APPLICATION OF NUMERICAL MODEL FOR GROUND-WATER MANAGEMENT OF PIEDMONT AQUIFER IN CENTRAL DINAJPUR, BANGLADESH

By

SYED ABUL HOSSAIN

Bachelor of Science
University of Dacca
Dacca, Bangladesh
1959

Master of Science
University of Dacca
Dacca, Bangladesh
1961

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 1981

APPLICATION OF NUMERICAL MODEL FOR
GROUND-WATER MANAGEMENT OF
PIEDMONT AQUIFER IN
CENTRAL DINAJPUR,
BANGLADESH

Thesis Approved:

Douglas C. Kent
Thesis Adviser

Wayne A. Pettyjohn

Zuhair al-Shaieb

Norman N. Durham
Dean of the Graduate College

PREFACE

In the present study, an attempt has been made to evaluate the geologic and hydrogeologic properties of the Piedmont aquifer in the Dinajpur district and to use those evaluated values in a ground-water management model. The primary objective is to determine the maximum annual yield and corresponding annual pumping allocation for the basin based on allowable limiting drawdown up to 22 feet from the surface. The computer model is used to determine the maximum annual yield based on predicted changes in the potentiometric surface (water table) caused by irrigation pumpage prior to 1977 and subsequent allocated irrigation pumpage until 1997.

The author is indebted to Dr. Douglas C. Kent, his thesis adviser, for his valuable advice and guidance during this study. It was only due to his sincere and untiring efforts that the author was able to undertake a research project in his own country and was able to present the paper in the present form and standard. Sincere appreciation is extended to him for always being patient beyond limits, constantly fair beyond measure, and extremely helpful in directing organization of data and this research.

The author is very much grateful to Dr. Wayne A. Pettyjohn, Chairman of Geology Department, committee member for showing personal interest in this project, for his suggestions and for providing his valuable publications pertinent to all this field of study. Special thanks and gratitude are due to Dr. Zuhair Al-Shaieb, committee member for his guidance and encouragement related to the completion of this thesis. Appreciation is also extended to Dr. Fred E. Witz, computer-system specialist, for his programming assistance in the adaptation of the U.S. Geological Survey ground-water model to the Piedmont aquifer in Dinajpur.

The author expresses his sincere respect and gratitude to the Bangladesh Water Development Board and United Nations for providing the opportunity and facilities for training the author in Hydrogeology. The author acknowledges with thanks the cooperation of the officials of Ground-water Circle, Bangladesh Water Development Board and for providing the geologic and hydrogeologic data as used for this study.

Finally, special gratitude is expressed to author's wife, Ranu, and his children Tutul, Shumi, and Rupa for their loving consideration and patience during the time of study. This thesis is dedicated to them.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table		Page

LIST OF FIGURES

CHAPTER I

ABSTRACT

The Piedmont alluvial deposit in the central part of Dinajpur district extends over an area of approximately 588 square miles. The aquifer, which consists of unconsolidated sand, gravel and clay, is the principal source of ground-water supply of 266 wells for irrigation and about 10,500 dug and handpump wells for domestic purposes. The water table contour map and lithology indicate that the aquifer is homogeneous, isotropic, and unconfined. The estimated values used for hydraulic conductivity and specific yield in all areas were 1.02×10^{-3} ft/sec and 0.25, respectively. The net surface recharge was determined to be 24 inches per year.

Concern has been expressed that extensive irrigation pumpage might lower the water table below 22 feet and stop or severely limit the use of 10,500 dug and hand pump wells for domestic purposes. In order to solve this problem, geological and hydrogeological investigations were conducted. A digital model (Trescott - Pinder model) was then used to provide a quantitative description of the aquifer and to predict maximum annual allocation for the aquifer based on allowable drawdown at a specified future time.

The computer simulation, with the help of Trescott-Pinder model, was first run for one year period in order to calibrate the model. It is assumed that the aquifer is in recharge - discharge equilibrium. After calibration was completed, a 20-year simulation was run to determine a mass balance relationship and maximum annual allocation for the aquifer based on irrigation requirements and allowable drawdown. An annual allocation of 2.23 acre-feet/acre was established for an allowable drawdown of 22 feet in the Piedmont aquifer.

CHAPTER II

INTRODUCTION

General

Bangladesh is located between 20° 45' and 26° 41' North latitude and 88° 03' and 92° 40' East longitude, with an area of 55,000 square miles. It is a major part of the Ganges-Bhramaputra Meghna delta which is roughly twice the size of the Mississippi delta (Figure 1). With the population averaging 1300 persons per square mile and the projection for the next 25 years exceeding 3000 per square mile, the internal food demand is likely to double. But the agricultural production is not sufficient to feed the rapidly increasing population. Thus the only way in which Bangladesh can meet her future food requirements is by expanding agriculture into the dry season by using irrigation. An extensive program is now underway to provide irrigation water from wells.

Location

The study area lies approximately between latitudes 26° 23' and 25° 57'N and longitudes 88° 18' and 88° 48'E (Figure 2). The area comprises 588 square miles and is

LOCATION MAP OF BANGLADESH

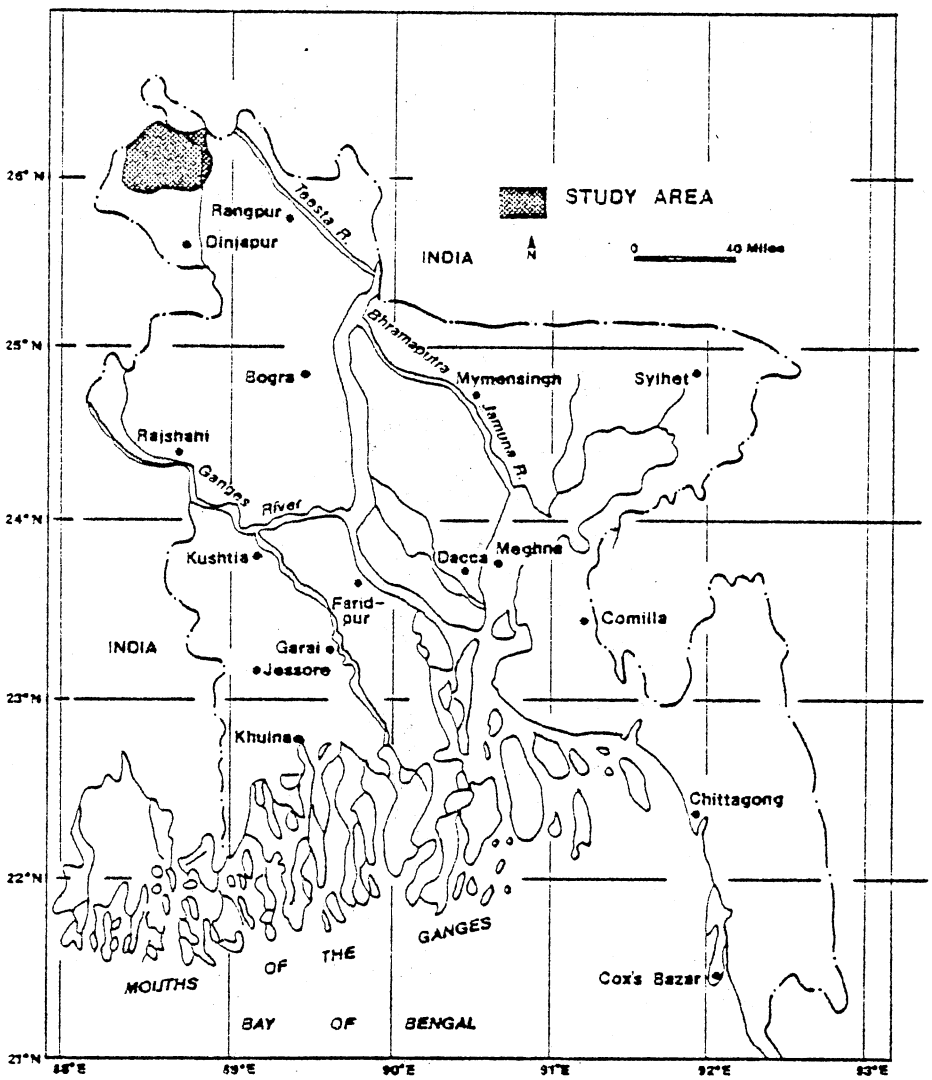


Figure 1. Location Map of Bangladesh

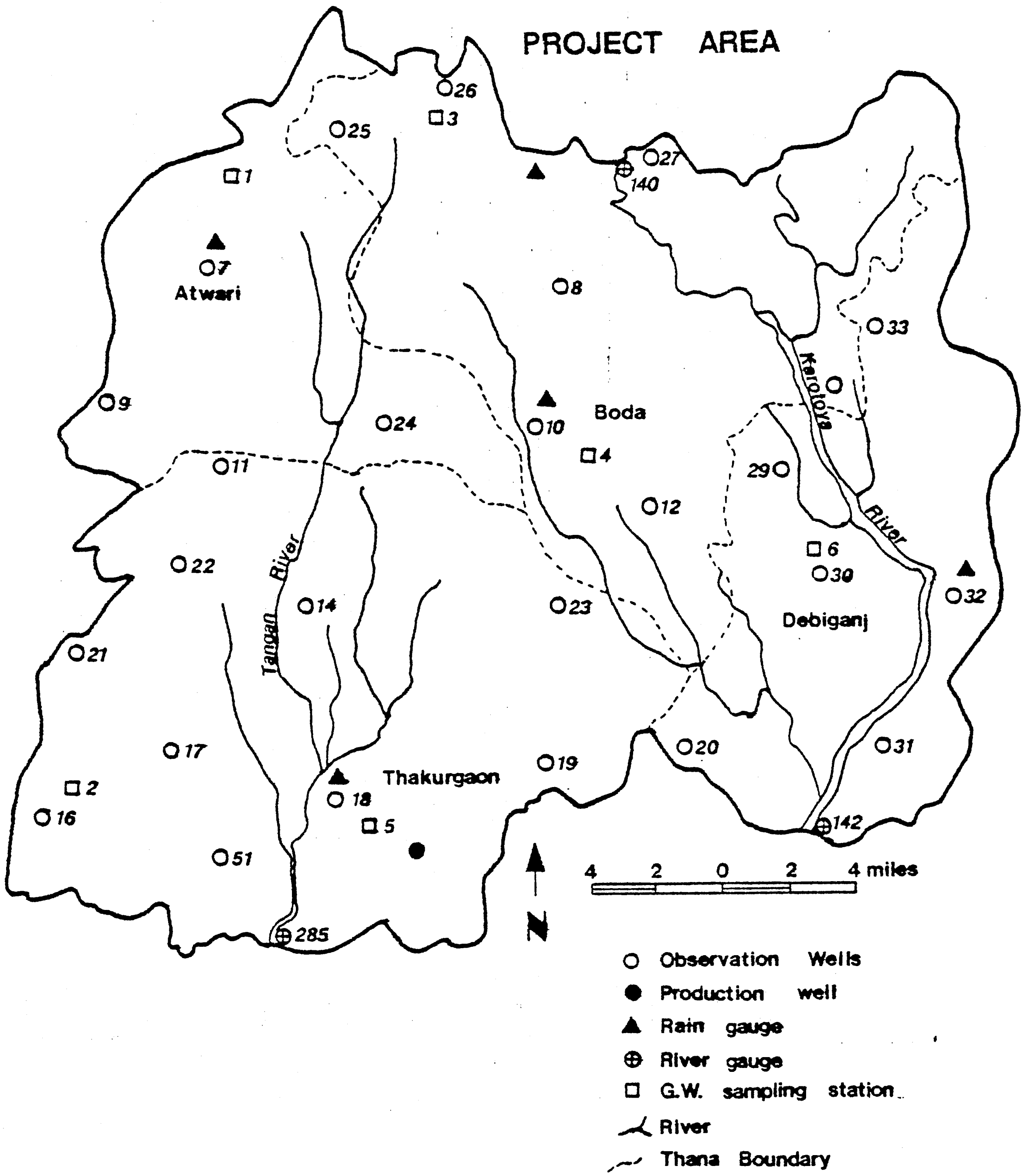


Figure 2. Location of Study Area

located in a central part of Dinajpur district of Bangladesh. The principal topographic features of the area are flat high plains lying in the Piedmont plain. Altitutdes range from 253 feet near the northeast edge to approximately 160 feet near the southernmost end of the area. It slopes almost uniformly from the north in a general south and southeast direction. The land gradient ranges from three feet per mile in the north to two feet per mile or less in the south and southeast parts of the area.

The area is drained by two major rivers Karotoya and Tangan. Karotoya has its source in India and enters Dinajpur district at Bardesware. Tangan river originates in or near the border of the country and flows southward past Thakurgaon.

According to the Soil Survey of Bangladesh (1973), the project area is characterized by a good sandy soil which responds well to irrigation and fertilizers. The very favorable geology, abundant recharge possibilities, the high water-table, and present performance of the well field indicate that the ground-water basin will be enough to supply required irrigation water.

Problem Assessment

Concern has been expressed by the Agricultural Authority that the existing dug wells and hand pump wells used for domestic purposes may be dewatered as a result of excessive drawdown due to irrigation pumping in the area. It appears that about 10,500 domestic water supply dug and hand pump wells which are now being used by 0.4 million people in this area will be severely affected. It is therefore proposed that the problem could be averted if the irrigation includes a plan allowing withdrawal of water for irrigation at a maximum annual yield for 20 years from the basin. This would maintain the allowable declination of water table down to 22 feet depth from surface throughout the irrigation period. Tentatively, the average yield of each irrigation well has been established by the Agricultural Authority to be two cfs spread over 70 acres of land at the rate of twelve hours per day for 150 days (November to March).

Objectives

The objectives of this study are to utilize the available hydrogeologic data of the area and to determine the maximum annual yield and pumping rate allocation in order to protect shallow domestic wells by limiting drawdown throughout the irrigated area. The study includes:

1. Determination of the occurrence and availability of ground-water in the area;

2. Calculation of a preliminary hydrologic budget for the area with emphasis on the ground-water component

of the budget;

3. Adaptation, calibration, and application of a mathematical model that will adequately simulate the operation of the hydrogeologic systems;

4. Determination of safe drawdown limit to protect shallow wells;

5. Computer simulation runs to establish maximum annual yield and pumping rate allocation in order to limit regional drawdown caused by irrigation to a safe limit.

Previous Works

Some previous studies have been made in this area. These have covered all aspects of the development potential in agriculture, irrigation, and mineral abstractions. The United Nation Food and Agriculture Organization (1966) studied the area on the assumption that good aquifers are present as revealed by exploring borings and electric resistivity surveys and recommended a detailed field investigation prior to undertaking any project for a large scale ground-water development. Karim (1968) and Mowla (1968) worked on this project and they determined the aquifer properties from limited available aquifer test data and from geologic analysis. The results were mostly based on qualitative data.

Some geological investigations have been made concerning the mineral development and it is only lately that attention has been directed towards the sediment loss, hydrogeology, and amount of the ground-water storage. Some general regional geological descriptions have been made but the data are very sparse and the detail required for the mathematical modeling of the aquifer was not available. The Soil Survey of Bangladesh (1973) and Publications by Brammer (1969) and Huizing (1971) provide very comprehensive information on soil genesis and soil properties related to the potential for agriculture development. These studies have been used to make general statements about the geology of the country on the basis of regional soil characteristics.

Most of the previous surveys report that there is an adequate aquifer available for exploitation within the study area and that irrigation wells drilled to 300 feet would yield at least two cfs. Some reservations about recharge potential and water quality have been expressed by McDonald. The same consultant has asserted that shallower tubewells drilled to approximately 180 feet deep could function as efficiently as 300 feet deep wells using a standard design, provided they were individually designed for the aquifer available.

CHAPTER III

METHOD OF APPROACH

Method of Study

In order to determine the maximum annual yield and a corresponding pumping rate allocation for the aquifer during a 20-year period from 1977 to 1997, the author accomplished the following tasks. A flow chart showing those tasks is shown in Figure 3 and details of the approach used are discussed in the following chapter under the respective headings.

Task I. Collection of Raw Data

It includes (a) geologic investigation, (b) collection of well records, (c) precipitation records, (d) prior requirements of water for irrigation, and (e) hydrogeologic investigations.

Geological information was collected from Geological Survey of Bangladesh; publications include those by Coleman (1969), Morgan (1959), and Brammer (1969). Lithological characteristics were determined from Driller's Log. Bore logs were collected from Bangladesh Agricultural Development Corporation and Water Development

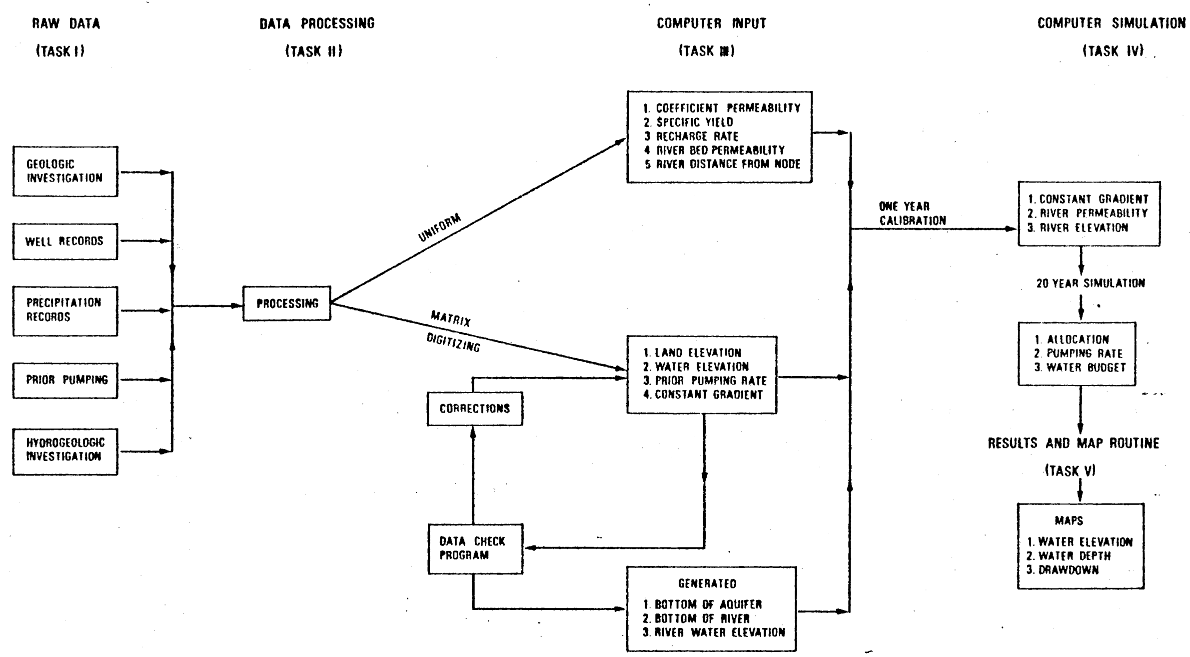


Figure 3. Flow Chart of Computer Modeling

Board. Aquifer materials are generally uniform, homogeneous, and unconsolidated Piedmont type deposits.

All of the records for the 266 existing irrigation wells were collected from Bangladesh Agricultural Development Corporation and the Water Development Board and were plotted on the maps. Precipitation records for the last ten years of five different stations in the study area were collected from the Meteorological Department. The average weighted annual rainfall in the study area is 86.11 inches.

Hydrogeological data such as long term aquifer testing data, well observation data, drawdown discharge of existing wells, and river discharge data were collected from the Water Development Board. The weekly water level survey which is published periodically by the Water Board consists of 26 observation wells in the study area.

Task II. Data Analysis and Processing

The collected raw data described in Task I were analyzed and processed for computer input. Contour maps were prepared and values were determined after processing the raw data. The details of these are discussed in the following chapters.

Task III. Data Digitizing and Use

The input data were divided into uniform matrix and nonuniform matrix parameters. Uniform matrix values were entered directly into the model.

The contoured data were gridded, digitized, and read into the IBM terminal for computer input. The 1:1.10 mile grid map shown in Figure 4 was drawn at the same scale as the base map and was overlaid onto each contour map. Values were assigned to each node of the grid. The grid spacing was used to establish a matrix.

The uniform matrix parameters included:

1. Recharge rates from precipitation and irrigation
2. Coefficient of permeability
3. Specific yield
4. Distance of river influence on node elevation
5. Coefficient of permeability of riverbed

Recharge was considered in two forms: precipitation and return flow from irrigation. Return flow from irrigation was estimated 20 percent of the total water pumped. The initial recharge rate was calculated to be 28% of precipitation. The permeability coefficient of the aquifer consisted of a homogeneous value and was calculated to be 600 gpd/ft^2. Similarly, the S_y of the upper part of the aquifer is estimated to be 20 percent. The riverbed consisted of silts and clays, and they

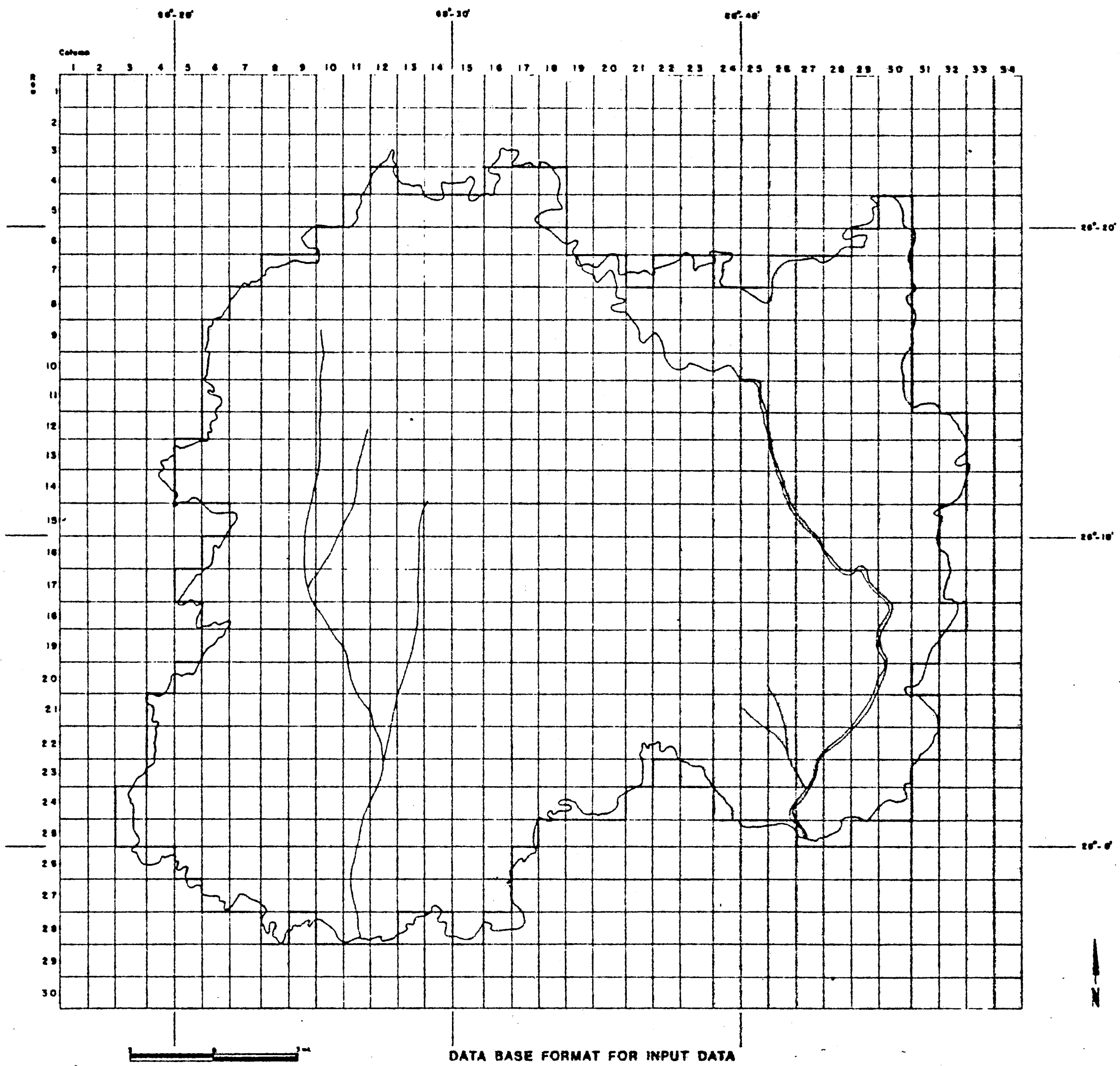


Figure 4. Data Base Format Map

served as a local aquitard to the underlying aquifer material.

The non-uniform matrix parameters included:

1. Water table elevations
2. Land elevations
3. Bottom elevations of the aquifer
4. Prior irrigation pumpage
5. Boundary subsurface flow gradient.

The water levels of existing 26 observation wells were obtained from Water Development Board and interpolated head values were used for the computer matrix. The land elevations were assigned to a grid by using the Survey of Bangladesh topographic map. The bottom of aquifer was considered as a function of the depth of well penetration and land elevations.

Prior irrigation pumping was determined by the Agriculture Department for use in the land irrigation at a rate for which a beneficial use can be shown. Prior pumping requirement rates (acre-ft/year) were acquired from the Agriculture Department converted to acre-ft per acre per year and assigned to nodes with respect to their one mile location (Figure 5). The pumping rate used from each irrigation well was two to three cfs or about four acre-feet per acre during 150 days irrigation period (November to March) at the rate of twelve hours continuous pumping per day. The well yields were determined by

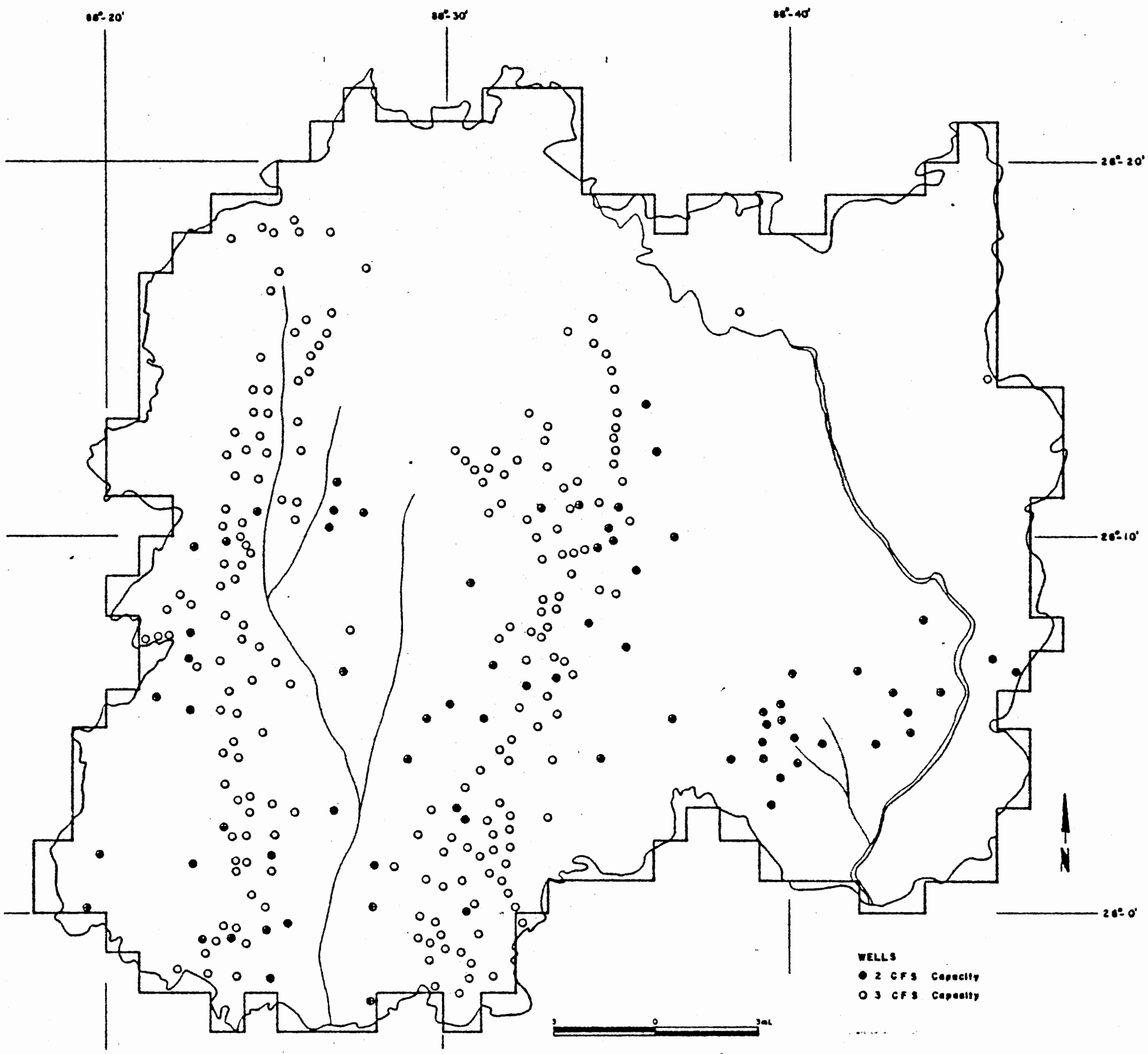


Figure 5. Well Location Map (1977)

dividing total withdrawal of water per year by the total number of wells pumped per year. The approximate range of discharge in thousand acre-feet per square mile at the end of 1977 are shown in Figure 6.

The 1977 water-table map of the study area was used to calculate the hydraulic gradients of recharge and discharge nodes located at the boundaries of the modeled aquifer. The fluxes at these boundaries were calculated using Darcy's law:

$$Q = KAI \quad (1)$$

where

Q = the amount of inflow or outflow (L^3/T)

K = hydraulic conductivity of aquifer material (L/T)

A = cross sectional area of the nodes through which flow takes place (L^2)

I = hydraulic head gradient or slope of water table (L/L)

The equation can be written as

$$Q = K(W \circ D)(\Delta H \div L) \quad (2)$$

where

D = saturated thickness of the adjacent node.

W = the cross sectional width of the node.

The distance between centers of nodes (L) are cancelled out when a square size node is used. The model calculates flux as the data entered was ΔH.

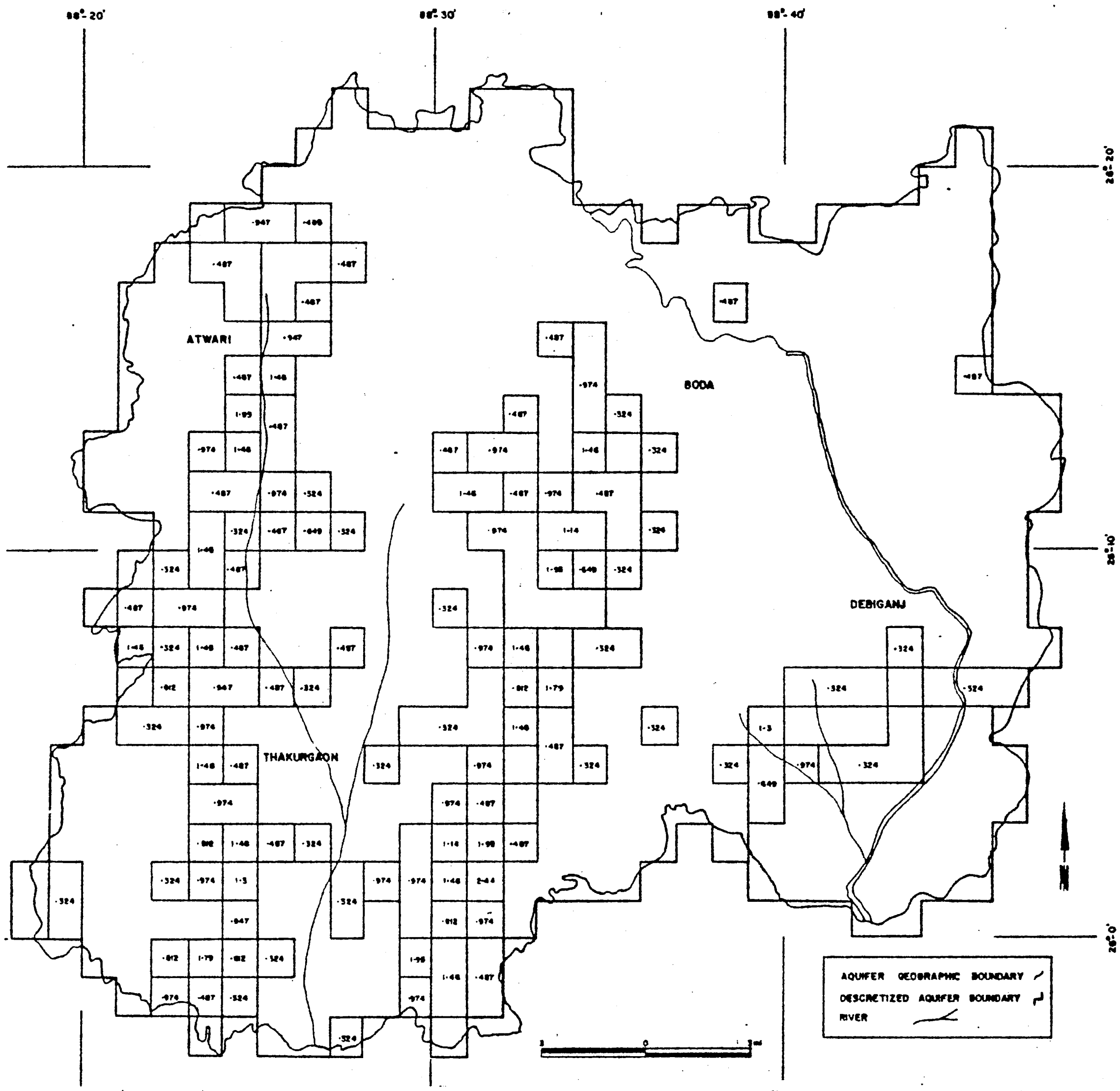


Figure 6. Map Showing Prior Pumping, in Thousands of Acre-feet per Square Mile (1977)

Task IV. Calibration

One year computer simulations were performed using the observed 1977 water head elevations. One year simulation runs were used to calibrate the model. Calibration was achieved by adjusting the river elevation and constant gradient.

Task V. Allocation

A 20-year computer simulation (1977 to 1997) was conducted using the results of calibration. Repeated runs were performed using prior irrigation pumping rate and various allocation pumping rates in order to prevent no more than 50 percent of the nodes from going below 22 feet. For comparison, an additional run was made using only the prior irrigation pumping rate.

Task VI. Model Output

Data output from these versions were plotted using the computer printer. The computer simulation results were summarized in the ground-water budget for the study area.

CHAPTER IV

GEOLOGY

Regional Geology

Bangladesh is generally covered by Quaternary sediments deposited by the Ganges-Brahmaputra Meghna river system. Most of the country is a vast alluvial plain of Recent age. Except for Tertiary hill ranges of Chittagong and Sylhet, and the uplifted Pleistocene Barind and Madhupur tracts and the Lalmai hills of Comilla.

Rocks ranging in age from Paleocene to Pleistocene are exposed in the eastern and northeastern parts of the country. The Tura sandstones of the Paleocene and early Eocene ages are the oldest rocks exposed (near Takerhat in Sylhet District) and represent the platform facies, whereas the folded rocks of the Chittagong area are exposed miogeosynclinal sediments of the Miocene Surma Group.

The western and nortwestern parts of the country are characterized by a stable shelf area in which rocks vary from Paleocene to Pleistocene in age and have been deposited on the eroded Archean Basement complex. The Basement complex consists mainly of crystalline metamorphic and igneous rocks which have been found in the bore holes at depths between 2030 and 7090 feet in the Bogra-Rajshahi district. Drilling at Madhyapara in the Dinajpur District has revealed the presence of quartz-diorite and granodiorite at very shallow depths between 422 and 506 feet below the surface. The age of this unit is uncertain.

Figure 7 represents the estimated thickness of the sedimentary deposits within Bangladesh. There is a major discontinuity which trend from the Shillong plateau to the soutwest towards Bogra. This zone is referred to as the "Hinge Line" which separates the shelf sediments of the northwest from the basin/geosynclinal sediments of the south and the east. The Sylhet basin is extremely deep and there is a strong gravimetric evidence of continuing subsidence. It can be noted in Figure 7 that minimum thicknesses, about 400 feet, occur in the Project area (Dinajpur District) but a very rapid increase to over 9000 feet in thickness occurs in the southern part (Pabna District). However, of these thicknesses only a small percentage of these thick sediments are economically exploitable aquifers which normally occur in the upper 300 feet.

The generalized stratigraphy of Bangladesh is summarized in Table I which includes a description of the water bearing properties.

Typical deep geological profiles for southern Dinajpur District are illustrated in Figure 8. In this

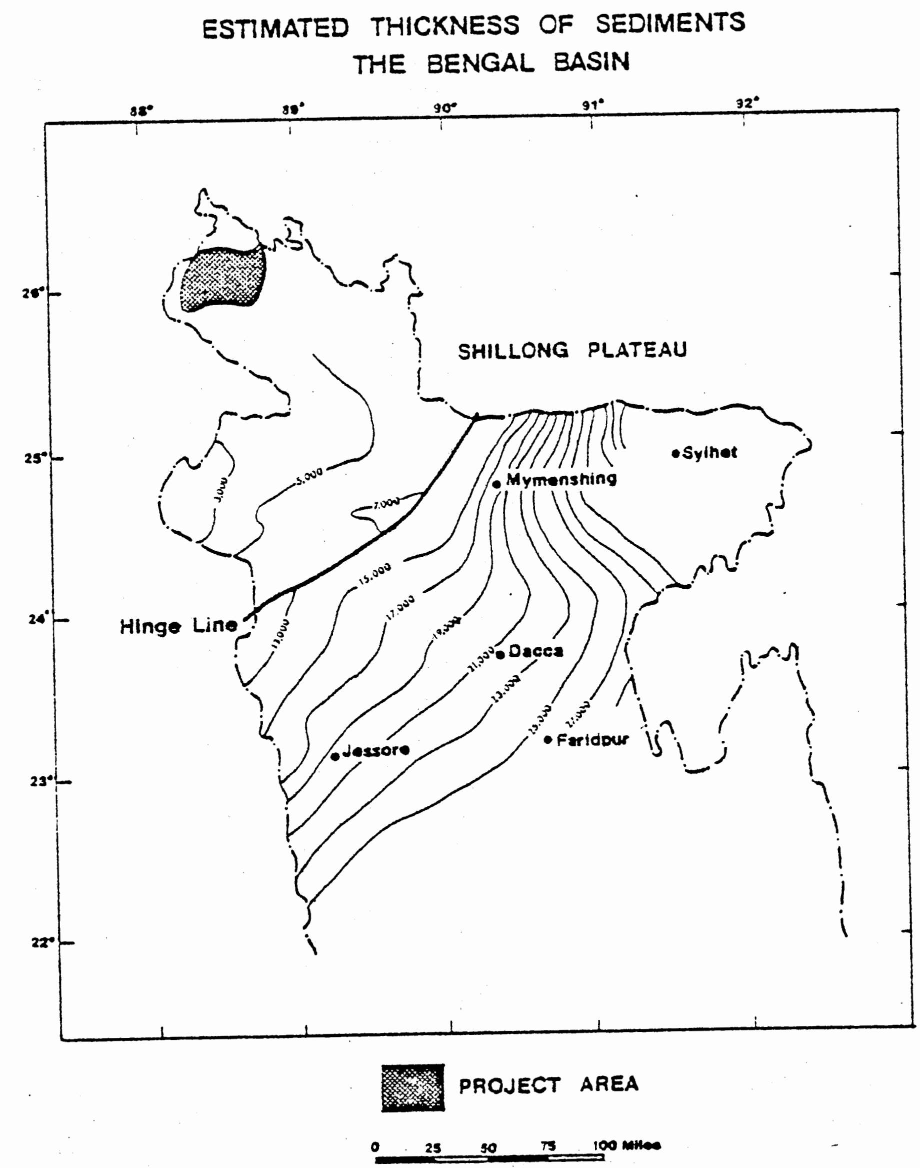


Figure 7. Map Showing Sediment Thickness in Bangladesh (After Brammer)

TABLE I

GENERALIZED STRATIGRAPHY AND AQUIFER UNITS IN BANGLADESH

Age	Geosynclinal Sediment South & East Bangladesh		Shelf Sediments Northwest Bangladesh		Physical Characteristics	
	Group	Formation	Group	Formation	Major Lithologic types	Water Bearing Properties
Pre Holocene Holocene		Alluvium		Alluvium (300)	Sand, Silt & clay	Escellent aquifer
unconformity						
Late Pliocene Pleistocene		Modhupur clay (151+)		Dihing (650)	Clay, Pebbly Sandstone	Aquiclude in Northwest
unconformity						
Late Miocene	Dupitila (6,000)	Upper Dupitila (over 5000)	Dupitila (1000)		Claystone, siltstone & gravel	Aquiclude, Aquitard
Middle-Pliocene		Lower Dupitila (2000)			Sandstone	Aquifer
unconformity						
Middle Miocene	Tipam (7132)	Giruhan clay (3188)	Surma (1355)	Jamal-gans (1355)	Claystone, siltstone & sandstone	Aquiclude
		Tipam Sandstone			Sandstone	Aquifer

TABLE I (Continued)

Early Miocene	Surma (19000)	Bakabil (6000)			Siltstone, shale, and sandstone	Aquiclude
		Bhuban (13000)			Siltstone, sandy shale & sandstone	Aquiclude/ Aquitard
			unconformity			
Oligocene	Barail	Jenam (2120+)	Barail (535)	Bogra (535)	Siltstone, Carbonaceous shale and sandstone shale	Aquiclude
Late Eocene				Kopill Alterna- tions (492+)	Shale, carbo- ceous fossi- lifarous with sandstone & limestone	Aquiclude
Middle Eocene			Jaintia (1091)	Sylhet Limestone (800)	Fossiliferous limestone w/ subordinate sandstone	Aquifer (mineralized)
Early Eocene				Tura sand- stone (790+)	Sandstone, coal & shale	Aquifer (mineralized)

TABLE I (Continued)

			unconformity			
Late-middle Cretaceous				Sibhani Trapwash	Shale, calcareous, sandstone, Ferrugenous shale, clay, sandstone	No aquifer
			unconformity			
Late Jurassic Early cretaceous				Rajmahal Trap (1790+)	Basalt flows	No aquifer
			Unconformity			
Late Permian			Raniganij	Paharpur (1443+)	Sandstone with thick coal seam	No aquifer
			unconformity			
Early Permian			Barakar	Kuchma (1601+)	Coarse grained sandstone w/ 170 thick coal reach	No aquifer
			unconformity			
Archean				Basement comples (393+)	Gneiss and Basalt	No aquifer

Note: Maximum thickness of rock shown in parenthesis (feet) section the width of the column is made proportional to grainsize.

The two boreholes sections as shown in Figure 8 are representative of deposits in most of the Dinajpur area although the age interpretation is exceedingly difficult because these formations are not fossiliferous. It is of interest to note the occurrence of kaolinite in both sections. This may indicate massive down faulting in the order of 600 feet between these boreholes following a time when this zone was weathered from the exposed basement diorites which were uniformly deposited over the area. The lowest sedimentary units classified are the Permian Gondwana deposits which were deposited in a lacustrine environment. The coarse conglomerates are possibly indicative of the marginal glaciofluvial deposition. Subsequent upfaulting to the southeast may have caused the removal of this Unit from the section GDH26 (Figure 8).

Above the kaolinized zone are sequences of shallow siltstone and muddy sandstones, provisionally classified as the Tura formation of Eocene Age by Geological Survey of Bangladesh. The upper part of this formation becomes increasingly coarse grained between 80 and 150 feet below the Pleistocene clay deposits. There is fairly strong evidence that the upper deposits may be unconformable with Tura and considerably younger Dupi Tila of Mio-Pliocene Age or even Pleistocene interglacial deposits.

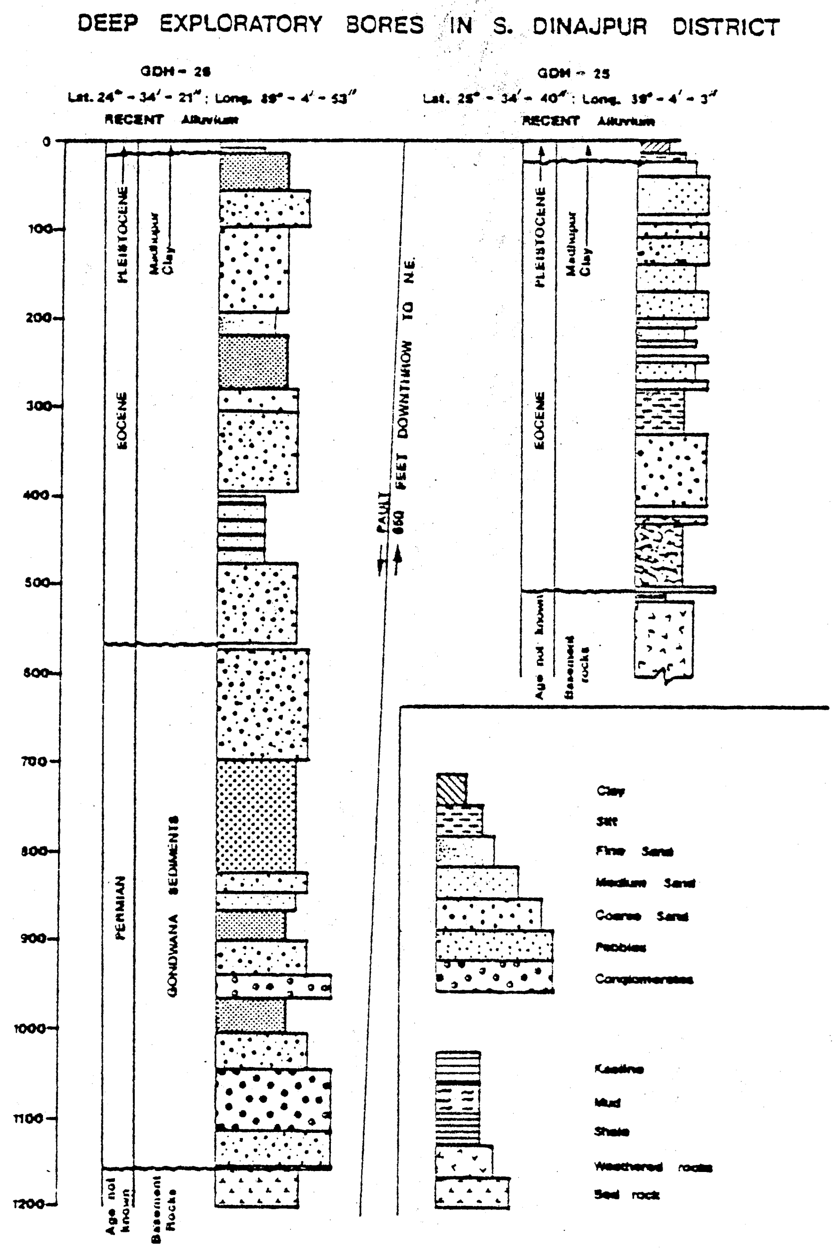


Figure 8. Geologic Profiles from Deep Bore Holes (After Sir MacDonald)

The Pliocene-Pleistocene Barind and Modhupur clay rests unconformably on the underlying sedimentary rocks. The impervious clay beds are 20 to 100 feet in thickness and are predominantly gray in color but are reddish in outcrop areas. These clay beds form the base of the unconfined aquifer. Covering these earlier deposits (Pleistocene clay) are Sub-Recent to Recent alluvium composed of grey and yellow sands, gravels, clay, and silty soils. These mark the areas of Recent and present fluvial activity. These more recent deposits constitute the aquifer which is modeled in this study.

The tectonic history of the region is complex and difficult to interpret because of the thicknesses of masking alluvium. Bangladesh geology is strongly affected by the fact of being squeezed between the Himalayan thrust and older Arakan anticline flanking the Burmese geosyncline. This bidirectional compression has generated much secondary folding and faulting over and above that normally found following the east-west Himalayan trend. The net result has been that many shear faults associated with the compression and vertical faults associated with the folding have remained active throughout the period of deposition. It is even claimed by some that fault throws can be detected at depth in the alluvial cross section.

Nevertheless, there can be little doubt that tectonics are responsible for the lateral separation of the Barind and Piedmont units and that faulting is probably responsible for many other observed lateral changes in aquifer characteristics as shown in Figure 9 which show the tectonic features and structures of Bangladesh.

Local Geology

The plain area of Bangladesh has been divided geomorphologically into seven units by Coleman (1969) and Brammer (1969) as shown in Figure 10. Generally the features used in the geomorphological classification are land gradients, erosional maturity, degree of weathering, flooding, drainage, and type of soil developed.

The two units, the Piedmont and Karotoya flood plains, are found in the Project area. The Piedmont represents the coarser grained, sandy deposits along the foot hills of the Himalayas. These areas have fairly well developed and drained soil profiles. In the northeastern part of the Project area, the Karotoya river debouches from the mountains and has modified the older Piedmont area, forming a well marked tract of recent flood plain alluvial deposits which trend north-south across the area. Both are of recent age but are flanked on the south by Pleistocene Barind alluvium. Nowhere is there evidence of the younger sediments being deposited on top of the older sediments. Within the depth limits of water well drilling, the change is always lateral, so that tectonic

MAJOR TECTONIC FEATURES AND STRUCTURE OF BANGLADESH

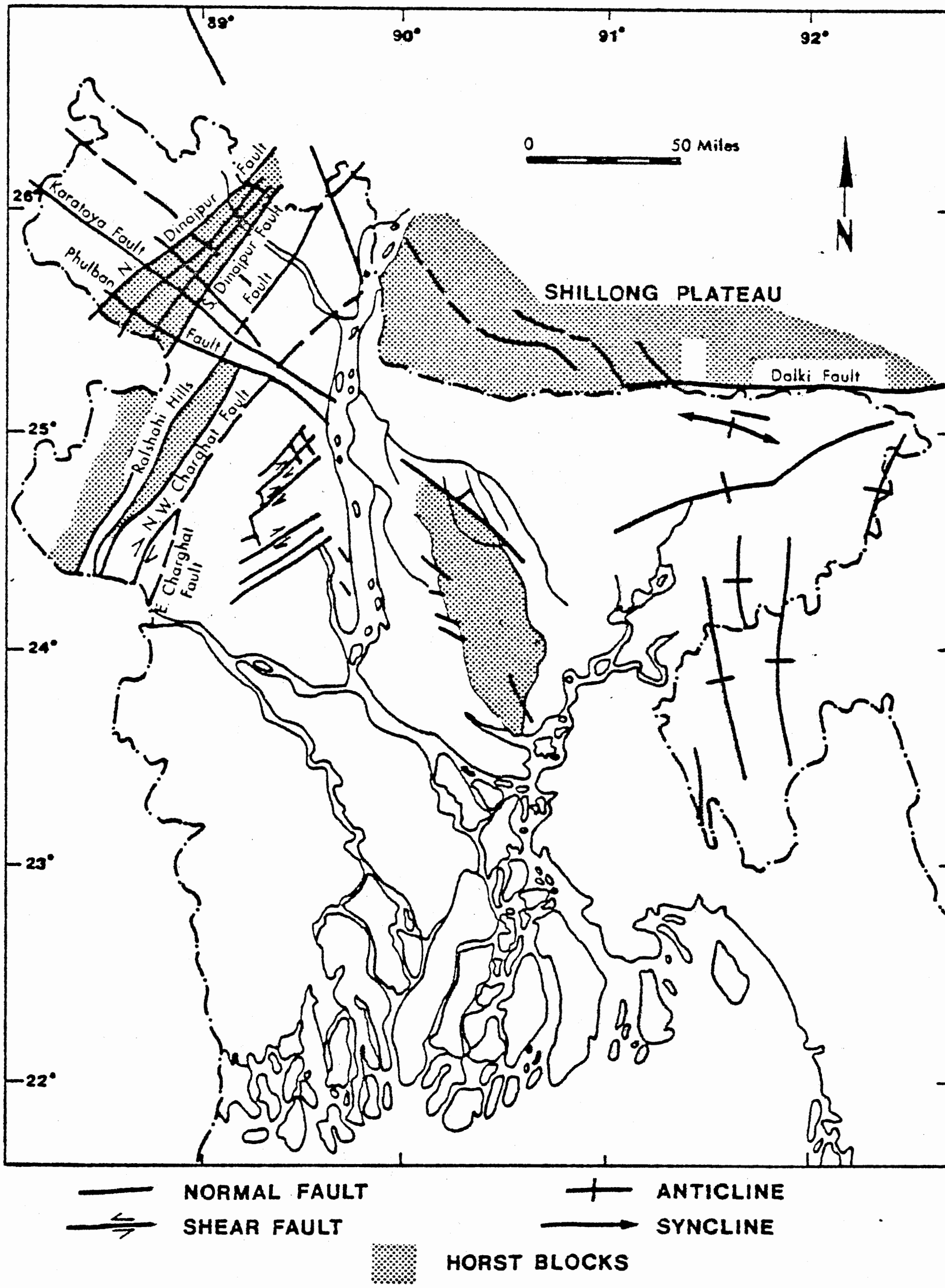


Figure 9. Tectonic Features and Structure Map (After Sir MacDonald)

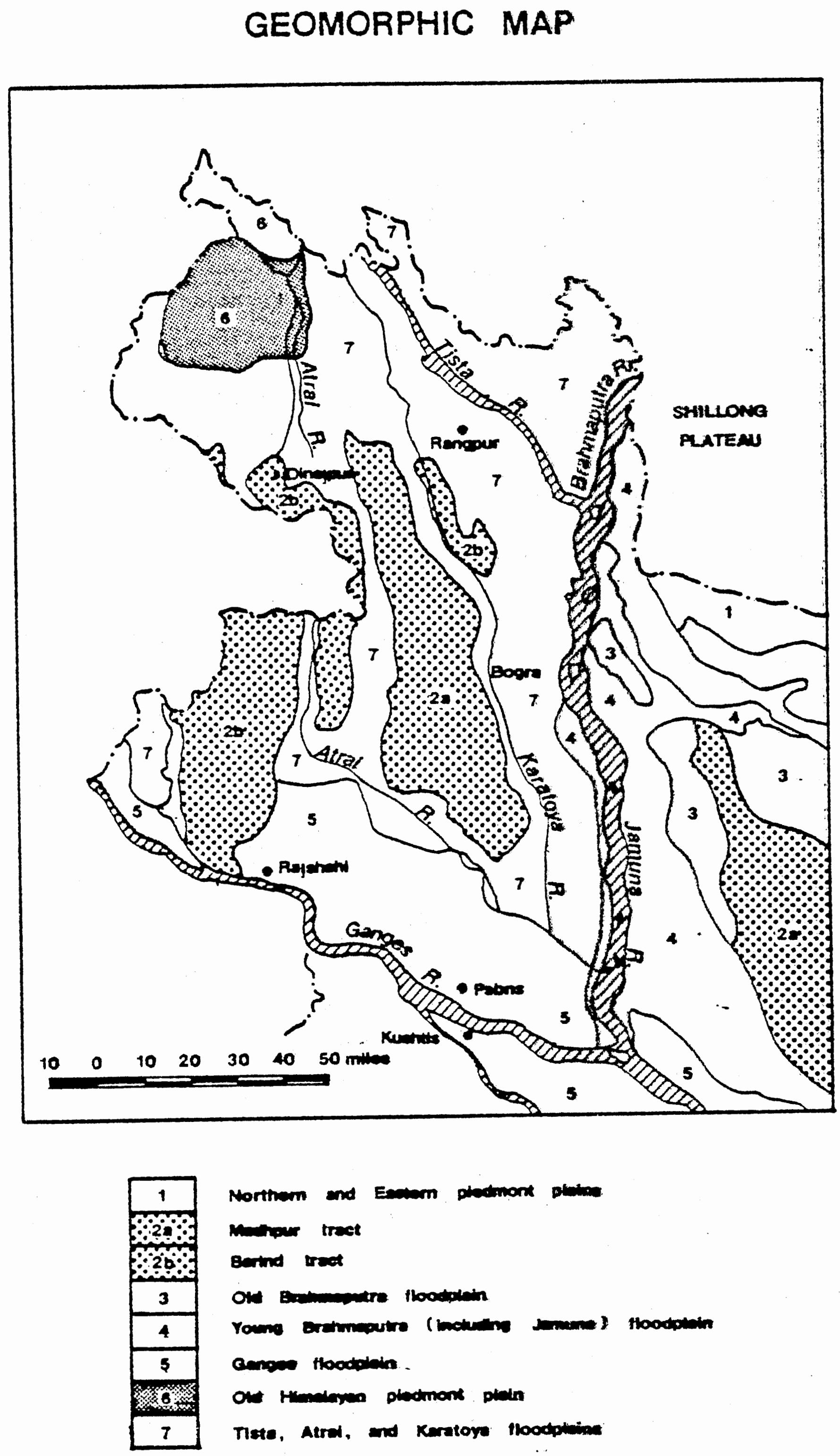


Figure 10. Geomorphologic Map of North of Bangladesh (After Sir MacDonald)

movements must have been effective in determining the boundaries.

It is agreed by many that the Karotoya represents a unit which is composed of a Piedmont material reworked by a river system; therefore the deposits can be named as Piedmont deposits. A north-south lithologic cross section as shown in Figure 11 represents a unit in the Project area which consists of mainly uniformly deposited medium to coarse sands with fine sands with less than ten percent clays and occasional gravels. The gravels increase in frequency and in grain size from south to north. Silt sizes seem to be completely absent which indicate intermittent outwash deposition rather than large, permanent river systems. The absence of silt is diagnostic of the unit.

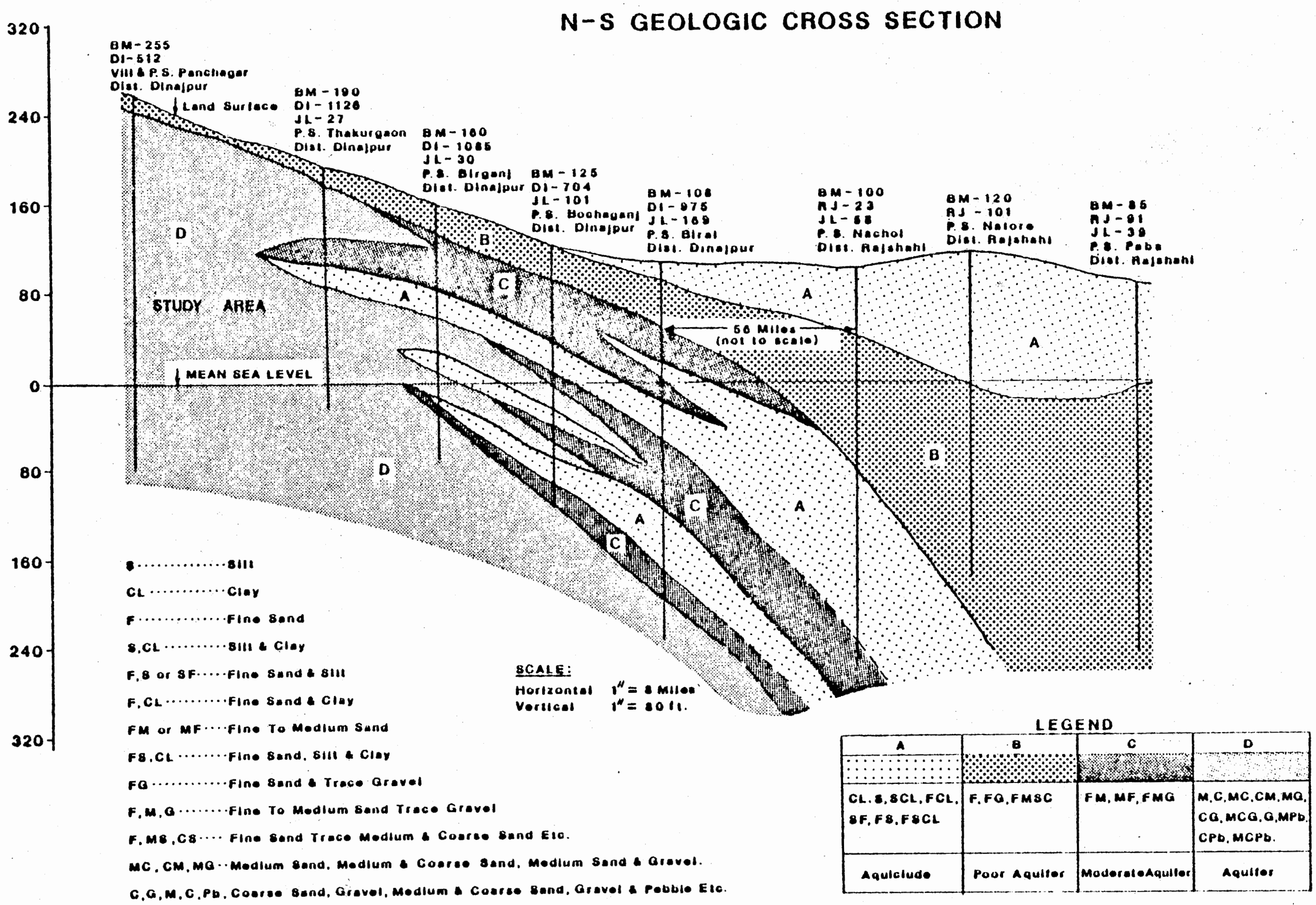


Figure 11. North-South Geologic Cross Section of Dinajpur and Rajshahi

CHAPTER V

HYDROGEOLOGY

General

Ground-water is the major, if not the sole, source of water for irrigation and domestic uses in the study area. The Piedmont aquifer is an unconfined aquifer. The base of this alluvial aquifer is impermeable in nature with finer materials preventing a downward loss of water. Therefore, rechargeable ground-water is restricted to the overlying alluvium. The aquifer appears to be a homogeneous north-south trending sand bounded by multilayered finer textured sediments (Figure 11).

Water-Table

The observed minimum and maximum depths to the water-table in the study area for the year 1977 are shown on maps in Figures 12 and 13. The minimum depth is associated with the July-August period which is during the monsoon; the maximum depths occur during April and May which is dry season. Data represented on the maps are collected from 26 number observation wells in the study area. The difference between the highest and the lowest

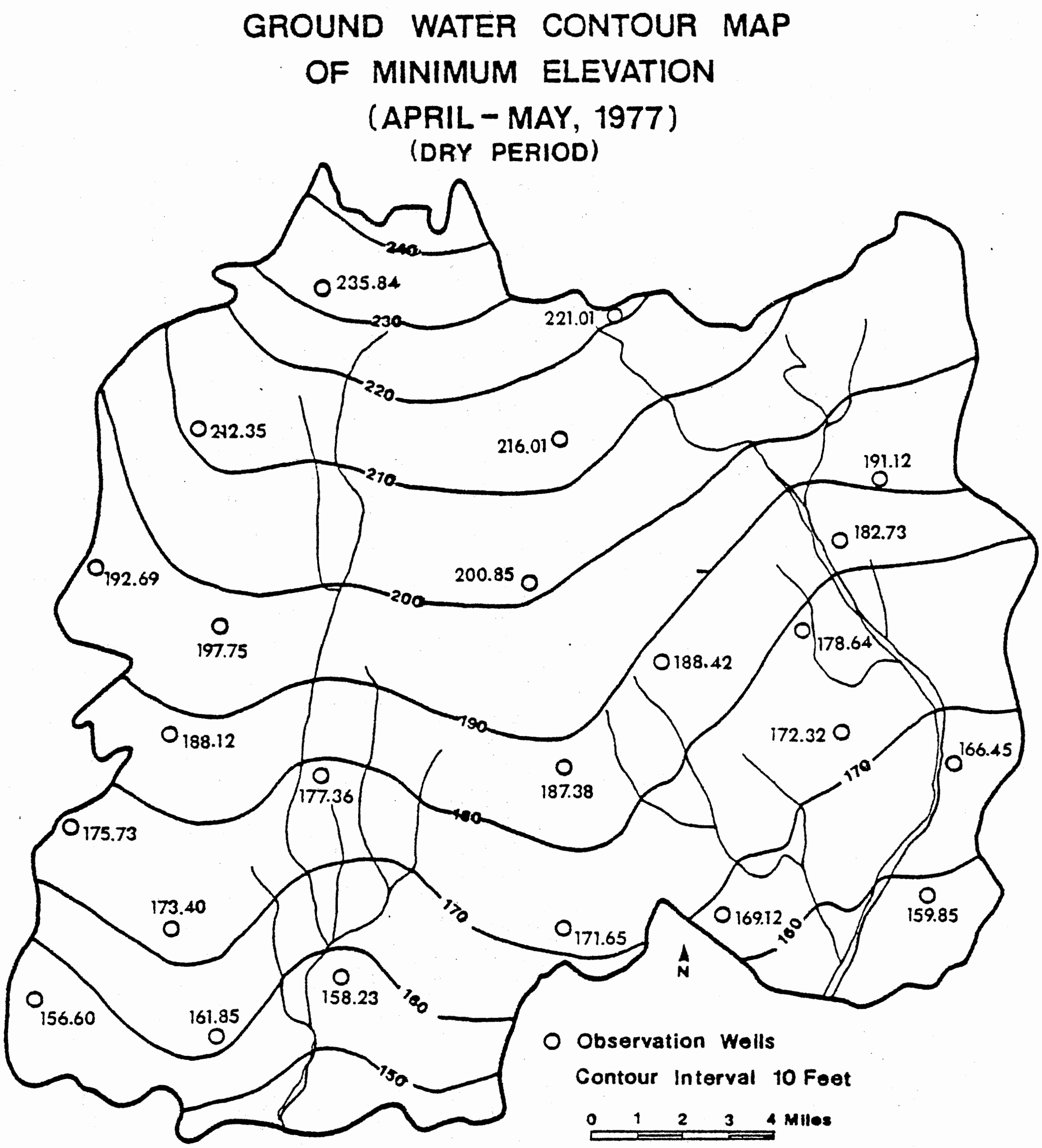


Figure 12. Minimum Water-table Elevation Contour Map (1977)

GROUND WATER CONTOUR MAP OF
MAXIMUM ELEVATION
(AUG-SEPT, 1977)
(WET PERIOD)

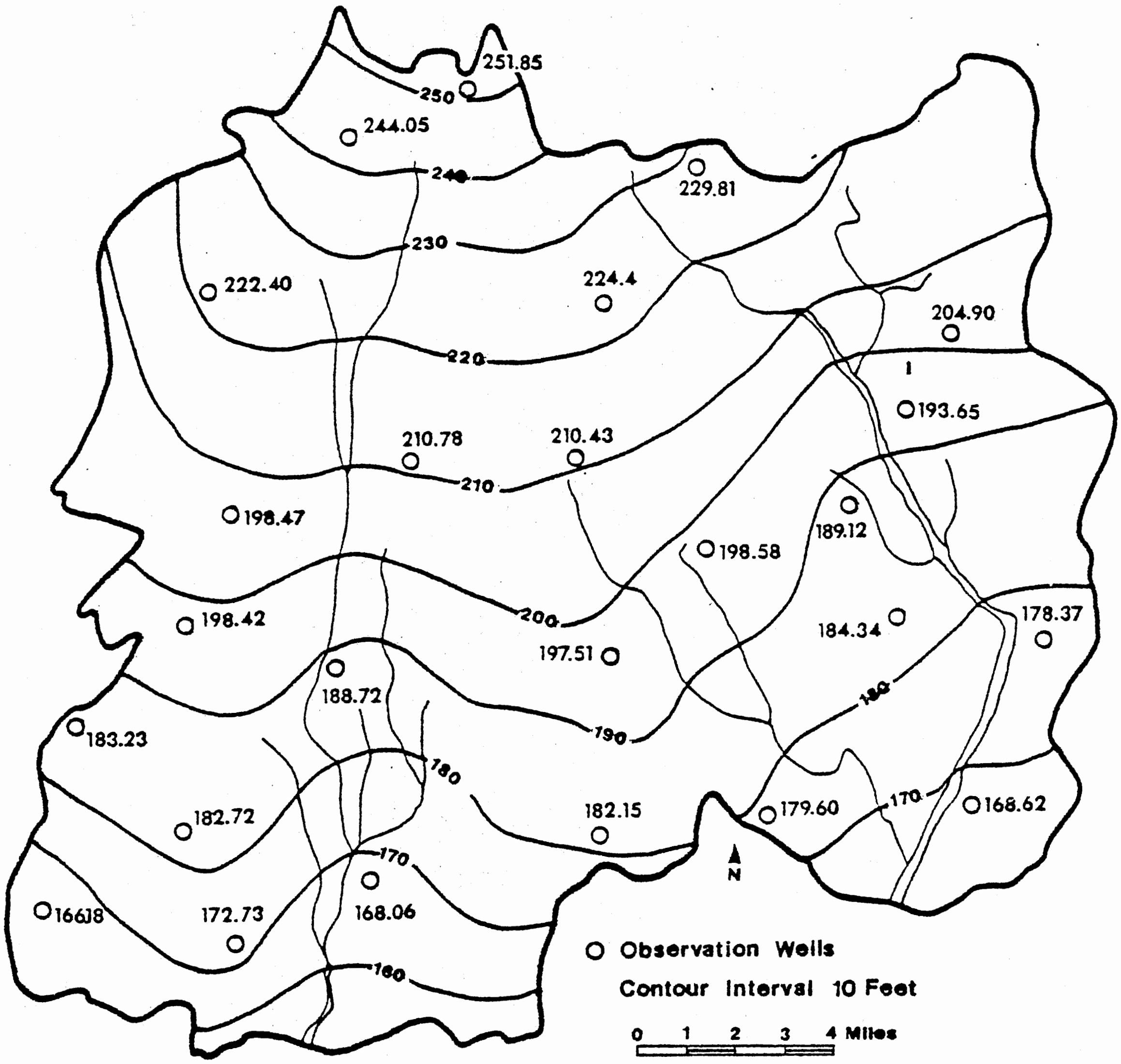


Figure 13. Maximum Water-table Elevation Contour Map (1977)

water levels ranges from eight to twelve feet in most wells. The water-table generally follows the topography of the area (Figure 14) and slopes generally to the south. The water-table contours also bend upstream, indicating that groundwater is discharging into the stream. A relatively uniform hydraulic conductivity (coefficient of permeability) gives rise to equidistant spacing of water-table contours. The average hydraulic gradient in the study area is approximately three feet per mile.

Climate

The study area experiences a tropical climate characterized by a very humid, wet, south-west monsoon from May to October. This accounts for some 85 percent of the annual rainfall. During the period between November and April, a cool, dry northeast monsoon blows from Central Asia bringing, initially, the lowest temperatures and humidities and later on, convectional storms. During the cool period, temperatures may vary from a minimum of 39°F to a maximum of 94°F and in the wet period temperatures vary from 54°F to 107°F.

The average of monthly precipitation and evapotranspiration between 1968 and 1977 for the same station in the study area are listed in Table II. The average annual precipitation of different stations in the study area is shown in Table III. The average annual precipi-

SURFACE ELEVATION CONTOUR MAP

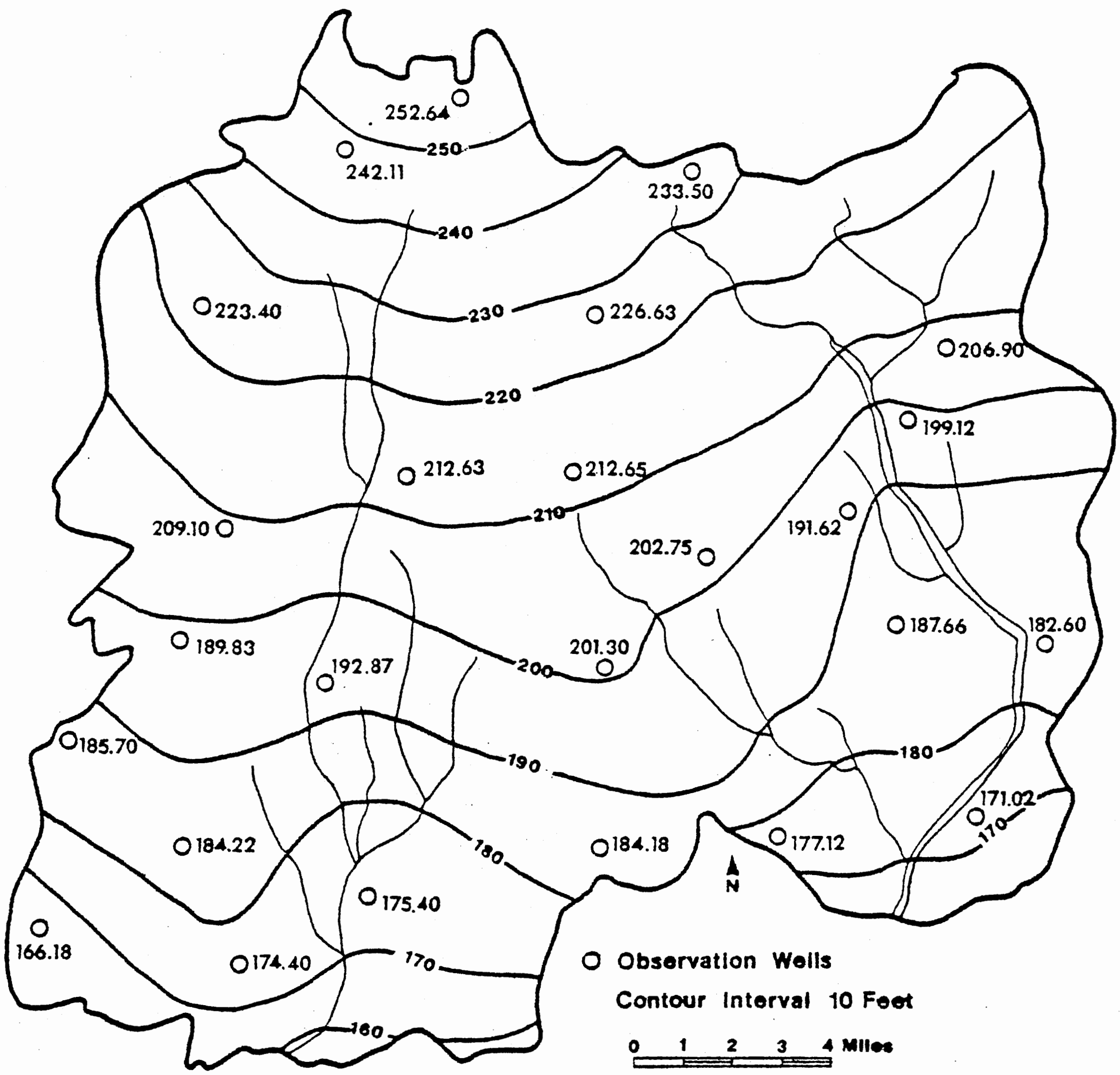


Figure 14. Surface Elevation Contour Map

TABLE II

AVERAGE MONTHLY PRECIPITATION AND EVAPOTRANSPIRATION (INCHES) (1968 - 1977)

Month	Average Precipitation (Inches)	Average Evapotranspiration (Inches)
January	0.40	2.08
February	0.44	2.79
March	0.62	4.48
April	2.85	5.90
May	10.33	6.10
June	15.96	5.31
July	20.82	5.11
August	15.52	5.0
September	14.48	4.80
October	4.50	4.21
November	0.145	2.80
December	0.045	2.08
Total	86.11	50.67

TABLE III

WEIGHTED AVERAGE PRECIPITATION

Station	Average Precipitation (in/yr)	Area (Sq. mi.)	Percentage of area	Weighted Average (in/yr)
Chotadap	89.12	76	13	11.58
Pachagarh	78.57	53	9	7.07
Buda	90.95	130	22	20.00
Debigan	88.14	100	17	14.98
Thakurgaon	83.25	229	39	32.46
Total		588	100%	86.11

tation is 86.11 inches.

Ground-water Recharge

The term natural recharge refers to water that percolates down through the unsaturated zone to the water-table and actually enters the dynamic ground-water flow system (Freeze, 1967). This definition excludes that portion of the soil moisture surplus that enters the ground and increases the soil moisture but does not enter the flow pattern itself. The rate of ground-water recharge depends on several factors including precipitation, permeability, depth to water-table, soil type, moisture conditions, topography, vegetation type and density and temperature.

Because of several variables and difficulties involved in attempting to measure ground-water recharge rates, experiments along these lines have been relatively few and exceedingly complex. Nonetheless, recharge rates can be of considerable importance in the design of well fields because the recharge that can be captured by the cone of depression surrounding a pumping well is equal or nearly equal to the long term yield that the well can provide without drawing from storage (W. A. Pettyjohn and Henning, 1979).

Recharge to the ground-water reservoir in the study area occurs mainly by percolation of rain water. The conditions are favourable for recharge during June to October. There is much precipitation in excess of evaporation and transpiration in the study area during this time. Duration of rainfall also varies from a few hours to a few days and is widely distributed. During these periods, the soil seldom becomes dry and little water is required to replenish soil moisture. Therefore the rainfall of the monsoon season is very favourable for recharge. The top soil is comprised of silt and fine sand which permits a good infiltration rate to occur. Practically the entire area is farmed which increases the soil permeability by loosening up the top soil. Puddling of the soil by rain drops is reduced by the vegetal cover which further lowers runoff, and enchances percolation. The universal practice of subdividing rice fields by the construction of low dikes increases infiltration since the dikes store two to four inches of water.

Figures 15 and 16 are well hydrographs which show that the water level in the observation wells rises during the rainy season and recedes during the dry season, reaching a maximum depth by the end of the dry period. During the months of May to July, the water level rise in the observation well is rapid. After that, the water levels in the wells slowly rise until the months of August and September when the water levels reach the minimum depth. The average fluctuation of ground-water level is ten feet.

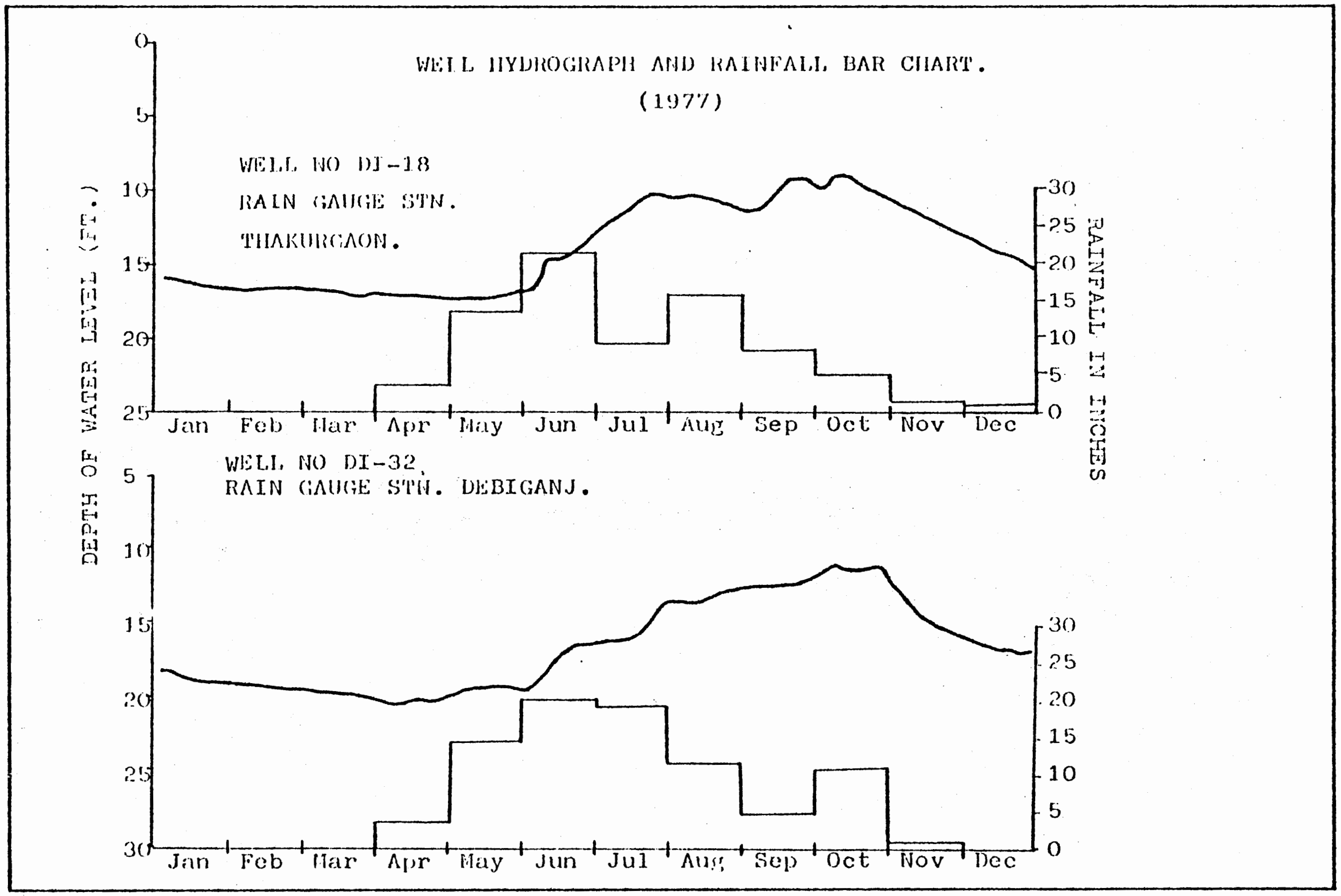


Figure 15. 1977 Well Hydrograph and Rainfall Bar Chart (W-18,32)

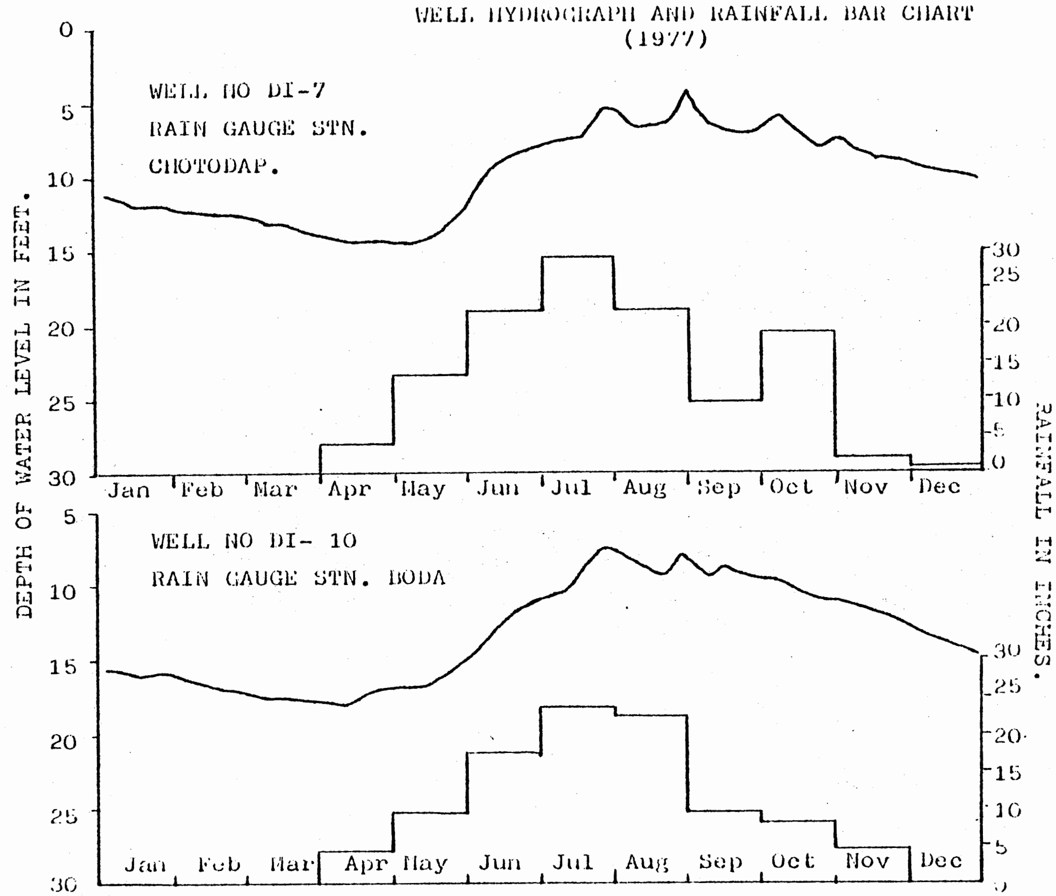


Figure 16. 1977 Well Hydrograph and Rainfall Bar Chart (W-7,10)

During the wet season ground-water is still continuously discharging in a small amount to the river (see Figure 13).

The minimum depth to the water-table as shown in Figure 13 is only two to three feet below the land surface over most of the area during the months of July and August. During this period, recharge by percolation of rainwater could not occur because the water table lies very close to land surface and is subject to evapotranspiration and runoff.

There are several methods by which estimates of regional ground-water recharge can be made. Three widely used methods will be summarized for the purpose of evaluating the accuracy of the results as well as some constraints inherent in each.

1. Actual Field Measurement

The volume of material which has been recharged or dewatered is calculated using the following equations:

$$\text{GW storage} = A.\Delta h.S_y \qquad (3)$$

where

A = surface recharge area

Δh = water-table fluctuation

S_y = specific yield

The recharge area is 376,640 acres. The average water-table fluctuation is ten feet and is derived from maximum and minimum depths to the water-table using the 1977 contour map. The specific yield for the upper twenty-five feet of the aquifer was estimated to be twenty percent based on lithologic characteristics. Therefore the annual ground-water storage will be

$$376{,}640 \text{ acre} \times 10 \text{ ft} \times .20 = 753{,}280 \text{ acre-ft}$$
$$\approx 753{,}000 \text{ acre-ft.}$$

When the storage is spreaded over the study area (376,640 acres) the ground-water storage will be two feet of water (753,280÷376,640 = 2). Estimates of natural recharge based on actual field measurement such as these will provide more realistic results than any of the other methods.

2. From Hydrologic Budget

The hydrologic budget is an accounting of all water that enters or leaves the area. The formulation of a hydrologic budget is essential for optimum development of water resource of an area. An expression for the hydrologic budget is given below (modified from Todd, 1959 and Kent, 1973).

$$\left[\begin{array}{l}\text{precipitation + surface}\\ \text{inflow + subsurface inflow}\\ \text{+ imported water + decrease}\\ \text{in ground-water storage +}\\ \text{decrease in surface storage}\end{array}\right] = \left[\begin{array}{l}\text{evapotranspiration + sur-}\\ \text{face outflow + subsurface}\\ \text{outflow + increase in}\\ \text{ground-water storage +}\\ \text{increase in surface stor-}\\ \text{age + exported water}\end{array}\right] \quad (4)$$

The difference between sub-surface inflow and sub-surface outflow in the study area is assumed to be negligible. There is no imported and exported water and the change of the surface storage from May to August is also assumed negligible. Therefore, the equation for the hydrologic budget for the periods can be written as follows:

$$\begin{bmatrix} \text{precipitation} \\ + \text{ surface inflow} \end{bmatrix} = \begin{bmatrix} \text{evapotranspiration +} \\ \text{surface outflow +} \\ \text{change of ground-} \\ \text{water storage} \end{bmatrix} \quad (5)$$

The records of river flow from discharge station (Figure 2) ground-water levels, rainfall, and evaporation have been analyzed to study the hydrologic budget of the area. The total May to August inflow and outflow of the two rivers Karotoya and Tangan, is given in Table IV for the year 1977.

TABLE IV

TOTAL SURFACE INFLOW AND OUTFLOW OF RIVERS IN THE STUDY AREA

Streams	Discharge Station No.	May to August	
		Inflow, Acre - feet	Outflow, Acre - feet
Karotoya	140	894,000	
	142		1,736,000
Tangan	285		265,000
Total		894,000	2,001,000

The average precipitation for the period May through August for the year 1977 has been computed from the daily records of the rainfall of five stations in the area (Figure 2). The pan evaporation data for the study area in the months of July and August are adjusted to be equivalent to evapotranspiration for these periods. The evapotranspiration for the months of May and June has been estimated using the Thornthwaite method (Thornthwaite, 1948). The total surface inflow and outflow, rainfall, evapotranspiration, and the change of ground-water level for the periods May through August are shown in Table V.

TABLE V

SUMMARY CHART OF THE SURFACE INFLOW AND OUTFLOW, PRECIPITATION, EVAPOTRANSPIRATION, AND THE CHANGE OF GROUNDWATER LEVEL FOR THE MONTHS MAY TO AUGUST, 1977

Total surface inflow as of Table IV		Total surface outflow as of Table IV		Precipitation	Evapotranspiration
Acre-ft	Inches	Acre-ft	Inches	Inches	Inches
894,000	28.48	2,000,000	63.75	69.75	14.42

Note: Study area for the hydrologic budget = 376,640 acres

The change of ground-water storage is computed by substituting the values in Table VI to Equation (6) as follows:

$$\text{GW storage} = \text{precipitation} + \text{surface inflow} - \text{evapotranspiration} - \text{surface outflow} \quad (7)$$

$$= 69.76 + 28.48 - 14.42 - 63.75$$

$$= 20.07 \text{ inches} \approx 20 \text{ inches}$$

using the computed change in ground-water storage, the ground-water recharge can be calculated from the following equation by Schicht and Walton (1962):

$$P_g = R_g + ET_g + U + \Delta S_g \quad (8)$$

where

P_g = Ground-water recharge

R_g = Groundwater component (baseflow) to river flow

ET_g = Groundwater evapotranspiration

U = Subsurface underflow

ΔS_g = Change in ground-water storage

The groundwater component (baseflow) to the river flows was adjusted using the separation method for river flow hydrographs. Estimated baseflow is 3.95 inches over the 3.77×10^5 acres study area comprising the river reach.

The component of evaporation (ET_g) from the ground-water table is included in the estimate of evapotranspiration. The change of subsurface inflow and outflow (U) is assumed to be negligible. Therefore by substituting the values of the change of ground-water storage and the

[illegible]

ground-water component to streams to Equation (8), ground-water recharge of the area is calculated for the period May through August, 1977. Equation (8) can be rewritten as follows:

$$P_g = R_g + \Delta S_g \tag{9}$$

$$= 3.95 + 20.07 = 24 \text{ inches}$$

3. Baseflow Recession

Baseflow represents withdrawal of ground-water from aquifer storage after ground-water recharge has ceased. If the logarithm of river discharge is plotted against time, a linear relationship between the two variables may be obtained for the recession period after peak discharge has occurred. A "best fit" line can be used for the ground-water recession portion of the river hydrograph. Hydrographs for two successive years, 1976-78 are shown in Figures 17, 18, 19, and 20 for the Tangan and Karotoya Rivers. The recession curve which represents baseflow, has been separated from the other runoff components by a linear "best fit" line. The total potential ground-water discharge into the river Q_{tp}, is the amount indicated by the recession line (Mayboom, 1961). The value of Q_{tp} is found from the following relation:

$$Q_{tp} = \frac{Q_0 t_1}{2.3} \tag{10}$$

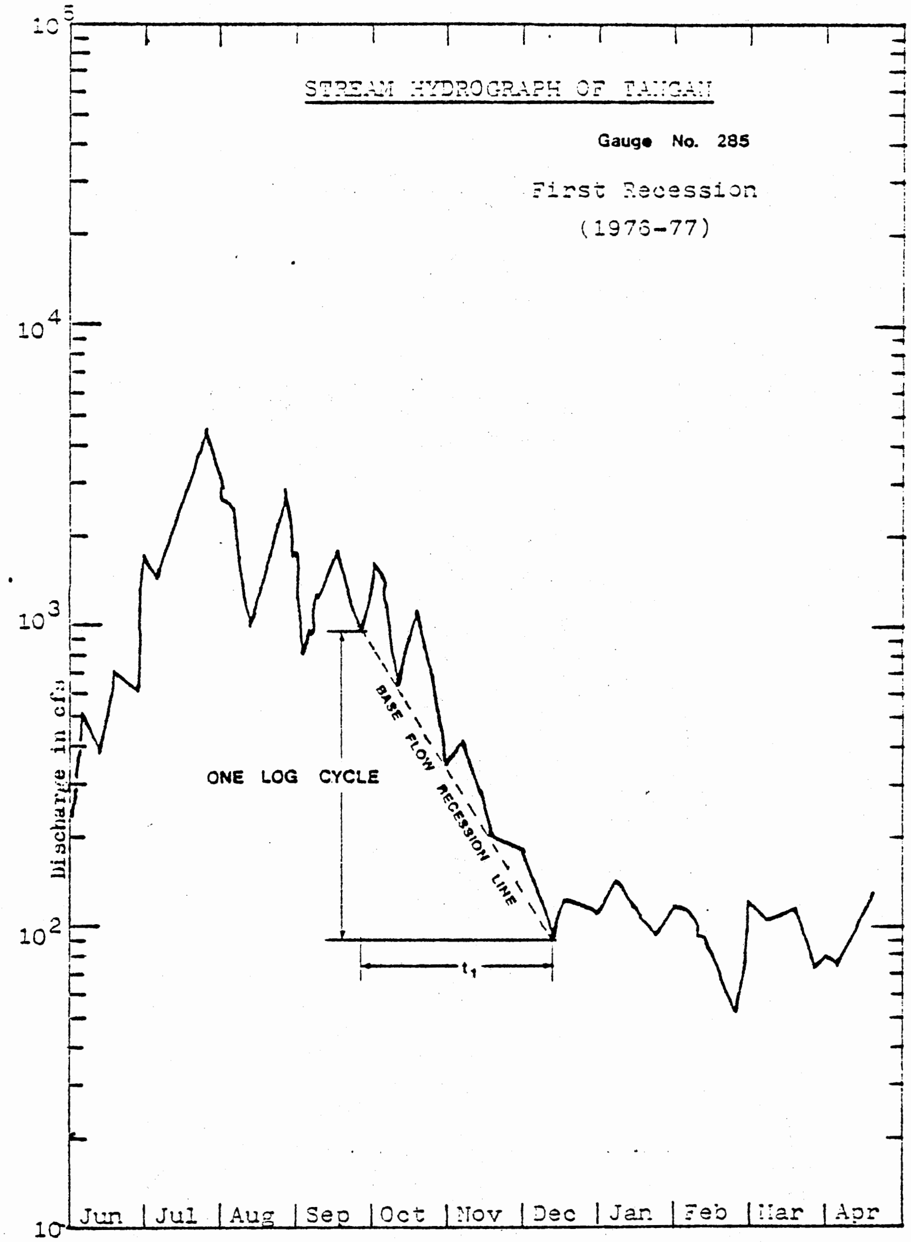


Figure 17. Stream Hydrograph of Tangan (1976-77)

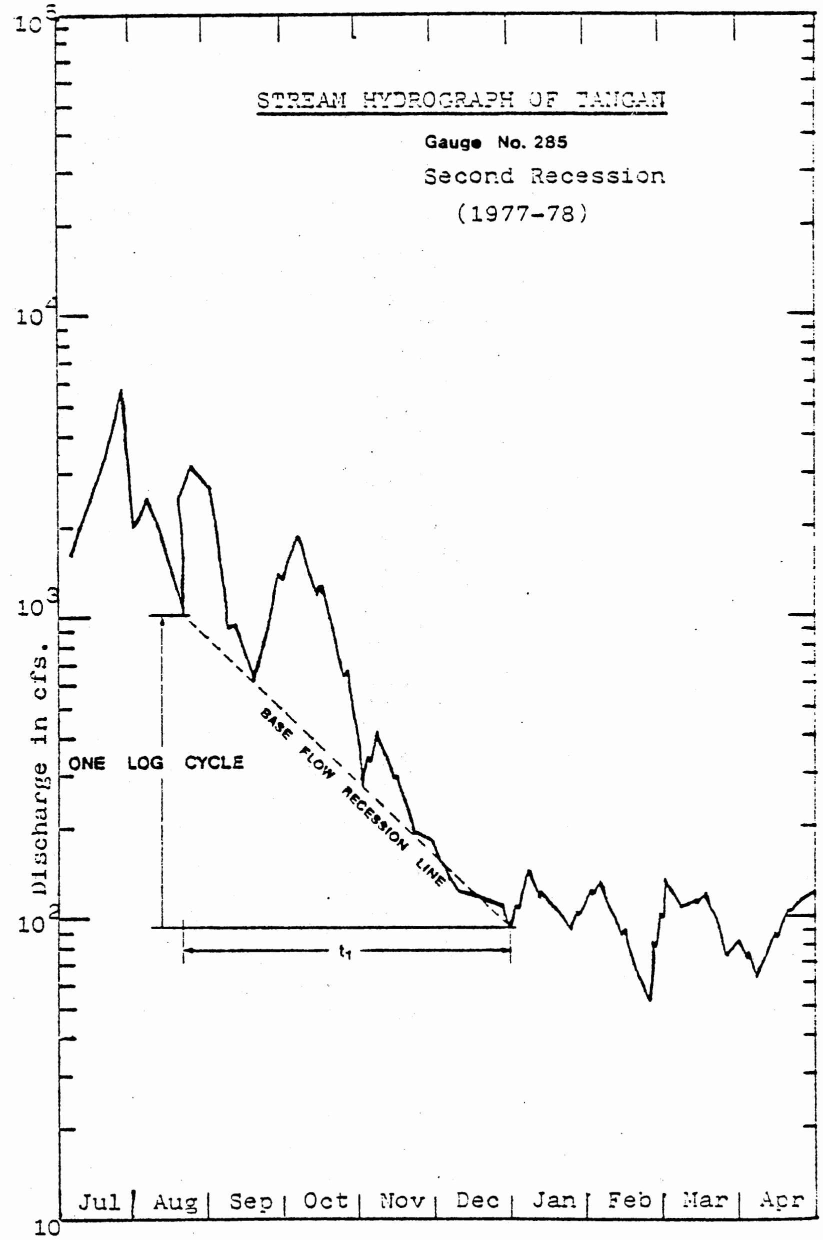


Figure 18. Stream Hydrograph of Tangan (1977-78)

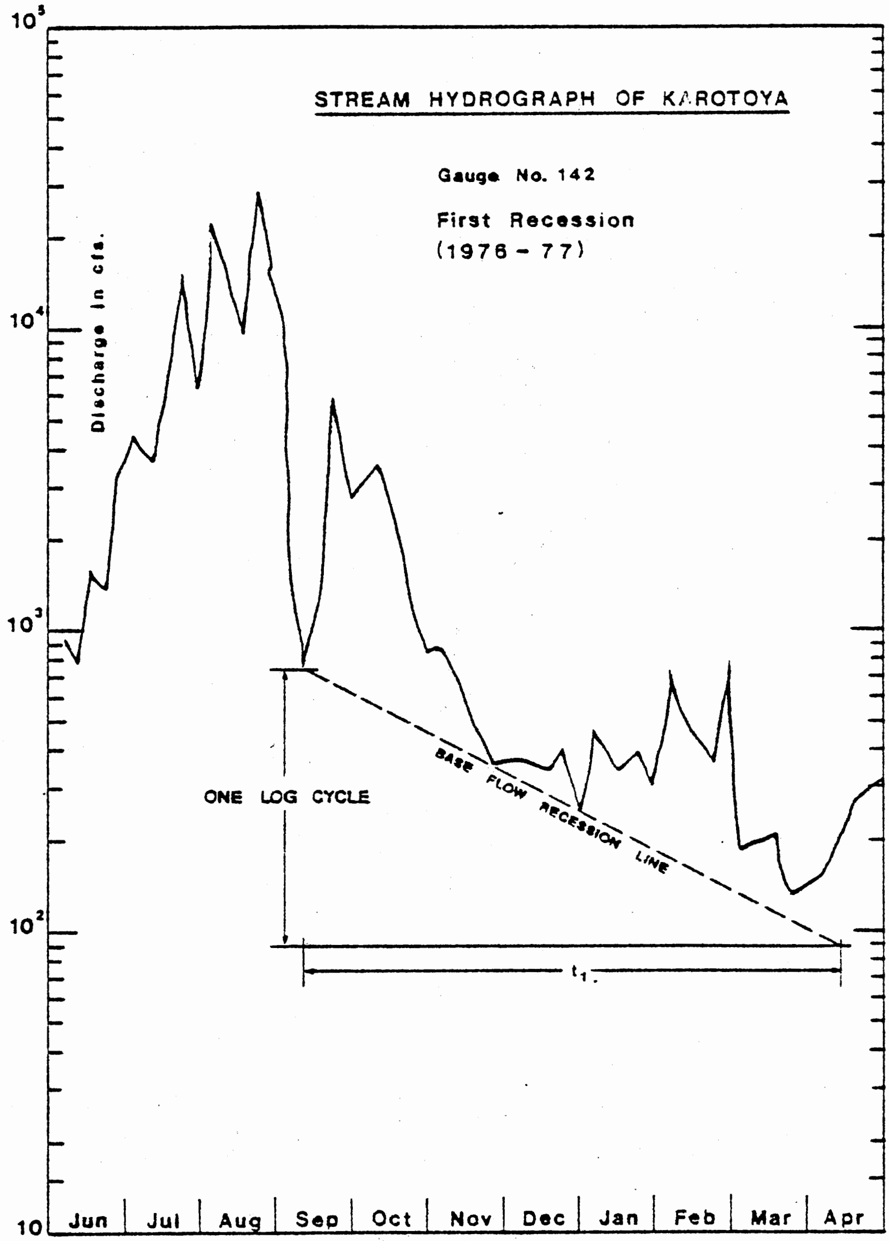


Figure 19. Stream Hydrograph of Karotoya (1976-77)

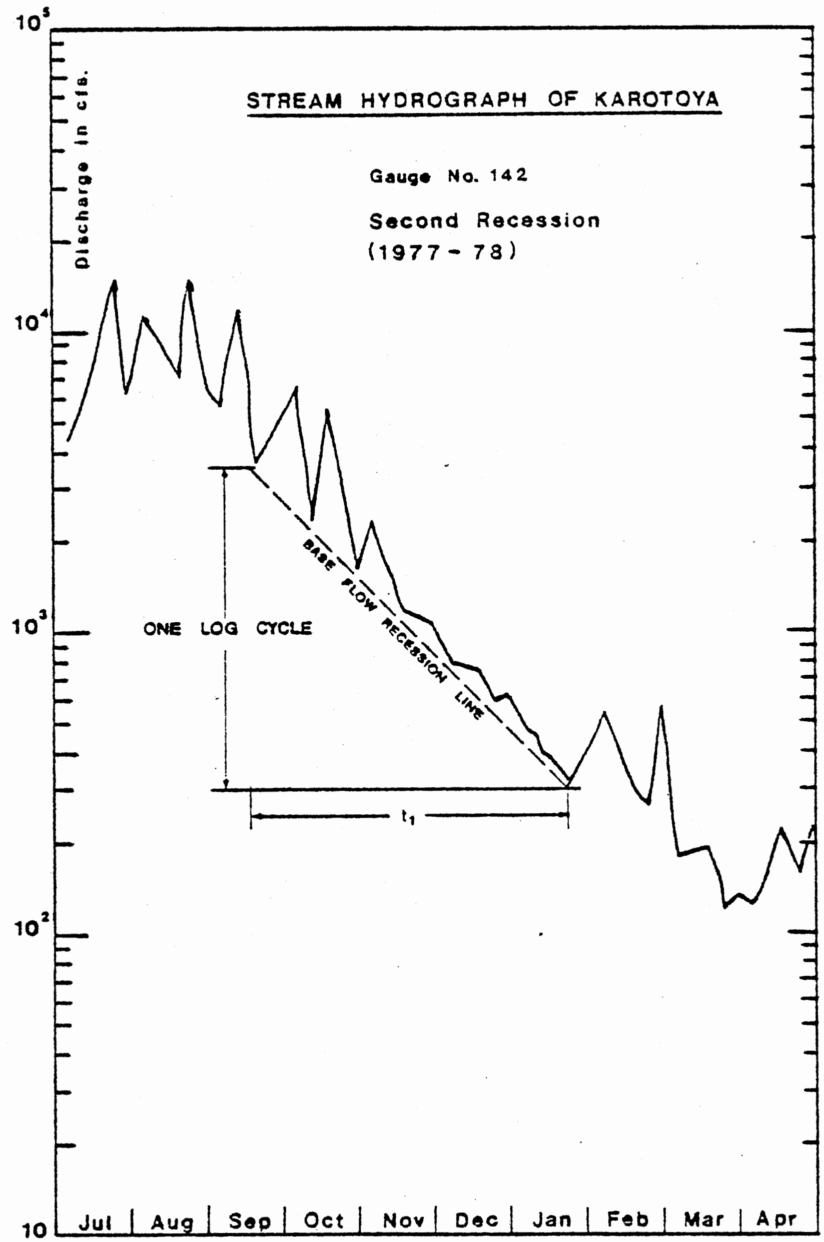


Figure 20. Stream Hydrograph of Karotoya (1977-78)

where

Q_0 is the baseflow at the start of the recession

t_1 is the time it takes for the baseflow to go from Q_0 to $0.1Q_0$.

The remaining potential ground-water discharge Q_t can be found from

$$Q_t = \frac{Q_{tp}}{10^{(t/t_1)}} \tag{11}$$

The amount of potential baseflow, Q_t, remains for a period t, after the start of a baseflow recession. The difference between the total potential ground-water discharge at the beginning of a baseflow recession period and the remaining potential ground-water discharge at the end of preceding baseflow recession period is a measure of ground-water recharge.

Ground-water discharge during the first recession period of the Tangan River hydrograph (Figure 17) is computed to be 900 ft^3/sec, Q_0, and it takes approximately three months for the discharge to reach $0.1Q_0$ (one log cycle). The total potential discharge is calculated from Equation (10).

$$Q_{tp} = \frac{Q_0 \; t_1}{2.3}$$

$$= (900 \text{ ft}^3/\text{sec} \times 3 \text{ months} \times 30 \text{ days/month} \times 86{,}400 \text{ sec/day})/2.3$$

or $$Q_{tp} = 3.04 \times 10^9 \text{ ft}^3$$

The value of Q_t at the end of recession which lasts for 7.5 months, is

$$Q_t = \frac{Q_{tp}}{10^{(t/t_1)}} = \frac{3.04 \times 10^9}{10^{(7.5/3)}} = 9.61 \times 10^6 \text{ ft}^3$$

The value of Q_0 or next year's recession (2nd recession) period as shown in Figure 18 is 1100 ft^3/sec, and t_1 is again 4.5 months. Therefore,

$$Q_{tp} = \frac{1100 \text{ ft}^3/\text{sec} \times 4.5 \times 30 \times 86{,}400}{2.3}$$

$$= 5.58 \times 10^9 \text{ ft}^3$$

The amount of recharge is equal to the total potential baseflow remaining at the end of the first baseflow recession subtracted from Q_{tp} for the beginning of the next recession:

$$\text{Recharge} = 5.58 \times 10^9 - 3.04 \times 10^6$$

$$= 5.57 \times 10^9 \text{ ft}^3$$

Similarly, the recharge, calculated from the baseflow recessions of the Karotoya River hydrograph in Figures 19 and 20, is 2.05×10^{10} ft^3. Therefore, the total recharge on the basin is

$$5.57 \times 10^9 + 2.05 \times 10^{10} = 2.60 \times 10^{10} \text{ ft}^3$$

This quantity represents approximately 19 inches of recharge in the 1977 year over the 588 square miles aquifer area. The result is comparable to results calculated by the former two methods. By comparing the results of the three methods as shown in Table VI, 24 inches of recharge can be considered as a representative annual ground-water recharge of the study area. Therefore, the recharge from deep percolation of precipitation is estimated to be 27.87 percent of total weighted annual rainfall of 86.11 inches (24 x 100/86.11 = 27.87%).

It is already stated as well as noted on the well hydrographs that the recharge may not occur during the months of August through September because the water table lies very close to the land surface. The recharge of the ground-water reservoir could be increased if the ground-water table was lowered by an additional ten to fifteen feet due to the withdrawal of ground-water in the dry season. Accordingly, the ground-water table would rise again to the same elevation in the months of September and October during normal years of rainfall.

Coefficient of Permeability

The hydraulic properties of the aquifer are needed as input to the model. This can be obtained accurately by using aquifer tests. Accuracy of results depends largely on how accurate and purposeful the pumping test data are and how closely the real aquifer resembles the ideal mathematical model on which all the equations of flow to wells depend. Many equations of flow to a pumping well are available in the literature and all of these equations

TABLE VI

COMPARABLE RESULTS OF ANNUAL RECHARGE
FROM DIFFERENT METHODS

Methods	Results (in/yr)
1. Field meaurements	24
2. Hydrologic budget	24
3. Baseflow recession from river discharge hydrograph	19

Note: 24 inches/year recharge was considered in the model.

are based on ideal mathematical models. But in nature, it is rarely found that an aquifer satisfies all the conditions of an ideal mathematical model, and moreover, sometimes data are inadequate and are not being collected in a planned way to fit them in a certain mathematical model. The selection of any given flow system model to represent the field conditions is, therefore, a matter of judgment left to the analyst, who must be aware of the geohydrologic boundaries of the aquifer in order to obtain accurate results as far as practicable.

In the study area, 266 wells have been drilled for irrigation, but practically no systematic and purposeful aquifer test was done in order to determine aquifer characteristics more precisely. Moreover, the thickness of the aquifer is not yet ascertained which complicates the situation still further. An attempt was made to analyze the data in several ways using several mathematical solutions which approximate field conditions.

1. Theim Method of Determining "T"

The transmissivity can be determined from the discharge and drawdown of the pumped well by the Theim equation assuming steady state was reached. Seventy representative wells of the 266 irrigation wells were selected in the entire study area to determine transmissivity. Johnson and others (1966) present the Theim equation as follows:

$$T = \frac{527.7\ Q}{S_w} \text{Log} \frac{r_e}{r_w} \tag{12}$$

where

Q = discharge in gallons per minute

T = transmissivity in gallons per day per ft

r_e = radius of influence which is assumed 1000 ft

r_w = well effective radius (0.83 ft)

$S_w = S_{wp} - CQ^2$

where S_w is the drawdown without loss

S_{wp} = observed drawdown (ft) in pumping well

C = constant for entrance loss which is estimated for average value of 0.5

Q = discharge in cfs

The values of S_w for ten selected wells are listed as examples in Column 6 of Table VII. Finally, Equation (12) has been reduced to

$$T = 1625 \times \frac{Q}{S_w} \tag{13}$$

where

Q/S_w is the specific capacity of the well

From this relationship, "T" and "K" have been computed for all 70 wells, 10 of which are shown in Table VII. The permeability "K" has been computed by dividing the "T" values calculated for all 70 wells with the saturated thickness of the well. The average "K" values range between

TABLE VII

HYDRAULIC PROPERTIES OF THE AQUIFER
COMPUTED FROM THE DEVELOPMENT
TEST DATA OF THE WELLS

Well No.	Aquifer thickness, (ft)	Static water level in well before pumping from GL (ft)	Drawdown (ft)	Computed drawdown (ft)	Depth to pumping level, (ft)	Drawdown per cfs, (ft)	Specific capacity, US gal/ft/minute	Transmissivity, US gal/ft/day	Coeff. of permeability $\frac{gpd}{ft^2}$	Remarks
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
HAK-E-28	250	8.33	16.46	14.66	27.82	5.49	91.81	149,000	596	Discharge
HAK-E-284	210	10.33	17.09	15.29	29.42	5.70	88.10	143,000	681	of
H - 74	180	7.60	19.79	17.99	27.08	6.60	74.87	122,000	678	each well
H - 76	275	5.67	25.74	13.94	32.67	8.58	96.62	157,000	571	is 3 cfs
H - 67	200	7.60	18.54	16.74	25.54	6.18	80.46	131,000	655	
H - 81	280	6.00	14.63	12.83	20.92	4.88	104.98	179,000	639	
H - 33	200	6.00	17.47	15.67	23.83	5.82	85.96	140,000	700	
H - 34	295	6.25	19.64	12.84	26.50	6.54	104.90	170,000	576	
H - 138	180	7.58	21.83	19.03	29.50	7.28	70.78	123,000	683	
H - 200	250	5.25	15.58	13.78	18.92	5.19	90.75	147,000	588	

500 and 680 gal/day/ft^2.

It is already stated that due to limited available data within the study area, the coefficient of permeability and transmissivity for this entire area could not be furnished directly. Therefore, another method was used to generate the coefficient of permeability and to distribute these values over the entire study area. Information related to thickness and lithology of the Piedmont deposit were obtained from drillers log in the study area. A number of Driller's Logs have been analyzed and frequency of occurrence of various types of materials is shown in Table VIII. The estimate based on the average percentage of occurrence for each class of material within a stated depth interval. The lithology was divided into 4 ranges: range one was associated with silt and clay; range two was very fine to fine sand; range three was medium to coarse sand; and range four was associated with coarse sand and gravel. A weighted average permeability was introduced by multiplying a weighting factor for the four grain size ranges by the percentage of the total saturated thickness which each range represents and finally is summed. The permeability values used for each range was obtained from the coefficient of permeability-grain size envelope developed by Kent et al. (1973) as shown in Figure 21. The ranges were chosen to represent various median grain sizes which correspond to the average

TABLE VIII

FREQUENCIES OF OCCURRENCES OF AQUIFER MATERIAL

	Material Percentage					
	Range 1		Range 2	Range 3	Range 4	
Depth Interval (ft)	Clay Percent	Silt Percent	Fine Sand Percent	Medium to Coarse Percent	Gravel Percent	
10- 20	40	0	30	25	2	
20- 40	16	0	28	55	1	
40- 70	16	0	17	61	6	
70-100	10	0	15	68	7	
100-130	7	0	12	74	7	
130-160	8	0	10	78	4	
160-180	13	0	19	68	0	
180-250	16	0	15	69	0	

Note: Percentages are averages based on data from 70 well logs.

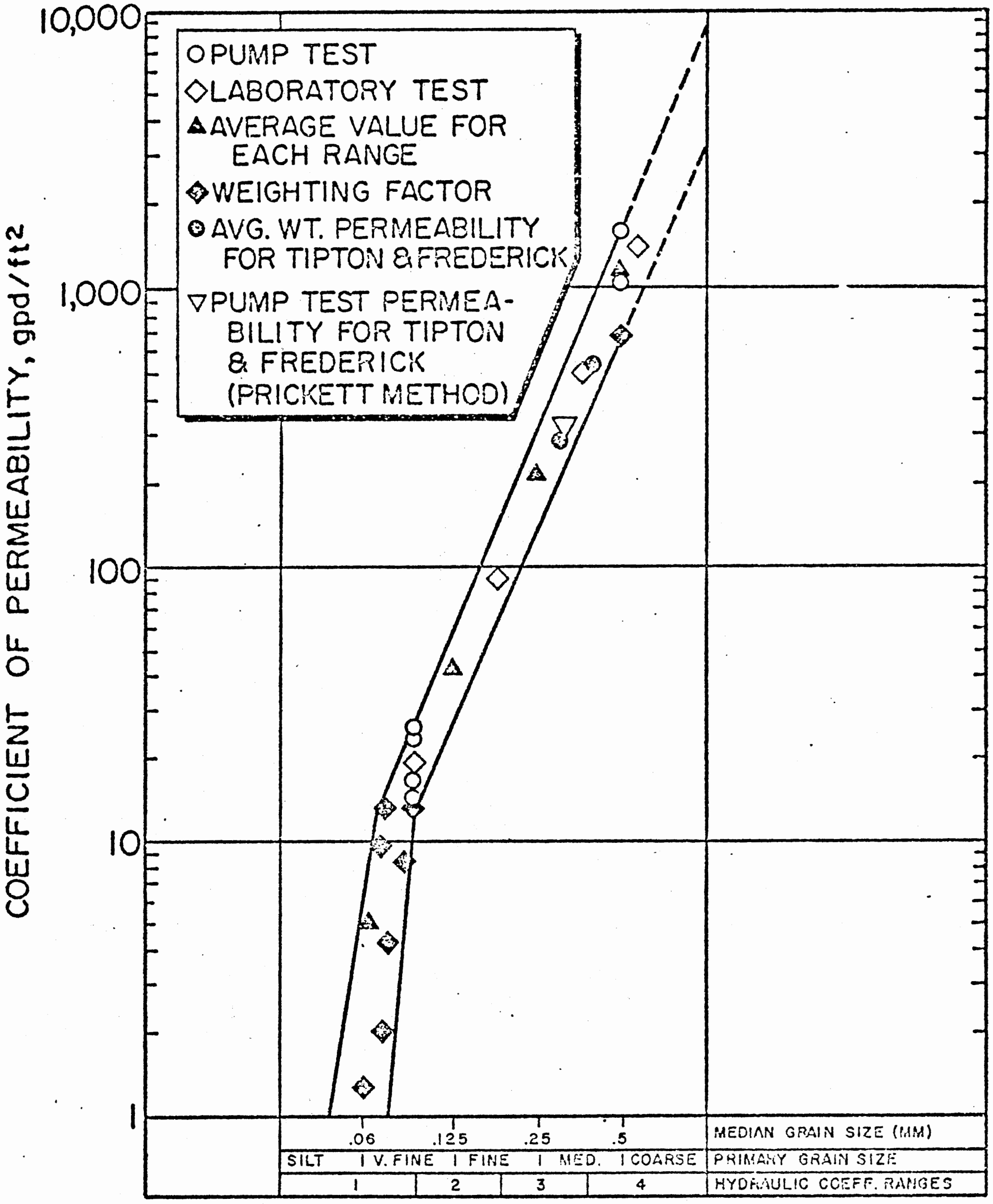


Figure 21. Coefficient of Permeability Vs. Grain Size Envelope (After Kent)

coefficient of permeability for each range as shown in Figure 21. Weighted average permeabilities were computed by this method for all available wells within the area and are shown in Table IX.

To supplement the permeability data and to verify computed values, the results of a long term aquifer test in the area were analyzed using various methods including the Jacob, Boulton, Hantush, and Pricket methods. The Pricket type curve is shown in Figure 22. The methods and the results are given in Tables X, XI, XII and corresponding Figures 23, 24, 25, and 26. These techniques were designed for aquifer tests conducted under varying groundwater conditions including consideration of delayed drainage due to gravity. Permeability coefficients from 520 to 576 gallons per day per foot squared are obtained from the above results. These results compare favorably with the weighted average of 588 gallons per day per foot squared (see Table IX) using the permeability envelope in Figure 21 and samples obtained from the same well. The favorable correlation was considered to be justification for using the permeability grain size envelope to determine an average permeability for each driller's log.

Specific Yield

It is anticipated that on the basis of the geology in the study area, the effective specific yield will

TABLE IX

WEIGHTED AVERAGE PERMEABILITY FOR KOCHABARI WELL AND OTHER WELLS IN STUDY AREA

Range	Layer Permeability, (gpd/ft^2)	Saturated Interval thickness, (ft)	Percentage of total thickness	Weighting of permeability coefficient, (gpd/ft^2)
Kochabari Well				
1	15	13	5	0.75
2	80	73	26	20.80
3	300	109	39	117
4	1500	85	30	450
				588.55
(K) Average weighted permeability				
All other available Driller's log in the study area				
1	15	509	11	1.65
2	80	330	7	5.6
3	300	2468	54	162.0
4	1500	1283	28	420.0
				589.25
(K) Average weighted permeability				

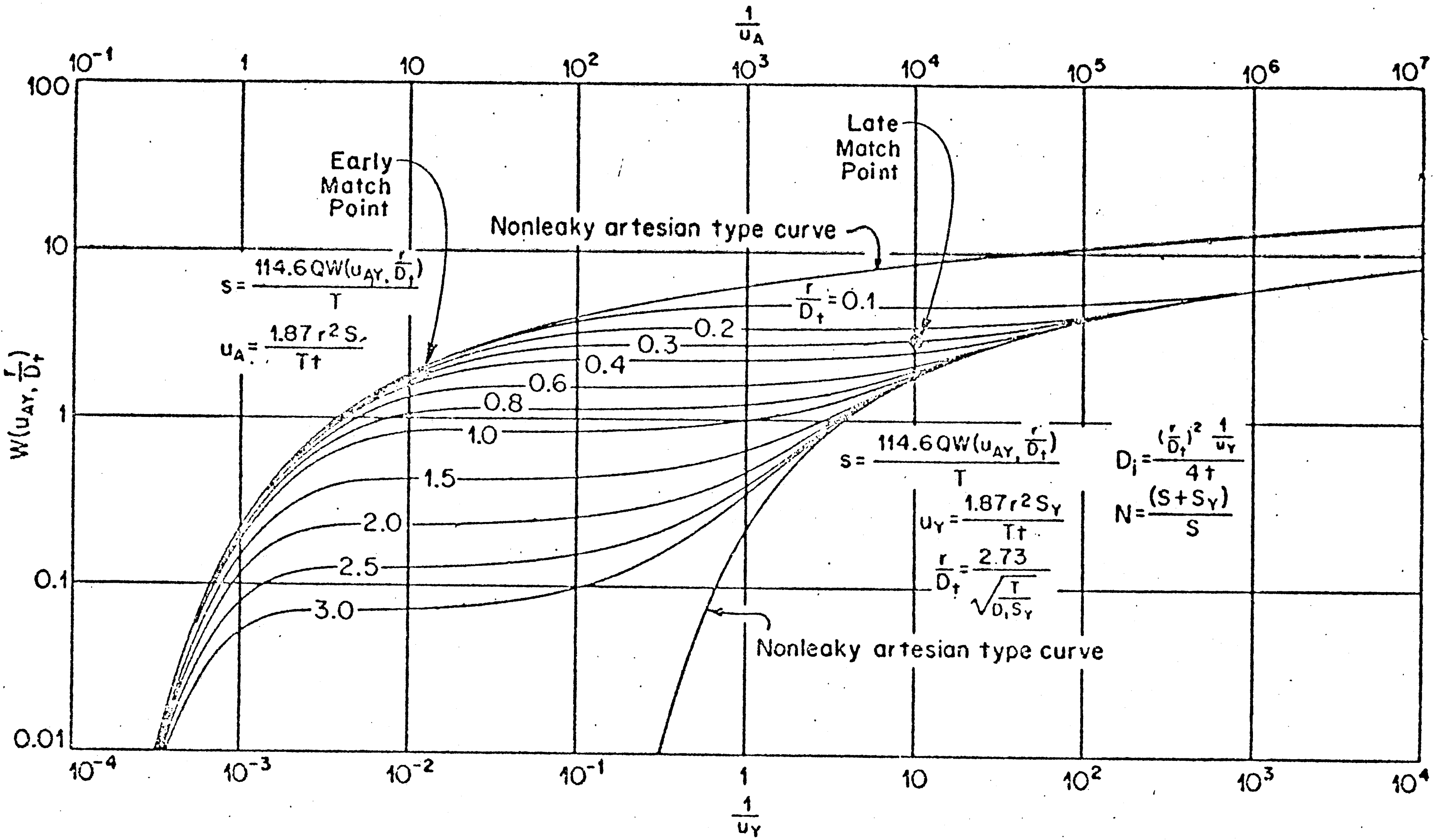


Figure 22. Prickett's Time-Drawdown Type Curve

TABLE X

PUMP TEST RESULT USING PRICKETT FOR
KOCHABARI OBSERVATION WELL #1

Saturated thickness = 250 ft   Q = 1600 gpm

r = 50 ft

Early match point

$$W(U_A \cdot \frac{r}{D_t}) = 3.5$$

$$S = 4.5 \text{ ft}$$

$$t = 2.7 \text{ minutes}$$

$$T = \frac{114.6\ Q}{S} (W_{U_A} \cdot \frac{r}{D_t}) \qquad K = \frac{T}{\text{Saturated thickness}}$$

$$T = \frac{(114.6)(1600)}{4.5}(3.5) \qquad K = \frac{142{,}613}{250}$$

$$T = 142{,}613 \text{ gpd/ft} \qquad K = 571 \text{ gpd/ft}^2$$

Late match point

$$W(U_y \cdot \frac{r}{D_t}) = 5.5$$

$$S = 7 \text{ ft}$$

$$t = 11 \text{ minutes}$$

$$T = \frac{114.6\ Q}{S}(W_{U_y} \cdot \frac{r}{D_t}) \qquad K = \frac{T}{\text{Saturated thickness}}$$

$$T = \frac{(114.6)(1600)}{7}(5.5) \qquad K = \frac{144069}{250}$$

$$T = 144069 \text{ gpd/ft} \qquad K = 576 \text{ gpd/ft}^2$$

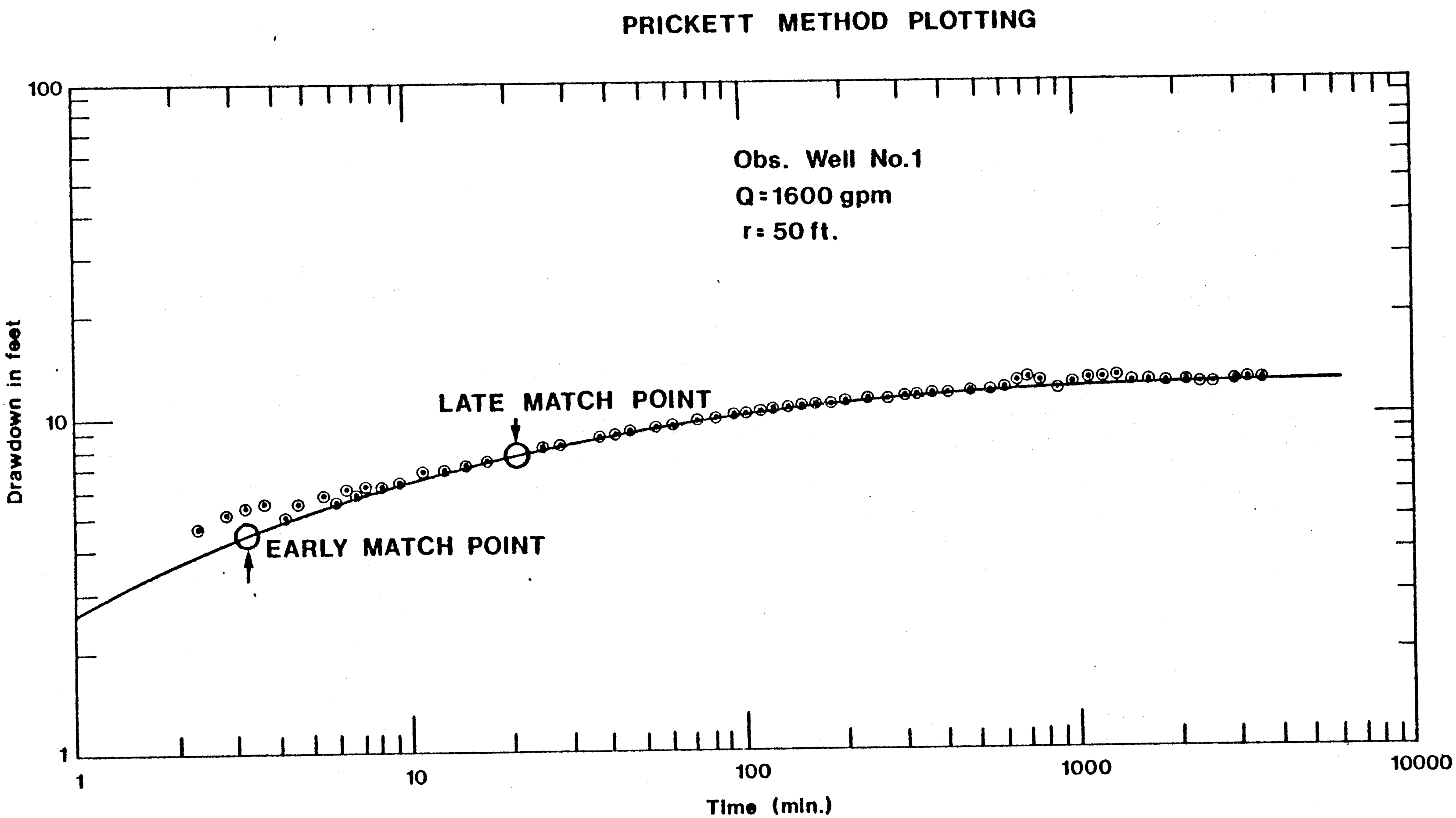


Figure 23. Prickett's Pump Test Time - Drawdown Graphs

TABLE XI

CALCULATION OF TRANSMISSIVITY "T" FOR OBSERVATION WELL #1 USING HANTUSH MODIFIED METHOD AND JACOB METHOD

$Q = 1600$ gpm

$r = 50$ ft

Saturated thickness $= 250$ ft

Hantush Method

$T = \frac{Q}{4\pi s} H(U,B)$ $\qquad$ $H(U,B) = 7.8$

$T = \frac{1600 \times 7.8 \times 1440}{4 \times 3.142 \times 11 \times 7.78}$ $\qquad$ $S = 11$ ft

$T = 17{,}379$ ft^2/day $\qquad$ $t = 150$ minutes

$T = 129{,}993$ gpd/day $\qquad$ $K = \frac{T}{\text{Sat. thickness}}$

$K = 520$ gpd/ft^2

Jacob Method

$Q = 1600$ gpm

$S = 3$ ft

$T = \frac{264\ Q}{S}$ $\qquad$ $K = \frac{140800}{250}$

$T = \frac{264 \times 1600}{3}$ $\qquad$ $K = 563$ gpd/ft^2

$T = 140{,}800$ gpd/ft

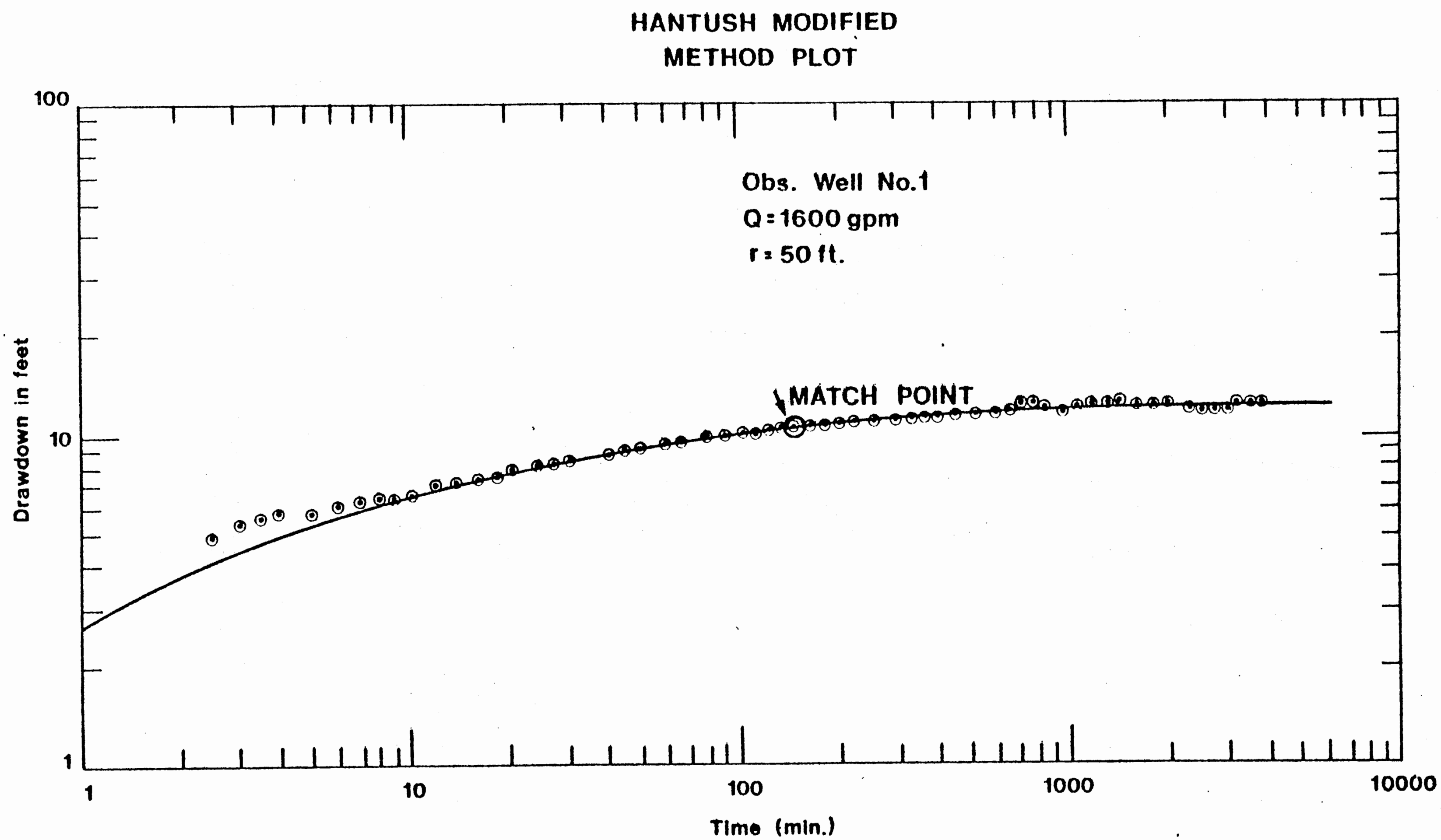


Figure 24. Hantush's Pump Test, Time-Drawdown Graph

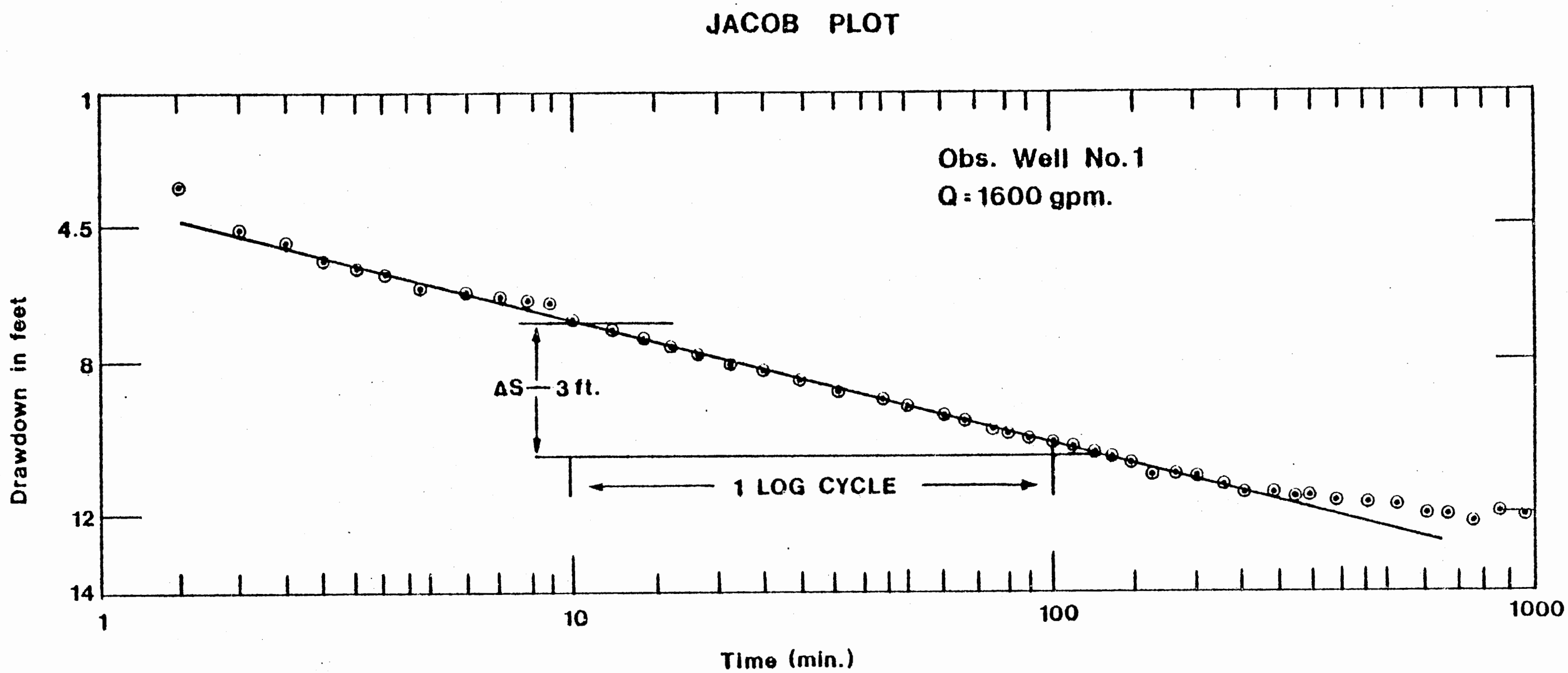


Figure 25. Jacob's Pump Test, Time-Drawdown Graph

TABLE XII

CALCULATION OF TRANSMISSIVITY "T" FOR OBSERVATION WELL NO. 1 USING BOULTON - THEIS METHOD

Saturated thickness = 250 ft

$Q = 3.56$ cfs

$r = 50$ ft

Early match point

$$W(U_A, \frac{r}{B}) = 6.7$$

$$S = 8.8 \text{ ft}$$

$$\frac{t}{r^2} = 0.011$$

$$T = \frac{Q}{4\pi S} W(U_A, \frac{r}{B}) \qquad K = \frac{T}{\text{Saturated thickness}}$$

$$T = \frac{(3.56)}{4 \times 3.142 \times 8.8} (6.7) \qquad K = \frac{139595}{250}$$

$$T = 0.216 \text{ ft}^2/\text{sec} \qquad K = 558 \text{ gpd/ft}^2$$

$$T = 139{,}595 \text{ gpd/ft}$$

Late match point

$$W(U_y, \frac{r}{B}) = 8.6$$

$$S = 11.5 \text{ ft}$$

$$\frac{t}{r^2} = 0.11$$

$$T = \frac{Q}{4\pi S} W(U_y, \frac{r}{B}) \qquad K = \frac{T}{\text{Saturated thickness}}$$

$$T = \frac{3.56 \times 8.6}{4 \times 3.142 \times 11.5} \qquad K = \frac{137656}{250}$$

$$T = 137{,}656 \text{ gpd/ft} \qquad K = 551 \text{ gpd/ft}^2$$

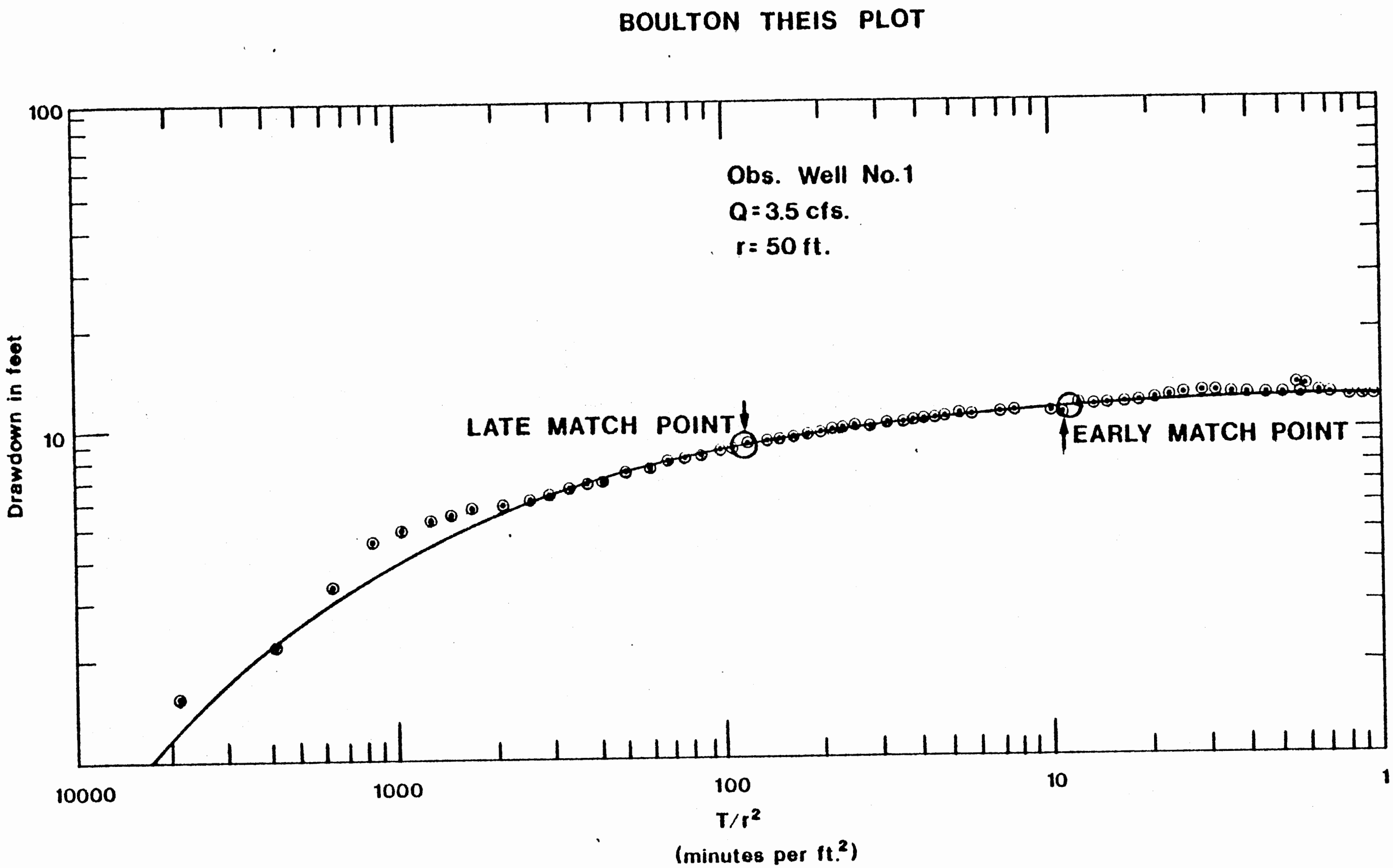


Figure 26. Boulton's Pump Test, Time-Drawdown Graph

positively correlate with lithologic type. Figure 27 illustrates representative lithologic types for unconsolidated sediments together with estimates of their associated specific yields, porosity and specific retention.

Direct methods of computing specific yield are either by hydrological balances over time or by longterm aquifer pumping tests with observation wells which frequently provide conflicting answers differing by several orders of magnitude. It has been noted by numerous observers that during longterm aquifer tests of the water-table aquifer, the rate of decline was not in accordance with the classical Theis non-equilibrium predictions. Boulton (1954) developed a semi-empirical mathematical model that regarded the water released from storage as the sum of two components: S, the volume of water instantaneously released by elastic compression of the aquifer, and Sy, the exponentially time varying drainable ground-water in storage that becomes quasi-constant after a finite time. It was noted by observers that this interpretation fitted the observed data fairly closely. Prickett (1965) attempted, with some success, to correlate Boulton's delay index with lithology. Walton (1970) and Dagan (1967) also observed that the calculated specific yield suggests that, apart from obvious lithological correlation, it varies both with time and the rate of water table decline (being a function of partially saturated flow).

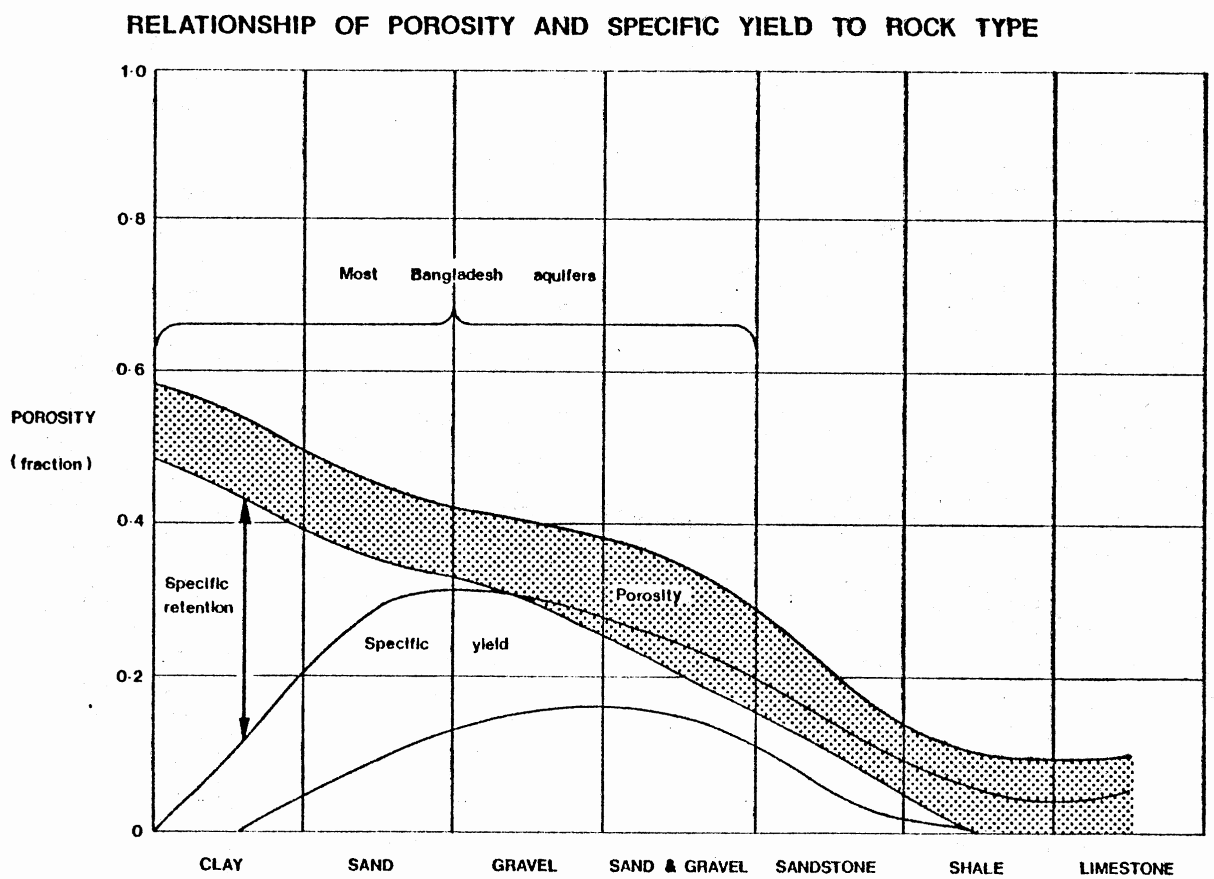


Figure 27. Relation of Specific Yields and Aquifer Materials

Hence specific yield evaluation from rising and falling water data of the project area must be conducted with some caution if realistic results are to be obtained. It is expected that analysis based on lithology will give good estimate of the specific yield over the depth dewatered. It has been noticed that the majority of unconfined aquifers are characterized by coarser sediments near the base. Thus, it is normal to have clays and silts at or near the surface with the aquifer material becoming coarser with depth. It is reasonable to expect that the specific yield will vary with sediment type, i.e., as the sediment becomes coarser with depth, the specific yield also increases with depth.

The graph shown in Figure 28 (after Johnson, 1967) was used to provide a relationship between median grain size and specific yield. The dominant grain sizes in Figure 21 were considered to be equivalent to the median grain sizes of the permeability envelope. The values of specific yield along with the corresponding permeability coefficient of the four ranges were plotted on semi-logarithmic paper to produce the relationship shown in Figure 29. Using this curve, the average specific yield in the study area was estimated to be 25 percent.

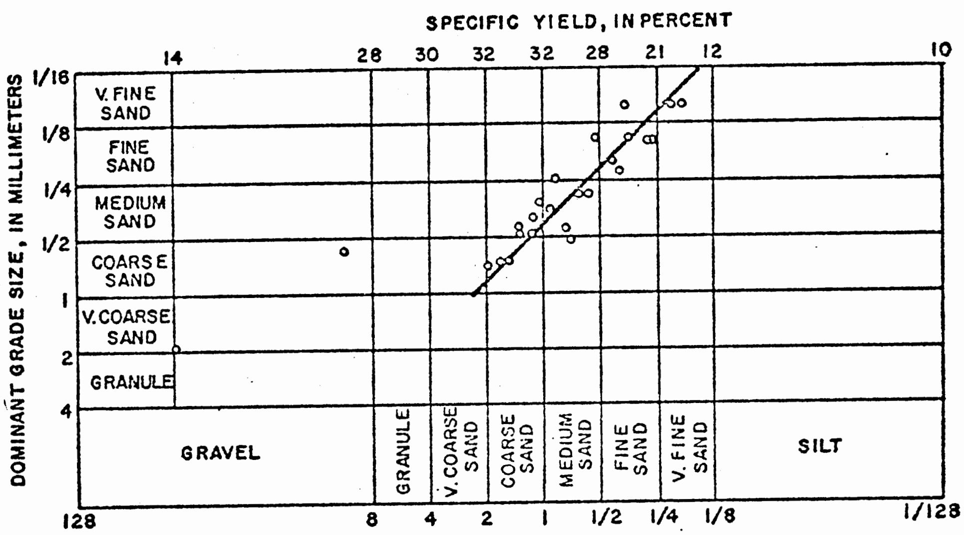


Figure 28. Specific Yield Vs. Grain Size

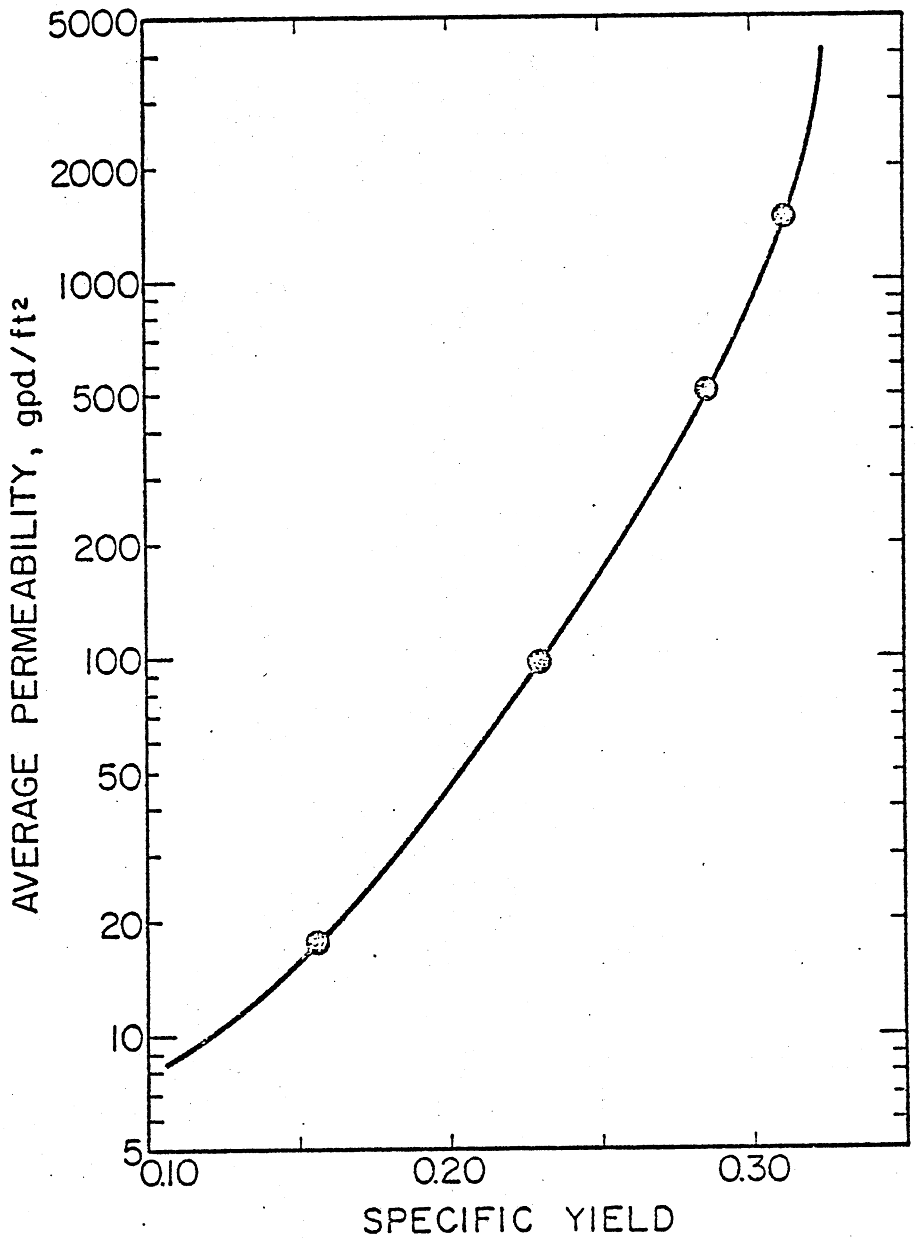


Figure 29. Specific Yield Vs. Coefficient of Permeability

Ground-water Quality

Ground-water quality must be considered in relation to rate of pumping and type of well completion in order to identify the modifying the insitu ground-water quality:

1. Possible public health hazards
2. Possible toxicity of irrigation water to plants
3. Alteration of soil zone physical properties due to ground-water use
4. Impairment of well performance due to corrosion and incrustation.

Chemical data for six wells are presented in Table XIII. Location of these wells are shown in Figure 2. The waters are of the bicarbonate type with cations of varying dominance are slightly alkaline, having a pH range of between 7 and 8. The total dissolved solids (TDS) concentration was generally low, having a value less than 250 mg/l. Most waters were undersaturated with respect to carbonate minerals, indicating propensity to dissolve soluble aquifer materials.

Sulfate was almost totally absent from the groundwater. This might be due to either the absence of sulfate source or to chemical reduction of available sulfate within the aquifer.

TABLE XIII

GROUNDWATER QUALITY ANALYSIS (1976-77)

Well No.	Ca^{++} mg/l	Mg^{++} mg/l	Na^{+} mg/l	HCO_3^{-} mg/l	SO_4^{--} mg/l	Cl^{-} mg/l	NO_3^{-} mg/l	Fe mg/l	SiO_2 mg/l	TDS mg/l	PH	SAR
1	25.0	22.9	4.2	116.0	0	8.0	T	0.25	36.0	201.6	7.9	6.85
2	26.0	9.0	34.9	130.0	T	11.0	n.d.	0.55	34.0	265.6	8.0	6.42
3	24.0	8.5	24.0	92.0	0	20.5	T	1.25	26.0	200.0	7.6	4.24
4	12.0	7.3	31.3	68.0	T	12.0	n.d.	0.27	33.5	151.6	7.7	2.83
5	17.0	7.3	30.4	88.0	T	5.5	T	0.15	35.5	204.2	7.65	2.57
6	18.0	7.3	31.0	88.0	0	7.0	T	0.55	36.5	160.0	8.0	8.51

T = trace

n.d. = not detected

The extensive use of irrigation could cause leaching of nitrate from fertilizers into the aquifer. This nitrate could pose a health hazard, especially since the people are drawing their drinking water from the top of the aquifer where the nitrates would enter.

CHAPTER VI

COMPUTER MODELING

Concept of Model

The rate of water level decline is the most important information for ground-water management in this study area. In particular, it is paramount to know how rapidly the water is being depleted, where and when the water level decline will seriously affect existing domestic shallow water supplies and what impacts alternative management practices will have on the system. Digital simulation offers a reliable means for evaluating the effects of various development alternatives on an aquifer system such as the Dinajpur Piedmont aquifer.

Obviously, it is impossible either to carry out experiments and tests in the aquifer itself in order to determine its response to various management alternatives proposed in the future or to make comparisons among responses to different possible alternatives in order to determine the most desirable one, according to some specified criteria. Whenever the treatment of real systems or phenomena is impossible or the cost of such treatment is prohibitive, models of the considered system or phenomena are introduced. Instead of treating the real system, we manipulate its model and use the results of these manipulations in order to make decisions regarding the operations of the real system.

A simulation model is a simplification of a complex physical reality and the processes in it. There is no need to elaborate on the fact that most real systems, and certainly aquifer systems, are indeed complicated beyond our capability to describe them exactly as they are. Simplifications are necessary. They take the form of a set of assumptions which should be kept in mind whenever the model is being employed in the course of investigations. On the basis of these simplifying assumptions, a model of an investigated ground-water system is constructed.

Ground-water flow obeys a well defined set of physical laws which can be expressed mathematically. The equations are balanceequations based on the principle of conservation of mass and Darcy's law. The mathematical equation describing the two dimensional ground-water flow through an areally extensive aquifer is given by the following partial differential equation (Bittinger and others, 1967):

$$\frac{\partial}{\partial x} Kb \frac{\partial H}{\partial x} + \frac{\partial}{\partial y} Kb \frac{\partial H}{\partial y} = S \frac{\partial H}{\partial t} + \frac{Q}{\Delta x \Delta y} \qquad (14)$$

where

K = hydraulic conductivity (L/T)

(L) denotes units of length, (T) denotes units of time.

b = saturated thickness (L)

H = hydraulic head (L)

S = specific yield (dimensionless)

Q = net withdrawal flow (L^3/T)

$\Delta x, \Delta y$ = incremental distances

t = time (T)

This equation is based on the continuity equation and Darcy's law and is consistent with the Dupuit assumptions. The Dupuit assumptions are (1) the velocity of ground-water flow is proportional to the slope of the hydraulic gradient and (2) the ground-water flow is horizontal and uniform every where in the vertical section (Todd, 1959). The above equation also assumes that both hydraulic conductivity and specific yield are isotropic and that the density of the fluid is constant in time and space.

The above equation has no general solution; therefore a finite difference approximation is used to allow a numerical solution with a digital computer. Applications of the finite difference approach requires subdivision of the study area into a system of finite grids or subregions. The differentials of the above equations can be approximated by first order finite difference expressions.

Characteristic values of all variables used in finite difference expressions are specified for each sub-region. These discrete values represent the value for the entire node.

One extremely versatile finite difference model has been developed by Trescott and Pinder (1973) and several problem options and input-output features have also been added. A modification of the 1974 version of this model has been used in this study. The computer methodology explains the steps used, as shown in Figure 3.

Data Input

Data input includes all data used in the model consisting of scalars, uniform matrices and non-uniform matrices. In addition, some matrices used by the program were generated from other matrix input. The scalar and uniform data were entered directly into the model. The matrix data were digitized and assigned to each node using an interactive data entry program running under IBM 370 TSO. Values were required for the following program variables:

A. Scalars

1. Number of rows used in the grid(DIML)
2. Number of columns used in the grid (DIMW)
3. Grid spacing in x-direction (DELX)

4. Grid spacing in y-direction (DELY)
5. Length of time steps (DELT)
6. Length of pumping period (TMAX)
7. Number of pumping periods in total simulation time (NPER)

B. Uniform Matrices

1. Coefficient of permeability (PERM)
2. Specific yield (SY)
3. Recharge rate (QRE)
4. Coefficient of permeability for riverbed (RATE)
5. Distance of river influence node elevation (M)

C. Non-uniform Matrices

1. Land elevation from topographic map (LAND)
2. 1977 - water table elevation (STRT)
3. Prior irrigation pumping rate (WELL)
4. Constant gradient (GRAD)

D. Computer Generated Matrices

1. Elevation of bottom of aquifer (BOTTOM)
2. Elevation of bottom of river (TOP)
3. River water elevation (RIVER)

The model area has approximately 588 square miles. A rectangular grid spacing (5280 feet x 5808 feet) was chosen and the resulting grid system consisting of 34 columns and 30 rows is shown in Figure 4. The model system contains 535 nodes. A one-year run of two periods was used for calibration. The allocation simulation period was divided into 40 pumping periods and a time step of 10 days was used. A time step is the period of time in which water table elevations are computed to accomodate recharge and/or discharge in the system. The length of the pumping period was 150 days for the dry period when the river was gaining water from the basin, and 210 days for the wet period when the basin was recharging from precipitation.

A homogeneous value of 600 gpd/ft^2 for permeability coefficient and 0.20 for specific yield were used as uniform matrices. Recharge rate from precipitation was calculated to be 24 inches per year. Water gained into the basin during the wet period was drained to the river during dry period. Therefore, the river was considered to be spread over three percent of the area. The coefficient of permeability for river bed was, therefore, adjusted in the model to be 180 gpd/ft^2 (600 x 3%). The distance of influence of the river to adjacent nodes was 100 feet.

The land and water table elevations were derived from topographic and water table maps, respectively, and

and the identified values were assigned to each node. Prior irrigation pumping rates were calculated from pumping rates in existing wells.

The elevation of the base of aquifer was set 300 feet below the surface. The river water elevation was set to the initial water elevation plus a constant calibration. The bottom of the river was set sixteen feet below the initial water elevations.

The model was used to simulate one-year and twenty-year periods. One-year simulation runs were used to calibrate the model. The twenty-year simulation runs were initiated on April, 1977 and terminated on March, 1997. The longer simulation period was used to assign predicted annual pumping allocation rates based on predicted drawdown.

Assumptions for Calibration

In order to protect the shallow dug and hand pump wells as well as to meet the existing irrigation requirements, the digital model used in this study is based on the following assumptions:

1. The maximum allowable drawdown is set at 22 feet of water depth from the surface
2. Time periods are assumed to be five months for the dry period and seven months for the wet period. These represent limiting recharge

during the wet period and river flows during the dry period.

Most of the hand pump wells are 1½ inches diameter cased well with average 100 feet deep. Each well yields approximately 3240 gallons per day at the rate of 0.1 cfs discharge with 12 hours continuous pumping per day. The drawdown is calculated using the following equation (Walton, 1970):

$$\frac{Q}{S} = \frac{T}{264 \text{ Log} \dfrac{Tt}{2693\ r_w^2\ S_y} - 65.5} \qquad (15)$$

where

$\frac{Q}{S}$ = specific capacity in gpm/ft

Q = discharge in gpm

S = drawdown in feet

T = coefficient of transmissivity in gpd/ft

S_y = specific yield, fraction

r_w = nominal radius of well, feet

t = time after pumping started, minutes

The calculated drawdown is less than 0.15 feet. It is very insignificant and is ignored in selecting pumping level. The practical suction lift for those shallow hand pump wells according to international standard (Campbell and Lehr, 1973) is 22 feet. The maximum drawdown for irrigation wells is therefore restricted to 22 feet only. The resulting water depth during 20 years simulation run were subdivided into ranges as shown in the 20-year sequence computed map in Figures 34, 35, 36, 37, and 38. The depth of 22 feet is considered critical depth because pumping below this depth will cause the existing shallow hand pump wells to be inoperative.

The conceptual models of the study area is shown in Figures 30, 31, and 32. These models were based on available data and is used with the computer model for better calibration. The water table rises during wet period and recedes during the dry period. Based on water table data, rainfall and river discharge data, April to October (7 months) was chosen as the wet period, when the river is gaining water from excess surface runoff and the ground-water table rises due to heavy infiltration from precipitation. Thus, the dry period is from November to March (5 months), when the ground-water table recedes and the river is gaining water from the basin. The annual result is a recharge-discharge equilibrium.

Calibration

Calibration was achieved by running the model on the basis of the conceptual model and comparing the 1977 observed head elevations with the computed values. The main objective of this calibration was to establish the recharge-discharge equilibrium. Equilibrium is established when the mass balance shows the inflow and the

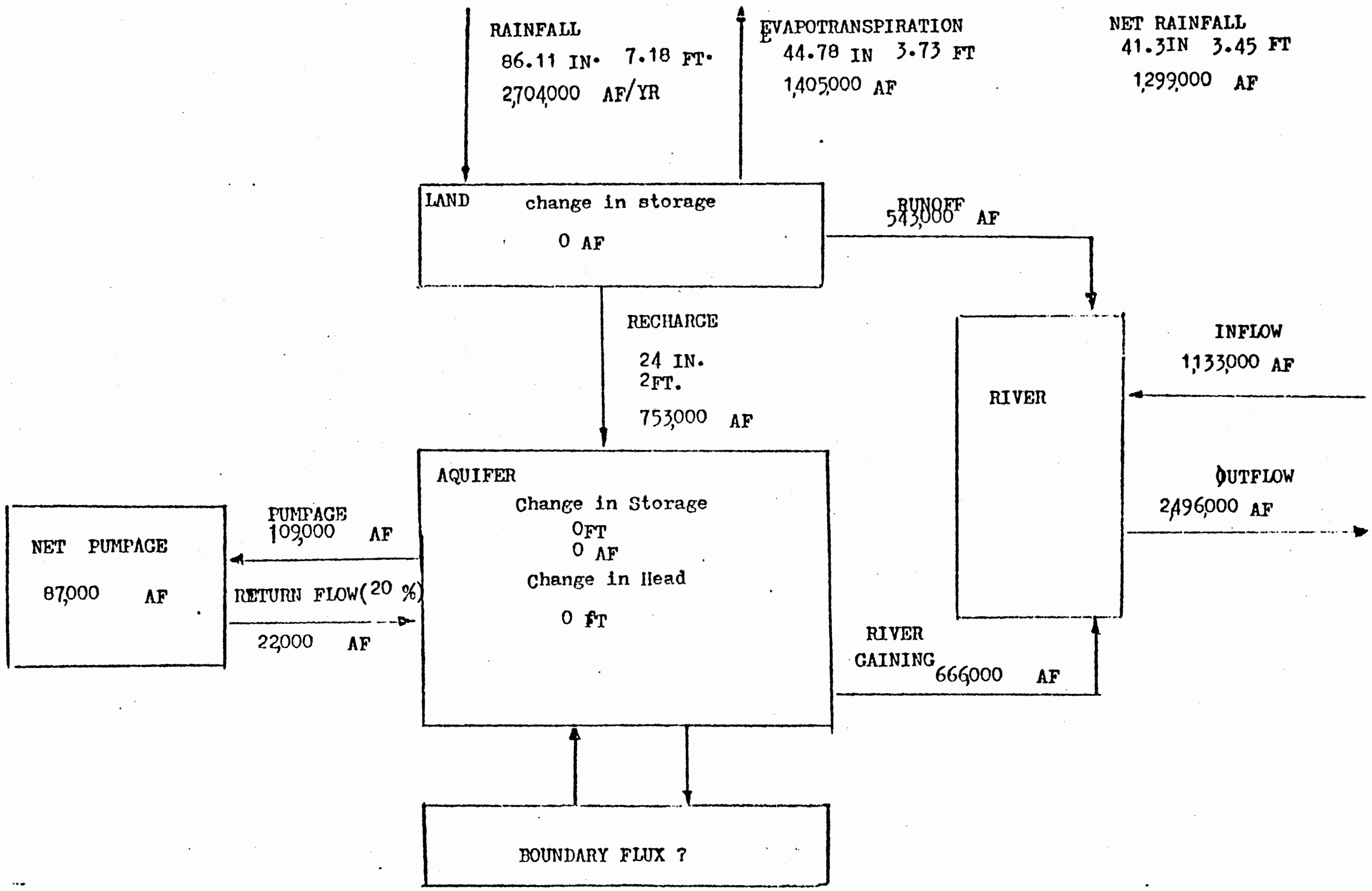


Figure 30. Conceptual Model for One Year Period

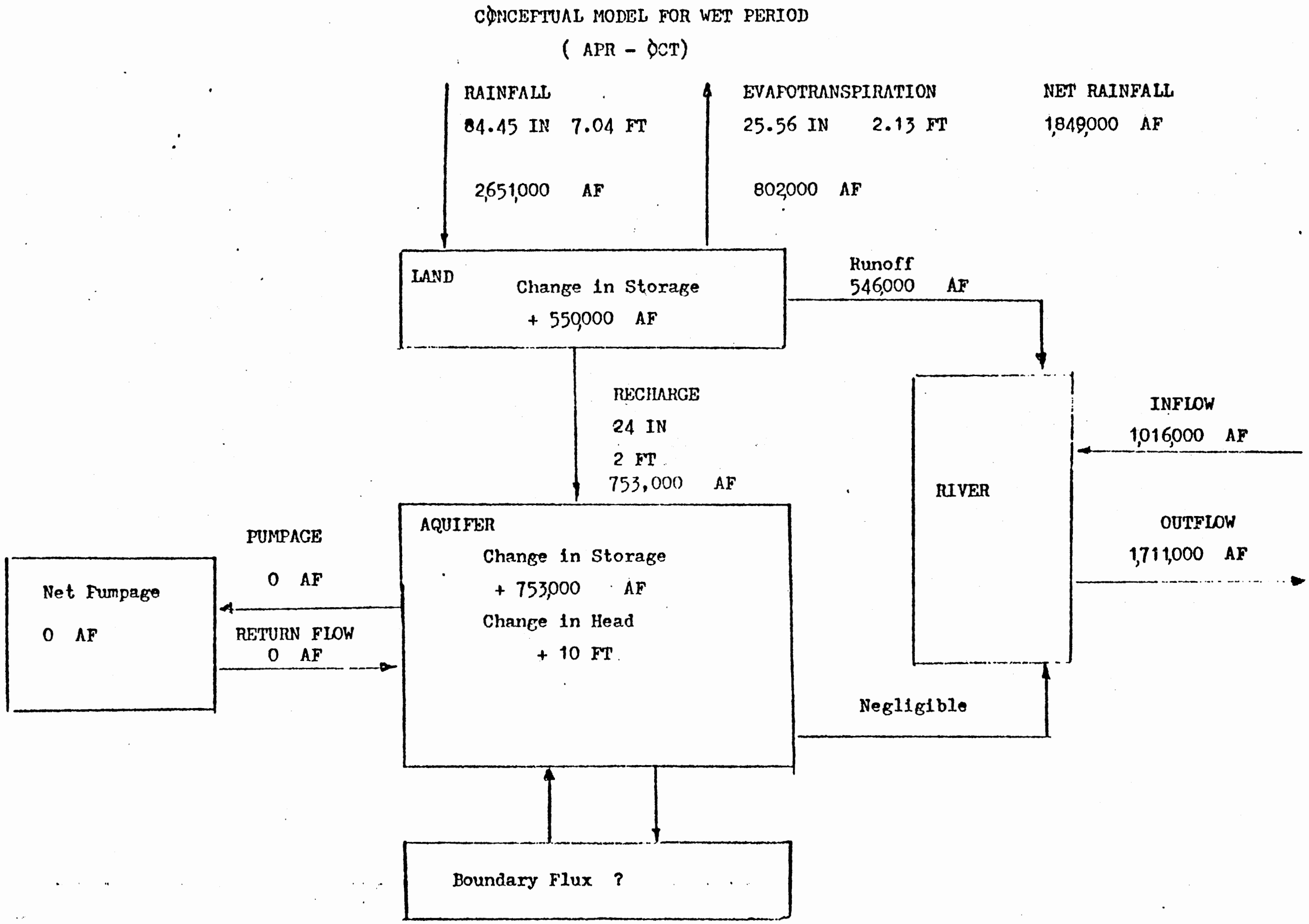


Figure 31. Conceptual Model (wet period)

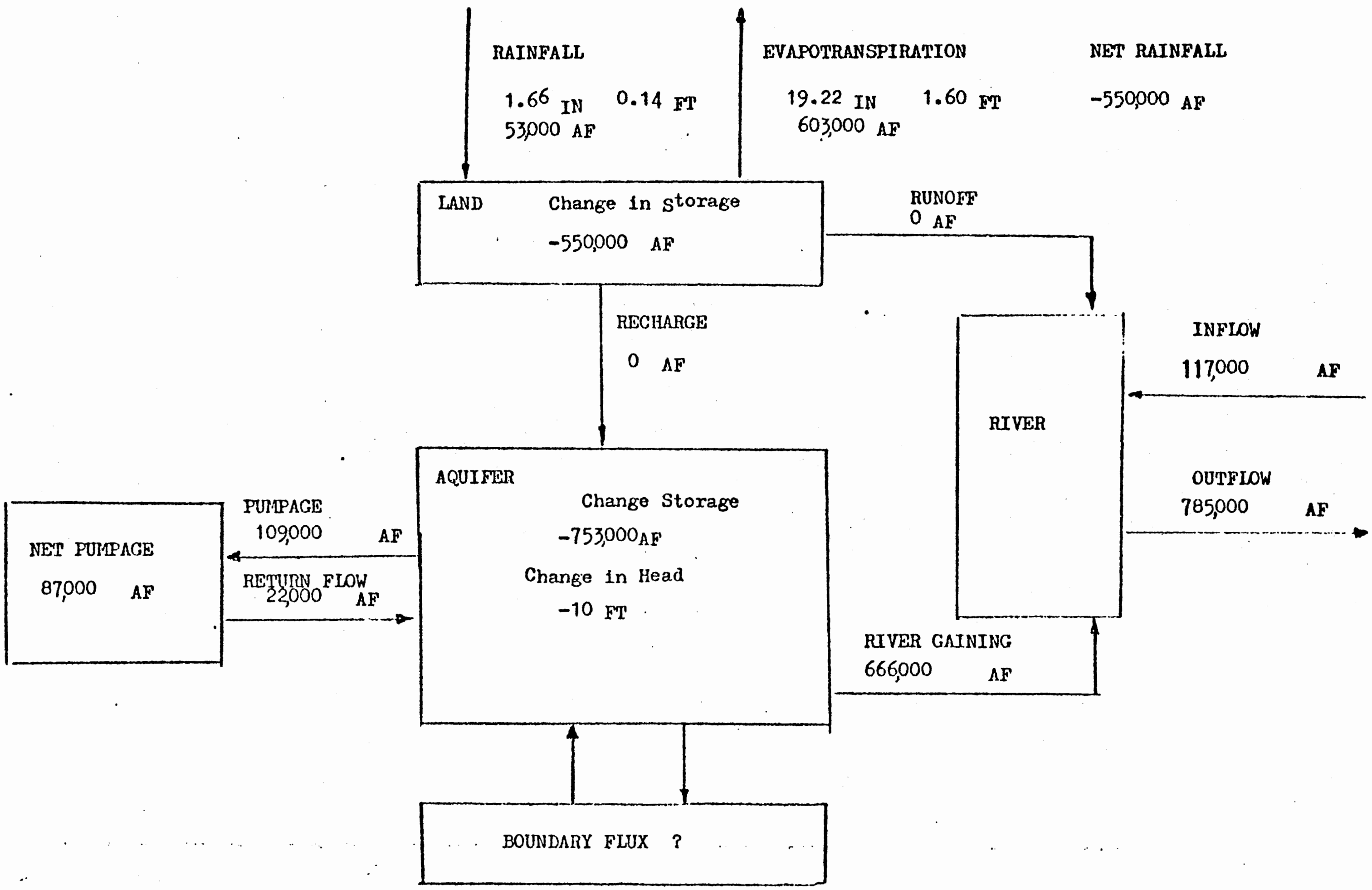


Figure 32. Conceptual Model (dry period)

outflow as being equal and is indicated by negligible fluctuations in the water table elevations.

After the initial calibration, using constant recharge, the discharge was calibrated to set the aquifer to the recharge-discharge equilibrium. The initial one year simulation resulted in an appreciable rise in the water table. Apparently the water could not be sufficiently drained. It was noted from the model that ground-water drainage concided with perennial rivers flowing over this area. The water which rises up to ten feet due to recharge during wet period was discharged fully to the river during the dry period. During calibration water was not shown sufficiently discharged into the rivers and removed from the ground-water system. In order to increase this discharge into perennial rivers, minor adjustment of river elevation was made. Other excessively rising nodes did not respond to river discharge due to their distant location from the river. Finally, the model was modified to allow for river discharge on all nodes in order to represent intermittent streams throughout the area. After making final adjustments, an equilibrium condition was achieved and model calibration was completed.

CHAPTER VII

RESULTS

A 20-year computer simulation was conducted for the 1977 to 1997 period for entire area using pumping rates of prior irrigation pumpage. This simulation was repeated with allocation pumping in conjunction with prior irrigation pumping. The allocation pumping was used when it was greater than the prior irrigation pumping for a node.

The model using one year simulation runs to achieve equilibrium, a 20-year simulation was conducted using prior irrigation pumpage. The mass balance is shown in Table XIV. The 1977 prior pumpage water head is shown in Figure 33. When comparing the 1997 water elevation map (Figure 33) to the corresponding 1977 (Figure 12) a slight fluctuation in head was noted and was considered negligible. Repeated runs were made using various allocation to achieve a maximum yield where no more than 50% of the nodes would be expected to go below critical depth (22 ft). Table XV shows the simulation results for different allocation with corresponding water depth, drawdowns and the percentage of nodes which are predicted to occur below the critical depth of 22 feet. The total maximum annual yield

TABLE XIV

MASS BALANCE FOR PRIOR IRRIGATION PUMPING
FROM 1977 to 1997

	Average Annual (Acre-feet)		Twenty-year total (Acre feet)	
	Inflow	Outflow	Inflow	Outflow
Recharge	753,216	...	15,064,313	
Pumpage	...	87,112	...	1,742,235
River leakage	4,124	651,172	82,486	13,023,437
Subsurface flow	4,852	23,506	97,037	470,127
Total	762,192	761,790	15,243,834	15,235,798
Net storage change	+402		+8036	

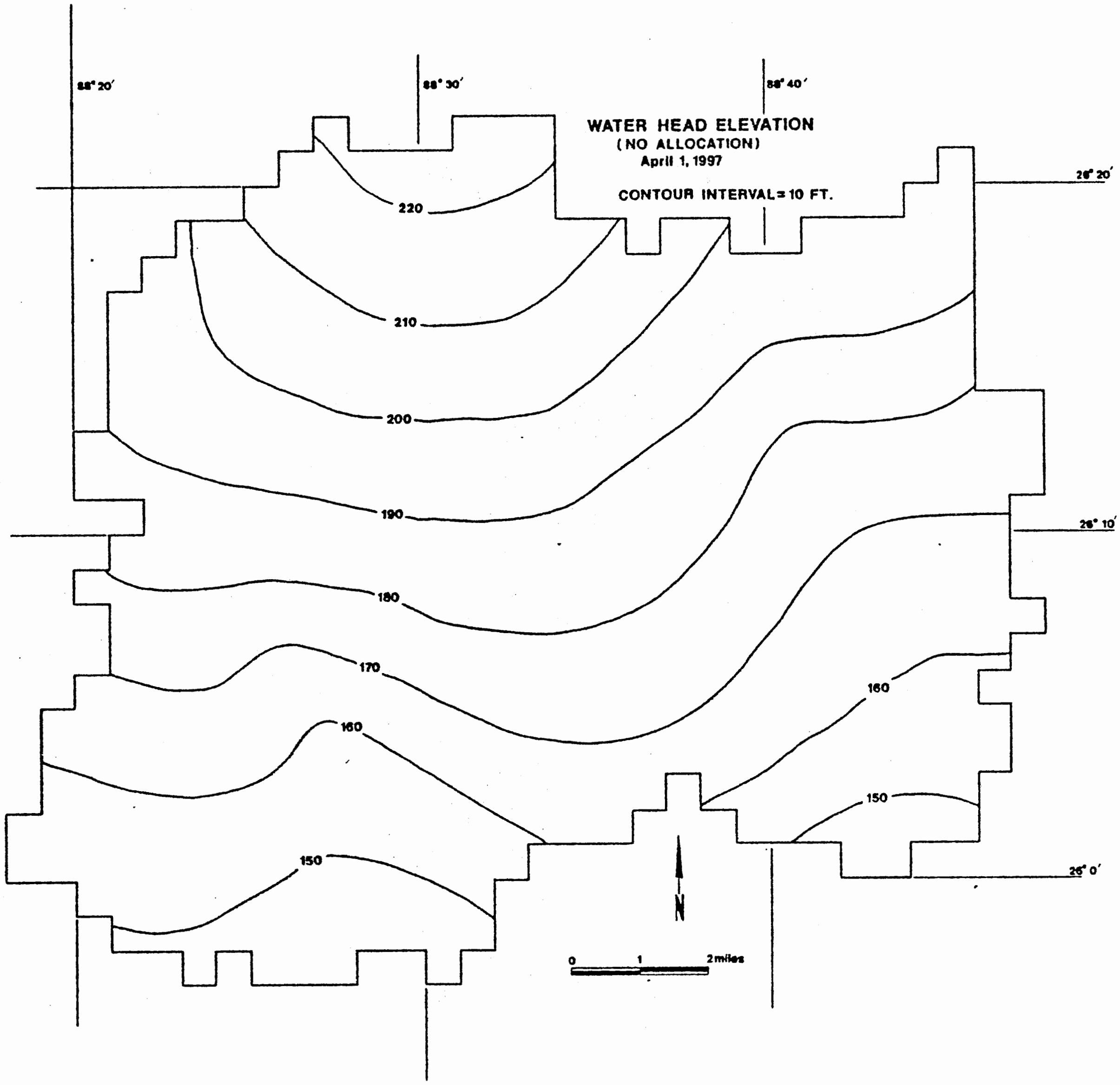


Figure 33. 1997 Water-table Contour Map (Prior Irrigation)

TABLE XV

SUMMARY OF SIMULATION RESULTS
WITH VARIOUS ALLOCATION

Allocation of pumping rate: cfs/acre	Allocation of pumping rate: Acre-ft/acre	Water withdrawal from storage (Acre-feet)	Drawdown (ft)	Depth of water from surface (ft)	No. of nodes below 22 feet (Critical depth)	Percent of nodes below 22 feet
0.0034	0.5	-50,000	0.6	13.0	0	0
0.0068	1.0	-147,000	2.1	14.8	0	0
0.010	1.5	-270,000	3.6	16.27	0	0
0.015	2.23	-685,000	8.4	21.03	200	40%
0.016	2.38	-935,000	11.31	23.94	352	66%
0.020	3.00	-4,314,000	44.0	80.0	535	100%

was determined to be 843,000 acre-feet per year using a pumping allocation of 2.23 acre-feet per acre per year. This yield was produced by dividing the total pumpage (1997) by the period of simulation of 20 years. The mass balance is shown in Table XVI. A 20-year sequence of maps showing areas which had crossed the depth limit (critical depth) are shown in Figures 34, 35, 36, 37, and 38. The final water table map of 1997 with allocation is shown in Figure 39. A 20-year ground-water budget was computed for the final computer allocation runs of the area is shown in Figure 40.

TABLE XVI

MASS BALANCE FOR PRIOR IRRIGATION PUMPING AND 2.23 ACRE-FT/ACRE ALLOCATION FROM 1977 to 1997

	Average Annual (Acre-feet)		Twenty-year total (Acre-feet)	
	Inflow	Outflow	Inflow	Outflow
Recharge	753,216		15,064,313	
Pumpage		674,159		13,483,187
River leakage	9,626	104,630	193,910	2,092,603
Subsurface flow	4,697	23,049	93,951	460,973
Total	767,609	801,838	15,352,173	16,036,762
Net storage change	-34,229		-684,589	

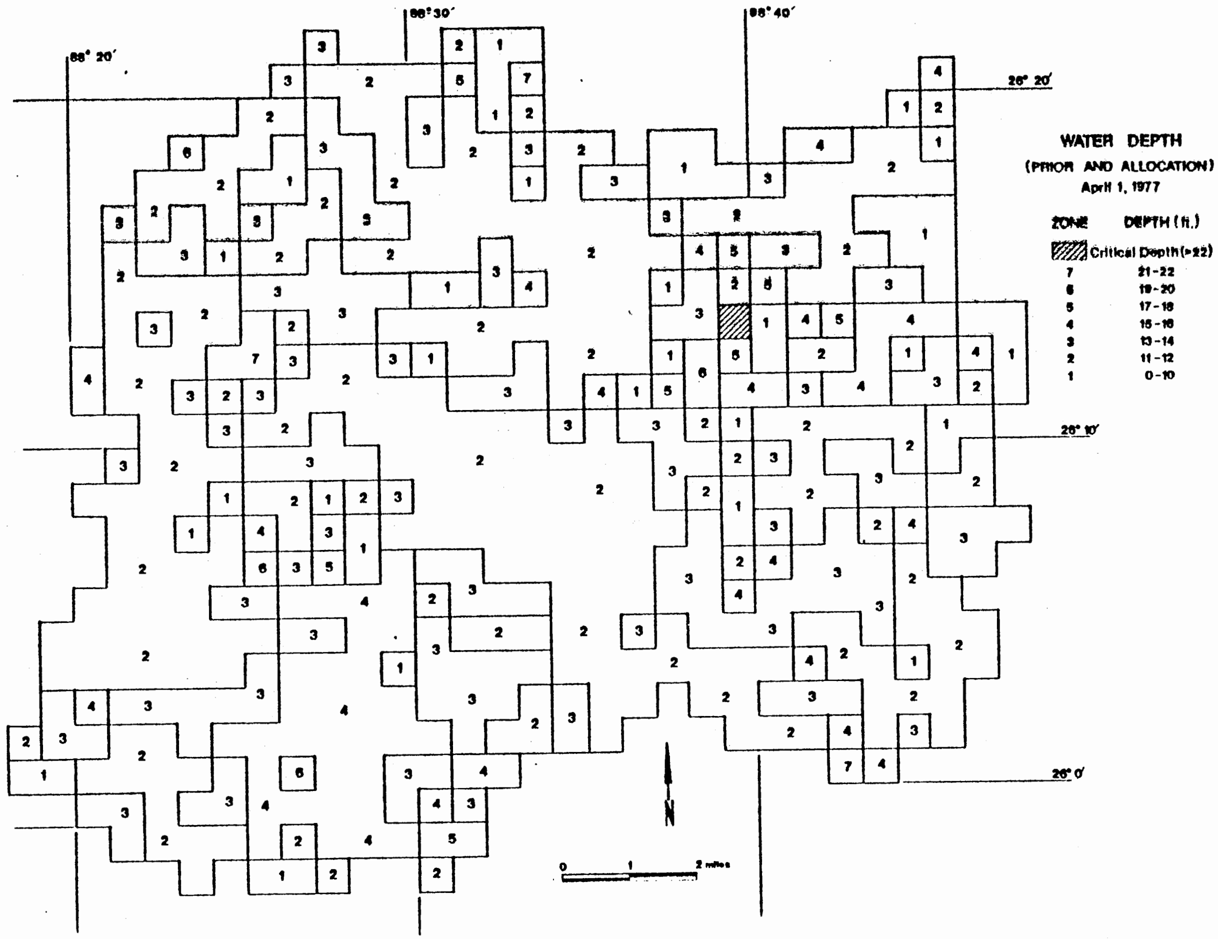


Figure 34. 1977 Water Depth Map (Allocation)

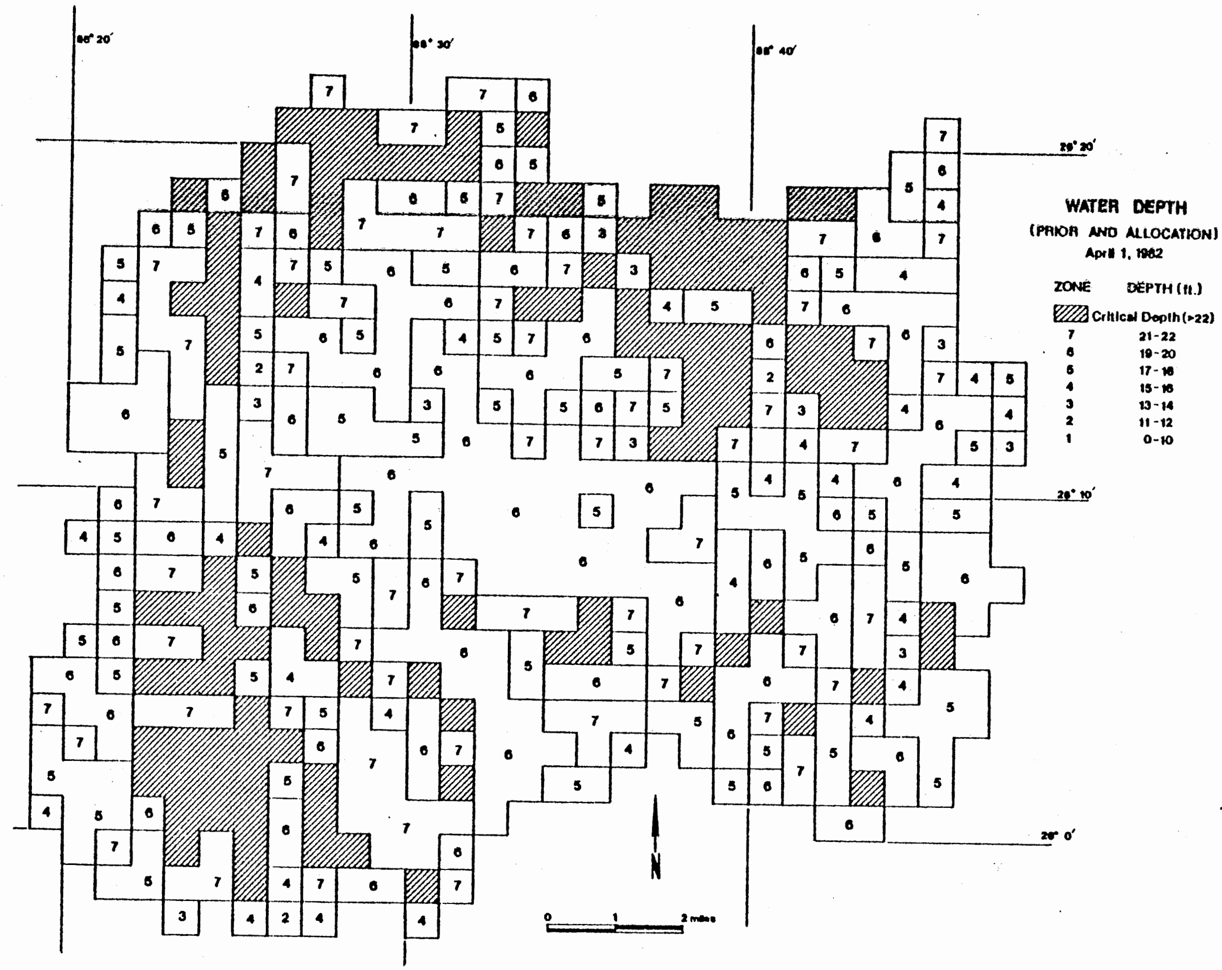


Figure 35. 1982 Water Depth Map (Allocation)

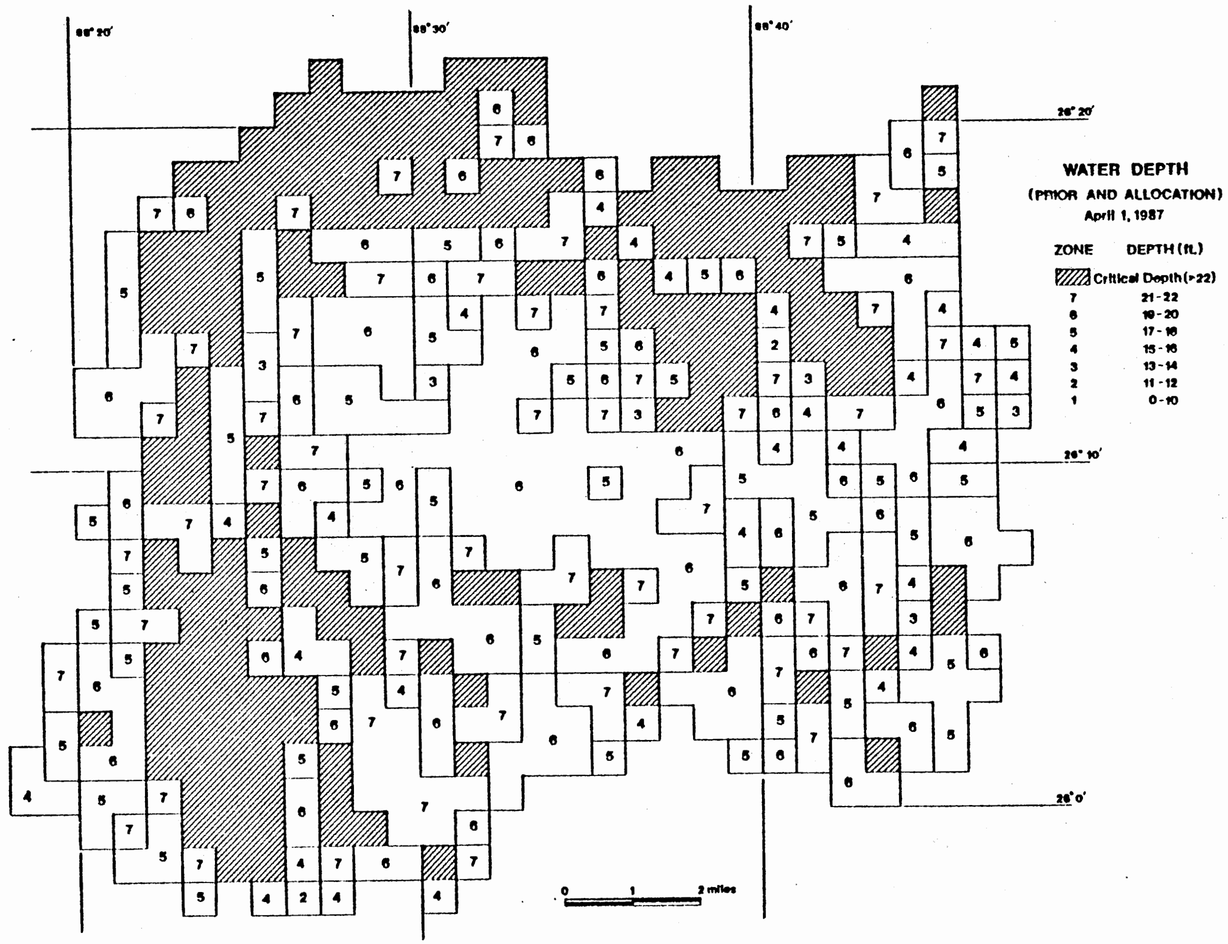


Figure 36. 1987 Water Depth Map (Allocation)

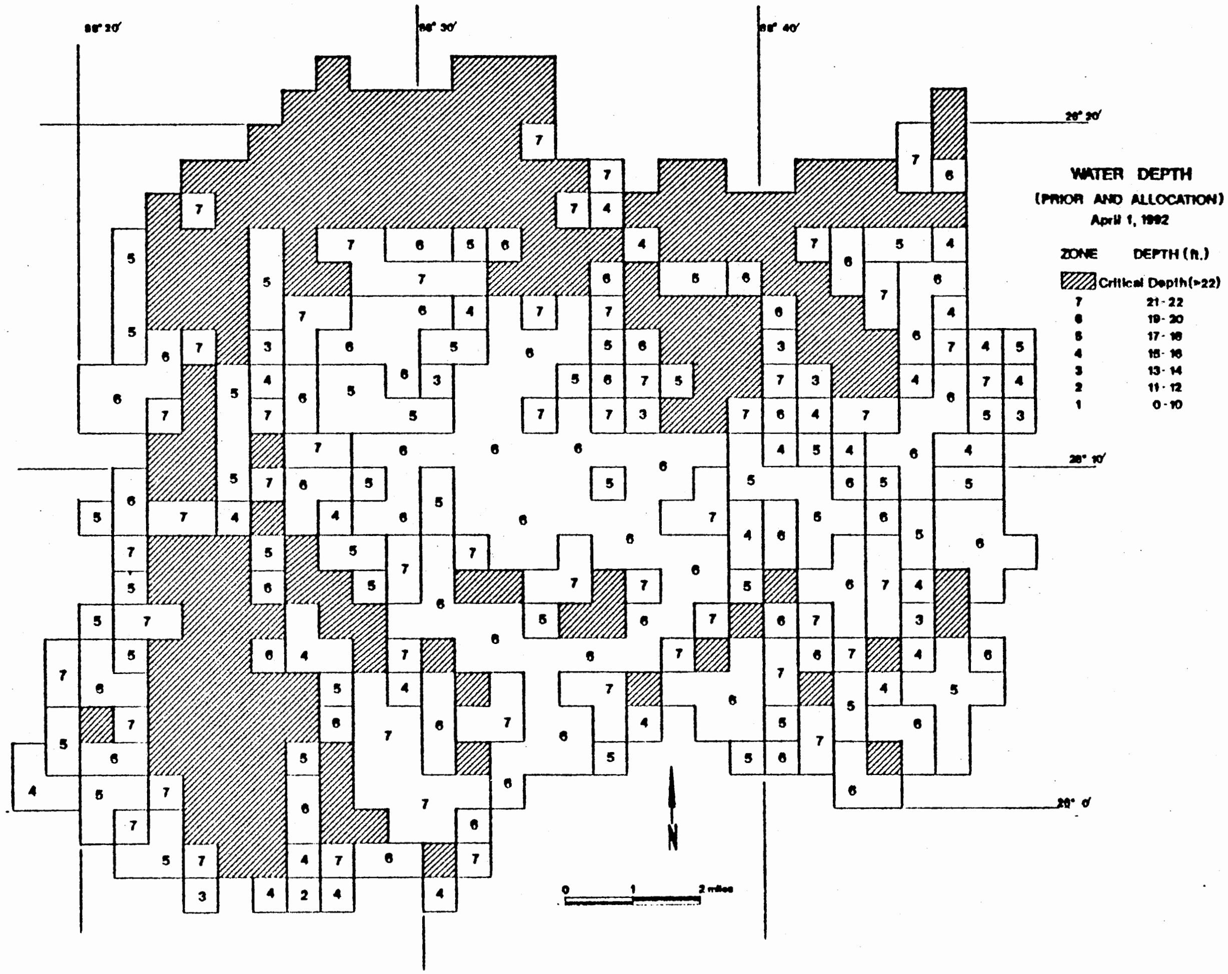


Figure 37. 1992 Water Depth Map (Allocation)

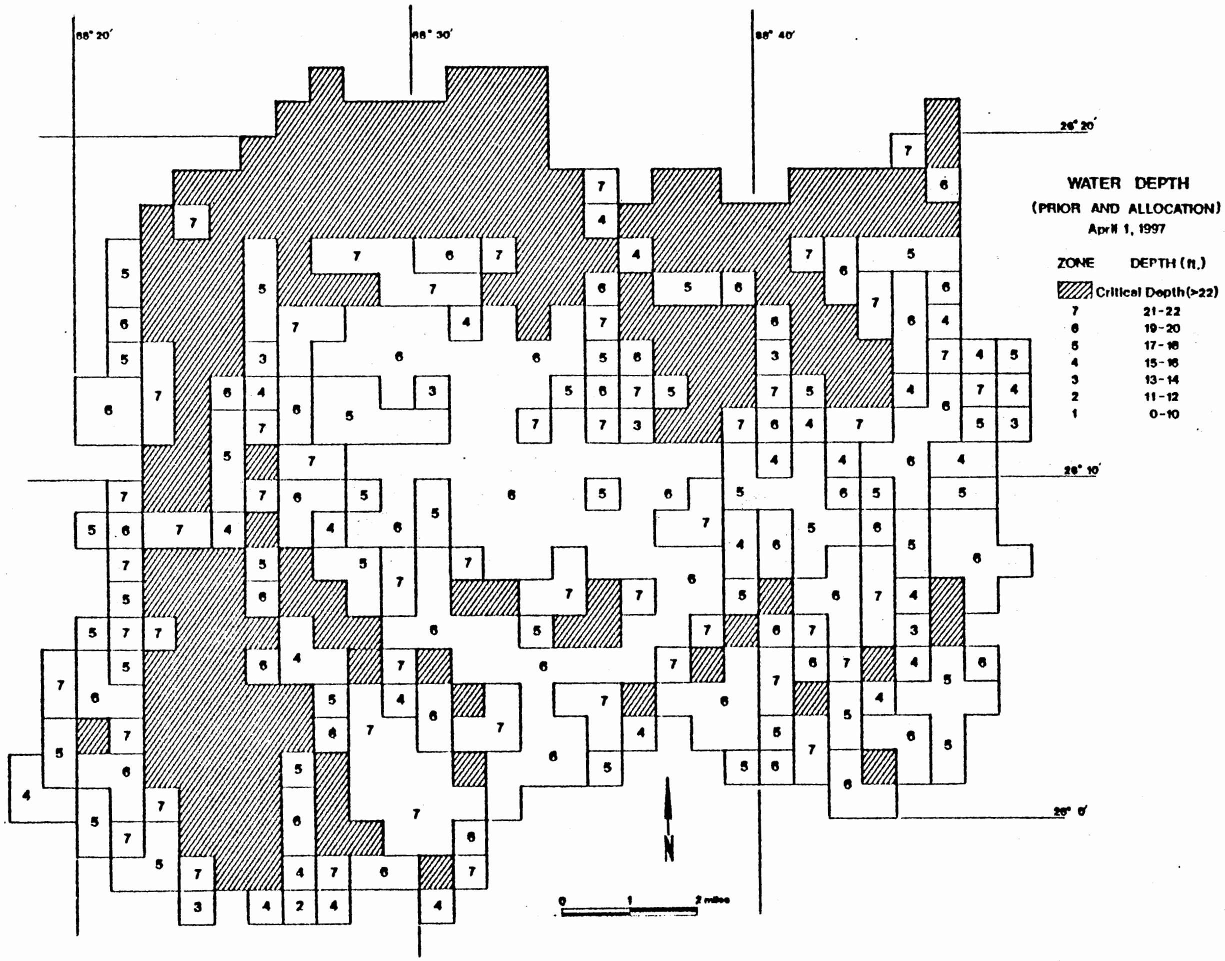


Figure 38. 1997 Water Depth Map (Allocation)

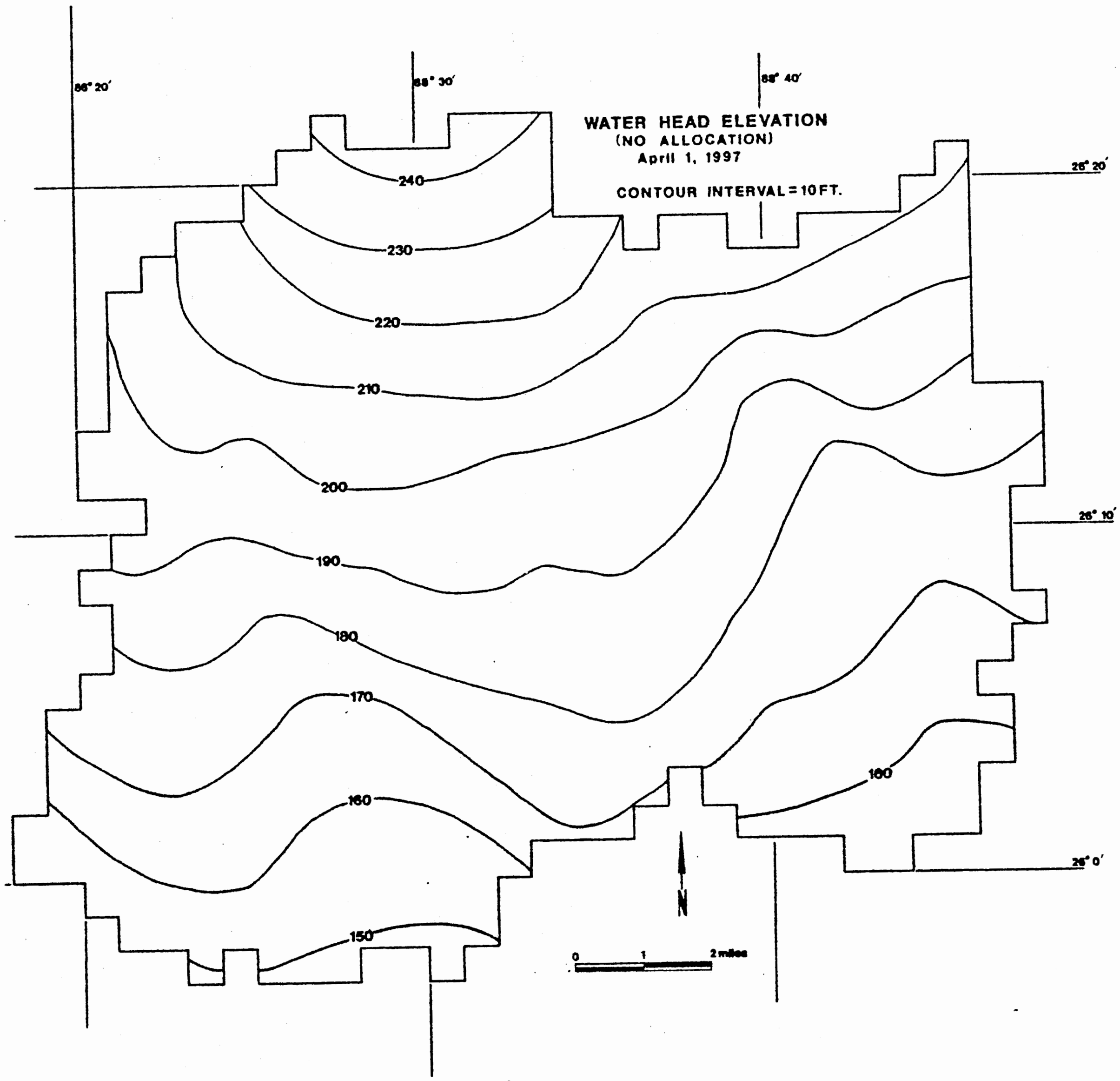


Figure 39. 1997 Water-table Contour Map (Allocation)

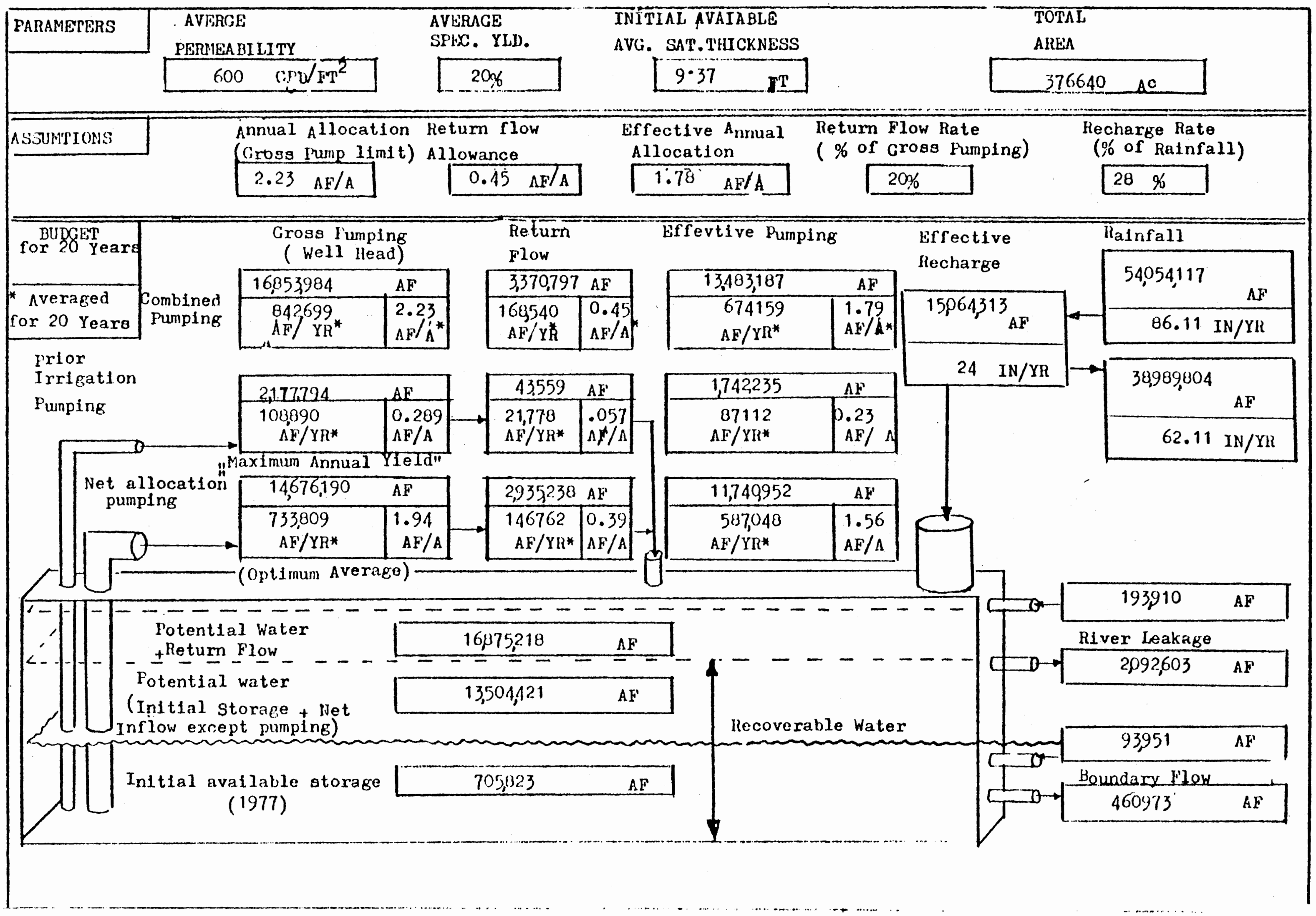


Figure 40. 20 Years Ground-Water Budget of the Study Area

CHAPTER VIII

SUMMARY AND CONCLUSIONS

Computer simulation is an effective tool for determining the maximum annual yield with limited drawdown for the ground-water basins in the central part of the Dinajpur district. The reliability of the output results of the model, however sophisticated, can only be as good as the data input upon which they are based. It is, therefore, essential in the modeling process to devote considerable care to the collection, interpretation, and validation of these data.

The principal parameters from which the annual yield was determined are:

1. The total land area is 676,640 acres overlying the Piedmont deposits in the aquifer;

2. The volume of water within 22 feet from surface storage in the aquifer as of 1977 is 705,823 acre-feet; this cumulative volume of water in storage for 20 years is 16,875,000 acre-feet;

3. The estimated rate of natural recharge is 24 inches per year;

4. The average specific yield in the upper part of the aquifer is 0.20; and

5. An average permeability is 600 gpd/ft^2.

With the prior irrigation pumpage rate the total water can be used at a rate of 109,000 acre-feet per year between 1977 to 1997. The mass balance is summarized in Figure 40. A total volume of 16,854,000 acre-feet represents the cumulative amount of water pumped from storage within 22 feet from the surface at the discharge rate of 2.23 acre-feet per acre during a 20-year period. A cumulative volume of 1,742,000 acre-feet is pumped at the prior irrigation pumpage rate during the same period. A cumulative volume of 16,875,000 acre-feet is stored within 22 feet from surface over the 20-year period. A groundwater storage within 22 feet of approximately 705,823 acre-feet is computed to have existed in 1977. An additional 16,169,000 acre-feet was accumulated due to recharge and return flow, expecially recharge during the non-pumping periods (7 months per year).

The optimum pumping rate (2.23 acre-feet per acre) is almost half of the existing crop demand as discussed in Chapter III (4 acre-feet per acre). Therefore, only fifty percent of the total area can be used for irrigation when using this rate. The pumping rate would be one cfs for twelve hours continuous pumping per day during the irrigation season. Therefore, ten wells of one cfs capacity or five wells of two cfs capacity can be alloted per square mile (640 acres). The result of modeling allow a choice of actions to produce various predictive results. A pumping limit of 1.5 acre-feet per acre is expected to prevent any shallow hand pump wells from being affected. A pumping limit of 2.23 acre-feet per acre is expected to affect 40% of the shallow wells. If the entire area is irrigated at the crop demand level of 4 acre-feet per acre, all shallow wells will be expected to be inoperative.

REFERENCES

Al-Sumait, Abdulaziz Jasen. 1978. A computer ground-water model for the Tillman Alluvium in Tillman County, Oklahoma. Unpublished Master's Thesis, Oklahoma State University.

Bangladesh Water Development Board. 1968-77. Hydrological Yearbooks of Bangladesh.

Bangladesh Water Development Board. 1977. Ground-water investigation in Rangpur and Dinajpur Districts, Water Supply Paper No. 414.

Bittinger, M. W., Duke, H. R., and R. A. Longenbaugh. 1967. Mathematical simulation for better aquifer management. International Association of Scientific Hydrology, Publication 72, Symposium of Haifa, p. 509-519.

Boulton, N. S. 1954. Unsteady radial flow to a pumped well allowing for delayed yield from storage. Int. Assoc. Sci. Hydrol. Rome, 2, pp 472.

Boulton, N. S. 1963. Analysis of data from nonequilibrium pumping tests allowing for delayed yield from storage. Proc. Inst. Civil Engrs., 26, pp. 469-482.

Brammer, H. 1969. Geology, physiology, hydrology and soils of East Pakistan, Directorate of Soil Survey. Soil Survey Project of East Pakistan.

Campbell, M. D., and J. H. Lehr. 1973. Water well technology, Chapter 10, p. 224.

Childs, E. C. 1969. Introduction to the physical basis of soil water. J. Wiley.

Coleman, J. M. 1969. Brahmaputra River: Channel forms and processes. Sedim. Geol., 3, 129-239.

Dagan, G. 1967. A method of determining the permeability and effective porosity of unconfined anisotropic aquifers. Water Resources Res., 3, pp. 1059-1071.

DeVries, R. N., and D. C. Kent. 1973. Sensitivity of ground-water flow model to vertical variability of aquifer constants. Water Resources Research, Vol. 9, No. 6, p. 38-41.

Fetter, C. W. Jr. 1980. Applied Hydrogeology. Bell and Howell Company, Columbus, Ohio.

Freeze, R. A. 1967. Program Potev. Ottawa, Department of Energy, Mines and Resources, Inland Waters Branch.

Hantush, M. S. 1960. Modification of the theory of leaky aquifers. J. Geophys. Res., 65, pp. 3713-3725.

Huizing, H. G. 1971. A reconnaissance study of the minerology of sand fractions from East Pakistan sediments and soils. Geoderma, 6, 109-133.

Jacob, C. E. 1940. Flow of ground-water. Engineering Hydraulics, ed. H. Rouse, John Wiley and Sons, New York, pp 321-386.

Johnson, A. I. 1967. Compilation of Specific Yield for various materials. US Geological Survey Water Supply Paper 1662-D.

Johnson, E. E., Inc. 1966. Ground-water and Wells, a reference book for the water well industry, Edward E. Johnson, Inc., St. Paul, Minnesota.

Karim, M. A. 1968. Optimization of water resources for irrigation in Dinajpur and Rangpur, East Pakistan. Master's Thesis in Hydrology, University of Arizona.

Kent, D. C. 1980. Evaluation of aquifer performance and water supply capabilities of alluvial and terrace deposits of the North Fork of the Red River in Beckham, Greer, Kiowa, and Jackson Counties, Oklahoma. Oklahoma State University.

Kent, D. C., Naney, J. W., and B. B. Barnes. 1973. An approach to computerization of geohydrologic data. Ground-water, Vol. 11, No. 4, p. 30-42.

Knowles, T. R., Claborn, B. J., and D. M. Wells. 1972. A computerized procedure to determine aquifer characteristics: Water Resources Centre 72-5, Texas Tech University, Lubbock, Texas, p. 103.

Mawla, G. 1968. The relationship of ground-water to alluvium to Dinajpur and Rangpur, East Pakistan. Master's Thesis in Geology, University of Arizona.

Mayboom, P. 1961. Estimating ground-water recharge from stream hydrographs. J. Hydrology, 2, pp. 1203-1214.

Morgan, J. P. 1959. Quaternary geology of the Bengal Basin, East Pakistan and India. Bull. Geol. Soc. Amer., 70, 319-342.

Pettyjohn, A. Wayne, and R., Henning. 1979. Preliminary estimate of ground-water recharge rates, related streamflow and water quality in Ohio, State of Ohio Water Resources Centre, The Ohio State University.

Prickett, T. A. 1965. Type curve solution to aquifer tests under water table conditions. Groundwater, 3, No. 3, pp. 5-14.

Remson, I., Hornberger, G. M., and F. J., Molz. 1971. Numerical methods in subsurface hydrology. New York, Wiley-Interscience, p. 389.

Schicht, R. J., and W. C. Walton. 1962. Hydrologic budgets for three small watersheds in Illinois, Illinois State Water Survey, Urbana.

Sir M. MacDonald and Partners. 1977. Report on aquifer conditions and economic tubewell depths. Paper 10. BADC/IDA Tubewell Project.

Soil Survey of Bangladesh. 1973. Reconnaissance soil surveys of Rajshahi, Pabna, Bogra, Rangpur and Dinajpur Districts, 6 Volumes.

Thornthwaite, C. W. 1948. An approach towards a rational classification of climate. Geography Review. Vol. 38, No. 1, pp. 45-94.

Todd, D. K. 1959. Groundwater Hydrology. John Wiley & Sons, New York.

Trescott, P. C., and G. F. Pinder. 1973. Iterative digital model for aquifer evaluation: US Geological Survey Open-File Report.

United Nation Food and Agriculture Organization. 1966. Hydrological survey in East Pakistan.

Walton, W. C. 1970. Ground-water resource evaluation. McGraw-Hill Book Company, New York.

APPENDIXES

APPENDIX A

LISTING OF COMPUTER PROGRAM

```
C     ************************************************************************
C                     DIGITAL MODEL FOR AQUIFER SIMULATION
C                       BY P.C. TRESCOTT AND G.F. PINDER
C                                JANUARY, 1974
C                 - FOR U.S. GEOLOGICAL SURVEY USE ONLY -
C     ************************************************************************
C     SUBPROGRAMS AND ENTRIES:
C
C                1. MNPROG                              4. COEF
C                                                          CLAY
C                2. DATAI                                  TRANS
C                      DATAIN                              TCOF
C                      ITER                                ETRATE
C                      MAP                                 STORAGE
C                      NEWPER                              LEAKAGE
C                                                          SOURCE
C                3. COMPUT                                 WELLHD
C                      NEWSTP
C                      NEWIT1                           5. CHECKI
C                      NEWITO                              CHECK
C                      ROW                                 CWRITE
C                      COLUMN
C                      STEADY                           6. PRNTAI
C                      OUTPUT                              PRNTA
C                      DRON
C                      DRY                              7. BLOCK DATA
C     ************************************************************************
C     MAIN PROGRAM TO DIMENSION DIGITAL MODEL
C     --------------------------------------
C     SPECIFICATIONS:
C*COM01*
C*COM02*
C*COM03*
C*COM04*
C*COM05*
C*COM06*
      COMMON / PLTCO / PHE, KEEP, STRT, ISUR, BOTTOM, TOP, LAND, RIVER
     * , M, S, SY, PERM, T, TR, TC, RATE, Q, QRE, WELLD, WELL, RRRR9
     * , KKLAY, TLAY1, TLAY, RRRR8
C*DIM01*
      DIMENSION BE(60), G(60), TEMP(60), PHI(50,60),
     1  KEEP(50,60), PHE(50,60), STRT(50,60), ISUR(50,60), T(50,60),
     2  TR(50,60), TC(50,60), S(50,60), DDN(60), DELX(60), DELY(60),
     3  PERM(50,60), BOTTOM(50,60), SY(50,60),
     4  RATE(50,60), RIVER(50,60), M(50,60), Q(50,60),
     5  TOP(50,60), LAND(50,60), QRE(50,60), WELL(50,60), WELLD(50,60),
     *  KKLAY(50,60), TLAY1(50,60), TLAY(50,60,8),
     5  WR(10), NWR(10,2)
C*DIM02*
      DIMENSION RRRR(50,60,40), RRRR9(50,60,5), RRRR8(50,60,5)
      DIMENSION IIII(50, 60, 40)
C*TYP01*
      REAL *4KEEP,M,ISUR,LAND
C*TYP02*
      REAL *8PHI,G,BE,TEMP
C*TYP03*
C*DTA01*
      EQUIVALENCE (PHE, RRRR, IIII)
C     ---INITIALIZE PARAMETERS DEFINING ARRAY SIZE---
      DATA  IZ,IP,IR,IC,IL,IQ,IW,IH  /50,50,50,50,50,50,50,10/
      DATA  JZ,JP,JR,JC,JL,JQ,JW,IMAX/60,60,60,60,60,60,60,60/
C*DTA02*
C*DTA03*
      DATA MNR, MNV / 40, 30 /, MLAY / 7 /
C*END00*
C
      WRITE(6,100)
  100 FORMAT(' **********','PROGRAM DIM  IS EXECUTED','**********')
 9999 FORMAT('      GWMOL    5/25/79 MOD 20, 21, 25F, 26, 27A')
      WRITE(6, 9999)
C     ---PASS DIMENSIONS OF PROBLEM TO SUBROUTINES---
      CALL DATAI(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RATE,M,RIV
     1ER,WELL,SY,TOP,LAND,PERM,BOTTOM,QRE,IZ,JZ,IR,JR,IW,JW,IP,JP,IL,JL,
     2IC,JC,IQ,JQ,WR,NWR,IH,WELLD, KKLAY,TLAY1,TLAY,MLAY)
      CALL COMPUT(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,DDN,G,BE,TE
     1MP,IMAX,IZ,JZ,TOP,IC,JC,BOTTOM,IP,JP,WELL,IW,JW, KKLAY,TLAY1,TLAY
     2 ,MLAY, PERM, SY)
      CALL COEF(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RATE,M,RIVE
```

```
     1R,WELL,SY,TOP,LAND,PERM,BOTTOM,GRE,IZ,JZ,IR,JR,IW,JW,IP,JP,IL,JL,I
     2C,JC,IQ,JQ,WR,NWR,IH, KKLAY,TLAY1,TLAY,MLAY)
      CALL CHECKI(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RATE,M,RI
     1VER,WELL,SY,TOP,LAND,PERM,BOTTOM,GRE,IZ,JZ,IR,JR,IW,JW,IP,JP,IL,JL
     2,IC,JC,IQ,JQ)
      CALL PRNTAI(PHI,ISUR,S,DELX,DELY,WELL,IZ,JZ,IW,JW)
      CALL PLTAI(6, RRRR, IZ, JZ, MNR, MNV, DELX, DELY, PHI, 1, IIII)
      WRITE(6, 9998) IZ, JZ
 9998 FORMAT('OMAIN DIMENSIONS:', 2I5)
C
C     ---START COMPUTATIONS---
      CALL MNPROG
      STOP
      END



      SUBROUTINE MNPROG
C     -----------------
C
C     SPECIFICATIONS:
C*CUM01*
      COMMON /DARRAY/ RHOP(20),CHK(10),VF4(11)
C*CUM02*
      COMMON /DPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH
C*CUM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQC,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDON,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
C*CUM04*
C*CUM05*
C*CUM06*
      COMMON / PLTC / LPLT,MPLT,IPLT,IRUN,IPER,ISTP,ISTPP,ITRY,IVAP
     * , TI, TPI
C*DIM01*
C*DIM02*
C*TYP01*
C*TYP02*
      REAL *8DBLE,RHOP,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,ER
     1OR,LEAK,RECH
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C*DTA01*
C*DTA02*
C*DTA03*
C*END00*
C     ..................................................................
C     ---READ AND WRITE DATA FOR A NEW PROBLEM---
   10 IPER = 0
      ISTP = 0
      ISTPP = 0
      ITRY = 0
      TPI = 0.
      CALL DATAIN
C
C     ---INITIALIZE TRANSMISSIVITY VALUES IN WATER TABLE PROBLEM---
C     IF(WATER.EQ.CHK(2)) CALL TRANS -- INITIAL TRANS. NOW CALC. IN
C                             SUBROUTINE DATAIN (REV. BY D.B. SAPIK).
C
C     ---COMPUTE ITERATION PARAMETERS---
      CALL ITER
C
C     ---INITIALIZE PARAMETERS FOR ALPHAMERIC MAP---
      IF (CONTR.EQ.CHK(3)) CALL MAP
C
C     ---COMPUTE T COEFFICIENTS FOR ARTESIAN PROBLEM---
      IF (WATER.NE.CHK(2)) CALL TCOF
      IPLT = -1
      IRUN = IRUN + 1
      IF(IPLT .GE. LPLT) CALL PLTX
C
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-
      KSTEP=0
   20 CALL NEWPER
```

```
C
      KT=0
      IFINAL=0
      IERR=0
      IPLT = -2
      IPER = KP
      ISTPP = 0
      ITRY = 0
      IF(IPLT .GE. LPLT) CALL PLTX
C
C     ---START NEW TIME STEP COMPUTATIONS---
   30 CALL NEWSTP
      IPLT = -3
      ISTP = ISTP + 1
      ISTPP = ISTPP + 1
      ITRY = 0
      TI = SUM
      TPI = SUMP
      IF(IPLT .GE. LPLT) CALL PLTX
C
C     ---COMPUTE LEAKAGE COEFFICIENTS---
      IF (LEAK.EQ.CHK(9)) CALL CLAY
C
C     ---START NEW ITERATION IF MAXIMUM NO. ITERATIONS NOT EXCEEDED ---
      CALL NEWIT0
      GO TO 50
   40 CALL NEWIT1
C
   50 IPLT = -4
      ITRY = ITRY + 1
      IF(IPLT .GE. LPLT) CALL PLTX
C     ---COMPUTE TRANSMISSIVITY IN WT OR WT-ARTESIAN CONVERSION PROBLEM-
      IF(WATER .NE. CHK(2) ) GO TO 70
      CALL TRANS
      IF (IERR.GT.0) GO TO 80
C
C     ---COMPUTE T COEFFICIENTS IN WATER TABLE PROBLEM---
   60 CALL TCOF
C
C     ---COMPUTE IMPLICITLY ALONG ROWS---
   70 CALL ROW
C
C     ---COMPUTE IMPLICITLY ALONG COLUMNS---
      CALL COLUMN
C
      IPLT = 4
      IF(IPLT .LE. MPLT) CALL PLTX
C     ---IF SOLUTION NOT OBTAINED START NEW ITERATION---
      IF (TEST.EQ.1.) GO TO 40
C
C     ---CHECK FOR STEADY STATE---
      CALL STEADY
C
      IPLT = 3
      IF(IPLT .LE. MPLT) CALL PLTX
C     ---PRINT OUTPUT AT DESIGNATED TIME STEPS---
      IF (MOD(KT,KTH).NE.0.AND.IFINAL.NE.1) GO TO 30
      CALL OUTPUT
      IF (IFINAL.NE.1) GO TO 30
C
      IPLT = 2
      IF(IPLT .LE. MPLT) CALL PLTX
C     ---CHECK FOR NEW PUMPING PERIOD---
      IF (KP.LT.NPER) GO TO 20
C
      IPLT = 1
      IF(IPLT .LE. MPLT) CALL PLTX
C        PRINT HYDROGRAPHS ---
   80 IF (KT.LE.0) GO TO 90
      CALL  DRY
      IF (KP.NE.NPER) GO TO 90
C
C     ---CHECK FOR NEW PROBLEM---
      READ (R,100,END=90)  NEXT
      IF (NEXT.EQ.0) GO TO 10
   90 RETURN
C
C     FORMAT:
```

```
C
C
  100 FORMAT (I10)
      END


      SUBROUTINE DATAI(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RATE
     1,M,RIVER,WELL,SY,TOP,LAND,PERM,BOTTOM,QRE,IZ,JZ,IR,JR,IW,JW,IP,JP,
     2IL,JL,IC,JC,IQ,JQ,WR,NWR,IH,WELLD,KKLAY,TLAY1,TLAY,MLAY)
C     ------------------------------------------------------------------
C
C  1. REVISIONS BY D.B. SAPIK (USGS - WRD, OKLA. CITY) - DEC. 1975.
C     A. EDIT INPUT DATA.
C     B. READ PROGRAM OPTION CARD.
C     C. READ HYDROGRAPH OPTION CARD.
C     D. READ THE FOLLOWING PARAMETERS FROM DISK FILE:
C           INITIAL MASS BALANCE PARAMETERS
C           HEAD (PHI)
C           HEAD (STRT)
C           RECHARGE
C     E. WRITE THE FOLLOWING PARAMETERS ON DISK FILE:
C           LEAKAGE RATES COMPUTED AT END OF FIRST TIME STEP
C           HEAD AT THE END OF A PUMPING PERIOD
C           CUMULATIVE MASS BALANCE PARAMETERS AT THE END OF SIMULATION
C           HEAD (PHI) AT THE END OF SIMULATION
C
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
C*COM01*
      COMMON /DARRAY/ RHOP(20), CHK(10), VF4(11)
C*COM02*
      COMMON /DPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH
C*COM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDCN,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
     * , WELBOT, WELTOP, SYB
C*COM04*
      COMMON /CK/ ETFLXT, QRET, QRETN, FLUXT, FLXNT, CHST, CHDT
     * , CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
C*COM05*
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(4),XN1,SYM(28),PRNT(122),BLA
     1NK(50),DIGIT(122),VF1(6),VF2(6),VF3(7),NA(4),XN(100),YN(13),XSF,NX
     2D,WIDTH,SPACNG,N1,N2,N3,N4,N6,N8,NC
C*COM06*
      COMMON / COPYC / ICPYW, KCPY
      COMMON / LAYER / SATL(20), PRMSY(10), SYPRM(10), SYC(10)
     * , NLAY, NLAYO, MSY, NSY, NSYO
C NOTE MSY MUST BE SET TO DIMENSION OF SY LIST AT END OF ENTRY DATAI
C*DIM01*
      DIMENSION PHI(IZ,JZ), KEEP(IZ,JZ), PHE(IZ,JZ), STRT(IZ,JZ), ISUR(I
     1Z,JZ), T(IZ,JZ), TR(IZ,JZ), TC(IZ,JZ), S(IZ,JZ), DELX(JZ), DELY(IZ
     2), Q(IR,JR), RATE(IR,JR), M(IR,JR), RIVER(IR,JR), WELL(IW,JW), SY(
     3IP,JP), TOP(IC,JC), LAND(IL,JL), PERM(IP,JP), BOTTOM(IP,JP), QRE(I
     4Q,JQ), WR(IH), NWR(IH,2), WELLD(IW, JW)
     5 , KKLAY(IZ,JZ),TLAY1(IZ,JZ),TLAY(IZ,JZ,MLAY)
C*DIM02*
      DIMENSION  SETID(20)
C*TYP01*
      REAL *4KEEP,M,ISUR,LAND
C*TYP02*
      REAL *8PHI,DBLE,RHOP,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONT
     1R,EROR,XLABEL,YLABEL,TITLE,XN1,HEADNG(16),MESUR,LEAK,RECH
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C*DTA01*
      DATA   SETID/4HSTRT, 1HS,     1HT,     4HPERM, 4HBOTT, 2HSY,
     1             3HTOP,  4HRATE, 4HRIVE, 1HM,     4HLAND, 3HQRE,
     2             4HDELX, 4HDELY, 4HWELL, 5*1H /
C*DTA02*
      DATA   EPS1, EPS2/1.0E-10, 1.0E-20/
      DATA THIN / .02 /
```

```
C*DTA03*
C*END00*
 9999 FORMAT(' SUB DATAI  12/19/73 MOD 26, 27A')
      WRITE(P, 9999)
      KCPY = 3
      KPLC = 3
      MSY = 10
      RETURN
C     *********************
      ENTRY DATAIN
C     *********************
C
C     ---DEFINE DISK (OR TAPE) FILES---
C
C     IFILE(1) = FILE NO. FOR WRITING LEAKAGE RATES COMPUTED AT THE END
C                OF THE FIRST TIME STEP (IF ISVFLX = 1).
C     IFILE(2) = FILE NO. FOR READING RECHARGE RATES AT THE START OF
C                SIMULATION (IF IRDRCH = 1).
C     IFILE(3) = FILE NO. FOR WRITING HEAD (SINGLE PRECISION PHI) AT THE
C                END OF A PUMPING PERIOD (IF ISVHED = 1).
C     IFILE(4) = FILE NO. FOR READING STARTING HEAD (STRT) AT THE
C                BEGINNING OF SIMULATION (IF IRDHED = 1).
C     IFILE(5) = FILE NO. FOR WRITING CUMULATIVE MASS BALANCE PARAMETERS
C                AND FINAL HEAD (PHI) AT THE END OF SIMULATION
C                (IF PNCH = 'PUNCH').
C     IFILE(6) = FILE NO. FOR READING CUMULATIVE MASS BALANCE PARAMETERS
C                AND HEAD (PHI) SAVED AT THE END OF A PREVIOUS
C                SIMULATION (IF IRDINI = 1).
C     IFILE(7) = FILE NO. FOR WRITING HYDROGRAPHS OF HEAD AT SELECTED
C                NODES (IF NHYDR .GT. 0).
C
      IFILE(1)=11
      IFILE(2)=12
      IFILE(3)=13
      IFILE(4)=14
      IFILE(5)=15
      IFILE(6)=16
      IFILE(7)=17
C                                          READ PRINT/PLOT PARAMETERS
      READ(R, 9000) ICPY, ICPY2, ICPYW, K1, MPRN, KPLT, K2
      IF(K1 .NE. 0) KCPY = K1
      IF(K2 .NE. 0) KPLC = K2
      IF(ICPY .GE. 1) WRITE(KCPY, 9012) ICPY, ICPY2, ICPYW, K1, MPRN
     * , KPLT, K2
      IF(ICPY .GE. 2) WRITE(KCPY, 9013) ICPY, ICPY2, ICPYW, KCPY, MPRN
     * , KPLT, KPLC
C
C     ---READ GROUP I DATA---
      READ (R,810) HEADNG
      IF(ICPY .GE. 1) WRITE(KCPY, 9020) HEADNG
      WRITE (P,800) HEADNG
C
      READ (R,940) WATER,LEAK,CONVRT,EVAP,RECH,CHCK,PNCH,NUM,HEAD
      IF(ICPY .GE. 1) WRITE(KCPY, 9030) WATER, LEAK, CONVRT, EVAP, RECH
     * , CHCK, PNCH, NUM, HEAD
C
      READ (R,850) CONTR,SCALE,DINCH,SPACNG,MESUR
      IF(ICPY .GE. 1) WRITE(KCPY, 9040) CONTR,SCALE,DINCH,SPACNG,MESUR
C
      IF(CONVRT.EQ.CHK(7)) WATER=CHK(2)
      WRITE (P,1100) WATER,LEAK,CONVRT,EVAP,RECH,CHCK,PNCH,NUM,HEAD,CONT
     1R
      IF (CONTR.EQ.CHK(3)) WRITE (P,1110) SCALE,MESUR,MESUF,DINCH,SPACNG
C
C     ..............................................................
C
C     ---READ PROGRAM OPTION CARD---
C     READ (R,841) KEDIT, NHYDR, MBRES, ISVFLX, IRDINI, IRDHED, IRDRCH
C       CARD DELETED
      IF(ICPY .GE. 1) WRITE(KCPY, 9050) KEDIT, NHYDR, MBRES, ISVFLX
     * , IRDINI, IRDHED, IRDRCH
C
      IF (KEDIT.LT.0 .OR. KEDIT.GT.19) KEDIT=0
      KED=KEDIT
      IF (KEDIT.LT.12 .AND. KEDIT.GT.5) KED=KED-6
      MESSG=1
      IF (KED.EQ.4 .OR. KED.EQ.5) MESSG=0
      IF(ICPY .GE. 2) WRITE(KCPY, 9050) KEDIT, NHYDR, MBRES, ISVFLX
```

```
     * , IRDINI, IRDHED, IRORCH
C
C     ...........................................................................
C
C     --- READ HYDROGRAPH OPTION CARD ---
      IF(NHYDR .EQ. 0) GO TO 6
C     --- READ NODE COORDINATES FOR A MAXIMUM OF 13 HYDROGRAPHS ---
      IF (NHYDR.GT.13) NHYDR=13
      IF(NHYDR .LT. -13) NHYDR = -13
      NHYDR2 = IABS(NHYDR)
      READ(R, 341) (IHYDR(J), JHYDR(J), J = 1, NHYDR2)
      IF(ICPY .GE. 1 .AND. NHYDR .GE. 0) WRITE(KCPY, 9060)
     * (IHYDR(J), JHYDR(J), J = 1, NHYDR2)
      IF(ICPY .GE. 1 .AND. NHYDR .LT. 0) WRITE(KCPY, 9062)
     * (IHYDR(J), JHYDR(J), J = 1, NHYDR2)
      JM1=IFILE(7)
      IF(NHYDR .GT. 0) REWIND JM1
    6 CONTINUE
C
C     ...........................................................................
C
C     ---READ GROUP II DATA---
      READ (R,340) NPER,DIML,DIMW,KTH,LENGTH,ERR,HMAX,ITMAX,EROR,SS,QET,
     1      ETDIST,FACTX,FACTY,SSRIV
      IF(ICPY .GE. 1) WRITE(KCPY, 9070) NPER, DIML, DIMW, KTH, LENGTH
     * , ERR, HMAX, ITMAX, EROR, SS, QET, ETDIST, FACTX, FACTY, SSRIV
      IF (ITMAX.GT.100) ITMAX=100
      IF (FACTX.LE.0.0) FACTX=1.0
      IF (FACTY.LE.0.0) FACTY=1.0
      IF (ETDIST.LE.0.0) ETDIST=1.0
      IF(ICPY .GE. 2) WRITE(KCPY, 9072) NPER, DIML, DIMW, KTH, LENGTH
     * , ERR, HMAX, ITMAX, EROR, SS, QET, ETDIST, FACTX, FACTY, SSRIV
      INO1=DIML-1
      JNO1=DIMW-1
      WRITE (P,380) NPER,DIML,DIMW,KTH,ERR,ITMAX,EROR,SS,SSRIV,QET,
     1    ETDIST,FACTX,FACTY
      IF(DIML.LE.IZ .AND. DIMW.LE.JZ) GO TO 7
      WRITE(P, 7007)
      GO TO 465
C
C
C     ---READ CUMULATIVE MASS BALANCE PARAMETERS FOR CONTINUATION OF
C        A PREVIOUS SUMULATION (GROUP II DATA)---
    7 IF(IRDINI .GT. 0) GO TO 8
      READ(R, 1090) SUM, SUMP, ETFLXT, QRET, QRETN, FLUXT, FLXNT
     * , CHST, CHDT, CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
      IF(ICPY .GE. 1) WRITE(KCPY, 9080) SUM, SUMP, PUMPT, CFLXPT, QRET
     * , QRETN, CHST, CHDT, CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
      GO TO 20
    8 JM1=IFILE(5)
      READ(JM1)        SUM, SUMP, ETFLXT, QRET, QRETN, FLUXT, FLXNT
     * , CHST, CHDT, CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
      IF(ICPY .GE. 2) WRITE(KCPY, 9080) SUM, SUMP, PUMPT, CFLXPT, QRET
     * , QRETN, CHST, CHDT, CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
      WRITE (P,830) SUM
C
C     ...........................................................................
C
C     ---READ GROUP III DATA---
C
C     ..........PHI (HEAD AT THE END OF A PREVIOUS SIMULATION)..........
      DO 10 I=1,DIML
      READ (JM1) (PHI(I,J), J=1,DIMW)
   10 WRITE (P,912) I, (PHI(I,J), J=1,DIMW)
C
C     ............................. STRT (STARTING HEAD) ............
C        (ADD 20,000 TO STARTING HEAD AT CONSTANT FLUX BOUNDARY NODES)
   20 IERR=0
      IERR2=0
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, NOWRNH
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN,NOWRNH
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN,NOWRNH
      SET=SETID(1)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 660) FACT
      A2=FACT
```

```
      LIST=0
      JM1=IFILE(4)
      IF (IRDHED.EQ.1) IVAR=2
      IF (IVAR.GT.0 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,820)
      DO 60 I=1,DIML
      IF (IVAR.EQ.1) READ (R,950) (STRT(I,J), J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9950)   (STRT(I,J),J=1,DIMW)
      IF (IRDHED.EQ.1)  READ (JM1) (STRT(I,J), J=1,DIMW)
      DO 50 J=1,DIMW
      IF (IVAR.GT.0) A2=STRT(I,J)*FACT
      ISUR(I,J)=A2
      IF(A2.LT.2.0E4) GO TO 48
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(I.GT.1 .AND. I.LT.DIML .AND. J.GT.1 .AND. J.LT.DIMW) GO TO 47
      IF(MESSG.EQ.1) WRITE(P,7010) I,J
      IERR2=1
   47 A2=A2-2.0E4
   48 IF(A2.GT.EPS1) GO TO 49
      WRITE(P,7012) I, J
      IERR=1
   49 STRT(I,J)=A2
      IF (SUM.LT.EPS2) PHI(I,J)=A2
   50 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9952)    (STRT(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,912) I, (ISUR(I,J), J=1,DIMW)
   60 CONTINUE
C WRITE MOVED BEFORE LOOP
C     ............................... S (STORAGE COEFFICIENT) .........
      READ (R,842) SETNAM, FACT, IVAR, IPRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN
      SET=SETID(2)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 670) FACT
      A2=FACT
      LIST=0
      IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,930)
      DO 90 I=1,DIML
      IF (IVAR.EQ.1) READ (R,860) (S(I,J),J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9860)      (S(I,J),J=1,DIMW)
      DO 80 J=1,DIMW
      IF(IVAR.EQ.1) A2=S(I,J)*FACT
      S(I,J)=A2
      IF(A2.GE.0.0) GO TO 80
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(I.GT.1 .AND. I.LT.DIML .AND. J.GT.1 .AND. J.LT.DIMW) GO TO 72
      IF(MESSG.EQ.1) WRITE(P,7015) I,J
      IERR2=1
      GO TO 80
   72 IF(ISUR(I,J).LT.2.0E4) GO TO 80
      IF(MESSG.EQ.1) WRITE(P,7020) I,J
      IERR2=1
   80 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9864)       (S(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,916) I, (S(I,J), J=1,DIMW)
   90 CONTINUE
C     .......................... T    (TRANSMISSIVITY), CR      ........
C WRITE MOVED BEFORE LOOP
C     .......................... PERM (HYDRAULIC CONDUCTIVITY) ........
C
C                                (READ TRANSMISSIVITY IF IPERM=0, OR
C                            READ HYDRAULIC CONDUCTIVITY IF IPERM=1)
C
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, IPERM, NLAY, PSCAL
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, IPERM
     * , NLAY, PSCAL
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(MPRN .LT. -1) IPRN = 2
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, IPERM
     * , NLAY, PSCAL
      SET=SETID(3)
      IF(IPERM.EQ.1) SET=SETID(4)
      IF(SETNAM.NE.SET) GO TO 485
```

```
      IF(PSCAL .EQ. 0.) PSCAL = 1.
      FACT1 = FACT
      FACT = FACT*PSCAL
      IF(IVAR .EQ. 0 .AND. IPERM .NE. 1) WRITE(P, 720) FACT
      IF(IVAR .EQ. 0 .AND. IPERM .EQ. 1) WRITE(P, 730) FACT
      LIST2=0
      IVPRM = IVAR
      IF (IVAR.NE.1 .OR. IPRN.EQ.1) GO TO 132
      LIST2=1
      IF(IPERM.NE.1) WRITE(P,390)
      IF(IPERM.EQ.1) WRITE(P,960)
  132 DO 160 I=1,DIML
      IF(IVAR.EQ.1) READ(R,850) (T(I,J), J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1) WRITE(KCPY,9850)      (T(I,J),J=1,DIMW)
      DO 150 J=1,DIMW
      IF(MLAY .GT. 1) KKLAY(I, J) = 0
      TR(I,J)=0.0
      TC(I,J)=0.0
      A2=FACT
      IF(IVAR.EQ.1) A2=T(I,J)*FACT
      IF(I.EQ.1 .OR. I.EQ.DIML .OR. J.EQ.1 .OR. J.EQ.DIMW) GO TO 148
      IF(A2.GT.0.0) GO TO 150
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(S(I,J).GE.0.0 .AND. ISUR(I,J).LT.2.0E4) GO TO 150
      WRITE(P,7025) I, J
      IERR=1
  148 A2=0.0
  150 T(I,J)=A2
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9864)      (T(I,J), J=1,DIMW)
      IF(LIST2.EQ.1) WRITE(P,900) I, (T(I,J), J=1,DIMW)
  160 CONTINUE
C WRITE MOVED BEFORE LOOP
C WRITE MOVED BEFORE LOOP
      IF(IPRN.EQ.2) LIST2=1
      NNX=1
      IF(LIST2.EQ.1) NNX=0
C
C . . . . . . . . . . . . . . . . . . . . . . TLAY . . . . . . . . . . . . . .
C                                            LAYER T OR PERM
      SAT = 0.
      IF(NLAY .EQ. 0) GO TO 179
      IF(WATER .NE. CHK(2) ) WRITE(P, 7100)
      IF(WATER .NE. CHK(2) ) GO TO 486
      NLAYO = NLAY
      IF(NLAY .LT. 0) NLAY = -NLAY
      IF(NLAY .GT. MLAY) WRITE(P, 7007)
      IF(NLAY .GT. MLAY) GO TO 486
      KLAY1 = 0
      IVAR1 = IVAR
      IPRN1 = IPRN
      IPRM1 = IPERM
      SR = 1.
      SDD = 10.
      SDD1 = 0.
      SD1 = 0.
      KSET1 = IVAR
      DO 173 KLAY = 1, NLAY
      SAT1 = SAT
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, IPRML, SD
      IF(ICPY .GE. 1) WRITE(KCPY, 9420)
     * SETNAM, FACT, IVAR, IPRN, IPRML, SD
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IPRMLO = IPRML
      IF(IPRML .GE. 2) IPRML = IPRML - 2
      IF(FACT .NE. 0) GO TO 172
      FACT = FACT1
      IVAR = IVAR1
      IPRN = IPRN1
      IPRML = IPRM1
 172  CONTINUE
      FACT1 = FACT
      IVAR1 = IVAR
      IPRN1 = IPRN
      IPRM1 = IPRML
      FACT = FACT*PSCAL
C                                            SAT, SD SET TO MATCH S2, S
C                                            OF PROGRAM (G808) PSEUDO-
```

```
C                                                   HYDROGRAPH GENERATION
C                                                   IF IPRML LT 2,
C                                                      SD, PERM/T FOR INTERVAL
C                                                   IF IPRML GE 2,
C                                                      SD, PERM/T CUMULATIVE
      IF(SD .LE. 0.) SD = SD1 + SDD*SR
      SDD = SD - SD1
      IF(SDD1 .GT. 0.) SR = SDD/SDD1
      SD1 = SD
      SDD1 = SDD
      IF(ICPY .GE. 2) WRITE(KCPY, 9422)
     * SETNAM, FACT, IVAR, IPRN, IPRML, SAT
      SET = SETID(3)
      IF(IPRML .EQ. 1) SET = SETID(4)
      IF(SETNAM .NE. SET) GO TO 485
      SAT = SD
      IF(IPRMLO .LT. 2) SAT = SAT1 + SAT
      SATL(KLAY) = SAT
      SD = SAT - SAT1
      IF(IVAR .EQ. 0 .AND. IPRML .NE. 1) WRITE(P, 720) FACT
      IF(IVAR .EQ. 0 .AND. IPRML .EQ. 1) WRITE(P, 730) FACT
      LIST = 0
      IF(IVAR .EQ. 1 .AND. IPRN .NE. 1) LIST = 1
      IF(LIST .EQ. 1 .AND. IPRML .NE. 1) WRITE(P, 860)
      IF(LIST .EQ. 1 .AND. IPRML .EQ. 1) WRITE(P, 960)
      IF(IPRML .EQ. 1) FACT = FACT*SD
      KSET = 0
      IF(KSET1 .EQ. 0 .AND. IVAR .EQ. 1) KSET = 1
      DO 177 I = 1, DIML
C                                                   BORROW DELX FOR TEMP
      IF(IVAR .EQ. 1) READ(R, 860) (DELX(J), J = 1, DIMW)
      IF(ICPY2 .GE. 1 .AND. IVAR .EQ. 1) WRITE(KCPY, 9860)
     * (DELX(J), J = 1, DIMW)
      DO 175 J = 1, DIMW
      A2 = FACT
      IF(IVAR .EQ. 1) A2 = DELX(J)*FACT
      IF(KLAY1 .GT. 0) A2 = A2 + TLAY(I, J, KLAY1)
      IF(A2 .EQ. 0) GO TO 173
      IF(T(I, J) .GT. 0.) GO TO 173
      A2 = 0.
      IF(IVAR .NE. 1) GO TO 173
      WRITE(P, 7105) I, J
      IERR = 1
  173 CONTINUE
      TLAY(I, J, KLAY) = A2
      IF(KSET .GT. 0 .AND. A2 .EQ. 0.) T(I, J) = 0.
  175 CONTINUE
      IF(ICPY2 .GE. 2 - IVAR) WRITE(KCPY, 9864)
     * (TLAY(I, J, KLAY), J = 1, DIMW)
      IF(LIST .EQ. 1) WRITE(P, 900) I, (TLAY(I, J, KLAY), J = 1, DIMW)
  177 CONTINUE
      KLAY1 = 0
      IF(IPRMLO .LT. 2) KLAY1 = KLAY
      IF(KSET .GT. 0) KSET1 = 1
  178 CONTINUE
  179 CONTINUE
      SAT1 = SAT
      SAT = SAT1
C     ****************************** BOTTOM (AQ. BOTTOM EL.) *********
C                 (CODE ISATHK=1 TO SUPPRESS PRINTING A SAT. THKNS. MAP)
      ISATHK=1
      NO=0
      KTOP=0
      IF(WATER.NE.CHK(2).AND.IPERM.EQ.0) GO TO 224
      READ (R,842) SETNAM, FACT, IVAR, IPRN, ISATHK
     * , WELBOT
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, ISATHK
     * , WELBOT
      IF(MPRN .GT. 0) ISATHK = 1
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(MPRN .LT. -1) ISATHK = 0
      IF(WELBOT .EQ. 0.) WELBOT = 5.
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, ISATHK
     * , WELBOT
      SET=SETID(5)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 740) FACT
```

```
      A2=FACT
      LIST=0
      IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,970)
      DO 190 I=1,DIML
      IF (IVAR.EQ.1) READ (R,860) (BOTTOM(I,J),J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9860)(BOTTOM(I,J),J=1,DIMW)
      DO 180 J=1,DIMW
      IF(IVAR.EQ.1) A2=BOTTOM(I,J)*FACT
      BOTTOM(I,J)=A2
  180 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9862) (BOTTOM(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,912) I, (BOTTOM(I,J), J=1,DIMW)
  190 CONTINUE
C WRITE MOVED BEFORE LOOP
C     ............................... SY (SPECIFIC YIELD) ............
      IF(WATER.NE.CHK(2)) GO TO 224
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, NSY, SYR, SYB
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
     * , NSY, SYR, SYB
      IF(IPRN .GT. 1) IPRN = 1
      IF(IPRN .LT. 0) IPRN = 0
      IF(SYR .EQ. 0.) SYR = 0.1
      IF(SYR .LT. 0.) SYR = 0.
      FACT2 = SYR*ALOG10(PSCAL)
      IF(SYB .EQ. 0) SYB = .01
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN
     * , NSY, SYR, SYB
      SET=SETID(6)
      IF(SETNAM.NE.SET) GO TO 485
C OPTION MOVED TO 13352
      IF(IVAR .EQ. 0) FACT = FACT + FACT2
      IF(IVAR .EQ. 0) WRITE(P, 680) FACT
C INITIALIZE MOVED TO 13685
      LIST=0
      IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,1120)
      PRMSY(1) = 0.
      IF(NSY .EQ. 0) GO TO 209
      NSY0 = NSY
      IF(NSY .LT. 0) NSY = -NSY
      IF(NSY .EQ. 1) NSY = 2
      READ(R, 840) (PRMSY(I), SYPRM(I), I = 1, NSY)
      IF(ICPY .GE. 1) WRITE(KCPY, 9950) (PRMSY(I), SYPRM(I), I = 1, NSY)
      A2 = PRMSY(1)
      SD = SYPRM(1)
      DO 208 I = 2, NSY
      A3 = A2
      A2 = PRMSY(I)
      SD1 = SD
      SD = SYPRM(I)
      IF(A3 .LE. 0.) WRITE(P, 7110)
      IF(A3 .LE. 0.) GO TO 486
      SYPRM(I) = SD1
      SYD(I) = (SD - SD1)/ALOG10(A2/A3)
  208 CONTINUE
  209 CONTINUE
      A2 = FACT
      DO 220 I = 1, DIML
      IF (IVAR.EQ.1) READ (R,860) (SY(I,J),J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9858)     (SY(I,J),J=1,DIMW)
      DO 210 J=1,DIMW
      IF(IVAR .EQ. 1) A2 = SY(I, J)*FACT + FACT2
      SY(I,J)=A2
  210 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9864)     (SY(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,916) I, (SY(I,J), J=1,DIMW)
  220 CONTINUE
C WRITE MOVED BEFORE LOOP
C     ............................... TOP (AQ. TOP ELEVATION) .........
C
C             VALUES OF TOP ARE REQUIRED FOR SIMULATION OF THE FOLLOWING:
C               1. ARTESIAN AQUIFERS WITH OR WITHOUT LEAKAGE,
C               2. W.T.-ARTESIAN SYSTEMS WITH OR WITHOUT LEAKAGE,
C               3. W.T. AQUIFERS WITH LEAKAGE FROM STREAMS.
C
      IF(CONVRT.NE.CHK(7)) GO TO 254
  224 KTOP=1
```

```
      READ(R, 542) SETNAM, FACT, IVAR, IPRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
      IF(IPRN .GT. 1) IPRN = 1
      IF(IPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN
      SET=SETID(7)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 750) FACT
      A2=FACT
      LIST=0
      IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,980)
      DO 250 I=1,DIML
C     IF (IVAR.EQ.1) READ (R,360) (TOP(I,J),J=1,DIMW)
      IF (IVAR.EQ.1) READ (R,950) (TOP(I,J),J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9950)    (TOP(I,J),J=1,DIMW)
      DO 240 J=1,DIMW
      IF(IVAR.EQ.1) A2=TOP(I,J)*FACT
      TOP(I,J)=A2
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(SNGL(PHI(I,J)).GT.A2 .OR. WATER.EQ.CHK(2)) GO TO 240
      WRITE(P,7030) I, J
      IERR=1
  240 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9952)     (TOP(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,912) I, (TOP(I,J), J=1,DIMW)
  250 CONTINUE
C WRITE MOVED BEFORE LOOP
C
C
C     ...CALCULATE TRANSMISSIVITY FOR WATER-TABLE OR ARTESIAN AQUIFER...
C                     (NEW PROCEDURE - ADDED BY D.B. SAPIK)
  254 IF (IPERM.EQ.0) GO TO 168
      IF(LIST2.EQ.1) WRITE(P,892)
      DO 164 I=1,DIML
      DO 162 J=1,DIMW
      A2=PHI(I,J)
      A3 = ABS(T(I, J) )
      IF(A3 .LE. EPS1) GO TO 159
      IF(A2 .GE. BOTTOM(I,J)+THIN) GO TO 256
      IF(NOWRNH .LE. 0) WRITE(P, 7032) I, J
      A2 = BOTTOM(I, J) + THIN
      IERR2 = 1
  256 IF(KTOP.NE.1) GO TO 161
      A2 = AMIN1(A2, TOP(I, J) )
      IF(A2 .GT. BOTTOM(I, J) ) GO TO 161
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
C      CARD DELETED
      IF(MESSG .EQ. 1 .AND. IVPRM .EQ. 1) WRITE(P, 7034) I, J
      IERR2=1
      ND=3
  159 A3=0.0
  161 IF(WATER.EQ.CHK(2)) PERM(I,J)=A3
      HMIN = A2 - BOTTOM(I, J)
      PHE(I, J) = HMIN
      T(I,J)=A3*HMIN
      IF(HMIN .GT. SAT) SAT = HMIN
  162 CONTINUE
      IF(LIST2.EQ.1) WRITE(P,900) I, (T(I,J), J=1,DIMW)
  164 CONTINUE
      IF(NNX.EQ.1) WRITE(P,894)
      GO TO 258
C
C
C     .............CALCULATE HYDRAULIC COND. FOR WATER-TABLE AQUIFER...
C                     (NEW PROCEDURE - ADDED BY D.B. SAPIK)
  168 IF(WATER.NE.CHK(2)) GO TO 260
      IF(LIST2.EQ.1) WRITE(P,962)
      DO 196 I=1,DIML
      DO 194 J=1,DIMW
      A2=STRT(I,J)
      TOPA=TOP(I,J)
      IF(KTOP.EQ.1) A2=AMIN1(A2,TOPA)
      HMIN=A2-BOTTOM(I,J)
      PHE(I,J)=HMIN
      A3=ABS(T(I,J))
      IF (HMIN.GT.1.001) GO TO 193
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(A3.LE.EPS1) GO TO 192
      IF(MESSG .EQ. 1 .AND. IVPRM .EQ. 1) WRITE(P, 7036) I, J
```

```
      IERR2=1
      ND=3
  192 T(I,J)=0.0
      A3=0.0
      HMIN=1.0
  193 PERM(I,J)=A3/HMIN
      IF(HMIN .GT. SAT) SAT = HMIN
  194 CONTINUE
      IF(LIST2.EQ.1) WRITE(P,900) I, (PERM(I,J), J=1,DIMW)
  196 CONTINUE
      IF(NNX.EQ.1) WRITE(P,964)
      IF(SUM.GT.0.0) ISATHK=1
C
C     ............ PRINT SATURATED THICKNESS MAP FOR WATER-TABLE OR ....
C     ............                 WATER-TABLE/ARTESIAN AQUIFER     ....
  258 CONTINUE
      IF(.NOT.(ISATHK.EQ.0 .OR. (ND.EQ.3 .AND. MESSG.EQ.1))) GO TO 260
      WRITE(P,974)
      DO 226 I=1,DIML
      WRITE(P,912) I, (PHE(I,J), J=1,DIMW)
  226 CONTINUE
C
C     .................................. RATE (CONF. BED CONDUCTIVITY) ...
  260 IF (LEAK.NE.CHK(9)) GO TO 350
      READ (R,842) SETNAM, FACT, IVAR, IPRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN
      SET=SETID(8)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 690) FACT
      A2=FACT
      LIST=0
      IVRAT = IVAR
      IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,870)
      DO 290 I=1,DIML
      IF (IVAR.EQ.1) READ (R,860) (RATE(I,J),J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9860)  (RATE(I,J),J=1,DIMW)
      DO 280 J=1,DIMW
      Q(I,J)=0.0
      IF(IVAR.EQ.1) A2=RATE(I,J)*FACT
      RATE(I,J)=A2
  280 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9864)    (RATE(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,900) I, (RATE(I,J), J=1,DIMW)
  290 CONTINUE
C WRITE MOVED BEFORE LOOP
C     .................................. RIVER (HEAD IN RIVER) ..........
C                     (CODE ISTEDY=1 TO SET RIVER HEAD EQUAL TO STARTING
C                      AQUIFER HEAD FOR STEADY-STATE SIMULATION.)
      READ (R,842) SETNAM, FACT, IVAR, IPRN, ISTEDY
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, ISTEDY
      IF(MPRN .GT. 1) IPRN = 1
      IF(MPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, ISTEDY
      SET=SETID(9)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0 .AND. ISTEDY .NE. 1) WRITE(P, 700) FACT
      A2=FACT
      IVRIV = IVAR
      LIST=0
      IF ((IVAR.EQ.1 .OR. ISTEDY.EQ.1) .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,1040)
      DO 320 I=1,DIML
      IF (IVAR.EQ.1) READ (R,860) (RIVER(I,J), J=1,DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9860) (RIVER(I,J),J=1,DIMW)
      DO 310 J=1,DIMW
      IF(IVAR.EQ.1) A2=RIVER(I,J)*FACT
      IF(ISTEDY.EQ.1) A2=STRT(I,J)+FACT
      RIVER(I,J)=A2
  310 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9862)   (RIVER(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,912) I, (RIVER(I,J), J=1,DIMW )
  320 CONTINUE
C WRITE MOVED BEFORE LOOP
      IF (ISTEDY.EQ.1) WRITE (P,1041) FACT
```

```
C       ................................ M (CONF. BED THICKNESS) ........
C                    (CODE NEGATIVE VALUES FOR STREAM BED THICKNESS)
        READ(R, 842) SETNAM, FACT, IVAR, IPRN
        IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
        IF(APRN .GT. 1) IPRN = 1
        IF(APRN .LT. 0) IPRN = 0
        IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN
        SET=SETID(10)
        IF(SETNAM.NE.SET) GO TO 435
        IF(IVAR .EQ. 0) WRITE(P, 710) FACT
        A2=FACT
        NO=0
        LIST=0
        IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
        IF (LIST.EQ.1) WRITE(P,1030)
        DO 350 I=1,DIML
        IF (IVAR.EQ.1) READ (R,860) (M(I,J),J=1,DIMW)
        IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9360)      (M(I,J),J=1,DIMW)
        DO 340 J=1,DIMW
        A2 = FACT
        IF(IVAR.EQ.1) A2=M(I,J)*FACT
        IF(IVAR .EQ. 0 .AND. RATE(I, J) .EQ. 0.) A2 = 0.
        M(I,J)=A2
        IF(A2.LT.0.0) NO=1
        IF(IVRAT .EQ. 0 .AND. A2 .EQ. 0.) RATE(I, J) = 0.
        IF(IVRIV .EQ. 0 .AND. A2 .EQ. 0.) RIVER(I, J) = 2E4
C       .....CHECK FOR ERRORS IN SPECIFICATION OF VALUES FOR RATE, M,
C            RIVER, TOP, ISUR, S, AND T.
        A3=0.0
        IF(CONVRT.EQ.CHK(7)) A3=TOP(I,J)
        IF(RATE(I, J) .EQ. 0.) GO TO 340
  333   HMIN=RIVER(I,J)
        IF (ISUR(I,J).GE.2.0E4) GO TO 340
        IF(A2.EQ.0.0 .OR. HMIN.EQ.0.0 .OR. S(I,J).LT.0.0 .OR.
     1  ISUR(I,J).GE.2.0E4 .OR. T(I,J).EQ.0.0) GO TO 338
        IF(A3.EQ.0.0 .OR. (A3.LT.2.0E4 .AND. HMIN.GT.A3) ) GO TO 340
  338   IERR=1
        WRITE(P,7045) I, J
  340   CONTINUE
        IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9862)        (M(I,J), J=1,DIMW)
        IF(LIST.EQ.1) WRITE(P,912) I, (M(I,J), J=1,DIMW)
  350   CONTINUE
C WRITE MOVED BEFORE LOOP
        IF(.NOT. (NO.EQ.1 .AND. WATER.EQ.CHK(2) .AND. CONVRT.NE.CHK(7)))
     1       GO TO 360
        WRITE(P,7040)
        IERR=1
C
C       ................................ LAND (LAND ELEVATION) ...........
  360   IF(EVAP.NE.CHK(6) .OR. WATER.NE.CHK(2) ) GO TO 400
        READ(R, 842) SETNAM, FACT, IVAR, IPRN, NOWRN
     * , WELTOP
        IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, NOWRN
     * , WELTOP
        IF(APRN .GT. 1) IPRN = 1
        IF(APRN .LT. 0) IPRN = 0
        IF(WELTOP .EQ. 0.) WELTOP = 2E4
        IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, NOWRN
     * , WELTOP
        SET=SETID(11)
        IF(SETNAM.NE.SET) GO TO 435
        IF(IVAR .EQ. 0) WRITE(P, 760) FACT
        A2=FACT
        LIST=0
        IF (IVAR.EQ.1 .AND. IPRN.NE.1) LIST=1
        IF (LIST.EQ.1) WRITE(P,990)
        DO 390 I=1,DIML
        IF (IVAR.EQ.1) READ (R,840) (LAND(I,J),J=1,DIMW)
        IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9950)  (LAND(I,J),J=1,DIMW)
        DO 380 J=1,DIMW
        IF(IVAR.EQ.1) A2=LAND(I,J)*FACT
        LAND(I,J)=A2
        IF(A2.GE.2.0E4) GO TO 380
C       .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
        A3=0.0
        IF(CONVRT.EQ.CHK(7)) A3=TOP(I,J)
        HMIN=0.0
        IF(LEAK.NE.CHK(9)) GO TO 376
```

```
      HMIN=H(I,J)
      IF(RATE(I,J).EQ.0.0) GO TO 376
      IF(HMIN) 372, 376, 374
  372 IF(MESSG .EQ. 1 .AND. NOWRN .EQ. 0) WRITE(P, 7058) I, J
      IERR2=1
      IF(A3.EQ.0.0 .OR. A2.GT.A3) GO TO 376
      IF(MESSG.EQ.1) WRITE(P,7057) I,J
      IERR2=1
      GO TO 376
  374 WRITE(P,7059) I,J
      IERR=1
      LAND(I,J)=2.0E4
      GO TO 380
  376 IF(STRT(I,J).LE.A2) GO TO 380
      IF(A3 .GT. 0. .AND. A2 .GE. A3) GO TO 380
      WRITE(P,7060) I,J
      IERR=1
  380 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9932)    (LAND(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,912) I, (LAND(I,J), J=1,DIMW)
  390 CONTINUE
C WRITE MOVED BEFORE LOOP
C     ................................. QRE (RECHARGE RATE) ............
  400 IF (RECH.NE.CHK(10)) GO TO 440
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, NOWRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, NOWRN
      IF(IPRN .GT. 1) IPRN = 1
      IF(IPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, NOWRN
      SET=SETID(12)
      IF(SETNAM.NE.SET) GO TO 485
      IF(IVAR .EQ. 0) WRITE(P, 770) FACT
      A2=FACT
      LIST=0
      JM1=IFILE(2)
      IF (IRDRCH.EQ.1) IVAR=2
      IF (IVAR.GT.0 .AND. IPRN.NE.1) LIST=1
      IF (LIST.EQ.1) WRITE(P,1000)
      DO 430 I=1,DIML
      IF(IVAR .GT. 0) READ(R, 860) (QRE(I, J), J = 1, DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9360)    (QRE(I,J),J=1,DIMW)
      IF (IRDRCH.EQ.1)  READ (JM1) (QRE(I,J), J=1,DIMW)
      DO 420 J=1,DIMW
      IF(IVAR.EQ.1) A2=QRE(I,J)*FACT
      IF(IVAR .EQ. 2) A2 = QRE(I, J) + FACT
      QRE(I,J)=A2
      IF(A2.EQ.0.0) GO TO 420
C     .....CHECK FOR ERRORS IN DATA SPECIFICATIONS.....
      IF(ISUR(I,J).LT.2.0E4 .AND. S(I,J).GE.0.0 .AND. T(I,J).GT.0.0)
     1      GO TO 404
      IF (ISUR(I,J).GE.2.0E4) GO TO 420
      IF(MESSG .EQ. 1 .AND. IVAR .EQ. 1) WRITE(P, 7065) I, J
      IERR2=1
      QRE(I,J)=0.0
      GO TO 420
  404 IF(LEAK.NE.CHK(9) .OR. CONVRT.NE.CHK(7)) GO TO 420
      IF(RATE(I,J).EQ.0.0) GO TO 420
      IF(H(I,J).GT.0.0) GO TO 406
      IF(MESSG .EQ. 1 .AND. NOWRN .EQ. 0) WRITE(P, 7066) I, J
      IERR2=1
      GO TO 420
  406 IF(MESSG.EQ.1) WRITE(P,7067) I,J
      IERR2=1
  420 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9864)     (QRE(I,J), J=1,DIMW)
      IF(LIST.EQ.1) WRITE(P,900) I, (QRE(I,J), J=1,DIMW)
  430 CONTINUE
C
C     ..................................   WELLD  ........................
  440 IF(JW .LE. 1) GO TO 450
C       CARD DELETED
C       CARD DELETED
      DO 441 I = 1, DIML
      DO 441 J = 1, DIMW
  441 WELLD(I, J) = 0.
      READ(R, 842) SETNAM, FACT, IVAR, IPRN, IWCUM, WSCALE, CFSCAL
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN, IWCUM
     * , WSCALE, CFSCAL
```

```
      IF(SETNAM .NE. SETID(15) ) GO TO 452
      IF(WSCALE .EQ. 0.) WSCALE = 1.
      IF(CFSCAL .EQ. 0.) CFSCAL = 1.
C      CARD DELETED
C      CARD DELETED
      IF(IVAR .GT. 0) GO TO 444
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, IWCUM
     * , WSCALE, CFSCAL
      WRITE(P, 775) FACT
      DO 443 I = 1, DIML
      DO 442 J = 1, DIMW
      IF(T(I, J) .GT. 0.) WELLD(I, J) = FACT
  442 CONTINUE
      IF(ICPY2 .GE. 3) WRITE(KCPY, 9864) (WELLD(I, J), J = 1, DIMW)
  443 CONTINUE
      GO TO 450
  444 IF(IPRN .GT. 1) IPRN = 1
      IF(IPRN .LT. 0) IPRN = 0
      IF(ICPY .GE. 2) WRITE(KCPY, 9422) SETNAM, FACT, IVAR, IPRN, IWCUM
     * , WSCALE, CFSCAL
      IF(IPRN .EQ. 0) WRITE(P, 1150)
      DO 448 I = 1, DIML
      READ(R, 860) (WELLD(I, J), J = 1, DIMW)
      IF(ICPY2 .GE. 1) WRITE(KCPY, 9860) (WELLD(I, J), J = 1, DIMW)
      DO 447 J = 1, DIMW
      A2 = WELLD(I, J)*FACT
      IF(ISUR(I, J) .GE. 2E4) A2 = WELLD(I, J)*CFSCAL
      IF(A2 .EQ. 0. .OR. T(I, J) .GT. 0.) GO TO 446
      IF(MESSG .EQ. 1) WRITE(P, 3015) I, J
      IERR2 = 1
      A2 = 0.
  446 CONTINUE
      WELLD(I, J) = A2
  447 CONTINUE
      IF(ICPY2 .GE. 2) WRITE(KCPY, 9864) (WELLD(I, J), J = 1, DIMW)
      IF(IPRN .EQ. 0) WRITE(P, 900) I, (WELLD(I, J), J = 1, DIMW)
  448 CONTINUE
C     ................................ DELX , DELY ....................
  450 READ(R, 842) SETNAM, FACT, IVAR, IPRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
  452 SET = SETID(13)
      IF(SETNAM .NE. SET) GO TO 485
      A2 = FACT
      IF(IVAR .EQ. 1) READ(R, 840) (DELX(J), J = 1, DIMW)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1)WRITE(KCPY,9950)  (DELX(J),   J=1,DIMW)
      DO 460 J=1,DIMW
      IF(IVAR.EQ.1) A2=DELX(J)*FACT
      DELX(J)=A2
  460 CONTINUE
      IF(ICPY2.GE.3-IVAR) WRITE(KCPY,9952)    (DELX(J),     J=1,DIMW)
      IF(IVAR.EQ.1.AND.IPRN.NE.1) WRITE(P,1020)  (DELX(J), J=1,DIMW)
      IF (IVAR.EQ.0) WRITE (P,780) FACT
C
      READ (R,842) SETNAM, FACT, IVAR, IPRN
      IF(ICPY .GE. 1) WRITE(KCPY, 9420) SETNAM, FACT, IVAR, IPRN
      SET=SETID(14)
      IF(SETNAM.NE.SET) GO TO 485
      A2=FACT
      IF (IVAR.EQ.1) READ (R,840) (DELY(I),I=1,DIML)
      IF(ICPY2.GE.1.AND.IVAR.EQ.1) WRITE(KCPY,9950)  (DELY(I),   I=1,DIML)
      DO 480 I=1,DIML
      IF(IVAR.EQ.1) A2=DELY(I)*FACT
      DELY(I)=A2
  480 CONTINUE
      IF(ICPY2.GE.3-IVAR)           WRITE(KCPY,9952)  (DELY(I),   I=1,DIML)
      IF(IVAR.EQ.1.AND.IPRN.NE.1) WRITE(P,1030)  (DELY(I), I=1,DIML)
      IF (IVAR.EQ.0) WRITE (P,790) FACT
C
      IF (KED.EQ.3) GO TO 484
      IF ((IERR.GT.0 .AND. (KED.LT.2 .OR. KED.EQ.4)) GO TO 486
      IF ((IERR2.GT.0 .AND. KED.EQ.0) GO TO 486
C
C     ..............................................................
C
C ...  CALCULATE THE INITIAL MASS BALANCE RESIDUAL AT EACH NODE AND ADD
C      TO RECHARGE.
C
      IF (MBRES.NE.1) GO TO 482
```

```
      WRITE (P,1085)
      IF (RECH.EQ.CHK(10)) WRITE (P,1087)
      WRITE (P,945)
      SUMRES=0.0
      DO 1250 I=1,DIML
      YVAL=DELY(I)
      DO 1230 J=1,DIMW
      PHE(I,J)=0.0
      IF (T(I,J).LT.EPS1 .OR. S(I,J).LT.0.0) GO TO 1230
      IM1=I-1
      IP1=I+1
      JM1=J-1
      JP1=J+1
      XVAL=DELX(J)
      HMIN=STRT(I,J)
      TT=T(I,J)
      TM=T(I,JM1)
      RES=     2.*TT*TM*FACTX*(STRT(I,JM1)-HMIN)/((TT*DELX(JM1)+TM*XVAL)*
     $      XVAL)
      TM=T(I,JP1)
      RES=RES+2.*TT*TM*FACTX*(STRT(I,JP1)-HMIN)/((TT*DELX(JP1)+TM*XVAL)*
     $      XVAL)
      TM=T(IM1,J)
      RES=RES+2.*TT*TM*FACTY*(STRT(IM1,J)-HMIN)/((TT*DELY(IM1)+TM*YVAL)*
     $      YVAL)
      TM=T(IP1,J)
      RES=RES+2.*TT*TM*FACTY*(STRT(IP1,J)-HMIN)/((TT*DELY(IP1)+TM*YVAL)*
     $      YVAL)
      PHE(I,J)=RES
      SUMRES=SUMRES+RES
      IF (RECH.EQ.CHK(10)) QRE(I,J)=QRE(I,J)+RES
 1230 CONTINUE
      WRITE (P,900) I, (PHE(I,J), J=1,DIMW)
 1250 CONTINUE
      WRITE (P,1086)  SUMRES
C
C     ..................................................................
C
C     ---INITIALIZE VARIABLES---
  482 CONTINUE
      ETQS=0.0
      ETQD=0.0
      SLEAK=0.0
      SUBS=0.0
      U=0.0
      TT=0.0
      IM=MIN0(6*DIMW+4,124)
      IM=(132-IM)/2
      VF4(3) = DIGIT(IM)
      VF4(3) = DIGIT(IM + 5)
      IF(NLAY .EQ. 0) GO TO 4829
      IF(NLAY .GE. MLAY) WRITE(P, 7007)
      IF(NLAY .GE. MLAY) GO TO 486
      NLAY1 = NLAY
      NLAY = NLAY + 1
      SAT = SAT + 1.
      SATL(NLAY) = SAT
      DO 4823 I = 1, DIML
      DO 4823 J = 1, DIMW
      A2 = PERM(I, J)*SAT
      IF(A2 .LE. TLAY(I, J, NLAY1) ) A2 = TLAY(I, J, NLAY1)
     * + PERM(I, J)*(SAT - SAT1)
      TLAY(I, J, NLAY) = A2
      IF(A2 .GT. 0.) KKLAY(I, J) = NLAY + 1
 4823 CONTINUE
 4829 CONTINUE
      IF(KPLT .GT. 0) CALL PLTXI(KPLT, KPLC, DIML, DIMW,DELX(1),DELY(1))
      IF(NLAY0 .LT. 0) CALL NEWSTP
      IF(NLAY0 .LT. 0) NLAY = 0
      RETURN
C
C     ---TERMINATE PROGRAM EXECUTION---
  484 WRITE(P,7001)
      STOP
  485 WRITE(P,7002) SET, SETNAM
  486 WRITE(P,7000)
      STOP 16
C
```

```
C     ..........................................................................
C
C     ---COMPUTE AND PRINT ITERATION PARAMETERS---
C     *******************
      ENTRY  ITER
C     *******************
      HMIN=2.
      XVAL=3.1415**2/(2.*DIMW**2)
      YVAL=3.1415**2/(2.*DIML**2)
      DO 490 I=2,DIML
      DO 490 J=2,DIMW
      IF (T(I,J).EQ.0.) GO TO 490
      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)
  490 CONTINUE
      ALPHA=EXP(ALOG(HMAX/HMIN)/(LENGTH-1))
      RHOP(1)=HMIN
      DO 500 NTIME=2,LENGTH
  500 RHOP(NTIME)=RHOP(NTIME-1)*ALPHA
      WRITE (P,1010) LENGTH,(RHOP(J),J=1,LENGTH)
      RETURN
C     ..........................................................................
C
C     ---INITIALIZE DATA FOR ALPHAMERIC PLOT---
C     *******************
      ENTRY  MAP
C     *******************
      WIDTH=0.
      DO 510 J=2,JNO1
  510 WIDTH=WIDTH+DELX(J)
      YDIM=0.
      DO 520 I=2,INO1
  520 YDIM=YDIM+DELY(I)
  530 XSF=DINCH*SCALE
      NYD=YDIM/XSF
      IF (NYD*XSF.LE.YDIM-DELY(INO1)/2.) NYD=NYD+1
      IF (NYD.LE.12) GO TO 540
      DINCH=YDIM/(12.*SCALE)
      WRITE(P,1130) DINCH
      IF(SCALE.LT.1.0) WRITE(P,1140)
      GO TO 530
  540 NXD=WIDTH/XSF
      IF (NXD*XSF.LE.WIDTH-DELX(JNO1)/2.) NXD=NXD+1
      N4=NXD*N1+1
      N5=NXD+1
      N6=NYD+1
      N3=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(5)=MESUR
      DO 560 I=1,NE
      NNX=N5-I
      NNY=I-1
      IF (NNY.GE.N6) GO TO 550
      YN(I)=XSF*NNY/SCALE
  550 IF (NNX.LT.0) GO TO 560
      XN(I)=XSF*NNX/SCALE
  560 CONTINUE
      RETURN
C     ..........................................................................
C
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-
C     ***********************
      ENTRY  NEWPER
C     ***********************
C
      SET=SETID(15)
      KED=KEDIT
      IF (KEDIT.GT.11) KED=KED-12
```

```
      IF (KEDIT.LT.6) GO TO 568
      MESSG=1
      IF (KED.EQ.4 .OR. KED.EQ.5) MESSG=0
      IERR=0
      IERR2=0
C
  568 READ (R,343,END=654) SETNAM, KP, KPM1, NWEL, TMAX, NUMT, CDLT,
     *        DELT, NW, WMULT, IPRDON, IPRHED, ISVHED, KTHP
      IF(ICPY .GE. 1) WRITE(KCPY, 9430) SETNAM, KP, KPM1, NWEL, TMAX
     * , NUMT, CDLT, DELT, NW, WMULT, IPRDON, IPRHED, ISVHED, KTHP
      IF(KTHP .NE. 0) KTH = KTHP
      IF (SETNAM.NE.SET) GO TO 656
C
C     ---COMPUTE ACTUAL DELT AND NUMT---
      DT=DELT/24.
      TM=0.0
      DO 570 I=1,NUMT
      DT=CDLT*DT
      TM=TM+DT
      J=I
      IF (TM.GE.TMAX) GO TO 580
  570 CONTINUE
      GO TO 590
  580 DELT=TMAX/TM*DELT
      NUMT=J
  590 CONTINUE
      IF(ICPY .GE. 2) WRITE(KCPY, 9430) SETNAM, KP, KPM1, NWEL, TMAX
     * , NUMT, CDLT, DELT, NW, WMULT, IPRDON, IPRHED, ISVHED, KTH
      WRITE(P, 1050) KP, TMAX, NUMT, DELT, CDLT
      DELT=DELT*3600.
      TMAX=TMAX*86400.
C
C     ---READ AND WRITE WELL PUMPING RATES---
      WRITE (P,1058) NWEL, KP
      IF (NWEL.GT.0) WRITE(P,1060)
      IF (KP.GT.KPM1) SUMP=0.
      A2 = 0.
      DO 605 I = 1, DIML
      IF(NW .LT. 0) READ(R, 360) (WELL(I, J), J = 1, DIMW)
      IF(NW.LT.0 .AND. ICPYW.GE.1) WRITE(KCPY,9360)(WELL(I,J),J=1,DIMW)
      DO 600 J=1,DIMW
      IF(KP.GT.KPM1) STRT(I,J)=PHI(I,J)
      IF(JW .LE. 1) GO TO 600
      IF(NW .LT. 0) A2 = WELL(I, J)
      IF(ISUR(I, J) .GE. 2E4) GO TO 598
      FLUX = WELLD(I, J) + A2*WSCALE
      IF(IWCUM .GT. 0) WELLD(I, J) = FLUX
      WELL(I, J) = FLUX*WMULT
      GO TO 600
  598 WELL(I, J) = WELLD(I, J) + A2*CFSCAL
  600 CONTINUE
      IF(NW.LT.0 .AND. ICPYW.GE.2) WRITE(KCPY,9364)(WELLD(I,J),J=1,DIMW)
  605 CONTINUE
      IF(JW .LE. 1) GO TO 650
      IF(NW .LE. 0) GO TO 620
      DO 610 I=1,NW
  610 WR(I)=0.
  620 IF(NWEL .LE. 0) GO TO 650
C
      KW=0
C      CARD DELETED
      DO 640 II=1,NWEL
      READ (R,840,END=655) I, J, FLUX, RADIUS
      IF(ICPYW .GE. 1) WRITE(KCPY, 9400) I, J, FLUX, RADIUS
      IF(ISUR(I, J) .LT. 2E4) FLUX = WELLD(I, J) + FLUX*WSCALE
      IF(ISUR(I, J) .GE. 2E4) FLUX = WELLD(I, J) + FLUX*CFSCAL
      IF(RADIUS .LT. 0.) FLUX = 0.
      IF(ICPYW .GE. 2) WRITE(KCPY, 9402) I, J, FLUX, RADIUS
      IF(ISUR(I, J) .GE. 2E4) GO TO 621
      IF(IWCUM .GT. 0) WELLD(I, J) = FLUX
C      CARD DELETED
      FLUX = FLUX*WMULT
  621 CONTINUE
      WELL(I, J) = FLUX
      IF(ICPYW .GE. 3) WRITE(KCPY, 9402) I, J, FLUX, RADIUS
C
      IF(ABS(FLUX).GT.0.0) GO TO 622
      IF(ISUR(I,J).LT.2.0E4) GO TO 622
```

```
      IF (MESSG.EQ.1) WRITE (P,3010) I, J
      IERR2=1
      GO TO 628
  622 IF(T(I,J).GT.0.0) GO TO 623
      WRITE(P,8015) I, J
      IERR=1
      GO TO 628
  623 IF(S(I,J).GE.0.0) GO TO 624
      WRITE(P,8020) I, J
      IERR=1
      GO TO 628
  624 IF(I.GT.1 .AND. I.LT.DIML .AND. J.GT.1 .AND. J.LT.DIMW) GO TO 628
      WRITE(P,8030) I, J
      IERR=1
  628 CONTINUE
C
      IF(RADIUS .LE. 0.) GO TO 630
      KW=KW+1
      IF (KW.GT.NW) GO TO 630
      NWR(KW,1)=I
      NWR(KW,2)=J
      WR(KW)=RADIUS
      WRITE (P,1070) I,J,WELL(I,J),WR(KW)
      IF (ISUR(I,J).GE.2.0E4) WRITE (P,1072)
      GO TO 640
  630 WRITE (P,1070) I,J,WELL(I,J)
      IF (ISUR(I,J).GE.2.0E4) WRITE (P,1072)
  640 CONTINUE
C
      IF (KEDIT.GT.17) GO TO 568
      IF (KED.EQ.3) GO TO 658
      IF (IERR.GT.0 .AND. (KED.LT.2 .OR. KED.EQ.4)) GO TO 657
      IF (IERR2.GT.0 .AND. KED.EQ.0) GO TO 657
  650 RETURN
C
C     ---TERMINATE PROGRAM EXECUTION---
  654 WRITE(P,7003)
      IF (KEDIT.LT.18) GO TO 657
  658 WRITE (P,7001)
      STOP
  655 WRITE (P,7004) KP
      GO TO 657
  656 WRITE(P,7002) SET, SETNAM
  657 WRITE(P,7000)
      STOP 16
C     ..............................................................
C
C     FORMATS:
C
C
  660 FORMAT ('0',63X,'STARTING HEAD =',G15.7)
  670 FORMAT ('0',57X,'STORAGE COEFFICIENT =',G15.7)
  680 FORMAT ('0',62X,'SPECIFIC YIELD =',G15.7)
  690 FORMAT ('0',37X,'HYDRAULIC CONDUCTIVITY OF CONFINING BED =',G15.7)
  700 FORMAT ('0',66X,'RIVER HEAD =',G15.7)
  710 FORMAT ('0',53X,'CONFINING BED THICKNESS =',G15.7)
  720 FORMAT ('0',62X,'TRANSMISSIVITY =',G15.7)
  730 FORMAT ('0',46X,'AQUIFER HYDRAULIC CONDUCTIVITY =',G15.7)
  740 FORMAT ('0',60X,'BOTTOM ELEVATION =',G15.7)
  750 FORMAT ('0',63X,'TOP ELEVATION =',G15.7)
  760 FORMAT ('0',62X,'LAND ELEVATION =',G15.7)
  770 FORMAT ('0',63X,'RECHARGE RATE =',G15.7)
  775 FORMAT('0', 64X, 'PUMPING RATE =', G15.7)
  780 FORMAT ('0',72X,'DELX =',G15.7)
  790 FORMAT ('0',72X,'DELY =',G15.7)
  800 FORMAT (1H1,16A8)
  810 FORMAT (10A8)
  820 FORMAT(1H1,56X, 20HSTARTING HEAD MATRIX/6H   ROW)
  830 FORMAT(1H1,15X,   '---CONTINUATION OF SIMULATION---  HEAD AFTER ',
     1  F13.2,    '  SECONDS OF PREVIOUS SIMULATION'/
     2  6H   ROW)
  840 FORMAT (3G10.0)
  841 FORMAT (16G5.0)
  842 FORMAT(A4, 6X, 7G10.0)
  843 FORMAT (A4, 1X, 15G5.0)
  850 FORMAT (A8,2X,3G10.0,A8)
  860 FORMAT (20F4.0)
  870 FORMAT(1H1,38X, 52HHYDRAULIC CONDUCTIVITY OF CONFINING BED OR RIVE
```

```
     .R BED/ 6H    ROW)
 880 FORMAT ('-',51X,'NUMBER OF PUMPING PERIODS =',I5/63X'NUMBER OF ROW
    1S =',I5/60X,'NUMBER OF COLUMNS =',I5/49X,'TIME STEPS BETWEEN PRINT
    2OUTS =',I5//51X,'ERROR CRITERIA FOR CLOSURE =',G15.7/39X,'MAXIMUM
    3PERMITTED NUMBER OF ITERATIONS =',I5/41X,'            STEADY STATE ER
    4ROR CRITERIA =',G15.7//44X,'SPECIFIC STORAGE OF CONFINING BED =',G
    ,15.7/ 48X,   'SPECIFIC STORAGE OF RIVER BED =',G15.7/
    5 54X, 'EVAPOTRANSPIRATION RATE =', G15.7/56X,   'EFFECTIVE DEPTH OF
    6 ET =',G15.7//22X,'MULTIPLICATION FACTOR FOR TRANSMISSIVITY IN X D
    7IRECTION =',G15.7/63X,'IN Y DIRECTION =',G15.7)
 890 FORMAT (1H1, 56X, 21HTRANSMISSIVITY MATRIX/ 6H    ROW)
 892 FORMAT (1H1,50X, 32HTRANSMISSIVITY MATRIX (COMPUTED)/ 6H    ROW)
 894 FORMAT(// 46X,  40HTRANSMISSIVITY VALUES HAVE BEEN COMPUTED)
 900 FORMAT(1H0, I5,2X, 10G11.4/ (8X, 10G11.4))
 910 FORMAT(1H0, I5,2X, 10F11.0/ (8X, 10F11.0))
 912 FORMAT(1H0, I5,2X, 10F11.2/ (8X, 10F11.2))
 916 FORMAT(1H0, I5,2X, 10F11.6/ (8X, 10F11.6))
 930 FORMAT(1H1,54X, 26HSTORAGE COEFFICIENT MATRIX/ 6H    ROW)
 940 FORMAT (A8)
 945 FORMAT(6H    ROW)
 950 FORMAT (3F10.0)
 960 FORMAT (1H1,42X, 29HHYDRAULIC CONDUCTIVITY MATRIX/ 6H    ROW)
 962 FORMAT (1H1,46X, 40HHYDRAULIC CONDUCTIVITY MATRIX (COMPUTED)/
   1    6H    ROW)
 964 FORMAT(// 42X,   48HHYDRAULIC CONDUCTIVITY VALUES HAVE BEEN COMPUT
    .ED)
 970 FORMAT(1H1,51X, 30HELEVATION AT BOTTOM OF AQUIFER/
   1   6H    ROW)
 974 FORMAT(1H1,52X, 28HSTARTING SATURATED THICKNESS/ 6H    ROW)
 980 FORMAT(1H1,40X, 53HELEVATION AT TOP OF AQUIFER OR AT BOTTOM OF LEA
    1KY BED/ 6H    ROW)
 990 FORMAT(1H1,54X, 25HELEVATION AT LAND SURFACE/ 6H    ROW)
1000 FORMAT(1H1,60X, 13HRECHARGE RATE/6H    ROW)
1010 FORMAT (///1H0,I5,22H ITERATION PARAMETERS:,8D12.3//6X,10D12.3)
1020 FORMAT(1H1,40X,  40HGRID SPACING IN PROTOTYPE -- X DIRECTION//
   1   (3X, 10F12.0))
1030 FORMAT(1H-,40X,  40HGRID SPACING IN PROTOTYPE -- Y DIRECTION//
   1   (3X, 10F12.0))
1040 FORMAT(1H1,31X,   'ELEV. OF W.S. IN RIVER OR HEAD IN OVERLYING (OR
    . UNDERLYING) AQUIFER' /  6H    ROW)
1041 FORMAT(/  5X,   'HEAD IN THE OVERLYING (OR UNDERLYING) AQUIFER IS
    &COMPUTED BY ADDING ', F10.4,   '  TO STARTING HEAD VALUE AT EACH N
    &ODE.'/)
1050 FORMAT ('-',50X,'PUMPING PERIOD NO.',I4,':',F10.2,' DAYS'/51X,38('
    1-')//53X,'NUMBER OF TIME STEPS=',I6//59X,'DELT IN HOURS =',F10.3//
    253X,'MULTIPLIER FOR DELT =',F10.3)
1058 FORMAT(//  50X, I4,  ' WELLS FOR PUMPING PERIOD NO. ', I4/
   1      48X, 42(1H-) )
1060 FORMAT(/50X, 1HI, 9X, 30HJ     PUMPING RATE  WELL RADIUS/)
1070 FORMAT (41X,2I10,2F13.2)
1072 FORMAT(1H+,90X, '*** CONSTANT FLUX BOUNDARY')
1080 FORMAT(1H1,45X, 39HTHICKNESS OF CONFINING BED OR RIVER BED/
   1    6H    ROW)
1085 FORMAT(1H1, 20X, '*** MASS BALANCE RESIDUALS (FT/SEC) AT SIMULATIO
    &N TIME = 0 ***' )
1086 FORMAT(1H-,   'TOTAL RESIDUAL = ', 1PE12.4 )
1087 FORMAT(1H0,35X,   '(RESIDUALS ARE ADDED TO RECHARGE)' )
1090 FORMAT(2G10.0, 3G20.0 / (4G20.0) )
1100 FORMAT ('-SIMULATION OPTIONS:   ',10(A8,3X))
1110 FORMAT ('0',40X,'ON ALPHAMERIC MAP:'/45X,'MULTIPLICATION FACTOR FO
    1R LENGTH =',G15.7/56X,'MAP SCALE IN UNITS OF  ',A11/50X,'NUMBER OF
    2 ',A5,' PER INCH =',G15.7/51X,'CONTOUR INTERVAL =',G15.7)
1120 FORMAT(1H1,56X, 21HSPECIFIC YIELD MATRIX/5H    ROW)
1130 FORMAT('0',25X,10('*'),' TO FIT MAP WITHIN 12 INCHES, DINCH REVISE
    1D TO',G15.7,1X,10('*'))
1140 FORMAT('0',45X,'NOTE: GENERALLY SCALE SHOULD BE > OR = 1.0')
1150 FORMAT('1', 30X, 'PUMPING RATE'/' ROW')
C
7000 FORMAT(//  76H ***** PROGRAM EXECUTION IS TERMINATED BECAUSE OF ER
    1RORS IN INPUT DATA ***** //)
7001 FORMAT(//   29H ***** END OF DATA EDIT ***** //)
7002 FORMAT(// 1X, 11H***ERROR***,    ' WRONG DATA SET IDENTIFIER READ F
    1ROM PARAMETER CARD.' // 10X,
    2   'LOOKING FOR IDENTIFIER = ',A4,   '  BUT READ IDENTIFIER = ',
    3  A4)
7003 FORMAT (//1X,   '***** END OF FILE FOR PUMPAGE DATA *****'/)
7004 FORMAT(//  1X, 11H***ERROR***,    ' END OF FILE BEFORE READING ALL
    . PUMPAGE DATA FOR PUMPING PERIOD NO. ', I4, 1H./)
```

```
7007 FORMAT(//1X, '***ERROR*** DIMENSIONS EXCEED MAXIMUM')
7010 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1   ', A CONSTANT FLUX BOUNDARY IS SPECIFIED ON THE NOFLOW BORDER
    2SURROUNDING' / 10X,  'THE MODEL (CONSTANT FLUX AT THIS NODE IS IG
    3NORED DURING SIMULATION).' /)
7012 FORMAT(/ 1X,  11H***ERROR***,   ' AT ROW ',I3, ' COL ', I3,
    1   ' STARTING HEAD VALUE IS ZERO.'/)
7015 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1   ', A CONSTANT HEAD BOUNDARY IS SPECIFIED ON THE NOFLOW BORDER
    2SURROUNDING' / 10X,  'THE MODEL (CONSTANT HEAD AT THIS NODE IS IG
    3NORED DURING SIMULATION).' /)
7020 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1   ', BOTH CONSTANT HEAD AND CONSTANT FLUX BOUNDARIES ARE' /
    2 10X,  'SPECIFIED (CONSTANT HEAD IS USED AT THIS NODE DURING SIMU
    3LATION).' /)
7025 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ', I3, '  COL ', I3,
    1   ', TRANSMISSIVITY (OR HYDRAULIC CONDUCTIVITY) IS ZERO FOR A CONS
    2TANT HEAD OR A CONSTANT FLUX BOUNDARY.'/)
7030 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ', I3, '  COL ', I3,
    1   ', STARTING HEAD IS BELOW TOP OF ARTESIAN AQUIFER (IF THIS IS T
    2HE CONTINUATION' / 10X,  'OF A PREVIOUS SIMULATION, THE STARTING
    3HEAD IS PHI).' /)
7032 FORMAT(/ 1X,13H***WARNING***,  ' AT ROW ',I3, ' COL ',I3,
    1   ' HEAD IS BELOW BOTTOM. HEAD CHANGED TO BOTTOM + THIN.')
7034 FORMAT(/ 1X,13H***WARNING***,  ' AT ROW ',I3, ' COL ',I3,
    1   ' SATURATED THICKNESS IS ZERO OR NEGATIVE, BUT HYDRAULIC CONDUC
    2TIVITY' / 15X, 'IS GREATER THAN ZERO (PROGRAM SETS PERM AND T TO Z
    3ERO).'/)
7036 FORMAT(/ 1X,13H***WARNING***,  ' AT ROW ',I3, ' COL ',I3,
    1   ' SATURATED THICKNESS IS ZERO OR NEGATIVE, BUT TRANSMISSIVITY I
    2S' / 15X,  'GREATER THAN ZERO (PROGRAM SETS PERM AND T TO ZERO).'
    3/)
7040 FORMAT(/ 1X, 11H***ERROR***,      ' CONVERSION IS NOT SPECIFIED FOR
    1 SIMULATION OF A W.T. AQUIFER WITH LEAKAGE FROM AN OVERLYING RIVER
    2.  SIMULATION RESULTS' / 10X,  'WILL BE IN ERROR (CANNOT COMPUTE
    3THE CORRECT VALUES FOR LEAKAGE AND TRANSMISSIVITY).  THE DATA SET
    4SPECIFYING ELEVATION' / 10X,  'AT TOP OF AQUIFER MUST BE READ AND
    5 CONVERT MUST BE SPECIFIED.' /)
7045 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ', I3, '  COL ', I3,
    1   ', VALUES FOR RIVER, RATE, M, TOP, STRT, S AND T ARE NOT CONSIS
    2TENT.' /)
7057 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1   ', ELEV. OF LAND SURFACE IS BELOW BOTTOM OF RIVER BED (THIS DOE
    2S NOT AFFECT' / 10X, 'THE RESULTS OF SIMULATION).' /)
7058 FORMAT(/ 1X,  13H***WARNING***,     ' AT ROW ', I3, '  COL ', I3,
    1   ', BOTH EVAPOTRANSPIRATION AND LEAKAGE FROM A RIVER MAY OCCUR D
    2URING SIMULATION.' /)
7059 FORMAT(/ 1X,  11H***ERROR***,       ' AT ROW ', I3, '  COL ', I3,
    1   ', BOTH EVAPOTRANSPIRATION AND LEAKAGE FROM AN OVERLYING AQUIFE
    2R ARE SPECIFIED (LAND(I,J) IS' / 10X,   'SET TO 20,000 AT THIS NOD
    3E TO PREVENT EVAPOTRANSPIRATION).' /)
7060 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ', I3, '  COL ', I3,
    1   ', HEAD IS ABOVE LAND SURFACE FOR A W.T. AQUIFER.' /)
7065 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1             ', RECHARGE IS SPECIFIED FOR A CONSTANT HEAD, CONSTANT
    2 FLUX, OR'/ 10X,  'IMPERMEABLE BOUNDARY (ZERO RECHARGE IS ASSIGNE
    3D TO THIS NODE).' /)
7066 FORMAT(/ 1X,  13H***WARNING***,     ' AT ROW ', I3, '  COL ', I3,
    1             ', RECHARGE IS SPECIFIED WHERE VERTICAL LEAKAGE OCCURS
    2 FROM AN OVERLYING' / 10X,   'RIVER (RECHARGE IS USED DURING SIMUL
    3ATION).' /)
7067 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1             ', RECHARGE IS SPECIFIED WHERE VERTICAL LEAKAGE OCCURS
    2 FROM AN OVERLYING' / 10X,   'AQUIFER (RECHARGE IS USED DURING SIM
    3ULATION).' /)
7100 FORMAT(// ' ***** ERROR ***** LAYER OPTION INVALID FOR ARTESIAN'
    * , ' AQUIFER.')
7105 FORMAT(' *** ERROR *** ROW ', I3, ' COL ', I3
    * , ' LAYER PERM OR T OUTSIDE PREVIOUS BOUNDARY.')
7110 FORMAT(' *** ERROR *** INVALID PERM (IN SY CURVE)')
8010 FORMAT(/ 1X, 13H***WARNING***, ' AT ROW ', I3, '  COL ', I3,
    1   ', WELL FLUX IS SPECIFIED AS ZERO FOR A CONSTANT FLUX BOUNDARY
    2(ZERO FLUX IS USED IN SIMULATION).' /)
8015 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ',I3,    '  COL ',I3,
    1   ', WELL FLUX IS SPECIFIED FOR AN IMPERMEABLE BOUNDARY.'/)
8020 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ',I3,    '  COL ',I3,
    1   ', WELL FLUX IS SPECIFIED FOR A CONSTANT HEAD BOUNDARY.'/)
8030 FORMAT(/ 1X, 11H***ERROR***,    ' AT ROW ',I3,    '  COL ',I3,
```

```
     1    ', WELL FLUX IS SPECIFIED ON THE NOFLOW BORDER SURROUNDING THE
     2MODEL.'/)
C
C     CARD DELETED
 9000 FORMAT(16I5)
 9012 FORMAT(81X, 'ICPY,ICPY2,ICPYW,KCPY,MPRN,KPLT,KPLC' / 1X, 16I5)
 9013 FORMAT(81X, 'WITH DEFAULTS' / 1X, 16I5)
 9020 FORMAT(81X, 'HEADING' / 1X, 10A8 / 1X, 10A8)
 9030 FORMAT(1X, A8)
 9040 FORMAT(81X, 'CONTR,SCALE,DINCH,SPACNG,MESUR'
     * / 1X, A8, 2X, 3G15.7, A8)
 9050 FORMAT(81X, 'KEDIT,NHYDR,MBRES,ISVFLX,IRDINI,IRDHED,IRDRCH'
     * / 1X, 16I5)
 9060 FORMAT(81X, 'HYDRO NODES' / 1X, 16I5)
 9062 FORMAT(81X, 'MASS BALANCE NODES' / 1X, 16I5)
 9070 FORMAT(81X, 'NPER,DIML,DIMW,KTH,LENGTH,ERR,HMAX,ITMAX'/ 1X, 8G15.7
     * / 81X, 'EROR,SS,QET,ETDIST,FACTX,FACTY,SSRIV'/ 1X, 8G15.7)
 9072 FORMAT(81X, 'WITH DEFAULTS' / 1X, 8G15.7 / 1X, 8G15.7)
 9080 FORMAT(81X, 'INITIAL MASS BAL' / 1X,2G15.7,3G30.16 / (1X,4G30.16))
 9420 FORMAT(81X, 'PARAMETER CARD (A8,7G10.0)'
     * / 1X, A4, 11X, G15.7, 3I15, 3G15.7)
 9422 FORMAT(81X, 'WITH DEFAULTS' / 1X, A4, 11X, G15.7, 3I15, 3G15.7)
 9950 FORMAT('0', 8G15.7 / (1X, 8G15.7) )
 9952 FORMAT(81X, 'WITH DEFAULTS' / (1X, 8G15.7) )
 9860 FORMAT('0', 20F6.0 / (1X, 20F6.0) )
 9862 FORMAT(81X, 'WITH DEFAULTS' / (1X, 20F6.0) )
 9864 FORMAT(81X, 'WITH DEFAULTS' / (1X, 8G15.7) )
 9866 FORMAT(1X, 20F6.3 / (1X, 20F6.3) )
 9430 FORMAT(81X, 'KP,KPM1,NWEL,TMAX,NUMT,CDLT,DELT,',
     * / 81X, ' NW,WMULT,IPRDON,IPRHED,ISVHED,KTH'
     * / 1X, A4, 1X, 3I5, F10.2, I5, F10.4, F10.2, I5, F10.4, 4I5)
 9400 FORMAT('0', 2I10, 2F10.4)
 9402 FORMAT(81X, 'WITH DEFAULTS' / 1X, 2I10, 2F10.4)
      END


      SUBROUTINE COMPUT(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,DDN,G
     * ,BE,TEMP,IMAX,IZ,JZ,TOP,IC,JC,BOTTOM,IP,JP,WELL,IW,JW
     2 ,KKLAY,TLAY1,TLAY,MLAY, PERM, SY)
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
C*COM01*
      COMMON /DARRAY/ RHOP(20),CHK(10),VF4(11)
C*COM02*
      COMMON /DPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH
C*COM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDON,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
     * , WELBOT, WELTOP
C*COM04*
      COMMON /CK/ ETFLXT, QRET, QRETN, FLUXT, FLXNT, CHST, CHDT
     * , CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
C*COM05*
C*COM05*
      COMMON / COPYC / ICPYW, KCPY
      COMMON / LAYER / SATL(20), PRMSY(10), SYPRM(10), SYD(10)
     * , NLAY, NLAY0, MSY, NSY, NSY0
C*DIM01*
      DIMENSION  BE(IMAX), G(IMAX), TEMP(IMAX), PHI(IZ,JZ),
     1  KEEP(IZ,JZ), PHE(IZ,JZ), STRT(IZ,JZ), ISUR(IZ,JZ), T(IZ,JZ),
     2  TR(IZ,JZ), TC(IZ,JZ), S(IZ,JZ), DELX(JZ), DELY(IZ), DDN(JZ)
     3  , TOP(IC,JC), BOTTOM(IP,JP), WELL(IW, JW)
     4  , KKLAY(IZ,JZ), TLAY1(IZ,JZ), TLAY(IZ,JZ,MLAY)
     5  , PERM(IP,JP), SY(IP,JP)
C*DIM02*
C*TYP01*
      REAL *4MINS,TEST3(102),KEEP,ISUR
C*TYP02*
      REAL *8PHI,DBLE,RHOP,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONT
     1R,EROR,XLABEL,YLABEL,TITLE,G,BE,TEMP,IMK,  DABS,D,W,T1,T2,T3,T4,RH
     2O,A,B,C,PARAM,TEST2,DMAX1,LEAK,RECH
```

```
C DELETED
C DELETED
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C*END03*
 9999 FORMAT('  SUB COMPUT   2/12/79  MCD 26, 27A')
      WRITE(P, 9999)
      RETURN
C     ..........................................................................
C     ---START A NEW TIME STEP---
C     ********************
      ENTRY NEWSTP
C     ********************
      IF(NLAY .EQ. 0) GO TO 1590
      DO 1580 I = 1, DIML
      DO 1580 J = 1, DIMW
      K = KKLAY(I, J)
      IF(K .LE. 0) GO TO 1580
      HED = PHI(I, J)
      IF(CONVRT .EQ. CHK(7) ) HED = AMIN1(HED, TOP(I, J) )
      SAT = HED - BOTTOM(I, J)
      IF(K .LE. 1) GO TO 1520
      IF(SATL(K - 1) .LT. SAT) GO TO 1520
      S1 = 0.
      T1 = 0.
 1510 K = K - 1
      IF(K .LE. 1) GO TO 1550
      IF(SATL(K - 1) .LT. SAT) GO TO 1540
      GO TO 1510
 1520 IF(K .GE. NLAY) GO TO 1580
      IF(SATL(K) .GE. SAT) GO TO 1580
 1530 K = K + 1
      IF(K .GE. NLAY) GO TO 1540
      IF(SATL(K) .GT. SAT) GO TO 1540
      GO TO 1530
 1540 S1 = SATL(K - 1)
      T1 = TLAY(I, J, K - 1)
 1550 CONTINUE
      KKLAY(I, J) = K
      TLAY1(I, J) = T1
      A2 = (TLAY(I, J, K) - T1)/(SATL(K) - S1)
      IF(NLAY0 .LT. 0 .AND. SAT .GT. 0.) A2 = (T1 + A2*(SAT - S1) )/SAT
      PERM(I, J) = A2
      P0 = PRMSY(1)
      IF(P0 .EQ. 0.) GO TO 1580
      DO 1568 JSY = 2, NSY
      ISY = JSY
      P1 = P0
      P0 = PRMSY(ISY)
      IF(P0 .GT. A2) GO TO 1569
 1568 CONTINUE
 1569 SY(I, J) = SYPRM(ISY) + SYD( ISY)*ALOG10(A2/P1)
 1580 CONTINUE
      IF(NLAY0 .LT. 0) RETURN
 1590 CONTINUE
      DELT = CDLT*DELT
      SUM = SUM + DELT
      SUMP = SUMP + DELT
      DAYSP = SUMP/86400.
      YRSP = DAYSP/360.
      HRS=SUM/3600.
      MINS=HRS*60.
      DAYS=HRS/24.
      YRS = DAYS/360.
      KT = KT + 1
      KSTEP = KSTEP + 1
      ISTEDY = 0
      KOUNT = 0
      DO 1880 I = 1, DIML
      DO 1880 J = 1, DIMW
 1880 KEEP(I, J) = PHI(I, J)
      RETURN
C
C     ---INITIALIZE DATA FOR A NEW ITERATION---
C     ***********************
      ENTRY NEWIT1
C     ***********************
      IF (KOUNT.LT.ITMAX) GO TO 20
```

```
      KOUNT=KOUNT+1
      WRITE(P,410) KT, DAYS
      GO TO 260
   20 KOUNT=KOUNT+1
      IF (MOD(KOUNT,LENGTH)) 30,30,40
C     ************************
      ENTRY NEWITO
C     ************************
   30 NTH=0
   40 NTH=NTH+1
      PARAM=RHOP(NTH)
      TEST3(KOUNT+1)=0.
      TEST=0.
      DO 50 I=2,INO1
      DO 50 J=2,JNO1
   50 PHE(I,J)=PHI(I,J)
      RETURN
C
C     ---COMPUTE IMPLICITLY ALONG ROWS---
C     ******************************************************************
      ENTRY ROW
C     ************************
      DO 60 J=1,DIMW
      BE(J)=0.0
      G(J)=0.0
   60 TEMP(J)=PHI(1,J)
      DO 140 I=2,DIML
      DO 100 J=2,JNO1
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(I,J)) 70,100,70
   70 T1=TR(I,J-1)/DELX(J)
      T2=TR(I,J)/DELX(J)
      T3=TC(I-1,J)/DELY(I)
      T4=TC(I,J)/DELY(I)
      QCOF=0.
      RW=0.
      IF (S(I,J).LT.0.) GO TO 80
C
C     ---COMPUTE ET RATE (ETQ), STEADY AND TRANSIENT LEAKAGE FACTOR
C     (SLEAK), STORAGE COEFFICIENT, WELL AND RECHARGE RATES---
      IF (EVAP.EQ.CHK(6)) CALL ETRATE(I,J)
      IF (LEAK.EQ.CHK(9)) CALL LEAKAG(I,J,QCOF)
      CALL SOURCE(I,J,RW)
      CALL STORAG(I,J)
C
      RHO=STORE/DELT
      GO TO 90
   80 RHO=1.0E40
C
C     ---CALCULATE VALUES FOR PARAMETERS USED IN THOMAS ALGORITHM ---
   90 IMK=PARAM*(T1+T2+T3+T4)
      A=T1
      B=-T1-T2-RHO-IMK-QCOF*U-ETQB
      C=T2
      W=B-A*BE(J-1)
      IF(DABS(W) .GT. 1.E-20) GO TO 95
      WRITE(P, 94) I, J, W, T1, T2, T3, T4
   94 FORMAT(' ***** IN ENTRY ROW, ROW', I3, ', COL', I3,
     * ', DENOM W =', G15.7, ',' / 7X, 'T1, T2, T3, T4 =', 4G15.7)
      W = 1.D0
   95 CONTINUE
      BE(J)=C/W
      D=-T3*PHI(I-1,J)+(T4+T3-IMK)*PHI(I,J)-T4*PHI(I+1,J)-RHO*KEEP(I,J)-
     1SLEAK+RW+ETQD-SUBS
      G(J)=(D-A*G(J-1))/W
  100 CONTINUE
C
C     ---CALCULATE HEAD VALUES FOR ROWS OF MATRIX AND PLACE THEM IN
C     TEMPORARY LOCATION TEMP;IF(I.GT.2) FIRST MAKE PHI(I-1,NO4)=
C     TEMP(NO4)---
      NO3=DIMW-2
      DO 130 KNO4=1,NO3
      NO4=DIMW-KNO4
      PHI(I-1,NO4)=TEMP(NO4)
      IF (T(I,NO4)) 120,110,120
  110 TEMP(NO4)=PHI(I,NO4)
      GO TO 130
```

```
  120 TEMP(NO4)=G(NO4)-BE(NO4)*TEMP(NO4+1)
  130 CONTINUE
  140 CONTINUE
      RETURN
C
C     ---COMPUTE IMPLICITLY ALONG COLUMNS---
C     ......................................................................
      ENTRY COLUMN
C     ************************
      DO 150 I=1,DIML
      BE(I)=0.0
      G(I)=0.0
  150 TEMP(I)=PHI(I,1)
      DO 230 J=2,DIMW
      DO 190 I=2,INC1
C
C     ---SKIP COMPUTATIONS IF NODE IS OUTSIDE AQUIFER BOUNDARY---
      IF (T(I,J)) 160,190,160
  160 T1=TR(I,J-1)/DELX(J)
      T2=TR(I,J)/DELX(J)
      T3=TC(I-1,J)/DELY(I)
      T4=TC(I,J)/DELY(I)
      QCOF=0.
      RW=0.
      IF (S(I,J).LT.0.) GO TO 170
C
C     ---COMPUTE ET RATE (ETQ), STEADY AND TRANSIENT LEAKAGE FACTOR
C     (SLEAK), STORAGE COEFFICIENT, WELL AND RECHARGE RATES---
      IF (EVAP.EQ.CHK(6)) CALL ETRATE(I,J)
      IF (LEAK.EQ.CHK(9)) CALL LEAKAG(I,J,QCOF)
      CALL SOURCE(I,J,RW)
      CALL STORAG(I,J)
C
      RHO=STORE/DELT
      GO TO 180
  170 RHO=1.0E40
C
C     ---CALCULATE VALUES FOR PARAMETERS USED IN THOMAS ALGORITHM---
  180 IMK=PARAM*(T1+T2+T3+T4)
      A=T3
      B=-T3-T4-RHO-IMK-QCOF*U-ETQB
      C=T4
      W=B-A*BE(I-1)
      IF(DABS(W) .GT. 1.E-20) GO TO 185
      WRITE(P, 184) I, J, W, T1, T2, T3, T4
  184 FORMAT(' ***** IN ENTRY COL, ROW', I3, ', COL', I3,
     * ', DENOM W =', G15.7, ',' / 7X, 'T1, T2, T3, T4 =', 4G15.7)
      W = 1.D0
  185 CONTINUE
      BE(I)=C/W
      D=-T1*PHI(I,J-1)+(T1+T2-IMK)*PHI(I,J)-T2*PHI(I,J+1)-RHO*KEEP(I,J)-
     1SLEAK+RW+ETQ-SLBS
      G(I)=(D-A*G(I-1))/W
  190 CONTINUE
C
C     ---CALCULATE HEAD VALUES FOR COLUMNS OF MATRIX AND PLACE IN TEMP-
C     ORARY LOCATION TEMP;IF(J.GT.2) FIRST MAKE PHI(NO4,J-1)=TEMP(NO4)---
      NO3=DIML-2
      A2=0.0
      DO 220 KNO4=1,NC3
      NO4=DIML-KNO4
      PHI(NO4,J-1)=TEMP(NO4)
      IF (T(NO4,J)) 210,200,210
  200 TEMP(NO4)=PHI(NO4,J)
      GO TO 220
  210 TEMP(NO4)=G(NO4)-BE(NO4)*TEMP(NO4+1)
      TCHK=ABS(SNGL(TEMP(NO4))-PHE(NO4,J))
      IF (TCHK.GT.ERR) TEST=1.
      A2=A2+TCHK
  220 CONTINUE
      TEST3(KOUNT+1)=TEST3(KOUNT+1)+A2
  230 CONTINUE
      RETURN
C
C     **********************************************************************
C
C     ---CHECK FOR STEADY STATE---
C     ***********************
```

```
      ENTRY STEADY
C     *************************
      TEST2=0.
      DO 240 I=2,INO1
      DO 240 J=2,JNO1
  240 TEST2=DMAX1(TEST2,DABS(DBLE(KEEP(I,J))-PHI(I,J)))
      IF (TEST2.GE.EPCR) GO TO 250
      WRITE(P,380) KT, DAYS
      IFINAL=1
      ISTEDY=1
      GO TO 260
  250 IF (KT.EQ.NUMT) IFINAL=1
  260 IF (CHCK.EQ.CHK(5)) CALL CHECK
C
C     --- SAVE HYDROGRAPHS FOR SELECTED NODES ---
      IF (NHYDR.LE.0) GO TO 258
      DO 254 J=1,NHYDR
      IM1=IHYDR(J)
      JM1=JHYDR(J)
  254 HYDR(J)=PHI(IM1,JM1)
      JM1=IFILE(7)
      WRITE (JM1) DAYS, (HYDR(J), J=1,NHYDR)
  258 CONTINUE
C
      IF (KOUNT.LE.ITMAX) RETURN
      KOUNT=KOUNT-1
C
C     ---PRINT OUTPUT AT DESIGNATED TIME STEPS---
C     ********************
      ENTRY OUTPUT
C     ********************
      WRITE (P,390) KT,DELT,SUM,MINS,HRS,DAYS,YRS,DAYSP,YRSP,KCUNT
      IF (CHCK.EQ.CHK(5)) CALL CWRITE
      IF(JW.LE.1 .OR. (WATER.NE.CHK(2) .AND. EVAP.NE.CHK(6)))GO TO 5090
      DO 5080 I = 1, DIML
      DO 5070 J = 1, DIMW
      IF(ISUR(I, J) .GE. 2E4 .OR. WELL(I, J) .LE. 0.) GO TO 5070
      IF(WATER .NE. CHK(2) ) GO TO 5050
      IF(PHI(I, J) .LT. BOTTOM(I, J) + WELBOT) WRITE(P, 500) I, J
 5050 IF(EVAP .NE. CHK(6) ) GO TO 5070
C     IF(PHI(I, J) .LT. LAND(I, J) - WELTOP) WRITE(P, 510) I, J
 5070 CONTINUE
 5080 CONTINUE
 5090 CONTINUE
      IF(WATER .NE. CHK(2) ) GO TO 2609
      IDRY = 0
      DO 2608 I = 1, DIML
      DO 2607 J = 1, DIMW
      IF(PHI(I, J) .GE. BOTTOM(I, J) ) GO TO 2607
      IDRY = IDRY + 1
 2607 CONTINUE
 2608 CONTINUE
      WRITE(P, 530) IDRY
 2609 CONTINUE
      IF (TT.NE.0.) WRITE (P,370) TMIN,TT
      WRITE (P,340) TEST2
      KOUNT=KOUNT+1
      WRITE (P,350) (TEST3(J),J=1,KOUNT)
C
      IF (CONTR.EQ.CHK(3) .AND. .NOT.(KT.EQ.NUMT .AND. IPRODN.LT.0))
     1     CALL  PRNTA
C
      IF (KT.NE.NUMT .OR. ISVHED.NE.1) GO TO 266
C     --- SAVE HEAD AT THE END OF A PUMPING PERIOD ---
      JM1=IFILE(3)
      DO 264 I=1,DIML
      DO 263 J=1,DIMW
  263 PHE(I,J)=PHI(I,J)
  264 WRITE (JM1) (PHE(I,J), J=1,DIMW)
C
C     ---PRINT HEAD MATRIX---
  266 IF (HEAD.NE.CHK(8) .OR. (KT.EQ.NUMT .AND. IPRHED.LT.0)) GO TO 280
      WRITE (P,360) KT, DAYS
      IF (ISTEDY.EQ.1) WRITE (P,334)
      WRITE (P,336)
      DO 270 I=1,DIML
  270 WRITE(P,329) I, (PHI(I,J), J=1,DIMW)
C
```

```
C       ---PRINT DRAWDOWN---
  280 IF (NUM.NE.CHK(4) .OR. (KT.EQ.NUMT .AND. IPRDDN.LT.0)) GO TO 310
      WRITE (P,330) KT, DAYS
      IF (ISTEDY.EQ.1) WRITE (P,334)
      WRITE (P,336)
C     ********************
      ENTRY DRDN
C     ********************
      DO 300 I=1,DIML
      DO 290 J=1,DIMW
      A2=ISUR(I,J)
      IF(A2.GE.2.0E4) A2=A2-2.0E4
  290 DDN(J) = PHI(I, J) - A2
  300 WRITE(P,331) I, (DDN(J), J = 1, DIMW)
C
      IF(IPRDDN .LT. 1) GO TO 310
      WRITE(P, 302)
  302 FORMAT(1H1, 41X, 'SATURATED THICKNESS')
      WRITE(P, 336)
      DO 306 I = 1, DIML
      DO 304 J = 1, DIMW
  304 DDN(J) = PHI(I, J) - BOTTOM(I, J)
  306 WRITE(P, 308) I, (DDN(J), J = 1, DIMW)
  308 FORMAT(1H0, I5, 2X, 10F12.3 / (8X, 10F12.3) )
C LINE 23424 CONTAINS MOD 17
  310 IF(KT .LT. NUMT .OR. IPRDDN .LT. 2) GO TO 315
      IF(ICPYW .GT. 0) WRITE(KCPY, 511)
      WRITE(P, 512)
      WRITE(P, 336)
  511 FORMAT(31X, 'CALIBRATION HEAD')
  512 FORMAT(1H1, 40X, 'CALIBRATION RESIDUAL')
  514 FORMAT(8F10.0)
  516 FORMAT(1H0, 8G15.7 / (1X, 8G15.7) )
      DO 5160 I = 1, DIML
      READ(R, 514) (DDN(J), J = 1, DIMW)
      IF(ICPYW .GT. 0) WRITE(KCPY, 516) (DDN(J), J = 1, DIMW)
      DO 5140 J = 1, DIMW
      IF(DDN(J) .GE. 2E4) DDN(J) = DDN(J) - 2E4
 5140 DDN(J) = PHI(I, J) - DDN(J)
 5160 WRITE(P, 331) I, (DDN(J), J = 1, DIMW)
  315 IF(NW .GT. 0 .AND. IERR .NE. 1) CALL WELLHD
      IF (KOUNT.LE.ITMAX) RETURN
C
C       --- SAVE FINAL MASS BALANCE PARAMETERS, FINAL HEAD, AND
C           PRINT HYDROGRAPHS ---
C     ********************
      ENTRY DRY
C     ********************
      IF (PNCH.NE.CHK(1)) GO TO 323
      JM1=IFILE(5)
      WRITE(JM1)      SUM, SUMP, ETFLXT, CRET, CRETN, FLUXT, FLXNT
     * , CHST, CHDT, CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
      DO 320 I=1,DIML
  320 WRITE (JM1) (PHI(I,J), J=1,DIMW)
      ENDFILE  JM1
C
C       --- PRINT HYDROGRAPHS FOR SELECTED NODES ---
  323 IF (NHYDR.LE.0) GO TO 328
      JM1=IFILE(7)
      ENDFILE  JM1
      REWIND  JM1
      WRITE (P,430)
      WRITE (P,434) (IHYDR(I), I=1,NHYDR)
      WRITE (P,435) (JHYDR(J), J=1,NHYDR)
      WRITE (P,405)
  326 READ (JM1,END=328)     DAYS, (HYDR(J), J=1,NHYDR)
      WRITE (P,440)          DAYS, (HYDR(J), J=1,NHYDR)
      GO TO 326
C
  328 CONTINUE
      JM1=IFILE(3)
      IF (ISVHED.EQ.1 .AND. KT.EQ.NUMT)  ENDFILE JM1
C
      IF (KOUNT.GT.ITMAX)  STOP
C
      RETURN
C
C
C     ...............................................................
```

```
C
C     FORMATS:
C
C
  329 FORMAT(1H0, I5, 2X, 10F12.6 / (8X, 10F12.6) )
  330 FORMAT(1H1, 35X, 37HRESIDUAL AT THE END OF TIME STEP NO. , I4,
     $  21H  (SIMULATION TIME = , F8.2, 6H DAYS) )
  331 FORMAT(1H0, I5, 2X, 10F12.3 / (8X, 10F12.3) )
  334 FORMAT(1H0,50X, 31H*** STEADY STATE CONDITIONS *** )
  336 FORMAT( 6H    RO* )
  340 FORMAT ('0MAXIMUM CHANGE IN HEAD FOR THIS TIME STEP =',F10.3)
  350 FORMAT ('0SUM OF THE ABSOLUTE VALUE OF HEAD CHANGES FOR EACH ITERA
     1TION:'/('0',10G12.4))
  360 FORMAT(1H1,37X, 33HHEAD AT THE END OF TIME STEP NO. , I4,
     $  21H  (SIMULATION TIME = , F8.2, 6H DAYS) )
  370 FORMAT ('0DIMENSIONLESS TIME FOR THIS STEP RANGES FROM',G15.7,'  T
     1O',G15.7)
  380 FORMAT(1H-,  '***** STEADY STATE AT TIME STEP NO. ', I4,
     1  '   (SIMULATION TIME = ', F8.2, '  DAYS) *****'/)
  390 FORMAT (1H1,55X,17HTIME STEP NUMBER=,I10/50X,29HSIZE OF TIME STEP
     1IN SECONDS=,F14.2//55X,'TOTAL SIMULATION TIME IN SECONDS=',F14.2/3
     20X,8HMINUTES=,F14.2/82X,6HHOURS=,F14.2/83X,5HDAYS=,F14.2/82X,'YEAR
     3S=',F14.2///45X,'DURATION OF CURRENT PUMPING PERIOD IN DAYS=',F14.
     42/82X,'YEARS=',F14.2///55X,'ITERATION NUMBER=',I10//)
  400 FORMAT (8F10.4)
  405 FORMAT(1H )
  410 FORMAT(1H-,    '***** EXCEEDED MAX. NO. OF ITERATIONS DURING TIME S
     *TEP NO. ', I4,    '  (SIMULATION TIME = ', F8.2, '  DAYS) *****'/)
  420 FORMAT (4G20.10)
  430 FORMAT(1H1,20X,    '*** HYDROGRAPHS FOR SELECTED NODES ***')
  434 FORMAT(1H0, 9H    TIME , 13( 6H  ROW ,I3) )
  435 FORMAT(1H0, 9H   (DAYS), 13( 6H  COL ,I3) )
  440 FORMAT(1H , F9.2, 13(1X,F8.2) )
  500 FORMAT('  DRY WELL AT NODE', 2I5)
  510 FORMAT(' DEEP WELL AT NODE', 2I5)
  530 FORMAT('-', I5, ' DRY NODES.')
      END


      SUBROUTINE COEF(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RATE,
     1M,RIVER,WELL,SY,TOP,LAND,PERM,BOTTOM,QRE,IZ,JZ,IR,JR,IW,JW,IP,JP,I
     2L,JL,IC,JC,IQ,JQ,WR,NWR,IH, KKLAY,TLAY1,TLAY,MLAY)
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
C*COM01*
      COMMON /DARRAY/ RHOP(20),CHK(10),VF4(11)
C*COM02*
      COMMON /DPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH
C*COM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDCN,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
     * , WELBOT, WELTOP, SYB
C*COM04*
C*COM05*
C*COM05*
      COMMON / LAYER / SATL(20), PRMSY(10), SYPRM(10), SYC(10)
     * , NLAY, NLAYO, MSY, NSY, NSYO
C*DIM01*
      DIMENSION PHI(IZ,JZ), KEEP(IZ,JZ), PHE(IZ,JZ), STRT(IZ,JZ), ISUR(I
     1Z,JZ), T(IZ,JZ), TR(IZ,JZ), TC(IZ,JZ), S(IZ,JZ), DELX(JZ), DELY(IZ
     2), Q(IR,JR), RATE(IR,JR), M(IR,JR), RIVER(IR,JR), WELL(IW,JW), SY(
     3IP,JP), TOP(IC,JC), LAND(IL,JL), PERM(IP,JP), BOTTOM(IP,JP), QRE(I
     4Q,JQ), WR(IH), NWR(IH,2)
     5 , KKLAY(IZ,JZ), TLAY1(IZ,JZ), TLAY(IZ,JZ,MLAY)
C*DIM02*
C*TYP01*
      REAL *4KEEP,M,ISUR,LAND
C*TYP02*
      REAL *8PHI,DBLE,RHOP,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONT
     1R,EROR,LEAK,RECH
C*TYP03*
```

```
      INTEGER DIML,DIMW,R,P,PU
C*DTA01*
      DATA PIE/3.141593/
      DATA THIN / .02 /
C*DTA02*
      DATA QREB / 0. /
C*DTA03*
C*END03*
 9999 FORMAT(' SUB COEF   12/05/73 MOD 26, 27')
      WRITE(P, 9999)
      RETURN
C     ..............................................................
C     ---COMPUTE Q COEFFICIENTS---
C     ********************
      ENTRY CLAY
C     ********************
      TMIN=1.E40
      TT=0.0
      PRATE=0.
      DO 60 I=1,DIML
      DO 60 J=1,DIMW
C
C     ---SKIP COMPUTATIONS IF T, RATE OR M = 0, OR IF CONSTANT
C        HEAD BOUNDARY---
      IF (RATE(I,J).LE.0..OR.T(I,J).EQ.0..OR.M(I,J).EQ.0..OR.S(I,J).LT.0
     1.) GO TO 60
C
C     ---CHECK FOR SS = 0---
      IF (SS.NE.0.) GO TO 10
      SUMN=0.0
      DENOM=1.0
      X2=ABS(M(I,J))
      GO TO 50
C     ---CHECK WHETHER VALUE FOR Q(I,J) WILL EQUAL VALUE FOR PREVIOUS
C        NODE---
   10 X2=ABS(M(I,J))
      IF (RATE(I,J)*X2.EQ.PRATE) GO TO 50
      DIMT=RATE(I,J)*SUMP/(X2*X2*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
      CK=(2.3-PPT)/(2.*PPT)
      SUMN=0.0
      DO 30 K=1,200
      POWER=K*K*PPT
      IF (POWER.LE.150.) GO TO 20
      POWER=150
   20 PEX=EXP(-POWER)
      SUMN=SUMN+PEX
      IF (PEX.GT.0.00009) GO TO 30
      IF (K.GT.CK) GO TO 40
   30 CONTINUE
C
C     ---COMPUTE DENOMINATER DEPENDING ON VALUE OF DIMT---
   40 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---
   50 Q1=RATE(I,J)/(X2*DENOM)
      Q(I,J)=Q1+2.0*Q1*SUMN
      PRATE=RATE(I,J)*X2
   60 CONTINUE
      TMIN=TMIN*3.0
      TT=TT*3.0
      RETURN
C
C     ---COMPUTE TRANSMISSIVITY IN WT OR WT-ARTESIAN CONVERSION PROBLEM-
C     ********************
      ENTRY TRANS
C     ********************
      DO 30 I=1,DIML
      DO 30 J=1,DIMW
      IF(PERM(I, J) .EQ. 0.) GO TO 30
      HED=PHI(I,J)
```

```
      IF(CONVRT .EQ. CHK(7) ) HED = AMIN1(HED, TOP(I, J) )
      IF(HED .LT. BOTTOM(I, J) + THIN) GO TO 70
      IF(NLAY .EQ. 0) GO TO 659
      K = KKLAY(I, J)
      IF(K .LE. 1) GO TO 659
      T(I, J) = TLAY1(I, J) + PERM(I, J)*(HED - BOTTOM(I, J)-SATL(K-1) )
      GO TO 30
 659  CONTINUE
      T(I,J)=PERM(I,J)*(HED-BOTTOM(I,J))
      GO TO 30
   70 T(I, J) = PERM(I, J)*THIN
C DRY NODE MESSAGE DELETED
   30 CONTINUE
      RETURN
C
C     ---COMPUTE T COEFFICIENTS---
C     ********************
      ENTRY TCOF
C     ********************
      DO 150 I=1,INO1
      DO 150 J=1,JNO1
      DENOM=(T(I,J)*DELX(J+1)+T(I,J+1)*DELX(J))
      IF (DENOM.NE.0.) GO TO 120
      TR(I,J)=0.
      GO TO 130
  120 TR(I,J)=(2.*T(I,J+1)*T(I,J))/DENOM*FACTX
  130 DENOM=(T(I,J)*DELY(I+1)+T(I+1,J)*DELY(I))
      IF (DENOM.NE.0.) GO TO 140
      TC(I,J)=0.
      GO TO 150
  140 TC(I,J)=(2.*T(I+1,J)*T(I,J))/DENOM*FACTY
  150 CONTINUE
      RETURN
C
C     ---COMPUTE ET RATE---
C     ********************
      ENTRY ETRATE(I,J)
C     ********************
      ETQB=0.0
      ETQD=0.0
      X=LAND(I,J)
      IF(X .GT. 2E4) X = X - 2E4
      IF (PHE(I,J).LE.X-ETDIST) GO TO 160
      ETQB=QET/ETDIST
      ETQD=ETQB*(ETDIST-X)
  160 RETURN
C
C     ---COMPUTE STORAGE COEFFICIENT---
C     ********************
      ENTRY STORAG(I,J)
C     ********************
      SUBS=0.0
      IF (WATER.NE.CHK(2)) GO TO 220
      IF (CONVRT.NE.CHK(7)) GO TO 200
      X=KEEP(I,J)-PHE(I,J)
      IF (X) 170,180,180
  170 HED1=PHE(I,J)
      HED2=KEEP(I,J)
      GO TO 190
  180 HED1=KEEP(I,J)
      HED2=PHE(I,J)
  190 X2=TOP(I,J)
      IF(X2 .GT. 2E4) X2 = X2 - 2E4
      IF (HED1-X2)200,200,210
  200 STORE=SY(I,J)
      IF(KEEP(I, J) .LT. BOTTOM(I, J) .AND. SYB .GT. 0) STORE = SYB
      GO TO 240
  210 IF (X2-HED2) 220,220,230
  220 STORE=S(I,J)
      GO TO 240
  230 STORE=SY(I,J)
      SUBS=(HED1-X2)*(S(I,J)-SY(I,J))/DELT
      IF (X.LT.0.) SUBS=-SUBS
  240 RETURN
C
C     ---COMPUTE STEADY AND TRANSIENT LEAKAGE FACTOR---
C     ********************
      ENTRY LEAKAG(I,J,QCOF)
```

```
C       ********************
        HED1=STRT(I,J)
        J=1.
        HED2=0.
        X2=TOP(I,J)
        IF(X2 .GE. 2E4) GO TO 272
        IF (CONVRT.EQ.CHK(7)) HED1=AMAX1(STRT(I,J),X2)
        IF (CONVRT.NE.CHK(7)) GO TO 250
        IF (PHE(I,J).GE.X2) GO TO 250
        HED2=X2
        J=0.
  250   X=0.
        X2=ABS(M(I,J))
        IF(X2.GT.1.0E-10) X=RATE(I,J)/X2
  270   SLEAK=X*(RIVER(I,J)-HED1)+Q(I,J)*(HED1-HED2)
        QCUF=Q(I,J)
        RETURN
  272   SLEAK = 0.
        QCUF = 0.
        RETURN
C
C       ---COMPUTE WELL AND RECHARGE RATES---
C       ********************
        ENTRY SOURCE(I,J,RW)
C       ********************
        IF(JW .LT. 2) GO TO 278
        IF(WELL(I, J) .EQ. 0.) GO TO 278
        IF(ISUR(I, J) .GE. 2E4) GO TO 276
        XWELL = 0.
        IF(WATER .EQ. CHK(2) ) XWELL = BOTTOM(I, J) + WELBOT
        IF(EVAP .NE. CHK(6) ) GO TO 274
        XTOP = LAND(I, J)
        IF(XTOP .GT. 2E4) XTOP = XTOP - 2E4
        XTOP = XTOP - WELTOP
        IF(XTOP .GT. XWELL) XWELL = XTOP
  274   IF(KEEP(I, J) .LT. XWELL) GO TO 278
  276   RW = RW - WELL(I, J)/(DELX(J)*DELY(I) )
  278   CONTINUE
        IF(RECH .NE. CHK(10) ) GO TO 2798
        X = QRE(I, J)
        IF(X .GE. 0. .OR. WATER .NE. CHK(2) ) GO TO 2795
        IF(KEEP(I, J) .LT. BOTTOM(I, J) ) X = X*QREB
 2795   RW = RW - X
 2798   CONTINUE
        RETURN
C
C       ---COMPUTE APPROXIMATE HEAD FOR PUMPING WELLS---
C       ********************
        ENTRY WELLHO
C       ********************
        WRITE (P,380)
        DO 320 KW=1,NW
        IF (WR(KW).EQ.0.) GO TO 320
        I=NWR(KW,1)
        J=NWR(KW,2)
        RE=(DELX(J)+DELY(I))/9.62
        IF (WATER.NE.CHK(2)) GO TO 280
        IF (CONVRT.NE.CHK(7)) GO TO 290
        IF (PHI(I,J).LT.X2) GO TO 290
  280   HW=PHI(I,J)+WELL(I,J)*ALOG(RE/WR(KW))/(2.*PIE*T(I,J))
        GO TO 310
  290   HED=PHI(I,J)-BOTTOM(I,J)
        ARG=HED*HED+WELL(I,J)*ALOG(RE/WR(KW))/(PIE*PERM(I,J))
        IF (ARG.GT.0.) GO TO 300
        WRITE (P,390) I,J
        GO TO 320
  300   HW=SQRT(ARG)+BOTTOM(I,J)
  310   DRAW=ISUR(I,J)-HW
        WRITE (P,370) I,J,WR(KW),HW,DRAW
  320   CONTINUE
        RETURN
C
C       ---FORMATS---
C
C
  340   FORMAT ('1',50X,'DRAWDOWN WHEN WELL WENT DRY')
  350   FORMAT ('1',32X,'DRAWDOWN FOR TIME STEP ',I4, ' (SIMULATION TIME =
       * ', F8.2, 6H DAYS) )
```

```
  370 FORMAT (' ',43X,2I5,3F11.2)
  380 FORMAT ('-',50X,'HEAD AND DRAWDOWN IN PUMPING WELLS'/51X,34('-')//
     143X,'I      J    WELL RADIUS     HEAD      DRAWDOWN'//)
  390 FORMAT (' ',43X,2I5,'  WELL IS DRY')
      END


      SUBROUTINE CHECKI(PHI,KEEP,PHE,STRT,ISUR,T,TR,TC,S,DELX,DELY,Q,RAT
     1E,M,RIVER,WELL,SY,TOP,LAND,PERM,BOTTOM,QRE,IZ,JZ,IR,JR,IW,JW,IP,JP
     2,IL,JL,IC,JC,IQ,JQ)
C     ------------------------------------------------------------------
C     ---THIS SUBROUTINE COMPUTES A MASS BALANCE---
C
C     SPECIFICATIONS:
C*COM01*
      COMMON /DARRAY/ RHOP(20),CHK(10),VF4(11)
C*COM02*
      COMMON /OPARAM/ WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONTR,EROR,LE
     1AK,RECH
C*COM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDDN,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
     * , WELBOT, WELTOP, SYB
C*COM04*
      COMMON /CK/ ETFLXT, QRET, QRETN, FLUXT, FLXNT, CHST, CHDT
     * , CFLXPT, CFLXNT, PUMPT, PUMPPT, STORT
C*COM05*
C*COM06*
C*DIM01*
      DIMENSION PHI(IZ,JZ), KEEP(IZ,JZ), PHE(IZ,JZ), STRT(IZ,JZ), ISUR(I
     1Z,JZ), T(IZ,JZ), TR(IZ,JZ), TC(IZ,JZ), S(IZ,JZ), DELX(JZ), DELY(IZ
     2), Q(IR,JR), RATE(IR,JR), M(IR,JR), RIVER(IR,JR), WELL(IW,JW), SY(
     3IP,JP), TOP(IC,JC), LAND(IL,JL), PERM(IP,JP), BOTTOM(IP,JP), QRE(I
     4Q,JQ)
C*DIM02*
      DIMENSION SAVE(32), SAVER(32)
C*TYP01*
      REAL *4KEEP,M,ISUR,LAND
C*TYP02*
      REAL *8PHI,DBLE,RHOP,CHK,WATER,CONVRT,EVAP,CHCK,PNCH,NUM,HEAD,CONT
     1R,EROR,LEAK,RECH
      REAL*8 HED2
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C*DTA01*
C     CARD DELETED
      DATA CHKITR /'ITER' /
      DATA QREB / 0. /
C*DTA02*
C*DTA03*
C*END00*
 9999 FORMAT(' SUB CHECKI 10/11/78 MOD 26, 27')
      WRITE(P, 9999)
C                                          D3 CONVERTS FT**3 TO ACRE FT
C                                          D3PT CONVERTS FT**3/SEC  TO
C                                            ACRE FT PER YEAR
      D3 = 1./43560.
      D3PT = 31557600.*D3
      D3PTS = D3PT
      D3PTR = D3PT
      RETURN
C     ..................................................................
C     ********************
      ENTRY CHECK
C     ********************
C
C     ---INITIALIZE VARIABLES---
      DO 10 ISV = 1, 32
      SAVE(ISV) = 0.
   10 SAVER(ISV) = 0.
      PUMP=0.
      PUMPP = 0.
      STOR=0.
```

```
      FLUXS=0.0
      CHD1=0.0
      CHD2=0.0
      QREFLX=0.
      QREFN = 0.
      CFLUX=0.
      FLUX=0.
      ETFLUX=0.
      FLXN=0.0
      CONFXN=0.0
      CONFXP=0.0
C     .........................................................*.............
C
C     ---COMPUTE RATES,STORAGE AND PUMPAGE FOR THIS STEP---
      DO 230 I=2,DIML
      DO 230 J=2,DIMW
      IF(T(I, J) .LE. 0. .OR. S(I, J) .LT. 0.) GO TO 230
      AREA=DELX(J)*DELY(I)
C
C     ---COMPUTE PUMPAGE OR FLOW ACROSS CONSTANT FLUX BOUNDARIES---
      IF (JW.LT.2) GO TO 30
      SAVE(13) = 0.
      SAVE(14) = 0.
      SAVE(17) = 0.
      SAVE(18) = 0.
      X=WELL(I,J)
      IF(X .EQ. 0.) GO TO 30
      IF (ISUR(I,J).LT.2.0E4) GO TO 20
      CONFXN=CONFXN+AMIN1(0.0, X)
      CONFXP=CONFXP+AMAX1(0.0, X)
      IF(X .GT. 0.) SAVE(13) = X
      IF(X .LT. 0.) SAVE(14) = X
      GO TO 30
C     LINE DELETED - MOVED TO 185 05200
   20 XWELL = 0.
      IF(WATER .EQ. CHK(2) ) XWELL = BOTTOM(I, J) + WELBOT
      IF(EVAP .NE. CHK(6) ) GO TO 25
      XTOP = LAND(I, J)
      IF(XTOP .GT. 2E4) XTOP = XTOP - 2E4
      XTOP = XTOP - WELTOP
      IF(XTOP .GT. XWELL) XWELL = XTOP
   25 IF(KEEP(I, J) .LT. XWELL) GO TO 30
      PUMP = PUMP + X
      IF(X .GT. 0.) PUMPP = PUMPP + X
      IF(X .GT. 0.) SAVE(17) = X
      IF(X .LT. 0.) SAVE(18) = X
C
C     ---COMPUTE FLUX FROM (OR TO) ADJACENT CONSTANT HEAD BOUNDARIES---
   30 SAVE(9) = CHD2
      SAVE(10)= CHD1
      SAVE(31) = 0.
      SAVE(32) = 0.
      IF(S(I, J-1) .GE. 0.) GO TO 55
      X=TR(I,J-1)*(STRT(I,J-1)-PHI(I,J))*DELY(I)
      IF (X) 40,60,50
   40 CHD1=CHD1+X
      GO TO 60
   50 CHD2=CHD2+X
      GO TO 60
   55 X = TR(I, J-1)*(PHI(I, J-1) - PHI(I, J) )*DELY(I)
      IF(X .GT. 0.) SAVE(31) = SAVE(31) + X
      IF(X .LT. 0.) SAVE(32) = SAVE(32) + X
   60 IF(S(I, J+1) .GE. 0.) GO TO 85
      X=TR(I,J)*(STRT(I,J+1)-PHI(I,J))*DELY(I)
      IF (X) 70,90,80
   70 CHD1=CHD1+X
      GO TO 90
   80 CHD2=CHD2+X
      GO TO 90
   85 X = TR(I, J)*(PHI(I, J+1) - PHI(I, J) )*DELY(I)
      IF(X .GT. 0.) SAVE(31) = SAVE(31) + X
      IF(X .LT. 0.) SAVE(32) = SAVE(32) + X
   90 IF(S(I-1, J) .GE. 0.) GO TO 115
      X=TC(I-1,J)*(STRT(I-1,J)-PHI(I,J))*DELX(J)
      IF (X) 100,120,110
  100 CHD1=CHD1+X
      GO TO 120
  110 CHD2=CHD2+X
```

```
      GO TO 120
  115 X = TC(I-1, J)*(PHI(I-1, J) - PHI(I, J) )*DELX(J)
      IF(X .GT. 0.) SAVE(31) = SAVE(31) + X
      IF(X .LT. 0.) SAVE(32) = SAVE(32) + X
  120 IF(S(I+1, J) .GE. 0.) GO TO 145
      X=TC(I,J)*(STRT(I+1,J)-PHI(I,J))*DELX(J)
      IF (X) 130,150,140
  130 CHD1=CHD1+X
      GO TO 150
  140 CHD2=CHD2+X
      GO TO 150
  145 X = TC(I, J)*(PHI(I+1, J) - PHI(I, J) )*DELX(J)
      IF(X .GT. 0.) SAVE(31) = SAVE(31) + X
      IF(X .LT. 0.) SAVE(32) = SAVE(32) + X
  150 SAVE(9) = CHD2 - SAVE(9)
      SAVE(10) = CHD1 - SAVE(10)
C
      IF(RECH .NE. CHK(10) ) GO TO 160
      X = QRE(I, J)*AREA
      IF(X) 158, 160, 156
 156  QREFLX = QREFLX + X
      SAVE(3) = X
      SAVE(4) = 0.
      GO TO 160
 158  IF(WATER .NE. CHK(2) ) GO TO 159
      IF(KEEP(I, J) .LT .BOTTOM(I, J) ) X = X*QREB
  159 QREFN = QREFN + X
      SAVE(3) = 0.
      SAVE(4) = X
  160 CONTINUE
C
C     ---COMPUTE ET RATE---
      IF (EVAP.NE.CHK(6)) GO TO 170
      SAVE(1) = 0.
      X = LAND(I, J)
      IF(X .GT. 2E4) X = X - 2E4
      IF(PHI(I, J) .LT. X - ETDIST) GO TO 170
C DELETED
C DELETED
      ETQ = QET/ETDIST*(PHI(I, J) + ETDIST - X)
      ETFLUX = ETFLUX - ETQ*AREA
      SAVE(1) = -ETQ*AREA
C
C     ---COMPUTE LEAKAGE RATE---
  170 IF (LEAK.NE.CHK(9)) GO TO 180
      SAVE( 5) = 0.
      SAVE( 6) = 0.
      IF (M(I,J).EQ.0.0) GO TO 180
      IF(TOP(I,J).GE.2.E4) GO TO 180
      HED1=STRT(I,J)
      IF (CONVRT.EQ.CHK(7)) HED1=AMAX1(STRT(I,J),TOP(I,J))
      HED2=PHI(I,J)
      IF(CONVRT .NE. CHK(7)) GO TO 175
      IF(TOP(I, J) .GT. HED2) HED2 = TOP(I, J)
  175 CONTINUE
      XX=ABS(M(I,J))
      XX=RATE(I,J)*(RIVER(I,J)-HED1)*AREA/XX
      YY=Q(I,J)*(HED1-HED2)*AREA
      FLUX=FLUX+XX
C LEAKAGE FROM PREVIOUS PERIOD CONSIDERED TOGETHER WITH CURRENT
      XX = XX + YY
      FLUXS = FLUXS + XX
      IF(XX .LT. 0.) FLXN = FLXN + XX
      IF(XX .GT. 0.) SAVE(5) = XX
      IF(XX .LT. 0.) SAVE(6) = XX
      IF(ISVFLX .EQ. 1) PHE(I, J) = XX/AREA
C
C     ---COMPUTE VOLUME FROM STORAGE---
  180 STORE=S(I,J)
      IF(WATER .NE. CHK(2) ) GO TO 185
      STORE = SY(I, J)
      IF(KEEP(I, J) .LT. BOTTOM(I, J) .AND. SYB .GT. 0.) STORE = SYB
 185  CONTINUE
      IF (CONVRT.NE.CHK(7)) GO TO 220
      X=KEEP(I,J)-PHI(I,J)
      IF(X) 190, 220, 200
  190 HED1=PHI(I,J)
      HED2=KEEP(I,J)
```

```
      X=ABS(X)
      GO TO 210
  200 HED1=KEEP(I,J)
      HED2=PHI(I,J)
  210 STORE=S(I,J)
      XX=TOP(I,J)
      IF(XX.GT.2.E4)  XX=XX-2.E4
      IF (HED1-XX.LE.0.) STORE=SY(I,J)
      IF ((HED1-XX)*(HED2-XX).LT.0.0) STORE=(HED1-XX)/
     1X*S(I,J)+(XX-HED2)/X*SY(I,J)
      IF(KEEP(I, J) .LT. BOTTOM(I, J) .AND. SYB .GT. 0.) STORE = SYB
  220 STOR=STOR+STORE*(KEEP(I,J)-PHI(I,J))*AREA
      SAVE(27) = -STORE*(KEEP(I, J) - PHI(I, J) )*AREA/DELT
      SAVE(21) = SAVE(3) + SAVE(5) +SAVE( 9)+SAVE(13)+SAVE(17)+SAVE(31)
      SAVE(22) = SAVE(1) + SAVE(6) +SAVE(10)+SAVE(14)+SAVE(18)+SAVE(32)
     * + SAVE(4)
      SAVE(25) = SAVE(21) + SAVE(22)
      SAVE(29) = SAVE(25) - SAVE(27)
      ZPC1 = 100.*SAVE(29)/AMAX1(SAVE(21), SAVE(22), 1E-30)
      IF(LEAK .NE. CHK(9) ) GO TO 225
      IF(W(I, J) .EQ. 0.) GO TO 225
      DO 223 ISV = 1, 32
  223 SAVER(ISV) = SAVER(ISV) + SAVE(ISV)*D3PTR
  225 IF(NHYDR .GE. 0) GO TO 230
      IF(HEAD.NE.CHKITR.AND.MOD(KT,KTH).NE.0.AND.IFINAL.NE.1)GO TO 230
      NHYDR2 = -NHYDR
      DO 228 KHYDR = 1, NHYDR2
      IF(I .NE. IHYDR(KHYDR) ) GO TO 228
      IF(J .NE. JHYDR(KHYDR) ) GO TO 228
      DO 227 ISV = 1, 32
  227 SAVE(ISV) = SAVE(ISV)*D3PTS
      WRITE(P, 250) I, J
      WRITE(P, 240)
      WRITE(P, 241) (SAVE(ISV), ISV = 1, 20)
      WRITE(P, 244) SAVE(31), SAVE(32)
      WRITE(P, 242) (SAVE(ISV), ISV = 21, 30), ZPC1
C     DELETE CARD
  228 CONTINUE
  230 CONTINUE
C     ..................................................................
C
C     ---COMPUTE CUMULATIVE VOLUMES, TOTALS, AND DIFFERENCES---
      STORT=STORT+STOR
      STOR=-STOR/DELT
      ETFLXT=ETFLXT-ETFLUX*DELT
      FLUXT=FLUXT+FLUXS*DELT
      FLXNT=FLXNT-FLXN*DELT
      FLXPT=FLUXT+FLXNT
      QRET=QRET+QREFLX*DELT
      QRETN = QRETN + QREFN*DELT
      CHDT=CHDT-CHD1*DELT
      CHST=CHST+CHD2*DELT
      PUMPT=PUMPT-PUMP*DELT
      PUMPPT = PUMPPT + PUMPP*DELT
      CFLUX=CONFXP+CONFXN
      CFLXNT=CFLXNT-CONFXN*DELT
      CFLXPT=CFLXPT+CONFXP*DELT
C
      RETURN
C     ..................................................................
C
C     ---PRINT RESULTS---
C     ************************
      ENTRY CWRITE
C     ************************
C                                           VARIABLE 1 = RATE IN,
C                                           2 = RATE OUT, 3 = TOTAL IN,
C                                           4 = TOTAL OUT
      ZET2 = D3PT*ETFLUX
      ZET4 = D3   *(-ETFLXT)
      ZRE1 = D3PT*QREFLX
      ZRE2 = D3PT*QREFN
      ZRE3 = D3   *QRET
      ZRE4 = D3   *QRETN
      ZLK1 = D3PT*(FLUXS - FLXN)
      ZLK2 = D3PT*FLXN
      ZLK3 = D3   *FLXPT
      ZLK4 = D3   *(-FLXNT)
```

```
      ZLK3 = D3PT*FLUX
      ZCH1 = D3PT*CHD2
      ZCH2 = D3PT*CHD1
      ZCH3 = D3   *CHST
      ZCH4 = D3   *(-CHDT)
      ZCF1 = D3PT*CONFXP
      ZCF2 = D3PT*CONFXN
      ZCF3 = D3   *CFLXPT
      ZCF4 = D3   *(-CFLXNT)
      ZP1 = D3PT*PUMPP
      ZP2  = D3PT*(PUMP - PUMPP)
      ZP3 = D3   *PUMPPT
      ZP4  = D3   *(-PUMPT - PUMPPT)
      ZT1 = ZRE1 + ZLK1 + ZCH1 + ZCF1 + ZP1
      ZT2 = ZET2 + ZLK2 + ZCH2 + ZCF2 + ZP2
     * + ZRE2
      ZT3 = ZRE3 + ZLK3 + ZCH3 + ZCF3 + ZP3
      ZT4 = ZET4 + ZLK4 + ZCH4 + ZCF4 + ZP4
     * + ZRE4
      ZN1 = ZT1 + ZT2
      ZN3 = ZT3 + ZT4
      ZST2 = D3PT*STOR
      ZST4 = D3   *(-STORT)
      ZE1 = ZN1 - ZST2
      ZE3 = ZN3 - ZST4
      ZPC1 = 100.*ZE1/AMAX1(ZT1, ZT2, 1E-30)
      ZPC3 = 100.*ZE3/AMAX1(ZT3, ZT4, 1E-30)
      WRITE(P, 240)
      IF(ZP3  - ZP4  .GT. 0.) WRITE(P, 2410) ZP1 , ZP2 , ZP3 , ZP4
      IF(ZLK3 - ZLK4 .GT. 0.) WRITE(P, 2411) ZLK1, ZLK2, ZLK3, ZLK4
      IF(ZCF3 - ZCF4 .GT. 0.) WRITE(P, 2412) ZCF1, ZCF2, ZCF3, ZCF4
      IF(ZCH3 - ZCH4 .GT. 0.) WRITE(P, 2413) ZCH1, ZCH2, ZCH3, ZCH4
      IF(     - ZET4 .GT. 0.) WRITE(P, 2414)       ZET2,       ZET4
      IF(ZRE3 - ZRE4 .GT. 0.) WRITE(P, 2415) ZRE1, ZRE2, ZRE3, ZRE4
      WRITE(P, 242) ZT1, ZT2, ZT3, ZT4, ZN1, ZN3
     * , ZST2, ZST4, ZE1, ZE3, ZPC1, ZPC3
C     --- WRITE ON DISK FILE THE LEAKAGE RATE AT EACH NODE ---
      IF (ISVFLX.NE.1) GO TO 310
      ISVFLX=0
      WRITE (P,9110) KT, DAYS
      JM1=IFILE(1)
      DO 280 I=1,DIML
      DO 279 J=1,DIMW
      IF (T(I,J).LT.1.0E-10 .OR. S(I,J).LT.0.0) PHE(I,J)=0.0
  279 CONTINUE
      WRITE (JM1) (PHE(I,J), J=1,DIMW)
      WRITE (P,9100) I, (PHE(I,J), J=1,DIMW)
  280 CONTINUE
      ENDFILE  JM1
C
  310 CONTINUE
      RETURN
C
C     FORMATS:
C
C
  240 FORMAT('0', 25X, 'M A S S   B A L A N C E' / 1X, 72('-'),
     * / ' TYPE OF FLOW', 4X, 'CURRENT RATE (ACRE FT/YR)',
     *  3X, 'CUMULATIVE (ACRE FT)' / 20X, 'INFLOW', 8X, 'OUTFLOW',
     *  9X, 'INFLOW', 8X, 'OUTFLOW' / 1X, 72('-') )
  241 FORMAT(' EVAPOTRANS.', 10X, F20.4, 10X, F20.4
     * / ' RECHARGE', F19.4, 10X, F20.4
     * / ' LEAKAGE', 6X, 2F14.4, '* ', 2F14.4
     * / ' CONSTANT HEAD', 2F14.4, 2X, 2F14.4
     * / ' CONSTANT FLUX', 2F14.4, 2X, 2F14.4
     * / ' PUMPING', 6X, 2F14.4, 2X, 2F14.4 )
 2410 FORMAT(' PUMPING       ', 2F14.0, 2X, 2F14.0)
 2411 FORMAT(' LEAKAGE       ', 2F14.0, 2X, 2F14.0)
 2412 FORMAT(' CONSTANT FLUX', 2F14.0, 2X, 2F14.0)
 2413 FORMAT(' CONSTANT HEAD', 2F14.0, 2X, 2F14.0)
 2414 FORMAT(' EVAPOTRANS.', 16X, F14.0, 16X, F14.0)
 2415 FORMAT(' RECHARGE      ', 2F14.0, 2X, 2F14.0)
  242 FORMAT(16X, 12('-'), 2X, 12('-'), 4X, 12('-'), 2X, 12('-')
     * / ' TOTAL', 8X, 2F14.0, 2X, 2F14.0
     * / 'ONET INFLOW', 3X, F14.0, 16X, F14.0
     * / ' STORAGE INCR.', F14.0, 16X, F14.0
     * / 16X, 12('-'), 18X, 12('-')
     * / ' ERROR', 8X, F14.0, 16X, F14.0
```

```
     * / ' PERCENT ERROR', F14.2, '% ', 14X, F14.2, '%'
C     DELETE CARD
     * / 1X, 72('-') )
C     DELETE CARD
  244 FORMAT(' LATERAL FLOW ', 2F14.4)
  250 FORMAT('-', 27X, 'NODE', 2I4)
  251 FORMAT('-', 27X, 'RIVER NODES')
 9100 FORMAT (1H0, I5, 2X, 1P10E12.4 / (8X, 1P10E12.4))
 9110 FORMAT(1H1,34X,    'LEAKAGE RATE (FT/SEC) COMPUTED AT THE END OF TI
     $ME STEP NO. ',   I4, 21H  (SIMULATION TIME = , F8.2, 6H DAYS) /
     $  64    RO# )
C
      END


      SUBROUTINE PRNTAI(PHI,ISUR,S,DELX,DELY,WELL,IZ,JZ,IW,JW)
C     --------------------------------------------------------------
C
C     SPECIFICATIONS:
C*COM01*
C*COM02*
C*COM03*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRDDN,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
C*COM04*
C*COM05*
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(4),XN1,SYM(23),PRNT(122),BLA
     1NK(50),DIGIT(122),VF1(6),VF2(6),VF3(7),NA(4),XN(100),YN(13),XSF,NX
     2D,WIDTH,SPACNG,N1,N2,N3,N4,N6,N8,NC
C*COM05*
C*DIM01*
      DIMENSION PHI(IZ,JZ), ISUR(IZ,JZ), S(IZ,JZ), DELX(JZ), DELY(IZ), W
     1ELL(IW,JW)
C*DIM02*
C*TYP01*
      REAL *4K,ISUR
C*TYP02*
      REAL *8XLABEL,YLABEL,TITLE,XN1,Z,PHI
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C*DTA01*
C*DTA02*
C*DTA03*
C*END00*
      RETURN
C     ......................................................................
C     ********************
      ENTRY PRNTA
C     ********************
C     ---INITIALIZE VARIABLES---
      DIST=WIDTH-DELX(JNO1)/2.
      JJ=JNO1
      LL=1
      Z=NXD*XSF
      WRITE (P,150) (TITLE(I),I=1,4)
      DO 140 I=1,N4
C
C     ---LOCATE X AXES---
      IF (I.EQ.1.OR.I.EQ.N4) GO TO 10
      PRNT(1)=SYM(23)
      PRNT(N8)=SYM(23)
      IF ((I-1)/N1*N1.NE.I-1) GO TO 30
      PRNT(1)=SYM(25)
      PRNT(N8)=SYM(25)
      GO TO 30
C
C     ---LOCATE Y AXES---
   10 DO 20 J=1,N8
      IF ((J-1)/N2*N2.EQ.J-1) PRNT(J)=SYM(25)
   20 IF ((J-1)/N2*N2.NE.J-1) PRNT(J)=SYM(24)
C
C     ---COMPUTE LOCATION OF NODES AND DETERMINE APPROPRIATE SYMBOL---
   30 IF (DIST.LT.0..OR.DIST.LT.Z-XN1*XSF) GO TO 90
```

```
      YLEN=DELY(2)/2.
      DO 90 L=2,INO1
      J=YLEN*N2/XSF+1.5
      K=(ISUR(L,JJ)-PHI(L,JJ))/SPACNG
      IF (K) 60,50,40
   40 K=AMOD(K,20.)
      N=K
      PRNT(J)=SYM(N+1)
      GO TO 70
   50 PRNT(J)=SYM(26)
      GO TO 70
   60 PRNT(J)=SYM(22)
   70 IF (S(L,JJ).LT.0.) PRNT(J)=SYM(27)
      IF (J.EQ.1) GO TO 80
      IF (WELL(L,JJ).NE.0.) PRNT(J)=SYM(28)
   80 YLEN=YLEN+(DELY(L)+DELY(L+1))/2.
      DIST=DIST-(DELX(JJ)+DELX(JJ-1))/2.
      JJ=JJ-1
   90 CONTINUE
C
C     ---PRINT AXES,LABELS, AND SYMBOLS---
      IF (I-NA(LL).EQ.0) GO TO 110
      IF ((I-1)/N1*N1-(I-1)) 120,100,120
  100 WRITE (P,VF1) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XN(1+(I-1)/6)
      GO TO 130
  110 WRITE (P,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XLABEL(LL)
      LL=LL+1
      GO TO 130
  120 WRITE (P,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8)
C
C     ---COMPUTE NEW VALUE FOR Z AND INITIALIZE PRNT---
  130 Z=Z-2.*XN1*XSF
      DO 140 J=1,N8
  140 PRNT(J)=SYM(11)
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,170) (YLABEL(I),I=1,6)
      WRITE (P,160) SPACNG
      RETURN
C
C     ---FORMATS---
C
C
  150 FORMAT ('1',46X,4A8//)
  160 FORMAT ('-LEGEND'/' ------'//' R = CONSTANT HEAD BOUNDARY'/' W = W
     1ELL LOCATION'/' 0 = NO-FLOW BOUNDARY'/' * = CONE OF IMPRESSION'//'
     2 CONTOUR INTERVAL =',F10.2//' THE FOLLOWING 20 SYMBOLS, STARTING W
     3ITH BLANK, ARE CYCLED:   1 2 3 4 5 6 7 8 9   1 3 3 D 5 F 7 H 9')
  170 FORMAT ('0',39X,6A8)
      END
```

```
      BLOCK DATA
C     ----------
C*COM01*
      COMMON /DARRAY/ RHOP(20), CHK(10), VF4(11)
C*COM02*
      COMMON /SPARAM/ SLEAK,U,SS,TT,TMIN,ETDIST,QET,IFINAL,TMAX,CDLT,DEL
     1T,SUM,SUMP,NUMT,KT,KP,NPER,KTH,ITMAX,LENGTH,NWEL,NW,ERR,DIML,DIMW,
     2JNO1,INO1,R,P,PU,SUBS,STORE,TEST,ETQB,ETQD,FACTX,FACTY,IERR,
     3DAYS,YRS,SSRIV,CFLEAK,QSUM,KEDIT,MESSG,IPRODN,IPRHED,ISVHED,ISVFLX
     4,IFILE(10), KOUNT, KSTEP, NHYDR,IHYDR(13),JHYDR(13),HYDR(13)
     * , WELBOT, WELTOP, SYB
C*COM04*
C*COM05*
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(4),XN1,SYM(28),PRNT(122),BLA
     1NK(60),DIGIT(122),VF1(6),VF2(6),VF3(7),NA(4),XN(100),YN(13),XSF,NX
     2D,WIDTH,SPACNG,N1,N2,N3,N4,N6,N8,NC
C*COM06*
C*TYP01*
C*TYP02*
      REAL *8RHOP,CHK,XLABEL,YLABEL,TITLE,XN1
C*TYP03*
      INTEGER DIML,DIMW,R,P,PU
C     *******************************************************************
C*DTA01*
      DATA CHK/'PUNCH','WATERTAB','CONTOUR','NUMERIC','CHECK','EVAPOTRA'
     1,'CONVERT','HEAD','LEAKAGE','RECHARGE'/,R,P,PU/5,6,7/
C*DTA02*
      DATA SYM/1H ,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H ,1H1,1H8,1H3,1
     1HD,1H5,1HF,1H7,1HH,1H9,1H ,1H*,1H|,1H-,1H+,1HO,1HR,1HW/
C*DTA03*
      DATA PRNT/122*' '/,N1,N2,N3,XN1/6,10,133,.08333333333D0/,BLANK/60*'
     1 '/,NA(4)/1000/
C*DTA04*
      DATA XLABEL/' X DIS- ','TANCE IN',' MILES '/,YLABEL/'DISTANCE','
     1FROM OR','IGIN IN ','Y DIRECT','ION, IN ','MILES    '/,TITLE/'ALPHA
     2MER','IC CONTO','URS OF D','RAWDOWN '/
C*DTA05*
      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'/
C*DTA06*
      DATA VF1/'(1H ',' ',' ',' ','A1,F','10.2',')'/
C*DTA07*
      DATA VF2/'(1H ',' ',' ',' ','A1,1','X,A3',')'/
C*DTA08*
      DATA VF3/'(1H0',' ',' ',' ','A1,F','3.1,','12F1','0.2)'/
C*DTA09*
C*COM03*
      DATA VF4/'(1H0',' ',' ',' ','X,I2',',2X,','20F6','.1/(',' ','X,2
     10','F6.1','))'/
C*DTA10*
C*END00*
C     *******************************************************************
      END
```

```
 3420 IF(VUTQ .GE. 10) CALL PLTTQ( 3420)
      RETURN
C                                           3.4 CQ EXIT = 2000 LOOPSQ
C                                           1.5 CX EOD
 3500 IF(VUTQ .GE. 10) CALL PLTTQ( 3500)
      IE = IES
      GO TO 2100
C                                           2.5 CM
 3520 IF(VUTQ .GE. 10) CALL PLTTQ( 3520)
      IF(VUTX .GE.  5) WRITE(UVC, 352)
 352  FORMAT(' EOD ON CONTROL INPUT')
      IF(NW .GE. LW) GO TO 3530
      IF(NAB .GE. LAB) GO TO 3530
      IF(NSAB .GT. 0) GO TO 3530
      IF(LE .NE. 0) LPLT = LE
      IF(ME .NE. 0) MPLT = ME
      IF(KE .NE. 0) IES = IIE(MOD(KE, MNIE) + 1)
C                                           STORE EXIT COMMAND
      JE = 4
      CALL PLTEQ
      NE = NE - 1
C                                           START EXECUTION
      VQ = 1
      RETURN
 3530 CALL PLTEE(5,  3530,   70)
      WRITE(UVC, 353)
 353  FORMAT(' EXECUTION INHIBITED DUE TO ERRORS.')
      GO TO 3900
C                                           3.5 CQ EOD
 3540 CALL PLTEE(4,  3540,   80)
      WRITE(UVC, 354)
 354  FORMAT(' MISSING END')
      GO TO 3900
C                                           1.6 CX TRACEM = SAB
C                                           2.6 CM TRACEM
C                                           3.6 CQ TRACEM
C                                           (4.3 SX TRACE)
 3600 IF(VUTQ .GE. 20) CALL PLTTQ( 3600)
      LE = MOD(LE, 5) + 1
      GO TO (3610, 3620, 3630, 3640, 3650), LE
      GO TO 9000
C                                           .6.0 LQ, MQ, KQ, IIE
 3610 IF(VUTQ .GE. 10) CALL PLTTQ( 3610)
      IF(VE .NE. 0) VUTI = VE
      IF(WE .NE. 0.)  VUTQ = WE
      IF(XE .NE. 0.)  VUTX = XE
      IF(ME .NE. -1) GO TO 3614
      WRITE(UVC, 361) ILQ, LQ, (LLQ(I), I = 1, MNLQ)
      WRITE(UVC, 361) ILQ, MQ, (MMQ(I), I = 1, MNLQ)
C     WRITE(UVC, 361) ILQ, KQ, (KKQ(I), I = 1, MNLQ)
 361  FORMAT(' LQ, MQ', T10, 2I5, (T20, 20I5) )
 3614 IF(ME .EQ. 1) WRITE(UVC, 130) LPLT, MPLT, (IIOO(I), I = 1, 9)
      IF(KE .NE. -1) GO TO 3616
      WRITE(UVC, 362) IE, IES, (IIE(I), I = 1, MNIE)
 362  FORMAT(' IIE', T10, 2I5, (T20, 20I5) )
 3616 IF(KE .EQ. 1) WRITE(UVC, 363) IE, YE
 363  FORMAT(' TRACE AT', I4, ' Y =', F10.4)
      GO TO 2100
C                                           .6.1 PLTET
 3620 IF(VUTQ .GE. 10) CALL PLTTQ( 3620)
      CALL PLTET(ME, KE, VE)
      GO TO 2100
C                                           .6.2 PLTAT
 3630 IF(VUTQ .GE. 10) CALL PLTTQ( 3630)
      CALL PLTAT(ME, KE, VE)
      GO TO 2100
C                                           .6.3 PLTUT
 3640 IF(VUTQ .GE. 10) CALL PLTTQ( 3640)
      CALL PLTUT(ME, KE, VE)
      GO TO 2100
C                                           .6.4 PLTFT
 3650 IF(VUTQ .GE. 10) CALL PLTTQ( 3650)
      CALL PLTFT(ME, KE, VE)
      GO TO 2100
C                                           .6.5 PLTPT
 3660 IF(VUTQ .GE. 10) CALL PLTTQ( 3660)
C     CALL PLTPT(ME, KE, VE)
      GO TO 2100
```

APPENDIX B

PRINT - PLOT PACKAGE LISTING

```
C                                                  GWPLT
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      DIMENSION RRRR(54, 60, 40), DDD(54, 60), XXDEL(60), YYDEL(54)
      EQUIVALENCE (RRRR, IIII)
      DATA MNI, MNJ, MNP, MNV / 54, 60, 40, 5 /
      DATA UVC / 6 /
      WRITE(UVC, 1) MNI, MNJ
    1 FORMAT('      GWPLT    2/11/73 MOD 27A'
     * , 20X, 'MAXIMUM DIMENSIONS', 2I5)
      CALL PLTAI(UVC, RRRR, MNI, MNJ, MNP, MNV, XXDEL, YYDEL, DDD, MND
     * , IIII)
      WRITE(UVC, 2)
    2 FORMAT('1 PRINT/PLOT PROGRAM')
      CALL PLTXI(5, 5, 10, 10, 1., 1.)
      WRITE(UVC, 300)
  300 FORMAT('-PRINT/PLOT RETURNED TO MAIN PROGRAM')
      STOP
      END


      SUBROUTINE PLTXI(UC0, UVC0, NI0, NJ0, XDEL0, YDEL0)
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      COMMON / PLTC / LPLT, MPLT, IIDO(9)
      COMMON / PLTCX / UC, UCQ, BCC, UVC, BW, VUTI, VUTQ, VUTX
     * , NW, MW, NAB, MAB, NSAB, KAB
     * , IMAP, LI, MI, KI, VI, NI, JMAP, LJ, MJ, KJ, VJ, NJ
     * , VG, JOP, KOP, IE, NE, MNE, JE, LE, ME, KE, VE, WE, XE, YE
     * , XDEL, YDEL
      DIMENSION JJJQ(10, 3), LLQ(10), MMQ(10), KKQ(10), IIE(20)
      EQUIVALENCE (IIDO(1), IPLT), (IIDO(7), IVAR)
      EQUIVALENCE (IIDO(3), TI  ), (IIDO(9), TPI )
C                                                 VERSION AND CORRECTION DATE
C                                                 ***   PLEASE MAINTAIN    ***
      DATA VERS, TCORR / 0, 77.1108 /
      DATA LLQ, MMQ, KKQ / 30*0 /
      DATA MNJOP, MNVG, MNLQ, MNIE / 10, 3, 10, 20 /
      DATA IIE /20*0 /
      DATA NEAB / 0 /
      MOD(I, N) = IABS(I + N - ( (I + N)/N)*N)
C                                                 STORE PARMS
      UC = UC0
      BW = UVC0 .GE. 0
      IF(UVC0 .GT.0)UVC=UVC0
      NI = NI0
      NJ = NJ0
      XDEL = XDEL0
      YDEL = YDEL0
      WRITE(UVC, 101) VERS, TCORR
  101 FORMAT('-', T30, 'PRINT/PLOT PACKAGE - VERSION', I3, ' CORRECTED'
     * , F10.4 ///)
C                                                 INIT COMMON
      UCQ = UC
      KAB = 0
      VAB = 2
      LW = 10
      MW = 100
      NW = 0
      LAB = 1
      MAB = 10
      NAB = 0
      NSAB = 0
      IMAP = 1
      LI = 1
      MI = NI
      KI = 1
      VI = NI
      JMAP = 1
      LJ = 1
      MJ = NJ
      KJ = 1
```

```
      VJ = NJ
      NE = 0
      IE = 0
      IES = 0
      JE = 1
C                                            INIT LOCAL
      NEAB = MNE
C                                            INIT TRACE
      CALL PLTTI(UVC)
      CALL PLTTM(1000)
C                                            LOAD TABLES
 121  FORMAT(16I5)
 122  FORMAT(' TABLE', T10, 16I5)
 123  FORMAT(36I2)
 124  FORMAT(' TABLE', T10, 36I2)
      READ(JC, 121) LE, VE, VUTI, VUTQ, VUTX, ME
      IF(VE .GT. 0) WRITE(UVC, 122) LE, VE, VUTI, VUTQ, VUTX, ME
      IF(LE .GT. 0) JC = LE
      BCC = VE .GE. 10
      CALL PLTEI
      READ(JC, 121) NJOP, NVQ
      IF(VE .GE. 10) WRITE(UVC, 122) NJOP, NVQ
      IF(NJOP .LT. 0) GO TO 1450
      READ(JC, 123) ( (JJJQ(JOP, VQ), VQ = 1, NVQ), JOP = 1, NJOP)
      IF(VE .GE. 20) WRITE(UVC, 124)
     *               ( (JJJQ(JOP, VQ), VQ = 1, NVQ), JOP = 1, NJOP)
      GO TO 1490
 1450 CONTINUE
      NJOP = -NJOP
      DO 1460 JOP = 1 , NJOP
      READ(JC, 121) (JJJQ(JOP, VQ), VQ = 1, NVQ)
      IF(VE .GE. 20) WRITE(UVC, 122) (JJJQ(JOP, VQ), VQ = 1, NVQ)
 1460 CONTINUE
 1490 CONTINUE
      IF(ME .GT. 0) WRITE(ME, 121) NJOP, NVQ
      IF(ME .GT. 0) WRITE(ME, 123)
     *               ( (JJJQ(JOP, VQ), VQ = 1, NVQ), JOP = 1, NJOP)
C                                            RESTART
 1600 IF(VUTX .GE. 40) CALL PLTTQ( 1600)
      ILQ = 0
      LQ = 0
      MQ = 0
      KQ = 0
      VQ = 2
      NEV = 0
      GO TO 2100
C
C                                            ENTRY PLTX
      IF(VUTQ .GT. 0) CALL PLTTM(1800)
      IF(VUTI .GE. 10) WRITE(UVC, 180) LPLT, MPLT, IIDC
 180  FORMAT('UTIME', T10, 2I5, 7I10, 2F15.5)
      IF(KAB) 9000, 2200, 8900
C                                            LOOPEQ
 2000 IF(VUTX .GE. 40) CALL PLTTQ( 2000)
      CALL PLTEQ
C                                            LOOPER
 2100 IF(VUTX .GE. 40) CALL PLTTQ( 2100)
      IF(KAB) 9000, 2120, 8900
 2120 CONTINUE
      CALL PLTER
C                                            LOOPJQ
 2200 CONTINUE
      JQ = JJJQ(JOP, VQ)
      IF(VUTX .GE. 100) WRITE(UVC, 240) VQ, JOP, JQ
 240  FORMAT(' JQ', T10, 16I5)
 2500 CONTINUE
      GO TO (2510, 2520, 2530, 2540, 2000, 2560, 2570, 2580, 2590, 2600
     * , 2610), JQ
      CALL PLTTQ(2500)
      GO TO 9000
C                                            FIRST LEVEL EXITS
C                                            1.1/1 CX CONTROL
 2510 IF(VUTQ .GE. 40) CALL PLTTQ( 2510)
      GO TO (3100, 3200, 3300, 3400, 3500, 9000, 3700, 3800, 9000, 4000
     * , 1600, 4400, 9000), KOP
      GO TO 9000
C                                            1.2/2 CM
```

```
 2520 IF(VUTQ .GE. 40) CALL PLTTQ( 2520)
      GO TO (3120, 3220, 3300, 3420, 3520, 3600, 3720, 3820, 3920, 4020
     * , 4120, 5720, 9000), KOP
      GO TO 9000
C                                           1.3/3 CQ
 2530 IF(VUTQ .GE. 40) CALL PLTTQ( 2530)
      GO TO (3140, 3220, 3300, 2000, 3540, 3600, 3740, 3840, 3920, 2000
     * , 2000, 3740, 9000), KOP
      GO TO 9000
C                                           2.1,2.2/4 SX STANDARD
 2540 IF(VUTQ .GE. 40) CALL PLTTQ( 2540)
      GO TO (5100, 2100, 3600, 5400, 5500, 5600, 5700, 5700, 5700, 5700
     * , 5700, 6200, 6300, 6400, 6500), KOP
      GO TO 9000
C                                           2.3,ETC/5 SQ = LOOPSQ
C                                           3.1,3.2/6 AX ARITHMETIC
 2560 IF(VUTQ .GE. 40) CALL PLTTQ( 2560)
      CALL PLTAX
      GO TO 2100
C                                           4.2,4.3/7 UM UNFORMATTED
 2570 IF(VUTQ .GE. 40) CALL PLTTQ( 2570)
      CALL PLTUM
      GO TO 2100
C                                           5.1,5.2/8 UX UNFORMATTED
 2580 IF(VUTQ .GE. 40) CALL PLTTQ( 2580)
      CALL PLTUX
      GO TO 2100
C                                           6.1,6.2/9 UF FORMATTED
 2590 IF(VUTQ .GE. 40) CALL PLTTQ( 2590)
      CALL PLTUF
      GO TO 2100
C                                           7.1,7.2/10 FX FORMATTED
 2600 IF(VUTQ .GE. 40) CALL PLTTQ( 2600)
      CALL PLTFX
      GO TO 2100
C                                           8.1,8.2/11 PX PLOTTER
 2610 IF(VUTQ .GE. 40) CALL PLTTQ( 2610)
C     CALL PLTPX
      GO TO 2100
C                                           SECOND LEVEL EXITS
C                                           1.1 CX UNRECOGNIZED (JE = 1)
 3100 CALL PLTEE(5,   3100,   10)
      WRITE(UVC, 310)
 310  FORMAT(' ERROR IN STORED PROGRAM')
      GO TO 8900
C                                           2.1 CM
 3120 CALL PLTEE(3,   3120,   20)
      WRITE(UVC, 312)
 312  FORMAT(' INVALID COMMAND CODE')
      GO TO 2100
C                                           3.1 CQ
 3140 CALL PLTEE(3,   3140,   20)
      WRITE(UVC, 312)
      GO TO 2000
C                                           1.2 CX UNSUPPORTED (JE = 2)
 3200 CALL PLTEE(5,   3200,   30)
      WRITE(UVC, 320)
 320  FORMAT(' UNDEFINED LABEL')
      GO TO 8900
C                                           2.2 CM
C                                           3.2 CX
 3220 CALL PLTEE(2,   3220,   40)
      WRITE(UVC, 322)
 322  FORMAT(' UNSUPPORTED OP CODE')
      GO TO 2100
C                                           1.3 CX ABORT (JE = 3)
C                                           2.3 CM
C                                           3.3 CQ
 3300 IF(VUTQ .GE. 10) CALL PLTTQ( 3300)
      IF(VUTX .GE.  5) WRITE(UVC, 330)
 330  FORMAT(' ABORT COMMAND')
      GO TO 8900
C                                           1.4 CX EXIT (JE = 4)
 3400 IF(VUTQ .GE. 10) CALL PLTTQ( 3400)
C                                           SET EOD
      KOP = 5
      RETURN
C                                           2.4 CM
```

```
C                                              1.7 CX BEGIN
 3700 IF(VOUTQ .GE. 10) CALL PLTTQ( 3700)
C                                              PUSH Q = BEGIN, END, WAIT
      IF(ILQ .GE. MNLQ) GO TO 3710
      ILQ = ILQ + 1
      LLQ(ILQ) = LQ
      MMQ(ILQ) = MQ
      LQ = IE
      MQ = VE
C     KQ = MIN(KQ, KE)
      GO TO 2100
 3710 CALL PLTEE(5,  3710, 100)
      WRITE(UVC, 371)
 371  FORMAT(' EXCESSIVE BEGINS FOR RUN TIME STACK')
      GO TO 2100
C                                              2.7 CM BEGIN
 3720 IF(VOUTQ .GE. 10) CALL PLTTQ( 3720)
      VQ = 3
C                                              3.7 CQ BEGIN
 3740 IF(VOUTQ .GE. 10) CALL PLTTQ( 3740)
C                                              PUSH Q = BEGIN, 0, 0
      IF(ILQ .GE. MNLQ) GO TO 3750
      ILQ = ILQ + 1
      LLQ(ILQ) = LQ
      MMQ(ILQ) = MQ
C                                              LQ = LOC WHERE BEGIN WILL BE
      LQ = NE + 1
      MQ = 0
C     KQ = 0
C     KE = 0
      VE = 0
      GO TO 2000
 3750 CALL PLTEE(4,  3750, 100)
      WRITE(UVC, 371)
      GO TO 2000
C                                              1.3 CX END
 3800 IF(VOUTQ .GE. 10) CALL PLTTQ( 3800)
      IF(ILQ .EQ. 0) GO TO 3810
      LQ = LLQ(ILQ)
      MQ = MMQ(ILQ)
      ILQ = ILQ - 1
      GO TO 2100
 3810 CALL PLTEE(5,  3810, 110)
      WRITE(UVC, 381)
 381  FORMAT(' UNMATCHED END')
      GO TO 3900
C                                              2.8 CM END
 3820 CALL PLTEE(3,  3820, 110)
      WRITE(UVC, 381)
      GO TO 2100
C                                              3.8 CQ END
 3840 IF(VOUTQ .GE. 10) CALL PLTTQ( 3840)
      IF(ILQ .EQ. 0) GO TO 9000
C                                              STORE END ADDRESS IN BEGIN
      CALL PLTEV(LQ, NE)
      LQ = LLQ(ILQ)
      MQ = MMQ(ILQ)
      ILQ = ILQ - 1
      IF(ILQ .LE. 0) VQ = 2
      GO TO 2000
C                                              1.9 CX LABEL = 9000 SAB
C                                              2.9 CM LABEL
C                                              3.9 CQ
 3920 IF(VOUTQ .GE. 10) CALL PLTTQ( 3920)
      LE = MOD(LE, MNIE) + 1
      IIE(LE) = NE
      GO TO 2100
C                                              1.10 CX GO TO
 4000 IF(VOUTQ .GE. 10) CALL PLTTQ( 4000)
      IE = IIE(MOD(LE, MNIE) + 1)
      GO TO 2100
C                                              2.10 CM GO TO
 4020 IF(VOUTQ .GE. 10) CALL PLTTQ( 4020)
      I = IIE(MOD(LE, MNIE) + 1)
      IF(I .GT. MNE) GO TO 4030
      NE = I
      GO TO 2100
 4030 CALL PLTEE(4,  4030,  30)
```

```
      WRITE(JVC, 403)
  403 FORMAT(' UNDEFINED LABEL')
      GO TO 2100
C                                                 3.10 CQ GO TO = 2000 LOOPSQ
C                                                 1.11 CX START = 1600 RESTART
C                                                 2.11 CM START
 4120 IF(VJTQ .GE. 10) CALL PLTTQ( 4120)
      IF(VUTX .GE.  5) WRITE(UVC, 412)
  412 FORMAT(' START COMMAND')
      IF(NW .GE. LW) GO TO 4130
      IF(NAB .GT. LAB) GO TO 4130
      IF(NSAB .GT. 0) GO TO 4130
      IF(LE .NE. 0) LPLT = LE
      IF(ME .NE. 0) MPLT = ME
      IF(KE .NE. 0) IES = IIE(MOD(KE, MNIE) + 1)
      CALL PLTEQ
      NE = NE - 1
      IE = IES
      VQ = 1
      GO TO 2100
 4130 CALL PLTEE(5,  4130,   70)
      WRITE(UVC, 353)
      GO TO 3900
C                                                 3.11 CQ START
C                                                 1.12 CX PRINTMAP
 4400 IF(VJTQ .GE. 10) CALL PLTTQ( 4400)
      IF(ILQ .GE. MNLQ) GO TO 4490
      ILQ = ILQ + 1
      LLQ(ILQ) = LQ
      MMQ(ILQ) = MQ
CL    KKQ(ILQ) = KQ
      LQ = IE
      MQ = VE
      DO 4450 IMAP = LI, MI, KI
      IE = LQ
 4450 CONTINUE
      CALL PLTEX
      CALL PLTUF
      IF(KAB) 9000, 4470, 8900
 4470 CONTINUE
      IF(IE .LT. MQ) GO TO 4450
 4480 CONTINUE
      IMAP = LI
      GO TO 2100
 4490 CALL PLTEE(5, 4490, 100)
      WRITE(UVC, 371)
      GO TO 3900
C                                                 2.12 CM PRINTMAP = 3700 BEGIN
C                                                 3.12 CQ PRINTMAP = 3700 BEGIN
C                                                 1.13 CX MAP
C                                                 UNSUPPORTED = 9000 SAB
C
C                                                 4 SX STANDARD EXITS
C                                                 4.1 STOP
 5100 IF(VUTQ .GE. 10) CALL PLTTQ( 5100)
      IF(VJTX .GE.  5) WRITE(JVC, 510)
  510 FORMAT(' STOP COMMAND')
      VAB = LE
      GO TO 3940
C                                                 4.2 NOP = 2100 LOOPER
C                                                 4.3 TRACE = 3600 2.6 TRACEM
C                                                 4.4 TESTAB
 5400 IF(VUTQ .GE. 10) CALL PLTTQ( 5400)
      IF(VE .NE. 0) VAB = VE
      IF(WE .NE. 0.) NEAB = WE
      IF(KE) 5410, 5420, 5415
 5410 CONTINUE
      IF(NW .GE. -KE) GO TO 5480
      GO TO 5419
 5415 CONTINUE
      IF(NAB .GE. KE) GO TO 5480
 5419 IF(NSAB .GT. 0) GO TO 5480
 5420 CONTINUE
      IF(LE .LT. 0) LW = -LE
      IF(LE .GT. 0) LAB = LE
      IF(ME .LT. 0) MW = -ME
      IF(ME .GT. 0) MAB = ME
      GO TO 2100
```

```
 5460 CALL PLTEE(1, 5480, 210)
      WRITE(JVC, 548)
  548 FORMAT(' TESTAB COMMAND - ERROR LIMIT EXCEED')
      GO TO 3940
C                                                4.5 LIMITI
 5500 IF(VUTQ .GE. 10) CALL PLTTQ( 5500)
      IF(LE .GT. 0) LI = LE
      IF(ME .GT. 0) MI = ME
      IF(ME .LT. 0) MI = NI
      IF(KE .GT. 0) KI = KE
      IF(VE .GT. 0) VI = VE
      IF(VE .LT. 0) VI = NI
      IF(WE .GT. 0.) NI = WE
      IMAP = LI
      GO TO 2100
C                                                4.6 LIMITJ
 5600 IF(VUTQ .GE. 10) CALL PLTTQ( 5600)
      IF(LE .GT. 0) LJ = LE
      IF(ME .GT. 0) MJ = ME
      IF(ME .LT. 0) MJ = NJ
      IF(KE .GT. 0) KJ = KE
      IF(VE .GT. 0) VJ = VE
      IF(VE .LT. 0) VJ = NJ
      IF(WE .GT. 0.) NJ = WE
      JMAP = LJ
      GO TO 2100
C                                                4.7 RUN
C                                                4.8 PERIOD
C                                                4.9 STEP
C                                                4.10 STEPP STEP IN PERIOD
C                                                4.11 TRY ITERATION
 5700 IF(VUTQ .GE. 10) CALL PLTTQ( 5700)
      IDO = IIDO(KOP - 5)
      IF(IDO .LT. LE) GO TO 5780
      IF(ME .LE. 0) GO TO 5730
      IF(IDO .GT. ME) GO TO 5780
 5730 IF(KE .LE. 0) GO TO 5740
      IF(MOD(IDO - LE, KE) .NE. 0) GO TO 5780
 5740 LDO = LE
      GO TO 2100
C                                                FAIL DO
 5780 CONTINUE
      IF(VQ .EQ. 2) RETURN
      IE = MQ
      IF(VE .GT. 0) IE = IIE(MOD(VE, MNIE) + 1)
      GO TO 2100
C                                                4.12 DEPTH
 6200 IF(VUTQ .GE. 10) CALL PLTTQ( 6200)
      I = IPLT
 6220 CONTINUE
      IF(I   .EQ. LE) GO TO 2100
      IF(ME  .EQ.  0) GO TO 6250
      IF(I   .EQ. ME) GO TO 2100
      IF(KE  .EQ.  0) GO TO 6250
      IF(I   .EQ. KE) GO TO 2100
      IF(VE  .EQ.  0) GO TO 6250
      IF(I   .EQ. VE) GO TO 2100
      IF(WE  .EQ. 0.) GO TO 6250
      IF(I   .EQ. IFIX(WE) ) GO TO 2100
      IF(XE  .EQ. 0.) GO TO 6250
      IF(I   .EQ. IFIX(XE) ) GO TO 2100
 6250 CONTINUE
      IF(VQ .EQ. 2) RETURN
      IE = MJ
      IF(YE .EQ. 0.) GO TO 2100
      IE = IIE(MOD(IFIX(YE), MNIE) + 1)
      IF(IE .LE. MNE) GO TO 2100
      CALL PLTEE(5, 6250, 30)
      WRITE(JVC, 320)
      GO TO 3900
C                                                4.13 VARIABLE
 6300 IF(VUTQ .GE. 10) CALL PLTTQ( 6300)
      I = IVAR
      GO TO 6220
C                                                4.14 TIME
 6400 IF(VUTQ .GE. 10) CALL PLTTQ( 6400)
      TDO = TI
 6420 CONTINUE
```

```
      IF(T  .LT. WE) GO TO 5780
      IF(XE .LE. 0.) GO TO 6430
      IF(T  .GT. XE) GO TO 5780
 6430 CONTINUE
C                                               TK DEFER
      TLDU = WE
      GO TO 2100
C                                               4.15 TIMEP TIME IN PERIOD
 6500 IF(VJTQ .GE. 10) CALL PLTTQ( 6500)
      TDU = TPI
      GO TO 6420
C
C
C                                               PROGRAM ABORT
 3900 CONTINUE
      IF(KAB .LT. 0) GO TO 9000
      WRITE(JVC, 390)
 390  FORMAT('- ***** PRINT/PLOT PROGRAM ABORT *****')
 8940 CONTINUE
      IF(VAB - 1) 8960, 9900, 9100
C                                               EXIT
 8960 CONTINUE
      WRITE(JVC, 896) NW, NAB, NSAB
 896  FORMAT(' PRINT/PLOT EXIT,', I5, ' WARNINGS,', I5, ' ERRORS,'
     * , I5, ' SEVERE ERRORS.')
      RETURN
C
C                                               SYSTEM ABORT
 9000 CONTINUE
      WRITE(UVC, 900)
 900  FORMAT('- ***** PRINT/PLOT SYSTEM ABORT *****')
 9100 CONTINUE
      VUTX = 9999
      VUTI = 9999
      CALL PLTET(0, 0, 999)
      CALL PLTEE(1, 9100, 0)
      WRITE(JVC, 361) ILQ, LQ, (LLQ(I), I = 1, MNLQ)
      WRITE(JVC, 361) ILQ, MQ, (MMQ(I), I = 1, MNLQ)
      WRITE(JVC, 361) ILQ, KQ, (KKQ(I), I = 1, MNLQ)
      WRITE(JVC, 362) IE, IES, (IIE(I), I = 1, MNIE)
      IF(NEAB .GT. 0) CALL PLTET(1, NEAB, 0)
 9900 CONTINUE
      WRITE(JVC, 990) NW, NAB, NSAB
 990  FORMAT(' PRINT/PLOT STOP,', I5, ' WARNINGS,', I5, ' ERRORS,'
     * , I5, ' SEVERE ERRORS.')
      STOP
      END



      SUBROUTINE PLTEI
C     SUB PLTEI 10/18/78 MOD 27
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      COMMON / PLTC / LPLT, MPLT, IIDO(9)
      COMMON / PLTCX / UC, UCQ, BCC, UVC, BW, VUTI, VUTQ, VUTX
     * , NW, MW, NAB, MAB, NSAB, KAB
     * , IMAP, LI, MI, KI, VI, NI, JMAP, LJ, MJ, KJ, VJ, NJ
     * , VQ, JOP, KOP, IE, NE, MNE, JE, LE, ME, KE, VE, WE, XE, YE
     * , XDEL, YDEL
      DIMENSION CC(10), CCOP(99), JJOP(99), KKOP(99)
      DIMENSION JJE(302), LLE(302), MME(302), KKE(302), VVE(302)
     * , WWE(302), XXE(302), YYE(302)
      DIMENSION CCEE(8)
      DATA MNOP, LOP, NOP1, MNEO / 99, 2, 0, 300 /
      DATA VJTE, IET / 0, 0 /
      DATA CSTOP / 'STOP' /
      DATA CCEE / 'NOTE', 'WARNING', 'ERROR', 'SEVERE E', 'FATAL ER'
     * , 'SYSTEM E', 'HALT ERR' /
C                                               PLTEI 10/17/78 MOD 27
C                                               INIT COMMON
      MNE = MNEO
      MNET = 0
C                                               LOAD OP TABLE
 101  FORMAT(10I5)
```

```
102   FORMAT(' OP TABLE', T10, 16I5)
103   FORMAT(6(2I2, A8) )
104   FORMAT(' OP TABLE', T10, 6(2I2, A8) )
      IF(VE .GE. 30 .AND. NOP1 .GT. 0) WRITE(UVC, 104)
     * (JJOP(IOP), KKOP(IOP), CCOP(IOP), IOP = 1, NOP1), LOP
      READ(UC, 101) NOP, JE, JOP
      IF(VE .GE. 10) WRITE(UVC, 102) NOP, JE, JOP
      IF(JE .GT. 0) LOP = JE
      IF(JOP .GT. 0) NOP1 = JOP - 1
      IF(NOP .LT. 0) GO TO 1240
      JE = NOP1 + 1
      MOP = NOP1 + NOP
      READ(UC, 103) (JJOP(IOP), KKOP(IOP), CCOP(IOP), IOP = JE, MOP)
      IF(VE .GE. 20) WRITE(UVC, 104)
     * (JJOP(IOP), KKOP(IOP), CCOP(IOP), IOP = JE, MOP)
      GO TO 1280
1240  CONTINUE
      JE = NOP1 + 1
      NOP = 0
      DO 1260 IOP = JE, MNOP
      READ(UC, 103) JJOP(IOP), KKOP(IOP), CCOP(IOP)
      IF(VE .GE. 20) WRITE(UVC, 104) JJOP(IOP), KKOP(IOP), CCOP(IOP)
      IF(JJOP(IOP) .EQ. 0) GO TO 1270
      NOP = NOP + 1
1260  CONTINUE
1270  CONTINUE
      MOP = NOP1 + NOP
1280  CONTINUE
      JOP = 0
      IF(ME .GT. 0) WRITE(ME, 101) JOP, JOP, NOP
      IF(ME .GT. 0) WRITE(ME, 103)
     * (JJOP(IOP), KKOP(IOP), CCOP(IOP), IOP = JE, MOP)
C                                               INIT EXEC TABLE
      DO 1480 I = 1, MNE
      JJE(I) = 1
1480  CONTINUE
      JJE(MNE + 1) = 1
      JJE(MNE + 2) = 2
      RETURN
                                                ENTRY PLTER
      IF(VU .EQ. 1) GO TO 3100
      IF(VUTX .GE. 30) CALL PLTTX(22000)
      READ(UC, 210, END=2800) CC, COP, LE, ME, KE, VE, WE, XE, YE
210   FORMAT(10A8, T1, A8, 2X, 4I5, 3E10.0)
      IET = NE + 1
      IF(SCC) WRITE(UVC, 220) IET, CC
220   FORMAT(' ', I5, T10, 10A8, ' CARD')
2400  CONTINUE
      DO 2460 IOP = LOP, MOP
      JE = IOP
      IF(COP .EQ. CCOP(IOP) ) GO TO 3500
2460  CONTINUE
      JE = 1
      GO TO 3500
2800  CONTINUE
      WRITE(UVC, 280)
280   FORMAT('0 *** END OF FILE ON CONTROL INPUT')
      WRITE(UVC, 281)
281   FORMAT('0', T10, 'STOP', 10X, 'GENERATED STATEMENT')
      COP = CSTOP
      LE = 0
      GO TO 2400
                                                ENTRY PLTEX
      IF(VUTX .GE. 35) CALL PLTTX(23000)
3100  CONTINUE
      IE = IE + 1
      IET = IE
      JE = JJE(IE)
      LE = LLE(IE)
      ME = MME(IE)
      KE = KKE(IE)
      VE = VVE(IE)
      WE = WWE(IE)
      XE = XXE(IE)
      YE = YYE(IE)
3500  CONTINUE
      JOP = JJOP(JE)
      KOP = KKOP(JE)
```

```
      IF(VUTE .GE. 30) WRITE(UVC, 350) VQ, JE, JOP, KOP, IET, CCOP(JE)
     * , LE, ME, KE, VE, WE, XE, YE
  350 FORMAT(' TRACE EX', T10, 5I5, A8, 2X, 4I5, 3G15.7)
      RETURN
                                          ENTRY PLTEQ
      IF(NE .GE. MNE) GO TO 4800
      NE = NE + 1
      IF(NE .GT. MNET) MNET = NE
      JJE(NE) = JE
      LLE(NE) = LE
      MME(NE) = ME
      KKE(NE) = KE
      VVE(NE) = VE
      WWE(NE) = WE
      XXE(NE) = XE
      YYE(NE) = YE
      RETURN
 4800 CONTINUE
      WRITE(UVC, 480) IET, CCOP(JE)
  480 FORMAT('-***FATAL ER # 300(24800) AT', I5, ' ', A8
     * / ' EXECUTE TABLE FULL')
      NSAB = NSAB + 1
      KAB = 1
      RETURN
                                          ENTRY PLTEV(NEO, VEO)
      VVE(NEO) = VEO
      RETURN
                                          ENTRY PLTEE(KEE, IEE, IAB)
      IF(VUTI .GT. 0) CALL PLTTQ(IEE)
      IF(VUTX .GE. 10) CALL PLTTX(29000)
      IF(VUTI .GE. 10) WRITE(UVC, 901) LPLT, MPLT, IIDC
  901 FORMAT('0TIME', T10, 2I5, 7I10, 2F15.5)
      IF(VUTE .GE. 10) WRITE(UVC, 350) VQ, JE, JOP, KOP, IET, CCOP(JE)
     * , LE, ME, KE, VE, WE, XE, YE
      WRITE(UVC, 902) CCEE(KEE), IAB, IEE, IET, CCOP(JE)
  902 FORMAT('0 *** ', A8, ' #', I4, '(', I5, ') AT', I5, ' ', A8)
      GO TO (9100, 9200, 9300, 9400, 9500, 9600, 9700), KEE
      WRITE(UVC, 909)
  909 FORMAT(' DOUBLE FAULT')
      GO TO 9700
 9100 CONTINUE
      RETURN
 9200 CONTINUE
      NW = NW + 1
      IF(NW .GE. MW) KAB = 1
      RETURN
 9300 CONTINUE
      NAB = NAB + 1
      IF(NSAB + NAB .GE. MAB) KAB = 1
      RETURN
 9400 CONTINUE
      NSAB = NSAB + 1
      IF(NSAB + NAB .GE. MAB) KAB = 1
      RETURN
 9500 CONTINUE
      NSAB = NSAB + 1
      KAB = 1
      RETURN
 9600 CONTINUE
      NSAB = NSAB + 1
      KAB = -1
      RETURN
 9700 CONTINUE
      WRITE(UVC, 970) NW, NAB, NSAB
  970 FORMAT('-PRINT/PLOT STOP', I5, ' WARNINGS,', I5, ' ERRORS,'
     * , I5, ' SEVERE ERRORS.')
      STOP
C
                                          ENTRY PLTET(MO, KO, VO)
      IF(VO .NE. 0) VUTE = VO
      IF(KO .LE. 0) RETURN
      M = KO
      IF(M .LE. 0) M = MNET
      DO 9955 I = MO, M
  995 FORMAT(' EXEC TABLE', T15, 4I5, 1X, A3, 1X, 4I5, 3G15.7)
      J = JJE(I)
      WRITE(UVC, 995) J, JJOP(J), KKOP(J), I, CCOP(J)
     * , LLE(I), MME(I), KKE(I), VVE(I), WWE(I), XXE(I), YYE(I)
```

```
 9953 CONTINUE
      RETURN
      END


      SUBROUTINE PLTAI(UVCO, RRRR, MNI, MNJ, MNR, MNV
     * , XXDEL, YYDEL, DDD, MND, IIII)
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      COMMON / PLTCX / UC, UCQ, BCC, UVC, BW, VUTI, VUTQ, VUTX
     * , NW, MW, NAB, MAB, NSAB, KAB
     * , IMAP, LI, MI, KI, VI, NI, JMAP, LJ, MJ, KJ, VJ, NJ
     * , VJ, JOP, KOP, IE, NE, MNE, JE, LE, ME, KE, VE, WE, XE, YE
     * , XDEL, YDEL
C                                             DIMENSION PARAMETERS
      DIMENSION PPPR(MNI, MNJ, MNR), XXDEL(MNJ), YYDEL(MNI),DDD(MNI,MNJ)
      DIMENSION IIII(MNI, MNJ, MNR)
C                                             DIMENSION LOCAL
      DIMENSION HHC(6)
C                                             DATA LOCAL
      DATA HHC / ' ', ' ', '0', '-', '1', '+' /
      DATA MNHC / 6 /
      DATA MVDIV, MIDIV, MDIV / 10, 100, 10 /
      DATA RLDIV, REDIV, RDIV / 0., 1E-20, 1E20 /
      DATA MVLOG, MILOG, MLOG / 10, 100, 10 /
      DATA RLLOG, RELOG, RLOG / 0., 1E-20, 1E20 /
      MOD(I, N) = IABS(I + N - ( (I + N)/N)*N)
C                                             ENTRY PLTAI
C                                             INIT COMMON
      UVC = UVCO
 1    FORMAT(' SUB PLTAI  12/17/73 MOD 27')
      WRITE(UVC, 1)
      NDIV = 0
      NLOG = 0
      RMDIV = AMAX1(RLDIV, REDIV)
      RMLOG = AMAX1(RLLOG, RELOG)
      CALL PLTUI(RRRR, MNI, MNJ, MNR, MNV)
      RETURN
C
C                                             6 PX ARITHMETIC
                                              ENTRY PLTAX
      IF(VUTQ .GE. 30) CALL PLTTQ( 0, 20000)
      GO TO (1000, 2000, 3000, 4000), KOP
      CALL PLTEE(3, 20000, 0)
      RETURN
C                                             6.1 PX ZDIV
 1000 IF(VUTQ .GE. 20) CALL PLTTQ(21000)
      ITRAC = 21000
      IF(LE - 1) 1100, 1200, 3000
 1100 IF(VUTQ .GE. 10) CALL PLTTQ( 21100)
      IF(ME .NE. 0) MVDIV = ME
      IF(KE .NE. 0) MIDIV = KE
      IF(VE .NE. 0) MDIV = VE
      IF(WE .NE. 0.) RLDIV = WE
      IF(XE .NE. 0.) REDIV = XE
      IF(YE .NE. 0.) RDIV = YE
      IF(RLDIV .EQ. -1.) RLDIV = 0.
      IF(REDIV .LT. 0.) REDIV = 0.
      RMDIV = AMAX1(RLDIV, REDIV)
      RETURN
 1200 IF(VUTQ .GE. 10) CALL PLTTQ( 21200)
      IF(ME .NE. 0) MVLOG = ME
      IF(KE .NE. 0) MILOG = KE
      IF(VE .NE. 0) MLOG = VE
      IF(WE .NE. 0.) RLLOG = WE
      IF(XE .NE. 0.) RELOG = XE
      IF(YE .NE. 0.) RLOG = YE
      IF(RLLOG .EQ. -1.) RLLOG = 0.
      IF(RELOG .LT. 0.) RELOG = 0.
      RMLOG = AMAX1(RLLOG, RELOG)
      RETURN
C                                             6.2 PX FUNCTION
 2000 IF(VUTQ .GE. 20) CALL PLTTQ( 22000)
      ME = MOD(ME - 1, MNR) + 1
```

```
      K   = MOD(KE - 1, MNR) + 1
      V   = MOD(VE - 1, MNR) + 1
      X = XE
      Y = YE
      IF(X .EQ. 0.) X = 1.
      IF(Y .EQ. 0.) Y = 1.
      L = IABS(LE) + 1
      GO TO (2100, 2200, 2300, 2400, 2500, 2600, 2700), L
      ITRAC = 22000
      GO TO 3000
C                                                 6.2.0 M = K - V
 2100 IF(VUTJ .GE. 10) CALL PLTTQ(22100)
      IF(VE .NE. 0) GO TO 2170
      IF(KE .NE. 0) GO TO 2140
      DO 2120 I = 1, NI
      DO 2120 J = 1, NJ
 2120 RRRR(I, J, ME) = WE
      RETURN
 2140 CONTINUE
      DO 2150 I = 1, NI
      DO 2150 J = 1, NJ
 2150 RRRR(I, J, ME) = RRRR(I, J, K)
      RETURN
 2170 CONTINUE
      DO 2180 I = 1, NI
      DO 2180 J = 1, NJ
 2180 RRRR(I, J, ME) = RRRR(I, J, K) - RRRR(I, J, V)
      RETURN
C                                                 6.2.1  M = W + XK +- YV
 2200 IF(VUTJ .GE. 10) CALL PLTTQ(22200)
      IF(VE .NE. 0) GO TO 2250
      DO 2240 I = 1, NI
      DO 2240 J = 1, NJ
 2240 RRRR(I, J, ME) = WE + X*RRRR(I, J, K)
      RETURN
 2250 CONTINUE
      IF(LE .LT. 0) Y = -Y
      DO 2280 I = 1, NI
      DO 2280 J = 1, NJ
 2280 RRRR(I, J, ME) = WE + X*RRRR(I, J, K) + Y*RRRR(I, J, V)
      RETURN
C                                                 6.2.2 M = W + XKV
 2300 IF(VUTJ .GE. 10) CALL PLTTQ(22300)
      IF(LE .LT. 0) GO TO 2350
      DO 2340 I = 1, NI
      DO 2340 J = 1, NJ
 2340 RRRR(I, J, ME) = WE + X*RRPR(I, J, K)*RRRR(I, J, V)
      RETURN
C                                                 6.2.-2 M = W + XK/V
 2350 CONTINUE
      IDIV = 0
      DO 2380 I = 1, NI
      DO 2380 J = 1, NJ
      R = RRRR(I, J, V)
      IF(ABS(R) .GT. RMDIV) GO TO 2376
      IF(ABS(R) .GT. RLDIV) GO TO 2373
      RRRR(I, J, ME) = 0.
      GO TO 2380
 2373 CONTINUE
      IDIV = IDIV + 1
      IF(IDIV .LE. MVDIV) WRITE(UVC, 237) K, I, J, R
  237 FORMAT(' *** DIVISION BY VARIABLE', I3, ', POSITION', 2I4
     * , ' VALUE', G15.7)
      RRRR(I, J, ME) = RLOG
      GO TO 2380
 2376 CONTINUE
      RRRR(I, J, ME) = WE + X*RRRR(I, J, K)/R
 2380 CONTINUE
      IF(IDIV .LE. 0) GO TO 2385
      IF(IDIV .LT. MVDIV) WRITE(UVC, 238) K, IDIV
  238 FORMAT('0*** DIVISION BY VARIABLE', I3, ',', I10, ' ZEROES.')
      NDIV = NDIV + 1
      IF(IDIV .GE. MIDIV .OR. NDIV .GE. MDIV) GO TO 8200
 2385 CONTINUE
      RETURN
C                                                 POWER/DEFER
 2400 IF(VUT2 .GE. 10) CALL PLTTQ(22400)
      ITRAC = 22400
```

```
      GO TO 3100
C                                                6.2.4 EXP
 2500 IF(VOTQ .GE. 10) CALL PLTTQ(22500)
      ITRAC = 22500
      IF(YE .EQ. 0.) YE = 10.
      IF(YE .LT. 1.) GO TO 9000
      IF(YE .GT. 1.) YE = ALOG(YE)
      IF(LE .LT. 0) GC TO 2550
      DO 2540 I = 1, NI
      DO 2540 J = 1, NJ
 2540 RRRR(I, J, ME) = WE + X*EXP(YE*RRRR(I, J, K) )
      IF(VE .NE. 0) GC TO 2900
      RETURN
C                                                6.2.-4 LOG
 2550 CONTINUE
      ITRAC = 22550
      X = X/YE
      ILOG = 0
      DO 2580 I = 1, NI
      DO 2580 J = 1, NJ
      R = RRRR(I, J, K)
      IF(R .GT. RMLOG) GO TO 2560
      IF(ABS(R) .GT. RLLOG) GO TC 2555
      RRRR(I, J, ME) = 0.
      GO TO 2580
 2555 CONTINUE
      ILOG = ILOG + 1
      IF(ILOG .LE. MVLOG) WRITE(UVC, 255) K, I, J, R
 255  FORMAT(' *** LOG OF VARIABLE', I3, ', PCSITION', 2I4
     * , ' VALUE', G15.7)
      RRRR(I, J, ME) = RLCG
      GO TO 2580
 2560 CONTINUE
      RRRR(I, J, ME) = WE + X*ALOG(R)
 2580 CONTINUE
      IF(ILOG .LE. 0) GO TO 2585
      IF(ILOG .GT. MVLOG) WRITE(UVC, 256) K, ILOG
 256  FORMAT('0*** LOG OF VARIABLE', I3, ',', I10, ' ZEROS.')
      NLOG = NLOG + 1
      IF(ILOG .GE. MILOG .OR. NLCG .GE. MLOG) GO TC 8300
 2585 CONTINUE
      IF(VE .NE. 0) GC TO 2900
      RETURN
C                                                6.2.5 PHI
 2600 IF(VOTQ .GE. 10) CALL PLTTQ(22600)
      DW = WE
      DX = X
      IF(LE .LT. 0) GC TO 2650
      IF(VE .NE. 0) GC TO 2630
      DO 2620 I = 1, NI
      DO 2620 J = 1, NJ
      DRK = RRRR(I, J, K)
 2620 DDD(I, J) = DW + DX*DRK
      RETURN
 2630 CONTINUE
      DY = Y
      DO 2640 I = 1, NI
      DO 2640 J = 1, NJ
      DRK = RRRR(I, J, K)
      DRV = RRRR(I, J, V)
 2640 DDD(I, J) = DW + DX*DRK + DY*DRV
      RETURN
 2650 CONTINUE
      IF(VE .NE. 0) GC TO 2670
      DO 2660 I = 1, NI
      DO 2660 J = 1, NJ
 2660 RRRR(I, J, ME) = DW + DX*DDD(I, J)
      RETURN
 2670 CONTINUE
      DY = Y
      DO 2680 I = 1, NI
      DO 2680 J = 1, NJ
      DRV = RRRR(I, J, V)
 2680 RRRR(I, J, ME) = DW + DX*DDD(I, J) - DY*DRV
      RETURN
C                                                6.2.6 INTEGER
 2700 CONTINUE
      IF(LE .LT. 0) GC TO 2750
```

```
      DO 2740 I = 1, NI
      DO 2740 J = 1, NJ
 2740 IIII(I, J, ME) = WE + X*RRRR(I, J, K)
      RETURN
C                                               6.2.-6 REAL
 2750 CONTINUE
      DO 2780 I = 1, NI
      DO 2780 J = 1, NJ
 2780 RRRR(I, J, ME) = WE + X*FLOAT(IIII(I, J, K) )
      RETURN
C                                               DIVIDE BY P(V)
 2900 CONTINUE
      ITRAC = 22900
      GO TO 3100
C                                               6.3 PX COMPARE
 3000 IF(VOTQ .GE. 20) CALL PLTTQ(23000)
      ME = MOD(ME - 1, MNR) + 1
      KE = MOD(KE - 1, MNR) + 1
      VE = MOD(VE - 1, MNR) + 1
      L = MOD(LE + 3, 7) + 1
      GO TO (3350, 3250, 3150, 3050, 3100, 3200, 3300), L
      ITRAC = 23000
      GO TO 3000
C                                               6.3.0 EQUAL
 3050 IF(VOTQ .GE. 10) CALL PLTTQ(23050)
      DO 3080 I = 1, NI
      DO 3080 J = 1, NJ
 3080 IF(RRRR(I, J, VE) .EQ. YE) RRRR(I, J, ME) = WE + XE*RRRR(I, J, KE)
      RETURN
C                                               6.3.1 GREATER OR EQUAL
 3100 IF(VOTQ .GE. 10) CALL PLTTQ(23100)
      DO 3140 I = 1, NI
      DO 3140 J = 1, NJ
 3140 IF(RRRR(I, J, VE) .GE. YE) RRRR(I, J, ME) = WE + XE*RRRR(I, J, KE)
      RETURN
C                                               6.3.-1 LESS OR EQUAL
 3150 IF(VOTQ .GE. 10) CALL PLTTQ(23150)
      DO 3180 I = 1, NI
      DO 3180 J = 1, NJ
 3180 IF(RRRR(I, J, VE) .LE. YE) RRRR(I, J, ME) = WE + XE*RRRR(I, J, KE)
      RETURN
C                                               6.3.2 GREATER
 3200 IF(VOTQ .GE. 10) CALL PLTTQ(23200)
      DO 3240 I = 1, NI
      DO 3240 J = 1, NJ
 3240 IF(RRRR(I, J, VE) .GT. YE) RRRR(I, J, ME) = WE + XE*RRRR(I, J, KE)
      RETURN
C                                               6.3.-2 LESS
 3250 IF(VOTQ .GE. 10) CALL PLTTQ(23250)
      DO 3280 I = 1, NI
      DO 3280 J = 1, NJ
 3280 IF(RRRR(I, J, VE) .LT. YE) RRRR(I, J, ME) = WE + XE*RRRR(I, J, KE)
      RETURN
C                                               6.3.3 MAX
 3300 IF(VOTQ .GE. 10) CALL PLTTQ(23300)
      IF(KE .EQ. VE) GO TO 3400
      DO 3340 I = 1, NI
      DO 3340 J = 1, NJ
 3340 RRRR(I, J, ME) = AMAX1(RRRR(I, J, KE), RRRR(I, J, VE) )
      IF(KE .EQ. VE) GO TO 3400
C                                               6.3.-3 MIN
 3350 IF(VOTQ .GE. 10) CALL PLTTQ(23350)
      DO 3380 I = 1, NI
      DO 3380 J = 1, NJ
 3380 RRRR(I, J, ME) = AMIN1(RRRR(I, J, KE), RRRR(I, J, VE) )
 3400 CONTINUE
      IF(XE .GE. YE) RETURN
      DO 3440 I = 1, NI
      DO 3440 J = 1, NJ
 3440 RRRR(I, J, ME) = AMIN1(AMAX1(XE, RRRR(I, J, ME) ), YE)
      RETURN
C                                               6.4 PX STAT
 4000 IF(VOTQ .GE. 20) CALL PLTTQ(24000)
      ME = MOD(ME - 1, MNR) + 1
C     KE = MOD(KE - 1, MNR) + 1
      VE = MOD(VE - 1, MNR) + 1
      HC = HHC(MOD(IFIX(WE), MNHC) + 1)
      NTR = 0
```

```
      TR = 0.
      TRR = 0.
      DO 4180 I = LI, MI, KI
      TRI = 0.
      TRRI = 0.
      DO 4170 J = LJ, MJ, KJ
      IF(RRRR(I, J, VE) .LE. YE) GO TO 4170
      NTR = NTR + 1
      R = RRRR(I, J, ME)
      TRI = TRI + R
      TRRI = TRRI + R*R
 4170 CONTINUE
      TR = TR + TRI
      TRR = TRR + TRRI
 4180 CONTINUE
      IF(NTR .LT. 1) GO TO 4900
      WNTR = NTR
      S = 0.
      IF(NTR .GE. 2) S = (TRR - TR*TR/WNTR)/(WNTR - 1)
      SD = SQRT(S)
      TR = TR/WNTR
      TRR = SQRT(TRR/WNTR)
      WRITE(JVC, 410) HC, NTR, TR, S, SD, TRR
  410 FORMAT(A1, I8, ' SAMPLES, MEAN =', G15.7, ', VARIANCE =', G15.7
     * , ', DEVIATION =', G15.7, ', ROOT MEAN SQUARE =', G15.7)
      WRITE(JVC, 411) HC
  411 FORMAT(A1)
      RETURN
 4900 CALL PLTEE(2, 24900, 440)
      WRITE(JVC, 490)
  490 FORMAT('- NO SAMPLES.')
      RETURN
C                                                ERROR EXITS
 8000 CALL PLTEE(3, ITRAC, 400)
      WRITE(JVC, 800)
  800 FORMAT(' ARITMETIC PROCESSOR - INVALID PARAMETER')
      RETURN
 8100 CALL PLTEE(3, ITRAC, 410)
      WRITE(JVC, 810)
  810 FORMAT(' UNSUPPORTED ARITHMETIC FUNCTION.')
      RETURN
 8200 CALL PLTEE(3, ITRAC, 420)
      WRITE(JVC, 820)
  820 FORMAT(' EXCESSIVE DIVISION BY ZERO.')
      RETURN
 8300 CALL PLTEE(3, ITRAC, 430)
      WRITE(JVC, 830)
  830 FORMAT(' EXCESSIVE LOG OF ZERO.')
      RETURN
                                                 ENTRY PLTAT(MO, KO, VO)
      RETURN
      END
```

```
      SUBROUTINE PLTUI(RRRR, MNI, MNJ, MNR, MNV)
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      COMMON / PLTC / LPLT, MPLT, IIDO(9)
      COMMON / PLTCX / UC, UCQ, BCC, UVC, BW, VUTI, VUTQ, VUTX
     * , NW, MW, NAB, MAB, NSAB, KAB
     * , IMAP, LI, MI, KI, VI, NI, JMAP, LJ, MJ, KJ, VJ, NJ
     * , VQ, JOP, KCP, IE, NE, MNE, JE, LE, ME, KE, VE, WE, XE, YE
     * , XDEL, YDEL
C                                            DIMENSION PARAMETERS
      DIMENSION RRRR(MNI, MNJ, MNR)
C                                            DIMENSION LOCAL
C                                            DIMENSION U
      DIMENSION BBUU(99), IIUU(99), NNUU(99), UUVV(99), VVUU(99)
      DIMENSION BBC(6), BBV(6), CCU(6), UU(6), UUV(6), IIU(10), JJU(10)
      DIMENSION CCV(50), JJDO(9), LLDO(9)
C                                            DIMENSION F
      DIMENSION CCHA(2), CCHI(25)
      DIMENSION HHA(3), HHC(6), HHI(100), HHT(10000)
      DIMENSION LLT(100), MMT(100), NNT(100), KKF(100), NNF(100)
      DIMENSION IIF(13), JJF(13), IIV(13), IIUF(13)
C                                            EQUIVALENCE
C                                            EQUIVALENCE U
      EQUIVALENCE (UU(5),UXR),(UUV(5),UVXR),(BBC(5),BCXR),(BBV(5),BVXR)
      EQUIVALENCE (UU(6),UXW),(UUV(6),UVXW),(BBC(6),BCXW),(BBV(6),BVXW)
      EQUIVALENCE (UU(1), UP), (IIDO(7), IVAR)
     * , (JJDO(1), JPLT), (JJDO(7), JVAR), (JJDO(8), TJ), (JJDO(9), TPJ)
C                                            EQUIVALENCE F
      EQUIVALENCE (HHA(1), CCHA(1) ), (HHI(1), CCHI(1) )
C                                            DATA LOCAL
C                                            DATA U
      DATA CCU / 'PRINTER', 'CONTROL', 'DATA', 'PUNCH','INPUT','OUTPUT'/
      DATA CCV / 50*'UUUUUUUU' /
      DATA LLDO / 9*0 /
      DATA UU / 6, 5, 0, 7, 20, 20 /
      DATA IIU / 1, 2, 3, 4, 5, 6, 3, 2, 3, 1 /
      DATA JJU / 1, 2, 3, 4, 5, 6, 2, 2, 2, 4 /
      DATA MNCV / 50 /
      DATA MLINE / 134 /
C                                            DATA F
      DATA CCHA / '(A1,(T9 ', ', 36A2))' /
      DATA CCHI /' 0 1 2 3',' 4 5 6 7',' 8 91011','12131415','16171819'
     * ,'20212223','24252627','28293031','32333435','36373839'
     * ,'40414243','44454647','48495051','52535455','56575859'
     * ,'60616263','64656667','68697071','72737475','76777879'
     * ,'80818283','84858687','88899091','92939495','96979899' /
      DATA HHC / ' ', ' ', '0', '-', '1', '+' /
      DATA JJF /  1,  2,  1,  2,  1,  3,  4,  5,  4,  5,  2 /
C               I,  I,  A,  G,  G,  G,  E,  E,  F,  F,  G
      DATA IIF / 98, 98, 99, 95, 95, 95, 94, 94, 93, 93, 95 /
      DATA IIV / 11*95 /
      DATA IIUF / 6*1, 3, 3, 4, 4, 1 /
      DATA  HLB, HLCOM / ' ', ',' /
      DATA MNHA, MNHC, MNHI, MNHT, MNT / 8, 6, 100, 10000, 100 /
      DATA NHT / 0 /
      MOD(I, N) = IABS(I + N - ( (I + N)/N)*N)
C                                            ENTRY PLTUI
      WRITE(UVC, 1)
    1 FORMAT(' SUB PLTUI    8/23/79 MCD 27A')
      DO 1020 I = 1, 9
 1020 IIDO(I) = 0
C                                            INIT LOCAL
      DO 1260 I = 1, 99
      IIUU(I) = 0
      VVUU(I) = 0
      BBUU(I) = .FALSE.
      UUVV(I) = UVC
      NNUU(I) = 0
 1260 CONTINUE
      DO 1380 I = 1, 6
       BBC(I) = .TRUE.
       BBV(I) = .FALSE.
       UUV(I) = UVC
 1380 CONTINUE
      DO 1480 I = 1, MNT
       LLT(I) = 0
```

```
      MMT(I) = 0
      NNT(I) = 0
      KKF(I) = -3
      NNF(I) = 0
 1460 CONTINUE
      CALL PLTFI(RRRR, MNI, MNJ, MNR, MNV, CCV, MNCV)
C     CALL PLTPI(RRRR, MNI, MNJ, MNR, MNV
C    * , IIO, HHT, NHT, MNHT, LLT, NNT, KKF, MNT, CCV, MNCV)
      RETURN
C                                          7 UM FILE IMMEDIATE
                                           ENTRY PLTUM
      IF(VUTQ .GE. 30) CALL PLTTQ(32000)
      GO TO (2100, 2200, 2300), KOF
      CALL PLTEE(5, 32000, 0)
      RETURN
C                                          7.1 TEXT
 2100 IF(VUTQ .GE. 10) CALL PLTTQ(32100)
      LE = MOD(LE - 1, MNT) + 1
      IF(ME .LE. 0) ME = 72
      LT = NHT + 1
      L = LT
      IF(KE .GE. 3) L = L + 8
      M = L + (ME - 1)/2
      IF(M .GT. MNHT) GO TO 2190
C                                          READ AND IGNORE/DEFER
      READ(UC, 210) (HHT(I), I = L, M)
  210 FORMAT(36A2)
      IF(BCC) WRITE(UVC, 211) (HHT(I), I = L, M)
  211 FORMAT(T10, 36A2, T90, ' CARD')
      NHT = M
      LLT(LE) = LT
      MMT(LE) = M
      NNT(LE) = ME
      IF(VE .LE. 0) VE = 1
      NNF(LE) = VE
      KKF(LE) = MINO(KE - 2, 0 )
      RETURN
 2190 CALL PLTEE(5, 32190, 420)
      WRITE(UVC, 219)
  219 FORMAT(' TEXT STORAGE FULL')
      RETURN
C                                          7.2 FORMAT
 2200 IF(VUTQ .GE. 10) CALL PLTTQ(32200)
      LE = MOD(LE - 1, MNT) + 1
      ME = MOD(ME - 1, MNT) + 1
      IF(KKF(ME) .LT. 0) GO TO 2290
      IF(LE .EQ. ME) GO TO 2220
      L = LLT(ME)
      M = MMT(ME)
      IF(NHT + M - L + 1 .GT. MNHT ) GO TO 2291
      LT = NHT + 1
      DO 2210 I = L, M
       NHT = NHT + 1
       HHT(NHT) = HHT(I)
 2210 CONTINUE
      LLT(LE) = LT
      MMT(LE) = NHT
      NNT(LE) = NNT(ME)
      KKF(LE) = KKF(ME)
      NNF(LE) = NNF(ME)
C                                          REFORMAT
 2220 CONTINUE
      LT = LLT(LE)
C                                          SET T FORMAT
      KE = MOD(KE, MNHI)
      IF(KE .GT. 0) HHT(LT + 9) = HHI(KE + 1)
C                                          SET REPETITION
      IF(VE .LE. 1) GO TO 2225
      VE = MOD(VE - 1, MNHI) + 1
      IF(VE .GT. 1) HHT(LT + 16) = HHI(VE)
C                                          + 1 - 1
      NNF(LE) = VE
 2225 CONTINUE
C                                          SET W.D
      IF(WE .LE. 0.) GO TO 2230
      I = WE
      J = MOD(IFIX(WE*100. + .5), 100)
      IF(J .GE. 30) J = J/10
```

```
      HHT(LT + 18) = HHI(MOD(I, MNHI) + 1)
      IF(HHT(LT + 21) .EQ. HLB) GO TO 2228
      HHT(LT + 20) = HHI(MOD(J, MNHI) + 1)
      IF(J .GT. 0) GO TO 2228
      IF(HHT(LT + 19) .NE. HLCOM) GO TO 2228
      HHT(LT + 19) = HLB
      HHT(LT + 20) = HLB
      HHT(LT + 21) = HLB
 2228 CONTINUE
      IF(XE .LE. 0.) GO TO 2235
C                                                  SET FIRST WOD
 2230 CONTINUE
      IF(XE .LE. 0.) GO TO 2240
      I = XE
      J = MOD(IFIX(XE*100. + .5), 100)
      IF(J .GE. 30) J = J/10
 2235 CONTINUE
      HHT(LT + 12) = HHI(MOD(I, MNHI) + 1)
      IF(HHT(LT + 15) .EQ. HLB) GO TO 2240
      HHT(LT + 14) = HHI(MOD(J, MNHI) + 1)
      IF(J .GT. 0) GO TO 2240
      IF(HHT(LT + 13) .NE. HLCOM) GO TO 2240
      HHT(LT + 14) = HLB
      HHT(LT + 15) = HLB
 2240 CONTINUE
      RETURN
 2290 CALL PLTEE(3, 32290, 430)
      WRITE(UVC, 229)
  229 FORMAT(' FORMAT NOT AVAILABLE')
      RETURN
 2291 CALL PLTEE(3, 32291, 420)
      WRITE(UVC, 219)
      RETURN
C                                                  7.3 NAME
 2300 IF(VUTQ .GE. 10) CALL PLTTQ(32300)
      LE = MOD(LE - 1, MNCV) + 1
      ME = LE + ME - 1
      IF(ME .GT. MNCV) GO TO 2390
      READ(UC, 230) (CCV(I), I = LE, ME)
  230 FORMAT(9A8)
      IF(BCC) WRITE(UVC, 231) (CCV(I), I = LE, ME)
  231 FORMAT(T10, 9A8, T90, ' CARD')
      RETURN
 2390 CALL PLTEE(5, 32390, 440)
      WRITE(UVC, 239)
  239 FORMAT(' END OF NAME TABLE')
      RETURN
C
C                                                  8 UX FILE EXECUTE
C                                                  ENTRY PLTUX
      IF(VUTQ .GE. 30) CALL PLTTQ(34000)
      GO TO (4100, 4200, 4300, 4400, 4500, 4600, 4700, 4800), KOP
      CALL PLTEE(5, 34000, 0)
      RETURN
C                                                  8.1 UNIT
 4100 IF(VUTQ .GE. 10) CALL PLTTQ(34100)
      LE = MOD(LE, 10) + 1
C                                                  SET DEVICE IU ON DEVICE JU
      IU = IIU(LE)
      JU = JJU(LE)
      J = MOD(ME, 100)
      IF(J .EQ. 0) U = UU(JU)
      IF(LE .LT. 7 .OR. LE .GT. 8) GO TO 4110
      J = UCS
      IF(LE .EQ. 3) UC = U
C                                                  INITIALIZE CONTROL AND CARD FILES
      IF(LE .NE. 7) GO TO 4110
      UU(1) = UVC
      BUC(2) = BW
      UUV(2) = UVC
      IIUU(UC) = 2
      IIUU(UVC) = 1
      V = 0
      IF(BW) V = 2
      VVUU(U) = V
      UUVV(U) = UVC
      BW = .TRUE.
 4110 CONTINUE
```

```
      IF(IIJJ(J) .EQ. 0) IIUU(U) = JC
      IF(IIUU(U) .NE. JJ) GO TO 4190
      V = VVUU(U)
      IF(U .EQ. UU(IU) ) GO TO 4120
      UU(IU) = U
      BBC(IU) = V .GE. 1
      BBV(IU) = V .GE. 2
      UUV(IU) = UUVV(U)
C                                               V = KE
 4120 CONTINUE
      IF(KE .EQ. 0) GO TO 4140
      V = KE
      BBC(IU) = V .GE. 1
      BBV(IU) = V .GE. 2
      IF(IU .EQ. JU) VVUU(U) = V
C                                               UV = VE
 4140 CONTINUE
      VE = MOD(VE, 100)
      IF(VE .EQ. 0) GO TO 4160
      IF(IIUU(VE) .EQ. 0) IIUU(VE) = 1
      IF(IIJJ(VE) .NE. 1) GO TO 4160
      UUV(IU) = VE
      IF(IU .EQ. JU) UUVV(U) = VE
 4160 CONTINUE
      UV = UUV(IU)
C                                               HEADING
      IF(WE .NE. 0.) WRITE(UV, 416) HHC(MOD(IFIX(WE), MNHC) + 1)
     * , CCU(IU), U
 416  FORMAT(A1, T30, A8, ' FILE', I3 / )
C                                               LITTLE OPEN
      IF(KE .EQ. 0.) GO TO 4170
      IF(BBJJ(U) ) GO TO 4170
      BBJJ(U) = .TRUE.
      NNJJ(U) = 0
 4170 CONTINUE
      IF(YE .NE. 0) BW = YE .GT. 0
      IF(IU .NE. 2) GO TO 4180
      IF(U .NE. UC) UCQ = UC
      UC = UU(2)
      BCC = BBC(2)
      UVC = UUV(2)
 4180 CONTINUE
      RETURN
 4190 CALL PLTEE(3, 34190, 500)
      WRITE(UVC, 419)
 419  FORMAT(' UNIT IS WRONG TYPE')
      RETURN
C                                               8.2 OPENR
 4200 IF(VOTO .GE. 10) CALL PLTTQ(34200)
      N = 0
      IF(IIUU(UXR) .EQ. 0) IIUU(UXR) = 4
      IF(IIJJ(UXR) .NE. 5) GO TO 4294
C                                               STANDARD FILE HEADER
      READ(UXR, END=4690) JJDO
      N = 1
      IF(BCXR) WRITE(UVXR, 420) HHC(MOD(VE, MNHC) + 1)
     * , CCU(5), UXR, JJDO
 420  FORMAT(A1, T30, A8, ' FILE', I3
     * / '0', T10, 'CODE, IDENT, TITLE, NI, NJ, NZ, XDEL, YDEL'
     * / T10, 7I10, 2F20.6 / )
      IF(JPLT .NE. -9) GO TO 4290
      IF(ME .GE. 0 .AND. JJDO(2) .NE. ME) GO TO 4291
      IF(JJDO(3) .LE. 0) GO TO 4220
      KE = MOD(KE - 1, MNT) + 1
      L = NHT + 1
      M = L + (JJDO(3) - 1)/2
      IF(M .GT. MNHT) GO TO 4292
C                                               IGNORE TITLE/DEFER
      READ(UXR, END=4690) (HHT(I), I = L, M)
      N = 2
      IF(BCXR) WRITE(UVXR, 421) (HHT(I), I = L, M)
 421  FORMAT('0HEADER', T10, 50A2)
      NHT = M + 1
      LLT(KE) = L
      MMT(KE) = M
      NNT(KE) = JJDO(3)
      KKF(KE) = -2
 4220 CONTINUE
```

```
      IF(IABS(LE) .GE. 2) GO TO 4230
      NI = JJDD(4)
      NJ = JJDD(5)
      MI = NI
      MJ = NJ
      VI = NI
      VJ = NJ
      XDEL = TJ
      YDEL = TPJ
      GO TO 4240
 4230 CONTINUE
      IF(JJDD(4) .NE. NI) GO TO 4293
      IF(JJDD(5) .NE. NJ) GO TO 4293
 4240 CONTINUE
      IF(LE .GT. 0) GO TO 4280
C                                                 PACKED FILE/DEFER
      CALL PLTEE(5, 34240, 410)
      WRITE(UWC, 424)
  424 FORMAT(' UNSUPPORTED OPERATICN')
      RETURN
 4280 CONTINUE
      NNUU(UXR) = N
      BBUU(UXR) = .TRUE.
      RETURN
 4290 CALL PLTEE(5, 34290, 510)
      WRITE(UWC, 429)
  429 FORMAT(' FILE HEADER INVALID')
      IIUU(U) = -1
      RETURN
 4291 CALL PLTEE(5, 34291, 520)
      WRITE(UWC, 428)
  428 FORMAT(' FILE HEADER ID DOES NOT MATCH.')
      IIUU(U) = -1
      RETURN
 4292 CALL PLTEE(5, 34292, 420)
      WRITE(UWC, 219)
      IIUU(U) = -1
      RETURN
 4293 CALL PLTEE(5, 34293, 530)
      WRITE(UWC, 427)
  427 FORMAT(' FILE HEADER DIMENSIONS INCOMPATABLE.')
      IIUU(U) = -1
      RETURN
 4294 CALL PLTEE(3, 34294, 500)
      WRITE(UWC, 419)
      IIUU(U) = -1
      RETURN
C                                                 8.3 OPENW
 4300 IF(VUTD .GE. 10) CALL PLTTQ(34300)
      IF(IIUU(UXW) .EQ. 0) IIUU(UXW) = 5
      IF(IIUU(UXW) .NE. 6) GO TC 4294
      N = 0
      JJDD(1) = -9
      JJDD(2) = ME
      JJDD(3) = 0
      JJDD(4) = NI
      JJDD(5) = NJ
      JJDD(6) = NI*NJ
      JJDD(7) = 0
      TJ = XDEL
      TPJ = YDEL
      IF(KE .EQ. 0) GC TO 4320
      KE = MOD(KE - 1, MNT) + 1
      L = LLT(KE)
      M = MMT(KE)
      JJDD(3) = NNT(KE)
 4320 CONTINUE
      IF(LE .GT. 0) GC TO 4340
C                                                 PACKED FILE/DEFER
      CALL PLTEE(5, 34320, 410)
      WRITE(UWC, 424)
      RETURN
 4340 CONTINUE
      IF(BCXW) WRITE(UVXW, 420) HHC(WCD(VE, MNHC) + 1)
     * , CCU(6), UXW, JJDC
      IF(BW) WRITE(UXW) JJDD
      IF(BW) W = 1
      IF(KE .EQ. 0) GC TO 4350
```

```
      IF(BCXW) WRITE(LVXW, 421) (HHT(I), I = L, M)
      IF(BW) WRITE(UXW) (HHT(I), I = L, M)
      IF(BW) N = 2
 4350 CONTINUE
C                                         PACKED FILE/DEFER
      BBUU(UXW) = .TRUE.
      NNUU(UXW) = N
      RETURN
C                                         8.4 CLOSE
 4400 IF(VUTQ .GE. 10) CALL PLTTQ(34400)
      L = MOD(ME, 100)
      IF(LE .GT. 0) GO TO 4450
      M = L
      IF(L .GT. 0) GO TO 4410
      L = 1
      M = 99
 4410 CONTINUE
      DO 4440 U = L, M
      IU = IIUU(U)
      IF(IU .LE. 0) GO TO 4440
      IF(IU .LT. 3 .AND. LE .GT. -2) GO TO 4435
      IF(NNUU(U) .LE. 0) GO TO 4435
      WRITE(UVC, 442) CCU(IU), U, NNUU(U)
  442 FORMAT('0', T10, A3, ' FILE', I3, ' CLOSED AFTER', I6,' RECORDS.')
      REWIND U
      NNUU(U) = 0
 4435 CONTINUE
      IF(MOD(LE, 2) .EQ. 1) IIUU(U) = 0
 4440 CONTINUE
      RETURN
 4450 CONTINUE
      IU = IIUU(L)
      WRITE(UVC, 445) CCU(IU), L, NNUU(L)
  445 FORMAT('0', T10, A8, ' FILE', I3, ' ENDED AFTER', I6, ' RECORDS.')
      ENDFILE L
      NNUU(L) = 0
      RETURN
C                                         8.5 LIMITR
 4500 IF(VUTQ .GE. 10) CALL PLTTQ(34500)
      IF(LE .NE. 0) LLDO(2) = LE
      IF(ME .NE. 0) LLDO(3) = ME
      IF(KE .NE. 0) LLDO(4) = KE
      IF(VE .NE. 0) LLDO(5) = VE
      IF(WE .NE. 0.) LLDO(6) = WE
      RETURN
C                                         8.6 READ
 4600 IF(VUTQ .GE. 10) CALL PLTTQ(34600)
      IF(IIUU(UXR) .NE. 5) GO TO 4294
      LE = MOD(LE - 1, MNR) + 1
      HC = HHC(MOD(VE, MNHC) + 1)
      GO TO 4620
 4610 CONTINUE
      READ(UXR, END = 4690)
      NNUU(U) = NNUU(U) + 1
 4620 CONTINUE
      READ(UXR, END = 4690) JJDO
      NNUU(U) = NNUU(U) + 1
      IF(BCXR) WRITE(UVXR, 460) HC, CCU(5), UXR, JJDO
  460 FORMAT(A1, A8, 'FILE', I3, 7I10, 2F15.5)
      DO 4640 I = 2, 6
      IF(JJDO(I) .LT. LLDO(I) ) GO TO 4610
 4640 CONTINUE
      IF(JPLT .LT. LPLT) GO TO 4610
      IF(JPLT .GT. MPLT) GO TO 4610
      IF(ME .EQ. 0) GO TO 4650
      IF(JVAR .NE. ME) GO TO 4610
      GO TO 4660
 4650 CONTINUE
      ME = JVAR
      DO 4655 I = 1, 9
 4655 IIUU(I) = JJDO(I)
 4660 CONTINUE
      READ(UXR, END = 4690) ( (RRRR(I, J, LE), J = 1, NJ), I = 1, NI)
      NNUU(UXR) = NNUU(UXR) + 1
      IF(.NOT. BVXR) RETURN
      KUP = 11
      U = UVXR
      VE = WE
```

```
      GO TO 6130
 4690 CALL PLTEE(5, 34690, 540)
      WRITE(JVC, 469) CCU(5), UXR, NNUU(UXR)
 469  FORMAT(' END OF ', A8, ' FILE', I3, ' AFTER', I6, ' RECORDS.')
      BBUU(UXR) = .FALSE.
      NNUU(UXR) = 0
      RETURN
C                                               8.7 WRITE
 4700 IF(VUTQ .GE. 10) CALL PLTTQ(34700)
      IF(IIJJ(UXW) .NE. 6) GO TO 4294
      LE = MOD(LE - 1, MNR) + 1
      HC = HHC(MOD(VE, MNHC) + 1)
      IF(ME .EQ. 0) ME = LE
      I = IVAR
      IVAR = ME
      IF(BCXW) WRITE(UVXW, 400) HC, CCU(6), UXW, IIDC
      IF(BW) WRITE(UXW) IIDC
      IVAR = I
      IF(BW) WRITE(UXW) ( (RRRR(I, J, LE), J = 1, NJ), I = 1, NI)
      IF(BW) NNUU(UXW) = NNUU(UXW) + 2
      IF(.NOT. BVXW) RETURN
      KOP = 11
      U = UVXW
      VE = WE
      GO TO 6130
C                                               8.8 TITLE
 4800 IF(VUTQ .GE. 10) CALL PLTTQ(34800)
      LU = UP
      IF(WE .GT. 0) LU = WE
      MU = LU
C                                               (9.12 TITLEI)
 4810 CONTINUE
      HC = HHC(MOD(VE, MNHC) + 1)
      IF(LE .EQ. 0) GO TO 4880
      LE = MOD(LE - 1, MNT) + 1
      IF(KKF(LE) .LE. -3) GO TO 4890
      IF(ME .EQ. 0) ME = 10
      IF(ME .GT. 0) GO TO 4820
      ME = -ME - NNT(LE)/2
      IF(ME .LT. 2) ME = 2
 4820 CONTINUE
      IF(ME .LT. 1) ME = 1
      IF(ME .GE. MNHI) ME = MNHI - 1
      IF(ME + KE .GT. MLINE) KE = 0
      IF(KE .LE. 0) KE = MLINE + KE - ME
      KE = (KE + 1)/2
      IF(KE .LT. 1) KE = 1
      IF(KE .GE. MNHI) KE = MNHI - 1
      HHA(4) = HHI(ME + 1)
      HHA(5) = HHI(KE + 1)
      L = LLT(LE)
      M = MMT(LE)
      DO 4850 JI = LU, MU
      U = JI
      IF(IIUU(U) .NE. 1) U = UP
 4850 WRITE(U, HHA) HC, (HHT(I), I = L, M)
      RETURN
 4880 CONTINUE
      DO 4885 U = LU, MU
 4885 WRITE(U, 488) HC
 488  FORMAT(A1)
      RETURN
 4890 CALL PLTEE(3, 34890, 550)
      WRITE(JVC, 489)
 489  FORMAT(' TEXT NOT SET')
      RETURN
C                                               9 UF FORMATTED I/O
                                                ENTRY PLTUF
      IF(VUTQ .GE. 30) CALL PLTTQ(36000)
      GO TO (6100, 6200, 6300, 6100, 6100, 6300, 6700, 6300, 6700, 6800
     * , 6100, 7200), KOP
      CALL PLTEE(6, 36000, 0)
      RETURN
C                                               9.1 HEAD
C                                               9.4 PRINT
C                                               9.5 PRINTI
C                                               9.11 VERIFY
C
 6100 IF(VUTQ .GE. 10) CALL PLTTQ(36100)
```

```
      LE = MOD(LE - 1, MNR) + 1
      IF(ME .EQ. 0) ME = LE
      ME = MOD(ME - 1, MNCV) + 1
C                                                   STANDARD FU
 6120 CONTINUE
      J = JP
C                                                   (READ, WRITE VERIFY)
 6130 CONTINUE
      IF(KE .NE. 0) KE = MOD(KE - 1, MNT) + 1
      IF(KE .EQ. 0) KE = IIF(KOP)
      IF(KKF(KE) .GE. -1) GO TO 6160
      IF(KKF(IIF(KOP) ) .GE. -1) GO TO 6150
      CALL PLTEE(3, 36140, 600)
      WRITE(UVC, 614)
 614  FORMAT(' INVALID FORMAT - NO DEFAULT')
      RETURN
 6150 CONTINUE
      CALL PLTEE(2, 36150, 610)
      WRITE(UVC, 615)
 615  FORMAT(' INVALID FORMAT IGNORED')
      KE = IIF(KOP)
 6160 CONTINUE
      CALL PLTFU(U, KOP, JJF(KOP), KKF(KE), HHT(LLT(KE) ) )
      RETURN
C                                                   9.2 HEADI
 6200 IF(VJT3 .GE. 10) CALL PLTTQ(36200)
      IF(LE .LE. 0) GC TO 6120
      IF(MOD(IMAP - LI, LE*KI) .EQ. 0) GO TO 6120
      IF(ME .EQ. 0) GO TO 6280
      LU = UP
      MU = LU + (MJ - LJ)/(VJ*KJ)
      HC = HHC(MOD(ME, MNHC) + 1)
      DO 6250 UI = LU, MU
      J = UI
      IF(IIJJ(J) .NE. 1) U = UP
 6250 WRITE(U, 620) HC
 620  FORMAT(A1)
 6280 CONTINUE
      RETURN
C                                                   9.3 HEADIJ
C                                                   9.6 PRINTIJ
 6300 IF(VJT3 .GE. 10) CALL PLTTQ(36300)
      GO TO 6120
C                                                   9.7 CARD
C                                                   9.9 PUNCH
 6700 IF(VJT3 .GE. 10) CALL PLTTQ(36700)
      JF = 2
      LE = MOD(LE - 1, MNR) + 1
C                                                  (9.8 CARDIJ)
C                                                  (9.10 PUNCHIJ)
 6720 CONTINUE
      J = JJ(IIUF(KOP) )
      IF(ME .NE. 0) ME = MOD(ME - 1, MNT) + 1
      IF(ME .EQ. 0) ME = IIF(KOP)
      IF(KKF(ME) .GE. -1) GO TO 6740
      I = 3
      IF(J .EQ. UC) I = 5
      CALL PLTEE(I, 36730, 650)
      WRITE(UVC, 673)
 673  FORMAT(' INVALID CARD FORMAT')
      RETURN
 6740 CONTINUE
      IF(.NOT. BBV(IIUF(KOP) ) ) GO TO 6780
      UV = UUV(IIUF(KCP) )
      IF(KE .NE. 0) KE = MOD(KE - 1, MNT) + 1
      IF(KE .EQ. 0) KE = IIV(KOP)
      IF(KKF(KE) .GE.-1) GO TO 6760
      IF(KKF(IIV(KOP) ) .GE. -1) GO TO 6750
      CALL PLTEE(2, 36740, 620)
      WRITE(UVC, 674)
 674  FORMAT(' INVALID VERIFY FORMAT - NO DEFAULT - VERIFY SUPPRESSED.')
      GO TO 6780
 6750 CONTINUE
      CALL PLTEE(2, 36750, 630)
      WRITE(UVC, 675)
 675  FORMAT(' INVALID VERIFY FORMAT IGNORED')
      KE = IIV(KOP)
 6760 CONTINUE
```

```
      IF(KE .NE. ME) GO TO 6770
      CALL PLTEE(2, 36760, 640)
      WRITE(UVC, 676)
 676  FORMAT(' SAME FORMAT - VERIFY SUPPRESSED.')
      GO TO 6730
 6770 CONTINUE
      CALL PLTFV(UV, JF, KKF(KE), HHT(LLT(KE) ) )
 6730 CONTINUE
      CALL PLTFU(U, KOP, JJF(KOP), KKF(ME), HHT(LLT(ME) ) )
      RETURN
C                                               9.8 CARDIJ
C                                               9.10 PUNCHIJ
 6800 IF(VUTQ .GE. 10) CALL PLTTQ(36800)
      JF = 3
      GO TO 6720
C                                               9.12 TITLEI
 7200 IF(VUTQ .GE. 10) CALL PLTTQ(37200)
      IF(MOD(IMAP - LI, VI*KI) .NE. 0) RETURN
      LU = UP
      MU = LU + (MJ - LJ)/(VJ*KJ)
      GO TO 4810
                                                ENTRY PLTUT(MO, KO, VO)
      DO 9180 I = 1, KO
      WRITE(UVC, 910) LLT(I), MMT(I), NNT(I), KKF(I), NNF(I)
 910  FORMAT(' TEXT TABLE', T15, 19I5)
 9180 CONTINUE
      RETURN
      END



      SUBROUTINE PLTFI(RRRR, MNI, MNJ, MNR, MNV, CCV, MNCV)
      IMPLICIT LOGICAL*4 (B)
      IMPLICIT REAL*8 (C-D)
      IMPLICIT INTEGER (F-G, U-V)
      IMPLICIT INTEGER*2 (H)
      COMMON / PLTCX / UC, UCQ, BCC, UVC, BW, VUTI, VUTQ, VUTX
     * , NW, MW, NAB, MAB, NSAB, KAB
     * , IMAP, LI, MI, KI, VI, NI, JMAP, LJ, MJ, KJ, VJ, NJ
     * , VQ, JUP, KOP, IE, NE, MNE, JE, LE, ME, KE, VE, WE, XE, YE
     * , XDEL, YDEL
C                                               DIMENSION PARAMETERS
      DIMENSION RRRR(MNI, MNJ, MNR), CCV(MNCV), HHF(100), HHV(100)
C                                               DIMENSION LOCAL
      DIMENSION HHC(6), HHHF(8, 5), CCHF(10), KKRJ(80), KKVJ(80),RRJ(80)
C                                               EQUIVALENCE
      EQUIVALENCE (HHHF(1, 1), CCHF(1) ), (KKRJ(1), KRJ), (KKVJ(1), KVJ)
C                                               DATA LOCAL
      DATA CCHF /'(A1,      ','    AB  , ','(A1,3HRO','W, I4  , '
     *           ,'(A1,      ','2   I4  , ','(',' ','(','2   I3 ', ' /
      DATA HHC / ' ', ' ', '0', '-', '1', '+' /
      DATA MNHC, MNHF, MKF, MNKRJ, NKRJ / 6, 8, 5, 80, 0 /
      DATA BV / .FALSE. /
      MOD(I, N) = IABS(I + N - ( (I + N)/N)*N)
C                                               ENTRY PLTFI
      WRITE(UVC, 1)
    1 FORMAT(' SUB PLTFI    1/19/79 MOD 27')
      DO 1020 I = 1, MNKRJ
      KKRJ(I) = 1
 1020 KKVJ(I) = 1
      RETURN
C
C
C                                               9 FV VERIFY FORMATTED
      ENTRY PLTFV(UV, JF, KF, HHV)
      IF(VUTQ .GE. 20) CALL PLTTQ(42000)
      IF(KF + 1) 2090, 2080, 2020
 2020 CONTINUE
      IF(JF .EQ. KF) GO TO 2080
      KF = JF
      DO 2060 I = 1 , MNHF
 2060  HHV(I) = HHHF(I, KF)
 2080 CONTINUE
      BV = .TRUE.
      RETURN
 2090 CONTINUE
      CALL PLTEE(6, 42090, 0)
```

```
      WRITE(JVC, 209)
 209  FORMAT(' INVALID FORMAT')
      RETURN
C
C
C                                                  9 FU FORMATTED I/O
      ENTRY PLTFU(UF, KOPF, JF, KF, HHF)
      IF(VUTQ .GE. 30) CALL PLTTQ(43000)
      IF(KF + 1) 2090, 3080, 3020
 3020 CONTINUE
      IF(JF .EQ. KF) GO TO 3080
      KF = JF
      DO 3060 I = 1 , MNHF
 3060  HHF(I) = HHHF(I, KF)
 3080 CONTINUE
      V = VE
      IF(V .GE. 10) V = MOD(V, 10)
      HC = HHC(MOD(V, MNHC) + 1)
      GO TO (3100, 3200, 3300, 3400, 3500, 3600, 3700, 3800, 3900, 4000
     * , 4100), KOPF
      CALL PLTEE(6, 43000, 0)
      RETURN
C                                                  9.1 HEAD
 3100 IF(VUTQ .GE. 20) CALL PLTTQ(43100)
      WRITE(JF, HHF) HC, CCV(ME), (J, J = LJ, MJ, KJ)
      RETURN
C                                                  9.2 HEADI
 3200 IF(VUTQ .GE. 20) CALL PLTTQ(43200)
      J = JF
      L = LJ
      V = (VJ - 1)*KJ
 3220 CONTINUE
      M = MIN0(L + V, MJ)
      WRITE(J, HHF) HC, IMAP, (J, J = L, M, KJ)
      J = J + 1
      L = M + 1
      IF(L .LE. MJ) GO TO 3220
      RETURN
C                                                  9.3 HEADIJ
 3300 IF(VUTQ .GE. 20) CALL PLTTQ(43300)
      L = LE
      IF(L .EQ. 0) L = 2
      L = MOD(L - 1, MNKRJ) + 1
      M = NKRJ
      IF(ME .GT. 0) M = L + ME - 1
      IF(M .GT. MNKRJ) M = MNKRJ
      WRITE(JF, HHF) HC, CCV(KVJ), (CCV(KKVJ(IK) ), IK = L, M)
      RETURN
C                                                  9.4 PRINT
 3400 IF(VUTQ .GE. 20) CALL PLTTQ(43400)
      DO 3460 I = LI, MI, KI
      IF(VE .LT. 30) GO TO 3450
      DO 3440 V = 30, VE, 30
 3440 WRITE(UF, 340)
 340  FORMAT('-')
 3450 CONTINUE
 3460  WRITE(UF, HHF) HC, I, (RRRR(I, J, LE), J = LJ, MJ, KJ)
      RETURN
C                                                  9.5 PRINTI
 3500 IF(VUTQ .GE. 20) CALL PLTTQ(43500)
      U = UF
      L = LJ
      V = (VJ - 1)*KJ
 3520 CONTINUE
      M = MIN0(L + V, MJ)
      WRITE(J, HHF) HC, CCV(ME), (RRRR(IMAP, J, LE), J = L, M, KJ)
      U = U + 1
      L = M + 1
      IF(L .LE. MJ) GO TO 3520
      RETURN
C                                                  9.6 PRINTIJ
 3600 IF(VUTQ .GE. 20) CALL PLTTQ(43600)
      L = LE
      IF(L .EQ. 0) L = 2
      L = MOD(L - 1, MNKRJ) + 1
      M = NKRJ
      IF(ME .GT. 0) M = L + ME - 1
      IF(M .GT. MNKRJ) M = MNKRJ
```

```
      DO 3680 I = LI, MI, KI
C                                                     HEAD ROW / DEFER
      DO 3680 J = LJ, MJ, KJ
      IF(RRRR(I, J, KRJ) .LE. WE) GO TO 3679
      WRITE(UF, HHF) HC, I, J, (RRRR(I, J, KKRJ(IK) ), IK = L, M)
 3679 CONTINUE
 3680 CONTINUE
      RETURN
C                                                     9.7 CARD
 3700 IF(VUTQ .GE. 20) CALL PLTTQ(43700)
      DO 3730 I = LI, MI, KI
      READ(UF, HHF, END = 3790) (RRRR(I, J, LE), J = LJ, MJ, KJ)
      IF(BV) WRITE(UV, HHV) HC, I, (RRRR(I, J, LE), J = LJ, MJ, KJ)
 3730 CONTINUE
      BV = .FALSE.
      RETURN
 3790 CALL PLTEE(5, 43790, 541)
      WRITE(UVC, 379) UF
 379  FORMAT(' END OF CARD FILE', I3)
      RETURN
C                                                     9.8 CARDIJ
 3800 CONTINUE
      L = LE/100
      LE = LE - L*100
      IF(L .EQ. 0) L = 2
      L = MOD(L - 1, MNKRJ) + 1
      M = NKRJ
      IF(LE .GT. 0) M = L + LE - 1
      IF(M .GT. MNKRJ) M = MNKRJ
 3810 CONTINUE
      READ(UF, HHF, END=3790) I, J, (RRJ(IK), IK = L, M)
      IF(BV) WRITE(UV, HHV) HC, I, J, (RRJ(IK), IK = L, M)
      IF(I .EQ. 0) GO TO 3880
      IF(I.GE.1 .AND. I.LE.NI .AND. J.GE.1 .AND. J.LE.NJ) GO TO 3830
      CALL PLTEE(3, 43810, 660)
      WRITE(UVC, 381) I, J
 381  FORMAT(' CARDIJ, ELEMENT', 2I5, ' OUT OF RANGE.')
      GO TO 3810
 3830 CONTINUE
      IF(WE .NE. 0.) RRRR(I, J, KRJ) = WE
      DO 3858 IK = L, M
 3858 RRRR(I, J, KKRJ(IK) ) = RRJ(IK)
      GO TO 3810
 3880 CONTINUE
      BV = .FALSE.
      RETURN
C                                                     9.9 PUNCH
 3900 CONTINUE
      DO 3980 I = LI, MI, KI
      IF(BV) WRITE(UV, HHV) HC, I, (RRRR(I, J, LE), J = LJ, MJ, KJ)
 3980 IF(BW) WRITE(UF, HHF)        (RRRR(I, J, LE), J = LJ, MJ, KJ)
      BV = .FALSE.
      RETURN
C                                                     9.10 PUNCHIJ
 4000 CONTINUE
      L = LE/100
      LE = LE - L*100
      IF(L .EQ. 0) L = 2
      L = MOD(L - 1, MNKRJ) + 1
      M = NKRJ
      IF(LE .GT. 0) M = L + LE - 1
      IF(M .GT. MNKRJ) M = MNKRJ
      DO 4080 I = LI, MI, KI
      DO 4080 J = LJ, MJ, KJ
      IF(RRRR(I, J, KRJ) .LE. WE) GO TO 4079
      IF(BV) WRITE(UV, HHV) HC, I, J, (RRRR(I, J, KKRJ(IK) ), IK = L, M)
      IF(BW) WRITE(UF, HHF)      I, J, (RRRR(I, J, KKRJ(IK) ), IK = L, M)
 4079 CONTINUE
 4080 CONTINUE
      BV = .FALSE.
      RETURN
C                                                     9.11 VERIFY
 4100 IF(VUTQ .GE. 20) CALL PLTTQ(44100)
      DO 4180 I = 1 , NI
 4180 WRITE(UF, HHF) HC, I, (RRRR(I, J, LE), J = 1, NJ)
      RETURN
C
C
```

```
C                                                    10 FX FORMATTED EXECUTE
      ENTRY PLTFX
      IF(VJTQ .GE. 30) CALL PLTTQ(46000)
      GO TO (6100, 6200, 6300), KOP
      CALL PLTEE(6, 46000, 0)
      RETURN
C                                                    10.1 IPRINT
 6100 IF(VJTQ .GE. 10) CALL PLTTQ(46100)
      IMAP = MOD(LE - 1, NI) + 1
      RETURN
C                                                    10.2 SETIJ
 6200 IF(VJTQ .GE. 10) CALL PLTTQ(46200)
      IF(LE .GT. 0) NKRJ = MOD(LE - 1, MNKRJ) + 1
      IF(LE .LT. 0) NKRJ = 0
      IF(ME .EQ. 0) RETURN
      ME = MOD(ME - 1, MNKRJ) + 1
      KKRJ(ME) = MOD(KE - 1, MNR) + 1
      KKVJ(ME) = MOD(VE - 1, MNCV) + 1
      RETURN
C                                                    10.3 VARIJ
 6300 IF(VJTQ .GE. 10) CALL PLTTQ(46300)
      I = LE/100
      J = LE
      IF(I .NE. 0) J = MOD(IABS(J), 100)
      IF(I .EQ. 0) I = J
      IF(NKRJ .GE. MNKRJ) GO TO 6380
      NKRJ = NKRJ + 1
      KKRJ(NKRJ) = MOD(I - 1, MNR) + 1
      KKVJ(NKRJ) = MOD(J - 1, MNCV) + 1
      IF(ME .EQ. 0) GO TO 6380
      I = ME/100
      J = ME
      IF(I .NE. 0) J = MOD(IABS(J), 100)
      IF(I .EQ. 0) I = J
      IF(NKRJ .GE. MNKRJ) GO TO 6380
      NKRJ = NKRJ + 1
      KKRJ(NKRJ) = MOD(I - 1, MNR) + 1
      KKVJ(NKRJ) = MOD(J - 1, MNCV) + 1
      IF(KE .EQ. 0) GO TO 6380
      I = KE/100
      J = KE
      IF(I .NE. 0) J = MOD(IABS(J), 100)
      IF(I .EQ. 0) I = J
      IF(NKRJ .GE. MNKRJ) GO TO 6380
      NKRJ = NKRJ + 1
      KKRJ(NKRJ) = MOD(I - 1, MNR) + 1
      KKVJ(NKRJ) = MOD(J - 1, MNCV) + 1
      IF(VE .EQ. 0) GO TO 6380
      I = VE/100
      J = VE
      IF(I .NE. 0) J = MOD(IABS(J), 100)
      IF(I .EQ. 0) I = J
      IF(NKRJ .GE. MNKRJ) GO TO 6380
      NKRJ = NKRJ + 1
      KKRJ(NKRJ) = MOD(I - 1, MNR) + 1
      KKVJ(NKRJ) = MOD(J - 1, MNCV) + 1
C                                                    SPLIT *E, XE, YE / DEFER
 6380 CONTINUE
      RETURN
C
C
      ENTRY PLTFT
      RETURN
      END
```

```
      SUBROUTINE PLTTI(UVC)
      INTEGER UVC
      DIMENSION II(10)
      DATA II / 10*0 /
      DATA MNI, NI / 10, 0 /
      RETURN
                                            ENTRY PLTTM(I)
      NI = 1
      II(NI) = I
      RETURN
                                            ENTRY PLTTQ(I)
      IF(NI .LT. MNI) GO TO 3800
      DO 3780 J = 2, MNI
 3780 II(J - 1) = II(J)
      NI = NI - 1
 3800 CONTINUE
      NI = NI + 1
      II(NI) = I
      RETURN
                                            ENTRY PLTTX(I)
      WRITE(UVC, 400) I, (II(J), J = 1, NI)
 400  FORMAT(' TRACE PROG', T15, I5, (T20, 10I10) )
      NI = 0
      RETURN
      END
```

APPENDIX C

INPUT DATA FOR MODEL

```
   1            1   3     2    5    8
A810610.2022   ALLOC=0.015CFS/AC;PERM=600GPD;RIVER=STRT+1.2;BOT=300
DPUR (S.A.HOSSAIN)
WATERTAB
LEAKAGE
CONVERT
EVAPOTRANSPIRATION
RECHARGE
CHECK
NO PUNCH
NO NUMERIC
NO HEAD
NO CONTO
    1
        40         30         34       9999          5       .005         1.         50
     1E-10         0.         0.         1.         1.         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
STRT                   1          1          1
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01        241       0.01       0.01       0.01        243
       242        240       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01       0.01        239        240        242        242        242        242
       241        238        234       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01       0.01        212       0.01       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
      0.01        234        236        238        240        240        240        240
       235        230        226       0.01       0.01       0.01       0.01       0.01
      0.01       0.01       0.01       0.01        211        210        208       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01       0.01       0.01
       228        230        232        235        238        238        237        238
       234        230        226       0.01       0.01       0.01       0.01       0.01
      0.01        211        211        211        210        208        206       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01       0.01        201        210
       220        229        231        231        230        230        230        230
       230        229        228        218        230        228        220        218
      0.01        210        210        208        206        205        204       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01       0.01        201        208        210
       219        222        224        229        227        229        229        229
       228        227        220        209        220        220        219        217
       210        209        208        207        206        205        204       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01        199        200        208        210
       219        210        220        220        220        220        220        220
       220        220        218        218        205        210        210        209
       208        203        200        199        198        197        196       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01        199        200        207        210
       215        208        219        217        218        218        218        219
       217        218        217        210        210        200        195        192
       206        200        198        198        198        197        196       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01        198        200        206        208
       210        205        210        210        210        210        214        211
       210        210        210        208        208        207        200        200
       189        199        197        194        190        188        186       0.01
      0.01       0.01
      0.01       0.01       0.01       0.01        192        198        200        205
       208        202        209        207        206        208        208        209
```

209	209	205	201	200	200	197	185
189	190	190	190	185	185	184	183
182	0.01						
0.01	0.01	0.01	191	192	198	200	203
204	200	203	203	203	203	205	205
204	203	200	200	199	195	190	190
190	178	189	185	182	180	180	179
178	0.01						
0.01	0.01	0.01	191	192	196	198	200
196	200	200	200	200	200	200	200
200	200	198	194	191	190	188	184
180	176	180	180	179	179	178	177
176	0.01						
0.01	0.01	0.01	0.01	0.01	196	197	198
190	198	199	197	198	198	198	198
198	198	195	194	190	189	188	184
178	178	174	178	178	177	176	175
0.01	0.01						
0.01	0.01	0.01	0.01	189	190	194	193
188	192	190	190	190	193	192	196
196	194	194	190	190	188	187	180
177	176	176	172	176	176	176	175
0.01	0.01						
0.01	0.01	0.01	187	188	188	191	190
186	190	189	185	188	190	190	192
192	192	192	190	188	187	186	180
178	174	174	173	169	174	174	0.01
0.01	0.01						
0.01	0.01	0.01	0.01	186	187	188	189
188	176	187	180	184	188	188	190
190	190	190	188	186	185	180	177
176	174	173	173	166	170	170	170
0.01	0.01						
0.01	0.01	0.01	0.01	0.01	180	185	185
186	170	180	176	180	180	183	187
188	189	188	188	186	180	178	175
175	173	170	170	163	170	169	168
0.01	0.01						
0.01	0.01	0.01	0.01	173	178	180	180
180	180	165	175	175	175	180	180
180	180	186	185	180	179	178	174
170	170	168	168	160	168	0.01	0.01
0.01	0.01						
0.01	0.01	0.01	172	172	172	178	178
178	171	163	163	170	170	177	178
178	178	180	180	178	178	177	170
170	168	168	167	159	161	161	159
0.01	0.01						
0.01	0.01	0.01	170	170	170	174	174
174	170	164	159	164	166	170	175
175	175	178	178	178	173	170	168
168	165	160	157	162	158	158	157
0.01	0.01						
0.01	0.01	157	160	164	167	171	170
170	170	163	158	160	163	167	170
170	172	170	175	170	171	168	165
160	160	155	159	158	157	156	0.01
0.01	0.01						
0.01	156	157	157	160	163	168	170
168	164	155	159	158	160	160	167
168	170	169	170	169	0.01	164	158
158	158	149	158	156	156	154	0.01
0.01	0.01						
0.01	155	156	156	158	160	163	168
167	160	150	157	156	160	160	160
160	159	167	169	0.01	0.01	0.01	151
156	156	142	150	154	0.01	0.01	0.01
0.01	0.01						
0.01	0.01	0.01	151	155	158	159	160
160	158	149	154	154	157	155	156
155	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01						
0.01	0.01	0.01	0.01	151	155	156	158
158	155	148	150	150	150	150	150
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01						

0.01	0.01	0.01	0.01	0.01	0.01	148	149
151	150	145	149	149	149	148	148
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01						
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01						
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
0.01	0.01						

S 0.20 0 1

PERM 928E -6 1 1 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	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	1	0	0	0	1	1	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	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0						
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0						
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0						
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	0	1	1	1	1	1	1	0	0	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0						
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0						
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	0	0	0						
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	0	0	0	0	0						
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	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	1	1	1	1	1	1	1	1	1	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	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						

BOTTOM 1 1 1 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	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	-49	0	0	0	-47	-48	-50	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	-51	-47	-48	-48	-48	-47	-49	-52	-56	0
0	0	0	0	0	0	0	0	0	-75	0	0	0	0						
0	0	0	0	0	0	0	0	0	-56	-50	-50	-48	-48	-48	-47	-55	-49	-64	0
0	0	0	0	0	0	0	0	-80	-75	-82	0	0	0						
0	0	0	0	0	0	0	0	-62	-58	-56	-51	-50	-50	-50	-50	-56	-58	-64	0
0	0	0	0	0	-79	-79	-79	-80	-80	-84	0	0	0						
0	0	0	0	0	0	-89	-70	-69	-60	-60	-55	-57	-58	-57	-59	-58	-58	-60	-70
-60	-63	-71	-72	0	-75	-75	-80	-82	-85	-86	0	0	0						
0	0	0	0	0	-86	-80	-78	-70	-68	-66	-60	-60	-60	-60	-60	-60	-63	-68	-78
-67	-70	-71	-74	-77	-80	-80	-82	-83	-83	-86	0	0	0						
0	0	0	0	-91	-87	-80	-76	-70	-77	-68	-69	-67	-67	-68	-69	-68	-68	-70	-71
-84	-77	-78	-80	-80	-85	-88	-91	-92	-93	-94	0	0	0						
0	0	0	0	-91	-88	-80	-77	-75	-80	-70	-70	-70	-70	-70	-70	-70	-70	-72	-78
-78	-88	-90	-91	-81	-87	-90	-90	-92	-93	-94	0	0	0						
0	0	0	0	-92	-88	-82	-80	-78	-82	-77	-76	-77	-76	-76	-79	-77	-75	-79	-80
-80	-83	-87	-89	-94	-90	-91	-93	-96	-102	-104	0	0	0						
0	0	0	0	-92	-90	-86	-83	-80	-90	-80	-80	-80	-80	-80	-80	-80	-80	-83	-87
-88	-87	-89	-90	-105	-94	-93	-94	-100	-100	-106	-107	-112	0						
0	0	0	-94	-92	-90	-88	-85	-88	-91	-84	-86	-85	-84	-88	-83	-84	-84	-88	-88
-89	-95	-90	-92	-100	-110	-100	-100	-108	-106	-105	-111	-112	0						
0	0	0	-94	-93	-92	-90	-86	-92	-87	-88	-89	-88	-88	-88	-87	-87	-86	-90	-90
-99	-92	-92	-100	-105	-111	-105	-105	-107	-107	-110	-113	-114	0						
0	0	0	0	0	-94	-91	-90	-96	-90	-90	-90	-90	-90	-90	-90	-90	-90	-92	-94
-96	-97	-100	-106	-110	-110	-114	-110	-110	-113	-114	-115	0	0						
0	0	0	0	-101	-96	-94	-95	-100	-95	-96	-96	-97	-95	-96	-92	-92	-94	-94	-98
-98	-99	-100	-108	-110	-112	-111	-115	-112	-114	-112	-115	0	0						
0	0	0	-103	-101	-100	-98	-99	-105	-98	-100	-105	-100	-97	-98	-96	-96	-96	-96	-98
-100	-100	-102	-111	-110	-114	-114	-114	-117	-114	-114	0	0	0						
0	0	0	0	-104	-101	-100	-101	-100	-108	-102	-107	-106	-100	-100	-98	-98	-98	-98	-100
-102	-103	-106	-113	-110	-114	-114	-115	-119	-116	-116	-116	0	0						
0	0	0	0	0	-108	-103	-103	-102	-111	-107	-107	-110	-105	-104	-100	-100	-100	-100	-100
-102	-106	-108	-114	-110	-114	-116	-116	-125	-116	-117	-122	0	0						
0	0	0	0	-115	-110	-108	-108	-107	-107	-120	-110	-110	-110	-108	-107	-107	-107	-102	-103
-108	-108	-109	-110	-116	-116	-113	-119	-129	-120	0	0	0	0						
0	0	0	-116	-116	-116	-110	-110	-110	-113	-124	-124	-115	-114	-110	-110	-110	-110	-108	-108
-109	-110	-110	-116	-117	-120	-120	-120	-130	-128	-128	-131	0	0						
0	0	0	-118	-118	-113	-114	-114	-115	-116	-120	-125	-120	-124	-116	-112	-112	-112	-110	-110
-110	-115	-118	-120	-120	-120	-128	-131	-129	-130	-130	-133	0	0						
0	0	-133	-127	-120	-120	-116	-116	-116	-116	-122	-126	-124	-122	-120	-117	-116	-116	-116	-113
-118	-117	-121	-123	-127	-126	-132	-129	-130	-131	-134	0	0	0						
0	-134	-132	-130	-127	-125	-120	-117	-117	-120	-130	-125	-126	-125	-124	-120	-120	-118	-118	-118
-121	0	-126	-130	-130	-130	-135	-130	-131	-132	-136	0	0	0						
0	-135	-134	-134	-130	-128	-125	-120	-120	-124	-131	-127	-128	-127	-126	-125	-124	-131	-123	-121
0	0	0	-139	-134	-134	-137	-135	-136	0	0	0	0	0						
0	0	0	-139	-132	-128	-130	-126	-127	-126	-135	-130	-130	-130	-130	-130	-135	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0						
0	0	0	0	-139	-132	-132	-130	-130	-130	-141	-134	-134	-134	-132	-132	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0						
0	0	0	0	0	0	-142	-140	-139	-140	-145	-140	-141	-141	-140	-142	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	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						

SY 0.20 0 1 0
TOP 1 1 1

9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00

9999.00	9999.00	9999.00	238.00	9999.00	9999.00	9999.00	240.00
239.00	236.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	234.00	236.00	238.00	238.00	238.00	238.00
233.00	228.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	208.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	228.00	230.00	233.00	236.00	236.00	235.00	236.00
232.00	228.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	208.00	206.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	208.00
218.00	227.00	229.00	229.00	228.00	228.00	228.00	228.00
228.00	227.00	226.00	216.00	9999.00	226.00	218.00	9999.00
9999.00	208.00	208.00	206.00	204.00	203.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	206.00	208.00
217.00	220.00	222.00	227.00	225.00	227.00	227.00	227.00
226.00	225.00	218.00	207.00	218.00	218.00	217.00	215.00
208.00	207.00	206.00	205.00	204.00	203.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	198.00	206.00	208.00
217.00	208.00	218.00	218.00	218.00	218.00	218.00	218.00
218.00	218.00	216.00	216.00	203.00	208.00	208.00	207.00
206.00	201.00	198.00	197.00	196.00	195.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	198.00	205.00	208.00
213.00	206.00	217.00	215.00	216.00	216.00	216.00	217.00
215.00	216.00	215.00	208.00	208.00	198.00	193.00	190.00
204.00	198.00	196.00	196.00	196.00	195.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	198.00	204.00	206.00
208.00	203.00	208.00	208.00	208.00	208.00	212.00	209.00
208.00	208.00	208.00	206.00	206.00	205.00	198.00	198.00
187.00	197.00	195.00	192.00	188.00	186.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	196.00	198.00	203.00
206.00	200.00	207.00	205.00	204.00	206.00	206.00	207.00
207.00	207.00	203.00	199.00	198.00	198.00	195.00	183.00
187.00	188.00	188.00	188.00	183.00	183.00	182.00	181.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	190.00	196.00	198.00	201.00
202.00	198.00	201.00	201.00	201.00	201.00	203.00	203.00
202.00	201.00	198.00	198.00	197.00	193.00	188.00	188.00
188.00	176.00	187.00	183.00	180.00	178.00	178.00	177.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	190.00	194.00	196.00	198.00
194.00	198.00	198.00	198.00	198.00	198.00	198.00	198.00
198.00	198.00	196.00	192.00	189.00	188.00	186.00	182.00
178.00	174.00	178.00	178.00	177.00	177.00	176.00	175.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	195.00	196.00
188.00	196.00	197.00	195.00	196.00	196.00	196.00	196.00
196.00	196.00	193.00	192.00	188.00	187.00	186.00	182.00
176.00	176.00	172.00	176.00	176.00	175.00	174.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	188.00	192.00	191.00
186.00	190.00	188.00	188.00	188.00	191.00	190.00	194.00
194.00	192.00	192.00	188.00	188.00	186.00	185.00	178.00
175.00	174.00	174.00	170.00	174.00	174.00	174.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	186.00	186.00	189.00	188.00
184.00	188.00	187.00	183.00	186.00	188.00	188.00	190.00
190.00	190.00	190.00	188.00	186.00	185.00	184.00	178.00
176.00	172.00	172.00	171.00	167.00	172.00	172.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	185.00	186.00	187.00
186.00	174.00	185.00	178.00	182.00	186.00	186.00	188.00
188.00	188.00	188.00	186.00	184.00	183.00	178.00	175.00
174.00	172.00	171.00	171.00	164.00	168.00	168.00	168.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	178.00	183.00	183.00
184.00	168.00	178.00	174.00	178.00	178.00	181.00	185.00
186.00	187.00	186.00	186.00	184.00	178.00	176.00	173.00
173.00	171.00	168.00	168.00	161.00	168.00	167.00	9999.00

9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	171.00	176.00	178.00	178.00
178.00	178.00	163.00	173.00	173.00	173.00	178.00	178.00
178.00	178.00	184.00	183.00	178.00	177.00	176.00	172.00
168.00	168.00	166.00	166.00	158.00	166.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	170.00	170.00	170.00	176.00	176.00
176.00	169.00	161.00	161.00	168.00	168.00	175.00	176.00
176.00	176.00	178.00	178.00	176.00	176.00	175.00	168.00
168.00	166.00	166.00	165.00	157.00	159.00	159.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	168.00	168.00	168.00	172.00	172.00
172.00	168.00	162.00	157.00	162.00	164.00	168.00	173.00
173.00	173.00	176.00	176.00	176.00	171.00	168.00	166.00
166.00	163.00	158.00	155.00	160.00	156.00	156.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	158.00	162.00	165.00	169.00	168.00
168.00	168.00	161.00	156.00	158.00	161.00	165.00	168.00
168.00	170.00	168.00	173.00	168.00	9999.00	166.00	163.00
158.00	158.00	153.00	157.00	156.00	155.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	155.00	155.00	158.00	161.00	166.00	168.00
166.00	162.00	153.00	157.00	156.00	158.00	158.00	165.00
166.00	168.00	167.00	168.00	9999.00	9999.00	9999.00	156.00
156.00	156.00	147.00	156.00	154.00	154.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	154.00	154.00	156.00	158.00	161.00	166.00
165.00	158.00	148.00	155.00	154.00	158.00	158.00	158.00
158.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	140.00	148.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	153.00	156.00	157.00	158.00
158.00	156.00	147.00	152.00	152.00	155.00	153.00	154.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	153.00	154.00	156.00
156.00	153.00	146.00	148.00	148.00	148.00	148.00	148.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	147.00
9999.00	148.00	143.00	147.00	9999.00	9999.00	146.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00	9999.00
9999.00	9999.00						

RATE 27.9E -6 1 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	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	0	0	0	0	0	0	0	1	0	0	0	1	1	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	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0						
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0						
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	0	0	1	1	1	1	1	0	0	0	0						
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0						
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

1	1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	C	0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1	1	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	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0

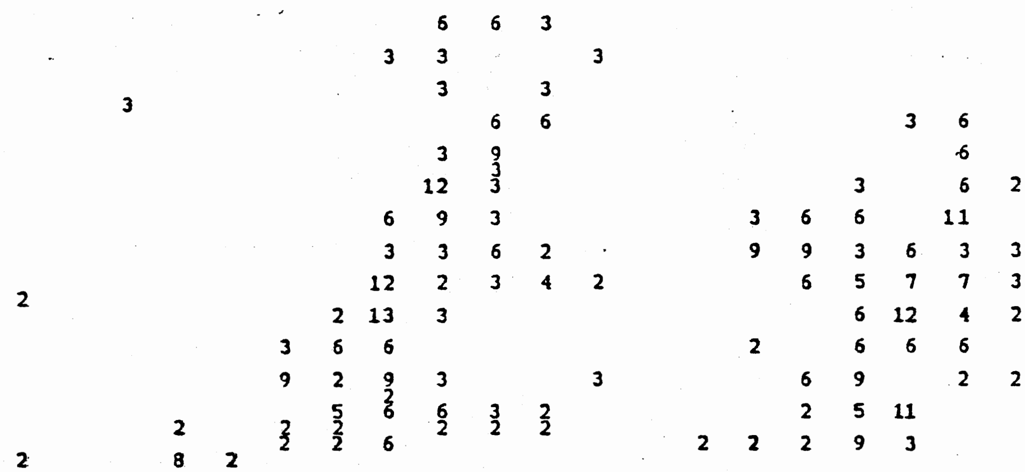
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	0	0	0
1	1	0	0	0	0
1	1	0	0	0	0
1	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

RIVER	1.2	0	1	1
M	-100	0	1	
LAND	1	1	1	1

9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999						
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999						
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	9999	251	9999	9999	9999	253
252	250	9999	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999						
9999	9999	9999	9999	9999	9999	9999	9999
9999	9999	249	253	252	252	252	253
251	248	244	9999	9999	9999	9999	9999
9999	9999	9999	9999	9999	225	9999	9999
9999	9999						
9999	9999	9999	9999	9999	9999	9999	9999
9999	244	250	250	252	252	252	253
245	251	236	9999	9999	9999	9999	9999
9999	9999	9999	9999	220	225	218	9999
9999	9999						
9999	9999	9999	9999	9999	9999	9999	9999
238	242	244	249	250	250	250	250
244	242	236	9999	9999	9999	9999	9999
9999	221	221	221	220	220	216	9999
9999	9999						
9999	9999	9999	9999	9999	9999	211	230
231	240	240	245	243	242	243	241
242	242	240	230	240	237	229	228
9999	225	225	220	218	215	214	9999
9999	9999						
9999	9999	9999	9999	9999	214	220	222

230	232	234	240	240	240	240	240
240	237	232	222	233	230	229	226
223	220	220	218	217	217	214	9999
9999	9999						
9999	9999	9999	9999	209	213	220	224
230	223	232	231	233	233	232	231
232	232	230	229	216	223	222	220
220	215	212	209	208	207	206	9999
9999	9999						
9999	9999	9999	9999	209	212	220	223
225	220	230	230	230	230	230	230
230	230	228	222	222	212	210	209
219	213	210	210	208	207	206	9999
9999	9999						
9999	9999	9999	9999	208	212	218	220
222	218	223	224	223	224	224	221
223	225	221	220	220	217	213	211
206	210	209	207	204	198	196	9999
9999	9999						
9999	9999	9999	9999	208	210	214	217
220	210	220	220	220	220	220	220
220	220	217	213	212	213	211	210
195	206	207	206	200	200	194	193
188	9999						
9999	9999	9999	206	208	210	212	215
212	209	216	214	215	216	212	217
216	216	212	212	211	205	210	208
200	190	200	200	192	194	195	189
188	9999						
9999	9999	9999	206	207	208	210	214
208	213	212	211	212	212	212	213
213	214	210	210	201	208	208	200
195	189	195	195	193	193	190	187
186	9999						
9999	9999	9999	9999	9999	206	209	210
204	210	210	210	210	210	210	210
210	210	208	206	204	203	200	194
190	190	186	190	190	187	186	185
9999	9999						
9999	9999	9999	9999	199	204	206	205
200	205	204	204	203	205	204	208
208	206	206	202	202	201	200	192
190	188	189	185	188	186	188	185
9999	9999						
9999	9999	9999	197	199	200	202	201
195	202	200	195	200	203	202	204
204	204	204	202	200	200	198	189
190	186	186	186	183	186	186	9999
9999	9999						
9999	9999	9999	9999	196	199	200	199
200	192	198	193	194	200	200	202
202	202	202	200	198	197	194	187
190	186	186	185	181	184	184	184
9999	9999						
9999	9999	9999	9999	9999	192	197	197
198	189	193	193	190	195	196	200
200	200	200	200	198	194	192	186
190	186	184	184	175	184	183	178
9999	9999						
9999	9999	9999	9999	185	190	192	192
193	193	180	190	190	190	192	193
193	193	198	197	192	192	191	190
184	184	182	181	171	180	9999	9999
9999	9999						
9999	9999	9999	184	184	184	190	190
190	182	176	176	185	186	190	190
190	190	192	192	191	190	190	184
183	180	180	180	170	172	172	169
9999	9999						
9999	9999	9999	182	182	182	186	186
185	184	180	175	180	176	184	188
188	188	190	190	190	185	182	180
180	180	172	169	171	170	170	167
9999	9999						
9999	9999	167	173	180	180	184	184
184	184	178	174	176	178	180	183
184	184	184	187	182	183	179	177
173	174	168	171	170	169	166	9999

	9999	9999						
	9999	166	168	170	173	175	180	183
	183	180	170	175	174	175	176	180
	180	182	182	182	179	9999	174	170
	170	170	165	170	169	168	164	9999
	9999	9999						
	9999	165	166	166	170	172	175	180
	180	176	159	173	172	173	174	175
	176	169	177	179	9999	9999	9999	161
	166	166	163	165	164	9999	9999	9999
	9999	9999						
	9999	9999	9999	161	168	172	170	174
	173	174	165	170	170	170	170	170
	165	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999						
	9999	9999	9999	9999	161	168	168	170
	170	170	159	166	166	166	168	168
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999						
	9999	9999	9999	9999	9999	9999	158	160
	161	160	155	160	159	159	160	158
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999						
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999						
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999	9999	9999	9999	9999	9999	9999
	9999	9999						
QRE		24	0	1	1			
WELL	-0.4000000		1	1	-0.4000000		1.0	



```
                               9   3               2           9       3   2
           2   4   6       2   2
                               6   6                       6
               6
                               5   9   3   2           6   7  12   3
               2           2   6   6           2   3   6   9  15
               2                   6           2       6   5   6
                           5  11   5   2              12   9   3   2
                           6   3   2                   6   9   3
                                               2
```

```
DELX             5280
DELY             5808
   90                                                                    00000800
TEXT           1  100                                                    00001100
A810610.2023  ALLOC=0.015CFS/AC;PERM=600GPD;RIVER=STRT+1.2;BOT=300
DPUR (S.A.HOSSAIN)
UNIT          -2       -1                                  NO CARD LIST00002400
START               3                                                  00002600
LABEL          1                                                       00002700
F              1   18   18              45.929E-10         SCALE       00006300
F              0   23                                      CWELL       00007800
F              0   21                                      GRAD  PATT 0 00010000
F              0   22                                      GRAD  VALUE 00010100
CARD          22   -8   -9                                 READ  GRAD  00010500
```

```
                                            1               1   1   2
                                       -1       2   2   2              -4
                                    2
                                   -2                                  -4
                                1
                               -2      -2
                                                                        4
                     1   1   3         -2
                        -9
   2          -2                       -1
                    -7
                                       -1
                -1
                                       -1
                -1
                                       -2
                -2
                                       -2
                -6
                                                -1
            -1
                                                -1
            -1
                                                -1
                    -1
                                            -1
                -1
                                            -1
            -1
                -1

                                            -1

                                            -2
                                            -1
```

```
                      -3
                -2                                          -1
                -1
          -1          -4                                    -2
                                                                                                 -1    -2    -1
                -1
                            -6   -2   -2          -4
                            -4                                                              -1

                                 -4

                                           -1       -7                   -1   -1       -2


F                            1   22   22                       1.0                    GRAD MULT     00011900
F                            0   21              1.                                   GRAD PATT     00012000
C                            0   21         22   0.                      0.           GRAD PATT     00012100
C                            2    4    4    21   20000.        1.        0.           ADD 20000.    00012200
C                            1   18          4                           20000.       NO CF RECH    00015200
C                            0   18          7                           9998.        NO LAKE RECH0 00015700
C                            0   19          7                           9998.        NO LAKE WELL0 00016000
C                            0   19          7                           9998.        NO LAKE WELL0 00016100
F                            0   24   18                                              SAVE RECH     00018700
F                            0   36    6                                              SAVE TOP      00018900
UNIT                         5   20    1    52                                        STD FILE OUT0 00019300
OPENW                        1   20    1                                                            00019400
WRITE                        4                                                        ISUR(STRT)    00019500
WRITE                        5                                                        BOTTOM        00019600
WRITE                        6                                                        TOP           00019700
WRITE                        7                                                        LAND          00019800
WRITE                        8                                                        RIVER         00019900
WRITE                        9                                                        H             00020000
WRITE                       10                                                        S             00020100
WRITE                       11                                                        SY            00020200
WRITE                       12                                                        PERM          00020300
WRITE                       13                                                        TRANS         00020400
WRITE                       16                                                        RATE(RIV K)   00020500
WRITE                       18                                                        QRS (RECH)    00020600
WRITE                       19                                                        WELLD         00020700
WRITE                       22                                                        GRAD          00020800
WRITE                       23                                                        CWELL         00020900
FUNCTION                    -5               5                                        SAT           00021000
WRITE                        0                                                        SAT           00021100
GO TO                        1                                                                      00021400
BEGIN                                                                                               00021600
FUNCTION                    -5               5                                        SAT           00021900
DEPTH                       -2                                                        BEFOR PERIOD0 00022200
BEGIN                                                                                               00023000
PERIOD                       1         2                                              ODD PERIODS0  00023100
F                            0   18   24                                              RECH ON       00023200
END                                                                                                 00023300
BEGIN                                                                                               00023400
PERIOD                       2         2                                              EVEN PERIODS0 00023500
F                            0   18                                                   RECH OFF      00023600
END                                                                                                 00023700
BEGIN                                                                                               00024100
PERIOD                       1         2                                              ODD PERIODS0  00024200
F                            0    6              20001.                               TOP OFF       00024300
END                                                                                                 00024400
BEGIN                                                                                               00024600
PERIOD                       2         2                                              EVEN PERIODS0 00024700
F                            0    6   36                                              TOP ON        00024800
END                                                                                                 00024900
END                                                                                                 00025100
BEGIN                                                                                               00025400
DEPTH                       -3                                                        BEFORE STEP   00025500
BEGIN                                                                                               00026000
PERIOD                       2         2                                              EVEN PERIODS0 00026100
F                            1   -1    0     5                                        HEAD          00026300
F                            0   -1   -1     8                                        HEAD-RIVER    00026400
C                            0   -1         16                                        ONLY RIVER    00026500
C                            2    6   36    -1                 1.                     >0= TOP ON    00026600
C                           -1    6   36    -1   20000.        1.                     <0= TOP OFF   00026700
END                                                                                                 00026900
BEGIN                                                                                               00028100
PERIOD                       1    1                                                                 00028200
```

```
STEPP           1    1                                                    MAY 1        00028300
F               1   -1    19    23              -.00355114-.001420451RR+CITY           00028500
C              -2   -1          -10.015                   0.015        ALLOC           00028800
F               1   -1    -1    23              281.6     0.400000     LESS CITY       00028900
C              -1   -1          12                                     BLANK BORDER00029000
C               0   -1           7                        9998.        NO LAKE WELL00029200
C              -2   19    -1     4 0.           -1.       20000.       NCT CF          00030000
WRITE          19                                                      WELL            00031600
WRITE          23                                                      CWELL           00031700
END                                                                                    00031800
F               1   -1     0                                           SAT             00032200
C              -2   -1          -1 0.                     0.           MIN SAT 0       00032300
F               0   -2     7     5                                     AQ DEPTH        00032400
C              -3   -1    -1    -2                                     MAX SAT         00032500
F               2   -1    -1    12                                     TRANS SAT*K     00032600
F               2   -1    -1    22                                     CF=GRAD*T       00032700
C               2   20    -1    21              1.                     TO WELL         00032800
END                                                                                    00033300
BEGIN                                                                                  00033800
DEPTH           3                                                      AFTER STEP      00033900
BEGIN                                                                                  00034200
PERIOD         10   40    10                                           5 YEAR          00034300
STEPP          15   15                                                 APRIL 30        00034400
WRITE          11   11                                                 SY              00034500
WRITE          13   13                                                 TRANS           00034600
END                                                                                    00034800
END                                                                                    00034900
BEGIN                                                                                  00035200
DEPTH           2                                                      AFTER PERIOD00035300
BEGIN                                                                                  00035600
PERIOD          1    2                                                 FIRST YEAR      00035700
WRITE          11   11                                                 SY              00035800
WRITE          13   13                                                 TRANS           00035900
END                                                                                    00036000
BEGIN                                                                                  00036300
PERIOD         40                                                      OCT 1993        00036500
WRITE          11   11                                                 SY              00036600
WRITE          13   13                                                 TRANS           00036700
WRITE           0                                                      SAT             00036800
CLOSE                                                                                  00036900
UNIT            1   59     1     8                                                     00037000
START           1   -1                                                                 00037100
END                                                                                    00037200
WRITE           0                                                      SAT             00037500
END                                                                                    00037600
EOD            -3    3                                                                 00037700
WELL      1     0    0   210  9999     1. 240.     0     0    -1    -1          3
WELL      2     1    0   150  9999     1. 240.     0     1    -1    -1          3
WELL      3     2    0   210  9999     1. 240.     0     0    -1    -1         99
WELL      4     3    0   150  9999     1. 240.     0     1    -1    -1
WELL      5     4    0   210  9999     1. 240.     0     0    -1    -1
WELL      6     5    0   150  9999     1. 240.     0     1    -1    -1
WELL      7     6    0   210  9999     1. 240.     0     0    -1    -1
WELL      8     7    0   150  9999     1. 240.     0     1    -1    -1
WELL      9     8    0   210  9999     1. 240.     0     0    -1    -1
WELL     10     9    0   150  9999     1. 240.     0     1    -1    -1
WELL     11    10    0   210  9999     1. 240.     0     0    -1    -1
WELL     12    11    0   150  9999     1. 240.     0     1    -1    -1         99
WELL     13    12    0   210  9999     1. 240.     0     0    -1    -1
WELL     14    13    0   150  9999     1. 240.     0     1    -1    -1
WELL     15    14    0   210  9999     1. 240.     0     0    -1    -1
WELL     16    15    0   150  9999     1. 240.     0     1    -1    -1
WELL     17    16    0   210  9999     1. 240.     0     0    -1    -1
WELL     18    17    0   150  9999     1. 240.     0     1    -1    -1
WELL     19    18    0   210  9999     1. 240.     0     0    -1    -1
WELL     20    19    0   150  9999     1. 240.     0     1    -1    -1
WELL     21    20    0   210  9999     1. 240.     0     0    -1    -1
WELL     22    21    0   150  9999     1. 240.     0     1    -1    -1         99
WELL     23    22    0   210  9999     1. 240.     0     0    -1    -1
WELL     24    23    0   150  9999     1. 240.     0     1    -1    -1
WELL     25    24    0   210  9999     1. 240.     0     0    -1    -1
WELL     26    25    0   150  9999     1. 240.     0     1    -1    -1
WELL     27    26    0   210  9999     1. 240.     0     0    -1    -1
WELL     28    27    0   150  9999     1. 240.     0     1    -1    -1
WELL     29    28    0   210  9999     1. 240.     0     0    -1    -1
WELL     30    29    0   150  9999     1. 240.     0     1    -1    -1
WELL     31    30    0   210  9999     1. 240.     0     0    -1    -1
WELL     32    31    0   150  9999     1. 240.     0     1    -1    -1         99
```

WELL	33	32	0	210	9999	1.	240.	0	0	-1	-1
WELL	34	33	0	150	9999	1.	240.	0	1	-1	-1
WELL	35	34	0	210	9999	1.	240.	0	0	-1	-1
WELL	36	35	0	150	9999	1.	240.	0	1	-1	-1
WELL	37	36	0	210	9999	1.	240.	0	0	-1	-1
WELL	38	37	0	150	9999	1.	240.	0	1	-1	-1
WELL	39	38	0	210	9999	1.	240.	0	0	-1	-1
WELL	40	39	0	150	9999	1.	240.	0	1	-1	-1

VITA 2

Syed Abul Hossain

Candidate for the Degree of

Master of Science

Thesis: APPLICATION OF NUMERICAL MODEL FOR GROUND-WATER MANAGEMENT OF PIEDMONT AQUIFER IN CENTRAL DINAJPUR, BANGLADESH

Major Field: Geology

Biographical:

Personal Data: Born in Bangladesh, March 1, 1937, the son of Mr. Syed Kayem Ali.

Education: Graduated in Intermediate of Science from Dacca Government College in 1957; received Bachelor of Science in 1959 and Master of Science in 1961, from University of Dacca, Dacca, Bangladesh, with a major in Geology; completed requirements for the Master of Science degree at Oklahoma State University in July, 1981.

Professional Experience: Geologist, Groundwater Circle, Bangladesh Water Development Board, 1961-67, and Deputy Director, 1967-79; attended groundwater group training course in Tokyo, Japan in 1973; graduate research assistant, Oklahoma State University, Department of Geology, 1980-81.