOGICAL SURVEY
WATER-RESOURCES INVESTIGATIONS REPORT 83-4251

Prepared in cooperation with the
RHODE ISLAND WATER RESOURCES BOARD

Providence, Rhode Island

1984

UNITED STATES DEPARTMENT OF THE INTERIOR

WILLIAM P. CLARK, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

Open-file report

For additional information write to:

U.S. Geological Survey
John O. Pastore Federal Bldg.
and U.S.P.O., Room 237
Providence, R.I. 02903
(Telephone: (401) 528-5135)

Copies of this report can be purchased from:

Open-File Services Section
Western Distribution Branch
U.S. Geological Survey
Box 25425, Federal Center
Denver, Colorado 80225
(Telephone: (303) 234-5888)

CONTENTS

ILLUSTRATIONS

TABLES

CONVERSION FACTORS

For use of those readers who may prefer to use metric (SI) units rather than inch-pound units, the conversion factors for terms used in this report are listed below.

Multiply inch-pound unit	By	To obtain metric (SI) unit
	LENGTH	
foot (ft)	0.3048	meter (m)
inch (in)	25.40	millimeter (mm)
	FLOW	
cubic foot per second (ft^3/s)	2.832×10^{-2}	cubic meter per second (m^3/s)
	TRANSMISSIVITY	
foot squared per day (ft^2/d)	0.0929	meter squared per day (m^2/d)
	HYDRAULIC CONDUCTIVITY	
foot per day (ft/d)	0.3048	meter per day (m/d)

A MODIFICATION OF THE FINITE-DIFFERENCE MODEL FOR SIMULATION OF TWO-DIMENSIONAL GROUND-WATER FLOW TO INCLUDE SURFACE-GROUND WATER RELATIONSHIPS

By Melih M. Ozbilgin* and David C. Dickerman**

ABSTRACT

The two-dimensional finite-difference model for simulation of ground-water flow described by Trescott and others (1976) was modified to enable simulation of the interaction between surface water and ground water during periods of low streamflow. Modifications were designed to allow calculation of surface-water heads for, and flow either to or from, contiguous surface-water bodies in simulation of aquifer response to imposed stresses; and to allow more convenient data input.

The method of data input and output in the 1976 program was modified and two entries (RSORT and HRIVER) were added to the COEF and CHECKI subroutines to calculate surface-water heads. A new subroutine CALC was added to the program which initiates the modified program for surface water calculations. If CALC is not specified as a simulation option, the program runs the original version. The subroutines which solve the ground-water flow equations were not changed.

Arrays were added or moved around in the modified program so that recharge, evapotranspiration, inflow to surface-water bodies, number of wells, pumping rate, and duration of pumping could be varied for any time period. The Manning formula was used to relate stream depth and discharge in surface water streams. The interaction between surface water and ground water is represented through the leakage term which is included in both the ground-water flow and surface-water mass balance equations.

The documentation includes a flow chart, data deck instructions, input data, output summary, and a complete program listing. The modified program has been tested under a variety of conditions simulated for idealized aquifers. It has been used to develop a field model of a stream-pond-aquifer system in the Beaver-Pasquiset ground-water reservoir in southern Rhode Island. Numerical results from the modified program are in good agreement with published analytical results; however, users of the modified program should be aware that undiscovered errors in logic may exist.

This report is intended as a manual for use by the Rhode Island Water Resources Board, as well as other users.

* Graduate student, University of Rhode Island, Kingston, R.I.
** Hydrologist, U.S. Geological Survey, Providence, R.I.

INTRODUCTION

Digital models that simulate ground-water flow are widely used in the management of ground-water resources in order to assess the impact of withdrawals on streamflow and ground-water levels. In Rhode Island, the Water Resources Board is primarily responsible for the management of the State's water resources. In order for the Water Resources Board to manage the State's ground-water resources more effectively, a model was needed that could simulate the interaction between surface and ground water during periods of low streamflow. The purpose of this study was to modify the ground-water flow model of Trescott and others (1976) in order to meet the management needs of the Rhode Island Water Resources Board.

The two-dimensional finite-difference model described by Trescott and others (1976) was designed to simulate the response of an aquifer to imposed stresses. The aquifer may be artesian, water table, or a combination of artesian and water table. Also, it may be heterogeneous and anisotropic, and may have irregularly shaped boundaries. The model permits leakage from confining beds, constant recharge, well discharge, and evapotranspiration as a linear function of depth to water. The user may choose either the Alternating Direction Implicit (ADI), Line Successive Overrelaxation (LSOR), or Strongly Implicit Procedure (SIP) equation solving schemes.

The model of Trescott and others (1976) was modified to allow (1) calculation of surface-water heads for, and flow either to or from, contiguous surface-water bodies, and (2) recharge, evapotranspiration, inflow to surface-water bodies, number of wells, pumping rate, and duration of pumping to be input conveniently and varied for any time period.

THEORETICAL DEVELOPMENT

The following four equations are necessary for the development of the mathematical model of flow in a surface-water/ground-water system. They are, (1) a ground-water flow equation, (2) an equation describing the relationship between stream depth and discharge, (3) a mass balance equation, and (4) an equation describing the interaction between surface water and ground water.

The ground-water flow equation used in the modified program is the same one developed and solved in the two-dimensional program by Trescott and others (1976). The reader is referred to Trescott and others (1976) for theoretical development of the ground-water flow equation and its finite-difference approximation.

Flow in natural streams is assumed to be gradually varied, which is a special class of steady nonuniform flow. The depth, area, roughness, bottom slope, and hydraulic radius are assumed to change very slowly, if at all, along the channel. It is also assumed that the rate of head loss, at a given section, is defined by the Manning formula for the same depth and discharge, regardless of trends in depth.

The Manning formula is widely used to relate stream depth and discharge in prismatic open channels. For incompressible, steady flow at constant depth, the Manning formula is written as:

$$Q = \frac{C}{n} \times A \times (R)^{2/3} \times (S)^{1/2}, \qquad (1)$$

in which

Q = discharge (L^3/T),

n = Manning's roughness coefficient,

A = cross-sectional area (L^2),

R = hydraulic radius = A/P (L),

P = wetted perimeter (L),

S = losses per unit weight per unit length of channel, or slope of the bottom of the channel, and

C = constant = 1.486 for inch-pound units or 1.0 for metric units.

For a rectangular channel, the area and the hydraulic radius may be expressed as:

$$A = w \times d, \text{ and} \qquad (2)$$

$$R = A/P = (w \times d)/[w + (2 \times d)], \qquad (3)$$

in which

d = depth of water in stream (L), and

w = width of the channel (L).

By substituting equations 2 and 3 in equation 1, stream depth may be determined from a known discharge by numerical integration or by trial and error assuming a value for stream depth to satisfy the equation. To simplify equation 1, it was assumed that small, natural streams usually have depths much less than widths. Therefore, the term [w + (2 x d)] in equation 3 may be approximated by the stream width (w). Using equation 3 this results in an approximation of the hydraulic radius (R) equal to stream depth (d). Equation 1 may now be rewritten as:

$$Q = \frac{C}{n} \times w \times (d)^{5/3} \times (S)^{1/2},$$

or

$$d = [(Q \times n)/(C \times w \times S^{1/2})]^{3/5}. \qquad (4)$$

Equation 4 is used in the modified program to determine stage from a known discharge.

The Manning's roughness coefficient is dependent upon surface roughness and upon the size and shape of the channel cross section (Streeter, 1971). The roughness coefficient (n) varies from 0.01 for a smooth, glass channel to 0.15 for a tree-lined floodplain (White, 1979). Experimental values of Manning's roughness coefficient are given in table 1. White (1979) indicates that Manning's approximation is accurate in an intermediate range of roughness coefficient, but predicts unrealistically low friction and high discharge for deep, smooth channels and shallow, rough channels. The roughness coefficient can be determined from stream depth and discharge measurements. If field data are not available, values can be chosen from table 1 and varied within the indicated range.

Table 1. Experimental values of Manning's roughness coefficient (White, 1979, p. 605)

	n
Excavated earth channels:	
Clean	0.022 ± 0.004
Gravelly	0.025 ± 0.005
Weedy	0.030 ± 0.005
Stony, cobbles	0.035 ± 0.010
Natural channels:	
Clean and straight	0.030 ± 0.005
Sluggish, deep pools	0.040 ± 0.010
Major rivers	0.035 ± 0.010
Floodplains:	
Pasture, farmland	0.035 ± 0.010
Light brush	0.050 ± 0.020
Heavy brush	0.075 ± 0.025
Trees	0.150 ± 0.050

Discharge at a surface-water segment can be determined from the mass balance of that segment. The mass balance equation, in its simplest form, may be expressed as:

$$Q_{out} = Q_{in} \pm \frac{\Delta V}{\Delta t}, \qquad (5)$$

in which

Q_{in} = inflow to a segment (L^3/T),

Q_{out} = discharge from a segment (L^3/T),

$\frac{\Delta V}{\Delta t}$ = change in storage = $(\Delta d \times A_s)/\Delta t$ (L^3/T),

d = change in depth (L),

A_s = area of the segment (L^2), and

t = elapsed time (T).

The inflow to a given stream segment may be written as:

$$Q_{in} = Q_p + Q_r - Q_e \pm Q_l,$$

in which

Q_r = rainfall (L^3/T),

Q_e = evaporation from the stream surface (L^3/T),

Q_p = inflow from previous stream segment (L^3/T), and

Q_l = leakage from (-) or into (+) the aquifer (L^3/T).

For streams where surface areas are small, changes in stream storage, rainfall to the stream, and evaporation from the stream surface may be neglected. The mass balance equation for a stream segment then becomes:

$$Q_{out} = Q_p \pm Q_l. \qquad (6)$$

Inflow from a previous stream segment (Q_p) is input to the first stream segment in the model or calculated by the computer program. Leakage either into or out of the aquifer (Q_l) is calculated from equation 10. The stream depth can be determined by substituting Q_{out} from equation 6, into equation 4.

For ponds where surface areas are large, changes in pond storage, rainfall to the pond, and evaporation from the pond surface cannot be neglected. The sum of the mass balances for all pond segments is used in order to calculate a flat water surface for ponds. The total mass balance for the pond may be expressed as:

$$Q_d = Q_p + \left[\sum_{i=1}^{m} (Q_r^i - Q_e^i \pm Q_l^i) \right] \pm \frac{\Delta d_p \times \sum_{i=1}^{m} A_s^i}{\Delta t}, \qquad (7)$$

in which

Q_d = discharge from the pond into a stream, if any (L^3/T),

Q_p = inflow to the pond from a stream, if any (L^3/T),

Δd_p = change in pond depth (L), and

m = number of segments in the pond.

When no discharge occurs from the pond segment to a stream segment ($Q_d = 0$), changes in pond level are calculated from equation 7. However, when a stream segment discharges from the pond, pond levels depend upon the critical depth in the stream.

Assuming constant atmospheric pressure, the Bernoulli equation for steady, frictionless, incompressible flow along a streamline between points 1 and 2 (fig. 1) can be written as (Streeter and Wylie, 1979):

$$d_1 + \frac{V_1^2}{2g} = d_2 + \frac{V_2^2}{2g}, \qquad (8\text{-a})$$

in which

g = acceleration of gravity (L/T^2), and

V = velocity (L/T).

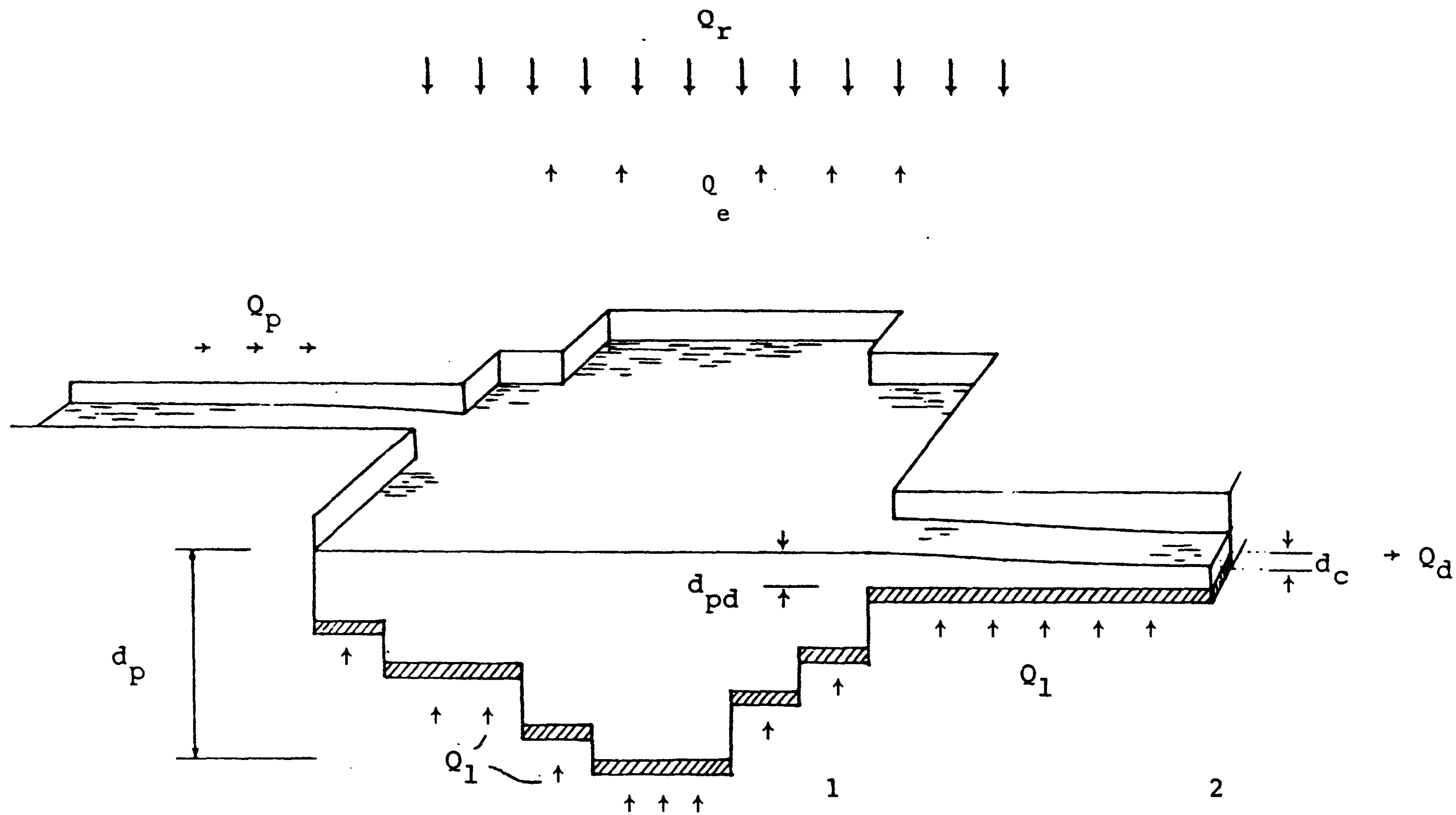


Figure 1.--Stream-pond-aquifer system representation.

For the pond, $V_1 = 0.0$ and $d_1 = d_{pd}$ (fig. 1); then

$$d_{pd} = d_2 + \frac{V_2^2}{2g}, \qquad (8\text{-b})$$

in which

$d_{pd} = d_p \pm \Delta d_p$ (L), and

d_p = pond depth at the beginning of time step(L).

The right side of equation 8-b is the specific energy (E) at point 2 in figure 1. It has a minimum at a certain value of d_2 called the critical depth. The critical depth (d_c) corresponds to a critical channel velocity (V_c) equal to the shallow-water wave-propagation speed, which is given as (White, 1979):

$$V_c = (g \times d_c)^{1/2}. \qquad (8\text{-c})$$

Substituting equation 8-c into equation 8-b will give:

$$d_{pd} = d_c + \frac{g \times d_c}{2g} = \frac{3}{2} d_c,$$

or

$$d_c = \frac{2}{3} (d_p \pm \Delta d_p). \qquad (8\text{-d})$$

For derivation of equation 8-d, specific energy (E) at the mouth of the stream is assumed equal to minimum energy (E_{min}). No flow occurs for $E<E_{min}$. For $E>E_{min}$ two types of flow occur; (1) subcritical flow, for large depth with $V<V_c$, and (2) supercritical flow, for shallow depth with $V>V_c$ (White, 1979). For small streams with shallow depths, subcritical flow does not occur. The major assumption is that channel velocity (V) will not exceed critical velocity (V_c), and supercritical flow will not occur.

Uniform critical flow can occur if the bottom slope of the channel is the critical slope (S_c) value for which the velocity is critical (White, 1979). For a wide rectangular channel where $R = d_c$ in equation 1, the uniform critical-flow rate would be:

$$Q_d = \frac{C}{n} \times w \times \left[\frac{2}{3} (d_p \pm \Delta d_p)\right]^{5/3} \times S_c^{1/2}. \qquad (9)$$

Equation 9 can be substituted into equation 7 and solved for the change in pond level, Δd_p.

The interaction between a surface-water body and an aquifer is represented in the model by the leakage term in equations 6 and 7. Leakage from a surface-water body either into or out of an aquifer is assumed to occur through a confining bed. When storage in the confining bed is neglected, the leakage term is expressed as (Trescott and others, 1976):

$$Q_l = \frac{K'}{m'} \times (H_s - H_a) \times A_a, \qquad (10)$$

in which

Q_l = leakage from (-) or into (+) the aquifer (L^3/T),

K' = hydraulic conductivity of the confining bed (L/T),

m' = thickness of the confining bed (L),

H_s = surface-water head (L),

H_a = aquifer head (L), and

A_a = area corresponding to aquifer node (L^2).

The model is capable of calculating transient effects of the leakage (equation 9; Trescott and others, 1976). However, for natural situations considered in this study, storage in the confining bed and time steps are small, and transient effects are negligible.

MODEL DEVELOPMENT

Streams and ponds within the model are treated as areas of either known aquifer inflow or outflow. A grid is superimposed on the stream and pond areas and the length of each stream segment is set equal to the length of the corresponding grid block. The width of each stream segment is set equal to the actual channel width. Figure 2 shows a generalized stream-aquifer grid representation. The area term in equation 10, which describes the interaction between the stream and aquifer, is the area contributing to the aquifer node. Therefore, streambed hydraulic conductivity is adjusted so calculations will be done for leakage through the actual stream area. This is done by applying the following relationship:

$$K_{adj} = K' \frac{A_s}{A}, \qquad (11)$$

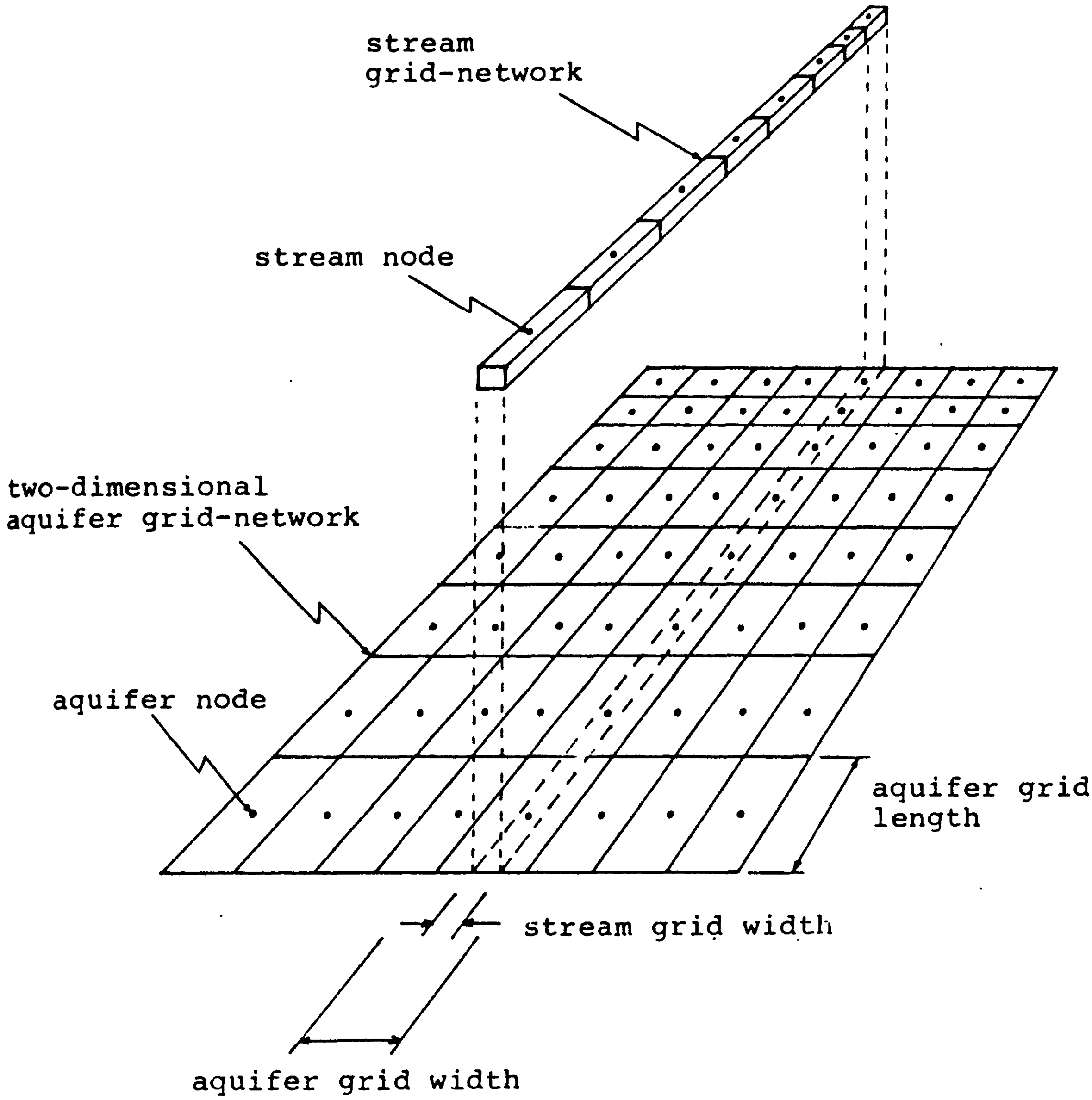


Figure 2.--Stream-aquifer grid representation.

in which

K_{adj} = streambed hydraulic conductivity adjusted for area (L/T),

K' = streambed hydraulic conductivity (L/T),

A_s = area of the streambed in the grid block (L^2), and

A = area of the grid block (L^2).

When the surface water area represents a pond, no adjustment is necessary as long as the pond and grid block areas are equal.

At the beginning of each time step, an initial surface-water head is determined at a node using known inflow and the leakage either to or from the aquifer that was computed in the previous time step. Net inflow or outflow to each aquifer node under the stream or pond is computed using the surface-water head, and input as a known recharge or discharge to the ground-water flow equation at that node. The finite-difference form of the ground-water flow equation is then solved for the entire aquifer. Next, new values for leakage at the surface-water nodes are calculated and these new values are then used to adjust surface-water heads. The ground-water flow equations are solved using the new surface-water heads. This procedure is continued until the largest difference in leakage between two successive trials is less than a given error. The procedure just described can be followed graphically on page 27 with the simplified flow chart for the modified program.

The error criterion needed to satisfy this difference in leakage may be set equal to any number by the user. However, this number should depend upon the expected leakage and the number of trials for each time period. The minimum value for the error criterion can be calculated from the following equation:

$$E_l = 4 \times E_h \times \frac{K'_{max}}{m'_{min}} \times A_{max}, \qquad (12)$$

in which

E_l = error criterion satisfying the interaction between surface water and ground water(L^3/T),

E_h = error criterion satisfying the change in aquifer head between two successive iterations (L),

K'_{max} = maximum confining bed hydraulic conductivity in the model (L/T),

m'_{min} = minimum confining bed thickness (L), and

A_{max} = maximum area contributing to an aquifer node (L^2).

It should be noted that the calculation of net leakage to surface-water bodies is determined in the model to four significant digits. Therefore, an E value less than 0.001 should not be used unless modifications to the model are made.

For treatment of pond areas, first the depth at the deepest point in the pond is determined by solving equation 7. Second, the altitude at remaining pond nodes is set equal to the altitude at the deepest pond node. This gives the pond a flat water surface. Third, if the water-surface altitude of the pond becomes less than the altitude of the pond bottom, the pond node is assumed to be dry. The pond altitude is then set equal to the altitude of the pond bottom at that node, and leakage from that node is set equal to zero.

For treatment of stream nodes, calculations are performed at each stream node and at appropriate stream reaches. At stream nodes one of the following three conditions will apply:

1. Flow occurs in the stream from the upstream node (Q_{in}), with leakage from the aquifer (Q_l) (fig. 3-a). In this case, outflow from the stream node (Q_{out}) is equal to the sum of the inflows (Q_{in} + Q_l), and stream depth is determined from the Manning formula.

2. Flow occurs in the stream from the upstream node (Q_{in}) with leakage to the aquifer (Q_l); for example, during ground-water pumpage near the stream. In this case, if the inflow rate (Q_{in}) is larger than leakage to the aquifer (Q_l), then outflow from the stream node (Q_{out}) is equal to inflow (Q_{in}) minus leakage (Q_l), and stream depth is determined from the Manning formula (fig. 3-b and 3-c). Figure 3-c shows a threshold condition that exists when the water table becomes detached from a flowing stream. Maximum leakage to the aquifer (Q_l) occurs when the aquifer head is equal to the altitude of the stream bottom. If, on the other hand, leakage from the stream node to the aquifer (Q_l) is larger than inflow (Q_{in}), then there will be no outflow (Q_{out}) from the stream node. The stream head, in this case, is equal to the altitude of the ground surface at the node, and leakage (Q_l) from the node is set equal to the inflow rate (Q_{in}) (fig. 3-d).

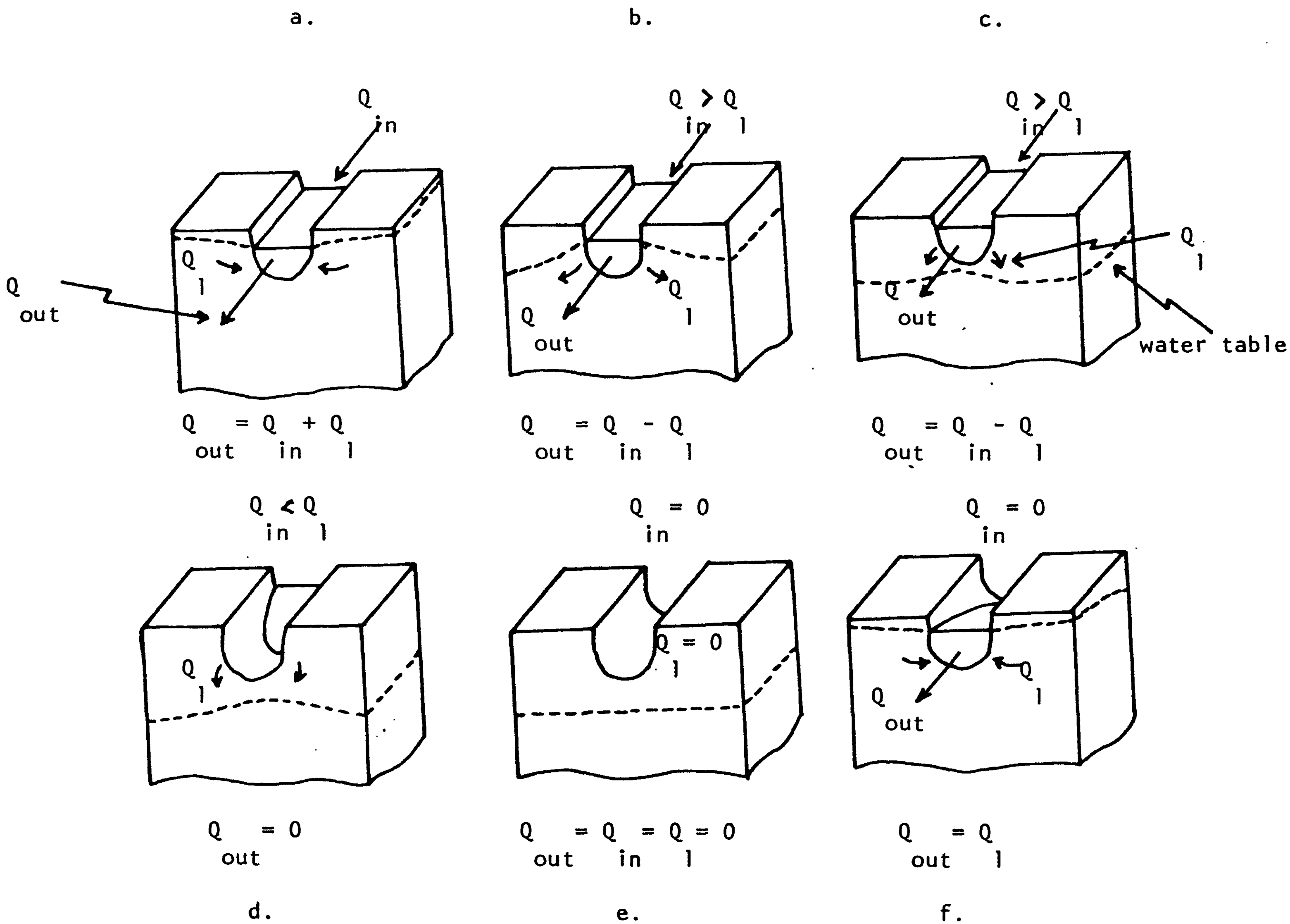


Figure 3.--Representation of stream-aquifer interaction.

3. No flow occurs in the stream from the upstream node (Q_{in} = 0), and stream head depends upon aquifer head. If the calculated aquifer head at the node is below the altitude of the stream bottom, outflow from the stream node (Q_{out}) is zero (fig. 3-e). If the aquifer head is above the altitude of the stream bottom, leakage from the aquifer (Q_l) to the stream node is calculated. In this case, outflow from the stream node (Q_{out}) is equal to leakage (Q_l), and stream depth is determined from the Manning formula (fig. 3-f).

Most main streams have tributaries feeding them. Treatment of each tributary stream node is the same as described above. Tributaries and ponds are numbered by segment(s_n) in downstream order. Numbering starts upstream, and increases sequentially as each tributary enters the main stream channel. Calculations of flow are done from lowest to highest order. In figure 4, for example, calculations of flow are done first in segments S_1 and S_2. The sum of the cumulative flows at the last downstream node of segments S_1 and S_2 are set equal to the inflow in the first upstream node of segment S_3. This procedure is continued until the nth segment is calculated. The pond segment is treated the same as stream segments in the numbering order sequence.

The average width, average bottom slope, and Manning's roughness coefficient are read into the modified program, as are inflow to the first upstream node and the stream segment number to which that stream segment discharges. In figure 4, for example, segments S_1 and S_2 discharge to S_3, and segments S_3 and S_4 discharge to S_5, etc. If no inflow is specified to the first upstream node, then the node is treated as described previously under condition number three.

MODEL TESTING

The modified program was tested first to insure that computational schemes in the original program had not changed. For this purpose, two steady-state problems were selected, one for idealized conditions and the other for an actual field situation. Both steady-state problems were run using the original and modified programs with SIP, ADI, and LSOR solver routines. The results from these programs were identical for both steady-state problems.

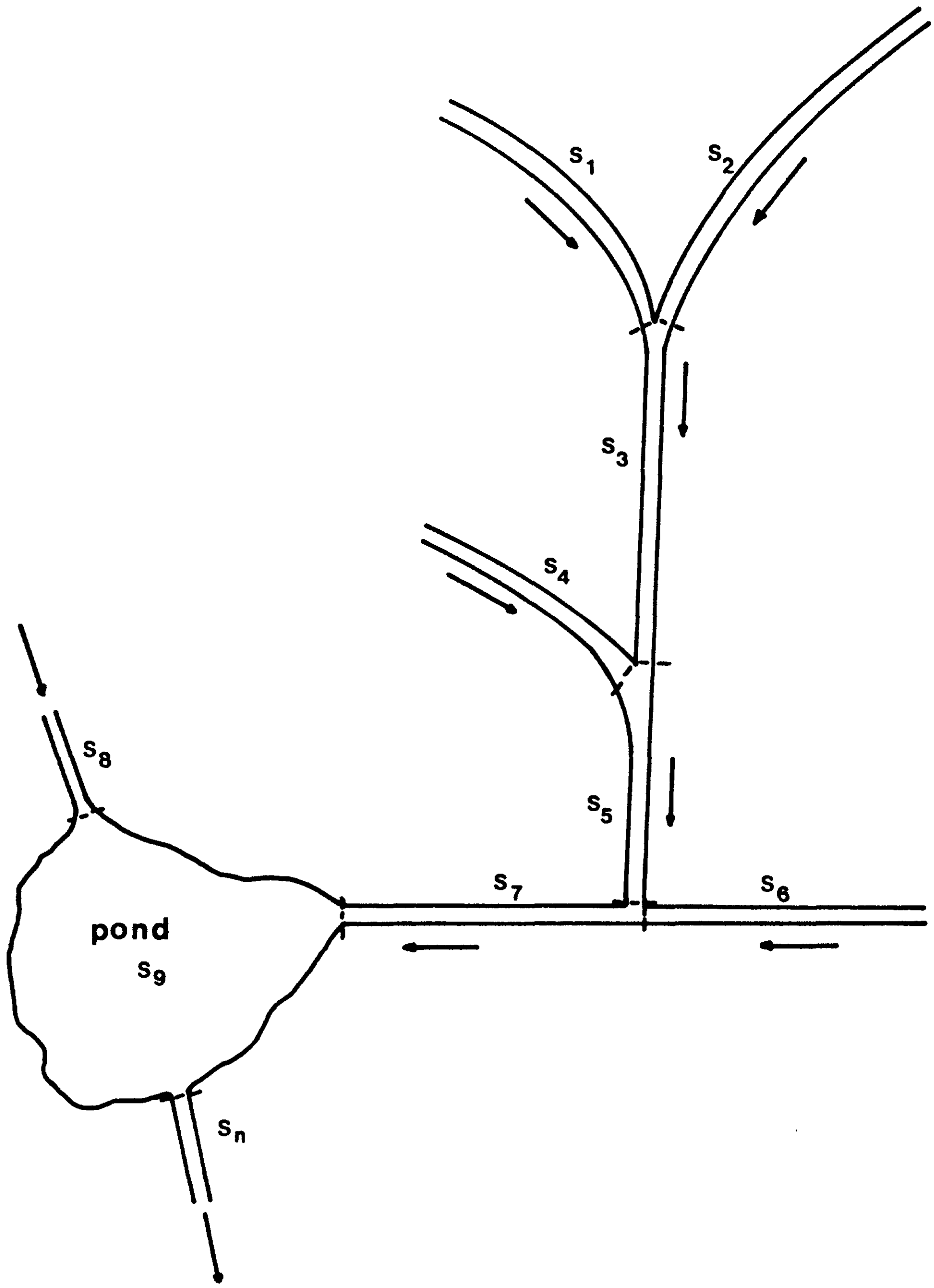


Figure 4.--Numbering sequence for surface-water segments.

The interaction between surface-water and ground-water systems was tested by modeling a hypothetical stream-aquifer system bounded on all sides by no-flow boundaries. A uniform recharge rate was applied to the aquifer and the model was simulated for steady-state. Using the steady-state heads as the starting condition, the model was simulated for one year. The results show that ground-water runoff from the total stream length at the end of the year equaled the recharge applied to the aquifer. This provided a useful check showing that inflow equaled outflow, and that the stream stage routine kept the mass balance of the total system intact.

To test the transient simulation capabilities of the modified program on a stream-aquifer system, the idealized aquifer shown in figure 5 was simulated. The results of numerical solutions were compared against analytical solutions given by Oakes and Wilkinson (1972). For this test the following values were assumed: storage coefficient, 0.2; streambed hydraulic conductivity, 1 ft/d; streambed thickness, 1 ft; and transmissivity, 3200 ft^2/d. Ground-water head in an observation well (fig. 5) and base flow from the entire aquifer area were simulated using the recharge distribution shown in figure 6. The model was simulated for six years to reach equilibrium, after which cyclic variations remained constant. The numerical results from the modified program for ground-water head (fig. 7) and base flow (fig. 8) are in good agreement with the analytical results presented by Oakes and Wilkinson (1972).

The comparisons indicate that the modified program: (1) is capable of simulating effects of transient stresses on stream-aquifer systems, and (2) can accurately simulate water-level and base-flow data for a given recharge distribution under constant stream depth and flow conditions. The capability of the modified program to simulate base flow for varying stream-head conditions was tested against a solution developed by Cooper and Rorabaugh (1963). The idealized aquifer shown in figure 5 was used to simulate flood wave passage through a stream. Comparison of base flows (fig. 9), for a unit length of stream (1 ft), indicates good agreement between numerical model simulations and analytical solutions developed by Cooper and Rorabaugh (1963). Small differences occur between numerical and analytical solutions because calculated stream heads in the model are not exactly the same as assumed in the analytical solution. This is because in the modified program, head changes as the stream loses or gains flow from the aquifer. Although small, the change in stream head results in a variation in base flow since leakage into (-) or from (+) the stream is a function of the stream head (equation 10).

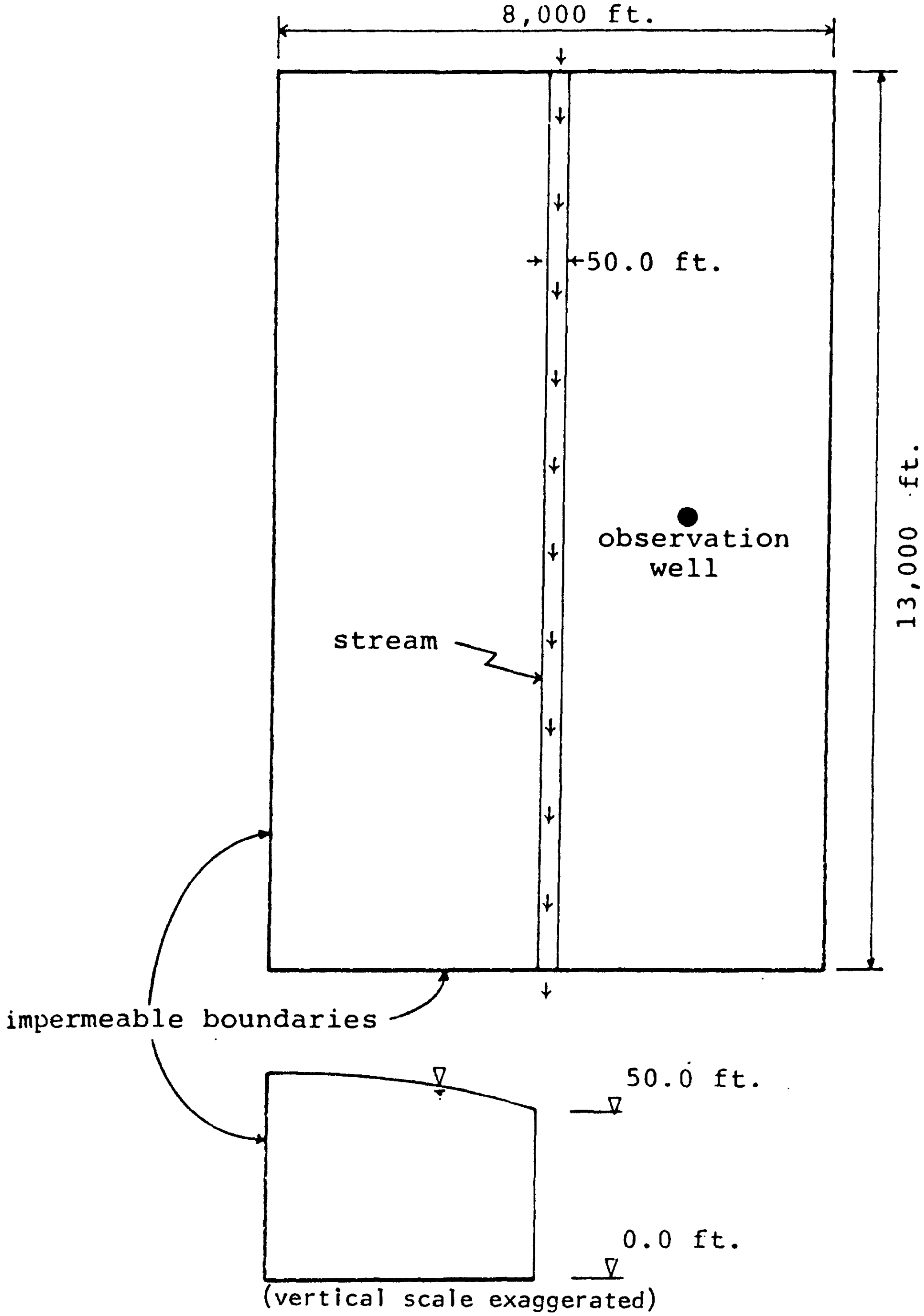


Figure 5.--Idealized aquifer for stream-aquifer system.

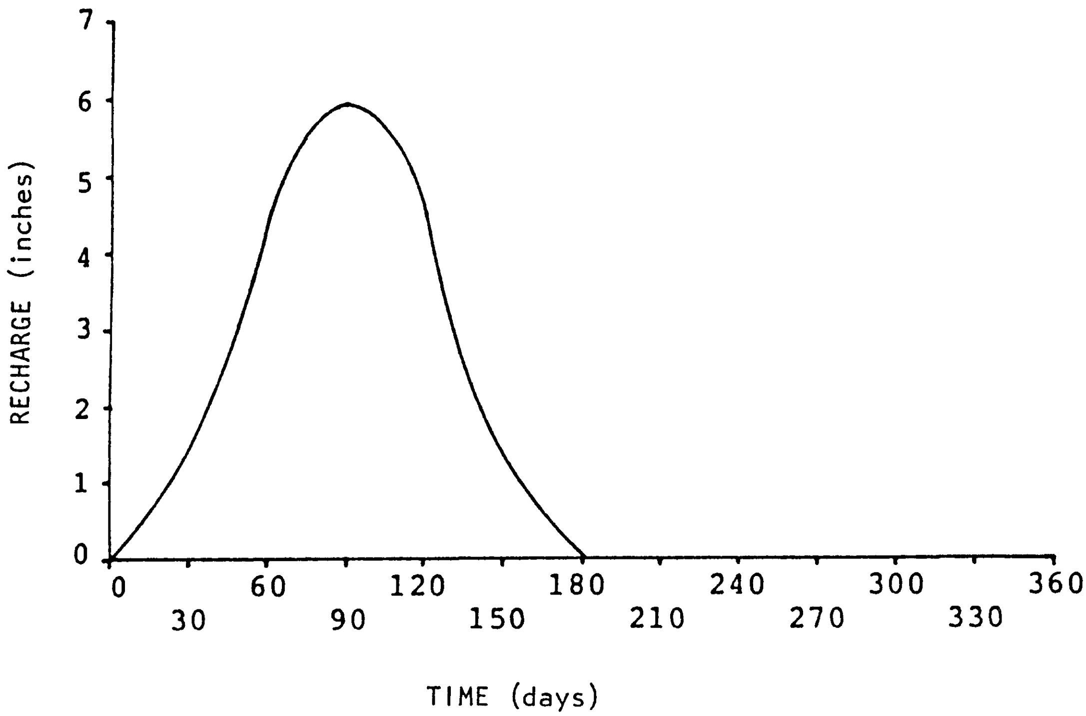


Figure 6.--Recharge distribution applied to the idealized aquifer for stream-aquifer system.

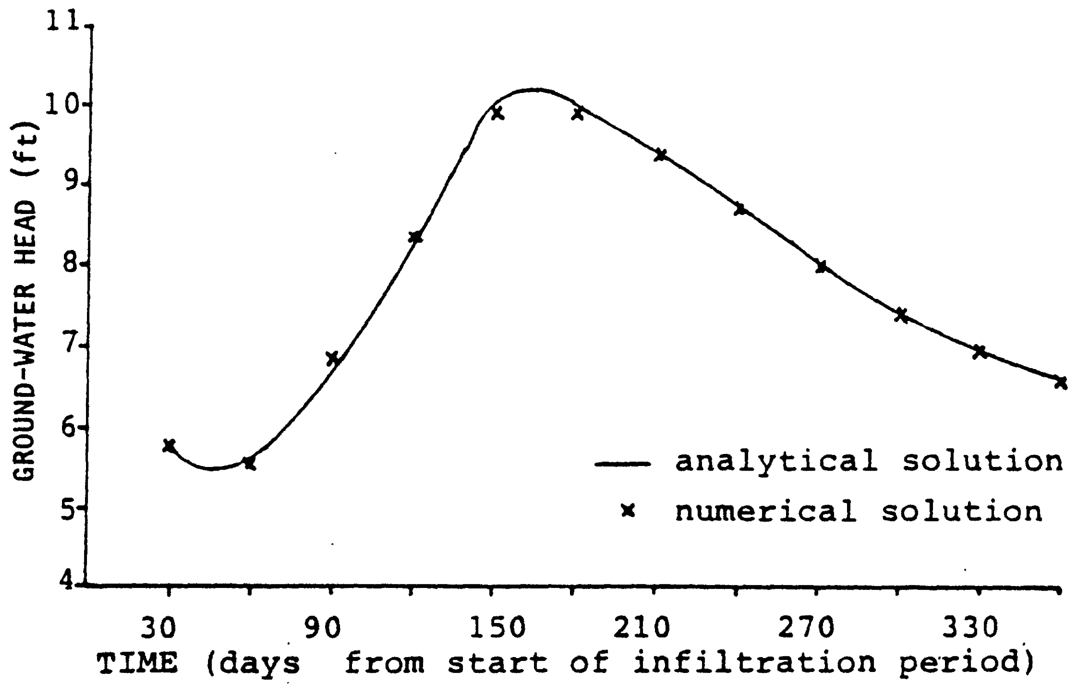


Figure 7.--Comparison of ground-water heads for numerical (model simulation) and analytical (Oakes and Wilkinson, 1972) solutions.

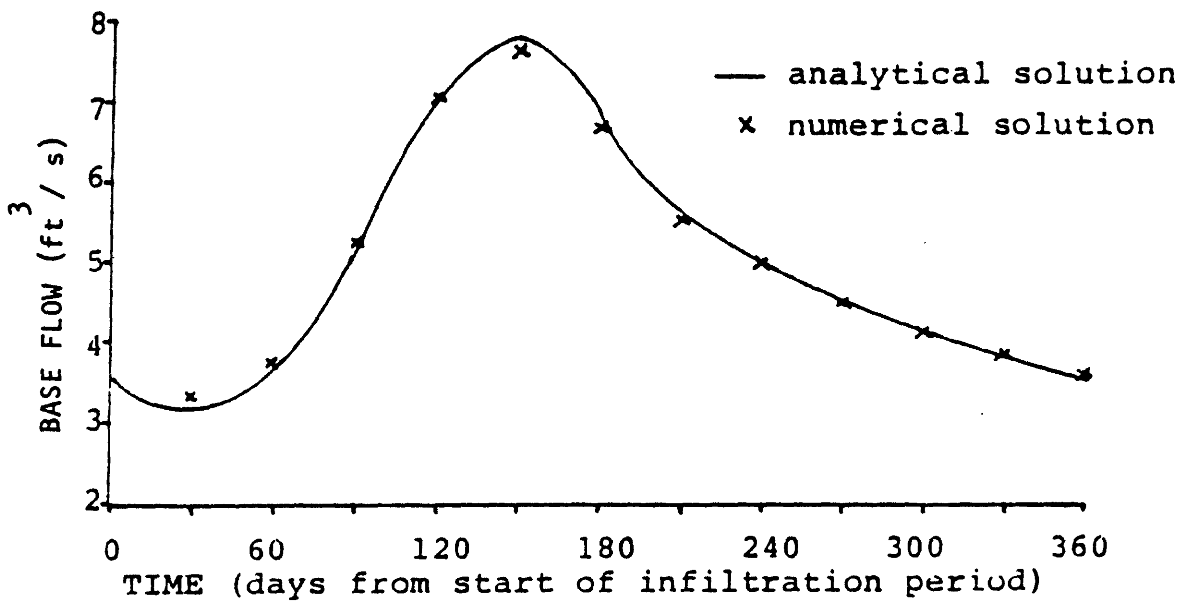
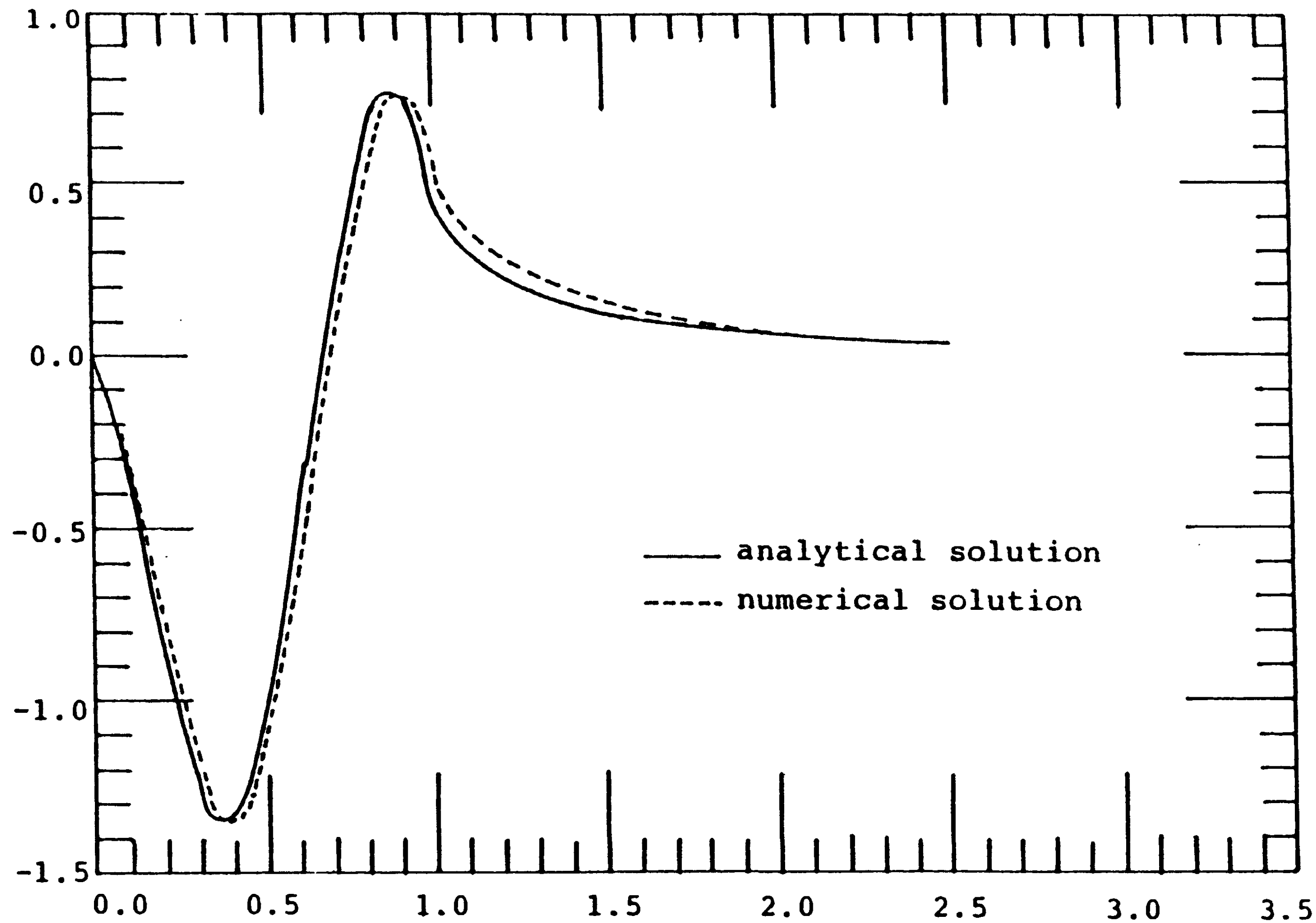


Figure 8.--Comparison of base flows for numerical (model simulation) and analytical (Oakes and Wilkinson, 1972) solutions.



$$\frac{t}{\tau} = \frac{\omega t}{2\pi}$$

$$\frac{Q}{(h_o/2)\sqrt{\omega TS}}$$

Figure 9.--Comparison of base flows due to sinusoidal flood wave for numerical (model simulation) and analytical (Cooper and Rorabaugh, 1963) solutions. (S=storage coefficient; T=transmissivity; h_o=peak flood wave height; ω=frequency of stage oscillation; and τ=period of stage oscillation).

The modified program was also tested against a one-dimensional transient model. The idealized aquifer's response to sinusoidal fluctuations in stream head were examined 1000 feet from the stream. The comparison shown in figure 10 indicates good agreement between one- and two-dimensional solutions.

Model output results, in table 2, show the modified program's capability to simulate a stream-pond-aquifer system. For this test, the idealized aquifer shown in figure 11 was simulated. Hydraulic conductivity of the confining layer beneath the stream and pond was assumed to be 1.0 ft/d and streambed thickness was assumed to be 1 ft. The streambed hydraulic conductivity was adjusted for a 20-foot wide stream using equation 11. The hydraulic conductivity of the homogeneous, isotropic, water-table aquifer was assumed to be 200 ft/d. The aquifer specific yield was assumed to be 0.2. If the pond were known to be in good hydraulic connection with the aquifer, then an appropriately higher specific yield should be assigned to aquifer grid blocks below the pond.

The model simulation was run for a six-year period, with ground-water recharge and evapotranspiration, to reach equilibrium conditions. A pumping well, located near the southern end of the pond, was pumped at 1.55 ft^3/s for one year. The objective of this test was to demonstrate that: (1) the modified program could handle stream reaches going dry for periods of time and then recover, (2) the interactions of the pond-aquifer, stream-aquifer, and pond-stream systems could be simulated, and (3) the mass balance of the system remained intact.

Table 2 shows that the mass balance of the system remained intact, since the sum of discharges from the aquifer (columns 2+4+7+9) closely matched the recharge applied to the aquifer. The small difference (0.50 ft^3/s) between recharge to (+70.40 ft^3/s), and discharge from (-70.90 ft^3/s) the aquifer was probably due to the four digit calculation performed on mass balances within the program.

The results shown in table 2 also indicate that the modified program can handle a stream-pond-aquifer system when stream reaches go dry during low-flow periods. Column 5 shows that discharge from the pond to the stream ceases when the pond altitude (column 3) is equal to or less than the altitude of the stream bottom (51.0 ft). However, column 6 shows that stream nodes during the 5th, 10th, and 11th months did not go dry, even though no discharge occurs to the stream from the pond. This happens because ground-water runoff to the stream (column 7) during those months is high enough to sustain some flow in the stream.

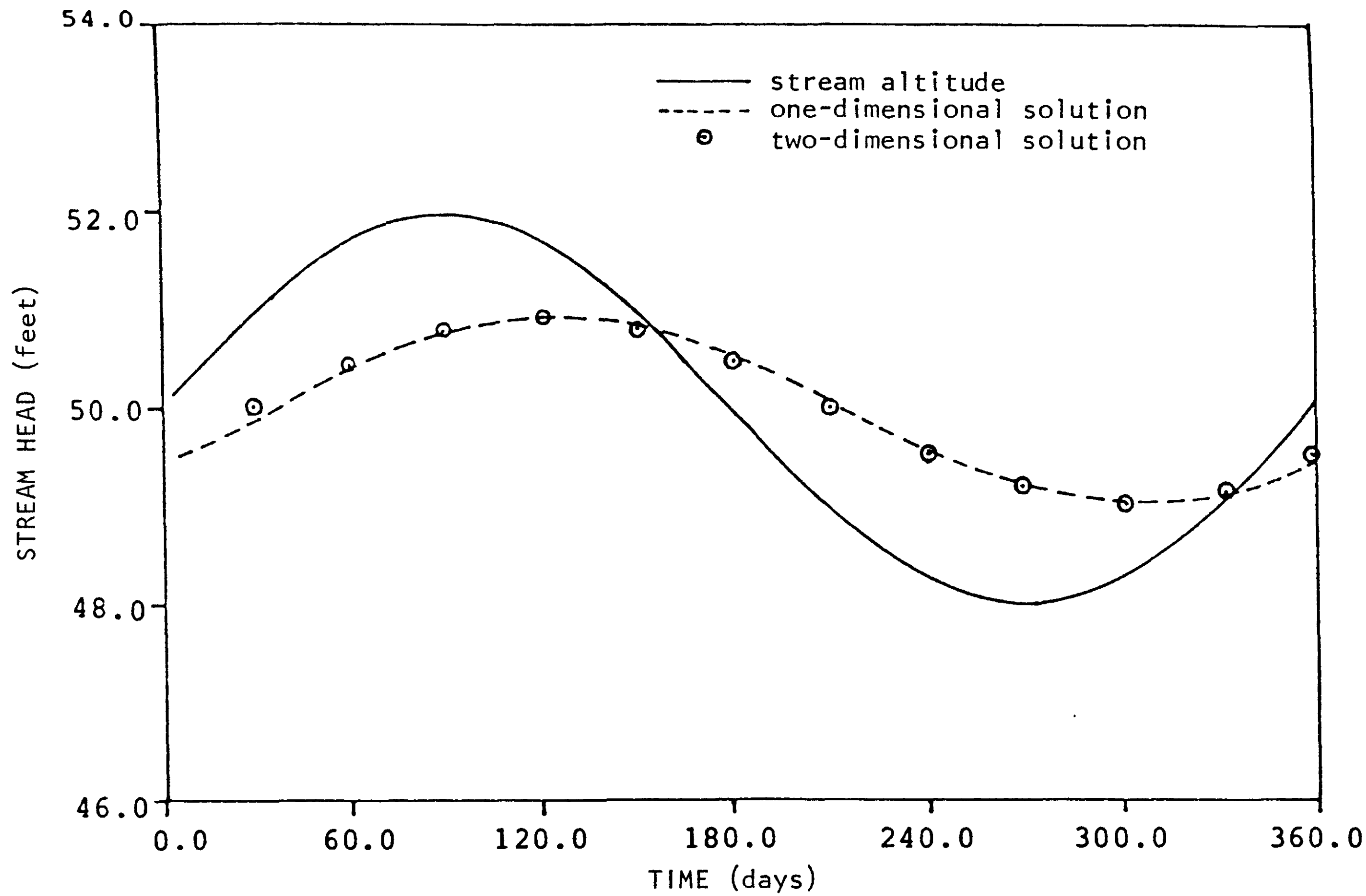


Figure 10.--Comparison of one- and two-dimensional solutions for a sinusoidal fluctuation of stream heads.

Table 2. Model results for stream-pond-aquifer interaction pumping 1.55 ft^3/s (- discharge from, + recharge into the aquifer).

(1) Month	(2) Well pumpage	(3) Pond altitude[1]	(4) Ground-water runoff to pond[2]	(5) Pond discharge to stream[2]	(6) Stream altitude[1,3]	(7) Ground-water runoff to stream[2]	(8) Stream discharge[2]	(9) Ground-water Evapotranspiration[2]	(10) Ground-water Recharge[2,4]
1	-1.55	51.4	-3.38	-3.25	51.3	-1.76	-5.01	-0.26	9.90
2	-1.55	51.6	-5.89	-5.76	51.4	-2.36	-8.12	-0.45	15.47
3	-1.55	51.3	-2.05	-1.92	51.2	-1.82	-3.74	-2.29	3.48
4	-1.55	51.1	+1.29	-0.56	51.1	-1.12	-1.68	-3.43	0.07
5	-1.55	50.5	+0.38	0.00	51.0	-0.75	-0.75	-4.52	2.54
6	-1.55	49.5	+0.41	0.00	DRY (2)	-0.30	-0.30	-5.00	0.16
7	-1.55	48.9	+0.50	0.00	DRY (3)	-0.12	-0.12	-4.78	0.84
8	-1.55	48.6	+0.41	0.00	DRY (3)	-0.12	-0.12	-3.45	2.31
9	-1.55	49.4	-0.39	0.00	DRY (1)	-0.64	-0.64	-1.93	10.40
10	-1.55	50.4	-0.42	0.00	51.0	-1.42	-1.42	-1.05	10.40
11	-1.55	51.0	-0.34	0.00	51.0	-1.26	-1.26	-0.25	5.73
12	-1.55	51.3	-2.00	-2.14	51.2	-1.52	-3.66	-0.22	9.10
Total	-18.60		-11.48	-13.63		-13.19	-26.82	-27.63	70.40

1 In feet. The altitude of the stream bottom is 51.0 feet.
2 In cubic feet per second (ft^3/s).
3 Altitude of the water surface at the first stream node discharging from the pond; numeral in parenthesis indicates the number of stream nodes downstream from the pond that went dry.
4 Total simulated discharge from the aquifer equals (columns 2+4+7+9) - 70.90.

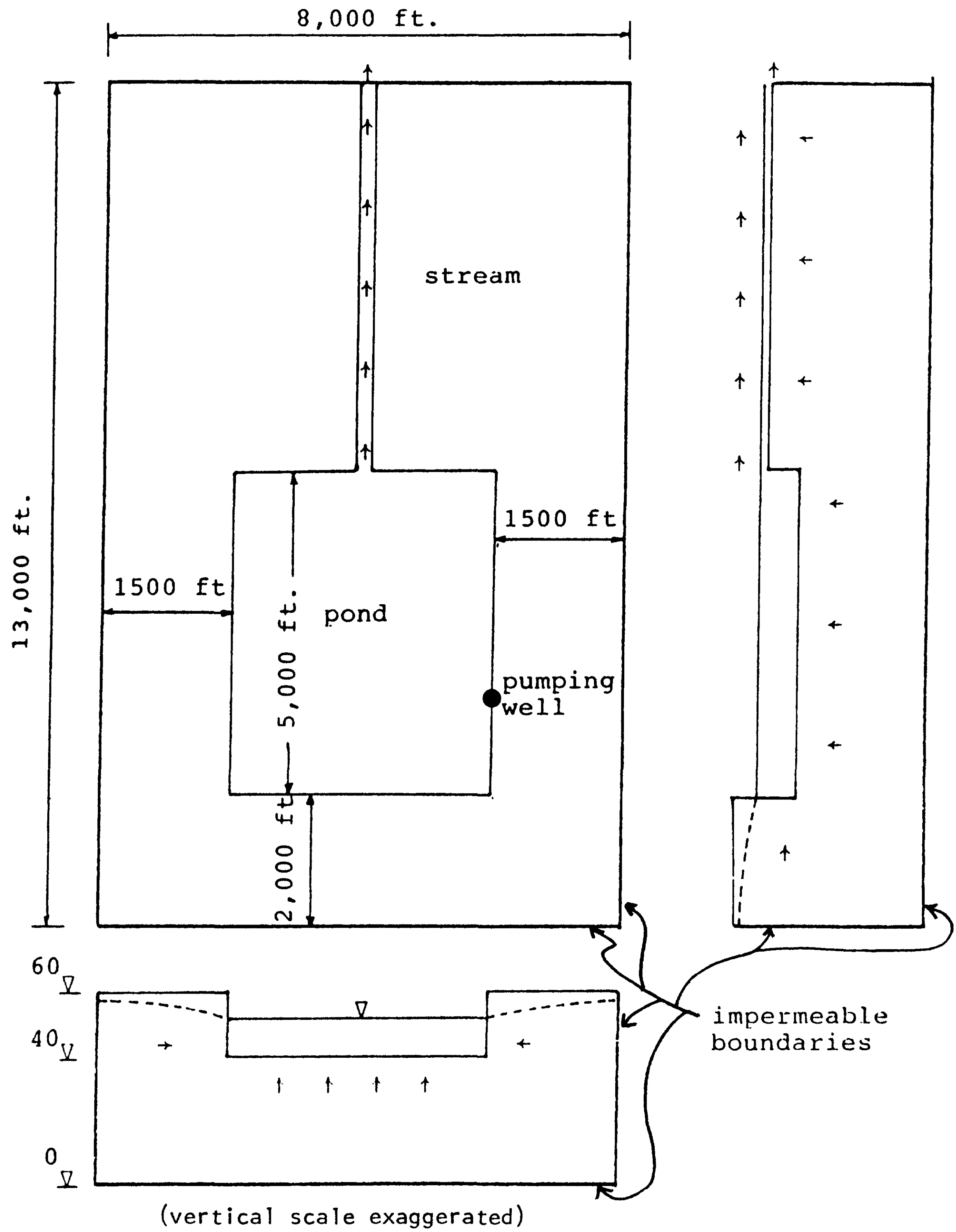


Figure 11.--Idealized aquifer for stream-pond-aquifer system.

The modified program described in this report was used to develop a field model of a stream-pond-aquifer system in the Beaver-Pasquiset ground-water reservoir in southern Rhode Island (Dickerman and Ozbilgin, written commun., 1983). The Beaver-Pasquiset model simulated breaks in the hydraulic connection between the stream-aquifer, and the pond-aquifer over stream and pond reaches influenced by drawdown from nearby pumping wells.

Complete documentation for the modified program includes: (1) explanation of modifications, (2) a flow chart, (3) data deck instructions, (4) input data, (5) output summary, and (6) a complete program listing. The modified program presented in this report can be used by the reader as a stand-alone manual. Users of the modified program should be aware that undiscovered errors in logic may exist.

REFERENCES

Cooper, H.H., Jr., and Rorabaugh, M.I., 1963, Ground-water movements and bank storage due to flood stages in surface streams: U.S. Geol. Survey Water-Supply Paper 1536-J, p. 343-366.

Oakes, D.B., and Wilkinson, W.B., 1972, Modeling of groundwater and surface water systems, I-Theoretical relationships between groundwater abstraction and base flow: Water Resources Board, Great Britain, Reading Bridge House, p. 37.

Streeter, V.L., 1971, Fluid mechanics: New York, McGraw-Hill Book Co., p. 278-280.

Streeter, V.L., and Wylie, E.B., 1979, Fluid mechanics: New York, McGraw-Hill Book Co., p. 227-232.

Trescott, P.C., Pinder, G.F., and Larson, S.P., 1976, Finite-difference model for aquifer simulation in two dimensions with results of numerical experiments: U.S. Geol. Survey Techniques of Water-Resources Inv., book 7, ch. Cl, 116 p.

White, F.M., 1979, Fluid mechanics: New York, McGraw-Hill Book Co., p. 602-607.

SUPPLEMENTAL DATA

Program Modifications

To allow for more convenient data handling, the method of data input and output for transient simulations was modified from the finite-difference model for aquifer simulation in two-dimensions (Trescott and others, 1976). A new entry was added to calculate transient surface-water heads from known flows. The generalized flow chart for the modified program is shown in figure 12.

The major modifications in the program were made in the MAIN program and in subroutines DATAI, COEF, and CHECKI. The rest of the program was modified only to the extent necessary to handle modifications in the MAIN program and in subroutines listed above. Format statements and specifications (for example, dimensions, common blocks ...etc.) were changed throughout the program. Except for the specifications part, computational schemes remain unchanged. The lines changed in the modified program were designated by dropping the first two letters in columns 73 and 74, and adding the "-" symbol to column 79 at the end of the line. For example, in the original program the line identified by code MAN 240 was changed to read N 2400-.

A new subroutine CALC was added to initiate the modified program for surface-water calculations. If CALC is not specified by the modeler as a simulation option, the program runs the original version. The CALC option is designed to work only with the LEAK or combined WATER/LEAK options.

Twelve new arrays were added to the program to accommodate calculations of surface-water heads. Five of the arrays have the size DIML x DIMW, while seven have the size NSEG. The dimensions of these arrays are calculated and passed to subroutines within the MAIN program. Statements calling the entry ARRAY in the MAIN program were moved to change the sequence of matrix data input. A new statement calling the entry ARRAY was added to read the new array RSEG (surface-water segment identification). Statements were added or moved between lines MAN2200 and MAN2500 to control the flow of computations in the modified program.

The method of data input and output for pumping periods in the entry NEWPER of subroutine DATAI was changed. Additional read statements were added to read new parameters and problem options. The rest of the changes in subroutine DATAI were made to handle those changes in entry NEWPER (for example, addition of new format statements, initialization of added variables, etc.).

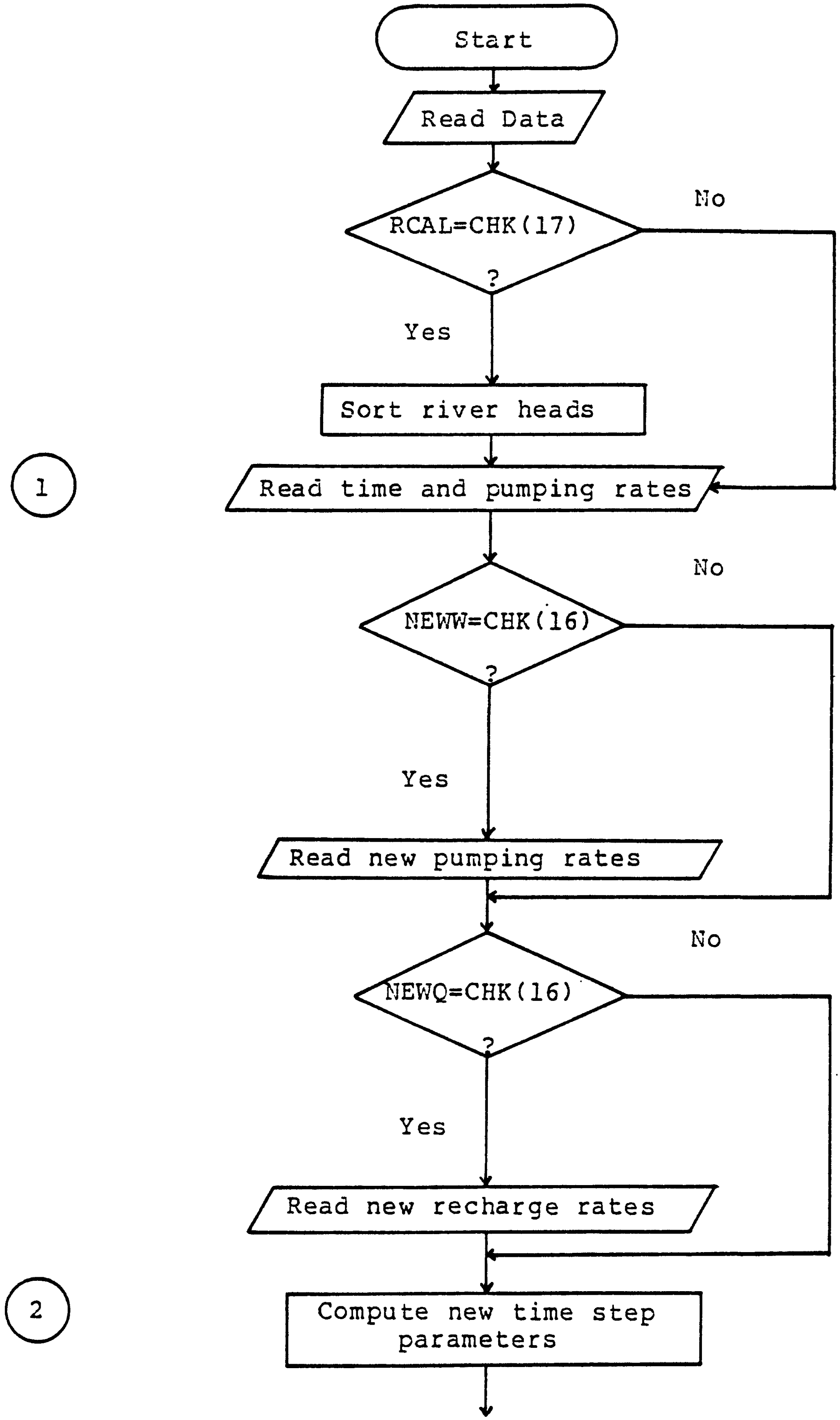
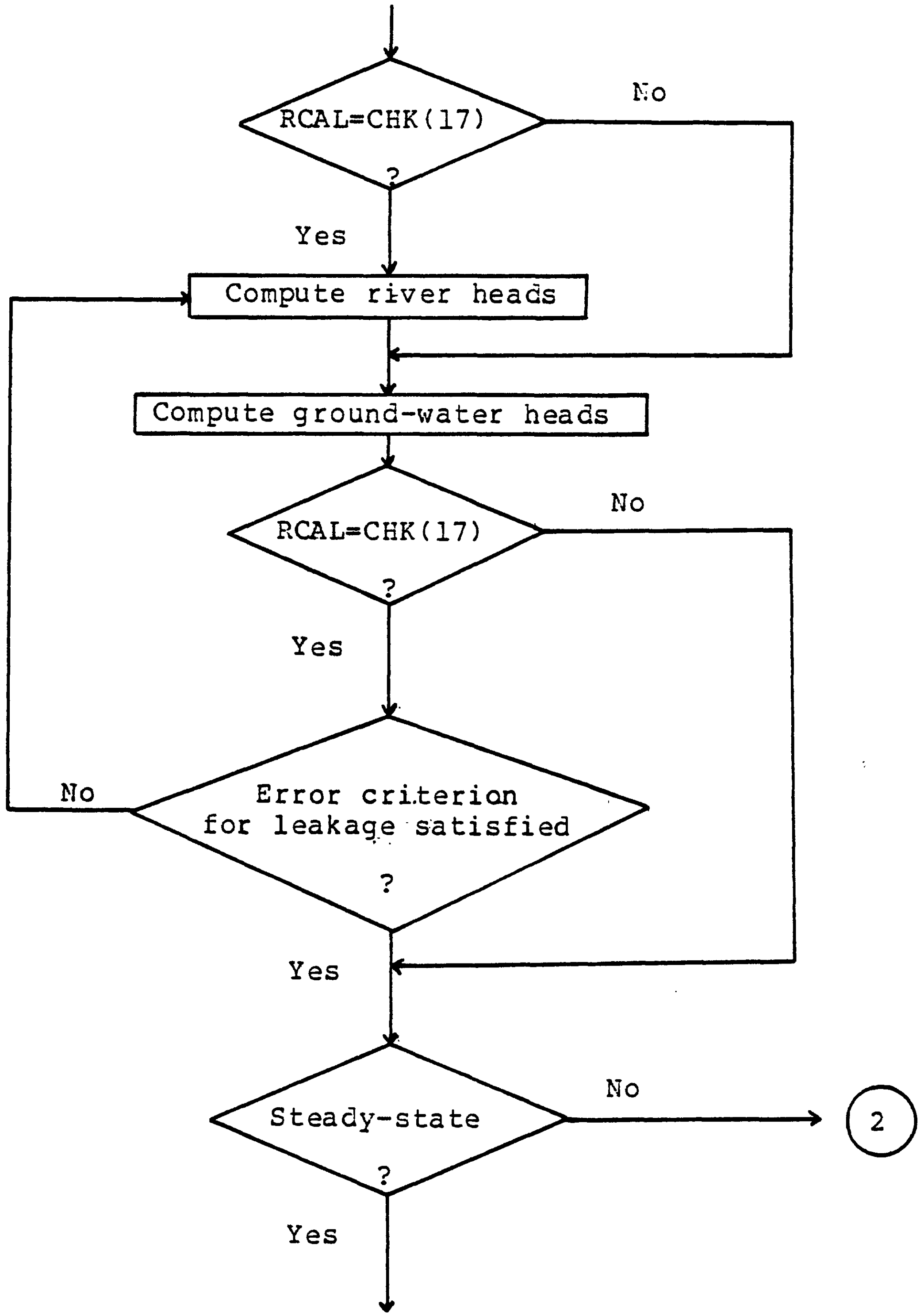
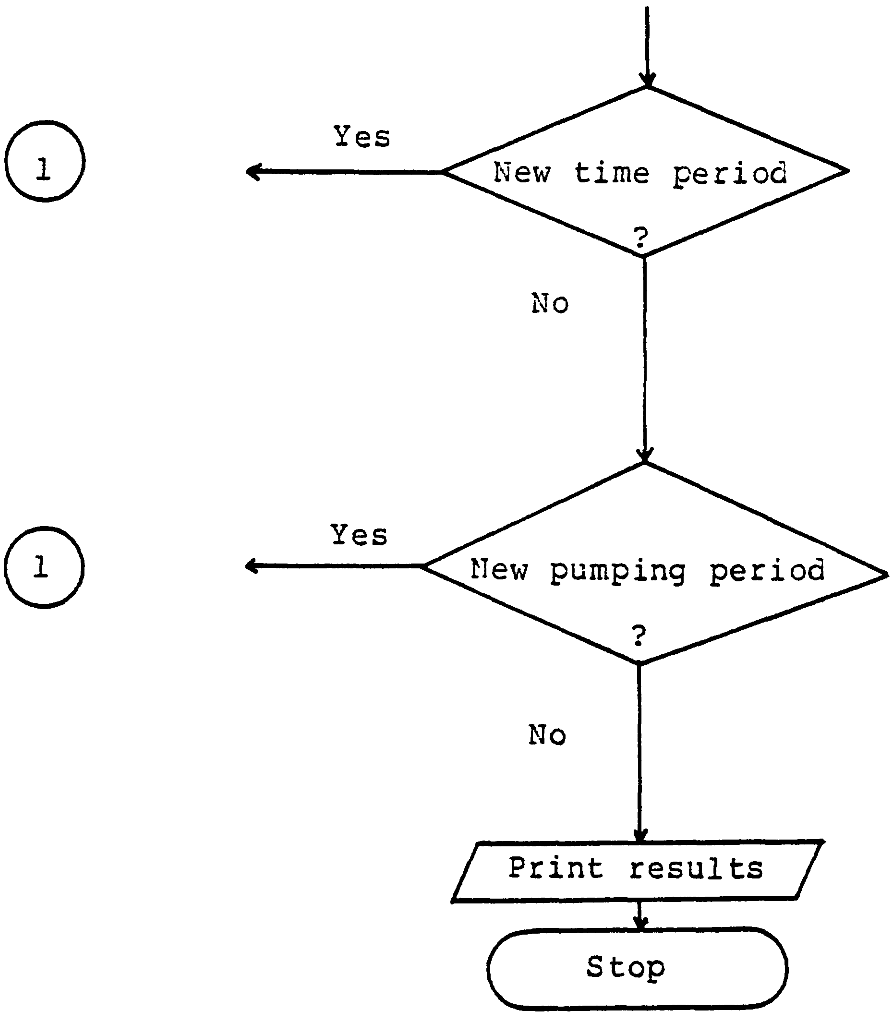


Figure 12.-- Simplified flow chart for the modified program.

flow chart - Continued



flow-chart - Continued



A new entry RSORT was added to subroutine COEF to order stream nodes by highest to lowest initial stream head, and store this order on array IRIV for later retrieval. This was necessary for both stream head calculations and an orderly printout of leakages from surface-water nodes.

The major changes in the modified program were made in subroutine CHECKI. A new entry HRIVER was added to calculate surface-water heads. The entry CWRITE was modified to handle transient output of leakages and to print calculated surface-water heads. Statements were added to the entry CHECK, as needed, to calculate leakages into surface-water bodies and to initialize some variables used in the new entry HRIVER.

Table 3.--Data deck instructions
(*Indicates new variable used in the modified program)

GROUP I: TITLE, SIMULATION OPTIONS, AND PROBLEM DIMENSIONS

This group of cards, which are read by the main program, contains data required to dimension the model. To specify an option on card 3, punch the characters underlined in the definition, starting in the first column of the field. For any option not used, leave the appropriate columns blank.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-80	20A4		
			HEADING	Any title the user wishes to print on one line at the start of the output.
2	1-48	12A4		
3	1-5	A4,1X	WATER	WATE for water table or combined water-table - artesian aquifer.
	6-10	A4,1X	LEAK	LEAK for an aquifer system including leakage from a surface water or confining bed. Required if CALC is specified on column 61.
	11-15	A4,1X	CONVRT	CONV for combined water-table - artesian aquifer.
	16-20	A4,1X	EVAP	EVAP to permit discharge by evapotranspiration.
	21-25	A4,1X	RECH	RECH to include a constant recharge rate.
	26-30	A4,1X	NUMS	SIP or LSOR or ADI to designate the equation-solving scheme.
	31-35	A4,1X	CHCK	CHEC to compute a mass balance.
	36-40	A4,1X	PNCH	PUNC for punched output at the end of the simulation.
	41-45	A4,1X	IDK1	DK1 to read initial head and mass balance parameters from disk (unit 4).

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	46-50	A4,1X	IDK2	DK2 to save (write) computed head, elapsed time, and mass balance parameters on disk (unit 4).
	51-55	A4,1X	NUM	NUME to print drawdown in numeric form.
	56-60	A4,1X	HEAD	HEAD to print the head matrix.
	61-54	A4,1X	*RCAL	CALC to calculate surface-water heads.
	(All variables on card 4 are integers)			
4	1-10	I10	DIML	Number of rows.
	11-20	I10	DIMW	Number of columns.
	21-30	I10	NW	Number of pumping wells for which drawdown is to be computed at a "real" well radius.
	31-40	I10	ITMAX	Maximum number of iterations per time step.
	41-50	I10	*NSEG	Number of surface-water segments to be specified.

NOTE - Steady-state simulations often require more than 50 iterations. Transient time steps usually require less than 30 iterations.

GROUP II: SCALAR PARAMETERS

The parameters required in every problem are underlined. The other parameters are required as noted; when not required, their location on the card can be left blank. The G format is used to read E, F, and I data. Minimize mistakes by always right-justifying data in the field. If F format data do not contain significant figures to the right of the decimal point, the decimal point can be omitted. Default typing of variables applies.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-4	A4	CONTR	CONT to generate a map of drawdown and (or) hydraulic head; for no maps insert a blank card.
	11-20	G10.0	XSCALE	Factor to convert model length unit to unit used in X direction on maps (that is, to convert from feet to miles, XSCALE=5280).
	21-30	G10.0	YSCALE	Factor to convert model length unit to unit used in Y direction on maps.
	31-40	G10.0	DINCH	Number of map units per inch.
	41-50	G10.0	FACT1	Factor to adjust value of drawdown printed.
	51-60	G10.0	FACT2	Factor to adjust value of head printed.
	61-68	A8	MESUR	Name of map length unit.

NOTE - For value of drawdown or head 52.57, for example:

FACT1 or FACT2	Printed value
.01	0
.1	5
1	52
10	525
100	***

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
2	1-10	G10.0	NPER	Number of pumping periods for this simulation.
	11-20	G10.0	KTH	Number of time steps between printouts.

NOTE - To print only the results for the final time step in a pumping period, make KTH greater than the expected number of time steps. The program always prints the results for the final time step.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	21-30	G10.0	ERR	Error criterion for closure (L).

NOTE - When the head change at all nodes on subsequent iterations is less than this value (for example 0.01 foot), the program has reached a solution for the time step.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	31-40	G10.0	EROR	Steady-state error criterion (L).

NOTE - If the head change between time steps in transient simulations is less than this amount, the pumping period is terminated.

	41-50	G10.0	*ERLEAK	Error criterion for surface water - ground water interaction (L^3/T).

NOTE - If the largest change in leakage from surface waters is less than this amount (for example 0.01 feet^3/sec.),the program has reached a solution for the time step when CALC option is specified in card 3 of Group I parameters. When CALC option is used, the program will try to satisfy ERR and ERLEAK simultaneously within the time step.

	51-60	G10.0	SS	Specific storage of confining bed (1/L)

NOTE - SS has a finite value only in transient simulations where leakage is a function of storage in the confining bed.

	61-70	G10.0	<u>LENGTH</u>	Definition depends on the numerical solution used: LSOR: number of LSOR iterations between 2-D corrections. ADI and SIP: number of iteration parameters; unless the program is modified, code 10 for SIP
	71-80	G10.0	<u>HMAX</u>	Definition depends on the numerical solution used: LSOR: acceleration parameter. ADI: maximum iteration parameter. SIP: value of B'.

NOTE - See the discussion of the numerical methods in Trescott and others (1976) for information on iteration parameters.

3	1-10	G10.0	<u>FACTX</u>	Multiplication factor for transmissivity in X direction.
	11-20	G10.0	<u>FACTY</u>	Multiplication factor for transmissivity in Y direction.

NOTE - FACTX=FACTY=1 for isotropic aquifers.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
4	1-20	G20.10	SUM	Parameters in which elapsed time and cumulative volumes
	21-40	G20.10	SUMP	for mass balance are stored. For the start of a simulation
	41-60	G20.10	PUMPT	insert three blank cards. For continuation of a
	61-80	G20.10	CFLUXT	previous run from punched output, remove the three
5	1-20	G20.10	QRET	blank cards and insert the first three cards of the
	21-40	G20-10	CHST	punched output from the previous run. If continuation
	41-60	G20.10	CHDT	is from interim storage on disk, the three blank cards
	61-80	G20.10	FLUXT	should remain.
6	1-20	G20.10	STORT	
	21-40	G20.10	ETFLXT	
	41-60	G20.10	FLXNT	

GROUP III: ARRAY DATA

Each of the following data sets, except for the first one (PHI), consists of a parameter card and, if the data set contains variable data, may include a set of data cards. Default typing applies except for M(I,J) which is a real array. Each parameter card contains five variables defined as follows:

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	1-10	G10.0	FACT	If IVAR=0, FACT is the value assigned to every element of the matrix; if IVAR=1, FACT is the multiplication factor for the following set of data cards.
	11-20	G10.0	IVAR	=0 if no data cards are to be read for this matrix; =1 if data cards for this matrix follow.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	21-30	G10.0	IPRN	=0 if input data for this matrix are to be printed; =1 if input data for this matrix are <u>not</u> to be printed.
	31-40	G10.0	IRECS	=0 if the matrix is being read from cards or if each element is being set equal to FACT; =1 if the matrix is to be read from disk (unit 2).
	41-50	G10.0	IRECD	=0 if the matrix is <u>not</u> to be stored on disk; =1 if the matrix being read from cards or set equal to FACT <u>is</u> to be stored on disk (unit 2) for later retrieval.

For the uniform starting head=100, FACT=100, IVAR=IPRN= IRECS=IRECD=0 and no data cards are required. The save matrix, which is being read from cards, on disk, for example (provided unit 2 has been defined on a DD statement; see technical information on pages 43-45 of original 1976 program), set FACT=1, IVAR=1, IPRN=IRECS=0, IRECD=1, and include the set of data cards (note that the input matrix will not be printed for IPRN=0). After this has been processed successfully, subsequent runs need only include a parameter card with the following: FACT=IVAR= IPRN=0, IRECS=1, IRECD=0. The set of data cards are not included and the matrix is input via unit 2 from disk storage.

When data cards are included, start each row on a new card.

To prepare a set of data cards for an array that is a function of space, the general procedure is to overlay the finite-difference grid on a contoured map of a parameter and record the average value of the parameter for each finite-difference block on coding forms according to the appropriate format. In general, record only significant digits and no decimal points (except for data set 2); use the multiplication factor to convert the data to their appropriate values. For example, if vertical hydraulic conductivity of the confining bed (RATE) ranges from 2 x 10^{-9} to 9 x 10^{-8} ft/sec., coded values should range from 2 to 90; the multiplication factor (FACT) would be 1.0E-09.

Arrays needed in every simulation are underlined. Omit parameter cards and data cards not used in the simulation (however, see the note for the S matrix).

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-80	8F10.4	PHI(I,J)	Head values for continuation of a previous run (L).

NOTE - For a new simulation this data set is omitted. Do not include a parameter card with this data set.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
2	1-80	8F10.4	STRT(I,J)	Starting head matrix (L).
3	1-80	20F4.0	S(I,J)	Storage coefficient (dimensionless).

NOTE - Always required. In addition to specifying storage coefficient values for artesian aquifers, this matrix is used to locate constant-head boundaries by coding a negative number at constant head nodes. At these nodes T or PERM must be greater than zero. For a problem with no constant-head nodes and that does not require S values, insert a blank parameter card.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
4	1-80	20F4.0	T(I,J)	Transmissivity (L^2/T)

NOTE - (1) Required for artesian aquifer simulation only.

(2) Zero values must be placed around the perimeter of the T or PERM matrix for reasons inherent in the computational scheme. If IVAR=0, zero values are automatically inserted around the border of the model.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
5	1-80	20F4.0	PERM(I,J)	Hydraulic conductivity(L/T) (see note 2 for data set 4).
6	1-80	20F4.0	BOTTOM (I,J)	Elevation of bottom of aquifer (L).
7	1-80	20F4.0	SY (I,J)	Specific yield (dimensionless).

NOTE - Data sets 5, 6, and 7 are required for water table or combined artesian - water table simulations.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
8	1-80	20F4.0	TOP(I,J)	Elevation of top of aquifer (L).

NOTE - Required only in combined artesian - water table simulations.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
9	1-80	20F4.0	RATE(I,J)	Hydraulic conductivity of confining bed (L/T).
10	1-80	20F4.0	RIVER(I,J)	Head on the other side of confining bed (also the surface-water head) (L).

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
11	1-80	20F4.0	M(I,J)	Thickness of confining bed (L).

NOTE - Data sets 9, 10, and 11 are required in simulations with leakage. They are required for calculation of surface-water heads. If the confining bed or streambed does not extend over the entire aquifer use the M matrix to locate the confining bed. If RATE and RIVER do not vary over the extent of the confining bed they can be initialized to a uniform value. For calculation of stream heads, a RIVER matrix is required. The RIVER matrix is used in the program to locate stream nodes in the order they appear from upstream to downstream. Therefore, stream head values must descend downstream to the end of a stream reach. This is not required for the pond areas.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
12	1-80	20F4.0	*RSEG(I,J)	Surface-water segment identification (dimensionless).

NOTE - Required in simulations with surface water option (when CALC is coded on column 61 of card 3 in Group I data, otherwise omit). The value left to the decimal point (whole part of a real number) designates the segment number. Mantissa (fraction part of a real number) is for classifying the segment as stream or pond, and to specify the type of calculation to be done. For example, for surface water segment number one;

mantissa = 0 if no calculation is to be done. Nodes are part of either a stream or pond segment. (RSEG=1.0).

mantissa=1 if no calculation is to be done, but leakage from each node of the segment is to be printed. Nodes are part of either a stream or a pond segment. (RSEG=1.1).

mantissa=2 if stream heads are to be calculated and leakage from each node of the segment is to be printed. Nodes are part of a stream segment (RSEG=1.2).

mantissa=3 if pond elevations are to be calculated and leakage from each node of the segment is to be printed. Nodes are part of a pond segment (RSEG=1.3).

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
13	1-80	20F4.0	GRND(I,J)	Land elevation (L).

NOTE - Required for simulations with surface water option and (or) evapotranspiration. For simulations with evapotranspiration option only, if GRND does not vary over the extent of the aquifer it can be initialized to a uniform value. For surface water simulations, value of GRND refers to the elevation of bottom of the pond or stream channel; and it may <u>not</u> be initialized to a uniform value.

14	1-80	8G10.0	DELX(J)	Grid spacing in X direction (L).
15	1-80	8G10.0	DELY(I)	Grid spacing in Y direction (L).

GROUP IV: PARAMETERS THAT CHANGE WITH THE PUMPING PERIOD

The program has three options for the simulation period.

1. To simulate a given number of time steps within a time period, set TMAX to a value larger than the expected simulation period. The program will use NUMT, CDLT, and DELT as coded.

2. To simulate a given time period within a pumping period, set NUMT larger than the number required for the simulation period (for example 100). The program will compute the exact DELT (which will be less than or equal to DELT coded) and NUMT to arrive exactly at TMAX at the last time step.

3. To simulate a given pumping period, set KNUM=KPER and TMAX, NUMT, CDLT, and DELT to appropriate values as described above.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-10	G10.0	KP	Number of the pumping period.
	11-20	G10.0	KPM1	Number of the previous pumping period.

NOTE - In general KPM1=0 if KP=1
KPM1=1 if KP=2, etc.
This causes the time parameter used in ENTRY CLAY to be set to zero and STRT to be initialized to PHI. However, for continuation of a previous pumping period KPM1=KP, and STRT and the time parameter are not affected.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	21-30	G10.0	*KPER	Number of time periods in this pumping period (for example, number of months in a pumping period).
	31-40	G10.0	*KNUM	Number of the present time period.
	41-50	G10.0	TMAX	Number of days in this time period.
	51-60	G10.0	NUMT	Number of time steps in this time period.
	61-70	G10.0	CDLT	Multiplying factor for DELT.

NOTE - 1.5 is commonly used.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
	71-80	G10.0	DELT	Initial time step in hours.
2	1-10	G10.0	*TPUMP	Number of days in this pumping period.
	11-20	G10.0	NWEL	Number of wells for this time period.
	21-30	G10.0	QET	Maximum evapotranspiration rate for this time period (L/T).
	31-40	G10.0	ETDIST	Depth at which evapotranspiration ceases below land surface (L).
	41-50	T10.0	*FACTQ	Multiplication factor for recharge matrix.

NOTE - The recharge rate from the previous time period is multiplied by FACTQ. If the present recharge rate is some factor of the previous recharge rate, assign the appropriate factor for FACTQ; otherwise FACTQ=1).

DATA SET 2. (NSEG cards)

COLUMNS	FORMAT	VARIABLE	DEFINITION
1-10	G10.0	*K	Surface water segment number.
11-20	G10.0	*WIDE(K)	Width of the stream channel (L), (WIDE(K)=0 for a pond segment).
21-30	G10.0	*SLOPE(K)	Slope of the stream channel bottom (dimensionless) (SLOPE(K)=0 for a pond segment).
31-40	G10.0	*ROUGH(K)	Manning's roughness coefficient for the stream segment (dimensionless) (ROUGH (K)=0 for a pond segment).
41-50	G10.0	*FLOWIN(K)	Inflow to a segment (for a stream segment, inflow to the first upstream node) (L^3/T).
51-60	G10.0	*IFLOUT(K)	The segment number to which segment K empties (IFLOUT(K)=0 if the surface water segment does not discharge into another surface water segment)

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-10	G10.0	*RAIN	Rainfall to a surface water segment, if pond (L/T).
	11-20	G10.0	*PANEVA	Evaporation rate from a pond surface (L/T).

If NEWQ=NO the following set of cards is to be omitted. When used, a parameter card is required for the data set (see definition of variables for the parameter card in Group III data).

DATA SET 3.

COLUMNS	FORMAT	VARIABLE	DEFINITION
1-80	20F4.0	QRE(I,J)	Recharge rate (L/T).

NOTE: (1) For each additional time period, another set of Group IV cards is required (that is, KPER sets of Group IV cards are required).

(2) For each additional pumping period, another KPER set of Group IV cards are required (that is, NPER x KPER sets of Group IV cards are required for the total simulation period).

(3) If another simulation is included in the same job, insert a blank card before the next Group I cards.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
3	1-5	A4,1X	*NEWW	YES to read new pumping rates for this time period; NO to use previous pumping rates.

NOTE - For the start of a simulation, code YES for NEWW to read the pumping rates; code NO for a problem with no pumping. For continuation, code NO for NEWW to use the pumping rates from previous time period; when YES is coded, the matrix WELL(I,J) is initialized to zero and new pumping rates are read for the time period.

COLUMNS	FORMAT	VARIABLE	DEFINITION
6-10	A4,1X	*NEWQ	YES to read new recharge rate for the time period; NO to use the recharge rate from previous time period.

NOTE - For the start of a simulation, code YES for NEWQ to read a constant recharge rate; code NO for a problem with no constant recharge. For continuation, code NO for NEWQ to use the recharge rate from previous time period; recharge rate from previous time period will be multiplied by FACTQ. When YES is coded for continuation of a problem, the matrix QRE(I,J) is initialized to zero and new recharge rate is read for the present time period.

If NEWW=NO, or for the start of a simulation NWEL=0, the following set of cards is omitted.

DATA SET 1. (NWEL cards)

COLUMNS	FORMAT	VARIABLE	DEFINITION
1-10	G10.0	I	Row location of well.
11-20	G10.0	J	Column location of well.
21-30	G10.0	WELL(I,J)	Pumping rate (L^3/T); negative for a pumping, positive for a recharge well.
31-40	G10.0	RADIUS	Real well radius (L).

NOTE - Radius is required only for those wells, if any, where computation of drawdown at a real well radius is to be made.

For problems without surface water option, the following set of cards is omitted.

Table 4.--Sample of input data for modified program.

```
-----------------
                                         STUDY OF PUMPING EFFECTS IN A POND-STREAM-AQUIFER SYSTEM
GROUP I                                   TESTED BY MELIH M. OZBILGIN
                  WATE LEAK NCON EVAP RECH SIP  CHEC PUNC NDK1 NDK2 NNUM HEAD CALC
                         11        15         0        100        2
-----------------
                          1       100      0.001      0.001      0.02      0.0      10
GROUP II                1.0       1.0

-----------------
          STRT         52.0         0
           S            0.0
          PERM     0.002900
          BOTTOM        0.0
           Sy           0.2
          RATE  {2.3140E-07         1
           A       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           T       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           E       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
                   0     0    050. 050. 050. 050. 050. 0     0     0     0     0     0     0     0
           M       0     0    050. 050. 050. 050. 050. 0     0     0     0     0     0     0     0
           A       0     0    050. 050. 050. 050. 050. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1 0
           T       0     0    050. 050. 050. 050. 050. 0     0     0     0     0     0     0     0
           R       0     0    050. 050. 050. 050. 050. 0     0     0     0     0     0     0     0
           I       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           X       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
                   0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
          RIVER        1.0              1
           I       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           V       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           E       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           R       0     0    052. 052. 052. 052. 052. 0     0     0     0     0     0     0     0
                   0     0    052. 052. 052. 052. 052. 0     0     0     0     0     0     0     0
GROUP III  M       0     0    052. 052. 052. 052. 052. 052. 051. 851. 651. 451. 351. 251. 1
           A       0     0    052. 052. 052. 052. 052. 0     0     0     0     0     0     0     0
           T       0     0    052. 052. 052. 052. 052. 0     0     0     0     0     0     0     0
           R       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           I       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           X       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           M           1.0              1
                   0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           M       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           A       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           T       0     0     0  1. 0  1. 0  1. 0  1. 0  1. 0     0     0     0     0     0     0     0
           R       0     0     0  1. 0  1. 0  1. 0  1. 0  1. 0     0     0     0     0     0     0     0
           I       0     0     0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0  1. 0
           X       0     0     0  1. 0  1. 0  1. 0  1. 0  1. 0     0     0     0     0     0     0     0
                   0     0     0  1. 0  1. 0  1. 0  1. 0  1. 0     0     0     0     0     0     0     0
                   0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
                   0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
                   0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
          RSEG         1.0              1
           S       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           E       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
           G       0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
                   0     0     0  1. 3  1. 3  1. 3  1. 3  1. 3     0     0     0     0     0     0     0
           M       0     0     0  1. 3  1. 3  1. 3  1. 3  1. 3     0     0     0     0     0     0     0
           A       0     0     0  1. 3  1. 3  1. 3  1. 3  1. 3  2. 2  2. 2  2. 2  2. 2  2. 2  2. 2  2. 2
           T       0     0     0  1. 3  1. 3  1. 3  1. 3  1. 3     0     0     0     0     0     0     0
```

```
        R      0    0    0  1.3  1.3  1.3  1.3  1.3    0    0    0    0    0    0    0
        I      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
        X      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
               0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
        GRND        1.0            1
        R     58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
        N     58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
        D     58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
OUP III       58   58   58   40   40   40   40   40   58   58   58   58   58   58   58
        M     58   58   58   40   40   40   40   40   58   58   58   58   58   58   58
        A     58   58   58   40   40   40   40   40   51   51   51   51   51   51   51
        T     58   58   58   40   40   40   40   40   58   58   58   58   58   58   58
        R     58   58   58   40   40   40   40   40   58   58   58   58   58   58   58
        I     58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
        X     58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
              58   58   58   58   58   58   58   58   58   58   58   58   58   58   58
        DELX     1000.0
        DELY        1.0            1
                   500.          500.       1000.       1000.       1000.       1000        1000.       1000
---------------   1000.          500.        500.
                      1             0          36           1          31.           1         1.0       744.0
                 1095.0            16.2226E-09          5.0         1.0
             YES   YES
                      5             8       -1.55
                      1           1.0       0.001       0.035         0.0           2
                      2          20.0       0.001       0.035         0.0           0
                     0.            0.
             8.7428E-08
        R             1             1          36           2          28.           1         1.0       672.0
        E        1095.0            16.8893E-09          5.0         1.0
        P    NO    YES
        E             1          20.0       0.001       0.035         0.0           2
        A             2          20.0       0.001       0.035         0.0           0
        T            0.            0.
        I    9.5162E-08
        N             1             1          36           3          31.           1         1.0       744.0
        G        1095.0            19.3339E-09          5.0         1.0
OUP IV       NO    YES
                      1          20.0       0.001       0.035         0.0           2
                      2          20.0       0.001       0.035         0.0           0
                     0.            0.
             1.4872E-07
        D             1             1          36           4          30.           1         1.0       720.0
        A        1095.0            15.4655E-08          5.0         1.0
        T    NO    YES
        A             1          20.0       0.001       0.035         0.0           2
                      2          20.0       0.001       0.035         0.0           0
                     0.            0.
             3.3436E-08
        S             1             1          36           5          31.           1         1.0       744.0
        E        1095.0            19.8006E-08          5.0         1.0
        T    NO    YES
        S             1          20.0       0.001       0.035         0.0           2
                      2          20.0       0.001       0.035         0.0           0
                     0.            0.
             6.4300E-10
                      1             1          36           6          30.           1         1.0       720.0
                 1095.0            11.3825E-07          5.0         1.0
             NO    YES
                      1          20.0       0.001       0.035         0.0           2
```

```
         2      20.0      0.001     0.035      0.0       0
        0.       0
2.4434E-08
         1         1         36         7      31.        1        1.0      744.0
    1095.0        11.6179E-07       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
1.5557E-09
         1         1         36         8      31.        1        1.0      744 0
    1095.0        11.5557E-07       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
8.0894E-09
         1         1         36         9      30.        1        1.0      720.0
    1095.0        11.1253E-07       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
2.2184E-08
         1         1         36        10      31.        1        1.0      744.0
    1095.0        16.2226E-08       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
1.0000E-07
         1         1         36        11      30.        1        1.0      720 0
    1095.0        13.2150E-08       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
1.0000E-07
         1         1         36        12      31.        1        1.0      744.0
    1095.0        17.7783E-09       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
5.5070E-08
         1         1         36        13      31.        1        1.0      744.0
    1095.0        16.2226E-09       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
8.7428E-08
         1         1         36        14      28.        1        1.0      672.0
    1095.0        16.8893E-09       5.0      1.0
NO    YES
         1      20.0      0.001     0.035      0.0       2
         2      20.0      0.001     0.035      0.0       0
        0.      0.
9.5162E-08
         1         1         36        15      31.        1        1.0      744.0
```

```
      1095.0         19.3339E-09      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
1.4872E-07
         1        1        36        16     30.      1     1.0     720.0
      1095.0         15.4655E-08      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
3.3436E-08
         1        1        36        17     31.      1     1.0     744.0
      1095.0         19.8006E-08      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
6.4300E-10
         1        1        36        18     30.      1     1.0     720.0
      1095.0         11.3825E-07      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
2.4434E-08
         1        1        36        19     31.      1     1.0     744.0
      1095.0         11.6179E-07      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
1.5557E-09
         1        1        36        20     31.      1     1.0     744.0
      1095.0         11.5557E-07      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
8.0894E-09
         1        1        36        21     30.      1     1.0     720.0
      1095.0         11.1253E-07      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
2.2184E-08
         1        1        36        22     31.      1     1.0     744.0
      1095.0         16.2226E-08      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
      0.        0.
1.0000E-07
         1        1        36        23     30.      1     1.0     720.0
      1095.0         13.2150E-08      5.0      1.0
NO    YES
         1     20.0     0.001     0.035     0.0      2
         2     20.0     0.001     0.035     0.0      0
```

0.	0.						
1.0000E-07							
1	1	36	24	31.	1	1.0	744.0
1095.0	17.7783E-09		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
5.5070E-08							
1	1	36	25	31.	1	1.0	744.0
1095.0	16.2226E-09		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
8.7428E-08							
1	1	36	26	28.	1	1.0	672.0
1095.0	16.8893E-09		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
9.5162E-08							
1	1	36	27	31.	1	1.0	744.0
1095.0	19.3339E-09		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
1.4872E-07							
1	1	36	28	30.	1	1.0	720.0
1095.0	15.4655E-08		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
3.3436E-08							
1	1	36	29	31.	1	1.0	744.0
1095.0	19.8006E-08		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
6.4300E-10							
1	1	36	30	30.	1	1.0	720.0
1095.0	11.3825E-07		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
2.4434E-08							
1	1	36	31	31.	1	1.0	744.0
1095.0	11.6179E-07		5.0	1.0			
NO YES							
1	20.0	0.001	0.035	0.0	2		
2	20.0	0.001	0.035	0.0	0		
0.	0.						
1.5557E-09							
1	1	36	32	31.	1	1.0	744.0
1095.0	11.5557E-07		5.0	1.0			

```
NO      YES
         1      20. 0     0. 001     0. 035     0. 0     2
         2      20. 0     0. 001     0. 035     0. 0     0
        0.         0.
8. 0894E-09
         1          1        36         33       30.      1      1. 0     720. 0
   1095. 0       11. 1253E-07         5. 0       1. 0
NO      YES
         1      20. 0     0. 001     0. 035     0. 0     2
         2      20. 0     0. 001     0. 035     0. 0     0
        0.         0.
2. 2184E-08
         1          1        36         34       31.      1      1. 0     744. 0
   1095. 0       16. 2226E-08         5. 0       1. 0
NO      YES
         1      20. 0     0. 001     0. 035     0. 0     2
         2      20. 0     0. 001     0. 035     0. 0     0
        0.         0.
1. 0000E-07
         1          1        36         35       30.      1      1. 0     720. 0
   1095. 0       13. 2150E-08         5. 0       1. 0
NO      YES
         1      20. 0     0. 001     0. 035     0. 0     2
         2      20. 0     0. 001     0. 035     0. 0     0
        0.         0.
1. 0000E-07
         1          1        36         36       31.      1      1. 0     744. 0
   1095. 0       17. 7783E-09         5. 0       1. 0
NO      YES
         1      20. 0     0. 001     0. 035     0. 0     2
         2      20. 0     0. 001     0. 035     0. 0     0
        0.         0.
5. 5070E-08
//
```

Table 5.--Program listing of computer source code.

```
C     ******************************************************************MAN  10
C                         FINITE-DIFFERENCE MODEL                       MAN  20
C                                   FOR                                 MAN  30
C                   SIMULATION OF GROUND-WATER FLOW                     MAN  40
C                            IN TWO DIMENSIONS                          MAN  50
C                                                                       MAN  60
C          BY P. C. TRESCOTT, G. F. PINDER AND S. P. LARSON             MAN  70
C                         U. S. GEOLOGICAL SURVEY                       MAN  80
C                             SEPTEMBER, 1975                           MAN  90
C                                                                       N  901-
C                     MODIFIED BY MELIH M. OZBILGIN                     N  902-
C                             SEPTEMBER, 1982                           N  903-
C     ******************************************************************MAN 100
C     MAIN PROGRAM TO DIMENSION DIGITAL MODEL AND CONTROL SEQUENCE       MAN 110
C     OF COMPUTATIONS                                                   MAN 120
C     ------------------------------------------------------------------MAN 130
C     SPECIFICATIONS:                                                   MAN 140
      REAL *4KEEP,M,HEADNG(32)                                          MAN 150
      REAL *8PHI,G,BE,TEMP,Z,YY                                         MAN 160
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,MAN 170
     1CONTR,LEAK,RECH,SIP,ADI,NEWW,NEWQ,RCAL                            N 1800-
C                                                                       MAN 190
      DIMENSION Y(70000), L(49), IFMT1(9), IFMT2(9), IFMT3(9), NAME(108)N 2000-
     1, YY(1)                                                           N 2100-
      EQUIVALENCE (YY(1),Y(1))                                          MAN 220
C                                                                       MAN 230
      COMMON /SARRAY/ VF4(11),CHK(17)                                   N 2400-
      COMMON /MPARAM/ FACTQ,NEWW,NEWQ,ERLEAK,DAYS                       N 2401-
      COMMON /ARSIZE/ IZ,JZ,IP,JP,IR,JR,IC,JC,IL,JL,IS,JS,IH,IMAX,IMX1,IN 2500-
     1K,IA,JA                                                           N 2501-
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LEMAN 260
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,MAN 270
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,MAN 280
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DIMAN 290
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL                       N 3000-
      COMMON /CK/ ETFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT,IN 3001-
     1RCHK,RAIN,PANEVA,IRCHK1                                           N 3002-
C                                                                       MAN 310
      DATA IFMT1/4H(1HO,4H,I5,,4H10E1,4H1.3/,4H(1H ,4H,5X,,4H10E1,4H1.3)MAN 320
     1,4H)   /                                                          MAN 330
      DATA IFMT2/4H('0',4H,I2,,4H2X,2,4HOF6.,4H1/(5,4HX,20,4HF6.1,4H))  MAN 340
     1,4H    /                                                          MAN 350
      DATA IFMT3/4H(1HO,4H,I5,,4H14F9,4H.5/(,4H1H ,,4H5X,1,4H4F9.,4H5)) MAN 360
     1,4H    /                                                          MAN 370
      DATA NAME/2*4H    ,4H STO,4HRAGE,4H COE,4HFFIC,4HIENT,4*4H    ,4H MAN 380
     1   T,4HRANS,4HMISS,4HIVIT,4HY   ,2*4H    ,4H   A,4HQUIF,4HER H,4HYDMAN 390
     2RA,4HULIC,4H CON,4HDUCT,4HIVIT,4HY   ,4H    ,4H   A,4HQUIF,4HER B,MAN 400
     34HASE ,4HELEV,4HATIO,4HN   ,3*4H    ,4H   S,4HPECI,4HFIC ,4HYIEL,4MAN 410
     4HD   ,4*4H    ,4HAQUI,4HFER ,4HTOP ,4HELEV,4HATIO,4HN   ,4H    ,4HMAN 420
     5CONF,4HININ,4HG BE,4HD HY,4HDRAU,4HLIC ,4HCOND,4HUCTI,4HVITY,3*4H MAN 430
     6    ,4H RIV,4HER H,4HEAD ,4*4H    ,4H   C,4HONFI,4HNING,4H BED,4H TMAN 440
     7HI,4HCKNE,4HSS  ,2*4H    ,4H   L,4HAND ,4HSURF,4HACE ,4HELEV,4HATIMAN 450
     8O,4HN   ,3*4H    ,4H ARE,4HAL R,4HECHA,4HRGE ,4HRATE,3*4H    ,4H  N 4600-
     9RI,4HVER ,4HNODE,4H IDE,4HNTIF,4HICAT,4HION ,4H    /              N 4601-
C                                                                       MAN 470
      DEFINE FILE 2(15,2624,U,KKK)                                      N 4800-
C     ..................................................................MAN 490
C                                                                       MAN 500
C     ---READ TITLE,PROGRAM OPTIONS AND PROGRAM SIZE---                 MAN 510
```

```
   10 READ (R,370) HEADNG                                                MAN 520
      WRITE (P,360) HEADNG                                               MAN 530
      READ (R,380) WATER,LEAK,CONVRT,EVAP,RECH,NUMS,CHCK,PNCH,IDK1,IDK2, MAN 540
     1NUM,HEAD,RCAL                                                      N 5500-
      WRITE (P,390) WATER,LEAK,CONVRT,EVAP,RECH,NUMS,CHCK,PNCH,IDK1,IDK2MAN 560
     1,NUM,HEAD,RCAL                                                     N 5700-
      IF (NUMS.EQ.CHK(11).OR.NUMS.EQ.CHK(12).OR.NUMS.EQ.CHK(13)) GO TO 2MAN 580
     10                                                                  MAN 590
      WRITE (P,350)                                                      MAN 600
      STOP                                                               MAN 610
   20 READ (R,320) DIML,DIMW,NW,ITMAX,NSEG                               N 6200-
      WRITE (P,340) DIML,DIMW,NW,ITMAX,NSEG                              N 6300-
C                                                                        MAN 640
C     ---COMPUTE DIMENSIONS FOR ARRAYS---                                MAN 650
      IZ=DIML                                                            MAN 660
      JZ=DIMW                                                            MAN 670
      INSEG=NSEG                                                         N 6701-
      IF (NSEG.EQ.0) INSEG=1                                             N 6702-
      IH=MAX0(1,NW)                                                      MAN 680
      IMAX=MAX0(DIML,DIMW)                                               MAN 690
      ISIZ=DIML*DIMW                                                     MAN 700
      ISUM=2*ISIZ+1                                                      MAN 710
      IMX1=ITMAX+1                                                       MAN 720
      L(1)=1                                                             MAN 730
      DO 30 I=2,4                                                        MAN 740
      L(I)=ISUM                                                          MAN 750
   30 ISUM=ISUM+2*IMAX                                                   MAN 760
      DO 40 I=5,16                                                       MAN 770
      L(I)=ISUM                                                          MAN 780
   40 ISUM=ISUM+ISIZ                                                     MAN 790
      IF (WATER.NE.CHK(2)) GO TO 60                                      MAN 800
      DO 50 I=17,19                                                      MAN 810
      L(I)=ISUM                                                          MAN 820
   50 ISUM=ISUM+ISIZ                                                     MAN 830
      IP=DIML                                                            MAN 840
      JP=DIMW                                                            MAN 850
      GO TO 80                                                           MAN 860
   60 DO 70 I=17,19                                                      MAN 870
      L(I)=ISUM                                                          MAN 880
   70 ISUM=ISUM+1                                                        MAN 890
      IP=1                                                               MAN 900
      JP=1                                                               MAN 910
   80 IF (LEAK.NE.CHK(9)) GO TO 100                                      MAN 920
      DO 90 I=20,22                                                      MAN 930
      L(I)=ISUM                                                          MAN 940
   90 ISUM=ISUM+ISIZ                                                     MAN 950
      IR=DIML                                                            MAN 960
      JR=DIMW                                                            MAN 970
      GO TO 120                                                          MAN 980
  100 DO 110 I=20,22                                                     MAN 990
      L(I)=ISUM                                                          MAN1000
  110 ISUM=ISUM+1                                                        MAN1010
      IR=1                                                               MAN1020
      JR=1                                                               MAN1030
  120 IF (CONVRT.NE.CHK(7)) GO TO 130                                    MAN1040
      L(23)=ISUM                                                         MAN1050
      ISUM=ISUM+ISIZ                                                     MAN1060
      IC=DIML                                                            MAN1070
      JC=DIMW                                                            MAN1080
      GO TO 140                                                          MAN1090
```

```
130 L(23)=ISUM                                                    MAN1100
    ISUM=ISUM+1                                                   MAN1110
    IC=1                                                          MAN1120
    JC=1                                                          MAN1130
140 IF (EVAP.NE.CHK(6).AND.RCAL.NE.CHK(17)) GO TO 150             N11400-
    L(24)=ISUM                                                    MAN1150
    ISUM=ISUM+ISIZ                                                MAN1160
    IL=DIML                                                       MAN1170
    JL=DIMW                                                       MAN1180
    GO TO 160                                                     MAN1190
150 L(24)=ISUM                                                    MAN1200
    ISUM=ISUM+1                                                   MAN1210
    IL=1                                                          MAN1220
    JL=1                                                          MAN1230
160 IF (NUMS.NE.CHK(11)) GO TO 180                                MAN1240
    DO 170 I=25,28                                                MAN1250
    L(I)=ISUM                                                     MAN1260
170 ISUM=ISUM+ISIZ                                                MAN1270
    IS=DIML                                                       MAN1280
    JS=DIMW                                                       MAN1290
    GO TO 200                                                     MAN1300
180 DO 190 I=25,28                                                MAN1310
    L(I)=ISUM                                                     MAN1320
190 ISUM=ISUM+1                                                   MAN1330
    IS=1                                                          MAN1340
    JS=1                                                          MAN1350
200 DO 210 I=29,31                                                MAN1360
    L(I)=ISUM                                                     MAN1370
210 ISUM=ISUM+DIMW                                                MAN1380
    DO 220 I=32,33                                                MAN1390
    L(I)=ISUM                                                     MAN1400
220 ISUM=ISUM+DIML                                                MAN1410
    L(34)=ISUM                                                    MAN1420
    ISUM=ISUM+IH                                                  MAN1430
    L(35)=ISUM                                                    MAN1440
    ISUM=ISUM+2*IH                                                MAN1450
    IF (MOD(ISUM,2).EQ.0) ISUM=ISUM+1                             MAN1460
    CONTINUE                                                      MAN1470
230 L(36)=ISUM                                                    MAN1480
    ISUM=ISUM+2*IMAX                                              MAN1490
    L(37)=ISUM                                                    MAN1500
    ISUM=ISUM+IMX1                                                MAN1510
    IF (LEAK.NE.CHK(9)) GO TO 235                                 N15101-
    DO 231 I=38,40                                                N15102-
    L(I)=ISUM                                                     N15103-
231 ISUM=ISUM+ISIZ                                                N15104-
    IF (RCAL.NE.CHK(17)) GO TO 233                                N15105-
    L(41)=ISUM                                                    N15106-
    ISUM=ISUM+ISIZ                                                N15107-
    L(42)=ISUM                                                    N15108-
    ISUM=ISUM+ISIZ                                                N15109-
    IA=DIML                                                       N15110-
    JA=DIMW                                                       N15111-
    DO 232 I=43,49                                                N15112-
    L(I)=ISUM                                                     N15113-
232 ISUM=ISUM+INSEG                                               N15114-
    IK=INSEG                                                      N15115-
    GO TO 237                                                     N15116-
233 DO 234 I=41,49                                                N15117-
    L(I)=ISUM                                                     N15118-
```

```
  234 ISUM=ISUM+1                                                    N15119-
      IK=1                                                           N15120-
      IA=1                                                           N15121-
      JA=1                                                           N15122-
      GO TO 237                                                      N15123-
  235 DO 236 I=38,49                                                 N15124-
      L(I)=ISUM                                                      N15125-
  236 ISUM=ISUM+1                                                    N15126-
      IK=1                                                           N15127-
      IA=1                                                           N15128-
      JA=1                                                           N15129-
  237 ISUM=ISUM+1                                                    N15130-
      WRITE (P,330) ISUM                                             MAN1520
C                                                                    MAN1530
C     ---PASS INTIIAL ADDRESSES OF ARRAYS TO SUBROUTINES---          MAN1540
      CALL DATAI(Y(L(1)),Y(L(7)),Y(L(8)),Y(L(9)),Y(L(10)),Y(L(11)),Y(L(1MAN1550
     12)),Y(L(13)),Y(L(14)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(18)),Y(L(19))MAN1560
     2,Y(L(20)),Y(L(21)),Y(L(22)),Y(L(23)),Y(L(24)),Y(L(29)),Y(L(32)),Y(MAN1570
     3L(34)),Y(L(35)),Y(L(38)),Y(L(39)),Y(L(40)),Y(L(44)),Y(L(45)),Y(L(4N15800-
     46)),Y(L(47)),Y(L(48)))                                         N15801-
      CALL STEP(Y(L(1)),Y(L(5)),Y(L(7)),Y(L(8)),Y(L(9)),Y(L(14)),Y(L(17)MAN1590
     1),Y(L(18)),Y(L(23)),Y(L(29)),Y(L(30)),Y(L(32)),Y(L(34)),Y(L(35)),YMAN1600
     2(L(37)),Y(L(13)),Y(L(38)),Y(L(42)))                            N16100-
      IF (NUMS.EQ.CHK(11)) CALL SOLVE1(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),YMAN1620
     1(L(5)),Y(L(6)),Y(L(7)),Y(L(9)),Y(L(12)),Y(L(13)),Y(L(14)),Y(L(15))MAN1630
     2,Y(L(16)),Y(L(25)),Y(L(26)),Y(L(27)),Y(L(28)),Y(L(29)),Y(L(31)),Y(MAN1640
     3L(32)),Y(L(33)),Y(L(37)),Y(L(10)),Y(L(11)),Y(L(24)),Y(L(19)),Y(L(2MAN1650
     43)),Y(L(20)),Y(L(22)),Y(L(21)))                                MAN1660
      IF (NUMS.EQ.CHK(12)) CALL SOLVE2(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),YMAN1670
     1(L(5)),Y(L(6)),Y(L(7)),Y(L(9)),Y(L(12)),Y(L(13)),Y(L(14)),Y(L(15))MAN1680
     2,Y(L(16)),Y(L(25)),Y(L(26)),Y(L(27)),Y(L(28)),Y(L(29)),Y(L(31)),Y(MAN1690
     3L(32)),Y(L(33)),Y(L(37)),Y(L(10)),Y(L(11)),Y(L(24)),Y(L(19)),Y(L(2MAN1700
     43)),Y(L(20)),Y(L(22)),Y(L(21)))                                MAN1710
      IF (NUMS.EQ.CHK(13)) CALL SOLVE3(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),YMAN1720
     1(L(5)),Y(L(6)),Y(L(7)),Y(L(9)),Y(L(12)),Y(L(13)),Y(L(14)),Y(L(15))MAN1730
     2,Y(L(16)),Y(L(25)),Y(L(26)),Y(L(27)),Y(L(28)),Y(L(29)),Y(L(31)),Y(MAN1740
     3L(32)),Y(L(33)),Y(L(36)),Y(L(37)),Y(L(10)),Y(L(11)),Y(L(24)),Y(L(1MAN1750
     49)),Y(L(23)),Y(L(20)),Y(L(22)),Y(L(21)))                       MAN1760
      CALL COEF(Y(L(1)),Y(L(5)),Y(L(6)),Y(L(7)),Y(L(8)),Y(L(9)),Y(L(10))MAN1770
     1,Y(L(11)),Y(L(12)),Y(L(14)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(18)),Y(MAN1780
     2L(19)),Y(L(20)),Y(L(21)),Y(L(22)),Y(L(23)),Y(L(24)),Y(L(29)),Y(L(3MAN1790
     32)),Y(L(40)),Y(L(41)),Y(L(43)))                                N18000-
      CALL CHECKI(Y(L(1)),Y(L(5)),Y(L(6)),Y(L(7)),Y(L(9)),Y(L(10)),Y(L(1MAN1810
     11)),Y(L(12)),Y(L(13)),Y(L(14)),Y(L(15)),Y(L(17)),Y(L(18)),Y(L(19))MAN1820
     2,Y(L(20)),Y(L(21)),Y(L(22)),Y(L(23)),Y(L(24)),Y(L(29)),Y(L(32)),Y(N18300-
     3L(38)),Y(L(39)),Y(L(40)),Y(L(41)),Y(L(43)),Y(L(44)),Y(L(45)),Y(L(4N18301-
     46)),Y(L(47)),Y(L(48)),Y(L(42)),Y(L(16)),Y(L(49)))              N18302-
      CALL PRNTAI(Y(L(1)),Y(L(8)),Y(L(9)),Y(L(12)),Y(L(14)),Y(L(29)),Y(LMAN1840
     1(32)))                                                         MAN1850
C     ...............................................................MAN1860
C                                                                    MAN1870
C     ---START COMPUTATIONS---                                       MAN1880
C     ************************                                       MAN1890
C     ---READ AND WRITE DATA FOR GROUPS II AND III---                MAN1900
      CALL DATAIN                                                    MAN1910
      CALL ARRAY(Y(L(12)),IFMT3,NAME(1),2)                           MAN1920
      IF (WATER.EQ.CHK(2)) GO TO 240                                 MAN1930
      CALL ARRAY(Y(L(9)),IFMT3,NAME(10),3)                           MAN1940
      GO TO 250                                                      MAN1950
  240 CALL ARRAY(Y(L(17)),IFMT1,NAME(19),4)                          MAN1960
```

```
      CALL ARRAY(Y(L(18)),IFMT2,NAME(28),5)                             MAN1970
      CALL ARRAY(Y(L(19)),IFMT3,NAME(37),6)                             MAN1980
  250 IF (CONVRT.EQ.CHK(7)) CALL ARRAY(Y(L(23)),IFMT2,NAME(46),7)       MAN1990
      IF (LEAK.NE.CHK(9)) GO TO 260                                     MAN2000
      CALL ARRAY(Y(L(20)),IFMT1,NAME(55),8)                             MAN2010
      CALL ARRAY(Y(L(21)),IFMT2,NAME(64),9)                             MAN2020
      CALL ARRAY(Y(L(22)),IFMT2,NAME(73),10)                            MAN2030
      CALL ARRAY(Y(L(40)),IFMT2,NAME(100),15)                           N20301-
  260 IF (EVAP.EQ.CHK(6).OR.RCAL.EQ.CHK(17)) CALL ARRAY(Y(L(24)),IFMT2,NN20400-
     1AME(82),11)                                                       N20401-
C     IF (RECH.EQ.CHK(10)) CALL ARRAY(Y(L(13)),IFMT1,NAME(91),12)       MAN2050
      CALL MDAT                                                         MAN2060
C                                                                       MAN2070
C     ---INITIALIZE TRANSMISSIVITY VALUES IN WATER TABLE PROBLEM---     MAN2080
      KT=0                                                              MAN2090
      IF (WATER.EQ.CHK(2)) CALL TRANS                                   MAN2100
C                                                                       MAN2110
C     ---COMPUTE ITERATION PARAMETERS---                                MAN2120
      IF (NUMS.EQ.CHK(11)) CALL ITER1                                   MAN2130
      IF (NUMS.EQ.CHK(12)) CALL ITER2                                   MAN2140
      IF (NUMS.EQ.CHK(13)) CALL ITER3                                   MAN2150
C                                                                       MAN2160
C     ---INITIALIZE PARAMETERS FOR ALPHAMERIC MAP---                    MAN2170
      IF (CONTR.EQ.CHK(3)) CALL MAP                                     MAN2180
C                                                                       MAN2190
C     ---COMPUTE T COEFFICIENTS FOR ARTESIAN PROBLEM---                 MAN2200
      IF (WATER.NE.CHK(2)) CALL TCOF                                    MAN2210
C                                                                       MAN2220
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-MAN2230
      IF (RCAL.EQ.CHK(17)) CALL RSORT                                   N22301-
  270 CALL NEWPER                                                       MAN2240
C                                                                       MAN2250
      KT=0                                                              MAN2260
      IFINAL=0                                                          MAN2270
      IERR=0                                                            MAN2280
      IF (RECH.NE.CHK(10)) GO TO 280                                    N22801-
      IF (NEWQ.EQ.CHK(16)) CALL ARRAY(Y(L(13)),IFMT1,NAME(91),12)       N22802-
C                                                                       MAN2290
C     ---START NEW TIME STEP COMPUTATIONS---                            MAN2300
  280 CALL NEWSTP                                                       MAN2310
      IRCHK1=1                                                          N23101-
      IRCHK=1                                                           N23102-
      IF (RCAL.EQ.CHK(17)) IRCHK=0                                      N23103-
      IF (RCAL.EQ.CHK(17)) IRCHK1=0                                     N23104-
      IF (RCAL.EQ.CHK(17)) CALL HRIVER                                  N23105-
C                                                                       MAN2320
C     ---COMPUTE TRANSIENT PART OF LEAKAGE TERM---                      MAN2330
  282 IF (LEAK.EQ.CHK(9).AND.SS.NE.0.) CALL CLAY                        MAN2340
C                                                                       MAN2350
C     ---ENTER APPROPIATE SOLUTION ROUTINE AND COMPUTE SOLUTION---      MAN2360
      IF (NUMS.EQ.CHK(11)) CALL NEWITA                                  MAN2370
      IF (NUMS.EQ.CHK(12)) CALL NEWITB                                  MAN2380
      IF (NUMS.EQ.CHK(13)) CALL NEWITC                                  MAN2390
C                                                                       N23901-
C     ---CALCULATE RIVER HEADS IF OPTION SPECIFIED---                   N23902-
C                                                                       N23903-
      IF (RCAL.NE.CHK(17)) GO TO 284                                    N23904-
      CALL CHECK                                                        N23905-
C     CALL CWRITE                                                       N23906-
      IF (IRCHK.EQ.1) GO TO 284                                         N23907-
```

```
      CALL HRIVER                                                       N23908-
      GO TO 282                                                         N23909-
C                                                                       MAN2400
C     ---CHECK FOR STEADY STATE AND PRINT OUTPUT AT DESIGNATED          MAN2410
C     TIME STEPS---                                                     MAN2420
  284 CALL STEADY                                                       MAN2430
C                                                                       MAN2440
C     ---LAST TIME STEP IN PUMPING PERIOD ?---                          MAN2450
      IF (IFINAL.NE.1) GO TO 280                                        MAN2460
C                                                                       MAN2470
C     ---CHECK FOR NEW PUMPING PERIOD---                                MAN2480
      IF (KP.LE.NPER) GO TO 270                                         N24900-
C                                                                       MAN2500
C     ---DISK OUTPUT IF DESIRED---                                      MAN2510
      IF (IDK2.NE.CHK(15)) GO TO 290                                    MAN2520
      CALL DISK                                                         MAN2530
C                                                                       MAN2540
C     ---PUNCHED OUTPUT IF DESIRED---                                   MAN2550
  290 IF (PNCH.NE.CHK(1)) GO TO 300                                     MAN2560
      CALL PUNCH                                                        MAN2570
C                                                                       MAN2580
C     ---CHECK FOR NEW PROBLEM---                                       MAN2590
  300 READ (R,320,END=310) NEXT                                         MAN2600
      IF (NEXT.EQ.0) GO TO 10                                           MAN2610
  310 STOP                                                              MAN2620
C     ..................................................................MAN2630
C                                                                       MAN2640
C     ---FORMATS---                                                     MAN2650
C     ------------------------------------------------------------------MAN2660
C                                                                       MAN2670
C                                                                       MAN2680
  320 FORMAT (5I10)                                                     N26900-
  330 FORMAT ('0',54X,'WORDS OF Y VECTOR USED =',I7)                    MAN2700
  340 FORMAT ('0',62X,'NUMBER OF ROWS =',I5/60X,'NUMBER OF COLUMNS =',I5MAN2710
     1/9X,'NUMBER OF WELLS FOR WHICH DRAWDOWN IS COMPUTED AT A SPECIFIEDMAN2720
     2 RADIUS =',I5,/,39X,'MAXIMUM PERMITTED NUMBER OF ITERATIONS =',I5,N27300-
     3/,43X,'NUMBER OF RIVER SEGMENTS SPECIFIED =',I5)                  N27301-
  350 FORMAT ('-',36X,'NO EQUATION SOLVING SCHEME SPECIFIED, EXECUTION TMAN2740
     1ERMINATED'/37X,58('*'))                                           MAN2750
  360 FORMAT ('1',60X,'U. S. G. S.'//55X,'FINITE-DIFFERENCE MODEL'/65X,'MAN2760
     1FOR'/51X,'SIMULATION OF GROUND-WATER FLOW'//60X,'JANUARY, 1975'//4N27700-
     28X,'MODIFIED BY MELIH M. OZBILGIN, PH.D.',//59X,'DECEMBER,  1982',N27800-
     3//133('*')/'0',32A4//133('*'))                                    N27801-
  370 FORMAT (20A4)                                                     MAN2790
  380 FORMAT (16(A4,1X))                                                MAN2800
  390 FORMAT ('-SIMULATION OPTIONS:  ',13(A4,4X))                       MAN2810
      END                                                               MAN2820-
```

```
      SUBROUTINE DATAI(PHI,STRT,SURI,T,TR,TC,S,QRE,WELL,TL,SL,PERM,BOTTODAT  10
     1M,SY,RATE,RIVER,M,TOP,GRND,DELX,DELY,WR,NWR,PICK,PICKUP,RSEG,FLOWIT  200-
     2N,WIDE,SLOPE,IFLOUT,ROUGH)                                          T  201-
C     ----------------------------------------------------------------DAT  30
C     READ AND WRITE INPUT DATA                                         DAT  40
C     ----------------------------------------------------------------DAT  50
C                                                                       DAT  60
C     SPECIFICATIONS:                                                   DAT  70
      REAL *8PHI,DBLE,XLABEL,YLABEL,TITLE,XN1,MESUR                     DAT  80
      REAL *4M                                                          DAT  90
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,DAT 100
     1CONTR,LEAK,RECH,SIP,ADI,NEWW,NEWQ,RCAL                            T 1100-
C                                                                       DAT 120
      DIMENSION PHI(IZ,JZ), STRT(IZ,JZ), SURI(IZ,JZ), T(IZ,JZ), TR(IZ,JZDAT 130
     1), TC(IZ,JZ), S(IZ,JZ), QRE(IZ,JZ), WELL(IZ,JZ), TL(IZ,JZ), SL(IZ,DAT 140
     2JZ), PERM(IP,JP), BOTTOM(IP,JP), SY(IP,JP), RATE(IR,JR), RIVER(IR,DAT 150
     3JR), M(IR,JR), TOP(IC,JC), GRND(IL,JL), DELX(JZ), DELY(IZ), WR(IH)DAT 160
     4, NWR(IH,2), PICK(IR,JR), PICKUP(IR,JR), RSEG(IR,JR), FLOWIN(IK), T 1700-
     5WIDE(IK), SLOPE(IK), IFLOUT(IK), ROUGH(IK), A(IZ,JZ), IN(9), IFMT(T 1701-
     69)                                                                T 1702-
C                                                                       DAT 180
      COMMON /SARRAY/ VF4(11),CHK(17)                                   T 1900-
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LEDAT 200
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,DAT 210
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,DAT 220
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DIDAT 230
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL                       T 2400-
      COMMON /CK/ ETFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT,IT 2500-
     1RCHK,RAIN,PANEVA,IRCHK1                                           T 2501-
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(5),XN1,MESUR,PRNT(122),BLANKDAT 260
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),DAT 270
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2                          DAT 280
      COMMON /ARSIZE/ IZ,JZ,IP,JP,IR,JR,IC,JC,IL,JL,IS,JS,IH,IMAX,IMX1,IT 2900-
     1K,IA,JA                                                           T 2901-
      COMMON /MPARAM/ FACTQ,NEWW,NEWQ,ERLEAK,DAYS                       T 2902-
      RETURN                                                            DAT 300
C     ..................................................................DAT 310
C     ********************                                              DAT 320
      ENTRY DATAIN                                                      DAT 330
C     ********************                                              DAT 340
C                                                                       DAT 350
C     ---READ AND WRITE SCALAR PARAMETERS---                            DAT 360
      READ (R,500) CONTR,XSCALE,YSCALE,DINCH,FACT1,FACT2,MESUR          DAT 370
      IF (CONTR.EQ.CHK(3)) WRITE (P,610) XSCALE,YSCALE,MESUR,MESUR,DINCHDAT 380
     1,FACT1,FACT2                                                      DAT 390
      READ (R,490) NPER,KTH,ERR,EROR,ERLEAK,SS,LENGTH,HMAX,FACTX,FACTY  T 4000-
      WRITE (P,520) NPER,KTH,ERR,EROR,ERLEAK,SS,FACTX,FACTY             T 4300-
C                                                                       DAT 440
C     ---READ CUMULATIVE MASS BALANCE PARAMETERS---                     DAT 450
      READ (R,600) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETFLDAT 460
     1XT,FLXNT                                                          DAT 470
      IF (IDK1.EQ.CHK(14)) GO TO 20                                     DAT 480
      IF (SUM.EQ.0.0) GO TO 40                                          DAT 490
      WRITE (P,480) SUM                                                 DAT 500
C     ..................................................................DAT 510
C                                                                       DAT 520
C     ---HEAD DATA TO CONTINUE PREVIOUS COMPUTATIONS READ HERE---       DAT 530
C     ------FROM CARDS:                                                 DAT 540
      DO 10 I=1,DIML                                                    DAT 550
```

```
      READ (R,540) (PHI(I,J),J=1,DIMW)
   10 WRITE (P,530) I,(PHI(I,J),J=1,DIMW)
      GO TO 40
C     ------READ AND WRITE DATA FROM UNIT 4 ON DISK RATHER THAN CARDS:
   20 READ (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETFL
     1XT,FLXNT
      WRITE (P,480) SUM
      DO 30 I=1,DIML
   30 WRITE (P,530) I,(PHI(I,J),J=1,DIMW)
      REWIND 4
C     ................................ STRT (STARTING HEAD) ............
   40 READ (R,490) FACT,IVAR,IPRN,IRECS,IRECD
      IF (IRECS.EQ.1) READ (2'1) STRT
      IF ((IVAR.EQ.1.OR.IRECS.EQ.1).AND.IPRN.NE.1) WRITE (P,470)
      DO 80 I=1,DIML
      IF (IVAR.EQ.1) READ (R,540) (STRT(I,J),J=1,DIMW)
      DO 70 J=1,DIMW
      IF (IRECS.EQ.1) GO TO 60
      IF (IVAR.NE.1) GO TO 50
      STRT(I,J)=STRT(I,J)*FACT
      GO TO 60
   50 STRT(I,J)=FACT
   60 SURI(I,J)=STRT(I,J)
      T(I,J)=0.
      TL(I,J)=0.
      SL(I,J)=0.
      TR(I,J)=0.
      TC(I,J)=0.
      WELL(I,J)=0.0
      PICK(I,J)=0.0
      PICKUP(I,J)=0.0
      QRE(I,J)=0.0
   70 IF (SUM.EQ.0.0.AND.IDK1.NE.CHK(14)) PHI(I,J)=STRT(I,J)
      IF (IVAR.EQ.0.AND.IRECS.EQ.0.OR.IPRN.EQ.1) GO TO 80
      WRITE (P,530) I,(STRT(I,J),J=1,DIMW)
   80 CONTINUE
      IF (IVAR.NE.1.AND.IRECS.NE.1) WRITE (P,420) FACT
      IF (IRECD.EQ.1) WRITE (2'1) STRT
      RETURN
C
C     ---READ REMAINING ARRAYS FROM CARDS OR DISK (AS SPECIFIED IN THE
C        OPTIONS) AND WRITE THEM ON DISK IF SPECIFIED IN THE OPTIONS---
C     **************
      ENTRY ARRAY(A,IFMT,IN,IRN)
C     **************
      READ (R,490) FACT,IVAR,IPRN,IRECS,IRECD
      IB=4*IRECS+2*IVAR+IPRN+1
      GO TO (90,90,110,110,140,140), IB
   90 DO 100 I=1,DIML
      DO 100 J=1,DIMW
  100 A(I,J)=FACT
      WRITE (P,430) IN,FACT
      GO TO 160
  110 IF (IB.EQ.3) WRITE (P,440) IN
      DO 130 I=1,DIML
      READ (R,510) (A(I,J),J=1,DIMW)
      DO 120 J=1,DIMW
  120 A(I,J)=A(I,J)*FACT
  130 IF (IB.EQ.3) WRITE (P,IFMT) I,(A(I,J),J=1,DIMW)
      GO TO 160
```

```
  140 READ (2'IRN) A
      IF (IB.EQ.6) GO TO 160
      WRITE (P,440) IN
      DO 150 I=1,DIML
  150 WRITE (P,IFMT) I,(A(I,J),J=1,DIMW)
  160 IF (IRECD.EQ.1) WRITE (2'IRN) A
      RETURN
C
C     ---INSERT ZERO VALUES IN THE T OR PERM MATRIX AROUND THE
C        BORDER OF THE MODEL---
C     **************
      ENTRY MDAT
C     **************
      DO 180 I=1,DIML
      DO 180 J=1,DIMW
      IF (WATER.EQ.CHK(2)) GO TO 170
      IF (I.EQ.1.OR.I.EQ.DIML.OR.J.EQ.1.OR.J.EQ.DIMW) T(I,J)=0.
      GO TO 180
  170 IF (I.EQ.1.OR.I.EQ.DIML.OR.J.EQ.1.OR.J.EQ.DIMW) PERM(I,J)=0.
  180 CONTINUE
C     ................................. DELX,DELY .......................
      READ (R,490) FACT,IVAR,IPRN,IRECS,IRECD
      IF (IRECS.EQ.1) GO TO 210
      IF (IVAR.EQ.1) READ (R,490) DELX
      DO 200 J=1,DIMW
      IF (IVAR.NE.1) GO TO 190
      DELX(J)=DELX(J)*FACT
      GO TO 200
  190 DELX(J)=FACT
  200 CONTINUE
      GO TO 220
  210 READ (2'13) DELX
  220 IF (IRECD.EQ.1) WRITE (2'13) DELX
      IF (IVAR.EQ.1.OR.IRECS.EQ.1.AND.IPRN.NE.1) WRITE (P,550) DELX
      IF (IVAR.NE.1.AND.IRECS.NE.1) WRITE (P,450) FACT
      READ (R,490) FACT,IVAR,IPRN,IRECS,IRECD
      IF (IRECS.EQ.1) GO TO 250
      IF (IVAR.EQ.1) READ (R,490) DELY
      DO 240 I=1,DIML
      IF (IVAR.NE.1) GO TO 230
      DELY(I)=DELY(I)*FACT
      GO TO 240
  230 DELY(I)=FACT
  240 CONTINUE
      GO TO 260
  250 READ (2'14) DELY
  260 IF (IRECD.EQ.1) WRITE (2'14) DELY
      IF (IVAR.EQ.1.OR.IRECS.EQ.1.AND.IPRN.NE.1) WRITE (P,560) DELY
      IF (IVAR.NE.1.AND.IRECS.NE.1) WRITE (P,460) FACT
C
C     ---INITIALIZE VARIABLES---
      JNO1=DIMW-1
      INO1=DIML-1
      IF (LEAK.NE.CHK(9).OR.SS.NE.0.) GO TO 280
      DO 270 I=2,INO1
      DO 270 J=2,JNO1
      IF (M(I,J).EQ.0.) GO TO 270
      TL(I,J)=RATE(I,J)/M(I,J)
  270 CONTINUE
  280 ETQB=0.0
```

```
      ETQD=0.0                                                          DAT1730
      SUBS=0.0                                                          DAT1740
      U=1.0                                                             DAT1750
      TT=0.0                                                            DAT1760
      IM=MINO(6*DIMW+4,124)                                             DAT1770
      IM=(132-IM)/2                                                     DAT1780
      VF4(3)=DIGIT(IM)                                                  DAT1790
      VF4(8)=DIGIT(IM+5)                                                DAT1800
      WIDTH=0.                                                          DAT1810
      DO 290 J=2,JNO1                                                   DAT1820
  290 WIDTH=WIDTH+DELX(J)                                               DAT1830
      YDIM=0.                                                           DAT1840
      DO 300 I=2,INO1                                                   DAT1850
  300 YDIM=YDIM+DELY(I)                                                 DAT1860
      RETURN                                                            DAT1870
C     ..................................................................DAT1880
C                                                                       DAT1890
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-DAT1900
C     ************************                                          DAT1910
      ENTRY NEWPER                                                      DAT1920
C     ************************                                          DAT1930
C                                                                       DAT1940
      READ (R,490) KP,KPM1,KPER,KNUM,TMAX,NUMT,CDLT,DELT                T19500-
      READ (R,490) TPUMP,NWEL,QET,ETDIST,FACTQ                          T19501-
      IF (ETDIST.LE.0.) ETDIST=1.                                       T19502-
      READ (R,630) NEWW,NEWQ                                            T19503-
C                                                                       DAT1960
C     ---COMPUTE ACTUAL DELT AND NUMT---                                DAT1970
      DT=DELT/24.                                                       DAT1980
      TM=0.0                                                            DAT1990
      DO 310 I=1,NUMT                                                   DAT2000
      DT=CDLT*DT                                                        DAT2010
      TM=TM+DT                                                          DAT2020
      IF (TM.GE.TMAX) GO TO 320                                         DAT2030
  310 CONTINUE                                                          DAT2040
      GO TO 330                                                         DAT2050
  320 DELT=TMAX/TM*DELT                                                 DAT2060
      NUMT=I                                                            DAT2070
  330 WRITE (P,620) KP,TPUMP,KPER                                       T20701-
      WRITE (P,570) KNUM,TMAX,NUMT,DELT,CDLT,QET,ETDIST,FACTQ           T20800-
      DELT=DELT*3600.                                                   DAT2090
      TMAX=TMAX*86400.                                                  DAT2100
      WRITE (P,640) NEWW,NEWQ                                           T21001-
C                                                                       DAT2110
C     ---INITIALIZE SUMP, STRT, SL, WELL AND WR---                      DAT2120
      WRITE (P,580) NWEL                                                DAT2130
      IF (KP.GT.KPM1) SUMP=0.                                           DAT2140
      DO 350 I=1,DIML                                                   DAT2150
      DO 350 J=1,DIMW                                                   DAT2160
      IF (KP.EQ.KPM1) GO TO 340                                         DAT2170
      STRT(I,J)=PHI(I,J)                                                DAT2180
  340 IF (LEAK.NE.CHK(9)) GO TO 350                                     DAT2190
      IF (M(I,J).EQ.0.) GO TO 350                                       DAT2200
      SL(I,J)=RATE(I,J)/M(I,J)*(RIVER(I,J)-STRT(I,J))                   DAT2210
  350 IF (NEWW.EQ.CHK(16)) WELL(I,J)=0.                                 T22200-
      IF (NW.EQ.0) GO TO 370                                            DAT2230
      DO 360 I=1,NW                                                     DAT2240
  360 WR(I)=0.                                                          DAT2250
  370 IF (NWEL.EQ.0) GO TO 410                                          DAT2260
      IF (NEWW.EQ.CHK(16)) GO TO 374                                    T22601-
```

```
      DO 372 I=1,DIML                                                   T22602-
      DO 372 J=1,DIMW                                                   T22603-
      RWELL=WELL(I,J)*DELX(J)*DELY(I)                                   T22604-
  372 IF (WELL(I,J).NE.0.) WRITE (P,590) I,J,RWELL                      T22605-
      GO TO 410                                                         T22606-
C                                                                       DAT2270
C     ---READ AND WRITE WELL PUMPING RATES AND WELL RADII---            DAT2280
  374 KW=0                                                              DAT2290
      DO 400 II=1,NWEL                                                  DAT2300
      READ (R,490) I,J,WELL(I,J),RADIUS                                 DAT2310
      IF (RADIUS.EQ.0.) GO TO 380                                       DAT2320
      KW=KW+1                                                           DAT2330
      IF (KW.GT.NW) GO TO 380                                           DAT2340
      NWR(KW,1)=I                                                       DAT2350
      NWR(KW,2)=J                                                       DAT2360
      WR(KW)=RADIUS                                                     DAT2370
      WRITE (P,590) I,J,WELL(I,J),WR(KW)                                DAT2380
      GO TO 390                                                         DAT2390
  380 WRITE (P,590) I,J,WELL(I,J)                                       DAT2400
  390 WELL(I,J)=WELL(I,J)/(DELX(J)*DELY(I))                             DAT2410
  400 CONTINUE                                                          DAT2420
  410 CONTINUE                                                          T24201-
C     --READ AND WRITE DATA IF RIVER HEADS TO BE COMPUTED---            T24202-
      IF (RCAL.NE.CHK(17)) GO TO 414                                    T24203-
      WRITE (P,650)                                                     T24204-
      DO 412 K=1,NSEG                                                   T24205-
      READ (R,490) KK,WIDE(KK),SLOPE(KK),ROUGH(KK),FLOWIN(KK),IFLOUT(KK)T24206-
  412 WRITE (P,660) KK,WIDE(KK),SLOPE(KK),ROUGH(KK),FLOWIN(KK),IFLOUT(KKT24207-
     1)                                                                 T24208-
      READ (R,490) RAIN,PANEVA                                          T24209-
  414 IF (KNUM.EQ.KPER) KP=KP+1                                         T24210-
      RETURN                                                            T24300-
C     ..................................................................DAT2440
C                                                                       DAT2450
C     FORMATS:                                                          DAT2460
C                                                                       DAT2470
C     ------------------------------------------------------------------DAT2480
C                                                                       DAT2490
C                                                                       DAT2500
  420 FORMAT ('0',63X,'STARTING HEAD =',G15.7)                          DAT2510
  430 FORMAT ('0',41X,9A4,'=',G15.7)                                    DAT2520
  440 FORMAT ('1',49X,9A4,/,65X,'MATRIX',/,50X,36('-'))                 DAT2530
  450 FORMAT ('0',72X,'DELX =',G15.7)                                   DAT2540
  460 FORMAT ('0',72X,'DELY =',G15.7)                                   DAT2550
  470 FORMAT ('1',60X,'STARTING HEAD MATRIX'/61X,20('-'))               DAT2560
  480 FORMAT ('1',40X,' CONTINUATION - HEAD AFTER ',G20.7,' SEC PUMPING DAT2570
     1'/42X,58('-'))                                                    DAT2580
  490 FORMAT (8G10.0)                                                   DAT2590
  500 FORMAT (A4,6X,5G10.0,A8)                                          DAT2600
  510 FORMAT (20F4.0)                                                   DAT2610
  520 FORMAT ('0',51X,'NUMBER OF PUMPING PERIODS =',I5/49X,'TIME STEPS BDAT2620
     1ETWEEN PRINTOUTS =',I5//51X,'ERROR CRITERION FOR CLOSURE =',G15.7/T26300-
     241X,'          STEADY STATE ERROR CRITERION =',G15.7/,26X,'ERROR CRT26400-
     3ITERION TO SATISFY RIVER-AQUIFER INTERACTION =',G15.7//44X,'SPECIFT26500-
     4IC STORAGE OF CONFINING BED  =',G15.7//22X,'MULTIPLICATION FACTOR T26600-
     5FOR TRANSMISSIVITY IN X DIRECTION =',G15.7/63X,'IN Y DIRECTION =',T26700-
     6G15.7)                                                            T26800-
  530 FORMAT ('0',I2,2X,20F6.1/(5X,20F6.1))                             DAT2690
  540 FORMAT (8F10.4)                                                   DAT2700
  550 FORMAT (1H1,46X,40HGRID SPACING IN PROTOTYPE IN X DIRECTION/47X,40DAT2710
```

```
     1('-')//('0',12F10.0))                                             DAT2720
 560 FORMAT (1H-,46X,40HGRID SPACING IN PROTOTYPE IN Y DIRECTION/47X,40DAT2730
     1('-')//('0',12F10.0))                                             DAT2740
 570 FORMAT ('-',50X,'   TIME PERIOD NO. ',I4,': ',F10.2,' DAYS'/51X,38('T27500-
     1-')//53X,'NUMBER OF TIME STEPS=',I6//59X,'DELT IN HOURS =',F10.3//DAT2760
     253X,'MULTIPLIER FOR DELT =',F10.3,//,49X,'EVAPOTRANSPIRATION RATE T27700-
     3=',G15.7,/,51X,'EFFECTIVE DEPTH OF ET =',G15.7,/,31X,'MULTIPLICATIT27701-
     4ON FACTOR FOR RECHARGE MATRIX =',G15.7)                          - T27702-
 580 FORMAT ('-',63X,I4,' WELLS'/65X,9('-')//50X,'I',9X,'J    PUMPING RDAT2780
     1ATE  WELL RADIUS'/)                                               DAT2790
 590 FORMAT (41X,2I10,2F13.3)                                           DAT2800
 600 FORMAT (4G20.10)                                                   DAT2810
 610 FORMAT ('0';30X,'ON ALPHAMERIC MAP:'/40X,'MULTIPLICATION FACTOR FODAT2820
     1R X DIMENSION =',G15.7/40X,'MULTIPLICATION FACTOR FOR Y DIMENSION DAT2830
     2=',G15.7/55X,'MAP SCALE IN UNITS OF  ',A11/50X,'NUMBER OF ',A8,' PDAT2840
     3ER INCH =',G15.7/43X,'MULTIPLICATION FACTOR FOR DRAWDOWN =',G15.7/DAT2850
     447X,'MULTIPLICATION FACTOR FOR HEAD =',G15.7)                     DAT2860
 620 FORMAT (1H1,///,24(' ->'),'PUMPING PERIOD NO. ',I4,': ',F10.2,' DAYST28601-
     1',/,72X,38('-'),//,81X,'NUMBER OF TIME PERIODS =',I5,////////)    T28602-
 630 FORMAT (2(A4,1X))                                                  T28603-
 640 FORMAT ('-',/,48X,'FOR THIS TIME PERIOD -> NEW PUMPING RATES USED T28604-
     1?............',A4,/,69X,'-> NEW AREAL RECHARGE RATE USED ?......',T28605-
     2A4,//)                                                            T28606-
 650 FORMAT ('-',///,10X,'RIVER SEGMENT',5X,'AVERAGE WIDTH',5X,'AVERAGET28607-
     1 SLOPE',5X,'MANNING COEF.',5X,'FLOWIN (L**3/T)',5X,'DISCHARGE TO ST28608-
     2EGMENT',/,10X,4(13('-'),5X),15('-'),5X,20('-'))                   T28609-
 660 FORMAT ('-',14X,I3,13X,F7.4,3(11X,F7.4),21X,I3)                    T28610-
      END                                                               DAT2870-
```

```
      SUBROUTINE STEP(PHI,KEEP,STRT,SURI,T,WELL,PERM,BOTTOM,TOP,DELX,DDN
     1,DELY,WR,NWR,TEST3,QRE,PICK,PREPIC)
C     ------------------------------------------------------------------
C     INITIALIZE DATA FOR TIME STEP, CHECK FOR STEADY STATE,
C     PRINT AND  PUNCH RESULTS
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,DBLE,DABS,TEST2,DMAX1,XLABEL,YLABEL,XN1,MESUR,TITLE
      REAL *4MINS,M,KEEP
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,NEWW,NEWQ,RCAL
C
      DIMENSION PHI(IZ,JZ), KEEP(IZ,JZ), STRT(IZ,JZ), SURI(IZ,JZ), T(IZ,
     1JZ), BOTTOM(IP,JP), WELL(IZ,JZ), PERM(IP,JP), TOP(IC,JC), DELX(JZ)
     2, DDN(JZ), DELY(IZ), WR(IH), NWR(IH,2), ITTO(200), TEST3(IMX1), QR
     3E(IZ,JZ), PICK(IR,JR), PREPIC(IA,JA)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /MPARAM/ FACTQ,NEWW,NEWQ,ERLEAK,DAYS
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
      COMMON /CK/ ETFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT,I
     1RCHK,RAIN,PANEVA,IRCHK1
      COMMON /ARSIZE/ IZ,JZ,IP,JP,IR,JR,IC,JC,IL,JL,IS,JS,IH,IMAX,IMX1,I
     1K,IA,JA
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(5),XN1,MESUR,PRNT(122),BLANK
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
C
      DATA PIE/3.141593/,YYY/Z00000000/
      RETURN
C     ..................................................................
C
C     ---START A NEW TIME STEP---
C     ********************
      ENTRY NEWSTP
C     ********************
      KT=KT+1
      KOUNT=0
      DO 10 I=1,DIML
      DO 10 J=1,DIMW
      QRE(I,J)=QRE(I,J)*FACTQ
   10 KEEP(I,J)=PHI(I,J)
      DELT=CDLT*DELT
      SUM=SUM+DELT
      SUMP=SUMP+DELT
      DAYSP=SUMP/86400.
      YRSP=DAYSP/365.
      HRS=SUM/3600.
      MINS=HRS*60.
      DAYS=HRS/24.
      YRS=DAYS/365.
      RETURN
C     ..................................................................
C
```

```
C     ---CHECK FOR STEADY STATE---
C     ********************
      ENTRY STEADY
C     ********************
      TEST2=0.
      DO 20 I=2,INO1
      DO 20 J=2,JNO1
   20 TEST2=DMAX1(TEST2,DABS(DBLE(KEEP(I,J))-PHI(I,J)))
      IF (TEST2.GE.EROR) GO TO 30
      WRITE (P,330) KT
      IFINAL=1
      GO TO 40
   30 IF (KT.EQ.NUMT) IFINAL=1
C
C     ---ENTRY FOR TERMINATING COMPUTATIONS IF MAXIMUM ITERATIONS
C     EXCEEDED---
C     ********************
      ENTRY TERM1
C     ********************
   40 IF (KT.GT.200) WRITE (P,400)
      ITTO(KT)=KOUNT
      IF (KOUNT.LE.ITMAX) GO TO 80
      IERR=2
      KOUNT=KOUNT-1
      ITTO(KT)=KOUNT
      IF (KT.EQ.1) GO TO 60
C
C     ---WRITE ON DISK OR PUNCH CARDS AS SPECIFIED IN THE OPTIONS---
      XXX=SUM-DELT
      IF (IDK2.EQ.CHK(15)) WRITE (4) ((KEEP(I,J),YYY,I=1,DIML),J=1,DIMW)
     1,XXX,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETFLXT,FLXNT
      IF (PNCH.NE.CHK(1)) GO TO 80
      WRITE (PU,360) XXX,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ET
     1FLXT,FLXNT
      DO 50 I=1,DIML
   50 WRITE (PU,350) (KEEP(I,J),J=1,DIMW)
      GO TO 80
   60 IF (IDK2.EQ.CHK(15)) WRITE (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST
     1,CHDT,FLUXT,STORT,ETFLXT,FLXNT
      IF (PNCH.NE.CHK(1)) GO TO 80
      WRITE (PU,360) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ET
     1FLXT,FLXNT
      DO 70 I=1,DIML
   70 WRITE (PU,350) (PHI(I,J),J=1,DIMW)
C
   80 IF (CHCK.EQ.CHK(5)) CALL CHECK
      IF (IERR.EQ.2) GO TO 90
C
C     ---PRINT OUTPUT AT DESIGNATED TIME STEPS---
      IF (MOD(KT,KTH).NE.0.AND.IFINAL.NE.1) RETURN
   90 WRITE (P,340) KT,DELT,SUM,MINS,HRS,DAYS,YRS,DAYSP,YRSP
      IF (CHCK.EQ.CHK(5)) CALL CWRITE
      IF (TT.NE.0.) WRITE (P,320) TMIN,TT
      KOUNT=KOUNT+1
      WRITE (P,300) (TEST3(J),J=1,KOUNT)
      WRITE (P,290) TEST2
      I3=1
      I5=0
  100 I5=I5+40
      I4=MIN0(KT,I5)
```

```
      WRITE (P,390) (I,I=I3,I4)
      WRITE (P,380)
      WRITE (P,370) (ITTO(I),I=I3,I4)
      WRITE (P,380)
      IF (KT.LE.I5) GO TO 110
      I3=I3+40
      GO TO 100
C
C     ---PRINT ALPHAMERIC MAPS---
  110 IF (CONTR.NE.CHK(3)) GO TO 120
      IF (FACT1.NE.0.) CALL PRNTA(1)
      IF (FACT2.NE.0.) CALL PRNTA(2)
  120 IF (HEAD.NE.CHK(8)) GO TO 140
C
C     ---PRINT HEAD MATRIX---
      WRITE (P,310)
      DO 130 I=1,DIML
  130 WRITE (P,VF4) I,(PHI(I,J),J=1,DIMW)
  140 IF (NUM.NE.CHK(4)) GO TO 170
C
C     ---PRINT DRAWDOWN---
      WRITE (P,280)
C     ********************
      ENTRY DRDN
C     ********************
      DO 160 I=1,DIML
      DO 150 J=1,DIMW
  150 DDN(J)=SURI(I,J)-PHI(I,J)
  160 WRITE (P,VF4) I,(DDN(J),J=1,DIMW)
  170 IF (NW.EQ.0.OR.IERR.EQ.1) GO TO 230
C     ..................................................................
C
C     ---COMPUTE APPROXIMATE HEAD FOR PUMPING WELLS---
      WRITE (P,260)
      DO 220 KW=1,NW
      IF (WR(KW).EQ.0.) GO TO 220
      I=NWR(KW,1)
      J=NWR(KW,2)
C
C     COMPUTE EFFECTIVE RADIUS OF WELL IN MODEL---
      RE=(DELX(J)+DELY(I))/9.62
      IF (WATER.NE.CHK(2)) GO TO 180
      IF (CONVRT.NE.CHK(7)) GO TO 190
      IF (PHI(I,J).LT.TOP(I,J)) GO TO 190
C
C     ---COMPUTATION FOR WELL IN ARTESIAN AQUIFER---
  180 HW=PHI(I,J)+WELL(I,J)*ALOG(RE/WR(KW))/(2.*PIE*T(I,J))*DELX(J)*DELY
     1(I)
      GO TO 210
C
C     ---COMPUTATION FOR WELL IN WATER TABLE AQUIFER
  190 HED=PHI(I,J)-BOTTOM(I,J)
      ARG=HED*HED+WELL(I,J)*ALOG(RE/WR(KW))/(PIE*PERM(I,J))*DELX(J)*DELY
     1(I)
      IF (ARG.GT.0.) GO TO 200
      WRITE (P,270) I,J
      GO TO 220
  200 HW=SQRT(ARG)+BOTTOM(I,J)
C
C     ---COMPUTE DRAWDOWN AT THE WELL AND PRINT RESULTS---
```

```
  210 DRAW=SURI(I,J)-HW
      WRITE (P,250) I,J,WR(KW),HW,DRAW
  220 CONTINUE
  230 IF (IERR.NE.2) RETURN
      STOP
C
C     ---DISK OUTPUT---
C     ****************
      ENTRY DISK
C     ****************
      WRITE (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETF
     1LXT,FLXNT
      RETURN
C     ...........................................................
C
C     ---PUNCHED OUTPUT---
C     ********************
      ENTRY PUNCH
C     ********************
      WRITE (PU,360) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ET
     1FLXT,FLXNT
      DO 240 I=1,DIML
  240 WRITE (PU,350) (PHI(I,J),J=1,DIMW)
      RETURN
C
C     ...........................................................
C
C     FORMATS:
C
C
C     -----------------------------------------------------------------
C
C
  250 FORMAT (' ',43X,2I5,3F11.2)
  260 FORMAT ('-',50X,'HEAD AND DRAWDOWN IN PUMPING WELLS'/51X,34('-')//
     148X,'I     J   WELL RADIUS     HEAD     DRAWDOWN'//)
  270 FORMAT (' ',43X,2I5,'  WELL IS DRY')
  280 FORMAT (1H1,60X,'DRAWDOWN'/61X,8('-'))
  290 FORMAT ('0MAXIMUM CHANGE IN HEAD FOR THIS TIME STEP =',F10.3/' ',5
     13('-'))
  300 FORMAT ('0MAXIMUM HEAD CHANGE FOR EACH ITERATION:'/' ',39('-')/('0
     1',10F12.4))
  310 FORMAT ('1',60X,'HEAD MATRIX'/61X,11('-'))
  320 FORMAT ('0DIMENSIONLESS TIME FOR THIS STEP RANGES FROM',G15.7,'  T
     1O',G15.7)
  330 FORMAT ('-*****STEADY STATE AT TIME STEP',I4,'*****')
  340 FORMAT (1H1,44X,57('-')/45X,'!',14X,'TIME STEP NUMBER =',I9,14X,'!
     1'/45X,57('-')//50X,29HSIZE OF TIME STEP IN SECONDS=,F14.2//55X,'TO
     2TAL SIMULATION TIME IN SECONDS=',F14.2/80X,8HMINUTES=,F14.2/82X,6H
     3HOURS=,F14.2/83X,5HDAYS=,F14.2/82X,'YEARS=',F14.2///45X,'DURATION 
     4OF CURRENT PUMPING PERIOD IN DAYS=',F14.2/82X,'YEARS=',F14.2//)
  350 FORMAT (8F10.4)
  360 FORMAT (4G20.10)
  370 FORMAT ('0ITERATIONS:',40I3)
  380 FORMAT (' ',10('-'))
  390 FORMAT ('0TIME STEP :',40I3)
  400 FORMAT ('0',10('*'),'THE NUMBER OF TIME STEPS EXCEEDS THE DIMENSIO
     1N OF THE VECTOR ITTO AND MAY CAUSE UNEXPECTED RESULTS IN ADDITIONA
     2L''0COMPUTATION.  AVOID PROBLEMS BY INCREASING THE DIMENSION OF T
     3HE VECTOR ITTO IN STEP',10('*'))
      END
```

```
      SUBROUTINE SOLVE1(PHI, BE, G, TEMP, KEEP, PHE, STRT, T, S, QRE, WELL, TL, SL, D
     1EL, ETA, V, XI, DELX, BET, DELY, ALF, TEST3, TR, TC, GRND, SY, TOP, RATE, M, RIVER
     2)
C     ---------------------------------------------------------------------
C       SOLUTION BY THE STRONGLY IMPLICIT PROCEDURE
C     ---------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI, DBLE, RHOP(20), G, BE, TEMP, DABS, W, TEST2, DMAX1, RHO, B, D, F, H,
     1B1, E, CH, GH, BH, DH, EH, FH, HH, ALFA, BETA, GAMA, RES
      REAL *4KEEP, M
      INTEGER R, P, PU, DIML, DIMW, CHK, WATER, CONVRT, EVAP, CHCK, PNCH, NUM, HEAD,
     1CONTR, LEAK, RECH, SIP, IORDER(21), ADI, RCAL
C
      DIMENSION PHI(1), BE(1), G(1), TEMP(1), KEEP(1), PHE(1), STRT(1),
     1T(1), S(1), QRE(1), WELL(1), TL(1), SL(1), DEL(1), ETA(1), V(1), X
     2I(1), DELX(1), BET(1), DELY(1), ALF(1), TEST3(1), TR(1), TC(1), GR
     3ND(1), SY(1), TOP(1), RATE(1), M(1), RIVER(1)
C
      COMMON /SARRAY/ VF4(11), CHK(17)
      COMMON /SPARAM/ WATER, CONVRT, EVAP, CHCK, PNCH, NUM, HEAD, CONTR, EROR, LE
     1AK, RECH, SIP, U, SS, TT, TMIN, ETDIST, QET, ERR, TMAX, CDLT, HMAX, YDIM, WIDTH,
     2NUMS, LSOR, ADI, DELT, SUM, SUMP, SUBS, STORE, TEST, ETQB, ETQD, FACTX, FACTY,
     3IERR, KOUNT, IFINAL, NUMT, KT, KP, NPER, KTH, ITMAX, LENGTH, NWEL, NW, DIML, DI
     4MW, JNO1, INO1, R, P, PU, I, J, IDK1, IDK2, NSEG, RCAL
      RETURN
C     ......................................................................
C
C     ---COMPUTE AND PRINT ITERATION PARAMETERS---
C     ********************
      ENTRY ITER1
C     ********************
C     ---INITIALIZE ORDER OF ITERATION PARAMETERS (OR REPLACE WITH A
C     READ STATEMENT)---
      DATA IORDER/1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 11*1/
      I2=INO1-1
      J2=JNO1-1
      L2=LENGTH/2
      PL2=L2-1.
      W=0.
      PI=0.
C
C     ---COMPUTE AVERAGE MAXIMUM PARAMETER FOR PROBLEM---
      DO 10 I=2, INO1
      DO 10 J=2, JNO1
      N=I+DIML*(J-1)
      IF (T(N).EQ.0.) GO TO 10
      PI=PI+1.
      DX=DELX(J)/WIDTH
      DY=DELY(I)/YDIM
      W=W+1.-AMIN1(2.*DX*DX/(1.+FACTY*DX*DX/(FACTX*DY*DY)), 2.*DY*DY/(1.+
     1FACTX*DY*DY/(FACTY*DX*DX)))
   10 CONTINUE
      W=W/PI
C
C     ---COMPUTE PARAMETERS IN GEOMETRIC SEQUENCE---
      PJ=-1.
      DO 20 I=1, L2
      PJ=PJ+1.
```

```
   20 TEMP(I)=1.-(1.-W)**(PJ/PL2)
C
C     ---ORDER SEQUENCE OF PARAMETERS---
      DO 30 J=1,LENGTH
   30 RHOP(J)=TEMP(IORDER(J))
      WRITE (P,370) HMAX
      WRITE (P,380) LENGTH,(RHOP(J),J=1,LENGTH)
      RETURN
C     .....................................................................
C
C     ---INITIALIZE DATA FOR A NEW ITERATION---
   40 KOUNT=KOUNT+1
      IF (KOUNT.LE.ITMAX) GO TO 50
      WRITE (P,360)
      CALL TERM1
   50 IF (MOD(KOUNT,LENGTH)) 60,60,70
C     ************************
      ENTRY NEWITA
C     ************************
   60 NTH=0
   70 NTH=NTH+1
      W=RHOP(NTH)
      TEST3(KOUNT+1)=0.
      TEST=0.
      N=DIML*DIMW
      DO 80 I=1,N
      PHE(I)=PHI(I)
      DEL(I)=0.
      ETA(I)=0.
      V(I)=0.
   80 XI(I)=0.
      BIGI=0.0
C
C     ---COMPUTE TRANSMISSIVITY AND T COEFFICIENTS IN WATER TABLE
C     OR WATER TABLE-ARTESIAN SIMUATION---
      IF (WATER.NE.CHK(2)) GO TO 90
      CALL TRANS
C
C     ---CHOOSE SIP NORMAL OR REVERSE ALGORITHM---
   90 IF (MOD(KOUNT,2)) 100,230,100
C     .....................................................................
C     ---ORDER EQUATIONS WITH ROW 1 FIRST - 3X3 EXAMPLE:
C                 1 2 3
C                 4 5 6
C                 7 8 9
C     .....................................................................
  100 DO 210 I=2,INO1
      DO 210 J=2,JNO1
      N=I+DIML*(J-1)
      NL=N-DIML
      NR=N+DIML
      NA=N-1
      NB=N+1
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 210
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(NL)/DELX(J)
      F=TR(N)/DELX(J)
```

```
      B=TC(NA)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 120
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 120
      IF (PHE(N).GT.GRND(N)) GO TO 110
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 120
  110 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  120 IF (CONVRT.EQ.CHK(7)) GO TO 130
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 200
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
  130 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 170
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 160
      IF (KEEP(N)-PHE(N)) 140,150,150
  140 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 170
  150 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  160 RHO=SY(N)/DELT
      GO TO 180
  170 RHO=S(N)/DELT
  180 IF (LEAK.NE.CHK(9)) GO TO 200
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 200
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 190
      HED2=TOP(N)
      U=0.
  190 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  200 CONTINUE
C
C     ---SIP 'NORMAL' ALGORITHM---
C     ---FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V---
      E=-B-D-F-H-RHO-TL(N)*U-ETQB
      CH=DEL(NA)*B/(1.+W*DEL(NA))
      GH=ETA(NL)*D/(1.+W*ETA(NL))
      BH=B-W*CH
      DH=D-W*GH
      EH=E+W*CH+W*GH
      FH=F-W*CH
      HH=H-W*GH
      ALFA=BH
      BETA=DH
      GAMA=EH-ALFA*ETA(NA)-BETA*DEL(NL)
      DEL(N)=FH/GAMA
      ETA(N)=HH/GAMA
      RES=-D*PHI(NL)-F*PHI(NR)-H*PHI(NB)-B*PHI(NA)-E*PHI(N)-RHO*KEEP(N)-
```

```
     1SL(N)-QRE(N)-WELL(N)+ETQD-SUBS-TL(N)*STRT(N)
      V(N)=(HMAX*RES-ALFA*V(NA)-BETA*V(NL))/GAMA
  210 CONTINUE
C
C     ---BACK SUBSTITUTE FOR VECTOR XI---
      DO 220 I=1,I2
      I3=DIML-I
      DO 220 J=1,J2
      J3=DIMW-J
      N=I3+DIML*(J3-1)
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 220
      XI(N)=V(N)-DEL(N)*XI(N+DIML)-ETA(N)*XI(N+1)
C
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERION---
      TCHK=ABS(XI(N))
      IF (TCHK.GT.BIGI) BIGI=TCHK
      PHI(N)=PHI(N)+XI(N)
  220 CONTINUE
      IF (BIGI.GT.ERR) TEST=1.
      TEST3(KOUNT+1)=BIGI
      IF (TEST.EQ.1.) GO TO 40
      RETURN
C
C     ..................................................................
C     ---ORDER EQUATIONS WITH THE LAST ROW FIRST - 3X3  EXAMPLE:
C                  7 8 9
C                  4 5 6
C                  1 2 3
C     ..................................................................
  230 DO 340 II=1,I2
      I=DIML-II
      DO 340 J=2,JNO1
      N=I+DIML*(J-1)
      NL=N-DIML
      NR=N+DIML
      NA=N-1
      NB=N+1
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 340
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(NL)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(NA)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 250
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 250
      IF (PHE(N).GT.GRND(N)) GO TO 240
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 250
  240 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  250 IF (CONVRT.EQ.CHK(7)) GO TO 260
```

```
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 330
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
  260 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 300
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 290
      IF (KEEP(N)-PHE(N)) 270,280,280
  270 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 300
  280 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  290 RHO=SY(N)/DELT
      GO TO 310
  300 RHO=S(N)/DELT
  310 IF (LEAK.NE.CHK(9)) GO TO 330
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 330
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 320
      HED2=TOP(N)
      U=0.
  320 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  330 CONTINUE
C
C     ---SIP 'REVERSE' ALGORITHM---
C     ---FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V---
      E=-B-D-F-H-RHO-TL(N)*U-ETQB
      CH=DEL(NB)*H/(1.+W*DEL(NB))
      GH=ETA(NL)*D/(1.+W*ETA(NL))
      BH=H-W*CH
      DH=D-W*GH
      EH=E+W*CH+W*GH
      FH=F-W*CH
      HH=B-W*GH
      ALFA=BH
      BETA=DH
      GAMA=EH-ALFA*ETA(NB)-BETA*DEL(NL)
      DEL(N)=FH/GAMA
      ETA(N)=HH/GAMA
      RES=-D*PHI(NL)-F*PHI(NR)-H*PHI(NB)-B*PHI(NA)-E*PHI(N)-RHO*KEEP(N)-
     1SL(N)-QRE(N)-WELL(N)+ETQD-SUBS-TL(N)*STRT(N)
      V(N)=(HMAX*RES-ALFA*V(NB)-BETA*V(NL))/GAMA
  340 CONTINUE
C
C     ---BACK SUBSTITUTE FOR VECTOR XI---
      DO 350 I3=2,IND1
      DO 350 J=1,J2
      J3=DIMW-J
      N=I3+DIML*(J3-1)
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 350
      XI(N)=V(N)-DEL(N)*XI(N+DIML)-ETA(N)*XI(N-1)
C
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERION---
      TCHK=ABS(XI(N))
      IF (TCHK.GT.BIGI) BIGI=TCHK
      PHI(N)=PHI(N)+XI(N)
```

```
  350 CONTINUE
      IF (BIGI.GT.ERR) TEST=1.
      TEST3(KOUNT+1)=BIGI
      IF (TEST.EQ.1.) GO TO 40
      RETURN
C
C     ..............................................................
C
C     ---FORMATS---
C
C     --------------------------------------------------------------
C
C
  360 FORMAT ('0EXCEEDED PERMITTED NUMBER OF ITERATIONS'//' ',39('*'))
  370 FORMAT ('-',44X,'SOLUTION BY THE STRONGLY IMPLICIT PROCEDURE'/45X,
     143('_'),//,61X,'BETA=',F5.2)
  380 FORMAT (1H0,I5,22H ITERATION PARAMETERS:,6D15.7/(/28X,6D15.7/))
      END
```

```
      SUBROUTINE SOLVE2(PHI,BE,G,TEMP,KEEP,PHE,STRT,T,S,QRE,WELL,TL,SL,D
     1EL,ETA,V,XI,DELX,BETA,DELY,ALFA,TEST3,TR,TC,GRND,SY,TOP,RATE,M,RIV
     2ER)
C     ------------------------------------------------------------------
C     SOLUTION BY LINE SUCCESSIVE OVERRELAXATION
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,DBLE,RHOP(20),G,BE,TEMP,IMK,DABS,W,PARAM,TEST2,DMAX1,B2
     1,A,C,B1,E,Q,RHO,B,D,F,H
      REAL *4KEEP,M
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,RCAL
C
      DIMENSION PHI(1), BE(1), G(1), TEMP(1), KEEP(1), PHE(1), STRT(1),
     1T(1), S(1), QRE(1), WELL(1), TL(1), SL(1), DEL(1), ETA(1), V(1), X
     2I(1), DELX(1), BETA(1), DELY(1), ALFA(1), TEST3(1), TR(1), TC(1),
     3GRND(1), SY(1), TOP(1), RATE(1), M(1), RIVER(1)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
      RETURN
C     ..................................................................
C
C     ---WRITE ACCELERATION PARAMETER---
C     ********************
      ENTRY ITER2
C     ********************
      WRITE (P,490)
      WRITE (P,500) HMAX,LENGTH
      RETURN
C     ..................................................................
C
C     ---INITIALIZE DATA FOR A NEW ITERATION---
   10 KOUNT=KOUNT+1
      IF (KOUNT.LE.ITMAX) GO TO 20
      WRITE (P,510)
      CALL TERM1
C     ************************
      ENTRY NEWITB
C     ************************
   20 TEST3(KOUNT+1)=0.
      TEST=0.
      N=DIML*DIMW
      DO 30 I=1,N
   30 PHE(I)=PHI(I)
      BIGI=0.0
C
C     ---COMPUTE TRANSMISSIVITY AND T COEFFICIENTS IN WATER TABLE
C     OR WATER TABLE-ARTESIAN SIMUATION---
      IF (WATER.NE.CHK(2)) GO TO 40
      CALL TRANS
C     ..................................................................
C
C     ---SOLUTION BY LSOR---
```

```
C     ---------------------
   40 NO3=DIMW-2
      TEMP(DIMW)=0.0
      DO 170 I=2,INO1
      DO 150 J=2,JNO1
      N=I+DIML*(J-1)
      NA=N-1
      NB=N+1
      NL=N-DIML
      NR=N+DIML
      BE(J)=0.0
      G(J)=0.0
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 150
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(N-DIML)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(N-1)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 60
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 60
      IF (PHE(N).GT.GRND(N)) GO TO 50
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 60
   50 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
   60 IF (CONVRT.EQ.CHK(7)) GO TO 70
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 140
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
   70 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 110
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 100
      IF (KEEP(N)-PHE(N)) 80,90,90
   80 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 110
   90 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  100 RHO=SY(N)/DELT
      GO TO 120
  110 RHO=S(N)/DELT
  120 IF (LEAK.NE.CHK(9)) GO TO 140
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 140
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 130
      HED2=TOP(N)
      U=0.
```

```
  130 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  140 CONTINUE
C
C     ---FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR G---
      E=-D-F-B-H-RHO-TL(N)*U-ETQB
      W=E-D*BE(J-1)
      BE(J)=F/W
      Q=-B*PHI(NA)-H*PHI(NB)-RHO*KEEP(N)-SL(N)-QRE(N)-WELL(N)+ETQD-SUBS-
     1TL(N)*STRT(N)-D*PHI(NL)-F*PHI(NR)-E*PHI(N)
      G(J)=(Q-D*G(J-1))/W
  150 CONTINUE
C
C     ---BACK SUBSTITUTE FOR TEMP---
      DO 160 KNO4=1,NO3
      NO4=DIMW-KNO4
      TEMP(NO4)=G(NO4)-BE(NO4)*TEMP(NO4+1)
  160 CONTINUE
C
C     ---EXTRAPOLATED VALUE OF PHI---
      DO 170 J=2,JNO1
      N=I+DIML*(J-1)
      PHI(N)=PHI(N)+HMAX*TEMP(J)
C
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERION---
      TCHK=DABS(TEMP(J))
      IF (TCHK.GT.BIGI) BIGI=TCHK
  170 CONTINUE
      IF (BIGI.GT.ERR) TEST=1.
      TEST3(KOUNT+1)=BIGI
      IF (KOUNT.EQ.0) GO TO 10
      IF (TEST.EQ.0.) RETURN
C
C     ---TEST FOR TWO DIMENSIONAL CORRECTION---
      IF (MOD(KOUNT,LENGTH).NE.0) GO TO 10
      GO TO 200
  180 DO 190 I=2,INO1
      DO 190 J=2,JNO1
      N=I+DIML*(J-1)
      IF (T(N).EQ.0.) GO TO 190
      PHI(N)=PHI(N)+ALFA(I)+BETA(J)
  190 CONTINUE
      GO TO 10
C     ..................................................................
C
C     ---TWO DIMENSIONAL CORRECTION TO LSOR---
C     ----------------------------------------
C
C     ---COMPUTE ALFA CORRECTION FOR ROWS---
  200 DO 210 I=1,DIML
      ALFA(I)=0.
      BE(I)=0.0
  210 G(I)=0.0
      DO 330 I=2,INO1
      A=0.
      B2=0.
      C=0.
      Q=0.
C
C     ---SUMMATION OF COEFFICIENTS FOR EACH ROW---
      DO 320 J=2,JNO1
```

```
      N=I+DIML*(J-1)
      NA=N-1
      NB=N+1
      NL=N-DIML
      NR=N+DIML
      IF (S(N).LT.0.) GO TO 330
      IF (T(N).EQ.0.) GO TO 320
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(N-DIML)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(N-1)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 230
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 230
      IF (PHE(N).GT.GRND(N)) GO TO 220
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 230
  220 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  230 IF (CONVRT.EQ.CHK(7)) GO TO 240
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 310
  240 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 280
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 270
      IF (KEEP(N)-PHE(N)) 250,260,260
  250 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 280
  260 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  270 RHO=SY(N)/DELT
      GO TO 290
  280 RHO=S(N)/DELT
  290 IF (LEAK.NE.CHK(9)) GO TO 310
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 310
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 300
      HED2=TOP(N)
      U=0.
  300 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  310 CONTINUE
C
      A=A-B
      B1=B+H+RHO+TL(N)*U+ETQB
      B2=B2+B1
      C=C-H
      Q=Q+(D*PHI(NL)+F*PHI(NR)+B*PHI(NA)+H*PHI(NB)+RHO*KEEP(N)+SL(N)+QRE
     1(N)+WELL(N)-ETQD+SUBS+TL(N)*STRT(N)-(D+F+B1)*PHI(N))
  320 CONTINUE
```

```
C
C     ---COMPUTATION OF INTERMEDIATE VECTOR G---
      W=B2-A*BE(I-1)
      BE(I)=C/W
      G(I)=(Q-A*G(I-1))/W
  330 CONTINUE
C
C     ---BACK SUBSTITUTE FOR ALFA---
      NO3=DIML-2
      DO 340 KNO4=1,NO3
      NO4=DIML-KNO4
  340 ALFA(NO4)=G(NO4)-BE(NO4)*ALFA(NO4+1)
C     ****************************************************************
C
C     ---COMPUTE BETA CORRECTION FOR COLUMNS---
      DO 350 J=1,DIMW
      BETA(J)=0.
      BE(J)=0.0
  350 G(J)=0.0
      DO 470 J=2,JNO1
      A=0.
      B2=0.
      C=0.
      Q=0.
C
C     ---SUMMATION OF COEFFICIENTS FOR EACH COLUMN---
      DO 460 I=2,INO1
      N=I+DIML*(J-1)
      NA=N-1
      NB=N+1
      NL=N-DIML
      NR=N+DIML
      IF (S(N).LT.0.) GO TO 470
      IF (T(N).EQ.0.) GO TO 460
      D=TR(N-DIML)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(N-1)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 370
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 370
      IF (PHE(N).GT.GRND(N)) GO TO 360
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 370
  360 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  370 IF (CONVRT.EQ.CHK(7)) GO TO 380
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 450
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
  380 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 420
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 410
```

```
      IF (KEEP(N)-PHE(N)) 390,400,400
  390 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 420
  400 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  410 RHO=SY(N)/DELT
      GO TO 430
  420 RHO=S(N)/DELT
  430 IF (LEAK.NE.CHK(9)) GO TO 450
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 450
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 440
      HED2=TOP(N)
      U=0.
  440 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  450 CONTINUE
C
      A=A-D
      B1=D+F+RHO+TL(N)*U+ETQB
      B2=B2+B1
      C=C-F
      Q=Q+(D*PHI(NL)+F*PHI(NR)+B*PHI(NA)+H*PHI(NB)+RHO*KEEP(N)+SL(N)+QRE
     1(N)+WELL(N)-ETQD+SUBS+TL(N)*STRT(N)-(B+H+B1)*PHI(N))
  460 CONTINUE
C
C     ---COMPUTATION OF INTERMEDIATE VECTOR G---
      W=B2-A*BE(J-1)
      BE(J)=C/W
      G(J)=(Q-A*G(J-1))/W
  470 CONTINUE
C
C     ---BACK SUBSTITUTE FOR BETA---
      NO3=DIMW-2
      DO 480 KNO4=1,NO3
      NO4=DIMW-KNO4
  480 BETA(NO4)=G(NO4)-BE(NO4)*BETA(NO4+1)
      GO TO 180
C     ..............................................................
C
C     ---FORMATS---
C
C     --------------------------------------------------------------
C
C
  490 FORMAT ('-',45X,'SOLUTION BY LINE SUCCESSIVE OVERRELAXATION'/46X,4
     12('_'))
  500 FORMAT ('-',26X,'ACCELERATION PARAMETER =',F6.3,'  TWO DIMENSIONAL
     1 CORRECTION EVERY',I5,' ITERATIONS')
  510 FORMAT ('0EXCEEDED PERMITTED NUMBER OF ITERATIONS'//' ',39('*'))
      END
```

```
      SUBROUTINE SOLVE3(PHI,BE,G,TEMP,KEEP,PHE,STRT,T,S,QRE,WELL,TL,SL,D
     1EL,ETA,V,XI,DELX,BETA,DELY,ALFA,XII,TEST3,TR,TC,GRND,SY,TOP,RATE,M
     2,RIVER)
C     ------------------------------------------------------------------
C     SOLUTION BY THE ALTERNATING DIRECTION IMPLICIT PROCEDURE
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,DBLE,RHOP(20),G,BE,TEMP,IMK,DABS,W,PARAM,TEST2,DMAX1,DT
     1ERMS,B1,E,Q,B,D,F,H,RHO,XII
      REAL *4KEEP,M
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,RCAL
C
      DIMENSION PHI(1), BE(1), G(1), TEMP(1), KEEP(1), PHE(1), STRT(1),
     1T(1), S(1), QRE(1), WELL(1), TL(1), SL(1), DEL(1), ETA(1), V(1), X
     2I(1), DELX(1), BETA(1), DELY(1), ALFA(1), XII(1), TEST3(1), TR(1),
     3 TC(1), GRND(1), SY(1), TOP(1), RATE(1), M(1), RIVER(1)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
      RETURN
C     ..................................................................
C
C     ---COMPUTE AND PRINT ITERATION PARAMETERS---
C     ********************
      ENTRY ITER3
C     ********************
      HMIN=2.
      IN4=DIMW-2
      IN5=DIML-2
      XVAL=3.1415**2/(2.*IN4*IN4)
      YVAL=3.1415**2/(2.*IN5*IN5)
      DO 10 I=2,INO1
      DO 10 J=2,JNO1
      N=I+DIML*(J-1)
      IF (T(N).EQ.0.) GO TO 10
      XPART=XVAL*(1/(1+DELX(J)**2*FACTY/DELY(I)**2*FACTX))
      YPART=YVAL*(1/(1+DELY(I)**2*FACTX/DELX(J)**2*FACTY))
      HMIN=AMIN1(HMIN,XPART,YPART)
   10 CONTINUE
      ALPHA=EXP(ALOG(HMAX/HMIN)/(LENGTH-1))
      RHOP(1)=HMIN
      DO 20 NTIME=2,LENGTH
   20 RHOP(NTIME)=RHOP(NTIME-1)*ALPHA
      WRITE (P,400)
      WRITE (P,410) LENGTH,(RHOP(J),J=1,LENGTH)
      RETURN
C     ..................................................................
C
C     ---INITIALIZE DATA FOR A NEW ITERATION---
   30 KOUNT=KOUNT+1
      IF (KOUNT.LE.ITMAX) GO TO 40
      WRITE (P,390)
      CALL TERM1
```

```
   40 IF (MOD(KOUNT,LENGTH)) 50,50,60
C     ************************
      ENTRY NEWITC
C     ************************
   50 NTH=0
   60 NTH=NTH+1
      PARAM=RHOP(NTH)
      TEST3(KOUNT+1)=0.
      TEST=0.
      N=DIML*DIMW
      DO 70 I=1,N
   70 PHE(I)=PHI(I)
      BIGI=0.0
C
C     ---COMPUTE TRANSMISSIVITY AND T COEFFICIENTS IN WATER TABLE
C     OR WATER TABLE-ARTESIAN SIMUATION---
      IF (WATER.NE.CHK(2)) GO TO 80
      CALL TRANS
C     ..................................................................
C
C     ---SOLUTION BY ADI---
C     ---------------------
C     ---COMPUTE IMPLICITLY ALONG ROWS---
   80 NO3=DIMW-2
      DO 90 J=1,DIMW
      N=1+DIML*(J-1)
   90 TEMP(J)=PHI(N)
      DO 230 I=2,DIML
      DO 200 J=2,JNO1
      N=I+DIML*(J-1)
      NA=N-1
      NB=N+1
      NL=N-DIML
      NR=N+DIML
      BE(J)=0.0
      G(J)=0.0
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 200
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(N-DIML)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(N-1)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 110
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 110
      IF (PHE(N).GT.GRND(N)) GO TO 100
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 110
  100 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  110 IF (CONVRT.EQ.CHK(7)) GO TO 120
      RHO=S(N)/DELT
```

```
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 190
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
  120 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 160
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 150
      IF (KEEP(N)-PHE(N)) 130,140,140
  130 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 160
  140 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  150 RHO=SY(N)/DELT
      GO TO 170
  160 RHO=S(N)/DELT
  170 IF (LEAK.NE.CHK(9)) GO TO 190
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 190
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 180
      HED2=TOP(N)
      U=0.
  180 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  190 CONTINUE
C
C     ---CALCULATE VALUES FOR PARAMETERS USED IN THOMAS ALGORITHM
C     AND FORWARD SUBSTITUTE TO COMPUTE INTERMEDIATE VECTOR G---
      IMK=(B+D+F+H)*PARAM
      E=-D-F-RHO-IMK-TL(N)*U-ETQB
      W=E-D*BE(J-1)
      BE(J)=F/W
      Q=-B*PHI(NA)+(B+H-IMK-E)*PHI(N)-H*PHI(NB)-RHO*KEEP(N)-SL(N)-QRE(N)
     1-WELL(N)+ETQD-SUBS-TL(N)*STRT(N)-D*PHI(NL)-F*PHI(NR)
      G(J)=(Q-D*G(J-1))/W
  200 CONTINUE
C
C     ---BACK SUBSTITUTE FOR HEAD VALUES AND PLACE THEM IN TEMP---
      XII(DIMW)=0.D0
      DO 220 KNO4=1,NO3
      NO4=DIMW-KNO4
      N=I+DIML*(NO4-1)
C
C     ---FIRST PLACE TEMP VALUES IN PHI(N-1)---
      PHI(N-1)=TEMP(NO4)
      IF (T(N).NE.0..AND.S(N).GE.0.) GO TO 210
      XII(NO4)=0.D0
      GO TO 220
  210 XII(NO4)=G(NO4)-BE(NO4)*XII(NO4+1)
  220 TEMP(NO4)=PHI(N)+XII(NO4)
  230 CONTINUE
C     ..................................................................
C
C     ---COMPUTE IMPLICITLY ALONG COLUMNS---
      NO3=DIML-2
      DO 240 I=1,DIML
  240 TEMP(I)=PHI(I)
      DO 380 J=2,DIMW
      DO 350 I=2,INO1
```

```
      N=I+DIML*(J-1)
      NA=N-1
      NB=N+1
      NL=N-DIML
      NR=N+DIML
      BE(I)=0.0
      G(I)=0.0
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(N).EQ.0..OR.S(N).LT.0.) GO TO 350
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(N-DIML)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(N-1)/DELY(I)
      H=TC(N)/DELY(I)
      IF (EVAP.NE.CHK(6)) GO TO 260
C
C     ---COMPUTE EXPLICIT AND IMPLICIT PARTS OF ET RATE---
      ETQB=0.
      ETQD=0.0
      IF (PHE(N).LE.GRND(N)-ETDIST) GO TO 260
      IF (PHE(N).GT.GRND(N)) GO TO 250
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-GRND(N))
      GO TO 260
  250 ETQD=QET
C
C     ---COMPUTE STORAGE TERM---
  260 IF (CONVRT.EQ.CHK(7)) GO TO 270
      RHO=S(N)/DELT
      IF (WATER.EQ.CHK(2)) RHO=SY(N)/DELT
      GO TO 340
C
C     ---COMPUTE STORAGE COEFFICIENT FOR CONVERSION PROBLEM---
  270 SUBS=0.0
      IF (KEEP(N).GE.TOP(N).AND.PHE(N).GE.TOP(N)) GO TO 310
      IF (KEEP(N).LT.TOP(N).AND.PHE(N).LT.TOP(N)) GO TO 300
      IF (KEEP(N)-PHE(N)) 280,290,290
  280 SUBS=(SY(N)-S(N))/DELT*(KEEP(N)-TOP(N))
      GO TO 310
  290 SUBS=(S(N)-SY(N))/DELT*(KEEP(N)-TOP(N))
  300 RHO=SY(N)/DELT
      GO TO 320
  310 RHO=S(N)/DELT
  320 IF (LEAK.NE.CHK(9)) GO TO 340
C
C     ---COMPUTE NET LEAKAGE TERM FOR CONVERSION SIMULATION---
      IF (RATE(N).EQ.0..OR.M(N).EQ.0.) GO TO 340
      HED1=AMAX1(STRT(N),TOP(N))
      U=1.
      HED2=0.
      IF (PHE(N).GE.TOP(N)) GO TO 330
      HED2=TOP(N)
      U=0.
  330 SL(N)=RATE(N)/M(N)*(RIVER(N)-HED1)+TL(N)*(HED1-HED2-STRT(N))
  340 CONTINUE
C
C     ---CALCULATE VALUES FOR PARAMETERS USED IN THOMAS ALGORITHM
C     AND FORWARD SUBSTITUTE TO COMPUTE INTERMEDIATE VECTOR G---
```

```
      IMK=(B+D+F+H)*PARAM                                               ADI2400
      E=-B-H-RHO-IMK-TL(N)*U-ETQB                                       ADI2410
      W=E-B*BE(I-1)                                                     ADI2420
      BE(I)=H/W                                                         ADI2430
      Q=-D*PHI(NL)+(D+F-IMK-E)*PHI(N)-F*PHI(NR)-RHO*KEEP(N)-SL(N)-QRE(N)ADI2440
     1-WELL(N)+ETQD-SUBS-TL(N)*STRT(N)-B*PHI(NA)-H*PHI(NB)              ADI2450
      G(I)=(Q-B*G(I-1))/W                                               ADI2460
  350 CONTINUE                                                          ADI2470
C                                                                       ADI2480
C     ---BACK SUBSTITUTE FOR HEAD VALUES AND PLACE THEM IN TEMP---      ADI2490
      XII(DIML)=0.DO                                                    ADI2500
      DO 370 KNO4=1,NO3                                                 ADI2510
      NO4=DIML-KNO4                                                     ADI2520
      N=NO4+DIML*(J-1)                                                  ADI2530
C                                                                       ADI2540
C     ---FIRST PLACE TEMP VALUES IN PHI(N-DIML)----                     ADI2550
      PHI(N-DIML)=TEMP(NO4)                                             ADI2560
      IF (T(N).NE.0..AND.S(N).GE.0.) GO TO 360                          ADI2570
      XII(NO4)=0.DO                                                     ADI2580
      TEMP(NO4)=PHI(N)                                                  ADI2590
      GO TO 370                                                         ADI2600
  360 XII(NO4)=G(NO4)-BE(NO4)*XII(NO4+1)                                ADI2610
      TEMP(NO4)=PHI(N)+XII(NO4)                                         ADI2620
C                                                                       ADI2630
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERION---          ADI2640
      TCHK=ABS(SNGL(TEMP(NO4))-PHE(N))                                  ADI2650
      IF (TCHK.GT.BIGI) BIGI=TCHK                                       ADI2660
  370 CONTINUE                                                          ADI2670
  380 CONTINUE                                                          ADI2680
      IF (BIGI.GT.ERR) TEST=1.                                          ADI2690
      TEST3(KOUNT+1)=BIGI                                               ADI2700
      IF (TEST.EQ.1.) GO TO 30                                          ADI2710
      RETURN                                                            ADI2720
C     ..................................................................ADI2730
C                                                                       ADI2740
C     ---FORMATS---                                                     ADI2750
C                                                                       ADI2760
C     ------------------------------------------------------------------ADI2770
C                                                                       ADI2780
C                                                                       ADI2790
  390 FORMAT ('0EXCEEDED PERMITTED NUMBER OF ITERATIONS'//' ',39('*'))  ADI2800
  400 FORMAT ('-',38X,'SOLUTION BY THE ALTERNATING DIRECTION IMPLICIT PRADI2810
     1OCEDURE'/39X,56('_'))                                             ADI2820
  410 FORMAT (///1H0,I5,22H ITERATION PARAMETERS:,8D12.3//28X,10D12.3)  ADI2830
      END                                                               ADI2840-
```

```
      SUBROUTINE COEF(PHI,KEEP,PHE,STRT,SURI,T,TR,TC,S,WELL,TL,SL,PERM,B
     1OTTOM,SY,RATE,RIVER,M,TOP,GRND,DELX,DELY,RSEG,IRIV,NRSEG)
C     -----------------------------------------------------------------
C     COMPUTE COEFFICIENTS
C     -----------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,DBLE,RHO,B,D,F,H
      REAL *4KEEP,M
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,RCAL
      DIMENSION ISORT(150)
C
      DIMENSION PHI(1), KEEP(1), PHE(1), STRT(1), SURI(1), T(1), TR(1),
     1TC(1), S(1), WELL(1), TL(1), SL(1), PERM(1), BOTTOM(1), SY(1), RAT
     2E(1), RIVER(1), M(1), TOP(1), GRND(1), DELX(1), DELY(1), RSEG(1),
     3IRIV(1), NRSEG(1)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
C
      DATA PIE/3.141593/
      RETURN
C     ...............................................................
C
C     ---COMPUTE COEFFICIENTS FOR TRANSIENT PART OF LEAKAGE TERM---
C     **********************
      ENTRY CLAY
C     **********************
      TMIN=1.E30
      TT=0.0
      PRATE=0.
      DO 50 I=1,DIML
      DO 50 J=1,DIMW
      N=I+DIML*(J-1)
C
C     ---SKIP COMPUTATIONS IF T, RATE OR M = 0, OR IF CONSTANT
C        HEAD BOUNDARY---
      IF (RATE(N).LE.0..OR.T(N).EQ.0..OR.M(N).EQ.0..OR.S(N).LT.0.) GO TO
     1 50
C
C     ---IF VALUE FOR TL(N  ) WILL EQUAL VALUE FOR PREVIOUS NODE,
C     SKIP PART OF COMPUTATIONS---
      IF (RATE(N)*M(N).EQ.PRATE) GO TO 40
      DIMT=RATE(N)*SUMP/(M(N)*M(N)*SS*3)
      IF (DIMT.GT.TT) TT=DIMT
      IF (DIMT.LT.TMIN) TMIN=DIMT
      PPT=PIE*PIE*DIMT
C
C     ---RECOMPUTE PPT IF DIMT WITHIN RANGE FOR SHORT TIME COMPUTATION--
      IF (DIMT.LT.1.0E-03) PPT=1.0/DIMT
      CC=(2.3-PPT)/(2.*PPT)
C
C     ---COMPUTE SUM OF EXPONENTIALS---
      SUMN=0.0
```

```
      DO 20 K=1,200
      POWER=K*K*PPT
      IF (POWER.LE.150.) GO TO 10
      POWER=150
   10 PEX=EXP(-POWER)
      SUMN=SUMN+PEX
      IF (PEX.GT.0.00009) GO TO 20
      IF (K.GT.CC) GO TO 30
   20 CONTINUE
C
C     ---COMPUTE DENOMINATER DEPENDING ON VALUE OF DIMT---
   30 DENOM=1.0
      IF (DIMT.LT.1.0E-03) DENOM=SQRT(PIE*DIMT)
C
C     ---HEAD VALUES ARE NOT INCLUDED IN COMPUTATION OF Q FACTOR SINCE
C        LEAKAGE IS CONSIDERED IMPLICITLY---
   40 Q1=RATE(N)/(M(N)*DENOM)
      TL(N)=Q1+2.*Q1*SUMN
      PRATE=RATE(N)*M(N)
   50 CONTINUE
      TMIN=TMIN*3.0
      TT=TT*3.0
      RETURN
C     ..............................................................
C
C     ---COMPUTE TRANSMISSIVITY IN WT OR WT-ARTESIAN CONVERSION PROBLEM-
C     ********************
      ENTRY TRANS
C     ********************
      DO 60 I=1,DIML
      DO 60 J=1,DIMW
      N=I+DIML*(J-1)
      IF (PERM(N).EQ.0.) GO TO 60
      HED=PHI(N)
      IF (CONVRT.EQ.CHK(7)) HED=AMIN1(SNGL(PHI(N)),TOP(N))
      T(N)=PERM(N)*(HED-BOTTOM(N))
      IF (T(N).GT.0.) GO TO 60
      IF (WELL(N).LT.0.) GO TO 70
C
C     ---THE FOLLOWING STATEMENTS APPLY WHEN NODES (EXCEPT WELL NODES)
C     GO DRY---
      PERM(N)=0.
      T(N)=0.0
      TR(N-DIML)=0.
      TR(N)=0.
      TC(N-1)=0.
      TC(N)=0.
      PHI(N)=SURI(N)
      WRITE (P,150) I,J
   60 CONTINUE
      IF (KT.EQ.0) RETURN
      GO TO 90
C
C     ---START PROGRAM TERMINATION WHEN A WELL GOES DRY---
   70 WRITE (P,120) I,J
      WRITE (P,130)
      IERR=1
      CALL DRDN
      DO 80 I=2,INO1
      DO 80 J=2,JNO1
```

```
      N=I+DIML*(J-1)
   80 PHI(N)=KEEP(N)
      SUM=SUM-DELT
      SUMP=SUMP-DELT
      KT=KT-1
      IF (KT.EQ.0) STOP
      IF (IDK2.EQ.CHK(15)) CALL DISK
      IF (PNCH.EQ.CHK(1)) CALL PUNCH
      IF (MOD(KT,KTH).EQ.0) STOP
      WRITE (P,140) KT,SUM
      CALL DRDN
      IF (CHCK.EQ.CHK(5)) CALL CWRITE
      STOP
C
C     ---COMPUTE T COEFFICIENTS---
C     ********************
      ENTRY TCOF
C     ********************
   90 DO 110 I=1,INO1
      DO 110 J=1,JNO1
      N=I+DIML*(J-1)
      NR=N+DIML
      NB=N+1
      IF (T(N).EQ.0.) GO TO 110
      IF (T(NR).EQ.0.) GO TO 100
      TR(N)=(2.*T(NR)*T(N))/(T(N)*DELX(J+1)+T(NR)*DELX(J))*FACTX
  100 IF (T(NB).EQ.0.) GO TO 110
      TC(N)=(2.*T(NB)*T(N))/(T(N)*DELY(I+1)+T(NB)*DELY(I))*FACTY
  110 CONTINUE
      RETURN
C
C
C     ---PREPARE A MATRIX WHICH WILL PUT RIVER SEGMENTS IN ORDER FROM
C        HIGHEST TO LOWEST RIVER HEADS---
C     ********************
      ENTRY RSORT
C     ********************
      DO 260 K=1,NSEG
      L=1
      DO 210 I=1,DIML
      DO 210 J=1,DIMW
      N=I+DIML*(J-1)
      IF (RIVER(N).EQ.0.) GO TO 210
      RR=RSEG(N)
      ISEG=INT(RR)
      IF (ISEG.NE.K) GO TO 210
      ISORT(L)=N
      L=L+1
  210 CONTINUE
      L=L-1
      LL=L-1
      LLL=L-1
      DO 240 II=1,LL
      DO 230 JJ=1,LLL
      IF (RIVER(ISORT(JJ+1)).GT.RIVER(ISORT(JJ))) GO TO 220
      GO TO 230
  220 LTEMP1=ISORT(JJ+1)
      LTEMP2=ISORT(JJ)
      ISORT(JJ)=LTEMP1
      ISORT(JJ+1)=LTEMP2
```

```
  230 CONTINUE                                                          F14631-
  240 CONTINUE                                                          F14632-
      DO 250 LJ=1,L                                                     F14633-
  250 IRIV(ISORT(LJ))=LJ                                                F14634-
      NRSEG(K)=L                                                        F14635-
  260 CONTINUE                                                          F14636-
      RETURN                                                            F14637-
C                                                                       COF1470
C     ---FORMATS---                                                     COF1480
C                                                                       COF1490
C     ------------------------------------------------------------------COF1500
C                                                                       COF1510
C                                                                       COF1520
  120 FORMAT ('-*************WELL',I3,',',I3,' GOES DRY**************') COF1530
  130 FORMAT ('1',50X,'DRAWDOWN WHEN WELL WENT DRY')                    COF1540
  140 FORMAT ('1',32X,'DRAWDOWN FOR TIME STEP',I3,'; SIMULATION TIME =',COF1550
     11PE15.7,' SECONDS')                                               COF1560
  150 FORMAT ('-',20('*'),' NODE ',I4,',',I4,' GOES DRY ',20('*'))      COF1570
      END                                                               COF1580-
```

```
      SUBROUTINE CHECKI(PHI,KEEP,PHE,STRT,T,TR,TC,S,QRE,WELL,TL,PERM,BOT
     1TOM,SY,RATE,RIVER,M,TOP,GRND,DELX,DELY,PICK,PICKUP,RSEG,IRIV,NRSEG
     2,FLOWIN,WIDE,SLOPE,IFLOUT,ROUGH,PREPIC,SL,FLOWAD)
C     ------------------------------------------------------------------
C     COMPUTE A MASS BALANCE
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,DBLE
      REAL *4KEEP,M
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,RCAL
C
      DIMENSION PHI(IZ,JZ), KEEP(IZ,JZ), PHE(IZ,JZ), STRT(IZ,JZ), T(IZ,J
     1Z), TR(IZ,JZ), TC(IZ,JZ), S(IZ,JZ), QRE(IZ,JZ), WELL(IZ,JZ), TL(IZ
     2,JZ), PERM(IZ,JZ), BOTTOM(IP,JP), SY(IP,JP), RATE(IR,JR), RIVER(IR
     3,JR), M(IR,JR), TOP(IC,JC), GRND(IL,JL), DELX(JZ), DELY(IZ), PICK(
     4IR,JR), PICKUP(IR,JR), RSEG(IR,JR), IRIV(IA,JA), NRSEG(IK), FLOWIN
     5(IK), WIDE(IK), SLOPE(IK), IFLOUT(IK), ROUGH(IK), PREPIC(IA,JA), S
     6L(IZ,JZ),FLOWAD(IK)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
      COMMON /CK/ ETFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT,I
     1RCHK,RAIN,PANEVA,IRCHK1
      COMMON /ARSIZE/ IZ,JZ,IP,JP,IR,JR,IC,JC,IL,JL,IS,JS,IH,IMAX,IMX1,I
     1K,IA,JA
      COMMON /MPARAM/ FACTQ,NEWW,NEWQ,ERLEAK,DAYS
      RETURN
C     ..................................................................
C     ********************
      ENTRY CHECK
C     ********************
C     ---INITIALIZE VARIABLES---
      PCHK=0.0
      PUMP=0.
      STOR=0.
      FLUXS=0.0
      CHD1=0.0
      CHD2=0.0
      QREFLX=0.
      CFLUX=0.
      FLUX=0.
      ETFLUX=0.
      FLXN=0.0
C     ..................................................................
C
C     ---COMPUTE RATES,STORAGE AND PUMPAGE FOR THIS STEP---
      DO 240 I=2,DIML
      DO 240 J=2,DIMW
      IF (T(I,J).EQ.0.) GO TO 240
      AREA=DELX(J)*DELY(I)
      IF (IRCHK.EQ.0) GO TO 235
      IF (S(I,J).GE.0.) GO TO 120
C
```

```
C     ---COMPUTE FLOW RATES TO AND FROM CONSTANT HEAD BOUNDARIES---
      IF (S(I,J-1).LT.0..OR.T(I,J-1).EQ.0.) GO TO 30
      X=(STRT(I,J)-PHI(I,J-1))*TR(I,J-1)*DELY(I)
      IF (X) 10,30,20
   10 CHD1=CHD1+X
      GO TO 30
   20 CHD2=CHD2+X
   30 IF (S(I,J+1).LT.0..OR.T(I,J+1).EQ.0.) GO TO 60
      X=(STRT(I,J)-PHI(I,J+1))*TR(I,J)*DELY(I)
      IF (X) 40,60,50
   40 CHD1=CHD1+X
      GO TO 60
   50 CHD2=CHD2+X
   60 IF (S(I-1,J).LT.0..OR.T(I-1,J).EQ.0.) GO TO 90
      X=(STRT(I,J)-PHI(I-1,J))*TC(I-1,J)*DELX(J)
      IF (X) 70,90,80
   70 CHD1=CHD1+X
      GO TO 90
   80 CHD2=CHD2+X
   90 IF (S(I+1,J).LT.0..OR.T(I+1,J).EQ.0.) GO TO 240
      X=(STRT(I,J)-PHI(I+1,J))*TC(I,J)*DELX(J)
      IF (X) 100,240,110
  100 CHD1=CHD1+X
      GO TO 240
  110 CHD2=CHD2+X
      GO TO 240
C
C     ---RECHARGE AND WELLS---
  120 QREFLX=QREFLX+QRE(I,J)*AREA
      IF (WELL(I,J)) 130,150,140
  130 PUMP=PUMP+WELL(I,J)*AREA
      GO TO 150
  140 CFLUX=CFLUX+WELL(I,J)*AREA
  150 IF (EVAP.NE.CHK(6)) GO TO 190
C
C     ---COMPUTE ET RATE---
      IF (PHI(I,J).GE.GRND(I,J)-ETDIST) GO TO 160
      ETQ=0.0
      GO TO 180
  160 IF (PHI(I,J).LE.GRND(I,J)) GO TO 170
      ETQ=QET
      GO TO 180
  170 ETQ=QET/ETDIST*(PHI(I,J)+ETDIST-GRND(I,J))
  180 ETFLUX=ETFLUX-ETQ*AREA
C
C     ---COMPUTE VOLUME FROM STORAGE---
  190 STORE=S(I,J)
      IF (WATER.EQ.CHK(2)) STORE=SY(I,J)
      IF (CONVRT.NE.CHK(7)) GO TO 230
      X=KEEP(I,J)-PHI(I,J)
      IF (X) 200,210,210
  200 HED1=PHI(I,J)
      HED2=KEEP(I,J)
      X=ABS(X)
      GO TO 220
  210 HED1=KEEP(I,J)
      HED2=PHI(I,J)
  220 STORE=S(I,J)
      IF (HED1-TOP(I,J).LE.0.) STORE=SY(I,J)
      IF ((HED1-TOP(I,J))*(HED2-TOP(I,J)).LT.0.0) STORE=(HED1-TOP(I,J))/
```

```
     1X*S(I,J)+(TOP(I,J)-HED2)/X*SY(I,J)                                CHK1110
  230 STOR=STOR+STORE*(KEEP(I,J)-PHI(I,J))*AREA                          CHK1120
C                                                                        CHK1130
C     ---COMPUTE LEAKAGE RATE---                                         CHK1140
  235 IF (LEAK.NE.CHK(9)) GO TO 240                                      CHK1150
      IF (M(I,J).EQ.0.) GO TO 240                                        CHK1160
      HED1=STRT(I,J)                                                     CHK1170
      IF (CONVRT.EQ.CHK(7)) HED1=AMAX1(STRT(I,J),TOP(I,J))               CHK1180
      HED2=PHI(I,J)                                                      CHK1190
      IF (CONVRT.EQ.CHK(7)) HED2=AMAX1(SNGL(PHI(I,J)),TOP(I,J))          CHK1200
      XX=RATE(I,J)*(RIVER(I,J)-HED1)*AREA/M(I,J)                         CHK1210
      YY=TL(I,J)*(HED1-HED2)*AREA                                        CHK1220
      FLUX=FLUX+XX                                                       CHK1230
      XNET=XX+YY                                                         CHK1240
      FLUXS=FLUXS+XNET                                                   CHK1250
      IF (XNET.LT.0.) FLXN=FLXN-XNET                                     CHK1260
      IF (RCAL.EQ.CHK(17)) PREPIC(I,J)=PICK(I,J)                         K12601-
      PICK(I,J)=XNET                                                     K12602-
      IF (RCAL.NE.CHK(17)) PICKUP(I,J)=PICKUP(I,J)+PICK(I,J)*DELT        K12603-
      IF (RCAL.NE.CHK(17)) GO TO 240                                     K12604-
      PCHK=AMAX1(PCHK,ABS(PICK(I,J)-PREPIC(I,J)))                        K12605-
  240 CONTINUE                                                           CHK1270
      IF (RCAL.NE.CHK(17)) GO TO 245                                     K12701-
      WRITE (P,930) PCHK,ERLEAK                                          K12702-
      IF (PCHK.GT.ERLEAK) GO TO 250                                      K12703-
      IRCHK1=IRCHK                                                       K12704-
      IRCHK=1                                                            K12705-
C     ...................................................................CHK1280
C                                                                        CHK1290
C     ---COMPUTE CUMULATIVE VOLUMES, TOTALS, AND DIFFERENCES---          CHK1300
  245 IF (IRCHK1.EQ.0) GO TO 250                                         K13001-
      STORT=STORT+STOR                                                   CHK1310
      STOR=STOR/DELT                                                     CHK1320
      ETFLXT=ETFLXT-ETFLUX*DELT                                          CHK1330
      FLUXT=FLUXT+FLUXS*DELT                                             CHK1340
      FLXNT=FLXNT+FLXN*DELT                                              CHK1350
      FLXPT=FLUXT+FLXNT                                                  CHK1360
      QRET=QRET+QREFLX*DELT                                              CHK1370
      CHDT=CHDT-CHD1*DELT                                                CHK1380
      CHST=CHST+CHD2*DELT                                                CHK1390
      PUMPT=PUMPT-PUMP*DELT                                              CHK1400
      CFLUXT=CFLUXT+CFLUX*DELT                                           CHK1410
      TOTL1=STORT+QRET+CFLUXT+CHST+FLXPT                                 CHK1420
      TOTL2=CHDT+PUMPT+ETFLXT+FLXNT                                      CHK1430
      SUMR=QREFLX+CFLUX+CHD2+CHD1+PUMP+ETFLUX+FLUXS+STOR                 CHK1440
      DIFF=TOTL2-TOTL1                                                   CHK1450
      PERCNT=0.0                                                         CHK1460
      IF (TOTL2.EQ.0.) GO TO 250                                         CHK1470
      PERCNT=DIFF/TOTL2*100.                                             CHK1480
  250 RETURN                                                             CHK1490
C     ...................................................................CHK1500
C                                                                        CHK1510
C     ---PRINT RESULTS---                                                CHK1520
C     ************************                                           CHK1530
      ENTRY CWRITE                                                       CHK1540
C     ************************                                           CHK1550
C                                                                        CHK1560
      IF (IRCHK1.EQ.0) GO TO 261                                         K15601-
      WRITE (P,260) STOR,QREFLX,STORT,CFLUX,QRET,PUMP,CFLUXT,ETFLUX,CHSTCHK1570
     1,FLXPT,CHD2,TOTL1,CHD1,FLUX,FLUXS,ETFLXT,CHDT,SUMR,PUMPT,FLXNT,TOTCHK1580
```

```
     2L2,DIFF,PERCNT
 261 IF (LEAK.NE.CHK(9)) GO TO 285
     IF (NSEG.EQ.0) GO TO 285
     WRITE (P,280)
     DO 275 K=1,NSEG
     PICSUM=0.
     PICKS=0.
     IF (RCAL.NE.CHK(17)) GO TO 262
     FLOSUM=0.0
     NNODE=NRSEG(K)
     DO 272 NN=1,NNODE
 262 DO 270 I=1,DIML
     DO 270 J=1,DIMW
     IF (RIVER(I,J).EQ.0.) GO TO 270
     RR=RSEG(I,J)
     ISEG=INT(RR)
     IF (ISEG.NE.K) GO TO 270
     IF (RCAL.NE.CHK(17)) GO TO 264
     IF (IRIV(I,J).NE.NN) GO TO 270
 264 RK=FLOAT(K)
     RRK=RR-RK
     RRK=RRK*10.0+0.1
     MR=INT(RRK)
     IF (MR.LT.1) GO TO 269
     IF (RCAL.NE.CHK(17)) GO TO 266
     IF (NN.EQ.1) FLOSUM=FLOWAD(K)
     FLOSUM=FLOSUM-PICK(I,J)
     IF (FLOSUM.LT.0.0.AND.MR.LT.3) PICK(I,J)=0.0
     IF (FLOSUM.LT.0.0) FLOSUM=0.
     PICKUP(I,J)=FLOSUM
 266 WRITE (P,290) I,J,PICK(I,J),PICKUP(I,J)
     PICSUM=PICSUM+PICK(I,J)
 269 PICKS=PICKS+PICK(I,J)
 270 CONTINUE
 272 CONTINUE
     IF (PICSUM.NE.0.) WRITE (P,310) PICSUM
     PICOUT=-PICSUM
     IF (IRCHK.EQ.1) WRITE (10,940) DAYS,PICOUT
 275 WRITE (P,300) K,PICKS
 285 RETURN
C
C    ..................................................................
C
C    ---CALCULATE RIVER HEADS---
C    ************************
     ENTRY HRIVER
C    ************************
C
     IF (NSEG.EQ.0) GO TO 510
     DO 390 K=1,NSEG
 390 FLOWAD(K)=FLOWIN(K)
     DO 500 K=1,NSEG
     FLOW=0.
     TOTAR=0.
     NNODE=NRSEG(K)
     DO 440 NN=1,NNODE
     DO 430 I=1,DIML
     DO 430 J=1,DIMW
     IF (RIVER(I,J).EQ.0.) GO TO 430
C    ---FIND THE RIVER SEGMENT---
```

```
      RR=RSEG(I,J)
      ISEG=INT(RR)
      IF (ISEG.NE.K) GO TO 430
C     ---FIND THE FIRST NODE OF RIVER SEGMENT---
      IF (IRIV(I,J).NE.NN) GO TO 430
      IF (NN.GT.1) GO TO 400
C     ---FIND THE CALCULATION OPTION FOR THE RIVER SEGMENT---
      GROUND=GRND(I,J)
      POND=RIVER(I,J)
      RK=FLOAT(K)
      RRK=RR-RK
      RRK=RRK*10.0+0.1
      MR=INT(RRK)
      MR=MR+1
  400 GO TO (500,500,410,420),MR
C     ---RIVER SEGMENT---
  410 IF (NN.EQ.1) FLOW=FLOWAD(K)
      FLOW=FLOW-PICK(I,J)
      IF (FLOW.LT.0.) FLOW=0.
      PAY=FLOW*ROUGH(K)
      PAYDA=1.486*WIDE(K)*(SQRT(SLOPE(K)))
      DEPTH=(PAY/PAYDA)**0.6
      RIVER(I,J)=GRND(I,J)+DEPTH
      IF (DEPTH.GT.0.0) GO TO 430
      RIVER(I,J)=GRND(I,J)
      WRITE (P,320) I,J
      GO TO 430
C     ---POND SEGMENT---
  420 IF (NN.EQ.1) FLOW=FLOWAD(K)*DELT
      IF (GRND(I,J).GT.GROUND) GO TO 425
      GROUND=GRND(I,J)
      POND=RIVER(I,J)
  425 AREA=DELX(J)*DELY(I)
      TOTAR=TOTAR+AREA
      RAINF=RAIN*AREA*DELT
      EVAPO=PANEVA*AREA*DELT
      FLOW=FLOW+RAINF-EVAPO-(PICK(I,J)*DELT)
      IF (NN.EQ.1) FLEAK=FLOWAD(K)
      FLEAK=FLEAK-PICK(I,J)
  430 CONTINUE
  440 CONTINUE
      GO TO (500,500,490,450),MR
C     ---POND SEGMENT---
  450 KK=IFLOUT(K)
      IF (KK.EQ.0) GO TO 465
      DO 460 I=1,DIML
      DO 460 J=1,DIMW
      IF (RIVER(I,J).EQ.0.) GO TO 460
      RR=RSEG(I,J)
      ISEG=INT(RR)
      IF (ISEG.NE.KK) GO TO 460
      IF (IRIV(I,J).NE.1) GO TO 460
      IKK=I
      JKK=J
  460 CONTINUE
      IF (POND.LT.GRND(IKK,JKK)) GO TO 465
      FLOWR=(TOTAR*(RAIN-PANEVA))+FLEAK
      IF (FLOWR.LT.0.) GO TO 465
  462 PAY=FLOWR*ROUGH(KK)
      PAYDA=1.486*WIDE(KK)*(SQRT(SLOPE(KK)))
```

```
      DST=(PAY/PAYDA)**0.6                                              K16020-
      PONDN=(3*DST/2)+GRND(IKK,JKK)                                     K16021-
      GO TO 467                                                         K16022-
  465 DEPTH=FLOW/TOTAR                                                  K16023-
      PONDN=POND+DEPTH                                                  K16024-
      IF (KK.EQ.0) GO TO 467                                            K16025-
      IF (PONDN.LT.GRND(IKK,JKK)) GO TO 467                             K16026-
      PONDIF=GRND(IKK,JKK)-POND                                         K16027-
      FLODIF=FLOW-(PONDIF*TOTAR)                                        K16028-
      FLOWR=FLODIF/DELT                                                 K16029-
      GO TO 462                                                         K16030-
  467 CONTINUE                                                          K16031-
      DO 480 I=1,DIML                                                   K16032-
      DO 480 J=1,DIMW                                                   K16033-
      IF (RIVER(I,J).EQ.0.) GO TO 480                                   K16034-
      RR=RSEG(I,J)                                                      K16035-
      ISEG=INT(RR)                                                      K16036-
      IF (ISEG.NE.K) GO TO 480                                          K16037-
      RIVER(I,J)=PONDN                                                  K16038-
      IF (RIVER(I,J).LT.GRND(I,J)) GO TO 470                            K16039-
      GO TO 480                                                         K16040-
  470 RIVER(I,J)=GRND(I,J)                                              K16041-
      WRITE (P,330) I,J                                                 K16042-
  480 CONTINUE                                                          K16043-
      IF (KK.GT.0.AND.FLOWR.GT.0.0) FLOWAD(KK)=FLOWR                    K16044-
      GO TO 500                                                         K16045-
  490 IF (IFLOUT(K).NE.0) FLOWAD(IFLOUT(K))=FLOWIN(IFLOUT(K))+FLOW      K16046-
  500 CONTINUE                                                          K16047-
C                                                                       K16048-
      DO 505 I=1,DIML                                                   K16049-
      DO 505 J=1,DIMW                                                   K16050-
      IF (M(I,J).EQ.0.) GO TO 505                                       K16051-
      SL(I,J)=RATE(I,J)/M(I,J)*(RIVER(I,J)-STRT(I,J))                   K16052-
  505 CONTINUE                                                          K16053-
      DO 800 I=1,DIML                                                   K16054-
  800 WRITE (P,990) I,(RIVER(I,J),J=1,DIMW)                             K16055-
  510 RETURN                                                            K16056-
C                                                                       CHK1610
C     ---FORMATS---                                                     CHK1620
C                                                                       CHK1630
C     ------------------------------------------------------------------CHK1640
C                                                                       CHK1650
C                                                                       CHK1660
  260 FORMAT ('0',10X,'CUMULATIVE MASS BALANCE:',16X,'L**3',23X,'RATES FCHK1670
     1OR THIS TIME STEP:',16X,'L**3/T'/11X,24('-'),43X,25('-')//20X,'SOUCHK1680
     2RCES:',69X,'STORAGE =',F20.4/20X,8('-'),68X,'RECHARGE =',F20.4/27XCHK1690
     3,'STORAGE =',F20.2,35X,'CONSTANT FLUX =',F20.4/26X,'RECHARGE =',F2CHK1700
     40.2,41X,'PUMPING =',F20.4/21X,'CONSTANT FLUX =',F20.2,30X,'EVAPOTRCHK1710
     5ANSPIRATION =',F20.4/21X,'CONSTANT HEAD =',F20.2,34X,'CONSTANT HEACHK1720
     6D:'/27X,'LEAKAGE =',F20.2,46X,'IN =',F20.4/21X,'TOTAL SOURCES =',FCHK1730
     720.2,45X,'OUT =',F20.4/96X,'LEAKAGE:'/20X,'DISCHARGES:',45X,'FROM CHK1740
     8PREVIOUS PUMPING PERIOD =',F20.4/20X,11('-'),68X,'TOTAL =',F20.4/1CHK1750
     96X,'EVAPOTRANSPIRATION =',F20.2/21X,'CONSTANT HEAD =',F20.2,36X,'SCHK1760
     $UM OF RATES =',F20.4/19X'QUANTITY PUMPED =',F20.2/27X,'LEAKAGE =',CHK1770
     $F20.2/19X,'TOTAL DISCHARGE =',F20.2//17X,'DISCHARGE-SOURCES =',F20CHK1780
     $.2/15X,'PER CENT DIFFERENCE =',F20.2//)                           CHK1790
  280 FORMAT (1H1,///,60X,'LEAKAGE INTO (-) AND OUT OF (+) THE RIVER',/,K17901-
     160X,41('-'),///,64X,'LEAKAGE RATE',10X,'  CUMULATIVE FLOW ',/,42X,K17902-
     2'I',9X,'J',13X,'(L**3/T)',18X,'(L**3/T)',/,40X,5('-'),5X,5('-'),5XK17903-
     3,20('-'),5X,20('-'),//)                                           K17904-
```

```
290 FORMAT (41X, I3, 7X, I3, 11X, F10. 4, 10X, F15. 4)                       K17905-
300 FORMAT ('-', //, 78X, 'RIVER SEGMENT NUMBER =', I5, /, 74X, 'SUM OF THE SK17906-
   2EEPAGE RATES =', F10. 4, 2X, '(L**3/T)', ////)                           K17907-
310 FORMAT ('-', /, 58X, 'TOTAL =', F10. 4)                                  K17908-
320 FORMAT ('-', 10X, '**********WARNING RIVER NODE', I5, ', ', I5, 'GOES DRYK17909-
   1**********')                                                             K17910-
330 FORMAT ('-', 10X, '***********WARNING POND NODE', I5, ', ', I5, 'GOES DRYK17911-
   1**********')                                                             K17912-
930 FORMAT ('-', 30X, 'MAXIMUM CHANGE IN LEAKAGE =', F10. 4, 'ERROR CRITERIK17913-
   1ON =', F10. 4)                                                           K17914-
990 FORMAT ('0', I2, 2X, 20F6. 1/(5X, 20F6. 1))                              K17915-
    END                                                                      CHK1800-
```

```
      SUBROUTINE PRNTAI(PHI,SURI,T,S,WELL,DELX,DELY)
C     ------------------------------------------------------------------
C     PRINT MAPS OF DRAWDOWN AND HYDRAULIC HEAD
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,Z,XLABEL,YLABEL,TITLE,XN1,MESUR
      REAL *4K
      INTEGER R,P,PU,DIML,DIMW,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,
     1CONTR,LEAK,RECH,SIP,ADI,RCAL
C
      DIMENSION PHI(IZ,JZ), SURI(IZ,JZ), S(IZ,JZ), WELL(IZ,JZ), DELX(JZ)
     1, DELY(IZ), T(IZ,JZ)
C
      COMMON /SARRAY/ VF4(11),CHK(17)
      COMMON /SPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH,SIP,U,SS,TT,TMIN,ETDIST,QET,ERR,TMAX,CDLT,HMAX,YDIM,WIDTH,
     2NUMS,LSOR,ADI,DELT,SUM,SUMP,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,
     3IERR,KOUNT,IFINAL,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,DIML,DI
     4MW,JNO1,INO1,R,P,PU,I,J,IDK1,IDK2,NSEG,RCAL
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(5),XN1,MESUR,PRNT(122),BLANK
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
      COMMON /ARSIZE/ IZ,JZ,IP,JP,IR,JR,IC,JC,IL,JL,IS,JS,IH,IMAX,IMX1,I
     1K,IA,JA
      RETURN
C     ..................................................................
C
C     ---INITIALIZE VARIABLES FOR PLOT---
C     ********************
      ENTRY MAP
C     ********************
   10 XSF=DINCH*XSCALE
      YSF=DINCH*YSCALE
      NYD=YDIM/YSF
      IF (NYD*YSF.LE.YDIM-DELY(INO1)/2.) NYD=NYD+1
      IF (NYD.LE.12) GO TO 20
      DINCH=YDIM/(12.*YSCALE)
      WRITE (P,310) DINCH
      IF (YSCALE.LT.1.0) WRITE (P,320)
      GO TO 10
   20 NXD=WIDTH/XSF
      IF (NXD*XSF.LE.WIDTH-DELX(JNO1)/2.) NXD=NXD+1
      N4=NXD*N1+1
      N5=NXD+1
      N6=NYD+1
      N8=N2*NYD+1
      NA(1)=N4/2-1
      NA(2)=N4/2
      NA(3)=N4/2+3
      NC=(N3-N8-10)/2
      ND=NC+N8
      NE=MAX0(N5,N6)
      VF1(3)=DIGIT(ND)
      VF2(3)=DIGIT(ND)
      VF3(3)=DIGIT(NC)
      XLABEL(3)=MESUR
      YLABEL(6)=MESUR
      DO 40 I=1,NE
```

```
      NNX=N5-I                                                          PRN 590
      NNY=I-1                                                           PRN 600
      IF (NNY.GE.N6) GO TO 30                                           PRN 610
      YN(I)=YSF*NNY/YSCALE                                              PRN 620
   30 IF (NNX.LT.0) GO TO 40                                            PRN 630
      XN(I)=XSF*NNX/YSCALE                                              PRN 640
   40 CONTINUE                                                          PRN 650
      RETURN                                                            PRN 660
C     ..................................................................PRN 670
C                                                                       PRN 680
C     ********************                                              PRN 690
      ENTRY PRNTA(NG)                                                   PRN 700
C     ********************                                              PRN 710
C     ---VARIABLES INITIALIZED EACH TIME A PLOT IS REQUESTED---         PRN 720
      DIST=WIDTH-DELX(JNO1)/2.                                          PRN 730
      JJ=JNO1                                                           PRN 740
      LL=1                                                              PRN 750
      Z=NXD*XSF                                                         PRN 760
      IF (NG.EQ.1) WRITE (P,280) (TITLE(I),I=1,2)                       PRN 770
      IF (NG.EQ.2) WRITE (P,280) (TITLE(I),I=3,5)                       PRN 780
      DO 270 I=1,N4                                                     PRN 790
C                                                                       PRN 800
C     ---LOCATE X AXES---                                               PRN 810
      IF (I.EQ.1.OR.I.EQ.N4) GO TO 50                                   PRN 820
      PRNT(1)=SYM(12)                                                   PRN 830
      PRNT(N8)=SYM(12)                                                  PRN 840
      IF ((I-1)/N1*N1.NE.I-1) GO TO 70                                  PRN 850
      PRNT(1)=SYM(14)                                                   PRN 860
      PRNT(N8)=SYM(14)                                                  PRN 870
      GO TO 70                                                          PRN 880
C                                                                       PRN 890
C     ---LOCATE Y AXES---                                               PRN 900
   50 DO 60 J=1,N8                                                      PRN 910
      IF ((J-1)/N2*N2.EQ.J-1) PRNT(J)=SYM(14)                           PRN 920
   60 IF ((J-1)/N2*N2.NE.J-1) PRNT(J)=SYM(13)                           PRN 930
C                                                                       PRN 940
C     ---COMPUTE LOCATION OF NODES AND DETERMINE APPROPRIATE SYMBOL---  PRN 950
   70 IF (DIST.LT.0..OR.DIST.LT.Z-XN1*XSF) GO TO 220                    PRN 960
      YLEN=DELY(2)/2.                                                   PRN 970
      DO 200 L=2,IND1                                                   PRN 980
      J=YLEN*N2/YSF+1.5                                                 PRN 990
      IF (T(L,JJ).EQ.0.) GO TO 140                                      PRN1000
      IF (S(L,JJ).LT.0.) GO TO 190                                      PRN1010
      INDX3=0                                                           PRN1020
      GO TO (80,90), NG                                                 PRN1030
   80 K=(SURI(L,JJ)-PHI(L,JJ))*FACT1                                    PRN1040
C     -TO CYCLE SYMBOLS FOR DRAWDOWN, REMOVE C FROM COL. 1 OF NEXT CARD-PRN1050
C     K=AMOD(K,10.)                                                     PRN1060
      GO TO 100                                                         PRN1070
   90 K=PHI(L,JJ)*FACT2                                                 PRN1080
  100 IF (K) 110,140,120                                                PRN1090
  110 IF (J-2.GT.0) PRNT(J-2)=SYM(13)                                   PRN1100
      N=-K                                                              PRN1110
      IF (N.LT.100) GO TO 130                                           PRN1120
      GO TO 170                                                         PRN1130
  120 N=K                                                               PRN1140
      IF (N.LT.100) GO TO 130                                           PRN1150
      IF (N.GT.999) GO TO 170                                           PRN1160
      INDX3=N/100                                                       PRN1170
      IF (J-2.GT.0) PRNT(J-2)=SYM(INDX3)                                PRN1180
```

```
      N=N-INDX3*100
  130 INDX1=MOD(N,10)
      IF (INDX1.EQ.0) INDX1=10
C     -TO CYCLE SYMBOLS FOR DRAWDOWN, REMOVE C FROM COL. 1 OF NEXT CARD-
C     IF (NG.EQ.1) GO TO 150
      INDX2=N/10
      IF (INDX2.GT.0) GO TO 160
      INDX2=10
      IF (INDX3.EQ.0) INDX2=15
      GO TO 160
  140 INDX1=15
  150 INDX2=15
  160 IF (J-1.GT.0) PRNT(J-1)=SYM(INDX2)
      PRNT(J)=SYM(INDX1)
      GO TO 200
  170 DO 180 II=1,3
      JI=J-3+II
  180 IF (JI.GT.0) PRNT(JI)=SYM(11)
  190 IF (S(L,JJ).LT.0.) PRNT(J)=SYM(16)
  200 YLEN=YLEN+(DELY(L)+DELY(L+1))/2.
  210 DIST=DIST-(DELX(JJ)+DELX(JJ-1))/2.
      JJ=JJ-1
      IF (JJ.EQ.0) GO TO 220
      IF (DIST.GT.Z-XN1*XSF) GO TO 210
  220 CONTINUE
C
C     ---PRINT AXES,LABELS, AND SYMBOLS---
      IF (I-NA(LL).EQ.0) GO TO 240
      IF ((I-1)/N1*N1-(I-1)) 250,230,250
  230 WRITE (P,VF1) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XN(1+(I-1)/6)
      GO TO 260
  240 WRITE (P,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XLABEL(LL)
      LL=LL+1
      GO TO 260
  250 WRITE (P,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8)
C
C     ---COMPUTE NEW VALUE FOR Z AND INITIALIZE PRNT---
  260 Z=Z-2.*XN1*XSF
      DO 270 J=1,N8
  270 PRNT(J)=SYM(15)
C
C     ---NUMBER AND LABEL Y AXIS AND PRINT LEGEND---
      WRITE (P,VF3) (BLANK(J),J=1,NC),(YN(I),I=1,N6)
      WRITE (P,300) (YLABEL(I),I=1,6)
      IF (NG.EQ.1) WRITE (P,290) FACT1
      IF (NG.EQ.2) WRITE (P,290) FACT2
      RETURN
C
C     ---FORMATS---
C
C     -----------------------------------------------------------------
C
C
  280 FORMAT ('1',53X,4A8//)
  290 FORMAT ('0EXPLANATION'/' ',11('-')//' R = CONSTANT HEAD BOUNDARY'/
     1' *** = VALUE EXCEEDED 3 FIGURES'/' MULTIPLICATION FACTOR =',F8.3)
  300 FORMAT ('0',39X,6A8)
  310 FORMAT ('0',25X,10('*'),' TO FIT MAP WITHIN 12 INCHES, DINCH REVIS
     1ED TO',G15.7,1X,10('*'))
  320 FORMAT ('0',45X,'NOTE: GENERALLY SCALE SHOULD BE > OR = 1.0')
```

```
      END
      BLOCK DATA
C     ----------
      REAL *8XLABEL, YLABEL, TITLE, XN1, MESUR, RHO, B, D, F, H
      INTEGER R, P, PU, DIML, DIMW, CHK, WATER, CONVRT, EVAP, CHCK, PNCH, NUM, HEAD,
     1CONTR, LEAK, RECH, SIP, ADI, NEWW, NEWQ, RCAL
C
      COMMON /DPARAM/ RHO, B, D, F, H
      COMMON /SARRAY/ VF4(11), CHK(17)
      COMMON /MPARAM/ FACTQ, NEWW, NEWQ, ERLEAK, DAYS
      COMMON /SPARAM/ WATER, CONVRT, EVAP, CHCK, PNCH, NUM, HEAD, CONTR, EROR, LE
     1AK, RECH, SIP, U, SS, TT, TMIN, ETDIST, QET, ERR, TMAX, CDLT, HMAX, YDIM, WIDTH,
     2NUMS, LSOR, ADI, DELT, SUM, SUMP, SUBS, STORE, TEST, ETQB, ETQD, FACTX, FACTY,
     3IERR, KOUNT, IFINAL, NUMT, KT, KP, NPER, KTH, ITMAX, LENGTH, NWEL, NW, DIML, DI
     4MW, JNO1, INO1, R, P, PU, I, J, IDK1, IDK2, NSEG, RCAL
      COMMON /PR/ XLABEL(3), YLABEL(6), TITLE(5), XN1, MESUR, PRNT(122), BLANK
     1(60), DIGIT(122), VF1(6), VF2(6), VF3(7), XSCALE, DINCH, SYM(17), XN(100),
     2YN(13), NA(4), N1, N2, N3, YSCALE, FACT1, FACT2
      COMMON /ARSIZE/ IZ, JZ, IP, JP, IR, JR, IC, JC, IL, JL, IS, JS, IH, IMAX, IMX1, I
     1K, IA, JA
C     ******************************************************************
C
      DATA IZ, JZ, IP, JP, IR, JR, IC, JC, IL, JL, IS, JS, IMAX/13*20/, IH/1/
      DATA CHK/'PUNC', 'WATE', 'CONT', 'NUME', 'CHEC', 'EVAP', 'CONV', 'HEAD', '
     1LEAK', 'RECH', 'SIP ', 'LSOR', 'ADI', 'DK1 ', 'DK2 ', 'YES ', 'CALC'/, R, P,
     2PU/5, 6, 7/, B, D, F, H/4*0. DO/
      DATA SYM/'1', '2', '3', '4', '5', '6', '7', '8', '9', '0', '*', '|', '-', '+', '
     1 ', 'R', 'W'/
      DATA PRNT/122*' '/, N1, N2, N3, XN1/6, 10, 133, . 833333333D-1/, BLANK/60*'
     1 '/, NA(4)/1000/
      DATA XLABEL/' X DIS- ', 'TANCE IN', '  MILES '/, YLABEL/'DISTANCE', '
     1FROM OR', 'IGIN IN ', 'Y DIRECT', 'ION, IN ', 'MILES    '/, TITLE/'PLOT
     2OF ', 'DRAWDOWN', 'PLOT OF ', 'HYDRAULI', 'C HEAD'/
      DATA DIGIT/'1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13'
     1, '14', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24', '25', '26',
     2'27', '28', '29', '30', '31', '32', '33', '34', '35', '36', '37', '38', '39', '
     340', '41', '42', '43', '44', '45', '46', '47', '48', '49', '50', '51', '52', '5
     43', '54', '55', '56', '57', '58', '59', '60', '61', '62', '63', '64', '65', '66
     5', '67', '68', '69', '70', '71', '72', '73', '74', '75', '76', '77', '78', ' 79
     6', '80', '81', '82', '83', '84', '85', '86', '87', '88', '89', '90', '91', '92'
     7, '93', '94', '95', '96', '97', '98', '99', '100', '101', '102', '103', ' 104'
     8, '105', '106', '107', '108', '109', '110', '111', '112', '113', '114', '115'
     9, '116', '117', '118', '119', '120', '121', '122'/
      DATA VF1/'(1H ', ', ', '     ', 'A1, F', '10. 2', ')'/
      DATA VF2/'(1H ', ', ', '     ', 'A1, 1', 'X, A8', ')'/
      DATA VF3/'(1H0', ', ', '     ', 'A1, F', '3. 1, ', '12F1', '0. 2)'/
      DATA VF4/'(1H0', ', ', '     ', 'X, I2', ', 2X, ', '20F6', '. 1/(', '     ', 'X, 2
     10', 'F6. 1', '))'/
C     ******************************************************************
      END
%%
```


✩ U.S. GOVERNMENT PRINTING OFFICE: 1984 700-907