US Army Corps of Engineers

ENVIRONMENTAL AND WATER QUALITY OPERATIONAL STUDIES

INSTRUCTION REPORT E-87-2

SELECT: A NUMERICAL, ONE-DIMENSIONAL MODEL FOR SELECTIVE WITHDRAWAL

by

Jack E. Davis, Jeffery P. Holland, Michael L. Schneider, Steven C. Wilhelms

Hydraulics Laboratory

DEPARTMENT OF THE ARMY
Waterways Experiment Station, Corps of Engineers
PO Box 631, Vicksburg, Mississippi 39180-0631

March 1987

Final Report

Approved For Public Release; Distribution Unlimited

Prepared for DEPARTMENT OF THE ARMY
US Army Corps of Engineers
Washington, DC 20314-1000

Under CWIS Work Unit 31604
(EWQOS Work Unit IIIA.4)

Monitored by Environmental Laboratory
US Army Engineer Waterways Experiment Station
PO Box 631, Vicksburg, Mississippi 39180-0631

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188
Exp. Date: Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) Instruction Report E-87-2
6a. NAME OF PERFORMING ORGANIZATION USAEWES Hydraulics Laboratory 6b. OFFICE SYMBOL (If applicable) WESHS-R	7a. NAME OF MONITORING ORGANIZATION USAEWES Environmental Laboratory
6c. ADDRESS (City, State, and ZIP Code) PO Box 631 Vicksburg, MS 39180-0631	7b. ADDRESS (City, State, and ZIP Code) PO Box 631 Vicksburg, MS 39180-0631
8a. NAME OF FUNDING/SPONSORING ORGANIZATION US Army Corps of Engineers 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) Washington, DC 20314-1000	10. SOURCE OF FUNDING NUMBERS See reverse PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
SELECT: A Numerical, One-Dimensional Model for Selective Withdrawal

12. PERSONAL AUTHOR(S)
Davis, Jack E., Holland, Jeffery P., Schneider, Michael L., and Wilhelms, Steven C.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final report	FROM _____ TO _____	March 1987	192

16. SUPPLEMENTARY NOTATION
Available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Hydraulic structures; Selective withdrawal; Release water quality; Stratified flow; Reservoir hydrodynamics

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report was developed to aid users of the program SELECT. It provides guidance on how to construct an input file and how to interpret the output file. It also overviews the concepts behind the program and the internal methodology of its execution. Examples of input and output are also provided.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code) / 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

10. SOURCE OF FUNDING (Continued).

CWIS Work Unit 31604
(EWQOS Work Unit IIIA.4)

PREFACE

Development of this report was sponsored by the Office, Chief of Engineers (OCE), US Army, as a part of the Environmental and Water Quality Operational Studies (EWQOS) Program, Work Unit IIIA.4 (CWIS Work Unit 31604), entitled "Techniques to Meet Environmental Quality Objectives for Reservoir Releases." The OCE Technical Monitors of the EWQOS Program were Mr. Earl E. Eiker, Dr. John Bushman, and Mr. James L. Gottesman. Program Manager of EWQOS was Dr. J. L. Mahloch.

This report was prepared by the Hydraulics Laboratory (HL), US Army Engineer Waterways Experiment Station (WES), under the general supervision of Mr. F. A. Herrmann, Jr., Chief of the HL; Mr. H. B. Simmons, former Chief of the HL; and Mr. John L. Grace, Jr., Chief of the Hydraulic Structures Division; and under the direct supervision of Mr. Jeffery P. Holland, Chief of the Reservoir Water Quality Branch. The report was written by Messrs. Jack E. Davis, Jeffery P. Holland, Michael L. Schneider, and Steven C. Wilhelms. Messrs. Terry Reeves and James A. Daub contributed in the preparation of the variable lists and Appendix A. The report was prepared for publication by Ms. Jessica S. Ruff of the WES Information Products Division of the Information Technology Laboratory.

COL Allen F. Grum, USA, was the previous Director of WES. COL Dwayne G. Lee, CE, is the present Commander and Director. Dr. Robert W. Whalin is Technical Director.

This report should be cited as follows:

Davis, J. E., et al. 1987. "SELECT: A Numerical, One-Dimensional Model for Selective Withdrawal," Instruction Report E-87-2, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

CONTENTS

	Page
PREFACE	1
PART I: INTRODUCTION	4
Background	4
Purpose and Scope	5
PART II: AN OVERVIEW OF SELECT AND ITS USE	6
Program Purpose	6
Conceptual Methodology	7
Overview of Program Execution	8
Available Assistance	13
PART III: PROGRAM DESCRIPTION	14
Main Program	14
DENINT	17
INTERP	23
OUTVEL	28
VPORT	34
VWEIR	50
SHIFT	59
XPRINT	67
DVPLOT	71
ERROR	75
XREAD	77
AERATE	83
VENTING	85
PART IV: ASSUMPTIONS AND LIMITATIONS	87
Geometry of Ports	87
Impoundment Width	87
Approach Path	88
Approach Curvature	88
Multiple Horizontal Ports	88
Weir Crest Above Thermocline	89
Hydraulic Integrity	89
Simultaneous Port-Weir Operation	89
PART V: INPUT DATA	90
Descriptions	90
Profile Formats	93
REFERENCES	94
APPENDIX A: INPUT FORMAT DESCRIPTION	A1
Notes on Input Format	A1
SELECT Program-Input Format Description	A2

	Page
APPENDIX B: INPUT FILE EXAMPLES	B1
Example File 1	B1
Example File 2	B2
Example File 3	B3
Example File 4	B4
APPENDIX C: OUTPUT EXAMPLE	C1
APPENDIX D: ERROR CODES	D1
APPENDIX E: PROGRAM CODE	E1

SELECT: A NUMERICAL, ONE-DIMENSIONAL MODEL FOR SELECTIVE WITHDRAWAL

PART I: INTRODUCTION

Background

1. As a result of increased public awareness and state and Federal legislation, Corps of Engineers (CE) water resources projects are being operated with an emphasis on water quality considerations. The use of a reservoir outlet works incorporating fixed or multilevel selective withdrawal structures is a primary method for the control of reservoir release quality. These structures release water from specified strata in a density-stratified reservoir, thereby allowing, through blending of flows or direct release, greater control of water quality. Two general conditions must be met to ensure that a given structure has the ability to selectively withdraw reservoir waters. First, if the structure exists, the outlets must be at appropriate elevations and have sufficient capacity such that particular outlets can be operated to withdraw water of a desired quality. If the structure is proposed, the number, locations, and capacities of outlets should be determined as part of the design procedure to ensure that the required level of release water can be maintained. Second, for efficient operation of an existing structure or optimal design of a proposed structure, the operator or designer must have the capability to describe the zone of withdrawal formed by a discharge through a selective withdrawal outlet for a given outlet geometry, location, and discharge. Additionally, proper hydraulic performance of such structures acts as a constraint for any solution.

2. Research at the US Army Engineer Waterways Experiment Station (WES) has provided extensive technology with which to assist the operator or designer in addressing the conditions listed above. Through laboratory experimentation, Bohan and Grace (1969) and Grace (1971)

described the withdrawal zone formed by releases from a density-stratified reservoir through ports (orifice flow) and over weirs (i.e., spillway flow, thermal berms). An outgrowth of this initial research was the development of the one-dimensional computer program SELECT, which computes the withdrawal zone formed by a given release through or over a specified outlet structure for a known reservoir density stratification. The program also computes the quality characteristics of the release for user-specified parameters (temperature, dissolved oxygen, etc.) treated as conservative substances. Additional work, which includes Dortch and Holland (1984), describes a numerical procedure that systematically evaluates the optimal number and locations of selective withdrawal intakes required to meet a specified release quality objective. Holland (1984) also overviews the steps that must be followed in the design of selective withdrawal structures.

3. Recent research efforts, building on the work of Bohan and Grace (1973) and many other researchers, have resulted in more-generalized techniques for the description of the withdrawal zone formed by a reservoir release through a port acting as a point sink. Smith et al. (1987) document these techniques. Subsequently, the results of these and other research efforts have been incorporated into an update of the SELECT program.

Purpose and Scope

4. The purpose of this report is to document the updated version of the SELECT program for field office use. This will be done by describing the computational methodologies and the sequence of operations in SELECT (Part II), the operations used in the subprograms (Part III), and the assumptions and limitations inherent to the code (Part IV). Definitions of the data required as input are provided in Part V. Additional information presented in the report includes a description of the input format (Appendix A), input file and output examples (Appendixes B and C), error codes (Appendix D), and program code (Appendix E).

PART II: AN OVERVIEW OF SELECT AND ITS USE

Program Purpose

5. The SELECT program is a one-dimensional numerical model that predicts the vertical extent and distribution of withdrawal from a reservoir of known density and quality distribution for a given discharge from a specified location. Using this prediction for the withdrawal zone, SELECT computes the quality of the release for user-specified parameters (such as temperature, dissolved oxygen (DO), turbidity, iron) treated as conservative substances. The release constituents are considered conservative through the selective withdrawal structure because the detention time in the structure is short compared with the time required for the constituents to physically or chemically change. For example, there would be no time for the water temperature to change significantly nor would there be time for iron to oxidize significantly. SELECT will predict, however, the improvement in DO that would occur due either to natural reaeration, as flow passes through gated-conduit outlet works, or to turbine venting.

6. It is important for the user to realize the purpose of SELECT. SELECT was developed based on the philosophy that the field office users require a tool to compute the withdrawal and release quality characteristics of a structure for given values of density stratification, outlet geometry, and discharge. SELECT is that tool. SELECT is not a water quality or thermal simulation model. It does not consider all the hydrodynamic and biochemical processes ongoing in a reservoir. Its purpose is to compute withdrawal and release quality characteristics.

7. Many times the computation of reservoir release characteristics is, within itself, sufficient to provide insight into the solutions to the posed problems. For example, the day-to-day operation of a multilevel outlet works or the initial design of a thermal berm could be performed with SELECT.

Conceptual Methodology

8. SELECT accomplishes its assigned tasks, subject to various assumptions and limitations, by first dividing the reservoir pool into a user-specified number of layers of equal thickness. (The user should first note the program assumptions and limitations outlined in Part IV.) The layers are assumed in the code to be longitudinally and laterally homogeneous in quality. The effects of density stratification are assumed to act only in the vertical dimension on the release from the project. The velocities induced by the release are also assumed to vary only in the vertical direction, in keeping with the one-dimensional assumption of the program.

9. SELECT computes the limits of withdrawal (defined as the vertical locations in the pool beyond which there is no contribution to the total release), the vertical distribution of withdrawal between those limits, the point of maximum withdrawal, and the outflow quality characteristics. To do this, SELECT first computes a normalized velocity profile for the region within the withdrawal zone of each port or weir. This profile is normalized in the sense that the individual point velocities are divided by the maximum profile velocity and subsequently scaled such that their integration over the profile would yield the discharge producing them.

10. The purpose of computing the normalized velocity profile is not to provide a prediction of actual velocities in the reservoir; rather, the normalized velocity profile is used to generate the reservoir withdrawal profile as a final product. The individual withdrawal profiles from each outlet are then summed, yielding a total withdrawal profile with corresponding withdrawal limits. With the total withdrawal profile known, the release quality can be determined by weighting the quality contribution of each layer with respect to the ratio of the layer discharge to the total discharge.

11. The user should also be alerted to the phrase "theoretical limit of withdrawal" used in this report. This phrase is used to denote a withdrawal condition for which the potential to withdraw flow from

much higher or lower in the pool would have existed had the withdrawal zone not been truncated by the physical boundaries of the reservoir, e.g., the water surface or bottom, respectively. The theoretical limit stands in contrast to the physical limit taken to be the water surface or bottom.

12. A final concept must be formalized prior to use of the model. SELECT uses the concept of "withdrawal angle" to incorporate the effects of many local topographical conditions upon port withdrawal zone formation. In plan view, the withdrawal angle can be thought of as the angle the outlet works structure makes with the local topography. The plan views in Figures 1a and 1b show effective withdrawal angles of π (180 deg) and $\pi/2$ (90 deg), respectively. These conditions typify the most common means of withdrawal through an outlet tower in the middle of a dam face and at the edge of a dam face near a wall.

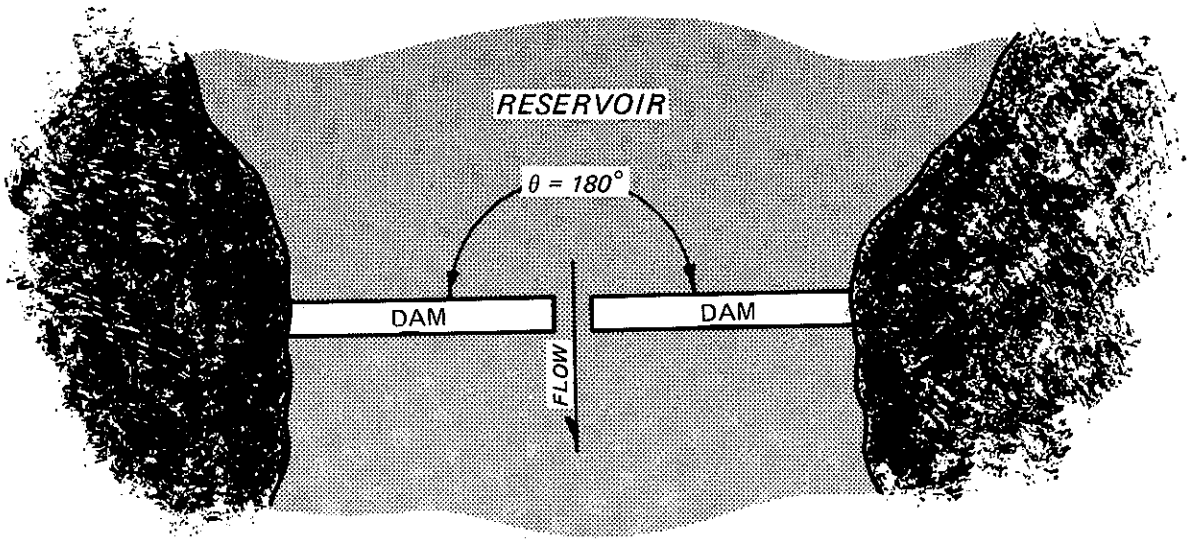
13. SELECT incorporates the concepts described above in a set of transcendental equations that must be iteratively solved to obtain the withdrawal zone characteristics.

Overview of Program Execution

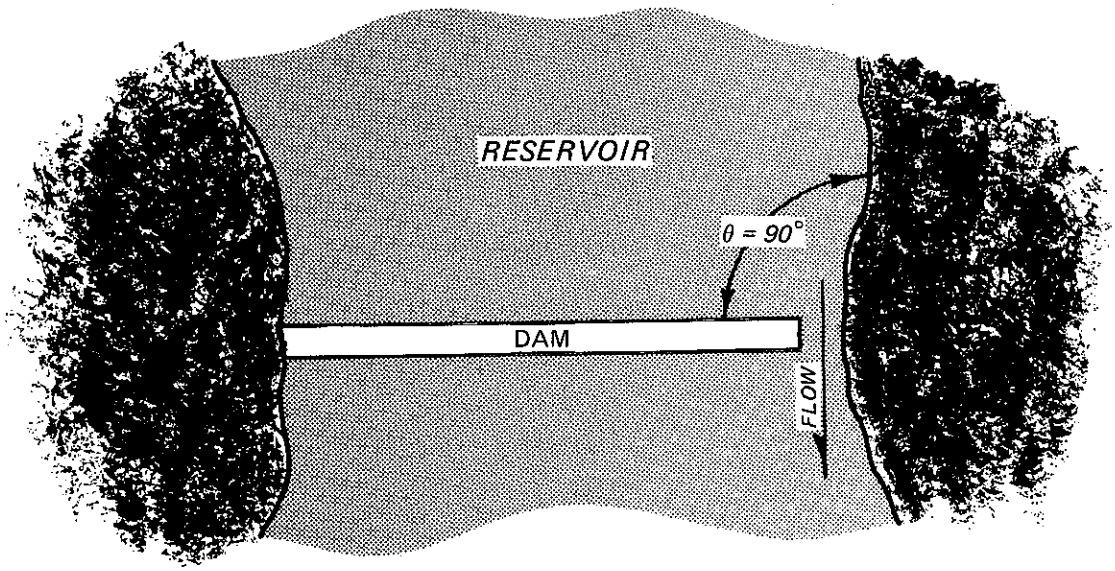
14. The following paragraphs outline the sequence of operations that SELECT executes to determine the withdrawal limits and release water quality. Figure 2 is a flowchart that depicts this sequence. The branches at decision points in the flowchart are labeled T and F for true or false. Figure 3 gives the program call sequence from each subprogram.

15. SELECT's main program orders the sequence in which the subprograms are executed. Subroutine XREAD is called first by the main program to read information from the input file concerning program control parameters such as the number of data sets in the input file. XREAD is then called again to read information about the impoundment, the outlets, and the vertical distribution of all water quality parameters being modeled.

16. After each water quality parameter distribution is entered,



a. Withdrawal angle, 180 deg



b. Withdrawal angle, 90 deg

Figure 1. Plan view of reservoirs showing 180- and 90-deg withdrawal angles (θ)

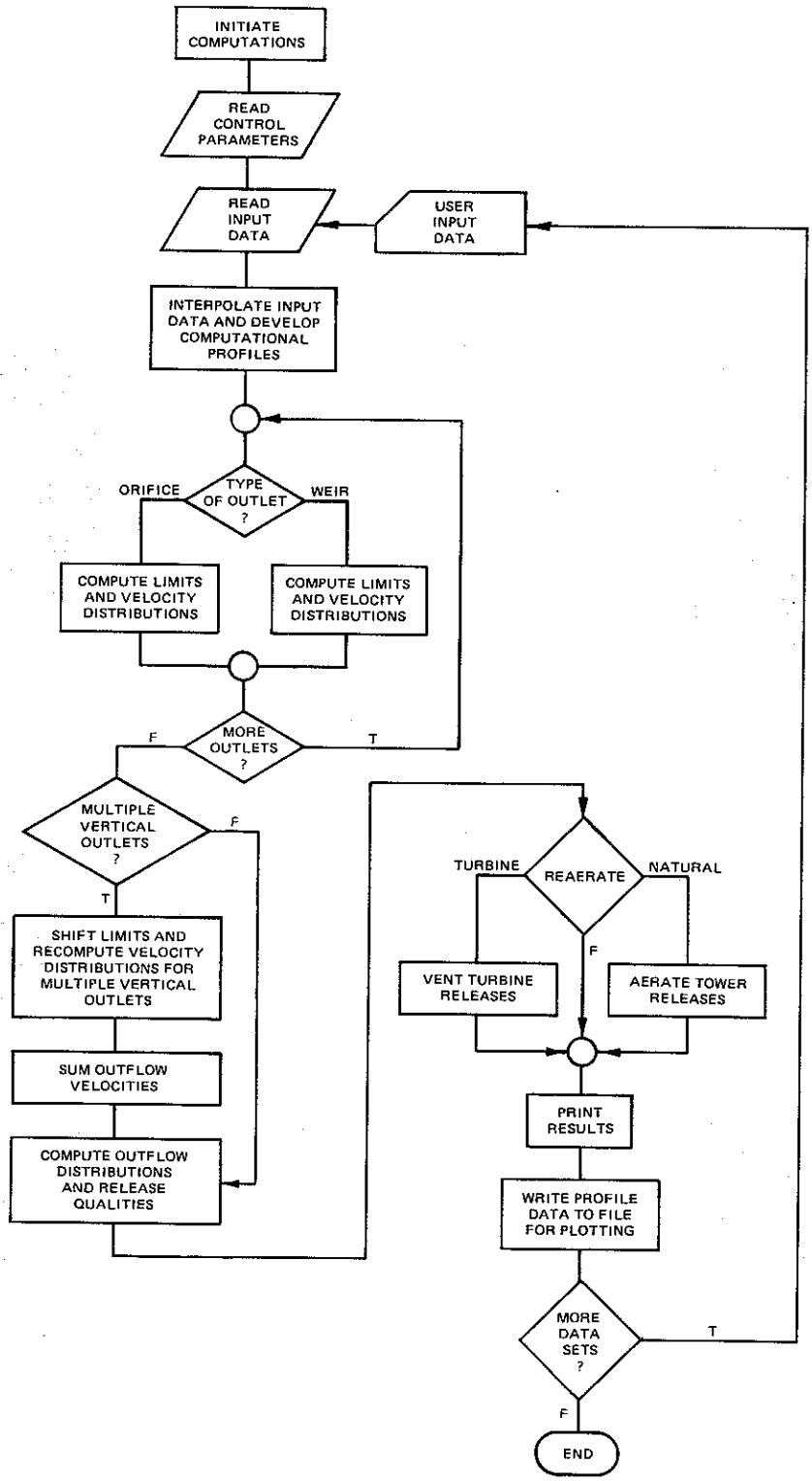


Figure 2. SELECT flowchart

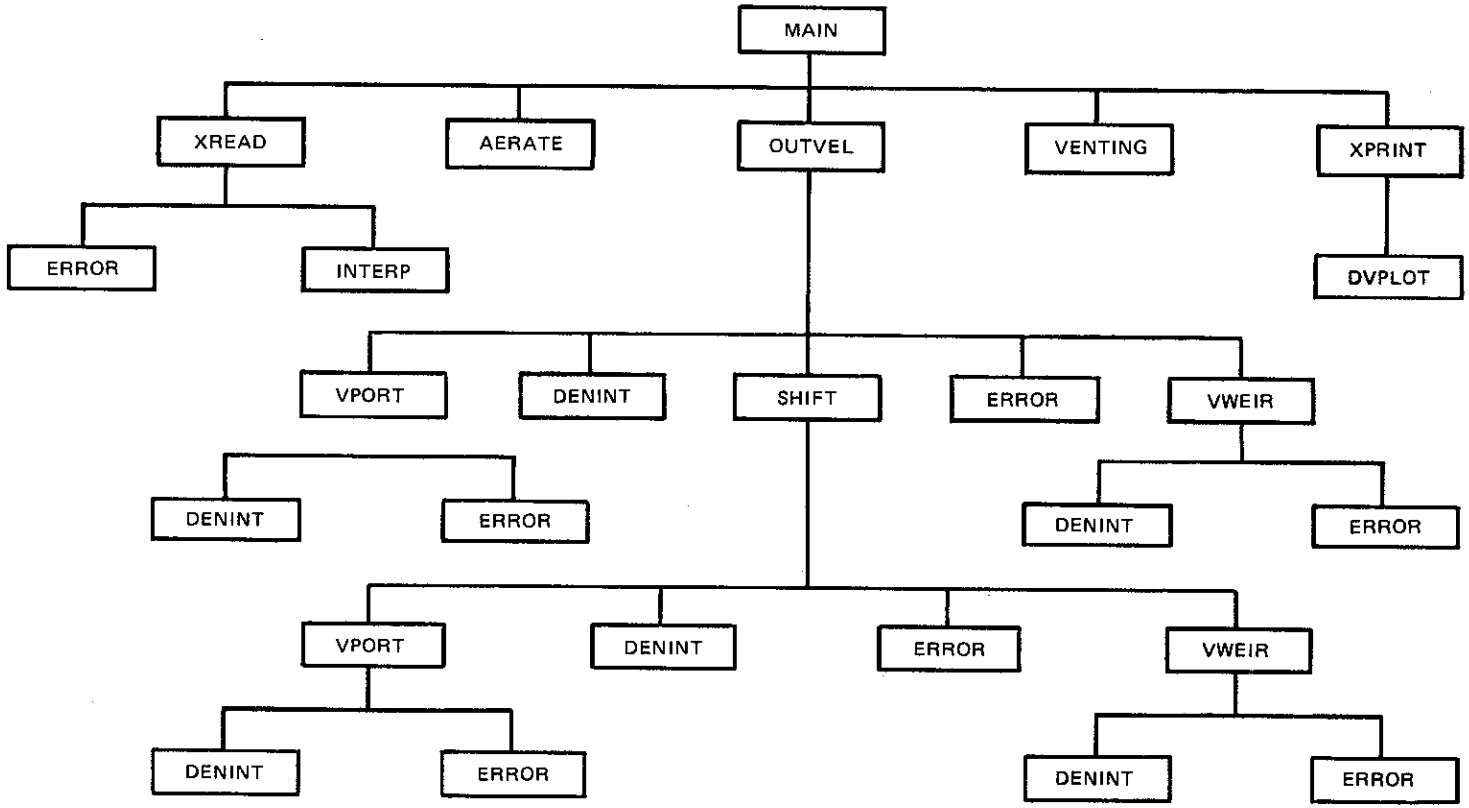


Figure 3. The branched system illustrates the call sequence from each subprogram

subroutine INTERP interpolates between input parameter values at their respective elevations to produce values at the center lines of evenly spaced and vertically distributed horizontal layers (see subroutine INTERP description for greater detail on horizontal layers). Vertical distributions for each parameter are thus created for computational purposes. SELECT bases all of its calculations on these distributions.

17. Subroutine OUTVEL is the next to be called in SELECT's execution. OUTVEL is a subprogram that carries out the organizational duties of the computational portion of the program. Based on whether a weir or a port is being modeled, OUTVEL calls subroutine VWEIR or VPORT, respectively, which are the main computational subroutines in SELECT. VWEIR and VPORT generate the withdrawal zone limits and compute the discharges or flow rates from each computational layer within the withdrawal zone. Program control is then moved back to OUTVEL where all of the withdrawal zone discharges for each outlet modeled in VPORT and VWEIR are summed. The subroutine then normalizes the withdrawal distribution, effectively creating the total normalized velocity distribution. OUTVEL then determines the total release water quality parameter values by summing the contributions of each layer for each parameter modeled. To compute these release values, OUTVEL assumes that each parameter acts as a conservative constituent during the withdrawal process, since detention time in the structure during the release is small compared with the time required for the constituents to change physically or chemically.

18. If withdrawal zones from different outlets overlap vertically, OUTVEL calls subroutine SHIFT to adjust the withdrawal zone limits and outflow profiles from each overlapping zone before the zones are summed as previously stated.

19. If adjustments to the DO content of the release flow are desired based on aeration of the release by the outlet structure (due to turbine venting or natural structural reaeration processes, as described in paragraphs 5-7), the adjustments are performed just prior to outputting the final results.

20. The results are output by subroutines XPRINT and DVPLOT. A plot of the normalized velocity and density distribution is generated by

DVPLOT, and a tabular listing of profile data such as DO, normalized velocity, withdrawal, temperature, and other quality constituents is generated by XPRINT along with information about the outlet characteristics and withdrawal zone limits. Details of each of these subprograms are given in Part III.

Available Assistance

21. Assistance in understanding, setting up, or executing the SELECT program or in analyzing its results is available to CE Field Operating Agency users. Copies of the SELECT code are available for use on the CYBERNET system, the WES DPS system, and certain personal computers.

PART III: PROGRAM DESCRIPTION

22. The following sections give detailed descriptions of the equations and logic in SELECT and its subprograms. A written description of each subprogram is given along with a flowchart diagramming the operational sequence of the subprogram. Note that throughout the report, T and F signify true and false at decision points in the given flowcharts. Listings of the variables in each subprogram and their definitions are also given.

Main Program

23. The main program of SELECT is very small, with the bulk of the program being found in the functions and subroutines. The main program basically orders the calling sequence of the subprograms to perform input operations, calculations, and output operations (Figure 4). Table 1 lists the variables used in the main program.

24. The main program also performs other tasks that are very important to the operation of SELECT. The first is the generation of a density profile based on a user-input temperature profile. Generally, the user does not have density profile information on the impoundment being modeled but does have temperature data. SELECT will use an input temperature profile to generate densities for each discrete layer. The equation used to convert temperature to density is

$$\rho_I = 1 - \frac{(T_I - 3.9863)^2}{508{,}929.2} * \frac{(T_I + 288.9414)}{(T_I + 68.12963)} \qquad (1)$$

where

ρ_I = density in grams per cubic centimeter for layer I

T_I = temperature in degrees Centigrade for layer I

I = index designating a specified layer

The main program also checks the stability of the density profile that was given or generated. The profile is stable if the density never <u>decreases</u> with increasing depth. If the profile is found unstable, the program will terminate execution and issue an error message.

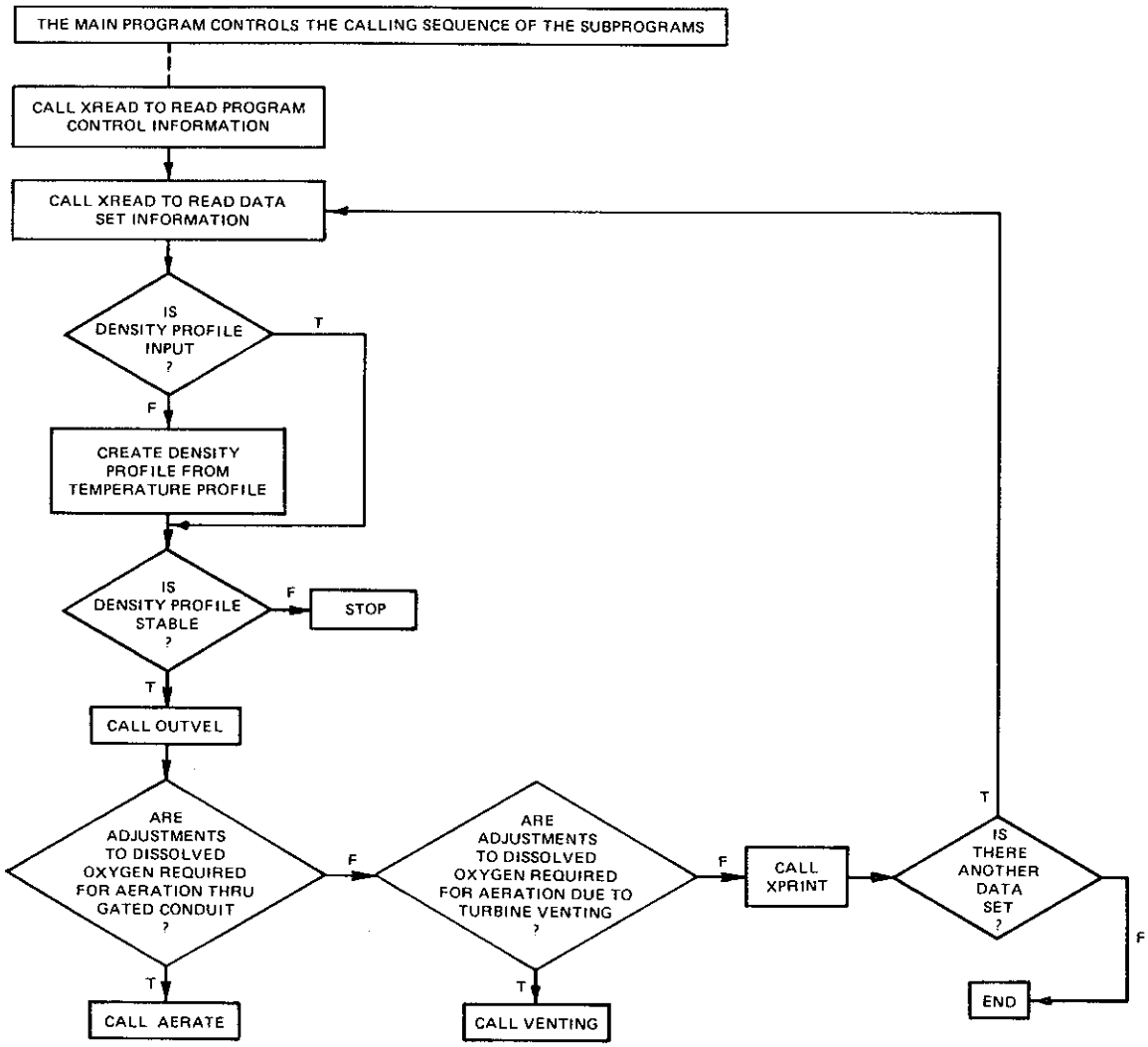


Figure 4. Main program flowchart

Table 1

Main Program Variables

Variable	Definition
C1	Constant = -3.9863 ; used in determining density correspondence to a particular temperature
C2	Constant = 508,929.2 ; used same as C1
C3	Constant = 288.9414 ; used same as C1
C4	Constant = 68.12963 ; used same as C1
DEN(J)	Density of layer J, g/ml
ISURF	Total number of layers in a profile
KFILE	Output file code
NSETS	Number of data sets in the input file
QAERA	Logical variable; true, consider DO uptake due to aeration through a gated structure outlet works; false, do not consider aeration
QCENT	Logical variable; true, temperatures are in degrees Centigrade; false, Fahrenheit
QDEN	Logical variable; true, density profile is input; false, develop density profile from temperature
QVENT	Logical variable; true, consider DO uptake due to turbine venting; false, do not consider uptake
TEMP(J)	Temperature of layer J, degrees Fahrenheit or Centigrade

DENINT

Description

25. Function DENINT determines the density at any elevation. The user-input density profile (or temperature profile which SELECT converts to a density profile) is interpolated by subroutine INTERP to give densities at each layer center line. When the density at some arbitrary elevation is desired, DENINT uses the layer center-line densities above and below the given elevation and linearly interpolates to determine the density at the given elevation. Figure 5 shows a schematic of the algorithm flowchart, and Table 2 lists the variables used in DENINT.

26. The interpolation equation is

$$\Gamma_A = \Gamma_{I+1} - \left(\frac{d}{D}\right)\left(\Gamma_{I+1} - \Gamma_I\right) \tag{2}$$

where

Γ_A = density at the given elevation of point A

Γ_{I+1} = density at the elevation of the layer I+1 center line immediately above point A

d = distance between the elevation of point A and the elevation of the layer center line immediately above

D = distance between the layer center lines immediately above and immediately below point A

Γ_I = density at the elevation of the layer I center line immediately below point A

A schematic defining these variables is shown in Figure 6.

Special considerations

27. If point A lies within 0.01 percent of the layer thickness of a center line, the density at the elevation of point A is assigned the value of the density at the layer center line.

28. When a density value is desired for an elevation that is above the impoundment surface or below the impoundment bottom, linear extrapolation is used (the need for such a calculation is explained in the section on subroutine VPORT). For an elevation above the surface, SELECT uses the change in density and difference in elevation between

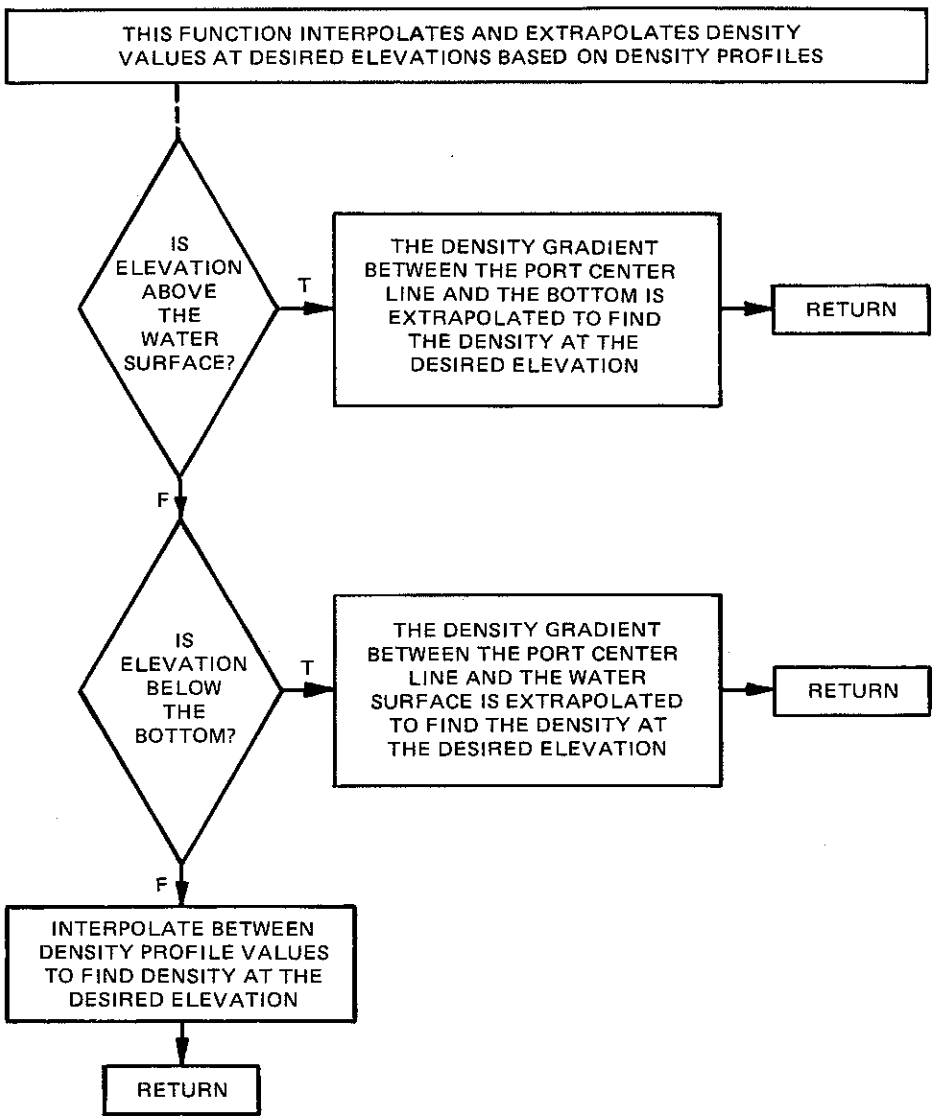


Figure 5. Flowchart for function DENINT

Table 2

DENINT Variables

Variable	Definition
DELZ	Layer thickness
DEN(J)	Density of layer J, g/cm^3
DEN(IJ)	Density of upper interpolation layer, g/cm^3
DEN(JK)	Density of lower interpolation layer, g/cm^3
DENPRT	Density at the port center line
DEPTH	Depth of pool
DGRD	The density gradient used to determine artificial densities outside the pool boundaries
DGRDB	The density gradient from port to bottom; slope equals the density difference over the vertical distance between the port center line and the bottom
DGRDT	The density gradient from port to surface; slope equals the density difference over the vertical distance between the port center line and the surface
DIFF	Absolute difference between locations at which density is to bedetermined and nearest layer midpoint
ELMID	Location of midpoint of the layer containing the location at which density is to be determined
ELTOP	Elevation of midpoint of upper interpolation layer
HGTPRT	Height above bottom of port center line
IJ	Subscript of upper interpolation layer
IJK	0 or 1; used to define interpolation layers
ISURF	Subscript of surface layer
JK	Subscript of lower interpolation layers
LAYER	Layer containing the location at which density is to be determined

(Continued)

Table 2 (Concluded)

Variable	Definition
LAYER1	Assigned the layer number value for layer calculated that lies outside the pool
NUMD	Total number of density values input for the density profile
OLDPRT	Assigned the value of HGTPRT
QEXTR	Logical variable; true when elevation of point is outside of pool; false when elevation of point is inside pool or at boundaries
SIGN	Equals 1, interpolation location is below midpoint of its layer; equals -1, interpolation location is above midpoint of its layer
SLOPE	Change in density between two interpolation layers divided by the vertical distance between the layers
SMALL	Essentially 0, used in check for constant density condition
X	Height above bottom at which density is desired
Y	Assigned the value of X for temporary storage

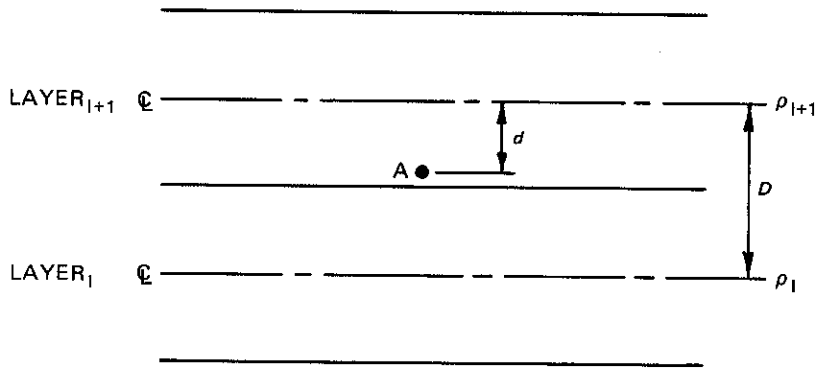


Figure 6. Schematic used to find density at arbitrary elevation A

the port center line and surface to extrapolate the required density value (Figure 7). For an elevation below the bottom, SELECT uses the change in density and difference in elevation between the port center line and the impoundment bottom elevation (Figure 8) to extrapolate the needed density. In actuality, a density value outside the pool is artificial and is necessary only for computations in subroutines VPORT, VWEIR, and SHIFT.

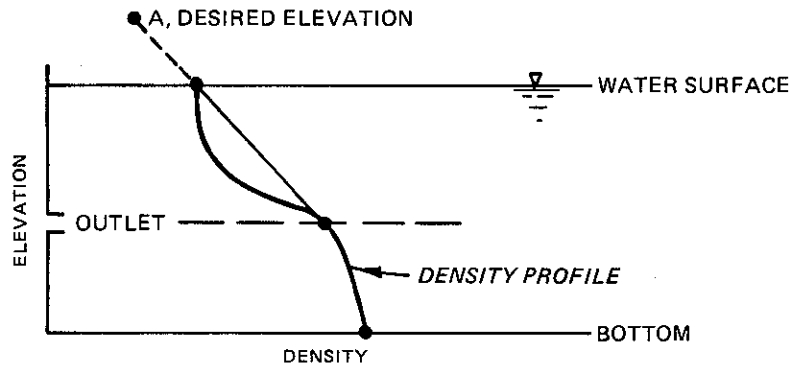


Figure 7. Diagram shows extrapolation of density gradient to determine density at an elevation above the surface

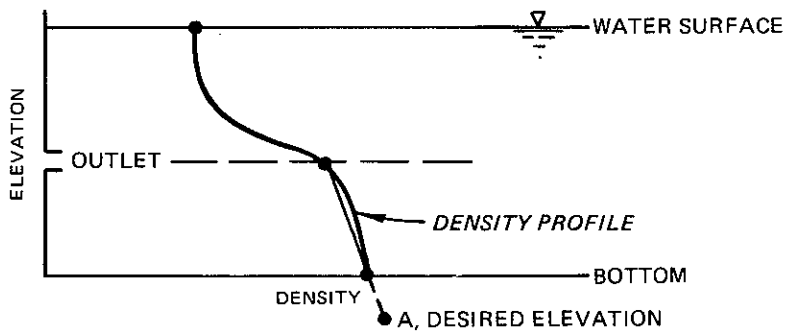


Figure 8. Diagram shows extrapolation of density gradient to determine density at an elevation below the bottom

INTERP

Description

29. Subroutine INTERP generates computational profiles based on user-input data profiles for all parameters to be modeled. INTERP uses linear interpolation of input profile values to determine a value for the center line of each computational layer. The necessity for profile values at the computational layer center lines is for numerical convenience and output organization only. For example, when the density at some arbitrary elevation is desired, the program determines which computational layer the elevation resides in and finds the layer center lines immediately above and below the elevation desired. It then interpolates between the two center-line density values to find the density at the desired elevation. Without the computational layer center-line values, the program would have to scan and compare every input data point elevation with the desired elevation until it found the data values immediately above and below the desired elevation, which is significantly less efficient. Since many of the algorithms in SELECT use profile information and are repeated many times, the use of computational layers is a timesaving necessity.

30. The use of computational layer center-line values may be less accurate than direct use of the actual data points since it results in interpolation between interpolated values; however, the errors incurred by this technique are not significant. Figure 9 is the computational flowchart of INTERP; Table 3 displays the variables used in INTERP.

Special considerations

31. Layer center-line elevations that lie in the pool above the highest input data point elevation are assigned the value of that highest data point. Layer center-line elevations that lie in the pool below the lowest input data point elevation are assigned the value of that lowest input data point.

32. If a layer center-line elevation lies within 0.01 percent of the layer thickness of a data point, the center line will be assigned the value of that data point.

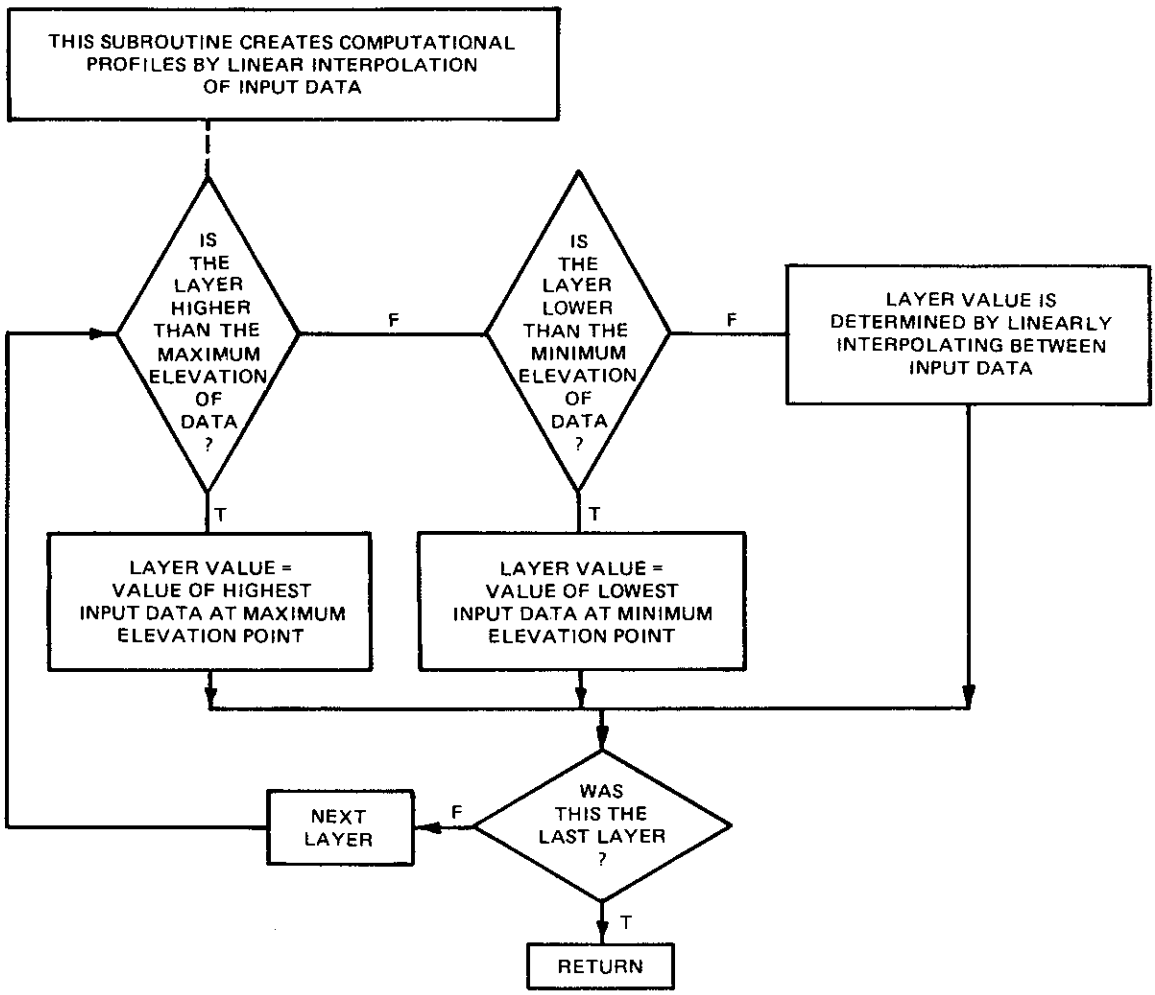


Figure 9. Flowchart of subroutine INTERP

Table 3

INTERP Variables

Variable	Definition
DIFF1	Absolute difference in elevations of present layer midpoint and nearest profile point below it
DIFF2	Absolute difference in elevations of present layer and the nearest profile point above it
DM(I)	Working storage used in resequencing heights into bottom-to-top order
ISURF	Total number of layers
J	Subscript of layer midpoint just above bottom of profile
K	Subscript of layer midpoint just below top of profile
L	Temporarily assigned layer values
NUMV	Number of input profile data points
NV	Number of input profile readings + 1; used in resequencing input profiles into bottom-to-top order
PM(I)	Working storage used in resequencing quality parameter values into bottom or top order
PQUAL(I)	Quality parameter value at midpoint of each layer after profile development in appropriate units
PVALUE(I)	Quality parameter values input as a profile in appropriate units
SIGN1	Variable that keeps track of present lower profile point's position in relation to present layer midpoint
SIGN2	Variable that keeps track of present higher profile point's position in relation to present layer midpoint
SMALL	Essentially zero; check for equality of two values
Y(I)	Height above bottom of the midpoint of each layer
YV(I)	Height above bottom of each input profile parameter

33. If neither of the above cases is true, the program interpolates to find the layer center-line values using the equation

$$q_I = \Delta q * \frac{d}{D} + q_1 \tag{3}$$

where

q_I = parameter value at the layer I center line

$\Delta q = q_2 - q_1$ = difference between parameter input

q_2 = parameter value of the input data immediately above the layer I center line

q_1 = parameter value of the input data point immediately below the layer I center line

d = elevation difference between the layer I center line and the lower input data point

D = elevation difference between the input data points immediately above and below the layer I center line

A schematic representation of this computation is given in Figure 10.

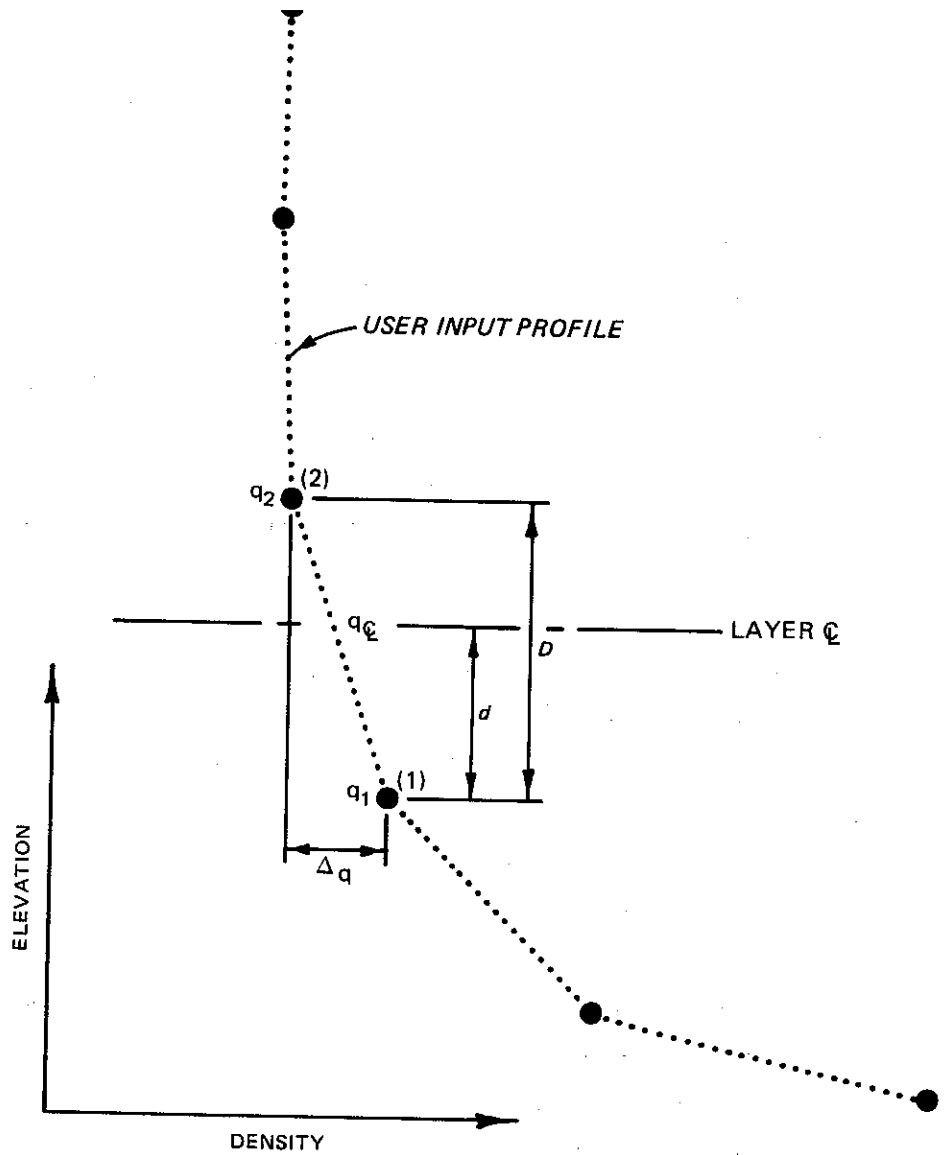


Figure 10. Schematic representation for Equation 3

OUTVEL

34. Subroutine OUTVEL controls the calling sequence of subroutines VWEIR, VPORT, and SHIFT. Based on the information returned from those subroutines, OUTVEL produces a total withdrawal profile and a total normalized velocity profile. OUTVEL also calculates the outflow density, temperature, and water quality concentrations as specified by the user. Figure 11 shows the algorithm flowchart. Table 4 lists descriptions of the subroutine variables.

Withdrawal profile computation

35. The withdrawal profile computed by VPORT or VWEIR is the profile for a port or weir, respectively. The profile values are given at the center-line elevations of the computational layers. When several outlets are modeled, each outlet generates an individual profile that contributes to the total withdrawal from each layer. The total withdrawal profile is found by summing the outflow values assigned to each layer resulting from flow through each outlet (Figure 12). Before this summation is carried out, however, subroutine SHIFT is executed to adjust the withdrawal limits of overlapping withdrawal zones, such as those shown in Figure 12. This accounts for reductions in shear resistance in the overlapping region, and the withdrawal profiles are adjusted accordingly.

Normalized velocity profile computation outflow parameters

36. The total normalized velocity distribution is developed from the total withdrawal distribution. The withdrawal from each layer is divided by the maximum layer withdrawal in the withdrawal zone. Thus, the layer of maximum discharge is assigned a normalized velocity of 1.0 and all others will have a velocity less than 1.0.

Outflow parameters

37. The outflow parameter values (density, temperature, and water qualities) are determined by a flow-weighted averaging technique using the equation

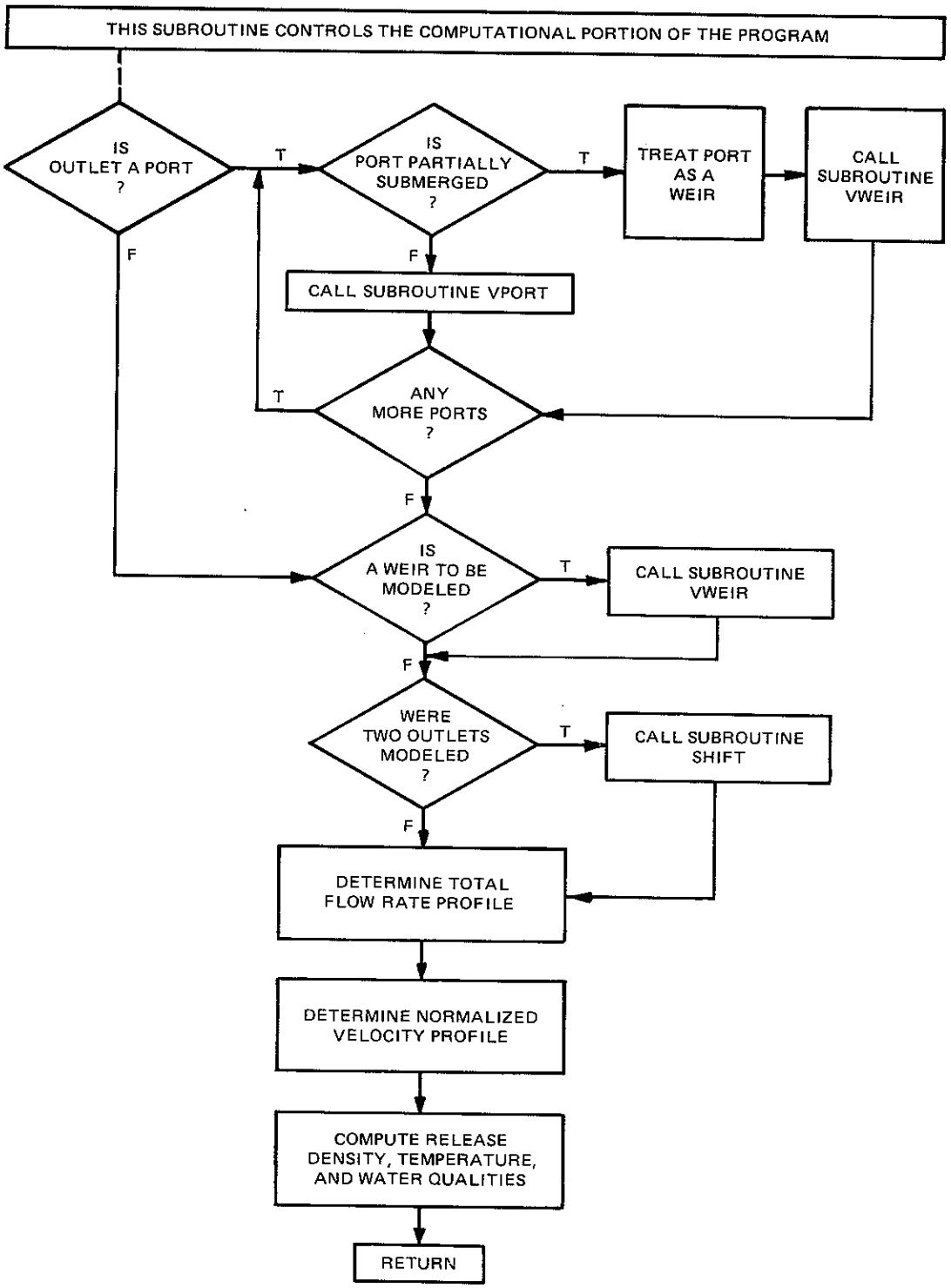


Figure 11. Flowchart for subroutine OUTVEL

Table 4

OUTVEL Variables

Variable	Definition
CREST	Weir crest height above bottom
DEN(I)	Density of layer I, g/cm^3
DENOUT	Release density
DENPRT	Density at port center line
DEPTH	Depth of pool
FLODEP	Depth of water between the top of the port and water surface; used in empirical determination of partial port submergence
FLORAT	Flow through one outlet
FLOW(I)	Release flow rate through I^{th} port
HDIM	Assigned value of PHDIM(K) locally
HGT(I)	Percentage of layer I that is filled with water
HGTLOW	Height above bottom of lower withdrawal limit
HGTPRT	Height above bottom of the port center line
HGTTOP	Height above bottom of upper withdrawal limit
ISURF	Total number of layers
LENGTH	Length of weir crest
LLOW(K)	Layer of pool where lower withdrawal limit for outlet K is located
LOWLIM	Layer of pool where lower withdrawal limit for one outlet is located
LTOP(K)	Layer of pool where upper withdrawal limit for outlet K is located
NOUTS	Number of selective withdrawal outlets

(Continued)

(Sheet 1 of 3)

Table 4 (Continued)

Variable	Definition
NPORTS	Number of selective withdrawal ports
NQUAL	Number of quality parameters input
PHDIM(K)	Horizontal dimension of port K
PHGT(K)	Height above bottom of port K center line
PVDIM(K)	Vertical dimension of port K
QALOUT(N)	Average release concentration of Nth water quality parameter, appropriate units
QPORT	Logical variable; true, ports are present as outlet devices; false, no ports
QPWEIR	Logical variable; true when a port is considered "partially submerged" and therefore modeled as a weir; false when the above is false
QSUB	Logical variable; true when weir is submerged; false when weir is <u>not</u> submerged
QTEMP	Logical variable; true for temperature profile input; false for no temperature profile
QUAL(J,I)	Value of the Jth quality parameter for layer I
QWEIR	Logical variable; true when a weir is present as an outlet device; false when there is no weir
SUMDF	Sum over layers of the product of withdrawal flow and density
SUMOUT	Total flow rate through all outlets
SUMQF	Sum over layers of the product of withdrawal flow and quality
SUMTF	Sum over layers of the product of withdrawal flow and temperature
TEMOUT	Average temperature of outflow, degrees Fahrenheit or Centigrade

(Continued)

(Sheet 2 of 3)

Table 4 (Concluded)

Variable	Definition
TEMP(K)	Temperature value at layer K, degrees Fahrenheit or Centigrade
TOPLIM	Layer of pool where upper withdrawal limit for a given outlet is located
V(I)	Velocity profile value at layer I for flow through one given outlet
VDIM	Assigned value of PVDIM(K) locally
VEL(I)	Total velocity profile value of layer I for all outlets
VHL	Empirical value in the determination of partial submergence of a port
VMAX	Maximum normalized velocity equal to 1.0
VS(I,K)	Velocity profile value of layer I for the Kth outlet
VW	Empirical value used in the determination of partial submergence of a port
WANGLE	Assigned the value WTHETA(K) locally
WRHGT	Weir crest height above bottom
WRLNG	Weir length
WTHDRW(I)	Withdrawal flow rate from layer I
WTHETA(K)	Withdrawal angle for port K
ZDN(K)	Height above bottom of the lower withdrawal limit of outlet K
ZUP(K)	Height above bottom of the upper withdrawal limit of outlet K

(Sheet 3 of 3)

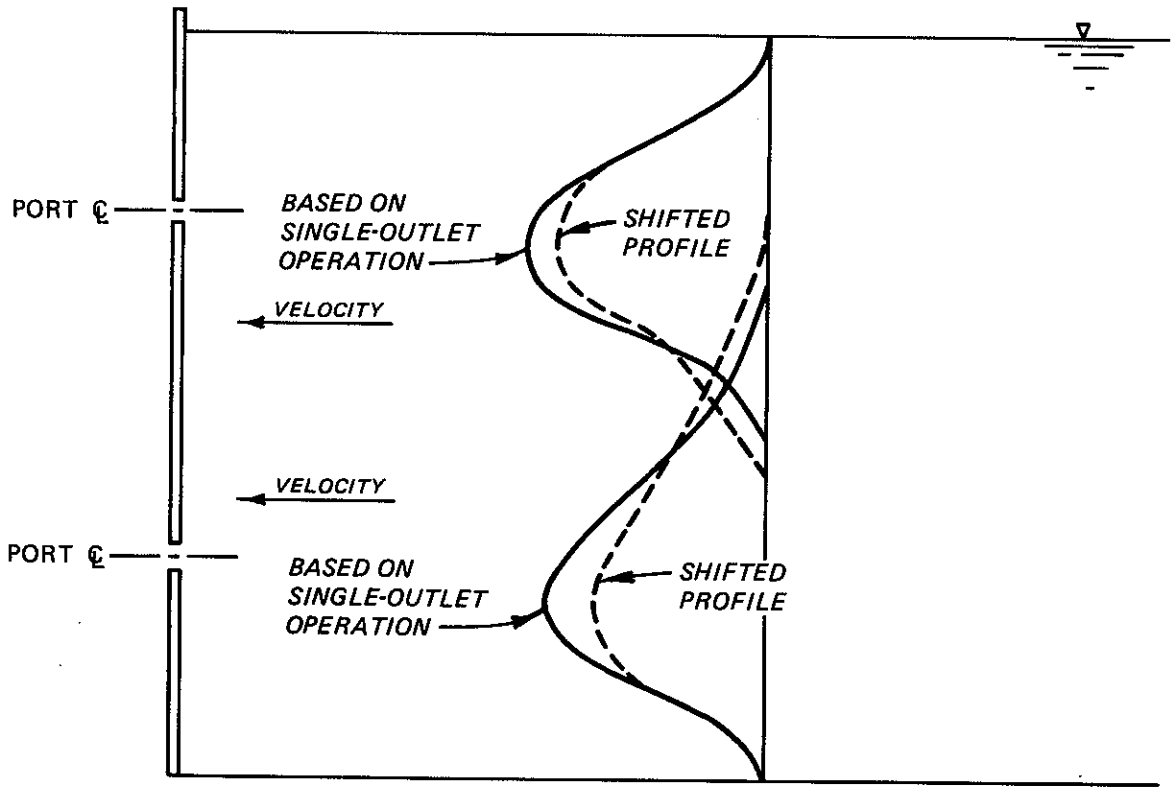


Figure 12. Schematic representation of overlapping withdrawal zones. The profiles are shifted (dashed lines) to better approximate actual profile

$$C_{TOT} = \frac{\sum(C_I * q_I)}{\Sigma q_I} \tag{4}$$

where

C_{TOT} = total outflow value or concentration

C_I = value or concentration of parameter for layer I

q_I = withdrawal from layer I

The concentrations or values of parameters in each layer are multiplied by the relative contribution of total flow from that layer and divided by the total flow rate from all outlets. Then, these flow-weighted parameter values for each layer are summed, yielding the total outflow parameter value. The outflow parameter calculations are performed for density, temperature, and quality constituents.

VPORT

38. Subroutine VPORT is the primary computational subprogram called by SELECT when port devices are modeled. VPORT determines the withdrawal zone limits and the associated flow rate profiles for a given port based on input data. The reader is reminded that subroutine OUTVEL performs tasks for the total release; VPORT performs its computations for only a single port per call. Figure 13 shows the computational flowchart for subroutine VPORT. Table 5 lists descriptions of the variables used by VPORT.

39. The following description of VPORT is presented in four parts: the determination of the withdrawal zone limits; the calculation of normalized velocities in the withdrawal zone; the generation of the profile; and the "point sink" calculation to verify that a point sink assumption is valid for the port being analyzed.

Withdrawal zone limits

40. The calculation of withdrawal zone limits is based on densimetric Froude number formulations developed by Bohan and Grace (1973) and Smith et al. (1987). The Bohan and Grace equation (modified for the withdrawal angle concept derived by Smith et al.) is used in three cases. One case is for no interference of the withdrawal zone with a boundary; the second is for simultaneous surface and bottom interference of the withdrawal zone; and the third is for determining the theoretical limit of a limit that experiences interference, while the other limit is free of interference.

41. The Smith et al. (1987) equation is used to find the limit that is free of interference when the other limit experiences interference. Boundary interference is defined to exist when the surface or bottom boundaries of the impoundment lie within the theoretical limits of the withdrawal zone, e.g., the limit cannot form freely within the pool because the impoundment surface or bottom interferes. The determination of limits is the same for no interference and for interference with both boundaries. The difference in the two conditions is not important for the calculation of the limits, but is important in

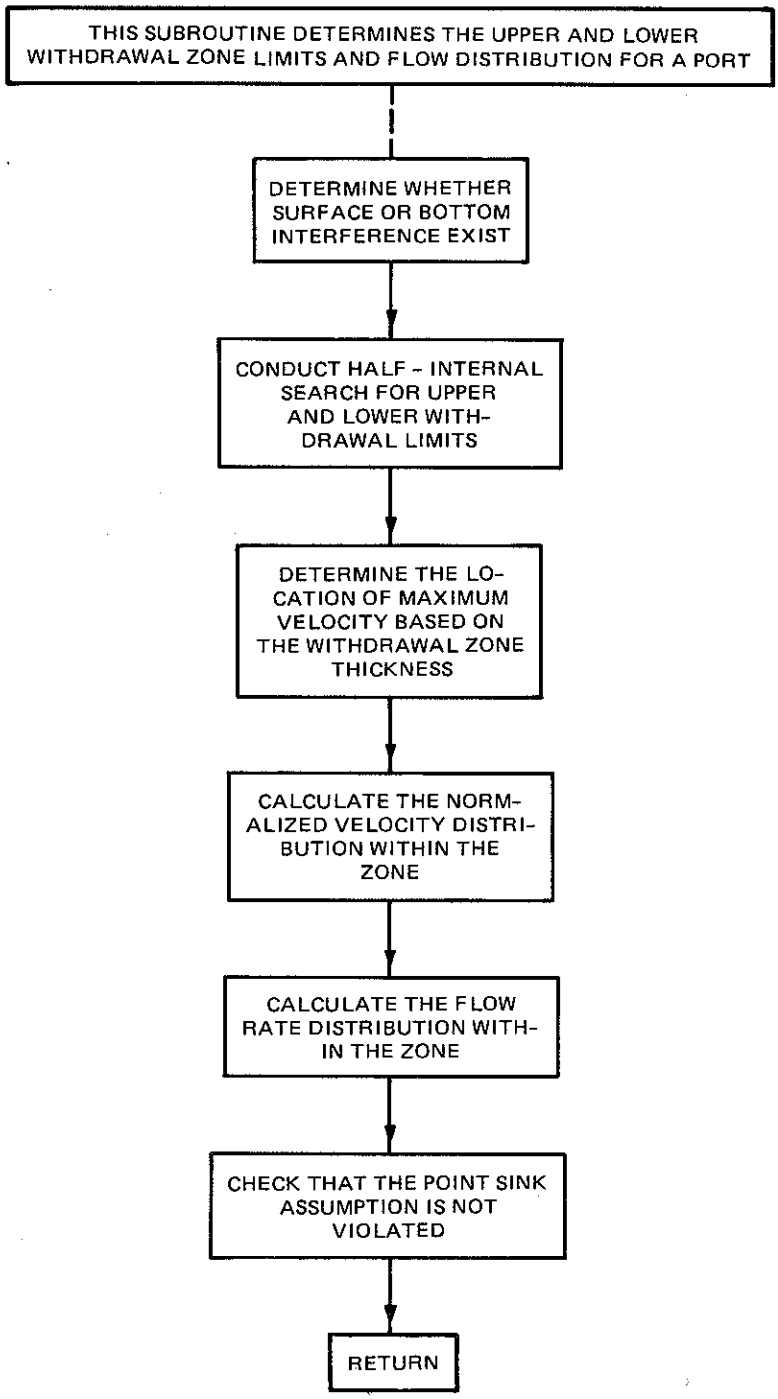


Figure 13. Flowchart for subroutine VPORT

Table 5

VPORT Variables

Variable	Definition
BONLIM	Assigned the value of zero for bottom interference or DEPTH for surface interference
C	Assigned the value of WANGLE
DELDEN	Density difference from layer of maximum velocity to local elevation, g/cm^3
DELZ	Layer thickness
DEN(I)	Density of layer I, g/cm^3
DENBOT	Density at the bottom of the impoundment
DENDIF	Density difference between fluid at layer of maximum velocity and a particular withdrawal limit, g/cm^3
DENLIM	Assigned value of DENUPP for surface interference or DENBOT for bottom interference
DENLOW	Density at lower withdrawal limit, g/cm^3
DENPRT	Density at port center line, g/cm^3
DENTOP	Density at upper withdrawal limit, g/cm^3
DENUPP	Density at surface of the impoundment
DEPTH	Depth of pool
DRBLIM	Density difference between fluid at the port center line and fluid at the port bottom
DRPBOT	Density difference between fluid at the port center line and the port invert
DRPTOP	Density difference between fluid at the port center line and the port top
DRTLIM	Density difference between fluid at the port center line and fluid at the port top
DVMAX	Density at location of maximum velocity, g/cm^3
F1	Value of withdrawal limit function FPORT(X) evaluated at XI
F2	Same as F1 evaluated at X2
F3	Same as F1 evaluated at X3
FLORAT	Flow rate through a port
G	Gravitational acceleration

(Continued)

(Sheet 1 of 4)

Table 5 (Continued)

Variable	Definition
H1	Working storage for result of function HPORT(X) evaluated at X1
H2	Same as H1 evaluated at X2
H3	Same as H1 evaluated at X3
HGT(I)	Percentage of layer I that is filled with water
HGTLOW	Height above bottom of lower limit
HGTPRT	Height above bottom of port center line
HGTTOP	Height above bottom of upper limit
ISURF	Total number of layers
LOWLIM	Layer of lower limit
LVMAX	Layer of maximum velocity
MAX	Number of search iterations allowed to determine withdrawal limits
PI	Assigned the value of $\pi = 3.14159$
PRTBOT	Height of the invert of port
PRTTOP	Height of the upper edge of port
QBLIM	Logical variable; true for bottom withdrawal interference; false for no interference from bottom
QSHIFT	Logical variable; true when VPORT is called from SHIFT; false when call is not from SHIFT
QSINK1	Logical variable; true when point sink description is adequate for determination of lower withdrawal limit; false when the above is not true
QSINK2	Same as QSINK1, except check is for the upper withdrawal limit
QTLIM	Logical variable; true for surface withdrawal interference; false for no interference from surface
RATIO	Ratio of the product of distance and density difference between the point of maximum velocity and a local point to the product of distance and density difference between the point of maximum velocity and a given limit

(Continued)

(Sheet 2 of 4)

Table 5 (Continued)

Variable	Definition
SINK1	Empirical value used in determination of validity of point sink description for calculations of lower withdrawal limit
SINK2	Same as SINK1, except now pertaining to upper withdrawal limit
SMALL	Essentially zero; check for approximate equality between two values
SUBR	Subroutine name
SUM	Sum over layers in the withdrawal zone of the velocity values for each layer
TINY	Essentially zero; used in value comparisons
TOPLIM	Layer of upper limit
TRUNCZ	For surface interference, it is the distance between the port center line and the surface; for bottom interference, it is the distance between the bottom and the port center line
V(I)	Normalized velocity profile value at layer I for a given port
VDIM	Assigned as value of PVDIM(K) locally
VDIM2	One-half the vertical dimension of the port; VDIM/2
VD2	Equals VDIM2 or, if upper edge of port is above the pool surface, the distance between the surface and the port center line
VM	Scaling factor
VMAX	Maximum velocity in the normalized velocity profile; assigned as 1.0
WANGLE	Withdrawal angle; equals WTHETA(K) from subroutine OUTVEL
X1	Elevation of a search limit
X2	A second limit search elevation
X3	A third limit search elevation
X4	A fourth limit search elevation
XDUMY	Assigned 0.0; used in ERROR () argument list
XDUMY1	Same as XDUMY
XDUMY2	Same as XDUMY
XDUMY3	Same as XDUMY

(Continued)

(Sheet 3 of 4)

Table 5 (Concluded)

Variable	Definition
XVMAX	Location of maximum velocity relative to the bottom
XXX	Used in label assignment statement
Y	Distance from elevation of maximum velocity to local elevation
YVMAX	Location of maximum velocity referenced to lower withdrawal limit
ZLOW	Distance between port center line and lower withdrawal limit
ZONE	Distance from lower withdrawal limit to upper withdrawal limit
ZONED	When surface or bottom interference exists (but not both), it equals the distance between the boundary of interference and the opposing withdrawal limit
ZTOP	Distance between port center line and upper limit

(Sheet 4 of 4)

the calculation of the velocities within the withdrawal zone.

42. The Bohan and Grace equation, modified to include the withdrawal angle, is

$$\frac{Q}{Z^3 N} = \frac{\theta}{\Pi} \tag{5}$$

where

$$N = \sqrt{\frac{\Delta\rho}{\rho}\frac{g}{Z}} \tag{6}$$

Q = flow rate

Z = distance between the port center line and the upper or lower withdrawal limit

θ = angle of withdrawal, in radians

$\Delta\rho$ = difference between the density at the upper or lower limit and the density at the port center line

ρ = density at the port center line

g = acceleration due to gravity 32.2

Figure 14 shows a schematic definition of these variables. Equation 5 is transcendental and cannot be solved directly since $\Delta\rho$ is a function of Z for the computation of N (Equation 6). Therefore, an iterative technique is needed to solve the equation for Z. Since Equation 5 and the Smith et al. (1987) equation, which follows, are transcendental and solved through iteration, a description of the iterative solution algorithm is withheld until after the Smith et al. equation is presented.

43. Smith et al. (1987) developed an equation that is an analytical extension of Equation 5 and is used to locate the limit that is free of interference when the other limit experiences boundary interference (Figure 15). The equation is

$$\frac{Q'}{D'^3 N} = \frac{0.125\phi}{X^3}\frac{\theta}{\Pi} \tag{7}$$

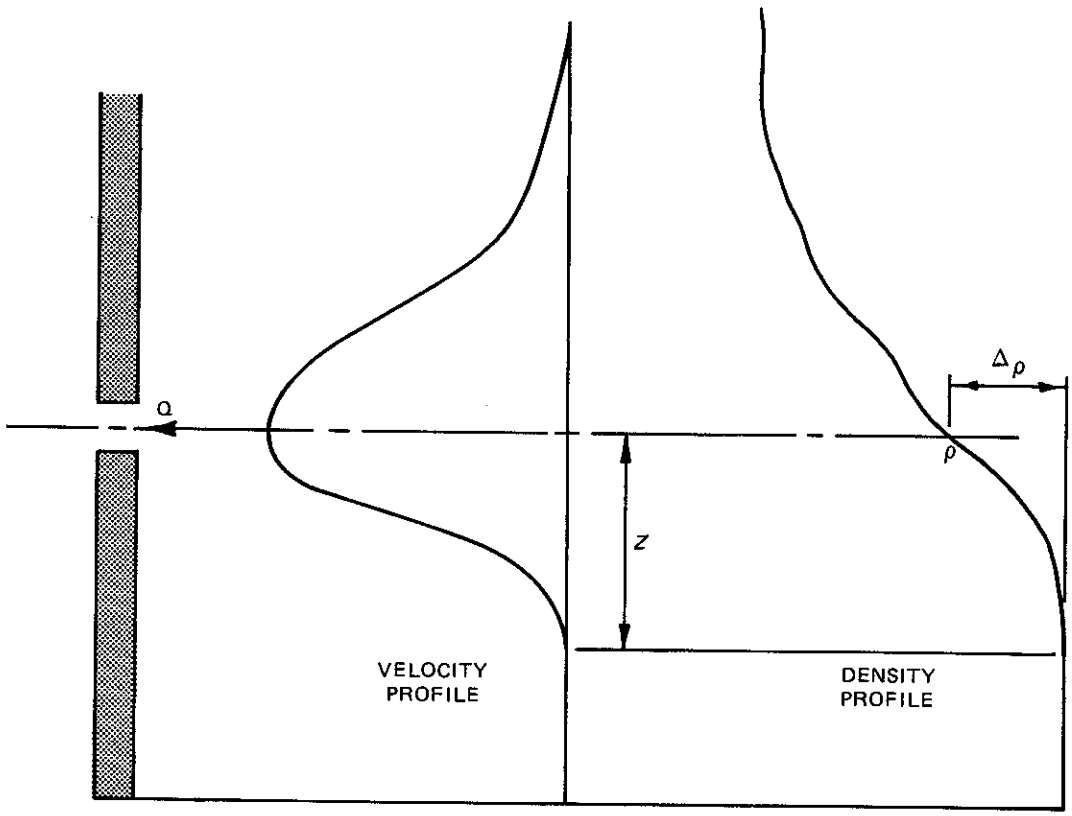


Figure 14. Variable definition schematic

where

$$\phi = \frac{1}{2}\left[1 + \frac{1}{\Pi}\sin\left(\frac{b/D'}{1 - b/D'}\Pi\right) + \frac{b/D'}{1 - b/D'}\right] \tag{8}$$

$$X = \frac{1}{2}\left[1 + \frac{b/D'}{1 - b/D'}\right] \tag{9}$$

where

Q' = discharge flow rate from the withdrawal zone

D' = distance between free withdrawal limit and boundary of interference

b = distance between port center line and boundary of interference

A schematic representation of this computation is given in Figure 15. Equation 7 is iteratively solved for D' . With D' known, the

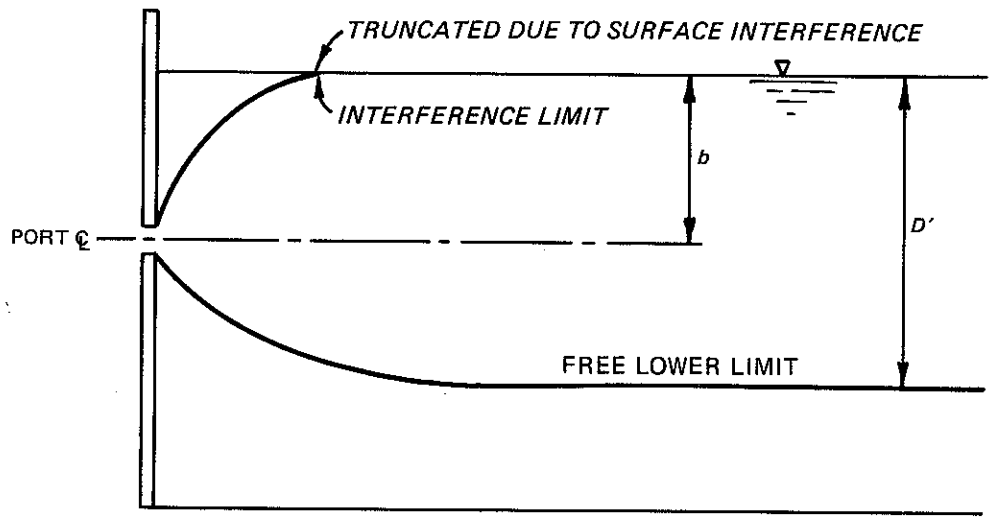


Figure 15. Schematic of a withdrawal zone with surface interference

location of the free limit (a distance D' from the limit of interference) is computed.

44. As mentioned earlier, Equations 5 and 7 are transcendental and therefore require an iterative solution. A half-interval search algorithm is used in SELECT and is discussed in the following paragraphs.

Half-interval search

45. Equations 5 and 7 are rearranged to give

$$Q - Z^3 N \frac{\theta}{\Pi} = 0 \tag{10}$$

$$Q' - (D'^3 N) \left[\frac{1 + \sin \frac{b/D'}{1 - b/D'} \Pi + \frac{b/D'}{1 - b/D'}}{\left(1 + \frac{b/D'}{1 - b/D'}\right)^3} \right] \frac{\theta}{2\Pi} = 0 \tag{11}$$

These equations are then used to iteratively solve for Z (Equation 10) and D' (Equation 11). Note that if, during the iteration process, an appropriate Z or D' is chosen and substituted into Equation 10 or 11, respectively, the above equalities become true and the assumed Z or D' would be considered the solution to the equation. If a smaller or larger value is chosen, the left-hand side (LHS) of the equalities

will be positive or negative, respectively. By systematically evaluating the magnitude and the sign of the LHS for various values of Z or D' , convergence to the solution can be achieved.

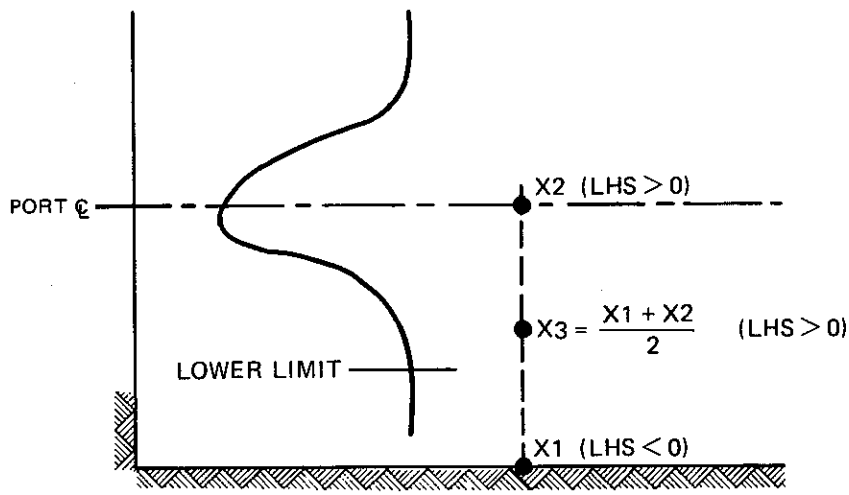
46. For the sake of illustration, the following description of the search for a lower limit in a withdrawal zone free of interference is given. We are solving Equation 10. Note that the half-interval search determines one limit at a time.

a. To begin the iteration process, two points are located: one point is assumed to lie above the limit (usually the outlet center-line elevation X2 = 0.0 since X2 is measured from the port center line downward); the other point is below the limit (a distance X1 below the port center line, usually at the bottom). Figure 16a illustrates this step.

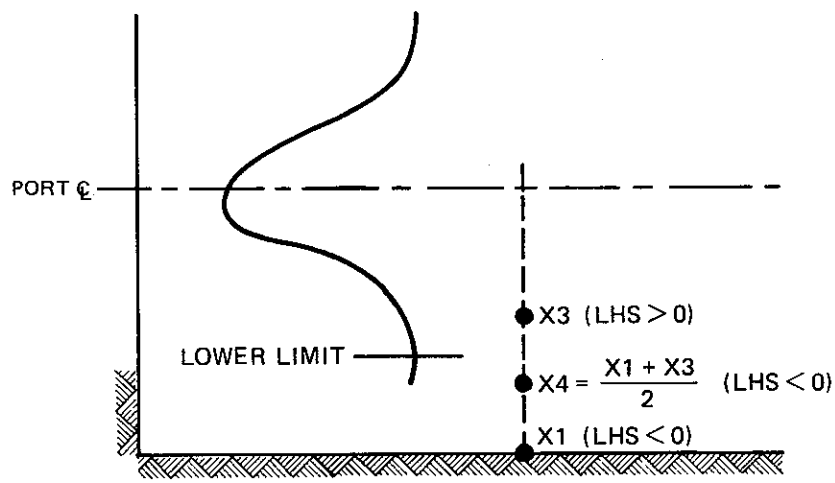
b. A third point (X3) is determined that is halfway between X1 and X2 , hence, the half-interval search technique. Point X3 is assumed to be the withdrawal limit, and its Z value and corresponding density difference are substituted into Equation 10. If the LHS is zero or within a tolerable deviation from zero (10 percent of the layer thickness), X3 is taken as the actual limit. If the LHS of Equation 10 is negative, then X3 is below the actual limit. If the LHS is positive, X3 is above the actual limit.

c. If X3 is not the solution after the first iteration is complete, then X3 replaces the limit (X1 or X2) that lies on the same side of the limit as it does. Thus, a new search region, which is the half of the original search region containing the solution, will be subjected to the same search technique. A new midpoint (X4) is found halfway between X3 and the remaining point (X1 or X2), at a quarter-point of the original search region. In this example, X4 is located on the opposite side of the limit from X3 (Figure 16b). It should be noted that X4 does not always reside on the opposite side of the limit from X3 .

d. The new point, X4, is assumed to be the withdrawal limit, and its Z value and corresponding density difference are substituted into Equation 10. The process described in c repeats itself until successive estimates for Z (X3 and X4) are found that lie within a tolerable distance from each other (10 percent of the layer thickness). The X3 or X4 value is then assigned as the actual limit.



a. Initial estimate of lower limit (X3)



b. Second estimate of lower limit (X4)

Figure 16. Illustration of search for lower withdrawal limit

e. If the search does not converge within 10 iterations, an error message is output and program execution is terminated.

47. The following points should also be noted:

a. When one limit experiences interference and the other does not, Equation 11 is used rather than Equation 10 in the search algorithm and the number of iterations allowed is increased from 10 to 20.

b. Although boundary interference may physically constrain a limit to the given boundary, a theoretical value for that limit is determined assuming no interference. This allows the computation of the normalized velocity profile based upon the theoretical extent of the limits, thereby expressing the velocity at the boundary in terms of its theoretical potential. The zone is truncated at the boundary, and all velocity values outside the boundary are omitted (Figure 17) in subsequent computations.

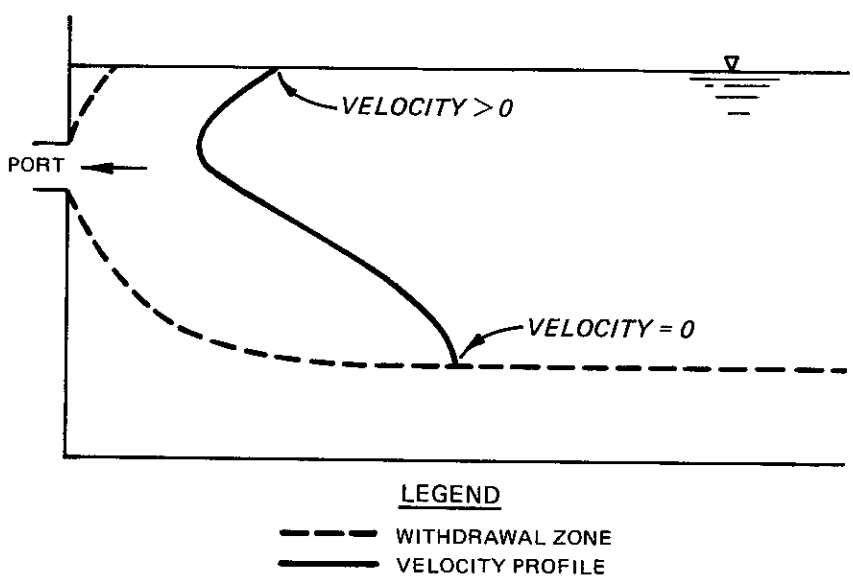


Figure 17. Velocities calculated above the water surface are truncated from velocity profile

Velocity profile computation

48. The velocity profile computations are based on the vertical location of the maximum velocity in the withdrawal zone. The location of maximum velocity is given by the equation (Bohan and Grace 1973)

$$Y_L = H * \sin\left(1.57 \frac{Z_L}{H}\right)^2 \tag{12}$$

where

Y_L = distance from lower limit to elevation of maximum velocity

H = vertical distance between the upper and lower withdrawal limits

Z_L = vertical distance between the outlet center line and the lower limit

Since the elevation of the lower limit is known, the elevation (and the layer) of maximum velocity can be determined. With the location of the layer of maximum velocity known, the velocity distribution can be determined.

49. Bohan and Grace (1973) found that Equations 13 and 14 described the vertical velocity distribution for withdrawal zones experiencing boundary interference and for zones free of interference, respectively.

$$\frac{V}{V_{MAX}} = 1 - \left(\frac{y}{Y} \frac{\Delta\rho}{\Delta\rho_{MAX}}\right)^2 \tag{13}$$

$$\frac{V}{V_{MAX}} = \left(1 - \frac{y}{Y} \frac{\Delta\rho}{\Delta\rho_{MAX}}\right)^2 \tag{14}$$

where

V = velocity value within a given layer of the withdrawal zone

V_{MAX} = maximum layer velocity value in the withdrawal zone

y = vertical distance from the elevation of maximum velocity to that of the given layer

Y = vertical distance between the elevation of maximum velocity and that of the upper or lower limit as appropriate

$\Delta\rho$ = difference in density between that at the elevation of maximum velocity and that at the given layer

$\Delta\rho_{MAX}$ = difference in density between that at the elevation of maximum velocity and that at the elevation of the upper or lower limit as appropriate

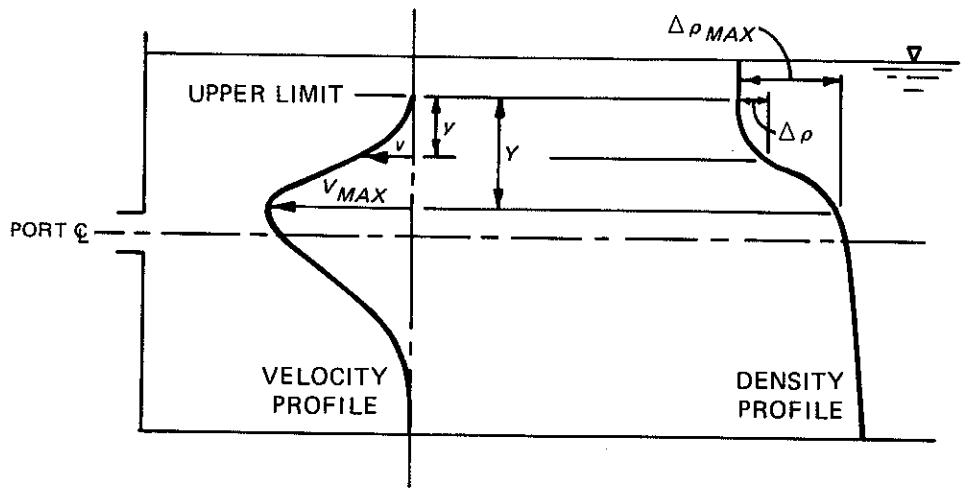


Figure 18. Schematic definition of variables for Equations 13 and 14

A definition sketch for these variables is given in Figure 18. Inspection of Equations 13 or 14 reveals that the product $(y/Y)(\Delta\rho/\Delta\rho_{MAX})$ has a maximum value of 1.0. Therefore, the maximum value of the right-hand side of this equation is 0.0. Since these velocity values will ultimately be scaled to provide a withdrawal profile (shown later), the assignment of V_{MAX} acts only to change the constant used for scaling these profiles. In order to develop the normalized velocity distribution (velocities with values between 0 and 1), V_{MAX} is assigned in VPORT to be 1.0. Thus, the V computed in Equations 13 and 14 is also a normalized velocity. The computation of velocity for each layer is then a straightforward calculation since the variables y, Y, $\Delta\rho_{MAX}$, and $\Delta\rho$ are known for each layer. The computation of the entire velocity profile is accomplished by individual operations for each portion of the withdrawal zone above and below the elevation of the point of maximum velocity.

Special considerations

50. The maximum velocity (1.0) will be assigned to all withdrawal layers above or below the elevation of maximum velocity if the layer densities do not vary from the density at the elevation of maximum velocity, i.e., $\Delta\rho = 0$.

Conversion of
velocities to flow rates

51. Once the normalized velocity distribution is determined for a port, the withdrawal profile, i.e., a flow rate, is calculated for each layer based on the total flow rate through the port. The equation used to compute the withdrawal profile is

$$q_I = \frac{V_I}{\Sigma V_I} Q_T \tag{15}$$

where

q_I = withdrawal from the Ith layer

V_I = normalized velocity at the Ith layer

Q_T = total flow rate through a port

Point sink assumption

52. One of the basic assumptions in the theory underlying SELECT is that ports may be considered point sinks, i.e., port dimensions are insignificant compared to withdrawal zone thickness. To ensure that the outlet configuration is consistent with the point sink assumptions, the program performs empirically based calculations (see next paragraph) to assess whether the point sink assumption is violated. If the assumption has been violated, a warning statement is issued in the output alerting the user to the violation of the assumption. Should the user receive such a statement, more extensive modeling may be required.

53. Using the illustration in Figure 19 to aid in the description of the variables, the following inequalities are defined to ascertain the applicability of the point sink assumption:

$$\frac{(\Delta\rho_L * Z_L)}{(\Delta\rho_B * h)} > 3.0 \tag{16}$$

$$\frac{(\Delta\rho_U * Z_U)}{(\Delta\rho_T * h)} > 3.0 \tag{17}$$

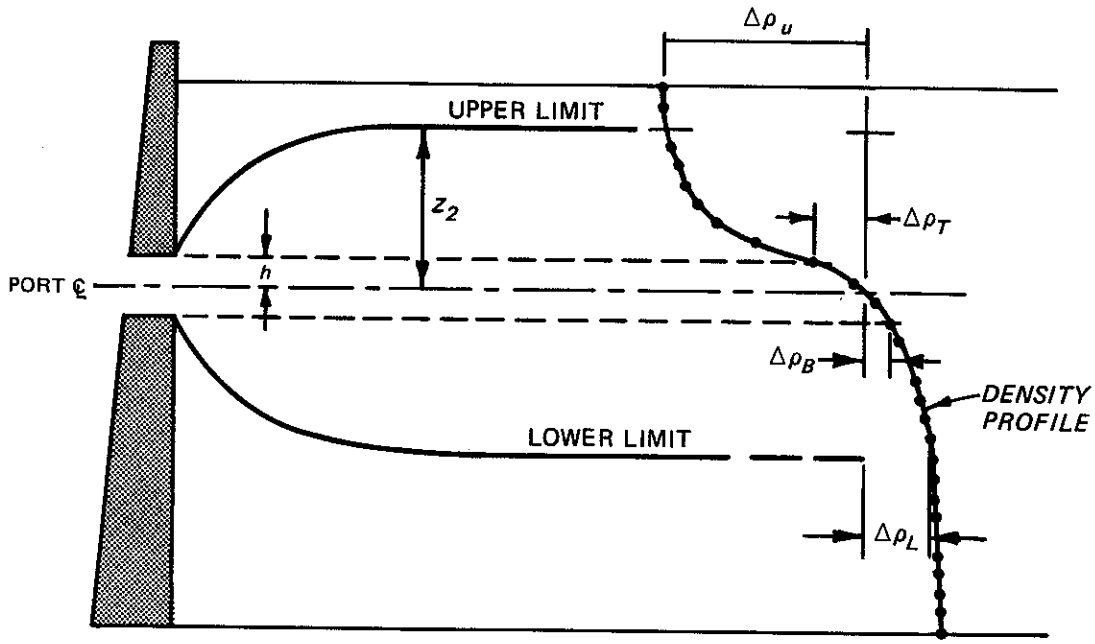


Figure 19. Definition sketch of variables for point sink assumptions

where

$\Delta\rho_L$ = density difference between the port center line and the predicted lower limit

Z_L = distance from the port center line to the lower limit

$\Delta\rho_B$ = density difference between the port center line and the invert of the port

h = one-half the vertical dimension of the port

$\Delta\rho_U$ = density difference between the port center line and the predicted upper limit

Z_u = distance from the port center line and the upper limit

$\Delta\rho_T$ = density difference between the port center line and the top of the port

When Equation 16 is true, a point sink description of withdrawal through the port is not applicable for determining the lower limit. When it is false, a point sink description is valid. When Equation 17 is true, a point sink description of withdrawal through the port is not valid for the upper limit. When it is false, a point sink description is valid.

VWEIR

54. Subroutine VWEIR is the computational subprogram called by OUTVEL when an outlet is modeled as a weir. VWEIR determines the withdrawal zone limits and associated profile for withdrawal over a weir under the assumption that the weir crest is above the thermocline. If this assumption is violated, SELECT may produce erroneous output. It should be noted that the impoundment surface is taken as the upper limit for any type of weir modeled. Also, if the bottom interferes with the lower limit, the bottom is taken as the lower limit.

55. The equation used in VWEIR to calculate the withdrawal zone limit is based on studies and analysis performed by Grace (1971) and additional analytical and empirical development at the WES. Subroutine VWEIR incorporates a formulation based on the densimetric Froude number F_D to describe weir withdrawal such that

$$F_D = \frac{\bar{V}H_w}{\sqrt{\frac{\Delta\rho}{\rho} g(Z + H_w)^3}} = C - D \frac{H_w}{(Z + H_w)} \tag{18}$$

where

$\bar{V}$ = average velocity in the withdrawal zone

H_w = head above the weir crest elevation

$\Delta\rho$ = difference in density of fluid at the weir crest and the lower limit

ρ = density at the weir crest elevation

g = acceleration due to gravity

Z = distance between the crest elevation and the lower withdrawal limit

$$C = 0.54 \quad \text{and} \quad D = 0 \quad \text{for} \quad \frac{Z + H_w}{H_w} \geq 2.0$$

$$C = 0.78 \quad \text{and} \quad D = 0.70 \quad \text{for} \quad \frac{Z + H_w}{H_w} < 2.0$$

56. Rearranging Equation 18 yields

$$0 = \overline{V} - C \frac{(Z + H_w)}{H_w} \sqrt{\frac{\Delta\rho}{\rho} g(Z + H_w)} + D \sqrt{\frac{\Delta\rho}{\rho} g(Z + H_w)} \qquad (19)$$

With Equation 19, it is possible to solve for the withdrawal limit Z by iteration. The half-interval search, outlined in VPORT, is used in a similar manner in VWEIR except that Equation 19 is used as the iterative equation. Figure 20 shows the computational flowchart of VWEIR. Table 6 lists descriptions of the variables used in subroutine VWEIR.

Velocity profile

57. The calculation of a normalized velocity profile for a weir is based on the equations developed by Bohan and Grace (1973). Two conditions influence which equation governs the description of the profile: (a) whether the portion above or below the elevation of maximum velocity is being described, and (b) whether the weir is free or submerged. The following listing accounts for the conditions under which the equations are used.

58. Submerged weir. The location of the point of maximum velocity is computed exactly as it is for an orifice, as described in paragraph 51 of the subroutine VPORT description. The portion of the profile above the elevation of maximum velocity is described by

$$\frac{V_2}{V_{MAX}} = 1 - \left(\frac{y_2 \Delta\rho_2}{Y_2 \Delta\rho_{2m}}\right)^2 \qquad (20)$$

where

V_2 = layer velocity in the zone of withdrawal at a distance y_2 above the elevation of maximum velocity

V_{MAX} = maximum layer velocity in the withdrawal zone, equals 1.0 for normalized distribution (see discussion of V_{MAX} in section on VPORT)

y_2 = vertical distance from the elevation of maximum velocity to that of the corresponding layer velocity V_2

$\Delta\rho_2$ = density difference of fluid between the elevations of the maximum velocity and the corresponding layer velocity V_2

Y_2 = vertical distance from the elevation of the maximum velocity to the upper limit of the zone of withdrawal

$\Delta\rho_{2m}$ = density difference of fluid between the elevation of the maximum velocity and the elevation of the upper limit

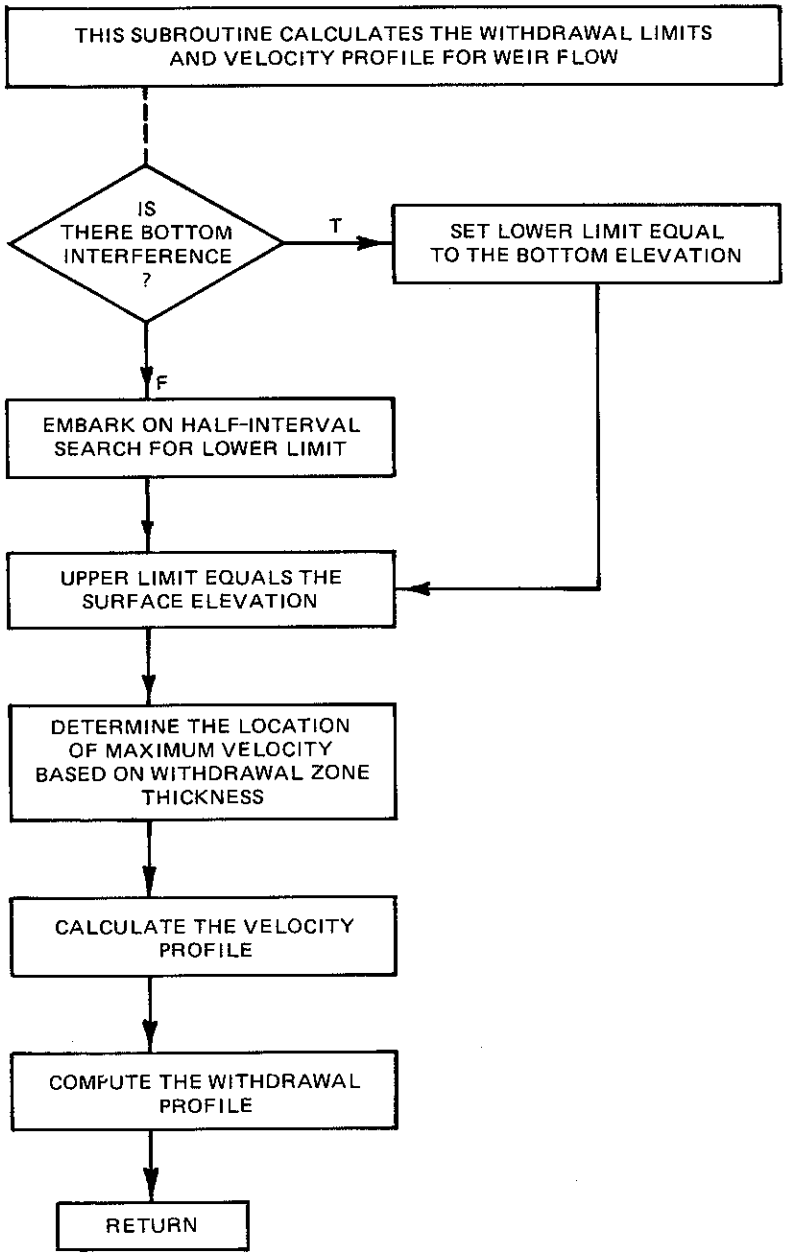


Figure 20. Flowchart for subroutine VWEIR

Table 6

VWEIR Variables

Variable	Definition
A	Coefficient used in determining the velocity profile exponent as a function of the discharge coefficient
AVGVEL	Average velocity over weir
B	Coefficient used in determining the velocity profile exponent as a function of discharge coefficient
C	Coefficient used in the FWEIR(X) function; has values of 0.54 or 0.78
COEF	Weir discharge coefficient
CREST	Weir crest height above bottom
D	Second coefficient used as C; possible values are 0.0 to 0.70
DELDEN	Density difference between fluid at layer of maximum velocity and fluid at a local elevation
DELZ	Layer thickness
DEN(I)	Density of layer I, g/cm^3
DENDIF	Density difference between fluid at layer of maximum velocity and fluid at a withdrawal limit, g/cm^3
DENLOW	Density at lower withdrawal limit, g/cm^3
DENTOP	Density at upper withdrawal limit, g/cm^3
DEPTH	Depth of pool
DVMAX	Density at location of maximum velocity, g/cm^3
EXPNT	Exponent for velocity profile equation for free weir flow
F1	Value of withdrawal limit function FWEIR(X) evaluated at X1
F2	Same as F1 evaluated at X2
F3	Same as F1 evaluated at X3

(Continued)

(Sheet 1 of 4)

Table 6 (Continued)

Variable	Definition
FLORAT	Flow rate over weir
G	Gravitational acceleration
HEAD	Head above weir crest
HGT(I)	Percentage of layer I that is filled with water
HGTLOW	Height above bottom of lower limit
HGTTOP	Height above bottom of upper limit
ISURF	Total number of layers
ITMAX	Number of search iterations allowed to determine withdrawal limits
LENGTH	Length of weir crest
LOWLIM	Layer of lower limit
LVMAX	Layer of maximum velocity
P	Exponent for velocity profile function
Q1	Logical variable; true for positive withdrawal limit function; false for negative
Q2	Logical variable; true for positive withdrawal limit function; false for negative
QZ	Logical variable; true when the vertical distance between the crest and the lower unit is less than the head above the crest, i.e., $Z_o/H_w < 1$; false, when the above is not true
QBLIM	Logical variable; true for bottom withdrawal interference; false for no interference from bottom
QSHIFT	Logical variable; true when VWEIR is called from SHIFT; false when not called from SHIFT
QSUB	Logical variable; true when the weir is submerged; false when not submerged

(Continued)

(Sheet 2 of 4)

Table 6 (Continued)

Variable	Definition
QTLIM	Logical variable; true for surface withdrawal interference; false for no interference from surface
QZ	Logical variable; true when the vertical distance between the crest and the lower limit is less than the head above
RATIO	Ratio of the product of distance and density difference between the point of maximum velocity and a local point to the product of the distance and density difference between the point of maximum velocity and a given limit
SMALL	Essentially zero; check for approximate equality of two values
SUM	Sum over layers of the velocities for each layer
SUBR	Subroutine name
TOPLIM	Upper withdrawal limit
V(I)	Normalized velocity profile for weir
VM	Scaling term equal to FLORAT/SUM
VMAX	Maximum velocity in the normalized velocity profile, 1.0
WRDEN	Density at weir crest, g/cm^3
X1	Elevation of an initial search limit
X2	Elevation of a second search limit
X3	Elevation of a third search limit
X4	Elevation of a fourth search limit
XCHECK	Local variable for the value of $Z + H_W/H_W$
XDUMY	Assigned 0.0; used in ERROR () argument list
XDUMY1	Same as XDUMY
XDUMY2	Same as XDUMY

(Continued)

(Sheet 3 of 4)

Table 6 (Concluded)

Variable	Definition
XDUMY3	Same as XDUMY
XVMAX	Distance from bottom to elevation of maximum velocity
Y	Distance from location of maximum velocity to local elevation
YLOW	Distance from location of maximum velocity to lower withdrawal limit
YTOP	Distance from location of maximum velocity to upper withdrawal limit
YVMAX	Location of maximum velocity referenced to lower withdrawal limit
ZLOW	Difference between crest elevation and lower withdrawal limit
ZONE	Distance from lower withdrawal limit to upper withdrawal limit

(Sheet 4 of 4)

59. A description of the portion of the profile below the elevation of maximum velocity for a submerged weir is given by

a. Without bottom interference

$$\frac{V_1}{V_{MAX}} = \left(1 - \frac{y_1 \Delta\rho_1}{Y_1 \Delta\rho_{1m}}\right)^3 \tag{21}$$

b. With bottom interference

$$\frac{V_1}{V_{MAX}} = 1 - \left(\frac{y_1 \Delta\rho_1}{Y_1 \Delta\rho_{1m}}\right)^2 \tag{22}$$

where

V_1 = layer velocity in the zone of withdrawal at a distance y_1 below the elevation of maximum velocity

V_{MAX} = maximum layer velocity in the withdrawal zone; taken as 1.0 for normalized distribution (see V_{MAX} discussion in section on VPORT)

y_1 = vertical distance from the elevation of maximum velocity to that of the corresponding layer velocity V_1

$\Delta\rho_1$ = density difference of fluid between the elevations of the maximum velocity and the corresponding layer velocity V_1

Y_1 = vertical distance from the elevation of the maximum velocity to the lower limit of the zone of withdrawal

$\Delta\rho_{1m}$ = density difference of fluid between the elevations of the maximum velocity and the elevation of the lower limit

60. Free weir. A description of the portion of the profile below the elevation of maximum velocity is given by

$$\frac{V_1}{V_{MAX}} = 1 - \left(\frac{y_1 \Delta\rho_1}{Y_1 \Delta\rho_{1m}}\right)^P \tag{23}$$

where the exponent P varies with the user's choice of a weir coefficient COEF such that for

$|COEF - 3.00| < 0.1$, $P = 1.5$

$|COEF - 3.33| < 0.1$, $P = 0.5$

$|COEF - 4.10| < 0.1$, $P = 0.2$

where $|\,|$ denotes the absolute value of the expression. If none of the above is true, $P = 4.35 - (1.04)COEF$.

61. The velocity for each layer is found directly since all variables y_i , Y_i , $\Delta\rho_{im}$, and $\Delta\rho_i$ (i = 1 or 2) are known for each layer in the withdrawal zone.

Special considerations

62. The maximum normalized velocity (1.0) is assigned for each withdrawal layer above and below the elevation of maximum velocity if the layer's density does not vary from the density at the elevation of maximum velocity, i.e., $\Delta\rho_i = 0$.

Conversion of velocities to flow rates

63. Once the normalized velocity profile is determined for the weir, the withdrawal profile is calculated based on the total flow rate over the weir. The flow value for each layer is scaled to sum to the magnitude of the total outflow over the weir. Equation 15 (in the VPORT section) is used to calculate the layer withdrawals based on the normalized velocities.

SHIFT

Description

64. Subroutine SHIFT adjusts the limits of withdrawal zones that overlap vertically when mulltilevel outlets are operated. The adjustment is necessary to correct the difference between the simple superposition of the predicted withdrawal zones for each outlet and the actual zone as observed during testing (see Figure 21). It is believed

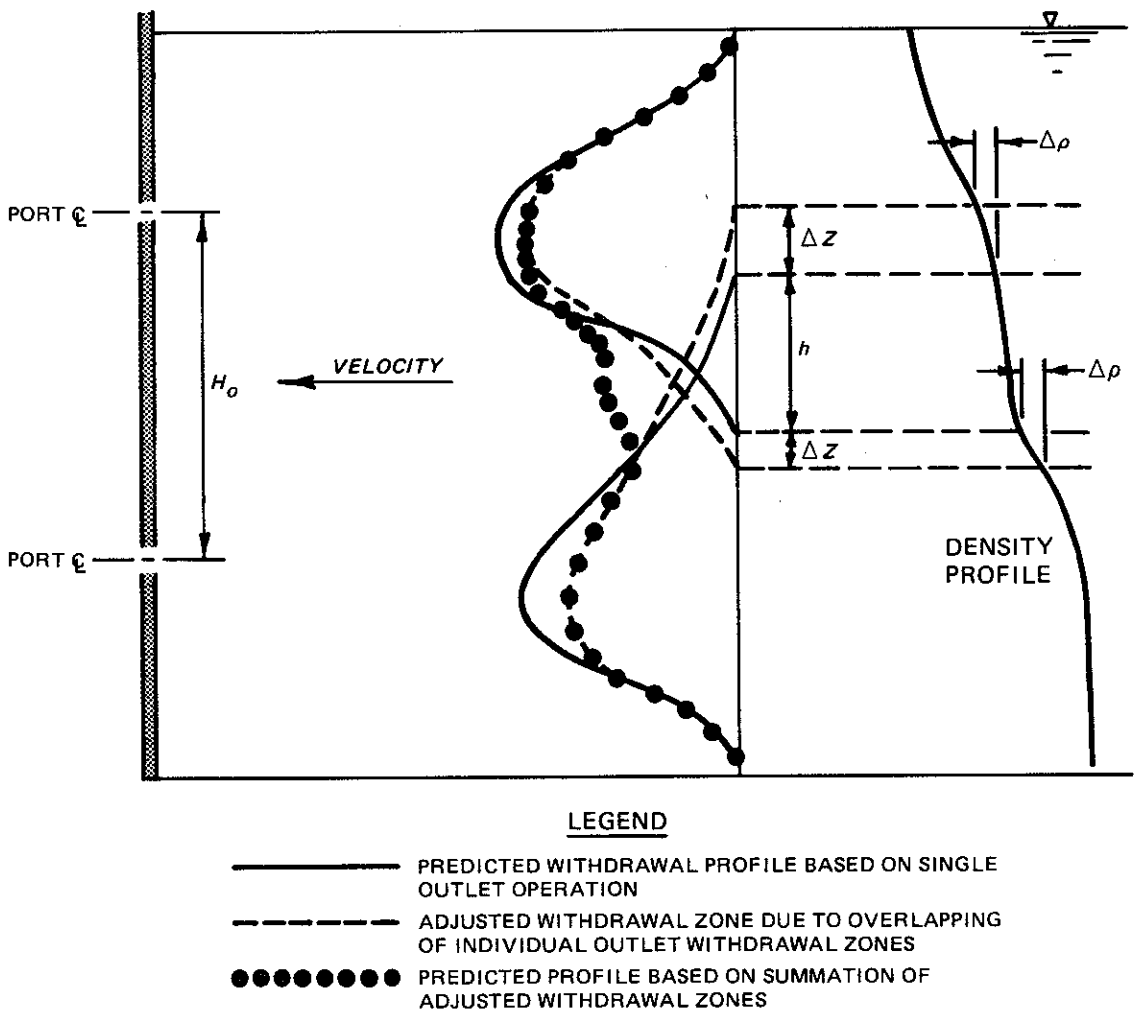


Figure 21. Schematic diagram of differences between initially calculated theoretical withdrawal zones and shifted withdrawal zones

that the difference exists because the fluid motion in one withdrawal zone tends to help the fluid movement in the other zone overcome the shear resistance forces in the region of overlap. Since the predicted velocity profile for each individual outlet does not take into account this apparent reduction in shear resistance, the superposition of the individually predicted profiles will not account for it either.

65. Bohan and Gloriod (1972) found that the discrepancy between the simple superposition of withdrawal zones and the actual zone was relatively consistent in laboratory testing. They concluded that the limits of the overlapping zones could be systematically adjusted to increase the zone thickness, thereby modifying the velocity distribution to produce individual withdrawal zones whose superposition closely approximated the actual zone. Bohan and Gloriod determined that the adjustment was a function of the amount of overlap of the two individual zones, the vertical spacing between the outlets, the density distribution of the impoundment, and the average velocity of each withdrawal zone in the region of overlap. The equation developed and used in SHIFT is

$$\frac{V_h}{\sqrt{\frac{\Delta\rho}{\rho} g\Delta z}} = 0.70 \left(\frac{h}{H_o}\right) 1.25 \qquad (24)$$

where

V_h = average velocity in the region of overlap of the upper or lower withdrawal zones as appropriate

$\Delta\rho$ = density difference between the old limit and the shifted limit

ρ = density at the original limit

g = acceleration due to gravity

Δz = vertical shift of the withdrawal limit

h = vertical distance of overlap

H_o = vertical distance between ports

66. Rearranging, Equation 24 becomes

$$V_h - 0.7 \left(\frac{h}{H_o}\right)^{1.25} \sqrt{\frac{\Delta\rho}{\rho} g\Delta z} = 0 \qquad (25)$$

The vertical shift Δz can be found by iteration. The iterative algorithm used is similar to the half-interval search described in the VPORT section. A flowchart of the SHIFT subroutine is shown in Figure 22. Table 7 describes the variables used in this algorithm.

Special considerations

67. If the surface interferes with the new shifted limit and the density at the surface is within 1×10^{-7} of density at the port, the shifted limit is assigned to the surface elevation. Also, if the bottom interferes with the new shifted limit and the density at the bottom is within 1×10^{-7} of the density at the port, the bottom is assigned as the limit. Thus, the upper limit of the lower port's withdrawal zone may experience surface interference and the lower limit of the upper port's withdrawal zone may experience bottom interference.

68. After the new limits are found for each withdrawal zone, SHIFT calls VPORT or VWEIR to recompute the flow rate profile for each zone based on the new limits and then transfers program control back to subroutine OUTVEL.

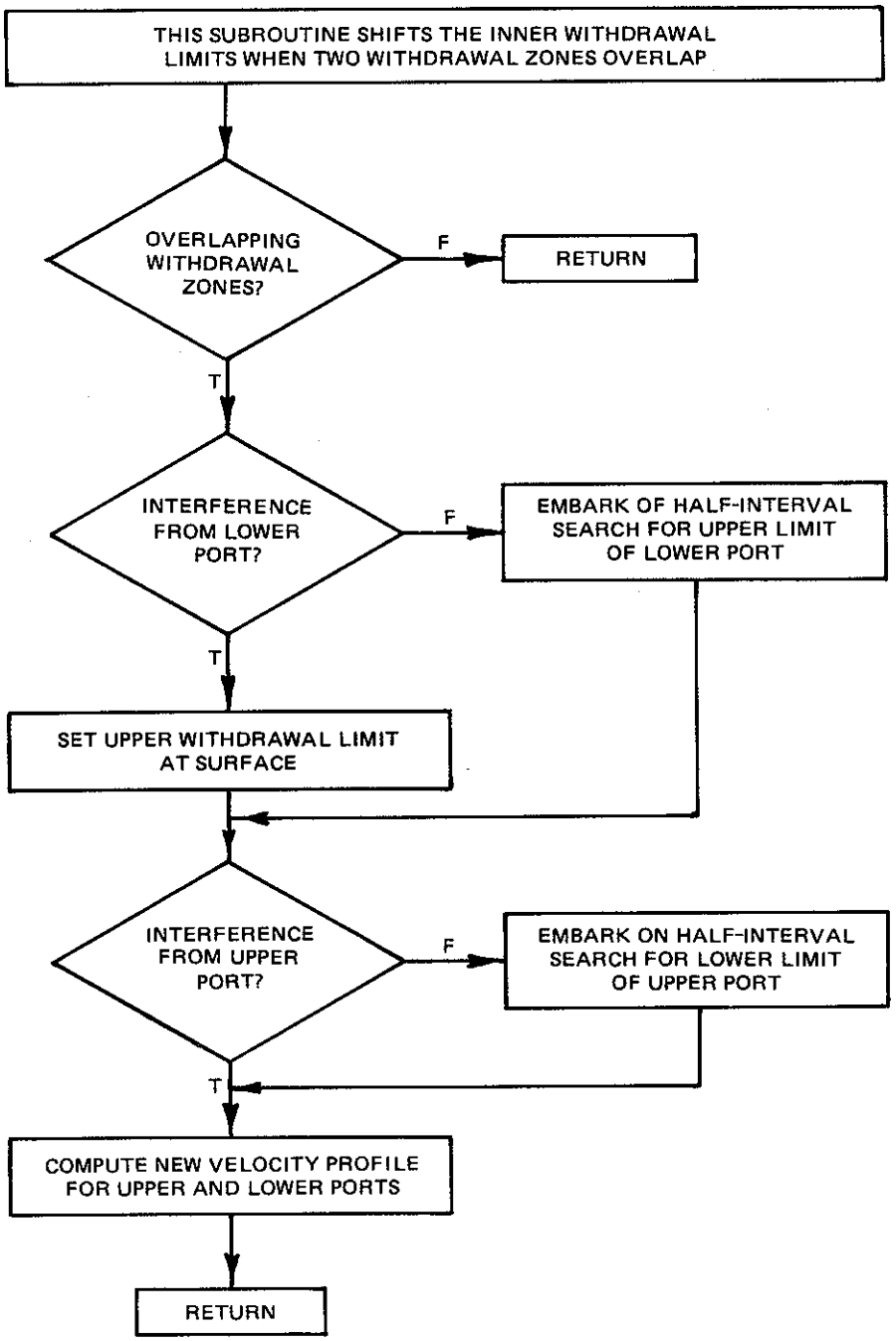


Figure 22. Flowchart for subroutine SHIFT

Table 7

SHIFT Variables

Variable	Definition
CREST	Weir crest height above bottom
DELZ	Layer thickness
DEN(I)	Density of layer I, g/cm^3
DENS1	Density at elevation of lower withdrawal limit for the outlet K + 1, g/cm^3
DENS2	Density at elevation of upper withdrawal limit for the outlet K, g/cm^3 (note: the elevation of port K is below that of port K + 1)
DEPTH	Depth of pool
F1	Value of withdrawal limit shift function FSHIFT (VH, X, D, ZL) evaluated at X1
F2	Same as F1 evaluated at X2
F3	Same as F1 evaluated at X3
FLORAT	Flow rate through one port or over a weir
FLOW(K)	Flow rate through port K
G	Gravitational acceleration
H	Distance between upper withdrawal limit of outlet K and lower withdrawal limit of outlet K + 1
HGT(I)	Percentage of layer I that is filled with water
HGTLOW	Height above bottom of lower withdrawal limit
HGTPRT	Height above bottom of center line of port
HGTTOP	Height above bottom of upper withdrawal limit
HO	Distance between vertical location of outlets K and K + 1
HTEST	Computed value used in withdrawal limit shift function

(Continued)

(Sheet 1 of 4)

Table 7 (Continued)

Variable	Definition
ISURF	Total number of layers
KFILE	File code; KFILE = 06 is output file
L1	Layer containing lower withdrawal limit of outlet K + 1
L2	Layer containing upper withdrawal limit of outlet K
LAY	Number of layers between L1 and L2
LENGTH	Length of weir crest
LLOW(K)	Layer of pool at which lower withdrawal limit for outlet K is located
LOWLIM	Layer of lower limit
LTOP(K)	Layer of pool at which upper withdrawal limit for outlet K is located
MAX	Number of search iterations allowed to determine withdrawal limits
NOUTS	Number of selective withdrawal outlets
PHGT(K)	Height above bottom of port K center line
Q1	Logical variable; true for positive value for withdrawal limit function FSHIFT (VH, X1, D, ZL); false, negative
Q2	Same as Q1 for FSHIFT evaluated at X2
QBLIM	Logical variable; true for bottom withdrawal interference; false for no interference from bottom
QPRINT	Logical variable; initially true, assigned false after print statement is executed; no other reassignment
QSBLIM	Logical variable; true when bottom withdrawal interference occurs after shifting limits; false when no such interference from bottom occurs
QSHIFT	Logical variable; true when VPORT or VWEIR is called; false when return back to SHIFT

(Continued)

(Sheet 2 of 4)

Table 7 (Continued)

Variable	Definition
QSTLIM	Logical variable; true when top withdrawal interference occurs after shifting limits; false when no such top interference occurs
QTLIM	Logical variable; true for surface withdrawal interference; false for no surface interference
QWEIR	Logical variable; true when a weir exists as an outlet device; false when there is no weir
SMALL	Essentially zero; used in value comparisons
SUBR	Subroutine name
TINY	Essentially zero; used in value comparisons
TOPLIM	Layer of upper limit
V(I)	Velocity profile value of layer I for an outlet
VH1	Average outflow of the outlet K in the zone of overlap of the K and K + 1 outlets
VH2	Average outflow of outlet K + 1 in the zone of overlap of the K and K + 1 outlets
VS(I,K)	Velocity profile value of layer I for the K outlet
WRFLOW	Discharge over weir
WRHGT	Weir crest height above bottom
WRLNG	Weir length
X1	Elevation of an initial search limit
X2	Elevation of a second search limit
X3	Elevation of a third search limit
X4	Elevation of a fourth search limit
XDUMY	Assigned 0.0; used in ERROR () argument list

(Continued)

(Sheet 3 of 4)

Table 7 (Concluded)

Variable	Definition
XDUMY1	Same as XDUMY
XDUMY2	Same as XDUMY
XDUMY3	Same as XDUMY
XXX	Used in label assignment statement
ZDN(K)	Height above bottom of the lower withdrawal limit for outlet K
ZL1	Height above the bottom of the lower withdrawal limit for the K + 1 outlet
ZL2	Height above bottom of the upper withdrawal limit for the K outlet
ZUP(K)	Height above bottom of the upper withdrawal limit for outlet K

(Sheet 4 of 4)

XPRINT

69. Subroutine XPRINT controls the output of all tabular information regarding elevations, depths, densities, normalized velocities, flow rates, temperatures, and water quality parameters. Each computational layer has characteristic values for all of the above categories, and subroutine XPRINT prints these parameter values for the layers at user-specified intervals such as at every layer or every third layer. XPRINT also summarizes input information, port center-line elevations, dimensions, crest elevation, crest length, and flow rate. A flowchart showing the organization of XPRINT is displayed in Figure 23. Table 8 lists descriptions of the variables used in XPRINT.

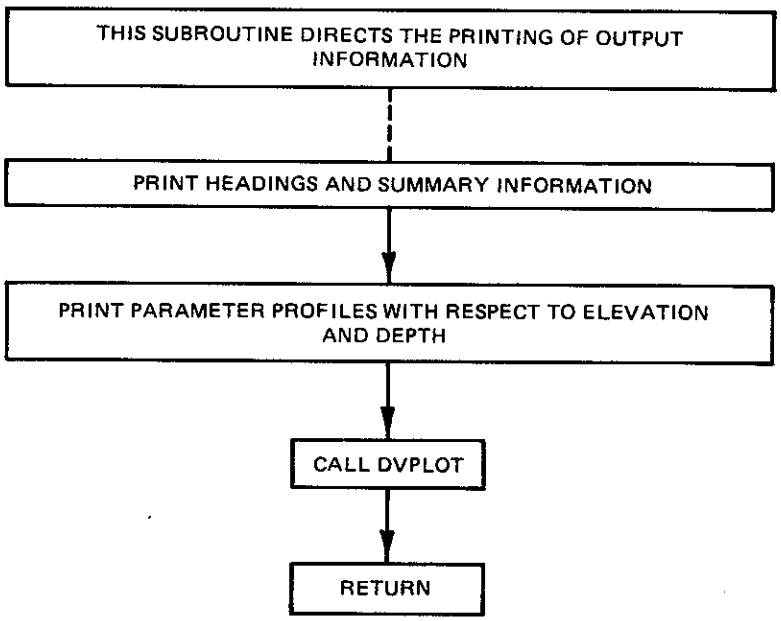


Figure 23. Flowchart for subroutine XPRINT

Table 8

XPRINT Variables

Variable	Definition
AWLBOT	Assigned result of intrinsic function AMAX1(X,Y); X = 0.0 , Y = ZDN(1) = lowest withdrawal limit
AWLUPP	Assigned result of intrinsic function AMIN1(X,Y); X = depth of pool, Y = ZUP (NOUTS) = uppermost withdrawal limit
AZDNEL	The elevation of AWLBOT, i.e., the elevation of the bottom or the lowest limit, whichever is highest
AZUPEL	The elevation of AWLUPP, i.e., the elevation of the surface or the uppermost withdrawal limit, whichever is lowest
BOTTOM	Elevation of bottom of pool
DEEP	Depth from surface to midpoint of layer
DELZ	Layer thickness
DEN(K)	Density of layer K
DENOUT	Average density of outflow, g/cm^3
DEPTH	Depth of pool
DIST	Units of length used
ELEV	Elevation of midpoint of a layer
FLOW(L)	Flow rate through port L
G	Gravitational acceleration
HEADING(K)	Data set title; up to 80 characters in length
INTER	Interval between layers to be printed in output
ISURF	Total number of layers
KFILE	File code for output; KFILE = 06
NAMEQ(K,J)	Name of the Jth water quality parameter
NOUTS	Total number of selective withdrawal outlets
NPORTS	Number of selective withdrawal ports
NQUAL	Number of quality parameters input
PELEV	Elevation of a port center line
PHGT(L)	Height above bottom of port L center line

(Continued)

(Sheet 1 of 3)

Table 8 (Continued)

Variable	Definition
PVDIM(K)	The vertical dimension of port K
QALOUT(I)	Average release concentration of the Ith water quality parameter
QCENT	Logical variable; true for temperatures in degrees Centigrade; false for temperatures in degrees Fahrenheit
QMETR	Logical variable; true when units are metric (SI); false when units are English (non-SI)
QPLOT	Logical variable; true if line printer plot of density and total velocity profiles is desired; false if not desired
QPORT	Logical variable; true if ports are present as outlets; false if no ports
QPWEIR	Logical variable; true if a port is considered "partially submerged" and therefore modeled as a weir; false otherwise
QSINK1	Logical variable; true when point sink description of port is adequate for calculation of lower withdrawal limit; false when otherwise (see point sink discussion in section on subroutine VPORT)
QSINK2	Same as QSINK1, except now pertaining to upper withdrawal limit
QTEMP	Logical variable; true for temperature profile input; false for no input temperature profile
QUAL(J,K)	K^{th} quality parameter value for layer J
QWEIR	Logical variable; true when a weir is included as an outlet device; false when there is no weir
SUMOUT	Total outflow through all outlets
SURFACE	Elevation of pool surface
TEMOUT	Average temperature of outflow, degrees Fahrenheit or Centigrade
TEMP(K)	Temperature of layer K, degrees Fahrenheit or Centigrade
TITLE(I)	Title of input file; up to 80 characters in length
VEL(K)	Total release velocity profile value of layer K
WELE	Elevation of weir crest

(Continued)

(Sheet 2 of 3)

Table 8 (Concluded)

Variable	Definition
WRFLOW	Flow rate over weir
WRHGT	Weir crest height above bottom
WRLNG	Weir length
WTHDRW(K)	Withdrawal flow rate for layer K
WTHETA(K)	Withdrawal angle for port K
XFEET	Equal to feet
XMETERS	Equal to meters
Y(K)	Height of midpoint of layer K above the bottom
ZDN(I)	Height above bottom of the lower withdrawal limit of outlet I
ZDNEL	Elevation of the lower withdrawal limit for the bottom outlet
ZUP(I)	Height above bottom of the upper limit of outlet I
ZUPEL	Elevation of upper withdrawal limit for the top outlet

(Sheet 3 of 3)

DVPLOT

70. Subroutine DVPLOT generates a line printer plot of elevation (and depth) versus density and velocity. The values are plotted at the user-defined interval. This subroutine is called from subroutine XPRINT. Figure 24 and Table 9 show the algorithm flowchart for DVPLOT and a list of variable descriptions, respectively.

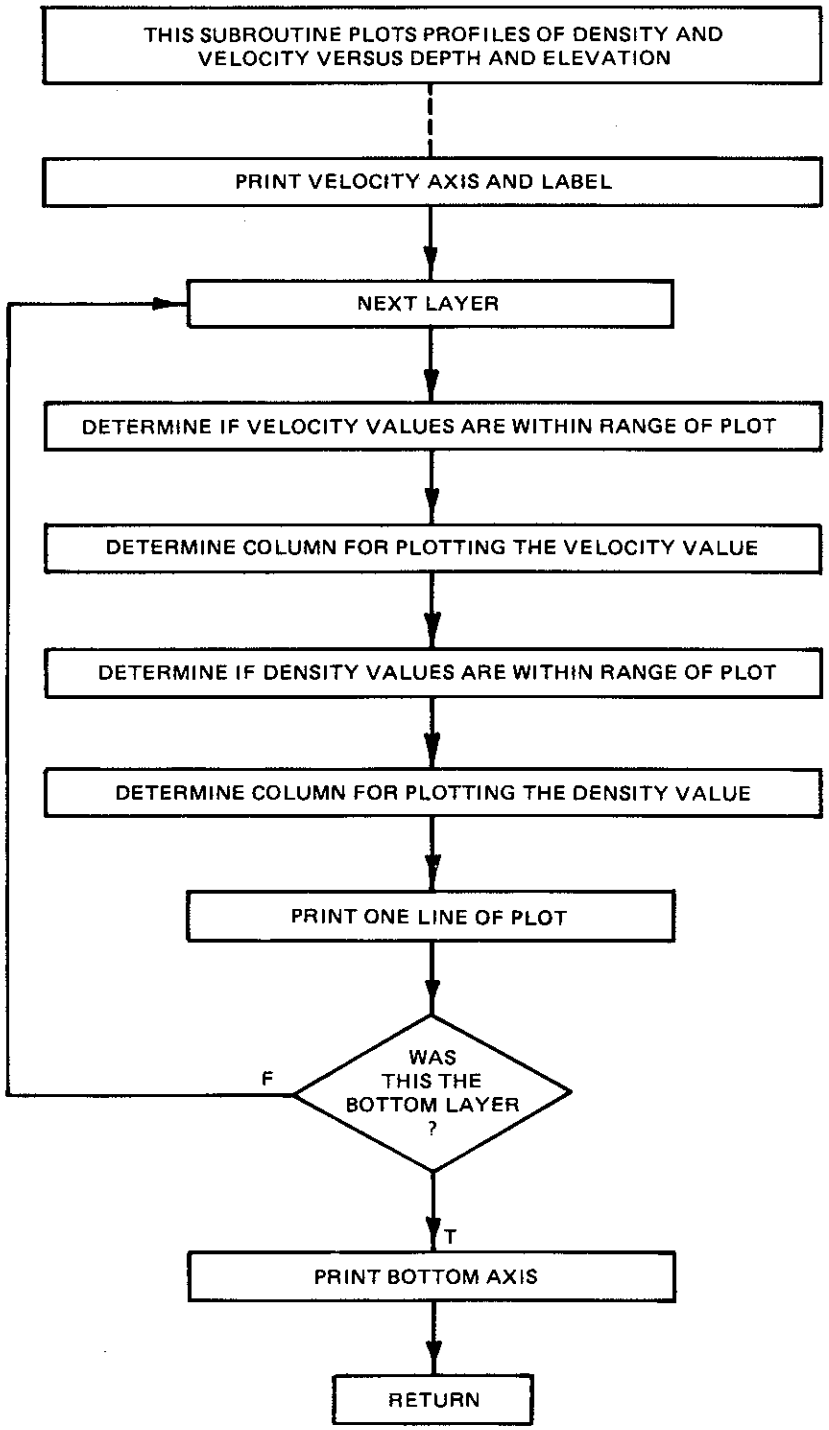


Figure 24. Flowchart for subroutine DVPLOT

Table 9

DVPLOT Variables

Variable	Definition
BLANK	Assigned value of 1H for FORMAT statements
BOTTOM	Elevation of pool bottom
CHANGE	Assigned value of 200; used to increment values along velocity axis of plot
CHANGE2	Assigned value of DENDIF; used to increment values along density axis of plot
COLUMN(K)	Assigned value of variable BLANK or V; K = 100 . There are 100 COLUMN values for each line printed. V is assigned to columns where a velocity greater than zero exists; used for plot marks
COLUMN2(K)	Same as COLUMN(K), except it is used for densities and symbol is D instead of V
D	Assigned value of 1HD in data statement; used as plot mark for density
DDIF	Difference between the maximum and minimum densities in the profile when the minimum value is truncated after the third decimal place
DEEP	Used in loop, it begins at zero and is incrementally increased by a layer thickness. Each increment is printed on the vertical axis, thereby listing depth from surface to bottom
DELZ	User-defined layer thickness
DEN(I)	The density of layer I
DENDIF	Difference between maximum and minimum densities on density axis
DMAX	The maximum density on the density axis of plot, assigned to be 1.00300 g/cm^3
DMIN	The minimum density on the density axis of plot, assigned to be 0.99300 g/cm^3
DN	Temporary assignment of DEN(I)
DSPACE	Value is incremented by CHANGE2, and new value is marked on density axis of plot producing the density scale
DUM	Used to truncate DMIN after the third decimal point

(Continued)

Table 9 (Continued)

Variable	Definition
ELEV	Used to print elevations of layers; first value is the elevation of the surface and following values are incrementally reduced by DELZ
FIRST	Assigned the value of variable BLANK or PEGGED; PEGGED is assigned when the density or velocity is less than or equal to zero. The value of FIRST is printed in the first column of the line being plotted
IJK	The column number in which the variable V or D is placed for variable array COLUMN and COLUMN2, respectively
ISURF	Number of layers
KFILE	Output file code
LAST	Assigned the value of variable BLANK or PEGGED; PEGGED is assigned when the velocity or density exceeds VELMAX or DMAX, respectively. LAST is printed in last column of plot
PEGGED	Assigned value of 1H*
PLUS	Assigned value of 2H+
QRANGE	Logical variable; true when velocity value for layer lies between the minimum and maximum velocity value on the plot scale or when the layer density lies between the minimum and maximum density value on the plot scale; false when the above are not true
V	Assigned the value 1HV; used as plot mark for velocity
VEL(I)	Total release velocity profile value of the Ith layer for all outlets combined
VELMAX	Assigned value of 2.0
VL	Assigned the value of VEL(I)
VMAX	Maximum velocity in the velocity profile
VSPACE	Value is incremented by value of CHANGE, and new value is marked on velocity axis producing the velocity scale

ERROR

71. Subroutine ERROR prints error messages and terminates program execution when one of the internal program checks (see Appendix D) has been failed. ERROR receives through its argument list an error number, the name of the subroutine in which the error occurred, the value of the variable CHECK (when applicable), and the values that were expected for CHECK (when applicable). The subroutine then prints the error number, the name of the subroutine-of-the-occurrence, and a brief statement as to the nature of the error, and terminates execution of the program. A flowchart of the subroutine is given in Figure 25, and a listing of variables used in subroutine ERROR is given in Table 10.

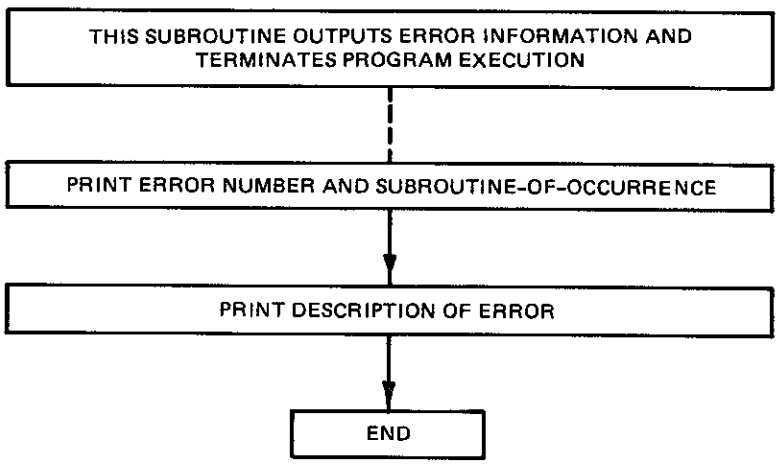


Figure 25. Flowchart for subroutine ERROR

Table 10

ERROR Variables

Variable	Definition
CHK	In subroutine argument list; carries the value of the string which is in error. This is used only when a string is checked in the program for correspondence with an expected string
ERR	In subroutine argument list; carries the value of the error code number
KFILE	Output file code
SUBR	In subroutine argument list; carries the name of the subroutine in which the error occurred
XCHK1	In subroutine argument list; carries the value of the string expected in the corresponding program check. If the program check was not a string comparison, value is zero
XCHK2	In subroutine argument list; when program check has more than one possible expected string, XCHK2 carries the value of one of the strings expected in the program check. If no strings or one string was expected in program check, value is assigned zero
XCHK3	Same as XCHK2, except that it is used when three possible values were expected in the program check. If zero, one, or two possible strings were expected, value is zero

XREAD

72. Subroutine XREAD transfers information from the input file into working storage. The input file contains program control parameters, outlet characteristics, and reservoir profile data. As the input data are read, they are checked to ensure that the user has observed proper order and format. A general flowchart of the subroutine is shown in Figure 26. Table 11 lists descriptions of the variables used in the subroutine.

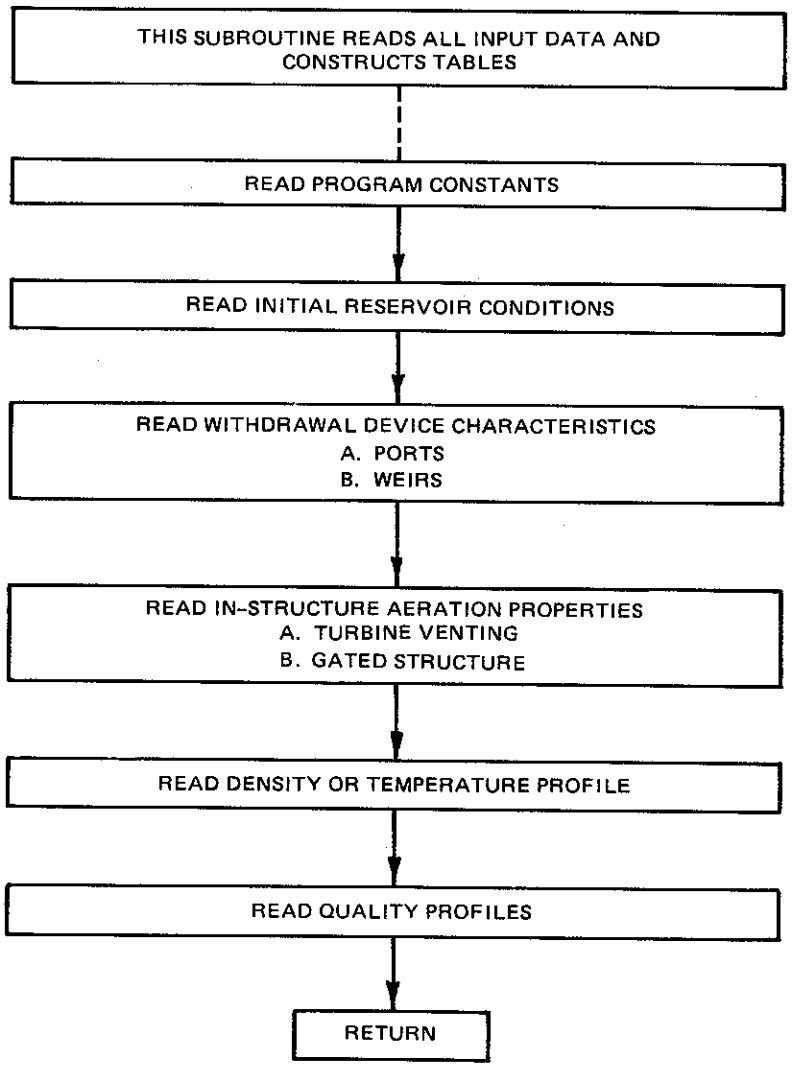


Figure 26. Flowchart for subroutine XREAD

Table 11

XREAD Variables

Variable	Definition
BOTTOM	Elevation of bottom of pool
CHECK	Assigned as first four characters encountered on a line in the user input data file
CHECK1	Same function as CHECK
CHECK2	Same function as CHECK
COEF	Discharge coefficient for free weir flow
DELZ	Layer thickness
DEN(M)	Density of layer M, g/cm^3
DEPTH	Depth of pool
DUMMY(I)	Line of input from data file for echo print of input
DUMQUAL(K)	Working storage assigned to qualities during interpolation of a complete computational profile, appropriate units
DUMYQ(K)	Working storage assigned to height of qualities above bottom during interpolation of a complete completion profile
FLODUM(K)	Working storage assigned to port flow during ordering of ports from bottom to top
FLOW(K)	Release flow rate through the Kth port
G	Gravitational acceleration
HDUM	Working storage for port horizontal dimension
HEADING(I)	Title of data set; up to 80 characters in length
HGT(I)	Percentage of layer I that is filled with water
HGTDUM(I)	Working storage assigned as port height above bottom; used during reordering of ports from bottom to top
IFILE	File code for input data file; equals 05
INTER	Interval of layers to be printed in output
ISURF	Total number of layers in a profile
JFILE	File code for auxiliary use; value should usually be specified by user as 00
KFILE	File code for output data file; equals 06
NAMEQ(J,I)	Name of the Ith water quality parameter

(Continued)

(Sheet 1 of 4)

Table 11 (Continued)

Variable	Definition
MFILE	Temporarily assigned IFILE or JFILE value; used in echo-print operation
NP	Number of ports + 1; used during ordering of ports from top to bottom
NPORTS	Number of selective withdrawal ports not including a weir
NQUAL	Number of water quality parameters
NSETS	Number of data sets in the input file
NUMD	Total number of density values in density profile
NUMQ(I)	Total number of quality values in the Ith quality parameter profile
NUMT	Total number of temperature values in temperature profile
PHDIM(K)	Horizontal dimension of port K
PHGT(K)	Height above bottom of the Kth port center line
PVDIM(K)	Vertical dimension of port K
QAERA	Logical variable; true, consider DO uptake due to aeration through a gated structure outlet works; false, do not consider aeration
QCENT	Logical variable; true, temperatures are in degrees Centigrade; false, Fahrenheit
QDEN	Logical variable; true when density is input; if false, program develops density profile
QECHO	Logical variable; true for echo print of input data file; false for no echo print
QFIRST	Logical variable; true, XREAD has been called once by the main program; false, XREAD has not been called by the main program
QMETR	Logical variable; true when units are metric (SI); otherwise, false and units are English (non-SI)
QPLOT	Logical variable; true if line printer plot of density and total velocity profiles is desired; false if not desired
QPORT	Logical variable; true when ports are present as outlet devices; false if there are no ports
QQUAL	Logical variable; true if quality profiles will be entered; false if quality profiles will not be entered

(Continued)

(Sheet 2 of 4)

Table 11 (Continued)

Variable	Definition
QSUB	Logical variable; true for a submerged weir; false for a free weir
QTAB1	Logical variable; true when profile data are input as one table; false when two tables are used: one for vertical locations and one for corresponding profile parameter values (see input description, Part V)
QTEMP	Logical variable; true when temperature profile is input; false otherwise
QTWFUN	Logical variable; true when a tailwater elevation function will be used; false when not used
QUAL(I,M)	M^{th} value for the Ith quality parameter, appropriate units
QWEIR	Logical variable; true when a weir is included as an outlet device; false otherwise
SUBR	Subroutine name
SURFACE	Elevation of pool surface
TABTYP	When equal to 1, profile data will be input as one table; when equal to 2, data will be input as two tables (see input description, Part V)
TEMP(M)	Temperature value of layer M, degrees Centigrade or Fahrenheit
TITLE(I)	Identification title for input file; up to 80 characters in length
TWEL	Value of tailwater elevation
UNITS	When equal to CENT, temperature is specified in degrees Centigrade; when equal to FAHR, Fahrenheit degrees
VDIMDM(I)	Temporarily assigned value of PVDIM(K); used in port reordering algorithm
VDUM	Working storage for port vertical dimension
WRFLOW	Outflow quantity over weir
WRHGT	Weir crest height above bottom
WRLNG	Weir length
WRTYPE	Equal to SUBM for a submerged weir; equal to FREE for a free weir
WTHETA(K)	Withdrawal angle for port K

(Continued)

(Sheet 3 of 4)

Table 11 (Concluded)

Variable	Definition
Y(I)	Height above bottom of the midpoint of layer I
YD(M)	Elevation, height above bottom or depth below surface of the Mth density profile value
YQ(M)	Elevation, height above bottom or depth below surface of the Mth quality profile value
YT(M)	Elevation, height above bottom or depth below surface of the Mth temperature profile value

All of the following variables are used to check for input errors and are assigned a character value equal to the four characters that follow the X.

XANGL	XDENS	XFILE	XINTE	XPORT	XSURF	XWEIR
XBOTT	XDEPT	XFLOW	XLENG	XPRIN	XTABL	
XCENT	XELEV	XFREE	XMETR	XQUAL	XTEMP	
XCOEF	XENGL	XHDIM	XNUMB	XSTOP	XTHIC	
XDATA	XFAHR	XHEIG	XPLOT	XSUBM	XVDIM	

(Sheet 4 of 4)

Echo print

73. XREAD has an "echo-print" feature that is optional to the user (option commands are discussed in Part IV). The echo-print feature prints the entire input file to the output, thus allowing the user to see the input and output together. This feature, along with the error checks (given in Appendix D), provide the user with the capability to diagnose error problems in input.

Computational layers

74. XREAD develops the computational layers used in program calculations. The number of computational layers is determined by dividing the impoundment depth by the user-specified layer thickness.

AERATE

75. Subroutine AERATE accounts for the reaeration (DO uptake) of release water during flow through a gated-conduit outlet works. The subroutine determines the upstream oxygen deficit based on the flow-weighted average of DO in the release (calculated in subroutine OUTVEL). AERATE computes the downstream deficit based on the "energy dissipation" model outlined by Wilhelms and Smith (1981) and subsequently calculates the release DO concentration. Figure 27 shows the flowchart for subroutine AERATE. Table 12 lists descriptions of the subroutine variables.

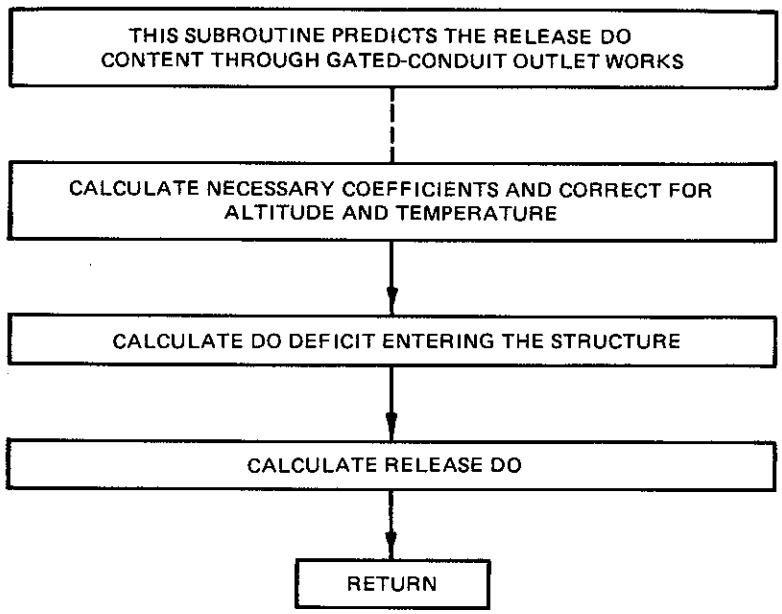


Figure 27. Flowchart for subroutine AERATE

Table 12

AERATE Variables

Variable	Definition
ALT	The elevation of the pool bottom
BOTTOM	The elevation of the pool bottom
C	Temperature-adjusted escape coefficient for oxygen
CSAT	Oxygen saturation concentration adjusted for altitude
C20	Equals 0.045 if English (non-SI) units are used; 0.1476 if metric (SI) units are used
DELH	Elevation difference between the pool surface and the tailwater surface
DEPTH	Depth of the pool
DF	DO deficit exiting structure
DI	DO deficit entering structure
QAERA	Logical variable; true, consider DO uptake due to aeration through a gated-structure outlet works; false, do not consider uptake
QALOUT(1)	The outflow DO concentration (weighted average) that is adjusted by subrouting AERATE
QMETR	Logical variable; true when metric (SI) units are used; false when English (non-SI) units are used
TEMOUT	Outflow temperature (weighted average)
TWEL	Elevation of tailwater

VENTING

76. Subroutine VENTING accounts for the DO uptake by release water if venting techniques are applied to turbine releases. VENTING is based on WES and Tennessee Valley Authority studies of Francis turbines which have shown that a maximum of 30-percent reduction of the initial (penstock) deficit is reasonable with turbine venting. Therefore, VENTING adjusts the flow-weighted average of DO in the release (as calculated in subroutine OUTVEL), to reflect a 30-percent reduction in the deficit. The results predicted by VENTING should be considered as a maximum DO uptake that can be achieved with venting. The reader is referred to Wilhelms (1984) and Wilhelms, Schneider, and Howington (1987) for additional details. Figure 28 is a flowchart of subroutine VENTING; descriptions of the subroutine variables are given in Table 13.

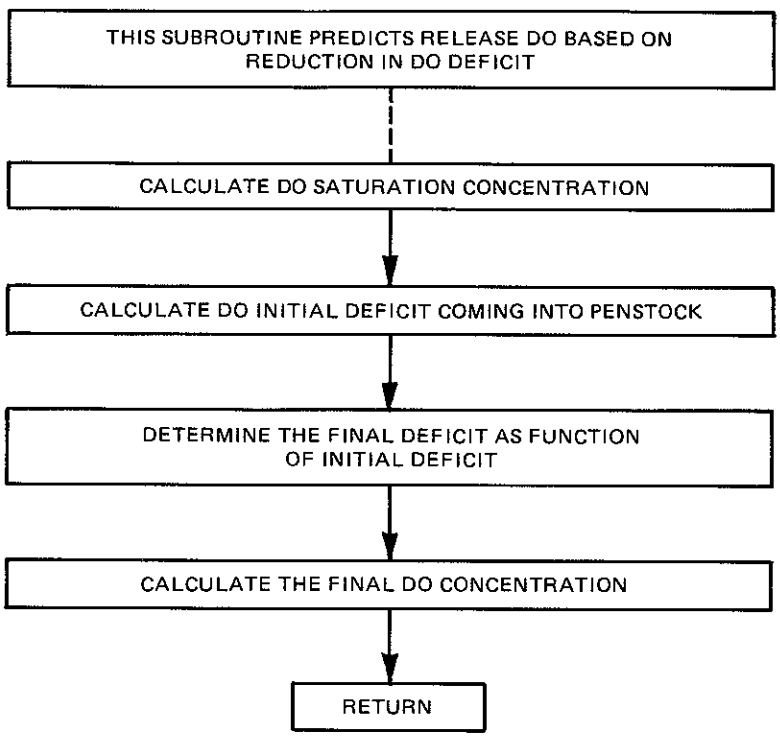


Figure 28. Flowchart for subroutine VENTING

Table 13

VENTING Variables

Variable	Definition
ALT	Elevation of the pool bottom
BOTTOM	Elevation of the pool bottom
CSAT	Oxygen saturation concentration adjusted for altitude
DF	DO deficit of release water from structure
DI	DO deficit of water entering structure
QALOUT(1)	The flow-weighted average of the outflow DO concentration which is adjusted by subroutine VENTING
TEMOUT	The flow-weighted average of the outflow temperature

PART IV: ASSUMPTIONS AND LIMITATIONS

77. In general, if the outlet and approach characteristics are simple, the results produced by SELECT will be accurate. However, if the characteristics are not of a simple nature, the assumptions inherent in the equations and theory of SELECT may be violated. The user should be aware of these assumptions and how they limit the accuracy of results. Thus, a listing of these assumptions is provided below. Violating these assumptions in a misapplication of the SELECT code will result in erroneous predictions. In some cases, more sophisticated modeling tools, such as multidimensional mathematical and physical models, may have to be employed to gain adequate results. In other cases, manipulation of the SELECT code itself might be possible. These solutions, however, must be considered on a site-specific basis. The user is encouraged to consult WES if any of the assumptions listed below are violated. Also included in the listing is a brief description of the limitation and effect of the assumption on SELECT predictions.

Geometry of Ports

78. Bohan and Grace (1969,1973) assumed that orifice geometries had no effect on the withdrawal zone. The dimensions of the orifice were small in comparison to the withdrawal zone thickness, e.g., point sink assumption. If this assumption is violated, SELECT will warn the user with an output statement but will continue to output results as if the assumption were valid.

Impoundment Width

79. The width of the impoundment in the region approaching the outlet works is assumed to be greater than the thickness of the withdrawal zone. Widths that are too narrow will cause lateral constrictions in the development of the withdrawal zone which will cause the extent of the withdrawal zone to increase beyond that predicted by

SELECT. Thus, if SELECT is run when this assumption is not valid, the calculated zone thickness may be less than the actual zone thickness.

Approach Path

80. The approach toward the outlet works is assumed free of obstruction. For example, a ridge at the bottom of the impoundment a short distance upstream from the outlet works could interfere with formation of the withdrawal zone. In such a case, the ridge may control withdrawal for one discharge with the outlet controlling for other discharges.

Approach Curvature

81. The approach toward the outlet works is assumed relatively straight. When the approach being modeled is curved, the withdrawal zone prediction may not be accurate.

Multiple Horizontal Ports

82. Use of SELECT in the past has shown that site-specific judgments need to be made concerning the modeling of multiple horizontal ports. The characteristics of withdrawal due to release through multiple horizontal ports remains largely undocumented. To illustrate, assume the existence of two ports that each withdraw as a point sink. It can be envisioned that, if the ports are far enough apart, the withdrawal zone of one port will not influence the withdrawal zone of the other. If the same ports are very near to each other, a withdrawal zone could be very closely approximated by the point sink equations by assuming that there is a single port with a flow rate equal to the combined magnitudes of the individual flow rates. The problems arise in the transition between ports that are hydrodynamically far apart or close together, because one port will affect the withdrawal characteristics of the others to some degree.

Weir Crest Above Thermocline

83. The equations used in the calculation of withdrawal limits for flow over a weir assume that the weir crest is above the thermocline. If this assumption is violated, the results from the program may be erroneous.

Hydraulic Integrity

84. SELECT assumes that it is hydraulically possible to proportion flows between multiple ports as input by the user. SELECT makes no check, for example, to ensure that multiple ports in the same wet well are not used. SELECT users should be aware of their responsibility to input operational scenarios which are viable for water quality control.

Simultaneous Port-Weir Operation

85. A port and a weir can be operated simultaneously only if each is releasing flow under its own control. For example, a spillway (weir) could be operated simultaneously with a water quality intake (port). Prediction of the withdrawal zone formed by a cofferdam (weir) directly in front of an outlet works is beyond the scope of SELECT. Such an operation would require more sophisticated modeling.

PART V: INPUT DATA

Descriptions

86. The following descriptions are presented to define the data that are needed as input for SELECT. All are listed in the order as they should appear in an input file. Formal descriptions of the input formats are given in Appendix A; example input files are given in Appendix B.

a. TITLE - user-specified label to identify the global input file. The title should not exceed 80 characters in length (including spaces, numbers and punctuation).

b. FILES - identifies the input and output file codes used. The first number in the field following FILES identifies the input source. The second number identifies the output source. The third number identifies a file that may be used for debugging and should be left blank by the user if not desired. This is followed by a number, NSETS, which indicates how many data sets are appended together in the input file.

c. DATA SETS - each must contain a heading, all port and weir information, and all necessary and desired parameter profiles. The ability to model several data sets in one execution saves the user time. That is, if the user intends to analyze several data sets, they can all be run in one program loading rather than several.

d. PRINT INPUT - tells the program that the user desires an echo print (a copy of the input file) along with the output.

e. HEADING - similar to TITLE except that it labels a single data set in an input file. Each data set in an input file must have a heading. The heading may be up to 80 characters in length (including numbers, blanks, and punctuation).

f. METRIC OR ENGLISH - indicates which system of units to be used.

English (non-SI) Units	Metric (SI) Units
feet, seconds	meters, seconds

Note: Densities are assumed to be in grams per cubic centimeter. The program default is English (non-SI) units.

g. TABLES - the program accepts two different formats for table listings of any type of profile information. Whichever format is chosen, that format must be used for all tables in a single data set.

(1) Format 1 - indicated by following the word TABLES on the input line by the number 1. This tells the program that the user wishes to enter profile data as one value of elevation and the associated parameter value per line.

(2) Format 2 - indicated by following the word TABLES on the input line by any number other than 1. This tells the program that the user wishes to enter data in the following way: the elevations will be listed eight values per line until all elevations are entered; then the associated parameter values will be listed eight values per line until all parameter values are entered. The positions of the parameter values on a line must match the line positions of the associated elevations.

h. THICKNESS - indicates the user-desired thickness of each computational layer. It is suggested that the thickness not exceed 5 ft (or 1.5 m) in order to preserve definition of the input profiles. The computational layer has parameter values assigned to it, such as velocity, density, temperature, and water quality. Since the parameter value is taken as constant throughout the layer, any variation in the actual parameter value inside the layer is neglected. Thus, in order to prevent any gross errors from occurring due to the differences between the actual and discretized profiles, a layer thickness of no more than 5 ft should be used. The layer thickness limit may vary from project to project.

i. INTERVAL - determines how often computational layer parameter values are to be output--every layer, every other layer, etc. An interval of "1" will output values for every layer. An interval of "2" will output values for every second layer, and so on. An interval "0" will not be accepted by the program.

j. SURFACE - the value of the surface elevation. This must be input as an elevation above the user-defined datum.

k. BOTTOM - the value of the bottom elevation. This must be input as an elevation above the user-defined datum.

l. PORTS - identifies the number of operating ports to be modeled by the program in a single data set. The number of operating ports is limited to five by the program dimensions.

m. VDIM - the value of the vertical dimension of the port. If more than one port is being modeled, the vertical dimensions for all of the ports must be listed on one line. They may be listed from the bottom port to the top port, or vice versa.

n. HDIM - the value of the horizontal dimension of the port. If more than one port is modeled, the horizontal dimensions of all the ports must be listed on one line. They may be listed from the bottom port to the top port, or vice versa; however, the ordering must match the ordering of VDIM.

o. PORT ELEVATION - the vertical port positions in the impoundment may be given as elevation above a user-defined datum, as depth below the surface, or as height above the bottom. Elevation (or depth or height) values for multiple ports must be listed on one line. The ordering may be from the bottom port to the top port, or vice versa. Again, however, the ordering must match the ordering used for VDIM.

p. FLOW - the value of the flow rate through a port or over a weir. If multiple ports are modeled, the flow rates for all the ports must be listed on one line. The ordering of the values may be from the top port to the bottom port, or vice versa. This ordering must also match the ordering of VDIM.

q. WITHDRAWAL ANGLE - the effective angle of withdrawal in the horizontal plane of the port. The withdrawal angles for all ports must be entered on one line and may be ordered from bottom port to top port, or vice versa. The ordering must match that of VDIM.

r. WEIR - indicates that a weir is to be modeled by the program.

s. WEIR TYPE - indicates whether the weir is a free weir (discharge unaffected by downstream pool) or a submerged weir (discharge affected by downstream pool).

t. COEFFICIENT - the weir coefficient to be used if the weir is not considered submerged.

u. WEIR LENGTH - the length of the crest of the weir.

v. WEIR CREST ELEVATION - the elevation above the user-defined datum, depth below the surface, or height above the bottom of the weir crest.

w. TURBINE VENTING - indicates that the improvement in DO content due to turbine aeration should be taken into account. The improvement is based on a 30-percent decrease in the penstock DO deficit. A DO water quality profile must be part of the input file if this command is used.

x. GATED STRUCTURE - indicates that the improvement in DO content due to aeration through gated-conduit structures should be taken into account. The improvement is based on results of Wilhelms and Smith (1981). A DO water quality profile must be part of the input file if this command is used.

y. TAILWATER ELEVATION - indicates that the tailwater elevation is entered. The tailwater elevation must be used only if the "GATED STRUCTURE" command is used.

z. TAILWATER FUNCTION - often the tailwater cannot be given directly but is a function of the discharge. In that case, a function subprogram must be written and appended to the SELECT code. This command may be used only if the GATED STRUCTURE command is used. For assistance with this input, the user should contact personnel of the Reservoir Water Quality Branch of the WES Hydraulics Laboratory.

Profile Formats

87. Input profile formats for a single data set must match the format specified on the "TABLES" card.

a. DENSITY - density profile is necessary for the operation of SELECT. If a density profile is not available, a temperature profile must be input from which the program can generate densities.

b. OTHER PARAMETERS - water quality parameter profiles are necessary if release water quality predictions are desired. If the gated structure or turbine venting input commands are used, a DO profile must be the first water quality profile listed after the density and/or temperature profiles. If a density profile is input to define reservoir stratification (to account for suspended or dissolved solids) and release temperature is of interest, the temperature profile may be input as a water quality parameter. Up to four other water quality parameter profiles may be entered.

REFERENCES

Bohan, J. P., and Gloriod, T. L. 1972. "Simultaneous, Multiple-Level Release from Stratified Reservoirs; Hydraulics Laboratory Investigation," Research Report H-72-3, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Bohan, J. P., and Grace, J. L., Jr. 1969. "Mechanics of Flow from Stratified Reservoirs In the Interest of Water Quality; Hydraulics Laboratory Investigation," Technical Report H-69-10, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Bohan, J. P., and Grace, J. L., Jr. 1973. "Selective Withdrawal from Man-Made Lakes; Hydraulics Laboratory Investigation," Technical Report H-73-4, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Dortch, M. S., and Holland, J. P. 1984. "A Technique to Optimally Locate Multilevel Intakes for Selective Withdrawal Structures; Hydraulics Laboratory Investigation," Technical Report HL-84-9, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Grace, J. L., Jr. 1971. "Selective Withdrawal Characteristics of Weirs; Hydraulics Laboratory Investigation," Technical Report H-71-4, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Holland, J. P. 1984. "Design of Selective Withdrawal Intake Structures," Proceedings of the Hydraulics Specialty Conference, Coeur D'Alene, Idaho, 14-17 August 1984, American Society of Civil Engineers.

Smith, D. R., et al. 1987. "Improved Description of Selective Withdrawal Through Point Sinks," Technical Report in preparation, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Wilhelms, S. C. 1984. "Hydropower and Water Quality," Proceedings of a Seminar on Applications in Water Quality Control, 31 January-1 February 1984, Portland, Oregon, US Army Corps of Engineers Committee on Water Quality, Washington, DC.

Wilhelms, S. C., Schneider, M. L., and Howington, S. E. 1987. "Improvement of Hydropower Release Dissolved Oxygen with Turbine Venting," Technical Report in preparation, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

Wilhelms, S. C., and Smith, D. R. 1981. "Reaeration Through Gated-Conduit Outlet Works," Technical Report E-81-5, US Army Engineer Waterways Experiment Station, Vicksburg, Miss.

APPENDIX A: INPUT FORMAT DESCRIPTION

Notes on Input Format

1. CARD refers to card image input if disc files or magnetic tapes are used as input devices.

2. The format code following the CARD number is the format by which the input line of data will be read. The input must match the fields of the appropriate format.

3. Following the CARD-FORMAT line is the list of variables used to store the input data. They are listed in the order by which they are entered. Following each variable name is the expected input for the first word(s) of the input line. When several expected input phrases are listed, only one is chosen.

4. Each time more than one card is required to input profile values, the remaining input cards will need to be renumbered to reflect the extra cards used. For example, when profile data are to be entered as one elevation value and one parameter value per line, such as CARD #25 when TABTYP = 1 , more than one profile data card will be required to define the profile. If, as a simple case, it is assumed four profile data input cards will be necessary, they would be numbered CARD #25, CARD #26, CARD #27, and CARD #28. The remaining input cards should be resequenced beginning with CARD #29.

5. Be sure that all input data units are consistent with the system of units specified on CARD #06. The exception to this is the units for density, which should always be entered as grams per cubic centimeter.

6. There must be a STOP statement at the end of each data set.

SELECT Program-Input Format Description

CARD #01 FORMAT (20A4)
1. TITLE - An arbitrary title not to exceed 80 characters in length

CARD #02 FORMAT (A4, 6X, 14I5)
1. CHECK - "FILES"
2. IFILE - Input File Code
3. KFILE - Output File Code for Printers
4. JFILE - Auxiliary Input File Code used for program debugging (enter 00)

CARD #03 FORMAT (A4, 6X, 14I5)
1. CHECK - "DATA"
2. NSETS - Number of data sets in input file

CARD #04 FORMAT (20A4)
1. CHECK - A) "PRINT INPUT" - Echo print input
B) "(ANYTHING ELSE)" - No echo print

CARD #05 FORMAT (20A4)
1. HEADING - An arbitrary descriptive heading for the simulation, not to exceed 80 characters in length

CARD #06 FORMAT (20A4)
1. CHECK - A) "METRIC" - Input and output are in Metric (SI) units
B) "ENGLISH" - Input and output are in English (non-SI) units

CARD #07 FORMAT (A4, 6X, 14I5)
1. CHECK - "TABLES"
2. TABTYP - A) "1" - Program reads profile input data in a table of one value of elevation, depth below the surface, or height above the bottom, and a corresponding value of density, temperature, or quality per line.

B) "(ANY OTHER NUMBER)" - Program reads profile data as a table of all the values (eight per line) of elevation, depth below the surface, or height above the bottom followed by a table of all the values of the corresponding density, temperature, or quality.

CARD #08 FORMAT (A4, 6X (7F10.0))
1. CHECK - "THICKNESS"
2. DELZ - Thickness of each computational layer (feet or meters)

CARD #09 FORMAT (A4, 6X, 14I5)
1. CHECK - "INTERVAL"
2. INTER - The interval used to select layer information for printout

CARD #10 FORMAT (A4, 6X (7F10.0))
1. CHECK - "SURFACE"
2. SURFACE - Elevation of the water surface (feet or meters)

CARD #11 FORMAT (A4, 6X (7F10.0))
1. CHECK - "BOTTOM"
2. BOTTOM - Elevation of the lake bottom (feet or meters)

CARD #12 FORMAT (A4, 6X, 14I5)
1. CHECK - A) "PORTS"
B) "WEIR"
2. NPORTS - Number of ports (not applicable if weir only)

If "WEIR" was specified on Card #12, go to Card #19; otherwise continue.

CARD #13 FORMAT (A4, 6X (7F10.0))
1. CHECK - "VDIM"
2. (PVDIM(K), K=1, NPORTS) - Vertical dimensions of the ports (feet or meters)

CARD #14 FORMAT (A4, 6X (7F10.0))
1. CHECK - "HDIM"
2. (PHDIM(K), K=1, NPORTS) - Horizontal dimensions of the ports (feet or meters)

CARD #15 FORMAT (A4, 6X (7F10.0))
1. CHECK - A) "ELEVATION"
B) "DEPTH"
C) "HEIGHT"
2. (PHGT(K), K=1, NPORTS) - Center-line elevation, depth or height of the ports (feet or meters)

CARD #16 FORMAT (A4, 6X (7F10.0))
1. CHECK - "FLOW"
2. (FLOW(K), K=1, NPORTS) - Flow through each of the ports (cubic feet per second or cubic meters per second)

CARD #17 FORMAT (A4, 6X, (7F10.0))
1. CHECK - "ANGLE"
2. (WTHETA(K), K=1, NPORTS) - The withdrawal angle for each port (radians)

If "PORTS" was specified on Card #12 and a weir is to be modeled also, use CARDS #18, #19, and #20; otherwise omit them.

CARD #18 FORMAT (A4, 6X, 14I5)
1. CHECK - "WEIR"

CARD #19 FORMAT (A4, 6X, 14I5)
1. CHECK - A) "FREE"
B) "SUBMERGED"

If "SUBMERGED" was specified on Card #19, omit Card #20; otherwise continue.

CARD #20 FORMAT (A4, 6X (7F10.0))
1. CHECK - "COEF"
2. COEF - Coefficient of discharge for free weir flow. These coefficients should be 3.0, 3.33, or 4.10.

CARD #21 FORMAT (A4, 6X (7F10.0))
1. CHECK - "LENGTH"
2. WRLNG - Length of the weir crest (feet or meters)

CARD #22 FORMAT (A4, 6X (7F10.0))
1. CHECK - A) "ELEVATION"
B) "DEPTH"
C) "HEIGHT"
2. WRHGT - Elevation, depth, or height of the weir crest (feet per second or meters per second)

CARD #23 FORMAT (A4, 6X (7F10.0))
1. CHECK - "FLOW"
2. WRFLOW - Flow over the weir (cubic feet per second or cubic meters per second)

Optional CARD FORMAT (20A4)
1. CHECK - A) "TURBINE VENTING" - If turbine venting is to be modeled
B) "GATED STRUCTURE" - If aeration due to gated conduit is to be modeled

If "GATED STRUCTURE" is entered, so must the following optional card.

Optional CARD FORMAT (A4, 6X, A4 6X, (6F10.0))
1. CHECK1 - "TAILWATER"
2. CHECK2 - A) "ELEVATION" - If tailwater elevation
B) "FUNCTION" - If tailwater function
3. TWEL - Value of the tailwater elevation

CARD #24 FORMAT (A4, 6X, A4, 6X, 12I5)
1. CHECK1 - "NUMBER OF"
2. CHECK2 - A) "DENSITIES"
B) "TEMPS"

3. NUMD - A) Number of density values in profile, or
B) Number of temperature values in profile

If "TEMPS" was specified on Card #24, go to Card #30; otherwise continue.

If TABTYP EQ. 1 (Card #07)

CARD #25 FORMAT (A4, 6X, A4)
1. CHECK1 - A) "ELEVATION"
B) "DEPTH"
C) "HEIGHT"
2. CHECK2 - "DENSITIES"

CARD #26 FORMAT (2F10.0)
1. (YD(M), DEN(M), M=1, NUMD) - Values of elevation, depth, or height (feet or meters) and the corresponding values of density (grams per cubic centimeter)

CARDS #27 and #28 not used

An adequate number of cards containing the profile elevation and density values using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

If TABTYP. NE. 1 (Card #07)

CARD #25 FORMAT (20A4)
CHECK - A) "ELEVATION"
B) "DEPTH"
C) "HEIGHT"

CARD #26 FORMAT (8F10.0)
1. (YD(M), M=1, NUMD) - Values of elevation, depth, or height corresponding with density values to be input (feet or meters)

An adequate number of cards containing profile elevation values using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

CARD #27 FORMAT (A4, 6X, 14I5)
1. CHECK - "DENSITY"

CARD #28 FORMAT (8F10.0)
1. (DEN(M), M=1, NUMD) - Values of density corresponding to the data on Card #26 (grams per cubic centimeters)

An adequate number of cards containing profile density values using the above format are required at this point. The remaining input cards should be renumbered to continue in this sequence.

CARD #29 FORMAT (A4, 16X, 12I5)

1. CHECK - A) "NUMBER OF TEMPS"
   B) "QUALITIES"
   C) "STOP" - Read sequence stops
2. NUMT - Number of temperature if "NUMBER OF TEMPS" is entered

If "STOP" was specified on Card #29, go to last line of this description (page A8).

If "QUALITIES" was specified on Card #29, go to Card #36.

If "NUMBER OF TEMPS" was specified on Card #29, continue.

CARD #30 FORMAT (A4, 16X, A4)

1. CHECK - A) "TEMPERATURE DEGREES"
2. UNITS - A) "FAHRENHEIT"
   B) "CENTIGRADE"

If TABTYP EQ. 1 (Card #07)

CARD #31 FORMAT (A4, 6X, A4)

1. CHECK1 - A) "ELEVATION"
   B) "DEPTH"
   C) "HEIGHT"
2. CHECK2 - "TEMPERATURE"

CARD #32 FORMAT (2F10.0)

1. (YT(M), TEMP(M), M=1, NUMT) - Values of elevation, depth, or height (feet or meters) and the corresponding values of temperature (units specified on Card #30)

CARDS #33 and #34 not used

An adequate number of cards containing the profile elevation and temperature values using the above format are required at this point. The remaining input cards should be renumbered to continue in this sequence.

If TABTYP NE. 1 (Card #07)

CARD #31 FORMAT (20A4)

1. CHECK - A) "ELEVATION"
   B) "DEPTH"
   C) "HEIGHT"

CARD #32 FORMAT (8F10.0)

1. (TY(M), M=1, NUMT) - Values of elevation, depth, or height corresponding with temperature values to be input (feet or meters)

An adequate number of cards containing profile elevations using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

CARD #33 FORMAT (20A4)

1. CHECK - "TEMPERATURE"

CARD #34 FORMAT (8F10.0)

1. (TEMP(M), M=1, NUMT) - Values of temperature corresponding to the data on Card #32 (units specified on Card #30)

An adequate number of cards containing profile temperatures using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

CARD #35 FORMAT (A4, 6X, 14I5)

1. CHECK - A) "QUALITIES"
   B) "STOP" - Read sequence stops
2. NQUAL - Number of quality parameters

If STOP was specified on CARD #35, go to last line of this description (page A8). If QUALITIES was specified on CARD #35, continue.

CARD #36 FORMAT (A4, 6X, 5A4, I5)

1. CHECK - "NUMBER OF"
2. (NAME(NM, I) NM=1,5) - Name of the Ith quality parameter (not to exceed 20 characters)
3. NUMQ(I) - Number of quality(I) values

If TABTYP EQ. 1 (Card #07)

CARD #37 FORMAT (A4, 6X, A4)

1. CHECK - A) "ELEVATION"
   B) "DEPTH"
   C) "HEIGHT"

CARD #38 FORMAT (2F10.0)

1. (YQ(I,M), QUAL(I,M), M=1, NUMQ(I)) - Values of elevation, depth, or height (feet or meters) and the corresponding values of the Ith quality parameter (appropriate units)

CARDS #39 and #40 not used

An adequate number of cards containing the profile elevation and quality values using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

If TABTYP NE. 1 (Card #07)

CARD #37 FORMAT (20A4)
1. CHECK - A) "ELEVATION
B) "DEPTH"
C) "HEIGHT"

CARD #38 FORMAT (8F10.0)
1. (YQ(I,M) M=1, NUMQ(I)) - Values of elevation, depth, or height corresponding with the Ith quality parameter values to be input (feet or meters)

An adequate number of cards containing the profile elevations using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

CARD #39 FORMAT (20A4)
1. CHECK - Name of the Ith quality parameter

CARD #40 FORMAT (8F10.0)
1. (QUAL(I,M), M=1, NUMQ(I)) - Values of the Ith quality parameter corresponding to the data on Card #38 (appropriate units)

An adequate number of cards containing the profile qualities using the above format are required at this point. The remaining input cards should be renumbered to continue in sequence.

If there are more water quality profiles, repeat CARDS #37-40 for each additional water quality parameter.

CARD #41 FORMAT (20A4)
1. CHECK - "STOP" - Stops read sequence

If there are more data sets, repeat CARDS #05-41 for each additional data set.

APPENDIX B: INPUT FILE EXAMPLES

1. The following are examples of simple input files. They are presented to help clarify the structure of an input file.

2. The impoundment being modeled has a pool that is 100 ft deep, with the elevation of the bottom taken as zero or datum.

Example File 1

3. Example File 1 models a single 5- by 5-ft port whose center line is located 20 ft below the pool surface. The port has a withdrawal angle of 180 deg (3.14 rad) and releases a flow rate of 100 cfs (2.8 cu m/sec).

4. The density profile needed to run SELECT will be generated by the program using the given temperature profile. The only water quality parameter modeled is temperature.

5. For the computations, the pool will be divided into layers 3 ft thick each. The printout interval for layer information will be at every layer (INTERVAL = 01).

```
ANONYMOUS LAKE EXAMPLE
FILES          05   06
DATA SETS      01
PRINT INPUT
EXAMPLE FOR INPUT/OUTPUT USING PORT
ENGLISH
TABLE          01
THICKNESS          3.0
INTERVAL       01
SURFACE          100.0
BOTTOM             0.0
PORT            1.
VDIM               5.0
HDIM               5.0
DEPTH             20.0
FLOW             100.0
ANGLE             3.14
NUMBER OF TEMP          12
TEMPERATURE DEGREES CENTIGRADE
HEIGHT     TEMP
      97.6       28.9
      90.0       28.2
      80.0       27.0
      69.8       26.0
      65.0       24.9
      60.0       23.0
      57.6       20.0
      53.9       18.5
      50.0       17.5
      40.0       16.7
      31.2       15.9
      20.0       15.0
STOP
```

Example File 2

6. Example File 2 models the same thing as Example File 1 with a few variations. One variation is that another port vertically separated from the first port is to be modeled. The two ports are considered to be operating simultaneously and independently. The second port is located 50 ft below the pool surface and has the same dimensions and withdrawal characteristics as the first port.

7. The presence of the dissolved oxygen (DO) profile following the temperature profile indicates that the user wishes to know the release DO concentration as well as temperature. The calculation of release DO will include DO uptake due to gated-conduit reaeration because the GATED STRUCTURE command is present following the port information in the input file.

```
ANONYMOUS LAKE EXAMPLE
FILES          05    06
DATA SETS      01
PRINT INPUT
EXAMPLE FOR INPUT/OUTPUT USING PORT
ENGLISH
TABLE          01
THICKNESS          3.0
INTERVAL       01
SURFACE          100.0
BOTTOM             0.0
PORT            2
VDIM               5.0        5.0
HDIM               5.0        5.0
DEPTH             50.0       20.0
FLOW             100.0      100.0
ANGLE             3.14       3.14
GATED STRUCTURE
TAILWATER ELEVATION          15.9
NUMBER OF TEMP          12
TEMPERATURE DEGREES CENTIGRADE
HEIGHT     TEMP
      97.6       28.9
      90.0       28.2
      80.0       27.0
      69.8       26.0
      65.0       24.9
      60.0       23.0
      57.6       20.0
      53.9       18.5
      50.0       17.5
      40.8       16.7
      31.2       15.9
      20.0       15.0
QUALITIES      1
NUMBER OF DISSOLVED OXYGEN        13
HEIGHT     DISSOLVED OXYGEN
      95.6        3.2
      90.2        3.2
      84.9        3.1
      78.7        3.0
      73.4        2.1
      69.5        1.8
      61.0        1.3
      58.4        0.5
      50.3        0.3
      40.6        0.2
      30.0        0.2
      21.7        0.1
      15.4        0.0
 STOP
```

Example File 3

8. Example File 3 models a weir. The weir is the submerged type with its crest 65 ft off the bottom. It is 100 ft in length and has a total flow rate of 200 cfs (5.6 cu m/sec) over its crest.

9. The input file dictates that the pool should be divided into layers that are 3 ft thick and that the output data be given at every layer (INTERVAL = 01). As in Example File 2, the temperature and DO concentration of the release are desired. There is no option for DO calculations based on reaeration through the outlet works for weir flow.

```
ANONYMOUS LAKE EXAMPLE
FILES          05   06
DATA SETS      01
PRINT INPUT
EXAMPLE FOR INPUT/OUTPUT USING A WEIR
ENGLISH
TABLE          01
THICKNESS          3.0
INTERVAL       01
SURFACE          100.0
BOTTOM             0.0
WEIR
SUBMERGED
LENGTH            100.
HEIGHT             65.0
FLOW              200.0
NUMBER OF TEMP          12
TEMPERATURE DEGREES CENTIGRADE
HEIGHT     TEMP
      97.6       28.9
      90.0       28.2
      80.0       27.0
      69.8       26.0
      65.0       24.9
      60.0       23.0
      57.6       20.0
      53.9       18.5
      50.0       17.5
      40.0       16.7
      31.2       15.9
      20.0       15.0
QUALITIES      1
NUMBER OF DISSOLVED OXYGEN        13
HEIGHT     DISSOLVED OXYGEN
      95.6        3.2
      90.2        3.2
      84.9        3.1
      78.7        3.0
      73.4        2.1
      69.5        1.8
      61.0        1.3
      58.4        0.5
      50.3        0.3
      40.6        0.2
      30.0        0.2
      21.7        0.1
      15.4        0.0
STOP
```

Example File 4

10. Example File 4 is an input file made up of Example Files 1 and 3. In other words, it is one input file containing two data sets. It is important to realize that this data file format will not cause the program to model the port and the weir simultaneously; rather, it will model the port only and produce output. Then it will model the weir only and produce output.

11. This technique of appending data sets saves time in loading and reloading the program for consecutive runs.

```
ANONYMOUS LAKE EXAMPLE
FILES          05    06
DATA SETS      02
PRINT INPUT
EXAMPLE FOR INPUT/OUTPUT USING PORT
ENGLISH
TABLE          01
THICKNESS          3.0
INTERVAL       01
SURFACE          100.0
BOTTOM             0.0
PORT            1
VDIM               5.0
HDIM               5.0
DEPTH             20.0
FLOW             100.0
ANGLE             3.14
NUMBER OF TEMP          12
TEMPERATURE DEGREES CENTIGRADE
HEIGHT     TEMP
      97.6       28.9
      90.0       28.2
      80.0       27.0
      69.8       26.0
      65.0       24.9
      60.0       23.0
      57.6       20.0
      53.9       18.5
      50.0       17.5
      40.0       16.7
      31.2       15.9
      20.0       15.0
STOP
EXAMPLE FOR INPUT/OUTPUT USING A WEIR
ENGLISH
TABLE          01
THICKNESS          3.0
INTERVAL       01
SURFACE          100.0
BOTTOM             0.0
WEIR
SUBMERGED
LENGTH            100.
HEIGHT             65.0
FLOW             100.0
NUMBER OF TEMP          12
TEMPERATURE DEGREES CENTIGRADE
HEIGHT     TEMP
      97.6       28.9
      90.0       28.2
      80.0       27.0
      69.8       26.0
      65.0       24.9
      60.0       23.0
      57.6       20.0
      53.9       18.5
      50.0       17.5
      40.0       16.7
      31.2       15.9
      20.0       15.0
QUALITIES       1
NUMBER OF DISSOLVED OXYGEN        13
HEIGHT     DISSOLVED OXYGEN
      95.6        3.2
      90.2        3.2
      84.9        3.1
      78.7        3.0
      73.4        2.1
      69.5        1.8
      61.0        1.3
      58.4        0.5
      50.3        0.3
      40.6        0.2
      30.0        0.2
      21.7        0.1
      15.4        0.0
STOP
```

APPENDIX C: OUTPUT EXAMPLE

1. This appendix presents typical SELECT output, which is divided as follows:

a. Listing of the input file. In this example, the PRINT INPUT command was included in the input file. If the command is omitted from the input file, no listing of the input file is given.

b. Summary of the output results. The summary includes the title of the input file, the title of the data set to which the results pertain, the units used for computations, each outlet's dimensions and flow rate, the total flow rate from all outlets, the locations of the upper and lower withdrawal limits, and the total outflow concentration of all water quality parameters modeled.

c. Tabular listing of the flow rates, velocities, and water quality parameter concentrations, given at layer centerline elevations. The interval between the layers at which information is given depends on the INTERVAL command listed in the input file. An interval of "1" will list layer information at every layer, an interval of "2" will list layer information at every other layer, and so on.

d. Line-printer plot of the velocity and density profile. The plot is a user option and is a result of inserting the PLOT OUTPUT command in the input file. If the command is omitted, no plot will be given. The data points on the plot coincide with the interval between layers chosen to list tabular data as in c above.

2. If the user chooses to use several data sets in one input file, the results from each data set will be listed individually and output one after the other in the results.

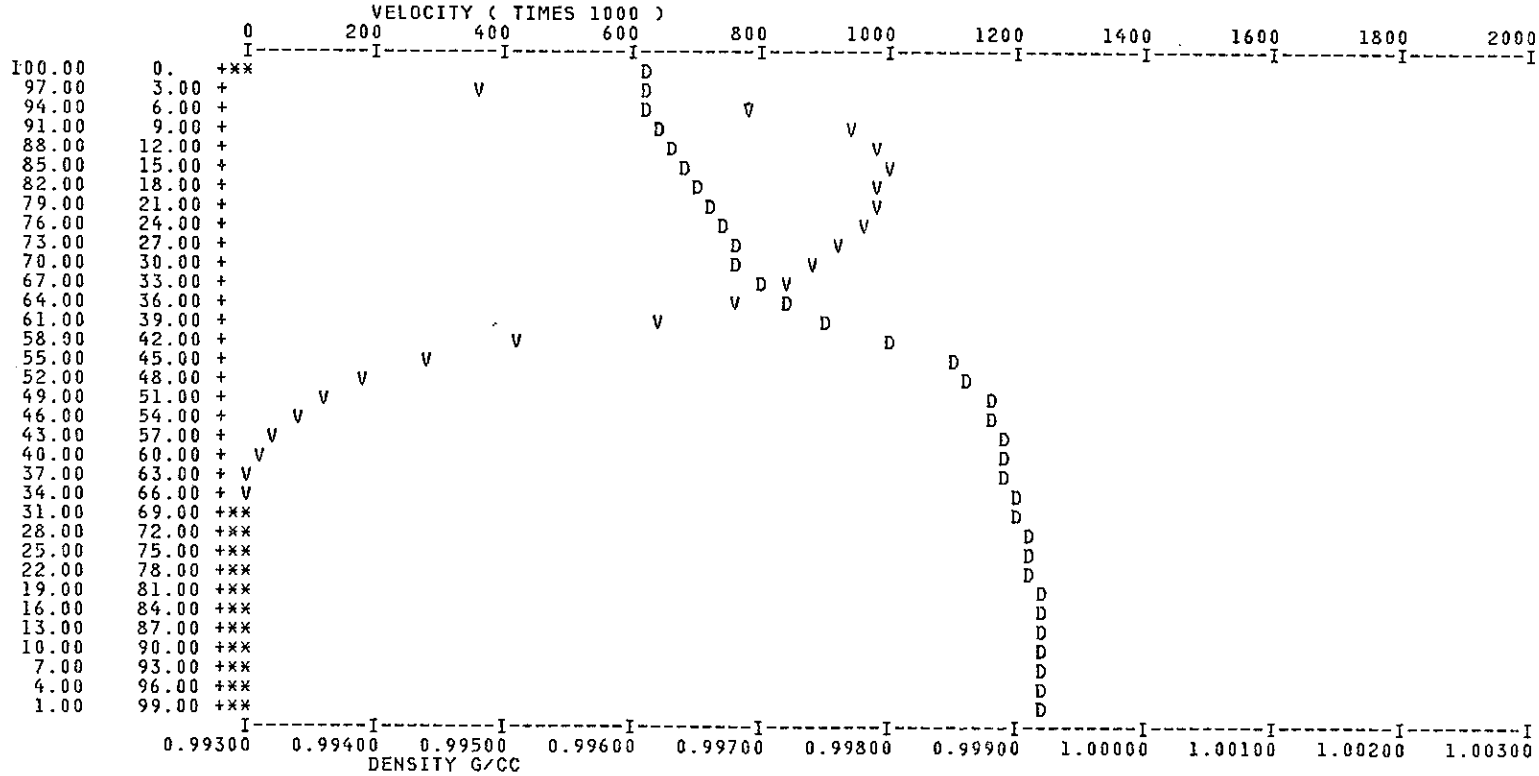
```
1000        ***ANONYMOUS LAKE EXAMPLE                                                  00000010
1010        ***FILES         05   06                                                   00000020
1020        ***DATA SETS     02                                                        00000030
1030        ***PRINT INPUT                                                             00000040
1040        ***EXAMPLE INPUT/OUTPUT USING PORT                                         00000050
1050        ***ENGLISH                                                                 00000060
1060        ***TABLE         01                                                        00000070
1070        ***THICKNESS         3.0                                                   00000080
1080        ***INTERVAL      01                                                        00000090
1090        ***SURFACE         100.0                                                   00000100
1100        ***BOTTOM            0.0                                                   00000110
1110        ***PORT           1                                                        00000160
1120        ***VDIM             20.0                                                   00000170
1130        ***HDIM             40.0                                                   00000180
1140        ***DEPTH            20.0                                                   00000190
1150        ***FLOW           4000.0                                                   00000200
1160        ***GATED STRUCTURE                                                         00000202
1170        ***TAILWATER ELEVATION       15.9                                          00000204
1180        ***NUMBER OF TEMP         12                                               00000210
1190        ***TEMPERATURE DEGREES CENTIGRADE                                          00000220
1200        ***HEIGHT    TEMP                                                          00000230
1210        ***      97.6       28.9                                                   00000240
1220        ***      90.0       28.2                                                   00000250
1230        ***      80.0       27.0                                                   00000260
1240        ***      69.8       26.0                                                   00000270
1250        ***      65.0       24.9                                                   00000280
1260        ***      60.0       23.0                                                   00000290
1270        ***      57.6       20.0                                                   00000300
1280        ***      53.9       18.5                                                   00000310
1290        ***      50.0       17.5                                                   00000320
1300        ***      40.0       16.7                                                   00000330
1310        ***      31.2       15.9                                                   00000340
1320        ***      20.0       15.0                                                   00000350
1330        ***QUALITIES      1                                                        00000370
1340        ***NUMBER OF DISSOLVED OXYGEN        13                                   00000380
1350        ***HEIGHT    DISSOLVED OXYGEN                                              00000385
1360        ***      95.6        3.2                                                   00000390
1370        ***      90.2        3.2                                                   00000400
1380        ***      84.9        3.1                                                   00000410
1390        ***      78.7        3.0                                                   00000420
1400        ***      73.4        2.1                                                   00000430
1410        ***      69.5        1.8                                                   00000440
1420        ***      61.0        1.3                                                   00000450
1430        ***      58.4        0.5                                                   00000460
1440        ***      50.3        0.3                                                   00000470
1450        ***      40.6        0.2                                                   00000480
1460        ***      30.0        0.2                                                   00000490
1470        ***      21.7        0.1                                                   00000500
1480        ***      15.4        0.0                                                   00000510
1490        ***STOP                                                                    00000530
1500        ***EXAMPLE INPUT/OUTPUT USING A WEIR                                       00000540
1510        ***ENGLISH                                                                 00000550
1520        ***TABLE         01                                                        00000560
1530        ***THICKNESS         3.0                                                   00000570
1540        ***INTERVAL      01                                                        00000580
1550        ***SURFACE         100.0                                                   00000590
1560        ***BOTTOM            0.0                                                   00000600
1570        ***WEIR                                                                    00000650
1580        ***SUBMERGED                                                               00000660
```

```
1590        ***LENGTH          100.                                    00000670
1600        ***HEIGHT          65.0                                    00000680
1610        ***FLOW          4000.0                                    00000690
1620        ***NUMBER OF TEMP         12                               00000700
1630        ***TEMPERATURE DEGREES CENTIGRADE                          00000000
1640        ***HEIGHT    TEMP                                          00000710
1650        ***       97.6       28.9                                  00000720
1660        ***       90.0       28.2                                  00000730
1670        ***       80.0       27.0                                  00000740
1680        ***       69.8       26.0                                  00000750
1690        ***       65.0       24.9                                  00000760
1700        ***       60.0       23.0                                  00000770
1710        ***       57.6       20.0                                  00000780
1720        ***       53.9       18.5                                  00000790
1730        ***       50.0       17.5                                  00000800
1740        ***       40.8       16.7                                  00000810
1750        ***       31.2       15.9                                  00000820
1760        ***       20.0       15.0                                  00000830
1770        ***QUALITIES     1                                         00000850
1780        ***NUMBER OF DISSOLVED OXYGEN        13                   00000860
1790        ***HEIGHT    DISSOLVED OXYGEN                              00000870
1800        ***       95.6        3.2                                  00000880
1810        ***       90.2        3.2                                  00000890
1820        ***       84.9        3.1                                  00000900
1830        ***       78.7        3.0                                  00000910
1840        ***       73.4        2.1                                  00000920
1850        ***       69.5        1.8                                  00000930
1860        ***       61.0        1.3                                  00000940
1870        ***       58.4        0.5                                  00000950
1880        ***       50.3        0.3                                  00000960
1890        ***       40.6        0.2                                  00000970
1900        ***       30.0        0.2                                  00000980
1910        ***       21.7        0.1                                  00000990
1920        ***       15.4        0.0                                  00001000
1930        ***STOP                                                    00001020
```

```
ANONYMOUS LAKE EXAMPLE


EXAMPLE INPUT/OUTPUT USING PORT

            UNITS ARE FEET

            PORT ELEVATION     80.000
            PORT VERTICAL DIMENSION      20.000
            DISCHARGE, VOLUME FLOW PER SEC.    4000.0000


            TOTAL DISCHARGE, VOLUME PER SEC    4000.0000

            LOWER WITHDRAWAL LIMIT, HEIGHT ABOVE BOTTOM     34.375    ELEVATION     34.375
            UPPER WITHDRAWAL LIMIT, HEIGHT ABOVE BOTTOM    100.000    ELEVATION    100.000
            OUTFLOW DENSITY  0.99675  G/CC
            OUTFLOW TEMPERATURE   26.10
            OUTFLOW CONCENTRATION OF DISSOLVED OXYGEN          8.14
```

ELEVATION	DEPTH	DENSITY	NORM. VEL.	FLOW	TEMPERATURE	DISSOLVED OX
99.500	0.50	0.99600	0.	0.	28.90	3.20
97.500	2.50	0.99601	0.3685	119.5295	28.89	3.20
94.500	5.50	0.99603	0.7951	257.8941	28.61	3.20
91.500	8.50	0.99617	0.9571	310.4569	28.34	3.20
88.500	11.50	0.99626	0.9976	323.5691	28.02	3.17
85.500	14.50	0.99636	1.0000	324.3637	27.66	3.11
82.500	17.50	0.99646	0.9946	322.6182	27.30	3.06
79.500	20.50	0.99656	0.9806	318.0696	26.95	3.01
76.500	23.50	0.99664	0.9602	311.4685	26.66	2.63
73.500	26.50	0.99672	0.9332	302.6989	26.36	2.12
70.500	29.50	0.99680	0.8999	291.8915	26.07	1.88
67.500	32.50	0.99695	0.8409	272.7727	25.47	1.68
64.500	35.50	0.99715	0.7606	246.7125	24.71	1.51
61.500	38.50	0.99743	0.6443	208.9817	23.57	1.33
58.500	41.50	0.99799	0.4400	142.7130	21.13	0.53
55.500	44.50	0.99841	0.2841	92.1540	19.15	0.43
52.500	47.50	0.99860	0.1920	62.2875	18.14	0.35
49.500	50.50	0.99872	0.1249	40.5051	17.46	0.29
46.500	53.50	0.99877	0.0823	26.7107	17.22	0.26

43.500	56.50	0.99881	0.0479	15.5320	16.98	0.23
40.500	59.50	0.99885	0.0222	7.2021	16.74	0.20
37.500	62.50	0.99889	0.0058	1.8687	16.47	0.20
34.500	65.50	0.99894	0.0000	0.0002	16.20	0.20
31.500	68.50	0.99898	0.	0.	15.93	0.20
28.500	71.50	0.99902	0.	0.	15.68	0.18
25.500	74.50	0.99906	0.	0.	15.44	0.15
22.500	77.50	0.99910	0.	0.	15.20	0.11
19.500	80.50	0.99913	0.	0.	15.00	0.07
16.500	83.50	0.99913	0.	0.	15.00	0.02
13.500	86.50	0.99913	0.	0.	15.00	0.
10.500	89.50	0.99913	0.	0.	15.00	0.
7.500	92.50	0.99913	0.	0.	15.00	0.
4.500	95.50	0.99913	0.	0.	15.00	0.
1.500	98.50	0.99913	0.	0.	15.00	0.


VELOCITY (TIMES 1000)
0 200 400 600 800 1000 1200 1400 1600 1800 2000
100.00 0.
97.00 3.00
94.00 6.00
91.00 9.00
88.00 12.00
85.00 15.00
82.00 18.00
79.00 21.00
76.00 24.00
73.00 27.00
70.00 30.00
67.00 33.00
64.00 36.00
61.00 39.00
58.00 42.00
55.00 45.00
52.00 48.00
49.00 51.00
46.00 54.00
43.00 57.00
40.00 60.00
37.00 63.00
34.00 66.00
31.00 69.00
28.00 72.00
25.00 75.00
22.00 78.00
19.00 81.00
16.00 84.00
13.00 87.00
10.00 90.00
7.00 93.00
4.00 96.00
1.00 99.00
0.99300 0.99400 0.99500 0.99600 0.99700 0.99800 0.99900 1.00000 1.00100 1.00200 1.00300
DENSITY G/CC

ANONYMOUS LAKE EXAMPLE

EXAMPLE INPUT/OUTPUT USING A WEIR

UNITS ARE FEET

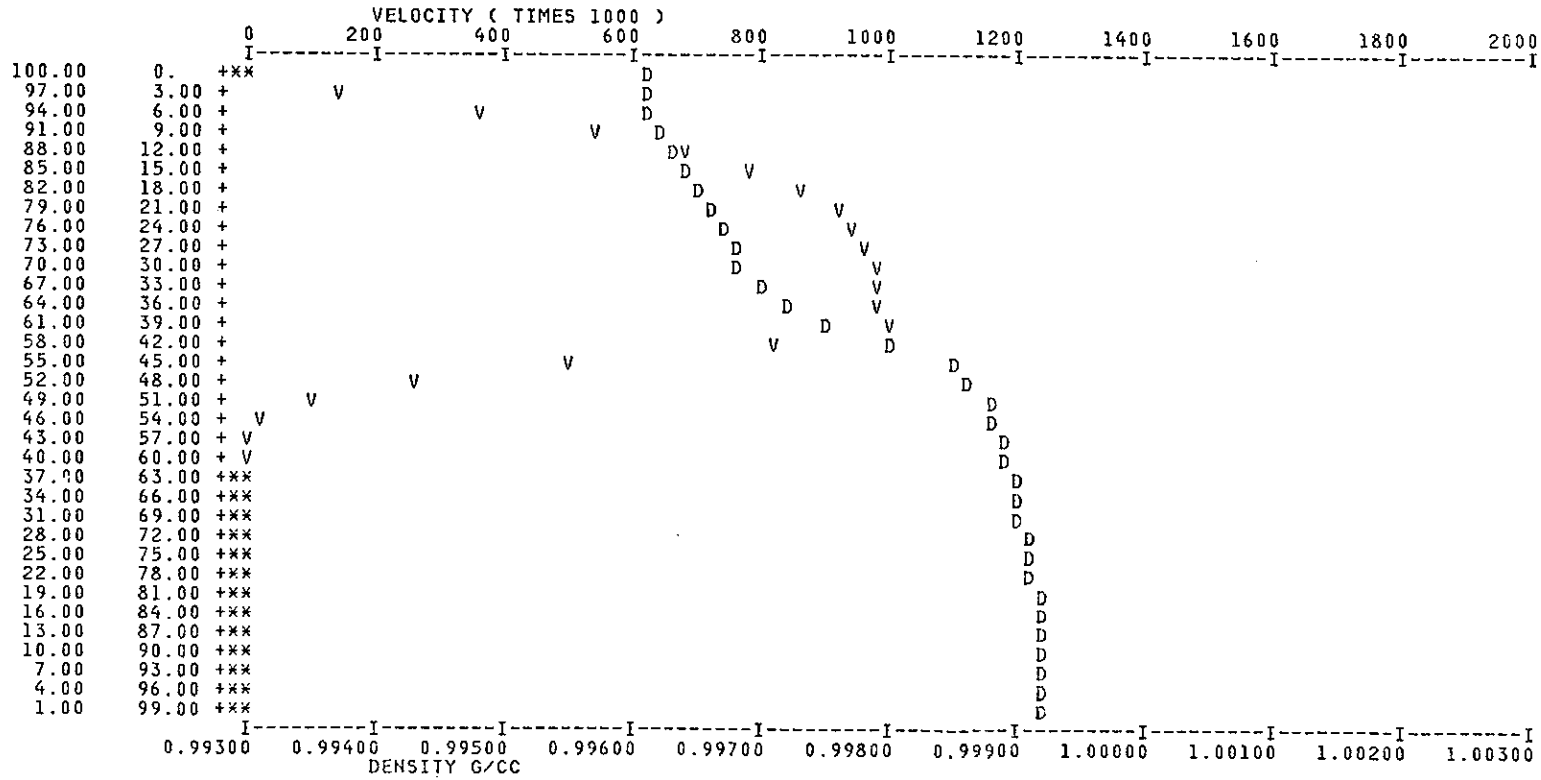
WEIR CREST ELEVATION 65.000
WEIR LENGHT 100.000
DISCHARGE, VOLUME FLOW PER SEC. 4000.0000

TOTAL DISCHARGE, VOLUME PER SEC 4000.0000

LOWER WITHDRAWAL LIMIT, HEIGHT ABOVE BOTTOM 40.117 ELEVATION 40.117
UPPER WITHDRAWAL LIMIT, HEIGHT ABOVE BOTTOM 100.000 ELEVATION 100.000
OUTFLOW DENSITY 0.99693 G/CC
OUTFLOW TEMPERATURE 25.43
OUTFLOW CONCENTRATION OF DISSOLVED OXYGEN 2.10

ELEVATION	DEPTH	DENSITY	NORM. VEL.	FLOW	TEMPERATURE	DISSOLVED OX
99.500	0.50	0.99600	0.	0.	28.90	3.20
97.500	2.50	0.99601	0.1507	50.1277	28.89	3.20
94.500	5.50	0.99609	0.3682	122.4636	28.61	3.20
91.500	8.50	0.99617	0.5409	179.8818	28.34	3.20
88.500	11.50	0.99626	0.6824	226.9224	28.02	3.17
85.500	14.50	0.99636	0.7930	263.7110	27.66	3.11
82.500	17.50	0.99646	0.8716	289.8517	27.30	3.06
79.500	20.50	0.99656	0.9246	307.5002	26.95	3.01
76.500	23.50	0.99664	0.9575	318.4385	26.66	2.63
73.500	26.50	0.99672	0.9784	325.3852	26.36	2.12
70.500	29.50	0.99680	0.9906	329.4421	26.07	1.88
67.500	32.50	0.99695	0.9978	331.8305	25.47	1.68
64.500	35.50	0.99715	0.9998	332.5085	24.71	1.51
61.500	38.50	0.99743	1.0000	332.5589	23.57	1.33
58.500	41.50	0.99799	0.8309	276.3341	21.13	0.53
55.500	44.50	0.99841	0.5158	171.5427	19.15	0.43
52.500	47.50	0.99860	0.2717	90.3624	18.14	0.35
49.500	50.50	0.99872	0.1124	37.3308	17.46	0.29
46.500	53.50	0.99877	0.0363	12.0827	17.20	0.26

43.500	56.50	0.99882	0.0050	1.6752	16.93	0.23
40.500	59.50	0.99886	0.0000	0.0000	16.67	0.20
37.500	62.50	0.99890	0.	0.	16.42	0.20
34.500	65.50	0.99894	0.	0.	16.17	0.20
31.500	68.50	0.99898	0.	0.	15.92	0.20
28.500	71.50	0.99902	0.	0.	15.68	0.18
25.500	74.50	0.99906	0.	0.	15.44	0.15
22.500	77.50	0.99910	0.	0.	15.20	0.11
19.500	80.50	0.99913	0.	0.	15.00	0.07
16.500	83.50	0.99913	0.	0.	15.00	0.02
13.500	86.50	0.99913	0.	0.	15.00	0.
10.500	89.50	0.99913	0.	0.	15.00	0.
7.500	92.50	0.99913	0.	0.	15.00	0.
4.500	95.50	0.99913	0.	0.	15.00	0.
1.500	98.50	0.99913	0.	0.	15.00	0.


VELOCITY (TIMES 1000)
0 200 400 600 800 1000 1200 1400 1600 1800 2000
100.00 0.
97.00 3.00
94.00 6.00
91.00 9.00
88.00 12.00
85.00 15.00
82.00 18.00
79.00 21.00
76.00 24.00
73.00 27.00
70.00 30.00
67.00 33.00
64.00 36.00
61.00 39.00
58.00 42.00
55.00 45.00
52.00 48.00
49.00 51.00
46.00 54.00
43.00 57.00
40.00 60.00
37.00 63.00
34.00 66.00
31.00 69.00
28.00 72.00
25.00 75.00
22.00 78.00
19.00 81.00
16.00 84.00
13.00 87.00
10.00 90.00
7.00 93.00
4.00 96.00
1.00 99.00
0.99300 0.99400 0.99500 0.99600 0.99700 0.99800 0.99900 1.00000 1.00100 1.00200 1.00300
DENSITY G/CC

APPENDIX D: ERROR CODES

1. SELECT has various internal checks that test for errors in the input file data and format, and for errors in some of the program's internal computations. When a check is failed, the program prints an appropriate error message and terminates program execution.

2. The error message contains an error number, a subroutine name, and a statement regarding the type of error. With this information, the line at which the error occurred (and the cause) can often be easily determined by the user through the use of the program code. It is suggested that the user locate the subroutine in which the error occurred and then scan the CALL ERROR () statements therein for the error number. (The error number is the first number in the ERROR argument list.) The appropriate CALL ERROR () statement should be near the line(s) of code in which the failure occurred. NOTE: Most errors (error numbers 1000-1380 excluding 1080 and 1345) are due to input file format problems. If the user determines that the error is in the input file, check for missing lines, misplaced lines, misspelled words, etc., in the input file.

Error Number	Subroutine of Occurrence	Input Format Card	Explanation
1000	XREAD	02	"FILE" was expected as first four characters of input line.
1010	XREAD	03	"DATA" was expected as first four characters of input line.
1020	XREAD	06	"METR" or "ENGL" was expected as first four characters of input line.
1030	XREAD	07	"TABL" was expected as first four characters of input line.
1040	XREAD	08	"THIC" was expected as first four characters of of input line.
1050	XREAD	09	"INTE" was expected as first four characters of input line.
1060	XREAD	10	"SURF" was expected as first four characters of input line.
1070	XREAD	11	"BOTT" was expected as first four characters of input line.
1080	XREAD	08	The number of computational layers dimensioned in the model has been exceeded. Probable solution is to increase the desired thickness of each layer.
1100	XREAD	12	"WEIR" or "PORT" was expected as first four characters of input line.

(Continued)

Error Number	Subroutine of Occurrence	Input Format Card	Explanation
1110	XREAD	13	"VDIM" was expected as first four characters of input line.
1120	XREAD	14	"HDIM" was expected as first four characters of input line.
1130	XREAD	15	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line.
1140	XREAD	16	"FLOW" was expected as first four characters of input line.
1150	XREAD	17	"ANGL" was expected as first four characters of input line.
1160	XREAD	18	"SUBM" or "FREE" was expected as first four characters of input line.
1170	XREAD	19	"COEF" was expected as first four characters of input line.
1180	XREAD	20	"LENG" was expected as first four characters of input line.
1200	XREAD	21	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line.
1210	XREAD	22	"FlOW" was expected as first four characters of input line.
1215	XREAD	23	"TAIL" was expected as first four characters of input line.

(Continued)

Error Number	Subroutine of Occurrence	Input Format Card	Explanation
1220	XREAD	23	"NUMB" was expected as first four characters of input line.
1225	XREAD	23	"TEMP" or "DENS" was expected as first four characters of the second alphanumeric format field.
1230	XREAD	24	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP not equal to 1.
1240	XREAD	26	"DENS" was expected as first four characters of input line. TABTYP not equal to 1.
1250	XREAD	24	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP not equal to 1.
1260	XREAD	24	"DENS" was expected as first four characters of the second alphanumeric format field. TABTYP not equal to 1.
1270	XREAD	28	"NUMB" or "QUAL" or "STOP" was expected as first four characters of input line.
1280	XREAD	29	"TEMP" or "STOP" was expected as first four characters of input line.
1290	XREAD	29	"FAHR" or "CENT" was expected as first four characters of the second alphanumeric format field.

(Continued)

Error Number	Subroutine of Occurrence	Input Format Card	Explanation
1300	XREAD	30	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP not equal to 1.
1310	XREAD	32	"TEMP" was expected as first four characters of input line. TABTYP not equal to 1.
1320	XREAD	30	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP equal to 1.
1330	XREAD	30	"TEMP" was expected as first four characters of the second alphanumeric format field. TABTYP equal to 1.
1340	XREAD	34	"QUAL" or "STOP" was expected as first four characters of input line.
1345	XREAD	--	Dissolved oxygen must be the first quality profile listed if aeration techniques are modeled.
1350	XREAD	35	"NUMB" was expected as first four characters of input line.
1360	XREAD	36	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP not equal to 1.
1370	XREAD	36	"DEPT" or "ELEV" or "HEIG" was expected as first four characters of input line. TABTYP equal to 1.

(Continued)

Error Number	Subroutine of Occurrence	Input Format Card	Explanation
1380	XREAD	40	"STOP" was expected as first four characters of input line.
1400	OUTVEL	16	Flow rate entered for a port was less than or equal to zero. Flow rates must be greater than zero.
1410	OUTVEL	15	Center-line height of port exceeded depth of water.
1420	VPORT	22	Flow rate for weir found to be less than or equal to zero. Flow rate must be greater than zero.
1500	VPORT	--	Convergence on the upper or lower withdrawal limit has not been reached.
1510	VPORT	--	Same as error number 1500.
1520	VPORT	--	Elevation of maximum velocity in withdrawal profile was found to be above the water surface or below the bottom.
1600	VWEIR	--	Internal computation check failure. Function FWEIR(X) must be positive at the crest elevation and negative at the bottom to allow convergence on the lower limit.
1610	VWEIR	--	Convergence on the lower withdrawal limit has not been reached.
1800	SHIFT	--	Number of iterations internally programmed for convergence on shifted limit is not sufficient.

APPENDIX E: PROGRAM CODE


```
*         PROGRAM SELECT
*
*  IT COMPUTES THE LIMITS OF THE ZONE OF WITHDRAWAL AND
*  THE DISTRIBUTION OF FLOW WITHIN THAT ZONE.  THE
*  PROGRAM ALSO COMPUTES THE OUTFLOW DENSITY AND QUALITY FROM
*  INPUT DENSITY AND QUALITY PROFILES.
*  LAST REVISION JULY 23, 1985                                          00001060
*
      COMMON / AA / QMETR, NSETS, G, HEADING(18), TITLE(18)
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT, QPWEIR
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5), PHDIM(5)
      COMMON / GG / COEF, QSUB, QQUAL
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4),
     *              QUAL(4,100), YQ(4,100)
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / MM / SUMOUT, VEL(100), FLORAT
      COMMON / NN / HGTPRT,VDIM,QTLIM,QBLIM,QSINK1,QSINK2,QFIRST
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100), VM
      COMMON / QQ / VS(100, 6), NOUTS
      COMMON / RR / ZUP(6), ZDN(6), LTOP(6), LLOW(6)
      COMMON / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      LOGICAL QDEN, QCENT
      LOGICAL QVENT, QAERA
      DATA C1, C2 / -3.9863, 508929.2 /
      DATA C3, C4 / 288.9414, 68.12963 /
*                                                                       00001330
*  DENSITIES FROM TEMPEATURES                                           00001340
*                                                                       00001350
      DENFUN ( X ) = 1. - ( X + C1 ) ** 2 / C2
     *               * ( X + C3 ) / ( X + C4 )
*                                                                       00001380
*   FAHRENHEIT TO CENTIGRADE                                            00001390
*                                                                       00001400
      TEMFUN ( T ) = ( 5. / 9. ) * ( T - 3 )
*
*  READ CONTROL INFORMATION
*
      CALL XREAD
*
*  INITIATE LOOP FOR THE NUMBER OF DATA SETS
*
      DO 130 I = 1, NSETS
*
*  READ INPUT DATA AND CONSTRUCT COMPLETE DATA TABLES
```

```
53      *
54              CALL XREAD
55      *
56      *  DEVELOP DENSITY PROFILE IF NOT GIVEN
57      *
58              IF ( QDEN ) GO TO 110
59              DO 100 J = 1, ISURF
60              IF ( .NOT. QCENT ) TEMP(J) = TEMFUN ( TEMP(J) )
61              DEN (J) = DENFUN ( TEMP(J) )
62       100    CONTINUE
63       110    CONTINUE
64      *
65      *  CHECK FOR STABLE DENSITY PROFILE
66      *
67              DO 120 K = 2, ISURF
68              IF ( DEN(K - 1) .GE. DEN(K) ) GO TO 120
69              WRITE(KFILE, 500)
70              STOP
71       120    CONTINUE
72      *
73      *  COMPUTE SELECTIVE WITHDRAWAL LIMITS AND VELOCITIES AND
74      *  RESULTANT OUTFLOW DENSITY AND QUALITIES
75      *
76              CALL OUTVEL
77      *
78      *  MODIFY OUTFLOW D.O. IF GATED STRUCTURE
79      *  AERATION IS USED.
80      *
81              IF ( QAERA ) CALL AERATE
82      *
83      *  MODIFY OUTFLOW D.O. IF TURBINE VENTING
84      *  IS USED.
85      *
86              IF ( QVENT ) CALL VENTING
87      *
88      *  PRINT RESULTS
89      *
90              CALL XPRINT
91       130    CONTINUE
92              STOP
93       500    FORMAT ( /// 10X, 27HDENSITY PROFILE UNSTABLE - ,
94          *                     16HPROGRAM STOPPED )
95              END
```

```
00000          FUNCTION DENINT ( X )
00005     *
00005     *  DETERMINE DENSITY AT ANY LOCATION
00005     *
00005          COMMON / CC / DELZ
00005          COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
00005          COMMON / LL / ISURF, HGT(100), DEPTH
00005          COMMON / NN / HGTPRT
00005          INTEGER SIGN
00005     *                                                                00002040
00005     *  DETERMINE LAYER NUMBER IN WHICH HEIGHT X EXISTS              00002050
00005     *                                                                00002060
00005          LAYER = 1. + X / DELZ
00005     *                                                                00002080
00005     *  IS HEIGHT X OUTSIDE THE POOL BOUNDARIES?                     00002090
00005     *                                                                00002100
00015          IF ( LAYER .GT. ISURF .OR. LAYER .LT. 1 ) GO TO 120
00023     *                                                                00002120
00023     *  TOLERANCE, 0.01 % OF THE LAYER THICKNESS                     00002130
00023     *                                                                00002140
00024          SMALL = 0.0001 * DELZ                                       00002150
00024     *                                                                00002151
00024     *
00024     *  FIND THE DENSITY INSIDE THE POOL
00024     *
00024     *                                                                00002190
00024     *  HEIGHT TO THE MIDPOINT OF THE LAYER OF POINT X               00002200
00024     *                                                                00002210
00027          ELMID = DELZ * ( FLOAT ( LAYER ) - 0.5 )
00033     *                                                                00002230
00033     *  DISTANCE BETWEEN POINT X AND ITS RESPECTIVE LAYERS MIDPOINT  00002240
00033     *                                                                00002250
00036          DIFF = ABS ( ELMID - X )
00044     *                                                                00002270
00044     *  ASSIGN THE DENSITY AT X THE VALUE OF ITS RESPECTIVE LAYER'S  00002280
00044     *  MIDPOINT IF  X IS IN LAYER 1 AND BELOW THE MIDPOINT; OR      00002290
00044     *  X IS IN THE TOP LAYER AND ABOVE THE MIDPOINT; OR             00002300
00044     *  X LIES VERY CLOSE TO THE MIDPOINT ITSELF                     00002310
00044     *                                                                00002320
00045          IF ( ( LAYER .EQ. 1 .AND. X .LE. ELMID ) .OR.
00065     *         ( LAYER .EQ. ISURF .AND. X .GE. ELMID ) .OR.
00065     *           DIFF .LT. SMALL ) GO TO 100
00065          GO TO 110
00066     100  CONTINUE
00066     *
00066          DENINT = DEN(LAYER)                                         00002380
00071          RETURN
00075     110  CONTINUE
00075     *
00075     *  INTERPOLATION CALCULATIONS                                   00002440
00075     *                                                                00002450
00075     *  IF SIGN IS POSITIVE, X IS BELOW THE CENTER LINE              00002460
                                                                          00002461
```

```
0075          *  IF SIGN IS NEGATIVE, X IS ABOVE THE CENTER LINE                00002462
0075          *                                                                 00002480
0075                 SIGN = ( ELMID - X ) / ABS ( ELMID - X )
0113                 IJK = - ( SIGN - 1 ) / 2
0116          *                                                                 00002510
0116          *  IJ IS THE UPPER LAYER SUBSCRIPT                                00002520
0116          *                                                                 00002530
0120                 IJ = LAYER + IJK
0120          *                                                                 00002550
0120          *  JK IS THE LOWER LAYER SUBSCRIPT                                00002560
0120          *                                                                 00002570
0122                 JK = IJ - 1
0122          *                                                                 00002590
0122          *  DENSITY GRADIENT BETWEEN TWO LAYERS                            00002600
0122          *                                                                 00002610
0124                 SLOPE = ( DEN(IJ) - DEN(JK) ) / DELZ
0132                 ELTOP = DELZ * ( FLOAT ( IJ ) - 0.5 )
0136          *                                                                 00002640
0136          *  INTERPOLATION EQUATION                                         00002650
0136          *                                                                 00002660
0141                 DENINT = DEN(IJ) - ( ELTOP - X ) * SLOPE
0150                 RETURN
0151           120   CONTINUE
0151          *
0151          *  FIND  HE DENSITY OUTSIDE THE POOL
0151          *
0151          *  IF THE LAYER IS OUTSIDE THE POOL, THE DENSITY IS EXTRAPOLATED  00002730
0151          *  BASED ON A LINEAR DENSITY GRADIENT EXTENDED FROM THE PORT      00002740
0151          *  CENTER LINE TO THE DESIRED BOUNDARY LAYER.                     00002750
0151          *                                                                 00002760
0151          *  DENSITY GRADIENT BETWEEN THE SURFACE AND THE PORT CENTER LINE  00002770
0151          *                                                                 00002780
0151                 DGRDT = ( DEN ( ISURF ) - DENPRT ) / ( DEPTH - HGTPRT )
0154          *                                                                 00002800
0154          *  DENSITY GRADIENT BETWEEN THE BOTTOM AND THE PORT CENTER LINE   00002810
0154          *                                                                 00002820
0161                 DGRDB = ( DEN ( 1 ) - DENPRT ) / HGTPRT
0163          *                                                                 00002840
0163          *  SELECT THE NEEDED DENSITY GRADIENT                             00002850
0163          *                                                                 00002860
0165                 IF ( LAYER .GT. ISURF ) DGRD = DGRDT
0172                 IF ( LAYER .LT. 1 ) DGRD = DGRDB
0175          *                                                                 00002911
0175          *  EXTRAPOLATION EQUATION                                         00002912
0175          *                                                                 00002913
0177                 DENINT = DGRD * ABS ( HGTPRT - X ) + DENPRT
0210                 RETURN
0211                 END
```

```
      SUBROUTINE XREAD
*
*   THIS SUBROUTINE READS ALL INPUT DATA AND CONSTRUCTS FULL TABLES
*
      COMMON / AA / QMETR, NSETS, G, HEADING(18), TITLE(18)
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5), PHDIM(5)
      COMMON / GG / COEF, QSUB, QQUAL
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100), YD(100), QDEN
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4),
     *              QUAL(4,100), YQ(4,100)
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      DIMENSION DUMMY(20)
      DIMENSION DUMQUAL(100), DUMYQ(100)
      INTEGER TABTYP, CHECK, CHECK1, CHECK2, HEADING, TITLE
      INTEGER XDATA, XPRIN, XENGL, XMETR, XELEV, XDEPT,
     *        XHEIG, XFILE, XTHIC, XINTE, XSURF, XBOTT,
     *        XNUMB, XPORT, XWEIR, XVDIM, XFLOW, XMINI,
     *        XFREE, XSUBM, XCOEF, XLENG, XDENS, XTEMP,
     *        XQUAL, XFAHR, XCENT, XTABL, XSTOP, XHDIM,
     *        XTURB, XGATE, XFUNC, XTAIL, XDISS, XMAXI
      INTEGER DUMMY, UNITS, XPLOT, XANGL
      LOGICAL QECHO, QMETR, QPORT, QWEIR, QSUB, QPLOT,
     *        QDEN, QCENT, QTABI, QTEMP, QQUAL
      LOGICAL  QVENT, QAERA, QTWFUN, QFIRST
      DATA XPLOT / 4HPLOT /
      DATA XDATA, XPRIN, XENGL / 4HDATA, 4HPRIN, 4HENGL /
      DATA XMETR, XELEV, XDEPT / 4HMETR, 4HELEV, 4HDEPT /
      DATA XHEIG, XFILE, XTHIC / 4HHEIG, 4HFILE, 4HTHIC /
      DATA XINTE, XSURF, XBOTT / 4HINTE, 4HSURF, 4HBOTT /
      DATA XNUMB, XPORT, XGATE / 4HNUMB, 4HPORT, 4HGATE /
      DATA XWEIR, XVDIM, XFLOW / 4HWEIR, 4HVDIM, 4HFLOW /
      DATA XFREE, XSUBM, XCOEF / 4HFREE, 4HSUBM, 4HCOEF /
      DATA XLENG, XDENS, XTEMP / 4HLENG, 4HDENS, 4HTEMP /
      DATA XQUAL, XFAHR, XCENT / 4HQUAL, 4HFAHR, 4HCENT /
      DATA XTABL, XSTOP, XANGL / 4HTABL, 4HSTOP, 4HANGL /
      DATA XHDIM, XDISS  / 4HHDIM, 4HDISS /
      DATA XTURB, XTAIL, XFUNC / 4HTURB, 4HTAIL, 4HFUNC /
      DATA QECHO, QMETR, QPORT, QWEIR / 4 * .FALSE. /
      DATA QSUB, QDEN, QCENT, QTABI, QTEMP / 5 * .FALSE. /
      DATA QVENT, QAERA, QTWFUN, QFIRST / 4 * .FALSE. /
      DATA IFILE / 05 /
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 5HXREAD /
*
*  PROGRAM CONTROL PARAMETERS
```

```
 53      *
 54              IF ( QFIRST ) GO TO 145
 55              QFIRST = .TRUE.
 56      *                                                                 00003500
 57      *  INPUT FILE TITLE                                               00003510
 58      *                                                                 00003520
 59              READ ( IFILE, 610 ) TITLE
 60      *                                                                 00003540
 61      *  FILE CODES                                                     00003550
 62      *                                                                 00003560
 63              READ ( IFILE, 620 ) CHECK, IFILE, KFILE, JFILE, LFILE
 64              IF ( CHECK .NE. XFILE )
 65        *         CALL ERROR ( 1000, SUBR, CHECK, XFILE, XDUMY2, XDUMY3 )
 66              IF ( JFILE .LE. 0 .OR. JFILE .GT. 99 ) JFILE = IFILE
 67              IF ( LFILE .LE. 0 .OR. LFILE .GT. 99 ) LFILE = KFILE
 68      *                                                                 00003620
 69      *  NUMBER OF DATA SETS                                            00003630
 70      *                                                                 00003640
 71              READ ( IFILE, 620 ) CHECK, NSETS
 72              IF ( CHECK .NE. XDATA )
 73        *         CALL ERROR ( 1010, SUBR, CHECK, XDATA , XDUMY2, XDUMY3 )
 74      *                                                                 00003680
 75      *  ECHO PRINT                                                     00003690
 76      *                                                                 00003700
 77              READ ( IFILE, 610 ) CHECK
 78              QECHO = CHECK .EQ. XPRIN
 79              IF ( .NOT. QECHO ) GO TO 140
 80              QECHO = .FALSE.
 81              NFILES = 1
 82      *                                                                 00003790
 83      *  DETERMINE IF JFILE EXISTS                                      00003800
 84      *                                                                 00003810
 85              IF ( JFILE .NE. IFILE ) NFILES = 2
 86      *                                                                 00003830
 87      *  LOOP FOR FILE SELECTION AND PRINTING                           00003840
 88      *                                                                 00003850
 89              DO 120 I = 1, NFILES
 90              IF ( I .EQ. 1 ) MFILE = IFILE
 91              IF ( I .EQ. 2 ) MFILE = JFILE
 92              REWIND MFILE
 93              WRITE ( KFILE, 600 )
 94      *                                                                 00003940
 95      *  INITIALIZE LINE NUMBERS.  ECHO PRINT FILE                      00003950
 96      *                                                                 00003960
 97              LINE = 1000
 98       100    CONTINUE
 99      *                                                                 00003990
100      *  PRINT LINE OF INPUT TO OUTPUT WITH EACH LOOP                   00004000
101      *                                                                 00004010
102              READ ( MFILE, 610, END = 110 ) DUMMY
103              WRITE ( KFILE, 630 ) LINE, DUMMY
104      *                                                                 00004040
```

```
*  INCREMENT LINE NUMBER                                                00004050
*                                                                       00004060
         LINE = LINE + 10
         GO TO 100
 110     CONTINUE
*                                                                       00004100
         REWIND MFILE
*                                                                       00004120
 120     CONTINUE
*                                                                       00004140
*  INCREMENT FILE POINTER TO PRIOR INPUT LINE                           00004150
*                                                                       00004160
         IK = 4
         DO 130 I = 1, IK
         READ ( IFILE, 610 ) DUMMY
 130     CONTINUE
 140     CONTINUE
         RETURN
*
*  ENTRY POINT TO READ THE INDIVIDUAL DATA SETS
*
 145     CONTINUE
*                                                                       00004270
*  INITIALIZE VARIABLES                                                 00004280
*                                                                       00004290
         QMETR = .FALSE.
         QPORT = .FALSE.
         QWEIR = .FALSE.
         QTEMP = .FALSE.
         QSUB = .FALSE.
         QDEN = .FALSE.
         QCENT = .FALSE.
         QTAB1 = .FALSE.
         NPORTS = 0
         NQUAL = 0                                                      00004390
         G = 32.18
*                                                                       00004440
*  DATA SET HEADING                                                     00004450
*                                                                       00004460
         READ ( IFILE, 610 ) HEADING
*                                                                       00004480
*  METRIC OR ENGLISH UNITS                                              00004490
*                                                                       00004500
         READ ( IFILE, 610 ) CHECK
         IF ( CHECK .NE. XMETR .AND.
     *        CHECK .NE. XENGL )
     *        CALL ERROR ( 1020, SUBR, CHECK, XENGL, XMETR, XDUMY3 )
         QMETR = CHECK .EQ. XMETR
         IF ( QMETR ) G = 9.807
*                                                                       00004570
*  FORM OF INPUT TABLES                                                 00004580
*                                                                       00004590
```

```
157              READ ( IFILE, 620 ) CHECK, TABTYP
158              IF ( CHECK .NE. XTABL )
159          *        CALL ERROR ( 1030, SUBR, CHECK, XTABL, XDUMY2, XDUMY3 )
160              QTAB1 = TABTYP .EQ. 1
161      *                                                                  00004640
162      *  LAYER THICKNESS                                                 00004650
163      *                                                                  00004660
164              READ ( IFILE, 650 ) CHECK, DELZ
165              IF ( CHECK .NE. XTHIC )
166          *        CALL ERROR ( 1040, SUBR, CHECK, XTHIC, XDUMY2, XDUMY3 )
167      *                                                                  00004700
168      *  LAYER INTERVALS FOR WHICH OUTPUT INFO IS DESIRED                00004710
169      *                                                                  00004720
170              READ ( IFILE, 620 ) CHECK, INTER
171              IF ( CHECK .NE. XINTE )
172          *        CALL ERROR ( 1050, SUBR, CHECK, XINTE, XDUMY2, XDUMY3 )
173      *                                                                  00004790
174      *  SURFACE ELECATION                                               00004800
175      *                                                                  00004810
176              READ ( IFILE, 650 ) CHECK, SURFACE
177              IF ( CHECK .NE. XSURF )
178          *        CALL ERROR ( 1060, SUBR, CHECK, XSURF, XDUMY2, XDUMY3 )
179      *                                                                  00004850
180      *  BOTTOM ELEVATION                                                00004860
181      *                                                                  00004870
182              READ ( IFILE, 650 ) CHECK, BOTTOM
183              IF ( CHECK .NE. XBOTT )
184          *        CALL ERROR ( 1070, SUBR, CHECK, XBOTT, XDUMY2, XDUMY3 )
185      *
186      *  CONSTRUCT LAYERS
187      *
188              DEPTH = SURFACE - BOTTOM
189      *                                                                  00004950
190      *  CALCULATE NUMBER OF LAYERS                                      00004960
191      *                                                                  00004970
192              ISURF = ( DEPTH / DELZ ) + .999
193              IF ( ISURF .GT. 100 )
194          *        CALL ERROR ( 1080, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
195              DO 150 I = 1, ISURF
196      *                                                                  00005020
197      *  PERCENTAGE OF LAYER FILLED WITH WATER                           00005030
198      *                                                                  00005040
199              HGT(I) = 1.0
200              Y(I) = ( DELZ * FLOAT ( I ) ) - ( .5 * DELZ )
201       150    CONTINUE
202              HGT(ISURF) = ( DEPTH - ( DELZ * ( ISURF - 1 ) ) ) / DELZ
203      *                                                                  00005090
204      *  TOP LAYER MAY NOT BE DELZ THICK                                 00005100
205      *                                                                  00005110
206              Y(ISURF) = DEPTH - ( HGT(ISURF) * DELZ / 2.0 )
207      *
208      *  DESCRIPTION OF WITHDRAWAL DEVICES
```

```
      READ ( IFILE, 620 ) CHECK, TABTYP
      IF ( CHECK .NE. XTABL )
     *      CALL ERROR ( 1030, SUBR, CHECK, XTABL, XDUMY2, XDUMY3 )
      QTAB1 = TABTYP .EQ. 1
*                                                                       00004640
*  LAYER THICKNESS                                                      00004650
*                                                                       00004660
      READ ( IFILE, 650 ) CHECK, DELZ
      IF ( CHECK .NE. XTHIC )
     *      CALL ERROR ( 1040, SUBR, CHECK, XTHIC, XDUMY2, XDUMY3 )
*                                                                       00004700
*  LAYER INTERVALS FOR WHICH OUTPUT INFO IS DESIRED                     00004710
*                                                                       00004720
      READ ( IFILE, 620 ) CHECK, INTER
      IF ( CHECK .NE. XINTE )
     *      CALL ERROR ( 1050, SUBR, CHECK, XINTE, XDUMY2, XDUMY3 )
*                                                                       00004790
*  SURFACE ELECATION                                                    00004800
*                                                                       00004810
      READ ( IFILE, 650 ) CHECK, SURFACE
      IF ( CHECK .NE. XSURF )
     *      CALL ERROR ( 1060, SUBR, CHECK, XSURF, XDUMY2, XDUMY3 )
*                                                                       00004850
*  BOTTOM ELEVATION                                                     00004860
*                                                                       00004870
      READ ( IFILE, 650 ) CHECK, BOTTOM
      IF ( CHECK .NE. XBOTT )
     *      CALL ERROR ( 1070, SUBR, CHECK, XBOTT, XDUMY2, XDUMY3 )
*
*  CONSTRUCT LAYERS
*
      DEPTH = SURFACE - BOTTOM
*                                                                       00004950
*  CALCULATE NUMBER OF LAYERS                                           00004960
*                                                                       00004970
      ISURF = ( DEPTH / DELZ ) + .999
      IF ( ISURF .GT. 100 )
     *      CALL ERROR ( 1080, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
      DO 150 I = 1, ISURF
*                                                                       00005020
*  PERCENTAGE OF LAYER FILLED WITH WATER                                00005030
*                                                                       00005040
      HGT(I) = 1.0
      Y(I) = ( DELZ * FLOAT ( I ) ) - ( .5 * DELZ )
 150  CONTINUE
      HGT(ISURF) = ( DEPTH - ( DELZ * ( ISURF - 1 ) ) ) / DELZ
*                                                                       00005090
*  TOP LAYER MAY NOT BE DELZ THICK                                      00005100
*                                                                       00005110
      Y(ISURF) = DEPTH - ( HGT(ISURF) * DELZ / 2.0 )
*
*  DESCRIPTION OF WITHDRAWAL DEVICES
```

```
*
*                                                                       00005160
*  PORT (AND TOTAL NUMBER THEREOF) OR WEIR                              00005170
*                                                                       00005180
         READ ( IFILE, 620 ) CHECK, NPORTS
         IF ( CHECK .NE. XPORT .AND.
     *        CHECK .NE. XWEIR )
     *        CALL ERROR ( 1100, SUBR, CHECK, XWEIR, XPORT, XDUMY3 )
*                                                                       00005230
*  DETERMINE TYPE OF WITHDRAWAL DEVICE                                  00005240
*                                                                       00005250
         QPORT = CHECK .EQ. XPORT
         QWEIR = CHECK .EQ. XWEIR
         IF ( QPORT ) GO TO 220
         IF ( QWEIR ) GO TO 300
 220     CONTINUE
*
*  PORT CHARACTERISTICS
*                                                                       00005330
*  PORT VERTICAL DIMENSIONS                                             00005340
*
         READ ( IFILE, 650 ) CHECK, ( PVDIM(K), K = 1, NPORTS )
         IF ( CHECK .NE. XVDIM )
     *        CALL ERROR ( 1110, SUBR, CHECK, XVDIM, XDUMY2, XDUMY3 )
*                                                                       00005390
*  PORT HORIZONTAL DIMENSIONS                                           00005400
*                                                                       00005410
         READ ( IFILE, 650 ) CHECK, ( PHDIM(K), K = 1, NPORTS )
         IF ( CHECK .NE. XHDIM )
     *        CALL ERROR ( 1120, SUBR, CHECK, XHDIM, XDUMY2, XDUMY3 )
*                                                                       00005450
*  PORT ELEVATIONS                                                      00005460
*                                                                       00005470
         READ ( IFILE, 650 ) CHECK, ( PHGT(K), K = 1, NPORTS )
         IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG .AND.
     *        CHECK .NE. XDEPT )
     *        CALL ERROR ( 1130, SUBR, CHECK, XDEPT, XHEIG, XELEV )
*                                                                       00005520
*  CONVERT ELEVATION TO HEIGHT ABOVE BOTTOM                             00005530
*                                                                       00005540
         IF ( CHECK .EQ. XHEIG ) GO TO 260
         IF ( CHECK .EQ. XDEPT ) GO TO 240
*                                                                       00005570
*  ELEVATION TO HEIGHT                                                  00005580
*                                                                       00005590
         DO 230 K = 1, NPORTS
         PHGT(K) = PHGT(K) - BOTTOM
 230     CONTINUE
         GO TO 260
 240     CONTINUE
*                                                                       00005650
*  DEPTHS TO HEIGHTS                                                    00005660
```

```
*
      DO 250 K = 1, NPORTS                                              00005670
      PHGT(K) = SURFACE - PHGT(K) - BOTTOM
 250  CONTINUE
 260  CONTINUE
*
*  PORT FLOW RATES                                                      00005720
*                                                                       00005730
                                                                        00005740
      READ ( IFILE, 650 ) CHECK, (FLOW(K), K=1, NPORTS)
      IF ( CHECK .NE. XFLOW )
    *       CALL ERROR ( 1140, SUBR, CHECK, XFLOW, XDUMY2, XDUMY3 )
*
*  WITHDRAWAL ANGLE
*
      READ ( IFILE, 650 ) CHECK, ( WTHETA(K), K=1, NPORTS )
      IF ( CHECK .NE. XANGL )
    *       CALL ERROR ( 1150, SUBR, CHECK, XANGL, XDUMY2, XDUMY3 )
      IF ( NPORTS .EQ. 1 ) GO TO 290
*
*  ORDER PORTS FROM BOTTOM TO TOP
*
      NP = NPORTS - 1
      DO 280 I = 1, NP
      K = I + 1
      DO 270 J = K, NPORTS
      IF ( PHGT(I) .LT. PHGT(J) ) GO TO 270
*                                                                       00005930
*  ASSIGN CHARACTERISTICS OF LOWER SUBSCRIPTED                          00005940
*  PORTS TO DUMMY VARIABLES                                             00005950
*                                                                       00005960
      HGTDUM = PHGT(I)
      VDUM = PVDIM(I)
      HDUM = PHDIM(I)
*                                                                       00006000
*  ASSIGN CHARACTERISTICS OF HIGHER SUBSCRIPTED PORTS TO LOWER SUBSCRIPT 00006010
*                                                                       00006020
      PHGT(I) = PHGT(J)
      PVDIM(I) = PVDIM(J)
      PHDIM(I) = PHDIM(J)
*                                                                       00006060
*  ASSIGN DUMMY VARIABLE  VALUES TO HIGHER SUBSCRIPTED PORT             00006070
*                                                                       00006080
      PHGT(J) = HGTDUM
      PVDIM(J) = VDUM
      PHDIM(J) = HDUM
 270  CONTINUE
 280  CONTINUE
 290  CONTINUE
*
*  CHECK FOR WEIR INPUT                                                 00006150
*                                                                       00006160
                                                                        00006170
      READ ( IFILE, 620 ) CHECK
```

```
313              QWEIR = CHECK .EQ. XWEIR
314              IF ( QWEIR ) GO TO 300
315              BACKSPACE IFILE
316              GO TO 340
317       300    CONTINUE
318      *
319      *  WEIR CHARACTERISTICS
320      *
321      *                                                                    00006270
322      *  SUBMERGED OR FREE                                                 00006280
323      *                                                                    00006290
324              READ ( IFILE, 620 ) CHECK
325              IF ( CHECK .NE. XFREE .AND.
326          *        CHECK .NE. XSUBM )
327          *        CALL ERROR ( 1160, SUBR, CHECK, XSUBM, XFREE, XDUMY3 )
328              QSUB = CHECK .EQ. XSUBM
329              IF ( QSUB ) GO TO 310
330      *                                                                    00006360
331      *  FREE WEIR COEFFICIENT                                             00006370
332      *                                                                    00006380
333              READ ( IFILE, 650 ) CHECK, COEF
334              IF ( CHECK .NE. XCOEF )
335          *        CALL ERROR ( 1170, SUBR, CHECK, XCOEF, XDUMY2, XDUMY3 )
336       310    CONTINUE
337      *                                                                    00006430
338      *  WEIR LENGTH                                                       00006440
339      *                                                                    00006450
340              READ ( IFILE, 650 ) CHECK, WRLNG
341              IF ( CHECK .NE. XLENG )
342          *        CALL ERROR ( 1180, SUBR, CHECK, XLENG, XDUMY2, XDUMY3 )
343      *                                                                    00006490
344      *  WEIR HEIGHT                                                       00006500
345      *                                                                    00006510
346              READ ( IFILE, 650 ) CHECK, WRHGT
347              IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG .AND.
348          *        CHECK .NE. XDEPT )
349          *        CALL ERROR ( 1200, SUBR, CHECK, XDEPT, XELEV, XHEIG )
350      *                                                                    00006560
351      *  CONVERT DEPTH OR ELEV TO HEIGHT ABOVE BOTTOM                      00006570
352      *                                                                    00006580
353              IF ( CHECK .EQ. XHEIG ) GO TO 330
354              IF ( CHECK .EQ. XDEPT ) GO TO 320
355      *                                                                    00006610
356      *  ELEVATION TO HEIGHT                                               00006620
357      *                                                                    00006630
358              WRHGT = WRHGT - BOTTOM
359              GO TO 330
360       320    CONTINUE
361      *                                                                    00006661
362      *  DEPTH TO HEIGHT                                                   00006662
363      *                                                                    00006663
364              WRHGT = SURFACE - WRHGT - BOTTOM
```

```
 330     CONTINUE
*
*  FLOW RATE OVER WEIR                                                  00006690
*                                                                       00006700
*                                                                       00006710
         READ ( IFILE, 650 ) CHECK, WRFLOW
         IF ( CHECK .NE. XFLOW )
     *       CALL ERROR ( 1210, SUBR, CHECK, XFLOW, XDUMY2, XDUMY3 )
 340     CONTINUE
*
*  TURBINE VENTING OR CONDUIT AERATION
*
         READ( IFILE, 610 ) CHECK
         QVENT = CHECK .EQ. XTURB
         QAERA = CHECK .EQ. XGATE
         IF( .NOT. QVENT .AND. .NOT. QAERA ) BACKSPACE IFILE
*
*  TAILWATER FUNCTION OR SINGLE ELEVATION.  NEEDED WHEN QAERA = .TRUE
*
         IF ( .NOT. QAERA ) GO TO 345
         READ ( IFILE, 635 ) CHECK1, CHECK2, TWEL
         IF ( CHECK1 .NE. XTAIL )
     *       CALL ERROR( 1215, SUBR, CHECK1, XTAIL, XDUMY2, XDUMY3 )
         QTWFUN = CHECK2 .EQ. XFUNC
 345     CONTINUE
*
*  INFORMATION FOR DENSITY OR TEMP PROFILE INCLUDING NUMBER OF DATA.    00006930
*
         READ ( IFILE, 640 ) CHECK1, CHECK2, NUMD
         IF ( CHECK1 .NE. XNUMB )
     *       CALL ERROR ( 1220, SUBR, CHECK1, XNUMB, XDUMY2, XDUMY3 )
         IF( CHECK2 .NE. XDENS .AND. CHECK2 .NE. XTEMP )
     *       CALL ERROR( 1225, SUBR, CHECK2, XDENS, XTEMP, XDUMY3)
         QDEN = CHECK2 .EQ. XDENS
         IF ( QDEN ) GO TO 350
         NUMT = NUMD
         GO TO 420
 350     CONTINUE
*
*  DENSITY
*
         IF ( QTAB1 ) GO TO 360
*                                                                       00007090
*  ELEVATION AND DENSITY VALUES LISTED IN SEPERATE TABLES               00007100
*                                                                       00007110
         READ ( IFILE, 610 ) CHECK
*                                                                       00007130
*  ELEVATIONS                                                           00007140
*                                                                       00007150
         IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG .AND.
     *       CHECK .NE. XDEPT )
     *       CALL ERROR ( 1230, SUBR, CHECK, XDEPT, XHEIG, XELEV )
         READ ( IFILE, 660 ) ( YD(M), M = 1, NUMD )
```

```
*                                                                       00007200
*  DENSITIES                                                            00007210
*                                                                       00007220
         READ ( IFILE, 620 ) CHECK
         IF ( CHECK .NE. XDENS )
     *       CALL ERROR ( 1240, SUBR, CHECK, XDENS, XDUMY2, XDUMY3 )
         READ ( IFILE, 660 ) ( DEN(M), M = 1, NUMD )
         GO TO 370
 360     CONTINUE
*                                                                       00007290
*  ELEVATION AND DENSITY VALUES LISTED IN ONE TABLE                     00007300
*                                                                       00007310
         READ ( IFILE, 670 ) CHECK1, CHECK2
         IF ( CHECK1 .NE. XDEPT .AND. CHECK1 .NE. XHEIG
     *       .AND. CHECK1 .NE. XELEV )
     *       CALL ERROR ( 1250, SUBR, CHECK1, XDEPT, XELEV, XHEIG )
         IF ( CHECK2 .NE. XDENS )
     *       CALL ERROR ( 1260, SUBR, CHECK2, XDENS, XDUMY2, XDUMY3 )
         READ ( IFILE, 680 ) ( YD(M), DEN(M), M = 1, NUMD )
         CHECK = CHECK1
 370     CONTINUE
*                                                                       00007410
*  CONVERT DEPTH OR ELEVATION TO HEIGHT ABOVE THE BOTTOM                00007420
*                                                                       00007430
         IF ( CHECK .EQ. XHEIG ) GO TO 410
         IF ( CHECK .EQ. XDEPT ) GO TO 390
*                                                                       00007460
*  ELEVATION TO HEIGHT                                                  00007470
*                                                                       00007480
         DO 380 M = 1, NUMD
         YD(M) = YD(M) - BOTTOM
 380     CONTINUE
         GO TO 410
 390     CONTINUE
*                                                                       00007540
*  DEPTH TO HEIGHT                                                      00007550
*                                                                       00007560
         DO 400 M = 1, NUMD
         YD(M) = SURFACE - YD(M) - BOTTOM
 400     CONTINUE
 410     CONTINUE
*                                                                       00007610
*  GENERATE COMPUTATIONAL DENSITY PROFILE                               00007620
*                                                                       00007630
         CALL INTERP ( DEN, YD, NUMD )
*                                                                       00007650
*  CHECK FOR TEMPERATURE OR QUALITY PROFILE (AND NUMBER OF DATA)        00007660
*  OR A STOP COMMAND                                                    00007670
*                                                                       00007680
         READ ( IFILE, 700 ) CHECK, NUMT
         IF ( CHECK .NE. XNUMB .AND. CHECK .NE. XQUAL .AND.
     *       CHECK .NE. XSTOP )
```

```
     *        CALL ERROR ( 1270, SUBR, CHECK, XNUMB, XQUAL, XSTOP )
          IF ( CHECK .EQ. XSTOP ) RETURN
          IF ( CHECK .EQ. XQUAL ) GO TO 490
*
*  TEMPERATURE
*
 420      CONTINUE
*                                                                      00007790
*  FAHRENHEIT OR CENTIGRADE                                            00007800
*                                                                      00007810

          READ ( IFILE, 690 ) CHECK, UNITS
          IF ( CHECK .NE. XTEMP .AND.
     *        CHECK .NE. XSTOP )
     *        CALL ERROR ( 1280, SUBR, CHECK, XTEMP, XSTOP, XDUMY3 )
          IF ( CHECK .EQ. XSTOP ) RETURN
          IF ( UNITS .NE. XFAHR .AND. UNITS .NE. XCENT )
     *       CALL ERROR ( 1290, SUBR, UNITS, XFAHR, XCENT, XDUMY3 )
          QCENT = UNITS .EQ. XCENT
          QTEMP = .TRUE.
          IF ( QTAB1 ) GO TO 430
*                                                                      00007930
*  ELEVATION AND TEMPERATURE LISTED IN SEPERATE TABLES                 00007940
*                                                                      00007950
          READ ( IFILE, 610 ) CHECK
*                                                                      00007970
*  ELEVATION TABLE                                                     00007980
*                                                                      00007990
          IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG .AND.
     *        CHECK .NE. XDEPT )
     *        CALL ERROR ( 1300, SUBR, CHECK, XHEIG, XELEV, XDEPT )
          READ ( IFILE, 660 ) ( YT(M), M = 1, NUMT )
*                                                                      00008040
*  TEMPERATURE TABLE                                                   00008050
*                                                                      00008060
          READ ( IFILE, 610 ) CHECK
          IF ( CHECK .NE. XTEMP )
     *        CALL ERROR ( 1310, SUBR, CHECK, XTEMP, XDUMY2, XDUMY3 )
          READ ( IFILE, 660 ) ( TEMP(M), M = 1, NUMT )
          GO TO 440
 430      CONTINUE
*                                                                      00008130
*  ELEVATION AND TEMP LISTED IN ONE TABLE                              00008140
*                                                                      00008150
          READ ( IFILE, 670 ) CHECK1, CHECK2
          IF ( CHECK1 .NE. XELEV .AND. CHECK1 .NE. XHEIG
     *       .AND. CHECK1 .NE. XDEPT )
     *        CALL ERROR ( 1320, SUBR, CHECK1, XDEPT, XELEV, XHEIG )
          IF ( CHECK2 .NE. XTEMP )
     *        CALL ERROR ( 1330, SUBR, CHECK2, XTEMP, XDUMY2, XDUMY3 )
          READ ( IFILE, 680 ) ( YT(M), TEMP(M), M = 1, NUMT )
          CHECK = CHECK1
```

```
 440     CONTINUE
*                                                                       00008250
*  CONVERT ELEVATION OR DEPTH TO HEIGHT ABOVE BOTTOM                    00008260
*                                                                       00008270
         IF ( CHECK .EQ. XHEIG ) GO TO 480
         IF ( CHECK .EQ. XDEPT ) GO TO 460
*                                                                       00008300
*  ELEVATION TO HEIGHT                                                  00008310
*                                                                       00008320
         DO 450 M = 1, NUMT
         YT(M) = YT(M) - BOTTOM
 450     CONTINUE

         GO TO 480
 460     CONTINUE
*                                                                       00008390
*  DEPTH TO HEIGHT                                                      00008400
*                                                                       00008410
         DO 470 M = 1, NUMT
         YT(M) = SURFACE - YT(M) - BOTTOM
 470     CONTINUE
 480     CONTINUE
*                                                                       00008460
*  GENERATE COMPUTATIONAL PROFILE                                       00008470
*                                                                       00008480
         CALL INTERP ( TEMP, YT, NUMT )
*                                                                       00008500
         GO TO 500
*
*  QUALITIES
*
 490     CONTINUE
         BACKSPACE IFILE
 500     CONTINUE
*                                                                       00008580
*  CHECK FOR QUALITY PROFILE ( AND NUMBER OF DATA)                      00008590
*  OR A STOP COMMAND                                                    00008600
*                                                                       00008610
         READ ( IFILE, 620 ) CHECK, NQUAL
         QQUAL = CHECK .EQ. XQUAL
         IF ( CHECK .NE. XQUAL .AND.
     *        CHECK .NE. XSTOP )
     *        CALL ERROR ( 1340, SUBR, CHECK, XQUAL, XSTOP, XDUMY3 )
         IF ( CHECK .EQ. XSTOP ) RETURN
         DO 590 I = 1, NQUAL
         READ ( IFILE, 710 ) CHECK, ( NAMEQ(NM,I), NM = 1, 5 ), NUMQ(I)
         IF ( CHECK .NE. XNUMB )                                        00008700
     *         CALL ERROR ( 1350, SUBR, CHECK, XNUMB, XDUMY2, XDUMY3 )  00008710
*                                                                       00008720
*  CHECK THAT THE FIRST QUALITY PROFILE IS DISSOLVED OXYGEN
*  WHEN AERATE OR VENTING SUBROUTINES ARE TO BE USED.
*
```

```
            IF( QVENT .OR. QAERA .AND. NAMEQ(1,1) .NE. XDISS )
     *          CALL ERROR ( 1345, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )        00008770
            NUMBER = NUMQ(I)
            IF ( QTAB1 ) GO TO 510
*                                                                               00008800
*  ELEVATION AND QUALITY LISTED IN SEPERATE TABLES                              00008810
*                                                                               00008820
            READ ( IFILE, 610 ) CHECK
*                                                                               00008840
*  ELEVATION TABLE                                                              00008850
*                                                                               00008860
            IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG .AND.
     *          CHECK .NE. XDEPT )
     *          CALL ERROR ( 1360, SUBR, CHECK, XDEPT, XHEIG, XELEV )
            READ ( IFILE, 660 ) ( YQ(I, M), M = 1, NUMBER )
*                                                                               00008910
*  QUALITY PARAMETERS                                                           00008920
*                                                                               00008930
            READ ( IFILE, 610 ) CHECK
            READ ( IFILE, 660 ) ( QUAL(I, M), M = 1, NUMBER )
            GO TO 520
 510        CONTINUE
*                                                                               00008980
*  ELEVATION AND QUALITY LISTED IN ONE TABLE                                    00008990
*                                                                               00009000
            READ ( IFILE, 670 ) CHECK
            IF ( CHECK .NE. XELEV .AND. CHECK .NE. XHEIG
     *          .AND. CHECK .NE. XDEPT )
     *          CALL ERROR ( 1370, SUBR, CHECK, XDEPT, XHEIG, XELEV )
            READ ( IFILE, 680 ) ( YQ(I, M), QUAL(I, M), M = 1, NUMBER )
 520        CONTINUE
*                                                                               00009070
*  CONVERT ELEVATION OR DEPTH TO HEIGHT ABOVE THE BOTTOM                        00009080
*                                                                               00009090
            IF ( CHECK .EQ. XHEIG ) GO TO 560
            IF ( CHECK .EQ. XDEPT ) GO TO 540
*                                                                               00009120
*  ELEVATION TO HEIGHT                                                          00009130
*                                                                               00009140
            DO 530 M = 1, NUMBER
            YQ(I, M) = YQ(I, M) - BOTTOM
 530        CONTINUE
            GO TO 560
 540        CONTINUE
*                                                                               00009200
*  DEPTH TO HEIGHT                                                              00009210
*                                                                               00009220
            DO 550 M = 1, NUMBER
            YQ(I, M ) = SURFACE - YQ(I, M) - BOTTOM
 550        CONTINUE
 560        CONTINUE
*                                                                               00009270
```

```
*  ASSIGN QUALITY VALUES TO DUMMY VARIABLES TO BE PASSED TO SUBR. INTERP      00009280
*                                                                            00009290
         NQ = NUMQ(I)
         DO 570 K = 1, NUMBER
         DUMQUAL(K) = QUAL(I,K)
         DUMYQ(K) = YQ(I,K)
 570     CONTINUE
*                                                                            00009350
*  GENERATE COMPUTATIONAL PROFILE                                            00009360
*                                                                            00009370
         CALL INTERP ( DUMQUAL, DUMYQ, NQ )
*                                                                            00009390
*  ASSIGN SUBR. INTERP OUTPUT TO ARRAY                                       00009400
*                                                                            00009410
         DO 580 K = 1, ISURF
         QUAL(I,K) = DUMQUAL(K)
 580     CONTINUE
 590     CONTINUE
*                                                                            00009460
*  STOP COMMAND                                                              00009470
*                                                                            00009480
         READ ( IFILE, 610 ) CHECK
         IF ( CHECK .NE. XSTOP )
     *        CALL ERROR ( 1380, SUBR, CHECK, XSTOP, XDUMY2, XDUMY3 )
 600     FORMAT ( 1H1 )
 610     FORMAT ( 20A4 )
 620     FORMAT ( A4, 6X, 14I5 )
 630     FORMAT ( 10X, I6, 7X, 3H***, 20A4 )
 635     FORMAT( A4, 6X, A4,  6X, 6F10.0 )
 640     FORMAT ( A4, 6X, A4, 6X, 12I5 )
 650     FORMAT ( A4, 6X, ( 7F10.0 ) )
 660     FORMAT ( 8F10.0 )
 670     FORMAT ( A4, 6X, A4 )
 680     FORMAT ( 2F10.0 )
 690     FORMAT ( A4, 16X, A4 )
 700     FORMAT ( A4, 16X, 12I5 )
 710     FORMAT ( A4, 6X, 5A4, I5 )
         RETURN
         END
```

```
00000          1              SUBROUTINE INTERP ( PQUAL, YV, NUMV )
00017          2       *
00017          3       *  PROGRAM TO DEVELOP COMPLETE DATA TABLES OR PROFILES
00017          4       *  BY LINEAR INTERPOLATION OF INPUT DATA
00017          5       *
00017          6              COMMON / CC / DELZ                                          00009720
00017          7              COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
00017          8              DIMENSION PQUAL(100), YV(100), PVALUE(100), DM(100), PM(100)
00017          9              INTEGER SIGN1, SIGN2
00017         10       *                                                                 00009760
00017         11       *  TOLERANCE, 0.01% OF LAYER THICKNESS                            00009770
00017         12       *                                                                 00009780
00017         13              SMALL = 0.0001 * DELZ                                      00009790
00017         14       *
00017         15       *  IF DATA IS ORDERED FROM TOP TO BOTTOM, IT MUST BE
00017         16       *  RESEQUENCED FROM BOTTOM TO TOP
00017         17       *
00022         18              IF ( YV(1) .LT. YV(2) ) GO TO 120
00026         19              NV = NUMV + 1
00026         20       *                                                                 00009860
00026         21       *  RESEQUENCE                                                     00009870
00026         22       *                                                                 00009880
00031         23              DO 100 I = 1, NUMV
00033         24              DM(I) = YV(NV - I)
00042         25              PM(I) = PQUAL(NV - I)
00050         26        100   CONTINUE
00055         27              DO 110 I = 1, NUMV
00057         28              YV(I) = DM(I)
00062         29              PQUAL(I) = PM(I)
00065         30        110   CONTINUE
00072         31        120   CONTINUE
00072         32       *                                                                 00009980
00072         33       *  ASSIGN PARAMETER VALUE TO INTERP VARIABLE                      00009990
00072         34       *                                                                 00010000
00072         35              DO 130 I = 1, NUMV
00075         36              PVALUE(I) = PQUAL(I)
00100         37        130   CONTINUE
00100         38       *
00100         39       *  START PROFILE DEVELOPMENT
00100         40       *
00100         41       *                                                                 00010070
00100         42       *  ANY CENTER LINE ELEVATION BELOW LOWEST PARAMETER POINT         00010080
00100         43       *  IS ASSIGNED THE VALUE OF THAT PARAMETER POINT                  00010090
00100         44       *                                                                 00010100
00105         45              DO 140 I = 1, ISURF
00107         46              IF ( Y(I) .GT. YV(1) ) GO TO 150
00114         47              PQUAL(I) = PVALUE(1)
00117         48        140   CONTINUE
00124         49        150   CONTINUE
00124         50              J = I
00124         51       *                                                                 00010170
00124         52       *  ANY CENTER LINE ELEV. ABOVE HEIGHEST PARAMETER POINT           00010180
```

```
00124          53      *  IS ASSIGNED THAT PARAMETER VALUE                                  00010190
00124          54      *                                                                    00010200
00126          55              DO 160 I = 1, ISURF
00131          56              L = ISURF + 1 - I
00135          57              IF ( Y(L) .LT. YV(NUMV) ) GO TO 170
00143          58              PQUAL(L) = PVALUE(NUMV)
00147          59       160    CONTINUE
00154          60       170    CONTINUE
00154          61      *                                                                    00010270
00154          62      *  FIRST CENTER LINE BELOW HIGHEST PARAMETER POINT                   00010280
00154          63      *                                                                    00010290
00154          64              K = L
00154          65      *                                                                    00010310
00154          66      *  FIRST CENTER LINE ABOVE LOWEST PARAMETER POINT                    00010320
00154          67      *                                                                    00010330
00156          68              I = J - 1
00161          69       180    CONTINUE
00161          70              I = I + 1
00162          71              IF ( I .GT. K ) GO TO 270
00166          72              NMINUS = NUMV - 1
00166          73      *                                                                    00010390
00166          74      *  LOCATE DATA POINTS ABOVE AND BELOW THE LAYER CENTER LINE          00010400
00166          75      *                                                                    00010410
00171          76              DO 230 M = 1, NMINUS
00173          77              DIFF1 = ABS ( YV(M) - Y(I) )
00204          78              IF ( DIFF1 .LT. SMALL ) GO TO 190
00207          79      *                                                                    00010450
00207          80      *  IF SIGN1 IS NEGATIVE, FIRST DATA POINT LIES BELOW CENTER LINE     00010460
00207          81      *  IF SIGN1 IS POSITIVE, POINT LIES ABOVE CENTER LINE                00010470
00207          82      *                                                                    00010480
00210          83              SIGN1 = ( YV(M) - Y(I) ) / DIFF1 * 1.2
00223          84              GO TO 200
00224          85       190    CONTINUE
00224          86              SIGN1 = 0
00225          87       200    CONTINUE
00225          88              DIFF2 = ABS ( YV(M+1) - Y(I) )
00236          89              IF ( DIFF2 .LT. SMALL ) GO TO 210
00241          90      *                                                                    00010560
00241          91      *  IF SIGN2 IS NEGATIVE, SECOND DATA POINT LIES BELOW CENTER LINE    00010570
00241          92      *  IF SIGN2 IS POSITIVE, POINT IS ABOVE CENTER LINE                  00010580
00241          93      *                                                                    00010590
00242          94              SIGN2 = ( YV(M+1) - Y(I) ) / DIFF2 * 1.2
00255          95              GO TO 220
00256          96       210    CONTINUE
00256          97              SIGN2 = 0
00257          98       220    CONTINUE
00257          99      *                                                                    00010650
00257         100      *  IF BOTH DATA POINTS ARE BELOW CENTER LINE, GO THRU LOOP AGAIN     00010660
00257         101      *                                                                    00010670
00257         102              IF ( SIGN1 .EQ. SIGN2 .AND. SIGN1 .EQ. -1 ) GO TO 230
00266         103              GO TO 240
00267         104       230    CONTINUE
```

```
00274          240     CONTINUE
00274         *                                                          00010720
00274         *  DOES CENTER LINE LIE VERY CLOSE TO DATA POINT           00010730
00274         *                                                          00010740
00274                 IF ( SIGN1 .EQ. 0 ) GO TO 250
00277                 IF ( SIGN2 .EQ. 0 ) GO TO 260
00301         *                                                          00010770
00301         *  INTERPOLATE BETWEEN DATA POINTS FOR VALUE AT CENTER LINE 00010780
00301         *                                                          00010790
00302                 PQUAL(I) = ( ( PVALUE(M+1) - PVALUE(M) ) * ( Y(I) -
00320            *                YV(M) ) / ( YV(M+1) - YV(M) ) )
00320            *                + PVALUE(M)
00320                 GO TO 180
00321          250    CONTINUE
00321         *                                                          00010850
00321         *  ASSIGN LOWER DATA POINT VALUE TO CENTER LINE            00010860
00321         *                                                          00010870
00321                 PQUAL(I) = PVALUE(M)
00325                 GO TO 180
00326          260    CONTINUE
00326         *                                                          00010910
00326         *  ASSIGN UPPER DATA POINT VALUE TO CENTER LINE            00010920
00326         *                                                          00010930
00326                 PQUAL(I) = PVALUE(M+1)
00332                 GO TO 180
00333          270    CONTINUE
00333                 RETURN
00336                 END
```

```
      SUBROUTINE OUTVEL
*
*  THIS IS THE CONTROL MODULE FOR THE COMPUTATION PORTION
*  OF THIS PROGRAM
*
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT, QPWEIR
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5), PHDIM(5)
      COMMON / GG / COEF, QSUB
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4),
     *              QUAL(4,100), YQ(4,100)
      COMMON / LL / ISURF, HGT(100), DEPTH
      COMMON / MM / SUMOUT,  VEL(100), FLORAT
      COMMON / NN / HGTPRT, VDIM
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100)
      COMMON / QQ / VS(100,6), NOUTS
      COMMON / RR / ZUP(6), ZDN(6), LTOP(6), LLOW(6)
      COMMON  / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      INTEGER TOPLIM
      LOGICAL QPORT, QWEIR, QSUB, QTEMP, QPWEIR
      REAL LENGTH
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 6HOUTVEL /
      LAYER ( X ) = 1. + X / DELZ
      NOUTS = NPORTS
      IF ( QWEIR ) NOUTS = NOUTS + 1
*
*  INITIALIZE THE TOTAL OUTFLOW FLOW RATE PROFILE
*
      DO 100 I = 1, ISURF
      VEL (I) = 0.0
 100  CONTINUE
      SUMOUT = 0.0                                                      00011360
*
*  DETERMINE TYPE OF WITHDRAWAL
*
      IF ( QPORT ) GO TO 110
      IF ( QWEIR ) GO TO 140
*
*  SELECTIVE WITHDRAWAL FOR ORIFICE FLOW
*
 110  CONTINUE
      DO 130 K = 1, NPORTS
*                                                                       00011470
*  VARIABLE ASSIGNMENTS                                                 00011480
*                                                                       00011490
      QPWEIR = .FALSE.
```

```
      FLORAT= FLOW(K)
      IF ( FLORAT .LE. 0. )
     *      CALL ERROR ( 1400, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
      SUMOUT = SUMOUT + FLORAT                                          00011540
      VDIM = PVDIM(K)
      HDIM = PHDIM(K)
      HGTPRT = PHGT(K)
      IF ( HGTPRT .GE. DEPTH )
     *      CALL ERROR ( 1410, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
      DENPRT = DENINT( HGTPRT )
      WANGLE = WTHETA(K)
*
*  CHECK FOR PARTIALLY SUBMERGED PORT
*
      FLODEP = DEPTH - HGTPRT + VDIM/2.
      IF ( VDIM .LE. FLODEP ) GO TO 115
      VW = FLORAT / ( FLODEP * HDIM )
      VHL = ( VW * FLODEP ** .5 ) / HDIM
      IF ( VHL .GT. 0.5 ) GO TO 115
*                                                                       00011700
*  PARTIALLY SUBMERGED PORT-
*  TREAT AS WEIR
*
      QPWEIR = .TRUE.
      QSUB = .TRUE.
      LENGTH = HDIM
      CREST = HGTPRT - VDIM/2.
*                                                                       00011780
      CALL VWEIR
*                                                                       00011800
      GO TO 116
 115  CONTINUE
*                                                                       00011830
*  FULLY SUBMERGED PORT                                                 00011840
*                                                                       00011850
      CALL VPORT
*                                                                       00011870
 116  CONTINUE
*                                                                       00011890
*  ASSIGN FLOW RATE PROFILE VALUES CALCULATED IN VPORT                  00011900
*  OR VWEIR TO AN ARRAY                                                 00011910
*                                                                       00011920
      DO 120 I = 1, ISURF
      VS(I,K) = V(I)
 120  CONTINUE
*                                                                       00011960
*  ASSIGN WITHDRAWAL LIMIT VALUES TO AN ARRAY                           00011970
*                                                                       00011980
      ZUP(K) = HGTTOP
      ZDN(K) = HGTLOW
      LTOP(K) = TOPLIM
      LLOW(K) = LOWLIM
```

```
                                                                                LABEL OUTVEL
 130      CONTINUE
 140      CONTINUE                                                                  00012040
          IF ( .NOT. QWEIR ) GO TO 160                                              00012050
*
*  SELECTIVE WITHDRAWAL FOR WEIR FLOW
*
          FLORAT = WRFLOW
          SUMOUT = SUMOUT + FLORAT
          IF ( FLORAT .LE. 0.)
     *          CALL ERROR ( 1420, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
          LENGTH = WRLNG
          CREST = WRHGT
*                                                                                   00012150
*  DETERMINE WITHDRAWAL LIMITS AND FLOW RATE PROFILES                               00012160
*                                                                                   00012170
          CALL VWEIR
*                                                                                   00012190
*  ASSIGN FLOW RATE PROFILE VALUES CALCULATED IN VWEIR TO AN ARRAY                  00012200
*                                                                                   00012210
          DO 150 I = 1, ISURF
          VS(I,NOUTS) = V(I)
 150      CONTINUE
*                                                                                   00012250
*  ASSIGN WITHDRAWAL LIMIT VALUES TO ARRAY                                          00012260
*                                                                                   00012270
          ZUP(NOUTS) = HGTTOP
          ZDN(NOUTS) = HGTLOW
          LTOP(NOUTS) = TOPLIM
          LLOW(NOUTS) = LOWLIM
 160      CONTINUE
*
*  IF MUTIPLE OUTLETS CALL SHIFT
*
          IF ( NOUTS .GT. 1 ) CALL SHIFT
*
*  DETERMINE TOTAL OUTFLOW FLOW RATE DISTRIBUTION
*
          DO 180 I = 1, ISURF
          DO 170 J = 1, NOUTS
          VEL(I) = VEL(I) + VS(I,J)
 170      CONTINUE
 180      CONTINUE
*
*  FIND MAXIMUM LAYER FLOW RATE
*
          VMAX = VEL(LOWLIM)
          LOWP1 = LOWLIM
          DO 185 I = 1, ISURF
          VMAX = AMAX1 ( VMAX, VEL(I) )
 185      CONTINUE
*
*  ASSIGN LAYER FLOW RATES TO WTHDRW(I)
```

```
157      *
158               DO 190 I = 1, ISURF
159               WTHDRW(I) = VEL(I)  * HGT(I)
160       190     CONTINUE
161               DO 200 I = 1, ISURF
162               VEL(I) = VEL(I) / VMAX
163       200     CONTINUE
164      *
165      *  COMPUTE THE RELEASE DENSITY
166      *
167               SUMDF = 0.
168               DO 210 I = 1, ISURF
169               SUMDF = SUMDF + DEN(I) * WTHDRW(I)
170       210     CONTINUE
171               DENOUT = SUMDF / SUMOUT
172      *
173      *  COMPUTE RELEASE TEMPERATURE
174      *
175               IF ( .NOT. QTEMP ) GO TO 230
176               SUMTF = 0.
177               DO 220 I = 1, ISURF
178               SUMTF = SUMTF + TEMP(I) * WTHDRW(I)
179       220     CONTINUE
180               TEMOUT = SUMTF / SUMOUT
181       230     CONTINUE
182      *
183      *  COMPUTE RELEASE QUALITIES
184      *
185
186               IF ( NQUAL .EQ. 0 ) GO TO 260
187               DO 250 J = 1, NQUAL
188               SUMQF = 0.
189               DO 240 I = 1, ISURF
190               SUMQF = SUMQF + QUAL(J,I) * WTHDRW(I)
191       240     CONTINUE
192               QALOUT(J) = SUMQF / SUMOUT
193       250     CONTINUE
194       260     CONTINUE
195               RETURN
196               END
```

```
      SUBROUTINE VPORT
*
*  CALCULATE WITHDRAWAL LIMITS
*  AND VELOCITY PROFILE FOR ORIFICE
*
      COMMON / AA / QMETR, NSETS, G
      COMMON / CC / DELZ
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
      COMMON / LL / ISURF, HGT(100), DEPTH
      COMMON / MM / SUMOUT, VEL(100), FLORAT
      COMMON / NN / HGTPRT, VDIM, QTLIM, QBLIM, QSINK1,QSINK2
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100), VM
      LOGICAL  QBLIM, QTLIM, QMETR, QSINK1, QSINK2, QSHIFT
      INTEGER XXX, TOPLIM
      DATA  MAX, VMAX, TINY / 10, 1., 1.0E-08 /
      DATA QSINK1, QSINK2, QSHIFT / .TRUE., .TRUE., .FALSE. /
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 5HVPORT /
*
*  FUNCTION STATEMENTS TO SOLVE
*  FOR ORIFICE WITHDRAWAL LIMITS
*  FOR INTERMEDIATE FLOW CONDITIONS
*
      LAYER ( X ) = 1. + X / DELZ
      THICK ( X ) = ABS ( HGTPRT - X )
      FROUD ( X ) = SQRT ( G * ABS ( 1. -
     *        DENINT ( X ) / DENPRT ) )
*                                                                  00013240
*  BASED ON BOHAN AND GRACE (1969)                                 00013250
*                                                                  00013260
      FPORT ( X ) = FLORAT -  C2 * FROUD ( X )
     *          * THICK ( X ) ** 2.5
*
*  FUNCTION STATEMENTS TO SOLVE
*  FOR THE UNBOUNDED WITHDRAWAL LIMIT
*  WHEN THERE IS SURFACE INTERFERENCE
*
      HEIGHT ( X ) = ABS ( BONLIM - X )
      HRATIO ( X ) = TRUNCZ / HEIGHT ( X ) /
     *       ( 1 - TRUNCZ / HEIGHT ( X ) )
      LEFT ( X ) = HEIGHT ( X ) ** 3 * SQRT (
     *       G * ABS ( DENLIM - DENINT ( X ) ) / HEIGHT ( X ) )
      RIGHT ( X ) = C / ( 2 * PI ) * ( 1 + 1 / PI
     *       * SIN ( HRATIO ( X ) * PI ) + HRATIO ( X ) ) /
     *       ( 1 + HRATIO ( X ) ) ** 3
*                                                                  00013420
*  BASED ON SMITH, ET AL (1985)                                    00013430
*                                                                  00013440
      HPORT ( X ) = FLORAT - LEFT ( X) * RIGHT ( X )
*                                                                  00013460
```

```
*  TOLERANCE, 10% OF LAYER THICKNESS                                    00013470
*                                                                       00013480
      SMALL = .10 * DELZ
*
*   SET THE VALUE  OF THE ANGLE OF WITHDRAWAL
*   COEFFICIENT FOR THE BOUNDARY INTERFERENCE EQUATION
*
*                                                                       00013540
*  CHECK TO SEE IF ENTERING FROM SUBROUTINE SHIFT                       00013550
*                                                                       00013560
      IF ( QSHIFT ) GO TO 185
      PI = 3.14159
      C = WANGLE
      C2 = WANGLE / PI
*
*
*  CHECK FOR BOUNDARY INTERFERENCE
*  FROM SURFACE OR BOTTOM USING
*  INTERMEDIATE FLOW EQUATION
*                                                                       00013660
      DENBOT = DENINT ( 0. )
      DENUPP = DENINT ( DEPTH )
      QBLIM = FPORT ( 0. ) .GE. 0.
      QTLIM = FPORT ( DEPTH ) .GE. 0.
*                                                                       00013710
*  DIRECT COMPUTATIONS BASED ON INTERFERENCE CHARACTERISTICS            00013720
*                                                                       00013730
      IF ( QTLIM .AND. QBLIM ) GO TO 540
      IF ( QTLIM ) GO TO 500
      IF ( QBLIM ) GO TO 510
      IF ( .NOT. QTLIM .AND. .NOT. QBLIM ) GO TO 540
 500  CONTINUE
**********
*   IF ONLY ONE  BOUNDARY EXPERIENCES INTERFERENCE, FIND
*   THE HEIGHT OF WITHDRAWAL USING ( SMITH - 1983 )
**********
*
*  DETERMINE THE HEIGHT OF THE TRUNCATED PORTION,
*  THE BOUNDARY LIMIT, THE SEARCH INTERVAL LIMITS,
*  AND THE FUNCTION SIGN AT THE SEARCH LIMITS.
*
*  SURFACE INTERFERENCE
*
      TRUNCZ = DEPTH - HGTPRT
      DENLIM = DENUPP
      BONLIM = DEPTH
      X1 = 0.
      X2 = DEPTH
      H1 = HPORT ( X1 )
      GO TO 530                                                         00013960
 510  CONTINUE
*
```

```
105      *  BOTTOM INTERFERENCE
106      *
107               TRUNCZ = HGTPRT
108               DENLIM = DENBOT
109               BONLIM = 0.
110               X1 = 0.
111               X2 = DEPTH
112               H1 = 1.
113       530     CONTINUE
114      *
115      *  FIND THE LIMIT USING A HALF-INTERVAL SEARCH
116      *
117      *                                                                    00014110
118      *  INITIALIZE X3                                                     00014120
119      *                                                                    00014130
120               X3 = SMALL
121      *                                                                    00014150
122      *  BEGIN ITERATION                                                   00014160
123      *                                                                    00014170
124               DO 560 I = 1, 2 * MAX
125               X4 = X3
126      *                                                                    00014200
127      *  ESTABLISH A THIRD POINT BETWEEN TWO EXISTING POINTS               00014210
128      *                                                                    00014220
129               X3 = ( X1 + X2 ) / 2.0
130      *                                                                    00014240
131      *  CALC. FUNCTION SIGN AT NEW POINT                                  00014250
132      *                                                                    00014260
133               H3 = HPORT ( X3 )
134               ZONED = ABS ( BONLIM - X3 )
135      *                                                                    00014290
136      *  IF NEW POINT IS SAME AS PREVIOS POINT, ITERATION IS COMPLETE      00014300
137      *                                                                    00014310
138               IF ( ABS ( X4 - X3 ) .LT. SMALL ) GO TO 570
139      *                                                                    00014330
140      *  USE AS NEW SEARCH LIMITS THE MOST RECENT POINT AND THE REMAINING  00014340
141      *  POINT OF OPPOSITE FUNCTION SIGN                                   00014350
142      *                                                                    00014360
143               IF ( H1 * H3  .LT. 0. ) GO TO 535
144               X1 = X3
145               H1 = H3
146               GO TO 560
147       535     CONTINUE
148               X2 = X3
149               H2 = H3
150       560     CONTINUE
151      *
152      *  CONVERGENCE WAS NOT REACHED
153      *
154               CALL ERROR ( 1500, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
155       570     CONTINUE
156      *
```

```
*   CALCULATE WITHDRAWAL LIMIT
*
          IF ( QTLIM .AND. .NOT. QBLIM  ) HGTLOW = DEPTH - ZONED
          IF ( QBLIM .AND. .NOT. QTLIM ) HGTTOP = ZONED
 540      CONTINUE
**********
*                                                                      00014570
*  USAGE FOR THE BOHAN AND GRACE EQUATION                              00014580
*  1. NO BOUNDARY INTERFERENCE                                         00014590
*  2. BOTH BOUNDARIES INTERFERE WITH WITHDRAWAL ZONE                   00014600
*  3. SINGLE BOUNDARY INTERFERENCE.  THEORETICAL LIMIT                 00014610
*     OF ONE INTERFERED WITH MUST BE DETERMINED                        00014620
*     (FREE LIMIT IS DETERMINED ABOVE WITH SMITH EQUATION              00014630
*                                                                      00014640
**********
          IF ( QTLIM .AND. .NOT. QBLIM  ) GO TO 150
*
*  EMBARK ON DETERMINATION OF
*  LOWER WITHDRAWAL LIMIT
*
*  IF LOWER LIMIT IS WITHIN THE POOL THEN
*  FIND IT WITH A HALF - INTERVAL SEARCH
*
*  INITIAL SEARCH LIMITS ARE X1 =0 AND X2 = HGTPRT                     00014740
*
          X1 = 0.0
*                                                                      00014770
*  IF BOTTOM BOUNDARY INTERFERENCE EXISTS                              00014780
*  (LOWER LIMIT OUTSIDE POOL), THEN X1 = - DEPTH                       00014790
*                                                                      00014800
          IF (QBLIM ) X1 = - DEPTH                                    00014810
          F1 = FPORT ( X1 )
          DENLIM = DENBOT
          X2 = HGTPRT
          X3 = -2. * SMALL
          ASSIGN 140 TO XXX
 110      CONTINUE
*
*  INITIATE ITERATION PROCESS
*
          DO 130 I = 1, MAX
*
*  ESTABLISH A THIRD POINT BETWEEN
*  THE TWO EXISTING POINTS
*
          X4 = X3
          X3 = ( X1 + X2 ) / 2.
*
*  CALCULATE FUNCTION SIGN AT NEW ELEVATION
*
          IF (  DENLIM .EQ. DENPRT ) GO TO XXX
          F3 = FPORT ( X3 )
```

```
209      *
210      *   IF NEW POINT IS THE SAME AS A PREVIOUS
211      *   POINT THEN SEARCH IS COMPLETE
212      *
213              IF ( ABS ( X4 - X3 ) .LT. SMALL ) GO TO XXX, ( 140, 170 )
214      *
215      *  USE AS NEW SEARCH LIMITS THE MOST
216      *  RECENTLY COMPUTED POINT AND THE
217      *  REMAINING POINT OF OPPOSITE SIGN
218      *
219              IF ( F1 * F3 .GT. 0. ) GO TO 120
220              X2 = X3
221              F2 = F3
222              GO TO 130
223       120    CONTINUE
224              X1 = X3
225              F1 = F3
226       130    CONTINUE
227      *
228      *  CONVERGENCE HAS NOT BEEN REACHED
229      *
230              CALL ERROR ( 1510, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3  )
231       140    CONTINUE
232      *
233      *  SET LOWER LIMIT ELEVATION
234      *
235              HGTLOW = X3
236       150    CONTINUE
237              IF ( QBLIM .AND. .NOT. QTLIM ) GO TO 180
238      *
239      *  APPLY SAME PROCEDURE FOR DETERMINING
240      *  UPPER WITHDRAWAL LIMIT FOR ORIFICE
241      *
242      *  DETERMINE ELEVATION, LAYER AND FUNCTION SIGN AT SEARCH LIMITS.          00015360
243      *  IF NIETHER LIMIT EXPERIENCES INTERFERENCE THE THE INITIAL SEARCH        00015370
244      *  LIMITS ARE X1 = HGTPRT AND X2 = DEPTH. HOWEVER, IF SURFACE              00015380
245      *  INTERFERENCE EXISTS ( UPPER LIMIT OUTSIDE POOL ), X2 = 2*DEPTH          00015390
246      *
247              X1 = HGTPRT
248              X2 = DEPTH                                                      00015420
249              IF ( QTLIM )  X2 = 2 * DEPTH
250              F1 = FPORT ( X1 )
251      *  USE THE PRIOR SEARCH PROCEDURE
252      *
253              ASSIGN 170 TO XXX
254              DENLIM = DENUPP
255              GO TO 110
256       170    CONTINUE
257              HGTTOP = X3
258       180    CONTINUE
259      *
260      *  CALCULATE LOCATION OF MAXIMUM VELOCITY
```

```
*  AND THICKNESS OF WITHDRAWAL LIMITS
*
 185     CONTINUE
         ZONE = HGTTOP - HGTLOW
         ZTOP = HGTTOP - HGTPRT
         ZLOW = HGTPRT - HGTLOW
*                                                                       00015610
*  BASED ON BOHAN AND GRACE                                             00015620
*                                                                       00015630
         YVMAX = ZONE * SIN ( 1.57 * ZLOW / ZONE ) ** 2
*                                                                       00015650
*  HEIGHT ABOVE BOTTOM                                                  00015660
*                                                                       00015670
         XVMAX = YVMAX + HGTLOW
         LVMAX = LAYER ( YVMAX + HGTLOW )
*                                                                       00015700
*  MAXIMUM VELOCITY OUTSIDE THE POOL                                    00015710
*                                                                       00015720
         IF ( ( XVMAX .LT. 0. ) .OR. ( XVMAX .GT. DEPTH ) )
    *        CALL ERROR ( 1520, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
*                                                                       00015750
*  ASSIGN DENSITIES AT LIMITS AND MAXIMUM VELOCITY                      00015760
*                                                                       00015770
         DVMAX = DENINT ( XVMAX )
         DENLOW = DENINT ( HGTLOW )
         DENTOP = DENINT ( HGTTOP )
*
*  WITHDRAWAL LAYER LIMITS
*
         IF ( HGTLOW .LT. 0. ) LOWLIM = LAYER ( 0. )
         IF ( HGTLOW .GE. 0. ) LOWLIM = LAYER ( HGTLOW )
         IF ( HGTTOP .GE. DEPTH ) TOPLIM = ISURF
         IF ( HGTTOP .LT. DEPTH ) TOPLIM = LAYER ( HGTTOP )
*
*  ZERO THE VELOCITY PROFILE
*  FOR THE CURRENT PORT
*
         DO 190 I = 1, ISURF
         V(I) = 0.
 190     CONTINUE
*
*  IF LOWER WITHDRAWAL LAYERS ARE OF
*  OF CONSTANT DENSITY THEN ASSIGN
*  CONSTANT VELOCITY TO EACH LAYER
*                                                                       00015990
         DENDIF = DENLOW - DVMAX
         IF ( DENDIF .GT. 0. ) GO TO 210
         DO 200 I = LOWLIM, LVMAX
         V(I) = VMAX
 200     CONTINUE
         GO TO 240
 210     CONTINUE
```

```
*
*  CALCULATE VELOCITY PROFILE FROM LAYER
*  OF MAXIMUM VELOCITY TO LOWER LIMIT
*
         IF ( LOWLIM .EQ. LVMAX ) GO TO 240
         DO 230 I = LOWLIM, LVMAX
         Y = DELZ * ( LVMAX - I )
         DELDEN = DEN(I) - DVMAX
*                                                                      00016150
*  BASED ON BOHAN AND GRACE                                            00016160
*                                                                      00016170
         RATIO = Y * DELDEN / ( ZLOW * DENDIF )
         RATIO = AMIN1 ( 1., RATIO )
         V(I) = VMAX * ( 1. - RATIO ) ** 2.0
 230     CONTINUE
 240     CONTINUE
*
*  IF UPPER WITHDRAWAL LAYERS ARE
*  OF CONSTANT DENSITY THEN ASSIGN
*  CONSTANT VELOCITY TO EACH LAYER
*
         DENDIF = DVMAX - DENTOP
         IF ( DENDIF .GT. 0. ) GO TO 260
         DO 250 I = LVMAX, TOPLIM
         V(I) = VMAX
 250     CONTINUE
         GO TO 290
 260     CONTINUE
*
*  DETERMINE VELOCITY PROFILE FROM LAYER
*  OF MAXIMUM VELOCITY TO UPPER LIMIT
*
         IF ( LVMAX .EQ. TOPLIM ) GO TO 290
         DO 280 I = LVMAX, TOPLIM
         Y = DELZ * ( I - LVMAX )
         DELDEN = DVMAX - DEN(I)
*                                                                      00016430
*  BASED ON BOHAN AND GRACE                                            00016440
*                                                                      00016450
         RATIO = Y * DELDEN / ( ZTOP * DENDIF )
         RATIO = AMIN1 ( 1., RATIO )
         V(I) = VMAX * ( 1. - RATIO ) ** 2.0
 280     CONTINUE
 290     CONTINUE
*
*  CONVERT NORMALIZED VELOCITIES TO FLOW RATES, I.E., DETERMINE        00016520
*  THE WITHDRAWAL FROM EACH LAYER                                      00016530
*
         SUM = 0.0
         DO 310 I = LOWLIM, TOPLIM
         SUM = SUM + V(I) *  HGT(I)
 310     CONTINUE
```

```
      VM = FLORAT / SUM
      DO 320 I = LOWLIM, TOPLIM
      V(I) = V(I) * VM
 320  CONTINUE
*
*  CHECK FOR POINT SINK DESCRIPTION
*
      VDIM2 = VDIM / 2.
      PRTTOP = HGTPRT + VDIM2
      VD2 = VDIM2
      IF ( PRTTOP .GT. DEPTH ) VD2 = DEPTH - HGTPRT
      IF ( PRTTOP .GT. DEPTH ) PRTTOP = DEPTH
      PRTBOT = HGTPRT - VDIM2
      IF ( PRTBOT .LT. 0. .AND. PRTBOT .GT. -.1 ) PRTBOT = 0.
      DRPTOP = DENPRT - DENINT ( PRTTOP )
      DRPBOT = DENINT ( PRTBOT ) - DENPRT
*
      DRTLIM = DENPRT - DENTOP
      DRBLIM = DENLOW - DENPRT
*
      IF ( DRPBOT .LT. TINY ) DRPBOT = TINY
      IF ( DRPTOP .LT. TINY ) DRPTOP = TINY
      IF ( DRBLIM .LT. TINY ) DRBLIM = TINY
      IF ( DRTLIM .LT. TINY ) DRTLIM = TINY
*                                                                      00016830
*  EMPIRICAL EQUATIONS DEVELOPED FOR POINTSINK VERIFICATION            00016840
*
      SINK1 = ( DRBLIM ) * ZLOW / ( DRPBOT * VDIM2 )
      SINK2 = ( DRTLIM ) * ZTOP / ( DRPTOP * VD2 )
      QSINK1 = SINK1 .GT. 3.0
      QSINK2 = SINK2 .GT. 3.0
      RETURN
      END
```

```
      SUBROUTINE VWEIR
*
*  CALCULATE WITHDRAWAL LIMITS AND
*  VELOCITY PROFILE FOR WEIR FLOW
*
      COMMON / AA / QMETER, NSETS, G
      COMMON / CC / DELZ
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / GG / COEF, QSUB
      COMMON / II / NUMD, DEN(100)
      COMMON / LL / ISURF, HGT(100), DEPTH
      COMMON / MM / SUMOUT, VEL(100), FLORAT
      COMMON / NN / HGTPRT, VDIM,QTLIM,QBLIM,QSINK1,QSINK2,QSHIFT
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100), VM
      LOGICAL QSUB, QMETER, QSHIFT,
     *        QBLIM, QTLIM, Q1, Q2, Q3, QZ
      REAL LENGTH
      INTEGER XXX, TOPLIM
      DATA A, B / 4.35, -1.04 /
      DATA VMAX, ITMAX / 1., 10 /
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 5HVWEIR /
      DATA QSHIFT / .FALSE. /
      LAYER ( X ) = 1. + X/DELZ                                         00017220
*
*  FUNCTION STATEMENTS TO SOLVE
*  FOR LOWER WITHDRAWALS LIMIT
*
      SIZE ( X ) = ABS ( CREST - X )
      RWEIR ( Z ) = SQRT ( Z + HEAD ) * ( 1. + Z / HEAD )
      R2WEIR ( Z ) = SQRT ( Z + HEAD )
*                                                                       00017300
*  BASED ON DORTCH AND WILHELMS                                         00017310
*                                                                       00017320
      FWEIR ( X ) = AVGVEL - C * RWEIR ( SIZE ( X ) ) *
     *      SQRT ( G * ABS ( 1. - DENINT (X) / WRDEN ))                 00017340
     *              + D * R2WEIR ( SIZE(X) ) *
     *             SQRT ( G * ABS ( 1. - DENINT(X) / WRDEN ) )
*                                                                       00017361
*  TOLERANCE, 10% OF THE LAYER THICKNESS                                00017362
*                                                                       00017363
      SMALL = 0.10 * DELZ                                               00017364
*  CHECK TO SEE IF
*  ENTERING  VWEIR
*  FROM SUBROUTINE
*  SHIFT
      IF ( QSHIFT ) GO TO 145
      IF ( QSUB ) GO TO 85
*                                                                       00017430
*  CALCULATE EXPONENT FOR USE WITH FREE WEIR                            00017440
*                                                                       00017450
```

```
      EXPNT = A + B * COEF
      IF ( ABS ( COEF - 3.00 ) .LT. .01 ) EXPNT = 1.5
      IF ( ABS ( COEF - 3.33 ) .LT. .01 ) EXPNT = 0.5
      IF ( ABS ( COEF - 4.10 ) .LT. .01 ) EXPNT = 0.2
 85   CONTINUE
      C = .54

      D = 0.
      QZ = .FALSE.
*
*  CALCULATE AVERAGE VELOCITY
*  OVER THE WEIR IN FT/SEC
*
      VMAX = 1.
      HEAD = DEPTH - CREST
      AVGVEL = FLORAT / ( HEAD * LENGTH )
*
*
*****
*
*  CHECK FOR INTERFERENCE FROM
*  BOTTOM. ASSUMED SURFACE
*  INTERFERENCE.
*
 90   CONTINUE
      WRDEN = DENINT ( CREST )
      QBLIM = FWEIR ( 0. ) .GE. 0.
      QTLIM = .TRUE.
*
*  EMBARK ON DETERMINATION OF
*  LOWER WITHDRAWAL LIMIT
*
      IF ( .NOT. QBLIM ) GO TO 100
*
*  IF BOTTOM INTERFERENCE EXISTS THEN
*  SET LOWER LIMIT AT THE BOTTOM
*
      HGTLOW = 0.
      LOWLIM = 1
      GO TO 140
100   CONTINUE
*
*  IF LOWER LIMIT IS WITHIN THE POOL THEN
*  FIND IT WITH A HALF - INTERVAL SEARCH
*
*  DETERMINE ELEVATION, LAYER, FUNCTION VALUE,
*  AND FUNCTION SIGN AT EACH SEARCH LIMIT
*  X1 BOTTOM OF POOL AND X2 WEIR ELEVATION
*
      X1 = 0.
      F1 = FWEIR ( X1 )
      Q1 = F1 .GT. 0.
```

```
      X2 = CREST
      F2 = AVGVEL
      Q2 = F2 .GT. 0.
      X3 = - 2. * SMALL
*
*  FUNCTION MUST BE POSITIVE AT THE WEIR
*  LEVEL AND NEGATIVE AT THE BOTTOM
*
      IF ( Q1 .OR. .NOT. Q2 )
     *     CALL ERROR ( 1600, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3 )
*
*  INITIATE ITERATION PROCESS
*
      DO 120 I = 1, ITMAX
*
*  ESTABLISH A THIRD POINT BETWEEN
*  THE TWO EXISTING POINTS
*
      X4 = X3
      X3 = ( X1 + X2 ) / 2.
*
*  CALCULATE FUNCTION SIGN AT NEW ELEVATION
*
      F3 = FWEIR ( X3 )
*
*  IF NEW POINT IS THE SAME AS A PREVIOUS
*  POINT THEN SEARCH IS COMPLETE
*
      IF ( ABS ( X4 - X3 ) .LT. SMALL ) GO TO 130
*
*  USE AS NEW SEARCH LIMITS THE MOST
*  RECENTLY COMPUTED POINT AND THE
*  REMAINING POINT OF OPPOSITE SIGN
*
      IF ( F1 * F3 .GT. 0. ) GO TO 110
      X2 = X3
      F2 = F3
      GO TO 120
 110  CONTINUE
      X1 = X3
      F1 = F3
 120  CONTINUE
*
*  CONVERGENCE HAS NOT BEEN REACHED
*
      CALL ERROR ( 1610, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3  )
 130  CONTINUE
*
*  CHECK FOR ( Z + H ) / H LESS THAN 2.0
*  IF TRUE, REASSIGN COEFFICIENTS C AND D AND REPEAT ITERATION PROCESS        00018470
*
      IF ( QZ ) GO TO 136
```

```
157              ZLOW = CREST - X3
158              XCHECK = ( ZLOW + HEAD ) / HEAD
159              IF ( XCHECK .GE. 2.0 ) GO TO 135
160              C = .78
161              D = .70
162              QZ = .TRUE.
163              GO TO 90
164       135    CONTINUE
165       136    CONTINUE
166      *
167      *  SET LOWER LIMIT ELEVATION AND LAYER
168      *
169              HGTLOW = X3
170              LOWLIM = LAYER ( X3 )
171       140    CONTINUE
172      *
173      *  SET UPPER LIMIT AT SURFACE
174
175      *
176       145    CONTINUE
177              HGTTOP = DEPTH
178              TOPLIM = ISURF
179      *
180      *  CALCULATE LOCATION OF MAXIMUM VELOCITY
181      *
182              ZONE = HGTTOP - HGTLOW
183              ZLOW = CREST  - HGTLOW
184      *                                                                  00018770
185      *  IF WEIR IS FREE, MAXIMUM VELOCITY IS AT THE SURFACE             00018780
186      *                                                                  00018790
187              IF ( .NOT. QSUB ) YVMAX = ZONE
188      *                                                                  00018810
189      *  BASED ON BOHAN AND GRACE                                        00018820
190      *                                                                  00018830
191              IF ( QSUB ) YVMAX = ZONE * SIN ( 1.57 * ZLOW / ZONE ) ** 2
192      *
193      *  COMPUTE THICKNESS OF WITHDRAWAL ZONE
194      *
195      *  HEIGHT OF MAX. VELOCITY ABOVE THE BOTTOM                        00018880
196      *                                                                  00018890
197              XVMAX = YVMAX + HGTLOW
198      *                                                                  00018910
199      *  LAYER NO. LOCATION OF MAX VELOCITY                              00018920
200      *                                                                  00018930
201              LVMAX = LAYER ( XVMAX )
202      *                                                                  00018950
203      *  DENSITY AT LAYER OF MAX. VELOCITY                               00018960
204      *                                                                  00018970
205              DVMAX = DENINT ( XVMAX )
206      *                                                                  00018990
207      *  DISTANCE BETWEEN PORT AND LOWER AND UPPER LIMITS RESPECTIVELY   00019000
208      *                                                                  00019010
```

```
209              YLOW = DELZ * ( LVMAX - LOWLIM )
210              YTOP = DELZ * ( TOPLIM - LVMAX )
211       *                                                                 00019040
212       *  DENSITY AT LIMITS                                              00019050
213       *                                                                 00019060
214              DENLOW = DENINT ( HGTLOW )
215              DENTOP = DENINT ( HGTTOP )
216       *
217       *  CALCULATE MAXIMUM VELOCITY
218       *
219              VMAX = 1.
220       *
221       *  ZERO THE VELOCITY PROFILE
222       *  FOR THE CURRENT WEIR
223       *
224              DO 150 I = 1, ISURF
225              V(I) = 0.
226        150   CONTINUE
227       *
228       *  IF LOWER WITHDRAWAL LAYERS ARE OF
229       *  CONSTANT DENSITY THEN ASSIGN
230       *  CONSTANT VELOCITY TO EACH LAYER
231       *
232              IF ( LVMAX .EQ. LOWLIM ) GO TO 200
233              DENDIF = DENLOW - DVMAX
234              IF ( DENDIF .GT. 0. ) GO TO 170
235              DO 160 I = LOWLIM, LVMAX
236              V(I) = VMAX
237        160   CONTINUE
238              GO TO 200
239        170   CONTINUE
240       *
241       *  CALCULATE VELOCITY PROFILE FROM LAYER
242       *  OF MAXIMUM VELOCITY TO LOWER LIMIT
243       *
244              DO 190 I = LOWLIM, LVMAX
245              Y = DELZ * ( LVMAX - I )
246              DELDEN = DEN(I) - DVMAX
247              RATIO = Y * DELDEN / ( YLOW * DENDIF )
248              RATIO = AMIN1 ( 1., RATIO )
249              IF ( QBLIM ) GO TO 180
250              P = 3.0
251       *                                                                 00019440
252       *  IF WEIR IS SUBMERGED                                           00019450
253       *                                                                 00019460
254              IF ( QSUB ) V(I) = VMAX * ( 1. - RATIO ) ** P
255       *                                                                 00019480
256       *  IF WEIR IS FREE                                                00019490
257       *                                                                 00019500
258              IF ( .NOT. QSUB ) V(I) = VMAX * ( 1. - RATIO ** EXPNT )
259              GO TO 190
260        180   CONTINUE
```

```
*                                                                       00019540
*  IF BOTTOM INTERFERENCE                                               00019550
*                                                                       00019560
         V(I) = VMAX * ( 1. - RATIO ** 2 )
 190     CONTINUE
 200     CONTINUE
*
*  IF FREE WEIR, GO TO 260                                              00019610
*                                                                       00019620
         IF ( .NOT. QSUB ) GO TO 260                                    00019630
*                                                                       00019640
*  IF UPPER WITHDRAWAL LAYERS ARE
*  OF CONSTANT DENSITY THEN ASSIGN
*  CONSTANT VELOCITY TO EACH LAYER
*
         IF ( LVMAX .EQ. TOPLIM ) GO TO 260
         DENDIF = DVMAX - DENTOP
         IF ( DENDIF .GT. 0. ) GO TO 220
         DO 210 I = LVMAX, TOPLIM
         V(I) = VMAX
 210     CONTINUE
         GO TO 250
 220     CONTINUE
*
*  DETERMINE VELOCITY PROFILE FROM LAYER
*  OF MAXIMUM VELOCITY TO UPPER LIMIT
*
         DO 240 I = LVMAX, TOPLIM
         Y = DELZ * ( I - LVMAX )
         DELDEN = DVMAX - DEN(I)
         RATIO = Y * DELDEN / ( YTOP * DENDIF )
         RATIO = AMIN1 ( 1., RATIO )
         V(I) = VMAX * ( 1. - RATIO ** 2 )
 240     CONTINUE
 250     CONTINUE
 260     CONTINUE
*
*  CONVERT NORMALIZED VELOCITIES TO FLOW RATES, I.E.,                   00019910
*  DETERMINE WITHDRAWAL FROM EACH LAYER                                 00019920
*
         SUM = 0.0
         DO 270 I = LOWLIM, TOPLIM
         SUM = SUM + V(I) * HGT(I)
 270     CONTINUE
         VM = FLORAT / SUM
         DO 280 I = LOWLIM, TOPLIM
         V(I) = V(I) * VM
 280     CONTINUE
         RETURN
         END
```

```
      SUBROUTINE SHIFT
*
*  THIS SUBROUTINE SHIFTS THE INNER WITHDRAWAL LIMITS
*  WHEN TWO WITHDRAWAL ZONES OVERLAP
*
      COMMON / AA / QMETR, NSETS, G
      COMMON / BB / IFILE, KFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / EE / NPORTS, QPORT, QWEIR
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5)
      COMMON / II / NUMD, DEN(100)
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / LL / ISURF, HGT(100), DEPTH
      COMMON / MM / SUMOUT, VEL(100), FLORAT
      COMMON / NN / HGTPRT, VDIM,QTLIM,QBLIM,QSINK1,QSINK2,QSHIFT
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100)
      COMMON / QQ / VS(100,6), NOUTS
      COMMON / RR / ZUP(6), ZDN(6), LTOP(6), LLOW(6)
      LOGICAL QPORT, QWEIR, QSBLIM, QSTLIM, QBLIM, Q1, Q2, QTLIM
      LOGICAL QPRINT,QSINK1,QSINK2,QSHIFT
      INTEGER TOPLIM, XXX
      DATA MAX / 10 /
      DATA TINY / 1.E-07 /
      DATA QPRINT / .TRUE. /
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 5HSHIFT /
*                                                                       00020310
*  LAYER NO. LOCATION OF X                                              00020320
*                                                                       00020330
      LAYER ( X ) = 1. + X / DELZ
      FROUD ( X, D, ZL ) = SQRT ( G * ABS ( 1. - DENINT ( X ) / D )
     *          * ABS ( X - ZL ) )
*                                                                       00020370
*  BASED ON BOHAN AND GRACE                                             00020380
*                                                                       00020390
      FSHIFT ( VH, X, D, ZL ) = VH - HTEST * FROUD ( X, D, ZL )
*                                                                       00020410
*  TOLERANCE, 10% OF THE LAYER THICKNESS                                00020420
*                                                                       00020430
      SMALL = .10 * DELZ
*                                                                       00020450
*  INITIALIZE LOGICAL VARIABLE                                          00020460
*                                                                       00020470
      QSTLIM = .FALSE.
      QSBLIM = .FALSE.
      QBLIM = .FALSE.
      QTLIM = .FALSE.
      QSHIFT = .FALSE.
*
*  CHECK FOR OVERLAP OF WITHDRAWAL ZONES.
*  IS LOWER LIMIT OF UPPER PORT BELOW UPPER LIMIT OF LOWER PORT         00020550
```

```
 53      *
 54              NMINUS = NOUTS - 1
 55              DO 260 K = 1, NMINUS
 56              H = ZDN(K+1) - ZUP(K)
 57              IF ( H .GE. 0. ) GO TO 260
 58      *
 59      *  SET UP PARAMETERS FOR SHIFTING LIMITS
 60      *
 61              H = ABS ( H )
 62              IF ( QWEIR .AND. K .EQ. NOUTS - 1 ) GO TO 100
 63              HO = PHGT(K+1) - PHGT(K)
 64              GO TO 110
 65       100    CONTINUE
 66      *                                                          00020690
 67      *  ZONES FROM WEIR AND A PORT OVERLAP                      00020700
 68      *                                                          00020710
 69              HO = CREST - PHGT(K)
 70       110    CONTINUE
 71              HTEST = .7 * ( H / HO ) ** 1.25
 72              VH1 = 0.
 73              VH2 = 0.
 74      *                                                          00020770
 75      *  K+1 IS THE UPPER PORT, K IS THE LOWER PORT              00020780
 76      *                                                          00020790
 77              L1 = LLOW(K+1)
 78              L2 = LTOP(K)
 79      *                                                          00020820
 80      *  NUMBER OF LAYERS BETWEEN OVERLAPPING LIMITS, INCLUSIVE  00020830
 81      *                                                          00020840
 82              LAY = L2 - L1 + 1
 83              DO 120 I = L1, L2
 84      *                                                          00020870
 85      *  DETERMINE AVERAGE VELOCITY IN OVERLAPPING REGION        00020880
 86      *                                                          00020890
 87              VH1 = VH1 + VS(I,K)
 88              VH2 = VH2 + VS(I,K+1)
 89       120    CONTINUE
 90              VH1 = VH1 / LAY
 91              VH2 = VH2 / LAY
 92      *                                                          00020950
 93      *  CALCULATE DENSITIES AT LIMITS                           00020960
 94      *                                                          00020970
 95              DENS2 = DENINT ( ZUP(K) )
 96              DENS1 = DENINT ( ZDN(K+1) )
 97      *                                                          00021000
 98      *  LIMIT VARIABLES REASSIGNED                              00021010
 99      *                                                          00021020
100              ZL2 = ZUP(K)
101              ZL1 = ZDN(K + 1)
102      *
103      *  CHECK FOR INTERFERENCE FROM SURFACE OR BOTTOM
104      *
```

```
105              QSTLIM = FSHIFT ( VH1, DEPTH, DENS2, ZL2 ) .GE. 0.
106              QSBLIM = FSHIFT ( VH2, 0., DENS1, ZL1 ) .GE. 0.
107              IF ( .NOT. QSTLIM ) GO TO 125
108       *
109       *
110       *  SOLVE FOR THE SHIFTED LIMIT
111       *  CHECK FIRST FOR DENSITY DIFFERENCE
112       *  BETWEEN THE PORT AND THE SURFACE
113       *
114              IF ( ABS( DEN( ISURF ) - DENINT( PHGT(K) ) ) .GT. TINY )
115           *        GO TO 125
116              ZUP(K) = DEPTH
117              GO TO 175
118        125   CONTINUE
119       *
120       *  DETERMINE FUNCTION SIGN AT EACH SEARCH LIMIT AND ORIGINAL
121       *  UPPER LIMIT OF LOWER PROFILE
122       *  IF THE FUNCTION SIGN IS POSITIVE , ASSUMED AMOUNT OF SHIFT          00021250
123       *  IS LESS THEN ACTUAL THEN ACTUAL AMOUNT OF SHIFT; IF FUNCTION        00021260
124       *  SIGN IS NEGATIVE, ASSUMED AMOUNT OF SHIFT IS GREATER THAN           00021270
125       *  ACTUAL AMOUNT OF SHIFT.                                             00021280
126       *
127              KOUNT = 0
128              X1 = DEPTH
129        130   CONTINUE
130              KOUNT = KOUNT + 1
131              IF ( KOUNT .GE. 5 ) GO TO 135
132              X2 = ZUP(K)
133              F2 = FSHIFT ( VH1, X2, DENS2, ZL2 )
134              Q2 = F2 .GT. 0.
135              X1 = X1 * 2.0
136              F1 = FSHIFT ( VH1, X1, DENS2, ZL2 )
137              Q1 = F1 .GE. 0.
138              X3 = -2. * SMALL
139       *
140       *  FUNCTION SIGN MUST BE POSITIVE AT THE ORIGINAL LIMIT
141       *  AND NEGATIVE AT THE NEW LIMIT, ELSE CHOOSE NEW LIMIT, X1, 2 TIMES GRE      ATER
142       *
143              IF ( Q1 .OR. .NOT. Q2 ) GO TO 130
144              ASSIGN 170 TO XXX
145              GO TO 140
146        135   CONTINUE
147       *
148       *  IF THEORETICAL SHIFTED UPPER LIMIT IS NOT FOUND,                    00021510
149       *  IT IS ASSIGNED TO 2 * DEPTH AND NOTED IN THE OUTPUT.                00021520
150       *
151              IF ( QPRINT ) WRITE ( KFILE, 500 )
152              QPRINT= .FALSE.
153              WRITE ( KFILE, 505 ) K
154              WRITE ( KFILE, 510 )
155              X3 = 2 * DEPTH
156              GO TO 170
```

```
 140     CONTINUE
*
*  INITIATE ITERATION PROCESS
*
         DO 160 I = 1, MAX

*
*  ESTABLISH A THIRD POINT BETWEEN TWO EXISTING POINTS
*
         X4 = X3
         X3 = ( X1 + X2 ) / 2.
*
*  CALCULATE FUNCTION SIGN AT NEW ELEVATION
*
         F3 = FSHIFT ( VH1, X3, DENS2, ZL2 )
*
*  IF NEW POINT IS THE SAME AS THE PREVIOUS POINT,
*  THEN SEARCH IS COMPLETE
*
         IF ( ABS ( X4 - X3 ) .LT. SMALL ) GO TO XXX, ( 170, 200 )
*
*  USE AS NEW SEARCH LIMITS THE MOST RECENTLY COMPUTED POINT
*  AND THE REMAINING POINT OF OPPOSITE SIGN
*
         IF ( F1 * F3 .GT. 0. ) GO TO 150
         X2 = X3
         F2 = F3
         GO TO 160
 150     CONTINUE
         X1 = X3
         F1 = F3
 160     CONTINUE
*
*  CONVERGENCE HAS NOT BEEN REACHED
*
         CALL ERROR ( 1800, SUBR, XDUMY, XDUMY1, XDUMY2, XDUMY3  )
 170     CONTINUE
*
*  SET UP SHIFTED UPPER LIMIT OF LOWER PROFILE
*
         ZUP(K) = X3
 175     CONTINUE
         IF ( .NOT. QSBLIM ) GO TO 180
*
*  CHECK DENSITY STRUCTURE FOR
*  LOWER LIMIT OF UPPER PORT
*
         IF ( ABS( DEN(1) - DENINT( PHGT(K+1) ) ) .GT. TINY )
     *         GO TO 180
         ZDN(K + 1) = 0.
         GO TO 210
 180     CONTINUE
```

```
*
*  DETERMINE FUNCTION SIGN AT SEARCH LIMITS (1) BOTTOM
*  (2) ORIGINAL LOWER LIMIT OF UPPER PROFILE
*
         X1 = - DEPTH / 2.
         KOUNT = 0
 185     CONTINUE
         KOUNT = KOUNT + 1
         IF ( KOUNT .GE. 5 ) GO TO 190
         X1 = X1 * 2.0
         F1 = FSHIFT ( VH2, X1, DENS1, ZL1 )
         Q1 = F1 .GE. 0.
         X2 = ZDN(K + 1)
         F2 = FSHIFT ( VH2, X2, DENS1, ZL1 )
         Q2 = F2 .GT. 0.
*
*  FUNCTION VALUE MUST BE NEGATIVE AT BOTTOM AND POSITIVE
*  AT ORIGINAL LIMIT
*
         IF ( Q1 .OR. .NOT. Q2 ) GO TO 185
*
*  IF LIMIT IS IN POOL, USE PRIOR SEARCH PROCEDURE
*
         ASSIGN 200 TO XXX
         GO TO 140
 190     CONTINUE
*
*  IF LIMIT IS OUTSIDE THE POOL
*  ASSIGN IT TO - DEPTH
*
         WRITE ( KFILE, 520 )
         X3 = - DEPTH
 200     CONTINUE
*
*  SET SHIFTED LOWER LIMIT OF UPPER PROFILE
*
         ZDN(K + 1) = X3
 210     CONTINUE
*
*  COMPUTE NEW NORMALIZED VELOCITIES
*  AND LAYER FLOW RATES FOR LOWER HALF
*  OF PROFILE.
*
*  ASSIGN LIMITS TO VARIABLES                                    00022550
*                                                                00022560
         HGTLOW = ZDN(K)
         LOWLIM = LAYER ( HGTLOW )
         HGTTOP = ZUP(K)
         TOPLIM = LAYER ( HGTTOP )
         HGTPRT = PHGT(K)

         FLORAT = FLOW(K)
```

```
      QBLIM = ZDN(K) .LE. 0.
      IF ( QSTLIM ) QTLIM = .TRUE.
*                                                                       00022660
*  CALL VPORT TO RECALCULATE VELOCITIES                                 00022670
*                                                                       00022680
      QSHIFT = .TRUE.
      CALL VPORT
*                                                                       00022710
      QSHIFT = .FALSE.
      DO 220 I = 1, ISURF
      VS(I, K) = V(I)
 220  CONTINUE
*
*  COMPUTE NEW NORMALIZED VELOCITIES
*  AND LAYER FLOW RATES FOR THE
*  UPPER HALF OF THE PROFILE.
*
*  ASSIGN LIMITS TO VARIABLES                                           00022810
*                                                                       00022820
      HGTLOW = ZDN(K + 1)
      LOWLIM = LAYER( HGTLOW )
      HGTTOP = ZUP(K + 1)
      TOPLIM = LAYER ( HGTTOP )
      IF ( QWEIR .AND. K .EQ. NOUTS - 1 ) GO TO 230
      HGTPRT = PHGT(K + 1)
      FLORAT = FLOW(K + 1)
      QTLIM = ZUP(K + 1) .GE. DEPTH
      IF ( QSBLIM ) QBLIM = .TRUE.
      QSHIFT = .TRUE.
*                                                                       00022930
*  CALL VPORT TO RECALCULATE VELOCITIES                                 00022940
*                                                                       00022950
      CALL VPORT
*                                                                       00022970
      QSHIFT = .FALSE.
      GO TO 240
 230  CONTINUE
*                                                                       00023010
*  ASSIGN WEIR INFO TO VARIABLES                                        00023020
*                                                                       00023030
      CREST = WRHGT
      LENGTH = WRLNG
      FLORAT = WRFLOW
      QTLIM = .TRUE.
      IF ( QSBLIM ) QBLIM = .TRUE.
      QSHIFT = .TRUE.
*                                                                       00023100
*  CALL VWEIR TO CALCULATE VELOCITIES                                   00023110
*                                                                       00023120
*  CALL VWEIR
      QSHIFT = .FALSE.
 240  CONTINUE
```

```
      DO 250 I = 1, ISURF
      VS(I,K + 1) = V(I)
250   CONTINUE
260   CONTINUE
500   FORMAT ( 1H1 )
510   FORMAT ( // , 5X, 29H*** THEORITICAL SHIFTED UPPER,
     *          43H LIMIT NOT FOUND. ASSIGNED TO 2 * DEPTH ***)
520   FORMAT ( // , 5X, 29H*** THEORETICAL SHIFTED LOWER,
     *                41H LIMIT NOT FOUND. ASSIGNED TO - DEPTH ***)
505   FORMAT ( // , 5X, 35H*** LOWER PORT FOR THIS CASE IS NO.,
     *            I5, 5H  ***)
      RETURN
      END
```

```
      SUBROUTINE XPRINT
*
*  PRINTS OUTPUT INFORMATION
*
      COMMON / AA / QMETR, NSETS, G, HEADING(18), TITLE(18)
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT, QPWEIR
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5)
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100)
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4), QUAL(4, 100)
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / MM / SUMOUT, VEL(100)
      COMMON / NN / HGTPRT, VDIM, QTLIM, QBLIM, QSINK1, QSINK2
      COMMON / QQ / VS(100, 6), NOUTS
      COMMON / RR / ZUP(6), ZDN(6)
      COMMON / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
      INTEGER HEADING, TITLE
      LOGICAL QMETR, QPORT, QWEIR, QCENT, QTEMP, QPLOT
      LOGICAL QSINK1, QSINK2, QPWEIR
      DATA XFEET, XMETERS / 4HFEET, 6HMETERS /
      TEMFUN ( T ) = 9. / 5. * T + 32.
*
*  PRINT HEADINGS AND SUMMARY INFORMATION
*
      WRITE ( KFILE, 510 ) TITLE
      WRITE ( KFILE, 520 ) HEADING
*                                                                       00023590
*  UNITS                                                                00023600
*                                                                       00023610
      DIST = XFEET
      IF ( QMETR ) DIST = XMETERS
      WRITE ( KFILE, 500 ) DIST
      IF ( .NOT. QPORT ) GO TO 110
*                                                                       00023660
*  PORT INFORMATION                                                     00023670
*                                                                       00023680
      DO 100 L = 1, NPORTS
      PELEV = PHGT(L) + BOTTOM
      WRITE ( KFILE, 530 ) PELEV, PVDIM(L), FLOW(L), WTHETA(L)
 100  CONTINUE
 110  CONTINUE
*                                                                       00023740
*  WEIR INFORMATION                                                     00023750
*                                                                       00023760
      IF ( QWEIR ) WELE = WRHGT + BOTTOM
      IF ( QWEIR ) WRITE ( KFILE, 540 ) WELE, WRLNG, WRFLOW
*                                                                       00023790
*  FLOW RATE INFORMATION                                                00023800
```

```
 53      *                                                                00023810
 54              WRITE ( KFILE, 550 ) SUMOUT
 55      *                                                                00023830
 56      *  WITHDRAWAL LIMIT INFORMATION                                  00023840
 57      *                                                                00023850
 58      *  THEORETICAL LIMITS                                            00023860
 59      *                                                                00023870
 60              ZUPEL = ZUP(NOUTS) + BOTTOM
 61              ZDNEL = ZDN(1) + BOTTOM
 62      *                                                                00023900
 63      *  ACTUAL LIMITS                                                 00023910
 64      *                                                                00023920
 65              AWLUPP = AMIN1 ( DEPTH, ZUP(NOUTS) )
 66              AWLBOT = AMAX1 ( 0.0, ZDN(1) )
 67              AZUPEL = AWLUPP + BOTTOM
 68              AZDNEL = AWLBOT + BOTTOM
 69              WRITE ( KFILE, 555 ) AWLBOT, AZDNEL
 70              WRITE ( KFILE, 560 ) ZDN(1), ZDNEL
 71              WRITE ( KFILE, 565 ) AWLUPP, AZUPEL
 72              WRITE ( KFILE, 570 ) ZUP(NOUTS), ZUPEL
 73      *                                                                00024010
 74      *  RELEASE DENSITY                                               00024020
 75      *                                                                00024030
 76              WRITE ( KFILE, 580 ) DENOUT
 77              IF ( .NOT. QTEMP ) GO TO 120
 78      *                                                                00024060
 79      *  RELEASE TEMPERATURE                                           00024070
 80      *                                                                00024080
 81              IF ( .NOT. QCENT ) TEMOUT = TEMFUN ( TEMOUT )
 82              WRITE ( KFILE, 590 ) TEMOUT
 83       120    CONTINUE
 84              IF ( NQUAL .EQ. 0 ) GO TO 130
 85      *                                                                00024130
 86      *  RELEASE QUALITY PARAMETER VALUE                               00024140
 87      *                                                                00024150
 88              WRITE( KFILE, 600 ) ( ( NAMEQ(NM, J ), NM = 1, 5 ),
 89         *                          QALOUT(J), J = 1, NQUAL )
 90       130    CONTINUE
 91      *                                                                00024190
 92      *  PORT MODELED AS WEIR                                          00024200
 93      *                                                                00024210
 94              IF ( QPWEIR ) WRITE ( KFILE, 607 )
 95      *                                                                00024230
 96      *  POINT SINK VERIFICATION INFORMATION                           00024240
 97      *                                                                00024250
 98              IF ( .NOT. QSINK1 ) WRITE ( KFILE, 605 )
 99               IF ( .NOT. QSINK2 ) WRITE ( KFILE, 606 )
100      *
101      *  PRINT TABULAR INFORMATION
102      *
103      *  TEMPERATURE                                                   00024310
104      *                                                                00024320
```

```
      WRITE ( KFILE, 610 )
      IF ( QTEMP ) WRITE ( KFILE, 620 )
      IF ( NQUAL .EQ. 0 ) GO TO 140
*                                                                       00024360
*  QUALITIES                                                            00024370
*                                                                       00024380
      WRITE ( KFILE, 630 ) ( ( NAMEQ(NM,J), NM = 1, 3 ),
     *          J = 1, NQUAL )
 140  CONTINUE
      DO 170 I = 1, ISURF, INTER
      K = ISURF - I + 1
      ELEV = Y(K) + BOTTOM
      DEEP = SURFACE - ELEV
*                                                                       00024460
*  ELEVATION, DENSITIES, NORMALIZED VELOCITY, AND LAYER WITHDRAWAL      00024470
*                                                                       00024480
      WRITE ( KFILE, 640 ) ELEV, DEEP, DEN(K),
     *                     VEL(K), WTHDRW(K)
      IF ( .NOT. QTEMP ) GO TO 150
      IF ( .NOT. QCENT ) TEMP(K) = TEMFUN ( TEMP(K) )
      WRITE ( KFILE, 650 ) TEMP(K)
 150  CONTINUE
      IF ( NQUAL .EQ. 0 ) GO TO 160
      WRITE ( KFILE, 660 ) ( QUAL(J, K), J = 1, NQUAL )
 160  CONTINUE
 170  CONTINUE
*
*  PLOT DENSITY AND VELOCITY PROFILES
*
      CALL DVPLOT
*                                                                       00024630
      RETURN
 500  FORMAT ( // 20X, 12HUNITS ARE IN, 1X, A6 )
 510  FORMAT ( 1H1 // 8X, 18A4 )
 520  FORMAT ( //// 8X, 18A4 )
 530  FORMAT ( // 20X, 14HPORT ELEVATION, 2X, F9.3 / 20X,
     *            23HPORT VERTICAL DIMENSION, 2X, F10.3 / 20X,
     *            31HDISCHARGE, VOLUME FLOW PER SEC., 2X, F11.4 / 20X,
     *            21HWITHDRAWAL ANGLE, RAD, 2X , F6.4 )
 540  FORMAT ( // 20X, 20HWEIR CREST ELEVATION, 2X, F9.3 /
     *           20X, 11HWEIR LENGHT, 2X, F10.3 / 20X,
     *           31HDISCHARGE, VOLUME FLOW PER SEC., 2X, F11.4 )
 550  FORMAT ( /// 20X, 31HTOTAL DISCHARGE, VOLUME PER SEC,
     *           2X, F11.4 )
 555  FORMAT ( // 20X, 33HLOWER WITHDRAWAL LIMIT ( ACTUAL ),
     *          20H HEIGHT ABOVE BOTTOM, 7X, F9.3, 4X, 9HELEVATION,
     *          2X, F9.3 )
 560  FORMAT ( 20X, 38HLOWER WITHDRAWAL LIMIT ( THEORETICAL ),
     *          20H HEIGHT ABOVE BOTTOM, 2X, F9.3, 4X, 9HELEVATION,
     *          2X, F9.3 )
 565  FORMAT ( 20X, 33HUPPER WITHDRAWAL LIMIT ( ACTUAL ),
     *          20H HEIGHT ABOVE BOTTOM, 7X, F9.3, 4X, 9HELEVATION,
```

```
     *                2X, F9.3 )
570      FORMAT ( 20X, 38HUPPER WITHDRAWAL LIMIT ( THEORETICAL ),
     *            20H HEIGHT ABOVE BOTTOM, 2X, F9.3, 4X, 9HELEVATION,
     *           2X, F9.3 )
580      FORMAT ( 20X, 15HOUTFLOW DENSITY, 2X, F7.5, 2X, 4HG/CC )
590      FORMAT ( 20X, 19HOUTFLOW TEMPERATURE, 2X, F6.2 )
600      FORMAT ( 20X, 24HOUTFLOW CONCENTRATION OF, 1X, 5A4,
     *            2X, F8.2 )
605      FORMAT ( /, 20X, 10H**********, 5X, 20HWARING-POINT SINK   ,
     *     41HDESCRIPTION NOT ADEQUATE FOR LOWER LIMIT, 5X,                  00024940
     *     10H********** )
606      FORMAT ( /, 20X, 10H**********, 5X, 20HWARING-POINT SINK   ,
     *      41H DESCRIPTION NOT ADEQUATE FOR UPPER LIMIT, 5X,
     *       10H********** )
607      FORMAT ( /, 20X, 5H*****, 3X, 'PARTIALLY SUBMERGED',
     *    ' PORT RESPONDED AS A WEIR FOR THIS CONDITION',
     *     ' SO VWEIR WAS USED' )
610      FORMAT ( 1H1 // 2X, 9HELEVATION, 3X, 5HDEPTH, 4X,
     *            7HDENSITY, 3X, 10HNORM. VEL.,
     *            4X, 4HFLOW )
620      FORMAT ( 1H+, 64X, 11HTEMPERATURE )
630      FORMAT ( 1H+, 74X, 4( 2X, 3A4 ) )
640      FORMAT ( // 2X, F9.3, 4X, F6.2, 4X, F7.5,
     *            5X, F6.4, 3X, F9.4 )
650      FORMAT ( 1H+, 69X, F5.2 )
660      FORMAT ( 1H+, 74X, 4F13.2 )
         END
```

```
      SUBROUTINE DVPLOT
*
*  THIS SUBROUTINE PLOTS PROFILES OF DENSITY
*  AND VELOCITY VERSUS DEPTH
*
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / II / NUMD, DEN(100), YD(100), QDEN
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / MM / SUMOUT,  VEL(100), FLORAT
      DIMENSION COLUMN(100), COLUMN2(100),
     *          VSPACE(11), DSPACE(11)
      INTEGER T, BLANK, PEGGED, FIRST, COLUMN, COLUMN2,
     *        V, D, VSPACE, PLUS
      LOGICAL QRANGE
      DATA PEGGED, BLANK, V, D, PLUS / 1H*, 1H , 1HV, 1HD, 2H +/
      DATA VELMAX / 2.0 /
      WRITE ( KFILE, 500 )
*
*  DETERMINE MAXIMUM VELOCITY
*
      VMAX = VEL(1)
      DO 100 I = 2, ISURF
      IF ( VEL(I) .GT. VMAX ) VMAX = VEL(I)
 100  CONTINUE
*
*  DETERMINE VELOCITY AXIS SPACING
*
      CHANGE = VELMAX * 100.
      VSPACE(1) = 0
      DO 110 K = 1, 10
      VSPACE(K + 1) = VSPACE(K) + CHANGE
 110  CONTINUE
*
*   DETERMINE MAXIMUM AND MINIMUM
*   DENSITIES FOR AXIS SPACING
*
      DMIN = DEN ( 1 )
      DMAX = DEN ( 1 )
      DO 115 I = 2, ISURF
      DMIN = AMIN1 ( DMIN, DEN ( I ) )
      DMAX = AMAX1 ( DMAX, DEN ( I ) )
 115  CONTINUE
      DUM = DMIN * 1000.
      DMIN = INT ( DUM ) / 1000.
      DDIF = DMAX - DMIN
      DMAX = DMIN + 0.01
      IF ( DDIF .GT. 0.01 ) DMAX = DMIN + 0.02
*
*  DETERMINE DENSITY AXIS SPACING
*
      DENDIF = DMAX - DMIN
```

```
 53              CHANGE2 = DENDIF / 10.
 54              DSPACE(1) = DMIN
 55              DO 120 J = 1, 10
 56              DSPACE(J + 1) = DSPACE(J) + CHANGE2
 57       120    CONTINUE
 58      *
 59      *  PRINT VELOCITY AXIS AND LABEL
 60      *
 61              WRITE ( KFILE, 510 )
 62              WRITE ( KFILE, 520 ) ( VSPACE(K), K = 1, 11 )
 63              WRITE ( KFILE, 530 )
 64      *
 65      *  BEGIN TO FILL IN  COLUMN ARRAY
 66              ELEV = BOTTOM + DEPTH + DELZ
 67              DEEP = - DELZ
 68              DO 170 I = 1, ISURF
 69              K = ISURF - I + 1
 70              DEEP = DEEP + DELZ
 71              ELEV = ELEV - DELZ
 72      *
 73      *  BLANK OUT VELOCITY COLUMN ARRAY
 74      *
 75              FIRST = BLANK
 76              LAST = BLANK
 77              DO 130 L = 1, 100
 78              COLUMN(L) = BLANK
 79       130    CONTINUE
 80      *
 81      *  DETERMINE IF VELOCITY VALUES
 82      *  ARE WITHIN RANGE OF PLOT
 83      *
 84              VL = VEL(K)
 85              IF ( VL .LE. 0 ) FIRST = PEGGED
 86      *
 87      *  DETERMINE COLUMN FOR PLOTTING
 88      *  EACH VELOCITY COMPONENT
 89      *
 90              QRANGE = VL .GT. 0 .AND. VL .LE. VELMAX
 91              IF ( .NOT. QRANGE ) GO TO 140
 92              IJK = ( VL / VELMAX ) * 100.
 93              IJK = IJK + 1
 94              COLUMN(IJK) = V
 95       140    CONTINUE
 96      *
 97      *  BLANK OUT DENSITY COLUMN ARRAY
 98      *
 99              DO 150 L = 1, 100
100              COLUMN2(L) = BLANK
101       150    CONTINUE
102      *
103      *  DETERMINE IF DENSITY VALUES
104      *  ARE WITHIN RANGE OF PLOT
```

```
105       *
106                 DN = DEN(K)
107                 IF ( DN .LE. 0. ) FIRST = PEGGED
108                 IF ( DN .GT. DMAX ) LAST = PEGGED
109       *
110       *  DETERMINE COLUMN FOR PLOTTING
111       *  EACH DENSITY COMPONENT
112       *
113                 QRANGE = DN .GT. 0. .AND. DN .LE. DMAX
114                 IF ( .NOT. QRANGE ) GO TO 160
115                 IJK = 100. * ( DN - DMIN ) / DENDIF
116                 IJK = IJK + 1
117                 COLUMN2(IJK) = D
118        160      CONTINUE
119       *
120       *  PRINT ONE LINE OF PLOT
121       *
122                 WRITE ( KFILE, 550 ) ELEV, DEEP, PLUS,
123            *                         FIRST, COLUMN, LAST
124                 WRITE ( KFILE, 560 ) FIRST, COLUMN2, LAST
125        170      CONTINUE
126       *
127       *  PRINT BOTTOM AXIS
128       *
129                 WRITE ( KFILE, 530 )
130                 WRITE ( KFILE, 570 ) ( DSPACE(J), J = 1, 11 )
131                 WRITE ( KFILE, 540 )
132       *
133       *  QUIT
134       *
135                 RETURN
136        500      FORMAT ( 1H1, /// )
137        510      FORMAT ( 30X, 23HVELOCITY ( TIMES 1000 ) )
138        520      FORMAT ( 11X, 11I10 )
139        530      FORMAT ( 20X, 10 ( 10HI--------- ), 1HI )
140        540      FORMAT ( 30X, 12HDENSITY G/CC )
141        550      FORMAT ( 1X, F7.2, 2X, F7.2, A2, 102A1 )
142        560      FORMAT ( 1H+, 19X, 102A1 )
143        570      FORMAT ( 11X, 11F10.5 )
144                 END
```

```
      SUBROUTINE ERROR ( ERR, SUBR, CHK, XCHK1, XCHK2, XCHK3 )
*
*  THERE ARE TWO TYPES OF ERROR CHECKS-
*  ONE FOR INPUT FORMATTING AND ONE FOR
*  PROGRAM COMPUTATIONS. FOR PROGRAM
*  COMPUTATIONS, ONLY  THE ERROR NUMBER AND
*  THE SUBROUTINE-OF-OCCURRANCE NAME ARE
*  PASSED TO THE ERROR SUBROUTINE.
*  THEN AN APPROPRIATE STATEMENT
*  IS MATCHED WITH THE ERROR NUMBER
*  AND PRINTED.  THE OTHER
*  ARGUMENTS PASSED ARE EQUAL TO ZERO
*  AND NOT USED.
*
*  FOR FORMAT ERRORS, THE ERROR NUMBER,
*  THE SUBROUTINE-OF-OCCURRANCE NAME,
*  THE STRING IN ERROR, AND THE STRING(S)
*  EXPECTED BY THE PROGRAM ARE PASSED.
*  UNUSED ARGUMENTS ARE PASSED AS
*  ZERO.
*
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      INTEGER ERR, SUBR, CHK, XCHK1, XCHK2, XCHK3
*
*  PRINT ERROR NUMBER AND SUBROUTINE-OF-OCCURRANCE
*
      WRITE ( KFILE, 500 ) ERR, SUBR
*
*  MATCH ERROR CODE WITH PROPER OUTPUT STATEMENT
*
*
*  ERROR CODES FOR IMPROPER INPUT VALUES
*
      IF ( ERR .EQ. 1400 ) WRITE ( KFILE, 510 )
      IF ( ERR .EQ. 1410 ) WRITE ( KFILE, 520 )
      IF ( ERR .EQ. 1420 ) WRITE ( KFILE, 530 )
*
*  ERROR CODES FOR INTERNAL COMPUTATIONS
*
      IF ( ERR .EQ. 2100 .OR. ERR .EQ. 2070 .OR.
     *     ERR .EQ. 2090 .OR. ERR .EQ. 2300 )
     *     WRITE ( KFILE, 540 )
      IF ( ERR .EQ. 2310 ) WRITE ( KFILE, 570 )
      IF ( ERR .EQ. 2320 ) WRITE ( KFILE, 580 )
*
*  ERROR CODES FOR HALF INTERVAL SEARCH CONVERGENCE ERRORS
*
      IF ( ERR .EQ. 2310 ) WRITE ( KFILE, 570 )
      IF ( ERR .EQ. 2080 .OR. ERR .EQ. 2110 )
     *     WRITE ( KFILE, 590 )
      IF ( ERR .EQ. 1080 ) WRITE ( KFILE, 600 )
*
```

```
*  IF THIS SUBROUTINE  USED ONE OF THE ABOVE 'IF' STATEMENTS
*  THEN THE FOLLOWING IF STATEMENT IS USED TO EXIT PROGRAM
*
        IF ( XCHK1 .EQ. 0 .AND. XCHK2 .EQ. 0
     *        .AND. XCHK3 .EQ. 0 ) GO TO 100
*
*  ERROR CODES FOR INPUT FORMAT FAILURE CHECKS
*
*
*  IF VARIABLE 'CHECK' WAS TESTED FOR ONLY ONE
*  STRING VALUE DURING PROGRAM OPERATION
*
        IF ( XCHK2 .EQ. 0 .AND. XCHK3 .EQ. 0 )
     *        WRITE ( KFILE, 610 ) CHK, XCHK1
*
*  IF VARIABLE 'CHECK' WAS TESTED FOR ONE OF TWO
*  STRING VALUES DURING PROGRAM OPERATION
*
        IF ( XCHK2 .NE. 0 .AND. XCHK3 .EQ. 0 )
     *        WRITE ( KFILE, 620 ) CHK, XCHK1, XCHK2
*
*  IF VARIABLE 'CHECK' WAS TESTED FOR ONE OF THREE
*  STRING VALUES DURING PROGRAM OPERATION
*
        IF ( XCHK2 .NE. 0 .AND. XCHK3 .NE. 0 )
     *        WRITE ( KFILE, 630 ) CHK, XCHK1, XCHK2, XCHK3
 100    STOP
 500    FORMAT ( /, 'ERROR NUMBER ', I4, ' OCCURRED IN SUBROUTINE ',A6)
 510    FORMAT (/, 'FLOW RATE FOR A PORT WAS FOUND TO BE', /,
     *        'LESS THAN OR EQUAL TO ZERO.', /, 'ENTERED FLOW RATE',
     *        ' MUST BE GREATER THAN  ZERO.')
 520    FORMAT ( /, 'CENTER LINE OF PORT WAS HIGHER ', /,
     *        'THAN THE WATER SURFACE.')
 530    FORMAT ( /, 'FLOW RATE FOR WEIR MUST BE GREATER THAN ZERO')
 540    FORMAT ( /,'*** COMPUTATIONAL ERROR IN PROGRAM ')
 570    FORMAT ( /, 'CONVERGENCE ON LIMIT IN SUBROUTINE SHIFT FAILED.')
 580    FORMAT ( /, 'IN HALF INTERVAL SEARCH, CONVERGENCE ON THE', /,
     *        'UPPER OR LOWER WITHDRAWAL LIMIT FAILED')
 590    FORMAT ( /, 'IN HALF INTERVAL SEARCH, CONVERGENCE ON ', /,
     *        'THE UPPER OR LOWER WITHDRAWAL LIMIT FAILED.')
 600    FORMAT ( /,'NUMBER OF LAYERS DIMENSIONED HAS BEEN EXCEEDED')
 610    FORMAT ( /, 'INTERNAL CHECK WAS "',A4, '"', /,
     *        'PROGRAM EXPECTED "', A4,'" AS FIRST FOUR', /,
     *        'CHARACTERS OF INPUT LINE.')
 620    FORMAT ( /,'INTERNAL CHECK WAS "', A4, '"', /,
     *        'PROGRAM EXPECTED "', A4, '" OR "', A4, '"', /,
     *        'AS FIRST FOUR CHARACTERS OF INPUT LINE.')
 630    FORMAT ( /, 'INTERNAL CHECK WAS "',A4, '"', /,
     *        'PROGRAM EXPECTED "', A4, '" OR "', A4, '"', /,
     *        'OR "', A4, '" AS FIRST FOUR CHARACTERS OF',
     *        ' INPUT LINE.')
        END
```

```
      SUBROUTINE AERATE
*
*   PRIDICTS THE RELEASE D.O. (MG/L)
*   THROUGH GATED CONDUIT OULET WORKS
*   USING THE ENERGY DISSIPATION MODEL
*   OUTLINED IN TR-E-81-5
*
      COMMON / AA / QMETR
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / LL / ISURF, HGT(100), DEPTH
      COMMON / MM / SUMOUT
      COMMON / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      LOGICAL QVENT, QAERA, QTWFUN
 9      LOGICAL QMETR
*
*   FUNCTIONS FOR CALCULATIONS
*
*   ADJUSTED ESCAPE COEFFICIENT
*
      CT( X ) = C20 * 1.022 ** ( X - 20.0 )
*
*   DISSOLVED OXYGEN SATURATION
*
      DOSAT( X ) = 1 / ( 0.00209 * X + 0.06719 )
*
*   ALTITUDE CORRECTION FACTOR
*
      BARO( X ) = 1.0 - 3.224 E-5 * X
      C20 = 0.045
      IF ( QMETR ) C20 = 0.1476
*
*   ADJUST ESCAPE COEFFICIENT
*
      C = CT( TEMOUT )
*
*   OXYGEN SATURATION CONCENTRATION ADJUSTED
*   FOR ALTITUDE
*
      ALT = BOTTOM
      IF ( QMETR ) ALT = BOTTOM * 3.28
      CSAT = DOSAT( TEMOUT ) * BARO( ALT )
*
*   DELTA-H THROUGH STRUCTURE
*
      DELH = DEPTH + BOTTOM - TWEL
*
*   DISSOLVED OXYGEN DEFICIT ENTERING STRUCTURE
*
      DI = CSAT - QALOUT(1)
*
*   REAERATE DISCHARGE
```

```
*
      DF = DI * EXP( - C * DELH )
*
*   RELEASE DISSOLVED OXYGEN
*
      QALOUT(1) = CSAT - DF
      RETURN
      END
```

```
00000          1                SUBROUTINE VENTING
00002          2      *
00002          3      *
00002          4      *   PREDICTS RESEASE D.O. BASED ON A MAXIMUM
00002          5      *   30% REDUCTION IN D.O. DEFICIT
00002          6      *
00002          7                COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
00002          8                COMMON / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
00002          9                COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
00002         10                LOGICAL QVENT, QAERA, QTWFUN
00002         11      *
00002         12      *   D.O. SATURATION
00002         13      *
00002         14                DOSAT( X ) = 1 / ( 0.00209 * X + 0.06719 )
00002         15      *
00002         16      *   ALTITUDE CORRECTION FACTOR
00002         17      *
00002         18                BARO( X ) = 1.0 - 3.224 E-5 * X
00002         19      *
00002         20      *   OXYGEN SATURATION CONCENTRATION ADJUSTED
00002         21      *   FOR ALTITUDE
00002         22      *
00002         23                ALT = BOTTOM
00004         24                CSAT = DOSAT( TEMOUT ) * BARO( ALT)
00013         25      *
00013         26      *   DEFICIT CALCULATIONS; DI = INITIAL DEFICIT
00013         27      *   DF = FINAL DEFICIT
00013         28      *
00017         29                DI = CSAT - QALOUT(1)
00022         30                DF = 0.70 * DI
00025         31                QALOUT(1) = CSAT - DF
00030         32                RETURN
00033         33               END                                          00028520
```

```
*         PROGRAM SELECT
*
*  IT COMPUTES THE LIMITS OF THE ZONE OF WITHDRAWAL AND
*  THE DISTRIBUTION OF FLOW WITHIN THAT ZONE.  THE
*  PROGRAM ALSO COMPUTES THE OUTFLOW DENSITY AND QUALITY FROM
*  INPUT DENSITY AND QUALITY PROFILES.
*  LAST REVISION JULY 23, 1985                                          00001060
*
      COMMON / AA / QMETR, NSETS, G, HEADING(18), TITLE(18)
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT, QPWEIR
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5), PHDIM(5)
      COMMON / GG / COEF, QSUB, QQUAL
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4),
     *              QUAL(4,100), YQ(4,100)
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / MM / SUMOUT, VEL(100), FLORAT
      COMMON / NN / HGTPRT,VDIM,QTLIM,QBLIM,QSINK1,QSINK2,QFIRST
      COMMON / OO / LENGTH, CREST, HDIM
      COMMON / PP / LOWLIM, TOPLIM, HGTLOW, HGTTOP, V(100), VM
      COMMON / QQ / VS(100, 6), NOUTS
      COMMON / RR / ZUP(6), ZDN(6), LTOP(6), LLOW(6)
      COMMON / SS / WTHDRW(100), DENOUT, TEMOUT, QALOUT(4)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      LOGICAL QDEN, QCENT
      LOGICAL QVENT, QAERA
      DATA C1, C2 / -3.9863, 508929.2 /
      DATA C3, C4 / 288.9414, 68.12963 /
*                                                                       00001330
*  DENSITIES FROM TEMPEATURES                                           00001340
*                                                                       00001350
      DENFUN ( X ) = 1. - ( X + C1 ) ** 2 / C2
     *               * ( X + C3 ) / ( X + C4 )
*                                                                       00001380
*   FAHRENHEIT TO CENTIGRADE                                            00001390
*                                                                       00001400
      TEMFUN ( T ) = ( 5. / 9. ) * ( T - 3 )
*
*  READ CONTROL INFORMATION
*
      CALL XREAD
*
*  INITIATE LOOP FOR THE NUMBER OF DATA SETS
*
      DO 130 I = 1, NSETS
*
*  READ INPUT DATA AND CONSTRUCT COMPLETE DATA TABLES
```

```
53      *
54              CALL XREAD
55      *
56      *  DEVELOP DENSITY PROFILE IF NOT GIVEN
57      *
58              IF ( QDEN ) GO TO 110
59              DO 100 J = 1, ISURF
60              IF ( .NOT. QCENT ) TEMP(J) = TEMFUN ( TEMP(J) )
61              DEN (J) = DENFUN ( TEMP(J) )
62       100    CONTINUE
63       110    CONTINUE
64      *
65      *  CHECK FOR STABLE DENSITY PROFILE
66      *
67              DO 120 K = 2, ISURF
68              IF ( DEN(K - 1) .GE. DEN(K) ) GO TO 120
69              WRITE(KFILE, 500)
70              STOP
71       120    CONTINUE
72      *
73      *  COMPUTE SELECTIVE WITHDRAWAL LIMITS AND VELOCITIES AND
74      *  RESULTANT OUTFLOW DENSITY AND QUALITIES
75      *
76              CALL OUTVEL
77      *
78      *  MODIFY OUTFLOW D.O. IF GATED STRUCTURE
79      *  AERATION IS USED.
80      *
81              IF ( QAERA ) CALL AERATE
82      *
83      *  MODIFY OUTFLOW D.O. IF TURBINE VENTING
84      *  IS USED.
85      *
86              IF ( QVENT ) CALL VENTING
87      *
88      *  PRINT RESULTS
89      *
90              CALL XPRINT
91       130    CONTINUE
92              STOP
93       500    FORMAT ( /// 10X, 27HDENSITY PROFILE UNSTABLE - ,
94          *                     16HPROGRAM STOPPED )
95              END
```

```
00000          1              FUNCTION DENINT ( X )
00005          2       *
00005          3       *  DETERMINE DENSITY AT ANY LOCATION
00005          4       *
00005          5              COMMON / CC / DELZ
00005          6              COMMON / II / NUMD, DEN(100), YD(100), QDEN, DENPRT
00005          7              COMMON / LL / ISURF, HGT(100), DEPTH
00005          8              COMMON / NN / HGTPRT
00005          9              INTEGER SIGN
00005         10       *                                                          00002040
00005         11       *  DETERMINE LAYER NUMBER IN WHICH HEIGHT X EXISTS         00002050
00005         12       *                                                          00002060
00005         13              LAYER = 1. + X / DELZ
00005         14       *                                                          00002080
00005         15       *  IS HEIGHT X OUTSIDE THE POOL BOUNDARIES?                00002090
00005         16       *                                                          00002100
00015         17              IF ( LAYER .GT. ISURF .OR. LAYER .LT. 1 ) GO TO 120
00023         18       *                                                          00002120
00023         19       *  TOLERANCE, 0.01 % OF THE LAYER THICKNESS                00002130
00023         20       *                                                          00002140
00024         21              SMALL = 0.0001 * DELZ                              00002150
00024         22       *                                                          00002151
00024         23       *
00024         24       *  FIND THE DENSITY INSIDE THE POOL
00024         25       *
00024         26       *                                                          00002190
00024         27       *  HEIGHT TO THE MIDPOINT OF THE LAYER OF POINT X          00002200
00024         28       *                                                          00002210
00027         29              ELMID = DELZ * ( FLOAT ( LAYER ) - 0.5 )
00033         30       *                                                          00002230
00033         31       *  DISTANCE BETWEEN POINT X AND ITS RESPECTIVE LAYERS MIDPOINT 00002240
00033         32       *                                                          00002250
00036         33              DIFF = ABS ( ELMID - X )
00044         34       *                                                          00002270
00044         35       *  ASSIGN THE DENSITY AT X THE VALUE OF ITS RESPECTIVE LAYER'S 00002280
00044         36       *  MIDPOINT IF  X IS IN LAYER 1 AND BELOW THE MIDPOINT; OR 00002290
00044         37       *  X IS IN THE TOP LAYER AND ABOVE THE MIDPOINT; OR        00002300
00044         38       *  X LIES VERY CLOSE TO THE MIDPOINT ITSELF                00002310
00044         39       *                                                          00002320
00045         40              IF ( ( LAYER .EQ. 1 .AND. X .LE. ELMID ) .OR.
00065         41            *        ( LAYER .EQ. ISURF .AND. X .GE. ELMID ) .OR.
00065         42            *          DIFF .LT. SMALL ) GO TO 100
00065         43              GO TO 110
00066         44        100   CONTINUE
00066         45       *                                                          00002380
00066         46              DENINT = DEN(LAYER)
00071         47              RETURN
00075         48        110   CONTINUE
00075         49       *                                                          00002440
00075         50       *  INTERPOLATION CALCULATIONS                              00002450
00075         51       *                                                          00002460
00075         52       *  IF SIGN IS POSITIVE, X IS BELOW THE CENTER LINE         00002461
```

```
00075          *  IF SIGN IS NEGATIVE, X IS ABOVE THE CENTER LINE                  00002462
00075          *                                                                   00002480
00075                 SIGN = ( ELMID - X ) / ABS ( ELMID - X )
00113                 IJK = - ( SIGN - 1 ) / 2
00116          *                                                                   00002510
00116          *  IJ IS THE UPPER LAYER SUBSCRIPT                                  00002520
00116          *                                                                   00002530
00120                 IJ = LAYER + IJK
00120          *                                                                   00002550
00120          *  JK IS THE LOWER LAYER SUBSCRIPT                                  00002560
00120          *                                                                   00002570
00122                 JK = IJ - 1
00122          *                                                                   00002590
00122          *  DENSITY GRADIENT BETWEEN TWO LAYERS                              00002600
00122          *                                                                   00002610
00124                 SLOPE = ( DEN(IJ) - DEN(JK) ) / DELZ
00132                 ELTOP = DELZ * ( FLOAT ( IJ ) - 0.5 )
00136          *                                                                   00002640
00136          *  INTERPOLATION EQUATION                                           00002650
00136          *                                                                   00002660
00141                 DENINT = DEN(IJ) - ( ELTOP - X ) * SLOPE
00150                 RETURN
00151          120    CONTINUE
00151          *
00151          *  FIND THE DENSITY OUTSIDE THE POOL
00151          *
00151          *  IF THE LAYER IS OUTSIDE THE POOL, THE DENSITY IS EXTRAPOLATED    00002730
00151          *  BASED ON A LINEAR DENSITY GRADIENT EXTENDED FROM THE PORT        00002740
00151          *  CENTER LINE TO THE DESIRED BOUNDARY LAYER.                       00002750
00151          *                                                                   00002760
00151          *  DENSITY GRADIENT BETWEEN THE SURFACE AND THE PORT CENTER LINE    00002770
00151          *                                                                   00002780
00151                 DGRDT = ( DEN ( ISURF ) - DENPRT ) / ( DEPTH - HGTPRT )
00154          *                                                                   00002800
00154          *  DENSITY GRADIENT BETWEEN THE BOTTOM AND THE PORT CENTER LINE     00002810
00154          *                                                                   00002820
00161                 DGRDB = ( DEN ( 1 ) - DENPRT ) / HGTPRT
00163          *                                                                   00002840
00163          *  SELECT THE NEEDED DENSITY GRADIENT                               00002850
00163          *                                                                   00002860
00165                 IF ( LAYER .GT. ISURF ) DGRD = DGRDT
00172                 IF ( LAYER .LT. 1 ) DGRD = DGRDB
00175          *                                                                   00002911
00175          *  EXTRAPOLATION EQUATION                                           00002912
00175          *                                                                   00002913
00177                 DENINT = DGRD * ABS ( HGTPRT - X ) + DENPRT
00210                 RETURN
00211                 END
```

```
      SUBROUTINE XREAD
*
*   THIS SUBROUTINE READS ALL INPUT DATA AND CONSTRUCTS FULL TABLES
*
      COMMON / AA / QMETR, NSETS, G, HEADING(18), TITLE(18)
      COMMON / BB / IFILE, KFILE, JFILE, LFILE
      COMMON / CC / DELZ, INTER, SURFACE, BOTTOM
      COMMON / DD / WTHETA(5), WANGLE
      COMMON / EE / NPORTS, QPORT, QWEIR, QPLOT
      COMMON / FF / PVDIM(5), PHGT(5), FLOW(5), PHDIM(5)
      COMMON / GG / COEF, QSUB, QQUAL
      COMMON / HH / WRLNG, WRHGT, WRFLOW
      COMMON / II / NUMD, DEN(100), YD(100), QDEN
      COMMON / JJ / NUMT, QCENT, TEMP(100), YT(100), QTEMP
      COMMON / KK / NQUAL, NUMQ(4), NAMEQ(5,4),
     *              QUAL(4,100), YQ(4,100)
      COMMON / LL / ISURF, HGT(100), DEPTH, Y(100)
      COMMON / TT / QVENT, QAERA, QTWFUN, TWEL
      DIMENSION DUMMY(20)
      DIMENSION DUMQUAL(100), DUMYQ(100)
      INTEGER TABTYP, CHECK, CHECK1, CHECK2, HEADING, TITLE
      INTEGER XDATA, XPRIN, XENGL, XMETR, XELEV, XDEPT,
     *        XHEIG, XFILE, XTHIC, XINTE, XSURF, XBOTT,
     *        XNUMB, XPORT, XWEIR, XVDIM, XFLOW, XMINI,
     *        XFREE, XSUBM, XCOEF, XLENG, XDENS, XTEMP,
     *        XQUAL, XFAHR, XCENT, XTABL, XSTOP, XHDIM,
     *        XTURB, XGATE, XFUNC, XTAIL, XDISS, XMAXI
      INTEGER DUMMY, UNITS, XPLOT, XANGL
      LOGICAL QECHO, QMETR, QPORT, QWEIR, QSUB, QPLOT,
     *              QDEN, QCENT, QTAB1, QTEMP, QQUAL
      LOGICAL  QVENT, QAERA, QTWFUN, QFIRST
      DATA XPLOT / 4HPLOT /
      DATA XDATA, XPRIN, XENGL / 4HDATA, 4HPRIN, 4HENGL /
      DATA XMETR, XELEV, XDEPT / 4HMETR, 4HELEV, 4HDEPT /
      DATA XHEIG, XFILE, XTHIC / 4HHEIG, 4HFILE, 4HTHIC /
      DATA XINTE, XSURF, XBOTT / 4HINTE, 4HSURF, 4HBOTT /
      DATA XNUMB, XPORT, XGATE / 4HNUMB, 4HPORT, 4HGATE /
      DATA XWEIR, XVDIM, XFLOW / 4HWEIR, 4HVDIM, 4HFLOW /
      DATA XFREE, XSUBM, XCOEF / 4HFREE, 4HSUBM, 4HCOEF /
      DATA XLENG, XDENS, XTEMP / 4HLENG, 4HDENS, 4HTEMP /
      DATA XQUAL, XFAHR, XCENT / 4HQUAL, 4HFAHR, 4HCENT /
      DATA XTABL, XSTOP, XANGL / 4HTABL, 4HSTOP, 4HANGL /
      DATA XHDIM, XDISS  / 4HHDIM, 4HDISS /
      DATA XTURB, XTAIL, XFUNC / 4HTURB, 4HTAIL, 4HFUNC /
      DATA QECHO, QMETR, QPORT, QWEIR / 4 * .FALSE. /
      DATA QSUB, QDEN, QCENT, QTAB1, QTEMP / 5 * .FALSE. /
      DATA QVENT, QAERA, QTWFUN, QFIRST / 4 * .FALSE. /
      DATA IFILE / 05 /
      DATA XDUMY, XDUMY1, XDUMY2, XDUMY3 / 0.0, 0.0, 0.0, 0.0 /
      DATA SUBR / 5HXREAD /
*
*  PROGRAM CONTROL PARAMETERS
```

```
      *
            IF ( QFIRST ) GO TO 145
            QFIRST = .TRUE.
      *                                                                 00003500
      *  INPUT FILE TITLE                                               00003510
      *                                                                 00003520
            READ ( IFILE, 610 ) TITLE
      *                                                                 00003540
      *  FILE CODES                                                     00003550
      *                                                                 00003560
            READ ( IFILE, 620 ) CHECK, IFILE, KFILE, JFILE, LFILE
            IF ( CHECK .NE. XFILE )
        *         CALL ERROR ( 1000, SUBR, CHECK, XFILE, XDUMY2, XDUMY3 )
            IF ( JFILE .LE. 0 .OR. JFILE .GT. 99 ) JFILE = IFILE
            IF ( LFILE .LE. 0 .OR. LFILE .GT. 99 ) LFILE = KFILE
      *                                                                 00003620
      *  NUMBER OF DATA SETS                                            00003630
      *                                                                 00003640
            READ ( IFILE, 620 ) CHECK, NSETS
            IF ( CHECK .NE. XDATA )
        *         CALL ERROR ( 1010, SUBR, CHECK, XDATA , XDUMY2, XDUMY3 )
      *                                                                 00003680
      *  ECHO PRINT                                                     00003690
      *                                                                 00003700
            READ ( IFILE, 610 ) CHECK
            QECHO = CHECK .EQ. XPRIN
            IF ( .NOT. QECHO ) GO TO 140
            QECHO = .FALSE.
            NFILES = 1
      *                                                                 00003790
      *  DETERMINE IF JFILE EXISTS                                      00003800
      *                                                                 00003810
            IF ( JFILE .NE. IFILE ) NFILES = 2
      *                                                                 00003830
      *  LOOP FOR FILE SELECTION AND PRINTING                           00003840
      *                                                                 00003850
            DO 120 I = 1, NFILES
            IF ( I .EQ. 1 ) MFILE = IFILE
            IF ( I .EQ. 2 ) MFILE = JFILE
            REWIND MFILE
            WRITE ( KFILE, 600 )
      *                                                                 00003940
      *  INITIALIZE LINE NUMBERS.  ECHO PRINT FILE                      00003950
      *                                                                 00003960
            LINE = 1000
  100       CONTINUE
      *                                                                 00003990
      *  PRINT LINE OF INPUT TO OUTPUT WITH EACH LOOP                   00004000
      *                                                                 00004010
            READ ( MFILE, 610, END = 110 ) DUMMY
            WRITE ( KFILE, 630 ) LINE, DUMMY
      *                                                                 00004040
```

```
*  INCREMENT LINE NUMBER                                                00004050
*                                                                       00004060
         LINE = LINE + 10
         GO TO 100
 110     CONTINUE
*
         REWIND MFILE                                                   00004100
*
 120     CONTINUE                                                       00004120
*
*  INCREMENT FILE POINTER TO PRIOR INPUT LINE                           00004140
*                                                                       00004150
         IK = 4                                                         00004160
         DO 130 I = 1, IK
         READ ( IFILE, 610 ) DUMMY
 130     CONTINUE
 140     CONTINUE
         RETURN
*
*  ENTRY POINT TO READ THE INDIVIDUAL DATA SETS
*
 145     CONTINUE
*
*  INITIALIZE VARIABLES                                                 00004270
*                                                                       00004280
         QMETR = .FALSE.                                                00004290
         QPORT = .FALSE.
         QWEIR = .FALSE.
         QTEMP = .FALSE.
         QSUB = .FALSE.
         QDEN = .FALSE.
         QCENT = .FALSE.
         QTAB1 = .FALSE.
         NPORTS = 0
         NQUAL = 0
         G = 32.18                                                      00004390
*
*  DATA SET HEADING                                                     00004440
*                                                                       00004450
         READ ( IFILE, 610 ) HEADING                                    00004460
*
*  METRIC OR ENGLISH UNITS                                              00004480
*                                                                       00004490
         READ ( IFILE, 610 ) CHECK                                      00004500
         IF ( CHECK .NE. XMETR .AND.
     *        CHECK .NE. XENGL )
     *        CALL ERROR ( 1020, SUBR, CHECK, XENGL, XMETR, XDUMY3 )
         QMETR = CHECK .EQ. XMETR
         IF ( QMETR ) G = 9.807
*
*  FORM OF INPUT TABLES                                                 00004570
*                                                                       00004580
                                                                        00004590
```