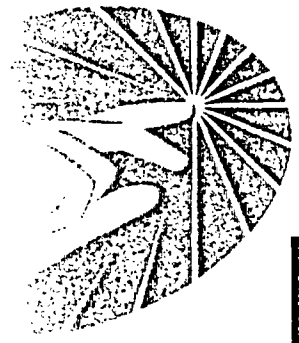


Dominion℠

R84-10

FINAL CALIBRATION OF THE COOLING LAKE MODEL FOR NORTH ANNA POWER STATION

by
EDMOND K. HO
and
E. ERIC ADAMS

RALPH M. PARSONS LABORATORY
AQUATIC SCIENCE AND ENVIRONMENTAL ENGINEERING

Report No. 295

Prepared under the Support of
Virginia Electric and Power Company
Richmond, Virginia

AUGUST 1984

MIT
DEPARTMENT
OF
CIVIL
ENGINEERING
SCHOOL OF ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Cambridge, Massachusetts 02139

R84-10

FINAL CALIBRATION OF THE COOLING LAKE MODEL FOR NORTH ANNA POWER STATION

by

Edmond K. Ho

and

E. Eric Adams

RALPH M. PARSONS LABORATORY

AQUATIC SCIENCE AND ENVIRONMENTAL ENGINEERING

DEPARTMENT OF CIVIL ENGINEERING

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Report No. 295

Prepared under the support of

Virginia Electric and Power Company

Richmond, Virginia

AUGUST 1984

ABSTRACT

The North Anna Power Station is a two-unit power plant located on Lake Anna and operated by Virginia Electric and Power Company (VEPCO). A complex cooling lake system, involving a diked-off portion of Lake Anna - known as the Waste Heat Treatment Facility (WHTF) - as well as the main lake, was designed to dissipate the waste heat rejected by the plant. In order to monitor the impacts that the waste heat discharge has on the natural environment and also to assess the efficiency of the cooling lake system, a segmented mathematical model was previously developed (Jirka et al., 1977).

Since Unit 1 of North Anna Power Station came on line in summer of 1978 (followed by Unit 2 in autumn of 1980), an extensive data collection network was established by VEPCO. The model was previously calibrated using data from the first three years of operational data (1978 to 1981, primarily one-unit operation). Since then more measurements were available and an independent validation was conducted by comparing model predictions with data for the following two years (1981 to 1983, with more two-unit operation) using the same model calibrations.

Surface temperature error analysis was made for four representative diagnostic control points over the cooling lake system. It was noticed that the model results for five years were similar to those for the first three years. However, temperature rise across the plant condenser was consistently over-predicted by an average of about 0.7°C over all five years. In addition, results of spectral analysis showed that the raw error at DIKE III was periodic at predominantly the annual frequency whereas the rest of the control points showed mild periodicities over a relatively wide range of frequencies.

Model recalibration aiming at improving the goodness-of-fit of the five years of measurement data at DISCHARGE and DIKE III was motivated by the results of the surface temperature error analysis. Two possible reasons for the transient errors characterized by annual periodicity of the raw error at DIKE III were identified, namely, errors associated with (1) the forcing function (surface heat transfer) and (2) system response of the model. Two candidates for each of the possible reasons were considered: (1a) atmospheric radiation, (1b) evaporation, (2a) spatially-averaged system response, and (2b) longitudinal system response. Regarding the steady state errors at DISCHARGE, a sensitivity analysis of the raw error with respect to changes in the plant operation data was made, and it was noticed that the mean raw error was reduced substantially by a moderate reduction of the waste heat discharged into the WHTF.

In all, two changes of the model were made as a result of recalibration: computation of residence times in the WHTF and adjustment of plant operation data. The recalibrated model was validated by comparing predictions with five years (1978-1983) of

measurement data in terms of both surface temperatures and vertical temperature profiles. Overall mean errors for the four diagnostic control points ranged (in terms of magnitude) from a minimum of 0.01 °C to a maximum of 0.16°C while standard deviations ranged from 0.80 °C to 1.37 °C. In comparision with the large peak-to-peak annual variation in water surface temperature of about 23 °C, the mean errors and standard deviations are both acceptably small. Comparisons between measured and predicted vertical temperature profiles in the main lake indicated that the dynamic nature of the data both in space and time was also modelled satisfactorily.

In addition to overall averages, monthly averages of the surface temperature raw errors were also computed for each of the diagnostic control points. These monthly error statistics can be applied by VEPCO to increase the accuracy of future model predictions.

ACKNOWLEDGEMENTS

This study was funded by Virginia Electric and Power Company, Richmond, Virginia (MIT OSP No. 93576). The report is the last in a series which describes the development, calibration and verification of the North Anna Cooling Lake Model. Earlier technical reports include

Watanabe, M., Harleman, D. and Connor, J., "Finite Element Model for Transient Two-Layer Cooling Pond Behavior", Technical Report No. 202, R.M. Parsons Laboratory, Department of Civil Engineering, MIT, 1975.

Brocard, D., Jirka, G. and Harleman, D., "A Model for the Convective Circulation in Side Arms of Cooling Lakes", Technical Report No. 223, R.M. Parsons Laboratory, Department of Civil Engineering, MIT, 1977.

Octavio, K., Jirka, G. and Harleman, D., "Vertical Heat Transport Mechanisms in Lakes and Reservoirs", Technical Report No. 227, R.M. Parsons Laboratory, Department of Civil Engineering, MIT, 1977.

Jirka, G., Brocard, D., Octavio, K., Watanabe, M. and Harleman, D., "Analysis of Cooling Effectiveness and Transient Long-Term Simulations of a Cooling Lake with Application to the North Anna Power Station", Technical Report No. 232, R.M. Parsons Laboratory, Department of Civil Engineering, MIT, 1977.

Wells, S., Adams, E. and Harleman, D., "Calibration and Verification of the Cooling Lake Model for North Anna Power Station during the Period July 1978 - September 1981", Technical Report No. 272, R.M. Parsons Laboratory, Department of Civil Engineering, MIT, 1982.

The authors wish to express their gratitude to VEPCO for their generous support and to Drs. Morris Brehmer and Jack Taylor, Messrs. John White, Judson White and Nat Wooding and Ms. Joyce Barton for their personal help and cooperation.

The authors also acknowledge the efforts of Drs. Masataka Watanabe, Dominique Brocard, Gerhard Jirka and Ms. Kathleen Octavio who were involved in the original model development; Drs. Peter Shanahan and Kevin Farley and Mr. Scott Wells who participated in early stages of model calibration; and Prof. Donald Harleman who provided helpful comments throughout both phases.

This report represents the Master's thesis of Mr. Edmond Ho submitted to the Department of Civil Engineering at Massachusetts Institute of Technology. Thesis supervision was provided by Dr. E. Eric Adams, Principal Research Engineer in the MIT Energy Laboratory and Lecturer in the Department of Civil Engineering.

The manuscript was typed by Mrs. Diane Westerink and some of the figures were drafted by Mrs. Izildinha Baptista. The authors express their sincere appreciation to both.

CONTENTS

1. INTRODUCTION 8
   1.1 Electric Power Plant and Waste Heat Disposal Problem 8
   1.2 North Anna Power Station 10
   1.3 The Cooling Lake System 10

2. REVIEW OF PREVIOUS INVESTIGATIONS 16
   2.1 Introduction 16
   2.2 Initial Model Development Stage 16
   2.3 Late Model Development Stage 18

3. NORTH ANNA COOLING LAKE MODEL 19
   3.1 Introduction 19
   3.2 Model for Waste Heat Treatment Facility 19
      3.2.1 Outline of Model 19
      3.2.2 Model for Side Arm 22
      3.2.3 Model for WHTF Ponds 26
   3.3 Model for Dike III Mixing 30
   3.4 Main Lake Model 33
      3.4.1 Outline of Model 33
      3.4.2 Finite Difference Model for Surface Layer 34
      3.4.3 Finite Difference Model for Subsurface Layer 35
   3.5 Summary of Model 36
   3.6 Previous Calibrations of Model 37

4. SURFACE TEMPERATURE ERROR ANALYSIS 40
   4.1 Introduction 40
   4.2 Raw Error 41
   4.3 Delta Error 51
   4.4 Error Statistics 52
   4.5 Spectral Analysis 58
   4.6 Summary of Analysis 66

5. MODEL RECALIBRATION 70
   5.1 Introduction 70
   5.2 Transient Errors at DIKE III 71
      5.2.1 General Discussion 71
      5.2.2 Periodic Errors in Surface Heat Transfer 80
         5.2.2.1 Long-Wave Radiation 80
         5.2.2.2 Evaporative Heat Transfer 87
      5.2.3 Errors in System Response 93
         5.2.3.1 Spatially Averaged Response 93
         5.2.3.2 Longitudinal System Response 95
      5.2.4 Conclusion of Transient Error Analysis 101
   5.3 Steady State Errors at DISCHARGE 102

6. MODEL VALIDATION 105
   6.1 Introduction 105
   6.2 Surface Temperature Error Analysis 105
   6.3 Vertical Temperature Profiles in Lake Anna 111

7. SUMMARY 120

APPENDIX 124
LIST OF REFERENCES 127
LIST OF FIGURES 129
LIST OF TABLES 131

1. INTRODUCTION

1.1 Electric Power Plant and Waste Heat Disposal Problem

The typical conversion efficiencies of present steam-electric power plants are about 38% for fossil fuel plants and 32% for nuclear plants, with internal plant losses of about 15% and 5%, respectively. The remaining bulk of energy is dissipated in the form of heat through a condenser cooling system, the ultimate heat sink of which is, not surprisingly, the atmosphere.

Since all steam power plants reject a significant amount of waste heat to the environment, the choice and design of a condenser cooling system is crucial in assessing the environmental impacts that an individual power plant has upon the neighboring area. An effective condenser cooling system is one which can transfer economically the waste heat from the condenser water to the atmosphere.

Three main categories of condenser cooling systems are commonly used in the electric power industry:

a. Once-through cooling

The cooling water is withdrawn from a large, nearby water body and the heat-loaded water from the condenser is discharged back into the water body. The waste heat is transferred from the water surface to the atmosphere by evaporation, conduction and back radiation. The "thermal inertia" (i.e., the ability of a cooling system to damp out meteorological transients and fluctuations in the power plant operation) of a once-through cooling system is usually high. Once-through systems have been very popular among the coastal cities.

b. Cooling lakes

A cooling lake is basically the same as once-through cooling except that the former is an enclosed water body with a limited volume. It is a closed system in the sense that condenser water is recirculated. The steady state performance of a cooling lake is dependent primarily on the lake area, geometry and inlet and outlet structures while their thermal inertia, dependent on lake depth, is comparable with once-through system. Cooling lake offers an attractive alternative of waste heat disposal for many inland power plants where land is readily available adjacent to a source of make-up water or where artificial impoundments can be (or have already been) constructed by damming a nearby river.

c. Cooling towers

Wet cooling towers emit heat directly to the atmosphere primarily through evaporation of the heated water and the process is enhanced by means of a moving air stream. In regions of scarce water supply dry and wet/dry towers are being considered. The principal heat transfer mechanism of a dry cooling tower is conduction between ambient air and the cooling water through the heat exchanger surface. In addition to the construction costs and the disadvantages of requiring regular maintenance, the "thermal inertia" of a cooling tower is relatively small, rendering it less desirable when compared with the other two cooling systems, despite the fact that modular design of cooling towers presents definite merits.

Overall, the availability of land and water, economic factors and social values, and the characteristics of the proposed plant combine together to determine the choice of a waste heat disposal facility.

1.2 North Anna Power Station

The North Anna Power Station is a nuclear power plant operated by Virginia Electric and Power Company, Richmond, Virginia (VEPCO). It is located in Louisa County in central Virginia, 66 km northwest of Richmond and 64 km east of Charlottsville (Figure 1.1). The Station is situated on the south bank of a lake formed by impounding the North Anna River (Figure 1.2) in January of 1972.

The power plant consists of two nuclear units; previous plans which called for third and fourth units have been cancelled. The two nuclear units generate 947 MW and 925 MW respectively with an average thermal efficiency of about 34% which translates into a waste heat load rejected in the condenser cooling system of about 1839 MW per unit.

The condenser cooling water flow rate is about 60 $m^3 s^{-1}$ per unit (4 pumps per unit at 15 $m^3 s^{-1}$ per pump), and the attendant temperature rise while passing through the condenser is about 8°C.

1.3 The Cooling Lake System

A cooling lake system was designed to dissipate the waste heat rejected by the nuclear power plant. Lake Anna was formed by impounding the North Anna River through construction of a dam (see Figure 1.2). Additional dikes were constructed and main channels dredged to form a separate chain of ponds known as the Waste Heat Treatment Facility (WHTF). Both the WHTF and Lake Anna share the burden of dissipating the waste heat load to the atmosphere, though the major portion is dissipated through the WHTF.

At a design elevation of 76 m above mean sea level (MSL), Lake Anna has a surface area of 39 km^2, a volume of 3.0×10^8 m^3, and an

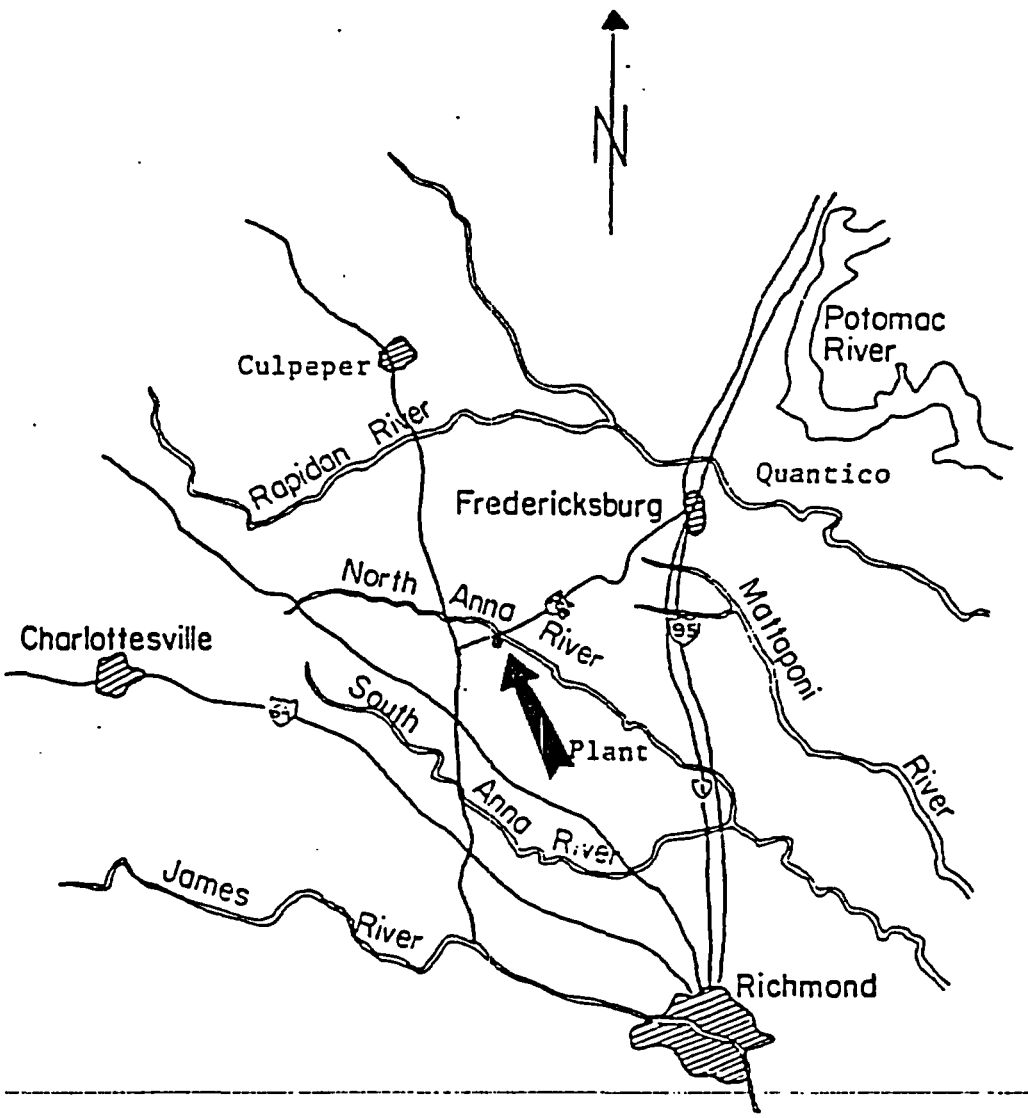


Figure 1.1 Geographical Location of North Anna Power Station, Virginia

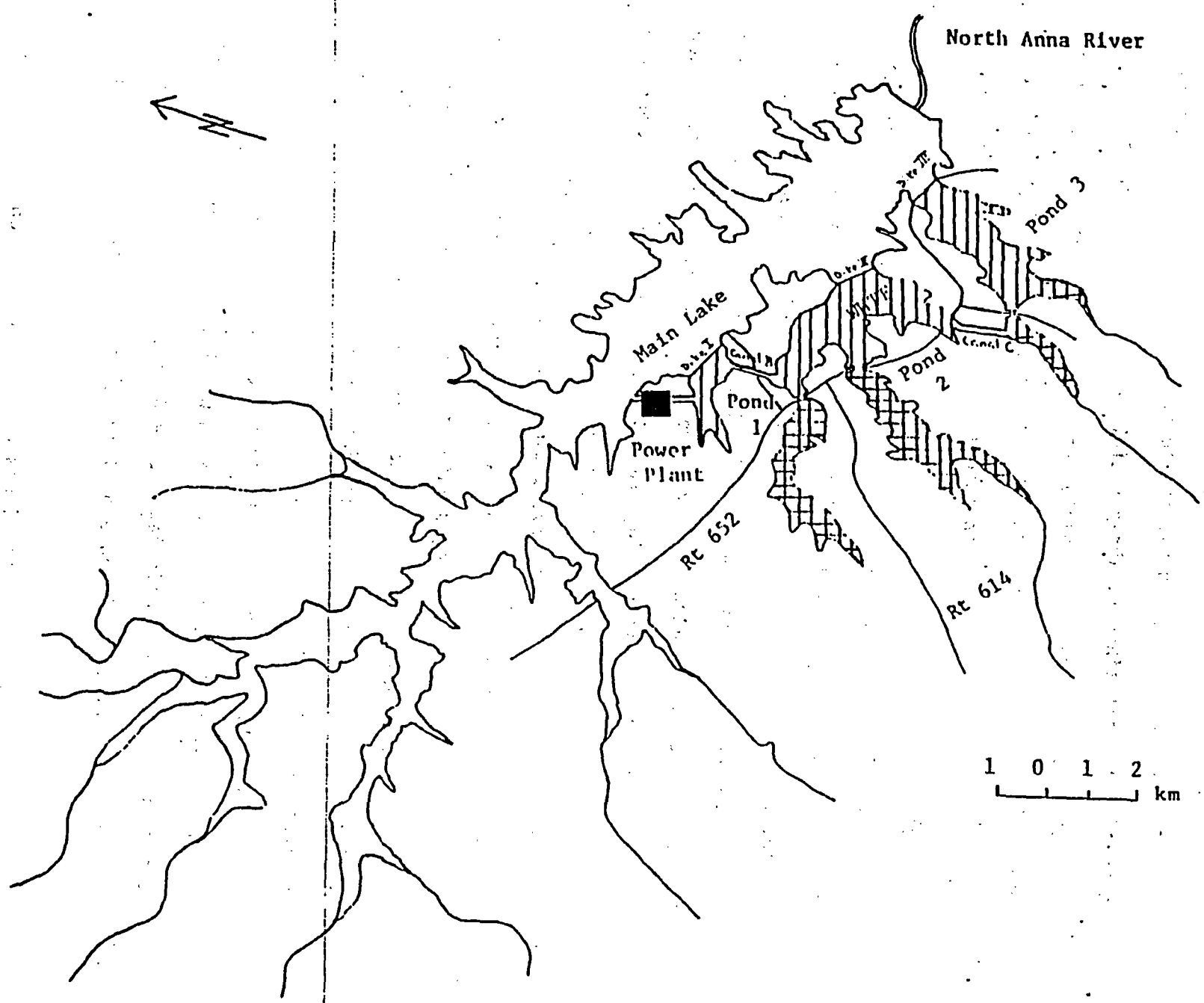


Figure 1.2 North Anna Cooling Lake System

average depth of 7.6 m *. The maximum depth at the dam is 24 m. The lake receives an average annual inflow of about 7.6 $m^3 s^{-1}$. The lake elevation is maintained by three radial gates at the dam and two near-surface skimmers. The outflow rate equals the inflow rate minus the rate of evaporation from the lake surface which is estimated at about 1.7 $m^3 s^{-1}$ for average conditions. Net groundwater seepage is negliglble.

The WHTF has a surface area of 14 km^2, a volume of 7.5×10^7 m^3, and an average depth of 5.5 m. The maximum depth is about 15 m in the vicinity of the dikes. Three dikes have been built to separate the WHTF from Lake Anna (see Figure 1.2). Dike 1 forms Pond 1 of the WHTF which receives the heated water via the discharge canal from the power plant (Figure 1.3). Connecting channels have been dredged between Pond 1 and Pond 2 (formed by Dike 2) and between Pond 2 and Pond 3 (formed by Dike 3). These channels have a constant trapezoidal cross-section of 7.6 m depth and 48.8 m width. After passing through Ponds 2 and 3, the cooling water is discharged into the main lake through a submerged discharge structure at Dike 3. After residence in the main lake, cooling water is withdrawn through intakes over the upper 9 m in the upstream vicinity of the station.

In summary, the cooling lake system of the North Anna Power Station is essentially a closed system which consists of a series of ponds (the WHTF) and a main lake (Lake Anna). A major characteristic of the system is the existence of long narrow side arms in the WHTF.

*all figures related to lake dimension, etc. in Section 1.3 are adopted from Wells et al (1982).

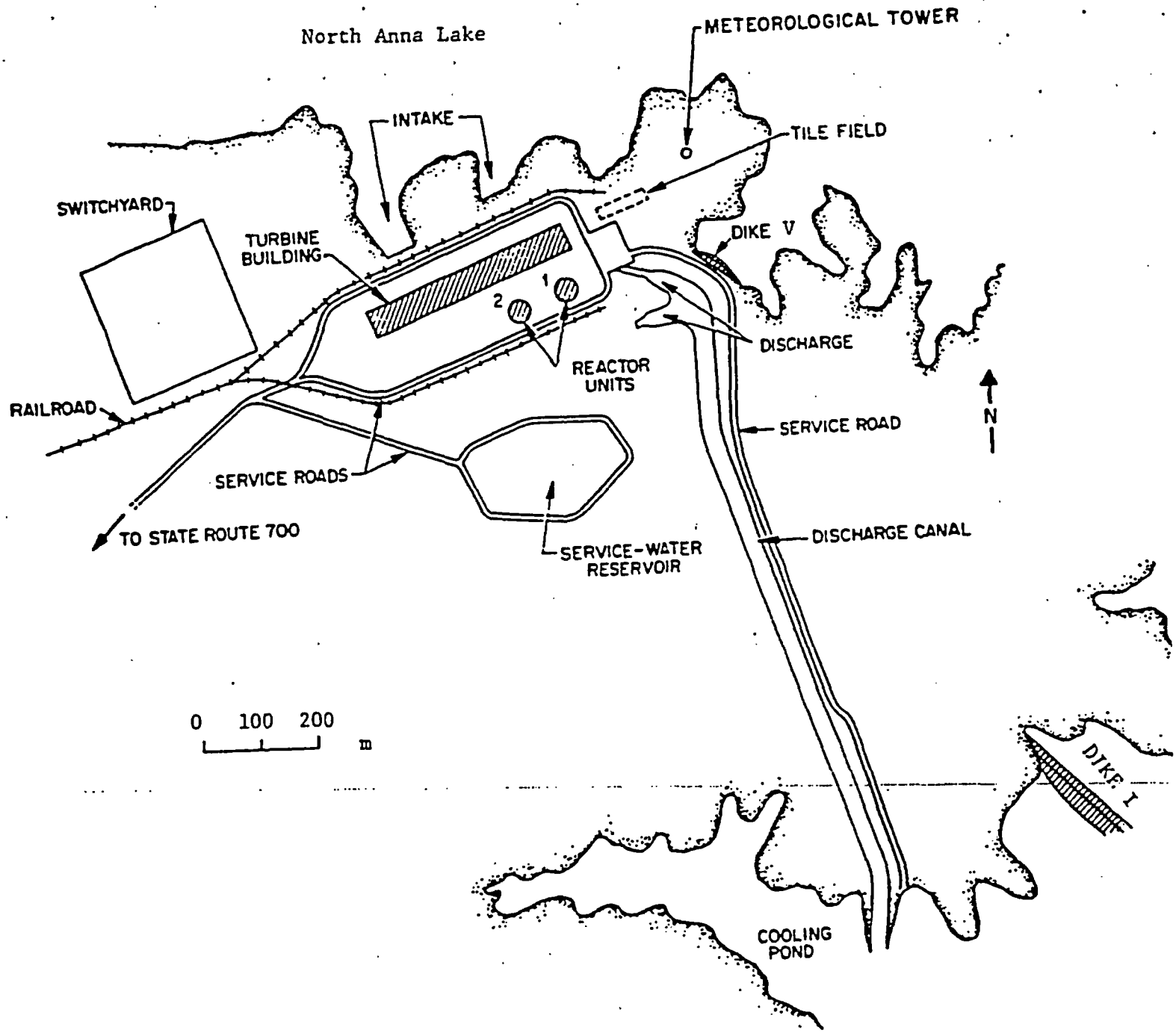


Figure 1.3--Discharge Canal from the Power Plant

These side arms comprise about 6.2 km^2 or 45% of the total surface area of the WHTF.

2. REVIEW OF PREVIOUS INVESTIGATIONS

2.1 Introduction

Proper understanding of the thermal structure of an artificially heated water body such as the cooling lake system of the North Anna Powr Station is fundamental in assessing the impacts that the waste heat discharge has on the natural environment. This should be apparent since most of the physical, chemical and biological processes which take place in an aquatic ecosystem are closely tied to the temperature regime. From the standpoint of the electric power generation industry, assessment of the efficiency of a cooling lake system also necessitates the study of the thermal structure of the cooling lakes through which waste heat is dissipated.

Mathematical models were developed to simulate the temperature regime of the cooling lake system at North Anna Power Station. This represents a continuous effort which commenced before the power plant first started its one-unit operation in July, 1978. Technical reports which resulted from the earlier development stage, and upon which the present research is based, include Watanabe et al. (1975), Brocard et al. (1977), Octavio et al. (1977), Jirka et al. (1977) and Wells et al. (1982). Since this research effort is coming to a natural conclusion, it is appropriate for the author to give a brief review of the previous investigations so that one may see the line along which the mathematical model has been developed.

2.2 Initial Model Development Stage

Watanabe et al. (1975) indicated the limitations of the then existing model for predicting cooling lake behavior. He rightly

assessed that steady-state models are of limited value in design and prediction owing to the inherently unsteady meteorology and plant heat loading. He developed a transient stratified cooling lake model which included a two-dimensional horizontal temperature prediction on the surface layer by improving Ryan and Harleman's model (1973).

Before Brocard et al. (1977), very little research effort had been made to determine the effects of side arms on the overall performance of a cooling lake, although it had been known for a long time that side arms are a common feature of cooling lakes with natural shorelines and that they may constitute a large portion of the cooling lake's total surface area. Brocard et al. (1977) developed a model for the side arm circulation applicable to both laminar and turbulent flows involving the division of the side arm in a "two layer flow region" and a "downflow region". Since then the side arm model has become an integral part of our present cooling lake model.

Octavio et al. (1977) dealt with basic research into the physical processes affecting the temperature structure of lakes and reservoirs. The M.I.T. Reservoir Model was modified to include the influence of wind via an iterative heating-wind mixing procedure. The natural (ambient) temperature regime of Lake Anna was predicted using the modified model and good agreement between predictions and measurements were reported.

The North Anna Cooling Lake Model began to assume its present form with the effort contributed by Jirka et al. (1977). A transient segmented cooling lake model was developed which links the mathematical models applicable to the main lake and those to the components of the WHTF. The model was then utilized in ten-year simulations (with

synthetic meteorological data series) to evaluate the thermal impacts of the power plant upon the lake system under one, two, three and four-unit operations.

2.3 Late Model Development Stage

While initial development of the North Anna Cooling Lake Model, calibration based on pre-operational data, and predictions under various levels of station operation for the historical period of 1957 - 1966 were documented in Jirka et al. (1977), no validation of the model was presented with operational data until Wells et al. (1982).

Since Unit 1 of the nuclear power plant came on line in summer of 1978 (followed by Unit 2 in autumn of 1980), an extensive data collection effort was undertaken to calibrate the model to operational data. Continuous measurements were made of meteorological variables, plant load, water temperature at various points in the system and current speeds in one of the WHTF side arms. Supplemental data, with greater spatial resolution, were collected at weekly or monthly intervals. Calibration of both the hydraulic and hydrologic components of the model were made in the following areas: (1) surface heat transfer processes, (2) temperatures in the main lake and (3) temperatures in the WHTF. (A review of the calibrations and changes in the model is given in Section 3.6.) Predictions given by the calibrated model were then compared against a three-year record of operational data (1978-1981).

3. NORTH ANNA COOLING LAKE MODEL

3.1 Introduction

A detailed description of the modelling approach as well as the development and interfacing of the various components of the North Anna Cooling Lake Model was given by Jirka et al. (1977). A summary of the basic structure, as originally developed, was included in Chapter 2 of Wells et al. (1982). Much of the material in the following sections of this chapter is summarized from their work.

The North Anna Cooling Lake Model is a segmented mathematical model which consists of different models for different parts of the North Anna lake system. A schematization of the geometry used in the modelling approach is illustrated in Figure 3.1. Specifically, three distinct features are significant:

(1) the three cooling ponds of the WHTF,

(2) the dead-end side arms of the WHTF and that of the main lake, and

(3) the main lake.

In the following sections a summary of the basic structure of the sub-models applied to each part is presented.

3.2 Model for Waste Heat Treatment Facility

3.2.1 Outline of Model

Three geometrically and structurally different components within the WHTF are recognized: (1) the three WHTF ponds, (2) the long dead-end side arms of the WHTF ponds, and (3) the interconnecting

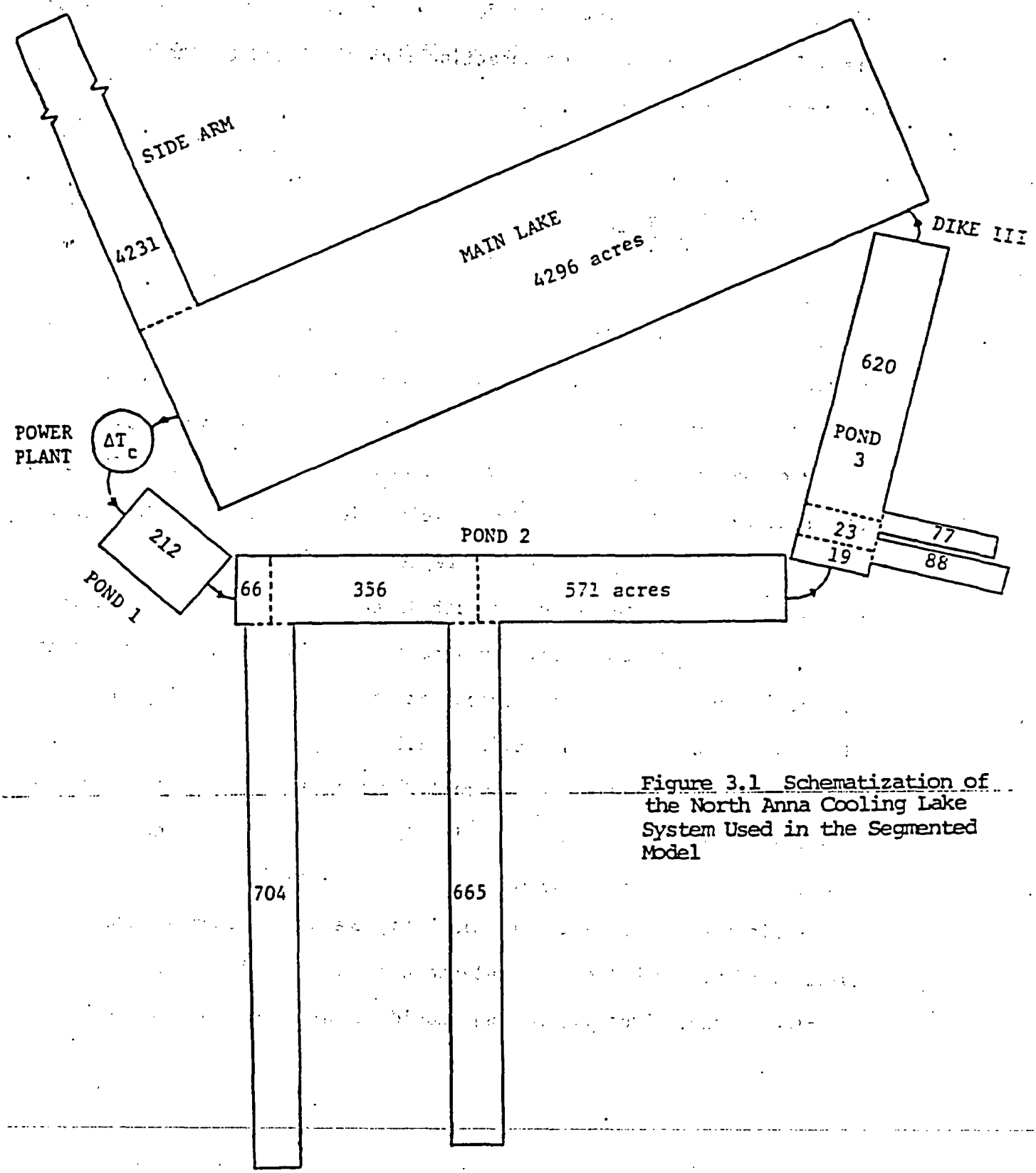


Figure 3.1 Schematization of the North Anna Cooling Lake System Used in the Segmented Model

channels. Side arm modelling is discussed in Section 3.2.2. Two different models are postulated for the vertical thermal structures of the WHTF ponds and the interconnecting channels based on their densimetric Froude numbers:

$$Fr = \frac{u}{\sqrt{\frac{\Delta\rho}{\rho} gh}} \qquad (3.1)$$

where u = characteristic velocity, g = gravitational acceleration, h = characteristic water depth, $\Delta\rho$ = characteristic density difference between upper and lower layers, and ρ = reference water density.

Because of the relatively large dimensions of the three WHTF ponds (refer to Section 1.3), the average velocities are low, and therefore $Fr < 1$. The WHTF ponds are thus expected to stratify; consequently, a two-layer model is used in which each layer is assumed to be vertically well mixed, with neither heat nor mass flux allowed to pass through the interface between the layers, except at the ends. (Modelling of the WHTF ponds is discussed in Section 3.2.3.)

Conversely, because of the small dimensions of the interconnecting channels, velocity is high, rendering $Fr > 1$, and thus the channels are modelled as a fully mixed system.

A significant parameter in modelling the WHTF system is the quantity of mixing between an interconnecting channel and the downstream pond. Ideally, mixing should be minimized to promote

maximum heat dissipation. The following empirical formula is utilized to calculate the dilution, D_s (Jirka, Abraham and Harleman, 1975):

$$D_s = 1.4 \sqrt{1 + Fr^2 \left(\frac{h}{b}\right)^{1/2}} \left[\frac{0.75}{h_{max}/H}\right]^{0.75} \qquad (3.2)$$

where D_s = dilution ratio, $(Q_o + Q_e)/Q_o$, Q_o = discharge flow rate, Q_e = entrained flow rate, h = depth of discharge channel, b = half-width of discharge channel, h_{max} = maximum jet penetration of deep-water jet, H = water depth, Fr = densimetric Froude number within the interconnecting channel.

The value of h_{max} is calculated from the buoyant surface jet model of Stolzenbach and Harleman (1971):

$$\frac{h_{max}}{H} = 0.42 \, F_r \left[\frac{h}{b}\right]^{1/4} \sqrt{hb} \qquad (3.3)$$

Data supporting Eqs. 3.2 and 3.3 can be found in Jirka et al. (1981).

3.2.2 Model for Side Arm

The convective circulation in a dead-end side arm (schematized in Figure 3.2) is a phenomenon whereby warm surface water from the main pond spreads into the side arm, gradually losing its heat to the atmosphere. The gradual decrease in density difference causes the inflowing water to sink and be replaced by new warm water. In the context of the North Anna Model, the entrance temperature distribution

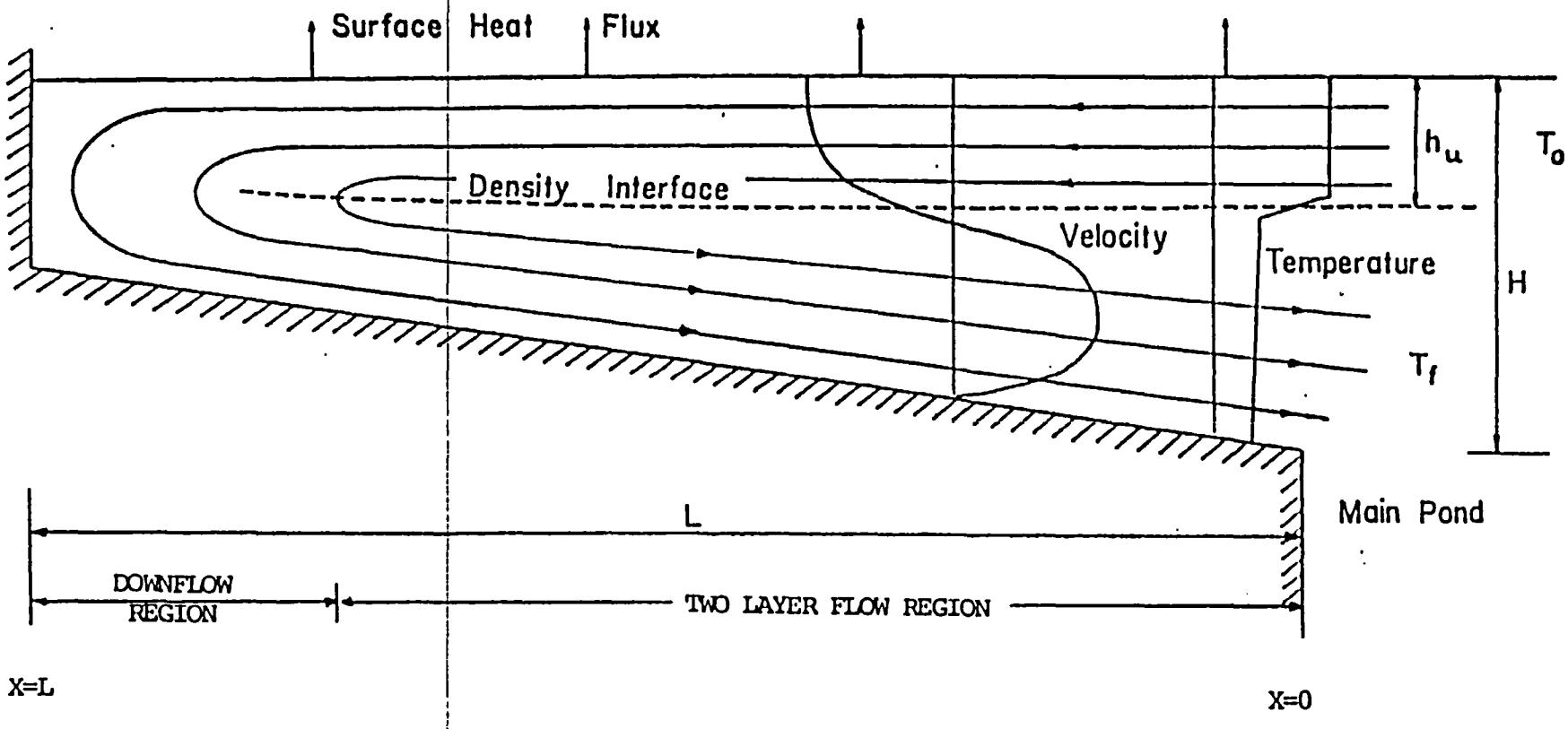


Figure 3.2 Schematization of Convective Circulation in a Dead-End Side Arm

and the surface heat fluxes are the independent variables, while the flow rate and the temperature of the return flow are to be determined.

This phenomena was studied by Brocard et al. (1977) who presented a model for solving the problem. The following assumptions were made to simplify the analysis:

(1) horizontal bottom, i.e., $dH/dx \approx 0$

where H = depth of the side arm,

x = longitudinal distance up the side arm;

(2) downflow occurs only at the end of the side arm;

(3) small values of $KL/\rho C_p q_o$,

where L = length of side arm,

q_o = side arm flow per unit width,

ρ = density of water,

C_p = specific heat of water,

K = surface heat exchange coefficient;

(4) negligible effect of the lateral bridge constriction on side arm flow.

Their equation derived for the side arm flow is:

$$\frac{q_o}{KL} = \left\{B \frac{f_o L}{H} \frac{1}{4h_o} \left[\alpha\left(\frac{1}{h_o} - \frac{\Delta q-1}{1-h_o}\right)^2 \left(\frac{1}{h_o} + \frac{1}{1-h_o}\right) + \frac{(\Delta q-1)^2}{(1-h_o)^3}\right]\right\}^{-1/3} \qquad (3.4)$$

where B = buoyancy term = $\dfrac{k^2 L^2}{\beta(T_o - T_E)gH^3}$

h_o = initial upper layer depth

β = coefficient of thermal expansion of water = $-\frac{1}{\rho}\frac{\partial \rho}{\partial T}$

$\Delta q = \frac{q_1 - q_2}{q_o} = 0$

$\alpha = \frac{f_i}{f_o} = 0.5$

T_o = initial temperature at side arm entrance

T_E = equilibrium temperature

f_o = bottom friction factor

f_i = interfacial friction factor

q_1 = upper layer flow

q_2 = lower layer flow

k = kinematic surface heat exchange coefficient = $\frac{K}{\rho C_p}$

The temperature distribution along the side arm can be determined once q_o is known. In the North Anna Model, the equation used for the return temperature has been

$$\frac{T_f - T_E}{T_o - T_E} = \exp(-0.8 \frac{kL}{q_o}) \qquad (3.5)$$

where T_f = final or return temperature. The 0.8 factor in Eq. 3.5 can be thought of a dispersion effect, reducing the "effective" length of the side arm by 20%.

In order to solve for the side arm flow rate and the return temperature, the mixed layer depth and the temperature of the mixed layer in the WHTF pond are inputs to the side arm model together with the side arm geometry and meteorological conditions.

3.2.3 Model for WHTF Ponds

Two situations exist with regard to the mathematical modelling of the temperature distribution in the three WHTF ponds:

(1) one pond with no side arms (Figure 3.3) and

(2) two ponds each with two side arms (Figure 3.4).

For the pond with no side arms, the temperature of the entrained water is equal to T_2, the temperature at the end of the pond, since no heat flux is allowed through the interface. By means of a heat balance, the temperature of the upper layer waters at the end of the mixing zone can be given by

$$T_1 = \frac{T_o + (D_s - 1)T_2}{D_s}$$

or

$$T_1 - T_E = \frac{(T_o - T_E) + (D_s - 1)(T_2 - T_E)}{D_s} \qquad (3.6)$$

If the pond is treated as a one-dimensional steady state system, the governing differential equation along it becomes

$$\frac{dT}{dA} = \frac{\phi_n}{\rho C_p Q} = \frac{-K(T - T_E)}{\rho C_p Q} \qquad (3.7)$$

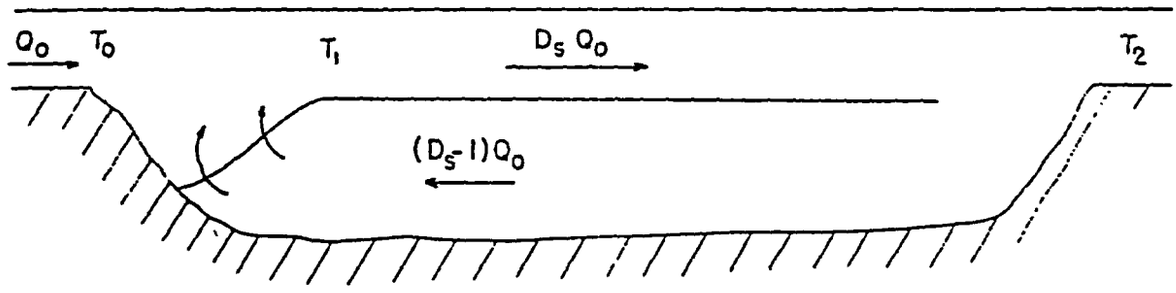
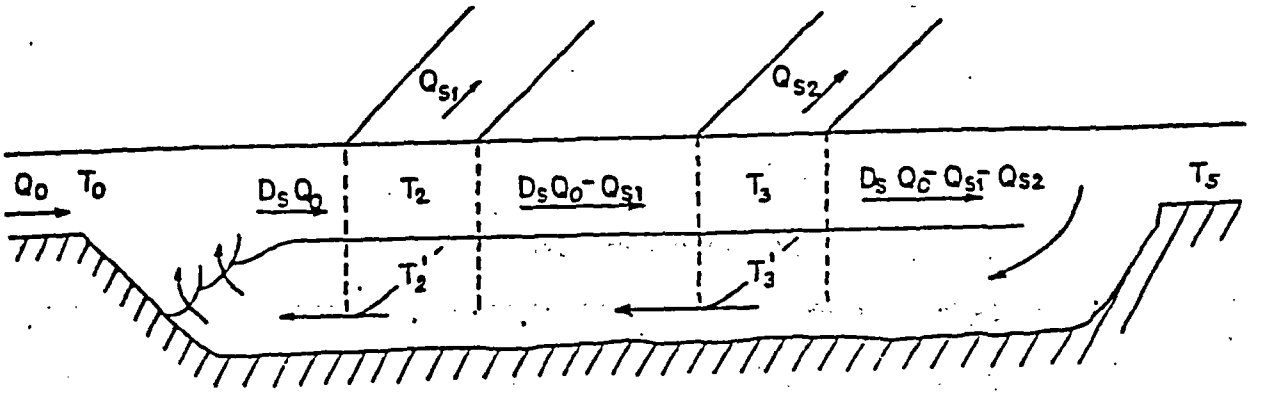
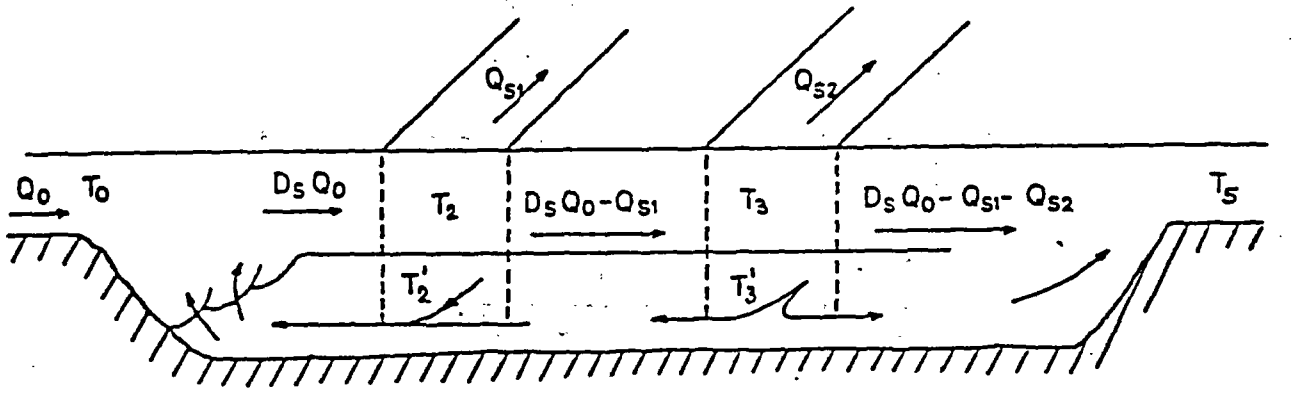


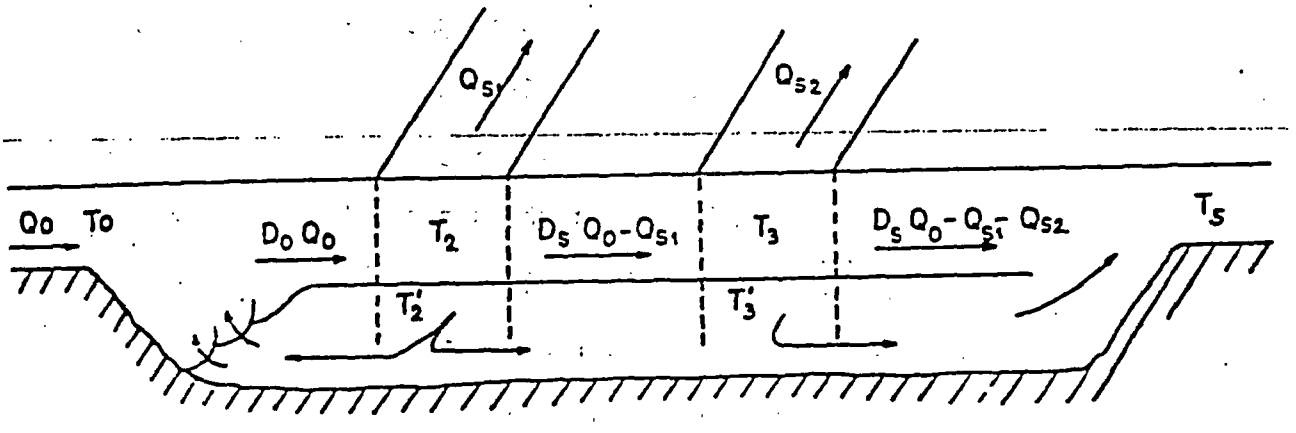
Figure 3.3 Flow Configuration of a Pond without Side Arm



A. Flow Configuration in a Reach with Two Side Arms for $(D_s-1)\ Q_o > Q_{s1} + Q_{s2}$



B. Flow Configuration in a Reach with Two Side Arms for $Q_{s1} < (D_s - 1)\ Q_o < Q_{s1} + Q_{s2}$



C. Flow Configuration in a Reach with Two Side Arms for $(D_s - 1)\ Q_o < Q_{s1}$

Figure 3.4 Flow Configurations of a Pond with Two Side Arms

where A = incremental area,

Q = flow rate in upper layer,

ϕ_n = net heat flux.

By integrating Eq. 3.7 across the pond, the steady state equation for the temperature distribution in the first WHTF pond (Pond 1) is

$$\frac{T_2 - T_E}{T_1 - T_E} = \exp\left(- \frac{KA}{\rho C_p D_s Q_o} \right) = e^{-r} \qquad (3.8)$$

From Eqs. 3.6 and 3.8 one obtains the following equation for Pond 1:

$$\frac{T_2 - T_E}{T_o - T_E} = \frac{e^{-r}}{D_s - (D_s - 1)e^{-r}} \qquad (3.9)$$

As shown in Figure 3.4, the flow in a pond with two side arms can take on three different possible configurations:

(1) the jet entrainment flow is greater than the sum of the side arm flows, i.e., $(D_s-1)Q_o > Q_{s1} + Q_{s2}$,

(2) the jet entrainment flow is greater than the first side arm flow but smaller than the sum of the side arm flows, i.e., $Q_{s1} < (D_s-1)Q_o < Q_{s1} + Q_{s2}$, and

(3) the jet entrainment flow is smaller than the first side arm flow, i.e., $(D_s-1)Q_o < Q_{s1}$.

Analysis of the thermal structure in the case of a pond with two side arms is simliar to that of a pond without side arm: the same governing equation (Eq. 3.7) is integrated over sections of the pond

between side arm locations and then simplified by a heat balance describing mixing of the flow within the pond and at the side arms' entrance or exit. Since the results are quite lengthy, they are not reproduced here; the reader is referred to pages 64-80 of Jirka et al. (1977). Also since the solution for the pond temperature in this case is complicated by the different possible configurations of the flow structure, an iterative process must be used in the computation.

3.3 Model for Dike III Mixing

A plan view of the submerged jet discharge at Dike 3 is given in Figure 3.5. The jet model of Stolzenbach and Harleman (see Eqs. 3.2 and 3.3) is invoked to evaluate entrance mixing. However, a condition of critical flow at the triangular constriction (Figure 3.6) dictates the maximum exchange flow:

$$\frac{(Q_o + Q_e)^2/A_T^2}{\frac{\Delta\rho}{\rho} g\, h_T} + \frac{Q_e^2/A_B^2}{\frac{\Delta\rho}{\rho} g h_B} = 1 \qquad (3.10)$$

where A_T = area of top section,

h_T = depth of top section,

A_B = area of bottom section,

h_B = depth of bottom section,

Q_e = entrained flow,

Q_o = jet discharge.

One can notice that the computed entrainment flow may exceed the flow which could be exchanged across the section. An analysis of critical

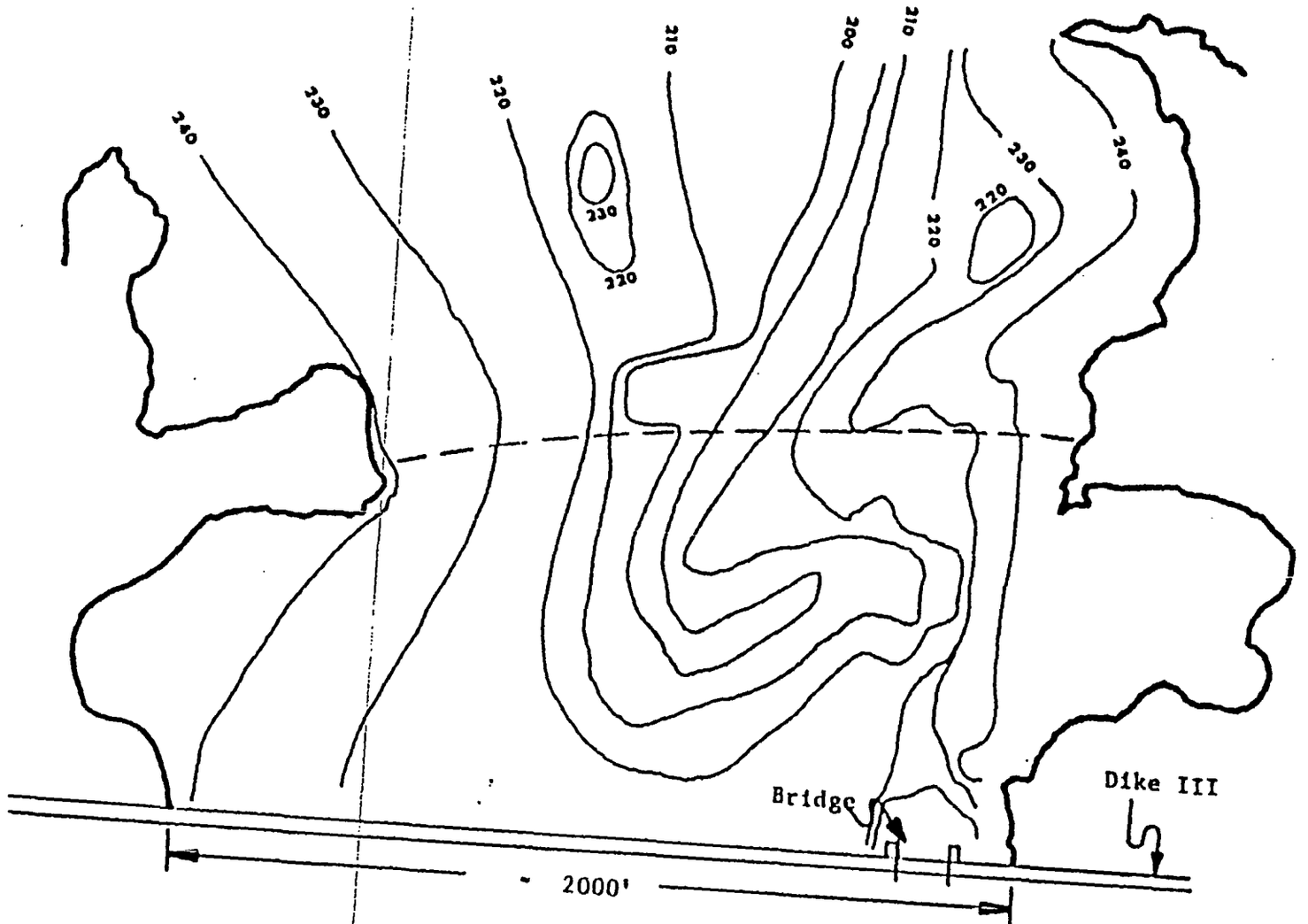


Figure 3.5 Plan View of Submerged Jet Discharge at DIKE III
(Contour Lines are in Feet above MSL)

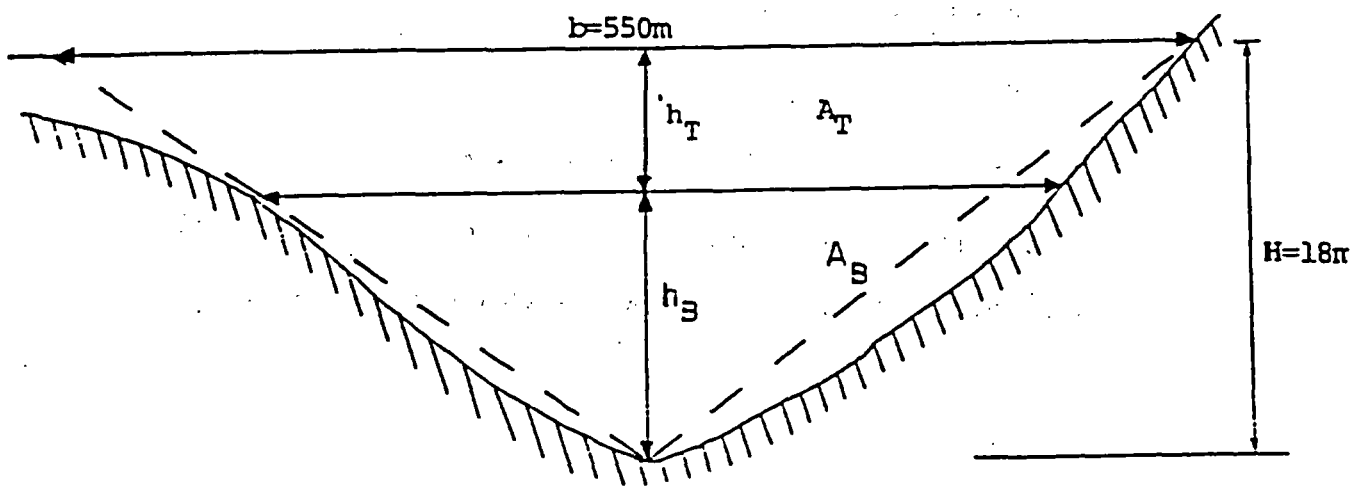


Figure 3.6 Cross Section of Dike III Constriction

flow provides an equation in dimensionless form to compute the entrance dilution based on the geometry of the constriction:

$$\frac{(1 + \frac{Q_e}{Q_o})^3 \, (\frac{Q_e}{Q_o})^2 \, (1 + \frac{Q_e}{Q_o})}{(\frac{h_T}{H})^3 \, (2 - \frac{h_T}{H})^2 \, (1 - \frac{h_T}{H})^5} = \frac{b^2 H^2}{4 Q_o^2} \, g \, (\frac{\Delta\rho}{\rho})_o \qquad (3.11)$$

where $H = h_T = h_B$ = total depth,

b = top width,

$(\frac{\Delta\rho}{\rho})_o$ = value of $\Delta\rho/\rho$ between jet water and entrained cold water.

Eq. 3.11 is solved by trial and error for the maximum value of Q_e/Q_o which obeyed the stipulation that $0 < h_T/H < 1$.

3.4 Main Lake Model

3.4.1 Outline of Model

The main lake (Lake Anna) is modelled according to its three different sections:

(1) a vertically well-mixed surface layer of constant thickness and horizontally-varying temperture distribution $T(x,t)$,

(2) a vertically stratified subsurface pool of uniform horizontal temperature distribution $T(z,t)$, and

(3) a side arm attached to the end of the main lake that has a return flow into the subsurface pool.

The combined surface and subsurface models are essentially modified from Ryan and Harleman (1973), while the analysis of the side arm flow

follows that discussed in Section 3.2.2.

3.4.2 Finite Difference Model for Surface Layer

A one-dimensional, transient model is used based on the following governing equation:

$$\frac{\partial T}{\partial t} = -\frac{Q}{H_s}\frac{\partial T}{\partial A} - \frac{\phi_n}{\rho C_p H_s} \qquad (3.12)$$

where H_s = depth of surface layer,

ϕ_n = net surface heat flux.

Eq. 3.12 is put into finite difference form, and the surface temperature along the lake, in 40 areal increments, is predicted as a function of time. Using a steady-state momentum equation for the heated surface layer, Watanabe et al. (1975) developed the following expression for the value of H_s:

$$H_s = \left(\frac{f_i Q_o^2 D_{sv}^3 L^3}{4\beta \Delta T_L g A_p^2}\right)^{1/4} \qquad (3.13)$$

where f_i = interfacial friction factor

Q_o = discharge flow rate

D_{sv} = vertical entrance dilution factor

A_p = total pond area

L = longitudinal pond dimension

β = thermal expansion coefficient

ΔT_L = surface temperature difference between dam and intake

g = gravitational acceleration

3.4.3 Finite Difference Model for Subsurface Layer

The hypolimnion of Lake Anna is modelled by a series of horizontally uniform layers which are vertically stratified. Heat transfer occurs by diffusion, advection, convective mixing and radiation absorption. The governing equations for this one-dimensional vertical model are as follows:

(1) Heat Transport Equation

$$\frac{\partial T}{\partial t} + \frac{1}{A}\frac{\partial}{\partial z}(Q_v T) = \frac{D_z}{A}\frac{\partial}{\partial z}\left(A\frac{\partial T}{\partial z}\right) + \frac{Bu_i T_i}{A} - \frac{Bu_o T}{A} - \frac{1}{\rho C_p A}\frac{\partial (A\phi_z)}{\partial z} \qquad (3.14)$$

(2) Surface Boundary Condition

$$D_z \frac{\partial T}{\partial z} = \phi_n \quad \text{at } z = z_s \qquad (3.15)$$

(3) Bottom/Side Boundary Condition

$$\frac{\partial T}{\partial z} = 0 \quad \text{at } z = 0 \qquad (3.16)$$

(4) Continuity Equation

$$Q_v = B\int_0^z u_i(z,t)dz - B\int_0^z u_o(z,t)dz \qquad (3.17)$$

where $\phi_z = \phi_o(1-\beta)e^{-\eta(z_s-z)}$

ϕ_o = net incident solar radiation

ϕ_n = net heat flux

β = fraction of short wave radiation absorbed at water surface (= 0.5)

η = extinction coefficient (= 0.75 m^{-1})

z_s = water surface elevation

B = width of main lake

u_i, u_o = velocity of inflow, outflow, respectively

Q_v = vertical flow (advection)

D_z = vertical diffusion coefficient

A = area of main lake

T_i = inflow temperature

These governing equations are put in finite difference form and solved along with the surface layer model using an explicit time scheme. At the end of each time step the vertical stability of the water column is checked and, if necessary, convective overturning is performed.

3.5 Summary of Model

In the North Anna Cooling Lake Model, the WHTF ponds and side-arms were formulated as basically steady state models, while the main lake formulation was transient. In order to account for the transients in the WHTF, a lagging criterion was used, based on the residence time of each pond. For example, the predicted temperature at day j, at the end of a pond with a residence time of n days, was calculated from the flow rate and initial temperatures for day j-n. In this manner temperatures were lagged throughout the WHTF. The flow rate and temperature computed for the end of Pond 3 became the inflow rate and inflow

temperature for the Dike 3 jet mixing formulation of the main lake.

The diluted flow in the upper layer of the main lake exceeds the condenser flow rate withdrawn at the intake by the entrainment associated with Dike 3 mixing (Figure 3.7). This extra flow is down-welled to the sub-surface model. This down-welling takes place from the last longitudinal segment in the finite difference representation of the surface layer model into the top segment of the finite difference representation of the sub-surface model. To summarize, the inputs to and from the stratified sub-surface model are the Dike 3 entrainment, the main lake side arm return flow, the intake withdrawal and down-welled surface flow. The North Anna River inflow and outflow were neglected because of their generally small magnitudes in comparison to the condenser flow rate.

A time step of one day along with daily-averged input data was used for all simulations.

3.6 Previous Calibrations of Model

Wells et al. (1982) calibrated the model using the first three years of operational data. Compared with the original model used by Jirka et al. (1977) which was outlined in the previous five sections of this chapter, Wells et al. introduced the following basic changes:

(1) a formula by Idso and Jackson (1969) for calculating the net long-wave atmospheric radiation was used to replace the one by Swinbank (1963) in order to increase predicted temperatures during the winter,

(2) a periodic function was applied to adjust the evaporation formula used in the WHTF to account for the non-uniformity of

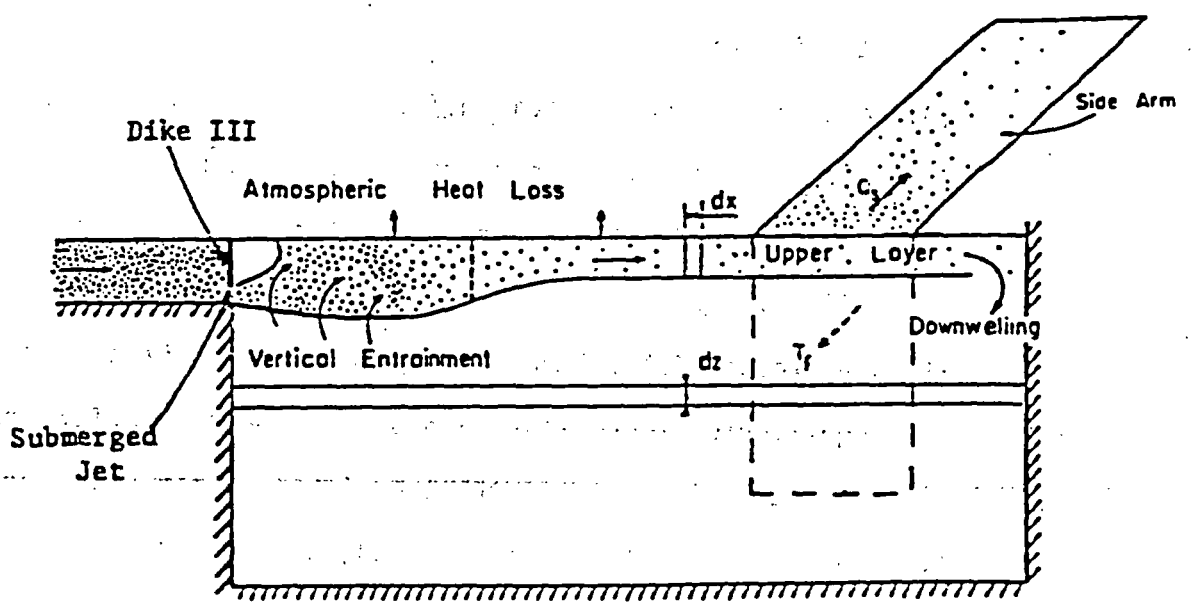


Figure 3.7 Schematization of Main Lake Model

air temperature, relative humidity and wind speed over the lake which were found to be seasonally dependent (see p. 74, Wells et al.),

(3) the condenser intake was allowed to withdraw water equally over the upper 9 m of the main lake to account for the two-dimensional characteristics of the thermal structure near the intake,

(4) a time-varying vertical diffusion coefficient, D_z, dependent on wind speed, condenser flow rate and vertical density gradient was used to replace the constant molecular diffusivity,

(5) an exponential filter was implemented for filtering both the equilibrium temperature and surface heat transfer coefficient to better account for the transients in the WHTF, and

(6) the side arm flow rates were adjusted based on the constrictions due to bridge piers (see also Adams and Wells, 1984).

In general, these modifications improved the predictive power of the model over the original version. A discussion of the performance of the calibrated model in a five-year simulation is given in the next chapter.

4. SURFACE TEMPERATURE ERROR ANALYSIS

4.1 Introduction

Since Unit 1 of North Anna Power Station came on line in summer of 1978 (followed by Unit 2 in autumn of 1980), an extensive data collection network was established by VEPCO. This program has continued, with minor modifications, through the autumn of 1983.

Input to North Anna Cooling Lake Model requires daily-averaged meteorological data (air temperature, wind speed, relative humidity, cloud cover and short wave solar radiation), plant flow rate, temperature rise across condenser, and an initial thermal structure of the main lake. These data are obtained along with information used for calibration purposes, such as water temperature at various points in the cooling lake system and current speeds in one of the WHTF side arms, and data used to evaluate downstream thermal impact and compliance measurements, such as flow data from the North Anna Dam. A detailed description of the data collection effort made by VEPCO is given in Chapter 3 of Wells et al. (1982) and thus will not be reproduced here.

Wells et al. (1982) calibrated the model using data segments from the first three years of operational data (1978-1981) and then validated the model by comparing the continuous model predictions with measurements over the same period. Since then more measurement data were available and a completely independent validation would be to compare model predictions with data for the following two years (1981-1983) using the same model calibrations.

Comparisons between model predictions and measurements are made for the surface temperatures at four diagnostic control points. For the sake of comparison, results of all five years of simulation (26 July 1978 to 30 September 1983 inclusive; a total of 1893 days) are presented in the plots (Figures 4.2 to 4.4). The first 1162 days are mere duplicates of what were presented in Wells et al. (1982).

4.2 Raw Error

Surface temperature error analysis is made for the four representative diagnostic control points over the cooling lake system (Figure 4.1). They are (1) at the discharge of Channel 1 into Pond 1 of the WHTF (hereafter referred to as DISCHARGE), (2) just upstream of Dike III in the WHTF (DIKE III), (3) in the main lake outside the Dike III jet mixing zone near Burrus Point (BR.PT.), and (4) in the main lake near the plant intake (INTAKE).

Figures 4.2a to 4.2d compare the observed surface temperatures with corresponding predictions for 1893 days since July 26, 1978. (Small gaps of missing measurement data of up to a continuous period of 15 days were filled in by curve-fitting.) While these plots show that the model simulates the surface temperatures with reasonable accuracy, a more rigorous error analysis necessitates plotting raw errors (henceforth defined as prediction - measurement) with time.

Figures 4.3a to 4.3d show the raw errors at the four diagnostic control points. (A positive error means that the model was over-predicting, and 15°C indicates missing data.) The raw errors seem to be occuring quite randomly at BR.PT. and INTAKE with mild amplitudes at the annual period; however, a conspicuous periodicity at the annual

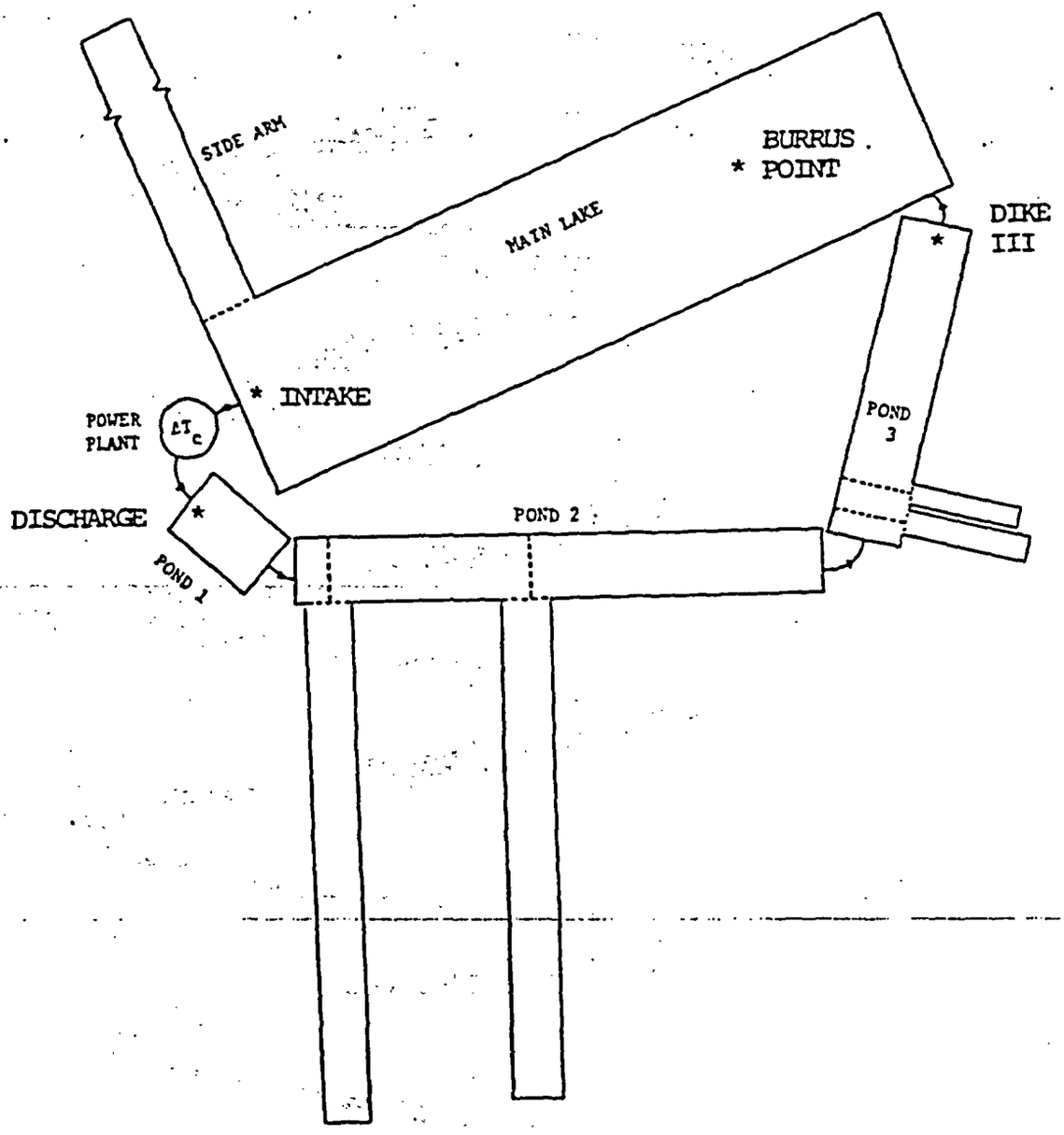


Figure 4.1 Schematization of the Locations of the Four Diagnostic Control Points

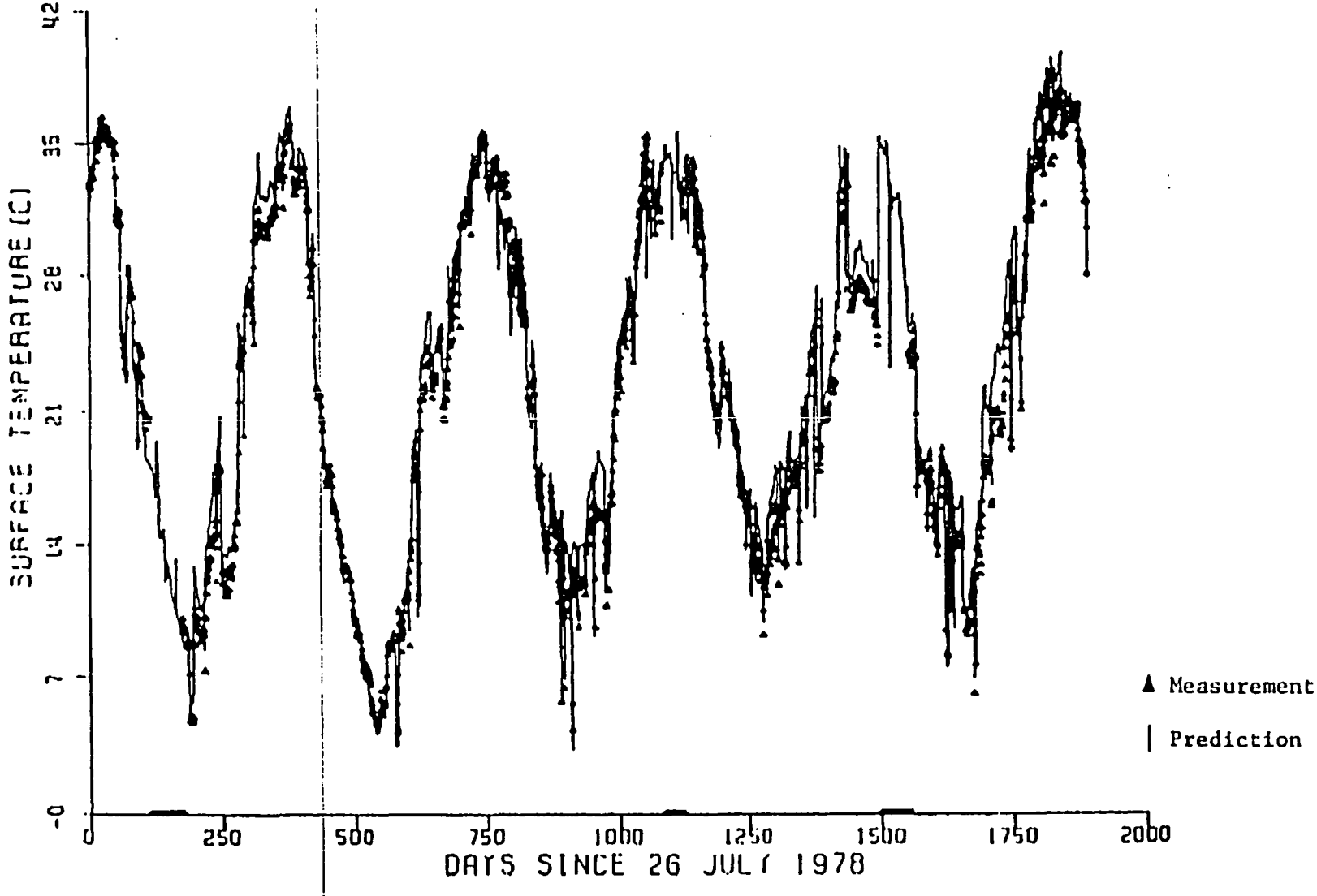


Figure 4.2a Measured vs. Predicted Surface Temperature at DISCHARGE

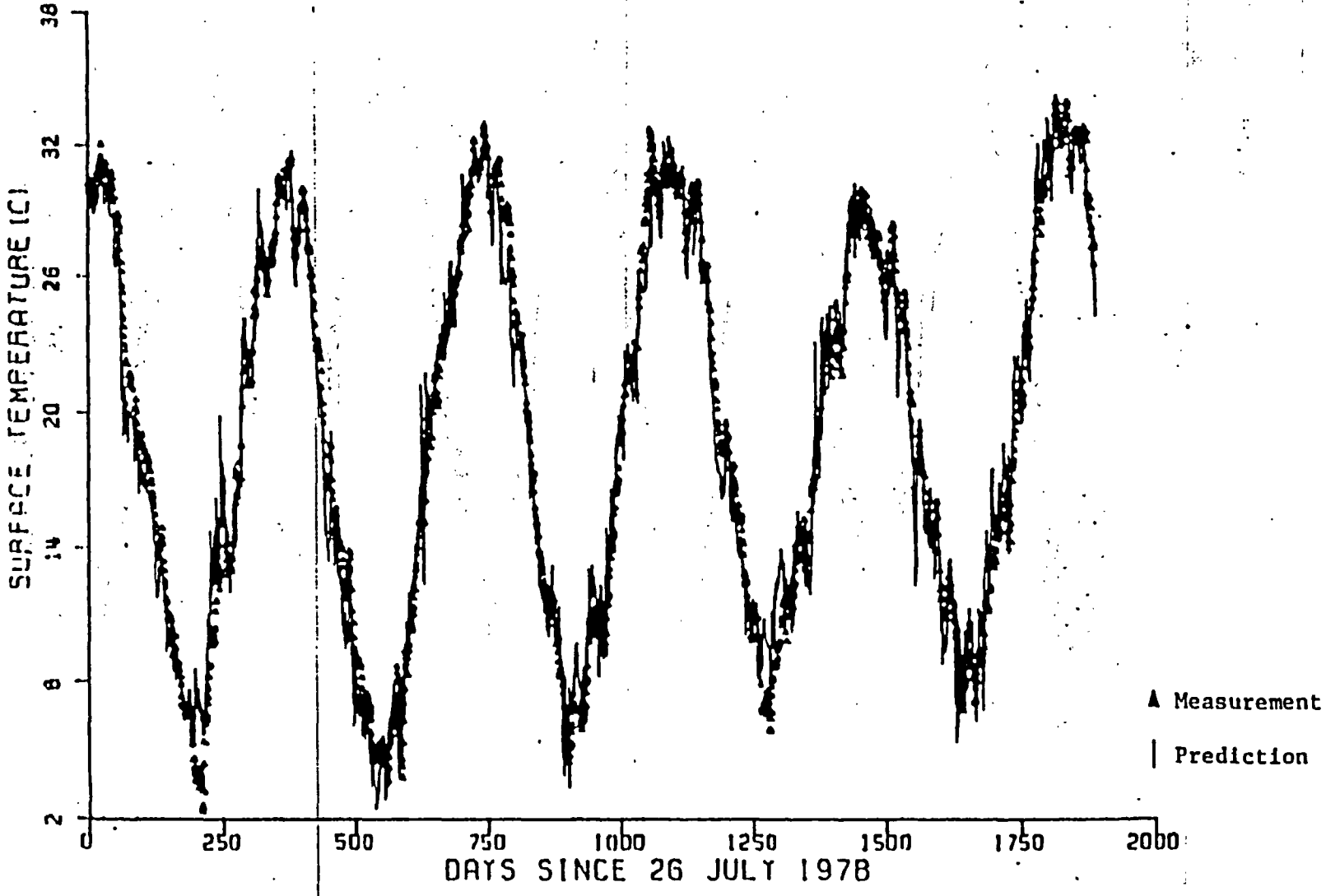


Figure 4.2b Measured vs. Predicted Surface Temperature at DIKE III

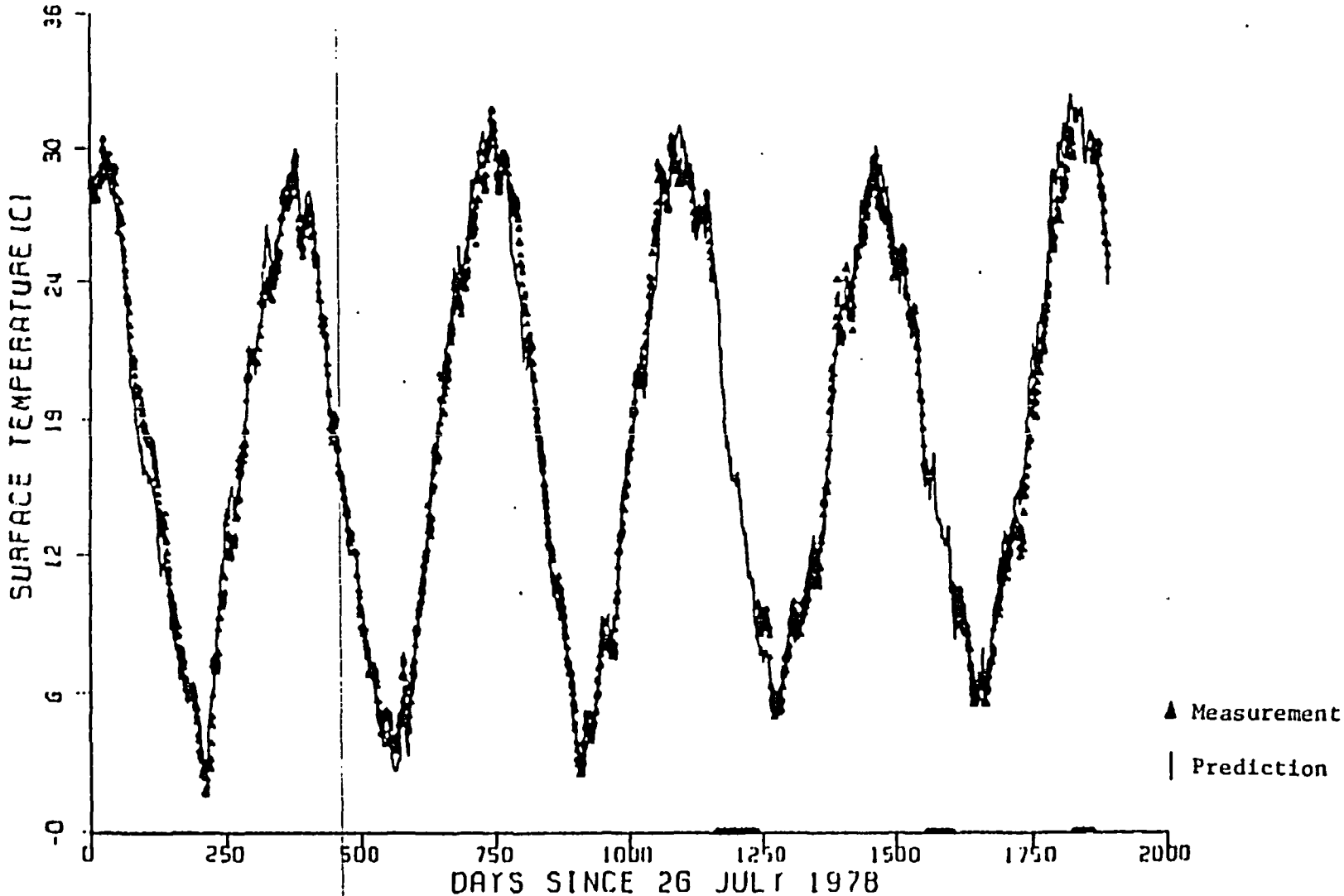


Figure 4.2c Measured vs. Predicted Surface Temperature at Bl .

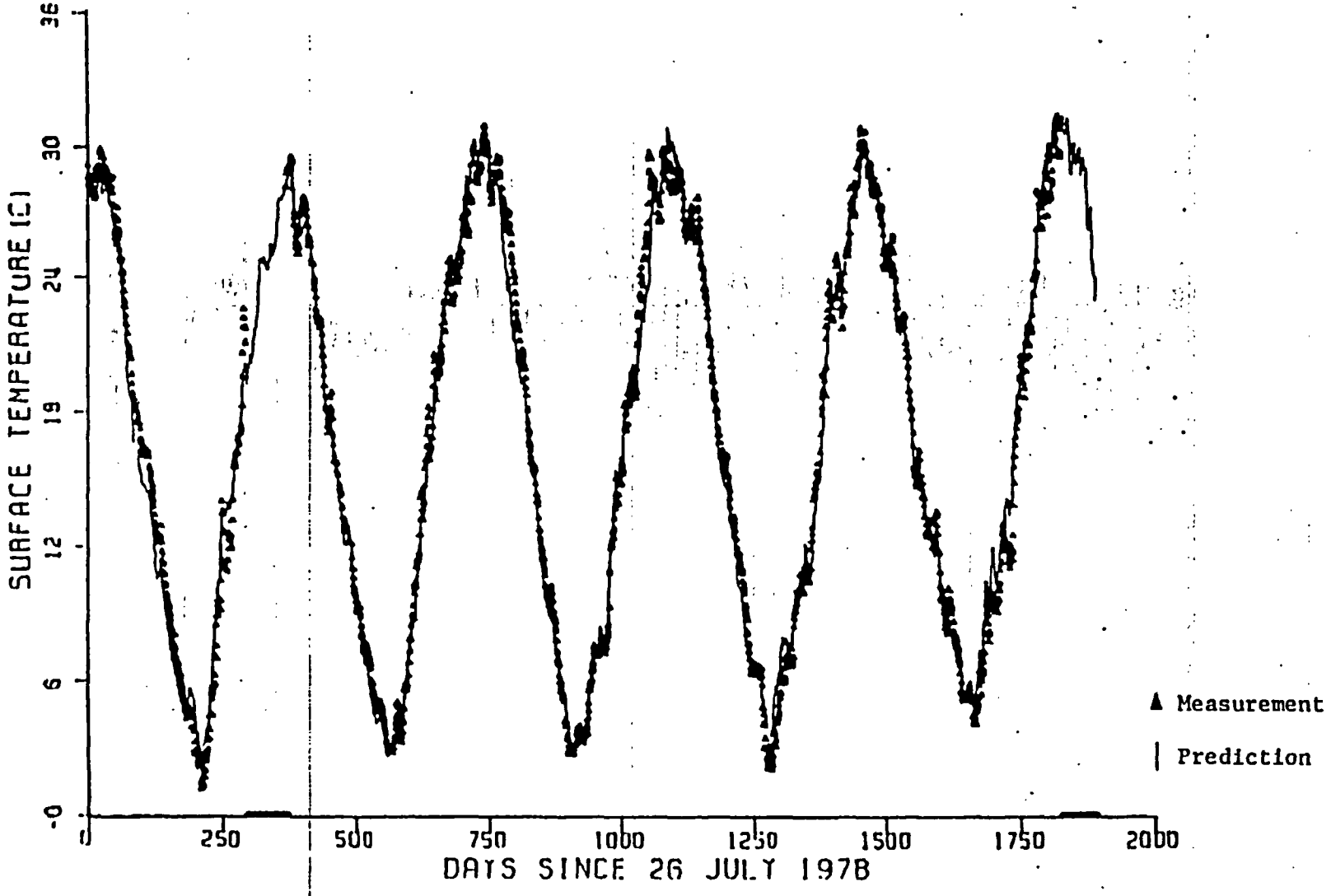


Figure 4.2d Measured vs. Predicted Surface Temperature at INTAKE

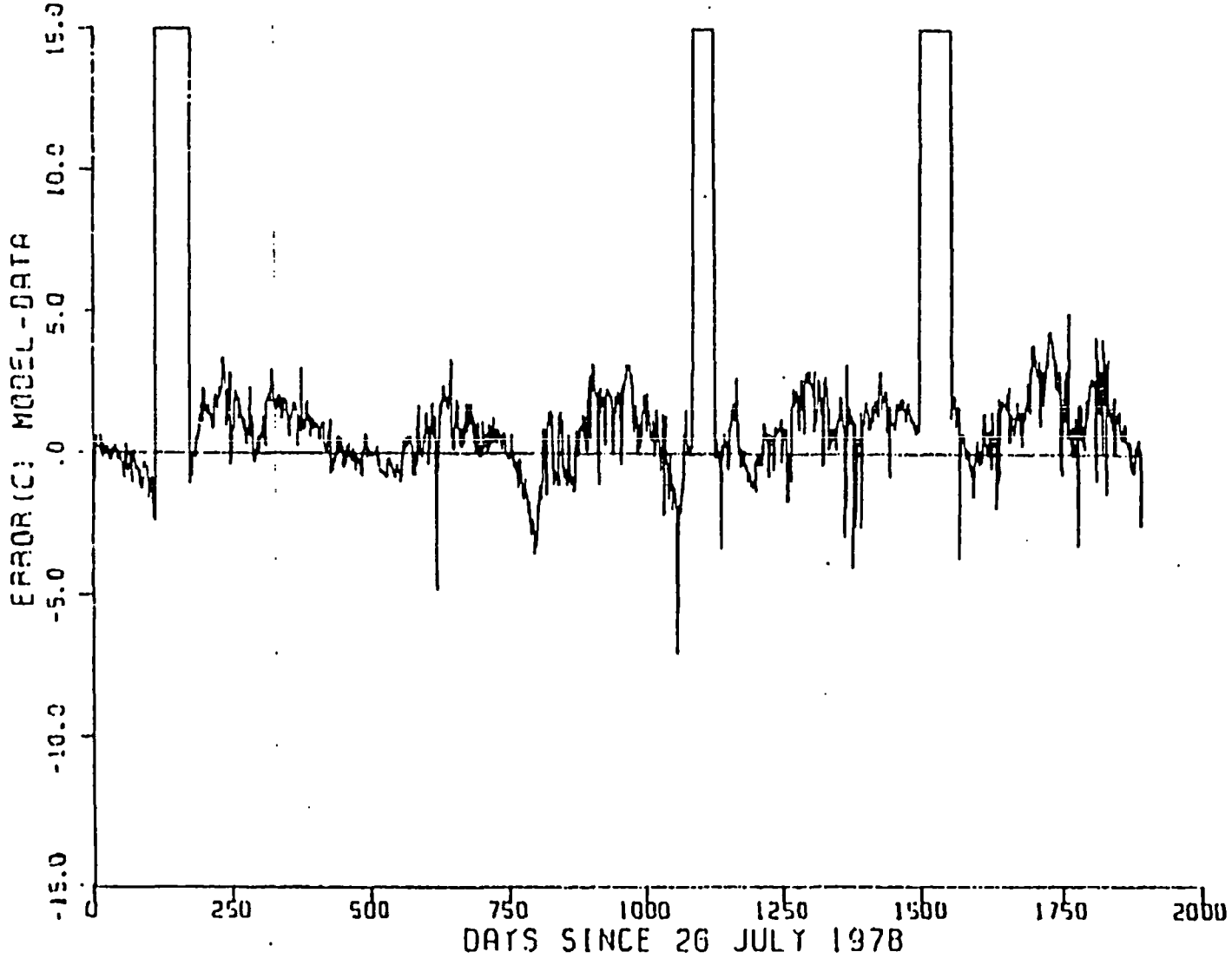


Figure 4.3a Raw Error at DISCHARGE

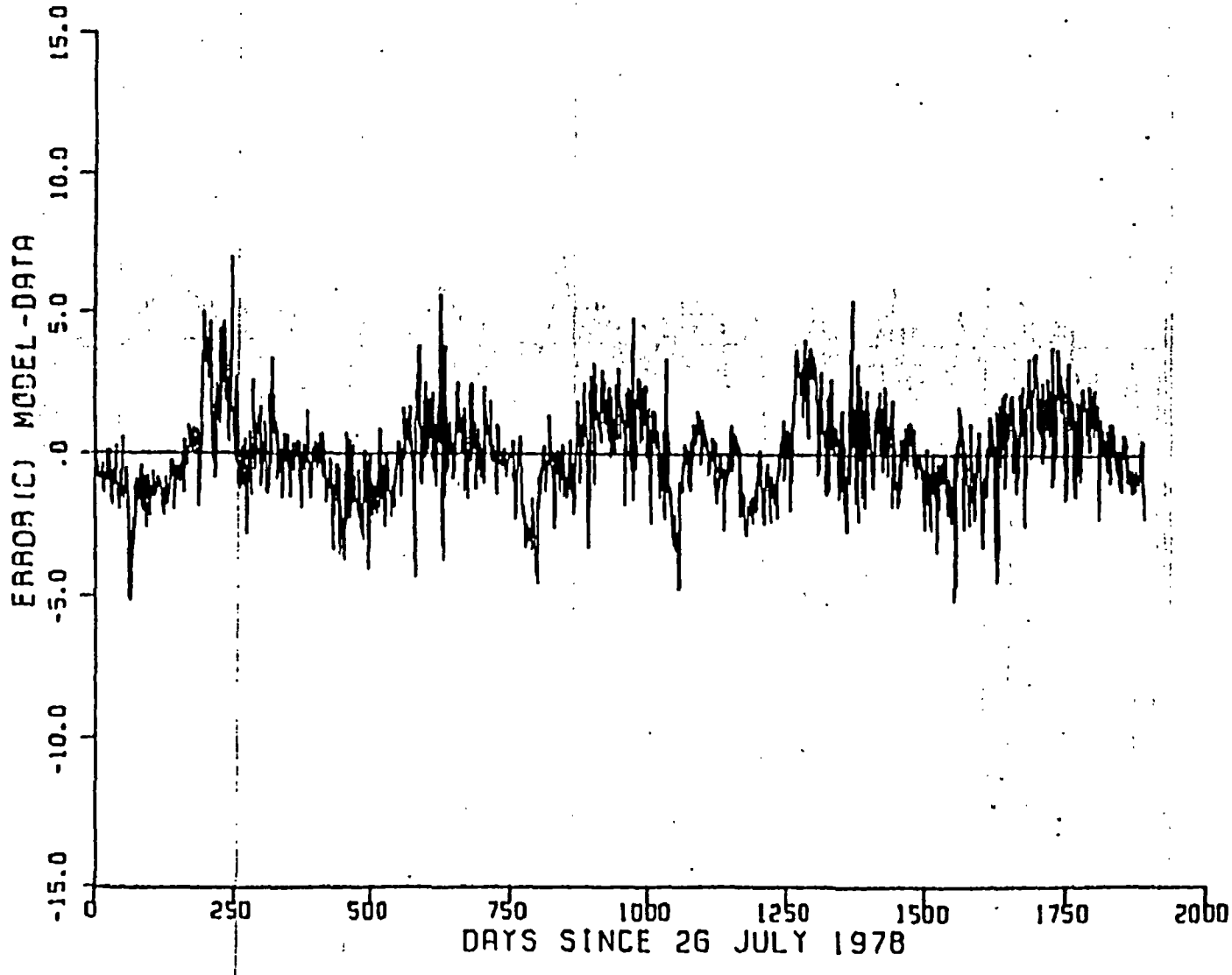


Figure 4.3b Raw Error at DIKE III

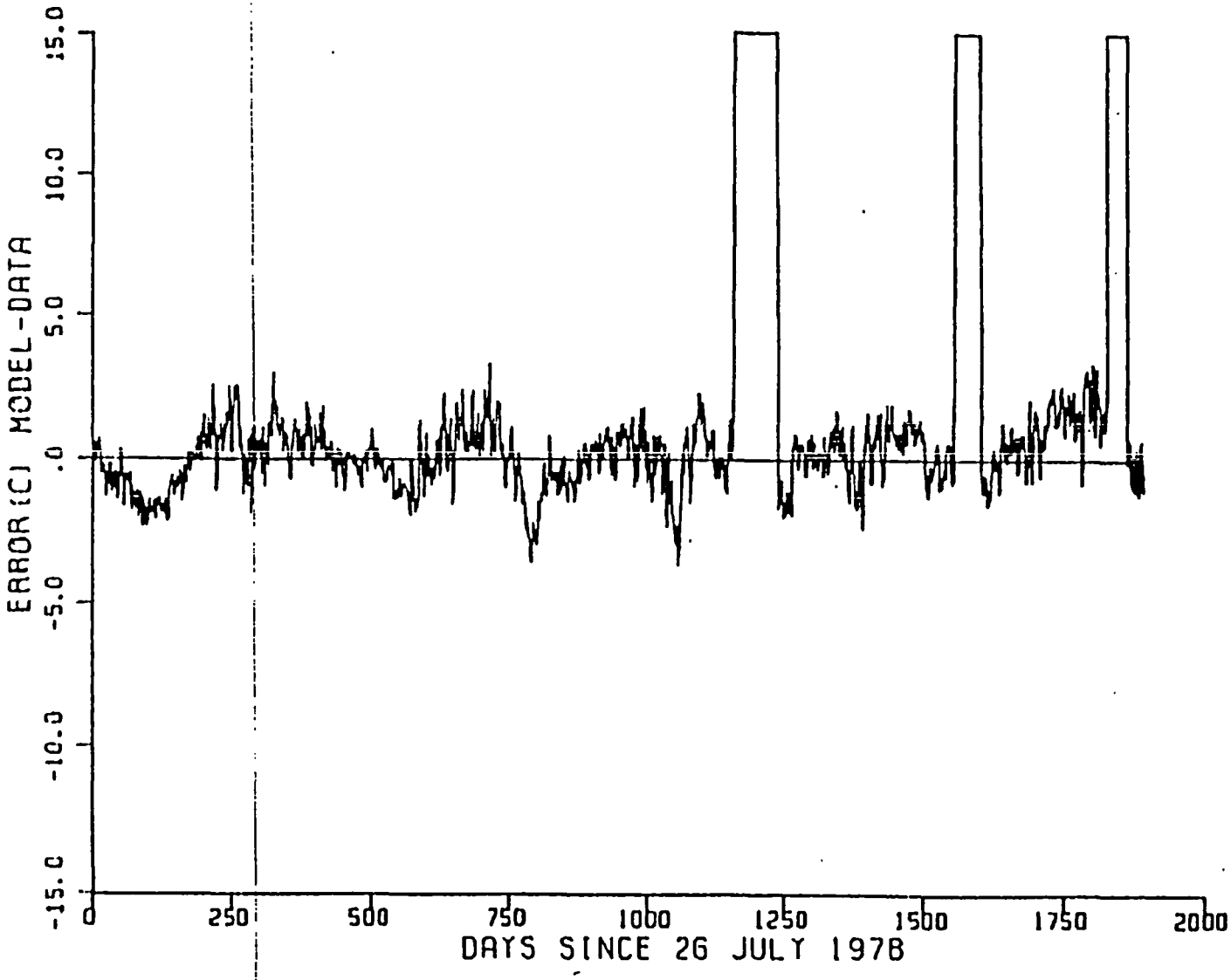


Figure 4.3c Raw Error at BR. PT.

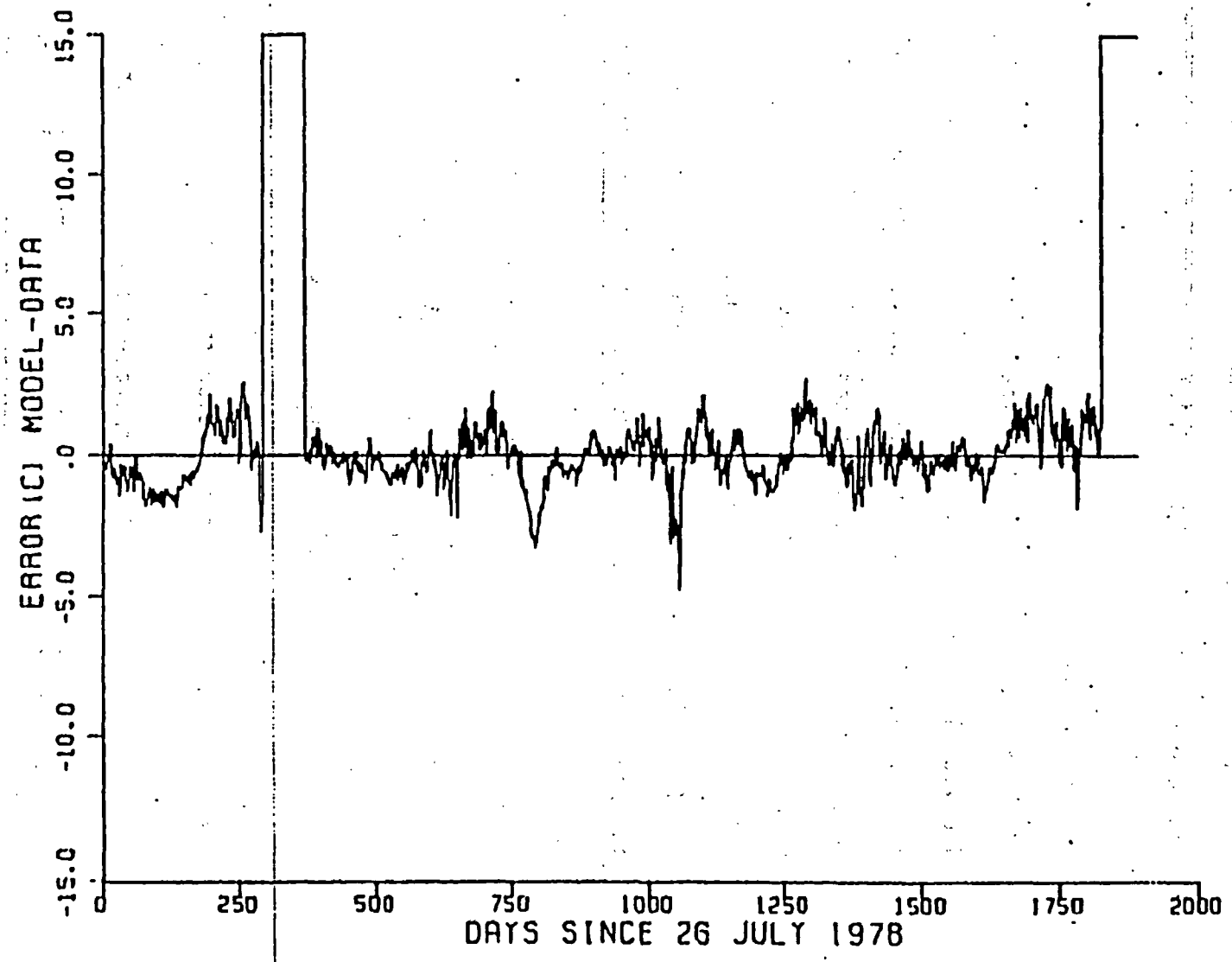


Figure 4.3d Raw Error at INTAKE

period with relatively large amplitude is observed at DIKE III, and the model is over-predicting at DISCHARGE most of the time. (Statistical data are provided in Section 4.4.)

4.3 Delta Error

Error analysis of the surface temperature can be done for individual points over the cooling lake system or for segments between the points. The former indicates how well the model is able to predict temperatures at specific locations, while the latter shows which segments of the cooling lake system are better modelled, and this information is useful for diagnosing individual model components. In the following analysis, the "delta" error over a segment is defined as the predicted temperature change over the segment minus the measured change. If the upstream control point is designated as i and the downstream point f, then the delta error for the segment is

$$\Delta T_{error} = (T_{i_{model}} - T_{f_{model}}) - (T_{i_{data}} - T_{f_{data}})$$

$$= (T_{i_{model}} - T_{i_{data}}) - (T_{f_{model}} - T_{f_{data}})$$

$$= \text{raw error at i} - \text{raw error at f} \qquad (4.1)$$

The following table is constructed to enhance clarity.

Table 4.1 Interpretation of Delta Errors

Raw Error at	-	Raw Error at	=	Delta Error across
INTAKE		DISCHARGE		Plant Condenser
DISCHARGE		DIKE III		WHTF
DIKE III		BR.PT.		Dike III Mixing
BR.PT.		INTAKE		Main Lake

Figures 4.4a to 4.4d show the "delta" errors associated with the above four segments of the cooling lake system. (Note that a positive error for all segments except across the plant condenser means that the model is cooling too much over that segment; however, a positive error for the segment across the plant condenser means that the model under-predicts the temperature rise due to waste heat rejected to the condenser cooling water.) The four "delta" error plots show that the WHTF and DIKE III Mixing are in general modelled with least accuracy, which can be inferred by the relatively poor performance of the model at DIKE III (see Figure 4.3b). It is also noticed that the model consistently over-predicts the temperature rise across the plant condenser.

4.4 Error Statistics

Statistics of the raw errors at the four diagnostic control points are given in Table 4.2. DIKE III shows the greatest variance but the least mean error, and the model is over-predicting by an average of 0.7 °C at DISCHARGE. Both BR.PT. and INTAKE show reasonably small mean raw errors and standard deviations.

Statistics for the first three years of simulation are reproduced from p. 165 of Wells et al. (1982) in Table 4.3. Comparison between the two tables shows that there is no major change in performance of the model associated with the two additional years of simulation.

In order to more effectively analyze the surface temperature prediction errors, monthly averages of the raw errors at the four diagnostic control points are compiled in addition to overall averages for the 5-year simulation period. This is done by lumping together the

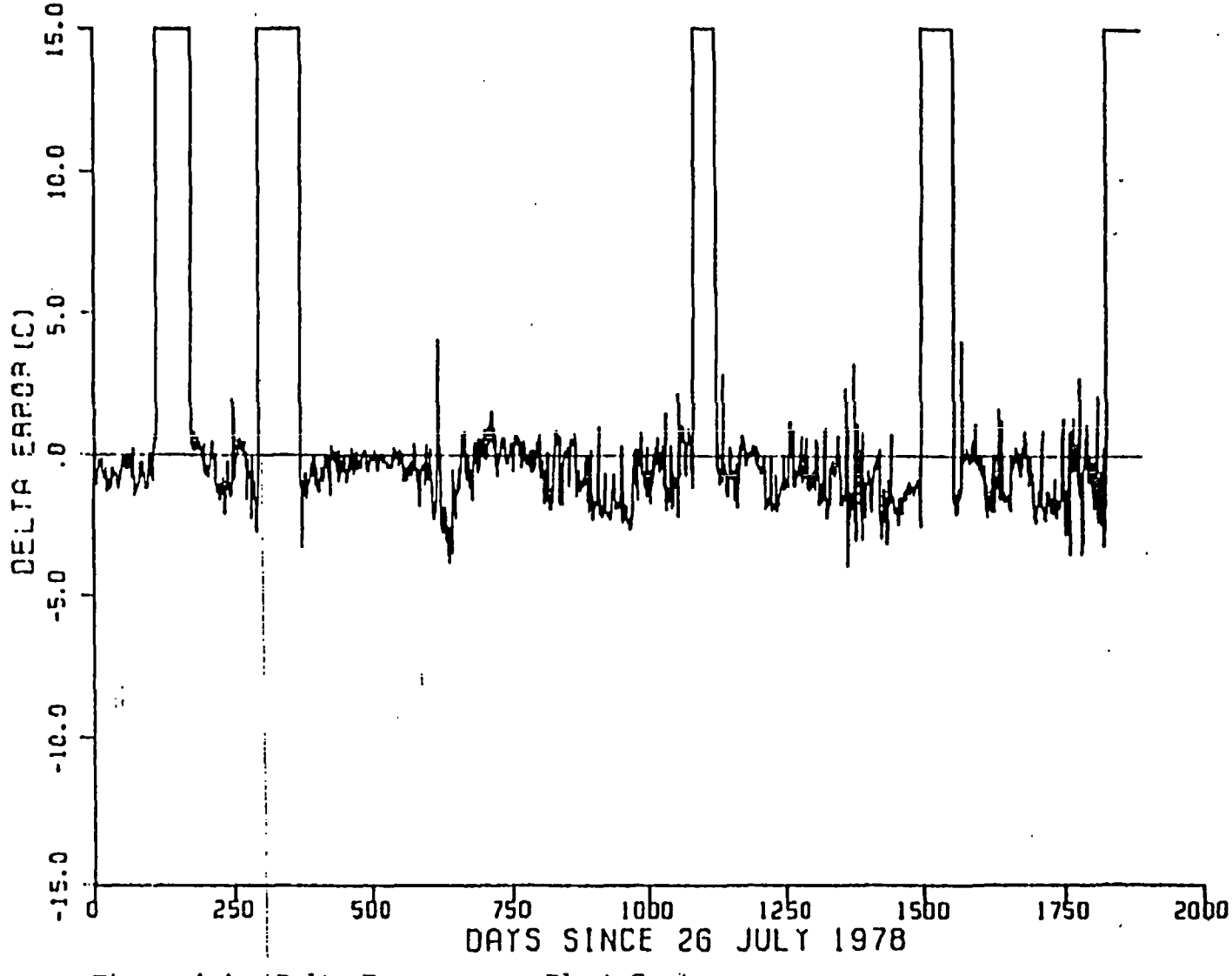


Figure 4.4a | Delta Error across Plant Condenser

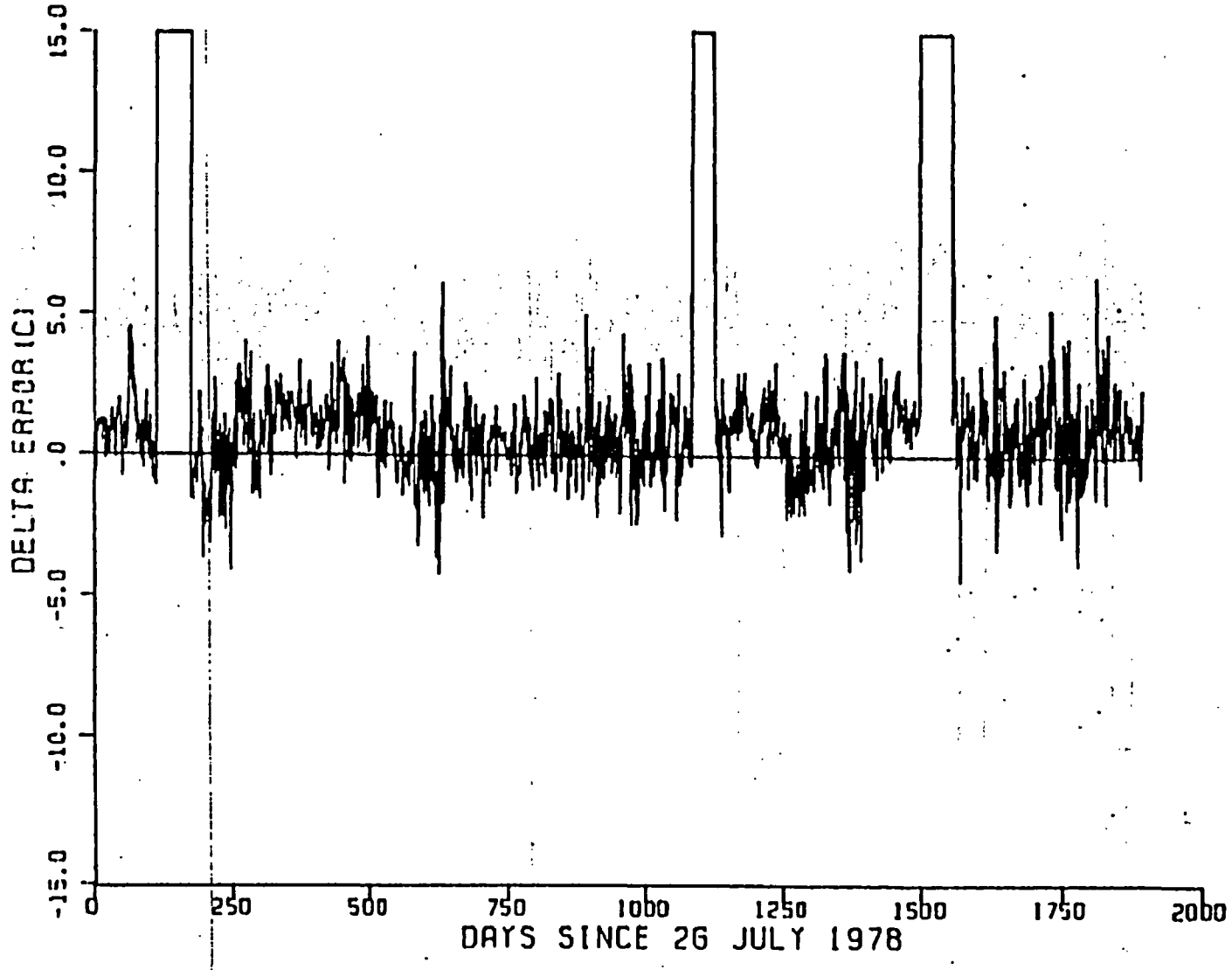


Figure 4.4b Delta Error across WMTF

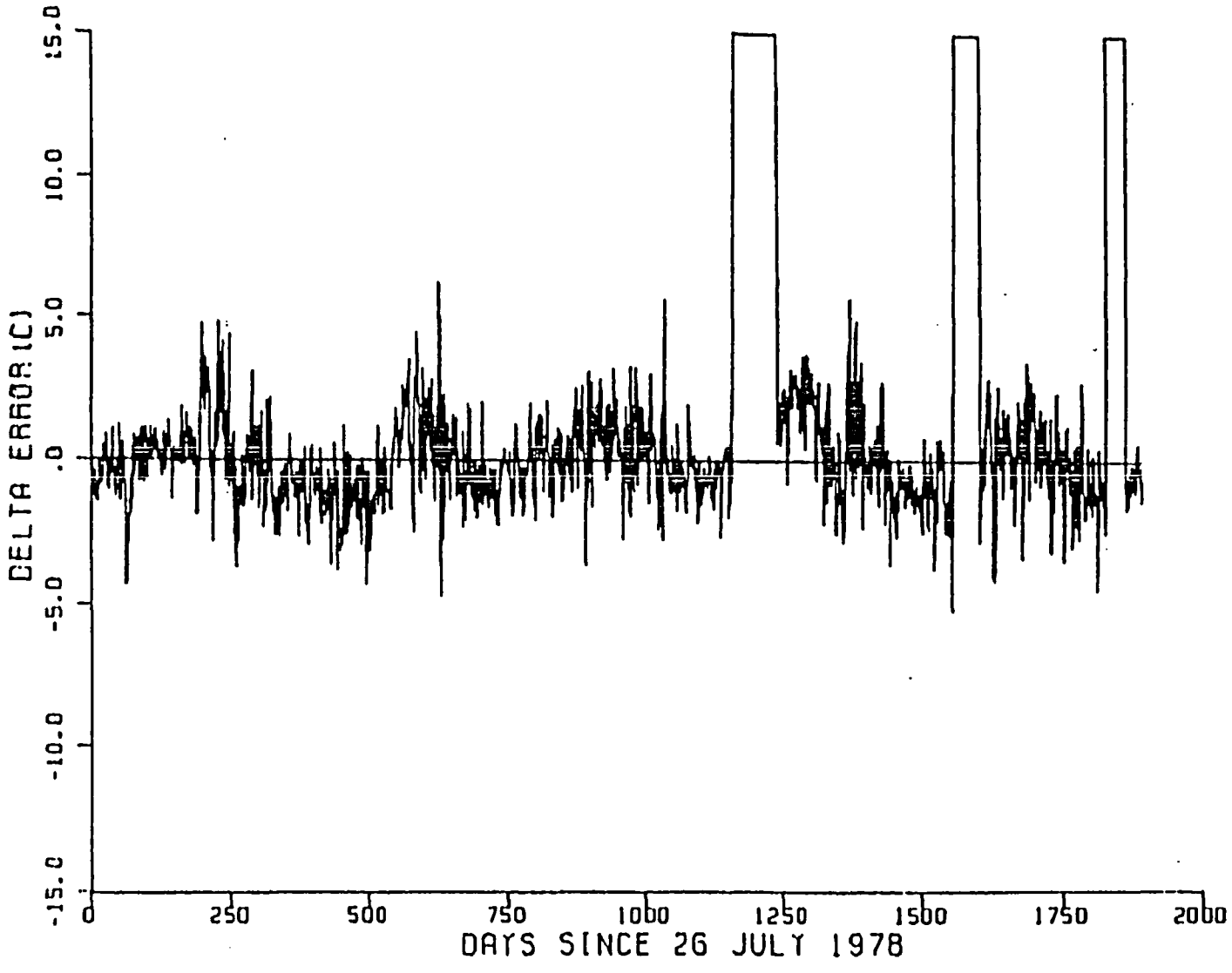


Figure 4.4c Delta Error across DIKE III Mixing

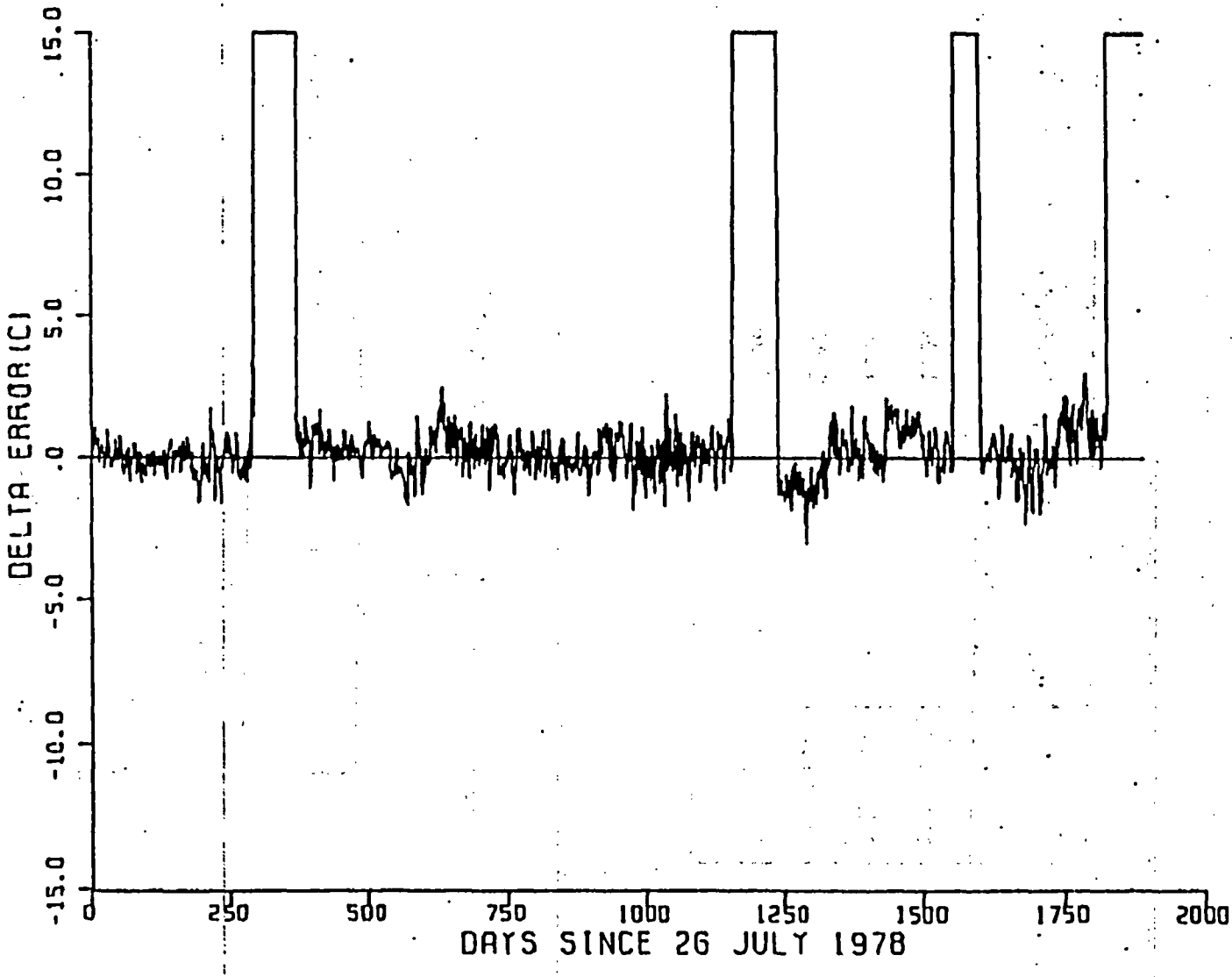


Figure 4.4d Delta Error across Main Lake

Table 4.2 Statistics of Raw Errors at Four Diagnostic Control Points (1978-1983)

	Mean (°C)	Standard Deviation (°C)	Sample Size (Day)
DISCHARGE	0.69	1.23	1633
DIKE III	-0.03	1.54	1893
BR. PT.	0.11	1.10	1733
INTAKE	-0.06	0.98	1752

Table 4.3 Statistics of Raw Errors at Four Diagnostic Control Points (1978-1981)

	Mean (°C)	Standard Deviation (°C)
DISCHARGE	0.48	1.27
DIKE III	-0.11	1.56
BR. PT.	-0.04	1.13
INTAKE	-0.22	1.02

monthly-averaged raw errors for a particular month for five years. The results are listed in Table 4.4. The number of months of data from which the average of a particular month is calculated is specified in the footnotes. The values in Table 4.4 are plotted in Figure 4.5 for each of the control points. Figure 4.5a shows that the raw error is predominantly positive (i.e., prediction > measurement) throughout the year at DISCHARGE. Figure 4.5b resembles a sine curve indicating that the raw error at DIKE III is largely periodic. It is also noted that the model over-predicts from January to July and under-predicts from August to December at that station. Figures 4.5c and d show that the model basically over-predicts from February to August and under-predicts from September to January at BR.PT. and INTAKE. Weak annual periodicity is observed and the monthly averages of the raw errors are bounded by ± 1°C at both stations.

4.5 Spectral Analysis

In many cases more insight is obtained by viewing results in the frequency domain; hence the method of spectral analysis is invoked in the surface temperature error analysis. The time series of raw errors at the four diagnostic control points are treated as stochastic processes. Stationarity of the stochastic processes is assumed, i.e., if $f(t) = \{x_{t_1}, x_{t_2}, \ldots x_{t_n}\}$ represents the raw error time series at a particular diagnostic control point, then we say that f(t) is stationary in the strict sense if the stochastic variables

$$x_{t_1}, x_{t_2}, x_{t_3}, \ldots x_{t_m}$$

Table 4.4 Monthly Averages* of Raw Errors at Four Diagnostic Control Points

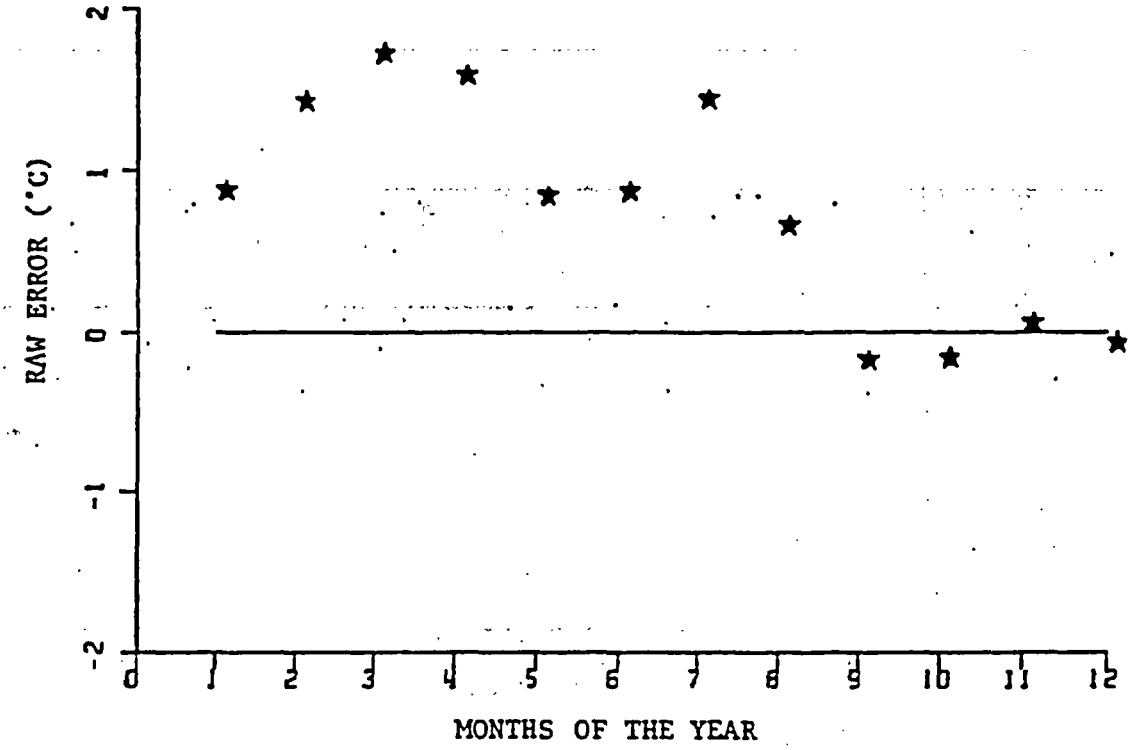
Month	DISCHARGE	DIKE III	BR. PT.	INTAKE
JAN	0.81[4]	0.52	-0.21	0.00
FEB	1.37	1.41	0.15	0.61
MAR	1.67	1.32	0.46	0.59
APR	1.53	0.64	0.63	0.44
MAY	0.78	0.46	0.43	-0.02[4]
JUN	0.80	0.11	0.56	-0.04[4]
JUL	1.38[4]	0.16	0.92[4]	0.52[3]
AUG	0.59[4]	-0.28[6]	0.38	0.02
SEP	-0.24	-1.20[6]	-0.48	-0.72
OCT	-0.22[4]	-1.43	-0.96[3]	-0.65
NOV	-0.01[4]	-1.01	-0.87[3]	-0.63
DEC	-0.13[4]	-0.65	-0.54[3]	-0.64

* all are five-year averages except as specified

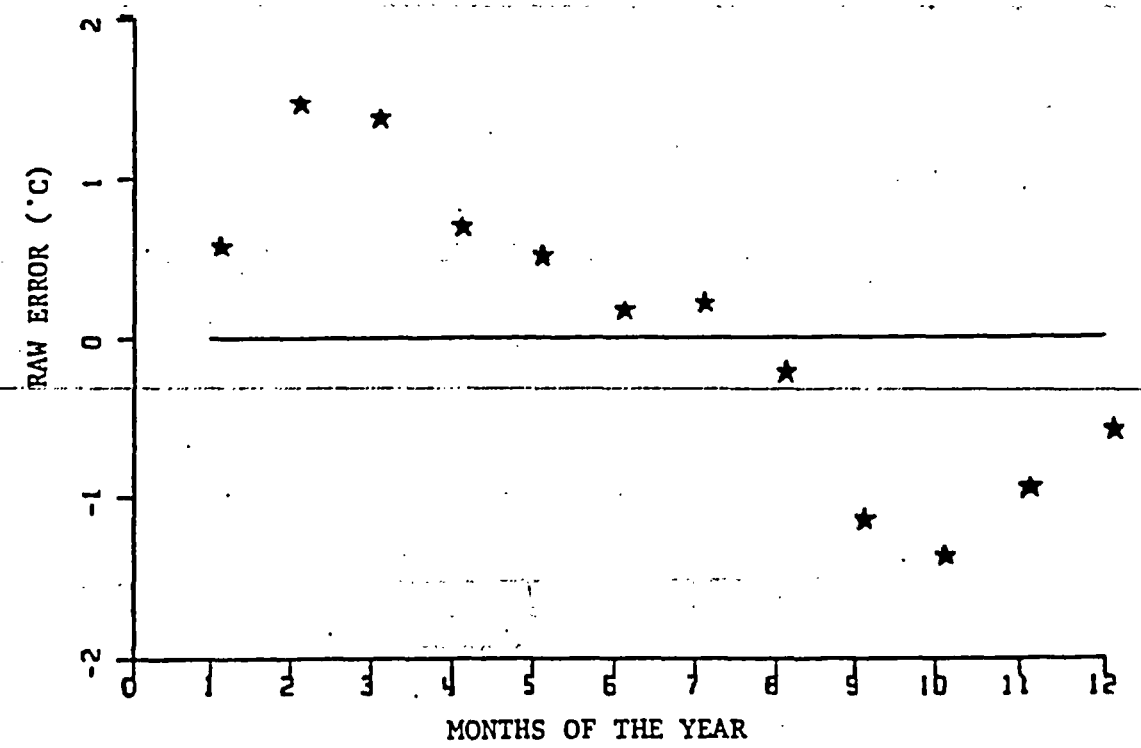
[6] six-year average

[4] four-year average

[3] three year average

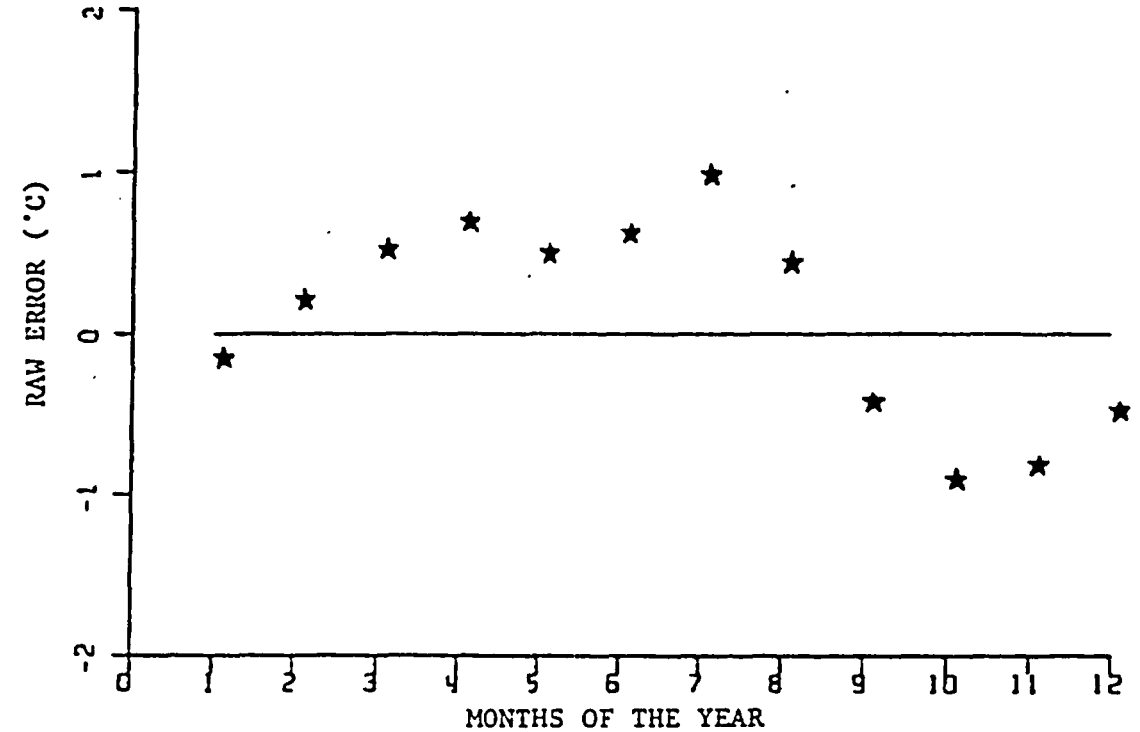


a. DISCHARGE

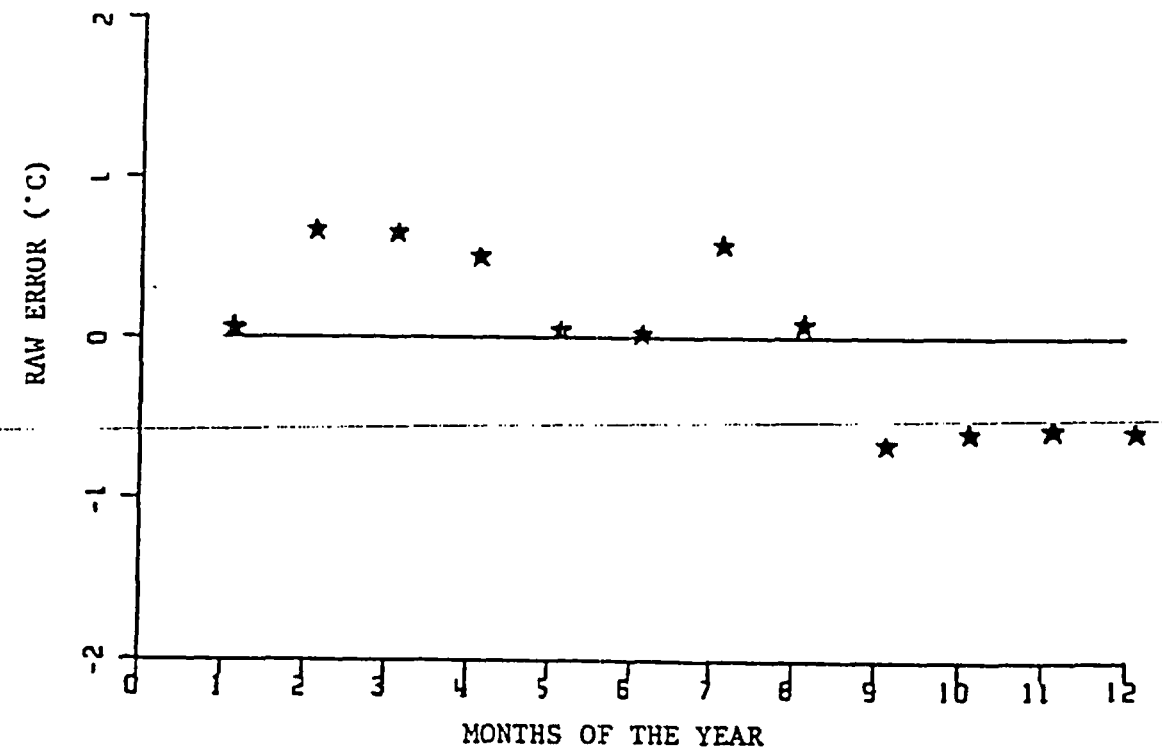


b. DIKE III

Figure 4.5 Monthly Averages of Raw Errors at Four Diagnostic Control Points



c. BURRUS POINT



d. INTAKE

Figure 4.5 Monthly Averages of Raw Errors at Four Diagnostic Control Points (cont'd)

have the same joint probability distribution as the variables

$$x_{t_1+\tau},\ x_{t_2+\tau},\ x_{t_3+\tau} \ldots x_{t_m+\tau}$$

for all m, t_1, t_2,...t_m and every τ (Grenander, 1957). (N.B., the analogous spatial property for stationarity is homogeneity.) And if we define the autocovariance of f(t) as

$$R_{ff} \equiv \mathrm{Cov}(f(t+\tau),\ f(t))$$

$$= E[f(t+\tau)f(t)] \qquad (4.2)$$

where Cov = covariance,

E = expected value,

τ = lag time,

the assumption of stationarity in f(t) will assure that R_{ff} is a function of τ and τ alone.

The Fourier transform of R_{ff} gives the corresponding power spectral density (or power spectrum), S(ω), such that

$$S(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega\tau} R_{ff}(\tau)\, d\tau \qquad (4.3)$$

$$R_{ff}(\tau) = \int_{-\infty}^{\infty} e^{i\omega\tau} S(\omega)\, d\omega \qquad (4.4)$$

Fourier transforms of the autocovariances of the raw errors at the four diagnostic control points are computed and Figure 4.6 presents

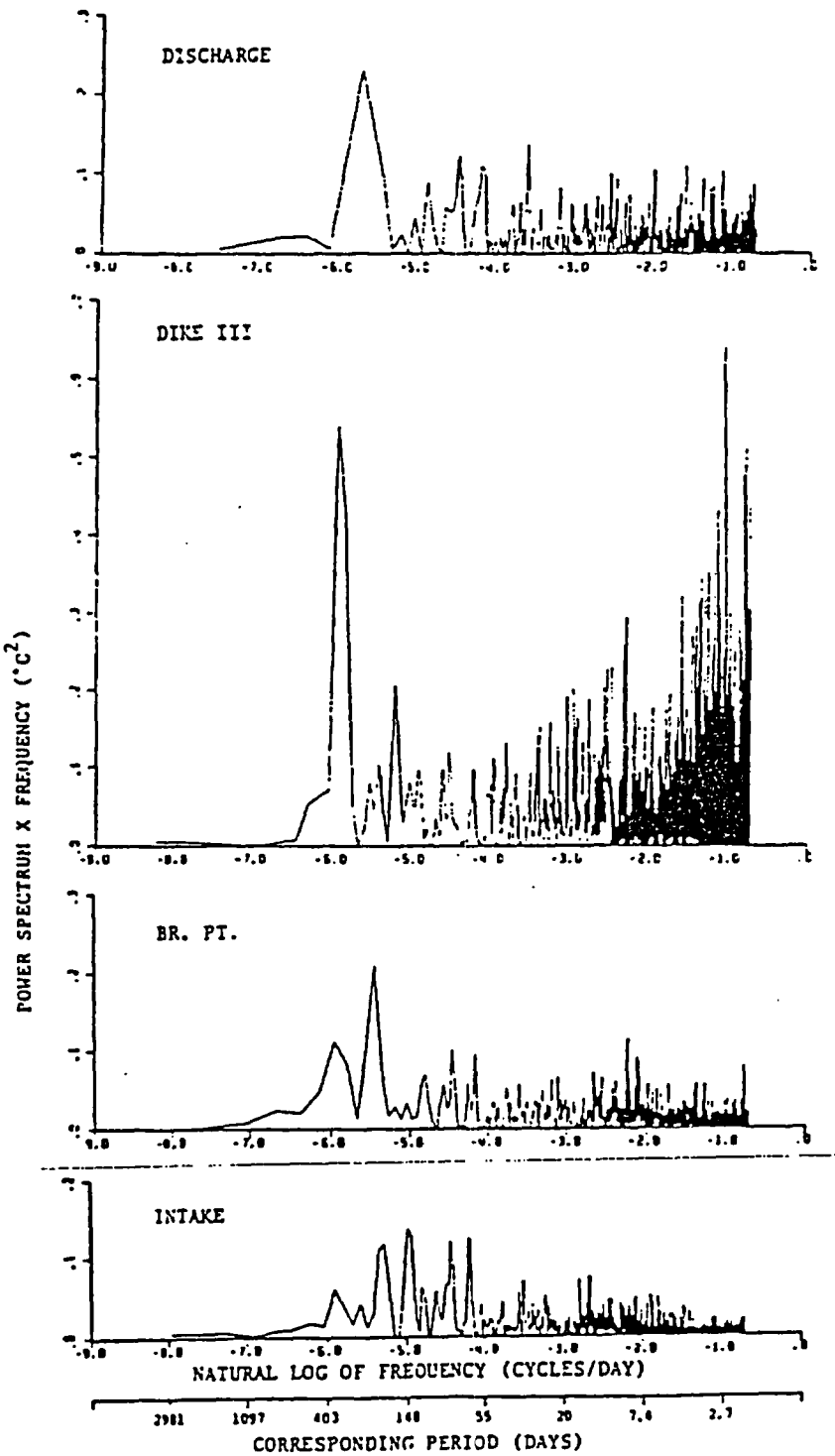


Figure 4.6 Power Spectra at Four Diagnostic Control Points

estimates of their corresponding power spectra. Note that the product between frequency and spectrum is plotted against natural log of frequency so that the "area under the curve" between two frequencies is proportional to the variance. This is because

$$\sigma^2(\Delta f) = \int_{f}^{f+\Delta f} S(\omega)\, d\omega$$

$$= \int_{f}^{f+\Delta f} \omega\, S(\omega)\, d \ln \omega \qquad (4.5)$$

where σ^2 = variance and Δf = window. In other words, variance of the raw error is represented by the total area under the curve.

Since the raw error time series are interrupted by gaps of missing measurement data, only the longest segment of continuous data is used for each control point. The lengths and starting dates of the time series used in computing the Fourier transforms are listed in Table 4.5.

Essentially a power spectrum shows how the variance or average power of the time series is distributed over frequency. Among the four control points, only DIKE III shows a strong and distinct peak at exactly the annual frequency (corresponds to ln (-5.9 cycles per day)). The rest of the diagnostic control points show a more evenly distributed variance over the frequencies ranging from about half a cycle per year to six cycles per year. This analysis confirms the observations made from Figures 4.3a through d that the raw error at DIKE III is periodic at predominantly the annual frequency whereas the

Table 4.5 Length and Starting Day of Time Series Used in Computing the Fourier Transforms

	Length of Time Series (Day)	Starting Day of Time Series (Days Since 26 July 1978)
DISCHARGE	911	174
DIKE III	1893	1
BR. PT.	1160	1
INTAKE	1458	371

rest of the control points show mild periodicities over the periods ranging from two months to two years.

4.6 Summary of Analysis

The State of Virginia experienced a hot summer in 1983. Meteorological records at Richmond showed that the monthly-averaged temperatures for June, July and August were about 0.8 °C above the long-term normals based on records for the 1941-1970 period (Local Climatological Data, Richmond, VA). North Anna Power Station operated at full capacity of two units for more than ninety days within the four-month period of June to September, providing the first experience with essentially two-unit summer operations. (Figures 4.7 and 4.8 show the number of nuclear units and number of pumps operating, respectively, during the period of model simulation.) Analysis in the previous sections indicates that the model shows a consistent performance even under "harsh" climatic and plant operation conditions.

Raw error analysis of the surface temperature shows that the model over-predicts in the months of January through July and under-predicts in August through December at DIKE III with an amplitude of about 1.5°C and a mean raw error near zero. For the rest of the diagnostic control points (INTAKE, BR.PT. and DISCHARGE), the raw error is occuring quite randomly in time with mild annual periodicity. The above observations are confirmed by error statistics and spectral analysis which show that the raw error at DIKE III is periodic at predominantly the annual frequency whereas the rest of the control points show mild periodicities over the periods ranging from two months to two years.

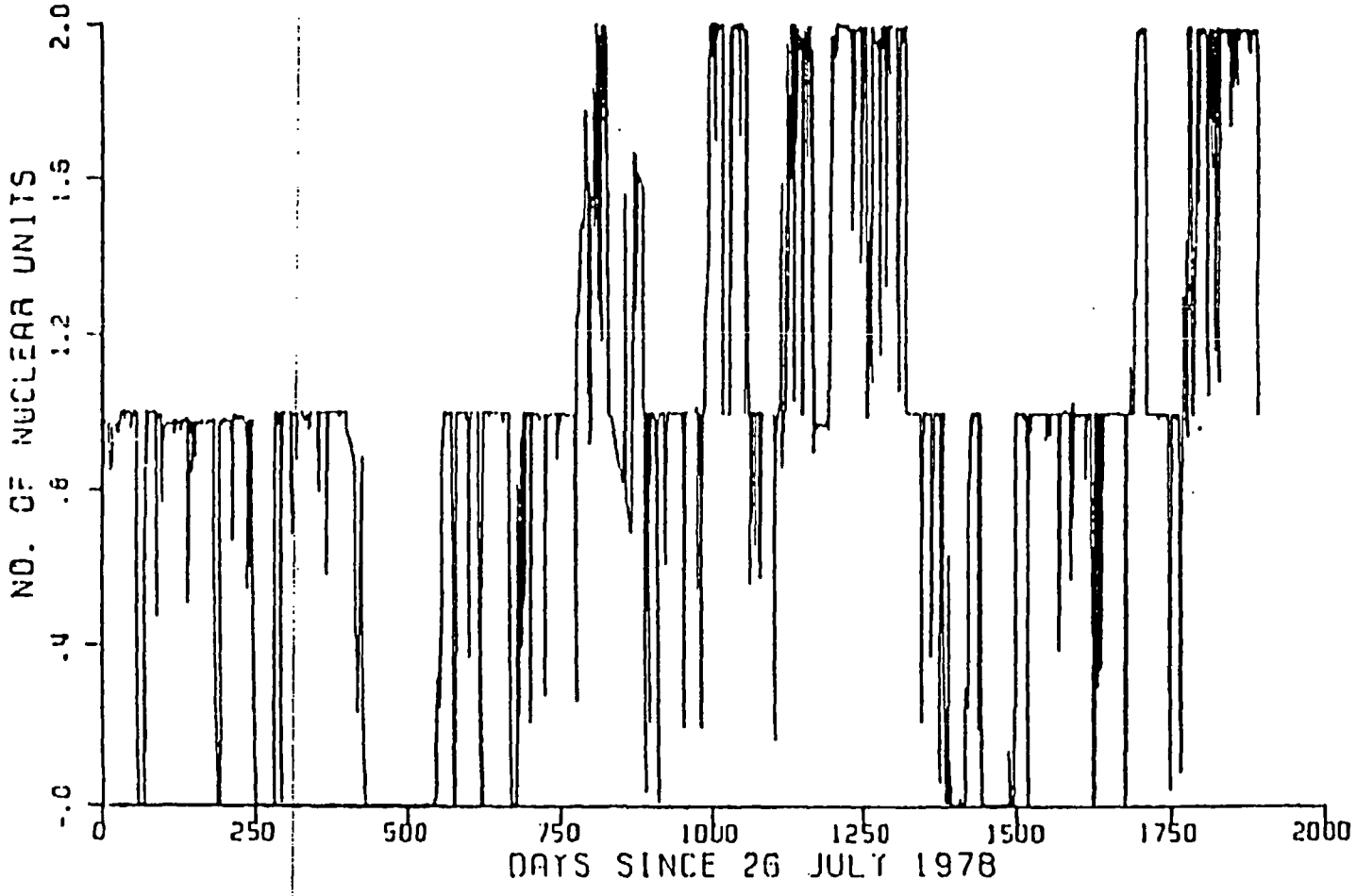


Figure 4.7 Number of Nuclear Units Operating During Simulation Period

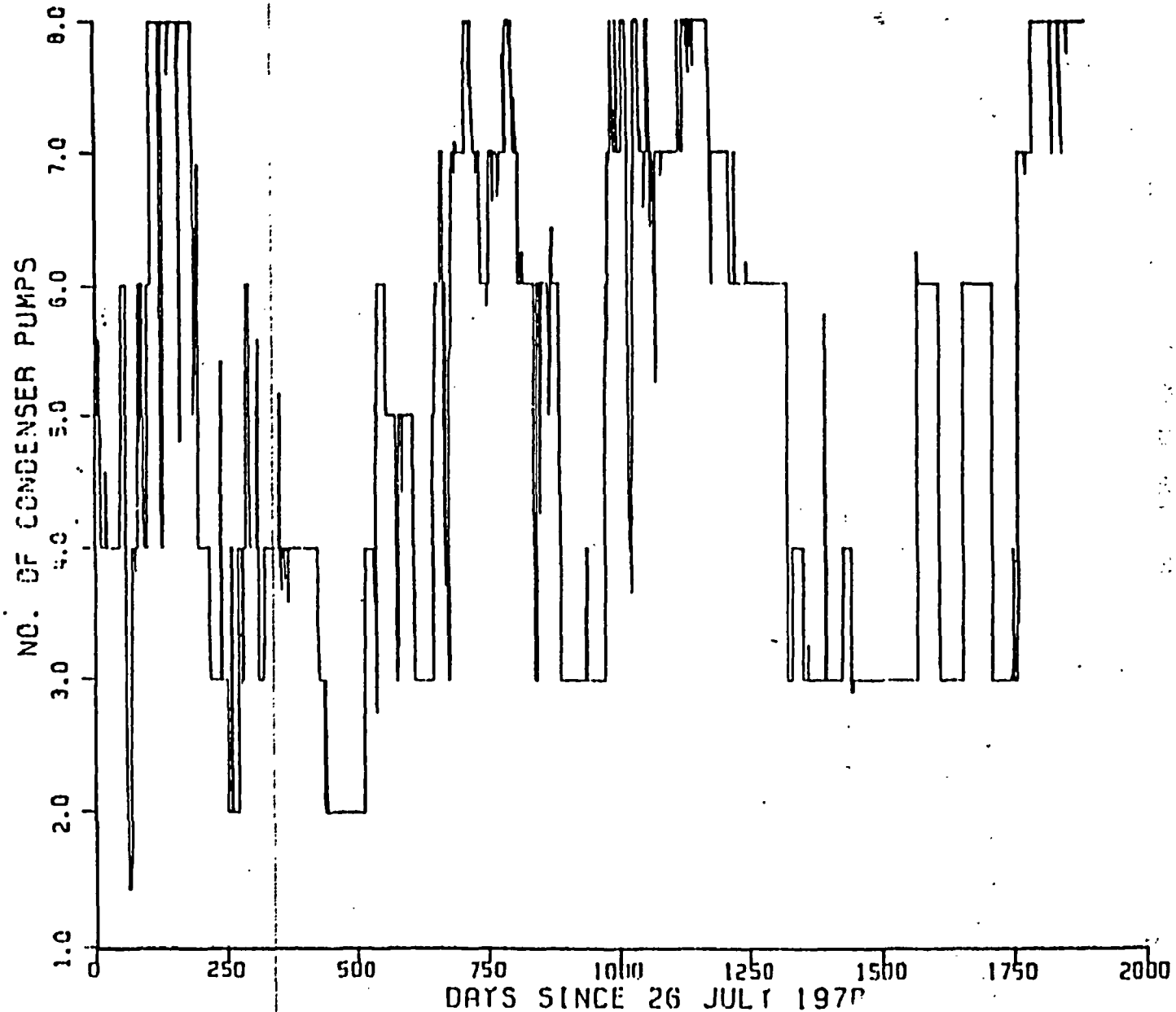


Figure 4.8 Number of Circulating Pumps Operating During Simulation Period
(Each pump corresponds to a flow rate of 15 m^3s^{-1})

"Delta" error analysis shows that the model consistently over-predicts the temperature rise across the plant condenser by about 0.7 °C.

5. MODEL RECALIBRATION

5.1 Introduction

Analysis of prediction errors in Chapter 4 shows that the model is relatively weak in predicting surface temperatures at two of the diagnostic control points, namely, DIKE III and DISCHARGE. Recalibration of the model aiming at improving the goodness-of-fit of the five years of measurement data at these two locations is presented in this chapter.

Transient error characterized by periodicity of the raw error at a distinct annual frequency is recognized at DIKE III. Section 5.2 deals with transient errors through a discussion of potential errors in (1) surface heat transfer and (2) system response of the cooling pond model.

Steady state error at DISCHARGE is characterized by an over-prediction of the surface temperature by an average of about 0.7°C. Given the fact that predictions at DISCHARGE should have minimal dependence on the hydraulics of the model or on the meteorological conditions, it is probable that the raw error is caused by uncertainty in the plant operation data. A sensitivity analysis of the raw error at DISCHARGE with respect to moderate changes in the plant operation data is given in Section 5.3.

5.2 Transient Errors at DIKE III

5.2.1 General Discussion

Periodicity of the raw error at DIKE III (see Figure 4.3b) can be looked at as the result of small differences in the highly periodic measurements and model predictions (see Figure 4.2b). Thus, if we let

$$E(t) = P(t) - M(t) \tag{5.1}$$

where E(t), P(t) and M(t) represent the raw error, prediction and measurement time series, respectivly, and if we fit a periodic function to P(t) and M(t) such that

$$P(t) = P^* \sin(\omega t + \phi_p) + \varepsilon_p(t) \tag{5.2}$$

$$M(t) = M^* \sin(\omega t + \phi_m) + \varepsilon_m(t) \tag{5.3}$$

where

P^*, M^* = amplitudes of periodic functions corresponding to P(t), M(t), respectively,

ϕ_p, ϕ_m = respective phases,

$\varepsilon_p, \varepsilon_m$ = remaining processes corresponding to P(t), M(t), respectively,

ω = $2\pi/T$; T = 365.25 days, and

t = Julian day of a year,

then, in general, the raw error can be caused by the fact that (1) $M^* \neq P^*$, or (2) $\phi_m \neq \phi_p$, or (3) some combinations of the above two. However, since the maximum raw errors do not occur during summer and winter (when both measurements and predictions are at their peaks), it is suspected that the raw error is caused more by (2) than by (1).

In order to confirm the above observation, harmonic analyses of the measurement and prediction time series are made.

Given any time series D(t), we can write

$$D(t) - C(t) = \varepsilon(t) \tag{5.4}$$

where C(t) is a periodic time series and $\varepsilon(t)$ is the remaining random process. If we let

$$C(t) = C^* \sin(\omega t + \phi) = A \sin \omega t + B \cos \omega t \tag{5.5}$$

where $\omega = 2\pi/T$ and T = period, then Eq. 5.4 becomes

$$D(t) = A \sin \omega t + B \cos \omega t + \varepsilon(t) \tag{5.6}$$

and, by direct integration,

$$\int_t^{t+nT} D(t) \sin\omega t \, dt = A \int_t^{t+nT} \sin^2 \omega t \, dt + B \int_t^{t+nT} \sin\omega t \cos\omega t \, dt + \int_t^{t+nT} \varepsilon(t) \sin\omega t \, dt = \frac{Ant}{2} \tag{5.7}$$

where n = 1, 2, 3,... Similarly,

$$\int_t^{t+nT} D(t) \cos\omega t \, dt = \frac{Bnt}{2} \tag{5.8}$$

Thus, we have the following Fourier coefficients

$$A(\omega) = \frac{2}{nT} \int_t^{t+nT} D(t) \sin\omega t \, dt \tag{5.9}$$

$$B(\omega) = \frac{2}{nT} \int_{t}^{t+nT} D(t) \cos\omega t \, dt \qquad (5.10)$$

And, from Eq. 5.5, we have the following relationships among A,B and H,ϕ:

$$C^* \cos\phi = A \qquad (5.11)$$

$$C^* \sin\phi = B \qquad (5.12)$$

or

$$C^{*2} = A^2 + B^2 \qquad (5.13)$$

$$\phi = \tan^{-1}\left(\frac{B}{A}\right) \qquad (5.14)$$

where C^* and ϕ can be interpreted as the amplitude and phase, respectively, characterizing the time series D(t).

In accordance with the above discussion, the appropriate time series are decomposed by computing the Fourier coefficients A and B using Eqs. 5.9 and 5.10, where $\omega = 2\pi/T$, T = 365.25 days and n = 4. A period of exactly four years (Jan. 1, 1979 to Dec. 31, 1982) of the time series is analyzed. Notice that for purposes of this analysis, a model prediction time series, P(t), is used with α set to a constant value of 0.75. (N.B., a periodic function was previously used to adjust the evaporation formula used in the WHTF in the calibrated model (p. 74, Wells et al.) such that

$$\alpha = 0.75 + 0.15 \cos\left(\frac{2\pi t}{365.25} - 2.60\right) \qquad (5.15a)$$

or, equivalently,

$$\alpha = 0.75 + 0.15 \sin\left(\frac{2\pi t}{365.25} - 1.03\right) \qquad (5.15b)$$

where t = Julian day of the year and α = correction factor multiplying the evaporative heat flux.)

Results of the harmonic analysis for P(t), M(t) and E(t) are listed in Table 5.1 in terms of C^* and ϕ, which are computed from A and B using Eqs. 5.13 and 5.14. From Table 5.1 it is observed that the amplitudes P* and M* are almost identical, while their phases are different by $\Delta\phi = \phi_p - \phi_m = 0.0939$. Thus it can be safely concluded that periodicity of raw error at DIKE III at annual frequency is caused by the fact that the measurements are lagging behind predictions by $\Delta\phi = 0.0939$ which amounts to $\Delta\phi/2\pi \times 365.25 \approx 5.5$ days along the time axis.

It is also observed from Table 5.1 that the raw error, E(t), is out-of-phase with the measurements and predictions by about $\pi/2$ (or 91 days), i.e., $\phi_e - (\phi_m + \phi_p)/2 \approx 1.58 \approx \pi/2$, which is consistent with the earlier qualitative observation that the greatest raw errors occured in spring and autumn and with the fact that the amplitudes P* and M* are almost equal. Regarding the latter point (refer to Fig. 5.1), Eq. 5.1 may be rewritten in terms of Eqs. 5.2 and 5.3 as

$$E(t) = P^* \sin(\omega t + \phi_p) - M^* \sin(\omega t + \phi_p) + \varepsilon(t) \qquad (5.16)$$

where $\varepsilon(t) = \varepsilon_p(t) - \varepsilon_m(t)$. Upon trigonometric expansion, Eq. 5.16 becomes

$$E(t) = \sin \omega t[P^*\cos\phi_p - M^* \cos\phi_m] + \cos \omega t[P^*\sin\phi_p - M^* \sin\phi_m] + \varepsilon(t) \qquad (5.17)$$

Table 5.1 Fourier Decomposition of Time Series

Description of Time Series D(t)	C^* (°C)	ϕ (radian)	ϕ (Julian Day)
Measurements at DIKE III [= M(t)]	11.687 (=M*)	-2.0793 ($=\phi_m$)	-120.9
Predictions at DIKE III with α = 0.75 [= P(t)]	11.683 (=P*)	-1.9854 ($=\phi_p$)	-115.4
Raw Errors at DIKE III with α = 0.75 [= E(t) = P(t) - M(t)]	1.096 (=E*)	-0.4576 ($=\phi_e$)	-26.6

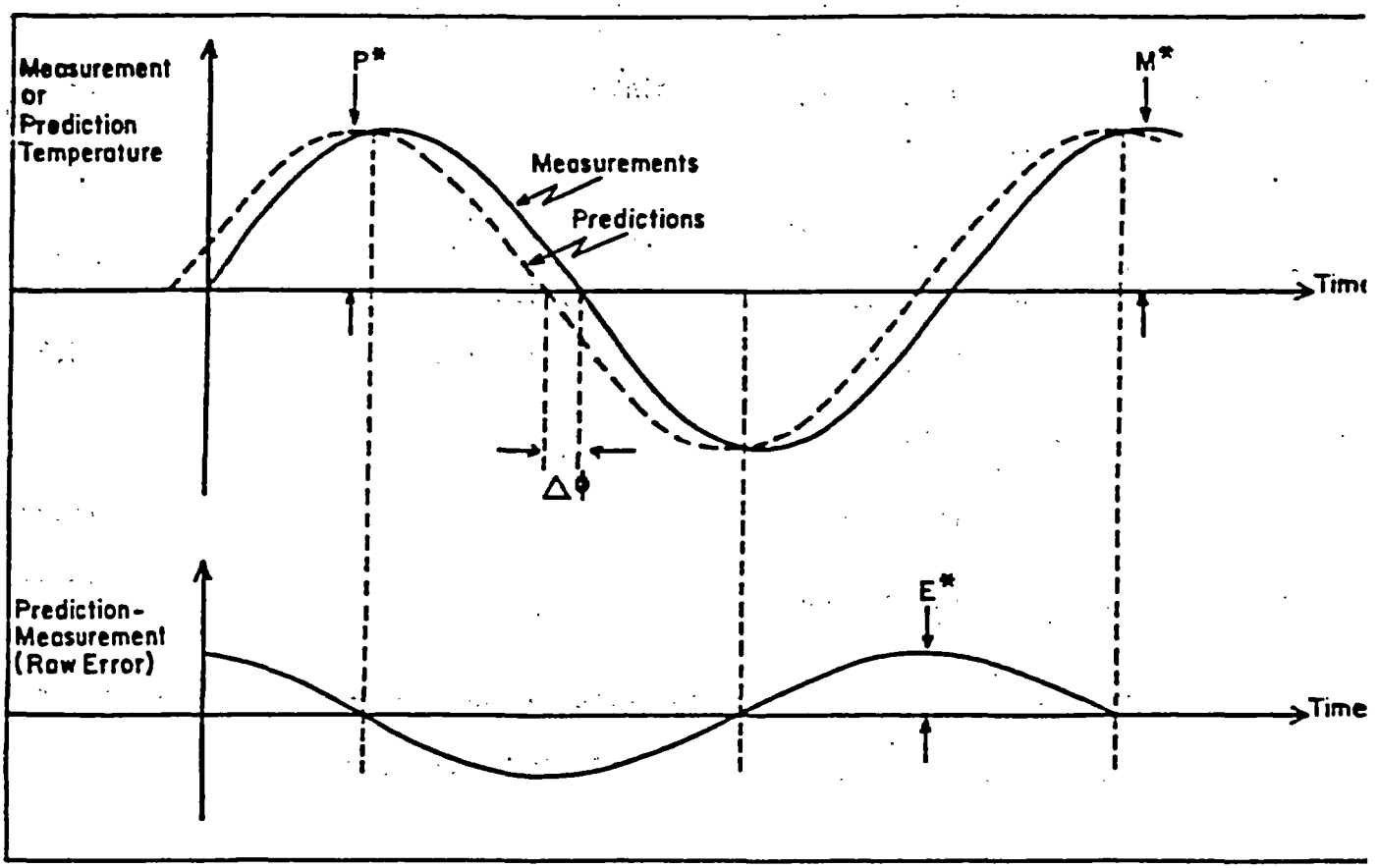


Figure 5.1 Schematic of Relationships between Measurements, Predictions and Raw Errors at DIKE III

If we let

$$A = P^* \cos\phi_p - M^* \cos\phi_m \qquad \text{and}$$

$$B = P^* \sin\phi_p - M^* \sin\phi_m \tag{5.18}$$

then Eq. 5.17 becomes

$$E(t) = A \sin\omega t + B \cos\omega t + \varepsilon(t) \tag{5.19}$$

which has the identical form as Eq. 5.6. Letting

$$E^* \sin(\omega t + \phi_e) = A \sin\omega t + B \cos\omega t \tag{5.20}$$

it follows that

$$(E^*)^2 = A^2 + B^2 \tag{5.21}$$

and

$$\phi_e = \tan^{-1}\left(\frac{B}{A}\right) \tag{5.22}$$

Now, since the harmonic analysis shows that $M^* = P^*$, putting Eq. 5.18 into Eqs. 5.21 and 5.22 will give, upon simplification,

$$(E^*)^2 = (M^*)^2[(\cos\phi_p - \cos\phi_m)^2 + (\sin\phi_p - \sin\phi_m)^2] \tag{5.23}$$

and

$$\phi_e = \tan^{-1}\left(\frac{\sin\phi_p - \sin\phi_m}{\cos\phi_p - \cos\phi_m}\right) \tag{5.24}$$

Defining $\phi_p = \phi_m + \Delta\phi$, we can rewrite Eq. 5.23 as

$$(E^*)^2 = (M^*)^2[(\cos\phi_m(\cos\Delta\phi - 1) - \sin\phi_m \sin\Delta\phi)^2 + (\sin\phi_m(\cos\Delta\phi - 1) + \cos\phi_m \sin\Delta\phi)^2] \tag{5.25}$$

which can be further simplified to give

$$(E^*)^2 = (M^*)^2[(\sin\phi_m \sin\Delta\phi)^2 + (\cos\phi_m \sin\Delta\phi)^2] \tag{5.26}$$

or

$$(E^*)^2 = (M^*)^2 \Delta\phi^2 \tag{5.27}$$

by recognizing that $\Delta\phi << 1$. (From Table 5.1, $\Delta\phi = \phi_p - \phi_m = 0.0939$.) Thus, we arrive at the following relationship

$$E^* = M^* \Delta\phi \tag{5.28}$$

which simply says that the amplitude of raw errors is directly proportional to the phase error given that the amplitudes of prediction and measurement are equal. (This result is confirmed by the calculation: $11.687 \times 0.0939 = 1.097 \approx 1.096$ where 11.687 and 1.096 are computed values of M^* and E^*, respectively. See Table 5.1.)

Eq. 5.24 can also be simplified to give

$$\phi_e = \tan^{-1} (-\cot \phi_m) \tag{5.29}$$

by writing $\phi_p = \phi_m + \Delta\phi$ and realizing that $\Delta\phi << 1$; this result is equivalent to

$$\phi_e = \phi_m \pm \frac{\pi}{2} \tag{5.30}$$

which says that, as long as $\phi_p - \phi_m = \Delta\phi << 1$, and $M^* = P^*$, ϕ_e will always be 90° out of phase with the measurement or prediction.

Two reasons can be postulated for the time lag between measurements and predictions: (1) the forcing function, as represented by the equilibrium temperature (T_e) is too high (thus causing maximum over-prediction) in spring and too low (causing maximum under-prediction) in autumn (Figure 5.2), or (2) the system response of the model is too quick. For each of these two general reasons, two possibilities are examined.

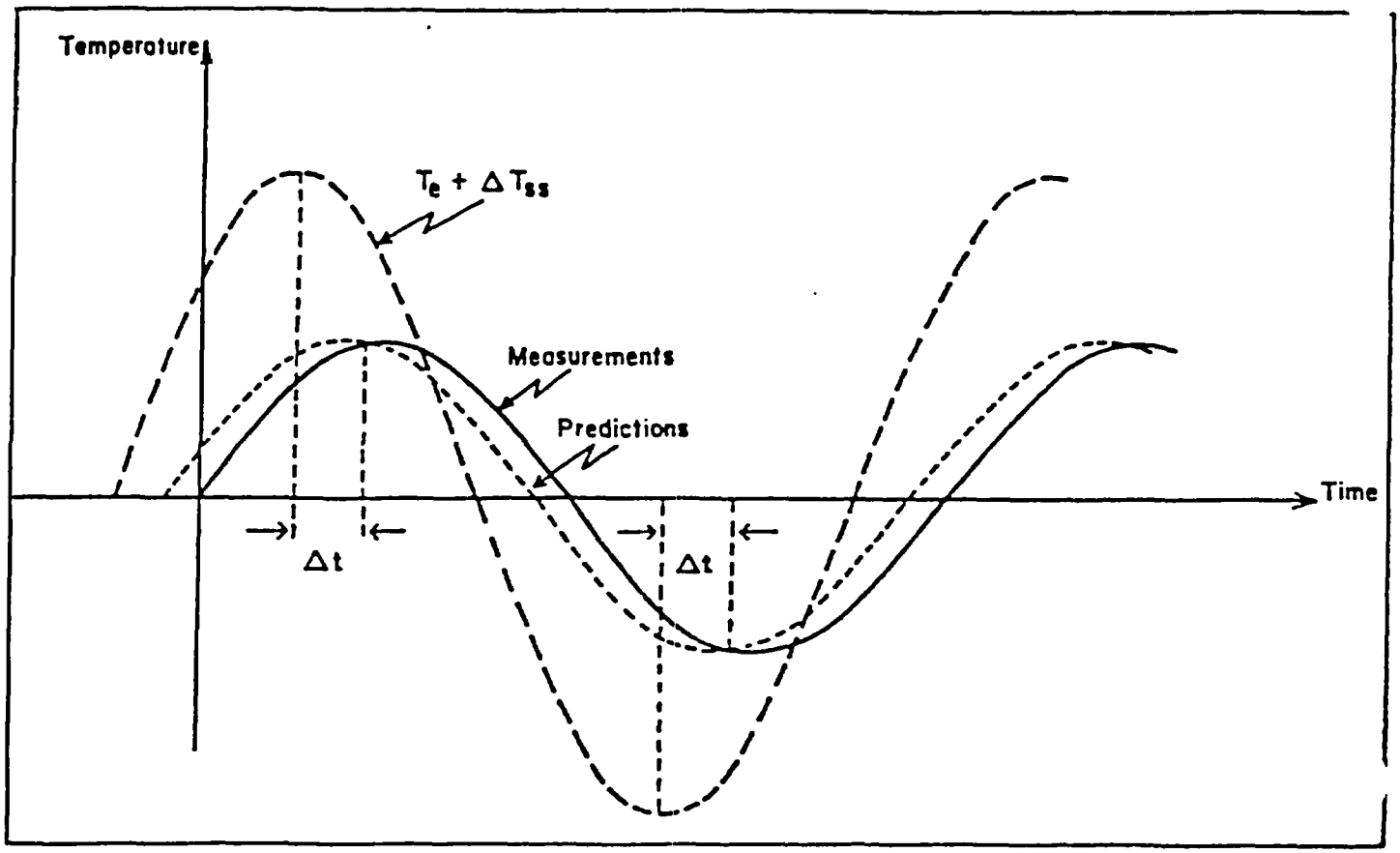


Figure 5.2 Schematic of Relationship between Raw Errors and Equilibrium Temperature at DIKE III

Regarding the meteorological forcing function, two candidates are examined: (a) long-wave radiation (discussed in Section 5.2.2.1) and (b) evaporative heat flux (discussed in Section 5.2.2.2). These two are chosen because they have the greatest uncertainties among the various surface heat transfer components.

Regarding the system response, again two candidates are examined: (a) spatially averaged response of the model to periodic warming and (b) longitudinal response based on the residence time in the WHTF. A brief discussion of each of the two candidates is given in Sections 5.2.3.1 and 5.2.3.2, respectively. Conclusion of this transient error analysis is given in Section 5.2.4.

5.2.2 Periodic Errors in Surface Heat Transfer

5.2.2.1 Long-Wave Radiation

The long-wave (atmopsheric) radiation formula proposed by Idso and Jackson (1969) replaced the formula due to Swinbank (1963) in the calibrated model as described by Wells et al. (1982) in order to improve the model's winter time performance. A comparison of several long-wave radiation calculation methods (see Table 5.2) developed over the past few decades is presented below using the air temperature and relative humidity data collected at North Anna Power Station. (N.B. Some of the formulae require water vapor pressure which can be readily obtained from air temperature and relative humidity by using the conversion formula rendered by Thackston (1974):

$$e_{sat} = 25.4 \times \exp(17.62 - \frac{9500.8}{T + 460})$$

Table 5.2 Summary of Long-Wave Radiation Calculation Methods (after Hatfield, 1983)

Method	Formulation for ε_{ac} *	Reference
Brunt	$0.51 + 0.066\ e_o^{1/2}$	Brunt (1932)
Swinbank	$0.92 \times 10^{-5}\ T_o^2$	Swinbank (1963)
Idso-Jackson	$1 - 0.261\ \exp[-7.77\times10^{-4}(273-T_o)^2]$	Idso and Jackson (1969)
Brutsaert 1	$0.553\ e_o^{1/7}$	Brutsaert (1975)
Satterlund	$1.08[1 - \exp(-e_o^{To/2016})]$	Satterlund (1979)
Brutsaert 2	$0.575\ e_o^{1/7}$	Idso (1981)
Idso 1	$0.179\ e_o^{1/7}\ \exp(350/T_o)$	Idso (1981)
Idso 2	$0.7 + 5.95\times10^{-5} e_o \exp(1500/T_o)$	Idso (1981)

* ε_{ac} = atmospheric emissivity under clear sky
e_o = water vapor pressure in millibars
T_o = absolute air temperature in degrees Kelvin

where e_{sat} = saturation vapor pressure (mm Hg) and T = air temperature in degrees Fahrenheit.)

Since no real-time on-site measurements of incoming atmospheric radiation are available, little can be said regarding the performance of the various formulae. The goal of this study, however, is to compare formulae and to identify if the raw error might be partially offset by use of a different formula.

In this comparision, only the downward component of atmospheric radiation under clear sky is calculated such that

$$\phi_{ac} = \varepsilon_{ac} \sigma T_a^4 \qquad (5.31)$$

where ϕ_{ac} = downward atmospheric radiation under clear sky (watt per m^2)

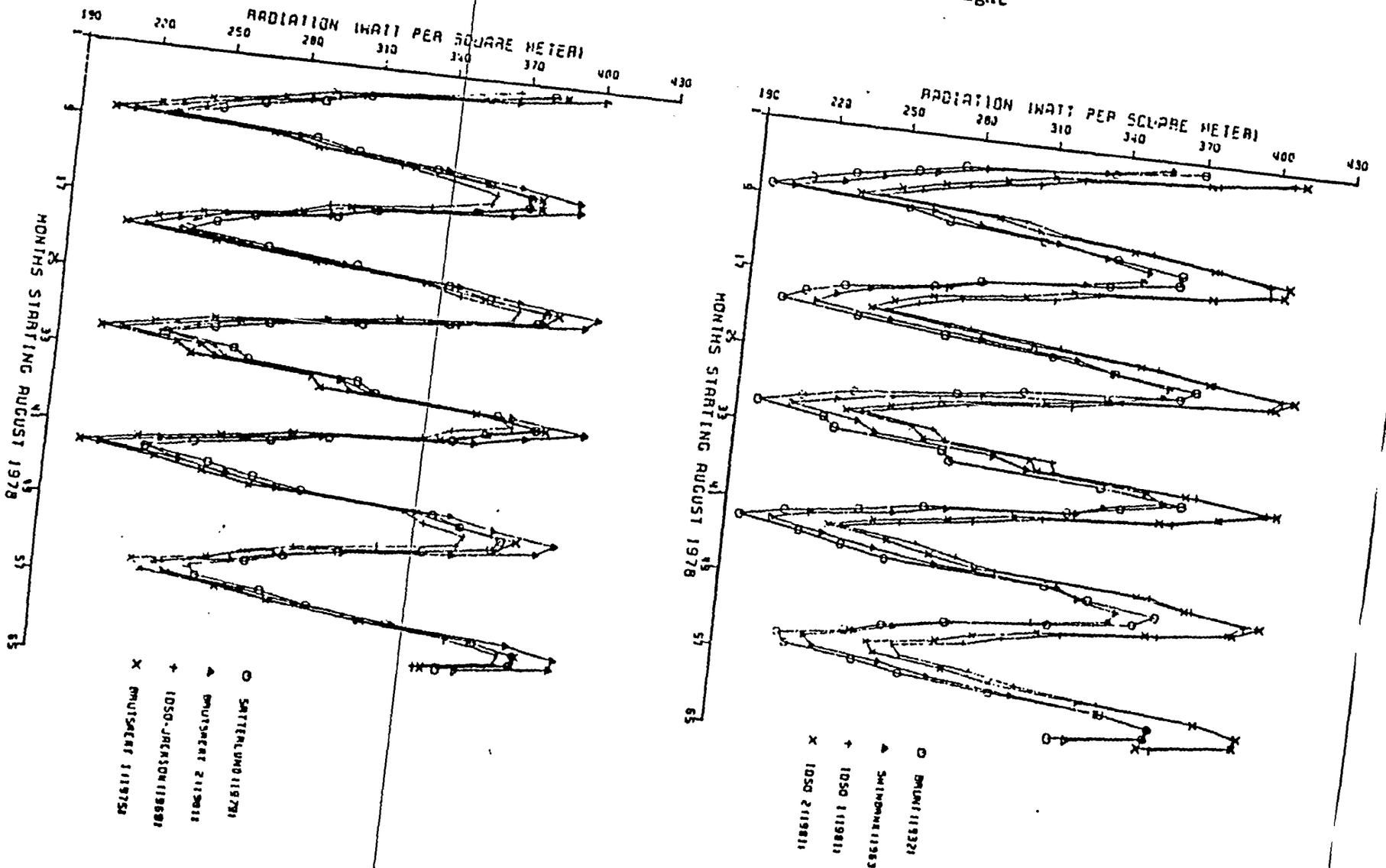
ε_{ac} = atmospheric emissivity under clear sky, as given in Table 5.2,

σ = Stefan-Boltzmann constant
= 5.6697×10^{-8} W m^{-2} K^{-4}, and

T_a = absolute air temperature in degrees Kelvin.

Figure 5.3 shows the monthly-averaged values of downward long-wave radiation under clear sky for more than five years since August, 1978 as calculated by using the eight different proposed formulae. (A detailed tabulation of the values and statistics is attached as an appendix.) It is found that the ones by Idso (Idso 1 and Idso 2) give the highest all-season estimates of atmospheric radiation whereas the ones due to Brunt and Swinbank give the lowest. The remaining four formulae give results that are in general bounded by the above-mentioned four. Of these, Idso-Jackson gives the most modest range (i.e., low in the summer and high in the winter), followed by Satterland, which is

Figure 5.3 Monthly-Averaged Values of Downward Long-Wave Radiation Under Clear Sky Given by Eight Different Formulae



consistently higher than Idso-Jackson with a more pronounced difference during the summer. Brutsaert 2 is identical with Brutsaert 1 except the former is shifted upward by about 29 W m^{-2}. Both have a larger range than those given by Satterland and Idso-Jackson.

A closer examination of Figure 5.3 shows that there is not a single formula which gives higher values of long-wave radiation (and thus higher water temperature) than Idso-Jackson does during the months from August through December (when the model under-predicts at DIKE III) and at the same time gives lower values of long-wave radiation (and thus lower water temperature) during the months from January through July (when the model over-predicts at DIKE III). The closest candidate is Brutsaert 1 which gives higher estimates than Idso-Jackson does from May to October and lower estimates from November to April. Figure 5.4 shows daily estimates of long-wave radiation given by the two formulae in the year of 1979. However, difference between the two estimates for each day is very small in relation to the gross atmospheric radiation (usually less than 10%) such that improvement in model prediction is insignificant when the effect is translated to changes in surface water temperatures. In addition to small difference between the formulae, the fact that an error in atmospheric radiation would affect surface temperature throughout the whole cooling lake system is not consistent with the observation of a periodic error primarily at DIKE III. Thus, no recalibration of the model is made in this respect.

It is known that long-wave radiation increases when water vapor condenses into droplets in the atmosphere, and that a cloud base near the ground radiates virtually like a black body (emissivity approaches

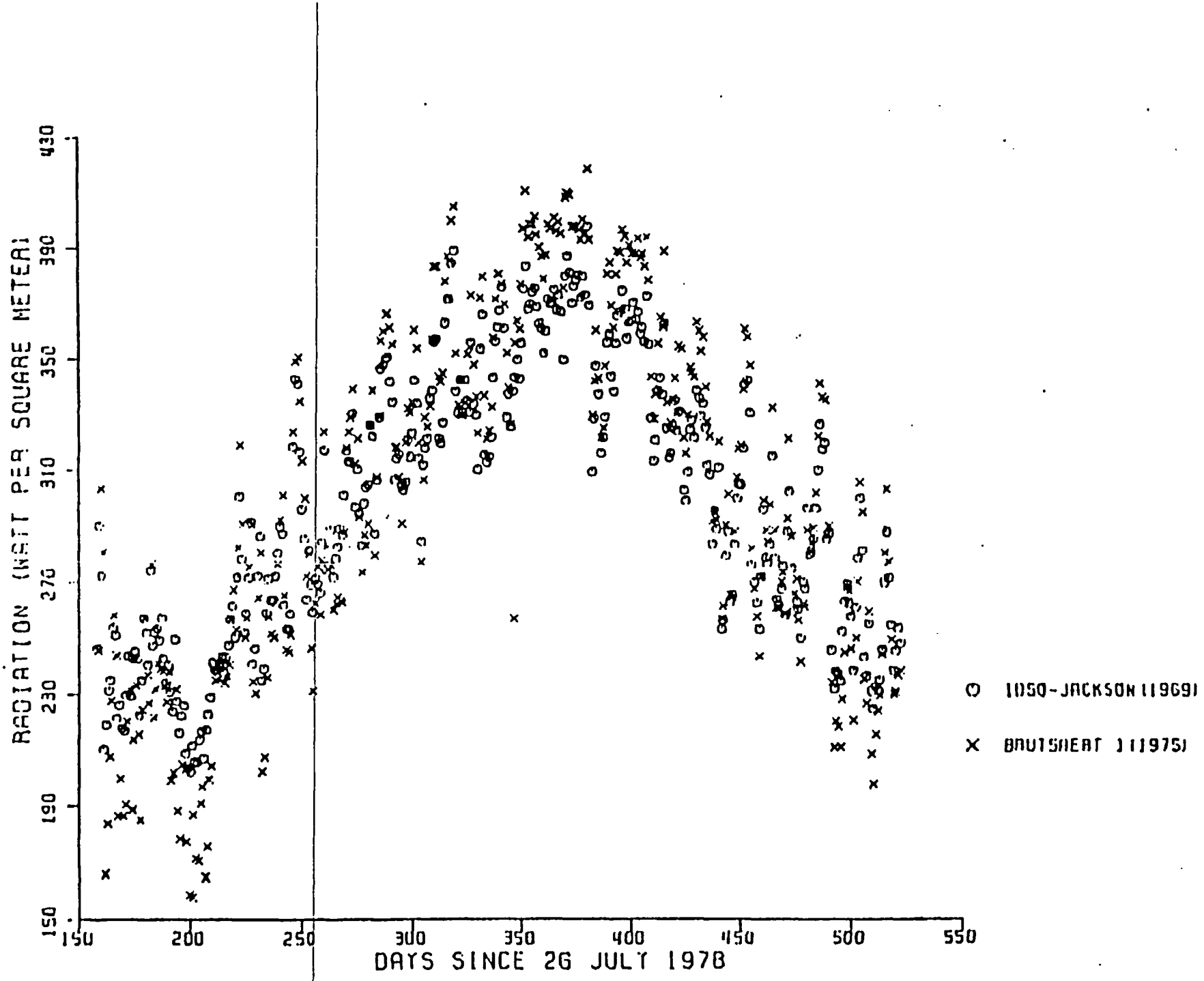


Figure 5.4 Daily Estimates of Long-Wave Radiation Given by Brutsaert 1 and Idso-Jackson

1.0). Kimball et al. (1982) concluded that the maximum increase in radiation over clear sky radiation could be about 40% with typical values of about 20%.

Several empirical approaches have been suggested for cloud cover correction. The one adopted in North Anna Cooling Pond Model uses the following relation:

$$\phi_a = \phi_{ac}(1 + k_1 C^2) \qquad (5.32)$$

where

ϕ_a = downward atmospheric radiation for cloudy sky

ϕ_{ac} = clear sky downward atmopsheric radiation

C = portion of sky covered by cloud (value varies from 0 to 1)

k_1 = constant that depends on the nature and height of clouds.

Value of k_1 ranges from 0.04 for cirrus to 0.25 for nimbostratus or fog according to Bolz (1949). (See also Table 6.7 of Brutsaert, 1982). A constant value of $k_1 = 0.17$ is used in the model, as suggested by Wunderlich (1972), since no records of cloud types are available.

Another approach applies the correction to the effective outgoing radiation under clear sky, ϕ_o, according to the equation (Sellers, 1965):

$$\phi = \phi_o(1 - k_2 C^m) \qquad (5.33)$$

where ϕ = effective outgoing radiation corrected for cloud cover, and k_2, m are constants to be determined from observations. In this approach, cloud effect is applied to both upward surface radiation and

downward sky radiation. However, the physical justification for applying a correction factor to the upward surface radiation is unclear and therefore the first approach is preferred to the second one (Salhotra et al., 1983).

5.2.2.2 Evaporative Heat Transfer

The fact that there is much stronger periodicity of the raw errors at DIKE III than at any other diagnostic control points suggest that the periodic error may be more associated with forced rather than with natural heat flux terms. This is suspected because the effects of forced heat fluxes decrease in the direction of flow from the plant condenser towards the plant intake, whereas effects of natural heat fluxes are relatively constant throughout the whole cooling lake system. And the fact that DIKE III is the nearest station to the condenser discharge among the diagnostic control points agrees with the hypothesis that periodicity of the raw error at the station is more related with the forced heat flux terms, namely, evaporation, conduction and back radiation; among these, evaporative transfer is most important because of its magnitude and uncertainty.

A periodic function was used to adjust the evaporation formula used in the WHTF in the calibrated model (p. 74, Wells et al.) such that

$$\alpha = 0.75 + 0.15 \sin\left(\frac{2\pi t}{365.25} - 1.03\right) \qquad (5.15b)$$

where t = Julian day of the year and α = correction factor multiplying the evaporative heat flux. This correction was intended to reduce the periodicity of the raw error. Justification for the correction was

that, owing to seasonal effects on wind direction, both air temperature and relative humidity are expected to be higher over the WHTF (relative to the meteorological tower measurements) in autumn, leading to lower evaporation than that predicted by meteorological tower data.

However, the model has also been run with a constant value of α = 0.75. (Figure 5.5 shows the corresponding raw error at DIKE III.) Comparison of Figures 4.3b and 5.5 shows that there is no significant improvement regarding the periodicity by applying the periodic function. In fact, harmonic analyses of the respective raw error time series show that amplitude of the raw error is only reduced from 1.096 °C to 1.080 °C (see Table 5.3), which is insignificant for practical purposes.

Fourier decomposition of the raw errors (with α = 0.75) gives ϕ_e = -0.4576 (i.e., $E(t) = 1.096 \sin(\omega t - 0.4576) + \varepsilon(t)$) which corresponds to a maximum over-prediction at t = 118 (or about April 28) and a maximum under-prediction at t = 301 (or about October 28). It is interesting, however, that the above phase is somewhat different from what one estimates visually from Figure 4.3b (i.e., peak on about June 1 and trough on about December 1). This deviation can be explained by the existence of "secondary" peaks to the left of the main peaks and these "secondary" peaks, though visually non-imposing, may carry considerable weight in determining the value of ϕ_e. The above discussion probably explains why Wells et al. (1982) employed a periodic function (Eq. 5.15) which gives a minimum value of evaporation over the WHTF on about December 1 and a maximum value on about June 1 in the calibrated model but was unable to reduce the periodicity of the raw error. Although there exist uncertainties associated with the

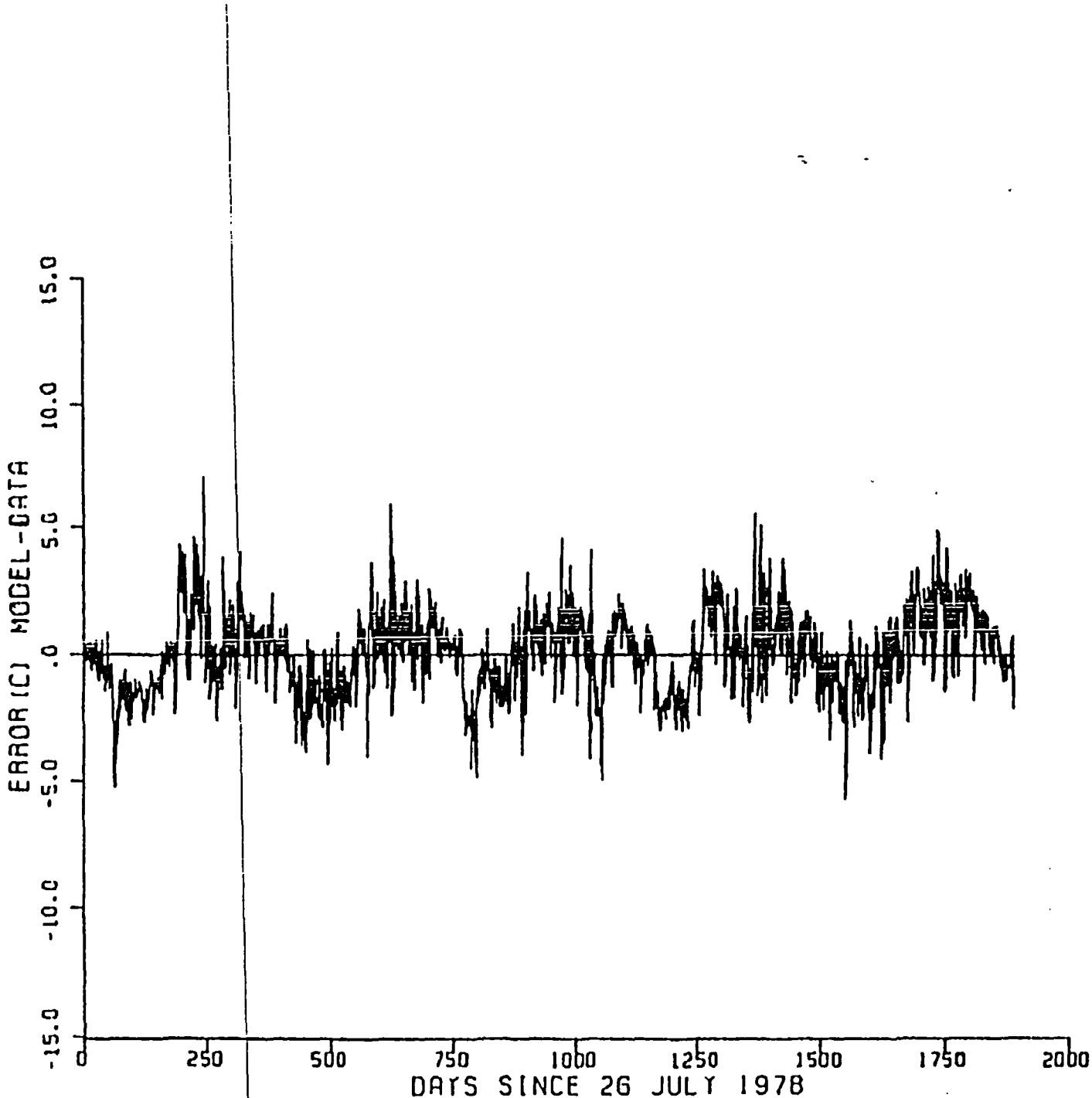


Figure 5.5 Raw Error at DIKE III (α = 0.75)

Table 5.3 Harmonic Analysis of Raw Error (DIKE III) Time Series

Description of Time Series	Raw Error Time Series		Measurements vs. Corresponing Predictions		Reference
	ϕ_e (radian)	E* (°C)	$\Delta\phi=\phi_p-\phi_m$ (radian)	$\Delta\phi$ (days)	
$\alpha = 0.75$	-0.4576	1.096	0.0939	5.46	Fig. 5.5
$\alpha = 0.75 + 0.15 \sin(\omega t - 1.0292)$	0.0769	1.080	0.0810	4.71	Fig. 4.3b
$\alpha = 0.75 + 0.15 \sin(\omega t - 0.4576)$	-0.3404	0.451	0.0383	2.23	-
$\alpha = 0.75 + 0.20 \sin(\omega t - 0.4576)$	-0.2249	0.232	0.0192	1.12	-
$\alpha = 0.75 + 0.25 \sin(\omega t - 0.4576)$	0.7276	0.045	0.0013	0.07	Fig. 5.6

annual variability in wind direction, the argument given by Wells et al. (pp. 72-74) provides the motivation for exploring a new periodic function for α so as to minimize the periodicity of the raw error.

Now that the immediate cause for the annual periodicity of the raw error at DIKE III is known (namely, measurements are lagging behind predictions by about 5.5 days), a new periodic function for α can be used so as to minimize the phase lag ($\Delta\phi = \phi_p - \phi_m$).

Knowing the value of ϕ_e for the raw error time series with α = 0.75 (see Table 5.1), we can try the following periodic function:

$$\alpha = 0.75 + 0.15 \sin(\frac{2\pi t}{365.25} - 0.4576) \qquad (5.34)$$

Table 5.3 shows that the amplitude of the resulting raw error is reduced to 0.45 °C. With a little trial and error, the following periodic function is found to be near optimal:

$$\alpha = 0.75 + 0.25 \sin(\frac{2\pi t}{365.25} - 0.4576) \qquad (5.35)$$

Eq. 5.35 gives maximum evaporation (α = 1) on about April 28 (t = 118) and minimum evaporation (α = 0.5) on about October 28 (t = 301). This new correction function of α essentially shifts predictions back in phase with the measurements (new $\Delta\phi$ = 0.0013 or 0.07 days along the time axis; see Table 5.3). Amplitude of the resulting raw error (Figure 5.6) is further reduced to 0.045 °C, less then 5% of the original value.

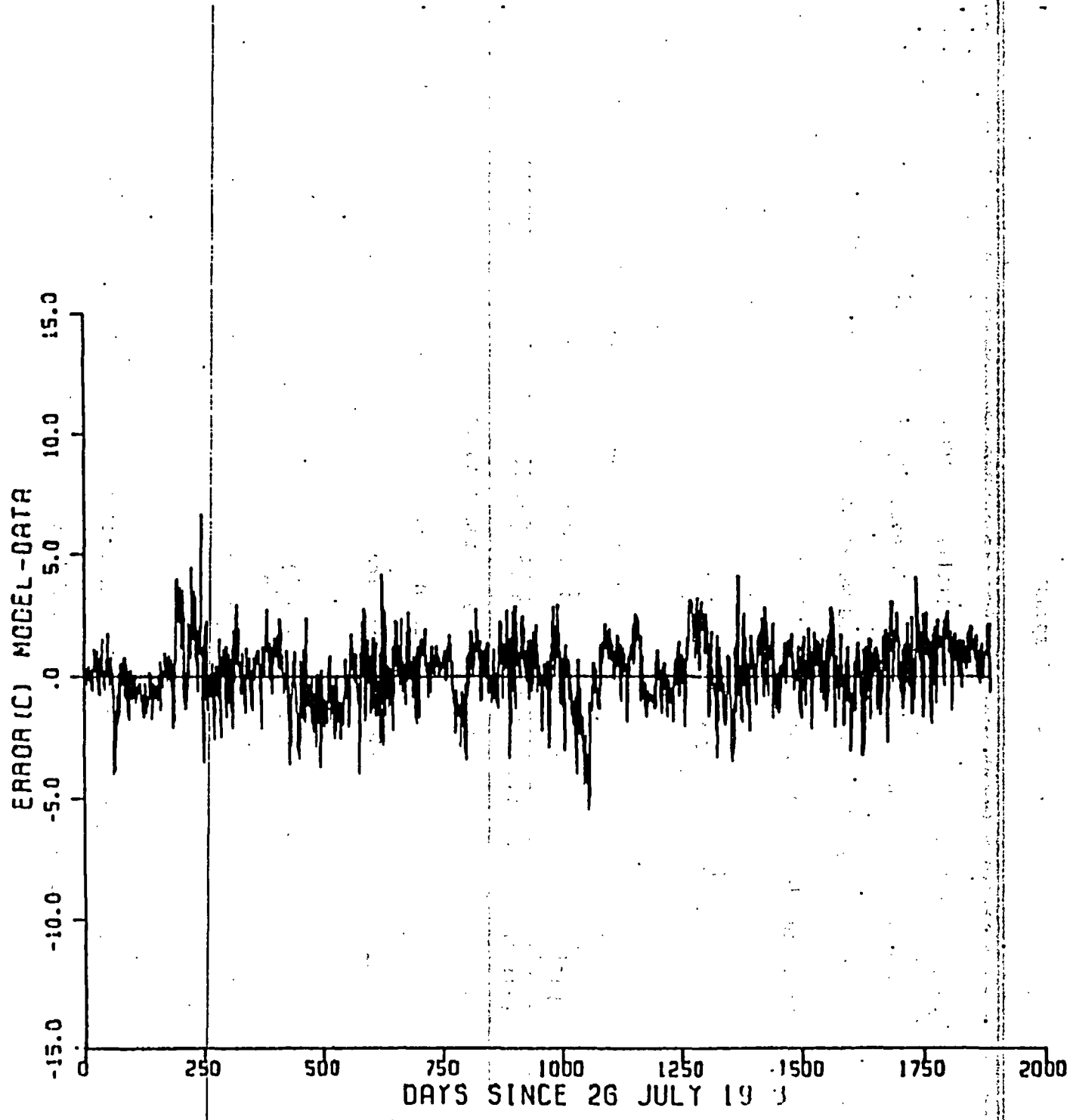


Figure 5.6 Raw Error at DIKE III ($\alpha = 0.75+0.25 \sin(\omega t-0.4576)$)

5.2.3 Errors in System Reponse

5.2.3.1 Spatially Averaged Response

A possible reason for the observed annual periodicity of the raw error at DIKE III is that the spatially averaged response of the model to periodic warming is too quick. In order to examine this possibility, a simple fully-mixed water body is considered with transient heat budget given by

$$\frac{dT}{dt} = - \frac{K}{\rho c \bar{h}} (T - T_e) \tag{5.36}$$

where T is the transient temperature of the water body, c is the specific heat capacity of water, $\bar{h}$ is the average pond depth (volume/surface area), and surface heat exchange has been linearized using the equilibrium temperature T_e and a constant surface heat exchange coefficient K. Furthermore, let the equilibrium temperature be represented by a periodic function such that

$$T_e(t) = T_e^* \sin(\omega t + \phi_{T_e}) \tag{5.37}$$

Where T_e^* and ϕ_{T_e} are the corresponding amplitude and phase, respectively.

If we assume that the true temperature T = T(t) is accurately represented by the measurements, M(t) (refer to Eq. 5.1), then it follows that we can write

$$T(t) = M^* \sin(\omega t + \phi_m) \tag{5.38}$$

Differentiating Eq. 5.38, we have

$$\frac{dT}{dt} = \omega M^* \cos(\omega t + \phi_m) \qquad (5.39)$$

Equating Eqs. 5.36 and 5.39, one can arrive at a simple relationship between the amplitudes of the two time series and also an expression for the phase shift, $\phi_{T_e} - \phi_m$ (Adams, 1982):

$$M^* = T_e^* \frac{1}{\sqrt{1 + (\omega\tau)^2}} \qquad (5.40)$$

and $$\phi_{T_e} - \phi_m = \tan^{-1}(\omega\tau) \qquad (5.41)$$

where $\tau = \rho ch/K$ is the time constant for surface warming (or cooling), $\omega = 2\pi/T$ and $T = 365.25$ days.

Resorting to the Imperial Units for a quick check, we have $\rho c \approx 62$ Btu ft^{-3} $°F^{-1}$; if we assume the values of $h = \bar{2}5$ ft and $K = 150$ Btu ft^{-2} $°F^{-1}$ day^{-1} for the North Anna cooling lake system, then $\tau \approx$ 10 days, $\omega\tau = 0.178$ and Eq. 5.41 will give $\phi_{T_e} - \phi_m = 0.176$ radians or $\approx$ 10 days when translated to the time axis. However, it is known from Section 5.2.1 that $\Delta\phi = \phi_p - \phi_m = 0.0939$ or about 5.5 days between predictions and measurements (see Table 5.1); thus it is doubtful that τ is off by a factor of 2 (N.B. $\tau = \rho ch/K$). Clearly, the values of depth used in the models are reasonably accurate and, if K were that far off, then the average error at DIKE III would not have been near zero. In addition, error in τ would have been manifest throughout the cooling lake system, not primarily at DIKE III. Based on the above argument, we can rule out the possibility of major errors associated with spatially averaged system response.

5.2.3.2 Longitudinal System Response

In order to account for transients in the WHTF a lag-time correction is incorporated in the model. This procedure was utilized, rather than a forward stepping transient simulation, because of the absence of transient analysis for the side arm flows. A residence time (t_r) is computed dynamically (i.e., once each day) for each pond within the WHTF based on flow rate and pond size. The temperature at the end of the pond on day t is determined from the upstream temperature at the entrance of the same pond on day $t - t_r$ and the degree of warming or cooling during the time span of t_r. In other words,

$$T(t) = f(Q(t - t_r), K(t), T_e(t), T_o(t - t_r)) \qquad (5.42)$$

where T_o is the upstream temperature of the pond, K is the surface heat exchange coefficient, T_e is the equilibrium temperature, Q is the condenser flow rate, and T refers to the temperature at the end of the pond.

Since meteorological variables act over the entire period of t_r days, rather than on one day, t, Wells et al. (1982) implemented an exponential filtering scheme, as described by Adams and Koussis (1980), for T_e and K in order to approximate the correct time response. The equation describing the exponential filter for T_e (and similar for K) was

$$\langle T_e(t) \rangle = \frac{\sum_{n=1}^{t_r/\Delta t} T_e(t - n\Delta t) \exp(-(n-1)\, k'\, \Delta t)}{\sum_{n=1}^{t_r/\Delta t} \exp(-(n-1)\, k'\, \Delta t)} \qquad (5.43)$$

where $k' = K'/\rho C_p h$ is the kinematic surface exchange coefficient, h is the averge depth of pond, $< >$ denotes the filtered variable, Δt is the time step (one day), and K' represents the filtered surface heat exchange coefficient. The filtered value of T_e is then used in a steady state analysis of heat loss within the ponds. Thus for a pond m with no side arm,

$$T_m(t) = \langle T_e(t) \rangle + [T_{m-1}(t-t_{r_m}) - \langle T_e(t) \rangle] e^{-kt_{r_m}} \quad (5.44)$$

In general, the lagging and filtering combination resulted in a much smoother and accurate prediction of transient temperatures throughout the WHTF (Wells et al., 1982). However, it is possible that the computed residence times may be off by several days, thus causing the phase lag between measurements and predictions of the surface temperature at DIKE III.

Typical values of the residence times (as computed by the North Anna Cooling Lake Model) for Ponds 1, 2, 3 of the WHTF for one-unit operation are 0.5, 3 and 2 days, respectively, with correspondingly lower values for 2-unit operation. In order to see how sensitive model predicitions are to different residence times, the model was run with various sets of fixed residence times for the three ponds of the WHTF. Statistical data were compiled and harmonic analyses made for the raw error at DIKE III. Results are given in Table 5.4. (Notice that these runs are made with a constant factor for evaporation ($\alpha = 0.75$).) The results show that predictions are generally improved by an increase of total residence time (t_r) of the WHTF as indicated by the decrease of

Table 5.4 Sensitivity Test of Raw Errors at DIKE III with Respect to Different Residence Times (Runs 1 - 7)

Run No.	Fixed Residence Times Used for Ponds 1, 2, 3 (Day)	Total Residence Time for WHTF (Day)	Mean (°C)	Standard Deviation (°C)	ϕ (radian)	E* (°C)
1	0, 0, 0	0	0.13	1.98	-0.4716	1.63
2	1, 1, 1	3	0.05	1.63	-0.4815	1.27
3	1, 2, 2	5	0.03	1.52	-0.4408	1.03
4	2, 3, 3	8	0.03	1.43	-0.4719	0.78
5	4, 5, 5	13	0.04	1.41	0.0842	0.33
6	5, 6, 6	17	0.04	1.43	1.2078	0.26
7	10,10,10	30	0.00	1.87	-4.2065	1.27
8	residence times computed dynamically for each day	based on whole volume	0.27	1.60	-4.3033	0.52
9		based on 75% of whole volume	0.31	1.44	0.9827	0.22
10		original computation	0.04	1.59	-0.4576	1.10

N.B. ϕ and E* denote the phase and amplitude of a sine curve fitted to the raw error time series (see Section 5.2.1).

E*. The above findings provide a motivation for re-examining the residence time calculation as employed by Wells et al. (1982).

It is noticed that in Wells et al. (1982) and previous versions of the cooling lake model, residence times of the WHTF were computed such that

$$\text{RESTIM} = \frac{\text{ULD}_1 \times A_1}{Q} + \frac{\text{ULD}_2 \times A_2}{Q - \text{QSA}_1} + \frac{\text{ULD}_3 \times A_3}{Q - \text{QSA}_1 - \text{QSA}_2} \qquad (5.45)$$

where RESTIM is the residence time of each reach of the WHTF, ULD_i represents the upper layer depth of section i within that reach, A_i is the surface area of section i, Q is the rate of flow entering the reach, and QSA_j denotes the side arm flow rate of the jth side arm. This routine is repeated for all three of the reaches within the WHTF.

Eq. 5.45 gives only the residence times of the upper layers of the main WHTF ponds while ignorning the volume of the lower layers and that of the side arms. Physically, this represents a lower limit on the true residence time. An upper limit (and perhaps more physically realistic estimate) would be based on the entire volume. Accordingly, Eq. 5.45 is replaced by

$$\text{RESTIM} = \frac{(A_1 + A_2 + A_3) \times D + H_1 \times \text{SA}_1 + H_2 \times \text{SA}_2}{Q_p} \qquad (5.46)$$

where SA_j and H_j represent the surface area and average depth, respectively, of side arm j, Q_p is the condenser flow rate, and D is the average depth of the main ponds within the WHTF.

The model was run using Eq. 5.46 (and $\alpha = 0.75$) to compute residence times dynamically. Statistics of the raw error are included

in Table 5.4 (Run 8). It is noticed that periodicity of the raw error at DIKE III is reduced substantially, as indicated by the reduction of the amplitude to less than half (E* = 0.52 °C, Table 5.4) of its original value (E* = 1.10 °C, Table 5.1). Typical values of the computed residence times in this run are around 1, 8 and 3 days (or a total of about 12 days), with maximum values reaching 3, 24, and 9 days for Ponds 1, 2 and 3, respectively. However, the value of $\Delta\phi$ ($= \phi_p - \phi_m$, see Section 5.2.1) in this run is about -2 days, whereas $\Delta\phi$ = 5.5 days with Eq. 5.45. In other words, on average, predictions lag behind the measurements, indicating that the computed residence time may have been too large.

A final sensitivity run was made using a dynamic residence time based on 75% of the entire volume (i.e., the right-hand side of Eq. 5.46 was multiplied by 0.75). A re-run of the model shows that periodicity of the raw error at DIKE III (Figure 5.7) is further reduced to give an amplitude of about 0.22 °C (refer to Run 9 of Table 5.4). (N.B. Phase lag between measurements and predictions in this case is almost reduced to zero ($\Delta\phi$ = 0.1 day) whereas P* = 11.463 °C and M* = 11.687 °C (refer to Eqs. 5.2 and 5.3).) This suggests that the finite raw error, although with greatly reduced amplitude, is largely caused by the fact that M* ≠ P* (see also Section 5.2.1). Notice that applying the correction factor to reduce the computed residence times is physically justified since some pockets of water in the WHTF (and particularly in the side arms) are stagnant and do not circulate very much (Adams and Wells, 1984). Still, the value of 75% is approximate, and though the computed residence times look reasonable, ideally they should be confirmed with a dye study. Such a study would elucidate not

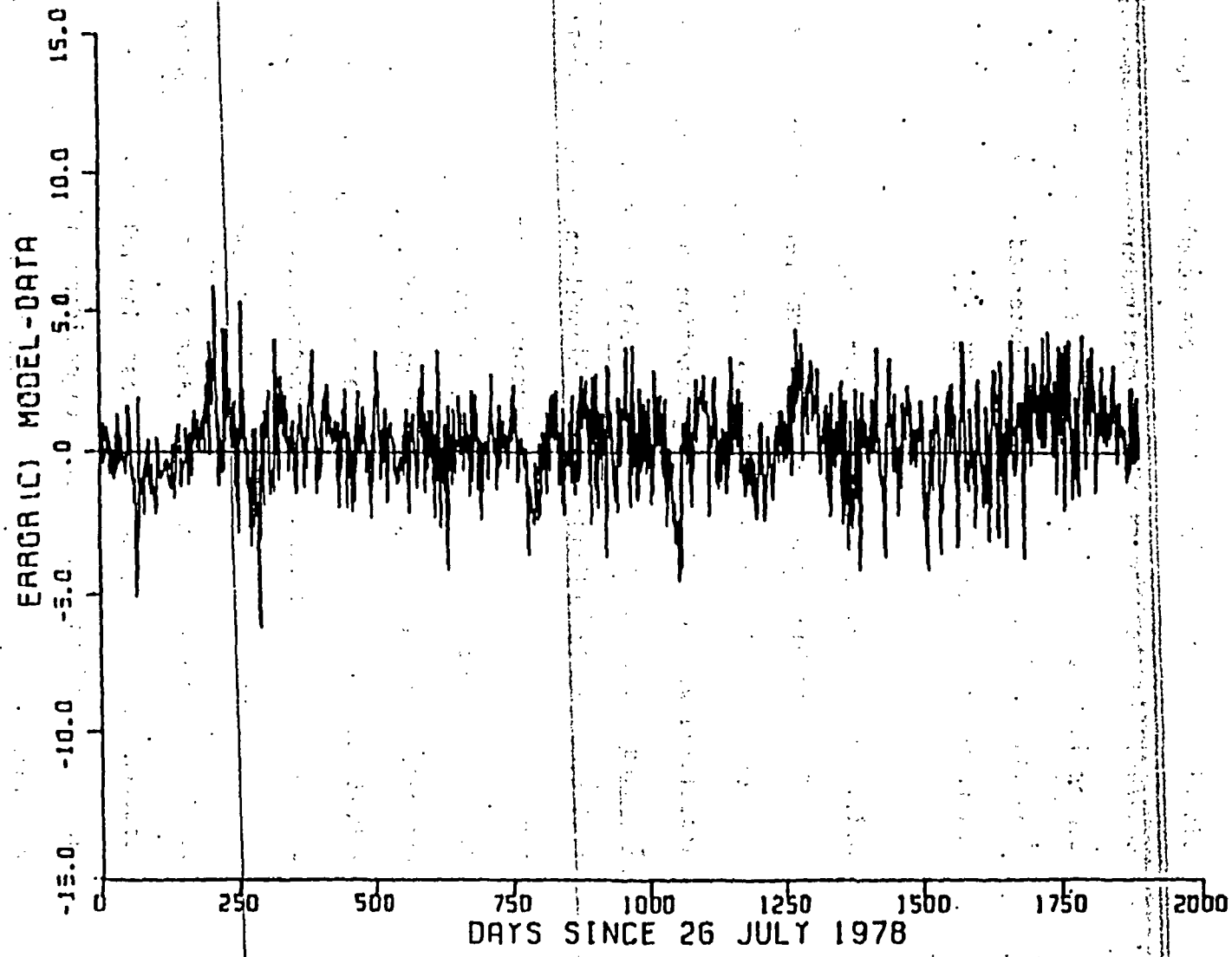


Figure 5.7 Raw Error at DIKE III (α = 0.75; lag time over WHTF based on 75% of whole volume)

only the effective residence time but also the complete residence time distribution which is needed if a more complete transient analysis is desired. In the absence of such confirmation, Eq. 5.46 with a 75% correction factor is used because it appears more reasonable than Eq. 5.45 and significantly reduces the observed phase lag between measurements and predictions at DIKE III.

5.2.4 Conclusion of Transient Error Analysis

Two possible reasons for the annual periodicity of the raw errors observed primarily at DIKE III have been identified, namely, errors associated with (1) the forcing function and (2) the system response. Two candidates for each of the possible reasons have been considered: (1a) atmospheric radiation, (1b) evaporation, (2a) spatially-averaged system response, and (2b) longitudinal system response. Among the four candidates, both (1a) and (2a) are ruled out. (1b) is possible since a moderate annual adjustment in the rate of evaporation over the WHTF ($\alpha = 0.75 \pm 0.25$) essentially eliminates the periodicity of the raw error. Some seasonal adjustment of evaporation rate can be justified by seasonal differences in wind direction. However, there is no way of verifying the exact magnitude because of insufficient meteorological measurements. (2b) seems to be most plausible because of the strong sensitivity of the periodicity to residence time and because of the uncertainties in the actual residence time. The adopted approach wherein residence time is based on a fraction (75%) of the entire volume seems more rational than that used previously and also results in a substantial reduction of the periodicity (reduced to less than a quarter of the original value in terms of amplitude of the raw error

and to near zero in terms of phase difference between measurement and prediction). However, we emphasize that there is still some uncertainties concerning the actual residence time distribution which would require a dye study for further confirmation.

5.3 Steady State Errors at DISCHARGE

Error analysis in Chapter 4 shows that the surface temperature at DISCHARGE is consistently over-predicted by the model by an average of about 0.7 °C while the mean error at INTAKE is 0°C. Predictions between INTAKE and DISCHARGE depend primarily on the condenser temperature rise which, in turn, depends on the rate of waste heat rejected to the WHTF and the circulating flow rate. These two variables are computed from daily-averaged values of electrical power production and the number of circulating water pumps using formulae suggested by VEPCO and reported by Wells et al. (1982).

To test model sensitivity to a small decrease in condenser temperature rise, three different runs of the model were made:

(0) no adjustment applied,

(1) $Q_1 = Q_o \times 1.1$, $\Delta T_1 = \Delta T_o/1.1$; i.e., total heat load remains unchanged while temperature rise is reduced by about 9%;

(2) $Q_1 = Q_o$, $\Delta T_1 = \Delta T_o/1.1$; i.e., both total heat rejected and temperature rise are reduced by about 9%.

In the above, Q_1 and ΔT_1 represent the adjusted flow rate and temperature rise across the plant condenser, respectively, and Q_o, ΔT_o are the reported values. (N.B., these runs are independent of the recalibration effort mentioned in Section 5.2, i.e., a value of α =

0.75 was used as the correction factor for evaporation over the WHTF, and the original residence time calculation was used.)

Mean raw errors at the four diagnostic control points for the three runs are listed in Table 5.5. It is noticed that the delta error across the plant (mean error at DISCHARGE minus mean error at INTAKE) is reduced from 0.74°C in Run 0 to about 0.20°C in Runs 1 and 2. This suggests that the optimal decrease in temperature rise should be about 13% (9%*0.74/(0.74-0.20)) and that both total heat load and condenser flow rate should be adjusted. Subsequent discussion with VEPCO confirmed that the average rate of waste heat rejection into the WHTF was about 8% less than what had previously been assumed, and that the condenser flow rate was quite sensitive to the pumping head. Another run (Run 3) was thus made based on the new heat rejection data from VEPCO and the objective of minimizing the delta error between INTAKE and DISCHARGE. Parameters were:

(3) $Q_1 = Q_0 \times 1.054$, $\Delta T_1 = \Delta T_0 \times 0.874$; i.e., total heat rejection is reduced by about 8% and temperature rise is reduced by about 13%.

Mean raw errors at the four diagnostic control points for Run 3 are also listed in Table 5.5. It is noticed that the delta error between DISCHARGE and INTAKE is greatly reduced, and that the raw error at DIKE III and BR. PT. are both acceptably small. The correction as employed in Run 3 is hence incorporated as part of the recalibration effort to serve as a basis for the final run.

Table 5.5 Mean Raw Errors for Runs 0, 1, 2 and 3

Mean Raw Error (°C)	Run 0	Run 1	Run 2	Run 3
DISCHARGE	0.74	0.29	0.11	0.00
DIKE III	0.29	0.25	0.01	0.05
BR. PT.	0.25	0.31	0.10	0.16
INTAKE	0.00	0.08	-0.09	-0.03

6. MODEL VALIDATION

6.1 Introduction

Chapter 5 described the efforts of model recalibration which were motivated by the results of surface temperature analysis in Chapter 4. Two changes of the model were made: computation of residence times in the WHTF (with a 75% correction factor applied) and adjustment of plant operation data (with reduced heat load and increased flow rate); both changes were implemented as options which could be switched off by changing an input data file to the model.

In this chapter continuous predictions given by the recalibrated model are compared with measurement data over the five-year simulation period (1978 - 1983). Section 6.2 will discuss the results of surface temperature analysis. Vertical temperature profiles in Lake Anna will be shown and discussed in Section 6.3.

6.2 Surface Temperature Error Analysis

Surface temperature error analysis is made for the four representative diagnostic control points over the cooling lake system (refer to Figure 4.1). Figures 6.1a to d show the raw errors and their statistical information is given in Table 6.1. Comparison with statistics given by the model described in Wells et al. (1982) shows that the mean raw error at DISCHARGE is reduced substantially without additional under-prediction at INTAKE, which has already been negligible. The model is also seen to have improved at other stations in terms of reduced standard deviations and/or magnitudes of means.

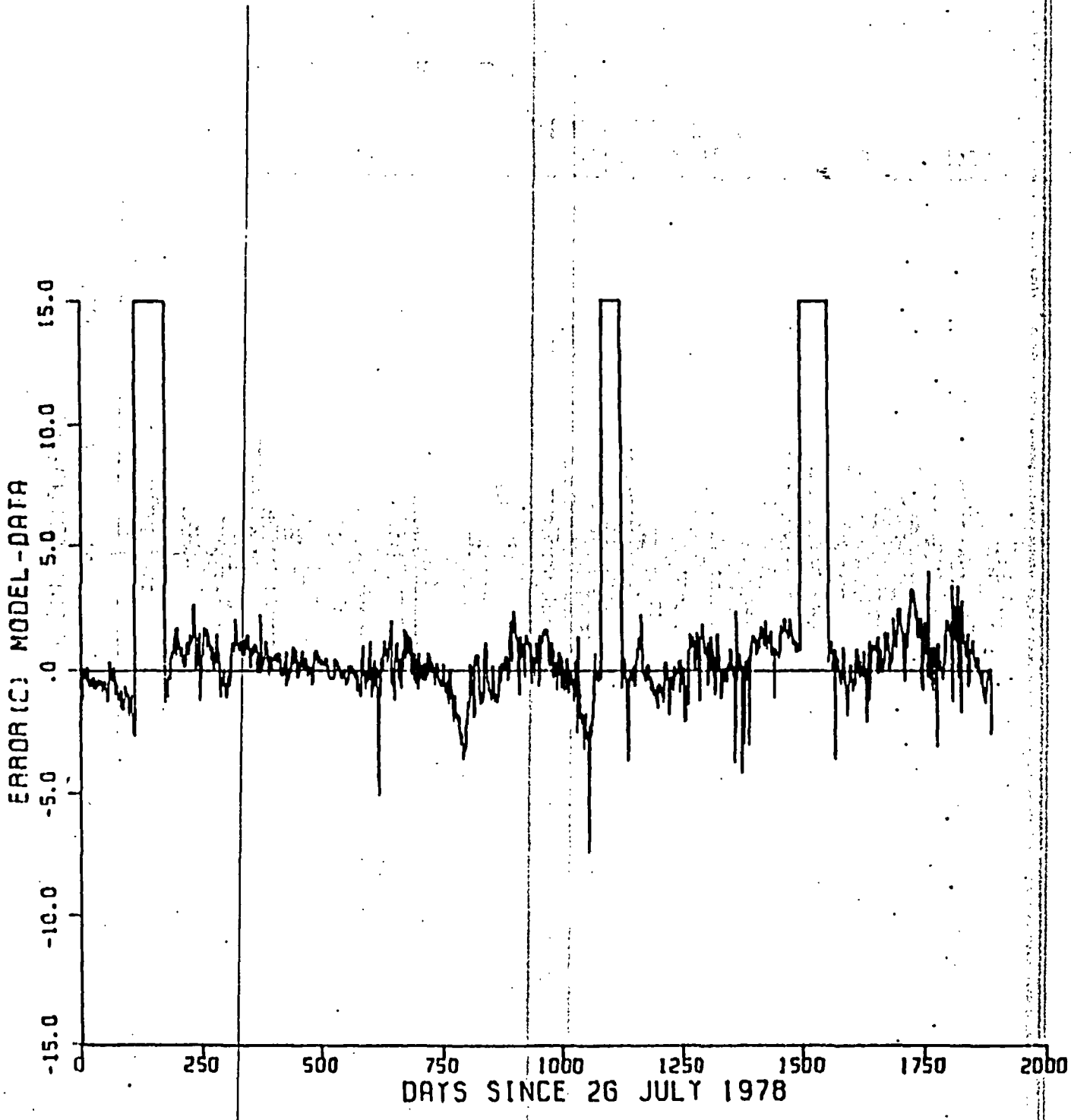


Figure 6.1a Raw Error at DISCHARGE (Recalibrated Model)

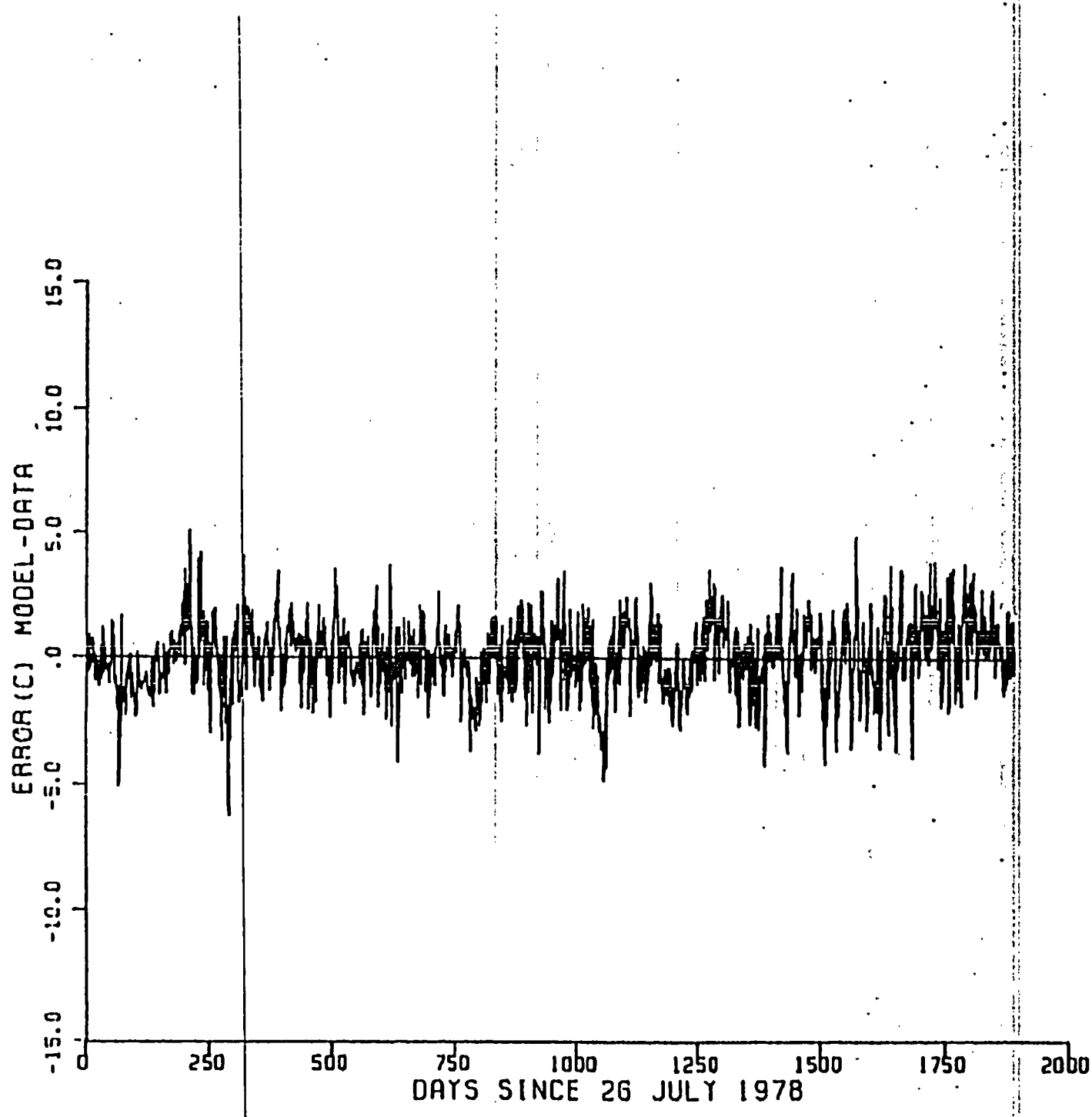


Figure 6.1b Raw Error at DIKE III (Recalibrated Model)

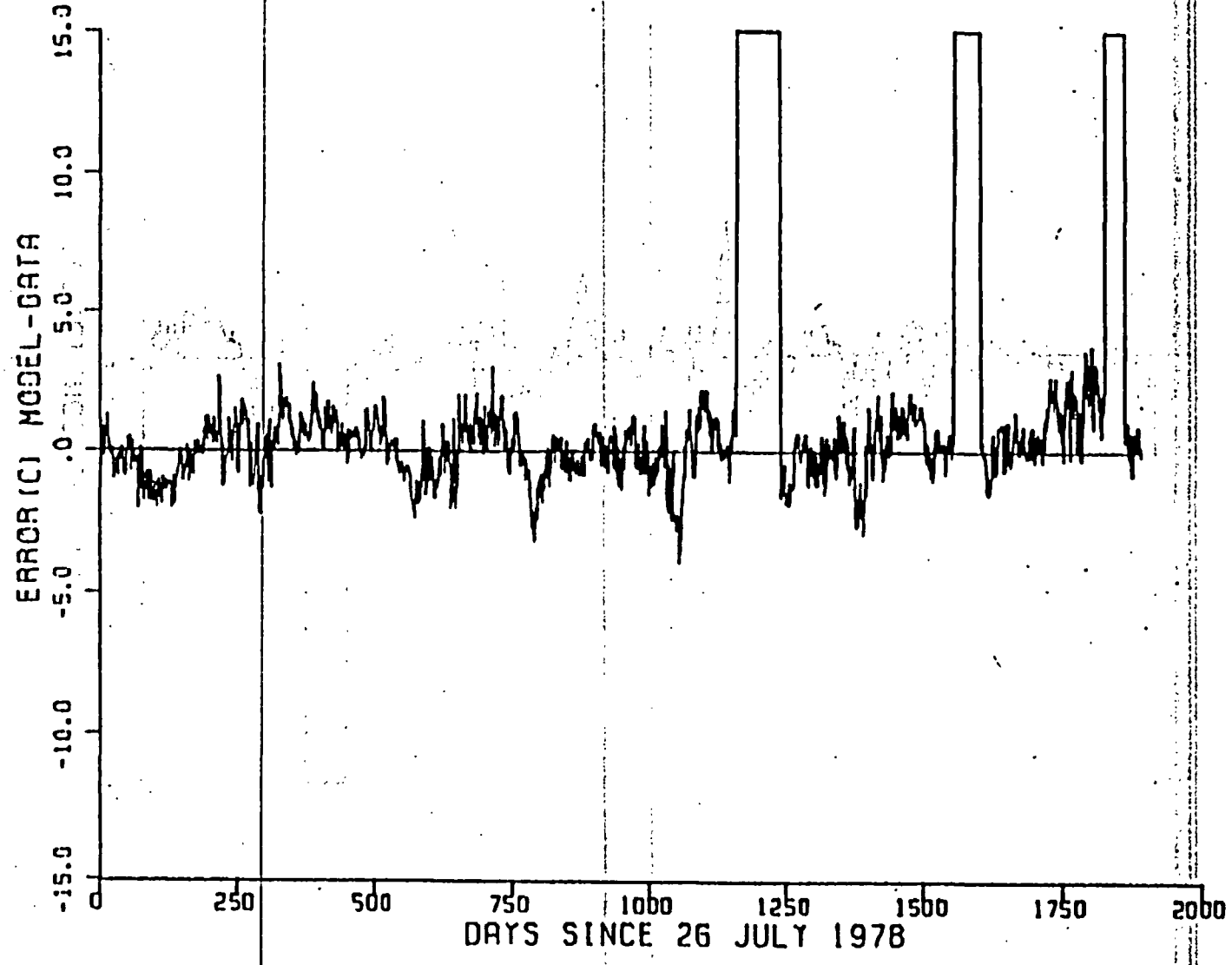


Figure 6.1c Raw Error at BR. PT. (Recalibrated Model)

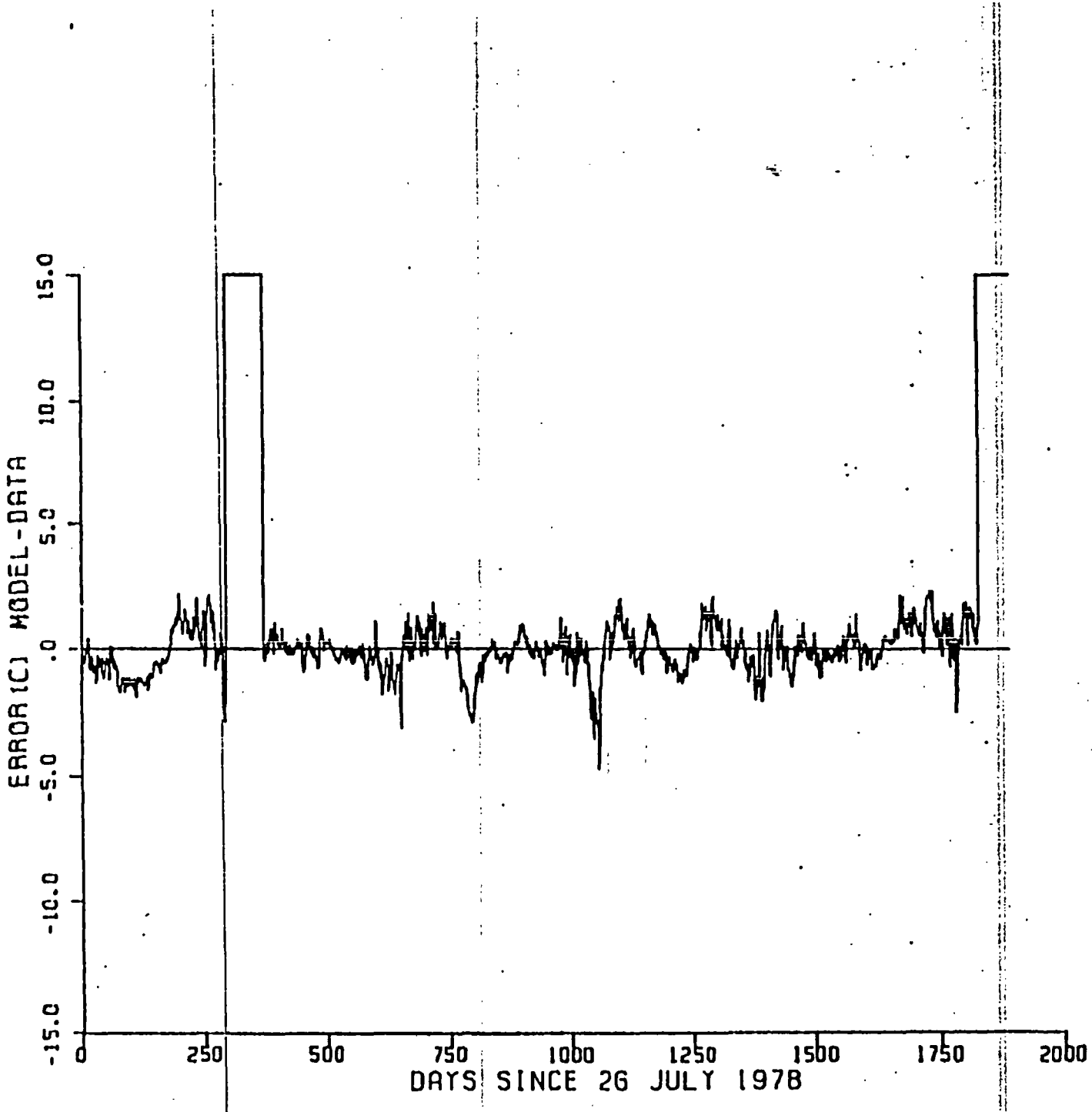


Figure 6.1d Raw Error at INTAKE (Recalibrated Model)

Table 6.1 Comparison of Statistics of Raw Errors at Four Diagnostic Control Points between Recalibrated Model and Model in Wells et al. (Five-Year Simulation for Both Cases).

Station	Mean (°C)		Standard Deviation (°C)		Sample Size
	Recalibrated Model	Wells et al.	Recalibrated Model	Wells et al.	(Day)
DISCHARGE	-0.02	0.69	1.11	1.23	1633
DIKE III	0.07	-0.03	1.37	1.54	1893
BR. PT.	0.16	0.11	1.04	1.10	1733
INTAKE	0.01	-0.06	0.80	0.98	1752

Comparison between Figures 4.3 b and 6.1 b shows that the annual periodicity at DIKE III is greatly reduced. Harmonic analysis gives an amplitude of E* = 0.12 °C, about 10% of the value (E* = 1.08°C, see Table 5.3) before recalibration. This remaining amplitude seems tolerable since in general we would expect to see some error at the annual period for all four stations because temperature changes with season and the model representation of heat fluxes is not perfect; as an example, we would probably see some differences due to the use of different atmospheric radiation formulae.

In addition to overall averages for the whole simulation period, monthly averages of the raw errors (see Section 4.4) at the four diagnostic control points are also compiled and shown in Table 6.2. When compared with Table 4.4, the former shows that magnitudes of the monthly averages of raw errors are in general reduced and also have become more random especially for DIKE III.

6.3 Vertical Temperature Profiles in Lake Anna

Figures 6.2a-6.2h show the measured and predicted vertical temperature profiles in the main lake for different seasons in 1982 and 1983 at two locations: (1) near the condenser intake and (2) near the dam at the downstream end of Lake Anna. (They correspond to Stations LA9 and LA13, respectively. See Figure 3.2 of Wells et al., 1982.) Comparisons between measurements and predictions indicate that the dynamic nature of the data both in space and time is satisfactorily represented by the model's characterization of the mixed layer and the vertical heat diffusivity in the hypolimnion. In addition, it is worth commenting that, firstly, recalibration of model did not change the

Table 6.2 Monthly Averages* of Raw Errors at Four Diagnostic Control Points (Recalibrated Model)

Month	DISCHARGE	DIKE III	BR. PT.	INTAKE
JAN	0.21[4]	0.42	0.01	0.22
FEB	0.36	0.63	-0.12	0.54
MAR	0.43	0.25	0.00	0.39
APR	0.40	-0.05	0.22	0.17
MAY	-0.06	0.04	0.02	-0.17[4]
JUN	-0.02	-0.16	0.40	-0.14[4]
JUL	0.74[4]	0.53	0.89[4]	0.43[3]
AUG	-0.10[4]	0.51[6]	0.73	0.17
SEP	-0.69	-0.22[6]	0.01	-0.46
OCT	-0.51[4]	-0.44	-0.46[3]	-0.36
NOV	-0.36[4]	-0.45	-0.35[3]	-0.31
DEC	-0.30[4]	-0.19	-0.15[3]	-0.34

* all are five-year averages except as specified

[6] six-year average

[4] four-year average

[3] three year average

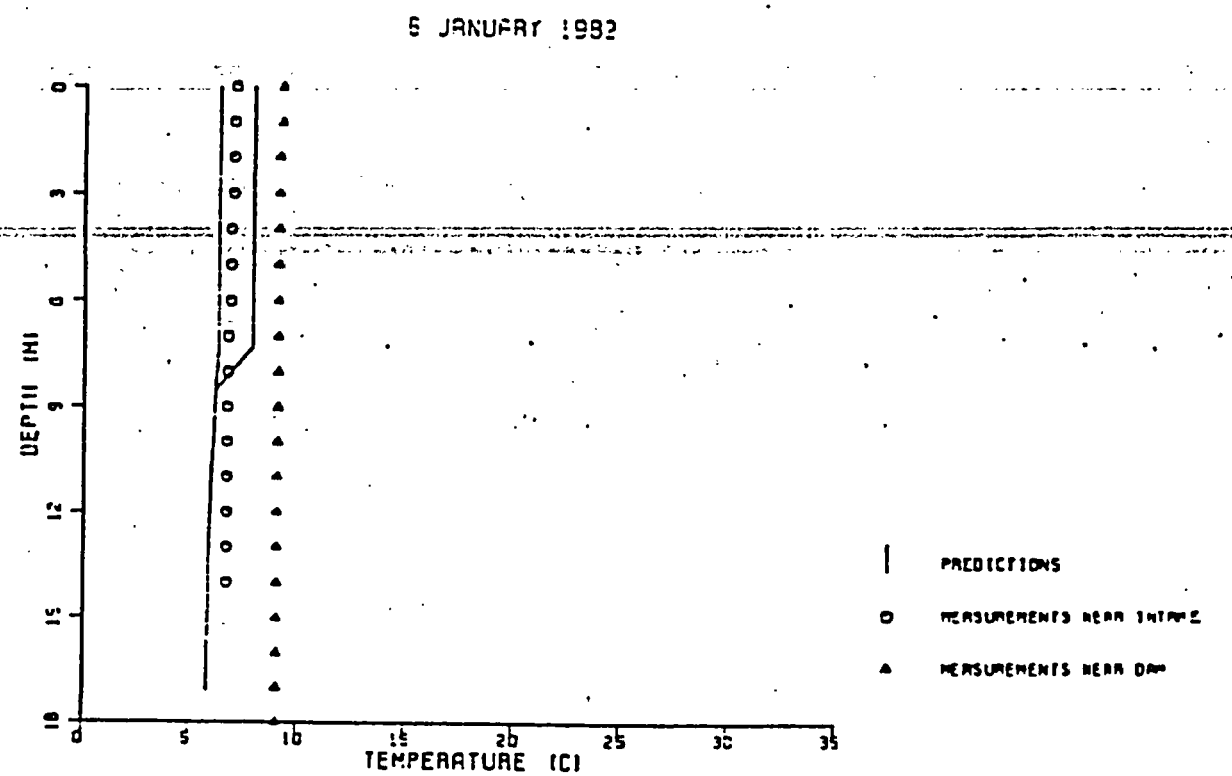


Figure 6.2a

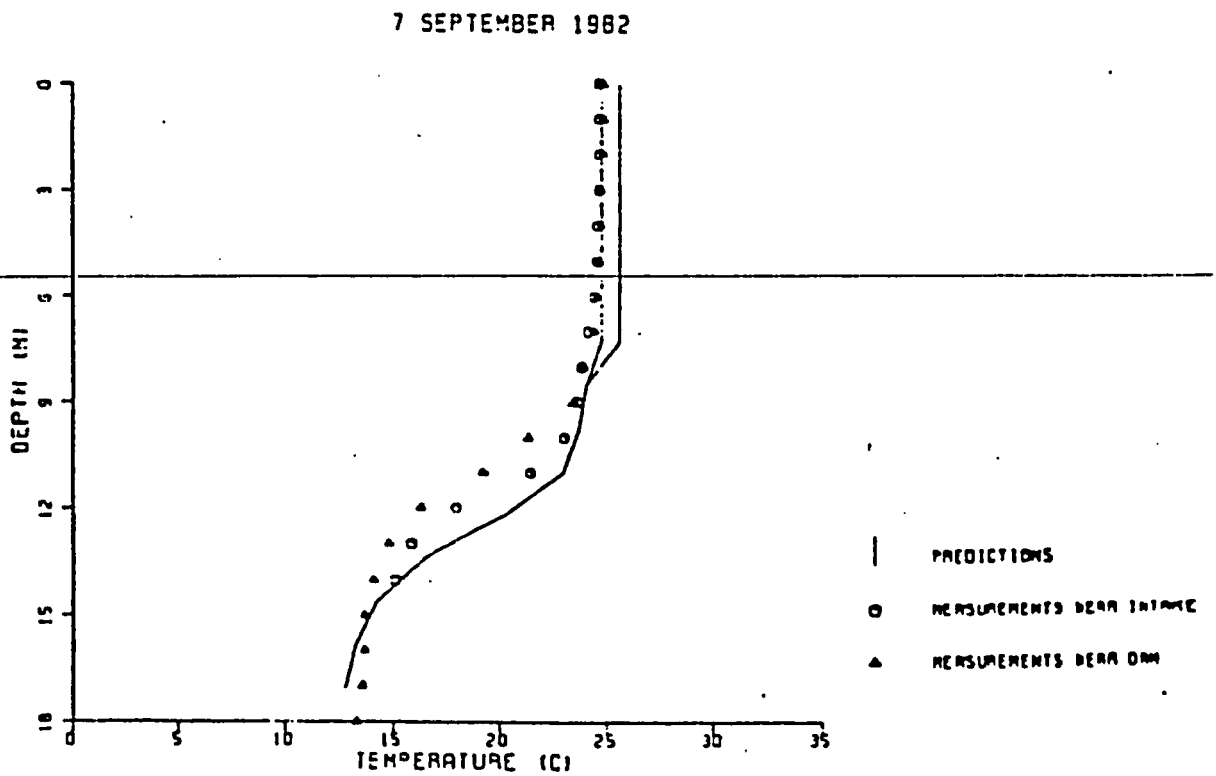


Figure 6.2b

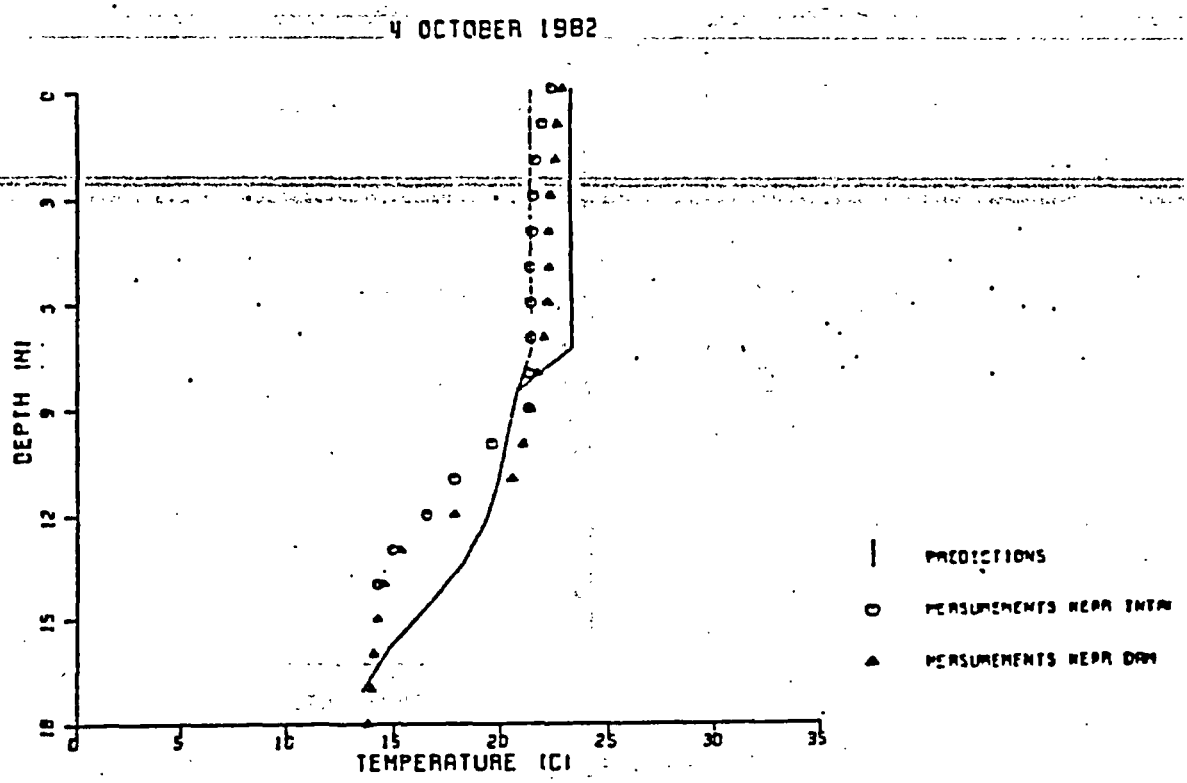


Figure 6.2c

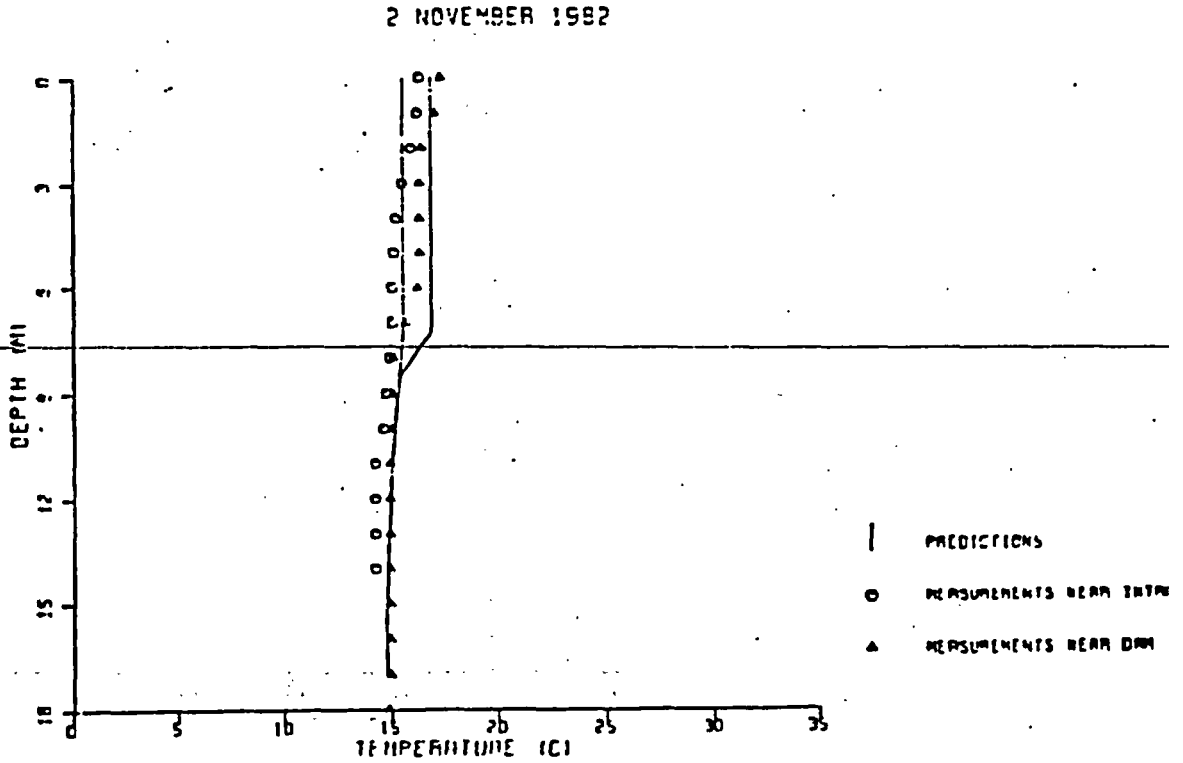


Figure 6.2d

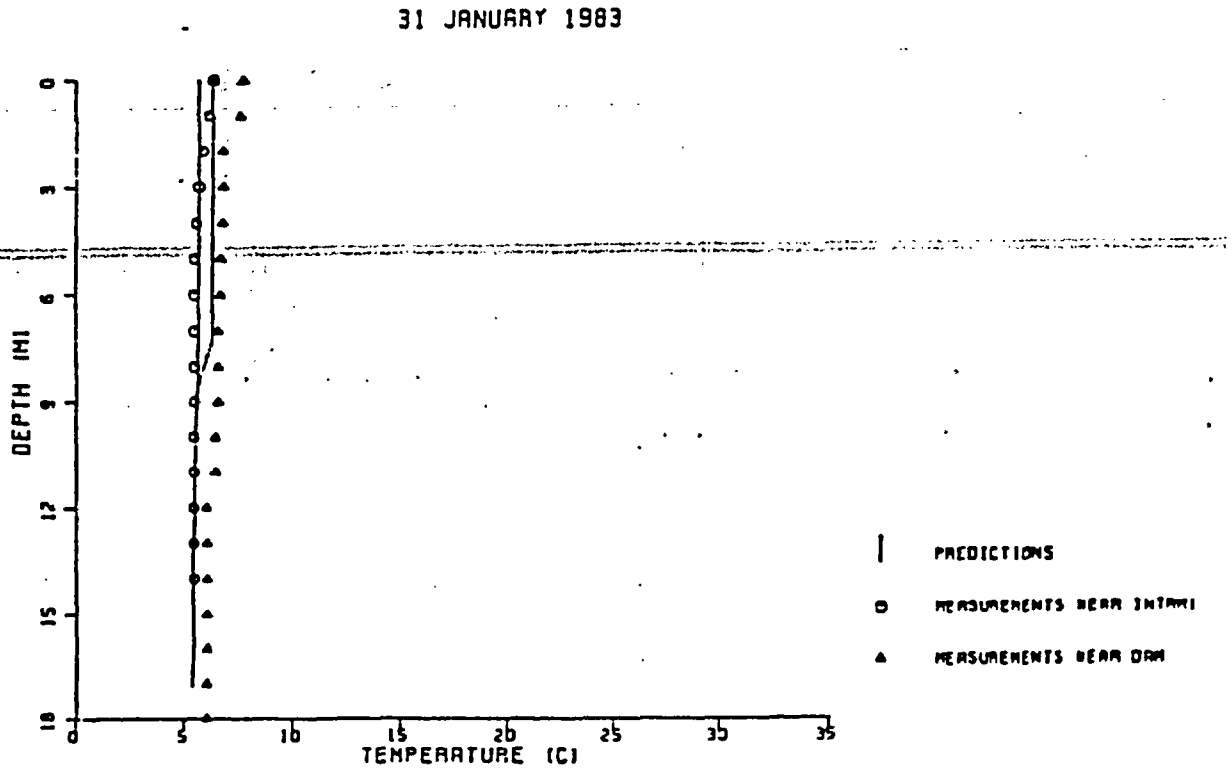


Figure 6.2e

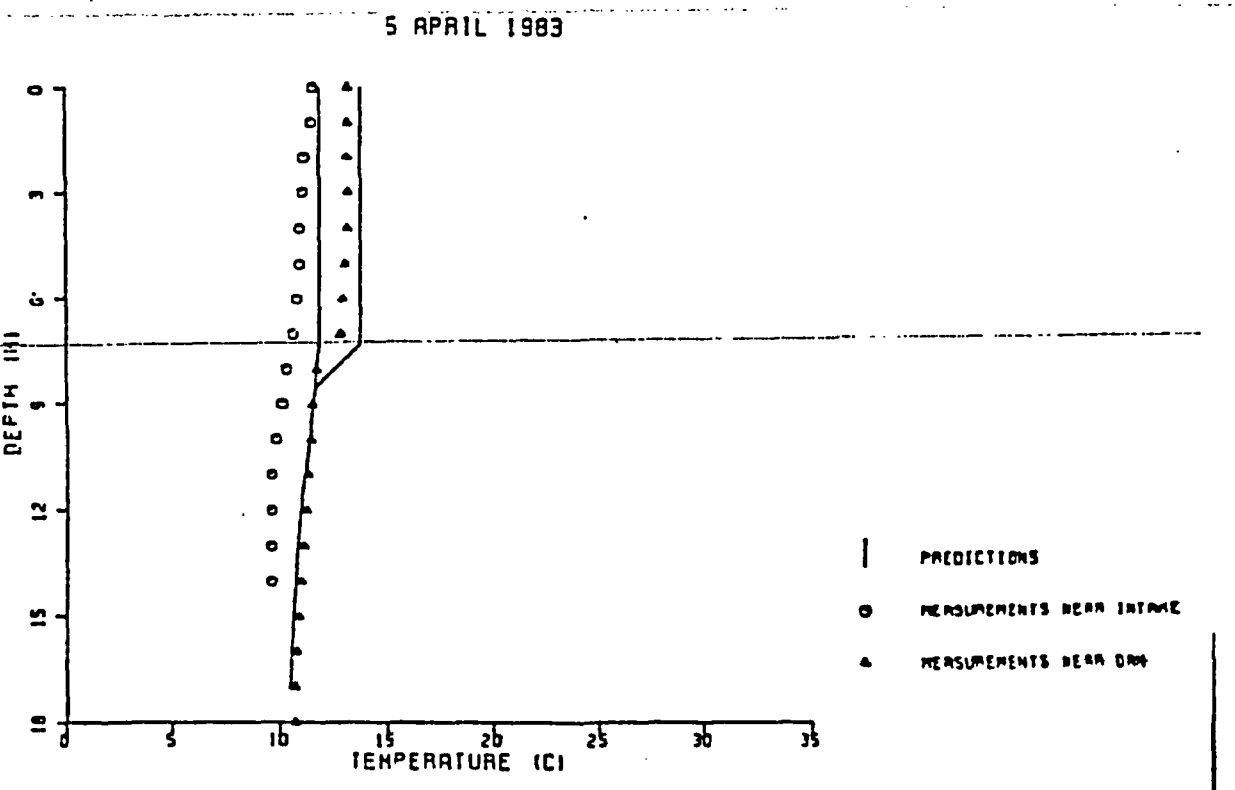


Figure 6.2f

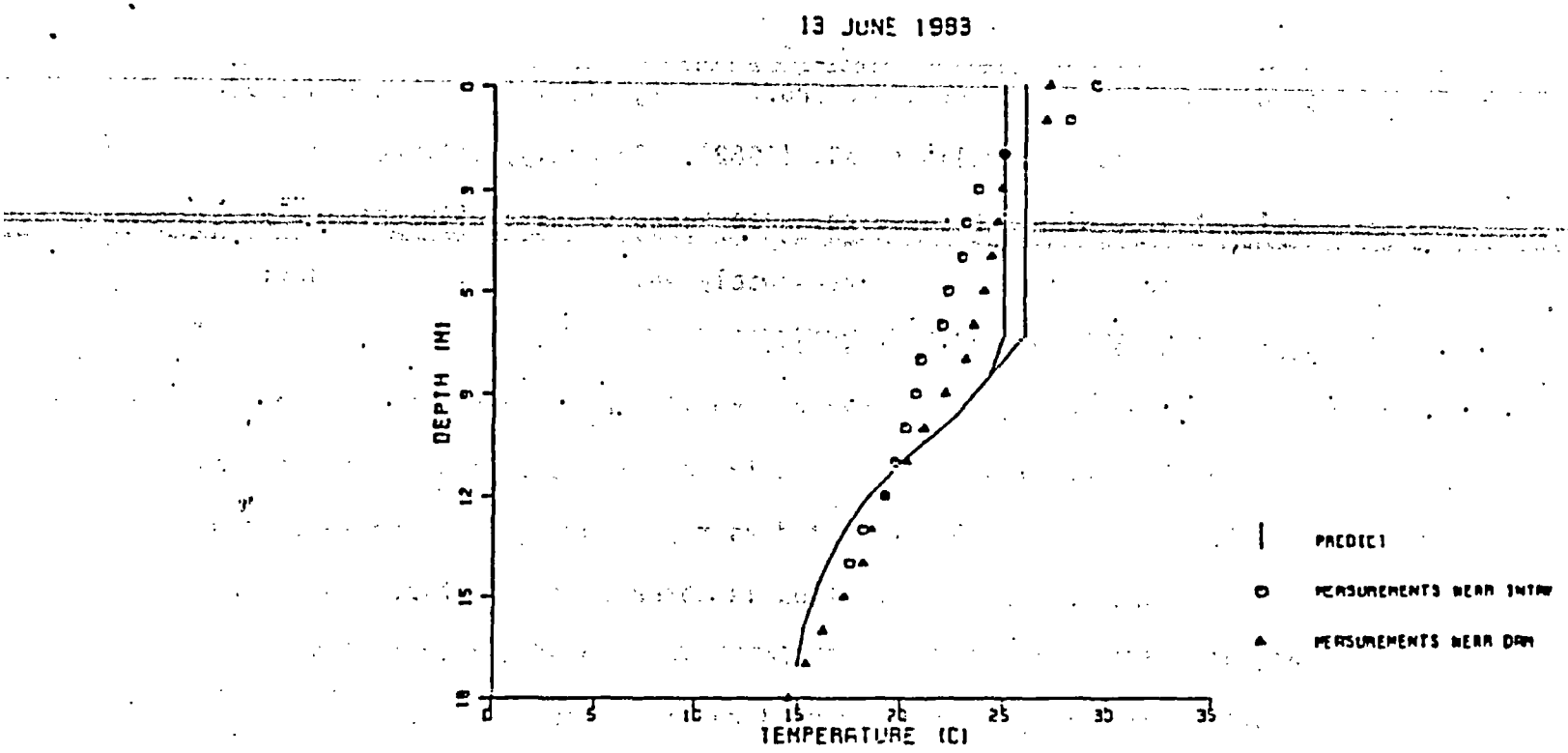


Figure 6.2g

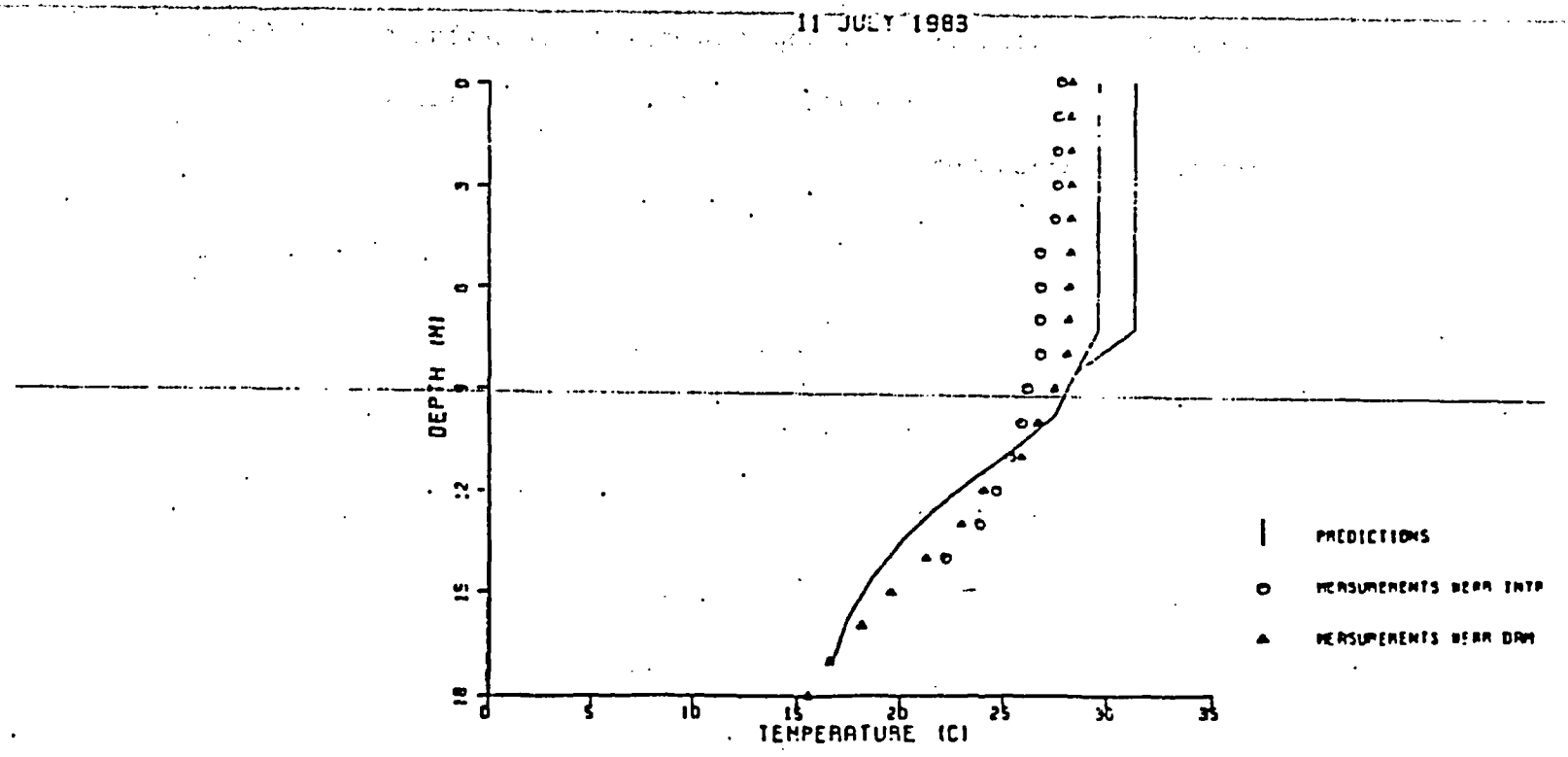
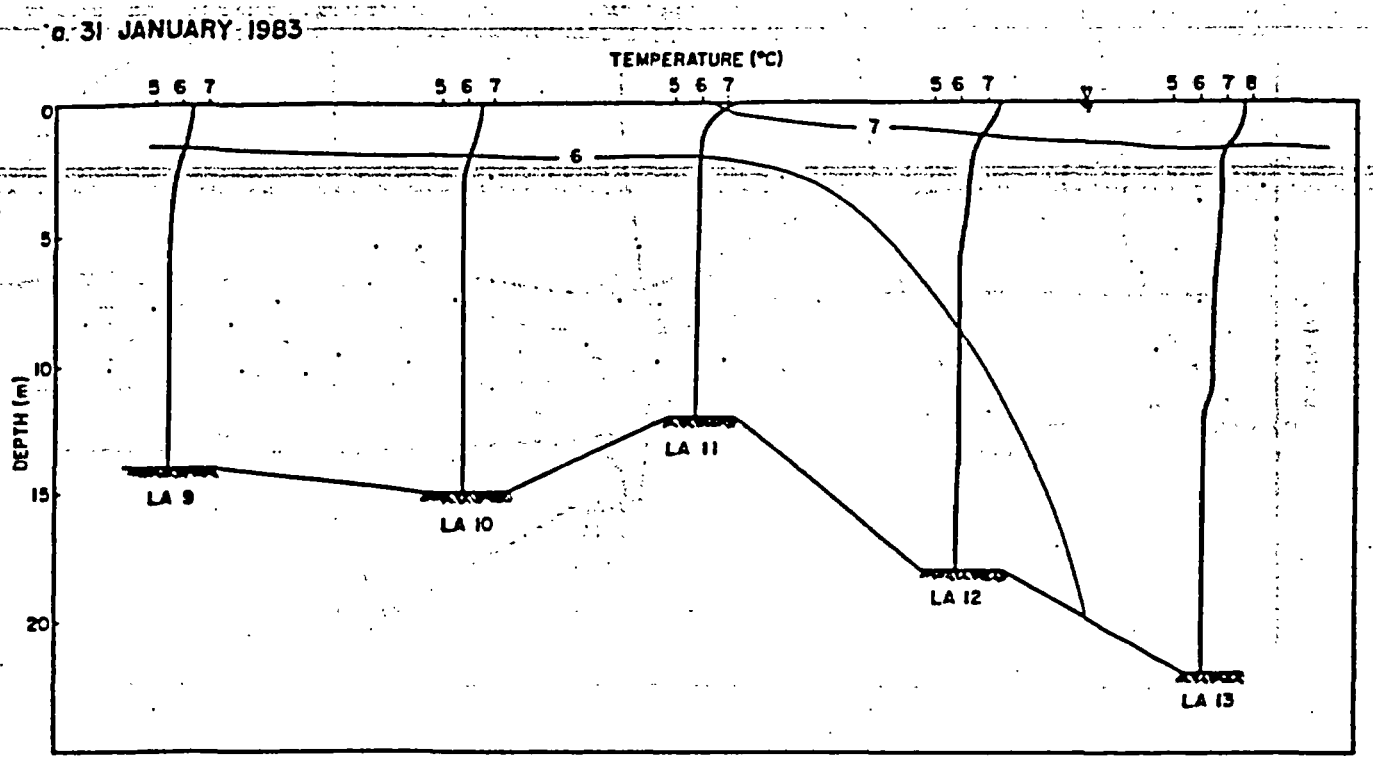


Figure 6.2h Predicted vs. Measured Vertical Temperature Profiles in Lake Anna

predicted vertical temperature profiles significantly, when compared with those shown in Wells et al. (1982). Secondly, calibration of Wells et al., although based on three years of data with primarily one-unit operation, holds up reasonably well for the fourth and fifth years with more two-unit operation.

Measured vertical temperature profiles and isotherms at five longitudinal locations (approximately equal-spaced) along the main lake are shown in Figure 6.3 for four days representing the four seasons in 1983. Slight horizontal variation is observed in the hypolimnion and some stratification occurs near the surface in winter (probably a transient daytime phenomenon caused by direct solar radiation). In general, however, the four plots collectively support the basic model assumptions of (1) a vertically well-mixed surface layer of constant thickness and horizontally-varying temperature distribution and (2) a vertically stratified subsurface layer of uniform horizontal temperature distribution.



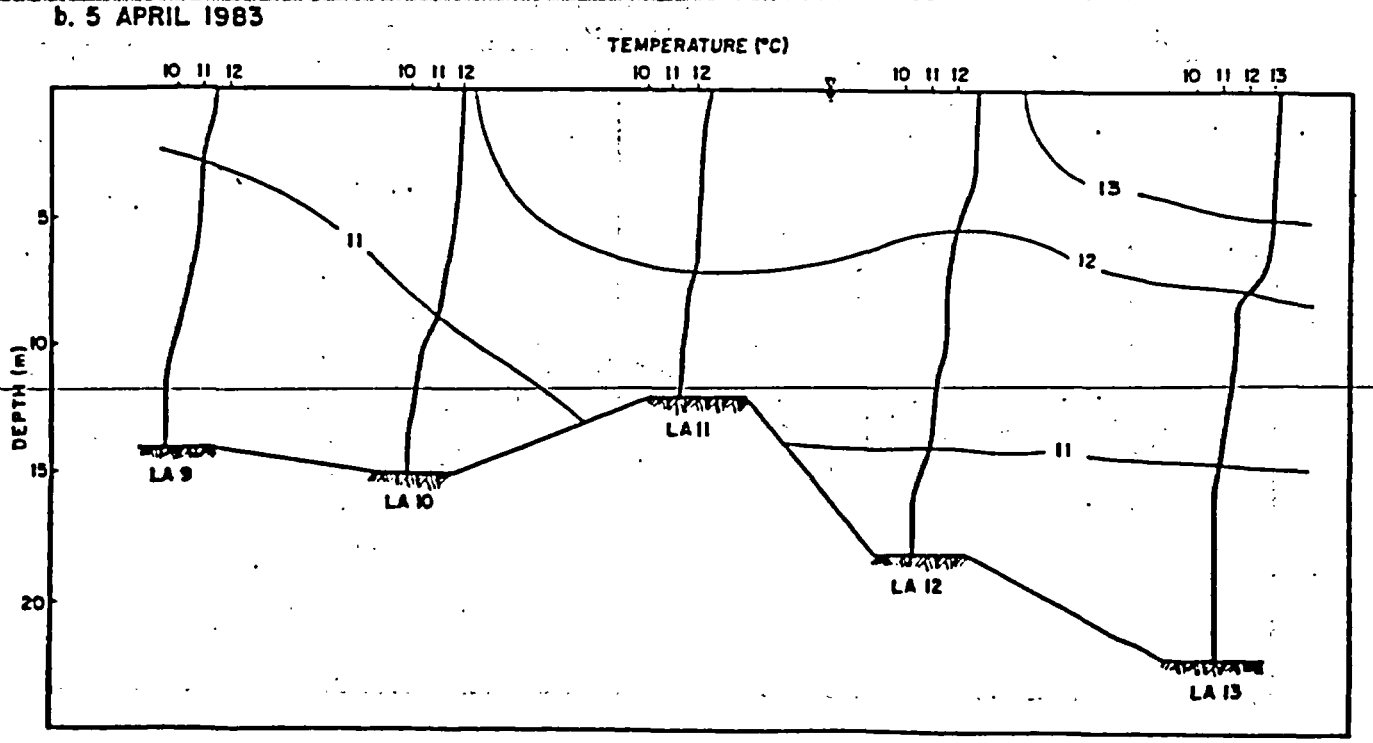
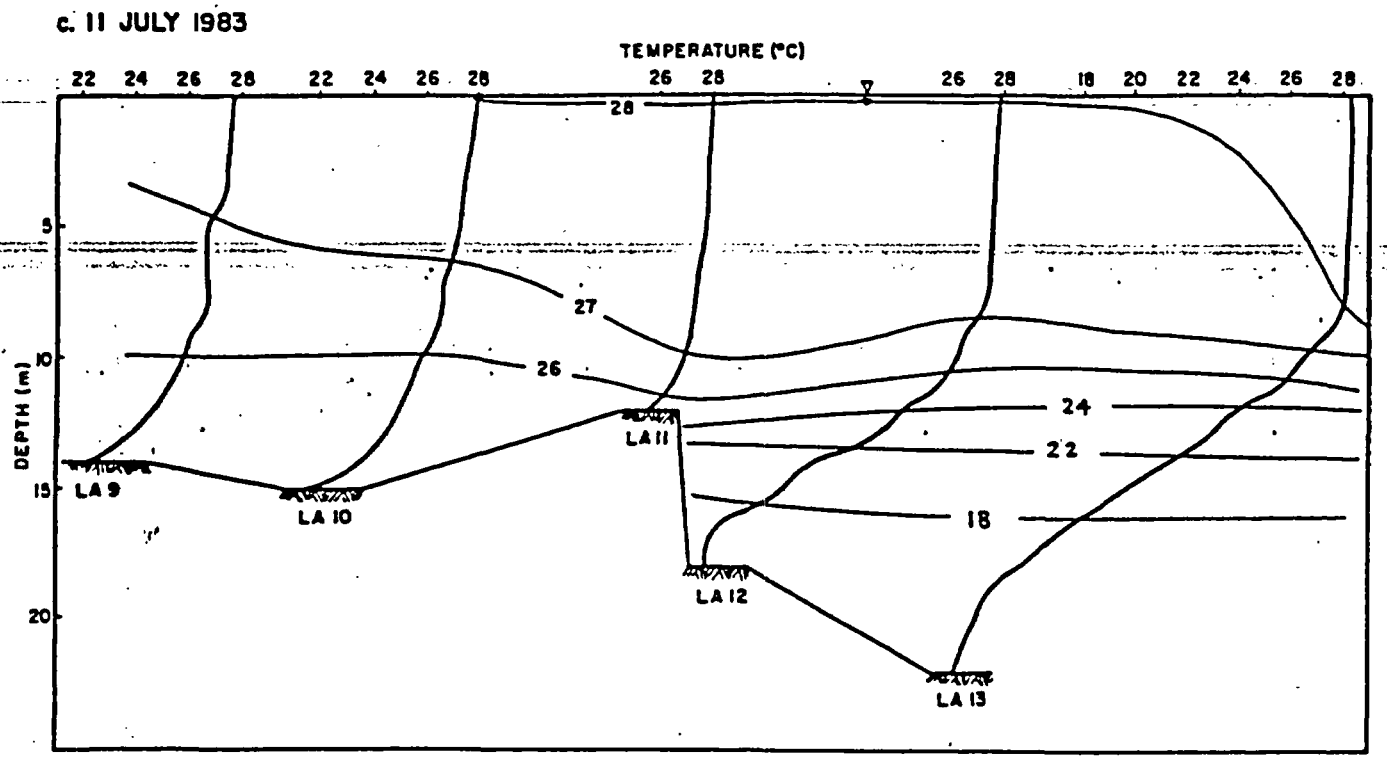


Figure 6.3 Measured Vertical Temperature Profiles and Isotherms in Lake Anna



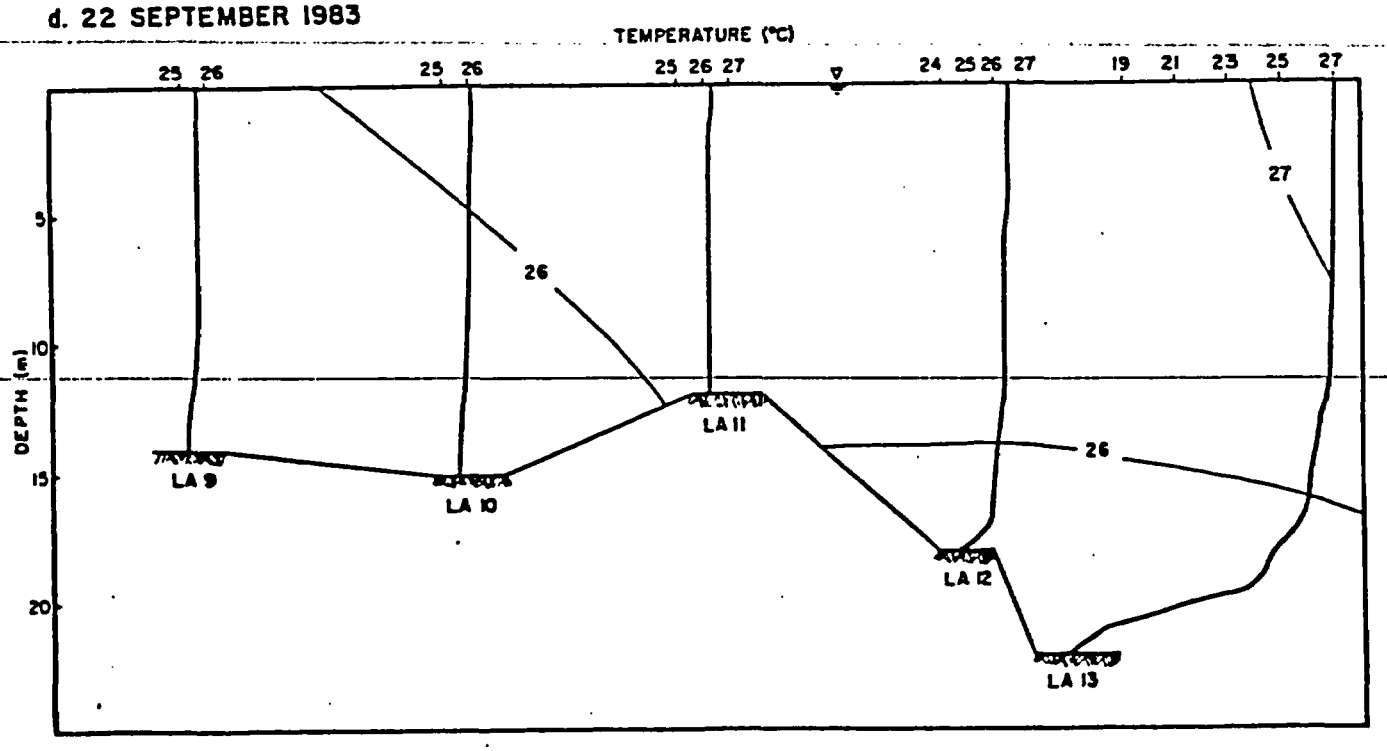


Figure 6.3 Measured Vertical Temperature Profiles and Isotherms in Lake Anna (cont'd)

7. SUMMARY

The North Anna Power Station is a two-unit power plant operated by Virginia Electric and Power Company (VEPCO). A complex cooling lake system was designed to dissipate the waste heat rejected by the plant. In order to monitor the impacts that the waste heat discharge has on the natural environment and also to assess the efficiency of the cooling lake system, a segmented mathematical model was previously developed to simulate the thermal structure of the North Anna lake system.

Since Unit 1 of North Anna Power Station came on line in summer of 1978 (follwed by Unit 2 in autumn of 1980), an extensive data collection network was established by VEPCO. Wells et al. (1982) calibrated the model using data from the first three years of operational data (1978 to 1981, primarily one-unit operation). Since then more measurements were available and an independent validation was conducted by comparing model predictions with data for the following two years (1981 to 1983, with more two-unit operation) using the same model calibrations. Continuous model input is now available for 62 months (26 July 1978 to 30 September 1983 inclusive, a total of 1893 days).

Surface temperature error analysis was made for four representative diagnostic control points over the cooling lake system. They are (1) at the discharge of Channel 1 into Pond 1 of the WHTF (designated DISCHARGE), (2) just upstream of DIKE III in the WHTF (DIKE III), (3) in the main lake near Burrus Point (BR. PT.), and (4) in the

main lake near plant intake (INTAKE). It was noticed that the model results for five years were similar to those for the first three years, as indicated by error statistics and observed periodicities of the raw errors. Moreover, error analysis indicated that the model showed a consistent performance even under "harsh" climatic and plant operation conditions, as provided by the experience of essentially two-unit operation and above-normal air temperatures during the summer of 1983. However, it was noticed that temperature rise across the plant condenser was consistently over-predicted by an average of about 0.7 °C over all five years. In addition, results of spectral analysis showed that the raw error at DIKE III was periodic at predominantly the annual frequency whereas the rest of the control points showed mild periodicities over a relatively wide range of frequencies.

Model recalibration aiming at improving the goodness-of-fit of the five years of measurement data at DISCHARGE and DIKE III was motivated by the results of the surface temperature error analysis. Two possible reasons for the transient errors characterized by annual periodicity of the raw error at DIKE III were identified, namely, errors associated with (1) the forcing function (surface heat transfer) and (2) system response of the model. Two candidates for each of the possible reasons were considered: (1a) atmospheric radiation, (1b) evaporation, (2a) spatially-averaged system response, and (2b) longitudinal system response. Among the four candidates, both (1a) and (2a) were ruled out. (1b) was considered possible since a moderate annual adjustment in the rate of evaporation over the WHTF virtually eliminated the periodicity of the raw error, but (2b) was found to be most plausible because of the strong sensitivity of the periodicity to input residence

time and because of the uncertainties in the actual residence time. The adopted approach wherein residence time was based on a fraction (75%) of the entire volume seemed more rational than that used previously and also resulted in a substantial reduction of the periodicity. However, it should be emphasized that there is still some uncertainty concerning the actual residence times which would require a dye study for further confirmation. Such a dye study would elucidate not only the effective residence time but also the residence time distribution within the WHTF which is needed if a more complete transient analysis is desired.

Steady state errors at DISCHARGE was characterized by an over-prediction of the surface temperature. Since predictions at DISCHARGE should have minimal dependence on the hydraulics of the model or on meteorological conditions, it seemed probable that the raw error was caused by uncertainty in the plant operation data. A sensitivity analysis of the raw error at DISCHARGE with respect to changes in the plant operation data was made, and it was noticed that the mean raw error was reduced substantially by a moderate (about 8%) reduction of the waste heat discharged into the WHTF. This adjustment of the plant operation data was confirmed by new data from VEPCO.

In all, two changes of the model were made as a result of recalibration: computation of residence times in the WHTF and adjustment of plant operation data. The recalibrated model was validated by comparing predictions with five years (1978 - 1983) of measurement data in terms of both surface temperatures and vertical temperature profiles. Overall mean errors for the four diagnostic control points ranged (in terms of magnitude) from a minimum of 0.01 °C

at INTAKE to a maximum of 0.16 °C at BR. PT. Standard deviations ranged from 0.80°C at INTAKE to 1.37°C at DIKE III. These values are in general less than those presented by Wells et al. (1982). In comparison with the large peak-to-peak annual variation in water surface temperature of about 23 °C, the mean errors and standard deviations are both acceptably small. Comparisons between measured and predicted vertical temperature profiles in the main lake indicated that the dynamic nature of the data both in space and time was also modelled satisfactorily.

In addition to overall averages, monthly averages of the surface temperature raw errors were also computed for each of the diagnostic control points. They ranged from a minimum of -0.69 °C for the month of September at DISCHARGE to a maximum of 0.89 °C for July at BR. PT. These monthly error statistics can be applied by VEPCO to increase the accuracy of future model predictions.

In conclusion, the recalibrated model was shown to be highly accurate in describing the thermal structure of the cooling lake system with one or two-unit operation of the power plant, and also under a wide-range of meteorological conditions, thus providing confidence for its use in a predictive mode.

APPENDIX

MONTHLY-AVERAGED CALCULATIONS OF DOWNWARD LONG-WAVE RADIATION UNDER CLEAR SKY GIVEN BY EIGHT DIFFERENT EMPIRICAL FORMULAE

MONTHLY AVERAGED CALCULATIONS OF DOWNWARD LONGWAVE RADIATION UNDER CLEAR SKY GIVEN BY EIGHT DIFFERENT EMPIRICAL FORMULAE •

MONTH	AIR TEMP.	REL. HUM.	LONGWAVE RADIATION GIVEN BY VARIOUS FORMULAE 1	2	3	4	5	6	7	8	MAX	MIN	MEAN	STAND DEV
8	23.8	0.85	369.6	356.7	366.2	385.2	379.8	400.6	405.4	410.6	410.6	356.7	384.3	18.5
9	19.9	0.81	334.0	330.4	338.6	352.0	353.9	366.0	375.9	372.5	375.9	330.4	352.9	16.4
10	12.4	0.74	273.6	282.3	289.9	291.8	305.9	303.4	321.8	310.0	321.8	273.6	297.3	14.8
11	8.9	0.80	254.5	262.2	271.1	271.7	287.7	282.5	304.2	291.1	304.2	254.5	278.1	15.3
12	4.7	0.71	229.1	240.5	250.1	242.2	263.6	251.8	275.7	266.3	275.7	229.1	252.8	14.4
1	1.6	0.67	211.5	224.7	240.1	222.3	246.9	231.1	256.9	248.3	256.9	211.5	235.2	14.4
2	-2.8	0.79	194.9	203.9	225.5	203.5	228.4	211.6	240.1	231.3	240.1	194.9	217.4	15.0
3	8.8	0.74	252.3	262.4	271.9	268.4	286.0	279.0	300.2	289.1	300.2	252.3	276.1	14.5
4	12.3	0.68	269.5	282.0	289.6	286.9	303.3	298.3	316.4	306.4	316.4	269.5	294.0	14.0
5	17.5	0.72	308.1	313.8	321.2	326.9	335.6	339.9	352.8	345.2	352.8	308.1	330.4	14.6
6	20.6	0.82	338.6	334.3	342.5	356.5	357.6	370.7	380.0	377.5	380.0	334.3	357.2	16.6
7	23.1	0.88	364.9	351.6	360.7	379.2	374.5	394.3	400.2	408.1	408.1	351.6	379.2	18.8
8	22.8	0.89	364.3	349.7	358.8	379.5	374.2	394.6	400.9	406.4	406.4	349.7	378.5	19.3
9	19.4	0.92	336.8	326.6	334.5	353.7	353.1	367.8	378.8	377.8	378.8	326.6	353.6	18.7
10	12.9	0.86	284.5	285.9	293.8	302.4	312.3	314.5	332.5	322.3	332.5	284.5	306.0	16.2
11	10.5	0.81	266.2	271.7	280.3	283.6	297.4	294.9	315.0	303.3	315.0	266.2	289.1	15.5
12	4.7	0.75	229.6	240.0	252.2	243.6	264.4	253.3	277.7	266.2	277.7	229.6	253.4	14.7
1	2.0	0.70	213.9	226.3	240.9	225.7	249.5	234.7	260.6	250.4	260.6	213.9	237.8	14.5
2	0.3	0.61	203.9	218.3	235.5	212.9	239.3	221.4	247.5	240.9	247.5	203.9	227.5	14.4
3	6.4	0.67	236.0	249.4	260.8	249.9	271.2	259.9	282.4	272.7	282.4	236.0	260.3	14.0
4	13.3	0.64	272.5	287.4	294.7	290.8	307.5	302.4	319.5	308.6	319.5	272.5	297.9	13.7
5	19.4	0.65	316.6	326.7	334.6	336.0	345.2	349.3	359.7	352.5	359.7	316.6	340.1	13.4
6	21.6	0.74	342.4	341.6	350.3	360.3	362.4	374.6	382.3	380.3	382.3	341.6	361.8	15.2
7	24.8	0.80	375.0	364.4	374.3	390.8	385.5	406.4	409.6	414.9	414.9	364.4	390.1	17.4
8	24.6	0.76	369.3	362.4	372.3	386.0	382.6	401.4	405.0	407.8	407.8	362.4	385.8	16.2
9	21.6	0.38	306.6	341.7	350.5	326.9	346.9	339.9	346.8	340.7	350.5	306.6	337.5	13.4
10	13.6	0.69	279.8	290.2	298.0	298.0	312.0	309.8	326.5	316.2	326.5	279.8	303.8	14.1
11	7.4	0.63	237.8	254.3	264.4	252.7	274.9	262.8	284.7	274.2	284.7	237.8	263.2	14.0
12	2.8	0.66	217.1	231.3	246.2	228.9	253.0	238.0	262.8	253.6	262.8	217.1	241.4	14.3
1	-0.9	0.63	198.6	212.6	231.6	207.2	234.0	215.5	242.3	235.1	242.3	198.6	222.1	14.7
2	4.6	0.64	226.4	240.1	253.3	238.6	261.5	248.1	271.6	263.4	271.6	226.4	250.4	14.0
3	6.1	0.61	231.3	247.7	258.9	244.6	267.9	254.4	277.0	268.1	277.0	231.3	256.2	13.9
4	14.8	0.54	276.3	296.9	304.5	294.2	312.9	305.9	320.9	312.2	320.9	276.3	303.0	12.9
5	17.0	0.42	279.5	310.8	318.4	297.9	320.3	309.7	322.1	315.1	322.1	279.5	309.2	13.3
6	24.2	0.52	341.8	359.7	369.4	361.8	370.7	376.2	380.1	375.6	380.1	341.8	366.9	11.5
7	25.1	0.76	374.2	366.8	376.9	390.6	386.4	406.1	408.8	412.8	412.8	366.8	390.3	16.2
8	22.0	0.81	350.6	343.8	352.5	367.9	366.7	382.5	389.9	390.0	390.0	343.8	368.0	17.0
9	20.3	0.71	329.4	332.8	341.1	348.2	353.3	362.1	371.4	366.3	371.4	329.4	350.6	14.5
10	12.0	0.75	271.6	280.3	288.1	289.6	303.9	301.1	319.7	308.3	319.7	271.6	295.3	14.8

11	8.2	0.66	245.1	259.2	269.0	260.5	280.7	270.8	292.1	281.7	292.1	245.1	269.9	14.0
12	1.8	0.76	215.0	225.3	240.3	227.2	249.9	236.3	262.5	251.6	262.5	215.0	238.5	14.7
1	-1.7	0.64	196.3	209.4	229.7	203.8	230.6	211.9	238.9	233.1	238.9	196.3	219.2	14.7
2	3.0	0.79	221.4	230.9	244.7	234.8	256.2	244.2	269.9	257.9	269.9	221.4	245.0	14.9
3	7.0	0.69	239.1	252.6	263.1	254.1	274.8	264.2	286.6	275.6	286.6	239.1	263.8	14.2
4	11.6	0.55	257.8	277.9	286.2	273.7	294.9	284.6	302.6	294.5	302.6	257.8	284.0	13.3
5	19.7	0.70	323.5	328.7	336.6	342.1	348.7	355.7	365.8	360.6	365.8	323.5	345.2	14.2
6	21.0	0.81	341.5	337.0	345.4	359.5	360.2	373.8	382.6	380.2	382.6	337.0	360.0	16.6
7	23.3	0.89	368.6	353.0	362.2	383.6	377.4	398.9	404.5	411.0	411.0	353.0	382.4	19.6
8	22.9	0.83	360.3	350.4	359.6	376.7	373.4	391.7	397.7	400.3	400.3	350.4	376.3	17.6
9	18.4	0.89	326.3	320.1	327.7	344.3	346.1	358.0	370.1	365.8	370.1	320.1	344.8	17.7
10	13.0	0.84	284.6	286.6	294.5	302.4	312.6	314.5	332.3	322.5	332.3	284.6	306.3	16.0
11	9.2	0.83	259.6	264.5	273.9	276.3	290.8	287.3	308.6	296.8	308.6	259.6	282.2	15.6
12	6.0	0.88	244.5	248.2	260.5	259.5	275.5	269.9	293.4	281.8	293.4	244.5	266.6	15.6
1	2.3	0.75	217.2	227.6	242.0	229.7	252.1	238.9	264.8	253.8	264.8	217.2	240.8	14.7
2	3.0	0.75	220.5	231.4	245.2	233.5	255.7	242.8	268.3	257.0	268.3	220.5	244.3	14.7
3	8.4	0.73	248.4	259.5	268.9	264.4	282.9	274.9	296.5	285.3	296.5	248.4	272.6	14.5
4	11.9	0.71	268.3	279.9	288.1	285.8	301.9	297.2	315.6	304.9	315.6	268.3	292.7	14.2
5	17.5	0.68	305.4	314.3	321.8	324.1	334.4	337.0	349.6	342.1	349.6	305.4	328.6	13.9
6	22.8	0.73	350.9	349.7	358.9	368.9	369.7	383.6	389.7	388.3	389.7	349.7	370.0	15.0
7	25.7	0.67	370.0	370.5	380.9	387.9	386.9	403.3	405.2	406.1	406.1	370.0	388.9	13.8
8	25.5	0.67	368.6	369.1	379.4	386.6	385.8	402.0	404.1	404.6	404.6	368.6	387.5	13.9
9	21.0	0.66	331.1	338.3	347.1	350.2	356.4	364.1	372.2	366.7	372.2	331.1	353.3	13.4

- TEMPERATURE IN DEGREES CELCIUS
  RADIATION IN WATTS PER SQUARE METER
  DATA START AUGUST 1978
  FORMULAE USED TO CALCULATE LONG-WAVE RADIATION:
  - BRUNT(1932) 1
  - SWINBANK(1963) 2
  - IDSO-JACKSON(1969) 3
  - BRUTSAERT 1(1975) 4
  - SATTERLUND(1979) 5
  - BRUTSAERT 2(1981) 6
  - IDSO 1(1981) 7
  - IDSO 2(1981) 8

LIST OF REFERENCES

Adams, E.E., The Transient Response of Cooling Ponds, Water Resources Research, Vol. 18, No. 5, pp. 1469-1478, Oct., 1982.

Adams, E.E. and Koussis, A.D., Transient Analysis for Shallow Cooling Ponds, J. of the Energy Division, ASCE, Vol. 106, No. EY2, pp. 141-153, Oct., 1980.

Adams, E.E. and Wells, S., Field Measurements on Side Arms of Lake Anna, VA., J. of Hydraulic Engineering, ASCE, Vol. 110, No. HY6, June, 1984.

Bolz, H.M., Die Abhängigkeit der infraroten Gegenstrahlung von der Bewölkung, Z. Meteorol. 3, pp. 201-203, 1949.

Brocard, D., Jirka, G. and Harleman, D., A Model for the Convective Circulation in Side Arms of Cooling Lakes, T.R. No. 223, R.M. Parsons Laboratory, Department of Civil Enginering, M.I.T., 1977.

Brutsaert, W., Evaporation into the Atmosphere: Theory, History and Applications, Reidel Publishing Co., Dordrecht, Holland, 1982.

Grenander, U. and Rosenblatt, M., Statistical Analysis of Stationary Time Series, Wiley & Sons, Inc., New York, 1957.

Hatfield, J.L., Reginato, R.J. and Idso, S.B., Comparison of Long-Wave Radiation Calculation Methods Over the United States, Water Resources Research, Vol. 19, No. 1, pp. 285-288, Feb., 1983.

Idso, S.B. and Jackson, R.D., Thermal Radiation from the Atmopshere, J. of Geophysical Research, Vol. 74, No. 23, Oct., 1969.

Jirka, G.H., Abraham, G. and Harleman, D.R.F., An Assessment of Techniques for Hydrothermal Predictions, T.R. No. 203, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1975.

Jirka, G.H., Adams, E.E., and Stolzenbach, K.D., Buoyant Surface Jets, J. of Hydraulics Div., ASCE, Vol. 107, No. HY11, Nov., 1981.

Jirka, G.H., Brocard, D., Octavio, K., Watanabe, M. and Harleman, D., Analysis of Cooling Effectiveness and Transient Long-Term Simulations of a Cooling Lake with Application to the North Anna Power Station, T.R. No. 232, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1977.

Kimball, Idso and Aase, A Model of Thermal Radiation from Partly Cloudy and Overcast Skies, Water Resources Research, Vol. 18, No. 4, pp. 931-936, 1982.

Local Climatological Data, Richmond, VA, publication of the NOAA, U.S. Department of Commerce, by the National Climatic Center, Asheville, North Carolina.

Octavio, K., Jirka, G. and Harleman, D., Vertical Heat Transport Mechanisms in Lakes and Reservoirs, T.R. No. 227, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1977.

Ryan, P. and Harleman, D., An Analytical and Experimental Study of Transient Cooling Pond Behavior, T.R. No. 161, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1973.

Salhotra, A., Adams, E. and Harleman, D., Evaporation and Stratification Studies for the Dead Sea, Progress Report No. 2, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1983.

Sellers, W., Physical Climatology, U. of Chicago Press, Chicago and London, 1965.

Stolzenbach, K. and Harleman, D., An Analytical and Experimental Investigation of Surface Discharges of Heated Water, T.R. No. 135, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1971.

Swinbank, W., Longwave Radiation from Clear Skies, Quarterly J. of the Royal Meteorological Society of London, Vol. 89, July 1963.

Thackston, E., Effect of Geophysical Location on Performance of Recirculating Cooling Ponds, EPA Report No. 660/2-74-085, Nov., 1974.

Watanabe, M., Harleman, D. and Connor, J., Finite Element Model for Transient Two-Layer Cooling Pond Behavior, T.R. No. 202, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1975.

Wells, S., Adams, E. and Harleman, D., Calibration and Verification of the Cooling Lake Model for North Anna Power Station, T.R. No. 272, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., 1982.

Wunderlich, W., Heat and Mass Transfer Between a Water Surface and the Atmosphere, Lab. Report No. 14, T.V.A. Engineering Laboratory, Norris, Tennessee.

LIST OF FIGURES

1.1	Geographic Location of North Anna Power Station, Virginia	11
1.2	North Anna Cooling Lake System	12
1.3	Discharge Canal from the Power Plant	14
3.1	Schematization of the North Anna Cooling Lake System Used in the Segmented Model	20
3.2	Schematization of Convective Circulation in a Dead-End Side Arm	23
3.3	Flow Configuration of a Pond without Side Arm	27
3.4	Flow Configurations of a Pond with Two Side Arms	28
3.5	Plan View of Submerged Jet Discharge at Dike III	31
3.6	Cross Section of Dike III Constriction	32
3.7	Schematization of Main Lake Model	38
4.1	Schematization of the Locations of the Four Diagnostic Control Points	42
4.2	Measured vs. Predicted Surface Temperatures at Four Diagnostic Control Points	43
4.3	Raw Errors at Four Diagnostic Control Points	47
4.4	Delta Errors at Four Diagnostic Control Points	53
4.5	Monthly Averages of Raw Errors at Four Diagnostic Control Points	60
4.6	Power Spectra at Four Diagnostic Control Points	63
4.7	Number of Nuclear Units Operating During Simulation Period	67
4.8	Number of Circulating Pumps Operating During Simulation Period	68
5.1	Schematic of Relationships between Measurements, Predictions and Raw Errors at DIKE III	76
5.2	Schematic of Relationship between Raw Errors and Equilibrium Temperature at DIKE III	79
5.3	Monthly-Averaged Values of Downward Long-Wave Radiation Under Clear Sky Given by Eight Different Formulae	83
5.4	Daily Estimates of Long-Wave Radiation Given by Brutsaert 1 and Idso-Jackson	85
5.5	Raw Error at DIKE III ($\alpha = 0.75$)	89
5.6	Raw Error at DIKE III ($\alpha = 0.75 + 0.25 \sin(\omega t - 0.4576)$)	92

5.7 Raw Error at DIKE III (α = 0.75; lag time over WHTF based on 75% of whole volume) 100

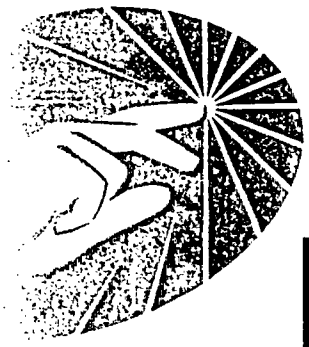
6.1 Raw Errors at Four Diagnostic Control Points (Recalibrated Model) 106

6.2 Predicted vs. Measured Vertical Temperature Profiles in Lake Anna 113

6.3 Measured Vertical Temperature Profiles and Isotherms in Lake Anna 118

LIST OF TABLES

4.1	Interpretation of Delta Errors	51
4.2	Statistics of Raw Errors at Four Diagnostic Control Points (1978-1983)	57
4.3	Statistics of Raw Errors at Four Diagnostic Control Points (1978-1981)	57
4.4	Monthly Averages of Raw Errors at Four Diagnostic Control Points	59
4.5	Length and Starting Day of Time Series Used in Computing the Fourier Transforms	65
5.1	Fourier Decomposition of Time Series	75
5.2	Summary of Long-Wave Radiation Calculation Methods (after Hatfield, 1983)	81
5.3	Harmonic Analysis of Raw Error (DIKE III) Time Series	90
5.4	Sensitivity Test of Raw Errors at DIKE III with Respect to Different Fixed Residence Times (Runs 1 - 7)	97
5.5	Mean Raw Errors for Runs 0, 1, 2 and 3	104
6.1	Comparison of Statistics of Raw Errors at Four Diagnostic Control Points between Recalibrated Model and Model in Wells et al. (Five-Year Simulation)	110
6.2	Monthly Averages of Raw Errors at Four Diagnostic Control Points (Recalibrated Model)	112



Dominion℠

USER'S MANUAL FOR LAKE ANNA COOLING POND MODEL

by

Edmond K. Ho

Scott A. Wells

E. Eric Adams

Ralph M. Parsons Laboratory for
Water Resources and Hydrodynamics

Department of Civil Engineering

Massachusetts Institute of Technology

prepared for
Virginia Electric and Power Co.
Richmond, Virginia

January 1983

TABLE OF CONTENTS

	PAGE
1. INTRODUCTION	1
2. INPUT DATA FILES	1
2.1 Meteorological Data	2
2.2 Plant Operation Data	2
2.3 Program Input Data	4
3. RUNNING THE PROGRAM	14
4. OUTPUT DATA FILES	15
4.1 North Anna Lake System Data	15
4.2 Main Lake Data	20
4.3 WHTF Data	20
APPENDIX 1. HISTORICAL "SYNTHETIC" INPUT DATA	23
A. Historical Meteorological Input Data	23
B. Historical Plant Operation Input Data	23
APPENDIX 2. DIAGNOSTIC EVALUATION OF RESULTS	24
A. To Analyse Errors at Four Control Points	24
B. To Compute Delta Errors Between Four Control Points	24
APPENDIX 3. PROCESSING NRC ENDECO DATA	26
APPENDIX 4. INDEX OF VARIABLES FOR PROGRAM INPUT DATA	28
APPENDIX 5. SELECTION FLAGS (CARD GROUP 41) USED IN FULLPOND FORTRAN	32
Attachment 1 (Main Program)	33
Attachment 1 (Subroutines: WLOSS, GETMET, LEAP, DATIME, DAXIME, TIMDAY, NDY, ZERO)	41
Attachment 1 (Subroutines: QQIN, QOUT, TTIN, TOUT, XINTKP, AVER, SPEED, COLDCK)	44

PAGE

Attachment 1 (Subroutine: FTEQ) 50

Attachment 1 (Subroutine: FKKK) 52

Attachment 1 (Subroutines: SURLAY, OLDSID) 53

Attachment 1 (Subroutines: WHTF, SIDARM, SIDE) 58

FW HTFSUB Attachment 1 (Subroutines: DSWHTF, DSPARM, PRWHTF, RDWHTF) 62

Attachment 2 (File: FULLPOND EXEC) 65

Attachment 3 (File: MET1305 DATA) 66

Attachment 4 (File: PLNT1305 DATA) 74

Attachment 5 (File: IN1305 DATA) 77

Attachment 6 (North Anna Cooling Pond Simulation – July 26, '78 through Feb. 19, '82) 78

Attachment 7 (File: RUN1305 DATA) 109

Attachment 8 (File: ML1305 DATA) 121

Attachment 9 (File: SAA1305 DATA) 133

Attachment 10 (File: ERR FORTRAN) 145

Attachment 11 (File: ERR EXEC) 146

Attachment 12 (File: SUBTR FORTRAN) 147

Attachment 13 (File: SUBTR EXEC) 148

Attachment 14 (File: AVERNRC FORTRAN) 149

Attachment 15 (File; AVERNRC EXEC) 150

LIST OF FIGURES

Fig. 4.1 Schematization of the North Anna Cooling System used in the Segmented Model and the locations of the four control points for diagnostic purpose 16

Fig. 4.2 Schematization of the Main Lake used in the Segmented Model 17

1. INTRODUCTION

This User's Manual is issued as a supplement to Technical Report No. 232, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., entitled "Analysis of Coding Effectiveness and Transient Long-Term Simulations of a Cooling Lake with application to North Anna Power Station" and to Technical Report No. 272, R.M. Parsons Laboratory, Department of Civil Engineering, M.I.T., entitled "Calibration and Verification of the Cooling Lake Model for North Anna Power Station". The mathematical model simulates thermal structure within the Lake Anna system. For a detailed account of the scope as well as the development of the model, the reader is referred to TR's 232 and 272.

The model contains a main program (filename FULLPOND) together with 27 subroutines and function sub-programs (Attachment 1), totalling 3045 lines as of December, 1982. The program is written in FORTRAN. This program and all other programs mentioned in this document have been run using the Conversational Monitor System (CMS) on M.I.T.'s IBM VM/370 computer system. Several references to CMS are made in this report. However, the programs should be able to run on any other mainframe system.

The program requires three input data files and generates three output data files, all of which are explained in the following sections.

2. INPUT DATA FILES

Three separate data files are used as input; these include meteorological data, plant operation data, and program input data. The program can be run with "recent" meteorological data and plant operation data (recorded since

plant start-up in July 1978), as well as historical "synthetic" meteorological and plant operation data. (See Appendix 1.)

2.1 Meteorological Data

The recent meteorlogical data are stored under the filename of METXXXX where XXXX (4 digits) designates the number of days of daily-averaged data since (and including) July 26, 1978. The data file contains five groups of meteorological data of equal size, with each group of data starting a new line in the file. Data parameters NMET and DTMET (Card groups 23 and 24, Table 2.3) must correspond to the values in METXXXX DATA. IOMET (Card group 25, Table 2.3) specifies the I/O unit from which this file is to be read. The format of this data file is 8F10.2 for every line. Table 2.1 gives a description of the meteorological parameters and the names of the corresponding input variables.

A sample input data file (MET1305) is included as Attachment 3.

2.2 Plant Operation Data

The recent plant operation data are stored under the filename of PLNTXXXX where XXXX (4 digits) designates the number of days of daily-averaged data since (and including) July 26, 1978. The data file contains two groups of plant operation data of equal size, with each group of data starting a new line in the file. Data parameters NDIS and DTDIS (Card groups 23 and 24, Table 2.3) must correspond to the values in PLNTXXXX DATA. IODIS (Card group 25, Table 2.3) specifies the I/O unit from which this file is to be read. Values for the second group of plant operation

Table 2.1 Meteorological Input Data

Card Group	Input Variable	Size	Description
1	TA(I)	NMET	Air temperatures at time intervals of DTMET (^{o}F)
2	SIGH(I)	NMET	Relative humidities at time intervals of DTMET (decimal)
3	WIND(I)	NMET	Wind speeds at 2 meters elevation at time intervals of DTMET (mph)
4	CLOUD(I)	NMET	Cloud cover at time intervals of DTMET (decimal)
5	FIN(I)	NMET	Values of solar radiation at time intervals of DTMET (Btu ft^{-2} day^{-1})

data depend on the value of ISELCT(8) (Card group 40, Table 2.3).

$$\text{ISELCT(8)} \begin{cases} = 0 \Rightarrow \text{once-through cooling simulation} \\ = 1 \Rightarrow \text{recirculating cooling simulation} \end{cases}$$

The format of this data file is 8F10.2 for every line. Table 2.2 gives a description of each of the two data groups and the names of the corresponding input variables.

A sample data file (PLNT1305) is included as Attachment 4.

2.3 Program Input Data

The program input data are stored under the filename of INXXXX where XXXX (4 digits) designates the number of days of model simulation since (and including) July 26, 1978. This input data file specifies various initial parameters, I/O units, data size values, and debugging/selection flag values. The data file consists of 41 data groups. Table 2.3 gives the format and description of each of the data groups as shown in the sample data given in Attachment 5. An index of the input variables and the corresponding card groups and the common blocks to which the input variables belong is given in Appendix 4. Parameters that usually need to be changed for a new run of the program are summarized in Table 2.4; they are also included in the attached sample data (Attachment 5).

Table 2.2 Plant Operation Data

Card Group	Input Variable	Size	Description
1	QPLNT(I)	NDIS	Circulating flow rate of plant condenser at time intervals of DTDIS (cfs)
2	TPLNT(I)	NDIS	For ISELCT(8) = 0: Circulating water temperature at plant discharge (°F) For ISELCT(8) = 1: Rise in circulating water temperature (ΔT) across plant condenser (°F)

Table 2.3 Program Input Data

Card Group	Input Variable	Format	Description
1	WH	20A4	80 character title for run
2	MMO	I5	Month..
	MDAY	I5	Day Date of first simulation day
	MYR	I5	Year
3	JM	I5	Initial number of layers in main pond
	JMP	I5	Maximum possible number of layers
	KITP	I5	= 2 for North Anna Lake System
	KSUR	I5	= 1 for constant water surface elevation = 2 for time variable water surface elevation
	KOH	I5	= 1 to use Koh's selective withdrawal eqn. = 2 to use Kao's selective withdrawal eqn.
	KQ	I5	= 1 for simulation with in- and outflow = 2 for simulation with neither in- nor outflow
	KLOSS	I5	= 1 for Rohwer evaporation eqn. = 2 for Idso & Jackson evaporation eqn.
	NPRINT	I5	Not used in North Anna Program
	KAREA	I5	= 2 for North Anna Lake System
	KMIX	I5	= 1 for no entrance mixing at Dike III = 2 for entrance mixing at Dike III
	MIXED	I5	Number of surface layers participating in entrance-mixing

Table 2.3 cont.

Card Group	Input Variable	Format	Description
	ITZERO	I5	= 0 for isothermal initial reservoir temperatur = 1 to input an initial temperature - depth profile
4	YSUR	F10.5	Initial water surface elevation (ft)
	DY	F10.5	Depth increment (ft)
	DT	F10.5	Time step (day)
	TSTOP	F10.5	Ending time of simulation (day)
	TZERO	F10.5	Initial isothermal reservoir temperature (^{o}F)
	PRAIR	F10.5	Air pressure (= 760 mm Hg)
	EFF	F10.5	Not used
5	ETA	F10.5	Radiation absorption coefficient
	BETA	F10.5	Fraction of solar radiation absorbed at sur
	RHO	F10.5	Density of water (62.4 lb ft^{-3})
	HCAP	F10.5	Specific heat of water (1 Btu lb^{-1} $^{o}F^{-1}$)
	RMIX	F10.5	Mixing ratio in eqn. 2.15 (=1.0)
6	NDD	I5	Number of values of diffusion coefficient to be read (=2)
	NIN	I5	Number of inflows (including power station discharge)
	NOUT	I5	Number of outflows (including power station intake)
	NSURF	I5	Number of surface elevations to be read
	NDTEMP	I5	Not used
7	NQI	16I5	Number of intake/discharge values to be read

Table 2.3 cont.

Card Group	Input Variable	Format	Description
8	DSURF	F10.5	Time interval between input values of SURF (card 20)
	DTDD	F10.5	Time interval between input values of DD (card 11)
	DTTMP	F10.5	Not used
9	DTQI	8F10.5	Time intervals (day) between input values of QI for each intake. Read but not used.
10	T	16F10.5	Temperatures of depth-temperature profile, beginning with bottom layer (^{o}F) (N.B. Number of values = JM)
11	DD(I)	F10.5	Parameters used to compute a time-variable diffusion coefficient.
	DD3	F10.5	
	DD4	F10.5	
12	DDCOEF	F10.0	=00.0 Parameters used to compute a time-variable diffusion coefficient
	XCOEF	F10.0	= 2.0
13	LOUT	16I5	Layer number(s) corresponding to the outlet elevation(s)
14	ELOUT	8F10.5	Outlet elevations (ft msl)
15	NARSF	I5	Number of area segments in main lake
16	NAA	I5	Number of areas to be read on card 18
	NXXL	I5	Number of lengths to be read on card 19
17	DAA	F10.5	Vertical distance increment between AA values (card 18) (ft)
	DXXL	F10.5	Vertical distance increment between XXL values (card 19) (ft)

Table 2.3 cont.

Card Group	Input Variable	Format	Description
	AAB	F10.5	Elevation corresponding to first (smaller) AA value (ft)
	XXLB	F10.5	Elevation corresponding to first (lowest) XXL value (ft)
	XWIND	F10.5	Not used
18	AA	8F10.5	Values of horizontal reservoir areas (acres) corresponding to elevations determined from AAB and DAA (card 17)
19	XXL	10F8.1	Values of horizontal reservoir length (ft) corresponding to elevations determined from XXLB and DXXL (card 17)
20	SURF	8F10.5	Values of surface elevation (ft)
21	NREACH	16I5	Number of reaches (ponds) in WHTF (normally set to be 3; set to be 4 when exploring rerouting)
22	SAFACT	F10.0	Side arm flow factor (it represents the fractional reduction factor to be applied to computed side arm flows in WHTF; no adjustment if 0.0)
	WGT	F10.0	Weighting factor for side arm iteration calculation (= 0.5)
	EFACT	F10.0	Evaporation factor (multiplicative factor which adjusts computer evaporative heat flux; default = 1.0)
23	NMET	I5	Number of values of meteorological data to be read (see Table 2.1)
	NDIS	I5	Number of values of plant operation data to be read (see Table 2.2)

Table 2.3 cont.

Card Group	Input Variable	Format	Description
24	DTMET	F10.0	Time intervals of meteorological data input (day
	DTDIS	F10.0	Time intervals of plant operation data input (da
25	IOMET	I5	I/O unit from which meteorological data are to be read
	IODIS	I5	I/O unit from which plant data are to be read
	ISMET	I5	Number of time intervals, DT, over which meteorological data are to be "smoothed" (i.e. moving averages constructed from data over ISMET periods)
	ISDIS	I5	Number of time intervals for plant operation data smoothing (N.B. if ISMET and ISDIS are left blank or equal to 0 no smoothing is done)
26	NCH	I5	Number of segments in discharge channel
	NCB	I5	Grid point at bottom of discharge channel
27	XCH	F10.0	Length of discharge channel (ft)
	WCH	F10.0	Average width of discharge channel (ft)
	DCH	F10.0	Average depth of discharge channel (ft)
	ACH	F10.0	Area of discharge channel (acre)
	DCHX	F10.0	Actual depth of discharge channel (ft)
	WCHX	F10.0	Actual width of discharge channel (ft)
	PLUMF	F10.0	Plume sector (1 for full plume, 0.5 for half plume)
	FRIFAC	F10.0	Friction factor of discharge channel
28	ZH(I)	F10.0	Entrance discharge channel depth (ft) for pond I
	ZB(I)	F10.0	Entrance discharge channel width (ft) for pond I

Table 2.3 cont.

Card Group	Input Variable	Format	Description
	ZBR(I)	F10.0	Reach width for pond I
	ZA1(I)	F10.0	Surface area before side arm 1 (acre) for pond I
	ZA2(I)	F10.0	Surface area before side arm 2 (acre) for pond I
	ZA3(I)	F10.0	Surface area after side arm 2 (acre) for pond I
	ZHM(I)	F10.0	Depth of pond at discharge channel (ft) entrance to pond I
29	NSA(I)	I5	Number of side arms
	LAGFIX(I)	I5	Travel time of circulating flow through reach I (time steps = days)
30	ZASA1(I)	F10.0	Area of side arm 1 (acre) of pond I
	ZASA2(I)	F10.0	Area of side arm 2 (acre) of pond I
	BSA1(I)	F10.0	Width of side arm 1 (ft) of pond I
	BSA2(I)	F10.0	Width of side arm 2 (ft) of pond I
	HSA1(I)	F10.0	Depth of side arm 1 (ft) of pond I
	HSA2(I)	F10.0	Depth of side arm 2 (ft) of pond I
	XLSA1(I)	F10.0	Length of side arm 1 (ft) of pond I
	XLSA2(I)	F10.0	Length of side arm 2 (ft) of pond I
31	HØØ	F10.5	Dike III jet depth (ft)
	BØØ	F10.5	Dike III jet width (ft)
32	RTW	F10.2	Top width of triangular control section downstream of Dike III (ft)
	RHØ	F10.2	Depth of triangular control section (ft)
	HBOT	F10.2	Bottom depth for jet interference calculation; inference is not computed if HBOT = Ø (ft)

Table 2.3 cont.

Card Group	Input Variable	Format	Description
33	KSA	I5	Number of side arms in main lake
34	XSAR(I)	8F10.2	Area of side arm I; I = 1 to KSA (acre)
35	XAREA(I)	8F10.2	Pond area separating side arms; for KSA = 1 pond area before the side arm (acre)
36	XH(I)	8F10.2	Entrance depth of side arm I (ft)
37	XB(I)	8F10.2	Average width of side arm I (ft)
38	ISTEQ	I5	Number of days for experimental filter to go ba
	HHH	F10.2	Value of depth for exponential filter (= 18 ft)
	KKK	I5	Value of K for exponential filter
	ISLCT(1to3)	3I5	Flags to input fixed HHH, KKK, or ISTEQ in that order; it will input fixed value if equals 0
39	IPDAT	10(I2, 1X, I2, 1X, I2)	Dates for detailed output; format is mo/da/yr e.g. 06/23/79. Detailed output will be printed on these 10 simulation days
	IPDAT	10(I2,1X,I2,1X,I2)	same as above
	IPDAT	10(I2,1X,I2,1X,I2)	same as above
	IPDAT	10(I2,1x,I2,1X,I2)	same as above
40	IDEBUG	16I5	Flags to turn on debugging output; normally leave card blank
41	ISELCT	16I5	Selection flags used to select various output and operation options (see the keys given in Appendix 5)

Table 2.4 Program Input Variables which usually need to be changed

Input Variable	Card Group
WH	1
TSTOP	4
NMET	23
NDIS	23
IPDAT	39

3. RUNNING THE PROGRAM

The program is run by using FULLPOND EXEC. A sample of the exec file is given as Attachment 2. Before every run of the program, both the input and the output files may have to be redefined*. For instance, PLNT1305 DATA C should be changed to PLNT1400 DATA B if a new data file PLNT1400 is used and the plant operation data are stored on disk B. In any case, changes are usually restricted to the following FILEDEF statements:

```
FILEDEF 5
FILEDEF 30
FILEDEF 31
FILEDEF 50
FILEDEF 51
FILEDEF 60
```

Also, it is note-worthy that the program input data (INXXXX DATA) is always defined to I/O unit 5.

Since considerable storage capacity is required in running the program, one has to increase the storage capacity of the virtual machine. This can be done on CMS by typing the statement:

```
CP DEFINE STORAGE AS 1000K
```

This command will terminate CMS; to reload, one can type

```
IPL CMS
```

The following commands are used to run the program

```
FULLPOND PR      or      FULLPOND TERM
```

*All data files must have reference to an I/O unit number defined via a FILEDEF statement in the EXEC file. In FULLPOND EXEC and all other EXEC files mentioned in this document, the data files have to be redefined as appropriate.

depending, respectively, on whether an output to the high-speed printer or to the terminal is desired. A hard-copy output of the sample run is given as Attachment 6.

4. OUTPUT DATA FILES

Three output data files, suitable as input to further computer analysis, are generated along with a hard-copy output from the main program (Attachment 6). The hard-copy output is described in this section; the other three output files are described in the following sections. The hard-copy output gives 15 columns of predicted temperature data for each day during the simulation period. In addition, detailed output for each date specified in Card group 39 (Table 2.3) is furnished. Table 4.1 gives a brief description of the data in the hard-copy output. See also Figs 4.1 and 4.2 which describe the horizontal and vertical locations of output temperature data. The write statement corresponding to this output is line no. FUL07100 in Attachment 1.

4.1 North Anna Lake System Data

This file of temperature data at selected locations of the North Anna Lake System (See Figs 4.1 and 4.2) is usually stored under the filename of RUNXXXX. A sample output (RUN1305) is included as Attachment 7. This output data file is primarily used for diagnostic evaluation of results (see Appendix 2). There are 13 columns of data in this file; an explanation is given in Table 4.2.

Table 4.1 Summary of "Hard Copy" Output

Column	Output Variable	Explanation*
1	MMO/MDAY/MYR	Date
2	TINIT	Temperature at plant discharge (°F)
3	TFIN(N,1)	Surface temperature at end of reach 1 (°F)
4	QSA1(2)	Flow rate of side arm 1 of reach 2 (cfs)
5	QSA2(2)	Flow rate of side arm 2 of reach 2 (cfs)
6	TEND(2)	Surface temperature at end of reach 2 (°F)
7	TEND(3)	Surface temperature at end of reach 3 (°F)
8	TSUR(1,1)	Surface temperature at (1,1) of main lake; TSUR (1,1) = THL (1,1) (°F)
9	THL(9,1)	Temperature at (9,1) of heat loss region of main lake (°F)
10	THL(40,1)	Temperature at (40,1) of heat loss region of main lake (°F)
11	XQCFS	Side arm flow rate of main lake (cfs)
12	TRSA(1)	Surface temperature of return side arm flow of main lake (°F)
13	TEQ	Filtered equilibrium temperature (°F) (see p. 100, TR272)
14	DELRHE	Density difference (g m^{-3}) as calculated between THL (40,1) and T(1,1) (See Fig 4.2)
15	DDRI	Eddy diffusivity (m^2 day^{-1}) in hypolimnion of main lake

*also see Fig 4.1 and Fig 4.2

Table 4.2 Summary of Lake Anna System Output

Column	Output Variable	Explanation*
1	MMO/MDAY/MYR	Date
2	TINIT	Temperature at plant discharge (°F)
3	TTEXIT(N,1)	Surface temperature at exit of discharge channel (entrance of reach 1) (°F)
4	TEND(2)	Surface temperature at end of reach 2 (°F)
5	TEND(3)	Surface temperature at end of reach 3 (°F)
6	TSUR(1,1)	Surface temperature at (1,1) of main lake; TSUR(1,1) = THL(1,1) (°F)
7	THL(9,1)	Temperature at (9,1) of heat loss region of main lake (°F)
8	THL(40,1)	Temperature at (40,1) of heat loss region of main lake (°F)
9	T(3,1)	see Fig 4.2 (°F)
10	T(4,1)	see Fig 4.2 (°F)
11	T(5,1)	see Fig 4.2 (°F)
12	TEQ	Filtered equilibrium temperature (°F)
13	DDRI	Eddy diffusivity (m^2 day^{-1}) in hypolimnion of main lake

*also see Fig 4.1 and Fig 4.2

4.2 Main Lake Data

This output data file gives temperatures at selected locations of the main lake and is extraneous unless specifically required. It is usually stored under the filename of MLXXXX; a sample output (ML1305) is included as Attachment 8. There are 14 columns of data in this file; an explanation is given in Table 4.3.

4.3 WHTF Data

This output data file gives miscellaneous data for the Waste Heat Treatment Facilities (WHTF) and is extraneous unless specifically required. It is usually stored under the filename of SAAXXXX. A sample output (SAA1305) is included as Attachment 9. There are 11 columns of data in this file; an explanation is given in Table 4.4.

Table 4.3 Summary of Main Lake Output

Column	Output Variable	Explanation
1	MMO/MDAY/MYR	Date
2	TSUR(1,1)	see Fig 4.2 (°F)
3	THL(3,1)	
4	THL(5,1)	
5	THL(7,1)	
6	THL(9,1)	
7	THL(15,1)	
8	THL(20,1) → Thermistors	
9	THL(25,1)	
10	THL(30,1)	
11	THL(40,1)	
12	TRSA(1)	Surface temperature of return side arm flow of main lake (°F)
13	TEQ	Filtered equilibrium temperature (°F)
14	TTEQ	Unfiltered equilibrium temperature (°F)

circled temperatures used by Marian Portman

Table 4.4 Summary of WHTF Output

Column	Output Variable	Explanation*
1	N	Reach number (1, 2 or 3).
2	BETAA	Coefficient of thermal expansion of water
3	QQSA1	Flow rate of side arm 1
4	QQSA2	Flow rate of side arm 2
5	TRS1(I)	Temperature of return flow of side arm 1
6	TRS2(I)	Temperature of return flow of side arm 2
7	TSA1(I)	Temperature at entrance of side arm 1
8	TSA2(I)	Temperature at entrance of side arm 2
9	XYXD	Upper layer depth of whole reach
10	DS	Dilution coefficient at entrance to the reach
11	FFSA1	Froude no. at the entrance of reach

APPENDIX 1. HISTORICAL "SYNTHETIC" INPUT DATA

A. Historical Meteorological Input Data

A ten year historical set of synthetically generated meteorological data for North Anna Power Station is available for the period April 1, 1957 through March 31, 1966. The data are in the form of two 5-year blocks stored on tape under the filenames of MET5761 and MET6266 respectively. Their format is identical to that of the recent (July 26, 1978 onwards) meteorological data (see Table 2.1) and can be readily used as input data files for the program. Note that this data is identical (but in a different format to be compatible as model input) to that listed in the Nov. 1977 Supplementary Data Report to "Analysis of Cooling Effectiveness and Transient Long-Term Simulation of a Cooling Lake with application to the North Anna Power Station".

B. Historical Plant Operation Input Data

Plant data are available for 1, 2 and 3 units of operation for five years under hypothetical, steady state (full load) conditions. They are stored on tape under the filenames of PL1UNIT, PL2UNIT and PL3UNIT respectively and are in format identical to that of the recent (July, 1978 onwards) plant operation data (see Table 2.2).

APPENDIX 2. DIAGNOSTIC EVALUATION OF RESULTS

A. To Analyse Errors at Four Control Points

A program was developed to compute differences ("errors") between model-predicted temperatures and field measurements at four control points, including TEXIT, TEND(3), THL(9,1) and THL(40,1). The locations of these four points are given in Fig 4.1. Measurements correspond to "NRC" Endeco recorders. The program is stored under the filename ERR FORTRAN and a copy of this is given as Attachment 10. One has to adjust the dimensions in the program to accomodate a new number of days and then recompile. The program is run by using ERR EXEC (Attachment 11).

Five input data files are required in this program. Table A2 gives the names and origins of these data files. Executing the program generates four output data files of raw errors between the predicted and measured temperatures.

B. To Compute Delta Errors Between Four Control Points

Raw errors at each control point obtained by executing ERR FORTRAN can be compared by using SUBTR FORTRAN and SUBTR EXEC (Attachments 12 and 13). Note that one may have to readjust the dimensions in SUBTR FORTRAN and redefine the I/O files in SUBTR EXEC as more data are added. Also note that SUBTR computes only one delta error file at a time.

Table A2 Input data files for running ERR FORTRAN

Filename	Origin
RUNXXXX	Obtained by running FULLPOND (see section 4.1)
NRC8	Daily averaged NRC Endeco temperature data for the main lake at the four control points (see Appendix 3)
NRC9	
NRC4	
NRC5	

APPENDIX 3. PROCESSING NRC ENDECO DATA

The NRC Endeco temperature data (hourly data) often contain gaps and various non-uniformities. To convert the data in a form compatible with ERR FORTRAN (see Appendix 2), one can use AVERNRC FORTRAN and AVERNRC EXEC. They are given as Attachments 14 and 15 respectively. Note that one may have to redefine the total number of data points and then recompile AVERNRC FORTRAN and redefine the I/O data files for AVERNRC EXEC. The program requires only one input data file which is the original NRC Endeco data as supplied by VEPCO and transferred from tape to disk. Nine output files of daily averaged temperature data at nine respective stations will be generated and usually they are merged with the existing already-processed data files. Temperature measurements at four out of the nine NRC Stations (see p. 49, TR 272) are compared with model prediction for diagnositc purposes (Appendix 2). Table A3 gives the relationship between the NRC temperature data and the corresponding output variable for the four control points.

Table A3 Correspondence between NRC data file and model output temperature

NRC data file	Output Variable in Model
NRC4	THL(40,1)
NRC5	THL(9,1)
NRC8	TEXIT
NRC9	TEND(3)

APPENDIX 4. INDEX OF VARIABLES FOR PROGRAM INPUT DATA

Input Variable	Card Group	Common Block*
AA	18	
AAB	17	
ACH	27	DISCH
BETA	5	INPOND
BSA1	30	WHTF1
BSA2	30	WHTF1
BØØ	31	WHTF1
DAA	17	
DCH	27	DISCH
DCHX	27	DISCH
DD	11	HYDRO
DSURF	8	
DT	4	PARAM
DTDD	8	TIMES
DTDIS	24	METEOE
DTMET	24	METEOE
DTQI	9	INB
DTTMP	8	TIMES
DXXL	17	
DY	4	MAINPB
EFACT	22	METEOD
EFF	4	
ELOUT	14	OUTB
ETA	5	CONST
FRIFAC	27	CONST
HBOT	32	WHTF3
HCAP	5	CONST
HHH	38	
HSA1	30	WHTF1

*The common blocks can be found in the COMMON statements at the beginning of FULLPOND FORTRAN and its subroutines.

APPENDIX 4., cont.

Input Variable	Card Group	Common Block
HSA2	30	WHTF1
HØØ	31	WHTF1
IDEBUG	40	BUGS
IODIS	25	METEOE
IOMET	25	METEOE
IPDAT	39	
ISLCT	38	
ISDLS	25	METEOE
ISELCT	41	BUGS
ISMET	25	METEOE
ISTEQ	38	
ITZERO	3	
JM	3	MAINPB
JMP	3	
KAREA	3	SWITCH
KITP	3	SWITCH
KKK	38	
KLOSS	3	SWITCH
KMIX	3	SWITCH
KOH	3	SWITCH
KQ	3	SWITCH
KSA	33	NUS
KSUR	3	SWITCH
LAGFIX	29	WHTF3
LOUT	13	OUTB
MDAY	2	
MIXED	3	MIXG
MMO	2	
MYR	2	
NAA	16	
NARSF	15	
NCB	26	DISCH

APPENDIX 4., cont.

Input Variable	Card Group	Common Block
NCH	26	DISCH
NDD	6	
NDTEMP	6	
NIN	6	INC
NMET	23	METEOE
NOUT	6	OUTB
NQI	7	INB
NSA	29	WHTF1
NSURF	6	
NXXL	16	
PLUMF	27	MAINPB
PRAIR	4	METEOD
RHO	5	CONST
RHØ	32	WHTF3
RMIX	5	MIXG
RTW	32	WHTF3
SAFACT	22	
SURF	20	SURFS
T	10	MAINPA
TSTOP	4	PARAM
TZERO	4	PARAM
WCH	27	DISCH
WCHX	27	DISCH
WGT	22	
WH	1	
XAREA	35	NUS
XB	37	NUS
XCH	27	DISCH
XH	36	NUS
XLSA1	30	
XLSA2	30	
XSAR	34	NUS

APPENDIX 4., cont.

Input Variable	Card Group	Common Block
XXL	19	
XXLB	17	
XWIND	17	
YSUR	4	PARAM
ZA1	28	WHTF1
ZA2	28	WHTF1
ZA3	28	WHTF1
ZASA1	30	WHTF1
ZASA2	30	WHTF1
ZB	28	WHTF1
ZBR	28	WHTF1
ZH	28	WHTF1
ZHM	28	WHTF1

APPENDIX 5. SELECTION FLAGS (CARD GROUP 41) USED IN FULLPOND FORTRAN

ISELCT(1) – Turns off WHTF input displays if .GT.Ø

ISELCT(2) – WHTF lags fixed at input lagtimes

ISELCT(3) – Dilution at Dike III based on jet or Control if .EQ.Ø;
on Control if .GT.Ø; on jet if .LT.Ø

ISELCT(4) – Starting time step for WHTF debug print

ISELCT(5) – Ending time step for WHTF debug print

ISELCT(6) – Starting time step for simulation if .GT.1

ISELCT(7) – Debug write unit

ISELCT(8) – Recirculating simulation mode if .GT.Ø

ISELCT(9) – Complete print out at every time step if .GT.Ø

ISELCT(10) – Heat loss calculations match hotpond program if .GT.Ø

ISELCT(11) – Storage file unit for WHTF output

ISELCT(12) – Output storage from main if .GT.Ø

ISELCT(13) – Selective withdrawal if .EQ.Ø;
depth average if .EQ.1;
compare if .EQ.2

ISELCT(14) – Water loss printout; off if .EQ.Ø; monthly if .EQ.1;
daily if .EQ.2

ISELCT(15) – Sub. aver. iteration limit:
N if .EQ.N;
50 if .EQ.Ø;
no limit if .EQ.99999

ISELCT(16) – CORRECTION FOR PLANT OPERATION DATA
[illegible] if .GT. 0
reduced heat load ($Q = Q \times 1.054$; $DT = DT \times 0.874$)
if .LE. 0

ISELCT(17) – MODIFICATION OF RESIDENCE TIMES CALCULATION WITH
no modification if .EQ. 0
based on whole volume x 0.7[illegible] if .EQ. 1

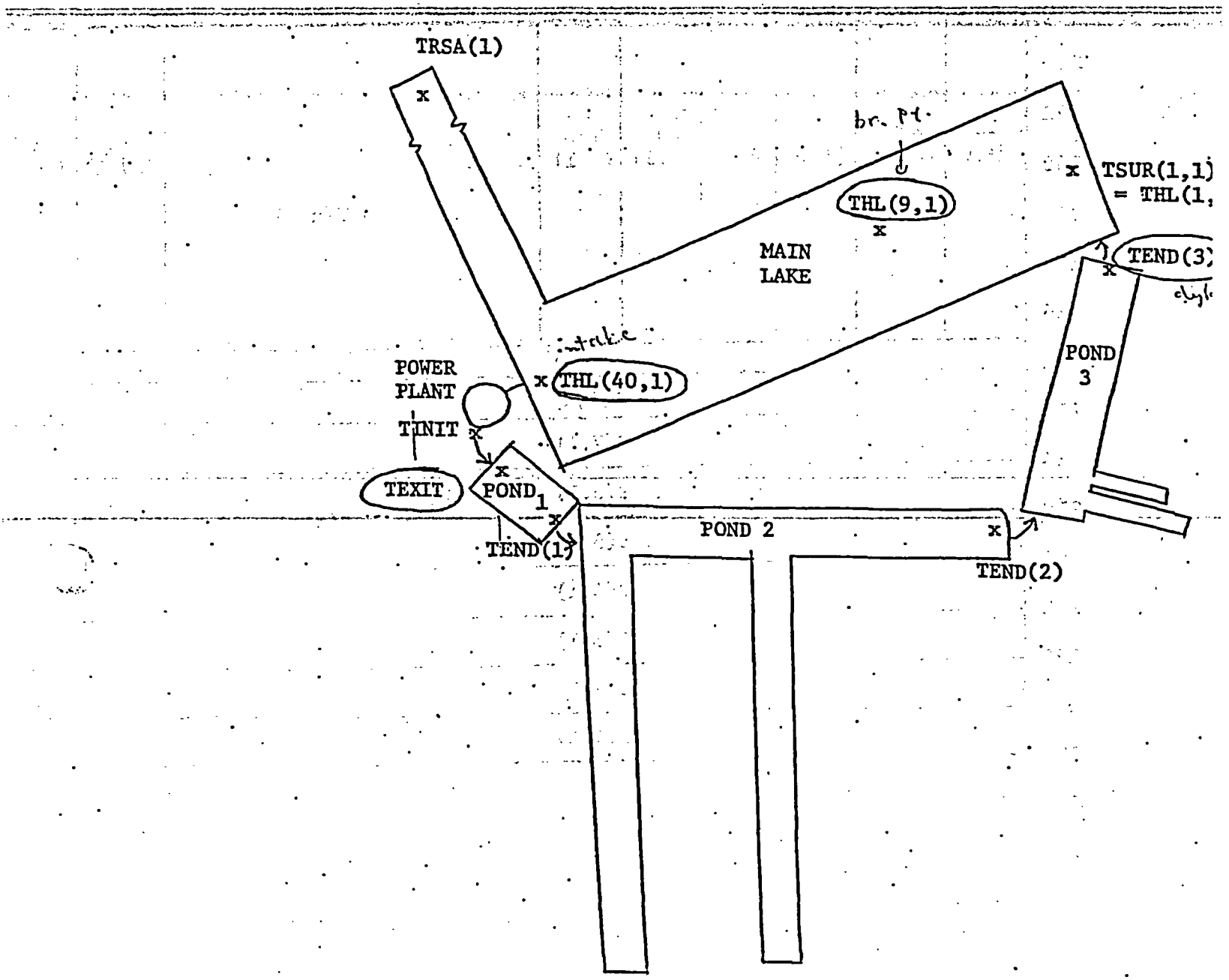


Fig. 4.1 Schematization of the North Anna Cooling System used in the Segmented Model and the locations of the four control points for diagnostic purpose.

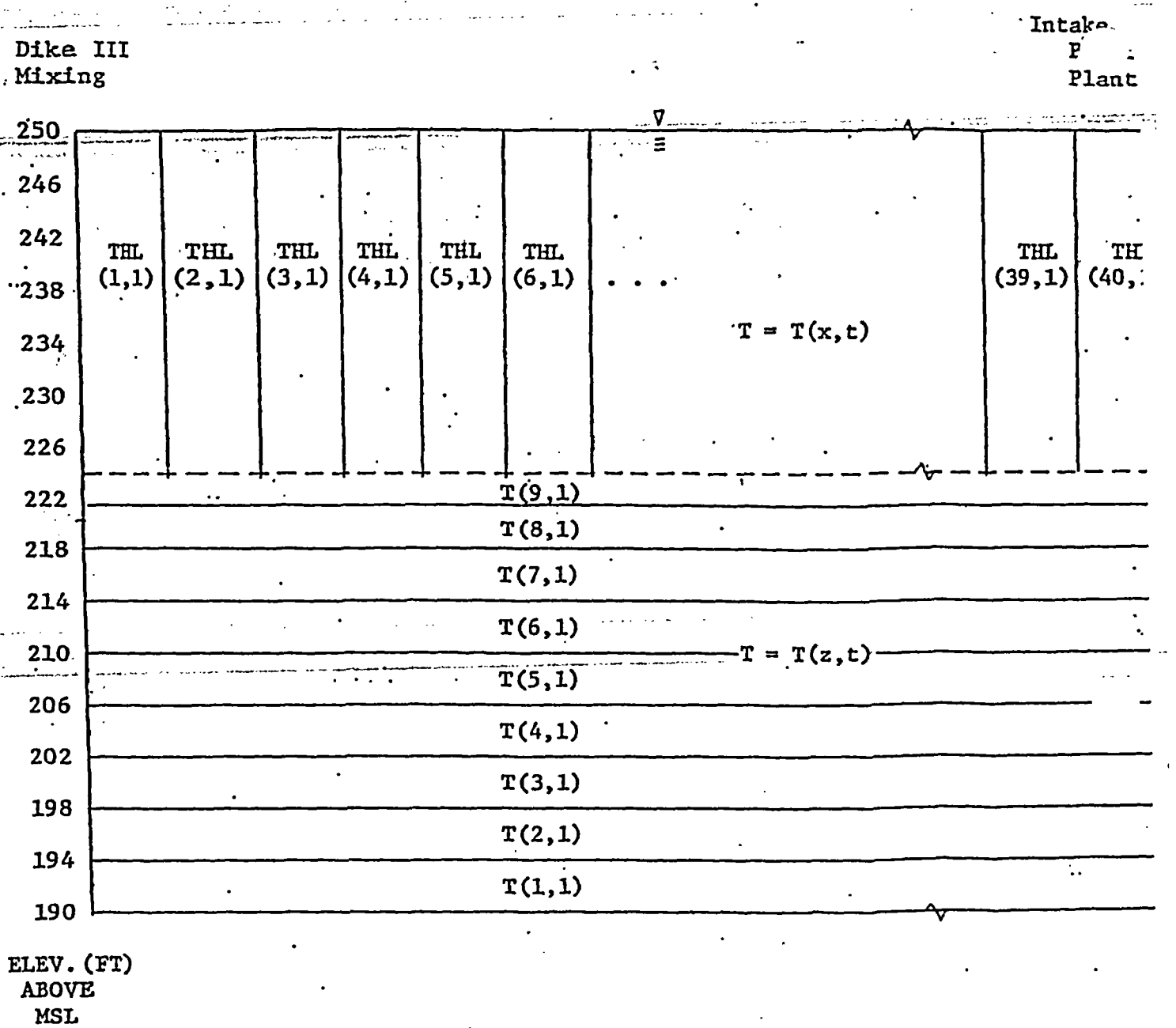


Fig. 4.2 Schematization of the Main Lake used in the Segmented Model

Attachment L (main program)

```
C     MAIN PROGRAM
C     THIS DECK MODIFIED FROM THE ORIGINAL
C
C     UPDATE    DATE                          DESCRIPTION
C     ------  --------  ------------------------------------------------
C        1    10/26/78  SPECIFICATION OF INITIAL TEMPERATURE PROFILE
C        2    10/26/78  CHANGES TO WHTF REACH LAGGING CODE
C        3    10/26/78  ADDITION OF SELECTION AND DEBUGGING OPTIONS
C        4    10/26/78  CALCULATION OF NATURAL AND FORCED WATER LOSS
C        5    10/26/78  CHANGES TO DIKE III DILUTION CALCULATION
C        6    10/26/78  SEPARATION OF WHTF CALCULATIONS INTO SUBROUTINE
C        7    10/26/78  CORRECTION OF EARLIER PROGRAMMING ERRORS
C        8    11/02/78  CHANGES TO MET DATA INPUT
C        9    11/02/78  CHANGE INPUT UNITS FOR FLOW FROM CFD TO CFS
C       10    11/07/78  ADD COMMON BLOCK IOPUT
C       11    11/20/78  CORRECTION OF JET MIXING FOR PLANT DISCHARGE
C       12    11/20/78  ADDITION TO TOUT TO OUTPUT
C       13    01/23/79  ADD MONTHLY PRINT-OUT OF WATER LOSS
C       14    02/21/79  RECODE WHTF TEMP CALCULATIONS
C       15    05/22/79  CMS VERSION CREATED
C       16    07/02/79  DEPTH AVERAGED WITHDRAWAL ADDED
C       17    08/13/79  SUB COLDCK ADDED FOR EXTREME COLD CONDITIONS
C       17    07/20/79  RECIRCULATING MODE INITIALIZED
C       18    02/20/81  FILTERED TEQ SUBROUTINE ADDED(& ASSORTED ERRORS)
C       19    12/31/81  FILTERED K SUBROUTINE ADDED ; USE OF IDSO AND
C                       JACKSON FORMULA; SA CUT-OFF FLOW CRITERION
C                       CHANGED; AND MANY MINOR CHANGES
C       20    10/06/83  BRUTSAERT FORMULA FOR LONG WAVE RADIATION USED;
C                       SWINBANK AND IDSO-JACKSON FORMULAE COMMENTED OUT
C                       IN SUBROUTINE FSURLAY
C       21    12/09/83  IDSO-JACKSON FORMULA REUSED; BRUTSAERT AND
C                       SWINBANK COMMENTED OUT
C       22    01/06/84  OPTIONS OF ADJUSTING PLANT OPERATION DATA ADDED
C       23    03/16/84  WHTF RESIDENCE TIMES CALC. BASED ON WHOLE VOLUME
C                       *0.75 ADDED AS AN OPTION
C     ------  --------  ------------------------------------------------
C
C
      COMMON /IOPUT/ IREAD,IRITE,IRR
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)
      COMMON/NEW/ TRSA(10),XQSA(10),ETSA(10)
      COMMON/NUS/ KSA,NESA(10),XSAR(10),XH(10),XB(10),XAREA(10)
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH,
     +        MIXD(5)
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR
      COMMON /PARAM2/ RTIME,RDATE,NSTEP
      DOUBLE PRECISION RTIME,RDATE
      COMMON /FLOW/ SPREAD,SIGMAI,SIGMAO,FLUXB,SPVOL,DERIV,HAFDEL,EPSIL
      COMMON /TIMES/ DTDD,DTTMP,DTTB
      COMMON /SURFS/ DYSUR,AYSUR,SURF(1893),TSUR(100,2),ELBSL
      COMMON /MR/ TMR(1893,2),QMR(1893,2),DMR(1893,2),AMR(1893,2)
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2),TCOOL(1893,2),
     +        QCOOL(1893,2)
      COMMON /THICK/ THICK1,THICK2,THICKB
```

FSURLAY

```
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR          FUL00560
      COMMON /RALPH/ DATRAD,ATRAD,AS(20,2),III,LAGTIM(1893),NH          FUL00570
      COMMON /PARSON/ EX(102),EXO(102),EXI(102),OX(102),UO(102,1)       FUL00580
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR  FUL00590
      COMMON /DRFH/ AH(50),TBOT,CONBOT,TB(20,2),TAB(40),NDYB            FUL00600
      COMMON /METEOA/ WIND(1893),TA(1893),SIGH(1893),FIN(1893),         FUL00610
     +       CLOUD(1893)                                                FUL00620
      COMMON /METEOB/ DTTA,DTSIGH,DTFIN,DTWIND,DCLOUD                   FUL00630
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                  FUL00640
      REAL LWRI,LWRO                                                    FUL00650
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                          FUL00660
      COMMON /METEOE/ NMET,DTMET,IOMET,NDIS,DTDIS,IODIS,ISMET,ISDIS     FUL00670
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB           FUL00680
      COMMON /OUTA/ NQO(5),DTQO(5),QO(1893,2),UOMAX(5),UOT(102,5)       FUL00690
      COMMON /TIME/ ATIME                                               FUL00700
      COMMON /OUTB/ JOUT,YOUT,ELOUT(5),TOUTC(5),NOUT,LOUT(5)            FUL00710
      COMMON /INA/ DTTI(5),TI(1893,2),TIN(1893),NTI(5)                  FUL00720
      COMMON /INB/ DTQI(5),QI(1893,2),QIN(1893),NQI(5)                  FUL00730
      COMMON /INC/ UIMAX(5),SIGMIN(5),NIN,JIN(5)                        FUL00740
      COMMON /INPOND/ AINTKP(5),VINTKP(5),YINTKP(5),TINTK,TINTK1        FUL00750
      COMMON /ENERGY/ EO,E1,E2,E3,ET,FLUXT                              FUL00760
      COMMON /MAINPA/ T(102,2),EL(102),XL(102),A(102),V(102,1),UI(102,5) FUL00770
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                 FUL00780
      COMMON/NDY/NDAYS(13)                                              FUL00790
      COMMON /SWITCH/ KSUR,KOH,KQ,KLOSS,KMIX,KAREA,KATRAD,KTRAV         FUL00800
      COMMON /WHTF1/ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO,    FUL00810
     + NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4),HSA2(4),ZBR(4), FUL00820
     + XLSA1(4),XLSA2(4)                                                FUL00830
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893),   FUL00840
     + QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)    FUL00850
      DIMENSION WH(20),AA(102),XXL(102),ISLCT(3)                        FUL00860
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                            FUL00870
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),    FUL00880
     +               QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH       FUL00890
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                    FUL00900
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                           FUL00910
      COMMON/FLAG/IWDFLG                                                FUL00920
      DIMENSION DATA1(1893),TEQL(1893),FTEQL(1893)                      FUL00930
      DIMENSION DATA2(1893),DATA3(1893),DATA4(1893),DATA5(1893)         FUL00940
      DIMENSION DATA6(1893),DATA7(1893)                                 FUL00950
      DIMENSION IPDAT(3,40)                                             FUL00960
C     ENDING DAY OF MONTH FOR THE YEAR APRIL TO MARCH                   FUL00970
      DIMENSION IDAZE(12)                                               FUL00980
      DATA IDAZE /30,61,91,122,153,183,214,244,275,306,334,365/         FUL00990
      CALL DATIME(RDATE,RTIME)                                          FUL01000
                                                                        FUL01010
C     READ IN ALL DATA FOR PROGRAM                                      FUL01020
                                                                        FUL01030
      IREAD = 5                                                         FUL01040
      IRITE = 6                                                         FUL01050
      READ (5,900) (WH(I),I=1,20)                                       FUL01060
      READ(IREAD,901)MMO,MDAY,MYR,IBUG                                  FUL01070
      CALL DAXIME(MDAY,MMO,MYR,ATIME)                                   FUL01080
      MYR1=MYR-1                                                        FUL01090
      CALL DAXIME(31,12,MYR1,ATIMEX)                                    FUL01100
```

```
      READ(5,901)JM,JMP,KITP,KSUR,KOH,KQ,KLOSS,NPRINT,KAREA,KMIX,MIXED, FULO1110
     +          ITZERO                                                  FULO1120
      READ (5,902) YSUR,DY,DT,TSTOP,TZERO,PRAIR,EFF                     FULO1130
      NSTEP=INT(TSTOP/DT)                                               FULO1140
      READ(5,902) ETA,BETA,RHO,HCAP,RMIX                                FULO1150
      READ(IREAD,901) NDD,NIN,NOUT,NSURF,NDTEMP                         FULO1160
      READ(5,901) (NQI(K),K=1,NIN)                                      FULO1170
      IF(NOUT.GT.1) READ(5,901) (NQO(K),K=2,NOUT)                       FULO1180
      READ(IREAD,902)DSURF,DTDD,DTTMP                                   FULO1190
      READ(5,902) (DTQI(K),K=1,NIN)                                     FULO1200
      IF(NOUT.GT.1) READ(5,902) (DTQO(K),K=2,NOUT)                      FULO1210
      IF(ITZERO.EQ.0)GO TO 219                                          FULO1220
C     READ INITIAL TEMPERATURE PROFILE                                  FULO1230
      READ(5,902) (T(I,1),I=1,JM)                                       FULO1240
      TZERO=T(JM,1)                                                     FULO1250
      JM1 = JM+1                                                        FULO1260
           DO 218 I=JM1,JMP                                             FULO1270
  218      T(I,1) = T(I,JM)                                            FULO1280
  219 CONTINUE                                                          FULO1290
      READ(5,902) (DD(I),I=1,NDD),DD3,DD4                               FULO1300
      READ(5,9902) DDCOEF,XCOEF                                         FULO1310
 9902 FORMAT(2F10.0)                                                    FULO1320
      IF(NIN.LT.2)GO TO 2222                                            FULO1330
         DO 222 K=2,NIN                                                 FULO1340
         NQII=NQI(K)                                                    FULO1350
         READ(5,975) (QI(I,K),I=1,NQII)                                 FULO1360
              DO 2221 I=1,NQII                                          FULO1370
 2221         QI(I,K)=QI(I,K)*86400.0                                   FULO1380
  222    CONTINUE                                                       FULO1390
 2222 IF(NOUT.EQ.1) GO TO 224                                           FULO1400
      DO 223 K=2,NOUT                                                   FULO1410
      NQOO=NQO(K)                                                       FULO1420
      READ(5,975) (QO(I,K),I=1,NQOO)                                    FULO1430
           DO 2231 I=1,NQOO                                             FULO1440
 2231      QO(I,K) = QO(I,K)*86400.0                                    FULO1450
  223 CONTINUE                                                          FULO1460
  224 CONTINUE                                                          FULO1470
      READ(5,901)(LOUT(I),I=1,NOUT)                                     FULO1480
      READ(5,902)(ELOUT(I),I=1,NOUT)                                    FULO1490
      GO TO (9,10),KITP                                                 FULO1500
    9 READ(5,902)(AINTKP(I),VINTKP(I),YINTKP(I),I=1,2)                  FULO1510
   10 YBOT=ELOUT(1)-DY*FLOAT(LOUT(1)-1)                                 FULO1520
      READ (5,901) NARSF                                                FULO1530
      GO TO (4,2), KAREA                                                FULO1540
C     READ IN DATA FOR OTHER THAN LABORATORY RESERVOIR IF INDICATED     FULO1550
    2 READ(IREAD,901) NAA,NXXL                                          FULO1560
      READ(IREAD,902) DAA,DXXL,AAB,XXLB,XWIND                           FULO1570
      READ (5,902) (AA(I),I=1,NAA)                                      FULO1580
      READ (5,961) (XXL(I),I=1,NXXL)                                    FULO1590
      READ(5,902)(SURF(I),I=1,NSURF)                                    FULO1600
      READ(5,901)NREACH                                                 FULO1610
      CALL RDWHTF                                                       FULO1620
      READ(IREAD,981)HOO,BOO                                            FULO1630
      DO 853 I=1,NAA                                                    FULO1640
  853 AA(I)=AA(I)*4840.*9.                                              FULO1650
```

```
      DO 3 I=1,JMP                                                      FUL01660
      EL(I)=YBOT+DY*FLOAT(I-1)                                          FUL01670
      RA=(EL(I)-AAB)/DAA                                                FUL01680
      L=RA-0.0001                                                       FUL01690
      A(I)=AA(L+1)+(RA-FLOAT(L))*(AA(L+2)-AA(L+1))                      FUL01700
      RA=(EL(I)-XXLB)/DXXL                                              FUL01710
      L=RA-0.0001                                                       FUL01720
      XL(I)=XXL(L+1)+(RA-FLOAT(L))*(XXL(L+2)-XXL(L+1))                  FUL01730
    3 B(I)=A(I)/XL(I)                                                   FUL01740
      GO TO 978                                                         FUL01750
    4 DO 8 I=1,JMP                                                      FUL01760
      B(I)=20.8                                                         FUL01770
      XL(I)=42.5                                                        FUL01780
      IF(I.LT.NCB)GO TO 12                                              FUL01790
      A(I)=877.5                                                        FUL01800
      GO TO 8                                                           FUL01810
   12 A(I)=840.                                                         FUL01820
    8 CONTINUE                                                          FUL01830
  978 CONTINUE                                                          FUL01840
      READ(5,981)RTW,RHO,HBOT                                           FUL01850
      READ(5,901) KSA                                                   FUL01860
      IF(KSA.EQ.0) GO TO 251                                            FUL01870
      READ(5,981) (XSAR(I),I=1,KSA)                                     FUL01880
      READ(5,981) (XAREA(I),I=1,KSA)                                    FUL01890
      READ(5,981) (XH(I),I=1,KSA)                                       FUL01900
      READ(5,981) (XB(I),I=1,KSA)                                       FUL01910
C     CONVERT AREAS FROM ACRES TO SQUARE FEET                           FUL01920
      DO 230 I=1,KSA                                                    FUL01930
      XSAR(I)=XSAR(I)*4840.*9.                                          FUL01940
230   XAREA(I)=XAREA(I)*4840.*9.                                        FUL01950
251   CONTINUE                                                          FUL01960
      READ(5,1555)ISTEQ,HHH,KKK,ISLCT(1),ISLCT(2),ISLCT(3)              FUL01970
 1555 FORMAT(I5,F10.2,I5,3I5)                                           FUL01980
      READ(5,913)IPDAT                                                  FUL01990
      READ(5,899)IDEBUG                                                 FUL02000
      IBUG = IDEBUG(1)                                                  FUL02010
      READ(5,899)ISELCT                                                 FUL02020
C                                                                       FUL02030
C       SELECT FLAGS                                                    FUL02040
C       1 - TURNS OFF WHTF INPUT DISPLAYS IF GT 0                       FUL02050
C       2 - WHTF LAGS FIXED AT INPUT LAGTIMES                           FUL02060
C       3 - DILUTION AT DIKE III BASED ON JET OR CONTROL IF .GT. 0      FUL02070
C       4 - STARTING TIME STEP FOR WHTF DEBUG PRINT                     FUL02080
C       5 - ENDING TIME STEP FOR WHTF DEBUG PRINT                       FUL02090
C       6 - STARTING TIME STEP FOR SIMULATION IF GT 1                   FUL02100
C       7 - DEBUG WRITE UNIT                                            FUL02110
C       8 - RECIRCULATING SIMULATION MODE IF .GT. 0                     FUL02120
C       9 - COMPLETE PRINT OUT AT EVERY TIME STEP IF .GT. 0             FUL02130
C      10 - HEAT LOSS CALCS MATCH HOTPOND PROGRAM IF .GT. 0             FUL02140
C      11 - STORAGE FILE UNIT FOR WHTF OUTPUT                           FUL02150
C      12 - OUTPUT STORAGE FROM MAIN IF .GT. 0                          FUL02160
C      13 - IF =0 SELECTIVE WDRAWAL, IF =1 DEPTH AVERAGE, IF =2 COMPAREFUL02170
C      14 - WATER LOSS PRINTOUT:  OFF=0, MONTHLY=1, DAILY=2             FUL02180
C      15 - SUB. AVER. ITERATION LIMIT: N IF .EQ. N; 50 IF .EQ. 0;      FUL02190
C           NO LIMIT IF .EQ. 99999                                      FUL02200
```

cf p. 12 User's Manual

```
C       16 - ADJUSTMENT FOR PLANT OPERATION DATA:                          FUL02210
C            NO ADJUSTMENT IF .GT. 0                                       FUL02220
C            SAME HEAT-LOAD (Q=Q*1.1;DT=DT/1.1) IF .LT. 0                  FUL02230
C            REDUCED HEAT LOAD (Q=Q*1.1;DT=DT/1.1) IF .EQ. 0               FUL02240
C       17 - CALCULATION OF RESIDENCE TIMES IN THE WHTF                    FUL02250
C            NO MODIFICATION IF .EQ. 0                                     FUL02260
C            BASED ON WHOLE VOLUME * 0.75 IF .EQ. 1                        FUL02270
C                                                                          FUL02280
C     DISPLAY INPUT DATA                                                   FUL02290
C                                                                          FUL02300
      CALL DSPARM(WH)                                                      FUL02310
      CALL DSWHTF                                                          FUL02320
      WRITE(IRITE,1002)                                                    FUL02330
      WRITE(IRITE,904)UM,YSUR,RHO                                          FUL02340
      WRITE(IRITE,905)(LOUT(I),ELOUT(I),I=1,NOUT)                          FUL02350
      WRITE(IRITE,906)DY,YBOT,ETA                                          FUL02360
      WRITE(IRITE,910)KITP,KSUR,KOH,KQ,KLOSS,KMIX                          FUL02370
      WRITE(IRITE,923)MIXED,RMIX                                           FUL02380
      IRR = 20                                                             FUL02390
      ISTRT = 1                                                            FUL02400
      IF(ISELCT(7).GT.0)IRR=ISELCT(7)                                      FUL02410
      IF(ISELCT(6).GT.ISTRT)ISTRT=ISELCT(6)                                FUL02420
      CALL GETMET(IOMET,ISMET,NMET,DTMET,NSTEP,DT,TA,DATA1)                FUL02430
      CALL GETMET(IOMET,ISMET,NMET,DTMET,NSTEP,DT,SIGH,DATA2)              FUL02440
      CALL GETMET(IOMET,ISMET,NMET,DTMET,NSTEP,DT,WIND,DATA3)              FUL02450
      CALL GETMET(IOMET,ISMET,NMET,DTMET,NSTEP,DT,CLOUD,DATA4)             FUL02460
      CALL GETMET(IOMET,ISMET,NMET,DTMET,NSTEP,DT,FIN,DATA5)               FUL02470
      CALL GETMET(IODIS,ISDIS,NDIS,DTDIS,NSTEP,DT,QPLNT,DATA6)             FUL02480
      CALL GETMET(IODIS,ISDIS,NDIS,DTDIS,NSTEP,DT,TPLNT,DATA7)             FUL02490
C     CONVERT INPUT PLANT FLOWS FROM CFS TO CFD                            FUL02500
        DO 2532 I=1,NSTEP                                                  FUL02510
 2532   QPLNT(I) = QPLNT(I)*86400.0                                         FUL02520
                                                                           FUL02530
C     ADJUSTMENT OF PLANT DATA                                             FUL02540
C     IPLNT .LE. 0 -- ADJUSTED FOR POSSIBLE INACCURACIES IN FLOW AND       FUL02550
C                     PLANT OPERATION LEVEL WITH TOTAL HEAT LOAD           FUL02560
C                     REDUCED BY ABOUT 7.9% (Q=Q*1.054;DT=DT*0.874)        FUL02570
C     IPLNT .GT. 0 -- NO ADJUSTMENT APPLIED.                               FUL02580
      IPLNT=ISELCT(16)                                                     FUL02590
      IF (IPLNT) 551,551,553                                               FUL02600
  551 DO 60 I=1,NSTEP                                                      FUL02610
      QPLNT(I)=QPLNT(I)*1.054                                              FUL02620
   60 TPLNT(I)=TPLNT(I)*0.874                                              FUL02630
  553 CONTINUE                                                             FUL02640
                                                                           FUL02650
C     INITIALIZE MANY VARIABLES                                            FUL02660
      WLT = 0.0                                                            FUL02670
      NCNT=1                                                               FUL02680
      WLTN = 0.0                                                           FUL02690
      WLTWH = 0.0                                                          FUL02700
      PWDCUM=0.                                                            FUL02710
      DTMP1=20.                                                            FUL02720
      DO 850 N=1,NMET                                                      FUL02730
      QIN(N)=0.0                                                           FUL02740
      TIN(N)=0.0                                                           FUL02750
```

Reduced heat load
(Q = Q * 1.054 ; DT = DT * 0.874)
if .LE. 0

```
      QEXIT(N,1)=0.                                                    FUL02760
      TEXIT(N,1)=0.                                                    FUL02770
      TMR(N,1)=0.                                                      FUL02780
      QMR(N,1)=0.                                                      FUL02790
      DMR(N,1)=DY                                                      FUL02800
      AMR(N,1)=0.                                                      FUL02810
      QFIN(N)=QPLNT(1)                                                 FUL02820
      DO 850 I=1,3                                                     FUL02830
      TFIN(N,I)=TZERO                                                  FUL02840
  850 CONTINUE                                                         FUL02850
      IF(ITZERO.GT.0)GO TO 849                                         FUL02860
           DO 848 I=1,JMP                                              FUL02870
           T(I,1) = TZERO.                                             FUL02880
  848      CONTINUE                                                    FUL02890
  849 DO 851 I=1,JMP                                                   FUL02900
  851 QQMIX(I)=0.0                                                     FUL02910
      NAR=NARSF+1                                                      FUL02920
      DO 11 I=1,NAR                                                    FUL02930
      THL(I,1)=T(JM,1)                                                 FUL02940
   11 THL(I,2)=T(JM,1)                                                 FUL02950
      TMIX=TZERO                                                       FUL02960
      TINTK1=TZERO                                                     FUL02970
      MIXH=1                                                           FUL02980
      SPREAD=1.96                                                      FUL02990
      DTT=DT                                                           FUL03000
      NPR=0                                                            FUL03010
      JXM=JM                                                           FUL03020
      N = 0                                                            FUL03030
      JMIXB = JM-MIXED                                                 FUL03040
      QMIX = 0.0                                                       FUL03050
      ET = 0.0                                                         FUL03060
      RAD = 0.0                                                        FUL03070
      EVAP = 0.0                                                       FUL03080
      TAIR = 0.0                                                       FUL03090
      EPSIL = 0.0                                                      FUL03100
      XTAVG=0.0                                                        FUL03110
      HAFDEL = 0.0                                                     FUL03120
      K111=NIN+KSA                                                     FUL03130
      DO 852 K=1,K111                                                  FUL03140
      MIXD(K)=MIXED                                                    FUL03150
  852 JIN(K)=JM                                                        FUL03160
      NIN1=NIN                                                         FUL03170
      DO 755 I=1,KSA                                                   FUL03180
      DTTI(NIN1+I)=DTT                                                 FUL03190
  755 DTQI(NIN1+I)=DTT                                                 FUL03200
      NIN=NIN+KSA                                                      FUL03210
      DO 854 K=1,NOUT                                                  FUL03220
  854 TOUTC(K)=TZERO                                                   FUL03230
      CUMQIN=0.0                                                       FUL03240
      CUMQOT=0.0                                                       FUL03250
      DYSUR=YSUR-EL(JM)+DY/2.0                                         FUL03260
      IF(YSUR-EL(JM)) 858,858,859                                      FUL03270
  858 AYSUR=A(JM)-(DY/2.0-DYSUR)*(A(JM)-A(JM-1))/DY                    FUL03280
      GO TO 860                                                        FUL03290
  859 AYSUR=A(JM)+(DYSUR-DY/2.0)*(A(JM+1)-A(JM))/DY                    FUL03300
```

```
  860 SAREA=(AYSUR+(A(JM)+A(JM-1))/2.0)/2.0                          FUL03310
      SAREA1=SAREA                                                   FUL03320
      DYSUR1=DYSUR                                                   FUL03330
      JM1=JM                                                         FUL03340
C     INITIALIZE VARIABLES USED IN ENERGY CHECK                      FUL03350
      FLUXT=0.                                                       FUL03360
      E2=0.0                                                         FUL03370
      E3=0.0                                                         FUL03380
      EO=A(1)/2.*T(1,1)+SAREA*DYSUR/DY*T(JM,1)                       FUL03390
      JMM=JM-1                                                       FUL03400
      DO 13 I=2,JMM                                                  FUL03410
   13 EO=EO+A(I)*T(I,1)                                              FUL03420
      EO=EO*DY*RHO*HCAP                                              FUL03430
      WRITE(6,930)EO                                                 FUL03440
      WRITE(6,1237)NAA                                               FUL03450
 1237 FORMAT(1X,'NAA=',I5)                                           FUL03460
      ANAREA=AA(NAA)/(4840.*9.)                                      FUL03470
      WRITE(6,798) ANAREA                                            FUL03480
  798 FORMAT(1X,'THE AREA OF THE MAIN BODY IS',F8.0,2X,'ACRES')      FUL03490
      WRITE(6,240) KSA                                               FUL03500
  240 FORMAT(/5X,'NUMBER OF SIDEARMS =',I5)                          FUL03510
      IF(KSA.EQ.0) GO TO 252                                         FUL03520
      DO 231 I=1,KSA                                                 FUL03530
      XXSAR=XSAR(I)/(4840.*9.)                                       FUL03540
      WRITE(6,232) I,XXSAR,I,XH(I),I,XB(I)                           FUL03550
  232 FORMAT(/5X,'AREA OF SIDEARM',I3,' =',F10.1,2X,'ENTRANCE DEPTH OF FUL03560
     +       SIDEARM',I3,' =',F10.1,2X,'AVERAGE WIDTH OF SIDEARM',   FUL03570
     +       I3,' =',F10.1)                                          FUL03580
  231 CONTINUE                                                       FUL03590
  252 CONTINUE                                                       FUL03600
      IF(JM-60) 15,15,16                                             FUL03610
   15 JP = JM                                                        FUL03620
      GO TO 17                                                       FUL03630
   16 JP=60                                                          FUL03640
   17 GO TO (191,18), KQ                                             FUL03650
   18 UIMAX(1)=0.0                                                   FUL03660
      UOMAX(1) = 0.0                                                 FUL03670
      TS=0.0                                                         FUL03680
      DO 19 J=1,JM                                                   FUL03690
      UI(J,1) = 0.0                                                  FUL03700
      UO(J,1)=0.0                                                    FUL03710
   19 V(J,1) = 0.0                                                   FUL03720
      SIGMA0 = 1.0                                                   FUL03730
      ID = 1                                                         FUL03740
  191 CONTINUE                                                       FUL03750
                                                                     FUL03760
C     THIS SUBROUTINE FILTERS THE EQUILIBRIUM TEMPS                  FUL03770
                                                                     FUL03780
      CALL FTEQ(NSTEP,ISTEQ,HHH,KKK,ISLCT,ISELCT,FTEQL,TEQL,NREACH)  FUL03790
C     STATEMENT 20 IS BEGINNING OF MAIN ITERATION LOOP OF PROGRAM    FUL03800
   20 CONTINUE                                                       FUL03810
      DO 323 N=ISTRT,NSTEP                                           FUL03820
      WINDY = WIND(N)                                                FUL03830
      TAIR = TA(N)                                                   FUL03840
      PSI = SIGH(N)                                                  FUL03850
```

```
      SR = FIN(N)                                                       FUL03860
      CC = CLOUD(N)                                                     FUL03870
      TEQ=FTEQL(N)                                                      FUL03880
      TTEQ=TEQL(N)                                                      FUL03890
                                                                        FUL03900
C     CALCULATION OF THE EDDY DIFFUSIVITY BASED ON A RICHARDSON CRITERIAFUL03910
                                                                        FUL03920
      DELRHE = (THL(40,1)-39.)**2-(T(1,1)-39.)**2                       FUL03930
      IF(DELRHE.LT.1.0)DELRHE=1.0                                       FUL03940
      DELRH2=DELRHE/500.                                                FUL03950
      IF(DELRH2.LT.1.0)DELRH2=1.0                                       FUL03960
      QQQQ=QPLNT(N)/(86400.*2160)                                       FUL03970
      WWWW=WIND(N)/5.0                                                  FUL03980
C     IF(QQQQ.LT.1.0)QQQQ=1.0                                           FUL03990
      IF(WWWW.LT.1.0)WWWW=1.0                                           FUL04000
      DDRI = DD(1)/((DDCOEF+DELRHE)**XCOEF)                             FUL04010
C     DDRI=DD(1)*(QQQQ**5)/(((DDCOF2*QQQQ**2)+DELRHE)**XCOEF)           FUL04020
      DDRI2=((DD3*(QQQQ**3))+(DD4*((WWWW**3)-1)))/DELRH2                FUL04030
      IF(DDRI2.GT.DDRI)DDRI=DDRI2                                       FUL04040
      IF(DDRI.GT.60.)DDRI=60.                                           FUL04050
      WRITE(6,1236)DELRHE,DDRI                                          FUL04060
1236  FORMAT(1X,'DELRHE=',F10.2,5X,'DDRI=',F10.2)                       FUL04070
                                                                        FUL04080
      ET=ET+DT                                                          FUL04090
      IF(ISELCT(8).GT.0)TI(N,1)=TOUTC(1)+TPLNT(N)                       FUL04100
      CALL SURLAY(N,TTEQ)                                               FUL04110
      IF(IBUG.GT.2)WRITE(20,1056)N,WLT                                  FUL04120
 1056 FORMAT(' *** MAIN-DEBUG *** AT TIME STEP ',I4,                    FUL04130
     +       ' TOTAL WATER LOSS (FORCED) IS ',E15.6,' AC-FT')           FUL04140
      XTAVG=XTAVG+(TOUTC(1)-TEQ)                                        FUL04150
      IF(KSA.EQ.0) GO TO 254                                            FUL04160
      DO 235 ISA=1,KSA                                                  FUL04170
      L=NIN1+ISA                                                        FUL04180
      TI(N,L)=TRSA(ISA)                                                 FUL04190
235   QI(N,L)=XQSA(ISA)                                                 FUL04200
254   CONTINUE                                                          FUL04210
      GO TO (21,503),KQ                                                 FUL04220
   21 GO TO (24,22), KMIX                                               FUL04230
C     MIX INFLOW WATER IF INDICATED                                     FUL04240
   22 DO 26 K=1,K111                                                    FUL04250
      IF(K.GT.1) GO TO 28                                               FUL04260
      QIN(K)=QCOOL(N,1)                                                 FUL04270
      TIN(K)=TCOOL(N,1)                                                 FUL04280
      JIN(K)=JM                                                         FUL04290
      GO TO 26                                                          FUL04300
   28 TP=0.0                                                            FUL04310
      JMIXB=JM-MIXD(K)                                                  FUL04320
      DO 23 J=JMIXB,JM                                                  FUL04330
   23 TP = TP+T(J,1)                                                    FUL04340
      TP=TP/FLOAT(MIXD(K)+1)                                            FUL04350
      TIN(K)=(TTIN(N,K)+TP*RMIX)/(1.0+RMIX)                             FUL04360
C     LOCATE ACTUAL LEVEL OF DAYS INPUT                                 FUL04370
      DO 27 I=1,JM                                                      FUL04380
      J = JM+1-I                                                        FUL04390
      IF(TIN(K)-T(J,1)) 27,30,30                                        FUL04400
```

```
   27 CONTINUE                                                          FULO4410
   30 JIN(K)=J+1                                                        FULO4420
      IF(JIN(K)-JM)32,32,33                                             FULO4430
   33 JIN(K)=JM                                                         FULO4440
   32 CONTINUE                                                          FULO4450
      QIN(K)=QQIN(N,K)*(1.0+RMIX)                                       FULO4460
   26 CONTINUE                                                          FULO4470
      GO TO 48                                                          FULO4480
   24 DO 25 K=1,NIN                                                     FULO4490
      TIN(K)=TTIN(N,K)                                                  FULO4500
   25 QIN(K)=QQIN(N,K)                                                  FULO4510
C     COMPUTATIONS WHEN SURFACE ELEVATION VARIES WITH TIME              FULO4520
   48 GO TO (45,31),KSUR                                                FULO4530
   31 JJM=JM                                                            FULO4540
      DO 331 K=1,NIN                                                    FULO4550
  331 CUMQIN=CUMQIN+QQIN(N,K)*DT                                        FULO4560
       DO 332 I=1,NOUT                                                  FULO4570
  332 CUMQOT=CUMQOT+QOUT(N,I)*DT                                        FULO4580
      QIO=CUMQIN-CUMQOT                                                 FULO4590
      IF(QIO) 34,34,35                                                  FULO4600
   35 SUM=-SAREA1*DYSUR1                                                FULO4610
      DO 36 M=1,JM                                                      FULO4620
      SUM=SUM+A(JM1+M-1)*DY                                             FULO4630
      IF(QIO-SUM) 37,37,36                                              FULO4640
   36 CONTINUE                                                          FULO4650
   34 SUM=DYSUR1*SAREA1                                                 FULO4660
      DO 38 M=1,JM                                                      FULO4670
      IF(ABS(QIO)-SUM) 39,39,38                                         FULO4680
   38 SUM=SUM+A(JM1-M)*DY                                               FULO4690
   37 YSUR=EL(JM1)+(M-0.5)*DY+(QIO-SUM)/A(JM1+M-1)                      FULO4700
      GO TO 40                                                          FULO4710
   39 YSUR=EL(JM1)-(M-0.5)*DY+(QIO+SUM)/A(JM1-M+1)                      FULO4720
   40 DYS=YSUR-EL(JM1)+DY/2.0                                           FULO4730
      IF(DYS) 41,42,42                                                  FULO4740
   42 M=IFIX(DYS/DY)                                                    FULO4750
      GO TO 43                                                          FULO4760
   41 M=IFIX(DYS/DY)-1                                                  FULO4770
   43 JM=JM1+M                                                          FULO4780
      DYSUR=YSUR-EL(JM)+DY/2.0                                          FULO4790
C     CALCULATE MEASURED SURFACE LEVEL                                  FULO4800
      R=ET/DSURF+1.0                                                    FULO4810
      L=R                                                               FULO4820
      RR=R-FLOAT(L)                                                     FULO4830
      SURMES=SURF(L)+RR*(SURF(L+1)-SURF(L))                             FULO4840
C     CALCULATE SURFACE AREA                                            FULO4850
      IF(YSUR-EL(JM)) 58,58,59                                          FULO4860
   58 AYSUR=A(JM)-(DY/2.0-DYSUR)*(A(JM)-A(JM-1))/DY                     FULO4870
      GO TO 61                                                          FULO4880
   59 AYSUR=A(JM)+(DYSUR-DY/2.0)*(A(JM+1)-A(JM))/DY                     FULO4890
   61 SAREA=(AYSUR+(A(JM)+A(JM-1))/2.0)/2.0                             FULO4900
C     CHECK ACCURACY OF SURFACE LEVEL                                   FULO4910
      SUMV=0.0                                                          FULO4920
      JMM=JM-1                                                          FULO4930
      IF(JM-JM1) 510,511,512                                            FULO4940
  512 DO 513 J=JM1,JMM                                                  FULO4950
```

```
  513 SUMV=SUMV+A(J)*DY                                                  FUL04960
  511 SUMV=SUMV+SAREA*DYSUR-SAREA1*DYSUR1                                FUL04970
      GO TO 515                                                          FUL04980
  510 JMM1=JM1-1                                                         FUL04990
      DO 514 J=JM,JMM1                                                   FUL05000
  514 SUMV=SUMV+A(J)*DY                                                  FUL05010
      SUMV=-(SUMV+SAREA1*DYSUR1-SAREA*DYSUR)                             FUL05020
  515 ERROR=SUMV-QIO                                                     FUL05030
      DYCOR=ERROR/SAREA                                                  FUL05040
      DYSUR=DYSUR-DYCOR                                                  FUL05050
      IF(DYSUR-0.5) 506,506,507                                          FUL05060
  506 DYSUR=DYSUR+DY                                                     FUL05070
      JM=JM-1                                                            FUL05080
  507 MM=JM-JJM                                                          FUL05090
      IF(MM)44,44,50                                                     FUL05100
   50 DO 51 I=1,MM                                                       FUL05110
      J=JM+1-I                                                           FUL05120
   51 T(J,1)=T(JJM,1)                                                    FUL05130
   44 T(JM,1)=T(JJM,1)                                                   FUL05140
      JMIXB=JM-MIXED                                                     FUL05150
      IF(JM-60) 53,53,54                                                 FUL05160
   53 JP=JM                                                              FUL05170
      GO TO 45                                                           FUL05180
   54 JP=60                                                              FUL05190
   45 CONTINUE                                                           FUL05200
C     SUB SPEED COMPUTES WITHDRAWAL THICKNESS AND VELOCITIES AT EACH     FUL05210
C     TIME STEP                                                          FUL05220
      CALL SPEED(N)                                                      FUL05230
C     STABILITY CHECK V*DT IS LESS THAN DY                               FUL05240
      VVV=ABS(V(2,1))                                                    FUL05250
      DO 501 J=3,JM                                                      FUL05260
      IF(VVV-ABS(V(J,1)))502,501,501                                     FUL05270
  502 VVV=ABS(V(J,1))                                                    FUL05280
  501 CONTINUE                                                           FUL05290
      VM=DY/DTT                                                          FUL05300
      IF(VVV-VM) 503,504,504                                             FUL05310
  504 DT=DY/VVV                                                          FUL05320
      IDT=DTT/DT+1                                                       FUL05330
      DT=DTT/IDT                                                         FUL05340
      GO TO 505                                                          FUL05350
  503 IDT=1                                                              FUL05360
  505 CONTINUE                                                           FUL05370
      DIFUS=DDRI*DT/DY/DY                                                FUL05380
      IDIF=DIFUS/0.4+1                                                   FUL05390
      DT=DT/IDIF                                                         FUL05400
      IDT=IDT*IDIF                                                       FUL05410
      DO 79 M=1,IDT                                                      FUL05420
C     HEAT TRANSPORT CALCULATIONS                                        FUL05430
      JMM=JM-1                                                           FUL05440
      DO 1114 J=2,JM                                                     FUL05450
      IF(EL(J).LT.ELBSL) GO TO 509                                       FUL05460
      T(J,2)=TCOOL(N,1)                                                  FUL05470
      GO TO 1114                                                         FUL05480
C     DIRECT ABSORPTION TERM                                             FUL05490
  509 ARJ1=(A(J)+A(J+1))/2.                                              FUL05500
```

```
      ARJ2=(A(J)+A(J-1))/2.0                                                FUL05510
      DELTA=(1.-BETA)*SR*.94*(EXP(-ETA*(YSUR-EL(J)-DY/2.))*ARJ1-            FUL05520
     &      EXP(-ETA*(YSUR-EL(J)+DY/2.0))*ARJ2)/A(J)/DY/HCAP/RHO            FUL05530
C     VERTICAL ADVECTION TERM                                               FUL05540
      IF(V(J,1)) 1160,1160,1161                                             FUL05550
 1160 IF(V(J+1,1))1170,1170,1171                                            FUL05560
 1170 DELTB=(V(J,1)*T(J,1)*(A(J)+A(J-1))/2.0-V(J+1,1)*T(J+1,1)*(A(J+1)+     FUL05570
     +      A(J))/2.0)/A(J)/DY                                             FUL05580
      GO TO 1162                                                            FUL05590
 1171 DELTB=(V(J,1)*T(J,1)*(A(J)+A(J-1))/2.0-V(J+1,1)*T(J,1)*(A(J+1)+      FUL05600
     +      A(J))/2.0)/A(J)/DY                                             FUL05610
      GO TO 1162                                                            FUL05620
 1161 IF(V(J+1,1))1172,1172,1173                                            FUL05630
 1173 DELTB=(V(J,1)*T(J-1,1)*(A(J)+A(J-1))/2.0-V(J+1,1)*T(J,1)*(A(J+1)+     FUL05640
     +      A(J))/2.0)/A(J)/DY                                             FUL05650
      GO TO 1162                                                            FUL05660
 1172 DELTB=(V(J,1)*T(J-1,1)*(A(J)+A(J-1))/2.0-V(J+1,1)*T(J+1,1)*(A(J+1)FUL05670
     +      +A(J))/2.0)/A(J)/DY                                            FUL05680
C     HORIZONTAL ADVECTION TERM                                             FUL05690
 1162 DELTC=-UO(J,1)*T(J,1)                                                 FUL05700
      DO 1175 K=1,NIN                                                       FUL05710
 1175 DELTC=DELTC+UI(J,K)*TIN(K)                                            FUL05720
      DELTC=DELTC/A(J)*B(J)                                                 FUL05730
      DELTD=DDRI*((T(J+1,1)-T(J,1))/DY*ARJ1-(T(J,1)-T(J-1,1))/DY*ARJ2)      FUL05740
     +      /A(J)/DY                                                       FUL05750
      DELT=(DELTA+DELTB+DELTC+DELTD)*DT                                     FUL05760
      T(J,2)=T(J,1)+DELT                                                    FUL05770
 1114 CONTINUE                                                              FUL05780
C     CALCULATIONS FOR BOTTOM HALF LAYER                                    FUL05790
      FLUXB=0.                                                              FUL05800
      SPVOL=RHO*HCAP*(A(1)*3.+A(2))/4.*DY/2.                                FUL05810
 1203 AAV=(A(1)+A(2))/2.                                                    FUL05820
      IF(V(2,1)) 1166,1167,1167                                             FUL05830
 1167 DELT1=DT*((1.-BETA)*SR*.94*EXP(-ETA*(YSUR-EL(1)-DY/2.))*AAV           FUL05840
     +      +DDRI*RHO*HCAP*(T(2,1)-T(1,1))*AAV/DY-FLUXB)/SPVOL             FUL05850
      GO TO 1168                                                            FUL05860
 1166 DELT1=DT*((1.-BETA)*SR*.94*EXP(-ETA*(YSUR-EL(1)-DY/2.))*AAV           FUL05870
     +      -V(2,1)*(A(2)+A(1))/2.0*(T(2,1)-T(1,1))                        FUL05880
     +      +DDRI*RHO*HCAP*(T(2,1)-T(1,1))*AAV/DY-FLUXB)/SPVOL             FUL05890
 1168 DO 1177 K=1,NIN                                                       FUL05900
 1177 DELT1=DELT1+UI(1,K)*B(1)*DY/2.0*(TIN(K)-T(1,1))*DT/A(1)/DY*2.0        FUL05910
      T(1,2)=T(1,1)+DELT1                                                   FUL05920
      DO 1178 J=1,JM                                                        FUL05930
 1178 T(J,1)=T(J,2)                                                         FUL05940
C     CHECK REASONABLENESS OF RESULTS                                       FUL05950
      IF (ABS(T(JM,2))-150.0) 63,57,57                                      FUL05960
   57 TSTOP = ET                                                            FUL05970
      GO TO 80                                                              FUL05980
C     SUBROUTINE AVER MIXES SURFACE LAYERS IN THE EVENT OF A SURFACE        FUL05990
C     INSTABILITY                                                           FUL06000
   63 CONTINUE                                                              FUL06010
      CALL AVER(N)                                                          FUL06020
   79 CONTINUE                                                              FUL06030
      DT=DTT                                                                FUL06040
      DO 78 I=1,NOUT                                                        FUL06050
```

```
      III=I                                                             FUL06060
      GO TO (115,116),KQ                                                FUL06070
  116 TOUTC(I)=0.0                                                      FUL06080
      GO TO 78                                                          FUL06090
  115 CALL TOUT(HEATOT,FLOWOT)                                          FUL06100
      TOUTC(I)=HEATOT/FLOWOT                                            FUL06110
   78 CONTINUE                                                          FUL06120
      GO TO (98,99),KITP                                                FUL06130
   98 CALL XINTKP(N)                                                    FUL06140
      GO TO 97                                                          FUL06150
   99 TINTK=TOUTC(1)                                                    FUL06160
C     ENERGY CHECK                                                      FUL06170
   97 E1=A(1)*DY/2.0*T(1,1)                                             FUL06180
      JMM=JM-1                                                          FUL06190
      DHL=DMR(N,1)                                                      FUL06200
      DO 111 J=2,JM                                                     FUL06210
      ELY=EL(J)                                                         FUL06220
      IF(ELY.GT.ELBSL) GO TO 114                                        FUL06230
  111 E1=E1+A(J)*DY*T(J,1)                                              FUL06240
  114 DC=DHL                                                            FUL06250
      IF(DHL.GT.DCH) DC=DCH                                             FUL06260
      ENSUR=ACH*DC*(TTIN(N,1)+TEXIT(N,1))/2.                            FUL06270
      DAMX=AS(1,1)*AYSUR-ACH                                            FUL06280
      ENSUR=ENSUR+DAMX*(TEXIT(N,1)+TSUR(1,1))/2.*DHL                    FUL06290
      ENSUR=ENSUR+DAHL*(THL(1,1)+THL(NH,1))*DHL/2.                      FUL06300
      NHL=NH-1                                                          FUL06310
      DO 117 I=2,NHL                                                    FUL06320
  117 ENSUR=ENSUR+DAHL*DHL*THL(I,1)                                     FUL06330
      E1=E1+ENSUR                                                       FUL06340
      E1=E1*RHO*HCAP                                                    FUL06350
      E2=E2+ SR*AYSUR*DT                                                FUL06360
      DO 113 K=1,NIN                                                    FUL06370
  113 E2=E2+QQIN(N,K)*DT*TTIN(N,K)*RHO*HCAP                             FUL06380
      E3=E3+FLUXT+FLUXB*DT                                              FUL06390
      DO 112 I=1,NOUT                                                   FUL06400
  112 E3=E3+QOUT(N,I)*DT*TOUTC(I)*RHO*HCAP                              FUL06410
      ENRAT=(E1-EO)/(E2-E3)                                             FUL06420
      TENRAT=(E1+E3)/(E2+EO)                                            FUL06430
      NDAY1=0                                                           FUL06440
  321 NDAYP=N                                                           FUL06450
      QSA1(2)=QSA1(2)/86400.0                                           FUL06460
      QSA2(2)=QSA2(2)/86400.0                                           FUL06470
      QSA1(3)=QSA1(3)/86400.0                                           FUL06480
      QSA2(3)=QSA2(3)/86400.0                                           FUL06490
      TINIT=TTIN(N,1)                                                   FUL06500
      KYR=MYR                                                           FUL06510
      IF(ISELCT(12).GT.0)WRITE(11,1060) KYR,NDAYP,TINIT,TEND(2),TEND(3),FUL06520
     +     TSUR(1,1),THL(40,1),THL(9,1),DMR(N,1),T(11,1),T(1,1)         FUL06530
 1060 FORMAT (1X,I4,I4,9F7.2)                                           FUL06540
      CALL TIMDAX(ATIME,MDAY,MMO,MYR)                                   FUL06550
      MYR = MYR-1900                                                    FUL06560
      IF(ISELCT(9).GT.0)GO TO 80                                        FUL06570
         DO 313 K=1,40                                                  FUL06580
         IF(IPDAT(1,K).EQ.MMO.AND.IPDAT(2,K).EQ.MDAY.AND.               FUL06590
     +        IPDAT(3,K).EQ.MYR)GO TO 80                                FUL06600
```

```
 313     CONTINUE                                                        FUL06610
       GO TO 280                                                         FUL06620
    80 NCNT=1                                                            FUL06630
C      NORTH ANNA COOLING POND SIMULATION-APRIL 1957 TO APRIL 1967       FUL06640
    81 WRITE(6,912) ET,N,YSUR,MMO,MDAY,MYR                               FUL06650
       W=WINDY                                                           FUL06660
       WRITE(6,915) PSI,TAIR,SR,W,CC                                     FUL06670
       F5=2.*HAFDEL                                                      FUL06680
       DO 96 K=1,NIN                                                     FUL06690
       JIK=JIN(K)                                                        FUL06700
       SIGMAI=SIGMIN(K)                                                  FUL06710
       WQQIN=QQIN(N,K)                                                   FUL06720
       WTTIN=TTIN(N,K)                                                   FUL06730
       WRITE(6,911) SIGMAI,WQQIN,WTTIN                                   FUL06740
    96 WRITE(6,922) TIN(K),QIN(K),JIN(K)                                 FUL06750
       DO 88 I=1,NOUT                                                    FUL06760
       IF(I.EQ.2) F5=THICKB                                              FUL06770
       WQOUT=QOUT(N,I)                                                   FUL06780
    88 WRITE(6,917) ELOUT(I),WQOUT,F5,TOUTC(I)                           FUL06790
       WRITE(6,926) ENRAT,E1,E2,E3                                       FUL06800
       WRITE(6,924) TENRAT,TEQ,FLUXB                                     FUL06810
       WRITE(6,942) TINIT,TEXIT(N,1),TMR(N,1),TCOOL(N,1)                 FUL06820
       WRITE(6,943) QCH,QEXIT(N,1),QMR(N,1),QCOOL(N,1),F2,WL             FUL06830
       CALL PRWHTF(N,MMO,MDAY,MYR)                                       FUL06840
       WRITE(6,1010)                                                     FUL06850
       WRITE(6,947)                                                      FUL06860
       ACN=ACH/AYSUR                                                     FUL06870
       WRITE (6,948) ACN,TTEXIT(N,1),TEXSTR                              FUL06880
       WRITE(6,1011)                                                     FUL06890
       WRITE(6,1050)                                                     FUL06900
       WRITE(6,1051) TTEXIT(N,1),TEND(1),TFIN(N,1),TSA1(2),TSA2(2),      FUL06910
      +              TEND(2),TFIN(N,2),TSA1(3),TSA2(3),TEND(3)          FUL06920
       WRITE(6,1052) FROUDE(1),FROUDE(2),FROUDE(3)                       FUL06930
       WRITE(6,1055) DISDIL(1),DISDIL(2),DISDIL(3)                       FUL06940
       WRITE(6,1053) ULAY1(1),ULAY4(1),ULAY1(2),ULAY2(2),ULAY3(2),       FUL06950
      +              ULAY4(2),ULAY1(3),ULAY2(3),ULAY3(3),ULAY4(3)       FUL06960
       WRITE(6,1054)QSA1(2),QSA2(2),QSA1(3),QSA2(3)                      FUL06970
       WRITE(6,1012)                                                     FUL06980
       WRITE(6,949)                                                      FUL06990
       NDI=1                                                             FUL07000
       NN=(NDI+1)/2                                                      FUL07010
       DO 990 I=1,NN                                                     FUL07020
   990 WRITE(6,944)(DIL(J),AS(J,1),TSTAR(J,1),TSUR(J,1),J=I,NDI,5)       FUL07030
       WRITE(6,1015) DMR(N,1)                                            FUL07040
       WRITE(6,1013)                                                     FUL07050
       WRITE(6,945)                                                      FUL07060
       DO 995 I=1,10                                                     FUL07070
   995 WRITE(6,946)(AH(J),THL(J,1),THSTAR(J,1),J=I,NH,10)                FUL07080
       IF(KSA.EQ.0) GO TO 253                                            FUL07090
       WRITE(6,243)                                                      FUL07100
   243 FORMAT(//45X,'SIDEARMS',/45X,8('*'))                              FUL07110
       WRITE(6,242)                                                      FUL07120
   242 FORMAT(/5X,'SIDEARM NUMBER',10X,'FLOW',10X,'RETURN TEMP',10X,     FUL07130
      +       'ENTRANCE TEMP')                                           FUL07140
       DO 244 ISA=1,KSA                                                  FUL07150
```

```
      XXQSA=XQSA(ISA)/86400.                                            FUL07160
      WRITE(6,245) ISA,XXQSA,TRSA(ISA),ETSA(ISA)                        FUL07170
  245 FORMAT(11X,I5,10X,F10.1,12X,F5.1,18X,F5.1)                        FUL07180
  244 CONTINUE                                                          FUL07190
  253 CONTINUE                                                          FUL07200
   89 WRITE(6,1014)                                                     FUL07210
      WRITE(6,920)                                                      FUL07220
      DO 90 I=1,10                                                      FUL07230
   90 WRITE (6,921) (J,EL(J),T(J,1),J=I,JP,10)                          FUL07240
      WRITE(6,1001)                                                     FUL07250
      IF(JM-60) 280,280,91                                              FUL07260
   91 WRITE (6,920)                                                     FUL07270
      IF(JM-70) 92,93,93                                                FUL07280
   92 LL = JM                                                           FUL07290
      GO TO 94                                                          FUL07300
   93 LL=70                                                             FUL07310
   94 DO 95 I=61,LL                                                     FUL07320
   95 WRITE (6,921) (J,EL(J),T(J,1),J=I,JM,10)                          FUL07330
      WRITE(6,1001)                                                     FUL07340
  280 CONTINUE                                                          FUL07350
      IF(NCNT.GT.1) GO TO 282                                           FUL07360
      NCNT=NCNT+1                                                       FUL07370
      WRITE(6,281)                                                      FUL07380
  281 FORMAT(6X,'DATE',8X,'TINIT',3X,'TEND(1)',6X,'QSA1',6X,            FUL07390
     +       'QSA2',3X,'TEND(2)',3X,'TEND(3)',6X,'TSUR',4X,'THL(9)',3X, FUL07400
     +       'THL(40)',6X,'QSA',6X,'TRSA',4X,'TEQ'/)                   FUL07410
  282 CONTINUE                                                          FUL07420
      XQCFS=XQSA(1)/86400.                                              FUL07430
      WRITE (6,322) IWDFLG,MMO,MDAY,MYR,TINIT,TFIN(N,1),QSA1(2),QSA2(2),FUL07440
     +    TEND(2),TEND(3),TSUR(1,1),THL(9,1),THL(40,1),XQCFS,TRSA(1),TEQFUL07450
  322 FORMAT(I2,3X,I2,'/',I2,'/',I2,4X,F6.2,4X,F6.2,4X,F6.1,4X,F6.1,    FUL07460
     +       5(4X,F6.2),4X,F6.0,3X,F6.1,3X,F6.2)                        FUL07470
C     THIS SECTION WRITES SOME OF THE VITAL OUTPUT TO A DISK FILE FOR    FUL07480 → RUN 1893
C     LATER USE ON FILE NUMBER 50 DEFINED IN FULLPOND EXEC               FUL07490
      WRITE(50,1099)MMO,MDAY,MYR,TINIT,TTEXIT(N,1),TEND(2),TEND(3),TSUR(FUL07500
     +              1,1),THL(9,1),THL(40,1),T(3,1),T(4,1),T(5,1),TEQ,DDRIFUL07510
 1099 FORMAT (3I2,11(F6.2),E8.3)                                        FUL07520
      WRITE(51,1098)MMO,MDAY,MYR,TSUR(1,1),THL(3,1),THL(5,1),THL(7,1),  FUL07530
     +      THL(9,1),THL(15,1),THL(20,1),THL(25,1),THL(30,1),THL(40,1), FUL07540
     +      TRSA(1),TEQ,TTEQ                                           FUL07550
 1098 FORMAT(3I2,13(F5.2))                                              FUL07560
C     COMPUTE DAY NO. RELATIVE TO FIRST OF YEAR                         FUL07570
      DAZE=ATIME-ATIMEX                                                 FUL07580
C     DECIDE ON WATER LOSS PRINTOUT INTERVAL                            FUL07590
      IF(ISELCT(14).EQ.0)GO TO 100                                      FUL07600
      IF(ISELCT(14).EQ.2)GO TO 283                                      FUL07610
      MMO1=MMO+1                                                        FUL07620
      IF(DAZE.LT.NDAYS(MMO1))GO TO 100                                  FUL07630
  283 WRITE(IRITE,498)WLT,WLTWH,WLTN                                    FUL07640
  100 CONTINUE                                                          FUL07650
  323 ATIME=ATIME+1.0                                                   FUL07660
    1 CONTINUE                                                          FUL07670
      XTAVG2=XTAVG/365.                                                 FUL07680
      WRITE(6,799) XTAVG2                                               FUL07690
  799 FORMAT(/5X,'THE YEARLY AVERAGE OF TI-TE IS',2X,F6.2)              FUL07700
```

```
      WRITE(6,498)WLT,WLTWH,WLTN                                        FULO7710
  498 FORMAT(5X,'THE TOTAL WATER LOSS WAS ',F8.0,' AC-FT IN MAIN POND'/ FULO7720
     +       26X,'AND ',F8.0,' AC-FT IN THE WHTF'/                      FULO7730
     +       5X,'THE NATURAL WATER LOSS RATE IS APPROXIMATELY ',F6.2,   FULO7740
     +       ' FT')                                                     FULO7750
  899 FORMAT (20I4)                                                     FULO7760
  900 FORMAT (20A4)                                                     FULO7770
  901 FORMAT (16I5)                                                     FULO7780
  902 FORMAT (8F10.5)                                                   FULO7790
  903 FORMAT(///////,2X,'TIME      TOUT      TSUR      TINTK')          FULO7800
  904 FORMAT(' NUMBER OF GRID POINTS=',I3,17X,'SURFACE ELEVATION=',     FULO7810
     +       F7.2,18X,'DENSITY=',E12.5)                                 FULO7820
  905 FORMAT (' OUTLET LEVEL=',I3,  26X,'OUTLET ELEVATION=',F8.2)       FULO7830
  906 FORMAT (' DY=',F6.2,33X,'BOTTOM ELEVATION=',F8.2,18X,'ETA=',F6.3) FULO7840
  907 FORMAT (' DT=',F6.2,33X,'INITIAL TEMPERATURE=',F6.2,17X,'BETA=',  FULO7850
     +       F5.2)                                                      FULO7860
  908 FORMAT(' CHANNEL AREA=',F10.2,10X,'CHANNEL LENGTH=',F10.2,10X,    FULO7870
     +       'EXIT DEPTH=',F10.2,10X,'EXIT WIDTH=',F10.2)               FULO7880
  909 FORMAT (' STOP AT TIME=',F7.2)                                    FULO7890
  910 FORMAT (' KITP  =',I2,15X,'KSUR=',I2,13X,'KOH =',I2,15X,'KQ=',    FULO7900
     +       I2,13X,'KLOSS=',I2,13X,'KAREA=',I2,10X,'KMIX=',I2)        FULO7910
  911 FORMAT (' INFLOW STD. DEV.=',F10.5,15X,'INFLOW RATE=',F12.1,19X,' FULO7920
     +       INFLOW TEMPERATURE =',F6.2,'F')                            FULO7930
  912 FORMAT ('1  ELAPSED TIME =',F7.2,22X,'NO. OF TIME STEPS =',I4,10X,FULO7940
     + 'SURF. ELEV.=',F9.2,/,3X,21('*'),/7X,'DATE',1X,I2,'/',I2,'/',I2/)FULO7950
  913 FORMAT(10(I2,1X,I2,1X,I2))                                        FULO7960
  915 FORMAT(' RELATIVE HUMIDITY =',F5.2, 5X,'AIR TEMP =',F6.2, 5X,     FULO7970
     +       'INSOLATION FLUX=',E12.5, 5X,'WIND SPEED=',F6.2,'MPH',5X,  FULO7980
     +       'CLOUD COVER=',F5.2)                                       FULO7990
  917 FORMAT (' OUTLET ELEVATION=',F10.5, 5X,' OUTFLOW RATE=',F12.1, 5X,FULO8000
     +       'WITHDR. THICKNESS=',F7.2,10X,' OUTFLOW TEMP=',F6.2,'F')   FULO8010
  920 FORMAT (/5('   J   ELEV    TEMP(F)  '))                           FULO8020
  921 FORMAT (5(I5,F10.3,F7.3,3X))                                      FULO8030
  922 FORMAT(' MIXED INFLOW TEMP.=',F6.2,15X,'MIXED INFLOW RATE=',F12.1 FULO8040
     +       ,15X,'LEVEL OF INFLOW=',I5)                                FULO8050
  923 FORMAT (' NO. GRID SPACES IN MIXED LAYER=',I3,8X,'MIXING RATIO='  FULO8060
     +       ,F5.2)                                                     FULO8070
  924 FORMAT (' TOTAL ENERGY RATIO=',F9.5,10X,'EQUIL. TEMP.=',F9.3,10X, FULO8080
     +       'BOTTOM FLUX=',F10.2)                                      FULO8090
  926 FORMAT(' ENERGY RATIO=',F7.3,5X,'ENERGY STORED=',E12.5,5X,'ENERGY FULO8100
     +       INFLOW=',E12.5, 3X,'ENERGY OUTFLOW=',E12.5)                FULO8110
  930 FORMAT (' INITIAL STORED ENERGY=',E12.5)                          FULO8120
  942 FORMAT (' TINIT=',F7.2,2X,'TEX=',F8.3,2X,'TMXG=',F8.3,2X,'TCOLD=' FULO8130
     +       ,F8.3)                                                     FULO8140
  943 FORMAT (' QCH=',F14.2,2X,'QEX=',F14.2,2X,'QMX=',F14.2,2X,'QHL=',  FULO8150
     +       F14.2,2X,'FR=',F6.2,2X,'WEDL=',F7.2,2X)                    FULO8160
  944 FORMAT(16X,2(F7.3,1X,F7.2,1X,F7.4,1X,F7.3,8X))                    FULO8170
  945 FORMAT (/,4('   AREA   TEMP    TSTR',10X))                        FULO8180
  946 FORMAT (4(F7.2,1X,F7.2,1X,F7.3,9X))                               FULO8190
  947 FORMAT (/(' AREA   TEMP   TSTR'))                                 FULO8200
  948 FORMAT(F7.2,1X,F7.2,1X,F7.3)                                      FULO8210
  949 FORMAT(/,16X,2('DILUTION   AREA  TSTR    TEMP',9X))               FULO8220
  961 FORMAT (10F8.1)                                                   FULO8230
  975 FORMAT(6F12.1)                                                    FULO8240
  981 FORMAT (8F10.2)                                                   FULO8250
```

```
 982 FORMAT(I10,6F10.0,I10)                                              FUL08260
 983 FORMAT(7F10.2)                                                      FUL08270
 984 FORMAT(5X,'RUN OF ',A8,5X,'AT ',A8//)                               FUL08280
1000 FORMAT (20A4,//)                                                    FUL08290
1001 FORMAT(///////)                                                     FUL08300
1002 FORMAT('1')                                                         FUL08310
1010 FORMAT (//,40X,'DISCHARGE CHANNEL',/,40X,17('*'))                   FUL08320
1011 FORMAT (//,38X,'WASTE HEAT TREATMENT FACILITY',/,38X,29('*'))       FUL08330
1012 FORMAT(//,41X,'JET MIXING ZONE',/,41X,15('*'))                      FUL08340
1013 FORMAT(//,40X,'HEAT LOSS REGION',/,40X,16('*'))                     FUL08350
1014 FORMAT(//,39X,'DEEP RESERVOIR REGION',/,39X,21('*'))                FUL08360
1015 FORMAT(/,1X,'UPPER LAYER DEPTH AT THE END OF THE JET REGION =',     FUL08370
     +       F10.2,'  FT')                                              FUL08380
1050 FORMAT (/,33X,'REACH 1',22X,'REACH 2',31X,'REACH 3',//,29X,         FUL08390
     +       'ENTRANCE  END',2(8X,'ENTRANCE  S.ARM 1  S.ARM 2  END'))   FUL08400
1051 FORMAT (1X,'TEMPERATURE  (DEG F)',8X,F6.2,3X,F6.2,2(6X,F6.2,3X,     FUL08410
     +       F6.2,3X,F6.2,3X,F6.2))                                      FUL08420
1052 FORMAT (1X,'FROUDE NUMBER',16X,F5.2,16X,F5.2,34X,F5.2)              FUL08430
1053 FORMAT (1X,'UPPER LAYER DEPTH (FT)',7X,F5.2,4X,F5.2,2(7X,F5.2,      FUL08440
     +       4X,F5.2,4X,F5.2,4X,F5.2))                                   FUL08450
1054 FORMAT (1X,'SIDE ARM FLOW   (CFS)',14X,2(24X,F6.1,3X,F6.1))         FUL08460
1055 FORMAT (1X,'DILUTION',21X,F5.2,16X,F5.2,34X,F5.2)                   FUL08470
      STOP                                                               FUL08480
      END                                                                FUL08490
```

```
      SUBROUTINE WLOSS(EVAP,A,DELT,WLT)                                 FSU00010
C                                                                       FSU00020
C     SUBROUTINE TO CALCULATE WATER LOSS DUE TO EVAPORATION FROM        FSU00030
C     AREA A DURING TIME PERIOD DELT                                    FSU00040
C                                                                       FSU00050
C     EVAP IN BTU/FT**2 DAY                                             FSU00060
C     A IN FT**2                                                        FSU00070
C     DELT IN DAYS                                                      FSU00080
C     WL AND WLT IN ACRE-FT                                             FSU00090
C                                                                       FSU00100
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00110
      IBUG = IDEBUG(11)                                                 FSU00120
      IF(EVAP.LT.0.0)RETURN                                             FSU00130
      WL = EVAP*A*3.471E-10*DELT                                        FSU00140
      WLT = WLT + WL                                                    FSU00150
      IF(IBUG.GT.2)WRITE(20,1000)EVAP,A,WL                              FSU00160
 1000 FORMAT(' *** WLOSS-DEBUG *** EVAP,A WL = ',3E12.6)                FSU00170
      RETURN                                                            FSU00180
      END                                                               FSU00190
      SUBROUTINE GETMET(METAPE,ISPER,NMET,DTMET,NOUT,DT,DOUT,DATA)      FSU00200
C     READS INPUT METEOROLOGY AND REFORMATS FOR RUN                     FSU00210
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00220
      COMMON /IOPUT/ IREAD,IRITE                                        FSU00230
      DIMENSION DATA(NMET), DOUT(NOUT)                                  FSU00240
      IBUG = IDEBUG(13)                                                 FSU00250
      IRBUG = 20                                                        FSU00260
      IRITE = 6                                                         FSU00270
      READ(METAPE,20)(DATA(I),I=1,NMET)                                 FSU00280
C     CHECK TIME SCALES                                                 FSU00290
      TSTOP = DT*FLOAT(NOUT)                                            FSU00300
      TIME = DTMET*FLOAT(NMET)                                          FSU00310
      IF(IBUG.GT.1)WRITE(IRBUG,3001)DT,DTMET,TSTOP,TIME,NOUT,NMET       FSU00320
      IF(ABS(DT-DTMET).LT.0.0005)GO TO 200                              FSU00330
      IF(DTMET.LT.DT)GO TO 300                                          FSU00340
C                                                                       FSU00350
C     MET DATA USES LONGER TIME STEP THAN SIMULATION                    FSU00360
C                                                                       FSU00370
C        CHECK IF DTMET IS INTEGRAL MULTIPLE OF DT                      FSU00380
  100 CONTINUE                                                          FSU00390
      REM = AMOD(DTMET,DT)                                              FSU00400
      IF(IBUG.GT.1)WRITE(IRBUG,3000)REM                                 FSU00410
      IF(REM.GT.0.0005) GO TO 600                                       FSU00420
      NRATIO = (DTMET+0.0005)/DT                                        FSU00430
      IF(IBUG.GT.2)WRITE(IRBUG,3004)NRATIO                              FSU00440
      N = 0                                                             FSU00450
           DO 180 I=1,NMET                                              FSU00460
              DO 160 J=1,NRATIO                                         FSU00470
              N = N+1                                                   FSU00480
              DOUT(N) = DATA(I)                                         FSU00490
              IF(N.GE.NOUT)GO TO 800                                    FSU00500
  160         CONTINUE                                                  FSU00510
  180      CONTINUE                                                     FSU00520
      GO TO 800                                                         FSU00530
C                                                                       FSU00540
C     MET DATA AND SIMULATION USE SAME TIME STEP                        FSU00550
```

```
C                                                                       FSU00560
 200  CONTINUE                                                          FSU00570
      IF(IBUG.GT.1)WRITE(IRBUG,3002)                                    FSU00580
           DO 250 I=1,NOUT                                              FSU00590
           DOUT(I) = DATA(I)                                            FSU00600
 250       CONTINUE                                                     FSU00610
      GO TO 800                                                         FSU00620
C                                                                       FSU00630
C     SIMULATION TIME STEP IS LONGER THAN MET DATA INTERVAL             FSU00640
C                                                                       FSU00650
 300  CONTINUE                                                          FSU00660
      REM = AMOD(DT,DTMET)                                              FSU00670
      IF(IBUG.GT.1)WRITE(IRBUG,3003)REM                                 FSU00680
      IF(REM.GT.0.0005) GO TO 500                                       FSU00690
      NRATIO = (DT+0.0005)/DTMET                                        FSU00700
      IF(IBUG.GT.2)WRITE(IRBUG,3004)NRATIO                              FSU00710
      N = 0                                                             FSU00720
           DO 380 I=1,NOUT                                              FSU00730
           DOUT(I) = 0.0                                                FSU00740
              DO 360 J=1,NRATIO                                         FSU00750
              N = N+1                                                   FSU00760
              DOUT(I) = DOUT(I)+DATA(N)                                 FSU00770
 360          CONTINUE                                                  FSU00780
 380       CONTINUE                                                     FSU00790
      GO TO 800                                                         FSU00800
C                                                                       FSU00810
C     DT AND DTMET ARE NOT INTEGRALLY RELATED                           FSU00820
C                                                                       FSU00830
C     DT GREATER THAN DTMET                                             FSU00840
 500  CONTINUE                                                          FSU00850
      WRITE(IRITE,2000)                                                 FSU00860
      CALL EXIT                                                         FSU00870
C     DTMET GREATER THAN DT                                             FSU00880
 600  N=1                                                               FSU00890
      ET = DT                                                           FSU00900
      ETM = DT                                                          FSU00910
           DO 680 I=2,NMET                                              FSU00920
           ETML = ETM                                                   FSU00930
           ETM = ETML + DTMET                                           FSU00940
           DELD = DATA(I)-DATA(I-1)                                     FSU00950
 620       RR = (ET-ETML)/DTMET                                         FSU00960
           DOUT(N) = DATA(I-1) + RR*DELD                                FSU00970
           IF(N.GE.NOUT)GO TO 800                                       FSU00980
           N = N+1                                                      FSU00990
           ET = ET + DT                                                 FSU01000
           IF(ET.LT.ETM)GO TO 620                                       FSU01010
           IF(ABS(ET-ETM).GT.0.001)GO TO 680                            FSU01020
           DOUT(N) = DATA(I)                                            FSU01030
           IF(N.GE.NOUT)GO TO 800                                       FSU01040
           N = N+1                                                      FSU01050
           ET = ET + DT                                                 FSU01060
 680       CONTINUE                                                     FSU01070
      IF(IBUG.GT.1)WRITE(IRBUG,3005)N                                   FSU01080
C                                                                       FSU01090
C     DATA SMOOTHING                                                    FSU01100
```

```
C                                                                       FSU01110
 800  IF(ISPER.LT.2)RETURN                                              FSU01120
      SPER = FLOAT(ISPER)                                               FSU01130
      IF(IBUG.GT.3)WRITE(IRBUG,3006)(DOUT(I),I=1,20)                    FSU01140
C          WORK BACKWARDS THROUGH DATA TO COMPUTE MOVING AVERAGES        FSU01150
           DO 850 I=ISPER,NOUT                                          FSU01160
           II = NOUT+ISPER-I                                            FSU01170
           JEND = II-1                                                  FSU01180
           JSTRT = II-ISPER + 1                                         FSU01190
              DO 830 J=JSTRT,JEND                                       FSU01200
              DOUT(II) = DOUT(II)+DOUT(J)                               FSU01210
 830          CONTINUE                                                  FSU01220
           DOUT(II) = DOUT(II)/SPER                                     FSU01230
 850       CONTINUE                                                     FSU01240
      ISPER1 = ISPER-1                                                  FSU01250
           DO 890 I=2,ISPER1                                            FSU01260
           II = ISPER1+2-I                                              FSU01270
           JEND = II-1                                                  FSU01280
           K = 1                                                        FSU01290
              DO 880    J=1,JEND                                        FSU01300
              DOUT(II) = DOUT(II) + DOUT(J)                             FSU01310
              K = K+1                                                   FSU01320
 880          CONTINUE                                                  FSU01330
           DOUT(II) = DOUT(II)/FLOAT(K)                                 FSU01340
 890       CONTINUE                                                     FSU01350
      IF(IBUG.GT.3)WRITE(IRBUG,3007)(DOUT(I),I=1,20)                    FSU01360
      RETURN                                                            FSU01370
C                                                                       FSU01380
C     ERROR EXIT                                                        FSU01390
 900. WRITE(IRITE,901)                                                  FSU01400
      CALL EXIT                                                         FSU01410
      RETURN                                                            FSU01420
C     FORMAT STATEMENTS                                                 FSU01430
 20   FORMAT(8F10.2)                                                    FSU01440
 901  FORMAT(' *** GETMET *** ERROR - INSUFFICIENT MET DAT SUPPLIED FOR FSU01450
     +RUN')                                                             FSU01460
 2000 FORMAT(' *** GETMET *** ERROR IN TIME INTERVALS - MET DATA AND SIMFSU01470
     +ULATION TIME STEP ARE NOT INTEGRALLY RELATED')                    FSU01480
 3000 FORMAT(' *** GETMET-DEBUG *** AT LINE 100, REM = ',E15.8)         FSU01490
 3001 FORMAT(' *** GETMET-DEBUG *** DT,DTMET,TSTOP,TIME,NOUT,NMET = ',  FSU01500
     +      4E15.6,2I8)                                                FSU01510
 3002 FORMAT(' *** GETMET-DEBUG *** AT LINE 200')                       FSU01520
 3003 FORMAT(' *** GETMET-DEBUG *** AT LINE 300, REM = ',E15.8)         FSU01530
 3004 FORMAT(' *** GETMET-DEBUG *** NRATIO = ',I10)                     FSU01540
 3005 FORMAT(' *** GETMET-DEBUG *** AFTER 680 LOOP - N = ',I8)          FSU01550
 3006 FORMAT(' *** GETMET-DEBUG *** BEFORE SMOOTHING, DOUT(1) TO (20) ='FSU01560
     +      ,(/5X,8F12.3))                                             FSU01570
 3007 FORMAT(' *** GETMET-DEBUG ***  AFTER SMOOTHING, DOUT(1) TO (20) ='FSU01580
     +      ,(/5X,8F12.3))                                             FSU01590
      END                                                               FSU01600
      FUNCTION LEAP(IYEAR)                                              FSU01610
C                                                                       FSU01620
C     COMPUTES LEAP YEAR DAY CORRECTION                                 FSU01630
C                                                                       FSU01640
      LEAP=0                                                            FSU01650
```

```
      IF( MOD(IYEAR,4).EQ.0)LEAP=1                                      FSU01660
      IF( MOD(IYEAR,100).EQ.0)LEAP=0                                    FSU01670
      IF( MOD(IYEAR,400).EQ.0)LEAP=1                                    FSU01680
      RETURN                                                            FSU01690
      END                                                               FSU01700
      SUBROUTINE DATIME(DATE,TIME)                                      FSU01710
C                                                                       FSU01720
C     ROUTINE TO RETURN DATE AND TIME IN REAL*8 FORM TO THE             FSU01730
C     CALLING PROGRAM - USES THE IBM 'TIME' MACRO                       FSU01740
C                                                                       FSU01750
      DIMENSION A(5)                                                    FSU01760
      DOUBLE PRECISION DATE,TIME,DUM1,DUM2                              FSU01770
      EQUIVALENCE (DUM1,A(1)),(DUM2,A(3))                               FSU01780
C     CALL IBM DATE AND TIME MACRO                                      FSU01790
      CALL WHEN(A)                                                      FSU01800
      DATE = DUM1                                                       FSU01810
      TIME = DUM2                                                       FSU01820
      RETURN                                                            FSU01830
      END                                                               FSU01840
      SUBROUTINE DAXIME(IDAY,MONTH,IYEAR,ANDAYS)                        FSU01850
C                                                                       FSU01860
C     SUBROUTINE TO CONVERT DATE INTO ABSOLUTE DAYS SINCE 1/1/1800      FSU01870
C                                                                       FSU01880
      COMMON /NDY/ NDAYS(13)                                            FSU01890
      JYEAR = IYEAR-1800                                                FSU01900
      NTDAY = JYEAR*365.25 + NDAYS(MONTH) + IDAY-1                      FSU01910
      IF(NTDAY.GT.36524)NTDAY = NTDAY-1                                 FSU01920
      IF(MONTH.LT.3)NTDAY = NTDAY-LEAP(IYEAR)                           FSU01930
      ANDAYS = NTDAY                                                    FSU01940
      RETURN                                                            FSU01950
      END                                                               FSU01960
      SUBROUTINE TIMDAX(ANDAYS,IDAY,MONTH,IYEAR)                        FSU01970
C                                                                       FSU01980
C     SUBROUTINE CONVERTS ABSOLUTE DAY SINCE 1/1/1800 TO DATE           FSU01990
C                                                                       FSU02000
      COMMON /NDY/ NDAYS(13)                                            FSU02010
      IF(ANDAYS.GT.365.0)GO TO 5                                        FSU02020
      IYEAR = 1800                                                      FSU02030
      IDAY = ANDAYS                                                     FSU02040
      ILEAP = 0                                                         FSU02050
      GO TO 9                                                           FSU02060
 5    AN = ANDAYS-365.0                                                 FSU02070
      IF(ANDAYS.GT.36889.0)AN = AN+1                                    FSU02080
      IYEAR = AN/365.25                                                 FSU02090
      IC = 365.25*IYEAR                                                 FSU02100
      IDAY = AN-IC                                                      FSU02110
      IF (IDAY.LT.1)GO TO 7                                             FSU02120
      IYEAR = IYEAR+1801                                                FSU02130
      ILEAP = LEAP(IYEAR)                                               FSU02140
      GO TO 9                                                           FSU02150
 7    IYEAR = IYEAR+1800                                                FSU02160
      MONTH = 12                                                        FSU02170
      IDAY = 31                                                         FSU02180
      RETURN                                                            FSU02190
 9    J = 2                                                             FSU02200
```

```
      IF(IDAY.LT.NDAYS(2))GO TO 20                                      FSU02210
           DO 10 J=3,12                                                 FSU02220
           IF(IDAY.LT.NDAYS(J)+ILEAP) GO TO 20                          FSU02230
10         CONTINUE                                                     FSU02240
      J = 13                                                            FSU02250
20    MONTH = J-1                                                       FSU02260
      IDAY = IDAY-NDAYS(MONTH)+1                                        FSU02270
      IF(MONTH.GT.2)IDAY = IDAY-ILEAP                                   FSU02280
      RETURN                                                            FSU02290
      END                                                               FSU02300
      BLOCK DATA                                                        FSU02310
      COMMON /NDY/ NDAYS(13)                                            FSU02320
      DATA NDAYS/0,31,59,90,120,151,181,212,243,273,304,334,365/        FSU02330
      END                                                               FSU02340
      SUBROUTINE ZERO                                                   FSU02350
      COMMON /SUMS/ STEQ,SQI1,STIN,STF1,STS1,SQS12,SQS22,STF2,          FSU02360
     +          STSA12,STRS12,STSA22,STRS22,STS2,SQS13,SQS23,STF3,      FSU02370
     +          STSA13,STRS13,STSA23,STRS23,STS3,NMO                    FSU02380
C                                                                       FSU02390
C     ZEROS SUMMATION ARRAYS FOR MAIN ROUTINE                           FSU02400
C                                                                       FSU02410
      NMO = 0                                                           FSU02420
      STEQ = 0.0                                                        FSU02430
      SQI1 = 0.0                                                        FSU02440
      STIN = 0.0                                                        FSU02450
      STF1 = 0.0                                                        FSU02460
      STS1 = 0.0                                                        FSU02470
      SQS12 = 0.0                                                       FSU02480
      SQS22 = 0.0                                                       FSU02490
      STF2 = 0.0                                                        FSU02500
      STSA12 = 0.0                                                      FSU02510
      STRS12 = 0.0                                                      FSU02520
      STSA22 = 0.0                                                      FSU02530
      STRS22 = 0.0                                                      FSU02540
      STS2 = 0.0                                                        FSU02550
      SQS13 = 0.0                                                       FSU02560
      SQS23 = 0.0                                                       FSU02570
      STF3 = 0.0                                                        FSU02580
      STSA13 = 0.0                                                      FSU02590
      STRS13 = 0.0                                                      FSU02600
      STSA23 = 0.0                                                      FSU02610
      STRS23 = 0.0                                                      FSU02620
      STS3 = 0.0                                                        FSU02630
      RETURN                                                            FSU02640
      END                                                               FSU02650
```

```
      FUNCTION QQIN(N,K)                                                FSU00010
C COMPUTE INFLOW RATE FROM READ IN VALUES.                              FSU00020
C READ IN VALUES TREATED AS A STEP FUNCTION.                            FSU00030
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU00040
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                            FSU00050
      COMMON /INB/ DTQI(5),QI(1893,2),QIN(1893),NQI(5)                  FSU00060
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00070
      IBUG = IDEBUG(9)                                                  FSU00080
      IF(K.EQ.1)GO TO 12                                                FSU00090
      ET=DTT*FLOAT(N)                                                   FSU00100
      R=ET/DTQI(K)                                                      FSU00110
      L=R-.0001                                                         FSU00120
      QQIN=QI(L+1,K)                                                    FSU00130
      RETURN                                                            FSU00140
   12 QQIN=QPLNT(N)                                                     FSU00150
      RETURN                                                            FSU00160
      END                                                               FSU00170
      FUNCTION QQOUT(N,I)                                               FSU00180
C COMPUTE OUTFLOW RATE FROM READ IN VALUES.                             FSU00190
C READ IN VALUES TREATED AS A STEP FUNCTION.                            FSU00200
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU00210
      COMMON /OUTA/ NQO(5),DTQO(5),QO(1893,2),UOMAX(5),UOT(102,5)       FSU00220
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                            FSU00230
      COMMON /INB/ DTQI(5),QI(1893,2),QIN(1893),NQI(5)                  FSU00240
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00250
      IBUG = IDEBUG(10)                                                 FSU00260
      IF(I.EQ.1)GO TO 12                                                FSU00270
      ET=DTT*FLOAT(N)                                                   FSU00280
      R=ET/DTQO(I)                                                      FSU00290
      L = R                                                             FSU00300
      QQOUT=QO(L+1,I)                                                   FSU00310
      RETURN                                                            FSU00320
   12 QQOUT = QPLNT(N)                                                  FSU00330
      RETURN                                                            FSU00340
      END                                                               FSU00350
      FUNCTION TTIN(N,K)                                                FSU00360
C COMPUTE INFLOW TEMPERATURE FROM READ IN VALUES.                       FSU00370
C LINEAR INTERPOLATION BETWEEN READ IN VALUES.                          FSU00380
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00390
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                            FSU00400
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU00410
      COMMON /INA/ DTTI(5),TI(1893,2),TIN(1893),NTI(5)                  FSU00420
      IBUG = IDEBUG(8)                                                  FSU00430
      IF(K.EQ.1)GO TO 12                                                FSU00440
      DTTI(1)=DTT                                                       FSU00450
      ET=DTT*FLOAT(N)                                                   FSU00460
      R=ET/DTTI(K)                                                      FSU00470
      L=R-0.0001                                                        FSU00480
      IF (L.EQ.0) GO TO 10                                              FSU00490
      RR = R-FLOAT(L)                                                   FSU00500
      TTIN=TI(L,K)+RR*(TI(L+1,K)-TI(L,K))                               FSU00510
      GO TO 11                                                          FSU00520
   10 TTIN=TI(1,K)                                                      FSU00530
   11 RETURN                                                            FSU00540
   12 TTIN=TPLNT(N)                                                     FSU00550
```

```
      IF(ISELCT(8).GT.0)TTIN=TI(N,1)                                    FSU00560
      RETURN                                                            FSU00570
      END                                                               FSU00580
       SUBROUTINE TOUT(HEATOT,FLOWOT)                                   FSU00590
C COMPUTE WEIGHTED AVERAGE OF OUTFLOW TEMPERATURE.                      FSU00600
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00610
      COMMON /SURFS/ DYSUR,AYSUR,SURF(1893),TSUR(100,2),ELBSL           FSU00620
      COMMON /RALPH/ DATRAD,ATRAD,AS(20,2),III,LAGTIM(1893),NH          FSU00630
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR  FSU00640
      COMMON /OUTA/ NQO(5),DTQO(5),QO(1893,2),UOMAX(5),UOT(102,5)       FSU00650
      COMMON /MAINPA/T(102,2),EL(102),XL(102),A(102),V(102,1),UI(102,5) FSU00660
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                 FSU00670
      IBUG = IDEBUG(7)                                                  FSU00680
      I=III                                                             FSU00690
      HEATOT=T(JM,1)*B(JM)*UOT(JM,I)*DYSUR+T(1,1)*B(1)*UOT(1,I)*DY/2.0  FSU00700
      FLOWOT=B(JM)*UOT(JM,I)*DYSUR+B(1)*UOT(1,I)*DY/2.0                 FSU00710
      JMM=JM-1                                                          FSU00720
      DO 2 J=2,JMM                                                      FSU00730
      HEATOT=HEATOT+UOT(J,I)*B(J)*DY*T(J,1)                             FSU00740
    2 FLOWOT=FLOWOT+UOT(J,I)*B(J)*DY                                    FSU00750
      IF(FLOWOT.EQ.0.0) FLOWOT=1.0                                      FSU00760
      CONTINUE                                                          FSU00770
      RETURN                                                            FSU00780
      END                                                               FSU00790
      SUBROUTINE XINTKP(N)                                              FSU00800
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU00810
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON  FSU00820
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU00830
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                  FSU00840
      REAL LWRI,LWRO                                                    FSU00850
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY                                FSU00860
      COMMON /OUTB/ JOUT,YOUT,ELOUT(5),TOUTC(5),NOUT,LOUT(5)            FSU00870
      COMMON /INPOND/ AINTKP(5),VINTKP(5),YINTKP(5),TINTK,TINTK1        FSU00880
      IBUG = IDEBUG(6)                                                  FSU00890
      W=WINDY                                                           FSU00900
      DO 5 I=1,2                                                        FSU00910
      IF(YINTKP(I).GT.YSUR) GO TO 10                                    FSU00920
    5 CONTINUE                                                          FSU00930
   10 Y1=YINTKP(I-1)                                                    FSU00940
      Y2=YINTKP(I)                                                      FSU00950
      A1=AINTKP(I-1)                                                    FSU00960
      A2=AINTKP(I)                                                      FSU00970
      V1=VINTKP(I-1)                                                    FSU00980
      V2=VINTKP(I)                                                      FSU00990
      AINTK=(A1+(YSUR-Y1)/(Y2-Y1)*(A2-A1))*43560.                       FSU01000
      VINTK=(V1+(YSUR-Y1)/(Y2-Y1)*(V2-V1))*43560.                       FSU01010
      EA= PSI*25.4*EXP(17.62-9500.8/(TAIR+460))                         FSU01020
      TAIRV=(TAIR+460)/(1.-.378*EA/PRAIR)-460                           FSU01030
      IDKT=QOUT(N,1)*DTT/VINTK+1                                        FSU01040
      DT1=DTT/IDKT                                                      FSU01050
      DO 20 J=1,IDKT                                                    FSU01060
      U=TINTK1                                                          FSU01070
      LWRO=4.E-8*(U+460)**4                                             FSU01080
      EW=25.4*EXP(17.62-9500.8/(U+460.))                                FSU01090
      TWV=(U+460.)/(1.-.378*EW/PRAIR)-460.                              FSU01100
```

```
      DTV=TWV-TAIRV                                                     FSU01110
      IF(DTV.LT.0.) DTV=0.                                              FSU01120
      EVAP=(22.4*(DTV**.3334)+14.*W)*(EW-EA)                            FSU01130
      BR=.255*(U-TAIR)/(EW-EA)                                          FSU01140
      CONDUC=BR*EVAP                                                    FSU01150
      IF(EVAP.LT.0.) EVAP=0.                                            FSU01160
      FLUX=LWRI-LWRO-EVAP-CONDUC+SR                                     FSU01170
      FLUX=FLUX*AINTK/RHO/HCAP                                          FSU01180
      TINTK2=(VINTK*TINTK1+QOUT(N,1)*DT1*(TOUTC(1)-TINTK1/2.)+FLUX*DT1) FSU01190
     1 /(VINTK+QOUT(N,1)*DT1/2.)                                        FSU01200
      TINTK=(TINTK1+TINTK2)/2.                                          FSU01210
   20 TINTK1=TINTK2                                                     FSU01220
      RETURN                                                            FSU01230
      END                                                               FSU01240
      SUBROUTINE AVER(N)                                                FSU01250
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU01260
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH     FSU01270
     *,MIXD(5)                                                          FSU01280
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU01290
      COMMON /SURFS/ DYSUR,AYSUR,SURF(1893),TSUR(100,2),ELBSL           FSU01300
      COMMON /MR/ TMR(1893,2),QMR(1893,2),DMR(1893,2),AMR(1893,2)       FSU01310
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR          FSU01320
      COMMON /RALPH/ DATRAD,ATRAD,AS(20,2),III,LAGTIM(1893),NH          FSU01330
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR  FSU01340
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY                                FSU01350
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB           FSU01360
      COMMON /MAINPA/T(102,2),EL(102),XL(102),A(102),V(102,1),UI(102,5) FSU01370
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                 FSU01380
      INTEGER COLD                                                      FSU01390
      IBUG = IDEBUG(5)                                                  FSU01400
C PERFORMS CONVECTIVE MIXING OF SURFACE LAYERS.                         FSU01410
      DMIX=DY                                                           FSU01420
      W=WINDY                                                           FSU01430
C LOOP COUNTS NO. OF ITERATIONS, ITT=MAX. NO. OF ITERATIONS             FSU01440
      LOOP=0                                                            FSU01450
      ITT=50                                                            FSU01460
  100 AV1=0.                                                            FSU01470
      K3=NH                                                             FSU01480
      LOOP=LOOP+1                                                       FSU01490
      IF(LOOP.GE.ITT) GO TO 89                                          FSU01500
      AV2=0.0                                                           FSU01510
      DHL=DMR(N,1)                                                      FSU01520
      COLD=1                                                            FSU01530
      DO 1 I=1,NH                                                       FSU01540
      TEMP=THL(I,1)                                                     FSU01550
      IF(TEMP.LT.40.0) COLD=2                                           FSU01560
    1 CONTINUE                                                          FSU01570
      DO 11 J=1,JM                                                      FSU01580
      TEMP=T(J,1)                                                       FSU01590
      IF(TEMP.LT.40.0)COLD=2                                            FSU01600
   11 CONTINUE                                                          FSU01610
      K4=1                                                              FSU01620
      JMM=JM-1                                                          FSU01630
      DO 5 I=1,JMM                                                      FSU01640
      J=JM-I+1                                                          FSU01650
```

```
      JJ=J-1                                                            FSU01660
      GO TO (13,12),COLD                                                FSU01670
   12 CALL COLDCK(T(J,1),T(JJ,1),IFLG,LOOP)                             FSU01680
      GO TO (7,6),IFLG                                                  FSU01690
   13 IF(T(J,1)-T(JJ,1)) 6,7,7                                          FSU01700
    6 CONTINUE                                                          FSU01710
      IF(J-2) 8,8,9                                                     FSU01720
    8 T(2,1)=(T(2,1)*A(2)+T(1,1)*A(1)/2.0)/(A(2)+A(1)/2.0)              FSU01730
      T(1,1)=T(2,1)                                                     FSU01740
      GO TO 7                                                           FSU01750
    9 ELY=EL(J)                                                         FSU01760
      DO 10 K=1,JJ                                                      FSU01770
      KJ=J+1-K                                                          FSU01780
      KJJ=KJ-1                                                          FSU01790
      ELX=EL(KJ)                                                        FSU01800
      IF(ELBSL-ELX) 2,2,3                                               FSU01810
    2 AREA=DAHL*DHL/DY/2.                                               FSU01820
      TAREA=AREA*THL(NH,1)                                              FSU01830
      NH1=NH-1                                                          FSU01840
      DO 37 K1=1,NH1                                                    FSU01850
      K2=NH-K1                                                          FSU01860
      TAR=THL(K2,1)                                                     FSU01870
      GO TO (15,14),COLD                                                FSU01880
   14 CALL COLDCK(TAR,T(JJ,1),IFLG,LOOP)                                FSU01890
      GO TO (4,35),IFLG                                                 FSU01900
   15 IF(TAR-T(JJ,1))35,4,4                                             FSU01910
   35 AREA=AREA+DAHL*DHL/DY                                             FSU01920
      TAREA=TAREA+DAHL*DHL/DY*TAR                                       FSU01930
   37 K3=K2                                                             FSU01940
      GO TO 4                                                           FSU01950
    3 AREA=A(KJ)                                                        FSU01960
      TAREA=T(KJ,1)*A(KJ)                                               FSU01970
    4 AV1=AV1+TAREA                                                     FSU01980
      AV2=AV2+AREA                                                      FSU01990
      TAV=AV1/AV2                                                       FSU02000
      GO TO (17,16),COLD                                                FSU02010
   16 CALL COLDCK(TAV,T(KJJ,1),IFLG,LOOP)                               FSU02020
      GO TO (20,10),IFLG                                                FSU02030
   17 IF(TAV-T(KJJ,1)) 10,20,20                                         FSU02040
   10 CONTINUE                                                          FSU02050
   20 DO 30 L=KJ,J                                                      FSU02060
   30 T(L,1)=TAV                                                        FSU02070
      IF(ELY.GT.ELBSL) GO TO 50                                         FSU02080
      GO TO 55                                                          FSU02090
   50 DMIX=YSUR-EL(KJ)+DY/2.                                            FSU02100
      K4=K3                                                             FSU02110
      DO 60 L=J,JM                                                      FSU02120
   60 T(L,1)=TAV                                                        FSU02130
      DO 70 L=K3,NH                                                     FSU02140
   70 THL(L,1)=TAV                                                      FSU02150
   55 CONTINUE                                                          FSU02160
    7 AV1=0.0                                                           FSU02170
      AV2=0.0                                                           FSU02180
    5 CONTINUE                                                          FSU02190
      DO 90 I=1,JMM                                                     FSU02200
```

```
      J=JM-I+1                                                          FSU02210
      JJ=J-1                                                            FSU02220
      GO TO (19,18),COLD                                                FSU02230
   18 CALL COLDCK(T(J,1),T(JJ,1),IFLG,LOOP)                             FSU02240
      GO TO (90,100),IFLG                                               FSU02250
   19 IF(T(J,1)-T(JJ,1)) 100,90,90                                      FSU02260
   90 CONTINUE                                                          FSU02270
      GO TO 91                                                          FSU02280
   89 WRITE(7,2000)ITT,N                                                FSU02290
 2000 FORMAT(' *** AVER ***  ',I5,'  ITERATIONS COMPLETED- JUMPING OUT  FSU02300
     + ON STEP ',I5)                                                    FSU02310
   91 CONTINUE                                                          FSU02320
      RETURN                                                            FSU02330
      END                                                               FSU02340
      SUBROUTINE SPEED(N)                                               FSU02350
C COMPUTATION OF VERTICAL AND SOURCE AND SINK VELOCITIES.               FSU02360
C ALSO, COMPUTATION OF WITHDRAWAL THICKNESS.                            FSU02370
C SOURCE AND SINK VELOCITIES ARE ASSUMED TO HAVE GAUSSIAN DISTRIBUTION. FSU02380
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FSU02390
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON  FSU02400
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH,    FSU02410
     + MIXD(5)                                                          FSU02420
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FSU02430
      COMMON /FLOW/ SPREAD,SIGMAI,SIGMAO,FLUXB,SPVOL,DERIV,HAFDEL,EPSIL FSU02440
      COMMON /TIMES/ DTDD,DTTMP,DTTB                                    FSU02450
      COMMON /SURFS/ DYSUR,AYSUR,SURF(1893),TSUR(100,2),ELBSL           FSU02460
      COMMON /MR/ TMR(1893,2),QMR(1893,2),DMR(1893,2),AMR(1893,2)       FSU02470
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2),TCOOL(1893,2),            FSU02480
     + QCOOL(1893,2)                                                    FSU02490
      COMMON /THICK/ THICK1,THICK2,THICKB                               FSU02500
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR          FSU02510
      COMMON /RALPH/ DATRAD,ATRAD,AS(20,2),III,LAGTIM(1893),NH          FSU02520
      COMMON /PARSON/ EX(102),EXO(102),EXI(102),QX(102),UO(102,1)       FSU02530
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR  FSU02540
      COMMON /DRFH/ AH(50),TBOT,CONBOT,TB(20,2),TAB(40),NDYB            FSU02550
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                  FSU02560
      REAL LWRI,LWRO                                                    FSU02570
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY                                FSU02580
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB           FSU02590
      COMMON /OUTA/ NQO(5),DTQO(5),QO(1893,2),UOMAX(5),UOT(102,5)       FSU02600
      COMMON /OUTB/ JOUT,YOUT,ELOUT(5),TOUTC(5),NOUT,LOUT(5)            FSU02610
      COMMON /INA/ DTTI(5),TI(1893,2),TIN(1893),NTI(5)                  FSU02620
      COMMON /INB/ DTQI(5),QI(1893,2),QIN(1893),NQI(5)                  FSU02630
      COMMON /INC/ UIMAX(5),SIGMIN(5),NIN,JIN(5)                        FSU02640
      COMMON /INPOND/ AINTKP(5),VINTKP(5),YINTKP(5),TINTK,TINTK1        FSU02650
      COMMON /ENERGY/ EO,E1,E2,E3,ET,FLUXT                              FSU02660
      COMMON /MAINPA/T(102,2),EL(102),XL(102),A(102),V(102,1),UI(102,5) FSU02670
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                 FSU02680
      COMMON /SWITCH/ KSUR,KOH,KQ,KLOSS,KMIX,KAREA,KATRAD,KTRAV         FSU02690
      COMMON/FLAG/IWDFLG                                                FSU02700
      IBUG = IDEBUG(4)                                                  FSU02710
C  COMPUTE INFLOW VELOCITY                                              FSU02720
C  COMPUTE EXPONENTIAL FACTOR                                           FSU02730
C CALC. STANDARD DEVIATION OF INFLOWS                                   FSU02740
      GRAV=32.*86400.*86400.                                            FSU02750
```

```
      FRD=0.25                                                          FSU02760
      MIXED=1                                                           FSU02770
      DO 330 K=1,NIN                                                    FSU02780
      IF(K.EQ.1) GO TO 350                                              FSU02790
      JIN1=JIN(K)                                                       FSU02800
      IF(JIN1.EQ.JM.OR.JIN1.EQ.2) GO TO 301                             FSU02810
      DERIV=(T(JIN1+1,1)-T(JIN1-1,1))/2.0/DY                            FSU02820
      IF(DERIV-.01) 311,311,315                                         FSU02830
C SURFACE AND BOTTOM INFLOWS                                            FSU02840
  301 DELRHO=.00000205*((T(JIN1,1)-39.)**2-(TIN(K)-39.)**2)             FSU02850
      DELRHO=ABS(DELRHO)                                                FSU02860
      IF(DELRHO-.000001) 311,311,302                                    FSU02870
  302 GPRIME=GRAV*DELRHO                                                FSU02880
      HDINF=(QIN(K)/FRD/B(JIN1)/SQRT(GPRIME))**.6667                    FSU02890
      DINM=4.0*DY                                                       FSU02900
      IF(HDINF.LT.DINM) HDINF=DINM                                      FSU02910
  306 SIGMAI=HDINF/SPREAD                                               FSU02920
      GO TO 314                                                         FSU02930
C INTERMEDIATE INFLOWS                                                  FSU02940
  315 EFSIL=2.0*(T(JIN1,1)-39.)*.00000205*DERIV                         FSU02950
      EFSIL=ABS(EFSIL)                                                  FSU02960
      QINXXX=2.0*QIN(K)                                                 FSU02970
      DINF=SQRT(QINXXX/B(JIN1)/FRD)*(1.0/GRAV/EFSIL)**0.25              FSU02980
      THICK1=DINF                                                       FSU02990
      DINM=4.0*DY                                                       FSU03000
      IF(DINF.LT.DINM) DINF=DINM                                        FSU03010
      HDINF=DINF/2.0                                                    FSU03020
      SIGMAI=DINF/2.0/SPREAD                                            FSU03030
      GO TO 314                                                         FSU03040
C CUTOFF DUE TO SHARP CHANGE IN GRADIENT                                FSU03050
  311 IF(JIN1.EQ.JM) GO TO 318                                          FSU03060
      JIN3=JIN1+2                                                       FSU03070
      JMM=JM-1                                                          FSU03080
      DO 312 J=JIN3,JMM                                                 FSU03090
      IF((T(J+1,1)-T(J,1))/DY-.05) 312,313,313                          FSU03100
  312 CONTINUE                                                          FSU03110
      SIGMAI=FLOAT(JM)*DY/4.0                                           FSU03120
      GO TO 319                                                         FSU03130
  313 HDINF=FLOAT(J-JIN1)*DY                                            FSU03140
      SIGMAI=HDINF/SPREAD                                               FSU03150
      GO TO 319                                                         FSU03160
  318 SIGMAI=FLOAT(JM)*DY/4.0                                           FSU03170
  319 IF(JIN1.EQ.2) GO TO 314                                           FSU03180
      JINO=JIN1-2                                                       FSU03190
      DO 321 I=1,JINO                                                   FSU03200
      J=JINO+2-I                                                        FSU03210
      IF((T(J,1)-T(J-1,1))/DY-.05) 321,321,322                          FSU03220
  321 CONTINUE                                                          FSU03230
      SIGMAI=FLOAT(JM)*DY/4.0                                           FSU03240
      GO TO 314                                                         FSU03250
  322 HDINF=FLOAT(JIN1-J)*DY                                            FSU03260
      SIGM1=HDINF/SPREAD                                                FSU03270
      IF(SIGM1.LT.SIGMAI) SIGMAI=SIGM1                                  FSU03280
      GO TO 314                                                         FSU03290
  350 SIGMAI=DMR(N,1)/SPREAD                                            FSU03300
```

```
      HDINF=2.0*DMR(N,1)                                                FSU03310
C ASSUME DEPTH OF MIXING LAYER =INFLOW DEPTH(APPROX.) PUT UPPER LIMIT   FSU03320
C OF HALF DEPTH                                                         FSU03330
  314 MIXD(K)=IFIX(HDINF/DY)+1                                          FSU03340
      HMJ=FLOAT(JM)/2.0                                                 FSU03350
      JMH=IFIX(HMJ)                                                     FSU03360
      IF(MIXD(K).GT.JMH) MIXD(K)=JMH                                    FSU03370
      IF(MIXD(K).GT.MIXED) MIXED=MIXD(K)                                FSU03380
      MIXD(K)=MIXED                                                     FSU03390
  330 SIGMIN(K)=SIGMAI                                                  FSU03400
      DO 9 K=1,NIN                                                      FSU03410
      DO 1 I=1,JM                                                       FSU03420
      S(I)=(DY*FLOAT(I-1))**2                                           FSU03430
      ARGI=S(I)/(2.0*(SIGMIN(K))**2)                                    FSU03440
      IF(ARGI-20.0)4,4,5                                                FSU03450
    4 EX(I)=EXP(-ARGI)                                                  FSU03460
      GO TO 1                                                           FSU03470
    5 EX(I)=0.0                                                         FSU03480
    1 CONTINUE                                                          FSU03490
      DO 2 J=1,JM                                                       FSU03500
      II=IABS( J-JIN(K))+1                                              FSU03510
    2 EXI(J)=EX(II)                                                     FSU03520
C  COMPUTE MAX INFLOW VEL.                                              FSU03530
      VOLIN=EXI(1)*B(1)*DY/2.0+EXI(JM)*B(JM)*DYSUR                      FSU03540
      JMM=JM-1                                                          FSU03550
      DO 3 J=2,JMM                                                      FSU03560
    3 VOLIN=VOLIN+EXI(J)*B(J)*DY                                        FSU03570
      UIMAX(1)=QIN(K)/VOLIN                                             FSU03580
      DO 6 J=1,JM                                                       FSU03590
    6 UI(J,K)=UIMAX(1)*EXI(J)                                           FSU03600
    9 CONTINUE                                                          FSU03610
C   COMPUTE OUTFLOW VELOCITIES                                          FSU03620
      DO 10 LT=1,NOUT                                                   FSU03630
      JOUT=LOUT(LT)                                                     FSU03640
      IF(LT.EQ.2) GO TO 200                                             FSU03650
      IF(ISELCT(13).EQ.1)GO TO 131                                      FSU03660
C COMPUTE WITHDRAWAL THICKNESS.                                         FSU03670
C NOTE THAT ONLY HALF THE WITHDRAWAL THICKNESS IS COMPUTED.             FSU03680
      DERIV = (T(JOUT+1,1)-T(JOUT-1,1))/2.0/DY                          FSU03690
      IF(DERIV-0.010) 11,11,15                                          FSU03700
   11 JOUT1=JOUT+2                                                      FSU03710
C CUTOFF DUE TO SHARP CHANGE IN DENSITY GRADIENT                        FSU03720
      DO 12 J=JOUT1,JMM                                                 FSU03730
      IF((T(J+1,1)-T(J,1))/DY-.05)12,13,13                              FSU03740
   12 CONTINUE                                                          FSU03750
      SIGMAO=100.0*DY                                                   FSU03760
      GO TO 19                                                          FSU03770
   13 HAFDEL=FLOAT(J-JOUT)*DY                                           FSU03780
      SIGMAO=HAFDEL/SPREAD                                              FSU03790
   19 JOUT2=JOUT-2                                                      FSU03800
      IF(JOUT2.LT.1) JOUT2=1                                            FSU03810
      DO 21 I=1,JOUT2                                                   FSU03820
      J=JOUT2+2-I                                                       FSU03830
      IF((T(J,1)-T(J-1,1))/DY-0.05) 21,21,22                            FSU03840
   21 CONTINUE                                                          FSU03850
```

```
      GO TO 14                                                          FSU03860
   22 HAFD1=FLOAT(JOUT-J)*DY                                            FSU03870
      SIGM1=HAFD1/SPREAD                                                FSU03880
      IF(SIGM1.LT.SIGMAO) SIGMAO=SIGM1                                  FSU03890
      GO TO 14                                                          FSU03900
C APPROXIMATING FORMULA USED FOR DENSITY IS RHO=1.0-0.00000205*(T-39.)**FSU03910
   15 EPSIL=2.0*.00000205*(T(JOUT,1)-39.0)*DERIV                        FSU03920
      GO TO (17,16,23),KOH                                              FSU03930
C CALCULATION OF WITHDRAWAL THICKNESS USING KAO FORMULA.                FSU03940
   16 QPUW=QOUT(N,LT)/B(JOUT)                                           FSU03950
      HAFDEL=4.5*SQRT(QPUW)/(EPSIL*32.*86400.*86400.)**.25              FSU03960
      GO TO 18                                                          FSU03970
C CALCULATION OF WITHDRAWAL THICKNESS USING KOH FORMULA.                FSU03980
   17 XO=10.                                                            FSU03990
      EPSIL=ABS(EPSIL)                                                  FSU04000
      AOO=(EPSIL*32.*86400.*86400./DD(1)/0.864)**0.16667                FSU04010
      ALPH=AOO/((3.5*QOUT(N,LT)/B(JOUT)/DD(1)/AOO/(XO**.667))**.133)    FSU04020
      HAFDEL=3.57*XO**0.334/ALPH                                        FSU04030
      GO TO 18                                                          FSU04040
   23 DELRHO=0.00000205*((T(JM,1)-39.)**2-(T(JOUT,1)-39.)**2)           FSU04050
      IF(DELRHO.LT.0.000001) GO TO 11                                   FSU04060
      HAFDEL=(QOUT(N,1)/2000.)**2/2./DELRHO/GRAV                        FSU04070
   18 SIGMAO = HAFDEL/SPREAD                                            FSU04080
      IF(SIGMAO.LT.0.0)SIGMAO=1.0                                       FSU04090
C      COMPARISON OF SELECTIVE WITHDRAWAL WITH                          FSU04100
C      DEPTH AVERAGE CALCULATION HERE                                   FSU04110
   14 IF(ISELCT(13).NE.2)GO TO 20                                       FSU04120
      IF(SIGMAO.LT.15.0)GO TO 131                                       FSU04130
 20   CONTINUE                                                          FSU04140
      IWDFLG=1                                                          FSU04150
C   COMPUTE EXP. FACTOR                                                 FSU04160
      DO 100 I=1,JM                                                     FSU04170
      S(I)=(DY*FLOAT(I-1))**2                                           FSU04180
      ARGO=S(I)/2.0/SIGMAO/SIGMAO                                       FSU04190
      IF(ARGO-20.0) 104,105,105                                         FSU04200
  104 OX(I)=EXP(-ARGO)                                                  FSU04210
      GO TO 100                                                         FSU04220
  105 OX(I)=0.0                                                         FSU04230
  100 CONTINUE                                                          FSU04240
      DO 110 J=1,JM                                                     FSU04250
      IO=IABS(J-JOUT)+1                                                 FSU04260
  110 EXO(J)=OX(IO)                                                     FSU04270
C FIRST COMPUTE MAXIMUM VELOCITIES, THEN OTHERS.                        FSU04280
      VOLOUT=EXO(1)*B(1)*DY/2.+EXO(JM)*B(JM)*DYSUR                      FSU04290
      JMM=JM-1                                                          FSU04300
      DO 120 J=2,JMM                                                    FSU04310
  120 VOLOUT=VOLOUT+EXO(J)*B(J)*DY                                      FSU04320
      UOMAX(LT)=QOUT(N,LT)/VOLOUT                                       FSU04330
      DO 130 J=1,JM                                                     FSU04340
  130 UOT(J,LT)=UOMAX(LT)*EXO(J)                                        FSU04350
      GO TO 10                                                          FSU04360
C                                                                       FSU04370
C     DEPTH AVERAGE CALCULATION                                         FSU04380
C                                                                       FSU04390
 131  JM1 = JM-1                                                        FSU04400
```

```
      SUM = 0.0                                                          FSU04410
      JJJ = (JM-LOUT(1))*2                                               FSU04420
      JJJ = JM-JJJ                                                       FSU04430
      DO 132 J=JJJ,JM1                                                   FSU04440
  132 SUM = SUM + B(J)*DY                                                FSU04450
      SUM = SUM + B(JM)*DYSUR                                            FSU04460
      VAV = QOUT(N,LT)/SUM                                               FSU04470
      DO 133 J=JJJ,JM                                                    FSU04480
  133 UOT(J,LT) = VAV                                                    FSU04490
      JJJ = JJJ-1                                                        FSU04500
      DO 134 J = 1,JJJ                                                   FSU04510
  134 UOT(J,LT) = 0.0                                                    FSU04520
      UOMAX(LT) = VAV                                                    FSU04530
      IWDFLG=2                                                           FSU04540
      GO TO 10                                                           FSU04550
C                                                                        FSU04560
C                                                                        FSU04570
  200 JMM=JM-1                                                           FSU04580
      WW=600.                                                            FSU04590
      JD=JOUT                                                            FSU04600
  223 TSUM=0.                                                            FSU04610
      JB=JM                                                              FSU04620
      DO 225 J=JD,JB                                                     FSU04630
  225 TSUM=TSUM+T(J,1)                                                   FSU04640
      TU=TSUM/FLOAT(JB-JD+1)                                             FSU04650
      TSUM=0.                                                            FSU04660
      JDD=JD-1                                                           FSU04670
      DO 226 J=1,JDD                                                     FSU04680
  226 TSUM=TSUM+T(J,1)                                                   FSU04690
      TL=TSUM/FLOAT(JDD)                                                 FSU04700
      DELRHO=0.00000205*((TU-39.)**2-(TL-39.)**2)                        FSU04710
      HD=FLOAT(JB-JOUT)*DY                                               FSU04720
      IF(DELRHO.LT.0.000001) GO TO 230                                   FSU04730
      QOUTX=12.*QOUT(N,2)                                                FSU04740
      IF(QOUTX.LT.1.728E+9) QOUTX=1.728E+9                               FSU04750
      IF(QOUTX.GT.4.32E+9) QOUTX=4.32E+9                                 FSU04760
      DELH=(QOUTX/WW/HD)**2/2./DELRHO/GRAV                               FSU04770
      HDD=FLOAT(JOUT-JD+1)*DY                                            FSU04780
      IF(DELH.GT.HDD) GO TO 230                                          FSU04790
      GO TO 235                                                          FSU04800
  230 JD=JD-1                                                            FSU04810
      IF(JD.EQ.1) GO TO 235                                              FSU04820
      GO TO 223                                                          FSU04830
  235 VOLOUT=0.                                                          FSU04840
      JX=JOUT-JD                                                         FSU04850
      IF(JX.LT.8) JD=JOUT-8                                              FSU04860
      THICKB=FLOAT(JB-JD+1)*DY                                           FSU04870
      SIGMAO=FLOAT(JOUT-JD+1)*DY/SPREAD                                  FSU04880
      DO 240 J=JD,JB                                                     FSU04890
      FAC=1.0                                                            FSU04900
      IF(J.EQ.1) FAC=0.5                                                 FSU04910
      IF(J.EQ.JM) FAC=DYSUR/DY                                           FSU04920
      EXO(J)=EXP(-(FLOAT(J-JOUT)*DY)**2/2./SIGMAO/SIGMAO)                FSU04930
      IF(J.GT.JOUT) EXO(J)=1.                                            FSU04940
  240 VOLOUT=VOLOUT+B(J)*DY*FAC*EXO(J)                                   FSU04950
```

```
      DO 245 J=1,JM                                                     FSU04960
  245 UOT(J,LT)=0.                                                      FSU04970
      DO 250 J=JD,JB                                                    FSU04980
  250 UOT(J,LT)=QOUT(N,2)*EXO(J)/VOLOUT                                 FSU04990
   10 CONTINUE                                                          FSU05000
      JMIXB=JM-MIXED                                                    FSU05010
      DO 36 J=1,JM                                                      FSU05020
      GO TO (31,32),KMIX                                                FSU05030
   32 IF(J-JMIXB) 31,33,33                                              FSU05040
   33 QOMIX(J)=0.                                                       FSU05050
      DO 34 K=1,NIN                                                     FSU05060
      QCH=QQIN(N,1)                                                     FSU05070
      JMIX=JM-MIXD(K)                                                   FSU05080
      IF(J.LT.JMIX) GO TO 34                                            FSU05090
      RMIY=(QCOOL(N,1)-QCH)/QCH                                         FSU05100
      RMIP=RMIX                                                         FSU05110
      IF(K.EQ.1) RMIP=RMIY                                              FSU05120
      QOMIX(J)=QOMIX(J)+QQIN(N,K)*RMIP/(MIXD(K)+1)                      FSU05130
   34 CONTINUE                                                          FSU05140
      UO(J,1)=QOMIX(J)/B(J)/DY                                          FSU05150
      IF(J.EQ.JM) UO(JM,1)=UO(J,1)*DY/DYSUR                             FSU05160
      GO TO 360                                                         FSU05170
   31 UO(J,1)=0.0                                                       FSU05180
  360 DO 35 LT=1,NOUT                                                   FSU05190
   35 UO(J,1)=UO(J,1)+UOT(J,LT)                                         FSU05200
   36 CONTINUE                                                          FSU05210
C COMPUTE VERTICAL ADVECTIVE VELOCITY                                   FSU05220
      V(1,1)=0.0                                                        FSU05230
      UIN=0.                                                            FSU05240
      DO 37 K=1,NIN                                                     FSU05250
   37 UIN=UIN+UI(1,K)                                                   FSU05260
      V(2,1)=(UIN     -UO(1,1))*B(1)*DY/(A(1)+A(2))*2                   FSU05270
      JMX=JM+1                                                          FSU05280
      DO 500 J=3,JMX                                                    FSU05290
      UIN=0.                                                            FSU05300
      DO 38 K=1,NIN                                                     FSU05310
   38 UIN=UIN+UI(J-1,K)                                                 FSU05320
      V(J,1)=(V(J-1,1)*(A(J-2)+A(J-1))/2.0+(UIN-UO(J-1,1))*B(J-1)       FSU05330
     1*DY)/(A(J)+A(J-1))*2.0                                            FSU05340
  500 CONTINUE                                                          FSU05350
      RETURN                                                            FSU05360
      END                                                               FSU05370
      SUBROUTINE COLDCK(T1,T2,IFLG,LOOP)                                FSU05380
C SUBROUTINE CHECKS FOR INSTABILITIES DUE TO DENSITY DIFFERENCES        FSU05390
C THIS SUB CALLED ONLY WHEN A LAYER TEMP. IS BELOW 40.0 DEG F           FSU05400
      IF(LOOP.GE.25)GO TO 15                                            FSU05410
      IF((T2.LT.39.175).AND.(T1.GT.T2)) GO TO 10                        FSU05420
      IF((T2.GE.0.0).AND.(T1.LT.T2)) GO TO 20                           FSU05430
C LAYER IS STABLE                                                       FSU05440
    5 IFLG=1                                                            FSU05450
      RETURN                                                            FSU05460
   10 IF(T1-39.175) 30,30,15                                            FSU05470
   15 T1X=.555*(T1-32.0)                                                FSU05480
      T2X=.555*(T2-32.0)                                                FSU05490
      D1= 1000.0-0.00663*(T1X-4.0)*(T1X-4.0)                            FSU05500
```

```
      D2= 1000.0-0.00663*(T2X-4.0)*(T2X-4.0)                          FSU05510
      IF(D1-D2) 5,5,30                                                FSU05520
   20 IF(T1-39.175) 25,25,30                                          FSU05530
   25 IF(T2-39.175) 30,30,15                                          FSU05540
C LAYER IS UNSTABLE                                                   FSU05550
   30 IFLG=2                                                          FSU05560
      RETURN                                                          FSU05570
      END                                                             FSU05580
```

```
      SUBROUTINE FTEQ(NSTEP,ISTEQ,HHH,KKK,ISLCT,ISELCT,FTEQL,TEQL,NREACHFTE00010
     *)                                                                 FTE00020
      COMMON/CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,         FTE00030
     * DELCON                                                           FTE00040
      COMMON/PLANT/QPLNT(1893),TPLNT(1893)                              FTE00050
      COMMON /TIME/ ATIME                                               FTE00060
      COMMON/MAINPB/NARSF,PLUMF,DY,JM                                   FTE00070
      COMMON/METEOA/WIND(1893),TA(1893),SIGH(1893),FIN(1893),CLOUD(1893)FTE00080
      DIMENSION Y(1893),TTEQL(1893),ISLCT(3),ISELCT(20),UU(1893),XX(1893FTE00090
     *),ISTQ(1893)                                                      FTE00100
      REAL LWRI,LWRO,KKK                                                FTE00110
      COMMON/METEOD/TEQ,PRAIR,CC,WINDY,EFACT                            FTE00120
      COMMON/SWITCH/KSUR,KOH,KQ,KLOSS,KMIX,KAREA,KATRAD,KTRAV           FTE00130
      COMMON/PARAM/DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR        FTE00140
      DIMENSION FTEQL(1893),TEQL(1893)                                  FTE00150
      DHL=DY                                                            FTE00160
      DAY=ATIME                                                         FTE00170
      DO 100 J=1,NSTEP                                                  FTE00180
      DAY=DAY+(J-1)                                                     FTE00190
      W =WIND (J)                                                       FTE00200
      TAIR=TA(J)                                                        FTE00210
      PSI=SIGH(J)                                                       FTE00220
      SR=FIN(J)                                                         FTE00230
      CC=CLOUD(J)                                                       FTE00240
   25 GO TO (27,28),KLOSS                                               FTE00250
C     CALC HEAT FLUX INDEP OF SURF TEMP                                 FTE00260
   27 LWRI=4.E-8*(TAIR+460)**4                                          FTE00270
      GO TO 29                                                          FTE00280
   28 CONTINUE                                                          FTE00290
C     LWRI=1.16E-13*((TAIR+460.)**6)*(1.0+0.17*CC*CC)                   FTE00300
      LWRI=(0.97*4.15E-8)*((TAIR+460.)**4)*(1.-(0.261*EXP(-2.4E-4*      FTE00310
     *(TAIR-32)**2)))*(1.+0.17*CC*CC)                                   FTE00320
   29 EA=PSI*25.4*EXP(17.62-9500.8/(TAIR+460))                          FTE00330
      TAIRV=(TAIR+460)/(1.-.378*EA/PRAIR)-460                           FTE00340
C     CALC EQUILIBRIUM TEMP                                             FTE00350
      C1=1.                                                             FTE00360
      I1=TAIR-15.                                                       FTE00370
      I2=I1+25                                                          FTE00380
   24 DO 37 I=I1,I2                                                     FTE00390
      U=I/C1                                                            FTE00400
      LWRO=4.E-8*(U+460.)**4                                            FTE00410
      EW=25.4*EXP(17.62-9500./(U+460.))                                 FTE00420
      GO TO (32,33),KLOSS                                               FTE00430
   32 EVAP=75.*(EW-EA)                                                  FTE00440
      GO TO 34                                                          FTE00450
   33 DTV=(U+460.)/(1.-.378*EW/PRAIR)-460.-TAIRV                        FTE00460
      DTVL=0.0024*W*W*W                                                 FTE00470
      IF(DTV.LT.DTVL)DTV=DTVL                                           FTE00480
C    THE EVAP REDUCTION IS THE RESULT OF A SEASONAL VARIATION IN WIND   FTE00490
C    SPEEDS BETWEEN THE MET TOWER AND THE LAGOONS                       FTE00500
C     EFACT1=EFACT+0.25*SIN(DAY*3.1416*2./365.25-0.4576)                FTE00510
      EFACT1=EFACT                                                      FTE00520
      EVAP=EFACT1*(22.4*DTV**.3334+14.*W)*(EW-EA)                       FTE00530
      SOL=SR*(1.-(1.-BETA)*EXP(-ETA*DHL))*0.94                          FTE00540
      IF(ISELCT(10).GT.0)SOL=SR*0.94                                    FTE00550
```

← as a counter

← essentially tests every single integer temp. within the range of $t_{air} - 15 \leq t \leq t_{air} + 10$ and sees which one gives $|\phi_{net}| \leq 3$

```
      GO TO 35                                                          FTE00560
   34 SOL=0.                                                            FTE00570
   35 BR=.255*(U-TAIR)/(EW-EA)                                          FTE00580
      CONDUC=BR*EVAP                                                    FTE00590
      IF(EVAP.LT.0.)EVAP=0.                                             FTE00600
      HN=SOL+LWRI-LWRO-EVAP-CONDUC                                      FTE00610
      IF(HN.LT.3)GO TO 36                                               FTE00620
   37 CONTINUE                                                          FTE00630
   36 IF(HN.GT.-3.)GO TO 38                                             FTE00640
      I1=10.*C1*U-20.                                                   FTE00650
      I2=I1+40                                                          FTE00660
      C1=C1*10.                                                         FTE00670
      GO TO 24                                                          FTE00680
   38 TEQL(J)=U                                                         FTE00690
  100 CONTINUE                                                          FTE00700
C                                                                       FTE00710
C     TEQ DATA EXP FILTERING                                            FTE00720
C *****************************                                         FTE00730
C   KKK=SURFACE HEAT EXCH. COEFF. APPROX. CALCULATED FROM CURVES        FTE00740
C        IN RYAN'S REPORT BASED ON AVERAGE WIND AND SURFACE TEMP.       FTE00750
C         FOR EACH SEASON...ALSO BASED ON THREE LEVELS OF HEAT          FTE00760
C         RISE ACROSS THE CONDENSER.                                    FTE00770
C   XX=A RESIDENCE TIME FOR THE ENTIRE WHTF,INCL. SIDEARMS              FTE00780
C   (THE RESIDENCE TIMES ARE AVERAGED OVER A 7 DAY PERIOD,IE            FTE00790
C      A BACKWARD RUNNING AVERAGE)                                      FTE00800
C     ISTEQ=THE NUMBER OF DAYS RUNNING BACKWARD THAT THE EXP.           FTE00810
C       FILTER IS OPERATIVE                                             FTE00820
C *****************************                                         FTE00830
C                                                                       FTE00840
      DAY=ATIME                                                         FTE00850
      DO 225 I=1,NSTEP                                                  FTE00860
      DAY=DAY+(I-1)                                                     FTE00870
C                                                                       FTE00880
C     TO INPUT VALUE OF HHH OR CALCULATE                                FTE00890
C                                                                       FTE00900
      IF(ISLCT(1).EQ.0)GO TO 20                                         FTE00910
C                                                                       FTE00920
C                                                                       FTE00930
C     TO INPUT VALUE OF KKK OR CALCULATE                                FTE00940
C   THIS SINUSOIDAL CURVE HAS ITS MAX AT JULY1 AND ITS MIN AT JAN 1     FTE00950
   20 IF(ISLCT(2).EQ.0) GO TO 21                                        FTE00960
      FLUXX=TPLNT(I)*QPLNT(I)/86400.                                    FTE00970
      IF(FLUXX.LT.10000.)KKK=85.+(22.5*COS((2.*DAY*3.14/365.25)+(2.*3.14FTE00980
     **182./365.25)))                                                   FTE00990
      IF(FLUXX.GE.10000..AND.FLUXX.LT.35000.)KKK=154.+(60.*COS((2.*DAY* FTE01000
     *3.14/365.25)+(182.*3.14*2./365.25)))                              FTE01010
      IF(FLUXX.GE.35000..AND.FLUXX.LT.70000.)KKK=165.+(62.*COS((2.*DAY* FTE01020
     *3.14/365.25)+(182.*3.14*2./365.25)))                              FTE01030
      IF(FLUXX.GE.70000.)KKK=179.+(64.*COS((2.*DAY*3.14/365.25)+(182.*  FTE01040
     *2.*3.14/365.25)))                                                 FTE01050
   21 IF(ISLCT(3).EQ.0) GO TO 30                                        FTE01060
      IF(I.GT.1)GO TO 23                                                FTE01070
      DO 99 J=1,NSTEP                                                   FTE01080
   99 XX(J)=2.66E+9/QPLNT(J)                                            FTE01090
      DO 200 K=1,NSTEP                                                  FTE01100
```

OK down to here

```
      II=7                                                              FTE01110
      X=0.                                                              FTE01120
      N=NSTEP-K+1                                                       FTE01130
      IF(N.LT.II)II=N                                                   FTE01140
      DO 300 J=1,II                                                     FTE01150
  300 X=XX(N-J+1)+X                                                     FTE01160
  200 UU(K)=X/FLOAT(II)                                                 FTE01170
      DO 400 J=1,NSTEP                                                  FTE01180
C   DIVIDING THE RES. TIME BY 3 APPROX. DIVIDES UP THE TIME             FTE01190
C    AMONG THE NREACH REACHES                                           FTE01200
      ISTQ(J)=INT(UU(J)/FLOAT(NREACH))                                  FTE01210
  400 IF(QPLNT(J).LT.1000.)ISTQ(J)=30                                   FTE01220
   23 ISTEQ=ISTQ(I)                                                     FTE01230
   30 X=0.                                                              FTE01240
      Z=0.                                                              FTE01250
      N=NSTEP-I+1                                                       FTE01260
      IF(N.LT.ISTEQ)ISTEQ=N                                             FTE01270
      DO 250 J=1,ISTEQ                                                  FTE01280
      TTEQL(N-J+1)=TEQL(N-J+1)                                          FTE01290
      Y(J)=EXP((-(J-1)*DT*KKK)/(RHO*HCAP*HHH))                          FTE01300
      X=(TTEQL(N-J+1)*Y(J))+X                                           FTE01310
      Z=Y(J)+Z                                                          FTE01320
  250 CONTINUE                                                          FTE01330
      FTEQL(N)=X/Z                                                      FTE01340
  225 CONTINUE                                                          FTE01350
      RETURN                                                            FTE01360
      END                                                               FTE01370
```

"screwed up" here.

output depends on # days of simulation because of filtering.

```
      SUBROUTINE FKKK(N,K,I)                                                FKK00010
      REAL K,KKK                                                            FKK00020
      DIMENSION RKK(8,1893)                                                 FKK00030
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                                FKK00040
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),        FKK00050
     * QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH                         FKK00060
      COMMON /TIME/ ATIME                                                   FKK00070
      COMMON /CONST/ RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON     FKK00080
C                                                                           FKK00090
C                                                                           FKK00100
C                                                                           FKK00110
C   THIS SUBROUTINE FILTERS THE SURFACE HEAT EXCHANGE COEFFICIENT,K,        FKK00120
C   IN EACH OF THE VARIOUS REACHES IN WHICH IT IS CALCULATED,IE THE         FKK00130
C   SIDEARMS(5 OF THEM) AND THE 3 WHTF REACHES...                           FKK00140
C                                                                           FKK00150
C                                                                           FKK00160
C   THE FILTER SHOULD BE OPERATIVE ON THE REACH IN QUESTION FOR THE LENG    FKK00170
C   TH OF ITS RESIDENCE TIME,SO THE FOLLOWING CALCULATIONS ARE APPROX       FKK00180
C   RES. TIMES(DAYS) FOR EACH SECTION GIVEN THAT THE FOLLOWING VOLUMES      FKK00190
C   CORRESPOND TO EACH:  SEGMENT     I=          VOL(FT**3)                FKK00200
C                          REACH1      1           1.65E8                   FKK00210
C                          REACH2      2           7.8E8                    FKK00220
C                          REACH3      3           5.14E8                   FKK00230
C             REACH2       SA1         4           5.5E8                    FKK00240
C                          SA2         5           5.2E8                    FKK00250
C             REACH3       SA1         6           5.8E7                    FKK00260
C                          SA2         7           6.9E7                    FKK00270
C                        MAIN LAKE SA  8           3.5E9                    FKK00280
C  NOTE:THE SAME VARIABLES ARE USED HERE AS DEFINED IN FTEQ SUBROUTINE      FKK00290
C   DETERMINE ISTEQ(RES TIME DAYS)                                          FKK00300
      IF(I.EQ.1)ISTEQ=INT(1.65E+8/QPLNT(N))                                 FKK00310
      IF(NREACH.EQ.4)GO TO 25                                               FKK00320
      IF(I.EQ.2)ISTEQ=INT(7.8E+8/QPLNT(N))                                  FKK00330
      IF(I.EQ.3)ISTEQ=INT(5.14E+8/QPLNT(N))                                 FKK00340
      IF(I.EQ.4)ISTEQ=5                                                     FKK00350
      IF(I.EQ.5)ISTEQ=5                                                     FKK00360
      GO TO 40                                                              FKK00370
   25 IF(I.EQ.2)ISTEQ=INT(9.4E+8/QPLNT(N))                                  FKK00380
      IF(I.EQ.3)ISTEQ=INT(9.1E+8/QPLNT(N))                                  FKK00390
      IF(I.EQ.4)ISTEQ=INT(5.14E+8/QPLNT(N))                                 FKK00400
   40 IF(I.EQ.6)ISTEQ=1                                                     FKK00410
      IF(I.EQ.7)ISTEQ=1                                                     FKK00420
      IF(I.EQ.8)ISTEQ=7                                                     FKK00430
C  VALUES OF K(FOR THE FILTER PARAMETER ONLY)                               FKK00440
      DAY=ATIME                                                             FKK00450
C   THE COS CURVE GIVES THE MAX AT JULY1 AND THE MIN AT JAN 1               FKK00460
      FLUXX=TPLNT(N)*QPLNT(N)/86400.                                        FKK00470
      IF(FLUXX.LT.10000.)KKK=85.+(22.5*COS((2.*DAY*3.14/365.25)+(182.*2     FKK00480
     **3.14/365.25)))                                                       FKK00490
      IF(FLUXX.GE.10000..AND.FLUXX.LT.35000.)KKK=154.+(60.*COS((2.*DAY*     FKK00500
     *3.14/365.25)+(182.*3.14*2./365.25)))                                  FKK00510
      IF(FLUXX.GE.35000..AND.FLUXX.LT.70000.)KKK=165.+(62.*COS((2.*DAY*     FKK00520
     *3.14/365.25)+(182.*3.14*2./365.25)))                                  FKK00530
      IF(FLUXX.GE.70000.)KKK=179.+(64.*COS((2.*3.14*DAY/365.25)+(2.*3.      FKK00540
     *14*182./365.25)))                                                     FKK00550
```

```
C  VALUE OF DEPTH                                                    FKK00560
      HHH=18.0                                                       FKK00570
      DT=1.0                                                         FKK00580
      JJ=N-ISTEQ                                                     FKK00590
      IF(JJ.LT.1)JJ=1                                                FKK00600
      RKK(I,N)=K                                                     FKK00610
      X=0.                                                           FKK00620
      Z=0.                                                           FKK00630
      DO 20 J=JJ,N                                                   FKK00640
      XX=RKK(I,J)                                                    FKK00650
      Y=EXP(-(DT*KKK*(N-J)/RHO*HCAP*HHH))                            FKK00660
      X=(XX*Y)+X                                                     FKK00670
      Z=Y+Z                                                          FKK00680
   20 CONTINUE                                                       FKK00690
      K=X/Z                                                          FKK00700
      RETURN                                                         FKK00710
      END                                                            FKK00720
```

```
      SUBROUTINE SURLAY(N,TTEQ)                                          FSU00010
C CALCULATION OF SURFACE LOSSES DUE TO EVAPORATION, CONDUCTION, AND RADIFSU00020
C  radiation                                                            FSU00030
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                                FSU00040
      COMMON /IOPUT/ IREAD,IRITE,IRR                                     FSU00050
      COMMON/NUS/ KSA,NESA(10),XSAR(10),XH(10),XB(10),XAREA(10)          FSU00060
      COMMON/NEW/ TRSA(10),XQSA(10),ETSA(10)                             FSU00070
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON   FSU00080
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH,     FSU00090
     + MIXD(5)                                                           FSU00100
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR       FSU00110
      COMMON /FLOW/ SPREAD,SIGMAI,SIGMAO,FLUXB,SPVOL,DERIV,HAFDEL,EPSIL  FSU00120
      COMMON /TIMES/ DTDD,DTTMP,DTTB                                     FSU00130
      COMMON /SURFS/ DYSUR,AYSUR,SURF(1893),TSUR(100,2),ELBSL            FSU00140
      COMMON /MR/ TMR(1893,2),QMR(1893,2),DMR(1893,2),AMR(1893,2)        FSU00150
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2),TCOOL(1893,2),             FSU00160
     + QCOOL(1893,2)                                                     FSU00170
      COMMON /THICK/ THICK1,THICK2,THICKB                                FSU00180
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR           FSU00190
      COMMON /RALPH/ DATRAD,ATRAD,AS(20,2),III,LAGTIM(1893),NH           FSU00200
      COMMON /PARSON/ EX(102),EXO(102),EXI(102),QX(102),UO(102,1)        FSU00210
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR   FSU00220
      COMMON /DRFH/ AH(50),TBOT,CONBOT,TB(20,2),TAB(40),NDYB             FSU00230
      COMMON /METEOA/ WIND(366),TA(366),SIGH(366),FIN(366),CLOUD(366)    FSU00240
      COMMON /METEOB/ DTTA,DTSIGH,DTFIN,DTWIND,DCLOUD                    FSU00250
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                   FSU00260
      REAL LWRI,LWRO                                                     FSU00270
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                           FSU00280
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB            FSU00290
      COMMON /OUTA/ NQO(5),DTQO(5),QO(1893,2),UOMAX(5),UOT(102,5)        FSU00300
      COMMON /OUTB/ JOUT,YOUT,ELOUT(5),TOUTC(5),NOUT,LOUT(5)             FSU00310
      COMMON /INA/ DTTI(5),TI(1893,2),TIN(1893),NTI(5)                   FSU00320
      COMMON /INB/ DTQI(5),QI(1893,2),QIN(1893),NQI(5)                   FSU00330
      COMMON /INC/ UIMAX(5),SIGMIN(5),NIN,JIN(5)                         FSU00340
      COMMON /INPOND/ AINTKP(5),VINTKP(5),YINTKP(5),TINTK,TINTK1         FSU00350
      COMMON /ENERGY/ EO,E1,E2,E3,ET,FLUXT                               FSU00360
      COMMON /MAINPA/T(102,2),EL(102),XL(102),A(102),V(102,1),UI(102,5)  FSU00370
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                  FSU00380
      COMMON /SWITCH/ KSUR,KOH,KQ,KLOSS,KMIX,KAREA,KATRAD,KTRAV          FSU00390
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                             FSU00400
      COMMON /WHTF1/ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO,     FSU00410
     . NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4),HSA2(4),ZBR(4),  FSU00420
     *XLSA1(4),XLSA2(4)                                                  FSU00430
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893),    FSU00440
     . QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)     FSU00450
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),     FSU00460
     *QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH                       FSU00470
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                     FSU00480
      COMMON/SDARM/HMAX,BSA                                              FSU00490
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                            FSU00500
      COMMON /TIME/ ATIME                                                FSU00510
      REAL K                                                             FSU00520
C INITIALIZE VARIABLES                                                   FSU00530
      IBUG = IDEBUG(2)                                                   FSU00540
      W = WINDY                                                          FSU00550
```

```
      IF(IBUG.GT.0)WRITE(IRR,5003)N,TAIR,W,CC,PSI,SR,EFACT                FSU00560
 5003 FORMAT(' *** SURLAY-DEBUG *** N,MET DATA = ',I3,6(F7.2,2X))         FSU00570
      FLUXT=0.                                                            FSU00580
      TINIT=TTIN(N,1)                                                     FSU00590
      OCH=QQIN(N,1)                                                       FSU00600
      IF(IBUG.GT.2)WRITE(IRR,5001)TINIT,OCH                               FSU00610
 5001 FORMAT(' *** SURLAY-DEBUG *** TINIT,OCH = ',2(F9.3,2X))            FSU00620
      DHL=DY                                                              FSU00630
   25 GO TO (27,28),KLOSS                                                 FSU00640
C  CALC HEAT FLUX INDEP OF SURF TEMP                                       FSU00650
   27 LWRI=4.E-8*(TAIR+460)**4                                            FSU00660
      GO TO 29                                                            FSU00670
   28 CONTINUE                                                            FSU00680
C     SWINBANK'S FORMULA                                                  FSU00690
C     LWRI=1.16E-13*((TAIR+460.)**6)*(1.0+0.17*CC*CC)-                    FSU00700
C     IDSO-JACKSON'S FORMULA                                              FSU00710
      LWRI=(.97*4.15E-8)*((TAIR+460.)**4)*(1.-(0.261*EXP(-2.4E-4*         FSU00720
     *(TAIR-32)**2)))*(1.+0.17*CC*CC)                                     FSU00730
C     THACKSTON'S FORMULA FOR CALCULATING EA(IN MB) FROM T(IN RANKINE)    FSU00740
C     EA=25.4*EXP(17.62-9500.8/(TAIR+460.))                               FSU00750
C     BRUTSAERT'S FORMULA FOR CALC. LW RADIATION                          FSU00760
C     LWRI=0.553*EA**(1./7.)*4.08E-8*((TAIR+460.)**4.)                    FSU00770
C    &      *(1.0+0.17*CC*CC)                                            FSU00780
   29 EA=.PSI*25.4*EXP(17.62-9500.8/(TAIR+460))                           FSU00790
      TAIRV=(TAIR+460)/(1.-.378*EA/PRAIR)-460                             FSU00800
      IF(IBUG.GT.2)WRITE(IRR,5004)LWRI,EA,TAIRV                           FSU00810
 5004 FORMAT(' *** SURLAY-DEBUG *** LWRI,EA,TAIRV = ',3(F7.2,2X))         FSU00820
      EW=25.4*EXP(17.62-9500./(TEQ+460.))                                 FSU00830
      DTV=(TEQ+460.)/(1.-.378*EW/PRAIR)-460.-TAIRV                        FSU00840
      DTVL=0.0024*W*W*W                                                   FSU00850
      IF(DTV.LT.DTVL) DTV=DTVL                                            FSU00860
C     NOTE THE SEASONAL VARIATION IN EFACT1                               FSU00870
      DAY=ATIME                                                           FSU00880
C     EFACT1=EFACT+0.25*SIN(DAY*3.1416*2./365.25-0.4576)                  FSU00890
      EFACT1=EFACT                                                        FSU00900
      EVAP=EFACT1*(22.4*DTV**.3334+14.*W)*(EW-EA)                         FSU00910
      IF(IBUG.GT.2)WRITE(IRR,5002)TEQ                                     FSU00920
 5002 FORMAT(' *** SURLAY-DEBUG *** TEQ = ',F7.2)                         FSU00930
C     CALCULATE APPROXIMATE WATER LOSS BASED ON ASSUMPTION THAT WATER     FSU00940
C     SURFACE TEMPERATURE IS ~~EVENTUALLY~~ TO EQUILIBRIUM TEMPERATURE    FSU00950
      ATOT = 43560.0                                                      FSU00960
      CALL WLOSS(EVAP,ATOT,DT,WLTN)                                       FSU00970
C CALC TEMP DROP IN CHANNEL                                               FSU00980
   31 DA=ACH/NCH                                                          FSU00990
      U=TINIT                                                             FSU01000
      T7=U                                                                FSU01010
      DO 40 I=1,NCH                                                       FSU01020
      T6=T7                                                               FSU01030
   30 LWRO=4.E-8*(U+460)**4                                               FSU01040
      EW=25.4*EXP(17.62-9500.8/(U+460.))                                  FSU01050
      TWV=(U+460.)/(1.-.378*EW/PRAIR)-460.                                FSU01060
      DTV=TWV-TAIRV                                                       FSU01070
      DTVL=0.0024*W*W*W                                                   FSU01080
      IF(DTV.LT.DTVL) DTV=DTVL                                            FSU01090
C     EFACT1=EFACT+0.25*SIN(DAY*3.1416*2./365.25-0.4576)                  FSU01100
```

Stefan - Boltzman Constant

$1.2\overline{5} \times 10^{-8}$

equal

```
      EFACT1=EFACT                                                      FSU01110
      EVAP=EFACT1*(22.4*(DTV**.3334)+14.*W)*(EW-EA)                     FSU01120
      BR=.255*(U-TAIR)/(EW-EA)                                          FSU01130
      CONDUC=BR*EVAP                                                    FSU01140
      IF(EVAP.LT.0.) EVAP=0.                                            FSU01150
      FLUX=LWRI-LWRO-EVAP-CONDUC+(SR*0.94)                              FSU01160
      IF(ISELCT(10).GT.0)FLUX=FLUX+(0.06*SR)                            FSU01170
      DELT=FLUX*DA/62.4/QCH                                             FSU01180
      T4=T7+DELT                                                        FSU01190
      X8=(T4+T6)/2.                                                     FSU01200
      X9=ABS(X8-U)                                                      FSU01210
      IF(X9.LT..1) GO TO 39                                             FSU01220
      U=X8                                                              FSU01230
      IF(IBUG.GT.4)WRITE(IRR,5005)U                                     FSU01240
 5005 FORMAT(' *** SURLAY-DEBUG *** IN CHANNEL LOOP, U= ',E15.6)        FSU01250
      GO TO 30                                                          FSU01260
   39 T7=T4                                                             FSU01270
      CALL WLOSS(EVAP,DA,DT,WLT)                                        FSU01280
      FLUXT=FLUXT-(LWRI-LWRO-EVAP-CONDUC)*DA  *DT                       FSU01290
      U=T7                                                              FSU01300
   40 CONTINUE                                                          FSU01310
C     COMPUTE MIXING AT ENTRANCE TO WHTF                                FSU01320
C     ****** FOLLOWING SECTION BY-PASSED BY P. SHANAHAN ******         FSU01330
C     ****** MIXING CALCULATION IS DUPLICATED IN WHTF   ******         FSU01340
      TMIX = TFIN(N,1)                                                  FSU01350
      TC1=(U-32.)*5./9.                                                 FSU01360
      TC2=(TMIX-32.)*5./9.                                              FSU01370
      RH01=.999973*(1.-(TC1-3.9863)**2/508929.*(TC1+288.9414)/          FSU01380
     1 (TC1+68.12963))                                                  FSU01390
      RH02=.999973*(1.-(TC2-3.9863)**2/508929.*(TC2+288.9414)/          FSU01400
     1 (TC2+68.12963))                                                  FSU01410
      DELRHO= RH02-RH01                                                 FSU01420
      IF(DELRHO.LT.0.) GO TO 45                                         FSU01430
      F=QCH/86400./WCHX/DCHX/(DELRHO*32.2*DCHX)**.5                     FSU01440
      GO TO 46                                                          FSU01450
   45 F=5.                                                              FSU01460
   46 CONTINUE                                                          FSU01470
   75 IF(F.LT.1.) F=1.                                                  FSU01480
C    CALC RES TIME  IN CHANNEL                                         FSU01490
      RESTIM=ACH*(DCH+HO)/2./QCH                                        FSU01500
      XLAGT=RESTIM/DT                                                   FSU01510
      LAGTYM=XLAGT+.5                                                   FSU01520
      NN=N+LAGTYM                                                       FSU01530
      QEXIT(NN,1)=QCH                                                   FSU01540
      TEXIT(NN,1)=T4                                                    FSU01550
      TEXSTR=(TEXIT(N,1)-TEQ)/(TINIT-TEQ)                               FSU01560
      IF(TEXIT(N,1).GT.1.) GO TO 77                                     FSU01570
   76 IF(N.GT.1) GO TO 800                                              FSU01580
      TEXIT(N,1)=TZERO                                                  FSU01590
      QEXIT(N,1)=QCH                                                    FSU01600
   77 TTEXIT(N,1)=TEXIT(N,1)                                            FSU01610
C                                                                       FSU01620
      CALL WHTF(N,U,EA,TAIRV,W,TTEQ)                                    FSU01630
C                                                                       FSU01640
C JET MODEL FOR NORTH ANNA DIKE III                                     FSU01650
```

```
C                                                                       FSU01660
C     THIS BLOCK OF CODE MOVED                                          FSU01670
      TMIX=0.                                                           FSU01680
      MIXK=MIXD(1)                                                      FSU01690
C  UNFORTUNATELY MIXD(1) IS REDEFINED FROM THE INPUT VALUE IN SPEED SUB FSU01700
C   IT WAS INPUTTED AS 5,BUT SPEED CHANGES IT TO 8                      FSU01710
C   ALSO IT WAS MIXING THE TOP '8' LAYERS WITH THE JET....              FSU01720
      DO 5 I=1,MIXK                                                     FSU01730
      J=JM+1-I                                                          FSU01740
    5 TMIX=TMIX+T(J,1)                                                  FSU01750
      TMIX=TMIX/MIXK                                                    FSU01760
C     END OF CODE BLOCK                                                 FSU01770
C CALC DENSIMETRIC FROUDE NUMBER AT THE ENTRANCE OF THE MAIN POND       FSU01780
      BO=BOO/2.                                                         FSU01790
      HO=HOO                                                            FSU01800
      U1=(TEXIT(N,1)+TMIX)/2.                                           FSU01810
      BETAA=(5.17E-2*(U1-40.)-3.84E-4*(U1-40.)*(U1-40.)+1.5E-6*(U1-40.) FSU01820
     . *(U1-40)*(U1-40.))*1.E-4                                         FSU01830
      IF(TEXIT(N,1).GT.TMIX.AND.BETAA.GT.0.) GO TO 2002                 FSU01840
      WRITE(6,5056)                                                     FSU01850
 5056 FORMAT ('   UPPER LAYER IS COLDER THAN LOWER ONE')                FSU01860
      QRATIO=1.                                                         FSU01870
      F2=99.99                                                          FSU01880
      GO TO 3008                                                        FSU01890
 2002 F=QEXIT(N,1)/(2.*BO*HO)/SQRT(BETAA*(TEXIT(N,1)-TMIX)*32.17*HO)    FSU01900
     . /86400.                                                          FSU01910
      F2=F                                                              FSU01920
C     CALCULATE DILUTION ALLOWED BY CONTROL SECTION                     FSU01930
      HRATIO=.5                                                         FSU01940
      NNN=1                                                             FSU01950
      DELHR=.1                                                          FSU01960
      NNM=1                                                             FSU01970
      DELQ=.1                                                           FSU01980
      QRATIO=1.                                                         FSU01990
      NTIME=0                                                           FSU02000
      ZZ=RTW*RTW*RHO*RHO*RHO/4./QEXIT(N,1)/QEXIT(N,1)*86400.*86400.     FSU02010
     . *32.17*BETAA*(TEXIT(N,1)-TMIX)                                   FSU02020
      QRATI1=QRATIO+DELQ                                                FSU02030
 3000 DELQ=ABS(QRATI1-QRATIO)+.01                                       FSU02040
      QRATI1=QRATIO                                                     FSU02050
 3001 XX=(1+QRATIO)*(1+QRATIO)*(1+QRATIO)/(HRATIO*HRATIO*HRATIO*        FSU02060
     . (2-HRATIO)*(2-HRATIO))+(1+QRATIO)*QRATIO*QRATIO/((1-HRATIO)**5)  FSU02070
      NTIME=NTIME+1                                                     FSU02080
      IF(NTIME.GE.500) WRITE(6,501)                                     FSU02090
  501 FORMAT('    *')                                                   FSU02100
      IF(NTIME.GE.500) GO TO 3008                                       FSU02110
      IF((ABS(XX-ZZ)).LE.(ZZ/1000.)) GO TO 3005                         FSU02120
      IF(XX.LE.ZZ) GO TO 3003                                           FSU02130
      NMM=1                                                             FSU02140
      IF(NMM.EQ.NNM) GO TO 3002                                         FSU02150
      DELQ=DELQ/2.                                                      FSU02160
 3002 QRATIO=QRATIO-DELQ                                                FSU02170
      NNM=NMM                                                           FSU02180
      GO TO 3001                                                        FSU02190
 3003 NMM=0                                                             FSU02200
```

```
      IF(NMM.EQ.NNM) GO TO 3004                                         FSU02210
      DELQ=DELQ/2.                                                      FSU02220
 3004 QRATIO=QRATIO+DELQ                                                FSU02230
      NNM=NMM                                                           FSU02240
      GO TO 3001                                                        FSU02250
 3005 IF(NNN.GT.1) GO TO 3006                                           FSU02260
      NNN=2                                                             FSU02270
      HRATI1=HRATIO                                                     FSU02280
      HRATIO=HRATIO+DELHR                                               FSU02290
      GO TO 3000                                                        FSU02300
 3006 IF((ABS(QRATIO-QRATI1)).LE.0.001.AND.(ABS(HRATIO-HRATI1)).LE.     FSU02310
     . 0.001) GO TO 3008                                                FSU02320
      HRATI1=HRATIO                                                     FSU02330
      IF(QRATIO.GT.QRATI1) GO TO 3007                                   FSU02340
      DELHR=-DELHR/2.                                                   FSU02350
 3007 HRATIO=HRATIO+DELHR                                               FSU02360
      GO TO 3000                                                        FSU02370
 3008 CONTINUE                                                          FSU02380
C     CALCULATE JET DILUTION WITHOUT CONTROL SECTION INFLUENCE           FSU02390
      ASPECT = (HO/BO)**0.25                                            FSU02400
      DJ = 1.4*SQRT(1.0+F*F)*ASPECT                                     FSU02410
      IF(HBOT.GT.0.0)DJ=DJ*(0.75*HBOT/(0.42*SQRT(HO*BO)*ASPECT*F))**0.75FSU02420
      DIL(1)=1.+QRATIO                                                  FSU02430
      IF(IBUG.GT.1)WRITE(20,778)N,DJ,DIL(1)                             FSU02440
  778 FORMAT(' *** SURLAY-DEBUG *** AT TIME STEP ',I3,' DJ,DIL = ',     FSU02450
     +       2E15.6)                                                    FSU02460
      IF(ISELCT(3).EQ.0)DIL(1)=AMIN1(DIL(1),DJ)                         FSU02470
      IF(ISELCT(3).LT.0) DIL(1)=DJ                                      FSU02480
      DIL(1)=AMAX1(DIL(1),1.5)                                          FSU02490
      QRATIO=DIL(1)-1.0                                                 FSU02500
      TSUR(1,1)=(TEXIT(N,1)+TMIX*QRATIO)/(1.+QRATIO)                    FSU02510
      IF(IBUG.GT.4)WRITE(IRR,5006)TSUR(1,1),TEXIT(N,1),TMIX,QRATIO      FSU02520
 5006 FORMAT(' *** SURLAY-DEBUG *** JET CALC, TSUR,TEXIT,TMIX,QRATIO=', FSU02530
     +       4F10.3)                                                    FSU02540
C  TO ELIMINATE ZERO DIVIDE                                           FSU02550
      EPSIL = TEXIT(N,1)-TEQ                                            FSU02560
      IF(EPSIL.LT.0.0001.AND.EPSIL.GT.-0.0001) EPSIL=0.0001             FSU02570
      TSTAR(1,1)=(TSUR(1,1)-TEQ)/EPSIL                                  FSU02580
      TMR(N,1)=TSUR(1,1)                                                FSU02590
      QMR(N,1)=QEXIT(N,1)*DIL(1)                                        FSU02600
      AMR(N,1)=1920000.                                                 FSU02610
      AS(1,1)=AMR(N,1)/AYSUR                                            FSU02620
      FRICIN=0.01                                                       FSU02630
      SHAPE=0.25                                                        FSU02640
      RADIUS=16000.                                                     FSU02650
      U1=(TMR(N,1)+TMIX)/2.                                             FSU02660
      BETAA=(5.17E-2*(U1-40.)-3.84E-4*(U1-40.)*(U1-40.)+1.5E-6*(U1-40.) FSU02670
     . *(U1-40)*(U1-40.))*1.E-4                                         FSU02680
      QQ=QMR(N,1)/86400.                                                FSU02690
      IF(BETAA.LE.0..OR.TMR(N,1).LE.TMIX) DHL=13.                       FSU02700
      IF(BETAA.LE.0..OR.TMR(N,1).LE.TMIX) GO TO 3009                    FSU02710
      DHL=1.5*(FRICIN*(QQ/SHAPE)*(QQ/SHAPE)/16./3.14/3.14/              FSU02720
     . 32.17/RADIUS/(BETAA*(TMR(N,1)-TMIX)))**0.25                      FSU02730
 3009 DASF=AYSUR/NARSF                                                  FSU02740
C                                                                       FSU02750
```

```
C     SET MIXED DEPTH TO 28 FEET                                        FSU02760
      DHL=22.                                                           FSU02770
      YBSUR=YSUR-DHL                                                    FSU02780
      LBSUR=(YBSUR-YBOT)/DY                                             FSU02790
      ELBSL=YBOT+LBSUR*DY-DY/2.                                         FSU02800
      DHL=YSUR-ELBSL                                                    FSU02810
      DMR(N,1)=DHL                                                      FSU02820
C HEAT LOSS REGION                                                      FSU02830
C CALC INCR. AREAS,VOLS, ETC.,                                          FSU02840
  120 AHL=AYSUR-AMR(N,1)                                                FSU02850
      NX=IFIX(AMR(N,1)/DASF)                                            FSU02860
      NHL=NARSF-NX                                                      FSU02870
      DAHL=AHL/NHL                                                      FSU02880
      VHL=DAHL*DMR(N,1)                                                 FSU02890
      QHL=QMR(N,1)                                                      FSU02900
      DTMAX=VHL/QHL                                                     FSU02910
      N1=IFIX(DTT/DTMAX)+1                                              FSU02920
      DT1=DTT/N1                                                        FSU02930
      ISN=2                                                             FSU02940
      KSA2=KSA+1                                                        FSU02950
      IF(KSA.EQ.0) GO TO 706                                            FSU02960
      IF(N.GT.1) GO TO 702                                              FSU02970
      NNNH=NHL+1                                                        FSU02980
      DO 704 ISA=1,KSA                                                  FSU02990
      NESA(ISA)=IFIX((XAREA(ISA)-AMR(N,1))/DAHL)+1                      FSU03000
 704  WRITE(6,754) NESA(ISA),NNNH                                       FSU03010
754   FORMAT(/5X,'THE ELEMENT NUM FOR THE SA IS',2X,I5,2X,'AND THE TOTAL FSU03020
     $ NUMBER OF ELENENTS IS',2X,I5)                                    FSU03030
702   CONTINUE                                                          FSU03040
706   CONTINUE                                                          FSU03050
C     LOOP OVER INTER-SIDE ARM SECTORS OF MAIN POND AND ASSOCIATED SIDE FSU03060
C           ARMS                                                        FSU03070
      DO 772 III=1,KSA2                                                 FSU03080
      IF(III.EQ.1) GO TO 780                                            FSU03090
      IF(NESA(III-1).EQ.(NHL+1)) GO TO 771                              FSU03100
780   CONTINUE                                                          FSU03110
C     LOOP OVER SUB TIME-STEPS                                          FSU03120
      IF(IBUG.GT.2)WRITE(20,779)N,N1                                    FSU03130
 779  FORMAT(' *** SURLAY-DEBUG *** AT MAIN POND: N,N1 = ',2I8)         FSU03140
      DO 250 J=1,N1                                                     FSU03150
      THL(1,2)=TMR(N,1)                                                 FSU03160
      IF(N.EQ.1) GO TO 130                                              FSU03170
      THL(1,1)=TMR(N-1,1)                                               FSU03180
      GO TO 140                                                         FSU03190
  130 THL(1,1)=T(JM,1)                                                  FSU03200
  140 NH=NHL+1                                                          FSU03210
      IF((KSA.NE.0).AND.(III.NE.KSA2)) NH=NESA(III)                     FSU03220
      DO 200 I=ISN,NH                                                   FSU03230
      U=(THL(I-1,1)+THL(I,1))/2.                                        FSU03240
      LWRO=4.E-8*(U+460.)**4                                            FSU03250
      EW=25.4*EXP(17.62-9500.8/(U+460.))                                FSU03260
      TWV=(U+460.)/(1.-.378*EW/PRAIR)-460.                              FSU03270
      DTV=TWV-TAIRV                                                     FSU03280
C     WHEN CONVECTIVE EFFECTS ARE NEGLIGIBLE, ROHWER FORMULA IS USED    FSU03290
      DTVL=((67.-4.*W)/22.4)*((67.-4.*W)/22.4)*((67.-4.*W)/22.4)        FSU03300
```

```
      IF(DTVL.LE.0.0) DTVL=0.0                                          FSU03310
      IF(DTV.LT.DTVL) DTV=DTVL                                          FSU03320
      EVAP=0.85*(22.4*(DTV**.3334)+14.*W)*(EW-EA)                       FSU03330
      BR=.255*(U-TAIR)/(EW-EA)                                          FSU03340
      CONDUC =BR*EVAP                                                   FSU03350
      IF(EVAP.LT.0.) EVAP=0.                                            FSU03360
      FLUX=LWRI-LWRO-EVAP-CONDUC+SR*(1.-(1.-BETA)*EXP(-ETA*DHL))*0.94   FSU03370
      DELT=(QHL*(THL(I-1,1)-THL(I,1))/VHL+FLUX*DAHL/62.4/VHL)*DT1       FSU03380
      THL(I,2)=THL(I,1)+DELT                                            FSU03390
      FLUXT=FLUXT-(LWRI-LWRO-EVAP-CONDUC)*DAHL*DT1                      FSU03400
C     COMPUTE EVAPORATIVE LOSS                                          FSU03410
      CALL WLOSS(EVAP,DAHL,DT1,WLT)                                     FSU03420
  200 CONTINUE                                                          FSU03430
      DO 220 I=1,NH                                                     FSU03440
      AH(I)=((I-1)*DAHL+AMR(N,1)+ACH)/AYSUR                             FSU03450
      THSTAR(I,1)=(THL(I,1)-TEQ)/(TINIT-TEQ)                            FSU03460
      THL(I,1)=THL(I,2)                                                 FSU03470
  220 CONTINUE                                                          FSU03480
  250 CONTINUE                                                          FSU03490
      IF((KSA.EQ.0).OR.(III.EQ.KSA2)) GO TO 771                         FSU03500
      ISA=III                                                           FSU03510
      I22=NESA(ISA)                                                     FSU03520
      HMAX=XH(ISA)                                                      FSU03530
      BSA=XB(ISA)                                                       FSU03540
      CALL OLDSID(XSAR(ISA),DMR(N,1),DMRN,THL(I22,2),R,XQSA(ISA),N,SAE,TFSU03550
     *TEQ)                                                              FSU03560
      CALL WLOSS(SAE,XSAR(ISA),DT,WLT)                                  FSU03570
      IF((XQSA(ISA)-.001).LE.0.0) GO TO 790                             FSU03580
      TRSA(ISA)=EXP(-R*QEXIT(N,1)/XQSA(ISA))*(THL(I22,2)-TEQ)+TEQ       FSU03590
      IF(TRSA(ISA).LT.39.)TRSA(ISA)=39.0                                FSU03600
      GO TO 791                                                         FSU03610
  790 TRSA(ISA)=THL(I22,2)                                              FSU03620
  791 CONTINUE                                                          FSU03630
      IF(XQSA(ISA).GE.QHL) XQSA(ISA)=QHL                                FSU03640
      QHL=QHL-XQSA(ISA)                                                 FSU03650
      ETSA(ISA)=THL(I22,2)                                              FSU03660
      ISN=NESA(ISA)+1                                                   FSU03670
      IF((QHL-.002).LE.0.0) GO TO 760                                   FSU03680
      DTMAX=VHL/QHL                                                     FSU03690
      N1=IFIX(DTT/DTMAX)+1                                              FSU03700
      DT1=DTT/N1                                                        FSU03710
      GO TO 771                                                         FSU03720
  760 N1=1                                                              FSU03730
      DT1=DTT/N1                                                        FSU03740
  771 CONTINUE                                                          FSU03750
  772 CONTINUE                                                          FSU03760
      TCOOL(N,1)=THL(NH,1)                                              FSU03770
      QCOOL(N,1)=QHL                                                    FSU03780
      GO TO 890                                                         FSU03790
  800 IF(N.GT.1) GO TO 820                                              FSU03800
      TCOOL(N,1)=TZERO                                                  FSU03810
      QCOOL(N,1)=QCH                                                    FSU03820
      GO TO 1000                                                        FSU03830
  810 IF(N.GT.1) GO TO 820                                              FSU03840
      TCOOL(N,1)=TZERO                                                  FSU03850
```

```
      QCOOL(N,1)=QEXIT(N,1)                                              FSU03860
      GO TO 1000                                                         FSU03870
  820 TCOOL(N,1)=TCOOL(N-1,1)                                            FSU03880
      QCOOL(N,1)=QCOOL(N-1,1)                                            FSU03890
  890 CONTINUE                                                           FSU03900
 1000 CONTINUE                                                           FSU03910
      RETURN                                                             FSU03920
      END                                                                FSU03930
      SUBROUTINE OLDSID(AREA,HO,HONEW,TO,R,Q,N,EVAP1,TTEQ)               FSU03940
C                                                                        FSU03950
C     DETERMINES THE FLOW IN THE SIDE ARM AND THE RETURNING TEMPERATURE  FSU03960
C     AS A FUNCTION OF THE MAIN POND CONDITIONS AT THE ENTRANCE          FSU03970
C                                                                        FSU03980
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                                FSU03990
      COMMON/SDARM/HMAX,BSA                                              FSU04000
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                   FSU04010
      REAL LWRI,LWRO                                                     FSU04020
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                           FSU04030
      REAL K,LSA                                                         FSU04040
       COMMON /TO/ TEXIT(1893,2),QEXIT(1893,2),TCOOL(1893,2),            FSU04050
     . QCOOL(1893,2)                                                     FSU04060
      IBUG = IDEBUG(3)                                                   FSU04070
      IF(TO.LE.TEQ) GO TO 10                                             FSU04080
      IF(TO.LT.39.)GO TO 10                                              FSU04090
      W=WINDY                                                            FSU04100
      EA= PSI*25.4*EXP(17.62-9500.8/(TAIR+460))                          FSU04110
      TAIRV=(TAIR+460)/(1.-.378*EA/PRAIR)-460                            FSU04120
      U=TO                                                               FSU04130
C                                                                        FSU04140
C CALCULATE HEAT LOSS COEFFICIENT                                        FSU04150
      LWRO=4.E-8*(U+460)**4                                              FSU04160
      EW=25.4*EXP(17.62-9500.8/(U+460.))                                 FSU04170
      TWV=(U+460.)/(1.-.378*EW/PRAIR)-460.                               FSU04180
      DTV=TWV-TAIRV                                                      FSU04190
      DTVL=0.0024*W*W*W                                                  FSU04200
      IF(DTV.LT.DTVL) DTV=DTVL                                           FSU04210
      EVAP=0.85*(22.4*(DTV**.3334)+14.*W)*(EW-EA)                        FSU04220
      BR=.255*(U-TAIR)/(EW-EA)                                           FSU04230
      CONDUC=BR*EVAP                                                     FSU04240
      IF(EVAP.LT.0.) EVAP=0.                                             FSU04250
      FLUX=LWRI-LWRO-EVAP-CONDUC+SR*0.94                                 FSU04260
      SK=-FLUX/(U-TTEQ)/62.4/86400.                                      FSU04270
      CALL FKKK(N,SK,8)                                                  FSU04280
C                                                                        FSU04290
C DETERMINATION OF THE SIDE ARM FLOW                                     FSU04300
      TSTAR=TO-0.6*(TO-TEQ)                                              FSU04310
      IF(TSTAR.GT.39.)GO TO 2008                                         FSU04320
      TSTAR=39.                                                          FSU04330
 2008 UB=(TSTAR+TO)/2.                                                   FSU04340
      LSA=AREA/BSA                                                       FSU04350
      BETAA=(5.17E-2*(UB-40.)-3.84E-4*(UB-40.)*(UB-40.)+1.5E-6*(UB-40.)  FSU04360
     . *(UB-40.)*(UB-40.))*1.E-4                                         FSU04370
      B=SK*SK*LSA*LSA/BETAA/(TO-TEQ)/32.17/HMAX/HMAX/HMAX                FSU04380
      IF(B.LE.0.) GO TO 10                                               FSU04390
      FO=0.02                                                            FSU04400
```

serves same purpose as SIDARM

```
      H=HO/HMAX                                                          FSU04410
      IF(H.GT.0.5) H=0.5                                                 FSU04420
      BF=B*FO*LSA/HMAX/4./H*(0.5*(1./H+1./(1.-H))**3+(1./(1.-H))**3)     FSU04430
      OMEGA=BF**(-0.333)                                                 FSU04440
      Q=OMEGA*SK*LSA*BSA                                                 FSU04450
      IF(Q.LE.0.0) Q=0.0                                                 FSU04460
      Q=Q*86400.                                                         FSU04470
      R=.8*SK*AREA/QEXIT(N,1)*86400.                                     FSU04480
      GO TO 20                                                           FSU04490
   10 Q=0.                                                               FSU04500
   20 CONTINUE                                                           FSU04510
      EVAP1 = EVAP                                                       FSU04520
      RETURN                                                             FSU04530
      END                                                                FSU04540
```

```
      SUBROUTINE WHTF(N,U,EA,TAIRV,W,TTEQ)                              FWH00010
C                                                                       FWH00020
C     ADDED SPECIALLY FOR NORTH ANNA WASTE HEAT TREATMENT FACILITY      FWH00030
C                                                                       FWH00040
      COMMON /IOPUT/ IREAD,IRITE,IRR                                    FWH00050
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON  FWH00060
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR      FWH00070
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2),TCOOL(1893,2),            FWH00080
     +            QCOOL(1893,2)                                         FWH00090
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                  FWH00100
      REAL LWRI,LWRO                                                    FWH00110
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                          FWH00120
      COMMON /WHTF1/ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO,    FWH00130
     +            NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4),     FWH00140
     +            HSA2(4),ZBR(4),XLSA1(4),XLSA2(4)                      FWH00150
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893),   FWH00160
     . QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)    FWH00170
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),    FWH00180
     +            QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH          FWH00190
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                    FWH00200
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                           FWH00210
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FWH00220
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                            FWH00230
      COMMON /TIME/ ATIME                                               FWH00240
      DIMENSION TENT(4)                                                 FWH00250
      REAL K                                                            FWH00260
      IBUG = IDEBUG(12)  ------> card prompt 46                         FWH00270
      IF(N.GE.ISELCT(4).AND.N.LE.ISELCT(5))IBUG=9                       FWH00280
      IF(IBUG.GT.1)WRITE(IRR,2005)N                                     FWH00290
 2005 FORMAT(' *** WHTF-DEBUG *** WHTF CALLED FOR TIME STEP ',I5)       FWH00300
C                                                                       FWH00310
C     LOOP OVER WHTF REACHES CALCULATING A STEADY STATE IN EACH         FWH00320
C                                                                       FWH00330
      DAY=ATIME                                                         FWH00340
      DO 1800 I=1,NREACH  =3                                            FWH00350
C     NNM COUNTS FROUDE NUMBER CONVERGENCE TRIALS                       FWH00360
      NNM=0                                                             FWH00370
C     MOLO COUNTS SIDE ARM FLOW CONVERGENCE TRIALS                      FWH00380
      MOLO=0                                                            FWH00390
      QSA1(I)=0.                                                        FWH00400
      QSA2(I)=0.                                                        FWH00410
      F1=2.0                                                            FWH00420
      F=2.0                                                             FWH00430
      IF(IBUG.GT.1)WRITE(IRR,2009)I,QEXIT(N,1),TEXIT(N,1)               FWH00440
 2009 FORMAT(' *** WHTF-DEBUG *** START REACH LOOP: I, QEXIT, TEXIT = ',FWH00450
     +          /15X,I2,2X,2(2X,E12.7))                                FWH00460
      TSEND(I)=TEXIT(N,1)                                               FWH00470
C                                                                       FWH00480
C     ITERATION LOOP TO CALCULATE STEADY STATE SOLUTION                 FWH00490
C                                                                       FWH00500
  200 U=(TEXIT(N,1)+TSEND(I))/2                                         FWH00510
      IF(IBUG.GT.3)WRITE(IRR,2010)I,K,U,TSEND(I)                        FWH00520
 2010 FORMAT(' *** WHTF-DEBUG *** I,K,U,TSEND = ',I3,3(1X,E11.5))       FWH00530
      OK = K                                                            FWH00540
      UA = U+460.0                                                      FWH00550
```

input data req'd to calc. resid. time
↳ read(I), I = 1, 2, 3

```
      MOLO=MOLO+1                                                       FWH00560
      IF(MOLO.GT.40)GO TO 1400                                          FWH00570
      LWRO=4.E-8*UA**4                                                  FWH00580
      EW=25.4*EXP(17.62-9500.8/UA)                                      FWH00590
      TWV=UA/(1.-.378*EW/PRAIR)-460.                                    FWH00600
      DTV=TWV-TAIRV                                                     FWH00610
      IF(DTV.LT.0.) DTV=0.                                              FWH00620
                                                                        FWH00630
C     SEASONAL VARIABILITY IN EFACT                                     FWH00640
C     EFACT1=EFACT+0.25*SIN(DAY*3.1416*2./365.25-0.4576)                FWH00650
      EFACT1=EFACT                                                      FWH00660
      EVAP=EFACT1*(22.4*(DTV**.3334)+14.*W)*(EW-EA)                     FWH00670
      BR=.255*(U-TAIR)/(EW-EA)                                          FWH00680
      CONDUC=BR*EVAP                                                    FWH00690
      IF(EVAP.LT.0.) EVAP=0.                                            FWH00700
      FLUX=LWRI-LWRO-EVAP-CONDUC+(SR*0.94)                              FWH00710
      IF(U.EQ.TTEQ) U=U+.01                                             FWH00720
      K=-FLUX/(U-TTEQ)                                                  FWH00730
      IF(K.LT.0.0)K=0.0                                                 FWH00740
      CALL FKKK(N,K,I)                                                  FWH00750
      IF(IBUG.GT.3)WRITE(IRR,2003)I,MOLO,NNM,K,FLUX                     FWH00760
 2003 FORMAT(' *** WHTF-DEBUG *** I,MOLO,NNM,K,FLUX = ',3I3,2(1X,E11.5))FWH00770
C                                                                       FWH00780
C     HEAT LOSS COEFFICIENTS FOR FLOW THROUGH REACHES                   FWH00790
      ZR1=K*ZA1(I)/62.4/QEXIT(N,1)                                      FWH00800
      ZR2=K*ZA2(I)/62.4/QEXIT(N,1)                                      FWH00810
      ZR3=K*ZA3(I)/62.4/QEXIT(N,1)                                      FWH00820
C                                                                       FWH00830
C     DETERMINE MIXING AT ENTRANCE TO REACH                             FWH00840
      F=(F1+F)/2.                                                       FWH00850
      DS=1.4*SQRT(1+F*F*SQRT(ZH(I)/ZB(I)))                              FWH00860
C     DILUTION REDUCTION DUE TO BOTTOM INTERFERENCE                     FWH00870
      HMX=0.42*F*(ZH(I)/ZB(I))**0.25*SQRT(ZH(I)*ZB(I))                  FWH00880
      REDUC=(0.75*ZHM(I)/HMX)**0.75                                     FWH00890
      IF(REDUC.GT.1.) REDUC=1.                                          FWH00900
      DS=DS*REDUC                                                       FWH00910
      IF(DS.LT.1.0)GO TO 1000                                           FWH00920
      IF(DS.LE.1.5) DS=1.5                                              FWH00930
      IF(F1.GE.99.0) GO TO 1000                                         FWH00940
      QO = QEXIT(N,1)                                                   FWH00950
      TO = TEXIT(N,1)                                                   FWH00960
      QS10 = QSA1(I)/QO                                                 FWH00970
      QS20 = QSA2(I)/QO                                                 FWH00980
C                                                                       FWH00990
C     REACHES WITHOUT SIDEARMS OR FIRST ITERATION                       FWH01000
C     IF(N.GT.190.AND.N.LT.210)WRITE(66,3938)N,QS10,QS20,DS,QO          FWH01010
C3938 FORMAT(1X,I4,1X,4E12.5)                                           FWH01020
  400 IF(QSA1(I).GT.0.0001) GO TO  600                                  FWH01030
      IF ((ZR1+ZR2+ZR3)/DS.GE.170.0) GO TO  450                         FWH01040
      DENOM=DS-(DS-1)*EXP(-(ZR1+ZR2+ZR3)/DS)                            FWH01050
      TEND(I)=TEQ+(TO-TEQ)*EXP(-(ZR1+ZR2+ZR3)/DS)/DENOM                 FWH01060
      TSA1(I)=TEQ+(TO-TEQ)*EXP(-ZR1/DS)/DENOM                           FWH01070
      TSA2(I)=TEQ+(TO-TEQ)*EXP(-(ZR1+ZR2)/DS)/DENOM                     FWH01080
      TENT(I)=TEND(I)                                                   FWH01090
      TSEND(I)=TEND(I)                                                  FWH01100
```

```
      IF(IBUG.GT.4)WRITE(IRR,2012)TEND(I),TSEND(I)                      FWH01110
 2012 FORMAT(' *** WHTF-DEBUG *** AFTER LINE 400, TEND,TSEND = ',       FWH01120
     +       2(1X,E11.4))                                               FWH01130
      GO TO  800                                                        FWH01140
  450 WRITE(IRR,2007)N,I                                                FWH01150
 2007 FORMAT(' *** WHTF-DEBUG *** AT LINE  450: N, I =',2I5)            FWH01160
      GO TO  800                                                        FWH01170
C                                                                       FWH01180
C     CALCULATE HEAT LOSSES, INCLUDING THOSE IN SIDE ARMS               FWH01190
  600 TERM1 = -ZR1/DS - ZR2/(DS-QS10) - ZR3/(DS-QS10-QS20)              FWH01200
      IF(QS20.LE.0.0)GO TO 601                                          FWH01210
      TERM2 = -ZR1/DS - ZR2/(DS-QS10) - ZRS2/QS20                       FWH01220
      GO TO 602                                                         FWH01230
  601 TERM2=-ZR1/DS-ZR2/(DS-QS10)-ZRS2/.009                             FWH01240
  602 TERM3 = -ZR1/DS - ZRS1/QS10                                       FWH01250
      TERM4 = -ZR1/DS - ZR2/(DS-QS10)                                   FWH01260
      IF(TERM1.LT.-50.) TERM1=-50.                                      FWH01270
      IF(TERM2.LT.-50.) TERM2=-50.                                      FWH01280
      IF(TERM3.LT.-50.) TERM3=-50.                                      FWH01290
C                                                                       FWH01300
C     CASE A  PAGE 74 IN NORTH ANNA REPORT (T.R. 232)                   FWH01310
      IF( (DS-1.0) .LT. (QS10+QS20) ) GO TO  620                        FWH01320
      DENOM = DS - (DS-1.0-QS10-QS20)*EXP(TERM1) - QS20*EXP(TERM2) -    FWH01330
     -                QS10*EXP(TERM3)                                   FWH01340
      TEND(I) = TEQ + (TO-TEQ)*EXP(TERM1)/DENOM                         FWH01350
      GO TO  660                                                        FWH01360
C                                                                       FWH01370
C     CASE B  PAGE 76 IN NORTH ANNA REPORT (T.R. 232)                   FWH01380
  620 IF( (DS-1.0) .LT. QS10 ) GO TO  640                               FWH01390
      DENOM = DS - QS10*EXP(TERM3) - (DS-1.0-QS10)*EXP(TERM2)           FWH01400
      TEND(I) = TEQ + (TO-TEQ)*( (DS-QS10-QS20)*EXP(TERM1) +            FWH01410
     +                (1.0-DS+QS10+QS20)*EXP(TERM2) )/DENOM             FWH01420
      GO TO  660                                                        FWH01430
C                                                                       FWH01440
C     CASE C  PAGE 76 IN NORTH ANNA REPORT (T.R. 232)                   FWH01450
  640 DENOM = DS - (DS-1.0)*EXP(TERM3)                                  FWH01460
      TEND(I) = TEQ + (TO-TEQ)*( (DS-QS10-QS20)*EXP(TERM1) +            FWH01470
     +                QS20*EXP(TERM2) + (QS10-DS+1.0)*EXP(TERM3) )/DENOM FWH01480
C                                                                       FWH01490
  660 TSA1(I) = TEQ + (TO-TEQ)*EXP(-ZR1/DS)/DENOM                       FWH01500
      TSA2(I) = TEQ + (TO-TEQ)*EXP(TERM4)/DENOM                         FWH01510
      TENT(I) = TEQ + (TO-TEQ)*(DS-DENOM)/(DS-1.0)/DENOM                FWH01520
      TSEND(I) = TEQ + (TO-TEQ)*EXP(TERM1)/DENOM                        FWH01530
      IF(IBUG.GT.3)WRITE(IRR,2011)I,TENT(I),TEND(I),TSEND(I),QS10,QS20  FWH01540
 2011 FORMAT(' *** WHTF-DEBUG *** AT LINE  660 - I,TENT,TEND,TSEND,QS1',FWH01550
     +       ',QS2 = '/15X,I3,2X,5(1X,E10.5))                           FWH01560
C                                                                       FWH01570
C     RECOMPUTE ENTRANCE MIXING                                         FWH01580
  800 U1=(TEXIT(N,1)+TENT(I))/2.0 - 40.0                                FWH01590
      IF(IBUG.GT.4)WRITE(IRR,2013)TEND(I),TSEND(I),XLSA1(I),XLSA2(I)    FWH01600
 2013 FORMAT(' *** WHTF-DEBUG *** AT LINE 800, TEND, TSEND SAL = ',     FWH01610
     +       4(1X,E11.4))                                               FWH01620
C     BETAA=COEFFICIENT OF THERMAL EXPANSION (1/DEG. F)                 FWH01630
      BETAA=(5.17E-2*U1 - 3.84E-4*U1*U1 + 1.5E-6*U1**3) * 1.0E-4        FWH01640
      IF(BETAA.LE.0.) GO TO 1000                                        FWH01650
```

```
      IF(TENT(I).GE.TEXIT(N,1))GO TO 1000                                FWH01660
      F1=QEXIT(N,1)/(ZH(I)*2*ZB(I)*SQRT(BETAA*(TEXIT(N,1)-TENT(I))      FWH01670
     . *32.17*ZH(I)))/86400.                                            FWH01680
C                                                                        FWH01690
C     SIDE ARM CALCULATIONS AND CONVERGENCE CHECK                        FWH01700
      ZLR=(ZA1(I)+ZA2(I)+ZA3(I))/ZBR(I)                                  FWH01710
      QCFS=QEXIT(N,1)/86400.                                             FWH01720
      ULAYD1=(0.005*QCFS*QCFS*DS*DS*DS*ZLR/BETAA/(TEXIT(N,1)-TENT(I))    FWH01730
     . /32.17/ZBR(I)/ZBR(I))**.25                                        FWH01740
      XYXD=ULAYD1                                                        FWH01750
      FROUDE(I)=F1                                                       FWH01760
      DISDIL(I)=DS                                                       FWH01770
      ULAYD2=ULAYD1                                                      FWH01780
      ULAYD3=ULAYD1                                                      FWH01790
      QQSA1=QSA1(I)                                                      FWH01800
      QQSA2=QSA2(I)                                                      FWH01810
C     IF FULLY MIXED, SKIP SIDE ARM CALCS                                FWH01820
C    IT SEEMS UNUSUAL FOR THIS CRITERION TO EXIST AS A CUT OFF FOR SIDEA FWH01830
C    RM FLOW;ESPECIALLY SINCE THE DATA SHOW FLOWS EXISTING FOR TEMPS<63  FWH01840
C    DEG FAR                                                            FWH01850
C     IF(BETAA.LE.0.0001)GO TO 1600                                      FWH01860
      IF(TENT(I).GE.TEXIT(N,1))GO TO 1600                                FWH01870
C     CALCULATE SIDE ARM FLOWS                                           FWH01880
      IF(NSA(I).EQ.0) GO TO 1200                                         FWH01890
      II=6                                                               FWH01900
      IF(I.EQ.2)II=4                                                     FWH01910
      CALL SIDARM(ZASA1(I),ULAYD1,ULAYD2,TSA1(I),ZRS1,                   FWH01920
     +          QSA1(I),DUM,HSA1(I),BSA1(I),XLSA1(I),TTEQ,N,II,SAFACT)   FWH01930
      ZRS1=ZRS1/QQ*86400.0                                               FWH01940
      QLIM = QLIM1(I)*DS*QEXIT(N,1)                                      FWH01950
      IF(QSA1(I).GT.QLIM) QSA1(I)=QLIM                                   FWH01960
      QSA1(I)=QSA1(I)*SAFACT                                             FWH01970
      II=7                                                               FWH01980
      IF(I.EQ.2)II=5                                                     FWH01990
      IF((NSA(I)-1).GT.0) CALL SIDARM(ZASA2(I),ULAYD2,ULAYD3,TSA2(I),    FWH02000
     +     ZRS2,QSA2(I),DUM,HSA2(I),BSA2(I),XLSA2(I),TTEQ,N,II,SAFACT)   FWH02010
      ZRS2=ZRS2/QQ*86400.0                                               FWH02020
      QLIM = QLIM2(I)*(DS*QEXIT(N,1)-QSA1(I))                            FWH02030
      IF(QSA2(I).GT.QLIM)QSA2(I)=QLIM                                    FWH02040
      QSA2(I) = QSA2(I)*SAFACT                                           FWH02050
C                                                                        FWH02060
C     TEST FOR CONVERGENCE OF SIDE ARM FLOW RATES AND FROUDE NUMBER      FWH02070
      IF(ABS(F-F1).GE.0.05)GO TO  200                                    FWH02080
      MOLO=0                                                             FWH02090
      IF((ABS(QQSA1-QSA1(I))).LE.(QSA1(I)/50.0).AND.                     FWH02100
     +       (ABS(QQSA2-QSA2(I))).LE.(QSA2(I)/50.0))GO TO 1600           FWH02110
      QSA1(I) = WGT*QSA1(I) + (1.0-WGT)*QQSA1                            FWH02120
      QSA2(I)=WGT*QSA2(I) + (1.0-WGT)*QQSA2                              FWH02130
      NNM=NNM+1                                                          FWH02140
      IF(NNM.LT.20)GO TO  200                                            FWH02150
      GO TO 1400                                                         FWH02160
C                                                                        FWH02170
C     FULLY MIXED REACH - NO SIDEARM FLOW, PLUG FLOW THROUGH REACH       FWH02180
 1000 DS=1.0                                                             FWH02190
      DISDIL(I)=DS                                                       FWH02200
```

```
      F1=99.99                                                          FWH02210
      FROUDE(I)=F1                                                      FWH02220
      ULAYD1=ZHM(I)                                                     FWH02230
      ULAYD2=ULAYD1                                                     FWH02240
      ULAYD3=ULAYD1                                                     FWH02250
      QSA1(I)=0.0                                                       FWH02260
      QSA2(I)=0.0                                                       FWH02270
      TSA1(I) = TEQ + (TO-TEQ)*EXP(-ZR1)                                FWH02280
      TSA2(I) = TEQ + (TO-TEQ)*EXP(-ZR1-ZR2)                            FWH02290
      TEND(I) = TEQ + (TO-TEQ)*EXP(-ZR1-ZR2-ZR3)                        FWH02300
      TSEND(I)=TEND(I)                                                  FWH02310
      TENT(I) = TO                                                      FWH02320
      ZRS1 = 0.0                                                        FWH02330
      ZRS2 = 0.0                                                        FWH02340
      IF(IBUG.GT.1)WRITE(IRR,2008)TEND(I),TSEND(I)                      FWH02350
 2008 FORMAT(' *** WHTF-DEBUG *** AT LINE 1000: TEND,TSEND =',          FWH02360
     +       2(1X,E11.4))                                               FWH02370
      IF(MOLO.GT.20)GO TO 1400                                          FWH02380
C     TEST FOR K CONVERGENCE                                            FWH02390
      IF(ABS(K-OK).GT.1.0)GO TO  200                                    FWH02400
C     TEST FOR FROUDE NUMBER CONVERGENCE                                FWH02410
 1200 IF(ABS(F-F1).GT.0.05)GO TO  200                                   FWH02420
      GO TO 1600                                                        FWH02430
C                                                                       FWH02440
C     CONVERGENCE FAILURE AFTER 20 TRYS                                 FWH02450
 1400 WRITE(6,3000) I                                                   FWH02460
 3000 FORMAT(' *** WHTF *** CONVERGENCE PROBLEM IN LOOP 2, REACH',I2)   FWH02470
C                                                                       FWH02480
C     SOLUTION HAS CONVERGED - END OF ITERATIONS                        FWH02490
 1600 ULAY1(I)=ULAYD1                                                   FWH02500
      ULAY2(I)=(ULAYD1+ULAYD2)/2.                                       FWH02510
      ULAY3(I)=(ULAYD2+ULAYD3)/2.                                       FWH02520
      ULAY4(I)=ULAYD3                                                   FWH02530
      IF(QS10.GT.0.0)TRS1(I) = TEQ + (TSA1(I)-TEQ)*EXP(-ZRS1/QS10)      FWH02540
      IF(QS10.LE.0.0)TRS1(I) = TEQ                                      FWH02550
      IF(QS20.GT.0.0)TRS2(I) = TEQ + (TSA2(I)-TEQ)*EXP(-ZRS2/QS20)      FWH02560
      IF(QS20.LE.0.0)TRS2(I) = TEQ                                      FWH02570
C   CALCULATE FROUDE NUMBER FOR FIRST SIDE ARM                          FWH02580
C                                                                       FWH02590
      UBET=((TSA1(I)+TRS1(I))/2.)-40.                                   FWH02600
      BETAA=((5.17E-2*UBET)-3.84E-4*(UBET**2)+1.5E-6*(UBET**3))*1.0E-4  FWH02610
      IF(BETAA.LE.0.0)GO TO 3938                                        FWH02620
      IF((TSA1(I)-TRS1(I)).LE.0.0)GO TO 3938                            FWH02630
      FFSA1=(QSA1(I)/(86400.*20.*110.))/(BETAA*(TSA1(I)-TRS1(I))*32.    FWH02640
     *2*40./2.)**0.5                                                    FWH02650
      GO TO 3937                                                        FWH02660
 3938 FFSA1=99.99                                                       FWH02670
 3937 CONTINUE                                                          FWH02680
C     CALL WLOSS(EVAP,ZASA1(I),DT,WLTWH)                                FWH02690
C     CALL WLOSS(EVAP,ZASA2(I),DT,WLTWH)                                FWH02700
C     CALL WLOSS(EVAP,ASUM(I),DT,WLTWH)                                 FWH02710
      IF(IBUG.EQ.1)WRITE(IRR,2006)N,I,K                                 FWH02720
 2006 FORMAT(1X,I6,'REACH ',I2,2X, 'K = ',F8.2)                         FWH02730
                                                                        FWH02740
C     ACCOUNT FOR RESIDENCE TIME IN REACH I                             FWH02750
```

```
      IF(ULAYD1.GT.25.0)ULAYD1=25.0                                     FWH02760
      IF(ULAYD2.GT.25.0)ULAYD2=25.0                                     FWH02770
      IF(ULAYD3.GT.25.0)ULAYD3=25.0                                     FWH02780
      IF(ULAYD4.GT.25.0)ULAYD4=25.0                                     FWH02790
C                                                                       FWH02800
C     OPTIONS OF RESIDENCE TIME CALCULATION                             FWH02810
C                                                                       FWH02820
      IF (ISELCT(17).EQ.1) GOTO 997                                     FWH02830
      RESTIM=ULAYD1*ZA1(I)/(DS*QEXIT(N,1))+ZA2(I)*ULAYD2/(DS*QEXIT(N,   FWH02840
     & 1)-QSA1(I))+ZA3(I)*ULAYD3/(DS*QEXIT(N,1)-QSA1(I)-QSA2(I))        FWH02850
      GO TO 998                                                         FWH02860
C     THE ABOVE CALCULATION DOESN'T CONSIDER VOLUME OF THE SIDE ARMS    FWH02870
C     AND THAT OF THE BOTTOM LAYERS WHICH COULD BE VERY SIGNIFICANT,    FWH02880
C     SO IT IS REPLACED BY THE FOLLOWING: ... E.HO 15 MARCH 1984        FWH02890
997   RESTIM=((ZA1(I)+ZA2(I)+ZA3(I))*25.0+                              FWH02900
     &        HSA1(I)*ZASA1(I)+HSA2(I)*ZASA2(I))/QPLNT(N)               FWH02910
C     CORRECTION FACTOR FOR RESTIM                                      FWH02920
      RESTIM=RESTIM*0.75                                                FWH02930
998   IF(IBUG.GT.0)WRITE(20,2001)N,I,RESTIM                             FWH02940
 2001 FORMAT(' *** WHTF-DEBUG *** N,I,RESTIM = ',2I8,E15.6)             FWH02950
      IF(RESTIM.LT.0.0) RESTIM = DT                                     FWH02960
      XLAGT=RESTIM/DT                                                   FWH02970
      LAGTYM=XLAGT+.5                                                   FWH02980
      IF(IBUG.GT.1)WRITE(IRR,2002)N,I,LAGTYM                            FWH02990
 2002 FORMAT(' *** WHTF-DEBUG *** N,I,LAGTYM = ',3I8)                   FWH03000
C     THE FOLLOWING CARD WAS MADE NECESSARY BY ABRUPT CHANGES IN        FWH03010
C     METEOROLOGICAL CONDITIONS.IN NO WAY IS IT GOOD AND THE PROBLEM    FWH03020
C     SHOULD BE SOLVED OTHERWISE IN FURTHER VERSIONS OF THIS PROGRAM    FWH03030
C     IF(LAGTYM.GT.1) LAGTYM=1                                          FWH03040
C   THE ABOVE CARD WAS ESSENTIALLY DELETED TO ALLOW A                   FWH03050
C    DYNAMICALLY CALCULATED LAGTIME TO BE USED....SW                    FWH03060
      IF(ISELCT(2).GT.0)LAGTYM=LAGFIX(I)                                FWH03070
      NN=N+LAGTYM                                                       FWH03080
      QQSA1=QSA1(I)/86400.                                              FWH03090
      QQSA2=QSA2(I)/86400.                                              FWH03100
      IF(I.EQ.2)WRITE(60,3939)N,BETAA,QQSA1,QQSA2,TRS1(I),TRS2(I),TSA1(IFWH03110
     *),TSA2(I),XYXD,DS,FFSA1                                           FWH03120
 3939 FORMAT(I4,1X,E9.3,6(1X,F7.2),3(1X,F5.2))                          FWH03130
      IF(TEND(I).GE.150.) TEND(I)=TEXIT(N,1)                            FWH03140
      TFIN(NN,I)=TEND(I)                                                FWH03150
      IF(TFIN(N,I).EQ.TZERO.AND.N.GT.1) TFIN(N,I)=TFIN(NN,I)            FWH03160
      TEXIT(N,1)=TFIN(N,I)                                              FWH03170
      QFIN(NN)=QEXIT(N,1)                                               FWH03180
      IF(LAGTYM.GE.N) TEXIT(N,1)=TEND(I)                                FWH03190
      IF(LAGTYM.GE.N) TFIN(N,I)=TEND(I)                                 FWH03200
      IF(IBUG.GT.1)WRITE(IRR,2004)I,ZR1,ZR2,ZR3,ZRS1,ZRS2,F1,TENT(I),  FWH03210
     +                    DENOM                                         FWH03220
      IF(IBUG.GT.1)WRITE(IRR,9993)I,F1,TENT(I),TEND(I),TSEND(I),ULAY1   FWH03230
 9993 FORMAT('I F1 TENT(I) TEND(I) TSEND(I) ULAY',I2,5(1X,E10.4))       FWH03240
 2004 FORMAT(' *** WHTF-DEBUG *** STEADY STATE FOR REACH ',I1/          FWH03250
     +       6X,'ZR1        ZR2        ZR3        ZRS1       ',       FWH03260
     +          'ZRS2       F1          TENT'/6X,7(E9.4,1X)/          FWH03270
     +       6X,'DENOM' /6X,7(E9.4,1X))                                 FWH03280
 1800 CONTINUE                                                          FWH03290
      RETURN                                                            FWH03300
```

```
      END                                                               FWH03310
      S[redacted]REA,HO,HONEW,TO,R,Q,EVAP,HMAX,BSA,LSA,TTEQ,N,I,FWH03320
     *SAFACT)                                                           FWH03330
C                                                                       FWH03340
C     DETERMINES THE FLOW IN THE SIDE ARM AND THE RETURNING TEMPERATURE FWH03350
C     AS A FUNCTION OF THE MAIN POND CONDITIONS AT THE ENTRANCE         FWH03360
C                                                                       FWH03370
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                               FWH03380
      COMMON /IOPUT/ IREAD,IRITE,IRR                                    FWH03390
      COMMON /METEOC/ EVAP1,RAD,TAIR,PSI,CONDUC,SR,LWRI                 FWH03400
      REAL LWRI,LWRO                                                    FWH03410
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                          FWH03420
      COMMON /TIME/ ATIME                                               FWH03430
      REAL K,LSA                                                        FWH03440
      IBUG = IDEBUG(3)                                                  FWH03450
      IF(TO.LE.TEQ) GO TO 10                                            FWH03460
      IF(TO.LT.39.)GO TO 10                                             FWH03470
      W=WINDY                                                           FWH03480
      DAY=ATIME                                                         FWH03490
      EA= PSI*25.4*EXP(17.62-9500.8/(TAIR+460))                         FWH03500
      TAIRV=(TAIR+460)/(1.-.378*EA/PRAIR)-460                           FWH03510
      U=TO                                                              FWH03520
C                                                                       FWH03530
C CALCULATE HEAT LOSS COEFFICIENT                                       FWH03540
      LWRO=4.E-8*(U+460)**4                                             FWH03550
      EW=25.4*EXP(17.62-9500.8/(U+460.))                                FWH03560
      TWV=(U+460.)/(1.-.378*EW/PRAIR)-460.                              FWH03570
      DTV=TWV-TAIRV                                                     FWH03580
      DTVL=0.0024*W*W*W                                                 FWH03590
      IF(DTV.LT.DTVL) DTV=DTVL                                          FWH03600
C                                                                       FWH03610
C     NOTE THE SEASONAL VARIABILITY IN THE EVAP FUNCTION                FWH03620
C     EFACT1=EFACT+0.25*SIN(DAY*2.*3.1416/365.25-0.4576)                FWH03630
      EFACT1=EFACT                                                      FWH03640
      EVAP=EFACT1*(22.4*(DTV**.3334)+14.*W)*(EW-EA)                     FWH03650
      BR=.255*(U-TAIR)/(EW-EA)                                          FWH03660
      CONDUC=BR*EVAP                                                    FWH03670
      IF(EVAP.LT.0.) EVAP=0.                                            FWH03680
      FLUX=LWRI-LWRO-EVAP-CONDUC+(SR*0.94)                              FWH03690
      K=-FLUX/(U-TTEQ)                                                  FWH03700
      IF(IBUG.GT.2)WRITE(IRR,2000)K,FLUX,U,TTEQ                         FWH03710
      CALL FKKK(N,K,I)                                                  FWH03720
 2000 FORMAT(' *** SIDARM-DEBUG *** K,FLUX,U,TEQ = ',4(1X,E11.4))       FWH03730
      TSTAR=TO-0.6*(TO-TEQ)                                             FWH03740
      IF(TSTAR.GT.39.)GO TO 2008                                        FWH03750
      TSTAR=39.                                                         FWH03760
 2008 UBETA=(TSTAR+TO)/2.                                               FWH03770
      UABS = UBETA-40.0                                                 FWH03780
      BETAA = (5.17E-2*UABS - 3.84E-4*UABS*UABS +                       FWH03790
     +      1.5E-6*UABS**3) * 1.0E-4                                    FWH03800
      IF(IBUG.GT.2)WRITE(IRR,2002)BETAA                                 FWH03810
 2002 FORMAT(' *** SIDARM-DEBUG *** BETAA = ',E11.4)                    FWH03820
      ENTRY SIDE(AREA,HO,HONEW,TO,R,Q,K1,BETAA1,HMAX,BSA,LSA)           FWH03830
      IF(IBUG.GT.2)WRITE(IRR,2933)K,HMAX,LSA,BETAA                      FWH03840
 2933 FORMAT(' *** SD-DEBUG*** K,HMAX,LSA,BETA',4(1X,E11.4))            FWH03850
```

specifies the first
[illegible] to be [illegible]

```
C                                                                       FWH03860
C DETERMINATION OF THE SIDE ARM FLOW                                    FWH03870
C                                                                       FWH03880
      SK = K/(62.4*86400.0)                                             FWH03890
      B = SK**2 * LSA**2 / (BETAA*(TO-TEQ)*32.2* HMAX**3)               FWH03900
      IF(B.LE.0.) GO TO 10                                              FWH03910
      FO=0.02                                                           FWH03920
      H=HO/HMAX                                                         FWH03930
      IF(H.GT.0.5) H=0.5                                                FWH03940
      ALP = 0.5                                                         FWH03950
      HF1 = 1.0 / H                                                     FWH03960
      HF2 = 1.0 / (1.0-H)                                               FWH03970
      BF = B*FO*LSA / (HMAX*4.0*H)                                      FWH03980
      BF = BF * (ALP*(HF1+HF2)**3 + HF2**3)                             FWH03990
      OMEGA=BF**(-0.333)                                                FWH04000
      Q=OMEGA*SK*LSA*BSA                                                FWH04010
      IF(IBUG.GT.2)WRITE(IRR,2001)Q,B,BF,OMEGA,SK                       FWH04020
 2001 FORMAT(' *** SIDARM-DEBUG *** Q,B,BF,OMEGA,SK = ',5(1X,E11.4))    FWH04030
      IF(Q.LE.0.0) GO TO 10                                             FWH04040
      IF(TSTAR.EQ.39.)GO TO 2009                                        FWH04050
      GO TO 2010                                                        FWH04060
 2009 AREA1=1.25*(Q*SAFACT)*(ALOG((39.-TEQ)/(TO-TEQ)))/SK               FWH04070
      GO TO 20                                                          FWH04080
 2010 AREA1=AREA                                                        FWH04090
   20 R=.8*SK*AREA1                                                     FWH04100
      Q=Q*86400.0                                                       FWH04110
      RETURN                                                            FWH04120
   10 Q=0.                                                              FWH04130
      R=0.0                                                             FWH04140
      RETURN                                                            FWH04150
      END                                                               FWH04160
```

```
                                                                      FWH00010
C                                                                     FWH00020
C     SUBROUTINE REPRINTS WHTF AND DISCHARGE CHANNEL INPUT DATA       FWH00030
C                                                                     FWH00040
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                             FWH00050
      COMMON /IOPUT/ IREAD,IRITE,IRR                                  FWH00060
      COMMON /CONST/ RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON FWH00070
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH   FWH00080
     *,MIXD(5)                                                        FWH00090
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR    FWH00100
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2)                         FWH00110
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR        FWH00120
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR FWH00130
      COMMON /METEOA/ WIND(366),TA(366),SIGH(366),FIN(366),CLOUD(366) FWH00140
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                FWH00150
      REAL LWRI,LWRO                                                  FWH00160
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                        FWH00170
      COMMON /METEOE/ NMET,DTMET,IOMET,NDIS,DTDIS,IODIS,ISMET,ISDIS   FWH00180
      COMMON /PARAM2/ RTIME,RDATE,NSTEP                               FWH00190
      DOUBLE PRECISION RTIME,RDATE                                    FWH00200
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB         FWH00210
      COMMON /ENERGY/ EO,E1,E2,E3,ET,FLUXT                            FWH00220
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                               FWH00230
      COMMON /WHTF1/ ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO, FWH00240
     +               NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4), FWH00250
     +               HSA2(4),ZBR(4),XLSA1(4),XLSA2(4)                 FWH00260
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893), FWH00270
     . QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)  FWH00280
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),  FWH00290
     +               QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH     FWH00300
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                  FWH00310
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                         FWH00320
      COMMON /PLANT/ QI(1893),TIN(1893)                               FWH00330
C                                                                     FWH00340
C     ECHO PRINT INPUT DATA                                           FWH00350
C                                                                     FWH00360
      IF(ISELCT(1).GT.0)GO TO 85                                      FWH00370
      WRITE(IRITE,1001)                                               FWH00380
      WRITE(IRITE,1004)NCH,XCH,WCH,DCH,ACH,DCHX,WCHX                  FWH00390
      WRITE(IRITE,1005)                                               FWH00400
      WRITE(IRITE,1006)                                               FWH00410
         DO 70 I=1,NREACH                                             FWH00420
         WRITE(IRITE,1007)I,ZH(I),ZB(I),ZHM(I),ZBR(I),ZA1(I),ZA2(I),  FWH00430
     +                 ZA3(I),LAGFIX(I),NSA(I)                      FWH00440
 70      CONTINUE                                                     FWH00450
      IF(ISELCT(2).GT.0)WRITE(IRITE,1016)                             FWH00460
      IF(ISELCT(2).EQ.0)WRITE(IRITE,1017)                             FWH00470
      WRITE(IRITE,1014)                                               FWH00480
      WRITE(IRITE,1009)                                               FWH00490
         DO 80 I=1,NREACH                                             FWH00500
         IF(NSA(I).EQ.0)GO TO 80                                      FWH00510
         WRITE(IRITE,1010)I,ZASA1(I),BSA1(I),HSA1(I),XLSA1(I),ZASA2(I), FWH00520
     +        BSA2(I),HSA2(I),XLSA2(I)                                FWH00530
 80      CONTINUE                                                     FWH00540
      IF(SAFACT.GT.0.0)WRITE(IRITE,1018)SAFACT                        FWH00550
```

```
      IF(NREACH.EQ.4)WRITE(IRITE,1019)NREACH                            FWH00560
 85   CONTINUE                                                          FWH00570
C                                                                       FWH00580
C     INITIALIZE VARIABLES                                              FWH00590
C                                                                       FWH00600
      SAFACT=1.0-SAFACT                                                 FWH00610
      WLTN=0.0                                                          FWH00620
      WLTWH=0.0                                                         FWH00630
         DO 100 I=1,1893                                                FWH00640
         QEXIT(I,1)=0.0                                                 FWH00650
         TEXIT(I,1)=0.0                                                 FWH00660
         QFIN(I)=0.0                                                    FWH00670
            DO 90 J=1,NREACH                                            FWH00680
            TFIN(I,J)=TZERO                                             FWH00690
 90         CONTINUE                                                    FWH00700
 100     CONTINUE                                                       FWH00710
C                                                                       FWH00720
C     UNIT CONVERSIONS AND PARAMETER CALCULATIONS                       FWH00730
C                                                                       FWH00740
         DO 110 I=1,NREACH                                              FWH00750
C        WIDTHS TO HALF-WIDTHS                                          FWH00760
         ZB(I)=0.5*ZB(I)                                                FWH00770
C        ACRES TO SQUARE FEET                                           FWH00780
         ZA1(I)=ZA1(I)*43560.0                                          FWH00790
         ZA2(I)=ZA2(I)*43560.0                                          FWH00800
         ZA3(I)=ZA3(I)*43560.0                                          FWH00810
         ZASA1(I)=ZASA1(I)*43560.0                                      FWH00820
         ZASA2(I)=ZASA2(I)*43560.0                                      FWH00830
C        DETERMINE SIDE ARM FLOW LIMITS                                 FWH00840
         IF(NSA(I).EQ.0)GO TO 110                                       FWH00850
         SHRAT1 = (BSA1(I)**0.66)*(ZASA1(I)**0.33) /                    FWH00860
     +     ((ZBR(I)**0.66)*((ZA1(I)+ZA2(I))**0.33) +                    FWH00870
     +     (BSA2(I)**0.66)*(ZASA2(I)**0.33))                            FWH00880
         QLIM1(I) = SHRAT1/(SHRAT1+1.0)                                 FWH00890
         IF(NSA(I).EQ.1)GO TO 110                                       FWH00900
         SHRAT2 = (BSA2(I)**0.66)*(ZASA2(I)**0.33)/                     FWH00910
     +     ((ZBR(I)**0.66)*(ZA2(I)**0.33))                              FWH00920
         QLIM2(I) = SHRAT2/(SHRAT2+1.0)                                 FWH00930
 110     CONTINUE                                                       FWH00940
         ACH=ACH*43560.0                                                FWH00950
      RETURN                                                            FWH00960
C                                                                       FWH00970
C     FORMAT STATEMENTS                                                 FWH00980
C                                                                       FWH00990
 1001 FORMAT('1')                                                       FWH01000
 1002 FORMAT(' ',20A4/' ',80('=')//)                                    FWH01010
 1004 FORMAT(5X,'DISCHARGE CHANNEL PARAMETERS'/5X,28('-')//             FWH01020
     +        10X,'NUMBER OF SEGMENTS',T45,I6/                          FWH01030
     +        10X,'CHANNEL LENGTH',T42,F9.2,2X,'(FEET)'/                FWH01040
     +        10X,'AVERAGE WIDTH',T45,F6.2,2X,'(FEET)'/                 FWH01050
     +        10X,'AVERAGE DEPTH',T45,F6.2,2X,'(FEET)'/                 FWH01060
     +        10X,'AREA',T45,F6.2,2X,'(ACRES)'/                         FWH01070
     +        10X,'DEPTH AT EXIT',T45,F6.2,2X,'(FEET)'/                 FWH01080
     +        10X,'WIDTH AT EXIT',T45,F6.2,2X,'(FEET)'/)                FWH01090
 1005 FORMAT(5X,'WHTF REACH PARAMETERS'/5X,21('-')/)                    FWH01100
```

to initialize all entries of the arrays QEXIT & TEXIT

```
C       TO FIT A SINE CURVE THROUGH PREDICTIONS AT DIKE 3                CUR00010
C       M*SIN(WT+PHI)=A*SIN(WT)+B*COS(WT)                                CUR00020
        DIMENSION PRED(1893)                                             CUR00030
        READ (20,1000) PRED                                              CUR00040
 1000   FORMAT (24X,F6.2)                                                CUR00050
        DO 9 I=1,1893                                                    CUR00060
      9 PRED(I)=(PRED(I)-32.)*5/9.                                       CUR00070
        PRODA=0.0                                                        CUR00080
        PRODB=0.0                                                        CUR00090
        DO 1 I=160,1620                                                  CUR00100
        PRODA=PRED(I)*SIN(2.*3.1416*(I-159)/365.25)+PRODA                CUR00110
      1 PRODB=PRED(I)*COS(2.*3.1416*(I-159)/365.25)+PRODB                CUR00120
        A=PRODA*2./4./365.25                                             CUR00130
        B=PRODB*2./4./365.25                                             CUR00140
        WRITE (5,1010) A,B                                               CUR00150
 1010   FORMAT (5X,'A=',F20.4,/,5X,'B=',F20.4)                           CUR00160
        STOP                                                             CUR00170
        END                                                              CUR00180
```

```
C       TO FIT A SINE CURVE THROUGH THE RAW ERROR TIME SERIES AT DIKE 3   CUR00010
C       M*SIN(WT+PHI)=A*SIN(WT)+B*COS(WT)                                 CUR00020
        DIMENSION ERR(1893)                                               CUR00030
        READ (20,1000) ERR                                                CUR00040
 1000   FORMAT (8F10.2)                                                   CUR00050
        PRODA=0.0                                                         CUR00060
        PRODB=0.0                                                         CUR00070
        DO 1 I=160,1620                                                   CUR00080
        PRODA=ERR(I)*SIN(2.*3.1416*(I-159)/365.25)+PRODA                  CUR00090
      1 PRODB=ERR(I)*COS(2.*3.1416*(I-159)/365.25)+PRODB                  CUR00100
        A=PRODA*2./4./365.25                                              CUR00110
        B=PRODB*2./4./365.25                                              CUR00120
        WRITE (5,1010) A,B                                                CUR00130
 1010   FORMAT (5X,'A=',F10.4,/,5X,'B=',F10.4)                            CUR00140
        STOP                                                              CUR00150
        END                                                               CUR00160
```

```
      DIMENSION ERR1(1893),ERR2(1893),ERR(1893)                          SUB00010
      READ(15,100)ERR1                                                   SUB00020
      READ(20,100)ERR2                                                   SUB00030
  100 FORMAT(8F10.2)                                                     SUB00040
      DO 200 J=1,1893                                                    SUB00050
      ERR(J)=ERR1(J)-ERR2(J)                                             SUB00060
      IF(ERR1(J).EQ.50.0.OR.ERR2(J).EQ.50.0)ERR(J)=50.0                  SUB00070
  200 CONTINUE                                                           SUB00080
      WRITE(30,100)ERR                                                   SUB00090
      STOP                                                               SUB00100
      END                                                                SUB00110
```

```
      DIMENSION T1(1893),T2(1893),T3(1893),T4(1893)                    ERR00010
      DIMENSION R1(1967),R2(1967),R3(1967),R4(1967)                    ERR00020
      DIMENSION ERR1(1893),ERR2(1893),ERR3(1893),ERR4(1893)            ERR00030
      DIMENSION RR1(1893),RR2(1893),RR3(1893),RR4(1893)                ERR00040
C                                                                      ERR00050
C     INPUT MODEL PREDICTION                                           ERR00060
C                                                                      ERR00070
      DO 11 J=1,1893                                                   ERR00080
11    READ (17,155) T1(J),T2(J),T4(J),T3(J)                            ERR00090
C                                                                      ERR00100
C     INPUT NRC ENDECO MEASURMENTS                                     ERR00110
C                                                                      ERR00120
      READ (14,100) R1                                                 ERR00130
      READ (15,100) R2                                                 ERR00140
      READ (16,100) R3                                                 ERR00150
      READ (18,100) R4                                                 ERR00160
      DO 200 J=1,1893                                                  ERR00170
      K=55+J                                                           ERR00180
      RR1(J)=R1(K)                                                     ERR00190
      RR2(J)=R2(K)                                                     ERR00200
      RR3(J)=R3(K)                                                     ERR00210
200   RR4(J)=R4(K)                                                     ERR00220
C                                                                      ERR00230
C     CALCULATE RAW ERROR (PREDICTION-MEASUREMENT) IN DEGREE CELSIUS   ERR00240
C                                                                      ERR00250
      DO 333 J=1,1893                                                  ERR00260
      ERR1(J)=(T1(J)-RR1(J))*5./9.                                     ERR00270
      IF (RR1(J).EQ.0.0) ERR1(J)=50.0                                  ERR00280
      ERR2(J)=(T2(J)-RR2(J))*5./9.                                     ERR00290
      IF (RR2(J).EQ.0.0) ERR2(J)=50.0                                  ERR00300
      ERR3(J)=(T3(J)-RR3(J))*5./9.                                     ERR00310
      IF (RR3(J).EQ.0.0) ERR3(J)=50.0                                  ERR00320
      ERR4(J)=(T4(J)-RR4(J))*5./9.                                     ERR00330
      IF (RR4(J).EQ.0.0) ERR4(J)=50.0                                  ERR00340
333   CONTINUE                                                         ERR00350
      WRITE (20,100) ERR1                                              ERR00360
      WRITE (21,100) ERR2                                              ERR00370
      WRITE (22,100) ERR3                                              ERR00380
      WRITE (23,100) ERR4                                              ERR00390
100   FORMAT (8F10.2)                                                  ERR00400
155   FORMAT (12X,F6.2,6X,F6.2,6X,F6.2,F6.2)                           ERR00410
      STOP                                                             ERR00420
      END                                                              ERR00430
```

```
      CALL OUTDAT(I,SIGH)                                                NEWO1110
      CALL OUTDAT(I,WIND)                                                NEWO1120
      CALL OUTDAT(I,CLOUD)                                               NEWO1130
      CALL OUTDAT(I,FIN)                                                 NEWO1140
      STOP                                                               NEWO1150
C                                                                        NEWO1160
C     FORMAT STATEMENTS                                                  NEWO1170
 5    FORMAT(7X,3I2,1X,I2,I3,1X,6(1X,F9.3))                              NEWO1180
 6    FORMAT(20A4)                                                       NEWO1190
 7    FORMAT(8F10.0)                                                     NEWO1200
 8    FORMAT(16I5)                                                       NEWO1210
1001 FORMAT('1')                                                         NEWO1220
1002 FORMAT(5X,I2,'/',I2,'/',I2,7X,6F10.2,5X,3I2)                        NEWO1230
1003 FORMAT('1',20A4/' ',80('=')//)                                      NEWO1240
1004 FORMAT(26X,'AIR',7X,'REL.',6X,'WIND',5X,'CLOUD',4X,'SOLAR',         NEWO1250
     +      4X,'CLEAR'/                                                 NEWO1260
     +      7X,'DATE',5X,4X,5X,'TEMP.',6X,'HUM.',5X,'SPEED',            NEWO1270
     +      5X,'COVER',5X,'RAD.',5X,'SKY',9X,'FLAGS'/                   NEWO1280
     +      5X,8('-'),2X,5X,5X,5('-'),3(5X,5('-')),3X,7('-'),           NEWO1290
     +      3X,7('-'),6X,5('-')//)                                      NEWO1300
1005 FORMAT(//)                                                          NEWO1310
1006 FORMAT(5X,' DATA WRITTEN TO FILE:')                                 NEWO1320
2000 FORMAT(' *** NEWMET-DEBUG *** RH = ',E12.4)                         NEWO1330
2001 FORMAT(' *** NEWMET-DEBUG *** CC = ',E12.4)                         NEWO1340
      END                                                                NEWO1350
      BLOCK DATA                                                         NEWO1360
C                                                                        NEWO1370
C     CURVE FIT COEFFICIENTS FOR DAILY AVERAGE CLEAR SKY SOLAR           NEWO1380
C     RADIATION SUBROUTINE - CSDAY                                       NEWO1390
C                                                                        NEWO1400
      COMMON /SRCOEF/ SRC1(21), SRC2(21), SRC3(21)                       NEWO1410
      DATA SRC1 /80.155, 79.371, 78.566, 77.604, 76.655, 76.041, 75.060,NEWO1420
     +          74.046, 73.161, 72.248, 71.390, 70.394, 69.350, 68.362,NEWO1430
     +          67.281, 66.240, 65.197, 64.113, 63.010, 61.911, 60.782/NEWO1440
      DATA SRC2 /29.207, 30.236, 31.219, 32.145, 33.156, 34.133, 35.194,NEWO1450
     +          35.938, 36.834, 37.699, 38.598, 39.413, 40.188, 40.982,NEWO1460
     +          41.706, 42.442, 43.128, 43.788, 44.471, 45.020, 45.639/NEWO1470
      DATA SRC3 / 1.679, 1.713, 1.710, 1.740, 1.728, 1.694, 1.737,       NEWO1480
     +            1.734, 1.727, 1.738, 1.721, 1.730, 1.741, 1.739,     NEWO1490
     +            1.742, 1.736, 1.740, 1.739, 1.739, 1.740, 1.735/     NEWO1500
      END                                                                NEWO1510
      SUBROUTINE CSDAY(XLAT,DAY,SR)                                      NEWO1520
C     RETURNS DAILY AVERAGE CLEAR SKY RADIATION AT A FUNCTION OF         NEWO1530
C          LATITUDE, TIME OF YEAR                                        NEWO1540
C     RADIATION IS IN UNITS OF BTU/FT**2/DAY                             NEWO1550
C          REFERERENCE: THACKSTON (1974)  "EFFECT OF GEOGRAPHICAL        NEWO1560
C          VARIATION ON RECIRCULATING COOLING PONDS"  EPA-660/2-74-085   NEWO1570
C                                                                        NEWO1580
      COMMON /IOPUT/ IREAD,IRITE                                         NEWO1590
      COMMON /SRCOEF/ SRC1(21), SRC2(21), SRC3(21)                       NEWO1600
C                                                                        NEWO1610
      LAT = IFIX(XLAT+0.5) - 25                                          NEWO1620
C     EQUATIONS ARE VALID FOR LATITUDES BETWEEN 26 AND 46                NEWO1630
      IF(LAT.GT.21)GO TO 900                                             NEWO1640
      IF(LAT.LT.1)GO TO 900                                              NEWO1650
```

```
      SR = SRC1(LAT) - SRC2(LAT)*SIN(2.0*3.14159*DAY/366.0 + SRC3(LAT)) NEW01660
      SR = SR*24.0                                                      NEW01670
      RETURN                                                            NEW01680
C     ERROR EXITS                                                       NEW01690
 900  WRITE(IRITE,901)XLAT                                              NEW01700
 901  FORMAT(' *** CSDAY *** FATAL ERROR.- LATITUDE OUT OF ALLOWABLE RANNEW01710
     +GE (26 TO 46 DEG)'/14X,'INPUT LATITUDE = ',F5.1)                  NEW01720
      CALL EXIT                                                         NEW01730
      END                                                               NEW01740
C                                                                       NEW01750
      SUBROUTINE OUTDAT(N,DATA)                                         NEW01760
      COMMON /IOPUT/ IREAD,IRITE,ITOUT,ITIN                             NEW01770
      DIMENSION DATA(N)                                                 NEW01780
      WRITE(ITOUT,5)DATA                                                NEW01790
      WRITE(IRITE,1003)                                                 NEW01800
      WRITE(IRITE,5)DATA                                                NEW01810
      WRITE(IRITE,1002)                                                 NEW01820
5     FORMAT(8F10.2)                                                    NEW01830
1002  FORMAT(//)                                                        NEW01840
1003  FORMAT(/' DATA WRITTEN TO FILES:')                                NEW01850
      RETURN                                                            NEW01860
      END                                                               NEW01870
C                                                                       NEW01880
      FUNCTION SATVP(T)                                                 NEW01890
C     RETURNS SATURATION VAPOR PRESSURE FOR INPUT TEMPERATURE           NEW01900
      SATVP = 25.4*EXP(17.62-9500.8/(T+460.0))                          NEW01910
      RETURN                                                            NEW01920
      END                                                               NEW01930
      FUNCTION LEAP(IYEAR)                                              NEW01940
C     COMPUTES LEAP YEAR DAY CORRECTION                                 NEW01950
      LEAP=0                                                            NEW01960
      IF( MOD(IYEAR,4).EQ.0)LEAP=1                                      NEW01970
      IF( MOD(IYEAR,100).EQ.0)LEAP=0                                    NEW01980
      IF( MOD(IYEAR,400).EQ.0)LEAP=1                                    NEW01990
      RETURN                                                            NEW02000
      END                                                               NEW02010
```

```
C     MAIN PROGRAM TO TRANSFORM NORTH ANNA MET DATA TO A FORM                NEW00010
C     COMPATIBLE WITH M.I.T. COOLING LAKE MODEL                              NEW00020
C                                                                            NEW00030
      COMMON /IOPUT/ IREAD,IRITE,ITOUT,ITIN                                  NEW00040
      DIMENSION TA(2000),SIGH(2000),WIND(2000),CLOUD(2000),FIN(2000)         NEW00050
      DIMENSION IDAZE(12)                                                    NEW00060
      DATA IDAZE/0,31,59,90,120,151,181,212,243,273,304,334/                 NEW00070
      DIMENSION TITLE(20)                                                    NEW00080
      IREAD=5                                                                NEW00090
      IRITE=6                                                                NEW00100
      ITOUT=16                                                               NEW00110
      ITIN = 15                                                              NEW00120
      LC = 57                                                                NEW00130
      READ(IREAD,6)TITLE                                                     NEW00140
      READ(IREAD,7)ZWIND,XLAT                                                NEW00150
      READ(IREAD,8)IBUG                                                      NEW00160
C     DETERMINE ADJUSTMENT FACTOR FOR WIND HEIGHT                            NEW00170
      FACTOR = 7.69/ALOG(ZWIND/0.003)                                        NEW00180
      WRITE(IRITE,1003)TITLE                                                 NEW00190
      WRITE(IRITE,1004)                                                      NEW00200
      I=0                                                                    NEW00210
      ILINE = 9                                                              NEW00220
C                                                                            NEW00230
C     LOOP OVER INPUT RECORDS                                                NEW00240
  100    IFLG1=0                                                             NEW00250
         IFLG2=0                                                             NEW00260
         I=I+1                                                               NEW00270
         TA(I)=0.0                                                           NEW00280
         WIND(I)=0.0                                                         NEW00290
         SIGH(I)=0.0                                                         NEW00300
         FIN(I)=0.0                                                          NEW00310
        DO 150 J=1,24                                                        NEW00320
         READ(ITIN,5,END=700)IYR,IMO,IDATE,IHR,IFLGO,WINDY,WDIR,             NEW00330
     +                       TAIR,DEW,SR,CC                                  NEW00340
         TA(I)=TA(I)+TAIR                                                    NEW00350
         WIND(I)=WIND(I)+WINDY                                               NEW00360
         SIGH(I)=SIGH(I)+DEW                                                 NEW00370
      IDAY = IDATE+IDAZE(IMO)                                                NEW00380
      IF(IMO.GT.2) IDAY = IDAY+LEAP(IYR)                                     NEW00390
      DAY = FLOAT(IDAY)                                                      NEW00400
C  CALCULATION OF SOLAR RADIATION IN CAL/CM**2/MIN FROM CLOUD COVER INFO     NEW00410
      IF(IFLGO.NE.004) GO TO 150                                             NEW00420
      CALL CSDAY(XLAT,DAY,CS)                                                NEW00430
      SR = CS*(1.0-0.65*CC**2.)/(1440.0*3.686)                               NEW00440
  150 FIN(I)=FIN(I)+SR                                                       NEW00450
C                                                                            NEW00460
C     AIR TEMPERATURE DATA                                                   NEW00470
C                                                                            NEW00480
         TA(I)=TA(I)/24.0                                                    NEW00490
C     RELATIVE HUMIDITY DATA                                                 NEW00500
C        DETERMINE R.H. FROM WET BULB                                        NEW00510
         DEW = SIGH(I)/24.0                                                  NEW00520
         RH = SATVP(DEW)/SATVP(TA(I))                                        NEW00530
         IF(IBUG.GT.0) WRITE(IRITE,2000)RH                                   NEW00540
         IF(RH.GE.0.0)GO TO 550                                              NEW00550
```

```
            RH = 0.0                                                    NEW00560
            IFLG2 = 1                                                   NEW00570
 550        IF(RH.LE.1.0)GO TO 570                                      NEW00580
            RH = 1.0                                                    NEW00590
            IFLG2 = 2                                                   NEW00600
 570        SIGH(I) = RH                                                NEW00610
C                                                                       NEW00620
C     WIND SPEED DATA                                                   NEW00630
C           ADJUST MEASURING HEIGHT TO 2 METERS                         NEW00640
C           METHOD ASSUMES LOG VELOCITY PROFILE  SEE: RYAN AND HARLEMAN NEW00650
            WIND(I) = WIND(I)/24.0*FACTOR                               NEW00660
C                                                                       NEW00670
C     SOLAR RADIATION DATA                                              NEW00680
C           CONVERT FROM CAL/CM**2/MIN TO BTU/FT**2/DAY                 NEW00690
            SR = FIN(I)/24.0                                            NEW00700
            SR = SR*1440.0                                              NEW00710
            SR = SR*3.686                                               NEW00720
            FIN(I) = SR                                                 NEW00730
C                                                                       NEW00740
C     CLOUD COVER DATA                                                  NEW00750
C           DETERMINE CLEAR SKY SOLAR RADIATION, CS                     NEW00760
            CALL CSDAY(XLAT,DAY,CS)                                     NEW00770
C           USE MEASURED AND CLEAR SKY RADIATION TO FIND CLOUD COVER    NEW00780
            IF(CS.LE.0.0)GO TO 425                                      NEW00790
            CC = (1.0-SR/CS)/0.65                                       NEW00800
            IF(IBUG.GT.0) WRITE(IRITE,2001)CC                           NEW00810
            IF(CC.LT.0.0)GO TO 420                                      NEW00820
            IF(CC.GT.0.0)CC = SQRT(CC)                                  NEW00830
            IF(CC.GT.1.0)GO TO 430                                      NEW00840
            GO TO 450                                                   NEW00850
 420        IFLG1 = 1                                                   NEW00860
 425        CC = 0.0                                                    NEW00870
            GO TO 450                                                   NEW00880
 430        CC = 1.0                                                    NEW00890
            IFLG1 = 2                                                   NEW00900
 450        CLOUD(I) = CC                                               NEW00910
C                                                                       NEW00920
C           WRITE CONVERTED DATA TO LINE PRINTER                        NEW00930
C           AND ALSO CREATE DATE FILE FOR CCCOR PROGRAM                 NEW00940
            WRITE(IRITE,1002)IMO,IDATE,IYR,TA(I),SIGH(I),WIND(I),       NEW00950
     +                       CLOUD(I),FIN(I),CS,IFLGO,IFLG1,IFLG2       NEW00960
      WRITE(17,3000) IYR,IMO,IDATE                                      NEW00970
 3000 FORMAT (3I2)                                                      NEW00980
            ILINE = ILINE+1                                             NEW00990
            IF(ILINE.LT.LC)GO TO 100                                    NEW01000
            WRITE(IRITE,1001)                                           NEW01010
            WRITE(IRITE,1004)                                           NEW01020
            ILINE = 5                                                   NEW01030
            GO TO 100                                                   NEW01040
C                                                                       NEW01050
C     END OF DATA READ IN - FINAL WRITE TO TAPE FILE                    NEW01060
 700  WRITE(IRITE,1005)                                                 NEW01070
      WRITE(ITOUT,6)TITLE                                               NEW01080
C     WRITE(IRITE,1006)                                                 NEW01090
      C     DAT(I,TA)                                                   NEW01100
```

```
C       MAIN PROGRAM                                                    MET00010
        DIMENSION DAT(5)                                                MET00020
        LC = 50                                                         MET00030
        IMLAST = 0                                                      MET00040
        IRITE = 6                                                       MET00050
        IREAD = 5                                                       MET00060
        ILINE = 7                                                       MET00070
        WRITE(IRITE,1002)                                               MET00080
        WRITE(IRITE,1006)                                               MET00090
100     READ(IREAD,5,END=500)IYR,IMO,IDAY,IHR,DAT                       MET00100
        IF(IMO.EQ.IMLAST) GO TO 110                                     MET00110
        ILINE = ILINE+1                                                 MET00120
        WRITE(IRITE,1003)                                               MET00130
        IMLAST = IMO                                                    MET00140
110     CONTINUE                                                        MET00150
C       MASK MISSING DATA                                               MET00160
        DO 120 I = 1,5                                                  MET00170
        IF(DAT(I).LT.0.0001) DAT(I)=9999999.9                           MET00180
120     CONTINUE                                                        MET00190
        WRITE(IRITE,1005)IMO,IDAY,IYR,IHR,DAT                           MET00200
        ILINE = ILINE+1                                                 MET00210
        IF(ILINE.LE.LC) GO TO 100                                       MET00220
        WRITE(IRITE,1001)                                               MET00230
        WRITE(IRITE,1006)                                               MET00240
        ILINE = 3                                                       MET00250
        GO TO 100                                                       MET00260
500     WRITE(IRITE,1003)                                               MET00270
        WRITE(IRITE,1004)                                               MET00280
        STOP                                                            MET00290
C                                                                       MET00300
C       FORMAT STATEMENTS                                               MET00310
5       FORMAT(7X,3I2,1X,I2,5X,5F10.3)                                  MET00320
1001    FORMAT('1')                                                     MET00330
1002    FORMAT('1 METEOROLOGICAL DATA FOR LAKE NORTH ANNA'/             MET00340
     +    ' ',39('=')//)                                               MET00350
1006    FORMAT(7X,'DATE',5X,'HOUR',4X,'WINDSPEED',10X,'WIND',           MET00360
     +         13X,'AIR',11X,'DEW POINT',10X,'SOLAR'/                   MET00370
     +   42X,'DIRECTION',6X,'TEMPERATURE',6X,'TEMPERATURE',8X,         MET00380
     +         'RADIATION',/)                                           MET00390
1003    FORMAT(/)                                                       MET00400
1004    FORMAT(' END OF FILE')                                          MET00410
1005    FORMAT(5X,I2,'/',I2,2X,I2,':00',5(5X,F7.3,5X))                  MET00420
        END                                                             MET00430
```

```
      COMMON/SRCOEF/ SRC1(21),SRC2(21),SRC3(21)                          CCC00010
      DIMENSION SR1(157),SR2(157),SR3(157),CC(157),CS(157)               CCC00020
      DIMENSION CC2(157),CC3(157),IFLAG(157)                             CCC00030
C  SR1=N.ANNA SR;SR2=RICHMOND CS SR;SR3=RICHMOND CORRECTED SR            CCC00040
C  SR4= FINAL COR. SR                                                    CCC00050
C  CC=RICH CC; CC2= METDT4(N.ANNA CC); CC3=FINAL CC                      CCC00060
      DIMENSION IDAZE(12),SR4(157)                                       CCC00070
      DATA IDAZE/0,31,59,90,120,151,181,212,243,273,304,334/             CCC00080
      READ(12,150)IYR,IMO,IDATE,IBUG                                     CCC00090
  150 FORMAT(4I2)                                                        CCC00100
C     READ INPUT DATA                                                    CCC00110
      READ(10,100)SR1                                                    CCC00120
      READ(14,100)CC2                                                    CCC00130
      READ (11,100)CC                                                    CCC00140
  100 FORMAT (8F10.2)                                                    CCC00150
C  THIS PROGRAM DETERMINES THE CLEAR SKY RADIATION FOR THE               CCC00160
C  LOCATION IN QUESTION (SHORT WAVE)                                     CCC00170
C     RADIATION IS IN UNITS OF BTU/FT**2/DAY                             CCC00180
C          REFERERENCE: THACKSTON (1974)  "EFFECT OF GEOGRAPHICAL        CCC00190
C          VARIATION ON RECIRCULATING COOLING PONDS"  EPA-660/2-74-085.  CCC00200
C                                                                        CCC00210
C                                                                        CCC00220
      XLAT=38.0                                                          CCC00230
      IDAY=IDATE+IDAZE(IMO)                                              CCC00240
      LAT=IFIX(XLAT+0.5)-25                                              CCC00250
      DO 200 N=1,157                                                     CCC00260
      IDY=IDAY+N                                                         CCC00270
      DAY=FLOAT(IDY)                                                     CCC00280
C     EQUATIONS ARE VALID FOR LATITUDES BETWEEN 26 AND 46                CCC00290
      IF(LAT.GT.21)WRITE(6,901)XLAT                                      CCC00300
      IF(LAT.LT.1)WRITE(6,901)XLAT                                       CCC00310
      SR2(N)=SRC1(LAT)-SRC2(LAT)*SIN(2.0*3.14159*DAY/366.0+SRC3(LAT))    CCC00320
      SR2(N) = SR2(N)*24.0                                               CCC00330
  901 FORMAT('***CSDAY***FATAL ERROR-LATITUDE OUT OF ALLOWABLE           CCC00340
     * RANGE(26 TO 46 DEG)'/14X,'INPUT LATITUDE=',F5.1)                  CCC00350
      IF(IBUG.GT.0)WRITE(6,902)LAT                                       CCC00360
  902 FORMAT('***LAT***',I5)                                             CCC00370
      SR3(N)=SR2(N)*(1.0-0.65*CC(N)**2.)                                 CCC00380
  200 CONTINUE                                                           CCC00390
C                                                                        CCC00400
C  CRITERION FOR ACCEPTING SR DATA FROM N. ANNA                          CCC00410
C  OR RICHMOND  CLOUD COVER IS WHETHER  THE ERROR                        CCC00420
C  IS WITHIN 75% OF RICHMOND CC .                                        CCC00430
      DO 700 J=1,157                                                     CCC00440
      IF(SR1(J).LT.SR3(J))GO TO 555                                      CCC00450
      ERR=(SR1(J)-SR3(J))/SR3(J)                                         CCC00460
      IF(ABS(ERR).GT.0.75)GO TO 20                                       CCC00470
      SR4(J)=SR1(J)                                                      CCC00480
      CC3(J)=CC2(J)                                                      CCC00490
      IFLAG(J)=0                                                         CCC00500
      GO TO 700                                                          CCC00510
  555 ERR=(SR1(J)-SR3(J))/SR3(J)                                         CCC00520
      IF(ABS(ERR).GT.0.15)GO TO 20                                       CCC00530
      SR4(J)=SR1(J)                                                      CCC00540
      CC3(J)=CC2(J)                                                      CCC00550
```

```
      IFLAG(J)=0                                                        CCC00560
      GO TO 700                                                         CCC00570
   20 SR4(J)=SR3(J)                                                     CCC00580
      CC3(J)=CC(J)                                                      CCC00590
      IFLAG(J)=1                                                        CCC00600
  700 CONTINUE                                                          CCC00610
C   WRITE OUTPUT                                                        CCC00620
      WRITE(6,500)                                                      CCC00630
      WRITE(6,502)                                                      CCC00640
      DO 777 N=1,157                                                    CCC00650
      WRITE(6,501)N,SR1(N),SR2(N),SR3(N),SR4(N),CC(N),CC2(N),CC3(N),    CCC00660
     * IFLAG(N)                                                         CCC00670
  501 FORMAT(6X,I5,7X,F10.2,5X,F10.2,5X,F10.2,5X,F10.2,2X,F5.1,2X,F5.1, CCC00680
     * 2X,F5.1,2X,I2)                                                  CCC00690
  777 CONTINUE                                                          CCC00700
      WRITE(13,100)SR4                                                  CCC00710
      WRITE(15,100)CC3                                                  CCC00720
  500 FORMAT(2X,'CORRECTIONS TO SOLAR RADIATION AND CLOUD COVER')       CCC00730
  502 FORMAT(//10X,'N. ANNA VS. RICHMOND',///,9X,'DAY',                 CCC00740
     *8X,'SR(N.ANNA)',7X,'SR(CS)',8X,'SR(RICH)',8X,'CORSR',4X,          CCC00750
     *'RHCC',3X,'NACC',3X,'CORCC',//,100('=')//)                        CCC00760
      STOP                                                              CCC00770
      END                                                               CCC00780
      BLOCK DATA                                                        CCC00790
C                                                                       CCC00800
C     CURVE FIT COEFFICIENTS FOR DAILY AVERAGE CLEAR SKY SOLAR          CCC00810
C     RADIATION SUBROUTINE - CSDAY                                      CCC00820
C                                                                       CCC00830
      COMMON /SRCOEF/ SRC1(21), SRC2(21), SRC3(21)                      CCC00840
      DATA SRC1 /80.155, 79.371, 78.566, 77.604, 76.655, 76.041, 75.060,CCC00850
     +           74.046, 73.161, 72.248, 71.390, 70.394, 69.350, 68.362,CCC00860
     +           67.281, 66.240, 65.197, 64.113, 63.010, 61.911, 60.782/CCC00870
      DATA SRC2 /29.207, 30.236, 31.219, 32.145, 33.156, 34.133, 35.194,CCC00880
     +           35.938, 36.834, 37.699, 38.598, 39.413, 40.188, 40.982,CCC00890
     +           41.706, 42.442, 43.128, 43.788, 44.471, 45.020, 45.639/CCC00900
      DATA SRC3 / 1.679, 1.713, 1.710, 1.740, 1.728, 1.694, 1.737,      CCC00910
     +            1.734, 1.727, 1.738, 1.721, 1.730, 1.741, 1.739,      CCC00920
     +            1.742, 1.736, 1.740, 1.739, 1.739, 1.740, 1.735/      CCC00930
      END                                                               CCC00940
```

```
C     NRC ENDECOS DATA                                                  AVE00010
C     PROGRAM TO CONVERT HOURLY DATA TO DAILY AVERAGES                  AVE00020
                                                                        AVE00030
      DIMENSION A(38664),B(9,178),IDY(4296)                             AVE00040
      DO 150 I=1,4296                                                   AVE00050
      J=(9*I)-8                                                         AVE00060
      K=9*I                                                             AVE00070
      READ(15,100)IDY(I),(A(II),II=J,K)                                 AVE00080
  100 FORMAT(12X,I2,10X,7F6.1,54X,F6.1,6X,F6.1)                         AVE00090
  150 CONTINUE                                                          AVE00100
      DO 101 IJ=1,9                                                     AVE00110
      N3=0                                                              AVE00120
      N2=0                                                              AVE00130
      DO 200 J=1,178                                                    AVE00140
      X=0.                                                              AVE00150
      N=0                                                               AVE00160
      N4=0                                                              AVE00170
      DO 225 I=1,25                                                     AVE00180
      K=IJ+(9*(I-1))+((J-1)*24*9)-(N2*9)                                AVE00190
      N4=N4+1                                                           AVE00200
      IF(A(K).EQ.0.0) GO TO 224                                         AVE00210
      X=X+A(K)                                                          AVE00220
      N=N+1                                                             AVE00230
  224 KK=N4+((J-1)*24)-N2                                               AVE00240
      IF(IDY(KK).EQ.IDY(KK+1)) GO TO 225                                AVE00250
      GO TO 226                                                         AVE00260
  225 CONTINUE                                                          AVE00270
  226 IF(N.EQ.0)GO TO 227                                               AVE00280
      B(IJ,J)=((X/N)*1.8)+32.                                           AVE00290
      GO TO 228                                                         AVE00300
  227 B(IJ,J)=32.                                                       AVE00310
  228 N2=24-N4+N3                                                       AVE00320
      N3=N2                                                             AVE00330
      IF(B(IJ,J).EQ.32.) B(IJ,J)=0.0                                    AVE00340
  200 CONTINUE                                                          AVE00350
  101 CONTINUE                                                          AVE00360
      DO 299 I=1,23                                                     AVE00370
      J=8*I-7                                                           AVE00380
      K=8*I                                                             AVE00390
      WRITE(16,300)(B(1,II),II=J,K)                                     AVE00400
      WRITE(17,300)(B(2,II),II=J,K)                                     AVE00410
      WRITE(18,300)(B(3,II),II=J,K)                                     AVE00420
      WRITE(19,300)(B(4,II),II=J,K)                                     AVE00430
      WRITE(20,300)(B(5,II),II=J,K)                                     AVE00440
      WRITE(21,300)(B(6,II),II=J,K)                                     AVE00450
      WRITE(22,300)(B(7,II),II=J,K)                                     AVE00460
      WRITE(23,300)(B(8,II),II=J,K)                                     AVE00470
      WRITE(24,300)(B(9,II),II=J,K)                                     AVE00480
  300 FORMAT(8F10.2)                                                    AVE00490
  299 CONTINUE                                                          AVE00500
      STOP                                                              AVE00510
      END                                                               AVE00520
```

```
C     MIT ENDECOS DATA                                                  AVE00010
C     PROGRAM TO CONVERT HOURLY DATA TO DAILY DATA                      AVE00020
                                                                        AVE00030
      DIMENSION A(42000),B(7,180),IDY(5136)                             AVE00040
      DO 150 I=1,5136                                                   AVE00050
      J=(7*I)-6                                                         AVE00060
      K=7*I                                                             AVE00070
      READ(15,100)IDY(I),(A(II),II=J,K)                                 AVE00080
  100 FORMAT(12X,I2,4X,2F6.1,60X,5F6.1)                                 AVE00090
  150 CONTINUE                                                          AVE00100
      DO 101 IJ=1,7                                                     AVE00110
      N3=0                                                              AVE00120
      N2=0                                                              AVE00130
      DO 200 J=1,180                                                    AVE00140
      X=0.                                                              AVE00150
      N=0                                                               AVE00160
      N4=0                                                              AVE00170
      DO 225 I=1,48                                                     AVE00180
      K=IJ+(7*(I-1))+((J-1)*24*7)-(N2*7)                                AVE00190
      N4=N4+1                                                           AVE00200
      IF(A(K).EQ.0.0) GO TO 224                                         AVE00210
      X=X+A(K)                                                          AVE00220
      N=N+1                                                             AVE00230
  224 KK=N4+((J-1)*24)-N2                                               AVE00240
      IF(IDY(KK).EQ.IDY(KK+1)) GO TO 225                                AVE00250
      GO TO 226                                                         AVE00260
  225 CONTINUE                                                          AVE00270
  226 IF(N.EQ.0)GO TO 227                                               AVE00280
      B(IJ,J)=((X/N)*1.8)+32.                                           AVE00290
      GO TO 228                                                         AVE00300
  227 B(IJ,J)=32.                                                       AVE00310
  228 N2=24-N4+N3                                                       AVE00320
      N3=N2                                                             AVE00330
      IF(B(IJ,J).EQ.32.) B(IJ,J)=0.0                                    AVE00340
  200 CONTINUE                                                          AVE00350
  101 CONTINUE                                                          AVE00360
      DO 299 I=1,23                                                     AVE00370
      J=8*I-7                                                           AVE00380
      K=8*I                                                             AVE00390
      WRITE(16,300)(B(1,II),II=J,K)                                     AVE00400
      WRITE(17,300)(B(2,II),II=J,K)                                     AVE00410
      WRITE(18,300)(B(3,II),II=J,K)                                     AVE00420
      WRITE(19,300)(B(4,II),II=J,K)                                     AVE00430
      WRITE(20,300)(B(5,II),II=J,K)                                     AVE00440
      WRITE(21,300)(B(6,II),II=J,K)                                     AVE00450
      WRITE(22,300)(B(7,II),II=J,K)                                     AVE00460
  300 FORMAT(8F10.2)                                                    AVE00470
  299 CONTINUE                                                          AVE00480
      STOP                                                              AVE00490
      END                                                               AVE00500
```

```
GLOBAL TXTLIB FULLPOND FORTLIB VFORTLIB CMSLIB
FILEDEF (5 DISK IN1305 DATA A (PERM
FILEDEF 30 DISK MET1305 DATA C (PERM
FILEDEF 20 DISK   (PERM LRECL 80 BLOCK 800 RECFM FB
FILEDEF 6 ) &1  (PERM
FILEDEF 7 TERM (PERM
FILEDEF 31 DISK PLNT1305 DATA A (PERM
FILEDEF 32 DISK PLNT1305 DATA A (PERM
FILEDEF 35 DISK  (PERM
FILEDEF 36 DISK  (PERM
FILEDEF 37 DISK  (PERM
FILEDEF 38 DISK  (PERM
FILEDEF 39 DISK JUNK DATA  (PERM
FILEDEF 50 DISK RUN1305 DATA K (PERM
FILEDEF 51 DISK ML1305 DATA K(PERM
FILEDEF 60 DISK SAA1305 DATA K (PERM
LOAD MAIN(START
```


Attachment 2 (File: FULLPOND EXEC)

```
C     PROGRAM TO COMPUTE THE SPECTRAL DENSITY OF A GIVEN TIME SERIES   SPE00010
C     USING IMSL SUBROUTINE FTFREQ.                                    SPE00020
C     JULY, 1983                                                       SPE00030
C                                                                      SPE00040
      INTEGER IND(6), IER                                              SPE00050
      REAL X(4000),XIND(4),XYMV(6),ACV(3000),FREQ(3000),PS(3000),      SPE00060
     *XCOV(3000),XSPECT(3000),AMPHAS(3000),COHER(3000)                 SPE00070
      REAL A(1893)                                                     SPE00080
C                                                                      SPE00090
      IND(1)=0                                                         SPE00100
      IND(3)=0                                                         SPE00110
      IND(5)=0                                                         SPE00120
      IND(6)=0                                                         SPE00130
      XIND(1)=1.0                                                      SPE00140
C                                                                      SPE00150
C     IND(2) LENGTH OF TIME SERIES                                     SPE00160
C     IND(4) NUMBER OF TIME LAGS                                       SPE00170
C     NSTART NUMBER OF DAY WHEN TIME SERIES STARTS                     SPE00180
C                                                                      SPE00190
      READ(6,*)IND(2),IND(4),NSTART                                    SPE00200
1000  FORMAT (3I5)                                                     SPE00210
      WRITE(6,1000)IND(2),IND(4),NSTART                                SPE00220
      NPOINT=IND(2)                                                    SPE00230
      NLAG=IND(4)                                                      SPE00240
      NLAGP1=NLAG+1                                                    SPE00250
C                                                                      SPE00260
C     READ IN BASE SERIES                                              SPE00270
C                                                                      SPE00280
      READ (50,1050) A                                                 SPE00290
      WRITE (7,1050) (A(I),I=1,8)                                      SPE00300
1050  FORMAT (8F10.2)                                                  SPE00310
      DO 10 J=1,NPOINT                                                 SPE00320
      K=J+NSTART-1                                                     SPE00330
10    X(J)=A(K)                                                        SPE00340
      WRITE (7,1050) (X(J),J=1,8)                                      SPE00350
C                                                                      SPE00360
      CALL FTFREQ (X,IND,XIND,XYMV,ACV,FREQ,PS,XCOV,XSPECT,AMPHAS,     SPE00370
     *XFER,COHER,IER)                                                  SPE00380
C                                                                      SPE00390
C     OUTPUT TO FILES AND TERMINAL                                     SPE00400
C                                                                      SPE00410
      WRITE (7,1000) IER                                               SPE00420
      WRITE (7,1060) (XYMV(I),I=1,2)                                   SPE00430
1060  FORMAT (1H ,8F12.5)                                              SPE00440
      WRITE (30) (FREQ(I),I=1,NLAGP1)                                  SPE00450
      WRITE (31) (PS(I),I=1,NLAGP1)                                    SPE00460
      STOP                                                             SPE00470
      END                                                              SPE00480
```

```
 1006 FORMAT(T10,'REACH',T21,'ENTRANCE CHANNEL',T46,'REACH',T62,          FWH01110
     +       'FLOW PATH AREAS  (ACRES)',T91,'TRAVEL',T101,'NUMBER'/      FWH01120
     +   T10,'NUMBER',T21,16('-'),                                       FWH01130
     +       T41,16('-'),T61,26('-'),T92,'TIME',T103,'OF'/               FWH01140
     +   20X,2('DEPTH',5X,'WIDTH',5X),'BEFORE',T71,'BETWEEN',           FWH01150
     +       T82,'AFTER',T91,'(TIME',T102,'SIDE'/                        FWH01160
     +   20X,4('(FEET)',4X),'ARM 1',T71,'1 AND 2',T82,'ARM 2',          FWH01170
     +       T91,'STEPS)',T102,'ARMS'/)                                  FWH01180
 1007 FORMAT(T10,I4,5X,4(F8.2,2X),2X,3(F7.2,3X),I4,5X,I5)                FWH01190
 1008 FORMAT(//)                                                         FWH01200
 1009 FORMAT(T10,'REACH',T35,'SIDE ARM 1',T75,'SIDE ARM 2'/              FWH01210
     +   T10,'NUMBER',T22,36('-'),T62,36('-')/                           FWH01220
     +   20X,2(12X,'FLOW',4X,'ENTRANCE',1X,'ENTRANCE',3X)/               FWH01230
     +   20X,2(2X,'AREA',5X,'WIDTH',5X,'DEPTH',4X,'LENGTH',4X)/          FWH01240
     +   20X,2(1X,'(ACRES)',3X,'(FEET)',4X,'(FEET)',3X,'(FEET)',4X)/)    FWH01250
 1010 FORMAT(T10,I4,6X,8(F8.2,2X))                                       FWH01260
 1014 FORMAT( )                                                          FWH01270
 1016 FORMAT(/T10,'WHTF REACH LAGS ARE SET AT TRAVEL TIMES SHOWN')       FWH01280
 1017 FORMAT(/T10,'WHTF REACH LAGS ARE SET AT 0 OR 1 TIME STEPS')        FWH01290
 1018 FORMAT(/T10,'SIDEARM FLOWS ARE REDUCED BY A FACTOR OF ',F4.2)      FWH01300
 1019 FORMAT(/T10,'THIS RUN CONSISTS OF',I2,2X,'REACHES IN THE WHTF,IE'/FWH01310
     *T10,'REROUTING OF THE FLOW THROUGH THE 2 MAIN SIDEARMS IN THE WHTFFWH01320
     * FOR MAXIMUM HEAT TRANSFER')                                     FWH01330
      END                                                              FWH01340
      SUBROUTINE DSPARM(TITLE)                                         FWH01350
C                                                                      FWH01360
C     SUBROUTINE TO REPRINT RUN PARAMETER INPUT DATA                   FWH01370
C                                                                      FWH01380
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                              FWH01390
      COMMON /IOPUT/ IREAD,IRITE                                       FWH01400
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                         FWH01410
      COMMON /METEOE/ NMET,DTMET,IOMET,NDIS,DTDIS,IODIS                FWH01420
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR     FWH01430
      COMMON /PARAM2/ RTIME,RDATE,NSTEP                                FWH01440
      DOUBLE PRECISION RTIME,RDATE                                     FWH01450
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),   FWH01460
     +       QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH              FWH01470
      DIMENSION TITLE(20)                                              FWH01480
C                                                                      FWH01490
      WRITE(IRITE,1001)                                                FWH01500
      WRITE(IRITE,1002)TITLE                                           FWH01510
      WRITE(IRITE,1003)RDATE,RTIME,DT,TSTOP,NSTEP,TZERO,NMET,DTMET,    FWH01520
     +       IOMET,NDIS,DTDIS,IODIS                                    FWH01530
      IF(ISMET.GT.1)WRITE(IRITE,1015)ISMET                             FWH01540
      IF(EFACT.LT.1.0)WRITE(IRITE,1021)EFACT                           FWH01550
      RETURN                                                           FWH01560
C                                                                      FWH01570
C     FORMAT STATEMENTS                                                FWH01580
C                                                                      FWH01590
 1001 FORMAT('1')                                                      FWH01600
 1002 FORMAT(' ',20A4/' ',80('=')//)                                   FWH01610
 1003 FORMAT(5X,'RUN PARAMETERS'/5X,14('-')/                           FWH01620
     +       10X,'RUN OF ',A8,' AT ',A8//                              FWH01630
     +       10X,'TIME STEP',T45,F6.2,2X,'(DAYS)'/                     FWH01640
     +       10X,'RUN STOPS AT TIME',T45,F7.2,2X,'(DAYS)'/             FWH01650
```

```
     +        10X,'NUMBER OF STEPS',T45,I6/                                  FWH01660
     +        10X,'INITIAL TEMPERATURE',T45,F6.2,2X,'(DEGREES F)'//          FWH01670
     +        10X,'NUMBER OF MET DATA POINTS',T45,I6/                        FWH01680
     +        10X,'MET DATA TIME STEP',T45,F6.2,2X,'(DAYS)'/                 FWH01690
     +        10X,'INPUT UNIT FOR MET DATA',T45,I6//                         FWH01700
     +        10X,'NUMBER OF PLANT DATA POINTS',T45,I6/                      FWH01710
     +        10X,'PLANT DATA TIME STEP',T45,F6.2,2X,'(DAYS)'/               FWH01720
     +        10X,'INPUT UNIT FOR PLANT DATA',T45,I6/)                       FWH01730
 1015 FORMAT(10X,'MET DATA SMOOTHING INTERVAL',T45,I6,2X,                    FWH01740
     +   '(TIME STEPS)'/)                                                    FWH01750
 1021 FORMAT(10X,'WIND FUNCTION ADJUSTMENT CONSTANT',T45,F6.2/)              FWH01760
      END                                                                    FWH01770
      SUBROUTINE PRWHTF(I,MMO,MDAY,MYR)                                      FWH01780
C                                                                            FWH01790
C     PRINTS RESULTS OF WHTF SIMULATION                                      FWH01800
C                                                                            FWH01810
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                                    FWH01820
      COMMON /IOPUT/ IREAD,IRITE,IRR                                         FWH01830
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON       FWH01840
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH          FWH01850
     *,MIXD(5)                                                               FWH01860
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR           FWH01870
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2)                                FWH01880
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR               FWH01890
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR       FWH01900
      COMMON /METEDA/ WIND(366),TA(366),SIGH(366),FIN(366),CLOUD(366)        FWH01910
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                       FWH01920
      REAL LWRI,LWRO                                                         FWH01930
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                               FWH01940
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB                FWH01950
      COMMON /ENERGY/ EQ,E1,E2,E3,ET,FLUXT                                   FWH01960
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                      FWH01970
      COMMON /WHTF1/ ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO,        FWH01980
     +          NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4),            FWH01990
     +          HSA2(4),ZBR(4),XLSA1(4),XLSA2(4)                             FWH02000
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893),        FWH02010
     . QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)         FWH02020
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),         FWH02030
     +          QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH                 FWH02040
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                         FWH02050
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                                FWH02060
      COMMON /PLANT/ QPLNT(1893),TPLNT(1893)                                 FWH02070
      COMMON /INA/ DTTI(5),TI(1893,2),TIN(1893),NTI(5)                       FWH02080
C                                                                            FWH02090
C                                                                            FWH02100
C        CONVERT UNITS FOR OUTPUT                                            FWH02110
C                                                                            FWH02120
         QPLNT(I)=QPLNT(I)/86400.0                                           FWH02130
C                                                                            FWH02140
C        PRINT RESULTS FOR THIS TIME STEP                                    FWH02150
C                                                                            FWH02160
      IF (ISELCT(8).EQ.0) TI(I,1)=TPLNT(I)                                   FWH02170
      WRITE(IRITE,1011)                                                      FWH02180
         WRITE(IRITE,1012)MMO,MDAY,MYR,TEQ,QPLNT(I),TI(I,1),                 FWH02190
     +      TTEXIT(I,1),TSEND(1),DISDIL(1),ULAY1(1)                          FWH02200
```

```
            DO 410 J=2,NREACH                                              FWH02210
            J1=J-1                                                         FWH02220
            WRITE(IRITE,1013)J,QSA1(J),QSA2(J),TFIN(I,J1),TSA1(J),         FWH02230
     +                       TRS1(J),TSA2(J),TRS2(J),TSEND(J),DISDIL(J),FWH02240
     +                       ULAY1(J)                                      FWH02250
 410         CONTINUE                                                      FWH02260
          IFILE =ISELCT(11)                                                FWH02270
          IF(IFILE.GT.0)WRITE(IFILE,2000)RDATE,MMO,MDAY,MYR,               FWH02280
     +       TI(I,1),TTEXIT(I,1),                                          FWH02290
     +       TSEND(1),TFIN(I,1),TSA1(2),TRS1(2),TSA2(2),TRS2(2),           FWH02300
     +       TSEND(2),TFIN(I,2),TSA1(3),TRS1(3),TSA2(3),TRS2(3),           FWH02310
     +       TSEND(3)                                                      FWH02320
          RETURN                                                           FWH02330
 1011 FORMAT(//38X,'DISCHARGE CHANNEL AND WHTF'/38X,26('*')/               FWH02340
     +    T36,'DISCHARGES (CFS)',                                         FWH02350
     +       T67,'TEMPERATURES (DEGREES FAHRENHEIT)',T121,'HEATED'/       FWH02360
     +    T14,'EQUIL.',T30,28('-'),T60,47('-'),T121,'LAYER'/              FWH02370
     +    T15,'TEMP.',T32,'PLANT',T43,'SIDE',T53,'SIDE',                  FWH02380
     +       T60,'PLANT',2X,'REACH',3X,'SIDE ARM 1',4X,                   FWH02390
     +       'SIDE ARM 2',T102,'REACH',T110,'ENTRANCE',T121,'DEPTH'/      FWH02400
     +    T7,'DATE',T14,'(DEG F)',T30,'DISCHARGE',T42,'ARM  1',T52,       FWH02410
     +       'ARM  2',T60,'DISCH',2X,'ENTR.',2X,'ENTR.',3X,'END ',2X,     FWH02420
     +       'ENTR.',3X,'END ',T103,'END',T110,'DILUTION',                FWH02430
     +       T121,'(FEET)'/)                                              FWH02440
 1012 FORMAT(T5,I2,'/',I2,'/',I2,T15,F5.2,T23,'REACH 1',T30,F8.1,T60,     FWH02450
     +    2(F6.2,2X),T100,3(F7.2,3X))                                     FWH02460
 1013 FORMAT(T23,'REACH',I2,T40,2(F8.1,2X),T66,6(F6.2,1X),T110,           FWH02470
     +       2(F7.2,3X))                                                  FWH02480
 2000 FORMAT(A8,3I2,15F5.1)                                               FWH02490
      END                                                                 FWH02500
      SUBROUTINE RDWHTF                                                   FWH02510
C                                                                         FWH02520
C     SUBROUTINE READS INPUT DATA FOR DISCHARGE CANAL AND WHTF            FWH02530
C                                                                         FWH02540
      COMMON /BUGS/ IDEBUG(16),ISELCT(20)                                 FWH02550
      COMMON /IOPUT/ IREAD,IRITE,IRR                                      FWH02560
      COMMON /CONST/RHO,HCAP,GRAV,VISCOS,ETA,BETA,FRIFAC,EVPCON,DELCON    FWH02570
      COMMON /MIXG/ QQMIX(102),QMIX,TMIX,RMIX,DMIX,MIXED,JMIXB,MIXH       FWH02580
     *,MIXD(5)                                                            FWH02590
      COMMON /PARAM/ DT,DTT,TSTOP,TZERO,YSUR,YBOT,SLOPE,NPRINT,NPR        FWH02600
      COMMON /TQ/ TEXIT(1893,2),QEXIT(1893,2)                             FWH02610
      COMMON /TEMP/ THL(100,2),TSTAR(20,2),THSTAR(50,2),TEXSTR            FWH02620
      COMMON /HYDRO/ RHOB,HCAPB,DAHL,DD(2),DIL(20),B(102),S(102),WL,AR     FWH02630
      COMMON /METEOA/ WIND(366),TA(366),SIGH(366),FIN(366),CLOUD(366)     FWH02640
      COMMON /METEOC/ EVAP,RAD,TAIR,PSI,CONDUC,SR,LWRI                    FWH02650
      REAL LWRI,LWRO                                                      FWH02660
      COMMON /METEOD/ TEQ,PRAIR,CC,WINDY,EFACT                            FWH02670
      COMMON /METEOE/ NMET,DTMET,IOMET,NDIS,DTDIS,IODIS,ISMET,ISDIS       FWH02680
      COMMON /DISCH/ XCH,WCH,DCH,ACH,DCHX,WCHX,QCH,F2,NCH,NCB             FWH02690
      COMMON /ENERGY/ EO,E1,E2,E3,ET,FLUXT                                FWH02700
      COMMON /MAINPB/ NARSF,PLUMF,DY,JM                                   FWH02710
      COMMON /WHTF1/ ZB(4),ZH(4),ZHM(4),ZA1(4),ZA2(4),ZA3(4),BOO,HOO,     FWH02720
     +               NSA(4),ZASA1(4),ZASA2(4),BSA1(4),BSA2(4),HSA1(4),    FWH02730
     +               HSA2(4),ZBR(4),XLSA1(4),XLSA2(4)                     FWH02740
      COMMON /WHTF2/ TEND(4),TFIN(1893,4),TSA1(4),TSA2(4),QFIN(1893),     FWH02750
```

```
     . QSA1(4),QSA2(4),FROUDE(4),ULAY1(4),ULAY2(4),ULAY3(4),ULAY4(4)      FWH02760
      COMMON /WHTF3/ LAGFIX(4),ASUM(4),RTW,RHO,HBOT,TRS1(4),TRS2(4),      FWH02770
     +               QLIM1(4),QLIM2(4),SAFACT,TSEND(4),WGT,NREACH         FWH02780
      COMMON /WLOSS1/ WLT,WLTN,WLTWH                                      FWH02790
      COMMON/STUPID/ TTEXIT(1893,1),DISDIL(4)                             FWH02800
      COMMON /PLANT/ QI(1893),TIN(1893)                                   FWH02810
C                                                                         FWH02820
      READ(IREAD,11)SAFACT,WGT,EFACT                                      FWH02830
      IF(EFACT.LT.0.0001)EFACT=1.0                                        FWH02840
      NSTEP=TSTOP/DT                                                      FWH02850
      READ(IREAD,10)NMET,NDIS                                             FWH02860
      READ(IREAD,11)DTMET,DTDIS                                           FWH02870
      READ(IREAD,10)IOMET,IODIS,ISMET,ISDIS                               FWH02880
      READ(IREAD,10)NCH,NCB                                               FWH02890
      READ(IREAD,11)XCH,WCH,DCH,ACH,DCHX,WCHX,PLUMF,FRIFAC                FWH02900
         DO 50 I=1,NREACH                                                 FWH02910
         READ(IREAD,11)ZH(I),ZB(I),ZBR(I),ZA1(I),ZA2(I),ZA3(I),ZHM(I)     FWH02920
         READ(IREAD,10)NSA(I),LAGFIX(I)                                   FWH02930
 50      READ(IREAD,11)ZASA1(I),ZASA2(I),BSA1(I),                         FWH02940
     +        BSA2(I),HSA1(I),HSA2(I),XLSA1(I),XLSA2(I)                   FWH02950
      RETURN                                                              FWH02960
   10 FORMAT(16I5)                                                        FWH02970
   11 FORMAT(8F10.0)                                                      FWH02980
   12 FORMAT(20A4)                                                        FWH02990
   15 FORMAT(2I5,7F10.0)                                                  FWH03000
      END                                                                 FWH03010
```

```
C      TO COMBINE TWO DATA FILES INTO ONE                          COM00010
       DIMENSION Q1(572), Q2(178), Q3(750)                         COM00020
       READ(10,100) Q1                                             COM00030
       READ(11,100) Q2                                             COM00040
  100  FORMAT(8F10.2)                                              COM00050
       DO 10 J=1,572                                               COM00060
       Q3(J)=Q1(J)                                                 COM00070
    10 CONTINUE                                                    COM00080
       DO 20 J=1,178                                               COM00090
       K=572+J                                                     COM00100
       Q3(K)=Q2(J)                                                 COM00110
    20 CONTINUE                                                    COM00120
       WRITE(12,100) Q3                                            COM00130
       STOP                                                        COM00140
       END                                                         COM00150
```

```
C     THIS PROGRAM CALCULATES THE MONTHLY AVERAGED PREDICTED TEMPERATUREMON00010
C     AT 7 DIFFERENT STATIONS OF THE COOLING POND SYSTEM AND ALSO       MON00020
C     THE AVERAGED FILTERED EQUILIBRIUM TEMPERATURE; MONTHLY VALUES OF  MON00030
C     STANDARD DEVIATION ARE ALSO PRODUCED.                             MON00040
C     IT REQUIRES THREE DATA FILES: DAY, MLXXXX AND RUNXXXX.            MON00050
C     THE LAST TWO ARE OBTAINED BY RUNNING FULLPOND.                    MON00060
C     FIRST EDITION ON 25 JANUARY 1983 ;REVISED ON 5 JANUARY 1984.      MON00070
                                                                        MON00080
      DIMENSION D1(2000),D2(2000),D3(2000),D4(2000),D5(2000),D6(2000),  MON00090
     *          D7(2000),M7(2000),TAV1(70),TAV2(70),TAV3(70),           MON00100
     *          TAV4(70),TAV5(70),TAV6(70),TAV7(70)                     MON00110
                                                                        MON00120
C     D1=TRSA,D2=INTAKE,D3=THIS,D4=BRPT,D5=DIKE3,D6=TEQ,D7=TEXIT        MON00130
C     M7=MONTH,D9=208                                                   MON00140
                                                                        MON00150
      READ(10,1000) NDAY, N1, N2                                        MON00160
 1000 FORMAT (3I5)                                                      MON00170
                                                                        MON00180
C     NDAY=TOTAL NO. OF DAYS;                                           MON00190
C     N1=FIRST DAY OF FIRST INTACT MONTH OF DATA;                       MON00200
C     N2=LAST DAY OF LAST INTACT MONTH OF DATA.                         MON00210
                                                                        MON00220
      WRITE (12,1005)                                                   MON00230
 1005 FORMAT (' MONTH',3X,'TEXIT',3X,'DYKE3',4X,'BRPT',4X,'THIS',5X,    MON00240
     *'INT',5X,'208',4X,'TRSA',4X,'TEQ',/)                              MON00250
      WRITE (15,1105)                                                   MON00260
 1105 FORMAT (' MONTH',3X,'TEXIT',3X,'DYKE3',4X,'BRPT',4X,'THIS',5X,    MON00270
     *'INT',4X,'TRSA',4X,'TEQ',/)                                       MON00280
                                                                        MON00290
      DO 1 I=1,NDAY                                                     MON00300
      READ (11,1002) D3(I),D1(I)                                        MON00310
      READ (13,1004) M7(I),D7(I),D5(I),D4(I),D2(I),D6(I)                MON00320
      D1(I)=(D1(I)-32.)*5./9.                                           MON00330
      D2(I)=(D2(I)-32.)*5./9.                                           MON00340
      D3(I)=(D3(I)-32.)*5./9.                                           MON00350
      D4(I)=(D4(I)-32.)*5./9.                                           MON00360
      D5(I)=(D5(I)-32.)*5./9.                                           MON00370
      D6(I)=(D6(I)-32.)*5./9.                                           MON00380
    1 D7(I)=(D7(I)-32.)*5./9.                                           MON00390
 1002 FORMAT (36X,F5.2,15X,F5.2)                                        MON00400
 1004 FORMAT (I2,10X,F6.2,6X,F6.2,6X,2F6.2,18X,F6.2)                    MON00410
                                                                        MON00420
C     SET INITIAL VALUES                                                MON00430
                                                                        MON00440
      K=0                                                               MON00450
      MON=M7(N1)                                                        MON00460
      TD1=0                                                             MON00470
      TD2=0                                                             MON00480
      TD3=0                                                             MON00490
      TD4=0                                                             MON00500
      TD5=0                                                             MON00510
      TD6=0                                                             MON00520
      TD7=0                                                             MON00530
      VA1=0                                                             MON00540
      VA2=0                                                             MON00550
```

```
      VA3=0                                                          MON00560
      VA4=0                                                          MON00570
      VA5=0                                                          MON00580
      VA6=0                                                          MON00590
      VA7=0                                                          MON00600
      N3=N2+1                                                        MON00610
      L=0                                                            MON00620
                                                                     MON00630
C     CALCULATE MEAN                                                 MON00640
                                                                     MON00650
      DO 2 I=N1,N3                                                   MON00660
      IF (M7(I).NE.MON) GO TO 3                                      MON00670
    4 TD1=TD1+D1(I)                                                  MON00680
      TD2=TD2+D2(I)                                                  MON00690
      TD3=TD3+D3(I)                                                  MON00700
      TD4=TD4+D4(I)                                                  MON00710
      TD5=TD5+D5(I)                                                  MON00720
      TD6=TD6+D6(I)                                                  MON00730
      TD7=TD7+D7(I)                                                  MON00740
      K=K+1                                                          MON00750
    2 CONTINUE                                                       MON00760
      GO TO 5                                                        MON00770
    3 TAVER1=TD1/K                                                   MON00780
      TAVER2=TD2/K                                                   MON00790
      TAVER3=TD3/K                                                   MON00800
      TAVER4=TD4/K                                                   MON00810
      TAVER5=TD5/K                                                   MON00820
      TAVER6=TD6/K                                                   MON00830
      TAVER7=TD7/K                                                   MON00840
      TAVER9=(2.0*TAVER2+TAVER1)/3.0                                 MON00850
      L=L+1                                                          MON00860
                                                                     MON00870
C     TAVER9=TEMP. AT 208                                            MON00880
                                                                     MON00890
      WRITE (12,1006) MON,TAVER7,TAVER5,TAVER4,TAVER3,TAVER2,TAVER9, MON00900
     *TAVER1,TAVER6                                                  MON00910
 1006 FORMAT(I6,8F8.2)                                               MON00920
      TAV1(L)=TAVER1                                                 MON00930
      TAV2(L)=TAVER2                                                 MON00940
      TAV3(L)=TAVER3                                                 MON00950
      TAV4(L)=TAVER4                                                 MON00960
      TAV5(L)=TAVER5                                                 MON00970
      TAV6(L)=TAVER6                                                 MON00980
      TAV7(L)=TAVER7                                                 MON00990
      MON=M7(I)                                                      MON01000
      TD1=0                                                          MON01010
      TD2=0                                                          MON01020
      TD3=0                                                          MON01030
      TD4=0                                                          MON01040
      TD5=0                                                          MON01050
      TD6=0                                                          MON01060
      TD7=0                                                          MON01070
      K=0                                                            MON01080
      GO TO 4                                                        MON01090
                                                                     MON01100
```

```
C     THIS PROGRAM COMPUTES DOWNWARD ATMOSPHERIC (LONGWAVE) RADIATION  LWR00010
C     UNDER CLEAR SKIES BY USE OF EIGHT DIFFERENT PROPOSED EMPIRICAL   LWR00020
C     FORMULAE AND THE RESULTS ARE PLOTTED FOR COMPARISION.            LWR00030
C     OPTIONS OF PLOTS: DAILY-AVERAGED OR MONTHLY-AVERAGED.            LWR00040
C     STATISTICS OF MONTHLY-AVERAGED DATA CAN BE PRINTED OUT IF DESIRED.LWR00050
C     TWO INPUT DATA FILES ARE USED: AIR TEMP. AND REL. HUM.           LWR00060
C     WRITTEN BY EDMOND KAMAN HO ON 8 SEPTEMBER 1983.                  LWR00070
                                                                       LWR00080
                                                                       LWR00090
      INTEGER MON(1893)                                                LWR00100
      REAL XMONTH(100),DAYS(1893)                                      LWR00110
      REAL RADM1(100),RADM2(100),RADM3(100),RADM4(100),RADM5(100),     LWR00120
     &     RADM6(100),RADM7(100),RADM8(100)                            LWR00130
      REAL TA(1893),RH(1893),TAIR(100),RHUM(100)                       LWR00140
      DOUBLE PRECISION SVP(1893),VP(1893),SBT4(1893)                   LWR00150
      DOUBLE PRECISION EM1(1893),EM2(1893),EM3(1893),EM4(1893),        LWR00160
     &                 EM5(1893),EM6(1893),EM7(1893),EM8(1893)         LWR00170
      REAL RAD1(1893),RAD2(1893),RAD3(1893),RAD4(1893),RAD5(1893),     LWR00180
     &     RAD6(1893),RAD7(1893),RAD8(1893)                            LWR00190
      READ (30,1000) TA                                                LWR00200
      READ (31,1000) RH                                                LWR00210
1000  FORMAT (8F10.2)                                                  LWR00220
C                                                                      LWR00230
C     ACQUIRE INFORMATION FROM USER                                    LWR00240
C                                                                      LWR00250
      WRITE (8,1010)                                                   LWR00260
1010  FORMAT ('DO YOU WANT DAILY-AVERAGED (0) OR MONTHLY-AVERAGED (1)',LWR00270
     &        ' PLOTS?')                                               LWR00280
      READ (8,*) IND1                                                  LWR00290
      IF (FLOAT(IND1)) 40,40,50                                        LWR00300
40    WRITE (8,1020)                                                   LWR00310
1020  FORMAT ('INDICATE FIRST DAY, LAST DAY, AND TOTAL NUMBER OF ',/,  LWR00320
     &        'DAYS OF DATA TO BE USED:')                              LWR00330
      READ (8,*) NSTART, NEND, NPTS                                    LWR00340
      GOTO 60                                                          LWR00350
50    WRITE (8,1030)                                                   LWR00360
1030  FORMAT ('INDICATE FIRST DAY OF FIRST INTACT MONTH OF DATA(=7),',/,LWR00370
     &        'LAST DAY OF LAST INTACT MONTH OF DATA, AND TOTAL ',/,    LWR00380
     &        'NUMBER OF MONTHS OF DATA TO BE USED:')                  LWR00390
      READ (8,*) NSTART, NEND, NPTS                                    LWR00400
      READ (33,1040) (MON(I),I=1,NEND)                                 LWR00410
1040  FORMAT (I2)                                                      LWR00420
      WRITE (8,1050)                                                   LWR00430
1050  FORMAT ('DO YOU WANT A PRINT-OUT OF THE STATISTICS ONLY(1),',     LWR00440
     &        ' OR PLOTS ONLY(0), OR BOTH(2)?')                        LWR00450
      READ (8,*) IND2                                                  LWR00460
C                                                                      LWR00470
C     ************************************************************** LWR00480
C     *                    MAIN LOOP OF PROGRAM                    * LWR00490
C     ************************************************************** LWR00500
60    DO 10 I=NSTART,NEND                                              LWR00510
C     CONVERT FAHRENHEIT TO CELCIUS                                    LWR00520
        TA(I)=(TA(I)-32.0)*5.0/9.0                                     LWR00530
C     CALCULATION OF SAT. VAPOUR PRESSURE FROM AIR TEMP.               LWR00540
C     CF. P.7, LECTURE 3 OF 1.71 CLASS NOTES (PROF. RAFAEL BRAS)       LWR00550
```

```
C     TEMP. IN CELCIUS, VAPOUR PRESSURE IN MILLIBAR                    LWR00560
      SVP(I)=33.8639*((0.00738*TA(I)+0.8072)**8                        LWR00570
     &          -0.000019*(1.8*TA(I)+48.0)+0.001316)                   LWR00580
C     CALCULATE VAPOUR PRESSURE FROM S.V.P. AND REL. HUM.              LWR00590
      VP(I)=RH(I)*SVP(I)                                               LWR00600
C                                                                      LWR00610
C     CALCULATE ATMOSPHERIC EMISSIVITY UNDER CLEAR SKIES               LWR00620
C                 BRUNT(1932) ..... EM1                                LWR00630
C                 SWINBANK(1963) ..... EM2                             LWR00640
C                 IDSO-JACKSON(1969) .. EM3                            LWR00650
C                 BRUTSAERT 1(1975) ... EM4                            LWR00660
C                 SATTERLUND(1979) .... EM5                            LWR00670
C                 BRUTSAERT 2(1981) ... EM6                            LWR00680
C                 IDSO 1(1981) ........ EM7                            LWR00690
C                 IDSO 2(1981) ........ EM8                            LWR00700
      EM1(I)=0.51+0.066*SQRT(VP(I))                                    LWR00710
      EM2(I)=0.92E-5*(TA(I)+273.0)**2                                  LWR00720
      EM3(I)=1-0.261*EXP(-7.77E-4*(TA(I))**2)                          LWR00730
      EM4(I)=0.553*VP(I)**(1.0/7.0)                                    LWR00740
      EM5(I)=1.08*(1-EXP(-VP(I)**((TA(I)+273.)/2016.)))                LWR00750
      EM6(I)=0.575*VP(I)**(1.0/7.0)                                    LWR00760
      EM7(I)=0.179*VP(I)**(1.0/7.0)*EXP(350.0/(TA(I)+273.0))           LWR00770
      EM8(I)=0.7+5.95E-5*VP(I)*EXP(1500./(TA(I)+273.))                 LWR00780
C     CALCULATE CLEAR SKY DOWNWARD LONGWAVE RADIATION                  LWR00790
C     SBT4 IS THE PRODUCT BETWEEN STEFAN-BOLTZMAN CONSTANT AND AIR     LWR00800
C     RAISED TO THE FOURTH POWER (SI UNIT: WATT PER METER SQUARED)     LWR00810
                                                                       LWR00820
                                                                       LWR00830
      SBT4(I)=5.6697E-8*(TA(I)+273.)**4                                LWR00840
      RAD1(I)=EM1(I)*SBT4(I)                                           LWR00850
      RAD2(I)=EM2(I)*SBT4(I)                                           LWR00860
      RAD3(I)=EM3(I)*SBT4(I)                                           LWR00870
      RAD4(I)=EM4(I)*SBT4(I)                                           LWR00880
      RAD5(I)=EM5(I)*SBT4(I)                                           LWR00890
      RAD6(I)=EM6(I)*SBT4(I)                                           LWR00900
      RAD7(I)=EM7(I)*SBT4(I)                                           LWR00910
      RAD8(I)=EM8(I)*SBT4(I)                                           LWR00920
10    CONTINUE                                                         LWR00930
C     ***************************************************************  LWR00940
C     *                    END OF MAIN LOOP                         *  LWR00950
C     ***************************************************************  LWR00960
C                                                                      LWR00970
C           CALCULATE MONTHLY-AVERAGED LONGWAVE                        LWR00980
C           RADIATION UNDER CLEAR SKIES IF IND1=1                      LWR00990
C                                                                      LWR01000
      IF (IND1.EQ.0) GOTO 110                                          LWR01010
C     SET INITIAL VALUES                                               LWR01020
      TRAD1=0.0                                                        LWR01030
      TRAD2=0.0                                                        LWR01040
      TRAD3=0.0                                                        LWR01050
      TRAD4=0.0                                                        LWR01060
      TRAD5=0.0                                                        LWR01070
      TRAD6=0.0                                                        LWR01080
      TRAD7=0.0                                                        LWR01090
      TRAD8=0.0                                                        LWR01100
```

```
          TTA=0.0                                                    LWR01110
          RRH=0.0                                                    LWR01120
          JMONTH=0                                                   LWR01130
          K=0                                                        LWR01140
          NENDP1=NEND+1                                              LWR01150
          MONTH=MON(NSTART)                                          LWR01160
                                                                     LWR01170
      DO 70 I=NSTART,NENDP1                                          LWR01180
      IF (MON(I).NE.MONTH) GOTO 80                                   LWR01190
90        TRAD1=TRAD1+RAD1(I)                                        LWR01200
          TRAD2=TRAD2+RAD2(I)                                        LWR01210
          TRAD3=TRAD3+RAD3(I)                                        LWR01220
          TRAD4=TRAD4+RAD4(I)                                        LWR01230
          TRAD5=TRAD5+RAD5(I)                                        LWR01240
          TRAD6=TRAD6+RAD6(I)                                        LWR01250
          TRAD7=TRAD7+RAD7(I)                                        LWR01260
          TRAD8=TRAD8+RAD8(I)                                        LWR01270
          TTA=TTA+TA(I)                                              LWR01280
          RRH=RRH+RH(I)                                              LWR01290
          K=K+1                                                      LWR01300
70    CONTINUE                                                       LWR01310
      GOTO 100                                                       LWR01320
                                                                     LWR01330
80    JMONTH=JMONTH+1                                                LWR01340
      RADM1(JMONTH)=TRAD1/K                                          LWR01350
      RADM2(JMONTH)=TRAD2/K                                          LWR01360
      RADM3(JMONTH)=TRAD3/K                                          LWR01370
      RADM4(JMONTH)=TRAD4/K                                          LWR01380
      RADM5(JMONTH)=TRAD5/K                                          LWR01390
      RADM6(JMONTH)=TRAD6/K                                          LWR01400
      RADM7(JMONTH)=TRAD7/K                                          LWR01410
      RADM8(JMONTH)=TRAD8/K                                          LWR01420
      TAIR(JMONTH)=TTA/K                                             LWR01430
      RHUM(JMONTH)=RRH/K                                             LWR01440
      MONTH=MON(I)                                                   LWR01450
      TRAD1=0.0                                                      LWR01460
      TRAD2=0.0                                                      LWR01470
      TRAD3=0.0                                                      LWR01480
      TRAD4=0.0                                                      LWR01490
      TRAD5=0.0                                                      LWR01500
      TRAD6=0.0                                                      LWR01510
      TRAD7=0.0                                                      LWR01520
      TRAD8=0.0                                                      LWR01530
      TTA=0.0                                                        LWR01540
      RRH=0.0                                                        LWR01550
      K=0                                                            LWR01560
      GOTO 90                                                        LWR01570
                                                                     LWR01580
C     ********************************************************************** LWR01590
C     *              PRINT OUT STATISTICS OF MONTHLY AVERAGES          * LWR01600
C     ********************************************************************** LWR01610
                                                                     LWR01620
100   IF (IND2.EQ.0) GOTO 105                                        LWR01630
      WRITE (35,1060)                                                LWR01640
1060  FORMAT ('MONTHLY-AVERAGED CALCULATIONS OF DOWNWARD LONGWAVE',  LWR01650
```

```
     &        ' RADIATION UNDER CLEAR SKY',.                                  LWRO1660
     &        ' GIVEN BY EIGHT DIFFERENT EMPIRICAL FORMULAE *',/              LWRO1670
     &        //,9X,'AIR',4X,'REL.',16X,'LONGWAVE RADIATION GIVEN BY ',       LWRO1680
     &        'VARIOUS FORMULAE',43X,'STAND ')                                LWRO1690
      WRITE (35,1065)                                                        LWRO1700
1065  FORMAT ('MONTH',3X,'TEMP.',                                            LWRO1710
     &        3X,'HUM.',6X,'1',8X,'2',8X,'3',8X,'4',8X,'5',8X,'6',           LWRO1720
     &        8X,'7',8X,'8',7X,'MAX',6X,'MIN',5X,'MEAN',6X,'DEV',//,         LWRO1730
     &        132('-'),//)                                                   LWRO1740
      DO 160 I=1,NPTS                                                        LWRO1750
      RMEAN=(RADM1(I)+RADM2(I)+RADM3(I)+RADM4(I)+RADM5(I)+RADM6(I)+          LWRO1760
     &       RADM7(I)+RADM8(I))/8.0                                          LWRO1770
      RADMAX=AMAX1(RADM1(I),RADM2(I),RADM3(I),RADM4(I),                      LWRO1780
     &             RADM5(I),RADM6(I),RADM7(I),RADM8(I))                      LWRO1790
      RADMIN=AMIN1(RADM1(I),RADM2(I),RADM3(I),RADM4(I),                      LWRO1800
     &             RADM5(I),RADM6(I),RADM7(I),RADM8(I))                      LWRO1810
      VAR=((RADM1(I)-RMEAN)**2+(RADM2(I)-RMEAN)**2+                          LWRO1820
     &     (RADM3(I)-RMEAN)**2+(RADM4(I)-RMEAN)**2+                          LWRO1830
     &     (RADM5(I)-RMEAN)**2+(RADM6(I)-RMEAN)**2+                          LWRO1840
     &     (RADM7(I)-RMEAN)**2+(RADM8(I)-RMEAN)**2)/8.0                      LWRO1850
      STDEV=SQRT(VAR)                                                        LWRO1860
      IMONTH=I+7                                                             LWRO1870
200   IF (IMONTH.GT.12) IMONTH=IMONTH-12                                     LWRO1880
      IF (IMONTH.LE.12) GOTO 210                                             LWRO1890
      GOTO 200                                                               LWRO1900
210   WRITE (35,1070) IMONTH,TAIR(I),RHUM(I),RADM1(I),RADM2(I),RADM3(I),     LWRO1910
     &                RADM4(I),RADM5(I),RADM6(I),RADM7(I),RADM8(I),          LWRO1920
     &                RADMAX,RADMIN,RMEAN,STDEV                              LWRO1930
1070  FORMAT (I4,F8.1,F8.2,12F9.1)                                           LWRO1940
160   CONTINUE                                                               LWRO1950
      WRITE (35,1080)                                                        LWRO1960
1080  FORMAT (//,2X,'* TEMPERATURE IN DEGREES CELCIUS',/,                    LWRO1970
     &        4X,'RADIATION IN WATTS PER SQUARE METER',/,                    LWRO1980
     &        4X,'DATA START AUGUST 1978',/,                                 LWRO1990
     &        4X,'FORMULAE USED TO CALCULATE LONG-WAVE RADIATION:',/,        LWRO2000
     &        9X,'BRUNT(1932) ..... 1',/,                                    LWRO2010
     &        9X,'SWINBANK(1963) ..... 2',/,                                 LWRO2020
     &        9X,'IDSO-JACKSON(1969) ..... 3',/,                             LWRO2030
     &        9X,'BRUTSAERT 1(1975) ..... 4',/,                              LWRO2040
     &        9X,'SATTERLUND(1979) ..... 5',/,                               LWRO2050
     &        9X,'BRUTSAERT 2(1981) ..... 6',/,                              LWRO2060
     &        9X,'IDSO 1(1981) ..... 7',/,                                   LWRO2070
     &        9X,'IDSO 2(1981) ..... 8',/)                                   LWRO2080
      IF (IND2.EQ.2) GOTO 105                                                LWRO2090
      GOTO 130                                                               LWRO2100
                                                                             LWRO2110
C     ************************************************************** LWRO2120
C     *                      PLOTTING ROUTINE                        * LWRO2130
C     ************************************************************** LWRO2140
105   CALL PLOTS (120,IDUM,25)                                               LWRO2150
      DO 120 I=1,NPTS                                                        LWRO2160
      XMONTH(I)=FLOAT(I)/8.0                                                 LWRO2170
      RAD[illegible](I)=(RADM1(I)-190.0)/30.0                                LWRO2180
      [illegible]I)=(RADM2(I)-190.0)/30.0                                    LWRO2190
      [illegible])=(RADM3(I)-190.0)/30.0                                     LWRO2200
```

```
      RADM4(I)=(RADM4(I)-190.0)/30.0                                    LWRO2210
      RADM5(I)=(RADM5(I)-190.0)/30.0                                    LWRO2220
      RADM6(I)=(RADM6(I)-190.0)/30.0                                    LWRO2230
      RADM7(I)=(RADM7(I)-190.0)/30.0                                    LWRO2240
120   RADM8(I)=(RADM8(I)-190.0)/30.0                                    LWRO2250
      CALL SYMBOL (8.1,3.05,.1,1,0.0,-1)                                LWRO2260
      CALL SYMBOL (8.4,3.0,.1,'BRUNT(1932)',0.0,11)                     LWRO2270
      CALL SYMBOL (8.1,2.55,.1,2,0.0,-1)                                LWRO2280
      CALL SYMBOL (8.4,2.5,0.1,'SWINBANK(1963)',0.0,14)                 LWRO2290
      CALL SYMBOL (8.1,2.05,0.1,3,0.0,-1)                               LWRO2300
      CALL SYMBOL (8.4,2.0,0.1,'IDSO 1(1981)',0.0,12)                   LWRO2310
      CALL SYMBOL (8.1,1.55,0.1,4,0.0,-1)                               LWRO2320
      CALL SYMBOL (8.4,1.5,0.1,'IDSO 2(1981)',0.0,12)                   LWRO2330
      CALL AXIS1 (0.0,0.0,'MONTHS STARTING AUGUST 1978',27,             LWRO2340
     &           8.0,0.0,1.0,8.0,0.0,0.1.0)                             LWRO2350
      CALL AXIS1 (0.0,0.0,'RADIATION (WATT PER SQUARE METER)',33,       LWRO2360
     &           8.0,90.0,190.0,30.0,0.0,0.1.0)                         LWRO2370
      CALL GRAPH (XMONTH,RADM1,NPTS,0.1,1)                              LWRO2380
      CALL GRAPH (XMONTH,RADM2,NPTS,0.1,2)                              LWRO2390
      CALL GRAPH (XMONTH,RADM7,NPTS,0.1,3)                              LWRO2400
      CALL GRAPH (XMONTH,RADM8,NPTS,0.1,4)                              LWRO2410
C                                                                       LWRO2420
      CALL PLOT (0.0,0.0,-3)                                            LWRO2430
      CALL SYMBOL (8.1,3.05,.1,1,0.0,-1)                                LWRO2440
      CALL SYMBOL (8.4,3.0,.1,'SATTERLUND(1979)',0.0,16)                LWRO2450
      CALL SYMBOL (8.1,2.55,.1,2,0.0,-1)                                LWRO2460
      CALL SYMBOL (8.4,2.5,0.1,'BRUTSAERT 2(1981)',0.0,17)              LWRO2470
      CALL SYMBOL (8.1,2.05,0.1,3,0.0,-1)                               LWRO2480
      CALL SYMBOL (8.4,2.0,0.1,'IDSO-JACKSON(1969)',0.0,18)             LWRO2490
      CALL SYMBOL (8.1,1.55,0.1,4,0.0,-1)                               LWRO2500
      CALL SYMBOL (8.4,1.5,0.1,'BRUTSAERT 1(1975)',0.0,17)              LWRO2510
      CALL AXIS1 (0.0,0.0,'MONTHS STARTING AUGUST 1978',27,             LWRO2520
     &           8.0,0.0,1.0,8.0,0.0,0.1.0)                             LWRO2530
      CALL AXIS1 (0.0,0.0,'RADIATION (WATT PER SQUARE METER)',33,       LWRO2540
     &           8.0,90.0,190.0,30.0,0.0,0.1.0)                         LWRO2550
      CALL GRAPH (XMONTH,RADM5,NPTS,0.1,1)                              LWRO2560
      CALL GRAPH (XMONTH,RADM6,NPTS,0.1,2)                              LWRO2570
      CALL GRAPH (XMONTH,RADM3,NPTS,0.1,3)                              LWRO2580
      CALL GRAPH (XMONTH,RADM4,NPTS,0.1,4)                              LWRO2590
      CALL ENDPLT (12.0,0.0,999)                                        LWRO2600
      GOTO 130                                                          LWRO2610
                                                                        LWRO2620
110   CALL PLOTS(120,IDUM,25)                                           LWRO2630
      DO 20 I=NSTART,NEND                                               LWRO2640
           RAD1(I-NSTART+1)=RAD1(I)                                     LWRO2650
           RAD2(I-NSTART+1)=RAD2(I)                                     LWRO2660
           RAD3(I-NSTART+1)=RAD3(I)                                     LWRO2670
           RAD4(I-NSTART+1)=RAD4(I)                                     LWRO2680
           RAD5(I-NSTART+1)=RAD5(I)                                     LWRO2690
           RAD6(I-NSTART+1)=RAD6(I)                                     LWRO2700
           RAD7(I-NSTART+1)=RAD7(I)                                     LWRO2710
           RAD8(I-NSTART+1)=RAD8(I)                                     LWRO2720
20    DAYS(I-NSTART+1)=FLOAT(I)                                         LWRO2730
C     CALL SYMBOL (8.1,3.05,0.1,3,0.0,-1)                               LWRO2740
C     CALL SYMBOL (8.4,3.0,0.1,'BRUNT(1932)',0.0,11)                    LWRO2750
```

```
C      CALL SYMBOL (8.1,2.55,0.1,2,0.0,-1)                              LWR02760
C      CALL SYMBOL (8.4,2.5,0.1,'SWINBANK(1963)',0.0,14)                LWR02770
       CALL SYMBOL (8.1,2.05,0.1,1,0.0,-1)                              LWR02780
       CALL SYMBOL (8.4,2.0,0.1,'IDSO-JACKSON(1969)',0.0,18)            LWR02790
       CALL SYMBOL (8.1,1.55,0.1,4,0.0,-1)                              LWR02800
       CALL SYMBOL (8.4,1.5,0.1,'BRUTSAERT 1(1975)',0.0,17)             LWR02810
       CALL PICTUR (8.0,7.0,'DAYS SINCE 26 JULY 1978',23,               LWR02820
     &              'RADIATION (WATT PER SQUARE METER)',33,             LWR02830
C    &              DAYS,RAD1,NPTS,0.1,3,                               LWR02840
C    &              DAYS,RAD2,NPTS,0.1,2,                               LWR02850
     &              DAYS,RAD3,-NPTS,0.07,1,                             LWR02860
     &              DAYS,RAD4,-NPTS,0.07,4)                             LWR02870
                                                                        LWR02880
       CALL PLOT (0.0,0.0,-3)                                           LWR02890
       CALL SYMBOL (8.1,3.05,.1,1,0.0,-1)                               LWR02900
       CALL SYMBOL (8.4,3.0,.1,'SATTERLUND(1979)',0.0,16)               LWR02910
       CALL SYMBOL (8.1,2.55,.1,2,0.0,-1)                               LWR02920
       CALL SYMBOL (8.4,2.5,0.1,'BRUTSAERT 2(1981)',0.0,17)             LWR02930
       CALL SYMBOL (8.1,2.05,0.1,3,0.0,-1)                              LWR02940
       CALL SYMBOL (8.4,2.0,0.1,'IDSO 1(1981)',0.0,12)                  LWR02950
       CALL SYMBOL (8.1,1.55,0.1,4,0.0,-1)                              LWR02960
       CALL SYMBOL (8.4,1.5,0.1,'IDSO 2(1981)',0.0,12)                  LWR02970
       CALL PICTUR (8.0,7.0,'DAYS SINCE 26 JULY 1978',23,               LWR02980
     &              'RADIATION (WATT PER SQUARE METER)',33,             LWR02990
     &              DAYS,RAD5,NPTS,0.1,1,                               LWR03000
     &              DAYS,RAD6,NPTS,0.1,2,                               LWR03010
     &              DAYS,RAD7,NPTS,0.1,3,                               LWR03020
     &              DAYS,RAD8,NPTS,0.1,4)                               LWR03030
       CALL ENDPLT (12.0,0.0,999)                                       LWR03040
       GOTO 130                                                         LWR03050
                                                                        LWR03060
130    STOP                                                             LWR03070
       END                                                              LWR03080
```

calculates monthly-averaged raw error.

```
      DIMENSION D1(1893),M7(1893),TAV1(70)                              MON00010
      READ(10,1000) NDAY, N1, N2                                        MON00020
1000  FORMAT (3I5)                                                      MON00030
C     NDAY=TOTAL NO. OF DAYS;                                           MON00040
C     N1=FIRST DAY OF FIRST INTACT MONTH OF DATA;                       MON00050
C     N2=LAST DAY OF LAST INTACT MONTH OF DATA.                         MON00060
      WRITE (12,1005)                                                   MON00070
 1005 FORMAT (' MONTH',3X,'MEAN (C)',/)                                 MON00080
      READ (11,1002) M7                                                 MON00090
      READ (13,1004) D1                                                 MON00100
 1002 FORMAT (I2)                                                       MON00110
 1004 FORMAT (8F10.2)                                                   MON00120
C     SET INITIAL VALUES                                                MON00130
      K=0                                                               MON00140
      MON=M7(N1)                                                        MON00150
      TD1=0                                                             MON00160
      N3=N2+1                                                           MON00170
      L=0                                                               MON00180
C     CALCULATE MEAN                                                    MON00190
      DO 2 I=N1,N3                                                      MON00200
      IF (M7(I).NE.MON) GO TO 3                                         MON00210
    4 TD1=TD1+D1(I)                                                     MON00220
      K=K+1                                                             MON00230
    2 CONTINUE                                                          MON00240
      GO TO 5                                                           MON00250
    3 TAVER1=TD1/K                                                      MON00260
      L=L+1                                                             MON00270
      WRITE (12,1006) MON,TAVER1                                        MON00280
 1006 FORMAT(I6,F8.2)                                                   MON00290
      TAV1(L)=TAVER1                                                    MON00300
      MON=M7(I)                                                         MON00310
      TD1=0                                                             MON00320
      K=0                                                               MON00330
      GO TO 4                                                           MON00340
    5 STOP                                                              MON00350
      END                                                               MON00360
```

```
   39 L=L+1                                                              MONO1110
                                                                         MONO1120
      WRITE (9,1030) MON,TAVER1,TAVER2,TAVER3,TAVER4,TAVER5              MONO1130
 1030 FORMAT(I6,5(5X,F5.2))                                              MONO1140
      TAV1(L)=TAVER1                                                     MONO1150
      TAV2(L)=TAVER2                                                     MONO1160
      TAV3(L)=TAVER3                                                     MONO1170
      TAV4(L)=TAVER4                                                     MONO1180
      TAV5(L)=TAVER5                                                     MONO1190
      MON=M6(I)                                                          MONO1200
      TD1=0                                                              MONO1210
      TD2=0                                                              MONO1220
      TD3=0                                                              MONO1230
      TD4=0                                                              MONO1240
      TD5=0                                                              MONO1250
      K1=0                                                               MONO1260
      K2=0                                                               MONO1270
      K3=0                                                               MONO1280
      K4=0                                                               MONO1290
      K5=0                                                               MONO1300
      GO TO 13                                                           MONO1310
                                                                         MONO1320
C     CALCULATE STANDARD DEVIATION                                       MONO1330
                                                                         MONO1340
   14 L=1                                                                MONO1350
      K1=0                                                               MONO1360
      K2=0                                                               MONO1370
      K3=0                                                               MONO1380
      K4=0                                                               MONO1390
      K5=0                                                               MONO1400
      VA1=0                                                              MONO1410
      VA2=0                                                              MONO1420
      VA3=0                                                              MONO1430
      VA4=0                                                              MONO1440
      VA5=0                                                              MONO1450
      MON=M6(N1)                                                         MONO1460
      DO 20 I=N1,N3                                                      MONO1470
      IF (M6(I).NE.MON) GO TO 21                                         MONO1480
   29 IF (TAV1(L).EQ.9999.) GO TO 22                                     MONO1490
      VA1=(D1(I)-TAV1(L))**2+VA1                                         MONO1500
      K1=K1+1                                                            MONO1510
   22 IF (TAV2(L).EQ.9999.) GO TO 23                                     MONO1520
      VA2=(D2(I)-TAV2(L))**2+VA2                                         MONO1530
      K2=K2+1                                                            MONO1540
   23 IF (TAV3(L).EQ.9999.) GO TO 24                                     MONO1550
      VA3=(D3(I)-TAV3(L))**2+VA3                                         MONO1560
      K3=K3+1                                                            MONO1570
   24 IF (TAV4(L).EQ.9999.) GO TO 25                                     MONO1580
      VA4=(D4(I)-TAV4(L))**2+VA4                                         MONO1590
      K4=K4+1                                                            MONO1600
   25 IF (TAV5(L).EQ.9999.) GO TO 20                                     MONO1610
      VA5=(D5(I)-TAV5(L))**2+VA5                                         MONO1620
      K5=K5+1                                                            MONO1630
   20 CONTINUE                                                           MONO1640
      GO TO 27                                                           MONO1650
```

```
 21 IF (TAV1(L).EQ.9999.) GO TO 40                                  MON01660
    STD1=SQRT(VA1/K1)                                               MON01670
    GO TO 41                                                        MON01680
 40 STD1=9999.                                                      MON01690
 41 IF (TAV2(L).EQ.9999.) GO TO 42                                  MON01700
    STD2=SQRT(VA2/K2)                                               MON01710
    GO TO 43                                                        MON01720
 42 STD2=9999.                                                      MON01730
 43 IF (TAV3(L).EQ.9999.) GO TO 44                                  MON01740
    STD3=SQRT(VA3/K3)                                               MON01750
    GO TO 45                                                        MON01760
 44 STD3=9999.                                                      MON01770
 45 IF (TAV4(L).EQ.9999.) GO TO 46                                  MON01780
    STD4=SQRT(VA4/K4)                                               MON01790
    GO TO 47                                                        MON01800
 46 STD4=9999.                                                      MON01810
 47 IF (TAV5(L).EQ.9999.) GO TO 48                                  MON01820
    STD5=SQRT(VA5/K5)                                               MON01830
    GO TO 49                                                        MON01840
 48 STD5=9999.                                                      MON01850
 49 WRITE (10,1030) MON,STD1,STD2,STD3,STD4,STD5                    MON01860
    MON=M6(I)                                                       MON01870
    VA1=0                                                           MON01880
    VA2=0                                                           MON01890
    VA3=0                                                           MON01900
    VA4=0                                                           MON01910
    VA5=0                                                           MON01920
    K1=0                                                            MON01930
    K2=0                                                            MON01940
    K3=0                                                            MON01950
    K4=0                                                            MON01960
    K5=0                                                            MON01970
    L=L+1                                                           MON01980
    GO TO 29                                                        MON01990
 27 STOP                                                            MON02000
    END                                                             MON02010
```

```
C     THIS PROGRAM CALCULATES THE MONTHLY AVERAGED MEASURED TEMPERATURE MON00010
C     AT 5 DIFFERENT STATIONS OF THE COOLING POND SYSTEM AND ALSO       MON00020
C     THEIR MONTHLY STANDARD DEVIATIONS. .                              MON00030
C     COMPLETED ON 7 JANUARY 1984.                                      MON00040
                                                                        MON00050
      DIMENSION D1(1967),D2(1967),D3(1967),D4(1967),D5(1967),           MON00060
     *          M6(2000),TAV1(70),TAV2(70),TAV3(70),                   MON00070
     *          TAV4(70),TAV5(70)                                      MON00080
                                                                        MON00090
C     D1=THURMAN IS., D2=INTAKE, D3=BR. PT., D4=DISCHARGE, D5=DYKE 3    MON00100
C     M6=MONTH                                                          MON00110
                                                                        MON00120
      READ(8,1000) NDAY, N1, N2                                         MON00130
 1000 FORMAT (3I5)                                                      MON00140
                                                                        MON00150
C     NDAY=TOTAL NO. OF DAYS;                                           MON00160
C     N1=FIRST DAY OF FIRST INTACT MONTH OF DATA;                       MON00170
C     N2=LAST DAY OF LAST INTACT MONTH OF DATA.                         MON00180
                                                                        MON00190
      WRITE (9,1005)                                                    MON00200
 1005 FORMAT (' MONTH',4X,'TH. IS.',3X,'INTAKE',4X,'BR. PT.',4X,        MON00210
     *'TEXIT',5X,'DIKE3',/)                                             MON00220
      WRITE (10,1006)                                                   MON00230
 1006 FORMAT (' MONTH',4X,'TH. IS.',3X,'INTAKE',4X,'BR. PT.',4X,        MON00240
     *'TEXIT',5X,'DIKE3',/)                                             MON00250
                                                                        MON00260
C     READ NRC DATA                                                     MON00270
                                                                        MON00280
      READ (11,1010) D1                                                 MON00290
      READ (12,1010) D2                                                 MON00300
      READ (13,1010) D3                                                 MON00310
      READ (14,1010) D4                                                 MON00320
      READ (15,1010) D5                                                 MON00330
      DO 1 I=1,NDAY                                                     MON00340
      IF (D1(I).EQ.0.0) D1(I)=212.0                                     MON00350
      D1(I)=(D1(I)-32.)*5./9.                                           MON00360
      IF (D2(I).EQ.0.0) D2(I)=212.0                                     MON00370
      D2(I)=(D2(I)-32.)*5./9.                                           MON00380
      IF (D3(I).EQ.0.0) D3(I)=212.0                                     MON00390
      D3(I)=(D3(I)-32.)*5./9.                                           MON00400
      IF (D4(I).EQ.0.0) D4(I)=212.0                                     MON00410
      D4(I)=(D4(I)-32.)*5./9.                                           MON00420
      IF (D5(I).EQ.0.0) D5(I)=212.0                                     MON00430
      D5(I)=(D5(I)-32.)*5./9.                                           MON00440
    1 CONTINUE                                                          MON00450
 1010 FORMAT (8F10.2)                                                   MON00460
                                                                        MON00470
C     READ CALENDAR                                                     MON00480
                                                                        MON00490
      DO 2 I=56,NDAY                                                    MON00500
    2 READ (16,1020) M6(I)                                              MON00510
 1020 FORMAT (I2)                                                       MON00520
                                                                        MON00530
C     SET INITIAL VALUES                                                MON00540
                                                                        MON00550
```

```
      K1=0                                                          MON00560
      K2=0                                                          MON00570
      K3=0                                                          MON00580
      K4=0                                                          MON00590
      K5=0                                                          MON00600
      MON=M6(N1)                                                    MON00610
      TD1=0                                                         MON00620
      TD2=0                                                         MON00630
      TD3=0                                                         MON00640
      TD4=0                                                         MON00650
      TD5=0                                                         MON00660
      N3=N2+1                                                       MON00670
      L=0                                                           MON00680
                                                                    MON00690
C     CALCULATE MEAN                                                MON00700
                                                                    MON00710
      DO 3 I=N1,N3                                                  MON00720
   13 IF (M6(I).NE.MON) GO TO 4                                     MON00730
      IF (D1(I).EQ.100.0) GO TO 5                                   MON00740
      TD1=TD1+D1(I)                                                 MON00750
      K1=K1+1                                                       MON00760
    5 IF (D2(I).EQ.100.0) GO TO 7                                   MON00770
      TD2=TD2+D2(I)                                                 MON00780
      K2=K2+1                                                       MON00790
    7 IF (D3(I).EQ.100.0) GO TO 9                                   MON00800
      TD3=TD3+D3(I)                                                 MON00810
      K3=K3+1                                                       MON00820
    9 IF (D4(I).EQ.100.0) GO TO 11                                  MON00830
      TD4=TD4+D4(I)                                                 MON00840
      K4=K4+1                                                       MON00850
   11 IF (D5(I).EQ.100.0) GO TO 3                                   MON00860
      TD5=TD5+D5(I)                                                 MON00870
      K5=K5+1                                                       MON00880
    3 CONTINUE                                                      MON00890
      GO TO 14                                                      MON00900
    4 IF (K1.GE.28) GO TO 30                                        MON00910
      TAVER1=9999.                                                  MON00920
      GO TO 31                                                      MON00930
   30 TAVER1=TD1/K1                                                 MON00940
   31 IF (K2.GE.28) GO TO 32                                        MON00950
      TAVER2=9999.                                                  MON00960
      GO TO 33                                                      MON00970
   32 TAVER2=TD2/K2                                                 MON00980
   33 IF (K3.GE.28) GO TO 34                                        MON00990
      TAVER3=9999.                                                  MON01000
      GO TO 35                                                      MON01010
   34 TAVER3=TD3/K3                                                 MON01020
   35 IF (K4.GE.28) GO TO 36                                        MON01030
      TAVER4=9999.                                                  MON01040
      GO TO 37                                                      MON01050
   36 TAVER4=TD4/K4                                                 MON01060
   37 IF (K5.GE.28) GO TO 38                                        MON01070
      TAVER5=9999.                                                  MON01080
      GO TO 14                                                      MON01090
   38 TAVER5=TD5/K5                                                 MON01100
```

```
C     CALCULATE STANDARD DEVIATION                                    MONO1110
                                                                      MONO1120
    5 L=1                                                             MONO1130
      K=0                                                             MONO1140
      MON=M7(N1)                                                      MONO1150
      DO 10 J=N1,N3                                                   MONO1160
      IF (M7(J).NE.MON) GO TO 12                                      MONO1170
   11 VA1=(D1(J)-TAV1(L))**2+VA1                                      MONO1180
      VA2=(D2(J)-TAV2(L))**2+VA2                                      MONO1190
      VA3=(D3(J)-TAV3(L))**2+VA3                                      MONO1200
      VA4=(D4(J)-TAV4(L))**2+VA4                                      MONO1210
      VA5=(D5(J)-TAV5(L))**2+VA5                                      MONO1220
      VA6=(D6(J)-TAV6(L))**2+VA6                                      MONO1230
      VA7=(D7(J)-TAV7(L))**2+VA7                                      MONO1240
      K=K+1                                                           MONO1250
   10 CONTINUE                                                        MONO1260
      GO TO 7                                                         MONO1270
   12 L=L+1                                                           MONO1280
      STD1=SQRT(VA1/K)                                                MONO1290
      STD2=SQRT(VA2/K)                                                MONO1300
      STD3=SQRT(VA3/K)                                                MONO1310
      STD4=SQRT(VA4/K)                                                MONO1320
      STD5=SQRT(VA5/K)                                                MONO1330
      STD6=SQRT(VA6/K)                                                MONO1340
      STD7=SQRT(VA7/K)                                                MONO1350
      WRITE (15,1100) MON,STD7,STD5,STD4,STD3,STD2,STD1,STD6          MONO1360
 1100 FORMAT (I6,7F8.2)                                               MONO1370
      MON=M7(J)                                                       MONO1380
      VA1=0                                                           MONO1390
      VA2=0                                                           MONO1400
      VA3=0                                                           MONO1410
      VA4=0                                                           MONO1420
      VA5=0                                                           MONO1430
      VA6=0                                                           MONO1440
      VA7=0                                                           MONO1450
      K=0                                                             MONO1460
      GO TO 11                                                        MONO1470
    7 STOP                                                            MONO1480
      END                                                             MONO1490
```

```
C     PROGRAM PROCESSES SIDEARM FIELD DATA AND                          SID00010
C     COMPARES WITH MODEL PREDICTIONS                                   SID00020
C     THIS PROGRAM HAS BEEN EXTENSIVELY REVISED FROM P. SHANAHAN'S      SID00030
C     ORIGINAL PROGRAM BY S. WELLS(MAY-JULY 1982)                       SID00040
C     AGAIN REVISED BY EDMOND K. HO (JULY,1983)                         SID00050
C     SUBROUTINE THERMO WAS WRITTEN BY E. HO (JULY,1982)                SID00060
C                                                                       SID00070
      REAL LSA                                                          SID00080
C                                                                       SID00090
      DOUBLE PRECISION TIME, DATE                                       SID00100
      DIMENSION TSTA(2), NAME(4), TEMP(100,20), VEL(100,20)             SID00110
      DIMENSION XLPS(11), QPS(11), TPS(11),QQ(2),GTB(2),GQB(2),AC(12),SASID00120
     *L(3),SAW(3),SAWC(3),SAH(3),SAHC(3),TEQL(200),SSK(200),            SID00130
     *IDAY(100),IHOUR(100),IYR(100),IMO(100),                           SID00140
     *NPTS(100),NTEMP(100),TEND(100),BF(2),BF2(2),TF(2),G(2),Q1(2),BFH(2SID00150
     *),OMEGA(2),DAY1(100),RLBDA(21),GLBDA(21),TRFB(2),XL(2),THS(2),TFH SID00160
     *(2)                                                               SID00170
      DATA XLPS /0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0/ SID00180
      DATA QPS / .48, .47, .45, .44, .43, .42, .41, .40, .40, .39, .38/ SID00190
      DATA TPS / 1.0, .73, .54, .39, .27, .18, .11, .07, .04, .02, 0.0/ SID00200
      DATA RLBDA /0.0,0.05,0.1,0.15,0.2,0.25,0.3,0.35,0.4,0.45,0.5,0.55,SID00210
     *0.6,0.65,0.7,0.75,0.8,0.85,0.9,0.95,1.0/                          SID00220
      DATA GLBDA /1.39,1.378,1.362,1.344,1.334,1.317,1.285,1.265,1.247, SID00230
     *1.225,1.198,1.16,1.095,0.986,0.85,0.65,0.406,0.23,0.11,0.04,0.005/SID00240
C  THE LAST DATA POINT ABOVE SHOULD BE 0.0 INSTEAD OF 0.005             SID00250
C  BUT BECAUSE OF COMPUTATIONAL PROBLEMS WITH THE ITERATION SCHEME      SID00260
C  A NON-ZERO VALUE WAS REQUIRED-THIS SHOULD NOT AFFECT THE RESULTS     SID00270
C                                                                       SID00280
      IRITE = 6                                                         SID00290
      READ(10,2001)NAME                                                 SID00300
      READ(10,2003)(SAL(I),I=1,3),(SAH(I),I=1,3),FO,ALPHA,TRF           SID00310
 2003 FORMAT(9F8.2)                                                     SID00320
      READ(10,2003)(SAW(I),I=1,3),(SAWC(I),I=1,3),(SAHC(I),I=1,3)       SID00330
      READ(10,2004)(AC(I),I=1,12)                                       SID00340
 2004 FORMAT(12F6.2)                                                    SID00350
      READ(10,2133)ATC,AT,ASUR,WCTOP                                    SID00360
      READ(10,2133)VOLSA,HBAR1,HBAR3,RSAL,RSAW,RSAH                     SID00370
 2133 FORMAT(6F12.1)                                                    SID00380
C     THE DEBUG CARDS DO THE FOLLOWING;                                 SID00390
C     NBUG1=1...OUTPUTS PARAMETERS OF THE HEAD LOSS CONVERGENCE         SID00400
C              ITERATION SCHEME                                         SID00410
C     NBUG2=2...OUTPUT OF DAILY RESULTS IS SUPRESSED, ONLY INITIAL      SID00420
C              DATA AND STATISTICS ARE PRINTED                          SID00430
C     (=1:ALL OUTPUT IS PRINTED,=2:DAILY STUFF SUPRESSED BUT CONDENSED  SID00440
C     FORM PRINTED,=3 OR MORE:SAME AS =1,=0:ONLY DAILY STUFF WRITTEN???)SID00450
C     NBUG3=3...TRF IS SET AT INPUT VALUE AND IS NOT COMPUTED           SID00460
C              BY THE SUBROUTINE TRED                                   SID00470
C     NBUG4=4...SEVERAL DIAGNOSTICS ARE CHECKED THROUGHOUT THE PROGRAM  SID00480
C              SUCH AS THERMO AND FLOW MODEL CALCULATIONS               SID00490
C     NBUG5=5...PROGRAM MERELY RUNS WITH DATA FROM MP2 INSIDE THE       SID00500
C              CONSTRICTION-JUST A COMPARISON OF MODEL PREDICTIONS       SID00510
C              WITHOUT THE INFLUENCE OF THE HEAD LOSS(SET NBUG2=0)      SID00520
C     NBUG6=6...OUTPUT OF THE FILTERING PARAMETERS                      SID00530
C              (IF >7 ONLY PARTIAL OUTPUT)                              SID00540
C                                                                       SID00550
```

```
      READ(10,2005)NDAYS,NGEOM,NBUG1,NBUG2,NBUG3,NBUG4,NBUG5,NBUG6      SID00560
      READ(10,2005)IBM,IBHM,ICM,ISM,IJM                                 SID00570
C                                                                       SID00580
C     THESE BUGS TURN ON OR OFF THE VARIOUS MODELS(IF=1:OFF-            SID00590
C     IF=0;ON)                                                          SID00600
C                                                                       SID00610
 2005 FORMAT(12I5)                                                      SID00620
C     READ DATA FOR EACH DAY                                            SID00630
      DO 23 J=1,NDAYS                                                   SID00640
         READ(10,2006)IMO(J),IDAY(J),IYR(J),IHOUR(J)                    SID00650
 2006    FORMAT(I2,1X,I2,1X,I2,1X,I5)                                   SID00660
         READ(10,2005)NPTS(J),NTEMP(J)                                  SID00670
         NNN=NPTS(J)                                                    SID00680
         IF(NBUG5.NE.5)READ(10,2004)(VEL(J,K),K=1,NNN)                  SID00690
         READ(10,2004)(TEMP(J,K),K=1,NNN)                               SID00700
         IF(NBUG5.EQ.5)READ(10,2004)HOI                                 SID00710
         IF(NBUG5.NE.5)READ(10,2003)TEND(J)                             SID00720
   23 CONTINUE                                                          SID00730
      DO 26 I=1,NDAYS                                                   SID00740
         NN=NPTS(I)                                                     SID00750
         DO 50 J=1,NN                                                   SID00760
   50    TEMP(I,J)=CTOF(TEMP(I,J))                                      SID00770
   26 TEND(I)=CTOF(TEND(I))                                             SID00780
                                                                        SID00790
      CALL TRANS(NDAYS,IDAY,IMO,IYR,TEQL,SSK,TEMP,NTEMP,VOLSA,          SID00800
     *HBAR1,HBAR3,RSAH,RSAW,RSAL,DEDVOL,NBUG6)                          SID00810
                                                                        SID00820
C ********************************************************************** SID00830
C *                BEGINNING OF MAIN LOOP                              * SID00840
C ********************************************************************** SID00850
                                                                        SID00860
      DO 3 JJ=1,NGEOM                                                   SID00870
      NGT=0                                                             SID00880
      NGO=0                                                             SID00890
      NGH=0                                                             SID00900
      NGSH=0                                                            SID00910
      DO 2 J=1,2                                                        SID00920
         GTB(J)=0.0                                                     SID00930
    2 GQB(J)=0.0                                                        SID00940
      NGH=0                                                             SID00950
      GTC=0.0                                                           SID00960
      GQC=0.0                                                           SID00970
      GTS=0.0                                                           SID00980
      GQS=0.0                                                           SID00990
      GQH=0.0                                                           SID01000
      GTH=0.0                                                           SID01010
      DAY2=0.0                                                          SID01020
                                                                        SID01030
      CALL DATIME(DATE,TIME)                                            SID01040
C                                                                       SID01050
C     PRINT BACK GEOMETRICAL AND BASIC DATA                             SID01060
C                                                                       SID01070
      WRITE(WRITE,3050)DATE,TIME                                        SID01080
      W     RITE,3051)NAME,SAL(JJ),SAH(JJ),SAW(JJ),SAWC(JJ),SAHC      FOSID01090
     *,TRF,AT,ATC,ASUR,VOLSA,WCTOP,DEDVOL                               SID01100
```

```
      WRITE(IRITE,3053)                                                 SID01110
      DO 683 N=1,12                                                     SID01120
         J=N-1                                                          SID01130
  683 WRITE(IRITE,3052)J,N,AC(N)                                        SID01140
 3052 FORMAT(10X,'FROM',1X,I3,1X,'TO',1X,I3,15X,F9.2)                   SID01150
 3053 FORMAT(////5X,'CROSS-SECTIONAL AREA PER DEPTH AT CONSTRICTION'/   SID01160
     *5X,46('-')//10X,'DEPTH LEVEL(METER)',10X,'AREA(FT*FT)'/)          SID01170
                                                                        SID01180
      BF1 = FO*SAL(JJ)**3/((62.4*86400.0)**2*32.2*4.0*SAH(JJ)**4)       SID01190
                                                                        SID01200
C     ******************************                                    SID01210
C     ANALYSIS OF FIELD OBSERVATIONS                                    SID01220
C     ******************************                                    SID01230
                                                                        SID01240
      IF(NBUG2.GE.1.AND.JJ.EQ.1)WRITE(15,5050)                          SID01250
      IF(NBUG2.GE.1)WRITE (16,5053)                                     SID01260
                                                                        SID01270
      DO 70 II=1,NDAYS                                                  SID01280
      NN=NPTS(II)                                                       SID01290
      IT=0                                                              SID01300
      TS=TEMP(II,1)                                                     SID01310
      IF(NTEMP(II).EQ.2)TS=TEMP(II,2)                                   SID01320
      CALL DDAY(IMO(II),IDAY(II),IYR(II),DAY)                           SID01330
      DAY1(II)=DAY                                                      SID01340
      IF(II.GT.1)DAY2=DAY-DAY1(II-1)                                    SID01350
      IF(DAY2.LT.0.0)DAY2=DAY+365-DAY1(II-1)                            SID01360
C     NOTE THAT THE SURFACE HEAT EXCHANGE IS BASED ON THE TS TEMP,      SID01370
C     WHILE THE DRIVING FORCE FOR THE MODEL PREDICTIONS IS THE          SID01380
C     TUPPER TEMP                                                       SID01390
      SK=SSK(II)                                                        SID01400
      TEQ=TEQL(II)                                                      SID01410
      SKK=SK/(86400.0*62.4)                                             SID01420
      IF(SKK.LE.0.0)SKK=0.0                                             SID01430
C                                                                       SID01440
C     EVALUATE MEASURED VELOCITY AND TEMPERATURE PROFILE                SID01450
C                                                                       SID01460
      IF(NBUG5.EQ.5)GO TO 6500                                          SID01470
      AA1=0.0                                                           SID01480
      QQ1 = 0.0                                                         SID01490
      TUPPER = 0.0                                                      SID01500
      DO 250 J=2,NN                                                     SID01510
         IF((VEL(II,J)*VEL(II,J-1)).GT.0.0)GO TO 240                    SID01520
         FRAC = VEL(II,J-1)/(VEL(II,J-1)-VEL(II,J))                     SID01530
         QQ1 = QQ1+FRAC*VEL(II,J-1)*AC(J-1)                             SID01540
         HOI = FLOAT(J-2)+FRAC                                          SID01550
         HO = 3.281*HOI                                                 SID01560
         TUPPER = TUPPER + ((2.0-FRAC)*TEMP(II,J-1)+FRAC*TEMP(II,J))*   SID01570
     &   0.50*FRAC*AC(J-1)*VEL(II,J-1)*0.5                              SID01580
         QUPPER = (QQ1/2.0)                                             SID01590
         TUPPER=TUPPER/QUPPER                                           SID01600
         AA1=AA1+(FRAC*AC(J-1))                                         SID01610
         VUPPER = (QUPPER/AA1)                                          SID01620
         GO TO 260                                                      SID01630
240      TUPPER = TUPPER + (TEMP(II,J)+TEMP(II,J-1))*AC(J-1)*0.5*       SID01640
     &   (VEL(II,J)+VEL(II,J-1))*0.5                                    SID01650
```

```
         QO1 = QO1+(VEL(II,J)+VEL(II,J-1))*(AC(J-1))                    SID01660
250   AA1=AC(J-1)+AA1                                                    SID01670
260   IF(NTEMP(II).EQ.2)TUPPER=TEMP(II,2)                                SID01680
C                                                                        SID01690
C     LOWER LAYER                                                        SID01700
C                                                                        SID01710
      AA2=0.0                                                            SID01720
      QO1=VEL(II,J)*(1.0-FRAC)*AC(J-1)                                   SID01730
      TLOWER = ( (1.0-FRAC)*TEMP(II,J-1)+(1.0+FRAC)*TEMP(II,J) )*0.5     SID01740
     &         *(1.0-FRAC)*AC(J-1)*VEL(II,J)*0.5                         SID01750
      J1 = J+1                                                           SID01760
      IF(J1.GE.NN)GO TO 280                                              SID01770
      DO 270 J=J1,NN                                                     SID01780
         QO1 = QO1+(VEL(II,J)+VEL(II,J-1))*(AC(J-1))                     SID01790
        TLOWER = TLOWER + (TEMP(II,J)+TEMP(II,J-1))*AC(J-1)*0.5*         SID01800
     &   (VEL(II,J)+VEL(II,J-1))*0.5                                     SID01810
         AA2=AC(J-1)+AA2                                                 SID01820
 270  CONTINUE                                                           SID01830
 280  QLOWER = (QO1/2.0)                                                 SID01840
      HOLI = NN-1-HOI                                                    SID01850
      HOL = HOLI*3.281                                                   SID01860
      AA2=AA2+(1.0-FRAC)*AC(J1-2)                                        SID01870
      VLOWER = (QLOWER/AA2)                                              SID01880
      TLOWER = TLOWER/QLOWER                                             SID01890
      IF(NTEMP(II).EQ.2)TLOWER=TEMP(II,NN)                               SID01900
      GO TO 6501                                                         SID01910
 6500 TUPPER=TS                                                          SID01920
C                                                                        SID01930
C    COEFF OF THERMAL EXPANSION                                          SID01940
C                                                                        SID01950
 6501 TSTAR=TUPPER-(0.6*(TUPPER-TEQ))                                    SID01960
      IF(TSTAR.GT.TUPPER)TSTAR=TUPPER                                    SID01970
      IF(TSTAR.LT.39.0)TSTAR=39.0                                        SID01980
      TBETA=(TSTAR+TUPPER)/2.0                                           SID01990
      BETA=EXCOEF(TBETA)                                                 SID02000
C                                                                        SID02010
C     DETERMINATION OF THERMOCLINE DEPTH                                 SID02020
C                                                                        SID02030
      HOT=HOI                                                            SID02040
      IF(NTEMP(II).NE.2)CALL THERMO(TEMP,II,HOT,HOI,NBUG4)               SID02050
      HOT=3.2808*HOT                                                     SID02060
      IF(NBUG5.EQ.5)HO=HOT                                               SID02070
      IF(NBUG5.EQ.5)GO TO 6560                                           SID02080
                                                                         SID02090
      QUPPER = ABS(QUPPER)                                               SID02100
      QLOWER = ABS(QLOWER)                                               SID02110
C                                                                        SID02120
C     COMPUTE ENTRANCE FROUDE NUMBER FROM OBSERVED DATA                  SID02130
C                                                                        SID02140
      TBAR = 0.5*(TLOWER+TUPPER)                                         SID02150
      BET2 = EXCOEF(TBAR)                                                SID02160
      DELRHO = BET2*(TUPPER-TLOWER)                                      SID02170
      UBAR=(ABS(VUPPER)+ABS(VLOWER))/2.0                                 SID02180
      P    *AA1/HO)-WCTOP                                                SID02190
           BS(VUPPER**2/(DELRHO*32.2*AA1))*BI                            SID02200
```

```
      FR2 = ABS(VLOWER**2/(DELRHO*32.2*AA2))*BI                          SID02210
      FR = FR1+FR2                                                       SID02220
      FR1 = SQRT(FR1)                                                    SID02230
      FR2 = SQRT(FR2)                                                    SID02240
C                                                                        SID02250
C     AVERAGE FROUDE NO.                                                 SID02260
C                                                                        SID02270
      ABAR=(AA1+AA2)/2.                                                  SID02280
      HBAR=SAH(JJ)*(1.-SQRT(1.-(2.*ABAR/(SAH(JJ)*WCTOP))))               SID02290
      BI2=(2.*ABAR/HBAR)-WCTOP                                           SID02300
      FRD=(UBAR**2)*BI2/(DELRHO*32.2*ABAR)                               SID02310
                                                                         SID02320
C     ********************************                                   SID02330
C     COMPUTATION OF MODEL PREDICTIONS                                   SID02340
C     ********************************                                   SID02350
C                                                                        SID02360
C     DETERMINE BROCARD MODEL PREDICITIONS                               SID02370
C                                                                        SID02380
6560  CONTINUE                                                           SID02390
      H1=HO/SAH(JJ)                                                      SID02400
      H1I=1.0/H1                                                         SID02410
      IF(IBM.EQ.1)GO TO 808                                              SID02420
                                                                         SID02430
      DO 13 NJ=1,2                                                       SID02440
      IF((TUPPER-TEQ).LE.0.0)GO TO 771                                   SID02450
      QRAT = QLOWER/QUPPER                                               SID02460
      IF(NJ.EQ.1)QRAT = 1.0                                              SID02470
      BF2(NJ) = ALPHA*(H1I+QRAT/(1.0-H1))**2*(H1I+1.0/(1.0-H1)) +        SID02480
     +          QRAT**2/(1.0-H1)**3                                      SID02490
      BF2(NJ) = BF2(NJ)/H1                                               SID02500
      BF(NJ) = BF1*SK**2/(BETA*(TUPPER-TEQ))*BF2(NJ)                     SID02510
      BF2(NJ) = BF(NJ)**(-0.3333)                                        SID02520
      MM=0                                                               SID02530
      TB1=TSTAR                                                          SID02540
  699 CONTINUE                                                           SID02550
      MM=MM+1                                                            SID02560
      TB=(TB1-TEQ)/(TUPPER-TEQ)                                          SID02570
      QO(NJ)=BF2(NJ)*SAL(JJ)*SKK*TB**0.333                               SID02580
 5900 FORMAT(1X,'BROCARD CALCULATION-TB,TBB,Q:',F7.4,1X,F7.4,2X,F10.4)   SID02590
      IF(NBUG3.NE.3)CALL TRED(BETA,TSTAR,TUPPER,HO,QO(NJ),SAL(JJ),TRF)   SID02600
      TBB=EXP(-TRF*SKK*SAL(JJ)/QO(NJ))                                   SID02610
      IF(NBUG4.GT.4)WRITE(7,5900)TB,TBB,QO(NJ)                           SID02620
      TRFB(NJ)=TRF                                                       SID02630
      TB1=TBB*(TUPPER-TEQ)+TEQ                                           SID02640
      TF(NJ)=TB1                                                         SID02650
      IF(TBB.LT.0.0001)GO TO 772                                         SID02660
      IF(MM.GT.35)GO TO 5001                                             SID02670
      IF(ABS(TBB-TB).GT.0.01)GO TO 699                                   SID02680
      GO TO 772                                                          SID02690
 5001 WRITE(7,5002)II                                                    SID02700
 5002 FORMAT(1X,'ERROR IN TEMP ITERATION IN BROCARDS MODEL ON DAY',I5)   SID02710
      GO TO 772                                                          SID02720
  771 TF(NJ)=TEQ                                                         SID02730
      IT=1                                                               SID02740
      BF2(NJ)=0.0                                                        SID02750
```

```
      QO(NJ)=0.0                                                        SID02760
  772 CONTINUE                                                          SID02770
      IF(TF(NJ).LT.39.0)TF(NJ)=39.0                                     SID02780
      IF(IT.EQ.0)GTB(NJ) = GTB(NJ)+(TF(NJ)-TLOWER)**2                   SID02790
C     ADJUST FLOW TO CFS                                                SID02800
      QO(NJ)=QO(NJ)*SAW(JJ)                                             SID02810
   13 IF(IT.EQ.0)GQB(NJ) = GQB(NJ)+(QO(NJ)-QUPPER)**2                   SID02820
                                                                        SID02830
      IF(IT.EQ.0)NGT = NGT+1                                            SID02840
      IF(IT.EQ.0)NGQ = NGQ+1                                            SID02850
C                                                                       SID02860
C DETERMINE CRITICAL FLOW MODEL PREDICTIONS FOR A LATERAL CONSTRICTION  SID02870
C                                                                       SID02880
  808 IF(ICM.EQ.1)GO TO 809                                             SID02890
      MOLO=0                                                            SID02900
      MOL=0                                                             SID02910
      AC3=ATC*(HBAR3/RSAH)**2                                           SID02920
      AAC=ATC-AC3                                                       SID02930
      H1C=RSAH*(1.-SQRT(1.-(AAC/RSAH/WCTOP)))                           SID02940
      BIC=WCTOP*(1.-H1C/RSAH)                                           SID02950
      TFAC=1.0                                                          SID02960
      LSA=SAL(JJ)                                                       SID02970
      IF((TUPPER-TEQ).LE.0.0)GO TO 773                                  SID02980
      GO TO 806                                                         SID02990
  805 LSA=(-QOC/SKK*TRF)*ALOG((39.0-TEQ)/(TUPPER-TEQ))                  SID03000
      MOLO=MOLO+1                                                       SID03010
C     THESE ARE PART OF THE OLDER CRITICAL MODEL FORMULATION            SID03020
C 806 BC = (SKK*LSA)**2/(BETA*(TUPPER-TEQ)*32.2*SAH(JJ)**3)             SID03030
C     CRATIO = SAWC(JJ)/SAW(JJ)                                         SID03040
C     BFC = BC*(((H1I)**3+(1./(1.-H1))**3)/CRATIO**2)                   SID03050
  806 BC=32.2*BETA*(TUPPER-TEQ)                                         SID03060
      CR=AAC**3/BIC/16.                                                 SID03070
      BFC=BC*CR                                                         SID03080
  807 TFF=TFAC                                                          SID03090
      MOL=MOL+1                                                         SID03100
      IF(NBUG3.NE.3)CALL TRED(BETA,TSTAR,TUPPER,HO,QOC,LSA,TRF)         SID03110
C     BF2C = BFC**(-0.5)*TFF**0.5                                       SID03120
      QOC = SQRT(BFC*TFAC)                                              SID03130
      TFAC=(1.0-EXP(-TRF*SKK*LSA*SAW(JJ)/QOC))                          SID03140
      IF(MOL.GT.35)GO TO 7001                                           SID03150
      IF(ABS(TFAC-TFF).GT.0.005)GO TO 807                               SID03160
      GO TO 7004                                                        SID03170
 7001 WRITE(7,7003)II                                                   SID03180
 7003 FORMAT(1X,'ERROR IN CRITICAL FLOW ITERATION ON DAY',I5)           SID03190
 7004 TRFC=TRF                                                          SID03200
      TFC =1.0-TFAC                                                     SID03210
      GO TO 774                                                         SID03220
  773 TFC=0.0                                                           SID03230
      BF2C=0.0                                                          SID03240
      QOC=0.0                                                           SID03250
  774 TFC = TFC*(TUPPER-TEQ)+TEQ                                        SID03260
      IF(TFC.LT.39.0.AND.MOLO.LT.20) GO TO 805                          SID03270
      IF(IT.EQ.0)GTC = GTC+(TFC-TLOWER)**2                              SID03280
      IF(IT.EQ.0)GQC = GQC+(QOC-QUPPER)**2                              SID03290
C                                                                       SID03300
```

```
C     DETERMINE JAIN MODEL PREDICTIONS                                  SID03310
C                                                                       SID03320
  809 IF(IJM.EQ.1)GO TO 810                                             SID03330
      IF((TUPPER-TEQ).LE.0.0)GO TO 775                                  SID03340
      A1 = 0.45                                                         SID03350
      A2 = 5.2                                                          SID03360
      C = 0.208                                                         SID03370
      QOJ = SAH(JJ)*SQRT(BETA*32.2*(TUPPER-TEQ)*SAH(JJ)/A2)             SID03380
      XLE = QOJ/(A2*SKK)                                                SID03390
      QOJ = C*A1*QOJ                                                    SID03400
      TFJ = (TUPPER-TEQ)*((XLE-SAL(JJ))/XLE)**2 + TEQ                   SID03410
      IF(TFJ.LT.39.0)TFJ=39.0                                           SID03420
      GO TO 776                                                         SID03430
  775 TFJ=TEQ                                                           SID03440
      QOJ=0.0                                                           SID03450
  776 IF(IT.EQ.0)GTJ = GTJ+(TFJ-TLOWER)**2                              SID03460
      QOJ=QOJ*SAW(JJ)                                                   SID03470
      IF(IT.EQ.0)GQJ = GQJ+(QOJ-QUPPER)**2                              SID03480
C                                                                       SID03490
C     DETERMINE STURM MODEL PREDICTION PARAMETERS                       SID03500
C                                                                       SID03510
  810 IF(ISM.EQ.1)GO TO 602                                             SID03520
      IF((TUPPER-TEQ).LE.0.0)GO TO 777                                  SID03530
      VISC = 2.11E-05                                                   SID03540
      BLMBDA=1.0-H1                                                     SID03550
      CALL ALI(BLMBDA,RLBDA,GLBDA,GLMBDA,21)                            SID03560
      VISCK = 1.09E-05                                                  SID03570
      HEAT=-SK*(TUPPER-TEQ)                                             SID03580
      PM = VISC*86400.0*32.2/(ABS(HEAT)/(TUPPER-TEQ)*SAH(JJ))           SID03590
      GR = 32.2*BETA*(TUPPER-TEQ)*SAH(JJ)**3/VISCK**2                   SID03600
      XLES = GLMBDA*GR**0.35*PM**0.54*SAH(JJ)                           SID03610
C                                                                       SID03620
C   THIS CALCULATION IS FOR A LAMBDA OF 0.65                            SID03630
C                                                                       SID03640
      XXX=0.986*GR**0.35*PM**0.54*SAH(JJ)                               SID03650
      XXLS=SAL(JJ)/XXX                                                  SID03660
      CALL ALI(XXLS,XLPS,QPS,QX,11)                                     SID03670
      QX=QX*SAW(JJ)*XXX*SK/(62.4*86400.)                                SID03680
      XLS = SAL(JJ)/XLES                                                SID03690
      CALL ALI(XLS,XLPS,QPS,QOS,11)                                     SID03700
      CALL ALI(XLS,XLPS,TPS,TFS,11)                                     SID03710
      QOS = QOS*SK*XLES/(62.4*86400.0)                                  SID03720
      TFS = TFS*(TUPPER-TEQ)+TEQ                                        SID03730
      IF(TFS.LT.39.0)GO TO 801                                          SID03740
      GO TO 778                                                         SID03750
  777 TFS=TEQ                                                           SID03760
      QOS=0.0                                                           SID03770
  778 IF(IT.EQ.0)GTS = GTS+(TFS-TLOWER)**2                              SID03780
      QOS=QOS*SAW(JJ)                                                   SID03790
      IF(IT.EQ.0)GQS = GQS+(QOS-QUPPER)**2                              SID03800
      GO TO 602                                                         SID03810
  801 TFS=(39.0-TEQ)/(TUPPER-TEQ)                                       SID03820
      CALL ALI2(XLS,XLPS,TPS,TFS,11)                                    SID03830
      CALL ALI(XLS,XLPS,QPS,QOS,11)                                     SID03840
      TFS=39.0                                                          SID03850
```

```
      GO TO 778                                                          SID03860
C                                                                        SID03870
C     HEAD LOSS AT CONSTRICTION BASED ON STURM'S MODEL                   SID03880
C                                                                        SID03890
  602 IF((TUPPER-TEQ).LE.0.0)GO TO 669                                   SID03900
      IF(NBUG5.EQ.5)GO TO 6502                                           SID03910
      H001=H1                                                            SID03920
      LSA=SAL(JJ)                                                        SID03930
      HMAX=SAH(JJ)                                                       SID03940
      B=(SKK**2*LSA**2)/(BETA*(TUPPER-TEQ)*32.2*HMAX**3)                 SID03950
      A=SAWC(JJ)/SAW(JJ)                                                 SID03960
      HF11=1.0/H1                                                        SID03970
      HF22=1.0/(1.0-H1)                                                  SID03980
      IF(ISM.EQ.1)GO TO 802                                              SID03990
      MAD=0                                                              SID04000
  666 N=0                                                                SID04010
      IF(NBUG1.EQ.1)WRITE(7,2017)BETA,LSA,HMAX,A,H001                    SID04020
 2017 FORMAT(1X,'BETA,LSA,H,A,H1:',5(1X,E9.3))                           SID04030
  600 N=N+1                                                              SID04040
C     THIS PROCEDURE IS  STEFFENSEN'S  SECOND ORDER METHOD FOR NON-      SID04050
C     LINEAR EQUATIONS                                                   SID04060
      DO 68 NJ=1,2                                                       SID04070
         BLMBDA=1.0-H001                                                 SID04080
         CALL ALI(BLMBDA,RLBDA,GLBDA,GLMBDA,21)                          SID04090
         XL(NJ) = GLMBDA*GR**0.35*PM**0.54*SAH(JJ)                       SID04100
         XLS = SAL(JJ)/XL(NJ)                                            SID04110
         CALL ALI(XLS,XLPS,QPS,Q,11)                                     SID04120
         CALL ALI(XLS,XLPS,TPS,T1,11)                                    SID04130
         Q1(NJ) = Q*SK*XL(NJ)/(62.4*86400.0)                             SID04140
         THS(NJ)=T1*(TUPPER-TEQ)+TEQ                                     SID04150
         IF(THS(NJ).LT.39.0)GO TO 601                                    SID04160
         GO TO 678                                                       SID04170
601      T=(39.0-TEQ)/(TUPPER-TEQ)                                       SID04180
         CALL ALI2(XLS,XLPS,TPS,T,11)                                    SID04190
         CALL ALI(XLS,XLPS,QPS,Q1(NJ),11)                                SID04200
         THS(NJ)=39.0                                                    SID04210
C                                                                        SID04220
C        CALCULATION OF THE CONTRACTIONS EFFECT ON HO                    SID04230
C                                                                        SID04240
678      G(NJ)=H1-H001-((((Q1(NJ)**2)*B)/(2.*(SKK**2)*(LSA**2)))*(HF11   SID04250
     &   **2+HF22**2)*(((1./A)**2)-1.))                                  SID04260
         IF(NBUG1.EQ.1)WRITE(7,2018)Q1(NJ),G(NJ),H001                    SID04270
 2018    FORMAT(1X,'Q,G,H001:',3(1X,F8.5))                               SID04280
         IF(NJ.EQ.2)GO TO 69                                             SID04290
         H001=G(NJ)+H001                                                 SID04300
         IF(H001.LE.0.0)H001=0.001*FLOAT(N)                              SID04310
   68 IF(H001.GE.1.0)H001=1.0-(0.001*FLOAT(N))                           SID04320
   69 IF(G(1).EQ.0.0)GO TO 6001                                          SID04330
      IF((G(2)-G(1)).EQ.0.0)GO TO 6003                                   SID04340
      GG=(G(2)-G(1))/G(1)                                                SID04350
      IF(N.GT.20)GO TO 663                                               SID04360
      H001=(H001-G(1))-(G(1)/GG)                                         SID04370
      IF(ABS(G(1)/GG).GT.0.013)GO TO 600                                 SID04380
      GO TO 6002                                                         SID04390
 6001 H001=H001-G(1)-0.001                                               SID04400
```

```
      GO TO 600                                                          SID04410
 6003 H001=0.001*FLOAT(N)                                                SID04420
      GO TO 600                                                          SID04430
 6002 QHS=Q1(1)                                                          SID04440
      TSH=THS(1)                                                         SID04450
      NGSH=1+NGSH                                                        SID04460
      GO TO 680                                                          SID04470
  669 TSH=TEQ                                                            SID04480
      H001=0.0                                                           SID04490
      QHS=0.0                                                            SID04500
  680 IF(IT.EQ.0)GTHS=GTHS+(TSH-TLOWER)**2                               SID04510
      QHS=QHS*SAW(JJ)                                                    SID04520
      IF(IT.EQ.0)GQHS=GQHS+(QHS-QUPPER)**2                               SID04530
      GO TO 802                                                          SID04540
  663 MAD=MAD+1                                                          SID04550
      H001=((H001-G(1))-(G(1)/GG)+H001-G(1))/2.                          SID04560
      IF(MAD.LE.3)GO TO 666                                              SID04570
      QHS=Q1(1)*SAW(JJ)                                                  SID04580
      TSH=THS(1)                                                         SID04590
      WRITE(7,5003)II                                                    SID04600
 5003 FORMAT(1X,'STURMS MODEL DID NOT CONVERGE ON DAY',I5)               SID04610
C                                                                        SID04620
C   HEAD LOSS AT CONSTRICTION FORMULATION(BASED ON BROCARD'S MODEL       SID04630
C                                                                        SID04640
  802 IF(IBHM.EQ.1)GO TO 994                                             SID04650
      IF((TUPPER-TEQ).LE.0.0)GO TO 779                                   SID04660
      H00=H1                                                             SID04670
      MAD=0                                                              SID04680
  667 N=0                                                                SID04690
      IF(NBUG1.EQ.1)WRITE(7,2007)BETA,B,LSA,HMAX,A,H00                   SID04700
 2007 FORMAT(1X,'BETA,B,LSA,H,A,H1:',6(1X,E9.3))                         SID04710
  700 N=N+1                                                              SID04720
C                                                                        SID04730
C     THIS PROCEDURE IS STEFFENSEN'S SECOND ORDER METHOD FOR NON-        SID04740
C     LINEAR EQUATIONS                                                   SID04750
C                                                                        SID04760
      DO 78 NJ=1,2                                                       SID04770
      HF1 = 1.0 / H00                                                    SID04780
      HF2 = 1.0 / (1.0-H00)                                              SID04790
      BFH(NJ) = B*FO*LSA / (HMAX*4.0*H00)                                SID04800
      BFH(NJ) = BFH(NJ) * (ALPHA*(HF1+HF2)**3 + HF2**3)                  SID04810
      OMEGA(NJ)=BFH(NJ)**(-0.333)                                        SID04820
      MM=0                                                               SID04830
      TB1=TSTAR                                                          SID04840
  698 CONTINUE                                                           SID04850
      MM=MM+1                                                            SID04860
      TB=(TB1-TEQ)/(TUPPER-TEQ)                                          SID04870
      Q1(NJ)=OMEGA(NJ)*LSA*SKK*TB**0.3333                                SID04880
      IF(NBUG3.NE.3)CALL TRED(BETA,TSTAR,TUPPER,HO,Q1(NJ),SAL(JJ),TRF)   SID04890
      TBB=EXP(-TRF*SKK*SAL(JJ)/Q1(NJ))                                   SID04900
      TB1=TBB*(TUPPER-TEQ)+TEQ                                           SID04910
      TFH(NJ)=TB1                                                        SID04920
      IF(MM.GT.35)GO TO 5005                                             SID04930
      TRFH=TRF                                                           SID04940
      IF(TBB.LT.0.0001)GO TO 5009                                        SID04950
```

```
      IF(NBUG4.GT.4)WRITE(7,5900)TB,TBB,Q1(NJ)                          SID04960
      IF(ABS(TBB-TB).GT.0.01)GO TO 698                                  SID04970
      GO TO 5009                                                        SID04980
 5005 WRITE(7,5002)II                                                   SID04990
C                                                                       SID05000
C     CALCULATION OF THE CONTRACTION'S EFFECT ON HO                     SID05010
C                                                                       SID05020
 5009 G(NJ)=H1-H00-((((Q1(NJ)**2)*B)/(2.*(SKK**2)*(LSA**2)))*(HF11**2+HFSID05030
     *22**2)*(((1./A)**2)-1.))                                          SID05040
      IF(NBUG1.EQ.1)WRITE(7,2008)HF1,HF2,BFH(NJ),Q1(NJ),G(NJ),H00       SID05050
 2008 FORMAT(1X,'HF1,HF2,BF,Q,G,H00:',6(1X,F8.5))                       SID05060
      IF(NJ.EQ.2)GO TO 79                                               SID05070
      H00=G(NJ)+H00                                                     SID05080
      IF(H00.LE.0.0)H00=0.001*FLOAT(N)                                  SID05090
   78 IF(H00.GE.1.0)H00=1.0-(0.001*FLOAT(N))                            SID05100
   79 GG=(G(2)-G(1))/G(1)                                               SID05110
      H00=(H00-G(1))-(G(1)/GG)                                          SID05120
      IF(H00.GE.1.0)H00=1.0-(0.001*FLOAT(N))                            SID05130
      IF(H00.LE.0.0)H00=0.001*FLOAT(N)                                  SID05140
      IF(N.GT.20)GO TO 993                                              SID05150
      IF(ABS(G(1)/GG).GT.0.0115)GO TO 700                               SID05160
      QH=Q1(1)                                                          SID05170
      NGH=1+NGH                                                         SID05180
      TRFH=TRF                                                          SID05190
      TBH=TFH(1)                                                        SID05200
      IF(TBH.LT.39.0)TBH=39.0                                           SID05210
      GO TO 780                                                         SID05220
  779 TBH=TEQ                                                           SID05230
      H00=0.0                                                           SID05240
      QH=0.0                                                            SID05250
  780 IF(IT.EQ.0)GTH=GTH+(TBH-TLOWER)**2                                SID05260
      QH=QH*SAW(JJ)                                                     SID05270
      IF(IT.EQ.0)GQH=GQH+(QH-QUPPER)**2                                 SID05280
      GO TO 994                                                         SID05290
  993 MAD=MAD+1                                                         SID05300
      H00=((H00-G(1))-(G(1)/GG)+H00-G(1))/2.                            SID05310
      IF(MAD.LE.3)GO TO 667                                             SID05320
      QH=Q1(1)*SAW(JJ)                                                  SID05330
      TBH=TFH(1)                                                        SID05340
      WRITE(7,6009)II                                                   SID05350
 6009 FORMAT(1X,'BROCARD HEAD LOSS DID NOT CONVERGE ON DAY',I5)         SID05360
C                                                                       SID05370
C     CALCULATION OF NORMALIZED(T*) TEMPERATURES                        SID05380
C                                                                       SID05390
  994 TDEN=TUPPER-TEQ                                                   SID05400
      TSD=(TLOWER-TEQ)/TDEN                                             SID05410
      TSBR=(TF(1)-TEQ)/TDEN                                             SID05420
      TSBH=(TBH-TEQ)/TDEN                                               SID05430
      TSC=(TFC-TEQ)/TDEN                                                SID05440
      TSS=(TFS-TEQ)/TDEN                                                SID05450
      TSSH=(TSH-TEQ)/TDEN                                               SID05460
      TSJ=(TFJ-TEQ)/TDEN                                                SID05470
      XD=999.99                                                         SID05480
C                                                                       SID05490
C     ( ) OF RESULTS                                                    SID05500
```

```
C                                                                       SID05510
 6502 IF(NBUG2.EQ.2)GO TO 4002                                          SID05520
      WRITE(IRITE,3000)NAME                                             SID05530
      WRITE(IRITE,3001)IMO(II),IDAY(II),IYR(II),IHOUR(II)               SID05540
C     WRITE(IRITE,3002)TAIR,SOLAR,HUM,XLWRI,W,XLWRO,CC,EVAP,SR,         SID05550
C    +       CONDUC,HEAT                                               SID05560
      WRITE(IRITE,3003)TEQ,SK                                           SID05570
      WRITE(IRITE,3004)HO,HOL,TS,TEND(II),QUPPER,QLOWER,TUPPER,TLOWER,  SID05580
     +                 VUPPER,VLOWER,FR1,FR2,FR,HOT                     SID05590
      WRITE(IRITE,3005)                                                 SID05600
      DO 300 J=1,NN                                                     SID05610
          J1 = J-1                                                      SID05620
          WRITE(IRITE,3006)J1,TEMP(II,J),VEL(II,J)                      SID05630
  300 CONTINUE                                                          SID05640
      WRITE(IRITE,3007)QO(1),QOC,TF(1),TFC,BF(1),BFC,BF2(1),BF2C,H1,CRATSID05650
                                                                        SID05660
     *IO,TRFB(1),TRFC                                                   SID05670
      WRITE(IRITE,3330)QO(2),QH,TF(2),TBH,BF(2),BFH(1),BF2(2),OMEGA(1),TSID05680
     *RFB(2),HOO,TRFH                                                   SID05690
      WRITE(IRITE,3008)QOJ,QOS,QX,TFJ,TFS,XLE,XLES,PM,GR                SID05700
      WRITE(IRITE,3331)QHS,TSH,XL(1),HOO1                               SID05710
 4002 HOO=HOO*SAH(JJ)                                                   SID05720
      HOO1=HOO1*SAH(JJ)                                                 SID05730
      IF(NBUG2.GE.1.AND.JJ.EQ.1)WRITE(15,5051)IMO(II),IDAY(II),IYR(II),TSID05740
     *S,TUPPER,TLOWER,TEQ,SK,QUPPER,QLOWER,HOT,HO,FR,FRD,XD             SID05750
      IF(NBUG2.GE.1)WRITE(16,5052)IMO(II),IDAY(II),IYR(II),QUPPER,TSD,  SID05760
     *QO(1),TSBR,QH,TSBH,HOO,QOC,TSC,QOS,TSS,QHS,TSSH,HOO1,QOJ,TSJ      SID05770
      WRITE(50,4001)DAY2,QUPPER,QLOWER,QO(1),QO(2),QOC,QH,QOJ,QOS,QHS,FRSID05780
     *,HO,HOO,QX                                                        SID05790
      WRITE(51,4003)DAY2,TLOWER,TUPPER,TEQ,TS,TF(1),TF(2),TFC,TBH,TFJ,TFSID05800
     *S,TSH,HOT,HOO1                                                    SID05810
 4001 FORMAT(F4.0,9F6.1,F6.3,F5.2,F5.2,F6.1)                            SID05820
 4003 FORMAT(F4.0,11F6.2,F5.2,F5.2)                                     SID05830
                                                                        SID05840
   70 CONTINUE                                                          SID05850
                                                                        SID05860
C     COMPUTE AND PRINT STATISTICS                                      SID05870
                                                                        SID05880
      IF(NGT.EQ.0)NGT=1                                                 SID05890
      IF(NGQ.EQ.0)NGQ=1                                                 SID05900
      IF(NGH.EQ.0)NGH=1                                                 SID05910
      IF(NGSH.EQ.0)NGSH=1                                               SID05920
      DO 14 J=1,2                                                       SID05930
      GTB(J) = SQRT(GTB(J)/FLOAT(NGT))                                  SID05940
   14 GQB(J) = SQRT(GQB(J)/FLOAT(NGQ))                                  SID05950
      GTC = SQRT(GTC/FLOAT(NGT))                                        SID05960
      GQC = SQRT(GQC/FLOAT(NGQ))                                        SID05970
      GTJ = SQRT(GTJ/FLOAT(NGT))                                        SID05980
      GQJ = SQRT(GQJ/FLOAT(NGQ))                                        SID05990
      GTS = SQRT(GTS/FLOAT(NGT))                                        SID06000
      GQS = SQRT(GQS/FLOAT(NGQ))                                        SID06010
      GQH=SQRT(GQH/FLOAT(NGH))                                          SID06020
      GTH=SQRT(GTH/FLOAT(NGH))                                          SID06030
      GQHS=SQRT(GQHS/FLOAT(NGSH))                                       SID06040
      GTHS=SQRT(GTHS/FLOAT(NGSH))                                       SID06050
      WRITE(IRITE,3020)NGT,NGQ,GTB(1),GQB(1),GTC,GQC,GTJ,GQJ,GTS,GQS    SID06060
```

```
      WRITE(IRITE,3049)GTB(2),GQB(2),GTH,GQH,GTHS,GQHS
3     CONTINUE
C ***********************************************************************
C *                    END OF MAIN LOOP                                 *
C ***********************************************************************
C +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
C +                    FORMAT STATEMENTS                                +
C +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 2001 FORMAT(6A4)
 3000 FORMAT('1FIELD DATA ANALYSIS FOR SIDE ARM ',6A4/)
 3001 FORMAT(5X,'DATE: ',I2,'/',I2,'/',I2,8X,'TIME: ',I4)
 3002 FORMAT(5X,'METEOROLOGICAL DATA:'/
     +       T10,'AIR TEMPERATURE',T38,F5.1,2X,'DEG F',
     +       T60,'NET SOLAR RADIATION',T88,F6.0,2X,'BTU/SQ FT-DAY'/
     +       T10,'RELATIVE HUMIDITY',T38,F5.1,2X,'%',
     +       T60,'INCOMING LONG WAVE',T88,F6.0,2X,'BTU/SQ FT-DAY'/
     +       T10,'WIND SPEED    ',T38,F5.1,2X,'MPH',
     +       T60,'OUTGOING LONG WAVE',T88,F6.0,2X,'BTU/SQ FT-DAY'/
     +       T10,'CLOUD COVER',T38,F5.2,
     +       T60,'EVAPORATIVE',T88,F6.0,2X,'BTU/SQ FT-DAY'/
     +       T10,'OBSERVED SOLAR',T38,F5.0,2X,'BTU/SQ FT-DAY',
     +       T60,'CONDUCTIVE',T88,F6.0,2X,'BTU/SQ FT-DAY'/
     +       T60,'NET HEAT INPUT',T88,F6.0,2X,'BTU/SQ FT-DAY')
 3003 FORMAT(T10,'EQUILIBRIUM TEMP',T38,F5.1,2X,'DEG F',
     +       T60,'HEAT EXCHANGE COEF',T88,F6.1,2X,'BTU/SQ FT-DAY',
     +       '-DEG F')
 3004 FORMAT(/,5X,'SIDE ARM OBSERVATIONS:-',/,
     +       T10,'UPPER LAYER DEPTH',T38,F5.2,2X,'FEET',
     +       T60,'LOWER LAYER DEPTH',T89,F5.2,2X,'FEET'/,
     +       T10,'SURFACE TEMPERATURE',T38,F5.1,2X,'DEG F',
     +       T60,'END SURFACE TEMPERATURE',T89,F5.1,2X,'DEG F'/,
     +       T10,'UPPER LAYER FLOW',T38,F5.2,2X,'CFS/FT',
     +       T60,'LOWER LAYER FLOW',T89,F5.2,2X,'CFS/FT'/,
     +       T10,'AVERAGE UPPER LAYER TEMP.',T38,F5.1,2X,'DEG F',
     +       T60,'AVERAGE LOWER LAYER TEMP.',T89,F5.1,2X,'DEG F'/,
     +       T10,'UPPER AVERAGE VELOCITY',T38,F5.2,2X,'FPS',
     +       T60,'LOWER AVERAGE VELOCITY',T89,F5.2,2X,'FPS'/,
     +       T10,'UPPER LAYER FROUDE NUMBER',T38,F5.2,
     +       T60,'LOWER LAYER FROUDE NUMBER',T89,F5.2/,
     +       T10,'ENTRANCE FROUDE NUMBER',T38,F5.2,T60,'THERMOCLINE ',
     +       'DEPTH',T89,F5.2,2X,'FEET'/)
 3005 FORMAT(5X,'TEMPERATURE-VELOCITY PROFILE:'/,
     +       T15,'DEPTH',T30,'TEMPERATURE',T45,'VELOCITY'/,
     +       T14,'(METER)',T30,'(DEGREES F)',T46,'(FPS)')
 3006 FORMAT(T16,I2,T32,F5.1,T45,F6.2)
 3007 FORMAT(/5X,'SIDE ARM PREDICTIONS:'/,
     +       T7,'BROCARD MODEL:',T57,'CRITICAL FLOW MODEL:'/
     +       T10,'UPPER LAYER FLOW',T37,F7.2,2X,'CFS/FT',
     +       T60,'UPPER LAYER FLOW',T87,F7.2,2X,'CFS/FT'/
     +       T10,'END TEMPERATURE',T37,F7.1,2X,'DEGREES F',
     +       T60,'END TEMPERATURE',T87,F7.1,2X,'DEGREES F'/
```

```
     +       T10,'BF PARAMETER',T37,F7.3,                                    SID06610
     +       T60,'BF PARAMETER',T87,F7.3/                                    SID06620
     +       T10,'NON-DIMENSIONAL FLOW',T37,F7.3,                            SID06630
     +       T60,'NON-DIMENSIONAL FLOW',T87,F7.3/,                           SID06640
     +       T10,'MIXED LAYER DEPTH',T37,F7.3,                               SID06650
     +       T60,'CONSTRICTION RATIO',T87,F7.2/T10,'TRF',T37,F7.4,T60,       SID06660
     *       'TRF',T87,F7.4)                                                 SID06670
 3008 FORMAT(T7,'JAIN MODEL:',T57,'STURM MODEL:'/                            SID06680
     +       T10,'UPPER LAYER FLOW',T37,F7.2,2X,'CFS/FT',                    SID06690
     +       T60,'UPPER LAYER FLOW',T87,F7.2,2X,'CFS/FT',2X,'(FOR HO=',      SID06700
     +       '0.35)Q=',F7.2,1X,'CFS'/T10,'END TEMPERATURE',T37,F7.1,2X,      SID06710
     +       'DEGREES F',T60,'END TEMPERATURE',T87,F7.1,2X,'DEGREES F'/       SID06720
     +       T10,'EQUILIBRIUM LENGTH',T35,F9.0,2X,'FEET',                    SID06730
     +       T60,'EQUILIBRIUM LENGTH',T85,F9.0,2X,'FEET'/                    SID06740
     +       T60,'MODIFIED PRANDTL NO.',T87,F7.4/                            SID06750
     +       T60,'GRASHOF NUMBER',T87,E11.4)                                 SID06760
 3330 FORMAT(T7,'BROCARD WITH UNEQUAL FLOWS:',T57,'BROCARD WITH HEAD LOSS     SID06770
    *S:'/T10,'UPPER LAYER FLOW',T37,F7.2,2X,'CFS/FT',T60,'UPPER LAYER        SID06780
     * FLOW',T87,F7.2,2X,'CFS/FT'/T10,'END TEMPERATURE',T37,F7.1,             SID06790
     *2X,'DEGREES F',T60,'END TEMPERATURE',T87,F7.1,2X,'DEGREES F'           SID06800
     */T10,'BF PARAMETER',T37,F7.3,T60,'BF PARAMETER',T87,F7.3/              SID06810
     *T10,'NON-DIMENSIONAL FLOW',T37,F7.3,                                   SID06820
     *T60,'NON-DIMENSIONAL FLOW',T87,F7.3/T10,'TRF',T37,F7.4,                SID06830
     *T60,'MIXED LAYER DEPTH',T87,F7.3/T60,'TRF',T87,F7.4)                   SID06840
 3331 FORMAT(T7,'STURM WITH HEAD LOSS:'/T10,'UPPER LAYER FLOW',              SID06850
     *T37,F7.2,2X,'CFS/FT'/T10,'END TEMPERATURE',T37,F7.1,2X,'DEGREES F'     SID06860
     */T10,'EQUILIBRIUM LENGTH',T37,F9.0,2X,'FEET'/                          SID06870
     *T10,'MIXED LAYER DEPTH',T37,F7.3)                                      SID06880
 3020 FORMAT('1ERROR ANALYSIS OF PREDICTIONS'/' ',29('=')//                  SID06890
     +       T10,'NUMBER OF TEMPERATURE COMPARISONS',T45,I4/                 SID06900
     +       T10,'NUMBER OF FLOW COMPARISONS',T45,I4//                       SID06910
     +       T10,'BROCARD MODEL:'/                                           SID06920
     +       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID06930
     +       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'//                 SID06940
     +       T10,'CRITICAL FLOW MODEL:'/                                     SID06950
     +       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID06960
     +       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'//                 SID06970
     +       T10,'JAIN MODEL:'/                                              SID06980
     +       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID06990
     +       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'//                 SID07000
     +       T10,'STURM MODEL:'/                                             SID07010
     +       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID07020
     +       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'/)                 SID07030
 3049 FORMAT(T10,'BROCARD WITH UNEQUAL FLOWS:'/                              SID07040
     *       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID07050
     *       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'//                 SID07060
     *       T10,'BROCARD WITH HEAD LOSS:'/                                  SID07070
     *       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID07080
     *       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS'//                 SID07090
     *       T10,'STURM WITH HEAD LOSS:'/                                    SID07100
     *       15X,'R.M.S. ERROR FOR TEMPERATURE',T50,F6.2,2X,'DEG F'/         SID07110
     *       15X,'R.M.S. ERROR FOR FLOW',T50,F6.1,2X,'CFS')                  SID07120
 3050 FORMAT('1 COOLING LAKE SIDE ARM ANALYSIS PROGRAM'/                     SID07130
     +       2X,38('=')//                                                    SID07140
     +       5X,'RUN OF ',A8,5X,'AT ',A8//)                                  SID07150
```

```
 3051 FORMAT(5X,'SUMMARY OF INPUT PARAMETERS'/5X,27('-')//                SID07160
     +       10X,'SIDE ARM NAME',T45,4A4/                                 SID07170
     +       10X,'SIDE ARM LENGTH',T45,F8.0,T55,'(FEET)'/                 SID07180
     +       10X,'SIDE ARM ENTRANCE DEPTH',T45,F8.1,T55,'(FEET)'/         SID07190
     +       10X,'SIDE ARM ENTRANCE WIDTH',T45,F8.1,T55,'(FEET)'/         SID07200
     +       10X,'BRIDGE CONSTRICTION WIDTH',T45,F8.1,T55,'(FEET)'/       SID07210
     +       10X,'BRIDGE CONSTRICTION DEPTH',T45,F8.2,T55,'(FEET)'/       SID07220
     +       10X,'BOTTOM FRICTION FACTOR',T45,F8.2/                       SID07230
     +       10X,'INTERFACIAL FRICTION COEFFICIENT',T45,F8.2/             SID07240
     +       10X,'TEMPERATURE DECAY FACTOR',T45,F8.3/10X,'ACTUAL TOTAL CSID07250
     *ROSS-SECTIONAL AREA',T45,F8.2,2X,'FT*FT'/10X,'ACTUAL AREA OF CONSTSID07260
     *RICTION',T45,F8.2,2X,'FT*FT'/10X,'SURFACE AREA',T45,F10.1,3X      SID07270
     *,'FT*FT'/10X,'VOLUME',T45,F11.1,2X,'FT*FT*FT'/10X,'TOP WIDTH AT COSID07280
     *NSTRICTION',T45,F8.2,2X,'FT'/10X,'ACTIVE VOLUME',T45,F11.1,2X,    SID07290
     *'FT*FT*FT')                                                        SID07300
 5050 FORMAT(1X,'SIDEARM DATA ANALYSIS :'/2X,                            SID07310
     *'DATE',7X,'TS',7X,'TU',7X,'TL',6X,'TEQ',8X,'K',9X,'QU',9X,        SID07320
     *'QL',8X,'HT',7X,'HV',4X,'FFR',4X,'FFR',6X,'XD'/                   SID07330
     *12X,'(F)',6X,'(F)',6X,'(F)',6X,'(F)',4X,'(BTU/FT*',5X,'(CFS)',6X, SID07340
     *'(CFS)',5X,'(FT)',5X,'(FT)',8X,'(MEAN)',4X,'(FT)'/                SID07350
     *47X,'FT/DAY)'/121('-'))                                           SID07360
 5051 FORMAT(1X,I2,'/',I2,'/',I2,1X,F6.1,3X,F6.1,3X,F6.1,3X,F6.1,3X,F7.1SID07370
     *,3X,F8.1,3X,F8.1,3X,F6.1,3X,F6.1,1X,F6.2,1X,F6.2,3X,F7.1)         SID07380
 5053 FORMAT(1X,'SIDEARM MODEL COMPARISONS:'/9X,':',6X,'DATA',9X,':',   SID07390
     *24X,'MODEL PREDICTIONS',50X,':'/9X,':',6X,'****',9X,':',24X,      SID07400
     *17('*'),50X,':'/2X,'DATE',3X,':',1X,'FLOW',3X,'NORMALIZED',      SID07410
     *1X,':',3X,'QB',3X,'T*B',5X,'QBH',3X,'T*BH',3X,'HB',5X,'QC',3X,    SID07420
     *'T*C',5X,'QS',3X,'T*S',5X,'QSH',3X,'T*SH',3X,'HS',5X,'QJ',3X,     SID07430
     *'T*J',1X,':'/                                                     SID07440
     *9X,':',1X,'(CFS)',5X,'TEMP',4X,':',1X,'(CFS)',8X,'(CFS)',9X,'(FT)'SID07450
     *,2X,'(CFS)',8X,'(CFS)',8X,'(CFS)',9X,'(FT)',2X,'(CFS)',6X,        SID07460
     *':'/121('-'))                                                     SID07470
 5052 FORMAT(I2,'/',I2,'/',I2,2X,F6.1,4X,F6.2,4X,F6.1,2X,F4.2,2X,F6.1,2XSID07480
     *,F4.2,2X,F4.1,2X,F6.1,1X,F4.2,2X,F6.1,1X,F4.2,                    SID07490
     *2X,F6.1,2X,F4.2,2X,F4.1,2X,F6.1,1X,F4.2)                          SID07500
      STOP                                                              SID07510
      END                                                               SID07520
                                                                        SID07530
                                                                        SID07540
                                                                        SID07550
      SUBROUTINE TRANS(N,ID,IM,IY,FTEQ,FK,TEMP,NTEMP,VOLSA,HBAR1,HBAR3,SSID07560
     *AH,SAW,LSA,VOL,IBUG)                                              SID07570
      REAL LSA                                                          SID07580
      DIMENSION ID(100),IM(100),IY(100),FTEQ(100),FK(100),TEMP(100,20), SID07590
     *NTEMP(100),TAIRA(1736),HUMA(1736),WA(1736),CCA(1736),SRA(1736),    SID07600
     *IDAY(100),TE(1736),SK(1736),TDAY(100)                             SID07610
      READ(40,100)TAIRA                                                 SID07620
      READ(41,100)HUMA                                                  SID07630
      READ(42,100)WA                                                    SID07640
      READ(43,100)CCA                                                   SID07650
      READ(44,100)SRA                                                   SID07660
                                                                        SID07670
      CALL DDAY(7,26,78,D)                                              SID07680
      NNN=0                                                             SID07690
      DO 10 J=1,N                                                       SID07700
```

```
        CALL DDAY(IM(J),ID(J),IY(J),DAY)                                  SID07710
        TDAY(J)=DAY                                                       SID07720
        IF(IY(J).GT.78)NNN=LEAP(IY(J)-1)                                  SID07730
        IDAY(J)=INT((DAY-D+1.0)+FLOAT(IY(J)-78)*365.0+FLOAT(NNN))         SID07740
        IF(IBUG.EQ.6.AND.J.LT.25)WRITE(7,141)IM(J),ID(J),IY(J),DAY,NNN    SID07750
     &  ,IDAY(J)                                                          SID07760
  141   FORMAT(1X,I2,'/',I2,'/',I2,2X,F5.1,1X,I2,1X,I5)                   SID07770
   10 CONTINUE                                                            SID07780
  100 FORMAT(8F10.2)                                                      SID07790
C                                                                         SID07800
C     DETERMINE RESIDENCE TIME FOR SIDEARM FOR FLOW OF 400 CFS            SID07810
C                                                                         SID07820
      VOL=VOLSA-((1./3.)*SAW*LSA*HBAR3**3/SAH**2)-(SAH*SAW*LSA*((1./3.    SID07830
     *)-(0.5*(1.-HBAR1/SAH))+((1./6.)*(1.-HBAR1/SAH)**3)))                SID07840
      TRES=(VOL/2./400./86400.)                                           SID07850
      IT=INT(TRES+0.5)                                                    SID07860
      IF(IBUG.GT.5)WRITE(7,134)VOL,TRES,IT                                SID07870
  134 FORMAT(1X,'VOL,TRES,IT:',E10.4,1X,E10.4,1X,I5)                      SID07880
                                                                          SID07890
      DO 24 J=1,N                                                         SID07900
      TS=TEMP(J,1)                                                        SID07910
      IF(NTEMP(J).EQ.2)TS=TEMP(J,2)                                       SID07920
      IA=IDAY(J)-IT+1                                                     SID07930
      IF(IA.LT.1)IA=1                                                     SID07940
      X=0.                                                                SID07950
      KDAY=IDAY(J)                                                        SID07960
                                                                          SID07970
      DO 25 JJ=IA,KDAY                                                    SID07980
      SR=SRA(JJ)                                                          SID07990
      W=WA(JJ)                                                            SID08000
      CC=CCA(JJ)                                                          SID08010
      RH=HUMA(JJ)                                                         SID08020
      TAIR=TAIRA(JJ)                                                      SID08030
      IF(IBUG.EQ.6.AND.J.LT.5)WRITE(7,140)TS,TAIR,RH,W,CC,SR              SID08040
  140 FORMAT(1X,'TS,TA,RH,W,CC,SR:',6(F8.2,2X))                           SID08050
      SOLAR = SR*0.94                                                     SID08060
C     IDSO AND JACKSON'S FORMULA .....                                    SID08070
      XLWRI=(0.97*4.15E-8)*((TAIR+460.)**4)*(1.0-(0.261*EXP(-2.4E-4*(TAI  SID08080
     *R-32.)**2)))*(1.0+0.17*CC*CC)                                       SID08090
C     SWINBANK'S FORMULA.....                                             SID08100
C     XLWRI = (1.16E-13)*(460.0+TAIR)**6*(1.0+0.17*CC**2)                 SID08110
      EA = RH*EE(TAIR)                                                    SID08120
      DAY=TDAY(J)-FLOAT(IT+JJ-IA)                                         SID08130
      CALL HEATO(TS,TAIR,W,WFF,EVAP,CONDUC,XLWRO,DAY,EA)                  SID08140
      HEAT = SOLAR+XLWRI-XLWRO-CONDUC-EVAP                                SID08150
C                                                                         SID08160
C     DETERMINE EQUILIBRIUM TEMPERATURE                                   SID08170
C                                                                         SID08180
      CALL EQUILO(TAIR,W,WFF,XLWRI,SOLAR,TE(JJ),DAY,EA)                   SID08190
      SK(JJ)=-HEAT/(TS-TE(JJ))                                            SID08200
      IF(SK(JJ).LE.0.0)SK(JJ)=0.0                                         SID08210
      X=SK(JJ)+X                                                          SID08220
   25 CONTINUE                                                            SID08230
C                                                                         SID08240
C     EXPONENTIAL FILTERING OF TE AND K                                   SID08250
```

```
C                                                                       SID08260
      YK=X/FLOAT(IDAY(J)-IA+1)                                          SID08270
      DT=1.0                                                            SID08280
      IF(IBUG.GT.5)WRITE(7,133)YK,(TE(KK),KK=IA,KDAY),(SK(KK),KK=IA,K   SID08290
     *DAY)                                                              SID08300
  133 FORMAT(1X,'FT:KBAR,TE,K:',11(F5.1,1X)/1X,10(F5.1,1X))             SID08310
      X=0.                                                              SID08320
      Z=0.                                                              SID08330
      DO 26 NNN=IA,KDAY                                                 SID08340
      XX=TE(NNN)                                                        SID08350
      Y=EXP(-(DT*YK*FLOAT(IDAY(J)-NNN))/62.4/HBAR1)                     SID08360
      X=(XX*Y)+X                                                        SID08370
      Z=Y+Z                                                             SID08380
   26 CONTINUE                                                          SID08390
      FTEQ(J)=X/Z                                                       SID08400
      X=0.                                                              SID08410
      Z=0.                                                              SID08420
      DO 27 NNN=IA,KDAY                                                 SID08430
      XX=SK(NNN)                                                        SID08440
      Y=EXP(-(DT*YK*FLOAT(IDAY(J)-NNN))/62.4/HBAR1)                     SID08450
      X=(XX*Y)+X                                                        SID08460
      Z=Y+Z                                                             SID08470
   27 CONTINUE                                                          SID08480
      FK(J)=X/Z                                                         SID08490
                                                                        SID08500
   24 CONTINUE                                                          SID08510
      RETURN                                                            SID08520
      END                                                               SID08530
                                                                        SID08540
                                                                        SID08550
      SUBROUTINE THERMO (T,I ,H,HO)                                     SID08560
      DIMENSION T(100,20) ,DT(20)                                       SID08570
      INTEGER P                                                         SID08580
C                                                                       SID08590
C     TO LOCATE THE THERMOCLINE TO THE NEAREST METER WITHIN             SID08600
C     THREE METERS OF HO                                                SID08610
C                                                                       SID08620
      DTMAX=0                                                           SID08630
      XLB=HO+3.0                                                        SID08640
      XUB=HO-3.0                                                        SID08650
      NLB=INT(XLB+0.5)                                                  SID08660
      NUB=INT(XUB-0.5)                                                  SID08670
      IF (NUB.LE.0) NUB=1                                               SID08680
      DO 20 N=NUB,NLB                                                   SID08690
      DT(N)=ABS(T(I,N+1)-T(I,N))                                        SID08700
      IF (DT(N)-DTMAX) 20,10,30                                         SID08710
C                                                                       SID08720
C     TO CHECK IF THE TWO SEGMENTS HAVING THE SAME SLOPE                SID08730
C     ARE ADJACENT TO EACH OTHER                                        SID08740
C                                                                       SID08750
   10 IF (N-P.EQ.2) M=N                                                 SID08760
      GO TO 20                                                          SID08770
   30 DTMAX=DT(N)                                                       SID08780
      P=                                                                SID08790
   20 E                                                                 SID08800
```

```
C                                                                       SID08810
C     IF THERE ARE ONLY TWO DATA POINTS IN THE TEMPERATURE              SID08820
C     PROFILE, THERMOCLINE CANNOT BE DETERMINED. ASSIGN H=HO            SID08830
C                                                                       SID08840
      IF (DTMAX.GE.10.0) GO TO 120                                      SID08850
C                                                                       SID08860
C     IF THREE POINTS ARE ON THE SAME LINE                              SID08870
C                                                                       SID08880
      IF (P+2.EQ.M) GO TO 40                                            SID08890
C                                                                       SID08900
C     IF LESS THAN THREE POINTS LIE ON A STRAIGHT LINE                  SID08910
C     FIT A THIRD DEGREE POLYNOMIAL EQUATION USING                      SID08920
C     FOUR POINTS AROUND THE INFLECTION POINT                           SID08930
C                                                                       SID08940
C     Y1=T(I,P)*(X-P-1)*(X-P-2)*(X-P-3)/(-6)                            SID08950
C     Y2=T(I,P+1)*(X-P)*(X-P-2)*(X-P-3)/2                               SID08960
C     Y3=T(I,P+2)*(X-P)*(X-P-1)*(X-P-3)/(-2)                            SID08970
C     Y4=T(I,P+3)*(X-P)*(X-P-1)*(X-P-2)/6                               SID08980
C     Y=Y1+Y2+Y3+Y4                                                     SID08990
C                                                                       SID09000
C     AFTER COLLECTING TERMS, THE THIRD DEGREE EQUATION HAS THE         SID09010
C     FORM OF Y=A*X**3+B*X**2+C*X                                       SID09020
C                                                                       SID09030
C     IF THE FOUR POINTS FORM A MONOTONOUSLY DECREASING OR INCREASING   SID09040
C     CURVE, ASSIGN H TO BE EQUAL TO THE LOWER BOUND OR UPPER BOUND     SID09050
C     VALUE                                                             SID09060
C                                                                       SID09070
      IF ((NLB-P).EQ.1) GO TO 70                                        SID09080
      IF ((NUB-P).EQ.1) GO TO 80                                        SID09090
      GO TO 110                                                         SID09100
   70 DT(NLB+1)=ABS(T(I,NLB+2)-T(I,NLB+1))                              SID09110
      IF (DT(NLB+1).GE.DTMAX) GO TO 90                                  SID09120
      GO TO 110                                                         SID09130
   80 DT(NUB-1)=ABS(T(I,NUB)-T(I,NUB-1))                                SID09140
      IF(DT(NUB-1).GE.DTMAX) GO TO 100                                  SID09150
      GO TO 110                                                         SID09160
C                                                                       SID09170
  110 A=T(I,P)/(-6)+T(I,P+1)/2-T(I,P+2)/2+T(I,P+3)/6                    SID09180
      B=T(I,P)*(3*P+6)/6-T(I,P+1)*(3*P+5)/2+T(I,P+2)*(3*P+4)/2-         SID09190
     *T(I,P+3)*(3*P+3)/6                                                SID09200
C                                                                       SID09210
C     TO FIND THE INFLECTION POINT                                      SID09220
C     THE SECOND DERIVATIVE OF Y W.R.T. X IS 6AX+2B                     SID09230
C     SET 6AX+2B=0, WE HAVE X=-B/(3*A)                                  SID09240
C                                                                       SID09250
      X=B/A/(-3.0)                                                      SID09260
      H=X-1.0                                                           SID09270
      GO TO 60                                                          SID09280
   40 H=FLOAT(M-1)                                                      SID09290
      GO TO 60                                                          SID09300
   90 H=FLOAT(NLB-1)                                                    SID09310
      GO TO 60                                                          SID09320
  100 H=FLOAT(NUB-1)                                                    SID09330
      GO TO 60                                                          SID09340
  120 H=HO                                                              SID09350
```

```
      GO TO 60                                                          SID09360
   60 WRITE (6,1000) I,H                                                SID09370
 1000 FORMAT (/,10X,'DAY',I2,8X,'DEPTH OF THERMOCLINE',2X,F6.2,'M')     SID09380
      RETURN                                                            SID09390
      END                                                               SID09400
                                                                        SID09410
                                                                        SID09420
                                                                        SID09430
      FUNCTION CTOF(T)                                                  SID09440
C                                                                       SID09450
C     CENTIGRADE TO FAHRENHEIT TEMPERATURE CONVERSION                   SID09460
C                                                                       SID09470
      CTOF = T*1.8 + 32.0                                               SID09480
      IF(CTOF.EQ.32.0)CTOF=0.0                                          SID09490
      RETURN                                                            SID09500
      END                                                               SID09510
      FUNCTION EE(T)                                                    SID09520
C                                                                       SID09530
C     VAPOR PRESSURE FUNCTION                                           SID09540
C                                                                       SID09550
      EE = 25.4*EXP(17.62-9500.8/(T+460.0))                             SID09560
      RETURN                                                            SID09570
      END                                                               SID09580
                                                                        SID09590
                                                                        SID09600
                                                                        SID09610
      SUBROUTINE EQUILO(TAIR,W,WFF,XLWRI,SOLAR,TEQ,DAY,EA)              SID09620
C                                                                       SID09630
C     SUBROUTINE TO COMPUTE EQUILIBRIUM TEMPERATURE .                   SID09640
C     USES RYANS WIND FUNCTION, ENGLISH UNITS                           SID09650
C                                                                       SID09660
      COMMON /IOPUT/ IREAD, IRITE                                       SID09670
C                                                                       SID09680
      DIMENSION UTEST(100),HNET(100)                                    SID09690
C                                                                       SID09700
C     PRAIR = 760.0                                                     SID09710
C     IT = 0                                                            SID09720
C     I1 = 1                                                            SID09730
C     I2 = 3                                                            SID09740
C     UTEST(1) = TAIR-5.0                                               SID09750
C     UTEST(2) = TAIR                                                   SID09760
C     UTEST(3) = TAIR+5.0                                               SID09770
C 10  DO 50 I=I1,I2                                                     SID09780
C        CALL HEATO(UTEST(I),TAIR,W,WFF,EVAP,CONDUC,XLWRO,DAY,EA)       SID09790
C        HNET(I) = XLWRI-XLWRO-EVAP-CONDUC+SOLAR                        SID09800
C        IF (ABS(HNET(I)).LT.3.0) GO TO 100                             SID09810
C 50  CONTINUE                                                          SID09820
C                                                                       SID09830
C     USE BISECTION METHOD TO DEFINE NEW UTEST VALUES                   SID09840
C     I1 = 2                                                            SID09850
C     I2 = 2                                                            SID09860
C     IF (HNET(1).LT.0.0) GO TO 80                                      SID09870
C     IF (HNET(3).GT.0.0) GO TO 90                                      SID09880
C     J1 = 3                                                            SID09890
C     IF ((HNET(1)*HNET(2)).GT.0.0)J1=1                                 SID09900
```

```
C     UTEST(J1) = UTEST(2)                                                    SID09910
C     HNET(J1) = HNET(2)                                                      SID09920
C 70  UTEST(2) = (UTEST(1)+UTEST(3))/2.0                                      SID09930
C     IT = IT+1                                                               SID09940
C     IF(IT.GT.50) GO TO 200                                                  SID09950
C     GO TO 10                                                                SID09960
C 80  UTEST(1) = UTEST(1)-5.0                                                 SID09970
C     I1 = 1                                                                  SID09980
C     GO TO 70                                                                SID09990
C 90  UTEST(3) = UTEST(3)+5.0                                                 SID10000
C     I2 = 3                                                                  SID10010
C     GO TO 70                                                                SID10020
                                                                              SID10030
      UTEST(1)=TAIR+5.0                                                       SID10040
      UTEST(2)=TAIR                                                           SID10050
      UTEST(3)=TAIR-5.0                                                       SID10060
      DO 50 I=1,3                                                             SID10070
         CALL HEATO (UTEST(I),TAIR,W,WFF,EVAP,CONDUC,XLWRO,DAY,EA)            SID10080
         HNET(I)=XLWRI-XLWRO-EVAP-CONDUC+SOLAR                                SID10090
         IF (ABS(HNET(I)).LT.3.0) GOTO 100                                    SID10100
50    CONTINUE                                                                SID10110
C                                                                             SID10120
C     USE VARIABLE SECANT METHOD TO DETERMINE NEW UTEST VALUES                SID10130
C                                                                             SID10140
      SLOPE=0.0                                                               SID10150
      DO 60 I=2,99                                                            SID10160
         SLOPE=(HNET(I)-HNET(I-1))/(UTEST(I)-UTEST(I-1))                      SID10170
         UTEST(I+1)=UTEST(I)-HNET(I)/SLOPE                                    SID10180
         CALL HEATO (UTEST(I+1),TAIR,W,WFF,EVAP,CONDUC,XLWRO,DAY,EA)          SID10190
         HNET(I+1)=XLWRI-XLWRO-EVAP-CONDUC+SOLAR                              SID10200
         IF (ABS(HNET(I+1)).LT.3.0) GOTO 100                                  SID10210
60    CONTINUE                                                                SID10220
                                                                              SID10230
C     EQUILIBRIUM TEMPERATURE FOUND                                           SID10240
 100  TEQ = UTEST(I)                                                          SID10250
      RETURN                                                                  SID10260
                                                                              SID10270
C     ERROR EXIT                                                              SID10280
 200  WRITE(IRITE,201)                                                        SID10290
 201  FORMAT(' *** EQUILO *** FAILURE TO CONVERGE')                           SID10300
      STOP                                                                    SID10310
      END                                                                     SID10320
                                                                              SID10330
                                                                              SID10340
                                                                              SID10350
      SUBROUTINE HEATO(TS,TAIR,W,WFF,EVAP,CONDUC,XLWRO,DAY,EA)                SID10360
C                                                                             SID10370
C     COMPUTES THE OUTGOING COMPONENTS OF THE LAKE HEAT BALANCE                SID10380
C                                                                             SID10390
      ES = EE(TS)                                                             SID10400
      TSV = (TS+460.0)/(1.0-0.378*ES/760.0)-460.0                             SID10410
      TAV = (TAIR+460.0)/(1.0-0.378*EA/760.0)-460.0                           SID10420
      DTV = TSV-TAV                                                           SID10430
      DTVL = 0.0024 * W*W*W                                                   SID10440
      DTV = AMAX1(DTV,DTVL)                                                   SID10450
```

```
      WFF=0.75+0.15*COS(((2.0*3.14/365.25)*(DAY-151.)))                 SID10460
      EVAP = WFF*(22.4*DTV**0.3333 + 14.0*W)                            SID10470
      ENAT = 17.0*W                                                     SID10480
      EVAP = AMAX1(EVAP,ENAT)                                           SID10490
      XLWRO = 1802.0*(TS/460.0+1.0)**4                                  SID10500
      CONDUC = 0.255*(TS-TAIR)*EVAP                                     SID10510
      EVAP = EVAP*(ES-EA)                                               SID10520
      IF(EVAP.LT.0.0)EVAP=0.0                                           SID10530
      RETURN                                                            SID10540
      END                                                               SID10550
                                                                        SID10560
                                                                        SID10570
                                                                        SID10580
      SUBROUTINE ALI(X,ARG,VAL,Y,NDIM)                                  SID10590
C                                                                       SID10600
C     LINEAR INTERPOLATION                                              SID10610
C                                                                       SID10620
      DIMENSION ARG(21),VAL(21)                                         SID10630
      IF(X.LT.0.0.OR.X.GT.1.0)GO TO 40                                  SID10640
      DO 10 K=1,NDIM                                                    SID10650
      IF(X.LT.ARG(K)) GO TO 20                                          SID10660
   10 CONTINUE                                                          SID10670
   20 Z1=ARG(K)-X                                                       SID10680
      Z2=ARG(K)-ARG(K-1)                                                SID10690
      Z3=VAL(K-1)-VAL(K)                                                SID10700
      Z4=(Z1*Z3)/Z2                                                     SID10710
      Y=VAL(K)+Z4                                                       SID10720
      GO TO 50                                                          SID10730
   40 IF(X.LT.0.0)Y=VAL(1)                                              SID10740
      IF(X.GT.1.0)Y=VAL(NDIM)                                           SID10750
   50 RETURN                                                            SID10760
      END                                                               SID10770
                                                                        SID10780
                                                                        SID10790
                                                                        SID10800
      SUBROUTINE ALI2(X,ARG,VAL,Y,NDIM)                                 SID10810
      DIMENSION ARG(11),VAL(11)                                         SID10820
      IF(Y.LT.0.0.OR.Y.GT.1.0)GO TO 40                                  SID10830
      DO 10 K=1,NDIM                                                    SID10840
      IF(Y.GT.VAL(K)) GO TO 20                                          SID10850
   10 CONTINUE                                                          SID10860
   20 S1=VAL(K-1)-Y                                                     SID10870
      S2=VAL(K-1)-VAL(K)                                                SID10880
      S3=ARG(K)-ARG(K-1)                                                SID10890
      S4=(S1*S3)/S2                                                     SID10900
      X=ARG(K-1)+S4                                                     SID10910
      GO TO 50                                                          SID10920
   40 IF(Y.LT.0.0)X=ARG(NDIM)                                           SID10930
      IF(Y.GT.1.0)X=ARG(1)                                              SID10940
   50 RETURN                                                            SID10950
      END                                                               SID10960
                                                                        SID10970
                                                                        SID10980
      SUBROUTINE DATIME(DATE,TIME)                                      SID10990
C                                                                       SID11000
```

```
C     DETERMINES DATE AND TIME OF RUN FROM IBM SYSTEM                    SID11010
C                                                                        SID11020
      DIMENSION A(5)                                                     SID11030
      DOUBLE PRECISION DATE,TIME,DUM1,DUM2                               SID11040
      EQUIVALENCE (DUM1,A(1)),(DUM2,A(3))                                SID11050
C                                                                        SID11060
      CALL WHEN(A)                                                       SID11070
      DATE = DUM1                                                        SID11080
      TIME = DUM2                                                        SID11090
      RETURN                                                             SID11100
      END                                                                SID11110
                                                                         SID11120
                                                                         SID11130
      FUNCTION EXCOEF(T)                                                 SID11140
C                                                                        SID11150
C     COMPUTES COEFFICIENT OF THERMAL EXPANSION                          SID11160
C     OF WATER AS A FUNCTION OF WATER TEMPERATURE                        SID11170
C                                                                        SID11180
      EXCOEF = (5.17E-2*(T-40.0)-3.84E-04*(T-40.0)**2+                   SID11190
     +        1.5E-06*(T-40.0)**3)*1.0E-04                               SID11200
      RETURN                                                             SID11210
      END                                                                SID11220
                                                                         SID11230
                                                                         SID11240
      SUBROUTINE DDAY(IMO,IDAY,IYR,DAY)                                  SID11250
      DIMENSION IDAZE(12)                                                SID11260
      DATA IDAZE/0,31,59,90,120,151,181,212,243,273,304,334/             SID11270
      ID=IDAY+IDAZE(IMO)                                                 SID11280
      IF(IMO.GT.2)ID=ID+LEAP(IYR)                                        SID11290
      DAY=FLOAT(ID)                                                      SID11300
      RETURN                                                             SID11310
      END                                                                SID11320
                                                                         SID11330
                                                                         SID11340
      FUNCTION LEAP(IYEAR)                                               SID11350
      LEAP=0                                                             SID11360
      IF(MOD(IYEAR,4).EQ.0)LEAP=1                                        SID11370
      IF(MOD(IYEAR,100).EQ.0)LEAP=0                                      SID11380
      IF(MOD(IYEAR,400).EQ.0)LEAP=1                                      SID11390
      RETURN                                                             SID11400
      END                                                                SID11410
                                                                         SID11420
                                                                         SID11430
      SUBROUTINE TRED(BETA,TSTAR,TS,HO,QO,SLSA,TRF)                      SID11440
C     ENTRAINMENT RATE ACCORDING TO MOORE AND LONG(1971)                 SID11450
C     WHERE THE CONSTANT IS APPROX ADJUSTED SINCE THE RI                 SID11460
C     NO.'S ARE SO LARGE IN COMPARISON TO THE LAB DATA                   SID11470
      FROUDE=(QO/HO)/SQRT(32.2*BETA*(TS-TSTAR)*HO)                       SID11480
      WI=1.0E-3*(QO/HO)*FROUDE**2                                        SID11490
      W=WI*SLSA                                                          SID11500
      R=W/QO                                                             SID11510
      TRF=1.0-R                                                          SID11520
      IF(TRF.LT.0.15)TRF=0.15                                            SID11530
      RETURN                                                             SID11540
      END                                                                SID11550
```

73.53	77.58	76.05	73.65	75.30	76.56	74.13	74.60
72.79	71.56	72.71	73.03	76.58	76.66	77.15	75.15
73.65	74.20	73.67	75.23	76.50	78.80	80.36	76.61
78.09	74.79	69.10	68.90	72.69	75.78	75.93	74.57
73.72	76.62	77.72	76.37	72.92	69.78	69.09	72.19
71.71	70.86	72.62	76.85	75.57	78.11	69.22	73.08
74.69	62.26	61.27	65.41	70.71	72.86	76.02	76.96
64.61	69.69	70.53	58.82	59.84	63.85	57.45	58.53
62.17	54.48	57.17	61.02	61.05	57.57	59.73	56.20
59.47	50.33	49.58	46.68	51.17	53.40	61.10	66.70
61.15	47.71	43.61	46.87	47.80	52.21	54.48	53.91
58.84	62.52	47.32	49.87	65.22	58.02	48.73	50.52
49.55	49.58	51.35	49.71	51.29	50.32	54.39	56.45
58.50	48.15	47.22	49.55	54.41	51.84	52.20	57.71
54.17	45.05	44.65	56.58	44.03	43.44	43.66	42.99
43.21	49.52	38.42	38.62	33.96	45.20	38.36	43.78
40.49	44.81	44.23	58.46	47.45	46.88	45.28	64.28
52.79	29.68	27.90	29.69	32.35	34.82	41.94	45.57
42.88	40.31	38.84	40.27	50.25	34.77	36.79	33.64
40.20	39.43	32.32	24.87	28.11	33.49	38.51	54.75
48.81	17.68	23.96	31.18	33.21	41.72	40.59	25.49
28.31	23.22	22.55	30.30	37.25	30.03	37.01	38.09
36.91	26.21	33.12	43.04	40.96	35.90	49.46	39.01
41.08	41.64	39.86	43.01	36.83	32.50	35.82	31.00
27.00	40.00	29.00	22.00	26.00	28.23	16.58	12.43
11.18	18.71	14.49	14.08	20.34	22.38	15.19	22.73
26.37	30.01	36.27	34.96	35.07	36.08	37.03	34.57
37.06	39.10	42.80	44.78	40.26	48.59	57.91	50.82
41.04	43.72	48.55	55.14	35.96	38.55	48.77	53.50
33.11	35.17	43.83	48.43	45.65	45.68	48.82	51.55
54.56	53.88	44.97	41.53	41.47	43.76	62.87	69.16
68.71	62.42	56.54	53.21	45.79	51.82	47.68	43.97
48.45	47.77	46.61	52.80	62.52	49.39	50.13	54.31
48.67	50.95	52.33	54.41	53.71	58.07	62.45	61.53
61.35	65.99	56.75	60.68	56.05	52.50	57.16	58.90
59.14	64.90	63.83	53.89	59.61	65.58	70.08	70.37
70.93	71.03	68.84	67.03	59.62	61.76	62.17	59.02
58.62	59.42	63.50	61.92	64.14	68.99	66.89	61.78
52.90	61.10	62.72	63.64	67.32	68.04	72.46	72.64
63.64	63.22	65.13	74.00	76.00	76.00	79.00	80.00
68.00	66.00	69.00	66.00	69.04	67.08	66.09	72.33
66.84	65.78	60.68	71.79	74.66	62.11	61.39	61.87
63.75	69.22	72.42	73.58	75.00	77.00	73.55	65.52
67.74	64.72	67.98	69.24	70.83	69.15	72.25	76.86
78.76	75.17	75.59	76.59	76.86	75.32	73.92	73.46
71.43	73.29	76.00	75.55	75.64	76.78	75.09	76.30
74.95	70.79	77.94	79.60	78.21	75.64	77.10	77.64
78.03	76.13	77.90	76.34	81.96	75.47	60.38	65.47
70.32	67.73	62.20	63.70	65.59	72.30	73.06	69.39
67.98	72.29	74.60	74.95	76.70	75.18	72.68	74.03
74.21	75.70	74.25	74.89	73.10	73.67	72.41	76.26
72.20	65.52	61.49	63.44	68.02	69.28	67.71	73.91
64.66	61.85	62.29	64.79	67.16	64.39	66.15	66.02
58.58	57.45	60.41	64.43	65.34	63.71	68.05	67.38
65.75	66.93	64.64	61.08	60.21	52.66	56.41	54.40

60.87	41.52	42.79	51.24	54.18	45.34	46.42	52.59
57.77	59.21	59.19	62.71	68.70	69.04	66.05	50.25
48.04	45.31	41.52	48.76	56.54	51.04	50.28	52.63
61.98	50.90	45.80	45.01	47.15	50.08	43.55	54.06
58.42	49.70	44.08	45.49	44.46	40.22	48.08	47.29
56.68	51.44	51.72	53.27	56.66	60.61	64.93	62.67
63.19	53.27	53.88	38.39	31.56	34.79	34.32	33.02
41.21	45.48	47.83	45.30	43.49	34.99	44.73	50.98
57.73	51.91	37.31	34.11	42.39	31.29	27.75	32.22
31.51	33.36	38.43	48.01	54.04	48.64	42.21	34.97
38.37	41.73	39.44	37.45	35.81	34.13	32.37	31.05
28.00	27.56	33.21	36.51	34.07	29.71	37.97	38.77
30.32	36.75	44.46	41.16	39.63	46.00	43.00	44.00
39.00	44.00	40.00	32.21	36.70	37.39	32.38	37.35
32.88	25.55	24.19	21.89	21.80	21.91	25.83	27.78
27.70	29.02	31.98	31.00	29.04	29.14	30.38	28.31
37.59	37.46	37.88	25.88	27.81	35.56	42.84	47.28
49.19	46.10	46.18	38.14	31.82	33.65	28.17	23.95
13.86	18.54	27.10	39.41	45.19	42.11	49.36	61.01
52.03	47.51	42.73	33.01	30.57	38.74	42.45	48.53
57.76	49.66	43.55	53.11	54.96	41.70	42.87	46.75
47.19	40.75	44.61	43.58	51.54	51.56	45.13	45.85
53.96	55.49	60.26	49.91	52.33	53.30	58.10	62.10
61.46	60.33	58.62	54.78	59.20	50.35	46.82	46.63
56.28	57.93	64.35	61.18	59.46	60.96	60.91	61.53
49.93	51.63	52.57	55.35	54.05	58.02	60.47	61.16
64.81	72.95	68.12	68.75	58.98	59.70	68.61	64.36
73.48	74.17	68.41	57.71	59.59	59.89	65.34	67.89
69.48	70.55	69.45	68.86	68.58	69.26	63.54	62.56
69.90	76.83	77.97	76.00	77.00	79.76	77.03	68.69
66.33	64.70	77.27	76.51	63.11	65.56	62.25	63.90
64.13	68.76	77.65	69.48	64.04	64.72	66.19	69.03
68.33	70.33	74.21	73.22	71.46	70.83	77.74	80.92
80.18	73.38	72.43	78.04	76.87	76.93	75.68	75.97
71.77	68.31	74.04	80.00	81.22	75.84	73.18	74.11
77.25	84.37	79.39	77.13	80.27	83.47	82.34	77.94
72.03	73.40	75.27	77.40	75.68	76.13	75.98	76.54
79.25	80.76	81.88	83.14	78.18	77.66	79.13	80.05
82.12	79.45	77.26	78.48	78.78	76.43	77.79	78.30
70.72	67.00	67.08	71.34	74.82	68.25	68.38	71.57
74.58	75.10	74.84	76.93	77.13	78.50	77.88	78.72
79.47	79.62	77.13	76.35	76.52	77.32	73.48	71.95
73.51	71.30	66.29	70.37	73.88	74.36	72.11	66.95
73.22	71.96	69.34	71.47	74.67	76.29	75.39	64.25
64.62	68.10	57.48	56.93	61.30	61.60	66.00	69.00
56.00	60.00	57.00	54.00	54.00	60.00	68.00	63.00
68.32	55.00	52.00	50.00	60.00	64.00	68.44	68.46
61.26	54.65	57.22	57.12	48.07	49.42	50.12	44.92
44.13	54.44	46.40	44.64	47.25	55.71	44.77	46.44
51.28	50.28	45.56	56.36	63.02	56.90	53.50	39.97
41.55	48.62	57.22	52.42	40.64	33.70	37.65	37.84
35.80	40.55	38.79	42.79	43.16	42.69	37.47	39.72
39.83	39.95	44.81	53.72	53.88	40.91	36.33	37.58
46.26	49.97	57.70	57.58	46.43	35.93	39.50	46.77
39.40	34.12	37.83	33.95	39.33	40.43	19.62	19.94

22.68	30.96	31.73	17.88	18.32	25.76	32.88	41.73
40.30	27.46	29.22	34.98	32.28	16.68	20.40	29.37
32.81	19.76	21.85	23.01	17.03	13.04	18.27	30.78
29.93	33.02	25.41	26.10	39.23	40.53	35.33	38.84
35.04	34.12	35.86	49.33	49.76	38.93	34.49	27.10
25.94	37.08	44.85	22.30	22.97	22.97	31.85	36.20
38.53	31.15	40.76	51.80	20.49	24.98	32.61	37.15
43.26	51.88	54.80	54.59	51.18	50.60	51.85	51.37
40.90	41.88	40.41	43.04	54.58	52.77	42.86	37.27
36.23	37.02	39.12	35.87	38.04	40.57	39.59	41.21
41.85	52.89	37.99	47.30	43.97	39.74	28.97	32.59
33.50	39.04	39.02	40.54	43.68	42.99	46.93	52.58
48.86	59.68	59.02	63.00	58.98	61.99	62.97	68.37
63.23	47.66	48.16	58.54	59.92	58.09	67.35	64.31
47.11	54.47	50.87	53.99	59.91	67.48	59.15	53.31
46.43	55.38	67.43	59.04	50.08	52.98	59.63	69.20
72.53	58.18	62.00	54.00	57.00	62.00	61.00	66.00
55.00	53.00	61.00	59.22	64.35	60.83	57.99	67.34
64.49	59.46	61.11	53.17	47.68	54.03	60.23	66.44
65.52	69.45	72.91	73.42	71.36	67.08	68.56	72.76
70.66	61.24	63.80	71.26	74.72	76.22	74.04	72.39
73.18	78.75	73.98	71.98	71.66	77.70	78.29	80.33
82.20	81.00	76.00	81.00	80.00	81.00	85.00	78.00
77.00	84.00	74.00	69.00	71.00	73.00	73.00	70.12
69.62	70.37	70.97	70.50	72.87	77.99	79.17	81.98
79.48	77.45	76.96	79.16	80.45	72.67	72.00	78.00
78.00	81.00	86.00	85.00	79.00	76.00	72.00	79.00
83.00	85.00	85.00	81.00	73.00	72.00	71.48	72.83
74.48	76.78	80.96	69.34	71.95	71.29	77.28	77.37
75.75	70.41	72.68	73.66	75.38	76.62	66.03	65.29
64.69	67.06	66.59	64.06	67.08	71.94	69.94	68.40
70.13	73.15	69.45	71.91	73.26	73.12	73.24	71.58
67.44	67.70	69.86	68.18	70.27	71.18	70.35	75.42
77.58	78.35	79.20	76.38	69.00	68.00	65.00	62.00
67.00	68.00	70.00	59.00	58.00	61.00	66.00	69.00
65.00	58.00	63.00	69.30	54.35	49.64	53.12	66.34
68.94	58.17	52.27	50.48	49.05	53.10	52.57	48.44
48.90	50.99	58.28	51.04	56.24	48.37	45.41	54.55
57.83	57.28	42.72	42.77	52.09	62.81	57.90	49.49
50.52	49.10	53.55	55.74	56.57	57.87	63.82	55.10
46.98	49.55	49.46	45.17	45.84	41.69	40.96	48.37
54.51	52.02	47.99	50.26	49.43	55.90	38.66	34.93
32.96	33.09	35.79	39.04	55.99	41.84	36.16	36.22
34.85	44.37	40.60	41.21	40.29	38.92	42.09	46.33
33.88	30.34	31.85	33.41	30.40	31.47	33.30	33.07
33.53	33.45	24.25	21.92	25.71	36.75	50.63	39.08
35.64	29.97	34.36	37.00	37.08	29.78	30.33	43.69
34.13	32.69	42.89	40.68	44.96	47.23	33.63	29.27
8.35	10.41	20.31	23.23	24.99	23.86	31.94	8.90
21.95	21.32	31.61	29.69	24.29	26.98	31.58	25.10
20.53	19.60	34.01	34.51	38.97	50.93	43.56	32.09
37.98	43.89	36.75	36.60	29.70	36.30	39.38	31.51
32.90	33.39	32.46	37.80	51.89	53.04	37.75	30.23
35.95	41.85	42.59	39.38	42.41	41.94	32.17	29.11
28.57	34.28	35.68	45.79	34.75	33.02	47.06	35.61

34.05	33.75	41.99	37.85	51.95	55.67	56.86	53.78
40.75	37.98	52.28	46.50	51.68	42.15	57.00	46.93
43.70	47.25	52.37	44.32	34.16	34.03	43.23	52.65
60.34	61.24	58.29	57.95	53.61	43.18	39.19	34.36
37.85	38.65	42.91	46.24	48.95	62.27	54.63	53.25
59.28	68.39	56.47	57.19	55.16	58.67	50.05	51.36
56.36	61.51	57.83	58.62	53.68	51.95	54.73	61.07
62.71	62.65	58.84	60.82	63.67	70.11	65.53	60.83
60.87	61.71	67.86	70.08	68.67	66.54	67.47	72.04
70.27	68.46	74.87	72.00	74.00	69.00	68.03	66.75
67.13	68.33	71.78	73.19	72.91	75.40	70.89	68.17
68.57	64.00	73.00	69.00	70.00	71.36	70.61	65.24
64.98	61.36	61.67	65.48	70.60	75.61	72.15	69.70
72.94	69.67	71.88	68.77	69.85	68.08	69.57	74.21
72.99	75.41	75.28	72.56	67.22	68.69	74.18	68.93
69.10	69.40	74.76	75.52	73.99	72.73	73.89	75.56
75.06	72.13	74.05	76.50	78.38	79.74	77.59	75.64
72.48	74.26	74.11	75.06	75.53	79.61	79.92	76.21
71.66	69.22	69.25	72.92	71.63	77.00	79.00	81.00
82.00	79.00	81.00	80.00	79.00	75.00	71.00	71.00
73.00	76.00	76.00	76.65	69.01	70.37	74.94	71.11
65.34	71.57	75.20	75.87	67.79	66.92	69.07	58.66
61.02	69.83	71.84	75.03	72.34	64.25	63.22	64.29
67.63	65.40	65.84	66.65	68.40	70.03	70.07	70.52
70.77	72.45	64.49	67.99	63.38	62.93	60.74	57.74
56.46	57.55	61.48	57.89	62.57	61.75	62.08	60.04
60.85	62.39	62.45	65.66	66.68	67.45	67.48	66.50
66.25	55.59	52.91	57.97	57.50	57.15	55.39	50.66
44.30	46.07	55.10	59.82	53.58	44.33	41.16	41.55
43.01	51.10	49.42	49.66	53.78	53.24	59.78	63.74
64.15	64.62	62.58	43.45	37.10	41.18	48.25	52.18
48.84	50.33	62.18	43.75	37.06	39.09	40.22	39.73
45.31	47.03	48.64	52.08	55.25	58.91	52.00	36.92
45.61	42.05	34.88	50.04	48.59	52.72	52.29	54.81
63.08	63.57	59.69	44.66	40.68	36.67	30.10	37.23
26.37	22.77	26.92	39.82	51.07	35.66	28.21	27.95
33.94	34.75	33.33	42.43	49.82	58.12	59.90	44.74
58.51	51.10	34.61	36.49	38.77	40.20	37.08	32.54
35.16	39.27	46.69	39.71	38.98	42.61	47.50	36.58
29.86	37.98	37.62	33.39	30.44	23.61	21.93	24.48
29.83	32.85	36.54	40.24	39.70	35.75	37.23	35.73
36.14	43.57	41.40	39.88	49.24	48.93	34.32	30.99
30.47	35.42	34.21	30.26	29.81	22.79	28.35	27.84
32.93	38.90	39.54	42.59	41.58	39.29	39.56	45.71
48.87	47.44	40.31	37.03	33.81	35.52	43.26	44.78
53.27	49.83	60.90	56.05	50.58	56.21	43.30	42.81
45.26	43.31	42.87	46.90	48.84	57.91	46.36	44.48
51.20	56.35	54.02	56.27	40.14	38.65	32.86	36.73
40.12	43.92	48.89	45.47	41.05	39.72	46.36	49.25
54.16	52.45	50.23	54.47	55.45	61.45	53.25	51.05
49.90	50.75	50.78	53.53	55.94	48.13	44.78	39.17
39.12	43.69	48.54	54.46	53.20	50.56	50.55	58.69
66.76	71.74	72.74	70.37	72.10	73.36	71.43	61.03
59.03	58.51	63.45	64.22	52.86	52.95	59.51	59.86
62.98	67.41	70.94	56.44	57.08	58.34	59.47	68.98

69.16	66.75	72.78	66.43	62.63	64.62	58.62	61.97
64.35	67.71	66.04	61.86	62.01	68.23	68.80	71.39
72.85	71.79	68.71	65.72	66.84	71.65	74.83	77.76
78.22	77.29	77.66	77.92	73.03	73.42	74.38	76.00
75.00	75.00	80.28	77.65	75.04	79.08	80.52	68.89
68.47	74.71	78.64	81.55	81.80	73.46	70.86	67.63
69.75	74.83	73.51	73.59	78.56	81.88	79.09	81.57
84.82	83.39	82.37	81.52	84.00	86.00	77.00	75.00
81.00	75.00	76.00	77.00	78.00	78.00	80.00	83.00
82.00	83.52	81.17	77.60	78.45	79.59	80.24	82.29
79.22	77.09	78.68	72.69	66.60	67.26	69.51	73.14
77.26	79.51	80.84	85.32	82.82	83.79	78.11	75.30
75.24	75.08	79.18	78.68	78.81	78.84	75.40	74.79
72.83	75.11	77.80	82.84	84.87	82.24	76.52	75.55
79.71	81.86	80.82	67.80	64.29	63.02	63.98	71.35
70.75	76.26	76.69	70.17	56.88	53.37	54.48	56.21
59.61	62.47	64.50	59.15	56.96			
0.86	0.82	0.73	0.70	0.84	0.81	0.90	0.89
0.95	0.96	0.92	0.99	0.88	0.85	0.82	0.76
0.87	0.90	0.97	0.88	0.94	0.83	0.73	0.71
0.77	0.69	0.70	0.74	0.74	0.76	0.80	0.87
0.94	0.89	0.81	0.85	0.96	0.88	0.82	0.83
0.77	0.71	0.77	0.71	0.80	0.71	0.73	0.81
0.80	0.95	0.80	0.89	0.85	0.84	0.83	0.79
0.92	0.95	0.92	0.92	0.85	0.76	0.68	0.80
0.73	0.63	0.85	0.95	0.74	0.83	0.99	1.00
0.72	0.70	0.64	0.72	0.77	0.81	0.79	0.74
0.74	0.65	0.92	0.86	0.76	0.75	0.63	0.68
0.69	0.63	0.56	0.62	0.60	0.54	0.77	0.71
0.67	0.80	0.73	0.70	0.66	0.80	0.67	0.61
0.56	0.83	0.86	0.94	0.96	0.88	0.92	0.73
0.93	1.00	1.00	0.72	0.78	0.73	0.84	0.86
0.97	0.68	0.63	0.61	1.00	0.73	0.82	1.00
0.81	0.74	0.95	1.00	1.00	0.77	0.95	0.92
1.00	0.68	0.73	0.86	0.81	0.53	0.39	0.49
0.42	0.34	0.59	0.89	0.54	0.61	0.67	1.00
0.56	0.51	0.46	0.59	0.57	0.57	0.97	0.94
0.93	0.48	0.52	0.61	0.94	1.00	0.75	0.50
0.61	0.63	0.78	1.00	0.72	0.35	0.42	0.70
0.45	0.44	0.86	0.87	0.59	0.71	0.79	0.45
0.50	0.63	0.70	0.51	0.78	1.00	1.00	0.46
0.75	0.56	0.38	0.51	0.92	0.69	0.86	0.61
0.67	1.00	0.83	0.84	1.00	1.00	0.59	0.43
0.74	0.61	1.00	1.00	1.00	1.00	1.00	1.00
0.85	0.79	0.87	0.97	1.00	1.00	1.00	1.00
0.86	0.81	0.84	0.67	0.89	0.61	0.62	0.61
0.42	0.41	0.44	0.53	0.58	0.56	0.59	0.65
0.74	0.97	0.90	0.73	0.72	0.69	0.71	0.69
0.74	0.93	1.00	1.00	1.00	0.70	0.42	0.38
0.61	0.90	0.64	0.60	0.73	0.94	0.71	0.50
0.56	0.48	0.45	0.36	0.38	0.48	0.65	0.80
0.90	0.74	0.96	0.82	0.66	0.56	0.51	0.40
0.47	0.62	0.89	0.57	0.67	0.63	0.74	0.77
0.82	0.82	0.90	0.94	0.84	0.70	0.53	0.48
0.68	0.89	0.80	0.89	0.79	0.87	0.92	0.72

0.59	0.57	0.80	0.69	0.58	0.89	1.00	0.98
1.00	1.00	0.90	0.79	0.79	0.76	0.77	0.77
0.81	0.66	0.61	0.61	0.51	0.87	1.00	0.83
0.82	0.71	0.85	0.85	0.77	1.00	0.77	0.78
0.79	0.81	0.80	0.86	0.76	0.60	0.70	1.00
0.64	0.64	0.09	0.78	0.79	0.87	0.90	0.88
0.94	0.96	1.00	0.92	0.94	0.96	1.00	0.98
1.00	1.00	0.97	0.98	0.98	0.94	1.00	0.96
1.00	1.00	0.97	0.86	0.96	0.98	0.87	0.83
0.79	0.87	0.85	0.89	0.82	0.92	1.00	0.83
0.78	0.69	0.72	0.68	0.91	0.96	0.97	1.00
0.97	0.96	0.92	0.88	0.88	0.96	1.00	1.00
0.93	0.83	0.93	0.97	1.00	0.98	1.00	0.87
0.93	0.85	0.88	0.89	0.87	0.93	1.00	0.97
0.76	0.79	0.80	0.76	0.73	0.72	1.00	1.00
1.00	0.98	1.00	1.00	0.89	1.00	1.00	1.00
1.00	1.00	0.85	0.90	0.87	0.87	0.67	0.75
0.79	1.00	1.00	0.96	0.95	0.88	0.78	0.80
0.79	0.87	0.89	0.99	0.90	0.83	0.88	0.86
0.73	0.72	0.69	0.76	0.73	0.93	0.90	1.00
0.92	0.94	0.75	0.80	0.84	0.70	0.87	0.78
1.00	1.00	1.00	1.00	0.75	0.74	0.63	0.66
0.56	0.75	0.78	0.78	0.77	0.84	0.85	0.94
0.88	0.73	0.74	0.71	0.66	0.65	0.64	0.57
0.45	0.49	0.75	0.92	0.62	0.65	0.61	0.59
0.76	1.00	0.81	0.85	0.99	0.63	0.62	0.72
1.00	1.00	0.95	1.00	1.00	0.87	0.76	0.86
0.64	0.56	0.72	0.75	0.65	0.65	0.74	0.89
0.92	0.78	0.73	0.64	0.70	0.78	0.89	0.44
0.54	0.93	0.79	0.64	0.85	0.96	0.74	0.63
0.60	0.80	0.83	0.58	0.75	0.59	0.49	0.53
0.60	0.43	0.64	0.47	0.48	0.54	0.52	0.54
0.77	0.64	0.55	0.72	0.62	0.62	0.46	0.52
0.50	0.69	0.67	0.43	0.47	0.49	0.72	0.76
0.93	0.93	0.69	0.83	0.52	0.43	0.78	0.50
0.69	0.53	0.34	0.31	0.71	0.57	0.71	0.84
0.67	0.81	0.74	0.65	0.93	0.58	0.44	0.46
0.75	0.48	0.52	0.72	0.67	0.47	0.59	0.83
0.74	0.67	0.58	0.85	0.89	0.92	0.96	0.74
0.65	0.67	0.65	0.45	0.48	0.71	0.81	0.73
0.47	0.45	0.67	0.82	0.91	0.52	0.48	0.52
0.43	0.52	0.44	0.50	0.46	0.50	0.64	0.65
0.80	1.00	0.98	0.81	0.82	0.77	0.71	0.67
0.53	0.39	0.55	0.51	0.60	0.41	0.35	0.67
0.70	0.66	0.64	0.58	0.58	0.62	0.88	0.86
0.83	0.73	0.74	0.90	0.94	0.84	0.48	0.53
0.57	0.57	0.62	0.71	0.74	0.69	0.77	0.63
0.57	0.85	0.85	0.64	0.52	0.62	0.65	0.53
0.65	0.69	0.77	0.82	0.62	0.69	0.75	0.68
0.68	0.77	0.80	0.86	0.95	1.00	0.83	0.85
0.87	0.81	0.78	0.78	0.95	0.88	0.92	0.76
0.70	0.98	0.69	0.77	0.69	0.85	0.76	0.64
0.69	0.69	0.87	0.84	0.79	0.73	0.74	0.85
1.00	0.89	0.81	0.71	0.75	0.84	0.88	0.80
0.76	0.70	0.68	0.72	0.81	0.89	0.76	0.76

0.70	0.74	0.84	0.83	0.78	0.64	0.73	0.81
0.65	0.71	0.91	0.94	0.79	0.96	0.85	0.78
0.74	0.69	0.71	0.67	0.73	0.73	0.70	0.73
0.74	0.51	0.36	0.37	0.41	0.37	0.31	0.27
0.30	0.27	0.28	0.32	0.31	0.36	0.35	0.37
0.40	0.40	0.41	0.41	0.42	0.41	0.36	0.37
0.48	0.28	0.28	0.38	0.35	0.46	0.87	0.81
0.93	0.60	0.67	0.60	0.62	0.65	0.66	0.81
0.64	0.53	0.49	0.59	0.56	0.63	0.67	0.89
0.89	0.63	0.60	0.67	0.63	0.79	0.89	0.50
0.63	0.88	0.65	0.75	0.63	0.40	0.52	0.60
1.00	0.63	0.57	0.49	0.58	0.80	0.52	0.53
0.53	0.41	0.34	0.51	0.75	0.86	0.72	0.66
0.75	0.61	0.67	0.61	0.98	0.73	0.60	0.63
0.70	0.61	0.52	0.41	0.43	0.37	0.51	0.49
0.49	0.56	0.64	0.80	0.81	0.71	0.66	0.60
0.52	0.66	0.73	0.61	0.61	0.55	0.59	0.64
0.68	0.84	0.95	0.61	0.66	0.68	0.91	0.90
0.78	0.81	0.93	0.54	0.60	0.56	0.62	0.53
0.65	0.58	0.70	0.60	0.64	0.74	0.73	0.56
0.90	0.69	0.54	0.65	0.51	0.61	0.93	0.70
0.71	0.71	0.69	0.50	0.58	0.58	0.56	0.54
0.52	0.48	0.31	0.44	0.52	0.51	0.54	0.64
0.76	0.45	0.72	0.83	0.50	0.58	0.69	0.60
0.66	0.81	0.57	0.95	0.94	0.73	0.95	0.93
0.81	0.55	0.60	0.47	0.48	0.58	0.62	0.50
0.76	0.97	0.54	0.52	0.62	0.61	0.59	0.55
0.47	0.37	0.44	0.28	0.43	0.48	0.74	0.60
0.50	0.54	0.58	0.63	0.61	0.77	0.68	0.79
0.64	0.68	0.99	0.73	0.84	0.37	0.36	0.43
0.64	0.49	0.43	0.49	0.75	0.28	0.45	0.54
0.61	0.54	0.40	0.23	0.73	0.54	0.43	0.66
0.39	0.55	0.67	0.60	0.50	0.53	0.76	0.61
0.55	0.83	0.58	0.53	0.39	0.40	0.68	0.68
0.44	0.49	0.65	0.56	0.49	0.36	0.29	0.30
0.43	0.32	0.29	0.36	0.48	0.36	0.27	0.25
0.29	0.31	0.31	0.33	0.31	0.51	0.47	0.39
0.42	0.43	0.50	0.47	0.45	0.36	0.44	0.32
0.38	0.37	0.44	0.34	0.43	0.44	0.48	0.45
0.41	0.70	0.71	0.72	0.72	0.70	0.61	0.54
0.67	0.63	0.60	0.53	0.50	0.58	0.69	0.87
0.88	0.95	0.97	0.92	0.97	0.79	0.74	0.77
0.86	0.75	0.82	0.82	0.71	0.62	0.82	0.72
0.67	0.65	0.65	0.70	0.69	0.64	0.82	0.77
0.74	0.70	0.70	0.67	0.56	0.64	0.70	0.77
0.86	0.85	0.80	0.99	0.94	1.00	0.81	0.84
0.94	0.95	0.80	0.77	0.81	0.80	0.68	0.65
0.91	0.78	0.64	0.87	0.76	0.76	0.74	0.80
0.76	0.64	0.86	0.91	0.84	0.84	0.79	0.84
0.95	0.96	0.92	0.90	0.90	0.61	0.58	0.58
0.70	0.68	0.63	0.75	0.87	0.84	0.73	0.61
0.61	0.68	0.66	0.56	0.56	0.58	0.61	0.66
0.57	0.60	0.65	0.75	0.77	0.65	0.69	0.60
0.79	0.51	0.59	0.63	0.83	0.80	0.68	0.73
0.83	0.72	0.64	0.59	0.92	0.54	0.53	0.48

0.63	0.99	0.74	1.00	1.00	1.00	0.81	0.95
0.88	0.96	0.95	0.91	0.71	0.78	0.76	0.69
0.45	0.42	0.58	0.82	0.80	0.61	0.46	0.36
0.45	0.51	0.86	0.61	0.60	0.62	0.54	0.60
0.57	0.89	0.80	0.70	0.81	0.68	0.64	0.57
0.97	0.99	0.91	0.92	0.73	0.49	0.63	0.60
0.51	0.47	0.57	0.61	0.71	0.94	1.00	0.85
0.92	0.64	0.51	0.55	0.59	0.97	0.83	0.83
0.98	0.94	0.94	0.86	0.65	0.65	0.89	0.77
0.53	0.92	0.82	0.45	0.47	0.59	0.38	0.42
0.45	0.46	0.51	0.91	0.91	0.61	0.71	0.40
0.48	0.85	0.85	0.93	0.84	0.98	0.45	0.62
0.59	0.57	0.48	0.58	0.60	0.97	0.72	0.89
1.00	1.00	0.90	0.98	0.65	0.50	0.91	0.80
0.76	0.66	0.78	0.57	0.67	1.00	0.98	0.99
0.92	0.74	0.77	0.62	0.61	0.73	0.72	0.49
0.94	0.78	0.74	0.64	0.83	0.96	0.89	0.94
1.00	0.68	0.49	0.59	0.76	0.61	0.73	0.31
0.72	1.00	0.78	0.78	0.69	0.99	0.66	0.48
0.62	0.62	0.68	0.49	0.44	0.45	0.47	0.54
0.76	0.35	0.31	0.72	0.24	0.47	0.60	0.37
0.49	0.78	0.50	0.48	0.53	0.46	0.48	0.57
0.74	0.75	0.56	0.50	0.74	0.55	0.34	0.32
0.55	0.51	0.99	0.97	0.56	0.46	0.60	0.53
0.59	0.58	0.50	0.48	0.59	0.60	0.69	0.49
0.42	0.40	0.56	0.70	0.58	0.66	0.69	0.64
0.71	0.86	0.73	0.79	0.85	0.93	0.91	0.93
0.90	0.92	0.91	0.85	0.88	0.78	0.89	0.68
0.76	1.00	0.84	0.71	0.71	0.75	0.77	0.89
0.93	0.85	0.97	0.78	0.79	0.58	0.91	0.74
0.72	0.66	0.78	0.94	0.71	0.70	0.80	0.84
0.94	0.86	0.93	0.86	0.65	0.71	0.84	0.90
0.78	0.83	0.88	0.89	0.97	1.00	1.00	0.92
0.90	1.00	1.00	0.92	0.92	0.86	0.92	0.92
0.78	0.84	0.96	0.90	0.86	0.78	0.88	0.96
0.85	1.00	1.00	0.88	0.85	0.71	0.74	0.79
0.77	0.85	0.82	0.82	0.82	0.90	0.82	0.76
0.79	0.79	0.85	0.80	0.87	0.86	0.79	0.78
0.89	0.98	0.90	0.79	0.74	1.00	0.87	0.71
0.87	0.88	0.91	0.90	0.89	0.78	0.80	0.80
0.82	0.95	0.85	0.88	0.86	0.87	0.93	0.86
0.91	0.89	0.72	0.81	0.86	1.00	1.00	1.00
0.88	0.85	0.82	1.00	0.91	0.88	0.93	1.00
0.94	0.86	0.77	0.91	1.00	0.92	0.89	0.92
1.00	0.98	0.90	1.00	1.00	0.83	0.71	0.60
0.68	0.74	0.80	0.94	0.88	0.70	0.73	0.87
1.00	0.71	0.79	0.78	0.70	0.74	0.68	0.79
0.93	0.91	1.00	0.75	0.67	0.65	0.61	0.71
0.69	0.79	0.89	0.84	0.82	0.79	0.62	0.90
0.95	1.00	1.00	1.00	0.92	0.87	0.80	0.65
0.81	0.82	1.00	0.96	0.88	1.00	1.00	1.00
0.96	0.98	0.94	0.69	0.77	0.74	0.75	1.00
1.00	0.79	0.82	0.80	0.96	0.77	0.75	1.00
0.89	0.65	0.75	0.78	0.94	0.96	0.87	1.00
0.95	1.00	0.77	0.91	1.00	1.00	0.79	0.68

0.96	0.85	0.67	0.66	0.86	1.00	0.78	0.69
0.62	0.58	0.84	0.49	0.63	0.59	0.53	0.50
0.50	1.00	1.00	0.85	0.71	0.77	0.88	0.80
0.70	0.87	0.81	0.77	1.00	0.74	0.52	0.54
0.94	0.84	0.57	0.72	0.83	1.00	0.66	0.62
0.93	0.68	0.81	0.87	0.69	0.67	0.66	0.69
0.64	0.98	0.73	0.96	0.65	0.54	0.62	1.00
0.45	0.57	0.50	0.83	1.00	1.00	1.00	1.00
1.00	0.85	0.61	0.46	0.57	0.56	0.63	0.95
1.00	0.82	0.64	0.77	0.51	0.48	0.62	0.45
0.42	1.00	0.87	0.65	0.58	0.94	0.68	0.89
0.80	0.63	0.74	1.00	1.00	1.00	1.00	1.00
0.69	0.65	0.75	0.82	1.00	0.66	0.66	0.67
0.62	0.50	0.47	0.43	0.77	1.00	0.48	0.40
0.44	0.39	0.50	0.61	0.66	0.66	0.72	0.75
0.58	0.45	0.62	0.78	0.53	0.51	0.47	0.58
0.68	0.83	0.79	1.00	0.55	0.64	0.82	0.93
0.99	1.00	0.82	0.50	0.48	0.58	0.40	0.45
0.74	0.90	0.61	0.67	0.57	0.56	0.72	0.75
0.85	0.74	0.58	0.68	0.62	0.67	0.65	0.65
0.71	0.71	0.70	0.69	0.84	0.92	0.89	0.82
0.74	0.69	0.62	0.65	0.71	0.79	0.81	1.00
0.89	0.88	0.80	0.75	0.77	0.89	0.79	0.60
0.64	0.60	0.57	0.65	0.68	0.68	0.56	0.62
0.64	0.64	0.62	0.76	0.72	0.65	0.64	0.67
0.67	0.71	0.64	0.60	0.56	0.62	0.56	0.53
0.65	0.65	0.67	0.78	0.80	0.73	0.67	0.63
0.68	0.61	0.74	0.70	0.72	0.60	0.60	0.62
0.62	0.74	0.70	0.60	0.47	0.57	0.79	0.75
0.61	0.59	0.63	0.75	0.74	0.61	0.77	0.72
0.75	0.62	0.80	0.68	0.59	0.60	0.53	0.61
0.51	0.43	0.61	1.00	0.86	0.52	0.64	0.63
0.59	0.57	0.63	0.86	0.56	0.59	0.62	0.64
0.69	0.72	0.72	0.67	0.89			
3.74	4.88	5.20	4.73	3.82	3.71	4.21	3.34
5.23	4.14	3.72	3.01	4.89	4.38	3.31	4.24
4.04	2.72	3.24	2.97	3.05	3.80	3.99	3.18
2.92	7.50	6.28	2.90	2.26	2.42	3.33	3.50
4.24	4.58	4.65	2.96	4.21	5.23	2.66	2.74
4.63	3.52	2.81	3.30	3.16	5.34	4.54	5.03
4.25	6.36	5.31	4.13	3.10	3.55	2.93	5.81
3.61	3.71	6.06	7.24	2.87	4.35	4.42	2.68
3.92	4.11	2.43	2.82	6.88	3.99	4.09	3.60
5.65	5.67	7.56	4.05	2.28	2.33	3.84	4.91
7.74	5.38	4.35	6.83	3.81	4.11	5.13	3.31
2.46	4.03	6.75	5.26	7.24	5.32	2.99	5.20
7.70	3.56	4.06	2.97	4.27	5.53	3.48	3.03
3.82	6.04	2.84	2.42	2.66	4.10	5.61	5.38
3.50	6.12	4.29	5.72	3.62	3.68	3.84	4.83
4.23	7.22	6.52	5.18	5.69	5.58	4.93	3.24
3.94	3.20	3.27	7.31	3.77	3.46	2.56	5.85
9.53	7.25	3.68	2.41	6.08	7.23	6.43	3.71
13.96	7.07	3.58	5.17	10.53	3.09	2.64	5.59
9.48	5.33	7.71	5.23	3.44	2.90	2.64	4.98
6.71	8.40	5.72	3.17	3.85	3.00	7.01	4.06

3.27	6.88	5.41	6.51	9.06	5.51	4.21	4.76
12.61	4.73	4.68	7.08	7.27	3.31	6.98	10.72
9.07	3.04	7.06	10.75	8.09	5.03	14.90	5.45
3.03	5.57	6.02	3.16	4.84	3.85	9.02	5.51
3.90	3.88	4.03	3.31	4.95	7.39	8.90	6.89
10.02	3.19	4.04	2.15	2.93	4.46	7.50	5.00
5.41	2.24	2.47	3.24	3.79	4.79	5.63	3.65
3.12	3.99	3.33	5.93	8.09	4.44	6.54	9.95
8.84	3.51	2.45	5.70	4.15	4.52	3.47	3.95
6.37	8.22	5.53	6.23	3.97	4.22	6.33	5.85
5.12	5.37	4.15	5.28	5.48	6.54	12.63	5.95
5.79	8.84	5.84	3.83	3.94	8.35	7.17	10.89
7.87	7.88	6.92	6.35	3.98	4.46	3.71	3.34
2.69	5.00	7.15	4.59	5.21	3.68	4.04	5.29
4.34	6.25	5.85	6.33	4.48	4.99	4.57	4.40
4.77	3.36	3.76	3.60	4.01	3.64	6.35	7.79
7.67	8.54	3.69	3.63	4.22	5.07	5.57	6.24
7.53	6.44	5.35	3.90	3.91	3.43	4.48	4.17
4.67	6.01	2.89	2.99	4.29	4.47	3.86	4.51
8.11	3.75	4.87	3.83	3.42	4.29	5.66	5.22
7.63	4.52	5.09	5.67	6.41	8.05	5.99	5.91
3.29	3.47	4.01	4.64	4.18	8.07	4.44	4.88
6.30	3.79	2.19	3.04	4.20	3.63	2.84	2.87
3.05	3.02	4.47	3.18	3.84	2.92	5.53	2.81
3.79	3.53	4.27	4.16	3.83	5.09	2.73	3.70
4.54	7.34	3.67	3.20	4.73	4.39	5.07	4.81
4.36	4.15	4.09	3.70	5.33	5.24	9.81	5.85
4.64	7.20	6.92	3.94	4.29	2.45	3.52	4.91
4.27	4.61	3.63	4.52	3.23	4.73	3.82	4.16
2.94	3.96	4.76	4.58	3.59	3.69	11.33	7.03
2.93	8.13	5.45	3.59	2.31	4.45	6.31	7.25
6.28	5.14	2.75	3.01	6.19	4.45	5.45	6.41
8.60	7.38	5.04	2.79	4.41	3.42	2.11	5.77
6.41	3.81	6.47	3.62	5.83	6.66	7.84	6.00
6.35	8.53	3.78	4.47	4.41	5.67	3.18	2.33
2.21	1.88	2.86	3.69	4.05	4.35	5.24	6.93
4.73	5.06	4.30	4.41	3.95	3.56	4.81	3.29
4.64	8.69	4.82	4.33	3.47	6.27	3.53	4.16
3.55	6.22	8.24	6.80	10.04	3.86	6.14	3.49
3.08	2.68	2.64	2.52	3.54	3.04	5.89	4.03
7.71	4.86	6.00	6.51	6.12	3.48	10.91	3.62
4.23	4.62	4.97	4.20	7.93	4.08	2.70	3.29
5.11	3.94	4.67	2.81	2.82	11.56	3.62	4.91
6.45	4.70	2.07	3.02	4.39	6.14	5.45	3.87
4.29	2.78	2.40	3.31	4.14	5.42	3.30	5.96
8.75	3.67	7.11	3.07	2.38	3.18	3.97	5.60
5.52	8.65	9.30	2.69	3.34	3.70	2.90	3.53
3.50	3.88	7.96	5.48	2.64	3.82	3.13	3.13
4.77	6.33	9.41	10.69	8.48	4.51	7.87	3.14
6.01	9.66	5.10	3.13	7.83	3.95	5.21	3.46
3.27	3.21	8.13	6.90	3.36	3.46	3.30	2.79
3.38	2.80	3.85	4.63	10.91	7.85	6.21	8.74
8.14	12.66	7.55	4.44	7.21	5.45	5.48	7.80
5.76	4.60	9.99	6.62	8.85	9.64	6.57	4.31
7.57	8.94	3.67	4.98	13.62	16.73	4.55	4.03

6.22	4.64	4.59	3.83	3.93	6.48	6.23	5.02
5.32	5.84	9.50	10.72	3.54	4.18	7.26	5.55
4.95	3.78	4.30	4.86	7.26	8.98	10.85	3.72
4.07	3.43	4.39	6.70	4.98	4.17	5.17	5.08
5.43	5.32	6.43	4.55	5.39	8.45	4.31	4.07
4.81	3.80	3.66	5.56	5.24	5.38	4.96	5.85
4.16	7.58	7.86	5.76	4.49	5.55	5.41	2.96
5.35	6.38	2.56	2.45	3.03	5.35	7.52	4.49
2.78	3.38	4.10	6.45	3.73	6.40	4.98	8.21
6.22	4.26	4.91	11.63	8.28	6.45	5.74	4.54
3.19	2.98	4.84	6.20	4.91	3.78	6.75	7.19
4.25	3.35	3.77	4.14	4.56	6.64	3.46	4.56
5.40	5.99	4.75	5.60	4.55	3.03	3.80	6.60
4.58	6.81	3.51	3.67	4.86	4.05	4.07	3.35
5.28	3.72	4.78	3.34	2.93	4.75	4.38	5.14
4.60	6.00	3.86	4.28	6.26	6.71	4.53	3.32
3.39	5.24	4.24	6.09	3.65	4.03	3.87	2.83
2.58	5.15	3.15	3.53	3.34	3.45	2.92	3.39
7.49	3.57	4.65	2.49	5.05	5.50	5.58	3.49
3.26	3.12	2.18	2.23	2.15	2.56	4.49	4.32
3.04	4.25	3.98	3.70	3.83	3.06	7.05	4.12
4.61	5.87	3.53	3.18	3.73	3.61	6.91	5.84
4.70	3.92	2.82	4.51	4.24	4.06	6.11	5.01
3.67	7.57	5.32	2.89	4.64	7.09	5.80	5.45
4.50	5.76	5.83	6.74	3.16	3.61	3.92	4.44
5.69	6.51	5.30	3.33	3.71	3.35	5.48	6.16
3.40	4.79	6.05	3.66	6.76	6.72	9.23	7.41
3.64	6.01	5.20	4.12	4.09	6.97	5.71	4.89
4.22	5.05	4.56	5.67	6.70	4.69	10.38	9.64
6.22	3.57	3.90	4.74	7.44	6.09	9.65	4.24
2.77	4.04	3.01	3.31	4.23	6.04	4.10	6.49
6.31	7.36	6.17	5.24	8.26	10.44	2.71	3.22
2.68	2.42	4.57	5.97	5.12	2.62	4.52	5.48
6.21	4.45	4.96	6.99	5.25	7.65	8.32	4.41
2.97	2.53	5.25	9.47	3.79	7.42	8.74	8.42
6.25	3.71	3.65	6.26	6.47	10.89	3.75	5.30
8.71	5.41	3.02	7.17	3.86	5.24	3.71	3.14
3.71	5.29	10.41	3.09	3.56	2.06	5.85	4.21
2.77	2.83	3.39	6.07	4.47	2.96	7.31	7.80
3.40	5.61	10.11	7.25	7.16	4.28	5.39	3.46
6.80	5.08	7.44	10.02	8.15	3.78	3.00	3.54
3.43	3.53	3.51	3.51	4.76	12.07	4.08	4.47
3.75	8.02	10.61	6.94	5.46	5.92	4.91	7.05
3.51	6.77	7.71	10.68	7.37	4.67	4.75	7.23
4.98	7.87	11.23	6.95	12.24	10.67	6.37	7.31
7.80	7.51	4.75	7.46	7.17	5.14	4.99	7.65
4.63	7.99	6.69	3.93	3.55	5.59	6.45	9.59
8.49	10.66	4.16	6.37	6.41	5.79	5.86	6.49
7.37	7.78	7.19	7.37	12.80	11.43	7.73	12.00
7.37	8.45	13.12	14.42	11.54	4.75	4.36	8.75
12.41	6.56	7.69	9.69	6.69	5.09	5.23	5.40
8.55	7.48	5.74	6.46	6.62	5.66	4.02	6.01
5.14	6.80	3.83	4.98	6.65	5.22	6.95	6.86
3.43	3.31	4.43	5.61	7.72	6.56	2.98	4.08
4.51	5.07	3.91	5.19	5.81	5.18	6.33	6.00

4.28	6.73	5.49	5.08	3.73	3.11	2.82	2.93
3.32	5.80	4.65	4.27	4.48	3.88	7.05	4.93
4.05	6.27	6.94	5.52	3.98	3.39	6.20	9.13
5.03	3.97	4.80	2.92	2.68	4.13	4.41	3.88
3.63	4.24	3.27	3.87	6.63	6.45	2.58	4.11
3.00	3.53	6.84	5.76	5.69	5.34	3.79	4.06
4.16	3.60	4.26	7.72	5.40	4.13	3.61	3.19
4.87	3.26	4.65	4.18	3.52	3.25	3.25	2.73
3.39	3.48	3.25	2.85	4.96	4.91	7.07	5.32
5.17	9.31	7.30	5.94	4.29	3.25	5.93	3.64
5.29	3.99	2.88	3.81	3.96	4.75	5.34	5.24
4.73	4.02	3.16	4.08	4.88	3.58	3.69	3.09
2.57	2.60	3.55	3.66	6.63	4.16	6.49	6.00
5.84	3.30	5.38	7.50	6.02	3.29	3.05	4.78
6.58	3.67	3.74	5.81	9.12	8.14	3.65	3.78
5.42	8.73	8.50	4.58	3.53	4.06	6.15	7.31
4.15	2.90	7.55	5.26	6.85	7.43	4.39	3.46
4.39	5.48	6.34	5.62	3.57	5.25	5.79	5.86
6.60	6.01	3.07	1.90	3.44	2.60	5.14	9.96
10.46	3.08	4.10	6.28	5.48	10.10	9.00	10.45
9.32	2.97	5.79	5.66	4.46	8.70	8.83	4.78
3.60	6.04	6.26	4.72	6.94	5.02	4.09	4.74
4.23	3.69	4.66	4.24	11.23	8.08	4.87	11.11
12.72	9.48	5.62	5.83	2.64	3.72	8.41	5.58
4.89	7.09	7.69	4.58	5.25	5.57	6.00	2.68
3.26	3.42	3.30	4.91	7.00	3.22	3.80	7.75
4.51	4.45	7.33	6.84	7.13	7.76	6.60	8.65
11.56	7.74	5.15	6.26	6.07	4.89	8.46	6.41
4.29	4.39	3.19	5.82	6.09	5.39	8.21	4.45
10.38	4.11	6.46	2.96	4.68	5.69	8.53	5.30
4.35	6.31	3.11	4.25	4.39	3.45	5.21	4.19
4.15	4.63	5.71	3.99	6.81	3.00	9.43	6.44
4.11	3.31	5.33	11.29	3.28	5.04	8.78	4.42
5.58	5.80	3.78	5.58	5.81	3.93	6.66	6.36
6.55	4.78	7.95	3.52	5.24	3.72	5.48	4.89
6.33	4.37	7.01	4.75	4.81	4.94	5.09	7.21
3.86	5.01	5.52	10.92	10.46	4.40	4.18	5.39
9.14	8.67	5.66	10.62	13.79	5.64	14.39	12.44
5.12	6.36	4.11	6.02	4.81	8.88	4.97	3.89
4.36	8.50	6.36	5.31	4.03	4.59	6.08	5.60
3.24	5.02	3.98	5.80	10.81	5.00	3.17	3.72
4.22	3.77	5.59	4.13	4.20	4.14	6.35	8.80
7.49	6.16	2.81	3.74	5.23	3.51	3.72	3.24
4.51	4.80	6.44	2.70	5.70	5.30	3.74	3.70
3.73	4.18	4.92	3.82	5.51	3.00	4.73	5.03
4.86	6.98	6.82	7.88	8.38	4.84	3.50	5.57
2.99	4.82	6.17	2.88	5.03	7.72	4.72	3.90
5.38	4.44	4.81	3.39	4.40	3.31	5.25	3.96
3.15	4.67	5.45	3.83	5.17	4.51	6.58	5.02
4.42	3.65	5.13	4.10	3.29	3.55	3.10	3.38
3.58	3.43	3.04	3.17	3.82	4.46	3.69	4.12
4.98	4.30	4.29	4.12	3.61	3.08	2.71	5.29
3.79	3.41	3.19	4.30	3.34	4.77	3.19	3.05
4.20	3.97	3.69	6.23	4.09	4.94	5.96	2.66
2.22	3.20	3.66	4.09	4.16	3.21	4.57	5.69

5.31	4.50	3.17	7.37	3.86	4.68	5.67	5.00
4.80	5.29	5.85	5.75	4.86	3.73	4.01	3.78
3.94	3.96	3.66	3.92	2.90	2.44	4.02	3.71
3.33	5.16	5.07	4.68	5.07	3.08	5.21	7.69
3.89	3.26	3.52	6.75	5.73	3.80	5.90	5.42
3.25	4.83	3.69	2.23	2.38	2.68	4.04	4.30
4.33	6.68	4.25	2.76	4.77	5.27	6.94	7.65
5.16	4.19	4.63	3.88	6.74	6.51	7.79	9.22
15.89	9.74	3.21	2.58	3.22	2.38	2.91	4.48
4.02	4.69	7.31	6.71	2.51	3.08	3.06	2.92
4.43	4.36	8.08	9.79	4.59	8.10	3.67	2.90
3.36	5.11	4.73	2.66	2.55	3.34	10.12	2.89
5.88	6.49	3.61	5.11	2.98	3.93	2.49	4.12
6.10	5.49	6.38	4.85	3.96	9.38	5.85	5.91
11.03	5.17	6.51	4.36	10.62	11.32	5.81	3.83
5.77	5.03	3.36	5.22	3.57	6.07	7.57	3.58
7.28	4.13	3.49	3.91	2.12	3.02	5.59	5.87
6.09	3.29	5.81	4.87	5.60	5.90	5.77	7.10
3.18	5.53	5.86	7.33	5.03	7.67	8.01	6.45
5.31	6.56	5.26	3.70	3.23	2.89	5.47	8.21
3.42	3.88	2.73	5.36	6.20	6.99	8.76	4.54
5.76	7.20	6.29	2.46	5.55	13.51	7.70	3.41
8.73	2.85	3.23	5.68	6.70	2.65	2.65	3.33
2.89	5.34	3.59	8.33	9.14	3.16	3.72	11.56
9.51	3.74	4.29	6.65	5.08	4.35	7.20	4.62
3.34	11.45	14.74	5.70	3.18	4.75	7.71	11.01
11.71	8.16	5.00	9.59	10.03	6.87	6.51	10.51
4.95	6.27	4.90	7.31	4.60	5.04	4.39	9.39
7.32	6.05	2.66	3.51	3.67	3.36	7.31	6.83
7.82	5.47	6.25	7.95	8.12	10.94	6.06	4.87
14.28	9.32	4.53	4.08	5.76	11.69	12.62	6.22
4.65	5.75	6.39	8.22	7.68	8.86	10.30	7.97
4.21	4.95	5.96	7.89	7.20	3.70	4.19	4.70
3.58	3.30	7.36	8.51	4.31	4.98	5.71	5.33
4.61	4.02	4.90	5.86	4.17	4.50	6.70	3.98
4.18	3.04	3.93	3.62	3.85	3.36	6.29	3.17
4.61	4.68	3.37	5.61	3.43	3.60	3.02	2.63
3.25	3.16	3.71	3.36	4.46	3.43	4.88	6.08
6.27	3.14	11.85	4.28	3.40	4.99	5.58	5.69
3.52	3.79	3.57	2.43	3.41	3.63	6.44	5.59
2.84	3.55	3.75	2.58	2.78	2.93	3.51	3.50
3.92	4.48	3.13	3.59	4.18	5.18	5.53	4.13
6.14	4.16	4.32	3.52	4.97	4.54	4.00	3.75
6.92	3.57	4.73	4.02	3.96	2.92	3.60	3.09
4.47	5.01	4.96	7.33	6.85	5.89	3.85	4.63
3.76	5.09	4.18	3.27	3.69	5.00	5.03	5.09
4.85	3.57	3.31	3.68	4.97	3.96	4.75	5.00
4.52	4.23	5.01	6.58	5.85	4.23	5.79	4.26
3.86	3.62	4.48	6.69	9.01	7.45	4.57	4.60
3.99	3.56	5.25	7.54	6.22	4.06	4.42	3.36
2.85	4.44	7.03	7.76	8.53			
0.69	0.68	0.76	0.50	0.90	0.00	0.94	0.73
0.93	0.98	0.60	0.90	0.73	0.64	0.56	0.44
0.61	0.75	0.92	0.68	0.74	0.20	0.36	0.35
0.32	0.39	0.00	0.16	0.33	0.38	0.70	0.63

0.80	0.40	0.46	0.59	0.81	0.90	0.48	0.66
0.35	0.06	0.17	0.21	0.51	0.45	0.90	0.79
0.45	1.00	0.80	0.69	0.50	0.30	0.33	0.51
0.70	0.20	0.58	1.00	0.70	0.00	0.00	0.00
0.37	0.00	0.59	1.00	0.00	0.29	0.98	0.90
0.15	0.66	0.53	0.00	0.00	0.00	0.11	0.90
0.79	0.25	0.50	0.96	0.00	0.73	0.80	0.00
0.00	0.50	0.20	0.20	0.70	0.40	0.80	0.40
0.40	0.40	0.00	0.00	0.00	1.00	0.00	0.00
0.62	0.84	0.00	0.84	0.64	0.86	1.00	0.00
1.00	1.00	1.00	0.00	0.00	0.90	0.90	0.99
1.00	0.00	0.18	1.00	1.00	0.83	1.00	1.00
0.80	0.00	0.87	1.00	0.70	0.00	0.53	0.80
1.00	0.00	0.70	0.00	0.00	0.00	0.00	0.14
0.00	0.70	0.82	1.00	0.00	0.00	0.00	1.00
0.00	0.40	0.20	0.00	0.00	1.00	1.00	1.00
1.00	0.00	0.00	0.80	0.90	1.00	0.90	0.00
0.40	1.00	1.00	1.00	0.70	0.20	0.80	1.00
0.00	0.96	1.00	0.87	0.50	0.90	0.90	0.00
0.00	0.00	1.00	0.00	0.00	0.92	0.00	0.00
0.78	0.00	0.00	0.90	1.00	0.00	0.90	0.00
1.00	1.00	0.47	1.00	1.00	0.89	0.50	1.00
0.00	0.00	1.00	0.57	1.00	1.00	1.00	1.00
0.00	0.00	0.78	0.80	1.00	1.00	1.00	1.00
0.00	0.90	0.00	0.53	0.59	0.00	0.00	0.80
0.00	0.00	0.00	0.34	0.00	0.00	0.00	0.00
0.90	0.92	0.62	0.00	0.19	0.05	0.19	0.00
0.25	0.53	1.00	1.00	1.00	0.00	0.00	0.00
0.55	0.90	0.00	0.80	0.84	1.00	0.00	0.51
0.93	0.52	0.00	0.00	0.00	0.00	0.41	0.98
1.00	1.00	1.00	0.80	0.85	0.27	0.00	0.00
0.00	0.70	0.90	0.00	0.33	0.09	0.38	0.41
1.00	0.42	0.80	1.00	1.00	0.33	0.00	0.00
1.00	0.98	0.46	0.97	1.00	0.89	0.90	0.70
0.40	0.80	0.70	0.30	0.60	1.00	0.90	0.90
1.00	0.70	0.40	0.50	0.40	0.70	0.70	0.60
0.70	0.20	0.10	0.20	0.30	1.00	0.70	0.50
0.70	0.30	0.90	0.61	0.66	1.00	0.25	0.62
0.30	0.51	1.00	0.70	0.50	0.53	0.44	0.90
0.41	0.28	0.54	0.59	0.54	0.80	0.57	0.79
0.70	0.80	0.80	0.77	0.73	0.87	0.79	0.94
0.93	0.89	0.71	0.77	0.87	0.88	0.70	0.71
0.81	1.00	1.00	0.10	0.68	0.80	0.30	0.57
0.30	0.47	0.70	0.56	0.20	0.40	1.00	0.37
0.32	0.46	0.40	0.54	0.90	0.76	0.50	0.00
0.77	0.80	0.71	0.30	0.45	0.67	0.83	0.70
0.40	0.38	0.53	0.67	0.90	0.62	1.00	0.50
0.00	0.50	0.43	0.00	0.33	0.40	1.00	0.80
0.43	0.00	0.00	0.00	0.23	1.00	1.00	1.00
0.99	1.00	1.00	0.93	0.82	1.00	0.94	1.00
0.77	0.50	0.00	0.44	0.90	0.00	0.60	0.00
0.70	1.00	0.65	0.60	1.00	0.00	0.00	0.93
0.31	0.20	0.32	0.23	0.00	0.00	0.82	0.50
0.00	0.70	0.00	0.70	0.00	0.00	0.00	0.63
0.93	0.82	0.00	0.00	0.00	0.00	0.63	0.33

1.00	1.00	1.00	1.00	0.63	0.41	0.00	0.00
0.00	0.00	0.00	0.34	0.00	0.82	1.00	0.75
0.75	0.00	0.46	0.41	0.00	0.00	0.00	0.00
0.21	0.00	1.00	0.00	0.00	0.00	0.00	0.00
0.00	1.00	0.00	0.00	0.94	0.00	0.67	1.00
1.00	1.00	0.88	1.00	1.00	0.82	0.00	0.87
0.00	0.00	0.80	0.70	0.20	0.00	0.48	1.00
1.00	0.00	0.94	1.00	0.96	0.90	1.00	0.95
0.92	1.00	0.00	0.80	1.00	1.00	0.00	0.40
0.00	1.00	0.72	0.64	0.67	1.00	0.84	0.60
1.00	0.61	0.90	0.00	0.00	0.00	0.80	0.00
1.00	0.00	0.70	1.00	0.00	0.00	0.00	0.00
0.00	0.93	0.88	0.00	0.00	0.80	0.00	0.37
0.87	0.95	0.00	1.00	0.18	0.58	1.00	0.00
1.00	0.80	0.00	0.00	0.80	0.00	0.59	0.92
0.00	0.45	0.00	1.00	1.00	0.00	0.00	0.00
1.00	0.40	0.50	1.00	0.80	0.40	0.50	1.00
0.80	0.80	0.30	1.00	0.90	1.00	1.00	0.10
0.34	0.07	0.69	0.00	0.00	0.88	0.90	0.57
0.38	0.00	0.96	0.86	0.90	0.68	0.60	0.00
0.00	0.00	0.03	0.00	0.00	0.19	0.43	0.74
1.00	1.00	1.00	0.55	0.57	0.84	0.63	0.42
0.00	0.14	0.53	0.00	0.90	0.43	0.00	0.76
0.57	0.85	0.68	0.00	0.00	0.75	0.68	0.70
1.00	0.90	0.56	0.87	0.86	0.68	0.00	0.40
0.18	0.00	0.72	0.70	0.70	0.63	0.83	0.13
0.00	0.93	0.56	0.39	0.00	0.55	0.20	0.12
0.12	0.19	0.52	0.92	0.00	0.77	0.20	0.18
0.44	0.06	0.50	1.00	0.93	0.99	0.44	0.40
0.66	0.50	0.23	0.29	1.00	0.76	0.61	0.00
0.00	0.90	0.57	0.80	0.50	0.60	0.10	0.10
0.10	0.00	0.80	0.60	0.20	0.10	0.30	0.70
1.00	0.40	0.20	0.10	0.50	0.90	0.70	0.20
0.20	0.70	0.10	0.40	0.60	0.50	0.60	0.40
0.20	0.70	0.80	0.50	0.80	0.00	0.00	0.84
0.00	0.90	0.98	0.80	0.42	1.00	0.60	0.31
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.80	0.37	0.40	0.34	0.19	0.00	0.00	0.00
0.30	0.80	0.00	0.30	0.10	0.40	0.50	0.50
0.63	0.61	1.00	0.80	0.70	0.00	0.07	0.77
0.93	0.80	0.60	1.00	0.80	0.95	1.00	0.00
1.00	0.00	1.00	0.00	0.00	0.00	0.00	0.90
0.58	0.30	0.00	0.30	0.30	0.10	0.52	0.90
0.48	0.00	0.00	0.00	1.00	0.48	1.00	0.00
0.00	0.99	0.81	1.00	0.00	0.00	0.00	0.00
1.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00
0.35	0.00	0.80	1.00	0.70	1.00	0.57	0.00
0.00	0.00	0.00	0.94	1.00	0.00	0.00	0.90
1.00	1.00	0.00	0.60	0.00	0.00	0.00	0.72
0.51	1.00	0.70	1.00	1.00	0.75	0.00	0.89
0.00	0.86	1.00	0.00	0.00	0.36	0.00	0.00
0.80	1.00	1.00	0.00	0.80	1.00	1.00	0.92
1.00	0.95	0.89	0.00	0.83	0.00	0.00	0.00
0.00	0.00	1.00	0.00	0.57	0.00	0.11	0.25
0.80	0.65	0.00	0.00	0.60	0.78	1.00	0.70

0.53	0.00	0.00	0.90	1.00	0.00	0.40	0.70
0.00	0.90	0.54	0.00	0.00	0.00	0.00	1.00
1.00	0.00	0.89	1.00	0.30	0.00	0.50	0.00
0.60	0.92	0.90	0.97	1.00	1.00	1.00	0.90
0.66	0.00	0.00	0.00	1.00	0.80	1.00	0.00
0.75	1.00	0.00	0.00	0.19	0.58	0.43	0.58
0.00	0.00	0.00	0.50	1.00	0.00	0.90	0.80
0.42	0.00	1.00	0.85	0.00	0.70	0.00	0.80
0.00	0.90	0.90	0.00	1.00	0.00	0.70	0.90
1.00	0.18	0.00	0.00	1.00	0.00	0.67	0.57
1.00	0.76	0.00	0.70	1.00	0.27	0.80	0.19
0.70	1.00	0.55	0.71	0.25	0.00	0.00	0.00
0.65	0.96	0.64	0.26	0.00	0.00	0.90	0.92
0.00	0.00	0.31	1.00	0.80	0.56	0.00	0.00
0.80	0.00	0.00	1.00	1.00	1.00	0.00	0.00
0.00	0.00	0.00	0.00	0.90	0.78	0.54	0.00
0.63	1.00	0.90	0.69	1.00	0.80	0.00	0.00
0.00	0.55	0.80	0.80	1.00	0.90	0.67	0.23
0.00	0.00	0.90	0.31	0.66	0.00	0.50	0.50
0.15	0.00	0.80	0.00	0.00	0.00	0.16	0.69
0.81	0.94	0.75	0.76	0.91	0.00	0.00	0.00
0.66	0.00	0.00	0.53	0.90	0.00	1.00	0.00
0.00	0.00	0.00	0.00	0.34	0.00	1.00	0.81
0.90	0.06	0.65	0.42	0.00	0.50	0.00	0.00
0.90	0.50	0.50	1.00	0.90	1.00	0.60	0.50
1.00	0.70	0.10	0.40	0.40	0.60	0.00	0.00
0.98	0.80	0.00	1.00	0.80	0.90	0.00	0.00
0.00	0.00	1.00	0.78	0.10	0.60	0.70	0.90
1.00	1.00	1.00	0.90	1.00	0.10	0.00	0.20
0.30	0.10	0.50	0.90	1.00	1.00	1.00	0.50
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.60	1.00	0.60	0.20	0.60	0.40
0.90	0.00	0.00	1.00	0.68	0.92	0.52	0.00
0.00	0.70	0.30	0.30	1.00	0.20	0.00	0.00
0.70	1.00	0.80	1.00	1.00	1.00	0.30	1.00
1.00	1.00	1.00	0.90	0.72	1.00	1.00	0.00
0.00	0.00	0.90	0.92	0.00	0.90	0.30	0.90
1.00	0.00	1.00	0.20	0.00	0.60	0.30	0.10
0.20	1.00	0.20	0.50	1.00	0.60	0.80	0.70
1.00	0.40	0.70	1.00	0.70	0.10	1.00	0.00
0.70	0.50	1.00	0.00	0.50	1.00	1.00	0.00
0.71	0.94	0.00	0.00	0.90	1.00	1.00	0.90
1.00	1.00	0.99	0.70	0.60	0.00	1.00	0.80
0.70	1.00	0.80	0.00	0.91	0.00	0.00	0.87
0.00	0.00	0.00	1.00	1.00	0.00	0.90	0.00
0.90	1.00	0.90	1.00	0.95	1.00	0.00	0.75
0.00	0.00	0.00	0.00	1.00	0.97	0.00	0.90
1.00	1.00	1.00	0.94	0.00	0.00	1.00	0.00
0.70	0.00	0.12	0.00	0.00	1.00	1.00	1.00
0.80	0.60	0.47	0.80	0.60	1.00	0.40	0.90
1.00	0.30	0.60	1.00	1.00	1.00	0.90	1.00
1.00	0.00	0.00	0.55	0.90	0.00	1.00	0.90
1.00	1.00	0.00	0.65	1.00	1.00	1.00	0.00
0.00	0.00	0.46	0.60	0.30	0.00	0.00	0.70
0.90	0.00	0.10	0.90	0.00	1.00	0.40	0.20

1.00	0.80	0.30	0.90	0.70	0.80	0.40	0.32
0.64	0.92	0.00	0.00	1.00	1.00	0.90	0.00
0.00	0.90	1.00	1.00	0.80	0.00	1.00	0.00
0.80	0.00	0.00	0.00	0.00	0.00	0.78	0.00
0.00	0.00	0.58	0.90	0.00	0.80	0.00	0.00
0.00	0.72	1.00	1.00	1.00	1.00	1.00	1.00
1.00	1.00	1.00	0.80	1.00	0.40	0.80	0.00
0.63	1.00	0.86	0.80	0.65	0.00	0.00	1.00
1.00	1.00	1.00	0.00	0.00	0.27	0.69	1.00
0.00	0.00	0.00	0.92	0.00	0.00	0.22	0.00
0.55	0.00	0.58	0.53	0.00	0.00	0.90	0.69
0.77	0.00	0.00	0.49	0.67	0.90	0.83	0.17
0.77	0.82	1.00	0.00	0.00	0.00	0.00	0.43
0.00	0.40	0.81	0.00	0.00	0.00	0.43	0.69
0.00	1.00	1.00	0.30	0.20	0.40	0.70	1.00
0.80	1.00	1.00	0.90	0.70	1.00	0.80	0.50
0.60	0.70	0.80	0.60	0.70	0.10	0.20	0.60
0.40	1.00	0.60	0.70	0.00	0.90	0.60	0.10
0.90	0.80	1.00	0.80	0.60	0.80	0.90	0.00
0.40	0.85	0.80	0.90	0.60	0.40	0.60	0.70
0.00	0.00	0.30	0.00	0.90	0.83	1.00	0.90
0.10	0.70	0.70	1.00	0.00	0.00	0.00	0.71
0.80	0.00	0.00	0.64	1.00	0.00	1.00	0.33
0.82	1.00	0.94	0.90	1.00	0.00	0.00	0.00
0.00	0.80	0.00	0.11	0.90	0.50	1.00	0.85
1.00	0.60	0.00	0.00	0.00	0.70	0.50	0.60
0.00	0.80	1.00	0.00	0.00	0.00	0.00	0.00
0.00	0.70	0.88	0.00	0.80	0.00	0.80	1.00
0.37	1.00	0.92	0.90	0.51	0.00	0.90	0.70
0.90	0.90	1.00	0.80	0.70	1.00	1.00	1.00
0.90	0.64	1.00	0.00	0.00	0.00	0.90	1.00
0.90	0.00	0.30	0.90	0.80	0.90	0.00	0.70
0.70	0.00	0.00	1.00	0.70	0.90	0.90	0.92
1.00	1.00	0.81	0.86	1.00	0.93	0.00	0.00
1.00	0.00	0.00	0.00	0.95	1.00	0.00	0.80
0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.70
0.80	1.00	1.00	0.90	0.36	0.00	1.00	0.70
0.00	0.88	0.00	0.00	1.00	0.65	0.00	0.80
1.00	0.80	0.00	0.00	0.91	1.00	0.50	1.00
1.00	0.00	0.00	1.00	0.50	0.00	0.00	0.00
0.90	0.95	0.80	1.00	0.00	0.60	1.00	1.00
0.00	0.00	0.00	0.00	1.00	0.99	1.00	1.00
1.00	0.89	0.60	0.00	0.00	0.00	1.00	1.00
1.00	0.70	0.80	0.00	0.00	0.70	1.00	0.70
0.40	1.00	0.70	0.50	0.90	1.00	0.10	1.00
0.50	0.30	0.94	1.00	1.00	1.00	1.00	0.23
0.35	0.00	0.00	1.00	1.00	0.00	0.29	0.62
0.00	0.37	0.00	0.00	1.00	1.00	0.00	0.00
0.00	0.00	1.00	0.90	0.90	1.00	1.00	0.26
0.00	0.00	0.80	0.81	0.00	0.00	0.00	0.00
0.00	0.51	0.52	1.00	0.00	0.00	0.88	0.90
1.00	1.00	0.23	0.90	0.00	0.47	0.00	0.55
0.96	0.62	0.90	0.80	0.00	0.67	0.78	0.00
0.80	0.80	0.00	0.30	0.00	0.00	0.00	0.00
0.00	0.06	0.00	0.00	0.40	1.00	1.00	1.00

0.57	0.00	0.00	0.00	0.00	0.00	0.58	1.00
0.80	0.90	0.00	0.00	0.00	1.00	0.23	0.00
0.00	0.00	0.00	0.90	0.00	0.33	0.00	0.00
0.00	0.21	0.00	0.39	0.90	0.30	0.00	0.56
0.40	0.77	0.00	0.00	0.00	0.00	0.00	0.00
0.70	1.00	0.00	1.00	0.58	0.30	0.00	0.00
0.39	0.00	0.43	0.50	0.87	0.00	0.00	0.00
0.00	0.84	0.00	0.00	0.00	0.00	0.81	0.62
0.00	0.00	0.00	0.50	0.44	0.00	0.90	0.00
1.00	0.70	0.60	0.00	0.00	0.00	0.90	0.00
0.00	0.00	0.27	1.00	1.00	0.00	0.80	0.00
0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
0.80	0.00	0.90	1.00	1.00			
1716.20	1722.83	1521.58	2050.15	1151.42	2437.18	1026.18	1559.18
1050.51	880.22	1822.36	1115.63	1534.85	1720.62	1842.26	2021.40
1738.32	1457.44	1032.82	1585.72	1435.33	2173.13	2032.46	2025.82
2043.52	1957.26	2185.06	2114.29	1983.80	1915.24	1426.48	1550.33
1211.54	1845.32	1758.22	1572.45	1152.24	946.15	1687.45	1417.63
1793.61	1930.73	1882.07	1851.11	1570.23	1621.10	878.33	1090.32
1583.50	632.12	1045.16	1223.01	1466.29	1638.79	1601.20	1422.06
1152.18	1630.57	1298.21	574.34	1107.05	1643.22	1802.45	1685.24
1422.06	1738.32	1183.20	528.22	1596.77	1395.52	548.48	683.85
1406.58	1017.33	1143.40	1618.89	1466.29	1437.54	1322.53	623.96
778.48	1234.07	1065.20	510.88	1344.65	798.39	708.02	1355.71
1335.80	979.12	1124.93	1111.31	768.15	997.67	642.38	973.61
961.81	950.17	1110.22	1227.44	1278.30	353.60	1180.99	1158.88
729.83	526.36	1225.22	508.67	685.60	479.92	318.15	1032.82
311.10	307.69	304.37	1114.65	1121.28	398.95	394.90	302.99
286.18	838.20	785.12	278.25	275.78	429.05	271.13	268.94
445.24	966.47	382.61	227.79	506.46	944.35	597.13	422.41
253.44	990.80	488.84	950.99	950.99	897.91	988.58	696.65
964.26	478.59	395.88	245.18	829.35	933.29	829.35	245.14
875.79	630.31	685.03	891.27	920.02	248.07	248.91	250.87
252.00	723.49	727.27	424.73	346.28	259.09	352.82	750.29
677.11	266.49	268.57	270.74	531.59	766.31	463.56	277.82
808.53	331.12	285.70	426.84	697.27	398.24	402.38	924.45
1271.67	1253.98	310.51	1278.30	1360.13	415.78	1368.98	1399.94
581.65	1218.59	1125.70	466.07	348.65	1180.99	483.17	1598.99
365.91	375.97	928.87	379.52	384.18	548.48	942.01	398.51
1718.41	1249.55	413.32	957.62	423.45	428.58	433.76	438.98
1537.06	1791.40	793.96	768.02	465.71	402.51	476.65	482.17
1786.97	667.35	1740.53	1189.84	1145.61	2056.79	1835.63	895.70
2043.52	2067.84	2123.14	1466.29	1935.15	2112.08	1983.80	1884.28
783.50	749.73	1276.09	1822.36	1698.51	1751.59	1729.47	1840.05
1729.47	1488.41	605.98	642.57	648.26	2019.19	2552.19	2647.28
1550.33	915.09	2563.24	1147.12	1081.47	698.44	2282.37	1702.93
908.97	1702.93	2379.68	2434.97	2244.77	2302.27	1913.03	800.60
755.90	760.84	765.74	1285.76	1169.94	2134.19	2629.59	2512.38
2490.26	1554.61	1086.12	2485.84	2160.73	2331.02	2136.40	2105.44
827.97	2114.29	1394.55	827.14	843.29	2262.47	2576.51	2634.01
857.39	933.29	2134.19	962.04	870.19	1218.71	1185.22	1716.67
2263.75	1479.74	1731.59	2398.59	1956.48	896.14	1215.19	1217.91
902.16	1760.17	2318.60	2171.11	2326.68	1772.48	1775.08	1997.90
1779.74	2546.51	2600.19	2551.52	2468.41	918.28	1789.10	2199.73
1790.68	2474.54	1239.87	1988.23	1884.72	920.08	2524.98	1961.02

2474.34	2185.06	918.38	1786.98	2194.27	2136.18	2287.01	1236.80
2318.42	2464.61	2101.90	2002.16	2096.37	1511.71	2034.01	1537.28
1748.27	1499.37	1495.89	1555.64	1656.04	1294.01	1491.72	1069.31
1115.09	1212.18	1688.78	1524.90	1270.56	1214.39	1683.31	1641.89
1395.52	854.57	851.10	2405.84	1687.23	1396.62	2250.01	1873.89
2229.06	2005.70	1593.90	1845.58	2260.28	2068.31	803.59	2071.83
2104.56	1930.73	2000.17	1799.80	1050.06	1363.23	1834.30	2176.25
1327.62	1254.37	1417.63	1995.00	1820.15	1477.79	1141.63	1403.56
1831.74	1826.56	1636.14	1400.38	922.01	1470.71	682.91	1605.84
1919.40	1575.98	1645.43	1884.72	1711.78	1647.49	617.04	1054.74
1562.05	1790.29	1757.34	1746.72	1646.98	597.52	591.73	585.94
593.37	574.35	487.44	689.80	890.83	551.21	665.03	539.69
926.66	1263.96	1548.78	1274.32	691.50	1533.96	1093.97	1440.63
951.51	483.12	976.42	1033.23	466.64	1306.39	1390.65	562.41
1176.57	1209.74	1146.27	1172.15	1220.14	1250.88	657.29	979.12
1249.55	767.65	1245.35	758.84	1118.63	1092.75	1060.90	774.28
459.13	573.25	1150.03	1118.63	1078.15	1035.91	714.57	885.75
333.17	329.30	325.51	321.79	668.79	789.54	955.19	1005.39
974.43	904.54	866.50	769.64	903.88	460.45	286.18	506.68
502.70	923.12	672.55	688.47	930.86	924.67	807.68	935.95
723.86	878.00	259.38	768.75	895.70	869.16	825.15	803.25
753.49	250.01	812.76	851.02	299.89	878.89	495.62	245.79
245.43	245.18	349.65	244.96	245.00	398.09	901.45	358.72
950.32	955.85	412.70	483.03	697.76	888.62	615.27	252.00
253.22	1066.21	311.39	257.48	297.46	352.82	262.60	313.83
350.54	268.57	986.15	455.54	242.39	277.82	988.80	736.90
1047.85	288.50	562.85	629.64	613.05	300.57	477.48	672.14
310.51	696.65	429.57	1161.31	1240.71	1311.04	554.80	1270.34
340.43	1243.36	678.87	352.86	1053.38	1323.20	1294.67	1458.55
1361.68	485.89	562.41	1459.21	1528.88	664.93	1260.39	1086.78
607.97	508.22	1325.85	428.58	1243.58	1000.97	444.25	1398.17
454.90	787.55	1737.87	1416.53	809.00	1656.05	1107.13	651.54
1608.05	1282.95	1860.40	515.84	521.52	1553.65	1888.70	1714.21
544.45	1438.14	1358.09	573.35	966.35	1497.45	1413.54	596.53
1005.00	1014.65	1651.31	619.63	846.06	631.13	636.86	1823.68
1711.78	1863.05	1293.12	2037.55	2140.61	954.30	907.53	1555.20
1796.26	2229.29	799.93	1052.94	952.22	1433.78	1584.17	2391.40
2193.24	2273.52	2129.99	2265.34	2226.42	2123.58	1926.30	1408.79
765.73	770.57	775.36	1814.84	1787.86	1232.52	1694.42	2041.08
2413.08	2302.50	1918.51	2628.53	1109.15	2089.08	2495.79	1497.69
1905.96	1295.94	1707.57	2585.14	2517.02	1556.74	1729.47	1675.92
863.95	1173.07	2001.94	1268.35	1318.33	1759.77	2562.14	2278.39
2496.04	2699.27	1687.01	1740.65	1744.92	1911.93	1430.02	2557.49
2628.48	1137.65	2071.16	2344.74	2684.66	2095.49	2549.31	2593.54
2595.31	2558.38	2165.60	1183.20	2705.01	1622.46	2559.95	2571.42
2296.08	2621.85	2209.61	920.08	1151.58	944.35	2294.76	2355.79
1876.54	2197.44	2527.86	2465.49	915.24	1635.48	1963.46	2609.91
2659.23	1229.90	2045.16	1511.71	2154.96	1966.63	2544.81	2538.63
2532.20	2542.03	1480.46	1936.28	2454.74	2496.17	2357.98	1701.19
870.71	2221.18	2405.78	2444.79	2052.98	1156.11	1657.22	2358.61
2348.51	1636.02	2374.28	2131.40	1813.55	1973.25	1795.87	2090.05
2260.27	1573.16	1340.83	1912.30	1326.00	2517.11	2331.91	1198.02
2432.32	1042.55	822.05	1287.15	1918.34	761.69	1626.85	1983.58
2400.69	2334.56	2353.14	2298.07	2198.99	2109.42	2134.19	2081.11
1176.01	1792.71	1745.39	1780.95	1875.21	1917.68	2123.80	2193.24

1776.93	1073.81	1822.40	1700.39	1778.01	1588.80	1471.28	1461.99
1284.50	1295.55	603.32	997.02	1152.19	2581.04	1619.38	983.12
705.33	939.03	1219.02	551.22	910.13	629.91	533.96	1982.85
522.52	1717.75	511.15	1873.22	1570.23	1700.50	1537.50	653.60
1045.94	1240.66	1302.35	1211.77	1197.48	1248.65	1022.26	583.42
1034.36	1512.29	1583.50	1350.18	414.19	968.68	404.24	1579.30
1307.72	397.20	620.57	380.33	1317.01	1408.12	1430.24	1342.88
357.89	1212.84	1361.24	1301.30	1066.87	956.94	951.94	940.88
837.53	1212.18	519.95	294.36	605.76	307.70	670.56	1193.82
1169.93	1180.99	1106.90	339.92	286.18	988.36	1100.27	376.44
275.79	273.41	1037.46	588.59	1036.13	1046.75	1036.13	487.66
610.18	257.75	498.90	254.78	235.09	454.70	920.25	343.24
816.08	364.47	247.46	997.65	925.77	642.25	1097.84	1134.55
408.83	244.96	245.00	1142.51	409.44	245.72	246.16	316.92
247.34	296.80	347.00	1112.65	395.88	1079.04	1179.00	860.75
814.97	1035.25	259.09	1169.27	599.34	1288.26	767.65	747.08
451.75	576.78	1154.68	1223.46	613.57	494.29	285.70	561.75
692.01	1352.39	1374.95	406.63	303.80	1257.29	794.88	611.35
1587.26	434.47	771.41	1579.30	1431.79	1581.51	1205.10	344.50
353.19	1640.78	512.43	313.16	984.27	1654.27	897.11	1751.59
840.79	509.55	532.58	451.17	403.39	408.32	413.31	565.97
893.26	1772.37	1831.65	1919.23	444.24	750.11	454.89	2082.66
869.60	471.15	1957.49	2131.98	1393.53	1127.69	1285.60	1148.93
2144.37	1902.42	2171.57	1261.59	543.83	1612.25	736.56	918.07
1438.87	2204.52	567.55	878.45	2411.53	1138.94	2346.73	995.34
2530.07	822.66	830.47	2529.41	625.38	1974.96	1240.04	869.30
648.26	1861.72	2781.53	2675.15	582.76	2708.77	1409.45	1576.20
692.97	1267.25	2887.24	1381.00	709.26	1991.54	1210.03	2068.95
1432.33	740.75	1728.58	1457.00	2097.48	2929.48	2256.27	2678.25
1633.71	907.20	1673.74	2183.51	3091.81	3001.36	1086.11	1055.82
3136.27	3119.46	2213.59	823.94	1381.51	1911.04	2968.19	2916.44
1410.56	2680.02	3031.00	853.98	857.38	860.71	3095.35	2958.46
2858.27	2851.86	2818.46	2457.53	1192.72	1549.89	2068.95	2889.45
1893.13	893.95	1212.35	1772.82	900.25	1505.31	997.88	2722.92
2981.46	2088.63	1532.19	1521.13	912.88	1236.54	1854.65	2529.63
2958.23	2708.32	1242.30	2460.62	1885.61	2899.85	2201.19	2201.57
2588.23	2868.22	1534.98	3143.57	3106.85	3067.05	2577.18	1804.89
1487.52	1113.32	1651.18	1615.57	1196.92	2874.64	2782.19	2702.80
1850.44	3046.92	2904.05	2090.18	1218.37	3103.76	896.52	2562.36
2785.07	2741.06	2531.17	2626.72	2315.77	2724.25	876.58	1426.70
1177.95	2455.32	1779.89	2158.30	3027.68	2044.88	2873.09	2407.99
1131.57	1992.24	1982.83	824.60	1110.12	816.40	1777.58	1933.26
803.58	1556.10	2260.80	2022.80	2011.02	1709.05	2397.37	2627.38
834.88	1287.16	2626.27	730.49	1262.71	1017.03	2419.05	2231.95
2413.52	2337.22	726.10	1218.81	2000.83	1530.63	1351.15	931.35
682.90	677.36	671.79	901.25	660.53	1859.02	1854.98	1790.91
1715.79	1688.66	1498.83	839.61	614.86	609.09	603.31	1429.78
1690.65	1674.10	1690.69	2121.81	1763.09	2008.57	1896.00	1919.89
2119.15	2011.01	1173.47	472.10	1021.28	1283.50	997.60	1153.21
602.24	2013.22	1871.68	429.47	956.52	599.79	1094.74	1321.87
1154.96	777.57	1061.21	1048.34	384.99	1058.37	1073.45	1060.45
713.96	362.25	597.16	353.60	349.37	345.22	917.64	337.11
333.17	329.30	325.51	435.33	680.73	314.58	311.10	1041.88
1277.20	1284.27	403.11	428.83	931.97	390.97	769.82	427.50
280.80	1074.17	275.78	760.86	774.65	588.59	717.79	751.74

731.67	261.10	721.81	616.76	256.22	557.61	422.89	491.07
251.06	640.02	484.95	248.21	481.83	700.56	246.25	845.71
477.90	586.67	245.02	1044.98	586.26	245.14	245.38	1031.27
471.51	301.22	970.89	1014.24	336.73	249.84	226.25	340.92
253.22	248.58	251.24	501.35	567.04	1154.90	262.60	441.33
518.89	268.57	451.76	1056.04	341.91	748.40	1154.01	382.61
1180.33	1181.88	1150.47	294.37	297.43	899.01	411.01	1240.04
420.07	313.98	429.57	321.16	345.89	328.65	1056.04	551.35
1353.72	1332.71	934.40	1446.61	357.15	363.59	1416.75	449.84
374.92	379.52	384.18	423.74	1540.60	1491.72	403.39	1547.23
804.79	1190.72	1073.51	1385.12	1326.74	438.98	444.25	449.56
759.04	904.58	1026.38	706.55	949.32	397.61	1123.80	601.01
449.56	1223.69	1007.37	465.71	476.65	482.17	659.81	493.29
498.89	1864.16	1885.17	1166.62	705.56	2086.42	519.06	728.79
544.46	550.22	1938.69	1146.93	567.56	573.35	573.35	1689.88
2244.99	1695.41	1470.27	1318.21	1635.78	1753.92	1770.39	1217.73
861.58	2640.43	1823.99	877.01	1868.40	659.59	1702.98	1866.83
676.41	1137.91	1849.34	937.50	1359.96	1174.45	1815.69	1905.79
1509.95	940.20	2856.50	2415.95	745.85	750.90	1022.62	2920.42
2904.27	1042.48	775.36	780.09	1309.43	2902.50	793.92	2604.16
1339.58	2539.14	2961.33	2918.20	2853.18	2375.70	1413.21	3038.96
3113.49	3051.34	1881.63	1145.78	2835.93	1424.93	2635.34	2741.06
2626.49	1647.86	867.11	870.19	873.18	876.08	878.90	881.63
884.28	886.83	889.29	1487.81	893.95	2294.13	1498.78	3171.65
1917.46	905.70	1343.54	1516.50	1881.19	2767.59	2845.66	914.02
915.07	916.02	916.87	2802.76	2716.06	2499.99	1823.68	919.65
2683.77	2750.12	2888.57	1191.83	3109.73	2748.35	2547.76	2667.63
2103.23	2919.53	2043.74	2150.34	2968.19	2877.07	1235.27	1811.30
1607.83	3089.83	2874.64	2184.40	1836.95	1220.93	1431.57	2526.75
1574.88	1453.02	892.07	2659.45	2607.70	2803.42	2764.28	2223.98
2879.72	2249.86	1422.50	2565.90	2503.53	2638.44	2159.18	1696.52
2822.89	847.55	847.55	2270.15	2338.20	2141.27	1621.15	828.65
1375.97	820.57	816.42	1098.81	1573.16	803.58	1333.47	1901.59
1729.31	1529.59	1302.98	1698.72	1502.03	2175.97	2119.67	1656.20
1924.50	746.72	1623.11	1434.07	2089.48	982.30	1577.58	2031.06
960.76	1175.99	699.37	1157.85	1506.68	1130.23	908.84	3224.51
1691.04	996.77	1083.30	870.63	1395.96	1618.22	1370.87	1208.45
1966.55	2467.92	1622.90	2622.51	800.52	928.87	580.14	777.01
1613.88	1095.78	1084.53	551.21	2223.76	1821.25	1970.31	1019.55
871.84	2233.94	2072.27	1066.65	499.86	1580.63	488.67	1299.76
783.57	472.10	577.01	623.96	455.82	1878.53	1271.88	1287.37
2042.19	716.60	1700.50	1202.22	560.33	979.12	404.24	613.94
394.50	852.92	1713.77	1658.48	1709.12	722.70	877.39	792.80
1535.73	590.00	349.37	1398.61	1625.97	1595.89	1532.20	1319.66
1083.02	626.57	457.36	1182.98	519.09	1432.23	507.86	301.13
784.01	294.89	377.52	390.97	683.38	1087.22	379.89	541.79
373.10	369.88	271.13	448.74	519.57	251.68	243.27	265.39
350.90	545.60	256.22	1213.28	1201.34	1090.76	339.64	250.01
336.94	709.17	665.65	333.89	410.88	332.52	1145.61	477.40
477.09	1123.05	1125.92	245.14	477.80	332.43	333.02	314.05
247.34	248.07	408.48	370.66	250.87	318.91	1173.25	1203.11
231.33	1085.89	1209.30	1119.73	305.86	264.50	1035.47	448.13
1254.20	1250.88	458.46	1200.23	930.64	1323.64	1382.91	561.76
486.21	294.37	297.43	406.64	794.18	1364.11	310.51	611.36
1420.51	458.02	1391.54	1238.72	332.50	699.75	1365.22	574.84

348.65	588.78	1637.69	1367.65	481.69	370.38	897.14	379.52
384.18	1806.66	1673.52	398.51	965.26	1821.47	1857.52	1783.21
572.86	513.53	723.76	438.98	2065.41	983.89	454.90	415.34
2134.86	2013.44	1948.20	2014.55	487.71	517.96	498.89	504.52
452.49	721.64	1141.40	2396.27	1526.45	2282.15	544.45	550.22
555.99	1094.30	947.01	1729.25	1908.17	1138.96	590.73	1161.52
1541.93	608.10	1195.29	1482.70	846.06	631.13	1807.77	642.57
1551.21	1759.10	810.77	659.94	670.83	676.41	681.96	1898.21
1819.92	2540.91	2316.87	709.25	714.61	2817.14	1962.35	1558.73
2480.09	1926.97	2905.38	2645.96	757.91	760.84	2250.74	2969.29
2648.83	2776.88	789.37	1074.06	1080.13	802.83	807.19	2204.30
2329.48	2783.52	1374.82	1355.49	2812.49	2846.11	2809.40	2749.02
2674.04	2010.56	1997.74	857.39	2808.95	2584.92	1212.84	1177.24
795.29	876.09	2417.06	1192.73	3073.90	2162.72	3094.03	2041.53
1023.08	1916.35	1215.19	1502.13	2752.11	1829.88	1550.77	3008.88
1516.50	1509.64	2783.30	2455.65	2885.69	2857.83	2827.53	2913.78
2672.05	2616.32	2946.07	2842.13	2354.29	919.90	920.06	920.12
2070.06	3025.69	2825.10	3025.69	3089.83	2814.70	2047.50	917.01
1528.70	1238.19	2805.63	2699.04	2795.90	910.53	2509.72	3229.82
3146.66	2793.69	3176.96	1218.36	2956.91	2383.88	2952.93	2764.06
2587.57	2468.14	2680.90	2283.70	1189.67	2363.54	2829.30	1985.13
2225.53	1531.53	2684.44	2853.63	2779.10	2885.47	2862.47	2664.53
1639.68	836.43	2604.60	828.65	1853.32	2217.35	2625.17	2549.31
2092.61	2517.91	2018.75	1912.15	1142.29	2881.71	2727.12	2612.12
2336.11	1204.44	2269.54	2450.89	2704.12	2280.38	1213.28	1592.35
2295.42	2609.91	2155.87	1724.38	1784.98	2437.40	946.15	2223.54
688.43	1329.73	1498.36	1965.67	2370.39	2043.96	886.00	2308.24
2361.55	2174.67	1733.01	626.38	620.62	2225.53	1016.31	1787.41
2210.71	2151.00	2000.39	580.14	2193.91	1748.05	2261.36	2194.57
919.73	1916.13	730.12	533.95	528.22			

4150.00	4155.00	3514.00	3180.00	3158.00	2650.00	2650.00	2650.00
2957.00	2851.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2427.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	[illegible]	[illegible]	2120.00	2120.00	2120.00	2120.00
2120.00	2559.00			3180.00	3180.00	3180.00	3180.00
3180.00	3180.00			750.00	750.00	750.00	750.00
750.00	750.00			1166.00	2067.00	2120.00	2120.00
2120.00	2120.00	← flow ofs 1 unit		2014.00	2120.00	2120.00	2120.00
2120.00	2120.00			3180.00	2968.00	2544.00	3180.00
3180.00	3180.00			2756.00	2120.00	2120.00	2120.00
2120.00	2120.00			3180.00	3180.00	3180.00	3180.00
3180.00	3577.50			4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	← 2 units		4240.00	4240.00	4240.00	4240.00
4240.00	4240.00			3180.00	2120.00	2120.00	2120.00
2120.00	4240.00			4240.00	4240.00	4240.00	4240.00
4240.00	4240.00			4240.00	4028.00	4240.00	4240.00
4240.00	4240.00			4240.00	4240.00	4240.00	4240.00
4240.00	4240.00			4240.00	4240.00	4240.00	4240.00
2915.00	2544.00			4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	[illegible]	[illegible]	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	3922.00	3339.00	2650.00	2650.00	3180.00	2809.00	2915.00
3339.00	3180.00	3657.00	3657.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	1722.50	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	2162.40	2872.60
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1213.70	1060.00	1060.00	1060.00
1060.00	1060.00	1150.10	2120.00	2120.00	1855.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1547.60	2120.00	2120.00	2120.00	2120.00	2120.00	1568.00	1590.00
1590.00	1833.80	2120.00	2120.00	2120.00	2120.00	2120.00	2957.40
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	2517.50	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2517.50	2957.00	1945.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1834.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2608.00
2740.00	2054.00	1945.00	2120.00	2120.00	2120.00	2120.00	2054.00
1988.00	2120.00	2120.00	2120.00	2120.00	2120.00	1897.00	2009.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	1811.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1104.00	1060.00

1060.00	1590.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00	1060.00
1060.00	1060.00	1060.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	1943.00	1899.00	2120.00	2120.00	2120.00
1458.00	2429.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	2805.00	2650.00	2650.00	2650.00	2650.00
2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00
2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2473.00	1590.00
1590.00	1590.00	1590.00	2650.00	2650.00	2650.00	2650.00	2650.00
2650.00	2650.00	2341.00	2650.00	2650.00	2650.00	2650.00	2650.00
2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00
2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00	2650.00
2562.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	2650.00	2650.00	2650.00
2650.00	2783.00	3180.00	3180.00	3180.00	3180.00	3092.00	3180.00
3180.00	3180.00	3180.00	3180.00	3357.00	3710.00	3710.00	3578.00
3688.00	3710.00	3621.70	1965.00	3180.00	3180.00	3180.00	3180.00
2208.00	1590.00	1590.00	1590.00	1590.00	1590.00	2893.00	2937.00
3180.00	3224.00	3710.00	3710.00	3710.00	3710.00	3710.00	3666.00
3622.00	3710.00	3754.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3843.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	3975.00	4240.00
4240.00	4240.00	3931.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3622.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3578.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3092.00	3180.00
3180.00	3180.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3511.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3666.00	3533.00	3578.00	3578.00	3666.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
4108.00	3754.00	3710.00	4196.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4152.00	4240.00	4240.00	4196.00	3710.00
3710.00	3710.00	3710.00	3931.00	3931.00	3710.00	3710.00	3710.00
3710.00	3401.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3313.00	3313.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	2253.00	1590.00
1590.00	1590.00	1590.00	1590.00	2783.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	2253.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	2871.00	2650.00	2650.00	2959.00	3180.00
3224.00	3401.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00

3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	2341.00
1723.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1723.00	2120.00	2076.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	2209.00	2694.00	3180.00
3180.00	3180.00	3180.00	2959.00	3710.00	3666.00	3710.00	3710.00
3710.00	3710.00	3843.00	4240.00	4240.00	4240.00	3843.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3843.00	4240.00	4108.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3931.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4108.00	3710.00	3048.00	2120.00	2120.00	2120.00	2120.00	2120.00
1943.00	2650.00	2959.00	3180.00	3710.00	3710.00	3710.00	4108.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4019.00	3710.00	3798.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3622.00	3489.00	3710.00	3710.00	3710.00	3754.00
4240.00	4240.00	4019.00	4108.00	3445.00	3401.00	3578.00	3710.00
3710.00	3710.00	3489.00	3180.00	3180.00	3180.00	2783.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3619.90	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3927.30	4240.00	3842.50	3710.00
3710.00	3710.00	3710.00	3710.00	3747.86	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4134.00	4240.00	4240.00	4240.00
4240.00	4028.00	4240.00	4240.00	4240.00	4134.00	4240.00	4240.00
4240.00	4059.80	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	3932.60
3710.00	3710.00	3180.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3710.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3270.10	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00

3180.00	3180.00	2517.50	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
1987.50	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1722.50	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	3047.50	1987.50	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00	2120.00
2120.00	2120.00	2120.00	2120.00	1540.00	1540.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	3180.00	3312.50	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	1987.50	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00	3180.00
3113.75	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00	1590.00
1590.00	1590.00	1590.00	1590.00	1590.00	1634.00	2120.00	2120.00
2120.00	1811.01	1590.00	1590.00	1590.00	1590.00	1590.00	1899.15

3180.00	3180.00	1811.01	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3621.50	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3710.00
3710.00	3710.00	3710.00	3710.00	3710.00	3710.00	3842.50	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	3710.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4019.00	3710.00	3710.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4107.50	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00	4240.00
4240.00	4240.00	4240.00	4240.00	4240.00			

7.83	7.84	7.34	9.14	9.23	11.01	11.01	11.04
9.86	10.22	13.75	13.74	12.68	12.06	12.47	11.99
12.40	13.54	13.60	13.61	13.63	11.92	13.63	13.71
13.71	13.77	13.76	13.45	13.33	13.56	13.77	14.05
14.05	14.16	14.18	14.17	14.17	14.20	13.78	14.18
14.17	14.12	14.11	14.18	14.18	14.12	14.16	14.17
14.12	11.58	8.99	9.35	9.45	9.46	9.46	9.31
9.39	9.45	8.62	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	1.37	8.78	1.82	8.15
12.66	14.18	14.20	14.19	13.82	14.15	14.15	14.17
14.16	14.20	11.58	9.46	9.47	10.11	11.77	9.42
9.43	9.46	7.48	4.44	10.78	13.98	14.07	14.05
14.06	14.08	10.81	8.89	9.12	8.99	8.73	9.05
9.06	8.11	6.88	6.86	6.85	6.88	6.88	6.86
6.84	6.86	6.87	6.84	6.89	6.87	6.89	6.93
6.94	6.68	6.93	6.89	9.13	13.77	13.79	13.82
13.75	6.92	6.89	6.75	6.90	6.94	6.90	6.91
6.92	6.93	6.91	6.94	6.97	3.80	4.65	6.90
6.90	6.90	6.85	6.83	6.87	5.96	6.85	6.85
6.89	6.28	6.25	6.83	6.90	6.91	6.87	6.87
10.00	11.48	7.16	6.88	6.93	6.87	6.88	6.89
6.89	6.86	6.88	6.88	6.88	6.89	6.89	6.91
6.94	6.92	6.91	6.91	6.92	6.92	6.92	6.93
6.95	4.62	0.0	0.0	0.0	0.0	0.0	0.0
1.56	8.80	7.96	7.99	13.78	13.82	13.82	13.92
13.90	13.85	13.84	13.89	13.91	13.85	13.80	13.80
13.84	13.83	13.81	13.87	13.91	13.01	9.48	13.96
14.13	14.16	17.33	18.79	18.51	18.67	18.61	18.65
18.53	18.74	18.58	18.57	18.57	18.62	18.69	18.51
18.53	18.28	17.98	18.29	16.66	10.24	13.47	10.21
17.05	11.32	18.21	18.38	18.52	18.48	18.39	11.50
0.20	0.04	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.27
13.09	16.33	4.54	13.81	14.18	14.15	13.75	9.99
9.45	9.44	9.02	0.05	8.08	9.43	11.96	14.13
12.92	14.19	14.19	14.20	14.14	14.20	14.18	14.19

14.18	14.20	14.20	14.20	12.52	9.66	7.00	14.60
18.90	18.90	18.90	18.90	18.90	16.50	18.90	18.90
18.90	16.40	14.20	14.20	14.20	14.00	14.10	14.00
14.10	14.00	13.90	13.90	13.80	14.00	14.10	14.10
14.10	14.10	13.50	14.10	14.20	14.20	14.20	14.20
14.20	14.20	14.20	14.20	14.20	14.20	14.20	11.50
11.00	14.10	12.30	14.10	14.20	14.20	14.20	14.70
15.10	14.20	14.20	14.20	14.20	14.20	10.30	8.70
14.20	14.20	14.20	14.20	14.20	14.20	14.20	14.20
14.20	14.20	14.20	14.20	14.20	14.20	14.20	14.20
14.20	14.20	14.20	14.20	14.00	14.20	14.10	14.20
14.20	14.20	14.20	14.20	14.20	14.20	14.20	13.80
13.60	13.30	13.40	13.20	13.00	13.00	13.10	12.80
12.80	12.60	12.60	12.30	12.10	11.30	6.10	4.90
3.30	5.00	5.00	7.00	8.30	10.70	11.60	8.60
5.90	12.60	3.80	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.11
0.16	1.76	0.39	2.79	3.00	2.90	2.80	3.05
3.18	2.35	3.97	2.80	5.99	7.92	10.26	11.03
11.06	11.35	11.29	11.36	11.36	11.24	11.33	11.36
11.36	11.34	11.36	11.35	11.36	11.36	7.60	0.0
0.0	0.0	2.46	4.73	10.59	11.34	11.34	11.34
11.36	11.36	12.83	11.35	11.36	11.36	11.34	11.36
11.36	11.36	11.36	11.10	11.36	11.35	11.36	11.35
10.75	4.23	10.60	11.33	11.36	11.36	11.36	11.36
11.75	18.92	18.92	18.93	18.93	18.93	18.93	18.93
18.93	18.93	0.0	0.0	0.0	4.15	9.25	18.19
18.93	18.93	18.93	18.93	18.92	18.93	18.93	18.93
18.91	18.70	18.93	18.93	18.93	18.93	18.93	18.93
18.93	18.93	18.93	18.93	18.93	11.36	11.36	11.36
11.36	10.82	9.47	9.47	9.47	9.47	9.74	9.47
9.21	9.47	9.47	9.47	8.97	8.11	8.11	8.41
8.16	8.11	7.96	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.06	1.96
7.10	7.66	1.87	2.06	2.36	4.17	7.61	6.18
3.31	6.46	7.96	8.11	8.11	4.15	4.71	7.94
8.11	8.11	8.11	8.09	8.09	5.04	1.67	5.24
8.11	8.11	8.11	7.82	7.04	7.08	7.08	7.10
7.08	7.06	7.03	7.04	6.93	7.03	7.50	7.04
7.04	7.03	7.58	8.07	5.67	2.24	6.61	8.04
8.04	8.08	8.25	8.09	8.08	8.09	8.09	8.05
8.09	8.40	9.43	9.44	9.45	9.43	9.46	9.45

9.41	9.25	8.31	9.44	9.45	9.45	9.47	9.46
9.45	9.46	8.10	8.11	8.10	8.10	8.10	8.10
8.04	8.51	8.10	8.10	8.10	8.10	8.10	8.10
8.10	8.10	8.17	8.39	8.32	8.32	8.17	6.76
2.12	5.26	10.05	10.27	10.35	10.96	11.36	10.65
10.76	11.95	12.11	10.74	10.40	11.27	12.63	12.61
12.57	11.00	7.48	8.55	10.28	11.33	6.58	7.58
10.28	12.62	12.62	14.11	14.10	11.99	11.94	13.06
15.86	17.68	18.82	18.02	18.55	12.22	13.78	18.63
12.25	10.78	15.64	18.86	18.86	18.68	18.79	18.85
15.79	18.65	18.70	18.48	18.18	9.48	9.40	9.38
9.39	9.37	9.38	9.33	9.16	9.06	12.70	17.75
17.62	17.27	17.09	17.02	9.58	8.31	8.26	8.17
8.12	8.05	8.01	11.14	7.94	7.74	7.78	9.24
11.44	14.80	7.45	7.31	7.26	7.21	7.05	6.96
6.77	6.73	6.63	7.29	7.84	8.37	14.65	15.83
15.59	14.72	15.59	11.90	13.53	15.35	15.27	15.23
15.17	15.17	15.10	15.07	14.98	14.93	9.75	12.84
17.43	11.20	0.57	11.98	18.90	18.91	18.93	18.04
3.91	17.67	18.90	15.49	16.01	18.93	18.93	18.93
18.93	18.63	16.79	18.51	18.58	18.55	18.24	0.11
6.51	18.38	18.93	18.93	18.93	18.93	18.93	18.93
18.90	18.93	18.93	11.39	15.26	18.93	18.93	18.92
18.93	18.93	18.93	18.93	18.93	18.93	18.90	18.93
17.48	14.20	14.50	18.93	18.93	18.89	18.85	18.93
18.93	18.93	18.93	18.93	18.89	18.93	17.96	18.93
18.93	18.93	3.70	11.73	18.93	18.93	18.93	18.93
18.93	18.93	18.91	18.93	18.93	18.93	18.93	18.93
18.91	18.92	18.93	18.93	18.93	13.80	11.17	9.47
5.12	9.03	8.78	10.17	8.11	6.18	1.60	3.75
8.05	7.98	7.83	7.19	7.36	8.88	10.09	10.55
10.52	11.26	13.80	15.17	16.05	15.55	14.17	14.07
16.22	16.23	16.23	16.19	15.87	13.80	16.21	16.19
15.32	13.95	14.20	14.10	14.20	14.20	14.20	14.20
14.66	15.60	9.87	14.20	14.20	14.20	14.20	14.20
15.48	11.36	10.17	9.41	8.11	9.06	15.93	14.62
9.40	12.19	14.20	14.20	14.20	14.20	14.20	14.17
14.18	14.89	16.23	15.82	16.23	16.23	15.84	13.87
16.23	16.23	16.63	17.26	16.15	16.06	16.10	15.93
7.11	14.05	14.73	11.60	8.72	8.77	8.42	8.11
7.05	4.52	8.60	6.96	9.46	9.46	10.82	8.11
8.11	5.33	8.11	8.11	8.11	8.11	8.11	8.11
4.99	8.11	4.90	4.74	8.11	8.11	8.11	8.11
8.11	8.11	8.11	8.11	8.11	8.11	8.11	8.11
8.11	8.11	8.11	8.11	8.11	8.11	8.11	8.11
7.43	1.33	6.35	8.11	8.11	7.75	8.11	8.11
8.11	8.46	11.50	11.38	12.23	7.90	6.72	8.11
8.11	8.12	8.14	8.37	11.39	13.81	13.86	13.40
9.99	8.61	11.68	11.43	12.01	14.20	14.20	14.20
14.17	7.71	14.20	14.20	13.11	12.31	13.07	14.11
14.10	14.54	13.80	13.99	13.98	13.06	7.10	7.12
11.67	13.88	13.99	14.08	14.09	13.70	11.76	13.77
14.05	14.20	14.20	14.01	14.17	14.15	7.80	13.72
13.70	7.14	6.39	6.82	6.83	6.95	6.82	7.35
7.87	7.87	9.17	7.90	7.91	7.88	7.87	7.87

7.85	7.86	7.81	7.84	7.87	7.88	7.97	7.88
7.87	7.88	7.74	10.06	12.89	10.31	15.43	15.26
15.98	15.98	15.99	16.03	15.56	15.54	15.98	15.74
16.16	16.23	16.23	16.23	16.23	18.93	18.93	18.93
18.93	18.93	18.93	18.93	18.93	18.88	18.89	18.93
16.23	18.92	18.93	18.84	18.84	18.78	18.84	18.84
13.95	17.35	18.88	18.66	16.89	18.84	18.84	18.84
18.84	18.84	18.86	18.81	18.89	18.93	18.37	13.14
18.52	16.62	18.84	18.84	18.87	18.91	18.88	15.93
9.37	9.41	9.72	12.86	14.05	14.07	13.57	10.20
14.75	18.86	18.84	18.58	18.62	18.52	18.48	18.55
18.53	18.36	18.35	18.66	18.39	10.90	13.32	18.09
18.90	18.92	18.91	18.93	18.93	18.88	16.18	12.52
18.80	18.93	18.91	18.91	17.72	18.83	18.93	18.93
18.91	18.90	18.93	18.93	18.93	18.93	18.93	18.93
18.93	18.93	12.20	12.31	10.04	12.00	16.12	18.93
18.93	18.53	18.63	18.92	18.91	18.93	18.93	18.61
9.55	9.47	11.96	18.93	18.93	18.93	18.93	18.93
18.93	18.93	18.93	18.93	14.20	14.16	14.20	14.20
14.20	14.20	14.20	14.20	14.20	14.20	14.20	14.20
14.20	9.47	3.02	7.31	13.38	14.20	14.20	14.20
15.15	18.93	18.93	18.93	18.90	18.93	18.93	18.93
11.75	7.02	11.87	10.22	15.02	16.14	18.93	18.93
18.85	18.93	18.93	18.93	18.93	18.93	18.60	1.06
6.83	18.93	18.93	18.93	18.93	18.93	7.57	0.03
0.03	0.03	0.03	0.03	0.29	5.56	12.16	2.46
0.03	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.01	0.37
0.37	0.0	0.0	0.0	0.0	0.0	0.0	0.27
2.96	5.36	5.68	5.68	5.68	4.65	9.57	14.77
16.28	18.43	18.69	14.20	13.39	13.71	14.11	14.01
14.16	14.20	14.20	14.20	14.20	14.14	2.69	8.80
14.19	14.20	13.82	0.30	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.57	0.0	0.0	0.0	0.0	0.0	0.0	0.16
2.86	5.68	16.66	18.93	18.93	18.90	18.93	18.93
18.88	18.84	18.93	18.93	18.93	18.93	18.73	18.61
18.76	18.93	18.93	18.90	18.73	18.27	15.87	00.00
13.56	18.91	18.84	18.88	18.10	18.48	18.93	18.93
18.90	18.90	18.93	18.93	18.93	18.93	18.93	18.93
18.90	18.93	18.93	18.93	18.93	18.93	18.93	18.88
18.93	18.93	18.93	18.93	18.93	17.66	18.91	18.91
18.93	17.85	18.93	18.93	18.93	18.86	18.93	18.93
18.93	18.93	18.93	18.93	18.90	18.93	18.93	18.93
18.93	4.27	3.57	6.11	9.47	9.47	9.47	9.47
9.47	9.50	9.51	9.58	9.55	9.47	9.47	9.47
9.47	9.45	9.47	9.47	5.56	5.42	9.53	9.61
9.74	6.96	8.50	9.47	9.47	9.47	9.47	9.45
9.47	9.47	9.47	9.38	9.47	9.47	9.42	9.47
9.47	9.47	9.47	15.15	18.93	17.35	15.60	18.93
18.93	18.93	18.93	18.93	18.89	18.93	18.68	1.97

0.17	0.00	0.13	6.67	18.93	18.93	18.93	18.93
18.93	9.11	5.68	11.02	18.93	8.00	17.27	18.89
6.44	16.71	18.93	18.93	18.93	18.93	18.93	18.93
18.83	18.93	18.93	18.93	18.93	18.93	18.93	18.93
18.93	18.93	9.47	9.47	9.47	9.47	9.47	9.47
9.47	9.47	9.47	9.47	9.47	9.42	9.47	9.47
9.47	9.47	9.47	9.47	9.42	6.31	0.00	2.05
8.68	9.47	9.47	9.56	9.56	9.53	10.60	9.47
9.47	9.47	9.47	9.48	12.11	12.29	12.30	14.37
17.68	18.85	18.42	18.69	18.91	18.93	18.93	18.93
18.80	18.93	18.93	18.93	18.93	18.93	18.91	9.73
9.67	18.93	18.93	18.93	18.93	18.93	18.86	18.93
18.93	18.93	18.93	18.93	18.93	18.93	18.93	18.93
18.93	18.93	18.93	18.93	18.70	18.93	18.93	18.93
18.93	18.93	18.88	18.71	18.93	18.93	18.93	18.93
18.93	18.93	18.74	18.93	13.61	0.64	2.09	6.17
13.05	16.62	18.93	18.93	18.93	18.93	18.93	15.84
9.36	9.47	16.62	8.11	8.11	8.11	8.08	0.80
0.65	5.32	7.24	10.55	10.55	10.54	10.53	10.40
10.93	7.91	11.75	7.65	16.21	16.13	16.21	16.23
16.23	16.23	14.29	8.11	8.11	8.07	8.12	8.83
9.56	14.01	14.20	14.19	14.08	11.01	11.57	14.20
13.83	14.17	14.10	14.20	14.20	14.19	14.20	14.20
14.09	13.96	12.94	7.46	10.16	13.79	14.20	14.19
14.12	11.63	12.41	14.20	14.20	14.20	14.20	11.29
8.25	12.98	14.20	14.20	16.23	14.16	7.68	10.68
14.20	14.20	14.20	14.20	14.17	14.10	14.20	14.20
14.20	14.20	14.98	16.22	16.11	13.87	13.97	12.41
14.20	14.20	14.20	14.00	13.15	14.20	14.20	14.20
14.20	14.20	13.57	14.20	14.20	14.20	14.20	14.20
13.97	14.13	14.12	14.20	14.20	14.20	14.20	14.20
14.15	14.13	14.20	14.20	14.20	14.20	13.90	13.77
13.76	14.20	14.20	14.20	14.20	14.20	14.18	14.11
14.20	14.20	13.75	7.10	8.49			

```
 NORTH ANNA COOLING POND SIMULATION - JULY 26, '78 THROUGH SEPT. 30, '83
   07   26 1978    4
   16   17    2    1    1    1    2  100    2    2    5    1
250.          4.          1.       1893.0     80.0     760.
   .234        .50     62.4          1.0       1.0
    2    1    1    1    2
    2
  365.0      365.0      365.0
365.
55.00      57.10      60.00      61.90      68.00      72.80      74.90      76.30
77.20      77.90      79.00      80.40      82.20      82.70      82.80      82.80
350000.    350000.       0.90       1.75
  00.0      2.0
   12
234.0
   39
    8    8
10.0       10.0       190.0      190.0      5.0
 825.      1412.      2147.      2763.      3438.      4200.      5000.      5000.
34400.  34400.  34400.  34400.  34400.  34400.  34400.  34400.
 250.
    3
0.59       0.5        0.75
 1893 1893
1.0        1.0
   30   31    0    0
   10   10
3600.0     103.9      25.0       12.1       25.0       170.0      1.0        0.0086
25.0       162.5      1000.0     212.12     0.0        0.0        25.0
    0    1
 0.0        0.0        0.0        0.0        0.0        0.0        0.0
25.0       162.5      1200.0     65.66      356.57     571.71     25.0
    2   10
 704.04    665.65      800.0      700.0      15.0       23.0       17700.0    16300.0
25.0       162.5      1000.0     19.2       23.23      620.20     25.0
    2    4
 87.88     76.77      500.0      500.0      12.0       12.0       5000.0     3000.0
     15.        90.
1800.0     60.0       20.0
    1
 4231.
 5000.
 40.
 2000.
    7       18.0  160    0    1    0
08/17/8109/17/8111/19/8101/06/8202/08/8202/26/8203/01/8205/12/8206/02/82
06/16/8206/29/8207/06/8208/05/8208/31/8209/07/8210/04/8211/02/8212/05/82
01/04/8301/31/8303/02/8304/05/8306/02/8306/13/8307/11/8307/18/8307/28/83
09/01/8309/07/8309/22/83
   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   1   0   0   0   0   1   0       0   1
```

Iselet (16)

Iselet (17)

p. 32.