
TPDWR2: Thermal Power Determination for Westinghouse Reactors, Version 2

User's Guide

**U.S. Nuclear Regulatory
Commission**

Office of Inspection and Enforcement

G. M. Kaczynski, R. W. Woodruff



8601070496 851231
PDR NUREG
1167 R PDR

NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 1717 H Street, N.W.,
Washington, DC 20555
2. The Superintendent of Documents, U.S. Government Printing Office, Post Office Box 37082,
Washington, DC 20013-7082
3. The National Technical Information Service, Springfield, VA 22161

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC Office of Inspection and Enforcement bulletins, circulars, information notices, inspection and investigation notices; Licensee Event Reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the GPO Sales Program: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, and NRC booklets and brochures. Also available are Regulatory Guides, NRC regulations in the *Code of Federal Regulations*, and *Nuclear Regulatory Commission Issuances*.

Documents available from the National Technical Information Service include NUREG series reports and technical reports prepared by other federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal and periodical articles, and transactions. *Federal Register* notices, federal and state legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Division of Technical Information and Document Control, U.S. Nuclear Regulatory Commission, Washington, DC 20555.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, 7920 Norfolk Avenue, Bethesda, Maryland, and are available there for reference use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018.

TPDWR2: Thermal Power Determination for Vestinghouse Reactors, Version 2

User's Guide

Manuscript Completed: October 1985
Date Published: December 1985

G. M. Kaczynski, R. W. Woodruff

**Division of Emergency Preparedness and Engineering Response
Office of Inspection and Enforcement
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555**



ABSTRACT

TPDWR2 is a computer program which was developed to determine the amount of thermal power generated by any Westinghouse nuclear power plant. From system conditions, TPDWR2 calculates enthalpies of water and steam and the power transferred to or from various components in the reactor coolant system and to or from the chemical and volume control system. From these results and assuming that the reactor core is operating at constant power and is at thermal equilibrium, TPDWR2 calculates the thermal power generated by the reactor core. TPDWR2 runs on the IBM PC and XT computers when IBM Personal Computer DOS, Version 2.00 or 2.10, and IBM Personal Computer Basic, Version D2.00 or D2.10, are stored on the same diskette with TPDWR2.

TABLE OF CONTENTS

| | | <u>Page</u> |
|---|-------------------------------------|-------------|
| | ABSTRACT..... | iii |
| 1 | INTRODUCTION..... | 1 |
| 2 | HEAT BALANCE..... | 3 |
| 3 | ENTHALPY AND SPECIFIC VOLUME..... | 5 |
| 4 | ORIENTATION DISPLAYS..... | 7 |
| 5 | MASTER MENU..... | 7 |
| | 5.1 Analysis..... | 9 |
| | 5.2 Blank Data Sheet..... | 9 |
| | 5.3 Calculator..... | 9 |
| | 5.4 Change Dockets..... | 9 |
| | 5.5 Parameters & Data..... | 9 |
| | 5.6 Preface..... | 10 |
| | 5.7 Printer..... | 10 |
| | 5.8 Reports..... | 10 |
| | 5.9 Steam Tables..... | 10 |
| | 5.10 Termination..... | 10 |
| 6 | SUBORDINATE MENUS..... | 10 |
| | 6.1 Analysis Menu..... | 11 |
| | 6.2 Parameters & Data Menu..... | 11 |
| | 6.3 Reports Menu..... | 13 |
| | 6.4 Steam Tables Menu..... | 13 |
| 7 | CONTINGENT MENUS..... | 14 |
| | 7.1 Data Handling Menu..... | 14 |
| | 7.2 Parameter Handling Menu..... | 14 |
| 8 | MODULES..... | 14 |
| | 8.1 Autostart Module..... | 14 |
| | 8.2 Command Start Module..... | 15 |
| | 8.3 Master Program Module..... | 15 |
| | 8.4 Introduction Module..... | 15 |
| | 8.5 Parameters & Data Module..... | 16 |
| | 8.6 Parameters Handling Module..... | 16 |
| | 8.7 Data Handling Module..... | 16 |
| | 8.8 Analysis Module..... | 17 |
| | 8.9 Display Module..... | 21 |
| | 8.10 Steam Tables Module..... | 21 |
| | 8.11 Reports Module..... | 22 |
| | 8.12 Termination Module..... | 22 |

TABLE OF CONTENTS (Continued)

| | <u>Page</u> |
|--------------------------------|-------------|
| 9 OPERATING PROCEDURE..... | 23 |
| 10 INSTALLATION PROCEDURE..... | 23 |

APPENDICES

| | |
|-----------------------------|--|
| A BATCH ROUTINES | |
| B TPDWR2.BAS | |
| C INTRO.BAS | |
| D PRM&DATA.BAS | |
| E PRM.BAS | |
| F DATA.BAS | |
| G ANALYSIS | |
| H DISPLAY.BAS | |
| I TABLES.BAS | |
| J REPORTS.BAS | |
| K END.BAS | |
| L PROGRAM PERFORMANCE TESTS | |

1 INTRODUCTION

Licenses for nuclear power plants restrict the maximum thermal power at which reactors may be operated. Technical Specifications appended to licenses set forth additional restrictions on thermal power for certain operating conditions. These restrictions ensure that fuel cladding will not fail if the fuel rods generate more heat than can be effectively dissipated by the coolant flow. To ensure that license requirements are met, licensees periodically determine thermal power by performing a heat balance. The result obtained from the heat balance is used in calibrating the neutron flux measuring instruments that provide input for control of reactor power level. The computer program, TPDWR2 (Thermal Power Determination for Westinghouse Reactors, Version 2), described in this report, was developed to provide reactor inspectors with the capability to independently perform a heat balance for any Westinghouse reactor whenever the reactor and its primary cooling system are at or near thermal equilibrium.

To perform a heat balance, the heat sources in the primary cooling system are equated to the heat sinks. The principal heat source is the reactor core, and the principal heat sinks are the steam generators. To calculate the heat transfer rate in each steam generator, the pressure at which the steam generator is operating and the temperature and flow rate of feedwater are needed. From these data, TPDWR2 calculates the enthalpies of steam and feedwater. The rate of heat transfer in each steam generator is equal to the enthalpy difference between feedwater and steam multiplied by the mass flow rate of feedwater. Reactor power is then found by summing the heat transferred by each steam generator and correcting for heat input by pumps, heat loss through thermal insulation, and other effects.

To demonstrate that the reactor is at equilibrium or to provide for small non-equilibrium corrections, two sets of data may be taken. From these data, corrections are made by TPDWR2 for the effects of changes in water levels in steam generators and the pressurizer. If the user wishes, the heat balance can be done with one set of data.

Before TPDWR2 can be used for the first time, the user must format a new diskette, copy BASIC.COM from the user's DOS diskette to the newly formatted diskette, and copy the modules from the TPDWR2 application diskette to the newly formatted diskette. A procedure for doing this is given in Section 10.

TPDWR2 consists of the following ten principal modules. The titles in parentheses are the file names used by the computer for accessing the modules as a result of selection of tasks by the user from various menus.

| | |
|----------------------------|------------------|
| TPDWR2 | (TPDWR2.BAS) |
| Introduction Module | (INTRO.BAS) |
| Parameters & Data Module | (PRM & DATA.BAS) |
| Parameters Handling Module | (PRM.BAS) |
| Data Handling Module | (DATA.BAS) |
| Analysis Module | (ANALYSIS.BAS) |
| Display Module | (DISPLAY.BAS) |

| | |
|---------------------|---------------|
| Steam Tables Module | (TABLES.BAS) |
| Reports Module | (REPORTS.BAS) |
| Termination Module | (END.BAS) |

Before doing the first heat balance for a given plant, values of certain design parameters are needed, for example, power and efficiency of the reactor coolant pumps. The Parameter Handling Module displays on the screen, in sequence, three short tables for inputting the values of the parameters. For essentially all of the parameters, the module provides default values, which will be adequate for most Westinghouse reactors. The Parameter & Data Module automatically creates a plant-specific parameters file on the TPDWR2 diskette. Values of parameters are stored there by the computer and are retained indefinitely.

After the plant parameters are stored and before doing a heat balance, certain test data are needed, including feedwater flow and temperature and steam pressure. The Data Handling Module displays a data sheet on the screen where the user makes the necessary data entries. The Parameter & Data Module automatically creates a test-specific data file for the plant on the TPDWR2 diskette and stores these data there. These data are retained on the diskette until data are entered for the next test for that plant.

The Analysis Module obtains values of parameters and data from the Parameters & Data Module and calculates enthalpy and specific volume of subcooled and saturated water and saturated steam as necessary. The Analysis Module then calculates the distribution of power among the various flow paths in the reactor coolant system, calculates reactor power, stores the results on the TPDWR2 diskette until a subsequent analysis is performed, and displays the results on command. If a printer is connected to the computer, then the Reports Module, on command, prints a tabulation of parameters and data and a report of the analysis. The Reports Module also prints, on command, a blank data sheet for the convenience of the user.

In preparation for entry of parameters and data and for performance of a heat balance, the Introduction Module queries the user for the docket number of the unit to be analyzed, provides information on the status of parameter and data storage files, flags the printer routines if a printer is used, and presents the master menu. From the master menu, the user can access additional menus to initiate the computer actions already described. In addition, the user can independently access the steam tables to find values of enthalpy and specific volume for subcooled and saturated water and for saturated and superheated steam. The user also can use the computer as a calculator if nonroutine calculations are necessary.

The Termination Module will save or delete data and results, depending on the user's needs, and make the computer available for other work.

Besides the ten principal modules, there are two batch modules in TPDWR2. These are:

| | |
|----------------------|----------------|
| Autostart Module | (AUTOEXEC.BAT) |
| Command Start Module | (T.BAT) |

The Autostart Module automatically starts the program after the diskette is latched in the A Drive and power is supplied. If the computer has been powered up and is waiting for a command, the Command Start Module will start the program.

TPDWR2 was written specifically for use with IBM PC and XT computers which have IBM Personal Computer DOS, Version 2.00 or 2.10, and IBM Personal Computer Basic, Version D2.00 or D2.10, available. The program functions with or without a printer connected to the computer.

2 HEAT BALANCE

The reactor core and reactor coolant pumps are sources of heat in the primary reactor coolant system. From these sources, heat flows in the primary coolant to the steam generators, which transfer heat from the primary coolant system to the secondary coolant system. The chemical and volume control system, which is connected to the primary coolant system, is also a heat sink. In addition, heat flows from coolant, through piping and thermal insulation, and is lost to the system that cools the reactor cavity and piping galleries. To form a heat balance equation for the primary coolant system, heat per unit of time generated by the heat sources is equated to heat per unit of time entering these sinks. To determine reactor power, the heat balance equation is solved for reactor power and the terms on the right side of the equation are evaluated.

Heat sink terms for the steam generators are found by writing a heat balance equation for the secondary side of each steam generator and solving for the heat per unit of time transferred across the tube walls of the steam generator. This quantity is equal to the sum of the heat per unit of time flowing in the steam and blowdown streams less the heat per unit of time flowing in the feedwater stream. Heat per unit of time flowing in each stream is equal to the product of enthalpy and mass flow rate for that stream.

Steam produced in the steam generator is saturated, not superheated; therefore, the enthalpy of the steam produced is a function only of the pressure at which the steam generator is operating and the quality of the steam. Pressure is monitored by the instrumentation system, and quality can be determined from a moisture carry-over test. Using known values of steam pressure and quality, the enthalpy of the wet steam produced can be determined by obtaining from steam tables the enthalpies of steam and water at saturation. The flow rate for the wet steam can be found by subtracting the blowdown flows from feedwater flow where these flows are expressed in mass units per unit of time.

Feedwater temperature is measured, and at the nozzle, feedwater pressure must be equal to steam generator pressure. From known temperature and pressure, feedwater enthalpy can be found from steam tables, specifically, the table addressing compressed water and superheated steam. If feedwater flow is expressed in volume units, it can be converted to mass units after obtaining specific volume (the reciprocal of density) from the table for compressed water.

Blowdown flow rates are usually expressed in gallons per minute, and these flow rates must be converted to mass units per unit of time. For surface blowdown, the temperature of the stream is assumed to be essentially at the saturation temperature. Given steam generator pressure, the temperature and specific volume at saturation can be found from the table for saturated steam and water. Using the specific volume, the flow rate can be converted to pounds per hour. For bottom blowdown, the temperature of the stream is assumed to be midway between feedwater temperature and saturation temperature. Again, the pressure at the nozzle for this stream is steam generator pressure and the specific volume is determined from the compressed water table. In addition to specific volume, the enthalpy must be found for the blowdown streams.

If steam pressure, feedwater flow rate and temperature, and blowdown flow rates are known, the power transferred from the primary side of the steam generator to the secondary side can be determined. Likewise, the power lost to the chemical and volume control system can be found if pressurizer pressure and letdown and charging flows and temperatures are known.

Thus far, the discussion has tacitly assumed that the flow distributions in the primary coolant system and the secondary sides of the steam generators are at equilibrium. If feedwater flow to a steam generator is in excess of the flow rate necessary to maintain the water level in the steam generator, then excess flow should be determined so that calculated steam flow can be corrected. Excess feedwater flow is determined by calculating the rate of change of water volume on the secondary side of the steam generator from two sets of data taken at different times and multiplying by the density of saturated water. Steam flow is then corrected by subtracting excess feedwater flow. Heat accumulated on the secondary side of the steam generator as a result of excess feedwater flow can be neglected.

Likewise, an excess of charging flow over letdown flow will result in flow to the pressurizer. The heat per unit time accumulated in the pressurizer is equal to the enthalpy of water in the hot leg of the primary cooling system times the mass flow rate which is found in the same way that excess flow is found for steam generators. If flow is out of the pressurizer, then the enthalpy and density of water in the pressurizer must be determined to find the heat per unit of time flowing into the primary cooling system. Heat accumulated in or returned to the primary coolant system can be neglected; nevertheless, TPDWR2 calculates this quantity when two sets of data are used.

The discussion also has tacitly assumed that the reactor and the primary coolant system are at thermal equilibrium. However, the reactor core may be generating more power than is being removed. The difference is stored in the reactor core and the reactor coolant system. For a 3900-MW reactor, the thermal capacity of the core and primary cooling system is of the order of 2 million Btu per Fahrenheit degree. If the average temperature of the core and primary cooling system change 0.1°F in 30 min, then the heat stored is 0.2 million Btu and the rate at which it is stored is roughly 110 Btu per second or 0.1 MW or 0.003% of licensed power. TPDWR2 does not compute a correction for deviation from thermal equilibrium.

TRPDWR2 does compute power lost through the thermal insulation on the reactor vessel, the primary reactor coolant system, and the steam generators. The types of thermal insulation, reflective or nonreflective, must be known as well as the surface area covered by each type. For nonreflective insulation, the average thickness of the insulation and its thermal conductivity must be known or estimated. Typical values might be 4 in. and 0.035 Btu per hour per foot per Fahrenheit degree. For reflective insulation, the heat loss coefficient must be known either from test or from purchase specification. A typical value might range from 50 to 100 Btu per hour per square foot. Total heat losses through the thermal insulation of a 3900-MW system might be as low as 0.3 MW.

The final term to be evaluated in the heat balance equation for the primary coolant system is heat input from the reactor coolant pumps. The heat input can be obtained from the pump name plate. The default value that TPDWR2 assumes for motor losses is 10%. When the default value is used, TPDWR2 multiplies

pump power by 90% and subtracts the product from the right side of the heat balance equation. For a four-loop, 3900-MW reactor, each reactor cooling pump motor might require a 6-MW or 8000-hp motor.

3 ENTHALPY AND SPECIFIC VOLUME

Depending on the temperature and pressure to which water is subjected in the reactor coolant system, the water can be in either the liquid or vapor phase or both. The boundary between the two phases is called the saturation line. This line terminates at the triple point and at the critical point. The triple point is at 32.018°F and at 0.08865 pounds per square inch absolute (psia). It is the point where water can exist simultaneously in the solid, liquid, and vapor phases. The critical point is at 705.47°F and 3208.2 psia. At temperatures, above this point, water vapor cannot be condensed at pressures as great as 15,500 psia.

To do a heat balance for the primary coolant system of a pressurized water reactor, values of enthalpy and specific volume are needed for subcooled (or compressed) water and for saturated water and steam. Internationally accepted values for these properties are given in the "ASME Steam Tables," Fourth Edition, published by the American Society of Mechanical Engineers. Table 1 in the ASME Steam Tables presents values of saturation pressure as a function of temperature from the triple point to the critical point. Table 3 presents values for enthalpy and specific volume for compressed and superheated water from 32°F to 1500°F and from 0.12 psia to 15,500 psia.

Appendix 1 to the ASME Steam Tables presents the equations that were used to calculate the values presented in Steam Tables 1 and 3. In developing these equations, contributors to the ASME Steam Tables found it convenient to express temperature and pressure in dimensionless units. A quantity called the reduced temperature (θ) is defined as absolute temperature of the fluid divided by the absolute temperature at the critical point. Likewise, reduced pressure (β) is defined as absolute pressure of the fluid divided by absolute pressure at the critical point. Equations for enthalpy and specific volume are given in Appendix 1 to the ASME Steam Tables as functions of θ and β for the rectangular temperature-pressure field bounded by the following four straight lines:

$$\theta = 0.423$$

$$\theta = 1.66$$

$$\beta = 0.00$$

$$\beta = 4.52$$

This field is divided into four sub-regions as shown in Figure 1 of this user's guide. For Sub-regions 1 and 4, water is in the liquid phase; and for the other two sub-regions, it is in the vapor phase. Appendix 1 to the ASME Steam Tables provides equations for enthalpy and specific volume for each sub-region. Appendix 1 also provides equations for the saturation line and other boundaries between sub-regions.

TPDWR2 uses the equations for enthalpy and specific volume for Sub-regions 1 and 2. In the direction of increasing β , Sub-region 1 extends far beyond the area of interest for light water reactors. For Sub-region 2, the area of interest is at the saturation line. Nevertheless, with two exceptions, TPDWR2 duplicates the values for enthalpy and specific volume in the ASME Steam Tables out

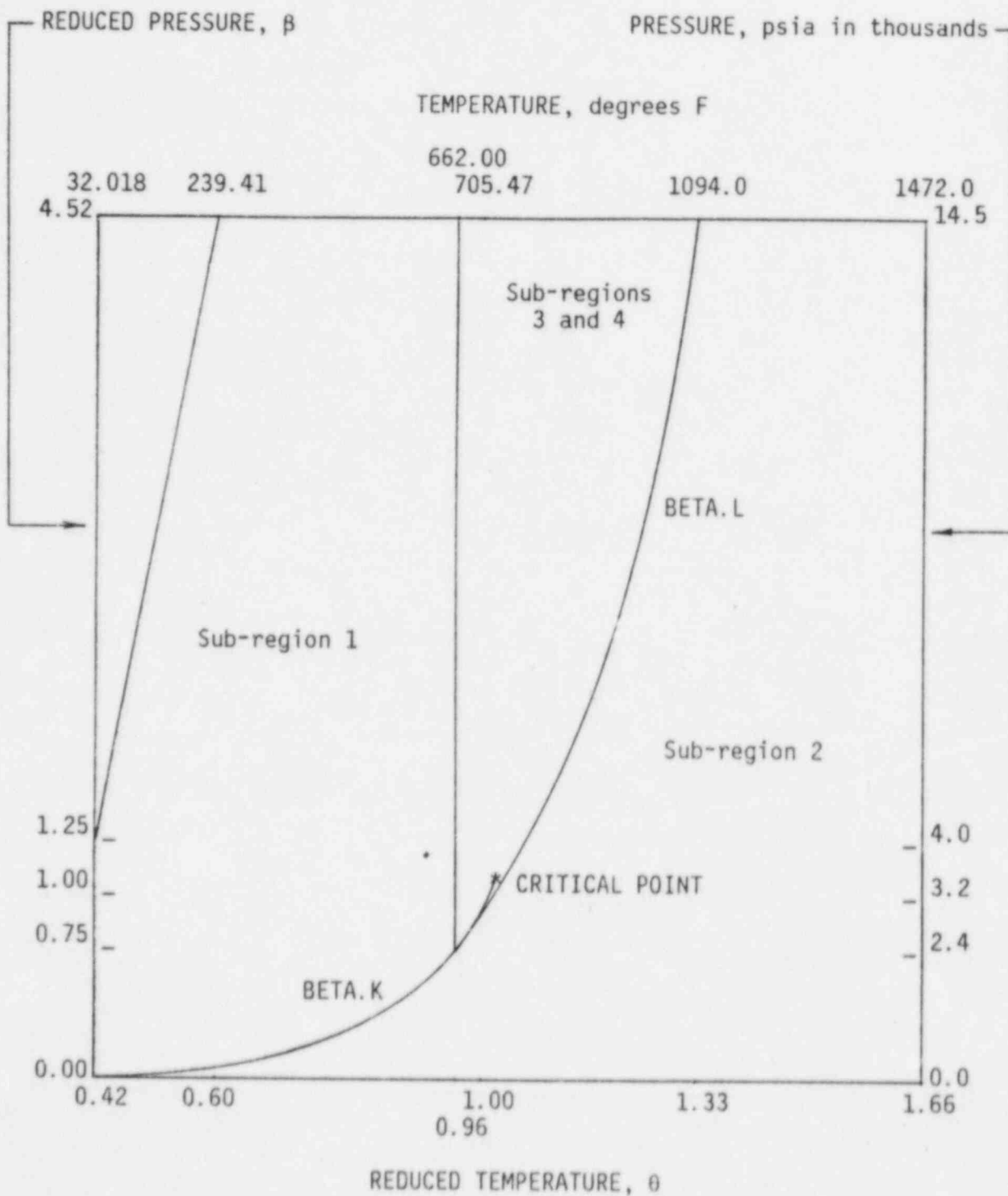


Figure 1: Boundary for Coolant Property Calculations

to the boundaries of the sub-regions, and TPDWR2 will calculate and display values of the properties in these areas if the user is inclined to do so.

In Sub-region 1 in the vicinity of the point, ($\theta = 0.5$, $\beta = 2.8$), TPDWR2 does not exactly reproduce values of enthalpy given in Table 3 of the ASME Steam Tables. However, the error is less than 1 part in 1300. Further, the absolute pressure in this area is roughly 9000 psia, thus this error has no effect on reactor calculations. The area to the left of the line through this point, although part of Sub-region 1, has been excluded from TPDWR2, because larger errors result from the calculation.

Sub-regions 3 and 4 surround the critical point and extend away from it in the direction of increasing β . Because light water reactors are not designed to operate in the critical region, these sub-regions have not been included in TPDWR2.

The saturation line is designated in the ASME Steam Tables as the K-Function or as β , subscript K. In Figure 1 and in Appendix G, ANALYSIS.BAS, of this user's guide, it is identified as BETA.K. The boundary between Sub-regions 2 and 3 is designated in the ASME Steam Tables as the L-Function or as β , subscript L. In Figure 1 and in Appendix G, it is identified as BETA.L.

The routines for calculating enthalpy and specific volume are in the Steam Tables Module which is documented in Appendix I of this user's guide. These routines have been incorporated directly in the Analysis Module to avoid repeated chaining between these two modules. A significant savings in computing time is accrued.

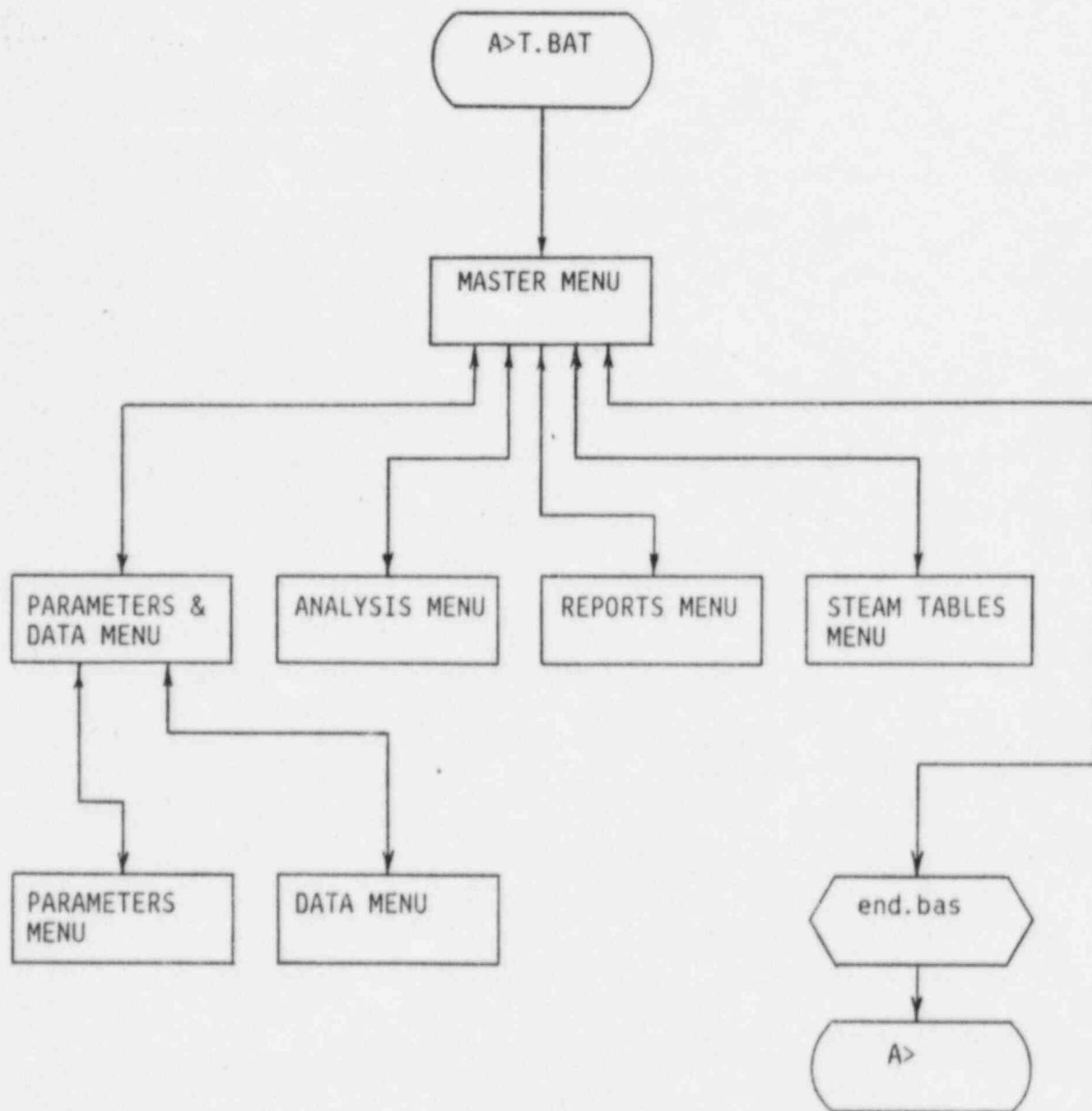
4 ORIENTATION DISPLAYS

The Introduction Module displays on the screen an initial series of orientation messages to identify the program, to tell the user how to obtain help, and to provide brief instructions for controlling the machine. The module then asks the user whether use of a printer is intended and, if so, it guides the user in preparing the printer for use. Further, the Instruction Module sets the print flag so that subsequent modules select the appropriate display formats for the printer. After the title of the program is displayed, the user can avoid the subsequent preliminary displays by typing "a" without the quotation marks.

The module then queries the user for the docket number of the plant and uses that information to determine whether the plant-specific parameters, data, and results files contain information or are empty. The module sets flags to keep track of the status of these files so that appropriate messages can be displayed for the user. After these preliminaries, the Introduction Module displays the Master Menu. This menu provides the user with the means for loading any of the other modules.

5 MASTER MENU

TPDWR2 has eight menus as shown in Figure 2. The Master Menu provides direct access to tasks listed on four subordinate menus, provides indirect access to tasks listed on two contingent menus, and provides the capability to initiate certain other supporting tasks. From any of the subordinate menus, the user can return to the Master Menu.



A> is the DOS prompt.

Figure 2: Menu Map

When the Master Menu is displayed, there may or may not be values of plant parameters stored on the TPDWR2 diskette for the docket number which was selected by the user. If values of plant parameters are not stored on the diskette, a banner will be displayed below the Master Menu informing the user of that fact and suggesting that the Parameters & Data task be selected. Nevertheless, the user can select any other task. Those tasks which can run without plant parameters, will run.

The tasks described in the following sections can be run from the Master Menu.

5.1 Analysis

Choice of this task causes the Analysis Module to be loaded in computer memory and the Analysis Menu to be displayed. See Section 6.1 for a description of the tasks which can be performed from that menu.

5.2 Blank Data Sheet

Choice of this task causes the Reports Module to be loaded into computer memory and a blank data sheet to be printed. If the printer option was not selected when the computer ran through the introductory portion of the program, then the computer loads the Introduction Module and presents the display in which the user can elect to use or not use a printer. After the printer is selected, the computer displays the Master Module again. If the user again selects Blank Data Sheet, the Reports Module will then print a data sheet for a four-loop PWR, reload the Introduction Module and again display the Master Menu.

5.3 Calculator

Choice of this task causes the computer to (1) display the Calculator Procedure across the top of the screen, (2) allow space on the screen in which the user can type the mathematical expression to be evaluated, and (3) leave TPDWR2 so that BASIC is available. The user can then type the mathematical expression according to the instructions in the Calculator Procedure. Logarithmic and trigonometric functions are available. When the enter key is pressed, the calculation is performed and the answer is displayed. When the calculations are completed, the user should return to TPDWR2 by following the instructions at the bottom of the screen, i.e., by pressing the F2 key. The computer will reload the Introduction Module and display the Master Menu.

5.4 Change Dockets

Before the user could access the Master Menu, it was necessary to select a docket by docket number. Anytime thereafter, when the Master Menu is displayed the user can change docket numbers by selecting this task. The computer will prompt for the new number and ask for the user to verify that the number is correct. After verification is received, the computer closes open files associated with the old docket number, opens new files as necessary, and again displays the Master Menu.

5.5 Parameters & Data

Choice of this task causes the Parameters & Data Menu to be loaded in computer memory and the Parameters & Data Menu to be displayed. See Section 6.2 for a description of the tasks which can be performed from that menu.

5.6 Preface

Choice of this task causes the computer to return to the first display following the NRC logo and to display information described in Section 4.

5.7 Printer

Choice of this task causes the computer to return to that orientation display which asks the user whether the printer will be used and then to return to the Master Menu.

5.8 Reports

Choice of this task causes the computer to determine whether the printer is connected to the system and power is on. If the printer is not connected or printer power is off, the computer presents on the screen the display which asks the user to decide whether or not the printer will be used. After the computer receives a response, the Master Menu is displayed, and the user can again select the reports task. The computer then loads the Reports Module into memory and displays the Reports Menu. See Section 6.3 for a description of the tasks which can be performed from this menu.

5.9 Steam Tables

Choice of this task causes the computer to load the Steam Tables Module into memory and to display the Steam Tables Menu. See Section 6.4 for a description of tasks which can be performed from this menu.

5.10 Termination

Choice of this task causes the computer (1) to load the Termination Module into memory; (2) to display the status of the parameters, data, and results storage files for the current docket; (3) to present an option to erase certain files for the current docket; (4) to present an option to return to the beginning of TPDWR2 or to exit from it; and (5) in the event that the user elects to exit, to present the option to remain in BASIC or return to DOS.

The options to erase certain current docket files are contingent on which files contain information. If all three files contain information, the user has the option of deleting the data and results files from memory. If parameters and data files contain information, the user has the option of deleting the data files. If only the parameters file has information, the user has the option of deleting it.

6 SUBORDINATE MENUS

The subordinate menus are accessible from the Master Menu and the Master Menu is accessible from any of the subordinate menus. Two of the subordinate menus, i.e., the Analysis and Reports Menus, also provide direct access to the Termination Module which performs the tasks and presents the options described in Section 5.10. The tasks that can be initiated from the subordinate menus are described in the following sections.

6.1 Analysis Menu

From this menu, heat balance data can be analyzed or, if the analysis has already been completed, the results of the analysis can be displayed on the screen.

If the analysis task is selected, the computer loads the Parameters & Data Module into memory, retrieves the values of the parameters and data from the storage files for the current docket, reloads the Analysis Module, performs the analysis, and again displays the Analysis Menu.

If the display task is selected and data have been analyzed, the computer loads the Display Module, recalls the results of the analysis from storage, displays the results, reloads the Analysis Module, and again displays the Analysis Menu. If the display task is selected and data have not been analyzed, the computer displays a message to that effect on the screen, reloads the Analysis Module, and again displays the Analysis Menu.

6.2 Parameters & Data Menu

From this menu, values of plant parameters can be entered or corrected, heat balance data can be entered or corrected, and values of parameters and data can be displayed on the screen.

6.2.1 Parameter Entry

If the task to enter values of plant parameter is selected, the computer loads the Parameters Handling Module into memory. If values of plant parameters are already in storage, the contingent Parameter Handling Menu is displayed. Tasks which can be initiated from that menu are described in Section 7.2. If values are not in storage, the computer asks for verification that the nuclear steam supply system was designed by Westinghouse. If the response is no, an appropriate message is displayed and the Master Menu is again displayed. If the response is yes, three displays are shown on the screen in sequence. In the first display the user is asked to enter the plant name, unit number, number of cooling loops, and licensed power level. In the second and third displays, the user is asked to enter the values of certain reactor coolant system and steam generator parameters.

For each display, highlighted parameter fields are presented to show the user where to make the entry and how many characters can be used for each entry. The displays for the reactor coolant system and steam generator parameters include default values which will be adequate for most plants. Each of these displays presents, on the lowest line of the screen, directions for selecting the default value, entering some other value, or requesting help. The help displays, as appropriate, provide instructions for moving the cursor, descriptions of the way in which the Parameters Module calculates the various default values, and other information that the user may need for performing an adequate heat balance.

6.2.2 Parameter Correction

If the task to correct values of plant parameters is selected, the Parameters & Data Module checks for the presence in storage of values of parameters. If values are not present, a message to that effect is displayed on the screen and

the Parameters & Data Menu is again displayed. If values of plant parameters are in storage, the computer loads the Parameters Handling Module into memory and shows three displays on the screen in sequence. The first display presents the current plant name and values for unit number, number or reactor coolant loops, and licensed thermal power. The second and third displays show current values for reactor coolant system and steam generator parameters.

For each display, the current value of the parameter is presented in a highlighted parameter field. The location of the cursor (an underscore) is visible and the cursor is blinking. Any value on the display can be changed by moving the cursor to the beginning of the selected parameter field, initiating entry, and typing over the current value. Guidance for doing this is presented in the instruction line at the bottom of the display. When corrections for a given display are completed, the user can exit to the next display. After the last display is shown, the computer reloads the Parameters & Data Module and again displays the Parameters & Data Menu.

6.2.3 Data Entry

If the task to enter heat balance data is selected, the computer loads the Data Handling Module into memory. If data are already in storage, the contingent Data Handling Menu is displayed. Tasks which can be initiated from that menu are described in Section 7.1. If data are not in storage, the computer asks whether the user will use one or two sets of data. If liquid levels and secondary system flow rates are at equilibrium, the user will find that one set of data is adequate. After the user responds, the computer will display a data sheet with highlighted data fields. The instruction line at the bottom of the screen provides guidance for data entry and for accessing a help display. Data can only be entered in the sequence requested by the computer. After the first data sheet is completed, the computer will display a second data sheet if the user indicated that two sets of data would be used. When data entry is completed, the computer reloads the Parameters & Data Module and again displays the Parameters & Data Menu.

6.2.4 Data Correction

If the task to correct heat balance data is selected, the Parameters & Data Module checks for the presence in storage of data. If data are not present a message to that effect is displayed on the screen and the Parameters & Data Menu is again displayed. If data are in storage, the computer loads the Data Handling Module into memory and shows on the screen either a single data sheet or two data sheets in sequence depending on whether one or two sets of data were taken. Current values of the data are presented in highlighted data fields. The location of the cursor (an underscore) is visible and the cursor is blinking. Any value on the data sheet can be changed by moving the cursor to the beginning of the selected data field, initiating entry, and typing over the current value. Guidance for doing this is presented in the instruction line at the bottom of the data sheet. When corrections for the first data sheet are completed, the user can exit to the second data sheet, if it exists. The computer then reloads the Parameters & Data Module and again displays the Parameters & Data Menu.

6.2.5 Display of Parameters and Data

If the task to display values of parameters and data is selected, the computer recalls these values from storage, if they have been stored, and presents

sequential displays on the screen. If either values of parameters or data or both are not in storage, appropriate messages are displayed. After the last display is presented, the Parameters & Data Module is again displayed.

6.3 Reports Menu

From this menu, reports in tabular form of values of parameters and data and of the results of a heat balance analysis can be printed. Blank data sheets can also be printed.

6.3.1 Parameters and Data

If the task to print a report of parameters and data is selected, the computer loads the Parameters & Data Module in memory, recalls the values of parameters and data from storage, reloads the Reports Module into memory, and prints the report. The report format is adjusted to accommodate units with two, three, or four loops. To maintain 1-inch margins on the report, it is printed in condensed mode, i.e., 17 characters per inch.

6.3.2 Heat Balance

If the task to print a report of a heat balance analysis is selected, the computer loads the Analysis Module into memory, recalls the results of the heat balance analysis from storage, reloads the Reports Module, and prints the report. The report format is adjusted to accommodate units with two, three, or four loops.

6.3.3 Blank Data Sheet

If the task to print a blank data sheet is selected, the computer prints a data sheet which is appropriate for any Westinghouse reactor.

6.4 Steam Tables Menu

From this menu, the enthalpy, specific volume, density, and phase can be determined for water in Sub-regions 1 and 2 as shown in Figure 1 and for saturated water at the boundary between Sub-regions 1 and 2.

6.4.1 Saturated Water at Known Temperature

If this task is selected, the computer first queries the user for temperature. It then calculates the saturation pressure of water, inputs temperature and saturation pressure into the Sub-region 1 equations for enthalpy and specific volume to obtain values for water and into the Sub-region 2 equations for enthalpy and specific volume to obtain values for steam. The reciprocal of the values for specific volume are calculated to obtain density. The results of the calculations are displayed on the screen, and if the printer has been selected, the results are also printed.

6.4.2 Saturated Water at Known Pressure

If this task is selected, the computer first queries the user for pressure. It then calculates the saturation temperature of water, using an iterative

technique. The Steam Tables Module assumes a value for saturation temperature, and the computer calculates the saturation pressure corresponding to that temperature. The difference between known pressure and calculated saturation pressure is found. From this error, the computer estimates a new value of saturation temperature and again calculates the value of saturation pressure. After each iteration, the difference between known and calculated saturation pressures is displayed on the screen. Iteration continues until the difference is acceptably small, i.e., until the results in the ASME Steam Tables are duplicated. Calculation then proceeds as in Section 6.4.1.

6.4.3 Unsaturated Water

If this task is selected, the computer first queries the user for values of temperature and pressure. The computer then calculates saturation pressure to determine whether the water is in the liquid phase (Sub-region 1) or the vapor phase (Sub-region 2). The computer then uses the appropriate equations to determine values of enthalpy, specific volume, and density. These values and the identity of the phase are displayed and, if the printer has been selected, are printed.

7 CONTINGENT MENUS

In addition to the tasks described below, the computer, on command, will return to the previous menu, i.e., the Parameters & Data Menu.

7.1 Data Handling Menu

When data are stored in the current docket and when the enter data task is selected from the Parameters & Data Menu, the Data Handling Module displays a contingent menu on the screen. From the contingent data handling menu, the user can (1) write over the existing set or sets of data or (2) if only one set of data is in storage, write a second set of data. Thus the user can take an initial set of data, do a heat balance analysis, review the results, take a second set of data later in time, and do a heat balance analysis with both sets of data to account for nonequilibrium effects.

7.2 Parameter Handling Menu

When values of parameters are stored in the current docket and when the enter parameters task is selected from the Parameters & Data Menu, the Parameters Handling Module displays a contingent menu on the screen. From the contingent parameters handling menu, the user can (1) write over the existing values of all the parameters or (2) correct existing values of selected parameters.

8 MODULES

Except for the Autostart and Command Start Modules, the software modules are written in BASIC. The Autostart and Command Start Modules are written for DOS. Each of these modules is described in the following subsections, and the actual program modules are given in Appendices A through K.

8.1 Autostart Module (AUTO EXEC. BAT)

After the TPDWR2 program diskette is inserted in the A Drive and power is connected to the computer, the computer automatically loads the Autostart Module

into computer memory and runs the module. Autostart loads TPDWR2.BAS into memory and runs it.

8.2 Command Start Module (T.BAT)

After the computer is booted with DOS and typing "t" (without quotation marks) and pressing the enter key in response to the DOS prompt, the computer will load and run Master Program Module, provided that the TPDWR2 working diskette is in the drive designated by the DOS prompt.

8.3 Master Program Module (TPDWR2.BAS)

The Master Program Module displays an NRC logo and dimensions those variable arrays that are common to two or more modules. The Master Program Module then chains to the Introduction Module.

8.4 Introduction Module (INTRO.BAS)

The Introduction Module displays on the screen an initial series of three orientation messages to identify the program, to tell the user how to obtain help, and to provide instructions for controlling the machine. The module then asks the user whether use of a printer is intended and, if so, it guides the user in preparing the printer for use. Further, the Introduction Module sets the print flag so that subsequent modules select the appropriate display formats for the printer and the screen. The module then asks the user to identify the docket number of the reactor that will be analyzed. The module checks the parameter, data, and results storage files to determine whether they contain information or are empty. If values are stored, as indicated by the appropriate flags, cautionary notes are displayed on the screen for the user.

After these preliminaries, the Introduction Module displays the Master Menu. This menu provides the user with the means for loading either the Parameters and Data, Analysis, Steam Tables, Reports, or Termination Modules. The Master Menu will also print a blank data sheet, put the machine in the calculator mode, turn the printer on or off, display the preliminary messages, and change dockets if the user wishes. After the user makes a selection from the menu, the machine executes that task. If the instruction is not compatible with the status of the system, the user will be so informed, and the user will be presented with the appropriate menu so that remedial action can be taken.

When the user is working in any of the other principal modules, the user can always return to the Master Menu in the Introduction Module. When such a request is executed, the preliminary messages in the Introduction Module are bypassed, and the user is taken directly to the Master Menu.

When the calculator mode is selected, instructions are presented on the screen which guide the user in exiting from TPDWR2, running BASIC as a calculator, and returning to TPDWR2. Re-entering TPDWR2 is accomplished with a single keystroke.

The calculator will evaluate any expression containing constants and/or functions preceded by a question mark. For example, the user can determine the value of expressions like

? 5+50*(1+sin(3.1416/6))+8*10^3

by first typing the line and then pressing RETURN. The following result is then displayed:

8080

To obtain correct results, the user must know the hierarchy of operations, i.e., order of precedence, which BASIC uses. BASIC works from left to right, evaluating innermost parentheses first and then proceeds outwards. Arithmetic operations are performed from left to right in the following order:

1. Exponentiation (^)
2. Multiplication (*) and Division (/)
3. Addition (+) and Subtraction (-)

8.5 Parameters & Data Module (PRM & DATA.BAS)

When the user enters the Parameters & Data Module, the module checks to determine whether or not the parameters storage flag is set. If it is not, the values for the parameters for the docket addressed have not been stored and must be entered as a prerequisite to entering data and performing a heat balance. The module displays a note to this effect on the screen and presents the Parameters & Data Menu. The user can either enter, correct, or display values of parameters and data; return to the Master Menu; or go to the Termination Module.

8.6 Parameters Handling Module (PRM.BAS)

If the user elects to enter values of plant parameters and if values are already stored in the Parameter Storage File, then the user has the option of entering a new value for each parameter, correcting selected parameters, or returning to the Parameters & Data Menu. If the user decides to enter new values for the parameter set, a request is presented on the screen, parameter by parameter, for entry of the necessary values. After each entry is keyed in, the user presses the enter key to advance the cursor to the next parameter. The module opens the parameters storage files for that docket, if they exist, or creates files on the program diskette. The values are stored on the diskette, and the module returns to the Parameters & Data Menu.

8.7 Data Handling Module (DATA.BAS)

If the user elects to enter data, the module first recalls the values of the plant parameters to obtain the number of cooling loops for that reactor. If the values are not in parameters storage files, the module so indicates on the screen and returns the user to the Parameters & Data Menu. Otherwise, the module displays a data sheet on the screen to prompt the user for the test date and time when data were taken. Time must be written in 24-hour format, for example, 1530 for 3:30 p.m. The data sheet also prompts for steam pressure, feedwater flow and temperature, surface and bottom blowdown, and water level in each steam generator. Finally, the data sheet prompts for letdown and charging line flows and temperatures, pressurizer pressure and water level, and reactor inlet temperature and average temperature (T ave). Next, the module stores these data in the data storage files.

Feedwater flow data must be presented to the computer in millions of pounds per hour. Where reactor instrumentation presents flow data as a pressure differential measured between pressure taps at an orifice plate or venturi, the user

will need to obtain calibration data to convert the pressure differential to flow in the required units. In doing this work, if the user needs the specific volume of feedwater, it can be obtained easily from the Steam Tables Module and the calculation can be done in the calculator mode.

Random access files are used for the parameters and data storage files.

While keying values for parameters and data into the computer and before pressing the enter key, errors can be corrected by using the left and right arrow keys to reposition the cursor and writing over the entry. After pressing the enter key, the entry is stored in machine memory and cannot be corrected until all entries have been completed and the user has returned to the Parameters & Data Menu. Then, if necessary, the user can select either the parameters or data corrections option.

From the Parameters & Data Menu, the user can display on the screen the values of the parameter and data sets. Printed reports can be obtained from the Reports Module.

8.8 Analysis Module (ANALYSIS.BAS)

From the Analysis Menu, the user can analyze data, display results of the analysis on the screen, return to the Master Menu, or terminate the session. If analysis of data is selected, the module first converts clock time for the data set on sets from 24-hour format to decimal format, calculates the time difference between the data sets, if required, and stores the time difference in machine memory for use later in this module. The module next analyzes the performance of the secondary side of the steam generators using the sign convention that flow into the steam generators is negative and flow out is positive.

For Data Set 1 and Steam Generator A, the Analysis Module inputs to the steam tables routine within the module, the values of feedwater temperature and steam pressure. The steam tables routine calculates the enthalpy of feedwater in Btu per hour. The Analysis Module continues by multiplying feedwater enthalpy by feedwater flow in pounds per hour to determine the heat per unit time delivered to the secondary side of the steam generator by the feedwater. Later, the module will determine the heat per unit time removed from the secondary side in the steam and other streams and will take the difference to find the heat per unit time supplied from the primary side by the reactor core and the pumps.

The module informs the user of the current status of the analysis as it progresses. When analysis starts, the module displays "Data Set 1" on the screen and then adds on the line below "Steam Generator A" followed by "Feedwater" and "enthalpy" to indicate that this quantity has been computed.

Next, the module calculates the heat per unit time lost in the stream from the surface blowdown line. At the beginning of this calculation, the display on the screen advances one line and adds the words "Surface Blowdown." Because water near the water-steam interface in the steam generator is at the saturation temperature, the Analysis Module inputs steam pressure to that steam tables routine which calculates the saturation temperature. The Analysis Module then returns to the steam tables routine to obtain values of enthalpy and density for these conditions. After values of enthalpy and density are returned, the Analysis Module converts the flow rate of surface blowdown to pounds per hour

and calculates the heat per unit time leaving the secondary side of the steam generator via this stream in Btu per hour.

At this point, the screen display is:

Data Set 1

| | |
|-----------------------------|------------------|
| Steam Generator A Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |

For bottom blowdown, the module assumes that the water temperature is midway between the feedwater temperature and saturation temperature. Using this value and the steam pressure, the module determines the enthalpy, density, and heat per unit of time in the bottom blowdown stream.

Using the saturation temperature, the steam pressure, and the steam tables routine, the Analysis Module obtains the enthalpies of dry steam and water (moisture carry-over). However, the calculation of the heat per unit of time in the steam stream cannot be calculated at this point because the steam flow is unknown. Later, steam flow will be calculated from feedwater and blowdown flows, steam generator water levels and the time difference between data sets, if two sets of data are used. The display on the screen now reads as follows:

Data Set 1

| | |
|-----------------------------|------------------|
| Steam Generator A Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |
| Bottom Blowdown | Enthalpy Density |
| Steam | Enthalpy |
| Moisture | Enthalpy |

The module repeats this routine for Steam Generator B and for the remaining steam generators, if the reactor has more than two loops. After completing these calculations, the routine is repeated for Data Set 2, if required.

The module now calculates the excess feedwater flow for each steam generator from the change in surface level between Data Sets 1 and 2. First, the module calculates the wet cross-sectional area of the dome of the steam generator, which is equal to the area of the dome less the areas of the swirl vane risers. Next, the module calculates the product of water level and average water density for Data Set 2 minus the product for Data Set 1. The module then multiplies this difference by the wet cross-sectional area of the steam generator dome and divides by the time difference.

Taking care to observe the sign convention for flows, steam flow is calculated for each steam generator by subtracting excess feedwater flow and surface and bottom blowdowns from the feedwater flow. Heat per unit of time in the steam stream is then calculated by adding the product of dry steam enthalpy and steam quality and the product of water enthalpy and the moisture carry-over.

The Analysis Module now starts to calculate heat per unit of time gains and losses in other parts of the reactor cooling system. The following lines are added to the display on the screen:

Data Set 1

Other Components Letdown Line

Enthalpy Density

Using pressurizer pressure and letdown temperature as input to the steam tables routine the Analysis Module obtains the enthalpy and density of the letdown coolant stream. The sign convention for flow is now generalized. Flow directed toward the reactor coolant system is taken as negative, and flow away from the system is taken as positive. The module converts letdown flow from gallons per minute to pounds per hour and then calculates the heat per unit of time for the letdown stream. In the same manner, the heat per unit of time for the charging stream also is determined.

If net water mass is being added to, or removed from, the reactor coolant system as a result of mass flow imbalance in the letdown and charging lines, the pressurizer acts as an accumulator. To calculate this effect, the module first determines whether the pressurizer level is rising or falling. If the level is rising, the module calculates hot leg temperature and then the enthalpy and density of hot leg coolant, using hot leg temperature, pressurizer pressure, and the steam tables routine. If the level is falling, the Analysis Module uses pressurizer pressure and the steam tables to calculate the temperature of coolant in the pressurizer, assuming that the coolant is at saturation. For this temperature and pressurizer pressure, the enthalpy and density of coolant in the pressurizer is calculated using the steam tables routine. In either event, the Analysis Module calculates the cross-sectional area of the pressurizer and the mass flow rate into or out of the pressurizer. The mass flow rate is given by Line 2340 in Appendix G.

The heat per unit of time removed from, or supplied to, the reactor cooling system via the pressurizer is given by the product of pressurizer flow and average enthalpy for Data Sets 1 and 2.

The Analysis Module next calculates the heat loss through the thermal insulation by taking the product of the surface area covered by reflective insulation and the heat loss coefficient and the product of the surface area covered by nonreflective insulation, the thermal conductivity of the insulation, and the difference between T_{ave} and ambient temperature which is assumed to be 100°F. The products are added to obtain the total insulation loss.

To obtain the power input to the coolant from each reactor coolant pump, the nameplate rating is multiplied by the product of the motor efficiency.

At this point the Analysis Module has completed calculations of all enthalpies, densities, and corrections. For a two-loop plant, the screen will have completed the status display as shown in Table 1. Because only summing of results remains to be done and because that is done rapidly, no more calculation status lines are displayed.

For each data set, the module now calculates the heat per unit of time transferred across the tube bundle in each steam generator by algebraically summing the heat per unit of time in the steam, feedwater, and blowdown streams. Heat per unit of time dissipated elsewhere in the reactor coolant system for each data set is calculated by algebraically summing the heat per unit of time in the letdown and charging streams, the heat per unit of time lost through the

Data Set 1

| | |
|-----------------------------|------------------|
| Steam Generator A Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |
| Bottom Blowdown | Enthalpy Density |
| Steam | Enthalpy |
| Moisture | Enthalpy |
| Steam Generator B Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |
| Bottom Blowdown | Enthalpy Density |
| Steam | Enthalpy |
| Moisture | Enthalpy |

Data Set 2

| | |
|-----------------------------|------------------|
| Steam Generator A Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |
| Bottom Blowdown | Enthalpy Density |
| Steam | Enthalpy |
| Moisture | Enthalpy |
| Steam Generator B Feedwater | Enthalpy |
| Surface Blowdown | Enthalpy Density |
| Bottom Blowdown | Enthalpy Density |
| Steam | Enthalpy |
| Moisture | Enthalpy |

Data Set 1

| | |
|-------------------------------|------------------|
| Other Components Letdown Line | Enthalpy Density |
| Charging Line | Enthalpy Density |
| Pressurizer | Enthalpy Density |

Data Set 2

| | |
|-------------------------------|------------------|
| Other Components Letdown Line | Enthalpy Density |
| Charging Line | Enthalpy Density |
| Pressurizer | Enthalpy Density |

Table 1: Screen Display during Analysis of Data

thermal insulation, and subtracting the heat input from the reactor cooling pumps. Reactor power is found by summing the heat per unit of time transferred in the steam generators and in other components and lines.

Analysis of the data is complete. The Analysis Module stores the results on the TPDWR2 working diskette in the results storage files where the results are available until they are overwritten after analysis of other data. Until that happens, this module in conjunction with the Display Module can be used to recall the results and display them on the screen; or the Reports Module can be used to print the results.

8.9 Display Module (DISPLAY.BAS)

The Display Module is loaded into memory by the computer when the user requests display of the heat balance results. When the display is terminated, the computer reloads the Analysis Module and displays the Analysis Menu.

8.10 Steam Tables Module

In Appendix I, most of the equations for enthalpy, specific volume, and related quantities are expressed in terms of θ , the reduced temperature, and β , the reduced pressure, to be comparable to equations in the ASME Steam Tables. However, the equations could have been expressed in terms of temperature and pressure, and the discussion that follows is written as though that were the case.

The Steam Tables Module can be accessed by the user directly from the Master Menu. When accessed from the master menu, the independent variables section of the Steam Tables Module queries the user for the condition of the coolant and values of temperature, pressure, or both. Specifically, the module asks whether the coolant is saturated at known temperature, saturated at known pressure, or unsaturated at known temperature and pressure. After data are entered, the module calculates the saturation pressure associated with temperature and the saturation temperature associated with pressure where those quantities exist. The module then calculates enthalpy, specific volume, and density for one or both phases, as appropriate. The results are displayed on the screen, or if a printer is connected and the printer option was selected when the Introduction Module was running, the results are printed.

Before doing any calculation, the independent variables section of the Steam Tables Module stores the input values of temperature and pressure in machine memory and calculates corresponding values of θ and β . Execution then proceeds to the section of the module on limits of applicability. This section compares the values of temperature and pressure against the limits of applicability for the steam tables. These limits are the noncontiguous boundaries of Sub-regions 1 and 2 as displayed in Figure 1. If an applicability limit is violated, the module displays an appropriate error message on the screen and chains to the Introduction Module which displays the Master Menu again.

If the applicability limits are not violated and if the coolant is not saturated, execution advances to the sub-region and phase identification section of the module. This section identifies the sub-region in which the temperature-pressure point resides. If the temperature of the fluid is less than or equal to 662°F and if the pressure of the fluid is greater than the saturation pressure (BETA.K), as shown in Figure 1, then the fluid is in the liquid phase;

otherwise, the fluid is in the steam phase. Execution then branches to either the section for Sub-region 1 (water) or the section for Sub-region 2 (steam), as appropriate.

For either sub-region, if the pressure of the fluid is less than the pressure at the critical point, the module calculates the saturation temperature associated with the fluid pressure. Because saturation pressure is a complicated function of temperature, temperature at saturation is found by an iterative technique. Iteration starts with an assumed value of saturation temperature equivalent to a θ of 0.5. The saturation pressure associated with the assumed value of saturation temperature is calculated. This value is subtracted from the given value of pressure to find the error in saturation pressure. From the error, a better value of saturation temperature is obtained. Iteration continues until a value is obtained that is as precise as the value given in Table 2 of the ASME Steam Tables. To do this, several iterations are required, and the computation may take a minute or so. Rather than leave the screen blank, the error associated with each iteration is displayed on the screen. Thus, the user knows that the machine is functioning. Countdowns are displayed during calculations of enthalpy and specific volume for the same reason.

The properties of water and steam obtained in the Steam Tables Module are calculated from equations presented in "ASME Steam Tables," Fourth Edition, published by the American Society of Mechanical Engineers. For Sub-region 1, the reduced volume and reduced enthalpy equations on pages 23 and 24 are used.

For Sub-region 2, the equations are on pages 24 and 25. The equation for the saturation line, BETA.K, is given on page 17, and the equation for BETA.L, the boundary between Sub-region 2 and 3, is given on page 23.

8.11 Reports Module

From the menu in the Reports Module, the user can print a blank data sheet to be used for collection of data, print input data including the plant parameters, print a report of the heat balance, return to the Master Menu or go to the Termination Module.

Data sheets and input data are printed in compressed font, i.e., 17 characters per inch.

Heat balance reports are printed with pica font, i.e., 10 characters per inch. One page is required per data set. Again, formatting is appropriate for the number of loops.

8.12 Termination Module (END.BAS)

When the Termination Module is accessed, it first checks the parameters, data, and results storage files, to determine whether information is stored in them. If so, an appropriate message is displayed for the user on the screen and the user can either erase or save stored information. If the user responds by commanding that the information be erased, the user is asked to confirm the command. The module executes the command and then returns to the beginning of the Introduction Module or exits to DOS.9 TPDWR2 DISKETTE.

TPDWR2 is written on an application diskette, which can be supplied by the Office of Inspection and Enforcement. The files stored on the application diskette are:

| | |
|--------------|--------------|
| AUTOEXEC.BAT | REPORTS.BAS |
| T.BAT | END.BAS |
| TPDWR2.BAS | CALC.STO |
| INTRO.BAS | 369 |
| PRM&DATA.BAS | PLANT-ID.369 |
| PRM.BAS | SG-DATA.369 |
| DATA.BAS | OC-DATA.369 |
| ANALYSIS.BAS | RCS-CALC.369 |
| DISPLAY.BAS | SG-CALC.369 |
| TABLES.BAS | OC-CALC.369 |

The 369 files on the application diskette provide the user with an example of an actual application of the program. The reports for this example are shown on pages L-16, -17, and -18 of Appendix L.

Before TPDWR2 can be run, the user must prepare a working diskette as described in the Installation Procedure in Section 10. By executing this procedure, the user will create a diskette which will have stored on it the user's licensed copies of COMMAND.COM, the DOS interface routine, and BASIC.COM, as well as a copy of each of the TPDWR2 modules and files listed above. This diskette can accommodate up to an additional 70 docket files.

To check the validity of the software during its life, the user can run any of the six Program Performance Tests that are documented in Appendix L.

9 OPERATING PROCEDURE

Before TPDWR2 can be run the first time, a working diskette must be prepared by following the procedure described in Section 10. After the working diskette is prepared, TPDWR2 can be run at any time on either an IBM PC or XT.

If power is off, place the working diskette in Drive A and turn power on. TPDWR2 will automatically be loaded into memory and execution will start. The program will provide sufficient prompts so that a user who has not previously run the program can do so.

If power is on and the DOS prompt is displayed, i.e., A> or B>, then the user should place the working diskette in either Drive A or B, as appropriate, and press the return key. Execution can then proceed as above.

10 INSTALLATION PROCEDURE

The following procedure can be used to prepare a working diskette using the IBM PC:

1. Put your DOS diskette in Drive A and a new diskette in Drive B.
2. Execute A>FORMAT B:/S/V.

3. Title the diskette TPDWR2.
4. Execute A>COPY BASIC.COM B:/V.
5. Replace the DOS diskette in Drive A with the new diskette in Drive B. Put the TPDWR2 application diskette (supplied by the Office of Inspection and Enforcement) in Drive B.
6. Execute A>COPY B:*.*/V.
7. To test the new working diskette, execute A>T.

The following procedure can be used to prepare a working diskette using the IBM XT:

1. Put your DOS diskette in Drive A and execute A>FORMAT/S/V.
2. Follow the instructions on the screen. Title the diskette TPDWR2.
3. When formatting is completed, leave the new diskette in the drive, execute A>COPY B:BASIC.COM/V, and follow the instructions on the screen.
4. When copying is completed, replace the new diskette with the TPDWR2 application diskette (obtained from the Office of Inspection and Enforcement) and execute first A>MD C:DKJE2 and then A>COPY *.* C:DKJE2/V.
5. Replace the TPDWR2 application diskette with the new diskette, and execute A>COPY C:DKJE2*.*/V.
6. When copying is completed, first execute A>ERASE C:DKJE2*. * and then A>RD C:DKJE2.
7. To test the new working diskette, execute A>L.

APPENDIX A
BATCH ROUTINES

AUTUEXEC.BAT

```
basic tpdwr2/d/s:167
```

T.BAT

```
cls  
basic tpdwr2/d/s:167
```


APPENDIX B

TPDWR2.BAS

APPENDIX B: TPDWR2.BAS

```

550 '
560 REM Output
570 DIM BOTTOM.BLOW.ENTHALPY(4,2),BOTTOM.BLOW.LB.PER.HR(4,2),BOTTOM.BLOW.POWER(4,2),CHARGING.ENTHALPY(2)
580 DIM CHARGING.FLOW.LB.PER.HR(2),CHARGING.POWER(2),FEEDWATER.ENTHALPY(4,2),FEEDWATER.POWER(4,2)
590 DIM INSULATION.POWER(2),LETDOWN.ENTHALPY(2),LETDOWN.FLOW.LB.PER.HR(2),LETDOWN.POWER(2),OTHER.POWER(2)
600 DIM OTHER.POWER.MW(2),PZR.ENTHALPY(2),REACTOR.POWER.MW(2),SG.ENTHALPY(4,2),SG.POWER(4,2)
610 DIM STEAM.ENTHALPY(4,2),STEAM.FLOW(4,2),STEAM.POWER(4,2),TOP.BLOW.ENTHALPY(4,2)
620 DIM TOP.BLOW.LB.PER.HR(4,2),TOP.BLOW.POWER(4,2)
630 '
640 REM Unique Arrays
650 DIM EXCESS.FEEDWATER.FLOW(4),HR(2),LETDOWN.DENSITY(2),MIN(2),SG.POWER.MW(4,2),T.HOT(2),TIME.HR(2)
660 DIM TTIME$(2),TOP.BLOW.DENSITY(4,2),TOTAL.SG.POWER(2),B(10,7),DE4(6),DV2(6),SB(9,3),XZ(9,3),ZZ(9,4)
670 '
680 '
690 '                ** BRIDGE **
700 '
710 CLS : LOCATE 12,29 : PRINT "TPDWR2 IS BEING LOADED" : CHAIN "INTRO"
720 END

```

APPENDIX C

INTRO. BAS

TABLE OF CONTENTS

| | Page |
|----------------|--------------|
| Messages | ---- C- 1 |
| Printer | C- 3 |
| Docket Entry | C- 3 |
| Storage Status | C- 4 |
| Master Menu | C- 5 |

```

10 *
20 * *****
30 * ** INTRO.BAS **
40 * *****
50 *
60 * *****
70 * * COMMON *
80 * *****
90 DEFDBL A-Z : KEY OFF
100 *
110 REM Parameters
120 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
130 COMMON RISER.NUMBER,RISER.OD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,REFLECT.AREA
140 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
150 *
160 REM Data
170 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
180 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
190 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
200 *
210 REM Auxiliary Results
220 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT#
230 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
240 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
250 *
260 CLS : COLOR 7,0 : IF X$="INTRO 1920" THEN GOTO 2170
270 *
280 OPEN "CALC.STO" AS #1 LEN=8 : FIELD #1,1 AS CF$, 3 AS D$, 1 AS DF$, 1 AS PF$, 1 AS PRF$, 1 AS RF$ : GET #1
290 CALC.FLAG$=CF$ : DOCKET$=D$ : DATA.FLAG$=DF$ : PRINT.FLAG$=PF$ : PARAMETER.FLAG$=PRF$ : RESULTS.FLAG$=RF$
300 IF CALC.FLAG$="S" THEN CLOSE #1 : KILL "CALC.STO" : GOTO 2170
310 CLOSE #1
320 *
330 *
340 * *****
350 * * MESSAGES *
360 * *****
370 *
380 L= 7 : C=25 : GOSUB 440
390 F=25 : GOSUB 980 : CLS : L= 2 : C= 3 : GOSUB 440 : M= 9 : D=33 : GOSUB 550
400 F=45 : GOSUB 980 : CLS : L= 2 : C=44 : GOSUB 440 : M= 5 : D=32 : GOSUB 550 : N= 9 : E= 1 : GOSUB 680
410 F=45 : GOSUB 980 : CLS : M=-1 : D=31 : GOSUB 550 : N= 3 : E= 1 : GOSUB 680 : L=13 : C=37 : GOSUB 860
420 F=23 : GOSUB 980 : GOTO 1050
430 *
440 COLOR 0,7
450 LOCATE L+1,C : PRINT * *
460 LOCATE L+2,C : PRINT * * TPDWR2 *
470 LOCATE L+3,C : PRINT * *
480 LOCATE L+4,C : PRINT * * THERMAL POWER DETERMINATION *
490 LOCATE L+5,C : PRINT * * FOR *
500 LOCATE L+6,C : PRINT * * WESTINGHOUSE REACTORS *
510 LOCATE L+7,C : PRINT * *
520 LOCATE L+8,C : PRINT * * SEPTEMBER 30, 1985 *
530 LOCATE L+9,C : PRINT * * : RETURN *
540 *

```



```

1040 ' *****
1050 ' * PRINTER *
1060 ' *****
1070 '
1080 CLS : M= 2 : E=22 : GOSUB 680 : L=12: C=1 : GOSUB 860 : M=19 : D=38 : COLOR 0,7
1090 '
1100 LOCATE M+1,D : PRINT " "
1110 LOCATE M+2,D : PRINT " Will you use the printer during this "
1120 LOCATE M+3,D : PRINT " session ("; : COLOR 16,7 : PRINT "Y/N"; : COLOR 0,7
1130 PRINT ") "
1140 LOCATE M+4,D : PRINT " " : LOCATE M+3,D+16 : INPUT X$
1150 COLOR 7,0
1160 '
1170 IF X$="Y" OR X$="y" THEN PRINT.FLAG$="S" : GOTO 1190
1180 IF X$="N" OR X$="n" THEN PRINT.FLAG$="C" : GOTO 1270 ELSE GOTO 1080
1190 '
1200 CLS : LOCATE 8,38 : PRINT "NOTE" : PRINT
1210 PRINT TAB(20);"1. Is printer power on? If not, turn it"
1220 PRINT TAB(20);" on now." : PRINT
1230 PRINT TAB(20);"2. Is the printer on line? If not press"
1240 PRINT TAB(20);" the "; : COLOR 0,7 : PRINT " ON LINE "; : COLOR 7,0 : PRINT " switch."
1250 '
1260 F=22 : GOSUB 980
1270 IF Y$="SKIP.TO.MENU" THEN Y$="" : GOTO 2170
1280 '
1290 '
1300 ' *****
1310 ' * DOCKET ENTRY *
1320 ' *****
1330 '
1340 CLS : COLOR 7,0
1350 LOCATE 10,19 : PRINT "What are the last three digits of the docket"
1360 PRINT TAB(19); " number, including any leading zeros?" : PRINT
1370 PRINT TAB(19); " After entering the docket number, press the" : COLOR 0,7
1380 LOCATE 14,19 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" " : COLOR 7,0 : PRINT " key." : LOCATE 11,57
1390 INPUT "", DOCKET$
1400 '
1410 CLS : LOCATE 12,26 : PRINT "The docket number is 50-";DOCKET$;"."
1420 LOCATE 25,21 : PRINT "After responding, strike the "; : COLOR 16,7
1430 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
1440 COLOR 7,0 : PRINT " key.";
1450 LOCATE 13,26 : INPUT "Is it correct (Y/N)? ",X$
1460 IF X$="Y" OR X$="y" THEN 1490 ELSE IF X$="N" OR X$="n" THEN 1340 ELSE LOCATE 13,47 : PRINT X$
1470 LOCATE 15,29 : COLOR 0,7 : PRINT " PLEASE TYPE Y OR N! " : COLOR 7,0 : GOTO 1450
1480 '
1490 DOCKET.ENTRY.FLAG$="S"
1500 '
1510 OPEN DOCKET$ AS #1 LEN=159 : FIELD #1,158 AS X$,1 AS PF$ : GET #1,1 : PARAMETER.FLAG$=PF$
1520 IF (PARAMETER.FLAG$(">")"S" AND PARAMETER.FLAG$(">")"C") THEN PARAMETER.FLAG$="C" : LSET PF$=PARAMETER.FLAG$
: PUT #1,1 : CLOSE #1
1530 CLOSE #1
1540 '
1550 '

```


APPENDIX C: INTRO.BAS

```

1560 ' *****
1570 ' * STORAGE STATUS *
1580 ' *****
1590 '
1600 CLS : F=23
1610 '
1620 OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59
1630 FIELD #1, 32 AS P$, 1 AS U$, 20 AS D$, 3 AS DS$, 1 AS DF$, 2 AS DTS$
1640 GET #1,1 : DATA.FLAG$=DF$ : DATA.SETSZ=CVI(DTS$) : X$=" "
1650 FOR I%=1 TO 32 : NON.PLANT$=RIGHT$(P$,I%)
1660 IF NON.PLANT$<>X$ THEN PLANT$=LEFT$(P$,32-I%+1) : GOTO 1670 ELSE X$=X$+" " : NEXT
1670 UNIT$=U$ : DOCKET.STORED$=DS$ : DDATE$=D$
1680 IF (DATA.FLAG$<>"S" AND DATA.FLAG$<>"C") THEN DATA.FLAG$="C"
1690 LSET DF$=DATA.FLAG$ : PUT #1,1 : CLOSE 1
1700 '
1710 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=45
1720 FIELD #1,1 AS RF$,8 AS PF$,8 AS PP$,8 AS RPB$,20 AS D$ : GET #1,1 : RESULTS.FLAG$=RF$
1730 IF (RESULTS.FLAG$<>"S" AND RESULTS.FLAG$<>"C") THEN RESULTS.FLAG$="C"
1740 LSET RF$=RESULTS.FLAG$ : PUT #1,1 : CLOSE 1
1750 '
1760 IF DATA.FLAG$="S" THEN 1780 ELSE 2010
1770 '
1780 CLS : LOCATE 5,35 : COLOR 0,7 : PRINT " CAUTION " : COLOR 7,0 : F=22
1790 LOCATE 8,19 : IF DATA.SETSZ=1 THEN PRINT "One set"; ELSE IF DATA.SETSZ=2 THEN PRINT "Two sets";
1800 PRINT " of data for ---" : PRINT
1810 PRINT TAB((80-LEN(PLANT$+UNIT$)-1)/2);PLANT$; " ; UNIT$ : PRINT TAB(37);"50-";DOCKET.STORED$ : PRINT
1820 PRINT TAB(19); : IF DATA.SETSZ=1 THEN PRINT "is "; ELSE IF DATA.SETSZ=2 THEN PRINT "are";
1830 PRINT " stored on the diskette. If you enter"
1840 PRINT TAB(19);"new data under the same docket number,"
1850 PRINT TAB(19);"previous data will be lost. Results of"
1860 PRINT TAB(19);"analysis of those data are " : IF RESULTS.FLAG$="S" THEN PRINT "also"; ELSE PRINT "not ";
1870 PRINT " stored on"
1880 PRINT TAB(19);"this diskette. ";
1890 IF RESULTS.FLAG$="S" THEN PRINT "Analysis of new data will" ELSE PRINT " " : PRINT : GOTO 1910
1900 PRINT TAB(19);"destroy those results." : PRINT
1910 PRINT TAB(19);"Options are described on the next screen."
1920 GOSUB 980
1930 '
1940 CLS : LOCATE 10,33 : PRINT "*** OPTIONS ***" : PRINT
1950 PRINT TAB(11);"From the MASTER MENU, you will be able to make a hard copy"
1960 PRINT TAB(11);"of the old data and results, to overwrite the old data with"
1970 PRINT TAB(11);"new data, or to change dockets." : GOSUB 980 : CLS
1980 '
1990 '

```

```

2000 ' *****
2010 ' * MASTER MENU *
2020 ' *****
2030 '
2040 IF Y$="SKIP.TO.MENU" THEN Y$="" : GOTO 2170
2050 CLS : LOCATE 10,24 : F=23 : IF CALC.FLAG$="S" THEN 2140
2060 LOCATE 10,24 : PRINT "NOTE: The computer system is ready."
2070 PRINT TAB(24); "The MASTER MENU will be displayed"
2080 PRINT TAB(24); "next. Except when you terminate the"
2090 PRINT TAB(24); "session, you will always be able to"
2100 PRINT TAB(24); "return to the MASTER MENU."
2110 '
2120 GOSUB 980
2130 '
2140 OPEN "calc.sto" AS #1 LEN=5 : FIELD #1, 1 AS CF$, 4 AS I$
2150 CALC.FLAG$="C" : LSET CF$=CALC.FLAG$ : PUT #1,1 : CLOSE #1
2160 '
2170 CLS : F=27 : IF PARAMETER.FLAG$="S" THEN GOTO 2200
2180 LOCATE 23,19 : COLOR 0,7 : PRINT " Plant Parameters are needed. Select (0) " : COLOR 7,0
2190 '
2200 LOCATE 25,21 : PRINT "After responding, strike the ";
2210 COLOR 16,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" " : COLOR 7,0 : PRINT " key."
2220 '
2230 IF PRINT.FLAG$(">S" THEN PRINT.FLAG$="C"
2240 TX= 5 : BX=19 : LX=11 : RX=70
2250 LOCATE TX,LX : PRINT CHR$(201) : LOCATE TX,RX : PRINT CHR$(187) : LOCATE BX,RX : PRINT CHR$(188)
2260 LOCATE BX,LX : PRINT CHR$(200)
2270 LOCATE TX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205); : NEXT
2280 LOCATE BX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205); : NEXT
2290 FOR IX=TX+2 TO BX : LOCATE IX-1,LX : PRINT CHR$(186) : NEXT
2300 FOR IX=TX+2 TO BX : LOCATE IX-1,RX : PRINT CHR$(186) : NEXT
2310 LOCATE 5,34 : PRINT " MASTER MENU " : A=17 : B=47
2320 LOCATE 8,A : PRINT "(A) Blank Data Sheet"; : PRINT TAB(B);"(F) Calculator"
2330 LOCATE 10,A : PRINT "(B) Parameters & Data"; : PRINT TAB(B);"(G) Change Dockets"
2340 LOCATE 12,A : PRINT "(C) Analysis"; : PRINT TAB(B);"(H) Printer"
2350 LOCATE 14,A : PRINT "(D) Steam Tables"; : PRINT TAB(B);"(I) Preface"
2360 LOCATE 16,A : PRINT "(E) Reports"; : PRINT TAB(B);"(J) Termination"
2370 LOCATE 19,62 : PRINT " " : LOCATE 19,17 : INPUT " Which would you like (A/B/C/D/E/F/G/H/I/J)";X$
2380 '
2390 CLS
2400 IF DIAGNOSTIC.FLAG$(">S" THEN GOTO 2430
2410 LPRINT TAB(12);"X$=";X$;" DIAGNOSTIC.FLAG$=";DIAGNOSTIC.FLAG$;" (INTRO, LINE 2410)"
2420 '
2430 IF X$="DIAGNOSTIC" OR X$="diagnostic" THEN 2440 ELSE 2450
2440 DIAGNOSTIC.FLAG$="S" : LOCATE 24,60 : COLOR 0,7 : PRINT "DIAGNOSTIC" : COLOR 7,0
2450 IF (X$="A" OR X$="a") AND PRINT.FLAG$="S" THEN X$="DATA SHEET" : GOSUB 2590 : CHAIN "REPORTS"
2460 IF (X$="A" OR X$="a") AND PRINT.FLAG$="C" THEN GOTO 2650
2470 IF X$="B" OR X$="b" THEN GOSUB 2600 : CHAIN "PRM&DATA"
2480 IF X$="C" OR X$="c" THEN GOSUB 2610 : CHAIN "ANALYSIS"
2490 IF X$="D" OR X$="d" THEN X$="MENU" : GOSUB 2620 : CHAIN "TABLES"
2500 IF (X$="E" OR X$="e") AND PRINT.FLAG$="S" THEN GOSUB 2590 : CHAIN "REPORTS"
2510 IF (X$="E" OR X$="e") AND PRINT.FLAG$="C" THEN GOTO 2650
2520 IF X$="F" OR X$="f" THEN CALC.FLAG$="S" : GOSUB 2980 : GOTO 2710
2530 IF X$="G" OR X$="g" THEN DOCKET.ENTRY.FLAG$="C" : Y$="SKIP.TO.MENU" : GOTO 1310

```

APPENDIX C: INT⁰.BAS

```

2540 IF X$="H" OR X$="h" THEN Y$="SKIP.TO.MENU" : GOTO 1050
2550 IF X$="I" OR X$="i" THEN Y$="SKIP.TO.MENU" : GOTO 350
2560 IF X$="J" OR X$="j" THEN GOSUB 2630 : CHAIN "END" ELSE GOTO 2170
2570 '
2580 PRINT CHR$(12) : LOCATE 13,35 : PRINT "PLEASE WAIT" : RETURN
2590 LOCATE 12,23 : PRINT "LOADING REPORTS MODULE INTO MEMORY" : RETURN
2600 LOCATE 12,18 : PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY" : RETURN
2610 LOCATE 12,23 : PRINT "LOADING ANALYSIS MODULE INTO MEMORY" : RETURN
2620 LOCATE 12,20 : PRINT "LOADING STEAM TABLES MODULE INTO MEMORY" : RETURN
2630 LOCATE 12,25 : PRINT "LOADING END MODULE INTO MEMORY" : RETURN
2640 '
2650 CLS : LOCATE 11,16
2660 PRINT "The printer option was not selected. Select the"
2670 PRINT TAB(16) "printer option so that data sheets or reports can"
2680 PRINT TAB(16) "be printed." : F=23
2690 GOSUB 980 : Y$="SKIP.TO.MENU" : CLS : M=10 : D=19 : GOTO 1050
2700 '
2710 CLS : KEY 2,"RUN"+CHR$(13)
2720 CLS : TX= 1 : BX=13 : LX= 1 : RX=80
2730 LOCATE TX,LX : PRINT CHR$(201) : LOCATE TX,RX : PRINT CHR$(187) : LOCATE BX,RX : PRINT CHR$(188)
2740 LOCATE BX,LX : PRINT CHR$(200)
2750 LOCATE TX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) : NEXT
2760 LOCATE BX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) : NEXT
2770 FOR IX=TX+2 TO BX : LOCATE IX-1,LX : PRINT CHR$(186) : NEXT
2780 FOR IX=TX+2 TO BX : LOCATE IX-1,RX : PRINT CHR$(186) : NEXT
2790 '
2800 LOCATE 1,29 : PRINT " CALCULATOR PROCEDURE "
2810 LOCATE 2, 5 : PRINT "You are in the calculator mode. When the prompt, OK, appears:"
2820 LOCATE 3, 7 : PRINT "(a) Type "; : COLOR 0,7 : PRINT "? "; : COLOR 7,0 : PRINT", then"
2830 LOCATE 4, 7 : PRINT "(b) Type the mathematical expression to be evaluated, and then"
2840 LOCATE 5, 7 : PRINT "(c) Press the "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
2850 COLOR 7,0 : PRINT " key."
2860 LOCATE 6, 5 : PRINT "Terms in the expression will be evaluated in the order of their rank"
2870 LOCATE 7, 5 : PRINT "in the hierarchy of operations and from left to right. The heirarchy is:"
2880 LOCATE 8, 7 : PRINT "(a) Inner parentheses,"
2890 LOCATE 9, 7 : PRINT "(b) Outer parentheses,"
2900 LOCATE 10, 7 : PRINT "(c) Exponentiation, which is indicated by "; : COLOR 0,7 : PRINT " ^ "; : COLOR 7,0
2910 PRINT " ,"
2920 LOCATE 11, 7 : PRINT "(d) Multiplication and division, which are indicated by "; : COLOR 0,7
2930 PRINT " * "; : COLOR 7,0 : PRINT " and "; : COLOR 0,7 : PRINT " / "; : COLOR 7,0 : PRINT " , and"
2940 LOCATE 12, 7 : PRINT "(e) Addition and subtraction, which are indicated by "; : COLOR 0,7 : PRINT " + ";
2950 COLOR 7,0 : PRINT " and "; : COLOR 0,7 : PRINT " - "; : COLOR 7,0 : PRINT " ."
2960 GOTO 3020
2970 '
2980 OPEN "CALC.STO" AS #1 LEN=8 : FIELD #1,1 AS CF$, 3 AS D$, 1 AS DF$, 1 AS PF$, 1 AS PRF$, 1 AS RF$
2990 LSET CF$=CALC.FLAG$ : LSET D$=DOCKET$ : LSET DF$=DATA.FLAG$ : LSET PF$=PRINT.FLAG$
3000 LSET PRF$=PARAMETER.FLAG$ : LSET RF$=RESULTS.FLAG$ : PUT #1 : CLOSE #1 : RETURN
3010 '
3020 LOCATE 25,20 : COLOR 7,0 : PRINT "To return to TPDWR2, press the "; : COLOR 16,7 : PRINT " F2 ";
3030 COLOR 7,0 : PRINT " key." : LOCATE 15,1
3040 END

```

APPENDIX D

PRM&DATA.BAS

TABLE OF CONTENTS

| | Page |
|---------|------|
| | ---- |
| Menu | D- 1 |
| Storage | D- 2 |
| Recall | D- 3 |
| Display | D- 5 |

```

10  *
20  *
30  *
40  *
50  *
60  *
70  *
80  *
90  DEFDSL A-Z : COLOR 7,0 : CLS
100 *
110 REM Parameters
120 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
130 COMMON RISER.NUMBER,RISER.OD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,REFLECT.AREA
140 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
150 *
160 REM Data
170 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
180 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
190 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
200 *
210 REM Auxiliary Results
220 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),X,P,PSAT$
230 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
240 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
250 *
260 IF DIAGNOSTIC.FLAG$<>"S" THEN GOTO 290
270 LPRINT TAB(12);"DIAGNOSTIC.FLAG$=";DIAGNOSTIC.FLAG$;" X$=";X$;" (DATA, LINE 362)"
280 *
290 IF X$="PRM&DATA 340" THEN 370
300 IF X$="PRM&DATA 740" THEN GOSUB 870 : GOSUB 1090 : ELSE IF X$="PRM&DATA 6270" THEN GOSUB 1060
310 IF X$="INPUT DATA" OR X$="ANALYSIS" OR X$="HEAT BALANCE" OR X$="DATA SET 2 PREP" THEN GOTO 1410
320 IF PARAMETER.FLAG$="S" THEN CLS : GOTO 370 ELSE CLS : LOCATE 20,12
330 COLOR 0,7 : PRINT " NOTE: Plant Parameters are needed. First, select (A). " : COLOR 7,0 : PRINT
340 *
350 *
360 *
370 *
380 *
390 *
400 GOSUB 700 : TX= 8 : BX=17 : LX= 6 : RX=76
410 LOCATE TX,LX : PRINT CHR$(201) : LOCATE TX,RX : PRINT CHR$(137) : LOCATE BX,RX : PRINT CHR$(188)
420 LOCATE BX,LX : PRINT CHR$(200)
430 LOCATE TX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205); : NEXT
440 LOCATE BX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205); : NEXT
450 FOR IX=TX+2 TO BX : LOCATE IX-1,LX : PRINT CHR$(186) : NEXT
460 FOR IX=TX+2 TO BX : LOCATE IX-1,RX : PRINT CHR$(186) : NEXT
470 LOCATE 8,29 : PRINT " PARAMETERS & DATA MENU " : LOCATE 11,10 : PRINT "Plant Parameters:";TAB(37);
480 PRINT "Test Data:";TAB(62);"Other:" : LOCATE 13,13 : PRINT "(A) Enter";TAB(37);"(C) Enter";TAB(58);
490 PRINT "(E) Display" : LOCATE 14,13 : PRINT "(B) Correct";TAB(37);"(D) Correct";TAB(58);"(F) MASTER MENU"
500 LOCATE 17,59 : PRINT " " : LOCATE 17,22 : INPUT " Which would you like (A/B/C/D/E/F)? ",X$
510 *
520 CLS
530 *

```

APPENDIX D: PRM&DATA.BAS

```

540 IF X$="A" OR X$="a" THEN GOSUB 650 : CHAIN "PRM"
550 IF X$="B" OR X$="b" THEN GOSUB 1440 : GOSUB 650 : X$="PRM 3840" : CHAIN "PRM"
560 IF X$="C" OR X$="c" THEN GOSUB 1440 : GOSUB 660 : X$="DATA" : CHAIN "DATA"
570 IF X$="D" OR X$="d" THEN GOSUB 1440 : GOSUB 1690 : GOSUB 660 : X$="DATA 1610" : CHAIN "DATA"
580 IF X$="E" OR X$="e" THEN GOSUB 1440 : GOSUB 1690 : GOTO 2300
590 IF X$="F" OR X$="f" THEN GOTO 3710 ELSE 370
600 '
610 '
620 '                ** MESSAGES **
630 '
640 CLS : LOCATE 13,20 : PRINT "LOADING INTRODUCTION MODULE INTO MEMORY" : RETURN
650 CLS : LOCATE 13,21 : PRINT "LOADING PARAMETERS MODULE INTO MEMORY" : RETURN
660 CLS : LOCATE 13,24 : PRINT "LOADING DATA MODULE INTO MEMORY" : RETURN
670 CLS : LOCATE 13,22 : PRINT "LOADING ANALYSIS MODULE INTO MEMORY" : RETURN
680 CLS : LOCATE 13,23 : PRINT "LOADING REPORTS MODULE INTO MEMORY" : RETURN
690 '
700 LOCATE 25,21 : PRINT "After responding, strike the "; : COLOR 16,7 : PRINT " ";
CHR$(17)+CHR$(196)+CHR$(217);" ";
710 COLOR 7,0 : PRINT " key." : RETURN
720 '
730 '
740 '                *****
750 '                * CONTINUE *
760 '                *****
770 '
780 LOCATE 25,23 : PRINT " To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0
790 PRINT ".";
800 Z$=INKEY$ : IF Z$(>)CHR$(32) THEN 800 ELSE CLS : RETURN
810 '
820 '
830 '                *****
840 '                * STORAGE *
850 '                *****
860 '
870 '                ** PLANT PARAMETERS **
880 '
890 PARAMETER.FLAG$="S"
900 OPEN DOCKET$ AS #1 LEN=167
910 FIELD #1,32 AS P$,1 AS U$,3 AS D$,2 AS L$,8 AS RPE$,8 AS RPP$,8 AS SI$,8 AS RO$,8 AS RN$,8 AS COA$,
8 AS COB$,8 AS COC$,8 AS COD$,8 AS PI$,8 AS RA$,8 AS LC$,8 AS NA$,8 AS IT$,8 AS C$,1 AS PF$,8 AS LP$
920 LSET P$ = PLANT$ : LSET U$ = UNIT$
930 LSET D$ = DOCKET$ : LSET PF$ = PARAMETER.FLAG$
940 LSET L$ = MKI$(LOOP$Z) : LSET RPE$ = MKD$(RC.PUMP.EFF)
950 LSET SI$ = MKD$(SG.ID) : LSET RPP$ = MKD$(RC.PUMP.PWR)
960 LSET RN$ = MKD$(RISER.NUMBER) : LSET RO$ = MKD$(RISER.OD)
970 LSET COA$ = MKD$(CARRYOVER.A) : LSET COB$ = MKD$(CARRYOVER.B)
980 LSET COC$ = MKD$(CARRYOVER.C) : LSET COD$ = MKD$(CARRYOVER.D)
990 LSET RA$ = MKD$(REFLECT.AREA) : LSET LC$ = MKD$(LOSS.COEFFICIENT)
1000 LSET NA$ = MKD$(NONREFLECT.AREA) : LSET C$ = MKD$(CONDUCTIVITY)
1010 LSET IT$ = MKD$(INSULATION.THICKNESS) : LSET PI$ = MKD$(PZR.ID)
1020 LSET LP$ = MKD$(LIC.POWER)
1030 PUT #1,1 : CLOSE #1 : RETURN
1040 '
1050 '

```

```

1060 '                                     ** TEST DATA **
1070 '
1080 DATA.FLAG$="S"
1090 DOCKET.STORED$ = DOCKET$
1100 '
1110 OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59
1120 FIELD #1, 32 AS P$, 1 AS U$, 20 AS D$, 3 AS DS$, 1 AS DF$, 2 AS DTS$
1130 LSET P$ = PLANT$ : LSET U$ = UNIT$ : LSET D$ = DDATE$ : LSET DS$ = DOCKET.STORED$
1140 LSET DF$ = DATA.FLAG$ : LSET DTS$ = MKI$(DATA.SETSZ) : PUT #1,1 : CLOSE #1
1150 '
1160 OPEN "SG-DATA."+DOCKET$ AS #2 LEN=48
1170 FIELD #2, 8 AS S$, 8 AS WF$, 8 AS WT$, 8 AS TB$, 8 AS BB$, 8 AS WL$
1180 '
1190 FOR TZ=1 TO 2 : FOR LP%=1 TO LOOPSZ
1200 '
1210 LSET S$ = MKD$(STEAM.PRES(LP%,TZ)) : LSET TB$ = MKD$(TOP.BLOW(LP%,TZ))
1220 LSET WF$ = MKD$(FEEDWATER.FLOW(LP%,TZ)) : LSET BB$ = MKD$(BOTTOM.BLOW(LP%,TZ))
1230 LSET WT$ = MKD$(FEEDWATER.TEMP(LP%,TZ)) : LSET WL$ = MKD$(SG.WATER.LEVEL(LP%,TZ))
1240 '
1250 PUT #2, (TZ-1)*LOOPSZ + LP% : NEXT LP%,TZ : CLOSE #2
1260 '
1270 OPEN "OC-DATA."+DOCKET$ AS #3 LEN=72
1280 FIELD #3,8 AS T$,8 AS LF$,8 AS LT$,8 AS CF$,8 AS CT$,8 AS PP$,8 AS PW$,8 AS TA$,8 AS TC$
1290 '
1300 FOR TZ= 1 TO 2
1310 LSET T$ = MKD$(TTIME(TZ)) : LSET PP$ = MKD$(PZR.PRES(TZ))
1320 LSET LF$ = MKD$(LETDOWN.FLOW(TZ)) : LSET PW$ = MKD$(PZR.WATER.LEVEL(TZ))
1330 LSET LT$ = MKD$(LETDOWN.TEMP(TZ)) : LSET TA$ = MKD$(T.AVE(TZ))
1340 LSET CF$ = MKD$(CHARGING.FLOW(TZ)) : LSET TC$ = MKD$(T.COLD(TZ))
1350 LSET CT$ = MKD$(CHARGING.TEMP(TZ))
1360 '
1370 PUT #3,TZ : NEXT : CLOSE #3 : RETURN
1380 '
1390 '
1400 '                                     *****
1410 '                                     * RECALL *
1420 '                                     *****
1430 '
1440 '                                     ** PLANT PARAMETERS **
1450 '
1460 IF PARAMETER.FLAG$="S" THEN GOTO 1480 ELSE 2230
1470 '
1480 OPEN DOCKET$ AS #1 LEN=167
1490 FIELD #1,32 AS P$,1 AS U$,3 AS D$,2 AS L$,8 AS RPE$,8 AS RPP$,8 AS SI$,8 AS RO$,8 AS RN$,8 AS COA$,
      8 AS COB$,8 AS COC$,8 AS COD$,8 AS PI$,8 AS RA$,8 AS LC$,8 AS NA$,8 AS IT$,8 AS C$,1 AS PF$,8 AS LP$
1500 '
1510 GET #1,1
1520 '
1530 PLANT$           = P$           : RISER.NUMBER       = CVD(RN$) : INSULATION.THICKNESS = CVD(IT$)
1540 UNIT$           = U$           : RISER.OD        = CVD(RO$) : CONDUCTIVITY         = CVD(C$)
1550 DOCKET$        = D$           : REFLECT.AREA    = CVD(RA$) : PZR.ID               = CVD(PI$)
1560 LOOPSZ         = CVI(L$)      : NONREFLECT.AREA = CVD(NA$) : RC.PUMP.EFF          = CVD(RPE$)
1570 RC.PUMP.PWR    = CVD(RPP$)    : SG.ID           = CVD(SI$) : LOSS.COEFFICIENT     = CVD(LC$)

```


APPENDIX D: PRM&DATA.BAS

```

1580 PARAMETER.FLAG$ = PF$      : CARRYOVER.A      = CVD(COA$) : CARRYOVER.B      = CVD(COB$)
1590 CARRYOVER.C      = CVD(COC$) : CARRYOVER.D      = CVD(COD$) : LIC.POWER      = CVD(LP$)
1600 '
1610 FOR IZ=32 TO 1 STEP -1
1620 IF MID$(PLANT$,IZ,1)<>' * THEN PLANT$=LEFT$(PLANT$,IZ) : GOTO 1650
1630 NEXT IZ
1640 '
1650 CLOSE #1
1660 IF X$="INPUT DATA" OR X$="ANALYSIS" OR X$="DATA SET 2 PREP" THEN GOTO 1690 ELSE RETURN
1670 '
1680 '
1690 '
1700 '
1710 OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59 : FIELD #1, 32 AS P$, 1 AS UN$, 20 AS D$, 3 AS DS$, 1 AS DF$, 2 AS DTS$
1720 '
1730 GET #1,1 : UNIT$ = UN$ : DDATE$ = D$ : DOCKET.STORED$ = DS$ : DATA.FLAG$ = DF$ : DATA.SETSZ = CVI(DTS$)
1740 '
1750 FOR IZ=20 TO 1 STEP -1
1760 IF MID$(DDATE$,IZ,1)<>' * THEN DDATE$=LEFT$(DDATE$,IZ) : GOTO 1790
1770 NEXT IZ
1780 '
1790 CLOSE #1
1800 '
1810 FOR IZ=19 TO 1 STEP -1
1820 IF MID$(DDATE$,IZ,1)<>' * THEN DDATE$=LEFT$(DDATE$,IZ) : GOTO 1850
1830 NEXT IZ
1840 '
1850 IF DATA.FLAG$="C" THEN GOTO 2190
1860 '
1870 OPEN "SG-DATA."+DOCKET$ AS #2 LEN=48
1880 FIELD #2, 8 AS S$, 8 AS WF$, 8 AS WT$, 8 AS TB$, 8 AS BB$, 8 AS WL$
1890 '
1900 FOR TZ=1 TO 2
1910 FOR LPZ=1 TO LOOPSZ
1920 '
1930 GET #2, (TZ-1) * LOOPSZ + LPZ
1940 STEAM.PRES(LPZ,TZ)      = CVD(S$) : FEEDWATER.FLOW(LPZ,TZ) = CVD(WF$) : TOP.BLOW(LPZ,TZ)      = CVD(TB$)
1950 SG.WATER.LEVEL(LPZ,TZ) = CVD(WL$) : FEEDWATER.TEMP(LPZ,TZ) = CVD(WT$) : BOTTOM.BLOW(LPZ,TZ) = CVD(BB$)
1960 '
1970 NEXT LPZ
1980 NEXT TZ
1990 '
2000 '
2010 CLOSE #2
2020 '
2030 OPEN "OC-DATA."+DOCKET$ AS #3 LEN=72
2040 FIELD #3,8 AS T$,8 AS LF$,8 AS LT$,8 AS CF$,8 AS CT$,8 AS PP$,8 AS PW$,8 AS TA$,8 AS TC$
2050 '
2060 FOR TZ=1 TO 2 : GET #3, TZ
2070 TTIME(TZ)      = CVD(T$) : CHARGING.FLOW(TZ)      = CVD(CF$) : PZR.WATER.LEVEL(TZ) = CVD(PW$)
2080 LETDOWN.FLOW(TZ) = CVD(LF$) : CHARGING.TEMP(TZ)      = CVD(CT$) : T.AVE(TZ)      = CVD(TA$)
2090 LETDOWN.TEMP(TZ) = CVD(LT$) : PZR.PRES(TZ)          = CVD(PP$) : T.COLD(TZ)      = CVD(TC$)
2100 '
2110 NEXT : CLOSE #3

```

```

2120 '
2130 IF DATA.FLAG$="S" THEN 2150 ELSE 2190
2140 '
2150 IF X$="ANALYSIS" THEN GOSUB 670 : CHAIN "ANALYSIS" ELSE IF X$="INPUT DATA" THEN GOSUB 680 : CHAIN "REPORTS"
2160 IF X$="DATA SET 2 PREP" THEN GOSUB 660 : CHAIN "DATA"
2170 RETURN
2180 '
2190 CLS : LOCATE 12,24 : PRINT "Data Storage is empty. Return to"
2200 PRINT TAB(24); "Parameters & Data Menu, and select"
2210 PRINT TAB(24); "Option (C)." : GOSUB 750 : CLOSE : GOTO 370
2220 '
2230 LOCATE 11,24 : PRINT "Parameter storage for Docket 50-";DOCKET$
2240 PRINT TAB(24); "is empty. From the Parameters & Data"
2250 PRINT TAB(24); "Menu, select Option (A)."

```

APPENDIX D: PRM&DATA.BAS

```

2660 LOCATE LZ+ 5,CZ-1 : PRINT USING "###.##"; PZR.ID
2670 LOCATE LZ+ 6,CZ : PRINT USING "##.##"; RC.PUMP.PWR
2680 LOCATE LZ+ 7,CZ : PRINT USING "##.##"; RC.PUMP.EFF
2690 LOCATE LZ+11,CZ-2 : PRINT USING "#####"; REFLECT.AREA
2700 LOCATE LZ+12,CZ : PRINT USING "###.##"; LOSS.COEFFICIENT
2710 LOCATE LZ+16,CZ-2 : PRINT USING "#####"; NONREFLECT.AREA
2720 LOCATE LZ+17,CZ+1 : PRINT USING "##.##"; INSULATION.THICKNESS
2730 LOCATE LZ+18,CZ+2 : PRINT USING "###.##"; CONDUCTIVITY : GOSUB 2970
2740 '
2750 '
2760 ' * Steam Generators *
2770 '
2780 CLS : COLOR 7,0 : NZ=-2 : IF LOOPSZ=4 THEN LZ=4 ELSE IF LOOPSZ=3 THEN LZ=5 ELSE LZ=6
2790 LOCATE LZ+ 1,27 : PRINT "STEAM GENERATOR PARAMETERS"
2800 LOCATE LZ+ 4,BZ : PRINT "Steam dome inside diameter (inches) "
2810 LOCATE LZ+ 6,BZ : PRINT "Number of swirl vane risers "
2820 LOCATE LZ+ 7,BZ : PRINT "Outside diameter of swirl vane risers (inches) "
2830 LOCATE LZ+ 9,BZ : PRINT "Measured or warranted moisture carry-over (%):"
2840 LOCATE LZ+11,BZ+NZ+4 : PRINT "Steam Generator A "
2850 LOCATE LZ+12,BZ+NZ+4 : PRINT "Steam Generator B " : IF LOOPSZ=3 OR LOOPSZ=4 THEN 2860 ELSE 2880
2860 LOCATE LZ+13,BZ+NZ+4 : PRINT "Steam Generator C " : IF LOOPSZ=4 THEN 2870 ELSE 2880
2870 LOCATE LZ+14,BZ+NZ+4 : PRINT "Steam Generator D "
2880 CZ=70
2890 LOCATE LZ+ 4,CZ-1 : PRINT USING "###.##"; SG.ID
2900 LOCATE LZ+ 7,CZ : PRINT USING "##.##"; RISER.OD
2910 LOCATE LZ+ 6,CZ : PRINT USING "##"; RISER.NUMBER
2920 LOCATE LZ+11,CZ+1 : PRINT USING "###.##"; CARRYOVER.A
2930 LOCATE LZ+12,CZ+1 : PRINT USING "###.##"; CARRYOVER.B : IF LOOPSZ=3 OR LOOPSZ=4 THEN 2940 ELSE 2960
2940 LOCATE LZ+13,CZ+1 : PRINT USING "###.##"; CARRYOVER.C : IF LOOPSZ=4 THEN 2950 ELSE 2960
2950 LOCATE LZ+14,CZ+1 : PRINT USING "###.##"; CARRYOVER.D
2960 GOSUB 2970 : GOTO 3000
2970 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT " To continue, press the ";
2980 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".";
2990 Z$=INKEY$ : IF Z$(<CHR$(32)) THEN 2990 ELSE RETURN
3000 '
3010 '
3020 ' ** DATA **
3030 '
3040 FOR IZ=1 TO DATA.SETSZ : CLS
3050 '
3060 LOCATE 1,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3070 LOCATE 3,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3080 LOCATE 5,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3090 LOCATE 14,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3100 LOCATE 19,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3110 LOCATE 24,2 : FOR IZ=2 TO 78 : PRINT CHR$(196); : NEXT
3120 FOR IZ= 2 TO 23 : LOCATE IZ, 1 : PRINT CHR$(179) : NEXT : LOCATE 4,40 : PRINT CHR$(179)
3130 FOR IZ=15 TO 23 : LOCATE IZ,40 : PRINT CHR$(179) : NEXT
3140 FOR IZ= 2 TO 23 : LOCATE IZ,79 : PRINT CHR$(179); : NEXT
3150 LOCATE 1, 1 : PRINT CHR$(218);: LOCATE 1,79 : PRINT CHR$(191);
3160 LOCATE 3, 1 : PRINT CHR$(195);: LOCATE 3,79 : PRINT CHR$(180);
3170 LOCATE 5, 1 : PRINT CHR$(195);: LOCATE 5,79 : PRINT CHR$(180);
3180 LOCATE 14, 1 : PRINT CHR$(195);: LOCATE 14,79 : PRINT CHR$(180);
3190 LOCATE 19, 1 : PRINT CHR$(195);: LOCATE 19,79 : PRINT CHR$(180);

```

```

3200 LOCATE 3,40 : PRINT CHR$(194); : LOCATE 5,40 : PRINT CHR$(193);
3210 LOCATE 14,40 : PRINT CHR$(194); : LOCATE 19,40 : PRINT CHR$(197);
3220 LOCATE 24,40 : PRINT CHR$(193);
3230 LOCATE 24, 1 : PRINT CHR$(192); : LOCATE 24,79 : PRINT CHR$(217); : LOCATE 1,1
3240 COLOR 7,0 : BX=5
3250 '
3260 LOCATE 1,65 : PRINT " Set";TZ;"of";DATA.SETSZ
3270 LOCATE 2,36 : PRINT "TEST DATA"
3280 LOCATE 4, 8 : PRINT "Date: "; : LOCATE 4,15 : PRINT USING "\          \"; DDATE$
3290 LOCATE 4,50 : PRINT "Time (hhmm): "; : LOCATE 4,64 : PRINT TTIME$(TZ)
3300 LOCATE 6, 3 : PRINT "Steam Generators:"
3310 LOCATE 8, 5 : PRINT "Steam pressure (psia)"
3320 LOCATE 9, 5 : PRINT "Feedwater flow (millions of lb/hr)"
3330 LOCATE 10, 5 : PRINT "Feedwater temperature (";CHR$(248);"F)"
3340 LOCATE 11, 5 : PRINT "Surface blowdown (gpm)"
3350 LOCATE 12, 5 : PRINT "Bottom blowdown (gpm)"
3360 LOCATE 13, 5 : PRINT "Water level (inches)"
3370 LOCATE 15, 3 : PRINT "Letdown Line:"
3380 LOCATE 17, 5 : PRINT "Flow (gpm)"
3390 LOCATE 18, 5 : PRINT "Temperature (";CHR$(248);"F)"
3400 LOCATE 20, 3 : PRINT "Pressurizer:"
3410 LOCATE 22, 5 : PRINT "Pressure (psia)"
3420 LOCATE 23, 5 : PRINT "Water Level (inches)"
3430 LOCATE 15,42 : PRINT "Charging Line:"
3440 LOCATE 17,44 : PRINT "Flow (gpm)"
3450 LOCATE 18,44 : PRINT "Temperature (";CHR$(248);"F)"
3460 LOCATE 20,42 : PRINT "Reactor:"
3470 LOCATE 22,44 : PRINT "T ave (";CHR$(248);"F)"
3480 LOCATE 23,44 : PRINT "T cold (";CHR$(248);"F)"
3490 FOR LPZ=1 TO LOOPSZ : IF LPZ=1 THEN S$="A" ELSE IF LPZ=2 THEN S$="B" ELSE IF LPZ=3 THEN S$="C" ELSE S$="D"
3500 '
3510 IF LOOPSZ=2 THEN CZ=52 ELSE IF LOOPSZ=3 THEN CZ=42 ELSE IF LOOPSZ=4 THEN CZ=32
3520 DZ=CZ+10*LPZ
3530 '
3540 LOCATE 6,DZ+2 : PRINT S$
3550 LOCATE 8,DZ-1 : PRINT USING "###.#";STEAM.PRES(LPZ,TZ)
3560 LOCATE 9,DZ : PRINT USING "#.###";FEEDWATER.FLOW(LPZ,TZ)
3570 LOCATE 10,DZ : PRINT USING "###.#";FEEDWATER.TEMP(LPZ,TZ)
3580 LOCATE 11,DZ : PRINT USING "###.#";TOP.BLOW(LPZ,TZ)
3590 LOCATE 12,DZ : PRINT USING "###.#";BOTTOM.BLOW(LPZ,TZ)
3600 LOCATE 13,DZ : PRINT USING "###.#";SG.WATER.LEVEL(LPZ,TZ) : NEXT
3610 LOCATE 17,33 : PRINT USING "###.#";LETDOWN.FLOW(TZ)
3620 LOCATE 19,33 : PRINT USING "###.#";LETDOWN.TEMP(TZ)
3630 LOCATE 22,32 : PRINT USING "###.#";PZR.PRES(TZ)
3640 LOCATE 23,33 : PRINT USING "###.#";PZR.WATER.LEVEL(TZ)
3650 LOCATE 17,72 : PRINT USING "###.#";CHARGING.FLOW(TZ)
3660 LOCATE 18,72 : PRINT USING "###.#";CHARGING.TEMP(TZ)
3670 LOCATE 22,72 : PRINT USING "###.#";T.AVE(TZ)
3680 LOCATE 23,72 : PRINT USING "###.#";T.COLD(TZ) : GOSUB 2970 : CLS : NEXT TZ
3690 CLS : GOTO 370
3700 '
3710 X$="INTRO 1920" : GOSUB 600 : CHAIN "INTRO"
3720 '
3730 END

```

APPENDIX E

PRM.BAS

TABLE OF CONTENTS

| | Page |
|------------------------|------|
| | ---- |
| MENU | E- 1 |
| INPUT | E- 2 |
| Plant Identification | E- 2 |
| Reactor Cooling System | E- 3 |
| Steam Generators | E- 4 |
| Docket Check | E- 6 |
| CORRECTIONS | E- 6 |
| Plant Identification | E- 6 |
| Reactor Cooling System | E- 7 |
| Steam Generators | E-10 |
| HELP MESSAGES | E-11 |
| DISPLAYS | E-15 |

```

10 ' *****
20 ' ** PRM.BAS **
30 ' *****
40 '
50 ' *****
60 ' * COMMON *
70 ' *****
80 '
90 DEFDBL A-Z : COLOR 7,0 : CLS
100 '
110 REM Parameters
120 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
130 COMMON RISER.NUMBER,RISER.OD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOP$,REFLECT.AREA
140 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
150 '
160 REM Data
170 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
180 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),T.TIME(),TOP.BLOW()
190 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
200 '
210 REM Auxiliary Results
220 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT$
230 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
240 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
250 '
260 '
270 ' *****
280 ' * MENU *
290 ' *****
300 '
310 IF X$="PRM 3840" THEN 3060
320 IF PARAMETER.FLAG$="S" THEN 340 ELSE 560
330 '
340 GOSUB 7710 : TX= 7 : BX=18 : LX= 15 : RX=66
350 LOCATE TX,LX : PRINT CHR$(201) : LOCATE TX,RX : PRINT CHR$(187) : LOCATE BX,RX : PRINT CHR$(188)
360 LOCATE BX,LX : PRINT CHR$(200)
370 LOCATE TX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) : NEXT
380 LOCATE BX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) : NEXT
390 FOR IX=TX+2 TO BX : LOCATE IX-1,LX : PRINT CHR$(186) : NEXT
400 FOR IX=TX+2 TO BX : LOCATE IX-1,RX : PRINT CHR$(186) : NEXT
410 LOCATE 9,20 : PRINT "Plant parameters for 50-";DOCKET$;" are stored on"
420 LOCATE 10,20 : PRINT "the diskette. You can:"
430 LOCATE 12,22 : PRINT "(A) Enter a new value for each parameter,"
440 LOCATE 14,22 : PRINT "(B) Correct selected parameters, or"
450 LOCATE 16,22 : PRINT "(C) Return to the previous menu."
460 LOCATE 18,55 : PRINT " " : LOCATE 18,24 : INPUT " Which would you like (A/B/C)? ",X$
470 '
480 IF X$="A" OR X$="a" THEN CLS : GOTO 560 ELSE IF X$="B" OR X$="b" THEN 3060
490 IF X$="C" OR X$="c" THEN CLS : X$="PRM&DATA 340" : CHAIN "PRM&DATA" ELSE CLS : GOTO 340
500 '
510 '
5 '

```

APPENDIX E: PRM.BAS

```

520 ' *****
530 ' * INPUT *
540 ' *****
550 '
560 GOSUB 7710 : LOCATE 11,22 : PRINT "This program runs for units designed"
570 PRINT TAB(22); "by Westinghouse. Was Westinghouse the"
580 PRINT TAB(22): INPUT "nuclear steam system supplier (Y/N)? ",X$
590 '
600 IF X$="M" OR X$="n" THEN 640 ELSE IF X$="Y" OR X$="y" THEN 730
610 LOCATE 16,15 : COLOR 0,7 : PRINT " Please! Type Y or N, either upper or lower case. "; : COLOR 7,0
620 GOTO 560
630 '
640 LOCATE 16,15 : PRINT SPC(60) : LOCATE 25,20 : PRINT SPC(55)
650 GOSUB 7490 : LOCATE 16,22 : PRINT "For information on the availability of"
660 PRINT TAB(22); "programs for other units, call the"
670 PRINT TAB(22); "Engineering and Generic Communications"
680 PRINT TAB(22); "Branch in the Office of Inspection and"
690 PRINT TAB(22); "Enforcement." : GOSUB 7520 : CLS : LOCATE 13,22
700 PRINT "LOADING INTRODUCTION MODULE INTO MEMORY" : X$="INTRO 1920" : CHAIN "INTRO"
710 '
720 '
730 ' ** PLANT IDENTIFICATION **
740 '
750 HELP.FLAG$="C"
760 KEY 4,"@"+CHR$(13)
770 CLS : LOCATE 25, 6 : PRINT "Press "; : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0
780 PRINT " for help -or- Key in the name or number, and press "; : COLOR 16,7
790 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
800 COLOR 7,0 : PRINT "."
810 '
820 LOCATE 8,30 : PRINT "Plant Identification"
830 LOCATE 11,18 : PRINT "Plant Name:" : LOCATE 13,18 : PRINT "Unit Number:" : LOCATE 15,18
840 PRINT "Licensed Power (Mwt):"
850 LOCATE 17,18 : PRINT "Number of Reactor Cooling Loops (2/3/4):"
860 COLOR 0,7 : LOCATE 11,30 : PRINT SPC(32) : LOCATE 13,31 : PRINT " " : LOCATE 15,40 : PRINT " "
870 LOCATE 17,59 : PRINT " " : COLOR 7,0
880 '
890 IF X$="PLANT$" THEN HELP.FLAG$="C"
900 LOCATE 11,30 : IF HELP.FLAG$="S" THEN 920
910 INPUT "", PLANT$ : IF PLANT$="@" THEN HELP.FLAG$="S" : X$="PLANT$" : GOTO 1140
920 LOCATE 11,30 : PRINT USING "\ \";PLANT$
930 '
940 IF X$="UNIT$" THEN HELP.FLAG$="C"
950 LOCATE 13,31 : IF HELP.FLAG$="S" THEN 970
960 INPUT "", UNIT$ : IF UNIT$="@" THEN HELP.FLAG$="S" : X$="UNIT$" : GOTO 1140
970 LOCATE 13,31 : PRINT USING "\ \";UNIT$
980 '
990 KEY 4,"-9"+CHR$(13)
1000 IF X$="LIC.POWER" THEN HELP.FLAG$="C"
1010 LOCATE 15,40 : IF HELP.FLAG$="S" THEN 1030
1020 INPUT "", LIC.POWER : IF LIC.POWER=-9 THEN HELP.FLAG$="S" : X$="LIC.POWER" : GOTO 1140
1030 LOCATE 15,40 : PRINT USING "####";LIC.POWER
1040 '

```



```

1050 IF X$="LOOPSZ" THEN HELP.FLAG$="C"
1060 LOCATE 17,59 : IF HELP.FLAG$="S" THEN 1100
1070 INPUT **, LOOPSZ : LOCATE 17,51 : PRINT "2/3/4"
1080 IF LOOPSZ=-9 THEN HELP.FLAG$="S" : X$="LOOPSZ" : GOTO 1140
1090 IF LOOPSZ<>2 AND LOOPSZ<>3 AND LOOPSZ<>4 THEN COLOR 23,0 : LOCATE 17,51 : PRINT "2/3/4" : COLOR 7,0 :
    GOTO 1050
1100 LOCATE 17,58 : PRINT LOOPSZ : GOSUB 7490
1110 '
1120 Z$=!NKEY$ : IF Z$(<)CHR$(32) THEN 1120 ELSE 1170
1130 '
1140 GOSUB 7480 : GOSUB 5760 : GOSUB 7520 : GOTO 770
1150 '
1160 '
1170 '
1180 '
1190 COLOR 7,0 : KEY OFF : KEY 4,"-9"+CHR$(13) : KEY 6,"-1"+CHR$(13)
1200 CLS : LOCATE 25,2 : PRINT "Press "; : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 : PRINT " for help, ";
1210 COLOR 0,7 : PRINT " F6 "; : COLOR 7,0 : PRINT " for default value, or key in value and press ";
1220 COLOR 16,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
1230 C1$="Default" : C2$="Actual" : GOSUB 7840
1240 D.PZR.ID = 6.01203*LIC.POWER^(1/3)
1250 D.RC.PUMP.PWR = .0062152*LIC.POWER/LOOPSZ
1260 D.RC.PUMP.EFF = 90!
1270 D.REFLECT.AREA = (19.34+32.18*LOOPSZ^(1/3))*LIC.POWER^(2/3)
1280 D.LOSS.COEFFICIENT = 55!
1290 D.NONREFLECT.AREA = 32.18*LOOPSZ^(1/3)*LIC.POWER^(2/3)
1300 D.INSULATION.THICKNESS = 4!
1310 D.CONDUCTIVITY = .035
1320 LOCATE LZ+ 5,CZ-1 : PRINT USING "###.##"; D.PZR.ID
1330 LOCATE LZ+ 6,CZ : PRINT USING "##.##"; D.RC.PUMP.PWR
1340 LOCATE LZ+ 7,CZ : PRINT USING "##.##"; D.RC.PUMP.EFF
1350 LOCATE LZ+11,CZ-2 : PRINT USING "#####"; D.REFLECT.AREA
1360 LOCATE LZ+12,CZ : PRINT USING "###.##"; D.LOSS.COEFFICIENT
1370 LOCATE LZ+16,CZ-2 : PRINT USING "#####"; D.NONREFLECT.AREA
1380 LOCATE LZ+17,CZ+1 : PRINT USING "##.##"; D.INSULATION.THICKNESS
1390 LOCATE LZ+18,CZ+2 : PRINT USING "#.###"; D.CONDUCTIVITY : GOSUB 7990
1400 '
1410 '
1420 IF X$="PZR.ID" THEN HELP.FLAG$="C"
1430 LOCATE LZ+ 5,CZ+11 : IF HELP.FLAG$="S" THEN 1470
1440 INPUT **, PZR.ID
1450 IF PZR.ID=-9 THEN HELP.FLAG$="S" : X$="PZR.ID" : GOSUB 7480 : GOSUB 6340 : GOSUB 7520 : GOTO 1190
1460 IF PZR.ID=-1 THEN PZR.ID=D.PZR.ID
1470 LOCATE LZ+ 5,CZ+11 : PRINT USING "###.##"; PZR.ID
1480 '
1490 IF X$="RC.PUMP.PWR" THEN HELP.FLAG$="C"
1500 LOCATE LZ+ 6,CZ+12 : IF HELP.FLAG$="S" THEN 1540
1510 INPUT **, RC.PUMP.PWR
1520 IF RC.PUMP.PWR=-9 THEN HELP.FLAG$="S" : X$="RC.PUMP.PWR" : GOSUB 7470 : GOSUB 6450 : GOSUB 7520 : GOTO 1190
1530 IF RC.PUMP.PWR=-1 THEN RC.PUMP.PWR=D.RC.PUMP.PWR
1540 LOCATE LZ+ 6,CZ+12 : PRINT USING "##.##"; RC.PUMP.PWR
1550 '
1560 IF X$="RC.PUMP.EFF" THEN HELP.FLAG$="C"
1570 LOCATE LZ+ 7,CZ+12 : IF HELP.FLAG$="S" THEN 1610

```

APPENDIX E: PRM.BAS

```

1580 INPUT **, RC.PUMP.EFF
1590 IF RC.PUMP.EFF=-9 THEN HELP.FLAG$="S" : X$="RC.PUMP.EFF" : GOSUB 7470 : GOSUB 6450 : GOSUB 7520 : GOTO 1190
1600 IF RC.PUMP.EFF=-1 THEN RC.PUMP.EFF=D.RC.PUMP.EFF
1610 LOCATE LZ+7,CZ+12 : PRINT USING "###.#"; RC.PUMP.EFF
1620 '
1630 IF X$="REFLECT.AREA" THEN HELP.FLAG$="C"
1640 LOCATE LZ+11,CZ+9 : IF HELP.FLAG$="S" THEN 1680
1650 INPUT **, REFLECT.AREA
1660 IF REFLECT.AREA=-9 THEN HELP.FLAG$="S" : X$="REFLECT.AREA" : GOSUB 7460 : GOSUB 6610 : GOSUB 7520 : GOTO 1190
1670 IF REFLECT.AREA=-1 THEN REFLECT.AREA=D.REFLECT.AREA
1680 LOCATE LZ+11,CZ+9 : PRINT USING "#####"; REFLECT.AREA
1690 '
1700 IF X$="LOSS.COEFFICIENT" THEN HELP.FLAG$="C"
1710 LOCATE LZ+12,CZ+11 : IF HELP.FLAG$="S" THEN 1750
1720 INPUT **, LOSS.COEFFICIENT
1730 IF LOSS.COEFFICIENT=-9 THEN HELP.FLAG$="S" : X$="LOSS.COEFFICIENT" : GOSUB 7480 : GOSUB 6800 : GOSUB 7520 : GOTO 1190
1740 IF LOSS.COEFFICIENT=-1 THEN LOSS.COEFFICIENT=D.LOSS.COEFFICIENT
1750 LOCATE LZ+12,CZ+11 : PRINT USING "###.#"; LOSS.COEFFICIENT
1760 '
1770 IF X$="NONREFLECT.AREA" THEN HELP.FLAG$="C"
1780 LOCATE LZ+16,CZ+9 : IF HELP.FLAG$="S" THEN 1820
1790 INPUT **, NONREFLECT.AREA
1800 IF NONREFLECT.AREA=-9 THEN HELP.FLAG$="S" : X$="NONREFLECT.AREA" : GOSUB 7460 : GOSUB 6860 : GOSUB 7520 : GOTO 1190
1810 IF NONREFLECT.AREA=-1 THEN NONREFLECT.AREA=D.NONREFLECT.AREA
1820 LOCATE LZ+16,CZ+9 : PRINT USING "#####"; NONREFLECT.AREA
1830 '
1840 IF X$="INSULATION.THICKNESS" THEN HELP.FLAG$="C"
1850 LOCATE LZ+17,CZ+12 : IF HELP.FLAG$="S" THEN GOTO 1890
1860 INPUT **, INSULATION.THICKNESS
1870 IF INSULATION.THICKNESS=-9 THEN HELP.FLAG$="S" : X$="INSULATION.THICKNESS" : GOSUB 7480 : GOSUB 7050 : GOSUB 7520 : GOTO 1190
1880 IF INSULATION.THICKNESS=-1 THEN INSULATION.THICKNESS=D.INSULATION.THICKNESS
1890 LOCATE LZ+17,CZ+12 : PRINT USING "###.#"; INSULATION.THICKNESS
1900 '
1910 IF X$="CONDUCTIVITY" THEN HELP.FLAG$="C"
1920 LOCATE LZ+18,CZ+13 : IF HELP.FLAG$="S" THEN GOTO 1960
1930 LOCATE LZ+18,CZ+13 : INPUT **, CONDUCTIVITY
1940 IF CONDUCTIVITY=-9 THEN HELP.FLAG$="S" : X$="CONDUCTIVITY" : GOSUB 7480 : GOSUB 7100 : GOSUB 7520 : GOTO 1190
1950 IF CONDUCTIVITY=-1 THEN CONDUCTIVITY=D.CONDUCTIVITY
1960 LOCATE LZ+18,CZ+12 : PRINT USING "###.###"; CONDUCTIVITY : LOCATE 25,1 : PRINT SPC(79); : GOSUB 7490
1970 GOTO 2020
1980 '
1990 GOSUB 7480 : GOSUB 5910 : GOSUB 7520 : GOTO 1190
2000 '
2010 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 2010 ELSE 1190
2020 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 2020 ELSE 2050
2030 '
2040 '
2050 '
2060 '
2070 COLOR 7,0 : KEY OFF : KEY 4,"-9"+CHR$(13) : KEY 6,"-1"+CHR$(13)
2080 CLS : LOCATE 25,2 : PRINT "Press " : COLOR 0,7 : PRINT " F4 " : COLOR 7,0 : PRINT " for help, ";

```

```

2090 COLOR 0,7 : PRINT " F6 "; : COLOR 7,0 : PRINT " for default value, or key in value and press ";
2100 COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
2110 COLOR 7,0 : PRINT "."
2120 GOSUB 8080
2130 LOCATE LZ+12,BZ+2 : PRINT "Steam Generator B " : IF LOOPSZ=3 OR LOOPSZ=4 THEN 2140 ELSE 2160
2140 LOCATE LZ+13,BZ+2 : PRINT "Steam Generator C " : IF LOOPSZ=4 THEN 2150 ELSE 2160
2150 LOCATE LZ+14,BZ+2 : PRINT "Steam Generator D "
2160 D.SG.ID      = 13.7089*(LIC.POWER/LOOPSZ)^(1/3)
2170 D.RISER.OB  = 2.0036*(LIC.POWER/LOOPSZ)^(1/3)
2180 D.CARRYOVER.A = .125 : D.CARRYOVER.C = .125 : D.CARRYOVER.B = .125 : D.CARRYOVER.D = .125
2190 D.RISER.NUMBER = 12! : CZ=59
2200 LOCATE LZ+ 4,CZ-1 : PRINT USING "###.#"; D.SG.ID
2210 LOCATE LZ+ 7,CZ  : PRINT USING "##.#"; D.RISER.OB
2220 LOCATE LZ+ 6,CZ  : PRINT USING "##"; D.RISER.NUMBER
2230 LOCATE LZ+11,CZ+1 : PRINT USING "#.###"; D.CARRYOVER.A
2240 LOCATE LZ+12,CZ+1 : PRINT USING "#.###"; D.CARRYOVER.B : IF LOOPSZ=3 OR LOOPSZ=4 THEN 2250 ELSE 2270
2250 LOCATE LZ+13,CZ+1 : PRINT USING "#.###"; D.CARRYOVER.C : IF LOOPSZ=4 THEN 2260 ELSE 2270
2260 LOCATE LZ+14,CZ+1 : PRINT USING "#.###"; D.CARRYOVER.D
2270 LOCATE LZ+ 4,CZ+10 : COLOR 0,7 : PRINT " . "
2280 LOCATE LZ+ 6,CZ+11 : PRINT " . "
2290 LOCATE LZ+ 7,CZ+11 : PRINT " . "
2300 LOCATE LZ+11,CZ+12 : PRINT " . "
2310 LOCATE LZ+12,CZ+12 : PRINT " . " : IF LOOPSZ=3 OR LOOPSZ=4 THEN 2320 ELSE COLOR 7,0 : GOTO 2340
2320 LOCATE LZ+13,CZ+12 : PRINT " . " : IF LOOPSZ=4 THEN 2330 ELSE COLOR 7,0 : GOTO 2340
2330 LOCATE LZ+14,CZ+12 : PRINT " . " : COLOR 7,0
2340 '
2350 IF X$="SG.ID" THEN HELP.FLAG$="C"
2360 LOCATE LZ+ 4,CZ+10 : IF HELP.FLAG$="S" THEN 2400
2370 INPUT "*", SG.ID
2380 IF SG.ID=-9 THEN HELP.FLAG$="S" : X$="SG.ID" : GOSUB 7480 : GOSUB 7160 : GOSUB 2860 : GOTO 2070
2390 IF SG.ID=-1 THEN SG.ID=D.SG.ID
2400 LOCATE LZ+ 4,CZ+10 : PRINT USING "###.#"; SG.ID
2410 '
2420 IF X$="RISER.NUMBER" THEN HELP.FLAG$="C"
2430 LOCATE LZ+ 6,CZ+11 : IF HELP.FLAG$="S" THEN 2470
2440 INPUT "*",RISER.NUMBER
2450 IF RISER.NUMBER=-9 THEN HELP.FLAG$="S" : X$="RISER.NUMBER" : GOSUB 7480 : GOSUB 7260 : GOSUB 2860 : GOTO 2070
2460 IF RISER.NUMBER=-1 THEN RISER.NUMBER=D.RISER.NUMBER
2470 LOCATE LZ+ 6,CZ+11 : PRINT USING "##"; RISER.NUMBER
2480 '
2490 IF X$="RISER.OB" THEN HELP.FLAG$="C"
2500 LOCATE LZ+ 7,CZ+11 : IF HELP.FLAG$="S" THEN 2540
2510 INPUT "*", RISER.OB
2520 IF RISER.OB=-9 THEN HELP.FLAG$="S" : X$="RISER.OB" : GOSUB 7480 : GOSUB 7260 : GOSUB 2860 : GOTO 2070
2530 IF RISER.OB=-1 THEN RISER.OB=D.RISER.OB
2540 LOCATE LZ+ 7,CZ+10 : PRINT USING "###.#"; RISER.OB
2550 '
2560 IF X$="CARRYOVER.A" THEN HELP.FLAG$="C"
2570 LOCATE LZ+11,CZ+12 : IF HELP.FLAG$="S" THEN 2610
2580 INPUT "*", CARRYOVER.A
2590 IF CARRYOVER.A=-9 THEN HELP.FLAG$="S" : X$="CARRYOVER.A" : GOSUB 7480 : GOSUB 7360 : GOSUB 2860 : GOTO 2070
2600 IF CARRYOVER.A=-1 THEN CARRYOVER.A=D.CARRYOVER.A
2610 LOCATE LZ+11,CZ+12 : PRINT USING "#.###"; CARRYOVER.A
2620 '

```

APPENDIX E: PRM.BAS

```

2630 IF X$="CARRYOVER.B" THEN HELP.FLAG$="C"
2640 LOCATE LZ+12,CZ+12 : IF HELP.FLAG$="S" THEN 2680
2650 INPUT **, CARRYOVER.B
2660 IF CARRYOVER.B=-9 THEN HELP.FLAG$="S" : X$="CARRYOVER.B" : GOSUB 7480 : GOSUB 7360 : GOSUB 2860 : GOTO 2070
2670 IF CARRYOVER.B=-1 THEN CARRYOVER.B=D.CARRYOVER.B
2680 LOCATE LZ+12,CZ+12 : PRINT USING "#.###"; CARRYOVER.B : IF LOOPSZ=2 THEN 2840
2690 '
2700 IF X$="CARRYOVER.C" THEN HELP.FLAG$="C"
2710 LOCATE LZ+13,CZ+12 : IF HELP.FLAG$="S" THEN 2750
2720 INPUT **, CARRYOVER.C
2730 IF CARRYOVER.C=-9 THEN HELP.FLAG$="S" : X$="CARRYOVER.C" : GOSUB 7480 : GOSUB 7360 : GOSUB 2860 : GOTO 2070
2740 IF CARRYOVER.C=-1 THEN CARRYOVER.C=D.CARRYOVER.C
2750 LOCATE LZ+13,CZ+12 : PRINT USING "#.###"; CARRYOVER.C : IF LOOPSZ=3 THEN 2840
2760 '
2770 IF X$="CARRYOVER.D" THEN HELP.FLAG$="C"
2780 LOCATE LZ+14,CZ+12 : IF HELP.FLAG$="S" THEN 2820
2790 INPUT **, CARRYOVER.D
2800 IF CARRYOVER.D=-9 THEN HELP.FLAG$="S" : X$="CARRYOVER.D" : GOSUB 7480 : GOSUB 7360 : GOSUB 2860 : GOTO 2070
2810 IF CARRYOVER.D=-1 THEN CARRYOVER.D=D.CARRYOVER.D
2820 LOCATE LZ+14,CZ+12 : PRINT USING "#.###"; CARRYOVER.D : IF LOOPSZ=4 THEN 2840
2830 '
2840 LOCATE 25, 1 : PRINT SPC(79) : GOSUB 7490 : GOTO 2880
2850 '
2860 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 2860 ELSE RETURN
2870 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 2870 ELSE 2070
2880 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 2880 ELSE CLS : GOTO 2890
2890 '
2900 '
2910 '
2920 '
2930 GOSUB 7710 : LOCATE 11,26 : PRINT "The docket number is 50-";DOCKET$;","
2940 LOCATE 12,29 : INPUT "Is it correct (Y/N)? ",X$
2950 '
2960 IF X$="N" OR X$="n" THEN LOCATE 14,29 : PRINT SPC(23) : LOCATE 16,59 : PRINT SPC(5) : OLD.DOCKET$=DOCKET$
: GOTO 2980
2970 IF X$="Y" OR X$="y" THEN LOCATE 14,29 : PRINT SPC(23) : GOSUB 3000 : GOTO 5680 ELSE 3020
2980 LOCATE 16,18 : INPUT "Type last three digits of docket number: ",DOCKET$ :LOCATE 12,50 : PRINT " "
2990 GOTO 2930
3000 IF OLD.DOCKET$(<)DOCKET$ AND OLD.DOCKET$(<)" THEN KILL OLD.DOCKET$ ELSE RETURN
3010 KILL "PLANT-ID."+OLD.DOCKET$ : KILL "RCS-CALC."+OLD.DOCKET$ : RETURN
3020 LOCATE 14,29 : COLOR 0,7 : PRINT " PLEASE! TYPE Y OR N. " : COLOR 7,0 : GOTO 2930
3030 '
3040 '
3050 '
3060 '
3070 '
3080 '
3090 '
3100 '
3110 L$="13"
3120 '
3130 GOSUB 7540
3140 '

```

```

3150 LOCATE 9,30 : PRINT "PLANT IDENTIFICATION"
3160 LOCATE 11,38 : PRINT "Current or Corrected"
3170 LOCATE 13,13 : PRINT "Plant Name:";TAB(35); : COLOR 0,7 : PRINT SPC(32); : LOCATE 13,35 : PRINT PLANT$
3180 COLOR 7,0
3190 LOCATE 15,13 : PRINT "Unit Number:";TAB(35); : COLOR 0,7 : PRINT SPC( 1); : LOCATE 15,35 : PRINT UNIT$
3200 COLOR 7,0
3210 LOCATE 17,13 : PRINT "Licensed Power (Mwt):";TAB(35); : COLOR 0,7 : PRINT SPC(4); : LOCATE 17,35
3220 PRINT USING "####";LIC.POWER : COLOR 7,0
3230 '
3240 IF L$="13" THEN 3260 ELSE IF L$="15" THEN 3320 ELSE 3380
3250 '
3260 LOCATE 13,35,1 : L$="13"
3270 Z$=INKEY$ : IF Z$="" THEN 3270 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3320
3280 IF Z$=CHR$(0)+CHR$(66) THEN 3500 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 3470
3290 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3380 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3300 ELSE 3260
3300 GOSUB 7730 : LOCATE 13,35 : COLOR 0,7 : INPUT "",PLANT$ : COLOR 7,0 : GOTO 3130
3310 '
3320 LOCATE 15,35,1 : L$="15"
3330 Z$=INKEY$ : IF Z$="" THEN 3330 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3380
3340 IF Z$=CHR$(0)+CHR$(66) THEN 3500 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 3470
3350 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3260 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3360 ELSE 3320
3360 GOSUB 7730 : LOCATE 15,35 : COLOR 0,7 : INPUT "",UNIT$ : COLOR 7,0 : GOTO 3130
3370 '
3380 LOCATE 17,35,1 : L$="17"
3390 Z$=INKEY$ : IF Z$="" THEN 3390 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3260
3400 IF Z$=CHR$(0)+CHR$(66) THEN 3500 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 3470
3410 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3320 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3420 ELSE 3380
3420 GOSUB 7730 : LOCATE 17,35 : COLOR 0,7 : INPUT "",LIC.POWER : COLOR 7,0 : GOTO 3130
3430 '
3440 GOSUB 7480
3450 Z$=INKEY$ : IF Z$="" THEN 3240 ELSE 3490
3460 '
3470 GOSUB 7610 : GOTO 6110
3480 '
3490 '
3500 '
3510 '
3520 L$="4" : H$="RCS"
3530 '
3540 GOSUB 7540 : C1$="Current" : C2$=" New" : GOSUB 8170
3550 LOCATE LZ+ 4,CZ+1 : PRINT USING "##"; : LOOPSZ
3560 LOCATE LZ+ 5,CZ-1 : PRINT USING "###.##"; : PZR.ID
3570 LOCATE LZ+ 6,CZ : PRINT USING "##.##"; : RC.PUMP.PWR
3580 LOCATE LZ+ 7,CZ : PRINT USING "##.##"; : RC.PUMP.EFF
3590 LOCATE LZ+11,CZ-2 : PRINT USING "#####"; : REFLECT.AREA
3600 LOCATE LZ+12,CZ : PRINT USING "###.##"; : LOSS.COEFFICIENT
3610 LOCATE LZ+16,CZ-2 : PRINT USING "#####"; : NONREFLECT.AREA
3620 LOCATE LZ+17,CZ+1 : PRINT USING "##.##"; : INSULATION.THICKNESS
3630 LOCATE LZ+18,CZ+2 : PRINT USING "###.##"; : CONDUCTIVITY : GOSUB 7990
3640 '
3650 IF L$="4" THEN 3700 ELSE IF L$="5" THEN 3790 ELSE IF L$="6" THEN 3850 ELSE IF L$="7" THEN 3940
3660 IF L$="11" THEN 4030 ELSE IF L$="12" THEN 4120 ELSE IF L$="16" THEN 4200 ELSE IF L$="17" THEN 4280
3670 IF L$="18" THEN 4360 ELSE PRINT "ERROR - PRM&DATA - 4392"
3680 '

```

APPENDIX E: PRM.BAS

```

3690
3700 LOCATE LZ+ 4,CZ+13,1 : L$="4"
3710 Z$=INKEY$: IF Z$="" THEN 3710 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3780
3720 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
3730 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 4360 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3740 ELSE 3710
3740 GOSUB 7760 : LOCATE LZ+ 4,CZ+13 : COLOR 0,7 : INPUT **, LOOPSZ : LOCATE LZ+4,BZ+33 : COLOR 7,0
3750 IF LOOPSZ<>2 AND LOOPSZ<>3 AND LOOPSZ<>4 THEN COLOR 23,0 : LOCATE LZ+4,BZ+33 : PRINT "2/3/4" : COLOR 7,0 :
GOTO 3740
3760 GOTO 3540
3770
3780 LOCATE LZ+ 5,CZ+11,1 : L$="5" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 3820
3790 Z$=INKEY$: IF Z$="" THEN 3790 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3850
3800 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
3810 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3700 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3820 ELSE 3790
3820 GOSUB 7660 : LOCATE LZ+ 5,CZ+11 : PZR.ID.R=PZR.ID : COLOR 0,7 : INPUT **, PZR.ID : COLOR 7,0
3830 IF PZR.ID=-9 THEN HELP.FLAG$="S" : PZR.ID=PZR.ID.R : GOSUB 7610 : GOSUB 6340 : GOTO 7650 ELSE GOTO 3540
3840
3850 LOCATE LZ+ 6,CZ+12,1 : L$="6" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 3890
3860 Z$=INKEY$: IF Z$="" THEN 3860 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3940
3870 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
3880 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3780 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3890 ELSE 3860
3890 GOSUB 7660 : LOCATE LZ+ 6,CZ+12 : RC.PUMP.PWR.R=RC.PUMP.PWR : COLOR 0,7
3900 INPUT **, RC.PUMP.PWR : COLOR 7,0
3910 IF RC.PUMP.PWR=-9 THEN HELP.FLAG$="S" : RC.PUMP.PWR=RC.PUMP.PWR.R : GOSUB 7610 : GOSUB 6450 : GOSUB 7650
3920 GOTO 3540
3930
3940 LOCATE LZ+ 7,CZ+12,1 : L$="7" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 3980
3950 Z$=INKEY$: IF Z$="" THEN 3950 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 4030
3960 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
3970 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3850 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 3980 ELSE 3950
3980 GOSUB 7660 : LOCATE LZ+ 7,CZ+12 : RC.PUMP.EFF.R=RC.PUMP.EFF : COLOR 0,7
3990 INPUT **, RC.PUMP.EFF : COLOR 7,0
4000 IF RC.PUMP.EFF=-9 THEN HELP.FLAG$="S" : RC.PUMP.EFF=RC.PUMP.EFF.R : GOSUB 7610 : GOSUB 6450 : GOSUB 7650
4010 GOTO 3540
4020
4030 LOCATE LZ+11,CZ+ 9,1 : L$="11" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 4070
4040 Z$=INKEY$: IF Z$="" THEN 4040 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 4120
4050 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
4060 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 3940 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 4070 ELSE 4040
4070 GOSUB 7660 : LOCATE LZ+11,CZ+9 : REFLECT.AREA.R=REFLECT.AREA : COLOR 0,7
4080 INPUT **, REFLECT.AREA : COLOR 7,0
4090 IF REFLECT.AREA=-9 THEN HELP.FLAG$="S" : REFLECT.AREA=REFLECT.AREA.R : GOSUB 7610 : GOSUB 6610 : GOSUB 7650
4100 GOTO 3540
4110
4120 LOCATE LZ+12,CZ+11,1 : L$="12" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 4160
4130 Z$=INKEY$: IF Z$="" THEN 4130 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 4200
4140 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
4150 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 4030 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 4160 ELSE 4130
4160 GOSUB 7660 : LOCATE LZ+12,CZ+11 : LOSS.COEFFICIENT.R=LOSS.COEFFICIENT : COLOR 0,7
4170 INPUT **, LOSS.COEFFICIENT : COLOR 7,0
4180 IF LOSS.COEFFICIENT=-9 THEN HELP.FLAG$="S" : LOSS.COEFFICIENT=LOSS.COEFFICIENT.R : GOSUB 7610 : GOSUB 6800
: GOSUB 7650 ELSE GOTO 3540
4190

```

```

4200 LOCATE LZ+16,CZ+ 9,1 : L$="16" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 4240
4210 Z$=INKEY$ : IF Z$="" THEN 4210 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 4280
4220 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
4230 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 4120 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 4240 ELSE 4210
4240 GOSUB 7660 : LOCATE LZ+16,CZ+9 : NONREFLECT.AREA.R=NONREFLECT.AREA : COLOR 0,7
4250 INPUT **, NONREFLECT.AREA : COLOR 7,0
4260 IF NONREFLECT.AREA=-9 THEN HELP.FLAG$="S" : NONREFLECT.AREA=NONREFLECT.AREA.R : GOSUB 7610 : GOSUB 6610 :
GOSUB 7650 ELSE GOTO 3540
4270 '
4280 LOCATE LZ+17,CZ+12,1 : L$="17" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 4320
4290 Z$=INKEY$ : IF Z$="" THEN 4290 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 4360
4300 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
4310 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 4200 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 4320 ELSE 4290
4320 GOSUB 7660 : LOCATE LZ+17,CZ+12 : INSULATION.THICKNESS.R=INSULATION.THICKNESS : COLOR 0,7
4330 INPUT **, INSULATION.THICKNESS : COLOR 7,0
4340 IF INSULATION.THICKNESS=-9 THEN HELP.FLAG$="S" : INSULATION.THICKNESS=INSULATION.THICKNESS.R : GOSUB 7610
: GOSUB 7050 : GOSUB 7650 ELSE GOTO 3540
4350 '
4360 LOCATE LZ+18,CZ+13,1 : L$="18" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 4400
4370 Z$=INKEY$ : IF Z$="" THEN 4370 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 3700
4380 IF Z$=CHR$(0)+CHR$(66) THEN 4450 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
4390 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 4280 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 4400 ELSE 4370
4400 GOSUB 7660 : LOCATE LZ+18,CZ+13 : CONDUCTIVITY.R=CONDUCTIVITY : COLOR 0,7
4410 INPUT **, CONDUCTIVITY : COLOR 7,0
4420 IF CONDUCTIVITY=-9 THEN HELP.FLAG$="S" : CONDUCTIVITY=CONDUCTIVITY.R : GOSUB 7610 : GOSUB 7100 : GOSUB 7650
4430 GOTO 3540
4440 '
4450 GOTO 4730
4460 '
4470 GOSUB 7610
4480 LOCATE 6, 7 : PRINT "1. The current value of each parameter is displayed.
4490 LOCATE 8, 7 : PRINT "2. To enter a new value for a particular parameter, first move the
4500 LOCATE 9,11 : PRINT "cursor to the appropriate line by pressing either the "; : COLOR 0,7
4510 PRINT " ";CHR$(24);" " : COLOR 7,0
4520 PRINT " or "; : COLOR 0,7 : PRINT " ";CHR$(25);" " : COLOR 7,0
4530 LOCATE 10,11 : PRINT "key."
4540 LOCATE 12, 7 : PRINT"3. Next press "; : COLOR 0,7 : PRINT " F6 "; : COLOR 7,0
4550 PRINT ". The bottom line on the display will change to"
4560 LOCATE 13,11 : PRINT "provide additional guidance."
4570 LOCATE 15, 7 : PRINT "4. To correct a value, use the "; : COLOR 0,7 : PRINT " ";CHR$(196)+CHR$( 16);" ";
4580 COLOR 7,0 : PRINT " and "; : COLOR 0,7 : PRINT " ";CHR$( 17)+CHR$(196);" "; : COLOR 7,0
4590 PRINT " keys to move the cursor"
4600 LOCATE 16,11 : PRINT "to the appropriate column in the highlighted field. Key in the"
4610 LOCATE 17,11 : PRINT "new value. If a wrong digit is keyed in, write over it. Do not"
4620 LOCATE 18,11 : PRINT "not press the "; : COLOR 0,7 : PRINT " ";CHR$( 17)+CHR$(196)+CHR$(217);" ";
4630 COLOR 7,0 : PRINT " key until the last digit is keyed in.
4640 LOCATE 20, 7 : PRINT "5. If the "; : COLOR 0,7 : PRINT " ";CHR$(196)+CHR$( 16);" "; : COLOR 7,0
4650 PRINT " and "; : COLOR 0,7 : PRINT " ";CHR$( 17)+CHR$(196);" "; : COLOR 7,0 : PRINT " keys";
4660 PRINT " do not work, press the "; : COLOR 0,7 : PRINT " Num Lock "; : COLOR 7,0 : PRINT " key,"
4670 LOCATE 21,11 : PRINT "and try again."
4680 GOTO 7650
4690 '
4700 IF H$="RCS" THEN GOTO 3540 ELSE IF H$="SG" THEN GOTO 4770
4710 '

```

APPENDIX E: PRM.BAS

```

4720 '
4730 '                ** STEAM GENERATORS **
4740 '
4750 L$="4" : H$="SG"
4760 '
4770 GOSUB 7540
4780 '
4790 GOSUB 8080
4800 LOCATE LZ+12,BZ+2 : PRINT "Steam Generator B " : IF LOOPSZ=3 OR LOOPSZ=4 THEN 4810 ELSE 4840
4810 LOCATE LZ+13,BZ+2 : PRINT "Steam Generator C " : IF LOOPSZ=4 THEN 4820 ELSE 4840
4820 LOCATE LZ+14,BZ+2 : PRINT "Steam Generator D "
4830 CZ=59
4840 LOCATE LZ+ 4,CZ-1 : PRINT USING "###.#"; SG.ID
4850 LOCATE LZ+ 7,CZ : PRINT USING "##.#"; RISER.DJ
4860 LOCATE LZ+ 6,CZ : PRINT USING "##"; RISER.NUMBER
4870 LOCATE LZ+11,CZ+1 : PRINT USING "#.###"; CARRYOVER.A
4880 LOCATE LZ+12,CZ+1 : PRINT USING "#.###"; CARRYOVER.B : IF LOOPSZ=3 OR LOOPSZ=4 THEN 4890 ELSE 4910
4890 LOCATE LZ+13,CZ+1 : PRINT USING "#.###"; CARRYOVER.C : IF LOOPSZ=4 THEN 4900 ELSE 4910
4900 LOCATE LZ+14,CZ+1 : PRINT USING "#.###"; CARRYOVER.D
4910 LOCATE LZ+ 4,CZ+10 : COLOR 0,7 : PRINT " . "
4920 LOCATE LZ+ 6,CZ+11 : PRINT " "
4930 LOCATE LZ+ 7,CZ+11 : PRINT " . "
4940 LOCATE LZ+11,CZ+12 : PRINT " . "
4950 LOCATE LZ+12,CZ+12 : PRINT " . " : IF LOOPSZ=3 OR LOOPSZ=4 THEN 4960 ELSE COLOR 7,0 : GOTO 4980
4960 LOCATE LZ+13,CZ+12 : PRINT " . " : IF LOOPSZ=4 THEN 4970 ELSE COLOR 7,0 : GOTO 4980
4970 LOCATE LZ+14,CZ+12 : PRINT " . " : COLOR 7,0
4980 '
4990 IF L$="4" THEN 5030 ELSE IF L$="6" THEN 5130 ELSE IF L$="7" THEN 5220 ELSE IF L$="11" THEN 5300
5000 IF L$="12" THEN 5390 ELSE IF L$="13" THEN 5490
5010 IF L$="14" THEN 5590 ELSE PRINT "ERROR - PRM.BAS - 6232"
5020 '
5030 LOCATE LZ+ 4,CZ+10,1 : L$="4" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5100
5040 Z$=INKEY$ : IF Z$="" THEN 5040 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5130
5050 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5060 IF (Z$=CHR$(0)+CHR$(72) OR Z$="8") AND LOOPSZ=2 THEN 5390
5070 IF (Z$=CHR$(0)+CHR$(72) OR Z$="8") AND LOOPSZ=3 THEN 5490
5080 IF (Z$=CHR$(0)+CHR$(72) OR Z$="8") THEN 5590
5090 IF Z$=CHR$(0)+CHR$(64) THEN 5100 ELSE 5040
5100 GOSUB 7660 : LOCATE LZ+ 4,CZ+10 : SG.ID.R=SG.ID : COLOR 0,7 : INPUT "", SG.ID : COLOR 7,0
5110 IF SG.ID=-9 THEN HELP.FLAG$="S" : SG.ID=SG.ID.R : GOSUB 7610 : GOSUB 7160 : GOTO 7650 ELSE GOTO 4770
5120 '
5130 LOCATE LZ+ 6,CZ+11,1 : L$="6" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5170
5140 Z$=INKEY$ : IF Z$="" THEN 5140 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5220
5150 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5160 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5030 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5170 ELSE 5140
5170 GOSUB 7660 : LOCATE LZ+ 6,CZ+11 : RISER.NUMBER.R=RISER.NUMBER : COLOR 0,7 : INPUT "", RISER.NUMBER
5180 COLOR 7,0
5190 IF RISER.NUMBER=-9 THEN HELP.FLAG$="S" : RISER.NUMBER=RISER.NUMBER.R : GOSUB 7610 : GOSUB 7260 : GOTO 7650
5200 GOTO 4770
5210 '
5220 LOCATE LZ+ 7,CZ+11,1 : L$="7" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5260
5230 Z$=INKEY$ : IF Z$="" THEN 5230 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5300
5240 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5250 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5130 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5260 ELSE 5230

```



```

5260 GOSUB 7660 : LOCATE LZ+7,CZ+11 : RISER.OD.R=RISER.OD : COLOR 0,7 : INPUT **, RISER.OD : COLOR 7,0
5270 IF RISER.OD=-9 THEN HELP.FLAG$="S" : RISER.OD=RISER.OD.R : GOSUB 7610 : GOSUB 7260 : GOTO 7650
5280 GOTO 4770
5290 '
5300 LOCATE LZ+11,CZ+12,1 : L$="11" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5340
5310 Z$=INKEY$ : IF Z$="" THEN 5310 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5390
5320 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5330 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5220 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5340 ELSE 5310
5340 GOSUB 7660 : LOCATE LZ+11,CZ+12 : CARRYOVER.A.R=CARRYOVER.A : COLOR 0,7 : INPUT **, CARRYOVER.A
5350 COLOR 7,0
5360 IF CARRYOVER.A=-9 THEN HELP.FLAG$="S" : CARRYOVER.A=CARRYOVER.A.R : GOSUB 7610 : GOSUB 7360 : GOTO 7650
5370 GOTO 4770
5380 '
5390 LOCATE LZ+12,CZ+12,1 : L$="12" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5440
5400 Z$=INKEY$ : IF Z$="" THEN 5400 ELSE IF (Z$=CHR$(0)+CHR$(80) OR Z$="2") AND (LOOP$=3 OR LOOP$=4) THEN 5490
5410 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5030
5420 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5430 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5300 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5440 ELSE 5400
5440 GOSUB 7660 : LOCATE LZ+12,CZ+12 : CARRYOVER.B.R=CARRYOVER.B : COLOR 0,7 : INPUT **, CARRYOVER.B
5450 COLOR 7,0
5460 IF CARRYOVER.B=-9 THEN HELP.FLAG$="S" : CARRYOVER.B=CARRYOVER.B.R : GOSUB 7610 : GOSUB 7360 : GOTO 7650
5470 GOTO 4770
5480 '
5490 LOCATE LZ+13,CZ+12 : L$="13" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5540
5500 Z$=INKEY$ : IF Z$="" THEN 5500 ELSE IF (Z$=CHR$(0)+CHR$(80) OR Z$="2") AND LOOP$=4 THEN 5590
5510 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5030
5520 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5530 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5390 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5540 ELSE 5500
5540 GOSUB 7660 : LOCATE LZ+13,CZ+12 : CARRYOVER.C.R=CARRYOVER.C : COLOR 0,7 : INPUT **, CARRYOVER.C
5550 COLOR 7,0
5560 IF CARRYOVER.C=-9 THEN HELP.FLAG$="S" : CARRYOVER.C=CARRYOVER.C.R : GOSUB 7610 : GOSUB 7360 : GOTO 7650
5570 GOTO 4770
5580 '
5590 LOCATE LZ+14,CZ+12,1 : L$="14" : IF HELP.FLAG$="S" THEN HELP.FLAG$="C" : GOTO 5630
5600 Z$=INKEY$ : IF Z$="" THEN 5600 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN 5030
5610 IF Z$=CHR$(0)+CHR$(66) THEN 5680 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4470
5620 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN 5490 ELSE IF Z$=CHR$(0)+CHR$(64) THEN 5630 ELSE 5600
5630 GOSUB 7660 : LOCATE LZ+14,CZ+12 : CARRYOVER.D.R=CARRYOVER.D : COLOR 0,7 : INPUT **, CARRYOVER.D
5640 COLOR 7,0
5650 IF CARRYOVER.D=-9 THEN HELP.FLAG$="S" : CARRYOVER.D=CARRYOVER.D.R : GOSUB 7610 : GOSUB 7360 : GOTO 7650
5660 GOTO 4770
5670 '
5680 CLS : LOCATE 12,18,0 : PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY"
5690 X$="PRM&DATA 740" : CHAIN "PRM&DATA"
5700 '
5710 '
5720 '
5730 ' *****
5740 ' * HELP MESSAGES *
5750 ' *****
5760 LOCATE 4,30 : PRINT "Plant Identification"
5770 LOCATE 9,12 : PRINT "1. Write only in the highlighted field in which the"
5780 LOCATE 10,16 : PRINT "cursor is located."
5790 LOCATE 12,12 : PRINT "2. For proper spacing of plant name and unit number in"

```

APPENDIX E: PRM.BAS

```

5800 LOCATE 13,16 : PRINT "screen displays and printed reports, start writing"
5810 LOCATE 14,16 : PRINT "the plant name at the beginning of its highlighted"
5820 LOCATE 15,16 : PRINT "field."
5830 LOCATE 17,12 : PRINT "3. To write over an incorrect character, use the "; : COLOR 0,7
5840 PRINT " ";CHR$(17)+CHR$(196);" "; : COLOR 7,0
5850 LOCATE 18,16 : PRINT "and "; : COLOR 0,7 : PRINT " ";CHR$(196)+CHR$(16);" "; : COLOR 7,0
5860 PRINT " keys to position the cursor. When the"
5870 LOCATE 19,16 : PRINT "entry is correct, press the "; : COLOR 0,7
5880 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT " key. If the"
5890 LOCATE 20,16 : PRINT "arrow keys are not working, then press "; : COLOR 0,7 : PRINT " Num Lock ";
5900 COLOR 7,0 : PRINT "." : RETURN
5910 LOCATE 4,31 : PRINT "Entry of Parameters"
5920 LOCATE 6, 7 : PRINT "1. Values must be entered in order from top to bottom."
5930 LOCATE 8, 7 : PRINT "2. The number pad should be locked out, so that the "; : COLOR 0,7
5940 PRINT " ";CHR$(196)+CHR$(16);" "; : COLOR 7,0 : PRINT " and ";
5950 COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196);" "; : COLOR 7,0
5960 LOCATE 9,11 : PRINT "keys (the number pad ";CHR$(34);"4";CHR$(34);" and ";CHR$(34);"6";CHR$(34);
5970 PRINT " keys) will move the cursor as"
5980 LOCATE 10,11 : PRINT "as desired. If the arrow keys are not working, press "; : COLOR 0,7
5990 PRINT " Num Lock "; : COLOR 7,0 : PRINT "."
6000 LOCATE 12, 7 : PRINT "3. To enter an actual value, move the cursor to the appropriate"
6010 LOCATE 13,11 : PRINT "column relative to the decimal point and then key in the digits."
6020 LOCATE 14,11 : PRINT "If a wrong digit is keyed in, write over it. The number of"
6030 LOCATE 15,11 : PRINT "digits entered should be limited to the highlighted field. Do"
6040 LOCATE 16,11 : PRINT "not press the "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
6050 COLOR 7,0 : PRINT " key until after the last digit is keyed in."
6060 LOCATE 18, 7 : PRINT "4. If an incorrect value is keyed in and entered, do not try to"
6070 LOCATE 19,11 : PRINT "correct it now. Complete the entry of parameter values and"
6080 LOCATE 20,11 : PRINT "return to the DATA & PARAMETERS MENU. Then select the Corrections"
6090 LOCATE 21,11 : PRINT "option." : GOSUB 7520 : GOTO 1190
6100
6110 LOCATE 6, 7 : PRINT "1. The current plant name, unit number, and licensed power are displayed.
6120 LOCATE 8, 7 : PRINT "2. To make corrections, first move the cursor to the appropriate line"
6130 LOCATE 9,11 : PRINT "by pressing either the "; : COLOR 0,7 : PRINT " ";CHR$(24);" "; : COLOR 7,0
6140 PRINT " or "; : COLOR 0,7 : PRINT " ";CHR$(25);" "; : COLOR 7,0 : PRINT " key."
6150 LOCATE 11, 7 : PRINT "3. Next press "; : COLOR 0,7 : PRINT " F6 "; : COLOR 7,0
6160 PRINT ". The bottom line on the display will change to"
6170 LOCATE 12,11 : PRINT "provide additional guidance."
6180 LOCATE 14, 7 : PRINT "4. To correct the plant name, use the "; : COLOR 0,7
6190 PRINT " ";CHR$(196)+CHR$(16);" "; : COLOR 7,0 : PRINT " and "; : COLOR 0,7
6200 PRINT " ";CHR$(17)+CHR$(196);" "; : COLOR 7,0 : PRINT " keys to move the"
6210 LOCATE 15,11 : PRINT "cursor to the appropriate position in the highlighted field. Write"
6220 LOCATE 16,11 : PRINT "over the incorrect characters, and then move the cursor one space"
6230 LOCATE 17,11 : PRINT "beyond the last character. Press "; : COLOR 0,7
6240 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT "."
6250 LOCATE 19, 7 : PRINT "5. To correct the unit number or licensed power, write over the "
6260 LOCATE 20,11 : PRINT "incorrect number and press "; : COLOR 0,7
6270 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT "."
6280 LOCATE 22, 7 : PRINT "6. If the "; : COLOR 0,7 : PRINT " ";CHR$(196)+CHR$(16);" "; : COLOR 7,0
6290 PRINT " and "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196);" "; : COLOR 7,0 : PRINT " keys";
6300 PRINT " do not work, press the "; : COLOR 0,7 : PRINT " Num Lock "; : COLOR 7,0 : PRINT " key,"
6310 LOCATE 23,11 : PRINT "and try again."
6320 I$=INKEY$ : IF I$(<)CHR$(32) THEN 6320 ELSE 3130
6330

```

```

6340 LOCATE 4,25 : PRINT "Pressurizer Internal Diameter"
6350 LOCATE 8, 7 : PRINT "1. The default value in inches is equal to licensed power in MW"
6360 LOCATE 9,11 : PRINT "raised to the 1/3 power times 6.01203. This gives the exact"
6370 LOCATE 10,11 : PRINT "value, 90.5 inches, for McGuire 1, a 3411 MW, four-loop unit."
6380 LOCATE 11,11 : PRINT "For other units, the default value will be approximate."
6390 LOCATE 13, 7 : PRINT "2. Thermal power is not very sensitive to pressurizer water level"
6400 LOCATE 14,11 : PRINT "changes and pressurizer diameter. The analysis can be used"
6410 LOCATE 15,11 : PRINT "to demonstrate that fact."
6420 LOCATE 17, 7 : PRINT "3. The actual internal diameter of the pressurizer can be obtained"
6430 LOCATE 18,11 : PRINT "from the final safety analysis report or pressurizer drawings." : RETURN
6440
6450 LOCATE 3,30 : PRINT "Reactor Coolant Pump"
6460 LOCATE 5, 7 : PRINT "1. The default value for pump power in MW is equal to licensed power"
6470 LOCATE 6,11 : PRINT "in MW divided by the number of loops times 0.0062152. This gives"
6480 LOCATE 7,11 : PRINT "the exact value, 5.3 MW, for McGuire 1, a 3411 MW, four-loop unit."
6490 LOCATE 8,11 : PRINT "For other units, the default value will be approximate."
6500 LOCATE 10, 7 : PRINT "2. Any error in pump power will be multiplied by the number of loops"
6510 LOCATE 11,11 : PRINT "and included in thermal power."
6520 LOCATE 13, 7 : PRINT "3. For a more accurate value of pump power, use the name plate value"
6530 LOCATE 14,11 : PRINT "for running power, as opposed to starting power; or use the product"
6540 LOCATE 15,11 : PRINT "of current, voltage, and power factor."
6550 LOCATE 17, 7 : PRINT "4. For pump efficiency, that is the ratio of the power transmitted"
6560 LOCATE 18,11 : PRINT "to the coolant by the impeller to the electrical power supplied"
6570 LOCATE 19,11 : PRINT "to the motor, use the value warranted by the vendor or the default"
6580 LOCATE 20,11 : PRINT "value."
6590 LOCATE 22, 7 : PRINT "5. The default value for pump efficiency is 90%." : RETURN
6600
6610 LOCATE 2,18 : PRINT "Reflective Insulation - Inside Surface Area"
6620 LOCATE 4, 7 : PRINT "1. For most newer plants, all thermal insulation on the reactor cool-"
6630 LOCATE 5,11 : PRINT "ing system is reflective insulation. For older plants, the reactor"
6640 LOCATE 6,11 : PRINT "vessel may be covered with reflective insulation, and the rest of"
6650 LOCATE 7,11 : PRINT "the system with nonreflective insulation."
6660 LOCATE 9, 7 : PRINT "2. The surface area that is required is the outside surface area of"
6670 LOCATE 10,11 : PRINT "the reactor vessel, the reactor coolant pumps, the steam generators,"
6680 LOCATE 11,11 : PRINT "and the connecting piping. The surface area of the pressurizer and"
6690 LOCATE 12,11 : PRINT "the surge pipe is not required."
6700 LOCATE 14, 7 : PRINT "3. If the system is entirely insulated with reflective insulation, the"
6710 LOCATE 15,11 : PRINT "default value may be used. Otherwise, the surface area that is"
6720 LOCATE 16,11 : PRINT "covered with reflective insulation should be estimated. Only modest"
6730 LOCATE 17,11 : PRINT "accuracy is required. An adequate estimate can be made by scaling"
6740 LOCATE 18,11 : PRINT "layout drawings in the final safety analysis report."
6750 LOCATE 20, 7 : PRINT "4. The default value in square feet is equal to 19.34 times licensed"
6760 LOCATE 21,11 : PRINT "power in MW raised to the 2/3 power plus 32.18 times the number of"
6770 LOCATE 22,11 : PRINT "loops raised to the 1/3 power times licensed power in MW raised to"
6780 LOCATE 23,11 : PRINT "the 2/3 power." : RETURN
6790
6800 LOCATE 4,18 : PRINT "Reflective Insulation - Heat Loss Coefficient"
6810 LOCATE 11, 7 : PRINT "1. The default value, 55 BTUs/hour/square foot, is realistic."
6820 LOCATE 13, 7 : PRINT "2. Thermal power is not very sensitive to this parameter."
6830 LOCATE 15, 7 : PRINT "3. The heat loss coefficient can be obtained from licensee or vendor"
6840 LOCATE 16,11 : PRINT "documents which specify or warrant its value." : RETURN
6850
6860 LOCATE 2,17 : PRINT "Nonreflective Insulation - Inside Surface Area"
6870 LOCATE 4, 7 : PRINT "1. For most newer plants, all thermal insulation on the reactor cool-"

```

APPENDIX E: PRM.BAS

```

6880 LOCATE 5,11 : PRINT "ing system is reflective insulation. For older plants, the reactor"
6890 LOCATE 6,11 : PRINT "vessel may be covered with reflective insulation, and the rest of"
6900 LOCATE 7,11 : PRINT "the system with nonreflective insulation."
6910 LOCATE 9, 7 : PRINT *2. The default value is the estimated outside surface area of the reactor*
6920 LOCATE 10,11 : PRINT "coolant piping, reactor coolant pumps, and steam generators. The"
6930 LOCATE 11,11 : PRINT "surface area of the pressurizer and surge pipe is not required and"
6940 LOCATE 12,11 : PRINT "is not included. The default value is 32.18 times times the number"
6950 LOCATE 13,11 : PRINT "of loops raised to the 1/3 power times licensed power in MW raised to"
6960 LOCATE 14,11 : PRINT "the 2/3 power."
6970 LOCATE 16, 7 : PRINT *3. If you use the default value for this parameter, the value for the*
6980 LOCATE 17,11 : PRINT "Inside Surface Area of Reflective Insulation should be the surface"
6990 LOCATE 18,11 : PRINT "area of the reactor vessel. Its estimated value is";
7000 PRINT USING "#####";D.REFLECT.AREA-D.NONREFLECT.AREA; : PRINT " square feet."
7010 LOCATE 20, 7 : PRINT *4. If necessary, note the value above, and later, use the correction*
7020 LOCATE 21,11 : PRINT "routine to enter it as the value of Inside Surface Area of Reflective"
7030 LOCATE 22,11 : PRINT "Insulation." : RETURN
7040
7050 LOCATE 4,18 : PRINT "Nonreflective Thermal Insulation - Thickness"
7060 LOCATE 11, 7 : PRINT *1. A rough, average value should be sufficient."
7070 LOCATE 13, 7 : PRINT *2. The default value, 4 inches, is realistic for surfaces covered*
7080 LOCATE 14,11 : PRINT "with nonreflective insulation." : RETURN
7090
7100 LOCATE 4,17 : PRINT "Nonreflective Insulation - Thermal Conductivity"
7110 LOCATE 11,11 : PRINT "The value of the thermal conductivity of reflective insulation"
7120 LOCATE 12,11 : PRINT "can be obtained from licensee or vendor documents which specify"
7130 LOCATE 13,11 : PRINT "or warrant the value; however, the default value, 0.035"
7140 LOCATE 14,11 : PRINT "BTUs/hour/foot/";CHR$(248);"F, is realistic." : RETURN
7150
7160 LOCATE 4,26 : PRINT "Steam Dome - Inside Diameter"
7170 LOCATE 9, 6 : PRINT *1. The default value in inches is equal to 13.7089 times the ratio"
7180 LOCATE 10,10 : PRINT "raised to the 1/3 power of licensed power in MW to the number"
7190 LOCATE 11,10 : PRINT "of loops. This gives the exact value, 130 inches, for McGuire 1,"
7200 LOCATE 12,10 : PRINT "a 3411 MW, four-loop unit, and approximate results for other units."
7210 LOCATE 14, 6 : PRINT *2. For steam generators which may be significantly different from"
7220 LOCATE 15,10 : PRINT "those at McGuire, obtain the inside diameter from vendor drawings."
7230 LOCATE 16,10 : PRINT "The value should be taken at the elevation for normal operating"
7240 LOCATE 17,10 : PRINT "water level." : RETURN
7250
7260 LOCATE 4,32 : PRINT "Swirl Vane Risers"
7270 LOCATE 8, 6 : PRINT *1. The default value for the number risers is for McGuire 1, a 3411 MW,"
7280 LOCATE 9,10 : PRINT "four-loop unit."
7290 LOCATE 11, 6 : PRINT *2. The default value for riser diameter in inches is equal to 2.0036"
7300 LOCATE 12,10 : PRINT "times the ratio raised to the 1/3 power of licensed power in MW to"
7310 LOCATE 13,10 : PRINT "the number of loops. This gives the exact value for McGuire 1 and"
7320 LOCATE 14,10 : PRINT "approximate values for other units."
7330 LOCATE 16, 6 : PRINT *3. For steam generators which may be significantly different from those"
7340 LOCATE 17,10 : PRINT "at McGuire 1, obtain the values from vendor drawings." : RETURN
7350
7360 LOCATE 4,31 : PRINT "Moisture Carry-over"
7370 LOCATE 9, 6 : PRINT *1. Use of the default value, 0.125%, is recommended unless the licensee"
7380 LOCATE 10,10 : PRINT "has actually measured moisture carry-over. Some licensees do this"
7390 LOCATE 11,10 : PRINT "periodically using a radioactive tracer technique."
7400 LOCATE 13, 6 : PRINT *2. Westinghouse usually warrants their steam generators for less"
7410 LOCATE 14,10 : PRINT "than 0.250% moisture carry-over."

```

```

7420 LOCATE 16, 6 : PRINT "3. If the the warranted value of moisture carry-over is used and if"
7430 LOCATE 17,10 : PRINT "the actual value is less than the warranted value, the calculated"
7440 LOCATE 18,10 : PRINT "value of power will be too small." : RETURN
7450
7460 CLS : LOCATE 1,28 : PRINT "HELP FOR PARAMETER INPUT" : GOTO 7490
7470 CLS : LOCATE 2,28 : PRINT "HELP FOR PARAMETER INPUT" : GOTO 7490
7480 CLS : LOCATE 3,28 : PRINT "HELP FOR PARAMETER INPUT"
7490 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT " To continue, press the ";
7500 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT "."; : RETURN
7510
7520 Z%=INKEY% : IF Z%<>CHR$(32) THEN 7520 ELSE RETURN
7530
7540 KEY OFF : KEY 4,"" : KEY 6,"" : KEY 8,""
7550 CLS : LOCATE 25,1 : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 : PRINT " = help, ";
7560 COLOR 0,7 : PRINT " ";CHR$(25);" "; : COLOR 7,0 : PRINT " = down, "; : COLOR 0,7
7570 PRINT " ";CHR$(24);" "; : COLOR 7,0
7580 PRINT " = up, "; : COLOR 16,7 : PRINT " F6 "; : COLOR 7,0 : PRINT " = initiate value entry, & ";
7590 COLOR 0,7 : PRINT " F8 "; : COLOR 7,0 : PRINT " = exit." : RETURN
7600
7610 CLS : LOCATE 3,26 : PRINT "HELP FOR PARAMETER CORRECTION"
7620 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT " To continue, press the ";
7630 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT "."; : RETURN
7640
7650 Z%=INKEY% : IF Z%<>CHR$(32) THEN 7650 ELSE GOTO 4700
7660 KEY 4,"-9"+CHR$(13) : LOCATE 25,1 : PRINT SPC(8) : LOCATE 25,6
7670 COLOR 7,0 : PRINT " Press "; : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 :PRINT " for help -or- ";
7680 PRINT "Key in the new value, and press ";
7690 COLOR 16,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT "."SPC(7); : RETURN
7700
7710 LOCATE 25,21 : PRINT "After responding, strike the ";
7720 COLOR 16,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT " key."; : RETURN
7730 LOCATE 25,1 : PRINT TAB(15);"Key in the correct name or number, and press "; : COLOR 16,7
7740 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT ".";SPC(13) : RETURN
7750
7760 LOCATE 25,1 : PRINT SPC(78) : LOCATE 25,21 : PRINT "Key in the new value and, press "; : COLOR 16,7
7770 PRINT " ";CHR$( 17)+CHR$(196)+CHR$(217);" "; : COLOR 7,0 : PRINT "."; : RETURN
7780
7790
7800
7810
7820
7830
7840 COLOR 7,0 : LZ=3 : BZ=5 : CZ=60
7850 LOCATE LZ ,29 : PRINT "REACTOR COOLING SYSTEM"
7860 LOCATE LZ+ 2,CZ+5 : PRINT "Values" : LOCATE LZ+3,CZ-1 : PRINT C1% : LOCATE ,CZ+11 : PRINT C2%
7870 LOCATE LZ+ 3,BZ-2 : PRINT "Piping and Components"
7880 LOCATE LZ+ 5,BZ : PRINT "Pressurizer internal diameter (inches) "
7890 LOCATE LZ+ 6,BZ : PRINT "Power input for each reactor coolant pump (MW) "
7900 LOCATE LZ+ 7,BZ : PRINT "Efficiency of reactor coolant pumps (%)"
7910 LOCATE LZ+ 9,BZ-2 : PRINT "Reflective Thermal Insulation"
7920 LOCATE LZ+11,BZ : PRINT "Inside surface area (square feet) "
7930 LOCATE LZ+12,BZ : PRINT "Heat loss coefficient (BTUs/hour/square foot) "
7940 LOCATE LZ+14,BZ-2 : PRINT "Nonreflective Thermal Insulation"
7950 LOCATE LZ+16,BZ : PRINT "Inside surface area (square feet) "

```

APPENDIX E: PRM.BAS

```

7960 LOCATE LZ+17,BZ : PRINT "Thickness (inches) "
7970 LOCATE LZ+18,BZ : PRINT "Thermal conductivity (Btu/hour/foot/";CHR$(248);"F) " : RETURN
7980
7990 LOCATE LZ+ 5,CZ+11 : COLOR 0,7 : PRINT " . "
8000 LOCATE LZ+ 6,CZ+12 : PRINT " . "
8010 LOCATE LZ+ 7,CZ+12 : PRINT " . "
8020 LOCATE LZ+11,CZ+ 9 : PRINT " "
8030 LOCATE LZ+12,CZ+11 : PRINT " . "
8040 LOCATE LZ+16,CZ+ 9 : PRINT " "
8050 LOCATE LZ+17,CZ+12 : PRINT " . "
8060 LOCATE LZ+18,CZ+13 : PRINT " . " : COLOR 7,0 : RETURN
8070
8080 COLOR 7,0 : BZ=5 : CZ=60 : IF LOOPSZ=4 THEN LZ=5 ELSE IF LOOPSZ=3 THEN LZ=6 ELSE LZ=7
8090 LOCATE LZ- 1,32 : PRINT "STEAM GENERATORS"
8100 LOCATE LZ+ 1,CZ+4 : PRINT "Values" : LOCATE LZ+2,CZ-2 : PRINT C1$; : LOCATE ,CZ+10 : PRINT C2$
8110 LOCATE LZ+ 4,8Z : PRINT "Steam dome inside diameter (inches) "
8120 LOCATE LZ+ 6,8Z : PRINT "Number of swirl vane risers "
8130 LOCATE LZ+ 7,8Z : PRINT "Outside diameter of swirl vane risers (inches) "
8140 LOCATE LZ+ 9,8Z : PRINT "Measured or warranted moisture carry-over (%):"
8150 LOCATE LZ+11,8Z+2 : PRINT "Steam Generator A " : RETURN
8160
8170 COLOR 7,0 : LZ=3 : BZ=5 : CZ=60
8180 LOCATE LZ-1 ,29 : PRINT "REACTOR COOLING SYSTEM"
8190 LOCATE LZ+ 1,CZ+5 : PRINT "Values" : LOCATE LZ+2,CZ-1 : PRINT C1$; : LOCATE ,CZ+11 : PRINT C2$
8200 LOCATE LZ+ 2,8Z-2 : PRINT "Piping and Components"
8210 LOCATE LZ+4,8Z : PRINT "Number of reactor cooling loops"
8220 LOCATE LZ+4,CZ+13 : COLOR 0,7 : PRINT " " : COLOR 7,0 : GOTO 7880
8230 END

```

APPENDIX F

DATA.BAS

TABLE OF CONTENTS

| | Page |
|------------------|--------------|
| INPUT | ---- F- 1 |
| CORRECTIONS | F- 4 |
| Date & Time | F- 5 |
| Steam Generators | F- 5 |
| Other Components | F- 6 |
| HELP MESSAGES | F- 8 |
| DISPLAY | F-11 |

APPENDIX F: DATA.BAS

```

550 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 550
560 X$="DATA SET 2 PREP" : CLS : LOCATE 13,18
570 PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY" : CHAIN "PRM&DATA"
580 '
590 GOSUB 4530 : LOCATE 12,28 : PRINT "Will you use one or two" : LOCATE 13,28 : PRINT "sets of data (1/2)?"
600 LOCATE 13,49 : INPUT " ",DATA.SETSZ
610 IF DATA.SETSZ=1 OR DATA.SETSZ=2 THEN 640 ELSE IF DATA.SETSZ=-9 THEN GOSUB 4300 ELSE LOCATE 16,28 :
COLOR 0,7 : PRINT " PLEASE, TYPE 1 OR 2! " : COLOR 7,0 : LOCATE 13,49,1
620 Z$=INKEY$ : IF Z$="" THEN INPUT " ",DATA.SETSZ : GOTO 610 ELSE 600
630 '
640 NZ=1
650 FOR TX=NZ TO DATA.SETSZ
660 GOSUB 3660 : GOSUB 4470 : GOSUB 5000 : COLOR 0,7 : IF TX=1 THEN LOCATE 4,15 : PRINT SPC(19)
670 LOCATE 4,64 : PRINT " " : COLOR 7,0
680 FOR IX=1 TO LOOPSZ : IF IX=1 THEN S$="A" ELSE IF IX=2 THEN S$="B" ELSE IF IX=3 THEN S$="C" ELSE S$="D"
690 IF LOOPSZ=2 THEN CZ=52 ELSE IF LOOPSZ=3 THEN CZ=42 ELSE IF LOOPSZ=4 THEN CZ=32
700 DX=CZ+10*IX
710 LOCATE 6,DX+2 : PRINT S$ : COLOR 0,7
720 LOCATE 8,DX-1 : PRINT " ."
730 LOCATE 9,DX : PRINT " ."
740 LOCATE 10,DX : PRINT " ."
750 LOCATE 11,DX : PRINT " ."
760 LOCATE 12,DX : PRINT " ."
770 LOCATE 13,DX : PRINT " ." : COLOR 7,0 : NEXT : COLOR 0,7
780 LOCATE 17,33 : PRINT " ."
790 LOCATE 18,33 : PRINT " ."
800 LOCATE 22,32 : PRINT " ."
810 LOCATE 23,33 : PRINT " ."
820 LOCATE 17,72 : PRINT " ."
830 LOCATE 18,72 : PRINT " ."
840 LOCATE 22,72 : PRINT " ."
850 LOCATE 23,72 : PRINT " ." : COLOR 7,0
860 '
870 IF TX=2 THEN LOCATE 4,15 : PRINT DDATE$ : GOTO 950
880 '
890 IF X$="DDATE$" THEN HELP.FLAG$="C"
900 LOCATE 4,15 : IF HELP.FLAG$="S" THEN 930
910 INPUT "", DDATE$ : IF DDATE$="@*" THEN HELP.FLAG$="S" : X$="DDATE$" : GOTO 3870
920 IF DDATE$="*" THEN 4440
930 LOCATE 4,15 : PRINT USING "\ " ; DDATE$
940 '
950 KEY 4,"-9"+CHR$(13) : KEY 6,"-8"+CHR$(13)
960 '
970 IF X$="TTIME" THEN HELP.FLAG$="C"
980 LOCATE 4,64 : IF HELP.FLAG$="S" THEN 1010
990 INPUT "", TTIME(TX) : IF TTIME(TX)=-9 THEN HELP.FLAG$="S" : X$="TTIME" : GOTO 4250
1000 IF TTIME(TX)=-8 THEN 4440
1010 LOCATE 4,64 : PRINT USING "####";TTIME(TX)
1020 IF TTIME(TX)<1000 THEN LOCATE 4,64 : PRINT "0"
1030 IF TTIME(TX)<100 THEN LOCATE 4,65 : PRINT "0"
1040 IF TTIME(TX)<10 THEN LOCATE 4,66 : PRINT "0"
1050 '
1060 FOR LPZ=1 TO LOOPSZ
1070 IF LOOPSZ=2 THEN CZ=52 ELSE IF LOOPSZ=3 THEN CZ=42 ELSE IF LOOPSZ=4 THEN CZ=32

```

```

1080 DZ=CZ+10*LPZ
1090 IF LPZ=1 THEN S$="A" ELSE IF LPZ=2 THEN S$="B" ELSE IF LPZ=3 THEN S$="C" ELSE IF LPZ=4 THEN S$="D"
1100 '
1110 IF X$="STEAM.PRES."+S$ THEN HELP.FLAG$="C"
1120 LOCATE 8,DZ-1 : IF HELP.FLAG$="S" THEN 1160
1130 INPUT **, STEAM.PRES(LPZ,TZ)
1140 IF STEAM.PRES(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="STEAM.PRES."+S$ : GOTO 4010
1150 IF STEAM.PRES(LPZ,TZ)=-8 THEN 4440
1160 LOCATE 8,DZ-1 : PRINT USING "###.#"; STEAM.PRES(LPZ,TZ)
1170 '
1180 IF X$="FEEDWATER.FLOW."+S$ THEN HELP.FLAG$="C"
1190 LOCATE 9,DZ : IF HELP.FLAG$="S" THEN 1230
1200 INPUT **, FEEDWATER.FLOW(LPZ,TZ)
1210 IF FEEDWATER.FLOW(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="FEEDWATER.FLOW."+S$ : GOTO 4010
1220 IF FEEDWATER.FLOW(LPZ,TZ)=-8 THEN 4440
1230 LOCATE 9,DZ : PRINT USING "#.###"; FEEDWATER.FLOW(LPZ,TZ)
1240 '
1250 IF X$="FEEDWATER.TEMP."+S$ THEN HELP.FLAG$="C"
1260 LOCATE 10,DZ : IF HELP.FLAG$="S" THEN 1290
1270 INPUT **, FEEDWATER.TEMP(LPZ,TZ)
1280 IF FEEDWATER.TEMP(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="FEEDWATER.TEMP."+S$ : GOTO 4010
1290 IF FEEDWATER.TEMP(LPZ,TZ)=-8 THEN 4440
1300 LOCATE 10,DZ : PRINT USING "###.#"; FEEDWATER.TEMP(LPZ,TZ)
1310 '
1320 IF X$="TOP.BLOW."+S$ THEN HELP.FLAG$="C"
1330 LOCATE 11,DZ : IF HELP.FLAG$="S" THEN 1360
1340 INPUT **, TOP.BLOW(LPZ,TZ) : IF TOP.BLOW(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="TOP.BLOW."+S$ : GOTO 4010
1350 IF TOP.BLOW(LPZ,TZ)=-8 THEN 4440
1360 LOCATE 11,DZ : PRINT USING "###.#"; TOP.BLOW(LPZ,TZ)
1370 '
1380 IF X$="BOTTOM.BLOW."+S$ THEN HELP.FLAG$="C"
1390 LOCATE 12,DZ : IF HELP.FLAG$="S" THEN 1430
1400 INPUT **, BOTTOM.BLOW(LPZ,TZ)
1410 IF BOTTOM.BLOW(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="BOTTOM.BLOW."+S$ : GOTO 4010
1420 IF BOTTOM.BLOW(LPZ,TZ)=-8 THEN 4440
1430 LOCATE 12,DZ : PRINT USING "###.#"; BOTTOM.BLOW(LPZ,TZ)
1440 '
1450 IF X$="SG.WATER.LEVEL."+S$ THEN HELP.FLAG$="C"
1460 LOCATE 13,DZ : IF HELP.FLAG$="S" THEN 1500
1470 INPUT **, SG.WATER.LEVEL(LPZ,TZ)
1480 IF SG.WATER.LEVEL(LPZ,TZ)=-9 THEN HELP.FLAG$="S" : X$="SG.WATER.LEVEL."+S$ : GOTO 4010
1490 IF SG.WATER.LEVEL(LPZ,TZ)=-8 THEN 4440
1500 LOCATE 13,DZ : PRINT USING "###.#"; SG.WATER.LEVEL(LPZ,TZ) : NEXT
1510 '
1520 IF X$="LETDOWN.FLOW" THEN HELP.FLAG$="C"
1530 LOCATE 17,33 : IF HELP.FLAG$="S" THEN 1560
1540 INPUT **, LETDOWN.FLOW(TZ) : IF LETDOWN.FLOW(TZ)=-9 THEN HELP.FLAG$="S" : X$="LETDOWN.FLOW" : GOTO 4010
1550 IF LETDOWN.FLOW(TZ)=-8 THEN 4440
1560 LOCATE 17,33 : PRINT USING "###.#"; LETDOWN.FLOW(TZ)
1570 '
1580 IF X$="LETDOWN.TEMP" THEN HELP.FLAG$="C"
1590 LOCATE 18,33 : IF HELP.FLAG$="S" THEN 1620
1600 INPUT **, LETDOWN.TEMP(TZ) : IF LETDOWN.TEMP(TZ)=-9 THEN HELP.FLAG$="S" : X$="LETDOWN.TEMP" : GOTO 4010

```

APPENDIX F: DATA.BAS

```

1610 IF LETDOWN.TEMP(TX)=-8 THEN 4440
1620 LOCATE 18,33 : PRINT USING "###.#"; LETDOWN.TEMP(TX)
1630 '
1640 IF X$="CHARGING.FLOW" THEN HELP.FLAG$="C"
1650 LOCATE 17,72 : IF HELP.FLAG$="S" THEN 1690
1660 INPUT **, CHARGING.FLOW(TX)
1670 IF CHARGING.FLOW(TX)=-9 THEN HELP.FLAG$="S" : X$="CHARGING.FLOW" : GOTO 4010
1680 IF CHARGING.FLOW(TX)=-8 THEN 4440
1690 LOCATE 17,72 : PRINT USING "###.#"; CHARGING.FLOW(TX)
1700 '
1710 IF X$="CHARGING.TEMP" THEN HELP.FLAG$="C"
1720 LOCATE 18,72 : IF HELP.FLAG$="S" THEN 1760
1730 INPUT **, CHARGING.TEMP(TX)
1740 IF CHARGING.TEMP(TX)=-9 THEN HELP.FLAG$="S" : X$="CHARGING.TEMP" : GOTO 4010
1750 IF CHARGING.TEMP(TX)=-8 THEN 4440
1760 LOCATE 18,72 : PRINT USING "###.#"; CHARGING.TEMP(TX)
1770 '
1780 IF X$="PZR.PRES" THEN HELP.FLAG$="C"
1790 LOCATE 22,32 : IF HELP.FLAG$="S" THEN 1820
1800 INPUT **, PZR.PRES(TX) : IF PZR.PRES(TX)=-9 THEN HELP.FLAG$="S" : X$="PZR.PRES" : GOTO 4010
1810 IF PZR.PRES(TX)=-8 THEN 4440
1820 LOCATE 22,32 : PRINT USING "####.#"; PZR.PRES(TX)
1830 '
1840 IF X$="PZR.WATER.LEVEL" THEN HELP.FLAG$="C"
1850 LOCATE 23,33 : IF HELP.FLAG$="S" THEN 1890
1860 INPUT **, PZR.WATER.LEVEL(TX)
1870 IF PZR.WATER.LEVEL(TX)=-9 THEN HELP.FLAG$="S" : X$="PZR.WATER.LEVEL" : GOTO 4010
1880 IF PZR.WATER.LEVEL(TX)=-8 THEN 4440
1890 LOCATE 23,33 : PRINT USING "###.#"; PZR.WATER.LEVEL(TX)
1900 '
1910 IF X$="T.AVE" THEN HELP.FLAG$="C"
1920 LOCATE 22,72 : IF HELP.FLAG$="S" THEN 1950
1930 INPUT **, T.AVE(TX) : IF T.AVE(TX)=-9 THEN HELP.FLAG$="S" : X$="T.AVE" : GOTO 4010
1940 IF T.AVE(TX)=-8 THEN 4440
1950 LOCATE 22,72 : PRINT USING "###.#"; T.AVE(TX)
1960 '
1970 IF X$="T.COLD" THEN HELP.FLAG$="C"
1980 LOCATE 23,72 : IF HELP.FLAG$="S" THEN 2010
1990 INPUT **, T.COLD(TX) : IF T.COLD(TX)=-9 THEN HELP.FLAG$="S" : X$="T.COLD" : GOTO 4010
2000 IF T.COLD(TX)=-8 THEN 4440
2010 LOCATE 23,72 : PRINT USING "###.#"; T.COLD(TX) : GOSUB 4390 : NEXT TX
2020 GOTO 4430
2030 '
2040 '
2050 '
2060 '
2070 '
2080 '
2090 FOR TX=1 TO DATA.SETS% : LOCATE 1,1,0 : COLOR 7,0 : L$="4L"
2100 '
2110 GOSUB 4580 : GOSUB 5030 : GOSUB 5460 : GOSUB 5480
2120 '
2130 IF L$="4L" THEN 2180 ELSE IF L$="4R" THEN 2270 ELSE IF X$="SG" THEN 2360 ELSE IF X$="DC" THEN 2930
2140 '

```

```

2150 '
2160 '                ** DATE & TIME **
2170 '
2180 LOCATE 4,15 : L$="4L" : COLOR 0,7
2190 IF TZ=1 THEN PRINT USING "\ ;DDATE$ : LOCATE 4,15,1 ELSE 2270
2200 Z$=INKEY$ : IF Z$="" THEN 2200
2210 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN LOCATE 4,15 : COLOR 7,0 : PRINT USING "\ ;DDATE$ :
      GOTO 2270
2220 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2230 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 4,15 : COLOR 0,7 : INPUT "",DDATE$ :
      COLOR 7,0 : GOTO 2110
2240 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN COLOR 7,0 : PRINT USING "\ ;DDATE$ : GOTO 3500
2250 GOTO 2200
2260 '
2270 LOCATE 4,64 : L$="4R" : COLOR 0,7 : PRINT TTIME$(TZ) : LOCATE 4,64,1
2280 Z$=INKEY$ : IF Z$="" THEN 2280
2290 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT TTIME$(TZ) : GOTO 2360
2300 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2310 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 4,64 : COLOR 0,7 : INPUT "",TTIME$(TZ) :
      COLOR 7,0 : GOTO 2110
2320 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 4,64 : COLOR 7,0 : PRINT TTIME$(TZ) : ELSE 2280
2330 IF TZ=1 THEN 2180 ELSE 3500
2340 '
2350 '
2360 '                ** STEAM GENERATORS **
2370 '
2380 X$="SG"
2390 DX=CX+LPX+10
2400 IF L$="8" THEN 2470 ELSE IF L$="9" THEN 2570 ELSE IF L$="10" THEN 2640 ELSE IF L$="11" THEN 2710
2410 IF L$="12" THEN 2770 ELSE IF L$="13" THEN 2830
2420 LPX=1 : DX=CX+10 : GOTO 2470
2430 LPX=2 : DX=CX+20 : GOTO 2470
2440 IF LOOPSZ=3 OR LOOPSZ=4 THEN LPX=3 : DX=CX+30 : GOTO 2470
2450 IF LOOPSZ=4 THEN LPX=4 : DX=CX+40 : GOTO 2470 ELSE 2930
2460 '
2470 LOCATE 8,DX-1 : L$="8" : COLOR 0,7 : PRINT USING "####.#";STEAM.PRES(LPZ,TX) : LOCATE 8,DX-1,1
2480 Z$=INKEY$ : IF Z$="" THEN 2480
2490 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "####.#";STEAM.PRES(LPZ,TX) : GOTO 2570
2500 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2510 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 8,DX-1 : COLOR 0,7 :
      INPUT "",STEAM.PRES(LPZ,TX) : COLOR 7,0 : GOTO 2110
2520 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 8,DX-1 : COLOR 7,0 : PRINT USING "####.#";STEAM.PRES(LPZ,TX)
      : GOTO 2530 ELSE 2480
2530 IF LPX=1 THEN 2270 ELSE IF LPX=2 THEN LPX=1 : DX=CX+10 : ELSE IF LPX=3 THEN LPX=2 : DX=CX+20
2540 IF LPX=4 THEN LPX=3 : DX=CX+30
2550 GOTO 2830
2560 '
2570 LOCATE 9,DX : L$="9" : COLOR 0,7 : PRINT USING "#.###";FEEDWATER.FLOW(LPZ,TX) : LOCATE 9,DX
2580 Z$=INKEY$ : IF Z$="" THEN 2580
2590 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "#.###";FEEDWATER.FLOW(LPZ,TX) : GOTO 2640
2600 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2610 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 9,DX : COLOR 0,7 :
      INPUT "",FEEDWATER.FLOW(LPZ,TX) : COLOR 7,0 : GOTO 2110

```

APPENDIX F: DATA.BAS

```

2620 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 9,DZ : COLOR 7,0 : PRINT USING "###.#";FEEDWATER.FLOW(LPZ,TZ)
: GOTO 2470 ELSE 2580
2630
2640 LOCATE 10,DZ : L$="10" : COLOR 0,7 : PRINT USING "###.#";FEEDWATER.TEMP(LPZ,TZ) : LOCATE 10,DZ
2650 Z$=INKEY$ : IF Z$="" THEN 2650
2660 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";FEEDWATER.TEMP(LPZ,TZ) : GOTO 2710
2670 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2680 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 10,DZ : COLOR 0,7 :
INPUT "",FEEDWATER.TEMP(LPZ,TZ) : COLOR 7,0 : GOTO 2110
2690 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 10,DZ : COLOR 7,0 : PRINT USING "###.#";
FEEDWATER.TEMP(LPZ,TZ) : GOTO 2570 ELSE 2650
2700
2710 LOCATE 11,DZ : L$="11" : COLOR 0,7 : PRINT USING "###.#";TOP.BLOW(LPZ,TZ) : LOCATE 11,DZ
2720 Z$=INKEY$ : IF Z$="" THEN 2720 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";
TOP.BLOW(LPZ,TZ) : GOTO 2770
2730 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2740 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 11,DZ : COLOR 0,7 : INPUT "",TOP.BLOW(LPZ,TZ)
: COLOR 7,0 : GOTO 2110
2750 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 11,DZ : COLOR 7,0 : PRINT USING "###.#";TOP.BLOW(LPZ,TZ) :
GOTO 2640 ELSE 2720
2760
2770 LOCATE 12,DZ : L$="12" : COLOR 0,7 : PRINT USING "###.#";BOTTOM.BLOW(LPZ,TZ) : LOCATE 12,DZ
2780 Z$=INKEY$ : IF Z$="" THEN 2780 ELSE IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";
BOTTOM.BLOW(LPZ,TZ) : GOTO 2830
2790 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2800 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 12,DZ : COLOR 0,7 : INPUT "",
BOTTOM.BLOW(LPZ,TZ) : COLOR 7,0 : GOTO 2110
2810 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 12,DZ : COLOR 7,0 : PRINT USING "###.#";BOTTOM.BLOW(LPZ,TZ)
: GOTO 2710 ELSE 2780
2820
2830 LOCATE 13,DZ : L$="13" : COLOR 0,7 : PRINT USING "###.#";SG.WATER.LEVEL(LPZ,TZ) : LOCATE 13,DZ
2840 Z$=INKEY$ : IF Z$="" THEN 2840
2850 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";SG.WATER.LEVEL(LPZ,TZ) : GOTO 2900
2860 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
2870 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 13,DZ : COLOR 0,7 : INPUT "",
SG.WATER.LEVEL(LPZ,TZ) : COLOR 7,0 : GOTO 2110
2880 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 13,DZ : COLOR 7,0 : PRINT USING "###.#";
SG.WATER.LEVEL(LPZ,TZ) : GOTO 2770 ELSE 2840
2890
2900 IF LPZ=LOOPS% THEN 2930 ELSE IF LPZ=1 THEN 2430 ELSE IF LPZ=2 THEN 2440 ELSE IF LPZ=3 THEN 2450 ELSE STOP
2910
2920
2930
2940
2950 IF L$="17L" THEN 2980 ELSE IF L$="18L" THEN 3080 ELSE IF L$="17R" THEN 3150 ELSE IF L$="18R" THEN 3220
2960 IF L$="22L" THEN 3270 ELSE IF L$="23L" THEN 3360 ELSE IF L$="22R" THEN 3430 ELSE IF L$="23R" THEN 3500
2970 X$="DC"
2980 LOCATE 17,33 : L$="17L" : COLOR 0,7 : PRINT USING "###.#";LETDOWN.FLOW(TZ) : LOCATE 17,33
2990 Z$=INKEY$ : IF Z$="" THEN 2990
3000 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";LETDOWN.FLOW(TZ) : GOTO 3080
3010 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3020 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 17,33 : COLOR 0,7 : INPUT "",LETDOWN.FLOW(TZ)
: COLOR 7,0 : GOTO 2110

```

*** OTHER COMPONENTS ***

```

3030 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 17,33 : COLOR 7,0 : PRINT USING "###.#";LETDOWN.FLOW(T%);
      GOTO 3040 ELSE 2990
3040 IF LOOPSZ=2 THEN LPZ=2 : DZ=CZ+20 : ELSE IF LOOPSZ=3 THEN LPZ=3 : DZ=CZ+30
3050 IF LOOPSZ=4 THEN LPZ=4 : DZ=CZ+40
3060 Z$="56" : GOTO 2830
3070 '
3080 LOCATE 18,33 : L$="18L" : COLOR 0,7 : PRINT USING "###.#";LETDOWN.TEMP(T%); LOCATE 18,33
3090 Z$=INKEY$ : IF Z$="" THEN 3090
3100 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";LETDOWN.TEMP(T%); GOTO 3150
3110 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3120 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 18,33 : COLOR 0,7 : INPUT "",LETDOWN.TEMP(T%)
      : COLOR 7,0 : GOTO 2110
3130 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 18,33 : COLOR 7,0 : PRINT USING "###.#";LETDOWN.TEMP(T%);
      GOTO 2980 ELSE 3090
3140 '
3150 LOCATE 17,72 : L$="17R" : COLOR 0,7 : PRINT USING "###.#";CHARGING.FLOW(T%); LOCATE 17,72
3160 Z$=INKEY$ : IF Z$="" THEN 3160
3170 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";CHARGING.FLOW(T%); GOTO 3220
3180 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3190 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 17,72 : COLOR 0,7 : INPUT "",CHARGING.FLOW(T%)
      : COLOR 7,0 : GOTO 2110
3200 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 17,72 : COLOR 7,0 : PRINT USING "###.#";CHARGING.FLOW(T%);
      GOTO 3080 ELSE 3160
3210 '
3220 LOCATE 18,72 : L$="18R" : COLOR 0,7 : PRINT USING "###.#";CHARGING.TEMP(T%); LOCATE 18,72
3230 Z$=INKEY$ : IF Z$="" THEN 3230
3240 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";CHARGING.TEMP(T%); GOTO 3290
3250 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3260 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 18,72 : COLOR 0,7 : INPUT "",CHARGING.TEMP(T%)
      : COLOR 7,0 : GOTO 2110
3270 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 18,72 : COLOR 7,0 : PRINT USING "###.#";CHARGING.TEMP(T%);
      GOTO 3150 ELSE 3230
3280 '
3290 LOCATE 22,32 : L$="22L" : COLOR 0,7 : PRINT USING "###.#";PZR.PRES(T%); LOCATE 22,32
3300 Z$=INKEY$ : IF Z$="" THEN 3300
3310 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";PZR.PRES(T%); GOTO 3360
3320 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3330 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 22,32 : COLOR 0,7 : INPUT "",PZR.PRES(T%)
      : COLOR 7,0 : GOTO 2110
3340 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 22,32 : COLOR 7,0 : PRINT USING "###.#";PZR.PRES(T%);
      GOTO 3220 ELSE 3300
3350 '
3360 LOCATE 23,33 : L$="23L" : COLOR 0,7 : PRINT USING "###.#";PZR.WATER.LEVEL(T%); LOCATE 23,33
3370 Z$=INKEY$ : IF Z$="" THEN 3370
3380 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";PZR.WATER.LEVEL(T%); GOTO 3430
3390 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3400 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 23,33 : COLOR 0,7 : INPUT "",
      PZR.WATER.LEVEL(T%); COLOR 7,0 : GOTO 2110
3410 IF Z$=CHR$(0)+CHR$(72) OR Z$="8" THEN LOCATE 23,33 : COLOR 7,0 : PRINT USING "###.#";PZR.WATER.LEVEL(T%)
      : GOTO 3290 ELSE 3370
3420 '
3430 LOCATE 22,72 : L$="22R" : COLOR 0,7 : PRINT USING "###.#";T.AVE(T%); LOCATE 22,72
3440 Z$=INKEY$ : IF Z$="" THEN 3440
3450 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";T.AVE(T%); GOTO 3500

```

APPENDIX F: DATA.BAS

```

3460 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3470 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 22,72 : COLOR 0,7 : INPUT **,T.AVE(TZ) :
COLOR 7,0 : GOTO 2110
3480 IF Z$=CHR$(0)+CHR$(72) OR Z$="B" THEN LOCATE 22,72 : COLOR 7,0 : PRINT USING "###.#";T.AVE(TZ) :
GOTO 3360 ELSE 3440
3490 '
3500 LOCATE 23,72 : L$="23R" : COLOR 0,7 : PRINT USING "###.#";T.COLD(TZ) : LOCATE 23,72
3510 Z$=INKEY$ : IF Z$="" THEN 3510
3520 IF Z$=CHR$(0)+CHR$(80) OR Z$="2" THEN COLOR 7,0 : PRINT USING "###.#";T.COLD(TZ) : GOTO 3560
3530 IF Z$=CHR$(0)+CHR$(66) THEN 3580 ELSE IF Z$=CHR$(0)+CHR$(62) THEN 4690
3540 IF Z$=CHR$(0)+CHR$(64) THEN COLOR 7,0 : GOSUB 4880 : LOCATE 23,72 : COLOR 0,7 : INPUT **,T.COLD(TZ) :
COLOR 7,0 : GOTO 2110
3550 IF Z$=CHR$(0)+CHR$(72) OR Z$="B" THEN LOCATE 23,72 : COLOR 7,0 : PRINT USING "###.#";T.COLD(TZ) :
GOTO 3430 ELSE 3510
3560 IF TZ=1 THEN 2180 ELSE 2270
3570 '
3580 NEXT TZ
3590 COLOR 7,0 : CLS : LOCATE 12,24 : PRINT "LOADING PARAMETERS & DATA MODULE" : I$="PRM&DATA 6270"
3600 CHAIN "PRM&DATA"
3610 '
3620 ' *****
3630 ' * HELP MESSAGES *
3640 ' *****
3650 '
3660 CLS : KEY 4,"@"+CHR$(13) : LOCATE 11,21 : COLOR 0,7
3670 PRINT " CAUTION " ; : COLOR 7,0 : PRINT " To obtain accurate results,"
3680 PRINT TAB(21);"auxiliary feedwater flow must be zero."
3690 PRINT TAB(21);"Tempering flow, if any, must be known"
3700 PRINT TAB(21);"or included in the measured value of"
3710 PRINT TAB(21);"main feedwater flow." : LOCATE 25,16
3720 PRINT "Press " ; : COLOR 0,7 : PRINT " F4 " ; : COLOR 7,0 : PRINT " for help, or to continue " ;
3730 COLOR 0,7 : PRINT " space bar " ; : COLOR 7,0 : PRINT " ." ;
3740 Z$=INKEY$ : IF Z$="@" THEN 3760 ELSE IF Z$(<)CHR$(32) THEN 3740 ELSE RETURN
3750 '
3760 CLS : LOCATE 9,11
3770 PRINT "The auxiliary feedwater nozzle may be located in the steam"
3780 PRINT TAB(11);"filled region of the steam generator. Tempering flow is"
3790 PRINT TAB(11);"sometimes provided to avoid water hammer in the auxiliary"
3800 PRINT TAB(11);"feedwater line. A sidestream from the main feedwater line"
3810 PRINT TAB(11);"to the auxiliary feedwater line keeps it free from steam."
3820 PRINT TAB(11);"Thus, cold auxiliary feedwater, when introduced, contacts"
3830 PRINT TAB(11);"water, not steam, downstream from the auxiliary feedwater"
3840 PRINT TAB(11);"valve and precludes water hammer." : GOSUB 4950
3850 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 3850 ELSE RETURN
3860 '
3870 GOSUB 4380 : LOCATE 4,32 : PRINT "System Conditions"
3880 LOCATE 7,7 : PRINT "1. To obtain an accurate result for thermal power, the reactor"
3890 LOCATE 8,11 : PRINT "core and cooling system must be at thermal equilibrium."
3900 LOCATE 9,11 : PRINT "Specifically, T ave must be constant. Otherwise, the result"
3910 LOCATE 10,11 : PRINT "for thermal power will be in error by an amount that is equal"
3920 LOCATE 11,11 : PRINT "to the energy per unit time stored in or released from the core"
3930 LOCATE 12,11 : PRINT "and cooling system."
3940 LOCATE 14,7 : PRINT "2. In addition, flows and liquid levels should be constant. If"
3950 LOCATE 15,11 : PRINT "they are not, their effect can be included in the analysis by"

```



```

3960 LOCATE 16,11 : PRINT      "taking a second set of data within the next few hours. The"
3970 LOCATE 17,11 : PRINT      "analytical routine will calculate the effect of these changes"
3980 LOCATE 18,11 : PRINT      "and provide two adjusted values of power which will correspond"
3990 LOCATE 19,11 : PRINT      "with the two sets of data taken." : GOSUB 4410
4000 '
4010 GOSUB 4380 : LOCATE 4,35 : PRINT "Data Entry"
4020 LOCATE 6, 7 : PRINT "1. Values must be entered in order from top to bottom."
4030 LOCATE 8, 7 : PRINT "2. The number pad should be locked out, so that the left and right"
4040 LOCATE 9,11 : PRINT "arrow keys (the number pad ";CHR$(34);"4";CHR$(34);" and ";CHR$(34);"6";
4050 PRINT CHR$(34);" keys) will move the"
4060 LOCATE 10,11 : PRINT "cursor as desired. If the arrow keys are not working, press "; : COLOR 0,7
4070 LOCATE 11,11 : PRINT " Num Lock "; : COLOR 7,0 : PRINT "."
4080 LOCATE 13, 7 : PRINT "3. To enter an actual value, move the cursor to the appropriate"
4090 LOCATE 14,11 : PRINT "column relative to the decimal point and then key in the digits."
4100 LOCATE 15,11 : PRINT "If a wrong digit is keyed in, write over it. The number of"
4110 LOCATE 16,11 : PRINT "digits entered should be limited to the highlighted field. Do"
4120 LOCATE 17,11 : PRINT "not press the "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4130 COLOR 7,0 : PRINT " key until after the last digit is keyed in."
4140 LOCATE 19, 7 : PRINT "4. If an incorrect value is keyed in and entered, do not try to"
4150 LOCATE 20,11 : PRINT "correct it now. Complete the entry of data and return to"
4160 LOCATE 21,11 : PRINT "the DATA & PARAMETERS MENU. Then select the Corrections option." : GOSUB 4410
4170 IF X$="DDATE$" THEN 4190 ELSE 660
4180 '
4190 GOSUB 4380 : LOCATE 4,38 : PRINT "Date"
4200 LOCATE 11,16 : PRINT "The date may be entered in any format. However,"
4210 PRINT TAB(16);"if a comma is used, for example, to separate the"
4220 PRINT TAB(16);"day and year, then the first character keyed in" : PRINT TAB(16);"must be a quotation mark."
4230 GOSUB 4410 : GOTO 660
4240 '
4250 GOSUB 4380 : LOCATE 4,38 : PRINT "Time"
4260 LOCATE 12,18 : PRINT "Time must be keyed in using 24-hour format, for"
4270 PRINT TAB(18);"example, 1435 for 2:35 pm. A colon may not be"
4280 PRINT TAB(18);"used to separate hours and minutes." : GOSUB 4410 : GOTO 660
4290 '
4300 CLS : LOCATE 3,31 : PRINT "HELP FOR DATA INPUT"
4310 LOCATE 12,19 : PRINT "One set of data is adequate if pressurizer"
4320 LOCATE 13,19 : PRINT "level, steam generator levels, and feed-"
4330 LOCATE 14,19 : PRINT "water temperature are at equilibrium."
4340 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT "To continue, press the ";
4350 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".";
4360 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 4360 ELSE 590
4370 '
4380 CLS : LOCATE 3,31 : PRINT "HELP FOR DATA INPUT"
4390 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT " To continue, press the ";
4400 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT "."; : RETURN
4410 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 4410 ELSE RETURN
4420 Z$=INKEY$ : IF Z$="" THEN 4420 ELSE 660
4430 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 4430 ELSE CLS : X$="PRM&DATA 6270" : GOSUB 4450 : CHAIN "PRM&DATA"
4440 CLS : X$="PRM&DATA 340" : GOSUB 4450 : CHAIN "PRM&DATA"
4450 LOCATE 13,18 : PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY" : RETURN
4460 '
4470 CLS : COLOR 7,0 : KEY OFF : KEY 4,"@"+CHR$(13) : KEY 6,"#" +CHR$(13)
4480 LOCATE 25, 2 : PRINT "Press "; : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 :PRINT " for help";
4490 PRINT ", "; : COLOR 0,7 : PRINT " F6 "; : COLOR 7,0 :PRINT " for menu";

```

APPENDIX F: DATA.BAS

```

4500 PRINT " -or- Key in the value, and press "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4510 COLOR 7,0 : PRINT "." : RETURN
4520 '
4530 CLS : COLOR 7,0 : KEY OFF : KEY 4,"-9"+CHR$(13)
4540 LOCATE 25, 9 : PRINT "Press "; : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 : PRINT " for help";
4550 PRINT " -or- Key in the value, and press "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4560 COLOR 7,0 : PRINT "." : RETURN
4570 '
4580 KEY OFF : KEY 4,"" : KEY 6,"" : KEY 8,""
4590 CLS : LOCATE 25,2 : COLOR 0,7 : PRINT " F4 "; : COLOR 7,0 : PRINT " =help, ";
4600 COLOR 0,7 : PRINT " ";CHR$(25);" "; : COLOR 7,0 : PRINT " =down/right, "; : COLOR 0,7
4610 PRINT " ";CHR$(24);" "; : COLOR 7,0
4620 PRINT " =up/left, "; : COLOR 16,7 : PRINT " F6 "; : COLOR 7,0 : PRINT " =initiate entry, & ";
4630 COLOR 0,7 : PRINT " F8 "; : COLOR 7,0 : PRINT " =exit."; : RETURN
4640 '
4650 LOCATE 25,21 : PRINT "After responding, strike the "; : COLOR 16,7
4660 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4670 COLOR 7,0 : PRINT " key."; : RETURN
4680 '
4690 GOSUB 4940
4700 LOCATE 6, 7 : PRINT "1. If more than one entry need to be corrected, corrections can be"
4710 LOCATE 7,11 : PRINT "done in any sequence by moving the cursor."
4720 LOCATE 9, 7 : PRINT "2. The number pad should be locked out, so that the left, right, up,"
4730 LOCATE 10,11 : PRINT "and down arrow keys (the number pad ";CHR$(34);"2";CHR$(34);"3";CHR$(34);"4";
4740 PRINT CHR$(34);"5";CHR$(34);"6";CHR$(34);"7, and";CHR$(34);"8";CHR$(34);" keys)"
4750 LOCATE 11,11 : PRINT "will move the cursor as desired. If the arrow keys are not"
4760 LOCATE 12,11 : PRINT "working, press the "; : COLOR 0,7 : PRINT " Num Lock "; : COLOR 7,0 : PRINT "."
4770 LOCATE 14, 7 : PRINT "3. To correct the expression for the date, use quotation marks if"
4780 LOCATE 15,11 : PRINT "the expression includes a comma. To correct time, use 24-hour"
4790 LOCATE 16,11 : PRINT "format without a colon separating hours and seconds."
4800 LOCATE 18, 7 : PRINT "4. To correct a numerical value, move the cursor to the appropriate"
4810 LOCATE 19,11 : PRINT "column relative to the decimal point and then key in the digits."
4820 LOCATE 20,11 : PRINT "If a wrong digit is keyed in, write over it. The number of"
4830 LOCATE 21,11 : PRINT "digits entered should be limited to the highlighted field. Do"
4840 LOCATE 22,11 : PRINT "not press the "; : COLOR 0,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4850 COLOR 7,0 : PRINT " key until after the last digit is keyed in."
4860 Z$=INKEY$ : IF Z$="" THEN 4810 ELSE 2110
4870 '
4880 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25, 2 : PRINT "Move cursor with "; : COLOR 0,7
4890 PRINT " ";CHR$(196)+CHR$(16);" "; : COLOR 7,0 : PRINT " and "; : COLOR 0,7
4900 PRINT " ";CHR$(17)+CHR$(196);" "; : COLOR 7,0 : PRINT " -and- ";
4910 PRINT "Key in the new value, and press "; : COLOR 16,7 : PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
4920 COLOR 7,0 : PRINT "."; : RETURN
4930 '
4940 COLOR 7,0 : CLS : LOCATE 3,28 : PRINT "HELP FOR DATA CORRECTION"
4950 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT "To continue, press the ";
4960 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT "."; : RETURN
4970 '
4980 '

```

```

4990 ' *****
5000 ' * DISPLAY *
5010 ' *****
5020 '
5030 LOCATE 1,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5040 LOCATE 3,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5050 LOCATE 5,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5060 LOCATE 14,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5070 LOCATE 19,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5080 LOCATE 24,2 : FOR IX=2 TO 78 : PRINT CHR$(196); : NEXT
5090 FOR IX= 2 TO 23 : LOCATE IX, 1 : PRINT CHR$(179) : NEXT : LOCATE 4,40 : PRINT CHR$(179)
5100 FOR IX=15 TO 23 : LOCATE IX,40 : PRINT CHR$(179) : NEXT
5110 FOR IX= 2 TO 23 : LOCATE IX,79 : PRINT CHR$(179); : NEXT
5120 LOCATE 1, 1 : PRINT CHR$(218);: LOCATE 1,79 : PRINT CHR$(191);
5130 LOCATE 3, 1 : PRINT CHR$(195);: LOCATE 3,79 : PRINT CHR$(180);
5140 LOCATE 5, 1 : PRINT CHR$(195);: LOCATE 5,79 : PRINT CHR$(180);
5150 LOCATE 14, 1 : PRINT CHR$(195);: LOCATE 14,79 : PRINT CHR$(180);
5160 LOCATE 19, 1 : PRINT CHR$(195);: LOCATE 19,79 : PRINT CHR$(180);
5170 LOCATE 3,40 : PRINT CHR$(194); : LOCATE 5,40 : PRINT CHR$(193);
5180 LOCATE 14,40 : PRINT CHR$(194); : LOCATE 19,40 : PRINT CHR$(197);
5190 LOCATE 24,40 : PRINT CHR$(193);
5200 LOCATE 24, 1 : PRINT CHR$(192); : LOCATE 24,79 : PRINT CHR$(217); : LOCATE 1,1
5210 COLOR 7,0 : BX=5
5220 LOCATE 1,65 : PRINT " Set";TX;"of";DATA.SETSZ
5230 LOCATE 2,36 : PRINT "TEST DATA"
5240 LOCATE 4, 8 : PRINT "Date: "
5250 LOCATE 4,50 : PRINT "Time (hhmm): "
5260 LOCATE 6, 3 : PRINT "Steam Generators:"
5270 LOCATE 8, 5 : PRINT "Steam pressure (psia)"
5280 LOCATE 9, 5 : PRINT "Feedwater flow (millions of lb/hr)"
5290 LOCATE 10, 5 : PRINT "Feedwater temperature (*;CHR$(249);*F)"
5300 LOCATE 11, 5 : PRINT "Surface blowdown (gpm)"
5310 LOCATE 12, 5 : PRINT "Bottom blowdown (gpm)"
5320 LOCATE 13, 5 : PRINT "Water level (inches)"
5330 LOCATE 15, 3 : PRINT "Letdown Line:"
5340 LOCATE 17, 5 : PRINT "Flow (gpm)"
5350 LOCATE 18, 5 : PRINT "Temperature (*;CHR$(248);*F)"
5360 LOCATE 20, 3 : PRINT "Pressurizer:"
5370 LOCATE 22, 5 : PRINT "Pressure (psia)"
5380 LOCATE 23, 5 : PRINT "Water Level (inches)"
5390 LOCATE 15,42 : PRINT "Charging Line:"
5400 LOCATE 17,44 : PRINT "Flow (gpm)"
5410 LOCATE 18,44 : PRINT "Temperature (*;CHR$(248);*F)"
5420 LOCATE 20,42 : PRINT "Reactor:"
5430 LOCATE 22,44 : PRINT "T ave (*;CHR$(248);*F)"
5440 LOCATE 23,44 : PRINT "T cold (*;CHR$(248);*F)" : RETURN
5450 '
5460 TTIME$(TX)=RIGHT$(STR$(TTIME(TX)+10000),4) : RETURN
5470 '
5480 FOR LP1X=1 TO LOOPSX : IF LP1X=1 THEN S$="A" ELSE IF LP1X=2 THEN S$="B" ELSE IF LP1X=3 THEN S$="C"
5490 IF LP1X=4 THEN S$="D"
5500 '
5510 IF LOOPSX=2 THEN CX=52 ELSE IF LOOPSX=3 THEN CX=42 ELSE IF LOOPSX=4 THEN CX=32
5520 DX=CX+10*LP1X

```

APPENDIX F: DATA.BAS

```
5530  
5540 LOCATE 4,15 : PRINT USING "\          \";DDATE$  
5550 LOCATE 4,64 : PRINT TTIME$(TZ)  
5560 LOCATE 6,DZ+2 : PRINT S$  
5570 LOCATE 8,DZ-1 : PRINT USING "####.#";STEAM.PRES(LP1Z,TZ)  
5580 LOCATE 9,DZ : PRINT USING "#.###";FEEDWATER.FLOW(LP1Z,TZ)  
5590 LOCATE 10,DZ : PRINT USING "###.#";FEEDWATER.TEMP(LP1Z,TZ)  
5600 LOCATE 11,DZ : PRINT USING "###.#";TOP.BLOW(LP1Z,TZ)  
5610 LOCATE 12,DZ : PRINT USING "###.#";BOTTOM.BLOW(LP1Z,TZ)  
5620 LOCATE 13,DZ : PRINT USING "###.#";SG.WATER.LEVEL(LP1Z,TZ) : NEXT  
5630 LOCATE 17,33 : PRINT USING "###.#";LETDOWN.FLOW(TZ)  
5640 LOCATE 18,33 : PRINT USING "###.#";LETDOWN.TEMP(TZ)  
5650 LOCATE 22,32 : PRINT USING "####.#";PZR.PRES(TZ)  
5660 LOCATE 23,33 : PRINT USING "###.#";PZR.WATER.LEVEL(TZ)  
5670 LOCATE 17,72 : PRINT USING "###.#";CHARGING.FLOW(TZ)  
5680 LOCATE 18,72 : PRINT USING "###.#";CHARGING.TEMP(TZ)  
5690 LOCATE 22,72 : PRINT USING "###.#";T.AVE(TZ)  
5700 LOCATE 23,72 : PRINT USING "###.#";T.COLD(TZ) : RETURN  
5710 END
```

APPENDIX G

ANALYSIS.BAS

TABLE OF CONTENTS

| | Page |
|-------------------------------------|------|
| | ---- |
| MENU | G- 1 |
| CALCULATIONS | G- 2 |
| Delta Time | G- 2 |
| Steam Generators | G- 2 |
| Other Components | G- 4 |
| Power Dissipated | G- 5 |
| Reactor Power | G- 6 |
| STORAGE | G- 6 |
| RECALL | G- 7 |
| TABLES | G- 8 |
| Reduced Variables | G- 8 |
| Limits of Applicablity | G- 8 |
| Sub-region and Phase Identification | G- 9 |
| Sub-region 1 | G- 9 |
| Sub-region 2 | G-10 |
| Saturation | G-13 |

```

10 *
20 * *****
30 * ** ANALYSIS.BAS **
40 * *****
50 *
60 * *****
70 *
80 *
90 DEFDBL A-Z
100 *
110 REM Parameters
120 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
130 COMMON RISER.NUMBER,RISER.OD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,REFLECT.AREA
140 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
150 *
160 REM Data
170 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
180 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
190 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
200 *
210 REM Auxiliary Results
220 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT$
230 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
240 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X,Y,Z$
250 *
260 IF X$="ANALYSIS" THEN GOTO 640 ELSE IF X$="HEAT BALANCE" THEN GOSUB 3250 : CHAIN "REPORTS"
270 IF X$="MENU" THEN 310
280 *
290 *
300 * *****
310 * * MENU *
320 * *****
330 *
340 CLS : TX=10 : BX=16 : LZ=11 : RZ=70
350 LOCATE TX,LZ : PRINT CHR$(201) : LOCATE TX,RZ : PRINT CHR$(187) : LOCATE BX,RZ : PRINT CHR$(188)
360 LOCATE BX,LZ : PRINT CHR$(200)
370 LOCATE TX,LZ+1 : FOR IX=1 TO RZ-LZ-1 : PRINT CHR$(205) : NEXT
380 LOCATE BX,LZ+1 : FOR IX=1 TO RZ-LZ-1 : PRINT CHR$(205) : NEXT
390 FOR IX=TX+2 TO BX : LOCATE IX-1,LZ : PRINT CHR$(186) : NEXT
400 FOR IX=TX+2 TO BX : LOCATE IX-1,RZ : PRINT CHR$(186) : NEXT
410 *
420 LOCATE 25,21 : PRINT "After responding, strike the "; : COLOR 16,7
430 PRINT " *";CHR$(17)+CHR$(196)+CHR$(217);" *";
440 COLOR 7,0 : PRINT " key.";
450 *
460 LOCATE 10,34 : PRINT " ANALYSIS MENU " : LOCATE 12,21 : PRINT "(A) Analyze Data" : LOCATE 12,46
470 PRINT "(C) MASTER MENU"
480 LOCATE 14,21 : PRINT "(B) Display Results" : LOCATE 14,46 : PRINT "(D) Termination"
490 LOCATE 16,24 : PRINT " Which would you like (A/B/C/D)? " : LOCATE 16,57 : INPUT "",X$
500 *
510 IF X$="A" OR X$="a" THEN X$="ANALYSIS" : GOSUB 550 : CHAIN "PRM&DATA"
520 IF X$="B" OR X$="b" THEN GOSUB 590 : CHAIN "DISPLAY"
530 IF X$="C" OR X$="c" THEN X$="INTRO 1920" : GOSUB 580 : CHAIN "INTRO"
540 IF X$="D" OR X$="d" THEN CHAIN "END" ELSE 310

```

APPENDIX G: ANALYSIS.BAS

```

550 '
560 CLS : LOCATE 13,18 : PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY" : RETURN
570 LOCATE 25,20 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT " .";
580 CLS : LOCATE 13,20 : PRINT "LOADING INTRODUCTION MODULE INTO MEMORY" : RETURN
590 CLS : LOCATE 13,23 : PRINT "LOADING DISPLAY MODULE INTO MEMORY" : RETURN
600 '
610 '
620 '
630 '
640 '
650 '
660 '
670 CLS : LOCATE 8,36 : PRINT "COMPUTING" : IF DATA.SETSZ=1 THEN 870
680 '
690 '
700 '
710 '
720 FOR TZ=1 TO 2
730 '
740 HR(TZ)=INT(TTIME(TZ)/100) : MIN(TZ)=(TTIME(TZ)/100-HR(TZ))*100 : TIME.HR(TZ)=HR(TZ)+MIN(TZ)/60
750 '
760 IF DIAGNOSTIC.FLAG$(">")="S" THEN GOTO 790 ELSE LPRINT : LPRINT TAB(12);"TZ=";TZ;" TIME=";TTIME(TZ)
770 LPRINT TAB(12);"HR(TZ)=";HR(TZ);" MIN(TZ)=";MIN(TZ);" TIME.HR=";TIME.HR(TZ)
780 '
790 NEXT TZ
800 '
810 IF TIME.HR(2) < TIME.HR(1) THEN TIME.HR(2)=TIME.HR(2)+24
820 DELTA.TIME=TIME.HR(2)-TIME.HR(1)
830 '
840 IF DIAGNOSTIC.FLAG$(">")="S" THEN GOTO 870 ELSE LPRINT : LPRINT TAB(12);"DELTA.TIME=";DELTA.TIME
850 '
860 '
870 '
880 '
890 SG.CROSS.SECTION=3.1416*(SG.ID*SG.ID - RISER.NUMBER*RISER.OO*RISER.OO) / 4
900 '
910 FOR TZ = 1 TO DATA.SETSZ
920 '
930 IF DIAGNOSTIC.FLAG$(">")="S" THEN GOTO 950 ELSE LPRINT : LPRINT TAB(12);"TZ=";TZ
940 '
950 PRINT : PRINT TAB(33);"Data Set";TZ;"of";DATA.SETSZ : PRINT
960 '
970 FOR LZ = 1 TO LOOPSZ
980 '
990 IF LZ=1 THEN L$="A" ELSE IF LZ=2 THEN L$="B" ELSE IF LZ=3 THEN L$="C" ELSE IF LZ=4 THEN L$="D"
1000 '
1010 IF DIAGNOSTIC.FLAG$(">")="S" THEN GOTO 1030 ELSE LPRINT : LPRINT TAB(12);"LZ=";LZ
1020 '
1030 PRINT TAB(16); "Steam Generator ";L$;
1040 '
1050 '

```



```

1060 '                                     * Feedwater *
1070 '
1080 PRINT TAB(34);"Feedwater"; : Z$="FEEDWATER" : T=FEEDWATER.TEMP(LZ,TZ) : P=STEAM.PRES(LZ,TZ) : GOSUB 3720
1090 FEEDWATER.ENTHALPY(LZ,TZ)=H
1100 FEEDWATER.POWER(LZ,TZ)=H*(-FEEDWATER.FLOW(LZ,TZ))/1000!
1110 '
1120 '
1130 '                                     * Surface Blowdown *
1140 '
1150 PRINT TAB(34);"Surface Blowdown"; : Y$="TSAT" : Z$="TOP BLOW" : P=STEAM.PRES(LZ,TZ) : GOSUB 3720
1160 TOP.BLOW.ENTHALPY(LZ,TZ)=H : TOP.BLOW.DENSITY(LZ,TZ)=1/V
1170 '
1180 IF DIAGNOSTIC.FLAG$("<>")="S" THEN GOTO 1210 ELSE LPRINT TAB(12);"V=";V;
1190 LPRINT " TOP.BLOW.DENSITY(LZ,TZ)=";TOP.BLOW.DENSITY(LZ,TZ)
1200 '
1210 X=TOP.BLOW(LZ,TZ)/7.4805195#*60*TOP.BLOW.DENSITY(LZ,TZ)/1000000!
1220 TOP.BLOW.LB.PER.HR(LZ,TZ)=X : REM E6 lb/hr
1230 TOP.BLOW.POWER(LZ,TZ)=TOP.BLOW.ENTHALPY(LZ,TZ)*TOP.BLOW.LB.PER.HR(LZ,TZ)/1000 : REM E9 BTUs/hr
1240 '
1250 '
1260 '                                     * Bottom Blowdown *
1270 '
1280 PRINT TAB(34);"Bottom Blowdown"; : Z$="BOTTOM BLOW" : T=(T+FEEDWATER.TEMP(LZ,TZ))/2 : GOSUB 3720
1290 BOTTOM.BLOW.ENTHALPY(LZ,TZ)=H : BOTTOM.BLOW.DENSITY(LZ,TZ)=1/V
1300 '
1310 IF DIAGNOSTIC.FLAG$("<>")="S" THEN GOTO 1340 ELSE LPRINT TAB(12);"V=";V;
1320 LPRINT " BOTTOM.BLOW.DENSITY(LZ,TZ)=";BOTTOM.BLOW.DENSITY(LZ,TZ)
1330 '
1340 BOTTOM.BLOW.LB.PER.HR(LZ,TZ)=BOTTOM.BLOW(LZ,TZ)/7.4805195#*60*BOTTOM.BLOW.DENSITY(LZ,TZ)/1000000!
1350 : REM E6 lb/hr
1360 BOTTOM.BLOW.POWER(LZ,TZ)=BOTTOM.BLOW.ENTHALPY(LZ,TZ)*BOTTOM.BLOW.LB.PER.HR(LZ,TZ)/1000!
1370 : REM E9 BTUs/hr
1380 '
1390 '
1400 '                                     * Steam *
1410 '
1420 PRINT TAB(34); "Steam"; : Z$="STEAM" : T=TSAT : GOSUB 3720 : DRY.STEAM.ENTHALPY(LZ,TZ)=H
1430 '
1440 IF DIAGNOSTIC.FLAG$("<>")="S" THEN GOTO 1470 ELSE LPRINT : LPRINT
1450 LPRINT TAB(12);"DRY.STEAM.ENTHALPY(";LZ;" ";TZ;" ) =" ; DRY.STEAM.ENTHALPY(LZ,TZ)
1460 '
1470 PRINT TAB(34); "Moisture"; : Z$="MOISTURE" : GOSUB 3720 : MOISTURE.ENTHALPY (LZ,TZ) = H
1480 '
1490 IF DIAGNOSTIC.FLAG$("<>")="S" THEN GOTO 1520 : LPRINT : LPRINT
1500 LPRINT TAB(12);"MOISTURE.ENTHALPY(";LZ;" ";TZ;" ) =" ; : LPRINT MOISTURE.ENTHALPY(LZ,TZ)
1510 '
1520 NEXT LZ,TZ : Z$=""
1530 '
1540 IF DRY4.SETS%=1 THEN 1680
1550 '
1560 FOR LZ=1 TO LOOPS%
1570 '
1580 X=SG.WATER.LEVEL(LZ,2)*BOTTOM.BLOW.DENSITY(LZ,2)-SG.WATER.LEVEL(LZ,1)*BOTTOM.BLOW.DENSITY(LZ,1)
1590 EXCESS.FEEDWATER.FLOW(LZ)=SG.CROSS.SECTION/1728 *X/DELTA.TIME/1000000! : REM E6 lb/hr

```

APPENDIX G: ANALYSIS.BAS

```

1600 '
1610 IF DIAGNOSTIC.FLAG*(<)*S" THEN GOTO 1660 : LPRINT : LPRINT TAB(12);"LZ=";LZ
1620 LPRINT TAB(12);"EXCESS.FEEDWATER.FLOW(LZ)=";EXCESS.FEEDWATER.FLOW(LZ)
1630 LPRINT TAB(12);"BOTTOM.BLOW.DENSITY(LZ,2)=";BOTTOM.BLOW.DENSITY(LZ,2)
1640 LPRINT TAB(12);"BOTTOM.BLOW.DENSITY(LZ,1)=";BOTTOM.BLOW.DENSITY(LZ,1)
1650 '
1660 NEXT LZ
1670 '
1680 FOR TZ=1 TO DATA.SETSZ : FOR LZ=1 TO LOOPSZ
1690 IF LZ=1 THEN CARRYOVER=CARRYOVER.A ELSE IF LZ=2 THEN CARRYOVER=CARRYOVER.B
1700 IF LZ=3 THEN CARRYOVER=CARRYOVER.C ELSE IF LZ=4 THEN CARRYOVER=CARRYOVER.D
1710 X=FEEDWATER.FLOW(LZ,TZ)-EXCESS.FEEDWATER.FLOW(LZ)
1720 STEAM.FLOW(LZ,TZ)=X-TOP.BLOW.LB.PER.HR(LZ,TZ)-BOTTOM.BLOW.LB.PER.HR(LZ,TZ) : REM E6 lb/hr
1730 STEAM.ENTHALPY(LZ,TZ)=DRY.STEAM.ENTHALPY(LZ,TZ)*(1-CARRYOVER/100)+MOISTURE.ENTHALPY(LZ,TZ)*CARRYOVER/100
1740 STEAM.POWER(LZ,TZ)=STEAM.ENTHALPY(LZ,TZ)*STEAM.FLOW(LZ,TZ)/1000 : REM E9 BTUs/hr
1750 '
1760 NEXT LZ,TZ
1770 '
1780 '
1790 ' ** OTHER COMPONENTS **
1800 '
1810 TZ=1
1820 PRINT : PRINT TAB(33);"Data Set";TZ;"of";DATA.SETSZ : PRINT
1830 '
1840 IF DIAGNOSTIC.FLAG*(<)*S" THEN GOTO 1860 ELSE LPRINT : LPRINT TAB(12);"TZ=";TZ:
1850 '
1860 PRINT TAB(16);"Other Components";
1870 '
1880 '
1890 ' * Letdown Line *
1900 '
1910 PRINT TAB(34);"Letdown Line"; : Z$="LETDOWN" : T=LETDOWN.TEMP(TZ) : P=PZR.PRES(TZ) : GOSUB 3720
1920 LETDOWN.ENTHALPY(TZ)=H
1930 LETDOWN.FLOW.LB.PER.HR(TZ)=LETDOWN.FLOW(TZ)/7.4805195*60/V/1000000! : REM E6 lb/hr
1940 LETDOWN.POWER(TZ)=H*LETDOWN.FLOW.LB.PER.HR(TZ)/1000! : REM E9 BTUs/hr
1950 '
1960 '
1970 ' * Charging Line *
1980 '
1990 PRINT TAB(34);"Charging Line"; : Z$="CHARGING" : T=CHARGING.TEMP(TZ) : P=PZR.PRES(TZ) : GOSUB 3720
2000 CHARGING.ENTHALPY(TZ)=H
2010 CHARGING.FLOW.LB.PER.HR(TZ)=-CHARGING.FLOW(TZ)/7.4805195*60/V/1000000! : REM E6 lb/hr
2020 CHARGING.POWER(TZ)=H*CHARGING.FLOW.LB.PER.HR(TZ)/1000! : REM E9 BTUs/hr
2030 '
2040 '
2050 ' * Insulation Losses *
2060 '
2070 IF INSULATION.THICKNESS = 0 THEN GOTO 2140
2080 '
2090 X=LOSS.COEFFICIENT*REFLECT.AREA
2100 INSULATION.POWER(TZ)=(X+CONDUCTIVITY*NONREFLECT.AREA*(T.AVE(TZ)-100)/INSULATION.THICKNESS*12)/1E+09
2110 '
2120 GOTO 2170
2130 '

```

```

2140 INSULATION.POWER(TZ)=(LOSS.COEFFICIEN)*REFLECT.AREA/1E+09
2150 '
2160 '
2170 '           * Pressurizer Level Change *
2180 '
2190 PRINT TAB(34);"Pressurizer"; : IF PZR.WATER.LEVEL(1)>PZR.WATER.LEVEL(2) THEN GOTO 2200 ELSE GOTO 2210
2200 Y$="TSAT" : Z$="PZR" : P=PZR.PRES(TZ) : GOSUB 3720 : GOTO 2230
2210 T.HOT(TZ)=2*T.AVE(TZ)-T.COLD(TZ)
2220 Z$="HOT LEG" : P=PZR.PRES(TZ) : T=T.HOT(TZ) : GOSUB 3720
2230 PZR.ENTHALPY(TZ)=H : PZR.DENSITY(TZ)=1/V
2240 '
2250 IF DIAGNOSTIC.FLAG$("<"S" THEN GOTO 2290 ELSE LPRINT TAB(12);"V=";V;TAB(46);"1/V =" ;1/V
2260 LPRINT TAB(12);"PZR.ENTHALPY(TZ)=";PZR.ENTHALPY(TZ); : LPRINT" PZR.DENSITY(TZ)=";PZR.DENSITY(TZ)
2270 LPRINT TAB(12);"PZR.DENSITY(1)=";PZR.DENSITY(1);" PZR.DENSITY(2) =" ;PZR.DENSITY(2)
2280 '
2290 IF TZ=1 AND DATA.SETSZ=2 THEN TZ=2 : GOTO 1820
2300 '
2310 IF DATA.SETSZ=1 THEN 2500
2320 '
2330 PZR.CROSS.SECTION=3.1416*PZR.ID*PZR.ID/4
2340 PZR.FLOW=PZR.CROSS.SECTION/1728*(PZR.WATER.LEVEL(2)*PZR.DENSITY(2)-
PZR.WATER.LEVEL(1)*PZR.DENSITY(1))/DELTA.TIME/1000000! : REM E6 lb/hr
2350 PZR.POWER=PZR.FLOW*(PZR.ENTHALPY(1)+PZR.ENTHALPY(2))/2/1000! : REM E9 BTUs/hr
2360 Z$=""
2370 '
2380 IF DIAENOSTIC.FLAG$("<"S" THEN GOTO 2500
2390 LPRINT:LPRINT TAB(12);"PZR.WATER.LEVEL(1)=";PZR.WATER.LEVEL(1);" PZR.WATER.LEVEL(2)=";PZR.WATER.LEVEL(2)
2400 LPRINT TAB(12);"PZR.ENTHALPY(1)=";PZR.ENTHALPY(1);" PZR.ENTHALPY(2)=";PZR.ENTHALPY(2)
2410 LPRINT TAB(12);"PZR.DENSITY(1)=";PZR.DENSITY(1);" PZR.DENSITY(2)=";PZR.DENSITY(2)
2420 LPRINT TAB(12);"PZR.WATER.LEVEL(2)*PZR.DENSITY(2)-PZR.WATER.LEVEL(1)*PZR.DENSITY(1)=";
2430 PRINT PZR.WATER.LEVEL(2)*PZR.DENSITY(2)-PZR.WATER.LEVEL(1)*PZR.DENSITY(1)
2440 LPRINT TAB(12);"PZR.FLOW*(PZR.ENTHALPY(1)+PZR.ENTHALPY(2))/2/1E3=";
2450 PRINT PZR.FLOW*(PZR.ENTHALPY(1)+PZR.ENTHALPY(2))/2/1000!
2460 '
2470 CLS : LOCATE 13,20 : PRINT "COMPLETING ANALYSIS AND STORING RESULTS"
2480 '
2490 '
2500 '           * Pumps *
2510 '
2520 RC.PUMP.POWER.BTU.PER.HR=-LOOPSZ*RC.PUMP.EFF/100*RC.PUMP.PWR/292.875 : REM E9 BTUs/hr
2530 '
2540 '
2550 '           ** POWER DISSIPATED **
2560 '
2570 '           * Steam Generators *
2580 '
2590 FOR TZ=1 TO DATA.SETSZ : FOR LZ=1 TO LOOPSZ
2600 SG.POWER(LZ,TZ)=STEAM.POWER(LZ,TZ)+TOP.BLOW.POWER(LZ,TZ)+BOTTOM.BLOW.POWER(LZ,TZ)+FEEDWATER.POWER(LZ,TZ)
2610 NEXT LZ
2620 '
2630 '           * Other Components *
2640 '
2650 OTHER.POWER(TZ)=LETDOWN.POWER(TZ)+CHARGING.POWER(TZ)+PZR.POWER +INSULATION.POWER(TZ)+RC.PUMP.POWER.BTU.PER.HR
2660 OTHER.POWER.MW(TZ)=292.875*OTHER.POWER(TZ)

```

APPENDIX G: ANALYSIS.BAS

```

2670 '
2680 NEXT TZ
2690 '
2700 CLS : LOCATE 13,20 : PRINT "COMPLETING ANALYSIS AND STORING RESULTS"
2710 '
2720 '
2730 '                ** REACTOR POWER **
2740 '
2750 FOR TZ=1 TO DATA.SETSZ : TOTAL.SG.POWER(TZ)=0
2760 FOR LZ=1 TO LOOPSZ : SG.POWER.MW(LZ,TZ)=292.875*SG.POWER(LZ,TZ)
2770 TOTAL.SG.POWER(TZ)=TOTAL.SG.POWER(TZ)+SG.POWER.MW(LZ,TZ) : NEXT LZ
2780 REACTOR.POWER.MW(TZ)=TOTAL.SG.POWER(TZ)+OTHER.POWER.MW(TZ) : NEXT TZ
2790 '
2800 '
2810 '                *****
2820 '                * STORAGE *
2830 '                *****
2840 '
2850 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=41 : FIELD #1,1 AS RF$,8 AS PH1$,8 AS PH2$,8 AS PF$,8 AS PP$,8 AS RPB$
2860 RESULTS.FLAG$="S"
2870 LSET RF$ = RESULTS.FLAG$ : LSET PH1$ = MKD$(PZR.ENTHALPY(1)) : LSET PH2$ = MKD$(PZR.ENTHALPY(2))
2880 LSET PF$ = MKD$(PZR.FLOW) : LSET PP$ = MKD$(PZR.POWER) : LSET RPB$=MKD$(RC.PUMP.POWER.BTU.PER.HR)
2890 PUT #1,1 : CLOSE #1
2900 '
2910 OPEN "SG-CALC."+DOCKET$ AS #2 LEN=104
2920 FIELD #2,8 AS SH$,8 AS SF$,8 AS SP$,8 AS FH$,8 AS FF$,8 AS FP$,8 AS TBH$,8 AS TBF$,8 AS TBP$,8 AS BBH$,
      8 AS BBF$,8 AS BBP$,8 AS SGP$
2930 '
2940 FOR TZ=1 TO DATA.SETSZ : FOR LPZ=1 TO LOOPSZ
2950 '
2960 LSET TBH$ = MKD$(TOP.BLOW.ENTHALPY(LPZ,TZ)) : LSET SH$ = MKD$(STEAM.ENTHALPY(LPZ,TZ))
2970 LSET TBF$ = MKD$(TOP.BLOW.LB.PER.HR(LPZ,TZ)) : LSET SF$ = MKD$(STEAM.FLOW(LPZ,TZ))
2980 LSET TBP$ = MKD$(TOP.BLOW.POWER(LPZ,TZ)) : LSET SP$ = MKD$(STEAM.POWER(LPZ,TZ))
2990 LSET BBH$ = MKD$(BOTTOM.BLOW.ENTHALPY(LPZ,TZ)) : LSET FH$ = MKD$(FEEDWATER.ENTHALPY(LPZ,TZ))
3000 LSET BBF$ = MKD$(BOTTOM.BLOW.LB.PER.HR(LPZ,TZ)) : LSET FF$ = MKD$(FEEDWATER.FLOW(LPZ,TZ))
3010 LSET BBP$ = MKD$(BOTTOM.BLOW.POWER(LPZ,TZ)) : LSET FP$ = MKD$(FEEDWATER.POWER(LPZ,TZ))
3020 LSET SGP$ = MKD$(SG.POWER(LPZ,TZ))
3030 '
3040 PUT #2, (TZ-1) * LOOPSZ + LPZ : NEXT LPZ,TZ : CLOSE #2
3050 '
3060 OPEN "OC-CALC."+DOCKET$ AS #3 LEN=88
3070 FIELD #3,8 AS LH$,8 AS LF$,8 AS LP$,8 AS CH$,8 AS CF$,8 AS CP$,8 AS IP$,8 AS OP$,8 AS OM$,8 AS RXP$,8 AS T$
3080 '
3090 FOR TZ = 1 TO DATA.SETSZ
3100 '
3110 LSET LH$ = MKD$(LETDOWN.ENTHALPY(TZ)) : LSET IP$ = MKD$(INSULATION.POWER(TZ))
3120 LSET LF$ = MKD$(LETDOWN.FLOW.LB.PER.HR(TZ)) : LSET OP$ = MKD$(OTHER.POWER(TZ))
3130 LSET LP$ = MKD$(LETDOWN.POWER(TZ)) : LSET OM$ = MKD$(OTHER.POWER.MW(TZ))
3140 LSET CH$ = MKD$(CHARGING.ENTHALPY(TZ)) : LSET RXP$ = MKD$(REACTOR.POWER.MW(TZ))
3150 LSET CF$ = MKD$(CHARGING.FLOW.LB.PER.HR(TZ)) : LSET T$ = MKD$(TTIME(TZ))
3160 LSET CP$ = MKD$(CHARGING.POWER(TZ))
3170 '
3180 PUT #3, TZ : NEXT : CLOSE #3 : GOTO 310
3190 '

```

```

3200 PRINT : PRINT : PRINT : PRINT TAB(23);"CHECK DIRECTORY FOR RESULT.";DOCKET$;" FILE" : PRINT : PRINT
3210 PRINT : STOP
3220 '
3230 '
3240 '
3250 '          *****
3260 '          * RECALL *
3270 '          *****
3280 '
3290 OPEN DOCKET$ AS #1 LEN=159
3300 FIELD #1,32 AS P$,1 AS U$,3 AS D$,2 AS L$,121 AS X$
3310 GET #1,1 : LOOPSZ = CVI(L$) : CLOSE #1
3320 '
3330 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=41 : FIELD #1,1 AS RF$,8 AS PH1$,8 AS PH2$,8 AS PF$,8 AS PP$,8 AS RPB$
3340 GET #1,1 : PZR.ENTHALPY(1) = CVD(PH1$) : PZR.ENTHALPY(2) = CVD(PH2$) : PZR.FLOW = CVD(PF$)
3350 PZR.POWER = CVD(PP$) : RC.PUMP.POWER.BTU.PER.HR = CVD(RPB$) : RESULTS.FLAG$ = RF$ : CLOSE #1
3360 '
3370 IF (RESULTS.FLAG$<>"S" AND RESULTS.FLAG$<>"C") THEN GOTO 3670
3380 '
3390 OPEN "SG-CALC."+DOCKET$ AS #2 LEN=104
3400 FIELD #2,8 AS SH$,8 AS SF$,8 AS SP$,8 AS FH$,8 AS FF$,8 AS FP$,8 AS TBH$,8 AS TBF$,8 AS TBP$,8 AS SBH$,
      8 AS BBF$,8 AS BBP$,8 AS SGP$
3410 '
3420 TZ=1 : LZ=1 : FOR TZ=1 TO DATA.SETSZ : FOR LPZ=1 TO LOOPSZ : GET #2, (TZ-1) * LOOPSZ + LZ
3430 TOP.BLOW.ENTHALPY(LPZ,TZ) = CVD (TBH$) : STEAM.ENTHALPY(LPZ,TZ) = CVD (SH$)
3440 TOP.BLOW.LB.PER.HR(LPZ,TZ) = CVD (TBF$) : STEAM.FLOW(LPZ,TZ) = CVD (SF$)
3450 TOP.BLOW.POWER(LPZ,TZ) = CVD (TBP$) : STEAM.POWER(LPZ,TZ) = CVD (SP$)
3460 BOTTOM.BLOW.ENTHALPY(LPZ,TZ) = CVD (BBH$) : FEEDWATER.ENTHALPY(LPZ,TZ) = CVD (FH$)
3470 BOTTOM.BLOW.LB.PER.HR(LPZ,TZ) = CVD (BBF$) : FEEDWATER.FLOW(LPZ,TZ) = CVD (FF$)
3480 BOTTOM.BLOW.POWER(LPZ,TZ) = CVD (BBP$) : FEEDWATER.POWER(LPZ,TZ) = CVD (FP$)
3490 SG.POWER(LPZ,TZ) = CVD (SGP$)
3500 '
3510 NEXT LPZ,TZ : CLOSE #2
3520 '
3530 OPEN "OC-CALC."+DOCKET$ AS #3 LEN=88
3540 FIELD #3,8 AS LH$,8 AS LF$,8 AS LP$,8 AS CH$,8 AS CF$,8 AS CP$,8 AS IP$,8 AS OP$,8 AS OM$,8 AS RXP$,8 AS T$
3550 '
3560 FOR TZ = 1 TO DATA.SETSZ : GET #3, TZ
3570 '
3580 LETDOWN.ENTHALPY(TZ) = CVD (LH$) : INSULATION.POWER(TZ) = CVD (IP$)
3590 LETDOWN.FLOW.LB.PER.HR(TZ) = CVD (LF$) : OTHER.POWER(TZ) = CVD (OP$)
3600 LETDOWN.POWER(TZ) = CVD (LP$) : OTHER.POWER.MW(TZ) = CVD (OM$)
3610 CHARGING.ENTHALPY(TZ) = CVD (CH$) : REACTOR.POWER.MW(TZ) = CVD (RXP$)
3620 CHARGING.FLOW.LB.PER.HR(TZ) = CVD (CF$) : TTIME(TZ) = CVD (T$)
3630 CHARGING.POWER(TZ) = CVD (CP$)
3640 '
3650 NEXT : CLOSE #3 : RETURN
3660 '
3670 PRINT : PRINT : PRINT : PRINT TAB(23);"CHECK DIRECTORY FOR RCS-CALC.";DOCKET$;" FILE" : PRINT : PRINT
3680 PRINT : STOP
3690 '
3700 '

```

APPENDIX 6: ANALYSIS.BAS

```

3710 ' *****
3720 ' * TABLES *
3730 ' *****
3740 '
3750 IF Y$="TSAT" THEN GOTO 4580
3760 '
3770 '
3780 ' ** REDUCED VARIABLES **
3790 '
3800 THETA = (T+459.67)/1165.14 : BETA = P/3208.2348#
3810 '
3820 IF DIAGNOSTIC.FLAG$("<"S" THEN GOTO 3850
3830 LPRINT:LPRINT TAB(12);"T=";T;" P=";P;" (ANALYSIS, LINE 6733)"
3840 '
3850 IF Y$="TSAT" THEN Y$="" : GOTO 4730
3860 '
3870 '
3880 ' ** LIMITS OF APPLICABILITY **
3890 '
3900 L0 = 15.74373327# : L1=-34.17061978# : L2 = 19.31380707#
3910 THETA.1 = .9626911787# : THETA.2 = 1.333462073# : THETA.3 = 1.657886606# : THETA.T = .4219960731#
3920 BETA.2 = 4.52079566# : BETA.L = L0+L1*THETA+L2*THETA^2 : BETA.L.PRIME = L1+2*L2*THETA
3930 '
3940 IF Z$="MOISTURE" THEN GOTO 4730 ELSE IF Z$="STEAM" THEN GOTO 5170
3950 '
3960 IF THETA < THETA.T THEN 3970 ELSE 3990
3970 PRINT TAB(18);"Temperature is out of range. It is below"
3980 PRINT TAB(18);"the triple point (32.018 F)." : GOTO 4510
3990 '
4000 IF BETA <= 0 THEN 4010 ELSE 4030
4010 PRINT TAB(18);"Absolute pressure is zero or negative."
4020 PRINT TAB(18);"Negative absolute pressure is impossible." : GOTO 4510
4030 '
4040 IF THETA > THETA.3 THEN 4050 ELSE 4070
4050 PRINT TAB(18);"Temperature is out of range. The upper limit is"
4060 PRINT TAB(18);"1472 degrees F." : GOTO 4510
4070 '
4080 IF THETA >= THETA.T AND THETA <= .6 AND BETA > 18.2*THETA-6.4 THEN 4090 ELSE 4150
4090 PRINT TAB(18);"Pressure is out of range. For temperature"
4100 PRINT TAB(18);"equal to "; : PRINT USING "####.#";T;
4110 PRINT " degrees F,"; : PRINT " the limit on pressure";
4120 PRINT TAB(18);"is ";
4130 PRINT USING "####.#";3208.235*(18.2*THETA-6.4);
4140 PRINT " psia." : GOTO 4510
4150 '
4160 IF THETA > .6 AND THETA <= THETA.1 AND BETA > BETA.2 THEN 4170 ELSE 4200
4170 PRINT TAB(18);"Pressure is out of range. For temperature in"
4180 PRINT TAB(18);"the range from 239.4 to 662.0 degrees F, the"
4190 PRINT TAB(18);"upper limit is 14,503 psia." : GOTO 4510
4200 '
4210 IF THETA > THETA.1 AND THETA < THETA.2 AND BETA > BETA.L THEN 4220 ELSE 4280
4220 PRINT TAB(18);"Pressure is out of range. For temperature"
4230 PRINT TAB(18);"equal to "; : PRINT USING "####.#";T;
4240 PRINT " degrees F,"; : PRINT " the limit on pressure";

```

```

4250 PRINT TAB(18);"is ";
4260 PRINT USING "#####.0";221.2*14.5037738#*BETA.L;
4270 PRINT " psia." : GOTO 4510
4280 '
4290 IF THETA >= THETA.2 AND THETA <= THETA.3 AND BETA > BETA.2 THEN 4300 ELSE 4560
4300 PRINT TAB(18);"Pressure is out of range. For temperature in"
4310 PRINT TAB(18);"the range from 1094 to 1472 degrees F, the"
4320 PRINT TAB(18);"upper limit is 14,503 psia." : GOTO 4510
4330 '
4340 IF T > 705.47 THEN 4350 ELSE 4380
4350 PRINT TAB(18);"Temperature is out of range. It is above the"
4360 PRINT TAB(18);"critical point (705.47 degrees F). Pressure"
4370 PRINT TAB(18);"will not condense supercritical steam." : GOTO 4510
4380 '
4390 IF T < 32.018 THEN 4400 ELSE 4580
4400 PRINT TAB(18);"Temperature is out of range. It is below"
4410 PRINT TAB(18);"the triple point (32.018 F)." : GOTO 4510
4420 '
4430 IF P > 3208.2 THEN 4440 ELSE 4460
4440 PRINT TAB(18);"Pressure is out of range. It is above the"
4450 PRINT TAB(18);"critical point (3208.2 psia)." : GOTO 4510
4460 '
4470 IF P < 8.864999E-02 THEN 4480 ELSE 4580
4480 PRINT TAB(18);"Pressure is out of range. It is below"
4490 PRINT TAB(18);"the triple point (0.08865 psia)." : GOTO 4510
4500 '
4510 LOCATE 25,1 : PRINT SPC(79); : LOCATE 25,22,0 : PRINT " To continue, press the ";
4520 COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".*";
4530 Z$=INKEY$ : IF Z$(<>CHR$(32)) THEN 4530 ELSE GOTO 310
4540 '
4550 '
4560 ** SUB-REGION AND PHASE IDENTIFICATION **
4570 '
4580 SK1#=-7.691234564# : SK2#=-26.08023696# : SK3#=-168.1706546# : SK4#= 64.23285504# : SK5#=-118.9646225#
4590 SK6#= 4.16711732# : SK7#= 20.9750676# : SK8#= 1E+09 : SK9#= 6
4600 '
4610 IF THETA > 1 THEN GOTO 4680 ELSE IF Y$="TSAT" THEN GOSUB 6680 : T=TSAT : GOTO 3790
4620 '
4630 C = 1 - THETA : SIGMA = SK1#*C+SK2#*C^2+SK3#*C^3+SK4#*C^4+SK5#*C^5
4640 BETA.K = EXP(SIGMA/(THETA*(1+SK6#*C+SK7#*C^2)))-C/(SK8#*C^2+SK9#)
4650 '
4660 IF THETA >= THETA.T AND THETA <= THETA.1 AND BETA > BETA.K THEN REGION$="1"
4670 IF THETA >= THETA.T AND THETA <= THETA.1 AND BETA < BETA.K THEN REGION$="2"
4680 IF THETA >= THETA.1 AND BETA <= BETA.L THEN REGION$="2"
4690 '
4700 IF REGION$="1" THEN GOTO 4730 ELSE IF REGION$="2" THEN GOTO 5170
4710 '
4720 '
4730 ** SUB-REGION 1 **
4740 '
4750 * Constants *
4760 '
4770 SA1 =.8438375405# : SA2 =.0005362162162# : SA3 =1.72 : SA4 =.07342278489# : SA5 =.0497585887#
4780 SA6 =.65371543# : SA7 =1.15E-06 : SA8 =1.5108E-05 : SA9 =.14188 : SA10=7.002753165#

```

APPENDIX G: ANALYSIS.BAS

```

4790 SA11=.0002995284926# : SA12=.204
4800 '
4810 A0 = 6824.687741# : A1 = -542.2063673# : A2 = -20966.66205# : A3 = 39412.86787# : A4 = -67332.77739#
4820 A5 = 99023.81028# : A6 = -109391.1774# : A7 = 85908.41667# : A8 = -45111.68742# : A9 = 14181.38926#
4830 A10 = -2017.271113# : A11 = 7.982692717# : A12 = -.02616571843# : A13 = .00152241179# : A14 = .02284279054#
4840 A15 = 242.1647003# : A16 = 1.269716088D-10 : A17 = .0000002074838328# : A18 = .0000000271402035#
4850 A19 = 1.105710498D-09 : A20 = 12.93441934# : A21 = .00001308119072# : A22 = 6.047626338D-14
4860 '
4870 IF X$="ANALYSIS" THEN GOTO 4900
4880 '
4890 '
4900 ' * Enthalpy *
4910 '
4920 Y=1-SA1*THETA^2-SA2*THETA^(-6)
4930 Y.PRIME=-2*SA1*THETA+6*SA2*THETA^(-7)
4940 Z=Y+(SA3*Y^2-2*SA4*THETA+2*SA5*BETA)^.5 : ZE=-5/17
4950 '
4960 CE1=A0*THETA+A1-A3*THETA^2-2*A4*THETA^3-3*A5*THETA^4-4*A6*THETA^5-5*A7*THETA^6-6*A8*THETA^7-7*A9*THETA^8
-B+A10*THETA^9
4970 CE2=A11*(Z*(17*(Z/29-Y/12)+5*THETA*Y.PRIME/12)+SA4*THETA -(SA3-1)*THETA*Y*Y.PRIME)*Z^2E
4980 CE3=(A12-A14*THETA^2+A15*(9*THETA+SA6)*SGN(SA6-THETA)*(ABS(SA6-THETA))^9
+A16*(20*THETA^19+SA7)*(SA7+THETA^19)^(-2))*BETA
4990 CE4=-(12*THETA^11+SA8)*(SA8+THETA^11)^(-2)*(A17*BETA+A18*BETA^2 +A19*BETA^3)
5000 CE5=A20*THETA^18*(17*SA9+19*THETA^2)*((SA10+BETA)^(-3)+SA11*BETA)
5010 CE6=+A21*SA12*BETA^3 : CE7=+21*A22*THETA^(-20)*BETA^4
5020 '
5030 EPSILON.1=CE1+CE2+CE3+CE4+CE5+CE6+CE7 : H=2.212E+07*.00317*EPSILON.1/2326 : PRINT TAB(51);"Enthalpy";
5040 IF Z$="MOISTURE" OR Z$="FEEDWATER" THEN PRINT : RETURN
5050 '
5060 '
5070 ' * Specific Volume *
5080 '
5090 CV1=A11*SA5*Z^2E+(A12+A13*THETA+A14*THETA^2 +A15*(ABS(SA6-THETA))^10+A16/(SA7+THETA^19))
5100 CV2=- (SA8+THETA^11)^(-1)*(A17+2*A18*BETA+3*A19*BETA^2)
5110 CV3=-A20*THETA^18*(SA9+THETA^2)*(-3*(SA10+BETA)^(-4)+SA11)+3*A21*(SA12-THETA)*BETA^2+4*A22*THETA^(-20)*BETA^3
5120 '
5130 CHI.1=CV1+CV2+CV3 : V=.00317*.45359237#*CHI.1/.3048^3 : PRINT TAB(60);"Density"
5140 IF Z$="TOP BLOW" OR Z$="BOTTOM BLOW" OR Z$="LETDOWN" OR Z$="CHARGING" OR Z$="PZR" OR Z$="HOT LEG" THEN RETURN
5150 '
5160 '
5170 ' ** SUB-REGION 2 **
5180 '
5190 ' * Constants *
5200 '
5210 ALPHA.0 = 0
5220 '
5230 B0 = 16.83599274# : B(0,1) = 28.56067796# : B(0,2) = -54.38923329# : B(0,3) = .4330662834#
5240 B(0,4) = -.6547711697# : B(0,5) = .08565182058# : B(1,1) = .06670375918# : B(1,2) = 1.388983801#
5250 B(2,1) = .08390104328# : B(2,2) = .02614670893# : B(2,3) = -.03373439453# : B(3,1) = .4520918904#
5260 B(3,2) = .1069036614# : B(4,1) = -.5975336707# : B(4,2) = -.08847535804# : B(5,1) = .5958051609#
5270 B(5,2) = -.5159303373# : B(5,3) = .2075021122# : B(6,1) = .1190610271# : B(6,2) = -.09867174132#
5280 B(7,1) = .1683998803# : B(7,2) = -.05809438001# : B(8,1) = .006552390126# : B(8,2) = .0005710218649#
5290 B(9,0) = 193.6587558# : B(9,1) = -1388.522425# : B(9,2) = 4126.607219# : B(9,3) = -6508.211677#
5300 B(9,4) = 5745.984054# : B(9,5) = -2693.088365# : B(9,6) = 523.5718623#

```



```

5310 '
5320 I1      = 4.260321148#
5330 '
5340 SB      = .763333333# : SB(6,1)= .4006073948# : SB(7,1)= .08636081627# : SB(8,1)=-.8532322921#
5350 SB(8,2)= .3460208861#
5360 '
5370 XZ(6,1)=14 : XZ(7,1)=19 : XZ(8,1)=54 : XZ(8,2)=27
5380 '
5390 ZX(1,1)=13 : ZX(1,2)=3 : ZX(2,1)=18 : ZX(2,2)=2 : ZX(2,3)=1 : ZX(3,1)=18 : ZX(3,2)=10 : ZX(4,1)=25
5400 ZX(4,2)=14 : ZX(5,1)=32 : ZX(5,2)=28 : ZX(5,3)=24 : ZX(6,1)=12 : ZX(6,2)=11 : ZX(7,1)=24 : ZX(7,2)=18
5410 ZX(8,1)=24 : ZX(8,2)=14
5420 '
5430 '
5440 '                * Enthalpy *
5450 '
5460 CE3=0 : CE4=0 : CE5=0 : DE4(1)=0 : DE4(2)=0 : DE4(3)=0 : DE4(4)=0 : DE4(5)=0 : DE6=0 : CE6=0
5470 '
5480 X = EXP(SB*(1-THETA)) : CE1=ALPHA.0 : CE2=B0*THETA
5490 '
5500 FOR MUZ = 1 TO 5
5510 CE3=CE3+B(0,MUZ)*(MUZ-2)*THETA^(MUZ-1)
5520 NEXT MUZ
5530 '
5540 FOR MUZ = 1 TO 5
5550 IF MUZ=1 THEN NTZ=2 ELSE IF MUZ=2 THEN NTZ=3 ELSE IF MUZ=3 THEN NTZ=2 ELSE IF MUZ=4 THEN NTZ=2
5560 IF MUZ=5 THEN NTZ=3
5570 FOR MUZ = 1 TO NTZ
5580 DE4(MUZ)=DE4(MUZ)+B(MUZ,MUZ)*(1+ZX(MUZ,MUZ)*SB*THETA)*X^ZX(MUZ,MUZ)
5590 IF DIAGNOSTIC.FLAG#("<"S" THEN GOTO 5620
5600 LPRINT TAB(12);"DE4(MUZ)=";DE4(MUZ);" B(MUZ,MUZ)=";B(MUZ,MUZ)
5610 LPRINT TAB(12);"(1+ZX(MUZ,MUZ)*SB*THETA)=";(1+ZX(MUZ,MUZ)*SB*THETA);" X^ZX(MUZ,MUZ)=",X^ZX(MUZ,MUZ)
5620 NEXT MUZ
5630 CE4=CE4+BETA^MUZ*DE4(MUZ)
5640 IF DIAGNOSTIC.FLAG#("<"S" THEN GOTO 5660
5650 LPRINT TAB(12);"CE4=";CE4;" DE4(MUZ)=";DE4(MUZ)
5660 NEXT MUZ
5670 '
5680 FOR MUZ = 6 TO 8
5690 DE5=0
5700 IF MUZ=6 THEN NTZ=2 : LTZ=1 ELSE IF MUZ=7 THEN NTZ=2 : LTZ=1 ELSE IF MUZ=8 THEN NTZ=2 : LTZ=2
5710 '
5720 FOR MUZ = 1 TO NTZ
5730 DNS=0 : DDS=0
5740 FOR LAMBAX= 1 TO LTZ
5750 DNS=DNS+XZ(MUZ,LAMBAX)*SB(MUZ,LAMBAX)*X^XZ(MUZ,LAMBAX)
5760 DDS=DDS+SB(MUZ,LAMBAX)*X^XZ(MUZ,LAMBAX)
5770 NEXT LAMBAX
5780 DE5=DE5+B(MUZ,MUZ)*X^ZX(MUZ,MUZ) * ((1+ZX(MUZ,MUZ)*SB*THETA)-(SB*THETA+DNS)/(1/BETA^(-2+MUZ)+DDS))
5790 NEXT MUZ
5800 CE5=CE5+DE5/(1/BETA^(-2+MUZ)+DDS)
5810 NEXT MUZ
5820 '

```

APPENDIX G: ANALYSIS.BAS

```

5830 FOR NUZ = 0 TO 6
5840 DE6=DE6+(1+THETA*(10*BETA.L.PRIME/BETA.L+NUZ*SB))*B(9,NUZ)*X^NUZ
5850 NEXT NUZ
5860 '
5870 CE6=BETA*(BETA/BETA.L)^10*DE6
5880 '
5890 EPSILON.2=CE1+CE2-CE3-CE4-CE5+CE6 : H=2.212E+07*.00317*EPSILON.2/2326 : PRINT TAB(51);"Enthalpy";
5900 '
5910 IF DIAGNOSTIC.FLAG$(<)*"S" THEN GOTO 5980
5920 LPRINT TAB(12);"THETA=";THETA;" BETA=";BETA
5930 LPRINT TAB(12);"CE1=";CE1;" CE2=";CE2;" CE3=";CE3
5940 LPRINT TAB(12);"CE4=";CE4;" CE5=";CE5;" CE6=";CE6
5950 LPRINT TAB(12);"EPSILON.2=";EPSILON.2;" H=";H
5960 '
5970 PRINT "TABLES, LINE 9278: LZ=";LZ
5980 IF Z$="STEAM" OR Z$="FEEDWATER" THEN RETURN
5990 '
6000 '
6010 ' * Specific Volume *
6020 '
6030 CV2=0 : CV3=0 : DV2(1)=0 : DV2(2)=0 : DV2(3)=0 : DV2(4)=0 : DV2(5)=0 : DV4=0
6040 '
6050 CV1=11*THETA/BETA
6060 '
6070 FOR MUZ = 1 TO 5
6080 IF MUZ=1 THEN NTZ=2 ELSE IF MUZ=2 THEN NTZ=3 ELSE IF MUZ=3 THEN NTZ=2 ELSE IF MUZ=4 THEN NTZ=2
6090 IF MUZ=5 THEN NTZ=3
6100 FOR NUZ = 1 TO NTZ
6110 DV2(MUZ)=DV2(MUZ)+B(MUZ,NUZ)*X^ZZ(MUZ,NUZ)
6120 NEXT NUZ
6130 CV2=CV2+MUZ*BETA^(MUZ-1)*DV2(MUZ)
6140 NEXT MUZ
6150 '
6160 FOR MUZ = 6 TO 8
6170 DN3=0 : DD3=0
6180 IF MUZ=6 THEN NTZ=2 : LZ=1 ELSE IF MUZ=7 THEN NTZ=2 : LZ=1 ELSE IF MUZ=8 THEN NTZ=2 : LZ=2
6190 FOR NUZ = 1 TO NTZ
6200 DN3=DN3+B(MUZ,NUZ)*X^ZZ(MUZ,NUZ)
6210 NEXT NUZ
6220 FOR LAMBAZ = 1 TO LZ
6230 DD3=DD3+SB(MUZ,LAMBAZ)*X^XZ(MUZ,LAMBAZ)
6240 NEXT LAMBAZ
6250 CV3=CV3+(MUZ-2)*BETA^(1-MUZ)*DN3/(BETA^(2-MUZ)+DD3)^2
6260 NEXT MUZ
6270 '
6280 FOR NUZ = 0 TO 6 : DV4=DV4+B(9,NUZ)*X^NUZ : NEXT
6290 '
6300 CV4=11*(BETA/BETA.L)^10*DV4
6310 '
6320 CHI.2=CV1-CV2-CV3+CV4 : V=.00317*.45359237*CHI.2/.3048^3 : PRINT TAB(60);"Density"
6330 '
6340 IF Z$="TOP BLOW" OR Z$="BOTTOM BLOW" OR Z$="LETDOWN" OR Z$="CHARGING" OR Z$="PZR" THEN RETURN
6350 '
6360 PRINT : PRINT : PRINT

```

```

6370 *
6380 *
6390 *           ** SATURATION **
6400 *
6410 *           * Pressure *
6420 *
6430 IF THETA <=1 THEN PSAT# = 3208.2348#*BETA.K ELSE GOTO 6560
6440 *
6450 IF PRINT.FLAG#="S" THEN GOTO 6580 ELSE GOTO 6470
6460 *
6470 PRINT TAB(A+4);"Saturation Pressure = ";
6480 *
6490 IF T>580 THEN PRINT USING "###.##";PSAT#;      : GOTO 6540
6500 IF T>430 THEN PRINT USING "###.##";PSAT#;      : GOTO 6540
6510 IF T>180 THEN PRINT USING "###.###";PSAT#;    : GOTO 6540
6520 IF T>105 THEN PRINT USING "###.###";PSAT#;    : GOTO 6540 ELSE PRINT USING "###.###";PSAT#;
6530 *
6540 PRINT " psia"
6550 *
6560 RETURN
6570 *
6580 LPRINT TAB(A+4);"Saturation Pressure = ";
6590 *
6600 IF T>580 THEN LPRINT USING "###.##";PSAT#;      : GOTO 6650
6610 IF T>430 THEN LPRINT USING "###.##";PSAT#;      : GOTO 6650
6620 IF T>180 THEN LPRINT USING "###.###";PSAT#;    : GOTO 6650
6630 IF T>105 THEN LPRINT USING "###.###";PSAT#;    : GOTO 6650 ELSE LPRINT USING "###.###";PSAT#;
6640 *
6650 LPRINT " psia" : RETURN
6660 *
6670 *
6680 *           * Temperature *
6690 *
6700 DUMMY.THETA#=.5
6710 DUMMY.C# = 1 - DUMMY.THETA#
6720 DUMMY.SIGMA# = SK1#*DUMMY.C#+SK2#*DUMMY.C#^2+SK3#*DUMMY.C#^3+SK4#*DUMMY.C#^4 +SK5#*DUMMY.C#^5
6730 DUMMY.BETA.K# = EXP(DUMMY.SIGMA#/(DUMMY.THETA#*(1+SK6#*DUMMY.C# +SK7#*DUMMY.C#^2))
-DUMMY.C#/(SK8#*DUMMY.C#^2+SK9#))
6740 PSAT#=3208.2348#*DUMMY.BETA.K#
6750 P.ERROR#= P- PSAT#
6760 *
6770 IF ABS(P.ERROR#)>2500 THEN DAMPING=1!
6780 IF ABS(P.ERROR#)<=2500 AND ABS(P.ERROR#)>2000 THEN DAMPING=.5
6790 IF ABS(P.ERROR#)<=2000 AND ABS(P.ERROR#)>1500 THEN DAMPING=.35
6800 IF ABS(P.ERROR#)<=1500 AND ABS(P.ERROR#)>1100 THEN DAMPING=.25
6810 IF ABS(P.ERROR#)<=1100 AND ABS(P.ERROR#)> 100 THEN DAMPING=.15
6820 IF 100>=ABS(P.ERROR#) THEN DAMPING=.1
6830 *
6840 IF P>1000 THEN ACCEPTABLE.ERROR=.001#
6850 IF P<=1000 AND P>50 THEN ACCEPTABLE.ERROR=.0001#
6860 IF P<=50 THEN ACCEPTABLE.ERROR=.00001#
6870 *
6880 IF ABS(P.ERROR#)>ACCEPTABLE.ERROR THEN 6890 ELSE GOTO 6930
6890 DUMMY.THETA# = DUMMY.THETA#+(((DAMPING*P.ERROR#)/P)*DUMMY.THETA#)

```

APPENDIX G: ANALYSIS.BAS

```
6900 IF DUMMY.THETA#>=1 THEN DUMMY.THETA#=.99999
6910 GOTO 6710
6920 '
6930 TSAT=1165.14*DUMMY.THETA#-459.67 : RETURN
6940 END
```

APPENDIX H

DISPLAY.BAS

```

10 ' *****
20 ' ** DISPLAY.BAS **
30 ' *****
40 '
50 ' ** COMMON **
60 '
70 DEFN A-Z
80 '
90 REM Parameters
100 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
110 COMMON RISER.NUMBER,RISER.OD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,R.FLECT.AREA
120 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
130 '
140 REM Data
150 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
160 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
170 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
180 '
190 REM Auxiliary Results
200 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT#
210 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
220 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
230 '
240 '
250 ' ** RECALL **
260 '
270 OPEN DOCKET$ AS #1 LEN=127
280 FIELD #1,32 AS P$,1 AS U$,3 AS D$,2 AS L$,8 AS RP$,8 AS SI$,8 AS RO$,8 AS RN$,8 AS CO$,8 AS PI$,8 AS RA$,
  8 AS LC$,8 AS NA$,8 AS IT$,8 AS C$,1 AS PF$
290 GET #1,1 : LOOPSZ = CVI(L$) : CLOSE #1
300 '
310 OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59 : FIELD #1, 33 AS X$, 20 AS D$ : GET #1,1 : DDATE$=D$ : CLOSE #1
320 '
330 FOR IX=19 TO 1 STEP-1 : IF MID$(DDATE$,IX,1)<>" " THEN DDATE$=LEFT$(DDATE$,IX) : GOTO 360
340 NEXT IX
350 '
360 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=41 : FIELD #1,1 AS RF$,8 AS PH1$,8 AS PH2$,8 AS PF$,8 AS PP$,8 AS RPB$
370 GET #1,1 : PZR.ENTHALPY(1) = CVD(PH1$) : PZR.ENTHALPY(2) = CVD(PH2$) : PZR.FLOW = CVD(PF$)
380 PZR.POWER = CVD(PP$) : RC.PUMP.POWER.BTU.PER.HR = CVD(RPB$) : RESULTS.FLAG$ = RF$ : CLOSE #1
390 '
400 IF (RESULTS.FLAG$<>"S" AND RESULTS.FLAG$<>"C") THEN GOTO 700
410 '
420 OPEN "SG-CALC."+DOCKET$ AS #2 LEN=104
430 FIELD #2,8 AS SH$,8 AS SF$,8 AS SP$,8 AS FH$,8 AS FF$,8 AS FP$,8 AS TBH$,8 AS TBF$,8 AS TBP$,8 AS BBH$,
  8 AS BBF$,8 AS BBP$,8 AS SGP$
440 '
450 TZ=1 : LPZ=1 : FOR TZ=1 TO DATA.SETSZ : FOR LPZ=1 TO LOOPSZ : GET #2, (TZ-1) * LOOPSZ + LPZ
460 TOP.BLOW.ENTHALPY(LPZ,TZ) = CVD (TBH$) : STEAM.ENTHALPY(LPZ,TZ) = CVD (SH$)
470 TOP.BLOW.LB.PER.HR(LPZ,TZ) = CVD (TBF$) : STEAM.FLOW(LPZ,TZ) = CVD (SF$)
480 TOP.BLOW.POWER(LPZ,TZ) = CVD (TBP$) : STEAM.POWER(LPZ,TZ) = CVD (SP$)
490 BOTTOM.BLOW.ENTHALPY(LPZ,TZ) = CVD (BBH$) : FEEDWATER.ENTHALPY(LPZ,TZ) = CVD (FH$)
500 BOTTOM.BLOW.LB.PER.HR(LPZ,TZ) = CVD (BBF$) : FEEDWATER.FLOW(LPZ,TZ) = CVD (FF$)
510 BOTTOM.BLOW.POWER(LPZ,TZ) = CVD (BBP$) : FEEDWATER.POWER(LPZ,TZ) = CVD (FP$)
520 SG.POWER(LPZ,TZ) = CVD (SGP$)

```

APPENDIX H: DISPLAY.BAS

```

530 '
540 NEXT LP%,TZ : TZ=1 : LP%=1 : CLOSE #2
550 '
560 OPEN "OC-CALC."+DOCKET$ AS #3 LEN=88
570 FIELD #3,8 AS LH$,8 AS LF$,8 AS LP$,8 AS CH$,8 AS CF$,8 AS CP$,8 AS IP$,8 AS OP$,8 AS OM$,8 AS RXP$,8 AS T$
580 '
590 FOR TZ = 1 TO DATA.SETSZ
600 GET #3, TZ
610 LETDOWN.ENTHALPY(TZ)      = CVD (LH$) : INSULATION.POWER(TZ) = CVD (IP$)
620 LETDOWN.FLOW.LB.PER.HR(TZ) = CVD (LF$) : OTHER.POWER(TZ)      = CVD (OP$)
630 LETDOWN.POWER(TZ)        = CVD (LP$) : OTHER.POWER.MW(TZ)    = CVD (OM$)
640 CHARGING.ENTHALPY(TZ)     = CVD (CH$) : REACTOR.POWER.MW(TZ)  = CVD (RXP$)
650 CHARGING.FLOW.LB.PER.HR(TZ) = CVD (CF$) : TTIME(TZ)          = CVD (T$)
660 CHARGING.POWER(TZ)       = CVD (CP$)
670 '
680 NEXT TZ : TZ=1 : CLOSE #3 : GOTO 740
690 '
700 PRINT : PRINT : PRINT
710 PRINT TAB(23);"CHECK DIRECTORY FOR RCS-CALC."+DOCKET$;" FILE"
720 PRINT : PRINT : PRINT : STOP
730 '
740 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=45 : FIELD #1,1 AS RF$,8 AS PF$,8 AS PP$,8 AS RPB$,20 AS D$
750 GET #1,1 : RESULTS.FLAG$ = RF$ : CLOSE #1
760 '
770 IF (RESULTS.FLAG$("<"S" AND RESULTS.FLAG$("<"C") THEN GOTO 910
780 '
790 OPEN "OC-DATA."+DOCKET$ AS #2 LEN=72
800 FIELD #2,8 AS T$,8 AS LF$,8 AS LT$,8 AS CF$,8 AS CT$,8 AS PP$,8 AS PM$,8 AS TA$,8 AS TC$
810 TZ=1 : FOR TZ=1 TO 2 : GET #2,TZ : TTIME(TZ)=CVD(T$) : NEXT : CLOSE #2
820 '
830 TZ=1
840 FOR TZ=1 TO DATA.SETSZ
850 TTIME$(TZ)=RIGHT$(STR$(TTIME(TZ)+10000),4) : REM Numeric to literal
860 NEXT TZ
870 TZ=1
880 '
890 IF RESULTS.FLAG$="S" THEN GOTO 970 ELSE GOTO 910
900 '
910 CLS : LOCATE 13,26 : PRINT "Data have not been analyzed."
920 LOCATE 25,1 : PRINT SPC(79) : LOCATE 25,22,0 : PRINT "To continue, press the "; : COLOR 16,7
930 PRINT " space bar "; : COLOR 7,0 : PRINT ".";
940 Z$=INKEY$ : IF Z$("<"CHR$(32) THEN 940 ELSE 1780
950 '
960 '
970 '
980 '
990 CLS
1000 '
1010 '
1020 A=B : B=A+21 : C=B+14 : D=C+13 : E=D+12
1030 '
1040 '

```

```

1050 LOCATE 11,35 : PRINT "HEAT BALANCE"
1060 PRINT TAB(INT((82-LEN(PLANT$+UNIT$)-1)/2)); PLANT$; " ";UNIT$
1070 PRINT TAB(INT((82-LEN(DDATE$))/2)); DDATE$
1080 '
1090 GOSUB 1810 : REM Stop
1100 '
1110 TX=1 : LPZ=1
1120 FOR TX=1 TO DATA.SETSZ : F=2
1130 '
1140 GOSUB 1850 : REM Table heading
1150 '
1160 FOR LPZ=1 TO LOOPSZ : F=2
1170 '
1180 IF LPZ=1 THEN Y$="A" ELSE IF LPZ=2 THEN Y$="B" ELSE IF LPZ=3 THEN Y$="C" ELSE IF LPZ=4 THEN Y$="D"
1190 '
1200 PRINT TAB(A);"STEAM GENERATOR ";Y$ : PRINT
1210 '
1220 PRINT TAB(A);" Steam";TAB(B+2); : PRINT USING "####.#";STEAM.ENTHALPY(LPZ,TX);
1230 PRINT TAB(C-2); : PRINT USING "##.###"; STEAM.FLOW(LPZ,TX); : PRINT TAB(D-2);
1240 PRINT USING "##.###";STEAM.POWER(LPZ,TX)
1250 '
1260 PRINT TAB(A);" Feedwater";TAB(B+2); : PRINT USING "####.#";FEEDWATER.ENTHALPY(LPZ,TX);
1270 PRINT TAB(C-3); : PRINT USING "###.###"; -FEEDWATER.FLOW(LPZ,TX); : PRINT TAB(D-2);
1280 PRINT USING "##.###"; FEEDWATER.POWER(LPZ,TX)
1290 '
1300 PRINT TAB(A);" Surface Blowdown";TAB(B+2); : PRINT USING "####.#";TOP.BLOW.ENTHALPY(LPZ,TX);
1310 PRINT TAB(C-1); : PRINT USING "#.#####"; TOP.BLOW.LB.PER.HR(LPZ,TX); : PRINT TAB(D-1);
1320 PRINT USING "#.#####"; TOP.BLOW.POWER(LPZ,TX)
1330 '
1340 PRINT TAB(A);" Bottom Blowdown";TAB(B+2); : PRINT USING "####.#";BOTTOM.BLOW.ENTHALPY(LPZ,TX);
1350 PRINT TAB(C-1); : PRINT USING "#.#####";BOTTOM.BLOW.LB.PER.HR(LPZ,TX); : PRINT TAB(D-1);
1360 PRINT USING "#.#####";BOTTOM.BLOW.POWER(LPZ,TX)
1370 PRINT TAB(D-1);"-----"
1380 '
1390 PRINT TAB(A);" Power Dissipated";TAB(D-2); : PRINT USING "##.####";SG.POWER(LPZ,TX);
1400 PRINT TAB(E-1); : PRINT USING "####.#";292.875*SG.POWER(LPZ,TX)
1410 PRINT
1420 '
1430 IF (LPZ=2 AND LOOPSZ=2) OR (LPZ=4 AND LOOPSZ=4) THEN F=6
1440 IF LPZ=2 OR LPZ=4 THEN GOSUB 1810 : GOSUB 1850
1450 '
1460 NEXT LPZ
1470 '
1480 PRINT TAB(A);"OTHER COMPONENTS" : PRINT
1490 '
1500 PRINT TAB(A);" Letdown Line";TAB(B+2); : PRINT USING "####.#";LETDOWN.ENTHALPY(TX);
1510 PRINT TAB(C-1); : PRINT USING "#.#####"; LETDOWN.FLOW.LB.PER.HR(TX); : PRINT TAB(D-1);
1520 PRINT USING "#.#####"; LETDOWN.POWER(TX)
1530 '
1540 PRINT TAB(A);" Charging Line";TAB(B+2); : PRINT USING "####.#";CHARGING.ENTHALPY(TX);
1550 PRINT TAB(C-2); : PRINT USING "##.#####";CHARGING.FLOW.LB.PER.HR(TX); : PRINT TAB(D-2);
1560 PRINT USING "##.#####";CHARGING.POWER(TX)
1570 '

```


APPENDIX H: DISPLAY.BAS

```

1580 IF DATA.SETS% = 1 THEN 1630
1590 PRINT TAB(A); " Pressurizer"; TAB(B+2); : PRINT USING "####.#"; PZR.ENTHALPY(T%);
1600 PRINT TAB(C-2); : PRINT USING "##.#####"; PZR.FLOW; : PRINT TAB(D-2);
1610 PRINT USING "##.#####"; PZR.POWER
1620 '
1630 PRINT TAB(A); " Pumps"; TAB(D-2); : PRINT USING "##.#####"; RC.PUMP.POWER.BTU.PER.HR
1640 '
1650 PRINT TAB(A); " Insulation Losses"; TAB(D-1); : PRINT USING "##.#####"; INSULATION.POWER(T%)
1660 PRINT TAB(D-1); "-----"
1670 '
1680 PRINT TAB(A); " Power Dissipated"; TAB(D-2); : PRINT USING "##.#####"; OTHER.POWER(T%);
1690 PRINT TAB(E-1); : PRINT USING "####.#"; OTHER.POWER.MW(T%)
1700 PRINT TAB(E-1); "-----"
1710 '
1720 PRINT TAB(A); "REACTOR POWER"; TAB(E-1); : PRINT USING "####.#"; REACTOR.POWER.MW(T%)
1730 '
1740 GOSUB 1810 : REM Stop
1750 '
1760 NEXT T%
1770 '
1780 CLS : LOCATE 13,22 : PRINT "LOADING ANALYSIS MODULE INTO MEMORY"
1790 X$="MENU" : CHAIN "ANALYSIS"
1800 '
1810 COLOR 7,0 : LOCATE 25,23,0 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar ";
1820 COLOR 7,0 : PRINT ".";
1830 Z$=INKEY$ : IF Z$ <> CHR$(32) THEN 1830 ELSE CLS : RETURN
1840 '
1850 CLS : LOCATE F,A : PRINT "DATA SET"; T%; TAB(B); "ENTHALPY"; TAB(C); "FLOW"; TAB(D-1); "POWER"; TAB(E); "POWER"
1860 PRINT TAB(A); TTIME$(T%); " hours"; TAB(B); "(BTUs/lb)"; TAB(C-3); "(E6 lb/hr)"; TAB(D-4); "(E9 BTUs/hr)";
1870 PRINT TAB(E); "(MWt)" : PRINT : RETURN
1880 '
1890 '
1900 END

```

APPENDIX I

TABLES. BAS

TABLE OF CONTENTS

| | Page |
|-------------------------------------|------|
| | ---- |
| Independent Variables | I- 1 |
| Limits of Applicability | I- 2 |
| Sub-region and Phase Identification | I- 3 |
| Sub-region 1 | I- 4 |
| Sub-region 2 | I- 8 |
| Saturation | I-13 |

```

10 .
20 . *****
30 . ** TABLES.BAS **
40 . *****
50 .
60 . *****
70 . ** COMMON **
80 . *****
90 DEFDBL A-Z
100 .
110 REM Parameters
120 COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
130 COMMON RISER.NUMBER,RISER.DD,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,REFLECT.AREA
140 COMMON RC.PUMP.EFF,RC.PUMP.PWR,SS.ID,DOCKET$
150 .
160 REM Data
170 COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
180 COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
190 COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
200 .
210 REM Auxiliary Results
220 COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT#
230 COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,M%,N%,DATA.FLAG$,DIAGNOSTIC.FLAG$
240 COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
250 .
260 .
270 . *****
280 . * INDEPENDENT VARIABLES *
290 . *****
300 .
310 . Ref: Chapter IV and Appendix 1, Sections 2.1 and 8.1
320 .
330 . ** STEAM TABLES MENU **
340 .
350 CLS : TX= 3 : BX=20 : LX= 12 : RX=69
360 LOCATE TX,LX : PRINT CHR$(201) : LOCATE TX,RX : PRINT CHR$(187) : LOCATE BX,RX : PRINT CHR$(188)
370 LOCATE BX,LX : PRINT CHR$(200)
380 LOCATE TX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) ; : NEXT
390 LOCATE BX,LX+1 : FOR IX=1 TO RX-LX-1 : PRINT CHR$(205) ; : NEXT
400 FOR IX=TX+2 TO BX : LOCATE IX-1,LX : PRINT CHR$(186) : NEXT
410 FOR IX=TX+2 TO BX : LOCATE IX-1,RX : PRINT CHR$(186) : NEXT
420 .
430 LOCATE 25,23 : PRINT "After responding, press the "; : COLOR 16,7
440 PRINT " ";CHR$(17)+CHR$(196)+CHR$(217);" ";
450 COLOR 7,0 : PRINT " key. ";
460 .
470 LOCATE 3,31 : PRINT " STEAM TABLES MENU "
480 LOCATE 6,19 : PRINT "You can calculate enthalpy, specific volume,
490 LOCATE 7,19 : PRINT "and density for any of the following coolant
500 LOCATE 8,19 : PRINT "conditions:"
510 LOCATE 10,22 : PRINT "(A) Saturated at known temperature,"
520 LOCATE 12,22 : PRINT "(B) Saturated at known pressure, "
530 LOCATE 14,22 : PRINT "(C) Unsaturated at known temperature"
540 LOCATE 15,27 : PRINT "and pressure, or"

```

APPENDIX I: TABLES.BAS

```

550 LOCATE 17,22 : PRINT "(D) Return to the MASTER MENU." : LOCATE 20,57 : PRINT " "
560 LOCATE 20,24 : INPUT " Enter your selection (A/B/C/D): ",W$
570 CLS
580 '
590 IF W$="D" OR W$="d" THEN X$="INTRO 1920" : CHAIN "INTRO"
600 '
610 IF (W$(">"A" AND W$(">"a" AND W$(">"B" AND W$(">"b" AND W$(">"C" AND W$(">"c") THEN 350 ELSE 620
620 IF W$="B" OR W$="b" THEN W$="B" : GOTO 670
630 '
640 CLS : LOCATE 12,19 : INPUT "Type coolant temperature in degrees F: ",T
650 IF W$="A" OR W$="a" THEN W$="A" : CLS : GOTO 740
660 '
670 LOCATE 13,23 : INPUT "Type system pressure in psia: ",P
680 '
690 IF W$="B" THEN PRINT : PRINT : PRINT : GOTO 1410
700 '
710 '
720 '                ** REDUCED VARIABLES **
730 '
740 PRINT : PRINT : PRINT
750 THETA = (T+459.67)/1165.14 : IF W$="A" THEN GOTO 1320
760 BETA = P/3208.2348#
770 '
780 '
790 '                *****
800 '                * LIMITS OF APPLICABILITY *
810 '                *****
820 '
830 '                Ref: Appendix 1, Section B
840 '
850 L0 = 15.74373327# : L1=-34.17061978# : L2 = 19.31380707#
860 '
870 THETA.1 = .9626911787# : THETA.2 = 1.333462073#
880 THETA.3 = 1.657886606# : THETA.T = .4219960731#
890 '
900 BETA.2 = 4.52079566#
910 BETA.L = L0+L1*THETA+L2*THETA^2 : BETA.L.PRIME = L1+2*L2*THETA
920 '
930 IF THETA < THETA.T THEN CLS : LOCATE 12 : GOTO 940 ELSE 960
940 PRINT TAB(18);"Temperature is out of range. It is below"
950 PRINT TAB(18);"the triple point (32.018 F)." : GOTO 8490
960 '
970 IF BETA <= 0 THEN CLS : LOCATE 12 : GOTO 980 ELSE 1010
980 PRINT TAB(18);"Absolute pressure is zero or negative."
990 PRINT TAB(18);"Negative absolute pressure is impossible."
1000 GOTO 8490
1010 '
1020 IF THETA > THETA.3 THEN CLS : LOCATE 12 : GOTO 1030 ELSE 1050
1030 PRINT TAB(18);"Temperature is out of range. The upper limit is"
1040 PRINT TAB(18);"1472 degrees F." : GOTO 8490
1050 '
1060 IF THETA >= THETA.T AND THETA <= .6 AND BETA > 18.2*THETA-6.4 THEN CLS : LOCATE 12 : GOTO 1070 ELSE 1130
1070 PRINT TAB(18);"Pressure is out of range. For temperature"
1080 PRINT TAB(18);"equal to "; : PRINT USING "####.#";T;

```

```

1090 PRINT " degrees F,"; : PRINT " the limit on pressure";
1100 PRINT TAB(18);"is ";
1110 PRINT USING "#####.##";3208.235*(18.2*THETA-6.4);
1120 PRINT " psia." : GOTO 8490
1130 '
1140 IF THETA > .6 AND THETA <= THETA.1 AND BETA > BETA.2 THEN CLS : LOCATE 12 : GOTO 1150 ELSE 1180
1150 PRINT TAB(18);"Pressure is out of range. For temperature in"
1160 PRINT TAB(18);"the range from 239.4 to 662.0 degrees F, the"
1170 PRINT TAB(18);"upper limit is 14,503 psia." : GOTO 8490
1180 '
1190 IF THETA > THETA.1 AND THETA < THETA.2 AND BETA > BETA.L THEN CLS : LOCATE 12 : GOTO 1200 ELSE 1260
1200 PRINT TAB(18);"Pressure is out of range. For temperature"
1210 PRINT TAB(18);"equal to "; : PRINT USING "#####.##";T;
1220 PRINT " degrees F,"; : PRINT " the limit on pressure";
1230 PRINT TAB(18);"is ";
1240 PRINT USING "#####.##";221.2*14.5037738**BETA.L;
1250 PRINT " psia." : GOTO 8490
1260 '
1270 IF THETA >= THETA.2 AND THETA <= THETA.3 AND BETA > BETA.2 THEN CLS : LOCATE 12 : GOTO 1280 ELSE 1510
1280 PRINT TAB(18);"Pressure is out of range. For temperature in"
1290 PRINT TAB(18);"the range from 1094 to 1472 degrees F, the"
1300 PRINT TAB(18);"upper limit is 14,503 psia." : GOTO 8490
1310 '
1320 IF T > 705.47 THEN CLS : LOCATE 12 : GOTO 1330 ELSE 1360
1330 PRINT TAB(18);"Temperature is out of range. It is above the"
1340 PRINT TAB(18);"critical point (705.47 degrees F). Pressure"
1350 PRINT TAB(18);"will not condense supercritical steam." : GOTO 8490
1360 '
1370 IF T < 32.018 THEN CLS : LOCATE 12 : GOTO 1380 ELSE 1580
1380 PRINT TAB(18);"Temperature is out of range. It is below"
1390 PRINT TAB(18);"the triple point (32.018 F)." : GOTO 8490
1400 '
1410 IF P > 3208.2 THEN CLS : LOCATE 12 : GOTO 1420 ELSE 1440
1420 PRINT TAB(18);"Pressure is out of range. It is above the"
1430 PRINT TAB(18);"critical point (3208.2 psia)." : GOTO 8490
1440 '
1450 IF P < 8.864999E-02 THEN CLS : LOCATE 12 : GOTO 1460 ELSE 1580
1460 PRINT TAB(18);"Pressure is out of range. It is below"
1470 PRINT TAB(18);"the triple point (0.08865 psia)." : GOTO 8490
1480 '
1490 '
1500 '
1510 '
1520 '
1530 '
1540 '
1550 '
1560 IF W$="A" THEN 1840 ELSE IF W$="B" THEN 1640
1570 '
1580 SK1#=-7.691234564# : SK2#=-26.08023696# : SK3#=-168.1706546#
1590 SK4#= 64.23285504# : SK5#=-118.9646225# : SK6#= 4.16711732#
1600 SK7#= 20.9750676# : SK8#= 1E+09 : SK9#= 6
1610 '
1620 IF W$="B" THEN CLS : LOCATE 5,25 : PRINT "Computing Saturation Temperature": GOTO 7090

```

APPENDIX I: TABLES.BAS

```

1630 '
1640 IF THETA > 1 GOTO 1770
1650 '
1660 C = 1 - THETA
1670 '
1680 SIGMA = SK1#C+SK2#C^2+SK3#C^3+SK4#C^4+SK5#C^5
1690 '
1700 BETA.K = EXP(SIGMA/(THETA*(1+SK6#C+SK7#C^2)))-C/(SK8#C^2+SK9#)
1710 '
1720 IF W$="A" THEN 6810
1730 IF W$="B" THEN 1840
1740 '
1750 IF THETA >= THETA.T AND THETA <= THETA.1 AND BETA > BETA.K THEN REGION$="1"
1760 IF THETA >= THETA.T AND THETA <= THETA.1 AND BETA < BETA.K THEN REGION$="2"
1770 IF THETA >= THETA.1 AND BETA <= BETA.L THEN REGION$="2"
1780 '
1790 IF REGION$="1" THEN 1840
1800 IF REGION$="2" THEN 3880
1810 '
1820 '
1830 '
1840 '
1850 '
1860 '
1870 '
1880 '
1890 IF (W$="A" OR W$="B") THEN 1910 ELSE 1900
1900 CLS : LOCATE 5,26 : PRINT "Coolant is in the WATER phase"
1910 '
1920 '
1930 '
1940 '
1950 SA1 =.8438375405# : SA2 =.0005362162162# : SA3 =1.72
1960 SA4 =.07342278489# : SA5 =.0497585887# : SA6 =.65371543#
1970 SA7 =1.15E-06 : SA8 =1.5108E-05 : SA9 =.14188
1980 SA10=7.002753165# : SA11=.0002995284926# : SA12=.204
1990 '
2000 A0 = 6824.687741# : A1 =-542.2063673# : A2 =-20966.66205#
2010 A3 = 39412.86787# : A4 =-67332.77739# : A5 = 99023.81028#
2020 A6 =-109391.1774# : A7 = 85908.41667# : A8 =-45111.68742#
2030 A9 = 14181.38926# : A10=-2017.271113# : A11= 7.982692717#
2040 A12=-.02616571843# : A13= .00152241179# : A14= .02284279054#
2050 A15= 242.1647003# : A16= 1.2697160880-10 : A17= .0000002074838328#
2060 A18= .0000000271402035# : A19= 1.1057104980-09 : A20= 12.93441934#
2070 A21= .00001308119072# : A22= 6.0476263380-14
2080 '
2090 '
2100 '
2110 '
2120 PRINT : PRINT : PRINT : PRINT TAB(27); "Computing Enthalpy of Water" : PRINT : PRINT
2130 '
2140 Y=1-SA1*THETA^2-SA2*THETA^(-6) : Y.PRIME=-2*SA1*THETA+6*SA2*THETA^(-7)
2150 PRINT TAB(40); "5"
2160 '

```

* SUB-REGION 1 *

<***

Ref: Appendix I; Sections 7.1.1 and 9.1

** CONSTANTS **

** ENTHALPY **

```

2170 Z=Y+(SA3*Y^2-2*SA4*THETA+2*SA5*BETA)^.5 : ZE=-5/17
2180 '
2190 CE1=A0*THETA+A1-A3*THETA^2-2*A4*THETA^3-3*A5*THETA^4-4*A6*THETA^5-5*A7*THETA^6-6*A8*THETA^7-7*A9*THETA^8-
      B*A10*THETA^9
2200 PRINT TAB(40); "4"
2210 '
2220 CE2=A11*(Z*(17*(Z/29-Y/12)+5*THETA+Y.PRIME/12)+SA4*THETA -(SA3-1)*THETA*Y*Y.PRIME)*Z^ZE
2230 PRINT TAB(40); "3"
2240 '
2250 CE3=(A12-A14*THETA^2+A15*(9*THETA+SA6)*SGN(SA6-THETA)*(ABS(SA6-THETA))^9
      +A16*(20*THETA^19+SA7)*(SA7+THETA^19)^(-2))*BETA
2260 PRINT TAB(40); "2"
2270 '
2280 CE4=-((12*THETA^11+SA8)*(SA8+THETA^11)^(-2)*(A17*BETA+A18*BETA^2 +A19*BETA^3)
2290 PRINT TAB(40); "1"
2300 '
2310 CE5=A20*THETA^18*(17*SA9+19*THETA^2)*((SA10+BETA)^(-3)+SA11*BETA)
2320 '
2330 CE6=+A21*SA12*BETA^3 : CE7=+21*A22*THETA^(-20)*BETA^4
2340 '
2350 EPSILON.1=CE1+CE2+CE3+CE4+CE5+CE6+CE7 : H=2.212E+07*.00317*EPSILON.1/2326
2360 H=2.212E+07*.00317*EPSILON.1/2326
2370 '
2380 PRINT : PRINT : PRINT
2390 '
2400 '
2410 '
2420 '
2430 PRINT TAB(23); "Computing Specific Volume of Water" : PRINT : PRINT
2440 '
2450 CV1=A11*SA5*Z^ZE+(A12+A13*THETA+A14*THETA^2 +A15*(ABS(SA6-THETA))^10+A16/(SA7+THETA^19))
2460 PRINT TAB(40); "4"
2470 '
2480 CV2=-((SA8+THETA^11)^(-1)*(A17+2*A18*BETA+3*A19*BETA^2)
2490 PRINT TAB(40); "3"
2500 '
2510 CV3=-A20*THETA^18*(SA9+THETA^2)*(-3*(SA10+BETA)^(-4)+SA11)+3*A21*(SA12-THETA)*BETA^2+4*A22*THETA^(-20)*BETA^3
2520 PRINT TAB(40); "2"
2530 '
2540 CHI.1=CV1+CV2+CV3 : V=.00317*.45359237*CHI.1/.3048^3
2550 PRINT TAB(40); "1";
2560 '
2570 '
2580 '
2590 '
2600 IF W$="A" OR W$="B" THEN 2620 ELSE 2640
2610 '
2620 VF=V : HF=H : GOTO 3880
2630 '
2640 IF PRINT.FLAG$="S" THEN 2900
2650 '
2660 A=15 : CLS : LOCATE 9,A+6 : PRINT "Water Temperature = "; : PRINT USING "####.##";T;
2670 PRINT " degrees F"

```


APPENDIX I: TABLES.BAS

```

2680 PRINT TAB(A+8);"System Pressure ="; : PRINT USING "#####.##";P;
2690 PRINT " psia"
2700 '
2710 PRINT:PRINT TAB(A+4);"WATER PROPERTIES : "
2720 PRINT : PRINT TAB(A+15);"Enthalpy = ";
2730 IF P>3000 THEN PRINT USING "###.##";H; ELSE PRINT USING "###.##";H;
2740 PRINT " BTUs/pound"
2750 '
2760 PRINT TAB(A+8);"Specific Volume = ";
2770 IF P>100 THEN PRINT USING "#.#####";V; ELSE PRINT USING "#.#####";V;
2780 PRINT " cubic foot/pound"
2790 '
2800 PRINT TAB(A+16);"Density = "; : PRINT USING "##.##";1/V;
2810 PRINT " pounds/cubic foot" : PRINT
2820 '
2830 PRINT : PRINT : PRINT
2840 LOCATE 25,23 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".";
2850 Z$=INKEY$: IF Z$<>CHR$(32) THEN 2850
2860 '
2870 GOTO 3270
2880 '
2890 '
2900 ' * Printer Routine *
2910 '
2920 A=10 : ON ERROR GOTO 3210
2930 ON ERROR GOTO 3210
2940 '
2950 LPRINT TAB(A+6);"Water Temperature ="; : LPRINT USING "#####.##";T;
2960 LPRINT " degrees F"
2970 LPRINT TAB(A+8);"System Pressure ="; : LPRINT USING "#####.##";P;
2980 LPRINT " psia"
2990 '
3000 LPRINT:LPRINT TAB(A+4);"WATER PROPERTIES : "
3010 LPRINT : LPRINT TAB(A+15);"Enthalpy = ";
3020 IF P>3000 THEN LPRINT USING "###.##";H; ELSE LPRINT USING "###.##";H;
3030 LPRINT " BTUs/pound"
3040 '
3050 LPRINT TAB(A+8);"Specific Volume = ";
3060 IF P>100 THEN LPRINT USING "#.#####";V; ELSE LPRINT USING "#.#####";V;
3070 LPRINT " cubic foot/pound"
3080 '
3090 LPRINT TAB(A+16);"Density = "; : LPRINT USING "##.##";1/V;
3100 LPRINT " pounds/cubic foot" : LPRINT
3110 '
3120 IF X$="ANALYSIS" THEN CHAIN "ANALYSIS"
3130 LPRINT : LPRINT : LPRINT
3140 IF PRINT.FLAG$="C" THEN GOTO 3150 ELSE GOTO 3170
3150 INPUT " To continue, press RETURN."; D$
3160 '
3170 A=0
3180 '
3190 LPRINT : LPRINT : LPRINT
3200 GOTO 2660
3210 ' PRINTER ERROR TRAP ROUTINE

```

```

3220 IF ERR=25 THEN 3230 ELSE RESUME
3230 SOUND 2800,5 : SOUND 32767,20
3240 CLS : LOCATE 13,25 : PRINT "PLEASE TURN ON THE PRINTER !!"
3250 RESUME
3260 '
3270 '                ** DIAGNOSTICS **
3280 '
3290 IF DIAGNOSTIC.FLAG$="S" THEN 3300 ELSE GOTO 8510
3300 IF PRINT.FLAG$="S" THEN A=30 : B=A+5 : GOTO 3620 ELSE A=4
3310 B=A+5
3320 PRINT:PRINT TAB(A);"Diagnostics for preceding computation:";PRINT
3330 '
3340 PRINT TAB(B);"THETA = ";THETA : PRINT TAB(B);"SIGMA = ";SIGMA
3350 PRINT TAB(B);"BETA.K = ";BETA.K : PRINT TAB(B);"BETA = ";BETA
3360 '
3370 PRINT TAB(B);"Y = ";Y : PRINT TAB(B);"Y.PRIME = ";Y.PRIME
3380 PRINT TAB(B);"Z = ";Z : PRINT TAB(B);"ZE = ";ZE
3390 '
3400 PRINT TAB(B);"CE1 = ";CE1 : PRINT TAB(B);"CE2 = ";CE2
3410 PRINT TAB(B);"CE3 = ";CE3 : PRINT TAB(B);"CE4 = ";CE4
3420 PRINT TAB(B);"CE5 = ";CE5 : PRINT TAB(B);"CE6=";CE6
3430 PRINT TAB(B);"CE7=";CE7
3440 '
3450 PRINT TAB(B);"EPSILON.1 = ";EPSILON.1
3460 '
3470 PRINT TAB(B);"CV1 = ";CV1 : PRINT TAB(B);"CV2 = ";CV2
3480 PRINT TAB(B);"CV3 = ";CV3
3490 '
3500 PRINT TAB(B);"CHI.1 = ";CHI.1
3510 '
3520 IF PRINT.FLAG$="C" THEN GOTO 3560
3530 '
3540 PRINT CHR$(12) : : A=0
3550 '
3560 GOTO 8510
3570 '
3580 '
3590 '                * Printer Routine *
3600 ON ERROR GOTO 3210
3610 '
3620 LPRINT:LPRINT TAB(A);"Diagnostics for preceding computation:";LPRINT
3630 '
3640 LPRINT TAB(B);"THETA = ";THETA : LPRINT TAB(B);"SIGMA = ";SIGMA
3650 LPRINT TAB(B);"BETA.K = ";BETA.K : LPRINT TAB(B);"BETA = ";BETA
3660 '
3670 LPRINT TAB(B);"Y = ";Y : LPRINT TAB(B);"Y.PRIME = ";Y.PRIME
3680 LPRINT TAB(B);"Z = ";Z : LPRINT TAB(B);"ZE = ";ZE
3690 '
3700 LPRINT TAB(B);"CE1 = ";CE1 : LPRINT TAB(B);"CE2 = ";CE2
3710 LPRINT TAB(B);"CE3 = ";CE3 : LPRINT TAB(B);"CE4 = ";CE4
3720 LPRINT TAB(B);"CE5 = ";CE5 : LPRINT TAB(B);"CE6=";CE6
3730 LPRINT TAB(B);"CE7=";CE7
3740 '
3750 LPRINT TAB(B);"EPSILON.1 = ";EPSILON.1

```

APPENDIX I: TABLES.BAS

```

3760 '
3770 LPRINT TAB(8);"CV1 = ";CV1      : LPRINT TAB(8);"CV2 = ";CV2
3780 LPRINT TAB(8);"CV3 = ";CV3
3790 '
3800 LPRINT TAB(8);"CHI.1= ";CHI.1
3810 '
3820 PRINT CHR$(12) : A=0
3830 '
3840 GOTO 8510
3850 '
3860 '
3870 '
3880 '
3890 '
3900 '
3910 '
3920 '
3930 IF (W$="A" OR W$="B") THEN 3950 ELSE 3940
3940 PRINT TAB(26);"Coolant is in the STEAM phase"
3950 '
3960 '
3970 '
3980 '
3990 ALPHA.0 = 0
4000 '
4010 B0      = 16.83599274#      : B(0,1) = 28.56067796#
4020 B(0,2) = -54.38923329#     : B(0,3) = .4330662834#
4030 B(0,4) = -.6547711697#    : B(0,5) = .08565182058#
4040 B(1,1) = .06670375918#    : B(1,2) = 1.388983801#
4050 B(2,1) = .08390104328#    : B(2,2) = .02614670893#
4060 B(2,3) = -.03373439453#   : B(3,1) = .4520918904#
4070 B(3,2) = .1069036614#    : B(4,1) = -.5975336707#
4080 B(4,2) = -.08847535804#   : B(5,1) = .5958051609#
4090 B(5,2) = -.5159303373#   : B(5,3) = .2075021122#
4100 B(6,1) = .1190610271#    : B(6,2) = -.09867174132#
4110 B(7,1) = .1683998803#    : B(7,2) = -.05809438001#
4120 B(8,1) = .006552390126#  : B(8,2) = .0005710218649#
4130 B(9,0) = 193.6587558#    : B(9,1) = -1388.522425#
4140 B(9,2) = 4126.607219#    : B(9,3) = -6508.211677#
4150 B(9,4) = 5745.984054#    : B(9,5) = -2693.088365#
4160 B(9,6) = 523.5718623#
4170 '
4180 I1      = 4.260321148#
4190 '
4200 SB      = .7633333333#     : SB(6,1) = .4006073948#
4210 SB(7,1) = .08636081627#   : SB(8,1) = -.8532322921#
4220 SB(8,2) = .3460208861#
4230 '
4240 ZX(6,1)=14 : ZX(7,1)=19 : ZX(8,1)=54 : ZX(8,2)=27
4250 '
4260 ZX(1,1)=13 : ZX(1,2)=3  : ZX(2,1)=18 : ZX(2,2)=2  : ZX(2,3)=1
4270 ZX(3,1)=18 : ZX(3,2)=10 : ZX(4,1)=25 : ZX(4,2)=14 : ZX(5,1)=32
4280 ZX(5,2)=28 : ZX(5,3)=24 : ZX(6,1)=12 : ZX(6,2)=11 : ZX(7,1)=24
4290 ZX(7,2)=18 : ZX(8,1)=24 : ZX(8,2)=14

```

```

4300 '
4310 '
4320 '                ** ENTHALPY **
4330 '
4340 PRINT : PRINT : PRINT : PRINT TAB(27); "Computing Enthalpy of Steam" : PRINT : PRINT
4350 '
4360 CE3=0 : CE4=0 : CE5=0 : DE6=0 : CE6=0
4370 '
4380 X = EXP(SB*(1-THETA)) : CE1=ALPHA.0 : CE2=B0*THETA
4390 '
4400 FOR MUZ = 1 TO 5
4410 CE3=CE3+B(0,MUZ)*(MUZ-2)*THETA^(MUZ-1)
4420 NEXT MUZ
4430 '
4440 FOR MUZ = 1 TO 5
4450 IF MUZ=1 THEN NTZ=2
4460 IF MUZ=2 THEN NTZ=3
4470 IF MUZ=3 THEN NTZ=2
4480 IF MUZ=4 THEN NTZ=2
4490 IF MUZ=5 THEN NTZ=3
4500 FOR MUZ = 1 TO NTZ
4510 DE4(MUZ)=DE4(MUZ)+B(MUZ,MUZ)*(1+ZX(MUZ,MUZ)+SB*THETA)*X^ZX(MUZ,MUZ)
4520 IF DIAGNOSTIC.FLAG*("<"S" THEN GOTO 4550
4530 LPRINT TAB(12); "DE4(MUZ)=";DE4(MUZ);" B(MUZ,MUZ)=";B(MUZ,MUZ)
4540 LPRINT TAB(12); "(1+ZX(MUZ,MUZ)+SB*THETA)="; (1+ZX(MUZ,MUZ)+SB*THETA);" X^ZX(MUZ,MUZ)="; X^ZX(MUZ,MUZ)
4550 NEXT MUZ
4560 CE4=CE4+BETA^MUZ*DE4(MUZ)
4570 IF DIAGNOSTIC.FLAG*("<"S" THEN GOTO 4590
4580 LPRINT TAB(12); "CE4=";CE4;" DE4(MUZ)=";DE4(MUZ)
4590 PRINT TAB(39);16-MUZ
4600 NEXT MUZ
4610 '
4620 FOR MUZ = 6 TO 8
4630 DE5=0
4640 IF MUZ=6 THEN NTZ=2 : LZ=1
4650 IF MUZ=7 THEN NTZ=2 : LZ=1
4660 IF MUZ=8 THEN NTZ=2 : LZ=2
4670 FOR MUZ = 1 TO NTZ
4680 DNS=0 : DDS=0
4690 FOR LAMBAZ= 1 TO LZ
4700 DNS=DNS+XZ(MUZ,LAMBAZ)+SB(MUZ,LAMBAZ)*X^XZ(MUZ,LAMBAZ)
4710 DDS=DDS+SB(MUZ,LAMBAZ)*X^XZ(MUZ,LAMBAZ)
4720 NEXT LAMBAZ
4730 DES=DES+B(MUZ,MUZ)*X^ZX(MUZ,MUZ) * ((1+ZX(MUZ,MUZ)+SB*THETA)-(SB*THETA*DNS)/(1/BETA^(-2+MUZ)+DDS))
4740 NEXT MUZ
4750 CES=CES+DES/(1/BETA^(-2+MUZ)+DDS)
4760 IF MUZ< 7 THEN PRINT TAB(39);16-MUZ
4770 IF MUZ>=7 THEN PRINT TAB(40);16-MUZ
4780 NEXT MUZ
4790 '
4800 FOR MUZ = 0 TO 6
4810 DE6=DE6+(1+THETA*(10*BETA.L.PRIME/BETA.L+MUZ*SB))*B(9,MUZ)*X^MUZ
4820 PRINT TAB(40);7-MUZ
4830 NEXT MUZ

```

APPENDIX I: TABLES.BAS

```

4840 '
4850 CE6=BETA*(BETA/BETA.L)^10*DE6
4860 '
4870 EPSILON.2=CE1+CE2-CE3-CE4-CE5+CE6
4880 H=2.212E+07*.00317*EPSILON.2/2326
4890 '
4900 PRINT : PRINT : PRINT
4910 '
4920 '
4930 '                ** SPECIFIC VOLUME **
4940 '
4950 IF X$(<)*ANALYSIS THEN PRINT TAB(23);"Computing Specific Volume of Steam" : PRINT : PRINT
4960 '
4970 CV2=0 : CV3=0 : DV4=0
4980 '
4990 CV1=I1*THETA/BETA : PRINT TAB(40);"16"
5000 '
5010 FOR MUZ = 1 TO 5
5020 IF MUZ=1 THEN NTZ=2
5030 IF MUZ=2 THEN NTZ=3
5040 IF MUZ=3 THEN NTZ=2
5050 IF MUZ=4 THEN NTZ=2
5060 IF MUZ=5 THEN NTZ=3
5070 FOR NUZ = 1 TO NTZ
5080 DV2(MUZ)=DV2(MUZ)+B(MUZ,NUZ)*X^ZZ(MUZ,NUZ)
5090 NEXT NUZ
5100 CV2=CV2+MUZ*BETA^(MUZ-1)*DV2(MUZ)
5110 PRINT TAB(39);16-MUZ
5120 NEXT MUZ
5130 '
5140 FOR MUZ = 6 TO 8
5150 DN3=0 : DD3=0
5160 IF MUZ=6 THEN NTZ=2 : LZ=1
5170 IF MUZ=7 THEN NTZ=2 : LZ=1
5180 IF MUZ=8 THEN NTZ=2 : LZ=2
5190 FOR NUZ = 1 TO NTZ
5200 DN3=DN3+B(MUZ,NUZ)*X^ZZ(MUZ,NUZ)
5210 NEXT NUZ
5220 FOR LAMBZ = 1 TO LZ
5230 DD3=DD3+SB(MUZ,LAMBZ)*X^XZ(MUZ,LAMBZ)
5240 NEXT LAMBZ
5250 CV3=CV3+(MUZ-2)*BETA^(1-MUZ)*DN3/(BETA^(2-MUZ)+DD3)^2
5260 IF MUZ<7 THEN PRINT TAB(39);16-MUZ ELSE PRINT TAB(40);16-MUZ
5270 NEXT MUZ
5280 '
5290 FOR NUZ = 0 TO 6
5300 DV4=DV4+B(9,NUZ)*X^NUZ
5310 PRINT TAB(40);7-NUZ
5320 NEXT NUZ
5330 '
5340 CV4=11*(BETA/BETA.L)^10*DV4
5350 '
5360 CHI.2=CV1-CV2-CV3+CV4 : V=.00317*.45359237*CHI.2/.3048^3
5370 '

```

```

5380 '
5390 '                                ** OUTPUT **
5400 '
5410 IF W$="A" OR W$="B" THEN 5430 ELSE 5450
5420 '
5430 VG=V : HG=H : GOTO 7420
5440 '
5450 IF PRINT.FLAG$="S" THEN 5800
5460 '
5470 A=15 : CLS : LOCATE 9,A+6 : PRINT "Steam Temperature = " : PRINT USING "####.##";T;
5480 PRINT " degrees F"
5490 PRINT TAB(A+8);"System Pressure ="; : PRINT USING "#####.##";P;
5500 PRINT " psia"
5510 '
5520 PRINT:PRINT TAB(A+4);"STEAM PROPERTIES :";
5530 PRINT : PRINT TAB(A+15); "Enthalpy = " : PRINT USING "####.##";H;
5540 PRINT " BTUs/pound"
5550 '
5560 PRINT TAB(A+8);"Specific Volume = ";
5570 IF P>200 THEN PRINT USING "####.####";V; : GOTO 5600
5580 IF P>25 THEN PRINT USING "####.####";V; : GOTO 5600
5590 IF P>4.4 THEN PRINT USING "####.###";V; : ELSE PRINT USING "####.##";V;
5600 IF V<=1 THEN PRINT " cubic foot/pound" ELSE PRINT " cubic feet/pound"
5610 '
5620 PRINT TAB(A+16);"Density = ";
5630 IF P>200 THEN PRINT USING "##.###";1/V; : GOTO 5660
5640 IF P>25 THEN PRINT USING "##.###";1/V; : GOTO 5660
5650 IF P>4.4 THEN PRINT USING "##.#####";1/V; ELSE PRINT USING "##.#####";1/V;
5660 PRINT " pounds/cubic foot" : PRINT
5670 '
5680 IF THETA >= 1 THEN 5690 ELSE 5730
5690 PRINT TAB(A+1);"Temperature is greater than the critical ";
5700 PRINT "temperature"
5710 PRINT TAB(A+1);"which is 705.47 degrees F. Saturation is not"
5720 PRINT TAB(A+1);"possible at any pressure." : PRINT
5730 '
5740 PRINT : PRINT : PRINT
5750 LOCATE 25,23 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".";
5760 Z$=INKEY$ : IF Z$(<>CHR$(32)) THEN 5760
5770 '
5780 GOTO 6180
5790 '
5800 '                                * Printer Routine *
5810 '
5820 '
5830 ON ERROR GOTO 3210
5840 '
5850 LPRINT CHR$(13); : LPRINT CHR$(127);CHR$(69);
5860 '
5870 LPRINT TAB(A+7);"Steam Temperature = " : LPRINT USING "####.##";T;
5880 LPRINT " degrees F"
5890 LPRINT TAB(A+8);"System Pressure ="; : LPRINT USING "#####.##";P;
5900 LPRINT " psia"
5910 '

```

APPENDIX I: TABLES.BAS

```

5920 LPRINT:LPRINT TAB(A+4);"STEAM PROPERTIES :";
5930 LPRINT : LPRINT TAB(A+15); "Enthalpy = "; : LPRINT USING "####.#";H;
5940 LPRINT " BTUs/pound"
5950 '
5960 LPRINT TAB(A+8);"Specific Volume = ";
5970 IF P>200 THEN LPRINT USING "####.####";V; : GOTO 6000
5980 IF P>25 THEN LPRINT USING "####.####";V; : GOTO 6000
5990 IF P>4.4 THEN LPRINT USING "####.####";V; : ELSE LPRINT USING "####.####";V;
6000 IF V<=1 THEN LPRINT " cubic foot/pound" ELSE LPRINT " cubic feet/pound"
6010 '
6020 LPRINT TAB(A+16);"Density = ";
6030 IF P>200 THEN LPRINT USING "##.###";1/V; : GOTO 6060
6040 IF P>25 THEN LPRINT USING "##.###";1/V; : GOTO 6060
6050 IF P>4.4 THEN LPRINT USING "##.#####";1/V; ELSE LPRINT USING "##.#####";1/V;
6060 LPRINT " pounds/cubic foot" : LPRINT
6070 '
6080 IF THETA >= 1 THEN 6090 ELSE 6130
6090 LPRINT TAB(A+1);"Temperature is greater than the critical ";
6100 LPRINT "temperature"
6110 LPRINT TAB(A+1);"which is 705.47 degrees F. Saturation is not"
6120 LPRINT TAB(A+1);"possible at any pressure." : LPRINT
6130 '
6140 LPRINT : LPRINT : LPRINT
6150 GOTO 5470
6160 '
6170 '
6180 ' ** DIAGNOSTICS **
6190 '
6200 IF DIAGNOSTIC.FLAG$="S" THEN GOTO 6210 ELSE GOTO 8510
6210 IF PRINT.FLAG$="S" THEN A=30 : B=A+5 : GOTO 6470 ELSE A=4
6220 B=A+5
6230 PRINT : PRINT TAB(A);"Diagnostics for the preceding computation:"
6240 PRINT
6250 '
6260 PRINT TAB(B);"THETA = ";THETA; : PRINT TAB(B);"BETA = ";BETA
6270 PRINT TAB(B);"BETA.L = ";BETA.L
6280 PRINT TAB(B);"BETA.L.PRIME = ";BETA.L.PRIME
6290 PRINT TAB(B);"CE1 = ";CE1 : PRINT TAB(B);"CE2 = ";CE2
6300 PRINT TAB(B);"CE3 = ";CE3 : PRINT TAB(B);"CE4 = ";CE4
6310 PRINT TAB(B);"CE5 = ";CE5 : PRINT TAB(B);"CE6 = ";CE6
6320 '
6330 PRINT TAB(B);"EPSILON.2 = ";EPSILON.2
6340 '
6350 PRINT TAB(B);"CV1 = ";CV1 : PRINT TAB(B);"CV2 = ";CV2
6360 PRINT TAB(B);"CV3 = ";CV3 : PRINT TAB(B);"CV4 = ";CV4
6370 '
6380 PRINT TAB(B);"CHI.2 = ";CHI.2
6390 '
6400 IF PRINT.FLAG$="C" THEN GOTO 6440
6410 '
6420 PRINT CHR$(12) : : A=0
6430 '
6440 GOTO 8510
6450 '

```

```

6460 '
6470 '                * Printer Routine *
6480 '
6490 LPRINT : LPRINT TAB(A);"Diagnostics for the preceding computation:"
6500 LPRINT
6510 '
6520 LPRINT TAB(B);"THETA = ";THETA; : LPRINT TAB(B);"BETA = ";BETA
6530 LPRINT TAB(B);"BETA.L = ";BETA.L
6540 LPRINT TAB(B);"BETA.L.PRIME = ";BETA.L.PRIME
6550 LPRINT TAB(B);"CE1 = ";CE1 : LPRINT TAB(B);"CE2 = ";CE2
6560 LPRINT TAB(B);"CE3 = ";CE3 : LPRINT TAB(B);"CE4 = ";CE4
6570 LPRINT TAB(B);"CE5 = ";CE5 : LPRINT TAB(B);"CE6 = ";CE6
6580 '
6590 LPRINT TAB(B);"EPSILON.2 = ";EPSILON.2
6600 '
6610 LPRINT TAB(B);"CV1 = ";CV1 : LPRINT TAB(B);"CV2 = ";CV2
6620 LPRINT TAB(B);"CV3 = ";CV3 : LPRINT TAB(B);"CV4 = ";CV4
6630 '
6640 LPRINT TAB(B);"CHI.2 = ";CHI.2
6650 '
6660 IF PRINT.FLAG$="C" THEN GOTO 6440
6670 '
6680 PRINT CHR$(12) : : A=0
6690 '
6700 GOTO 8510
6710 '
6720 '
6730 '                *****
6740 '                * SATURATION *
6750 '                *****
6760 '
6770 '                ** PRESSURE **
6780 '
6790 '                Ref: Appendix 1; Sections 5 and 7.1.5
6800 '
6810 IF THETA <=1 THEN PSAT# = 3208.2348#*BETA.K ELSE 6950
6820 IF W$="A" THEN P=PSAT# : GOTO 760
6830 '
6840 IF PRINT.FLAG$="S" THEN GOTO 6970 ELSE GOTO 6860
6850 '
6860 PRINT TAB(A+4);"Saturation Pressure = ";
6870 '
6880 IF T>580 THEN PRINT USING "####.#";PSAT#; : GOTO 6930
6890 IF T>430 THEN PRINT USING "####.##";PSAT#; : GOTO 6930
6900 IF T>180 THEN PRINT USING "####.###";PSAT#; : GOTO 6930
6910 IF T>105 THEN PRINT USING "####.####";PSAT#; : GOTO 6930 ELSE PRINT USING "####.#####";PSAT#;
6920 '
6930 PRINT " psia"
6940 '
6950 RETURN
6960 '
6970 LPRINT TAB(A+4);"Saturation Pressure = ";
6980 '

```


APPENDIX I: TABLES.BAS

```

6990 IF T>580 THEN LPRINT USING "####.#";PSAT#; : GOTO 7040
7000 IF T>430 THEN LPRINT USING "####.##";PSAT#; : GOTO 7040
7010 IF T>180 THEN LPRINT USING "####.###";PSAT#; : GOTO 7040
7020 IF T>105 THEN LPRINT USING "#####";PSAT#; : GOTO 7040 ELSE LPRINT USING "####.#####";PSAT#;
7030 '
7040 LPRINT " psia"
7050 '
7060 RETURN
7070 '
7080 '
7090 '
7100 '
7110 PRINT : PRINT : PRINT
7120 DUMMY.THETA#=.5
7130 DUMMY.C# = 1 - DUMMY.THETA#
7140 DUMMY.SIGMA# = SK1#*DUMMY.C#+SK2#*DUMMY.C#^2+SK3#*DUMMY.C#^3+SK4#*DUMMY.C#^4 +SK5#*DUMMY.C#^5
7150 DUMMY.BETA.K# = EXP(DUMMY.SIGMA#/(DUMMY.THETA#*(1+SK6#+DUMMY.C#+SK7#*DUMMY.C#^2))
-DUMMY.C#/(SK8#+DUMMY.C#^2+SK9#))
7160 PSAT#=3208.2348#*DUMMY.BETA.K#
7170 P.ERROR#= P- PSAT#
7180 '
7190 PRINT TAB(25);"PSAT# Convergence is "; : PRINT USING "#####";P.ERROR#
7200 '
7210 IF ABS(P.ERROR#)>2500 THEN DAMPING=1!
7220 IF ABS(P.ERROR#)<=2500 AND ABS(P.ERROR#)>2000 THEN DAMPING=.5
7230 IF ABS(P.ERROR#)<=2000 AND ABS(P.ERROR#)>1500 THEN DAMPING=.35
7240 IF ABS(P.ERROR#)<=1500 AND ABS(P.ERROR#)>1100 THEN DAMPING=.25
7250 IF ABS(P.ERROR#)<=1100 AND ABS(P.ERROR#)> 100 THEN DAMPING=.15
7260 IF 100>ABS(P.ERROR#) THEN DAMPING=.1
7270 '
7280 IF P>1000 THEN ACCEPTABLE.ERROR=.001#
7290 IF P<=1000 AND P>50 THEN ACCEPTABLE.ERROR=.0001#
7300 IF P<=50 THEN ACCEPTABLE.ERROR=.00001#
7310 '
7320 IF ABS(P.ERROR#)>ACCEPTABLE.ERROR THEN 7330 ELSE GOTO 7370
7330 DUMMY.THETA# = DUMMY.THETA#+(((DAMPING*P.ERROR#)/P)*DUMMY.THETA#)
7340 IF DUMMY.THETA#>=1 THEN DUMMY.THETA#=.99999
7350 GOTO 7130
7360 '
7370 TSAT=1165.14*DUMMY.THETA#-459.67
7380 IF W#="B" THEN T=TSAT : GOTO 740
7390 '
7400 RETURN
7410 '
7420 '
7430 '
7440 '
7450 IF PRINT.FLAG#="S" THEN 7950
7460 '
7470 CLS : A=15 : LOCATE 6,A+6
7480 PRINT "Saturation Temperature = "; : PRINT USING "####.##";T;
7490 PRINT " degrees F"
7500 PRINT TAB(A+9);"Saturation Pressure = ";
7510 '

```

** TEMPERATURE **

** OUTPUT **

```

7520 IF T>580 THEN PRINT USING "####.#";PSAT#; : GOTO 7570
7530 IF T>430 THEN PRINT USING "####.##";PSAT#; : GOTO 7570
7540 IF T>180 THEN PRINT USING "###.###";PSAT#; : GOTO 7570
7550 IF T>105 THEN PRINT USING "####.####";PSAT#; : GOTO 7570 ELSE PRINT USING "####.####";PSAT#;
7560 '
7570 PRINT " psia"
7580 '
7590 PRINT:PRINT TAB(A+4);"WATER PROPERTIES : "
7600 PRINT : PRINT TAB(A+15);"Enthalpy = ";
7610 IF P>25 THEN PRINT USING "###.#";HF; ELSE PRINT USING "###.#";HF;
7620 PRINT " BTUs/pound"
7630 '
7640 PRINT TAB(A+8);"Specific Volume = ";
7650 IF P>100 THEN PRINT USING "#.#####";VF; ELSE PRINT USING "#.#####";VF;
7660 PRINT " cubic foot/pound"
7670 '
7680 PRINT TAB(A+16);"Density = "; : PRINT USING "##.##";1/VF;
7690 PRINT " pounds/cubic foot"
7700 '
7710 PRINT:PRINT TAB(A+4);"STEAM PROPERTIES : "
7720 PRINT : PRINT TAB(A+15);"Enthalpy = "; : PRINT USING "####.#";HG;
7730 PRINT " BTUs/pound"
7740 '
7750 PRINT TAB(A+8);"Specific Volume = ";
7760 IF P>200 THEN PRINT USING "####.#####";VG; : GOTO 7790
7770 IF P>25 THEN PRINT USING "####.#####";VG; : GOTO 7790
7780 IF P>4.4 THEN PRINT USING "####.###";VG; ELSE PRINT USING "####.###";VG;
7790 IF V<=1 THEN PRINT " cubic foot/pound" ELSE PRINT " cubic feet/pound"
7800 '
7810 PRINT TAB(A+16);"Density = ";
7820 IF P>200 THEN PRINT USING "##.###";1/VG; : GOTO 7850
7830 IF P>25 THEN PRINT USING "##.###";1/VG; : GOTO 7850
7840 IF P>4.4 THEN PRINT USING "##.#####";1/VG; ELSE PRINT USING "##.#####";1/VG;
7850 PRINT " pounds/cubic foot" : PRINT
7860 '
7870 PRINT : PRINT : PRINT
7880 '
7890 LOCATE 25,23 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".";
7900 I%=INKEY% : IF I%(>CHR$(32)) THEN 7900
7910 '
7920 GOTO 8510
7930 '
7940 '
7950 ' * Printer Routine *
7960 '
7970 ON ERROR GOTO 3210
7980 '
7990 LPRINT CHR$(18); : LPRINT CHR$(27);CHR$(69);
8000 '
8010 LPRINT TAB(A+6);"Saturation Temperature = "; : LPRINT USING "####.##";T;
8020 LPRINT " degrees F"
8030 LPRINT TAB(A+9);"Saturation Pressure = ";
8040 '

```

APPENDIX I: TABLES.BAS

```

8050 IF T>580 THEN LPRINT USING "####.#";PSAT#; : GOTO 8100
8060 IF T>430 THEN LPRINT USING "####.##";PSAT#; : GOTO 8100
8070 IF T>180 THEN LPRINT USING "####.###";PSAT#; : GOTO 8100
8080 IF T>105 THEN LPRINT USING "####.####";PSAT#; : GOTO 8100 ELSE LPRINT USING "####.#####";PSAT#;
8090 '
8100 LPRINT " psia"
8110 '
8120 LPRINT:LPRINT TAB(A+4);"WATER PROPERTIES : "
8130 LPRINT : LPRINT TAB(A+15);"Enthalpy = ";
8140 IF P>25 THEN LPRINT USING "###.#";HF; ELSE LPRINT USING "###.#";HF;
8150 LPRINT " BTUs/pound"
8160 '
8170 LPRINT TAB(A+8);"Specific Volume = ";
8180 IF P>100 THEN LPRINT USING "#.#####";VF; ELSE LPRINT USING "#.#####";VF;
8190 LPRINT " cubic foot/pound"
8200 '
8210 LPRINT TAB(A+16);"Density = "; : LPRINT USING "##.##";1/VF;
8220 LPRINT " pounds/cubic foot" : LPRINT
8230 '
8240 LPRINT:LPRINT TAB(A+4);"STEAM PROPERTIES : "
8250 LPRINT : LPRINT TAB(A+15);"Enthalpy = "; : LPRINT USING "####.#";HG;
8260 LPRINT " BTUs/pound"
8270 '
8280 LPRINT TAB(A+8);"Specific Volume = ";
8290 IF P>200 THEN LPRINT USING "####.#####";VG; : GOTO 8320
8300 IF P>25 THEN LPRINT USING "####.####";VG; : GOTO 8320
8310 IF P>4.4 THEN LPRINT USING "####.###";VG; : ELSE LPRINT USING "####.##";VG;
8320 IF V<=1 THEN LPRINT " cubic foot/pound" ELSE LPRINT " cubic feet/pound"
8330 '
8340 LPRINT TAB(A+16);"Density = ";
8350 IF P>200 THEN LPRINT USING "##.###";1/VG; : GOTO 8380
8360 IF P>25 THEN LPRINT USING "##.###";1/VG; : GOTO 8380
8370 IF P>4.4 THEN LPRINT USING "##.#####";1/VG; ELSE LPRINT USING "##.#####";1/VG;
8380 LPRINT " pounds/cubic foot" : LPRINT
8390 '
8400 LPRINT : LPRINT : LPRINT : GOTO 7470
8410 '
8420 IF PRINT.FLAG#="C" THEN GOTO 8430 ELSE GOTO 8450
8430 INPUT " To continue, press RETURN."; D#
8440 '
8450 A=0
8460 '
8470 LPRINT : LPRINT : LPRINT : GOTO 7470
8480 '
8490 LOCATE 25,23 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0 : PRINT ".*";
8500 Z#=#INKEY# : IF Z#<>CHR$(32) THEN 8500
8510 CLS : LOCATE 13,20 : PRINT "LOADING INTRODUCTION MODULE INTO MEMORY"
8520 *#="INTRO 1920" : CHAIN"INTRO"
8530 GOTO 280
8540 '
8550 END

```

APPENDIX J
REPORTS.BAS

TABLE OF CONTENTS

| | Page |
|--------------|------|
| Menu | J- 1 |
| Input Data | J- 2 |
| Heat Balance | J- 6 |
| Data Sheet | J- 9 |

APPENDIX J: REPORTS.BAS

```

550 *                                     *****
560 *                                     * INPUT DATA *
570 *                                     *****
580 *
590 *                                     ** RECALL **
600 *
610 LPRINT CHR$(15);           : REM Condensed Font
620 *
630 OPEN DOCKET$ AS #1 LEN=159 : FIELD #1,158 AS X$, 1 AS PF$ : GET #1,1 : PARAMETER.FLAG$=PF$
640 CLOSE : X$="INPUT DATA" : GOSUB 660 : CHAIN "PRM&DATA"
650 *
660 CLS : LOCATE 13,18 : PRINT "LOADING PARAMETERS & DATA MODULE INTO MEMORY" : RETURN
670 *
680 GOSUB 2980
690 *
700 *
710 *                                     * Printout *
720 *
730 GOSUB 3630 : A=18 : B=A+1 : C=A+34 : D=A+43 : E=A+63 : F=A+97 : G=A+106 : LPRINT CHR$(15);
740 *
750 IF LOOPS%=2 THEN LPRINT : LPRINT : LPRINT
760 *
770 LPRINT : LPRINT
780 WIDTH "LPT1:" ,132
790 LPRINT TAB(A+47); "HEAT BALANCE DATA"
800 LPRINT TAB(A+(111-(LEN(PLANT$+UNIT$)+1))/2); PLANT$; " " ; UNIT$
810 LPRINT TAB(A+(111-LEN(DDATE$))/2); DDATE$
820 *
830 LPRINT TAB(B); "PLANT PARAMETERS:"
840 *
850 LPRINT
860 LPRINT TAB(B+1); "REACTOR COOLANT SYSTEM";
870 LPRINT TAB(E+1); "REFLECTIVE INSULATION"
880 LPRINT TAB(B+1); " Pump Power (MW each)"; TAB(D+1); : LPRINT USING "##.#"; RC.PUMP.PWR;
890 LPRINT TAB(E+1); " Inside Surface Area (sq ft)"; TAB(G-1); : LPRINT USING "##,###"; REFLECT.AREA
900 LPRINT TAB(B+1); " Pump Efficiency (%); TAB(D+1); : LPRINT USING "##.#"; RC.PUMP.EFF;
910 LPRINT TAB(E+1); " Heat Loss Coefficient (BTUs/hr sq ft)"; TAB(G-1);
920 LPRINT USING "###.##"; LOSS.COEFFICIENT
930 LPRINT TAB(B+1); " Pressurizer Inside Diameter (inches)"; TAB(D); : LPRINT USING "###.#"; PZR.ID
940 LPRINT TAB(E+1); "NONREFLECTIVE INSULATION"
950 LPRINT TAB(E+1); "STEAM GENERATORS";
960 LPRINT TAB(E+1); " Inside Surface Area (sq ft)"; TAB(G-1);:LPRINT USING "##,###"; NONREFLECT.AREA
970 LPRINT TAB(B+1); " Dome Inside Diameter (inches)"; TAB(D-1); : LPRINT USING "###.##"; SG.ID;
980 LPRINT TAB(E+1); " Thickness (inches)"; TAB(G+1) ;: LPRINT USING "##.#"; INSULATION.THICKNESS
990 LPRINT TAB(B+1); " Riser Outside Diameter (inches)"; TAB(D) ; : LPRINT USING "##.##"; RISER.OD;
1000 LPRINT TAB(E+1); " Thermal Conductivity (BTUs/hr ft F)"; TAB(G-1); : LPRINT USING "##.###"; CONDUCTIVITY
1010 LPRINT TAB(B+1); " Number of Risers"; TAB(D+3);:LPRINT USING "##"; RISER.NUMBER
1020 LPRINT TAB(B+1); " Moisture Carry-over (%) in A"; TAB(D); : LPRINT USING "#.###"; CARRYOVER.A;
1030 LPRINT TAB(E+1); "LICENSED THERMAL POWER (Mwt)"; TAB(G+1); : LPRINT USING "####"; LIC.POWER
1040 LPRINT TAB(B+1); " Moisture Carry-over (%) in B"; TAB(D); : LPRINT USING "#.###"; CARRYOVER.B
1050 IF LOOPS%=2 THEN 1080 ELSE LPRINT TAB(B+1); " Moisture Carry-over (%) in C"; TAB(D); :
LPRINT USING "#.###"; CARRYOVER.C
1060 IF LOOPS%=3 THEN 1080 ELSE LPRINT TAB(B+1); " Moisture Carry-over (%) in D"; TAB(D); :
LPRINT USING "#.###"; CARRYOVER.D

```

```

1070
1080 LPRINT : LPRINT
1090
1100 LPRINT TAB(B);"DATA:"; : IF DATA.SETSZ=1 THEN LPRINT : GOTO 1160
1110 LPRINT TAB(C);"SET 1"; : LPRINT TAB(D);"SET 2"; : LPRINT TAB(F);"SET 1"; : LPRINT TAB(G);"SET 2"
1120
1130 LPRINT : LPRINT TAB(B+1);"TIME"; TAB(C+1); TTIME$(1); TAB(D+1); TTIME$(2);
1140 LPRINT TAB(E+1);"TIME"; TAB(F+1); TTIME$(1); TAB(G+1); TTIME$(2) : LPRINT : GOTO 1180
1150
1160 LPRINT : LPRINT TAB(B+1);"TIME"; TAB(D+1); TTIME$(1); TAB(E+1); "TIME"; TAB(G+1); TTIME$(1) : LPRINT
1170
1180 LPRINT TAB(B+1);"STEAM GENERATOR A"; : LPRINT TAB(E+1);"STEAM GENERATOR B"
1190
1200 LPRINT
1210
1220 LPRINT TAB(B+2); "Steam Pressure (psia)"; : IF DATA.SETSZ=1 THEN 1250
1230 LPRINT TAB(C-1); : LPRINT USING "####.#"; STEAM.PRES(1,1); : LPRINT TAB(D-1);
1240 LPRINT USING "####.#"; STEAM.PRES(1,2); : GOTO 1260
1250 LPRINT TAB(D-1); : LPRINT USING "####.#"; STEAM.PRES(1,1);
1260 LPRINT TAB(E+2); "Steam Pressure (psia)"; : IF DATA.SETSZ=1 THEN 1290
1270 LPRINT TAB(F-1); : LPRINT USING "####.#"; STEAM.PRES(2,1); : LPRINT TAB(G-1);
1280 LPRINT USING "####.#"; STEAM.PRES(2,2); : GOTO 1300
1290 LPRINT TAB(G-1); : LPRINT USING "####.#"; STEAM.PRES(2,1)
1300 LPRINT TAB(B+2); "Feedwater Flow (E6 lb/hr)"; : IF DATA.SETSZ=1 THEN 1330
1310 LPRINT TAB(C-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(1,1); : LPRINT TAB(D-1);
1320 LPRINT USING "##.###"; FEEDWATER.FLOW(1,2); : GOTO 1340
1330 LPRINT TAB(D-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(1,1);
1340 LPRINT TAB(E+2); "Feedwater Flow (E6 lb/hr)"; : IF DATA.SETSZ=1 THEN 1370
1350 LPRINT TAB(F-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(2,1); : LPRINT TAB(G-1);
1360 LPRINT USING "##.###"; FEEDWATER.FLOW(2,2); : GOTO 1380
1370 LPRINT TAB(G-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(2,1)
1380 LPRINT TAB(B+2); "Feedwater Temperature (F)"; : IF DATA.SETSZ=1 THEN 1410
1390 LPRINT TAB(C); : LPRINT USING "###.#"; FEEDWATER.TEMP(1,1); : LPRINT TAB(D);
1400 LPRINT USING "###.#"; FEEDWATER.TEMP(1,2); : GOTO 1420
1410 LPRINT TAB(D); : LPRINT USING "###.#"; FEEDWATER.TEMP(1,1);
1420 LPRINT TAB(E+2); "Feedwater Temperature (F)"; TAB(F); : IF DATA.SETSZ=1 THEN 1450
1430 LPRINT TAB(F); : LPRINT USING "###.#"; FEEDWATER.TEMP(2,1); : LPRINT TAB(G);
1440 LPRINT USING "###.#"; FEEDWATER.TEMP(2,2); : GOTO 1460
1450 LPRINT TAB(G); : LPRINT USING "###.#"; FEEDWATER.TEMP(2,1);
1460 LPRINT TAB(B+2); "Surface Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 1490
1470 LPRINT TAB(C); : LPRINT USING "###.#"; TOP.BLOW(1,1); : LPRINT TAB(D);
1480 LPRINT USING "###.#"; TOP.BLOW(1,2); : GOTO 1500
1490 LPRINT TAB(D); : LPRINT USING "###.#"; TOP.BLOW(1,1);
1500 LPRINT TAB(E+2); "Surface Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 1530
1510 LPRINT TAB(F); : LPRINT USING "###.#"; TOP.BLOW(2,1); : LPRINT TAB(G);
1520 LPRINT USING "###.#"; TOP.BLOW(2,2); : GOTO 1540
1530 LPRINT TAB(G); : LPRINT USING "###.#"; TOP.BLOW(2,1)
1540 LPRINT TAB(B+2); "Bottom Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 1570
1550 LPRINT TAB(C); : LPRINT USING "###.#"; BOTTOM.BLOW(1,1); : LPRINT TAB(D);
1560 LPRINT USING "###.#"; BOTTOM.BLOW(1,2); : GOTO 1580
1570 LPRINT TAB(D); : LPRINT USING "###.#"; BOTTOM.BLOW(1,1);
1580 LPRINT TAB(E+2); "Bottom Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 1610
1590 LPRINT TAB(F); : LPRINT USING "###.#"; BOTTOM.BLOW(2,1); : LPRINT TAB(G);
1600 LPRINT USING "###.#"; BOTTOM.BLOW(2,2); : GOTO 1620

```


APPENDIX J: REPORTS.BAS

```

1610 LPRINT TAB(6); : LPRINT USING "###.#"; BOTTOM.BLOW(2,1)
1620 LPRINT TAB(B+2); "Water Level (inches)"; : IF DATA.SETSZ=1 THEN 1650
1630 LPRINT TAB(C); : LPRINT USING "###.#"; SG.WATER.LEVEL(1,1); : LPRINT TAB(D);
1640 LPRINT USING "###.#"; SG.WATER.LEVEL(1,2); : GOTO 1660
1650 LPRINT TAB(D); : LPRINT USING "###.#"; SG.WATER.LEVEL(1,1);
1660 LPRINT TAB(E+2); "Water Level (inches)"; : IF DATA.SETSZ=1 THEN 1690
1670 LPRINT TAB(F); : LPRINT USING "###.#"; SG.WATER.LEVEL(2,1); : LPRINT TAB(G);
1680 LPRINT USING "###.#"; SG.WATER.LEVEL(2,2); : GOTO 1700
1690 LPRINT TAB(G); : LPRINT USING "###.#"; SG.WATER.LEVEL(2,1)
1700 '
1710 IF LOOPSZ=2 THEN 2540
1720 IF LOOPSZ=4 THEN 2030
1730 '
1740 LPRINT : LPRINT : LPRINT TAB(B);"STEAM GENERATOR C" : LPRINT
1750 '
1760 LPRINT TAB(B+2); "Steam Pressure (psia)"; : IF DATA.SETSZ=1 THEN 1790
1770 LPRINT TAB(C-1); : LPRINT USING "####.#"; STEAM.PRES(3,1); : LPRINT TAB(D-1);
1780 LPRINT USING "####.#"; STEAM.PRES(3,2); : GOTO 1800
1790 LPRINT TAB(D-1); : LPRINT USING "####.#"; STEAM.PRES(3,1)
1800 LPRINT TAB(B+2); "Feedwater Flow (E6 lb/hr)"; : IF DATA.SETSZ=1 THEN 1830
1810 LPRINT TAB(C-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(3,1); : LPRINT TAB(D-1);
1820 LPRINT USING "##.###"; FEEDWATER.FLOW(3,2); : GOTO 1840
1830 LPRINT TAB(D-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(3,1)
1840 LPRINT TAB(B+2); "Feedwater Temperature (F)"; : IF DATA.SETSZ=1 THEN 1870
1850 LPRINT TAB(C); : LPRINT USING "###.#"; FEEDWATER.TEMP(3,1); : LPRINT TAB(D);
1860 LPRINT USING "###.#"; FEEDWATER.TEMP(3,2); : GOTO 1880
1870 LPRINT TAB(D); : LPRINT USING "###.#"; FEEDWATER.TEMP(3,1)
1880 LPRINT TAB(B+2); "Surface Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN GOTO 1910
1890 LPRINT TAB(C); : LPRINT USING "###.#"; TOP.BLOW(3,1); : LPRINT TAB(D);
1900 LPRINT USING "###.#"; TOP.BLOW(3,2); : GOTO 1920
1910 LPRINT TAB(D); : LPRINT USING "###.#"; TOP.BLOW(3,1)
1920 LPRINT TAB(B+2); "Bottom Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 1950
1930 LPRINT TAB(C); : LPRINT USING "###.#"; BOTTOM.BLOW(3,1);
1940 LPRINT TAB(D); : LPRINT USING "###.#"; BOTTOM.BLOW(3,2); : GOTO 1960
1950 LPRINT TAB(D); : LPRINT USING "###.#"; BOTTOM.BLOW(3,1)
1960 LPRINT TAB(B+2); "Water Level (inches)"; : IF DATA.SETSZ=1 THEN 1990
1970 LPRINT TAB(C); : LPRINT USING "###.#"; SG.WATER.LEVEL(3,1); : LPRINT TAB(D);
1980 LPRINT USING "###.#"; SG.WATER.LEVEL(3,2); : GOTO 2010
1990 LPRINT TAB(D); : LPRINT USING "###.#"; SG.WATER.LEVEL(3,1)
2000 '
2010 IF LOOPSZ=3 THEN GOTO 2540
2020 '
2030 LPRINT : LPRINT : LPRINT TAB(B);"STEAM GENERATOR C"; : LPRINT TAB(E);"STEAM GENERATOR D" : LPRINT
2040 '
2050 LPRINT TAB(B+2); "Steam Pressure (psia)"; : IF DATA.SETSZ=1 THEN 2080
2060 LPRINT TAB(C-1); : LPRINT USING "####.#"; STEAM.PRES(3,1); : LPRINT TAB(D-1);
2070 LPRINT USING "####.#"; STEAM.PRES(3,2); : GOTO 2090
2080 LPRINT TAB(D-1); : LPRINT USING "####.#"; STEAM.PRES(3,1);
2090 LPRINT TAB(E+2); "Steam Pressure (psia)"; : IF DATA.SETSZ=1 THEN 2120
2100 LPRINT TAB(F-1); : LPRINT USING "####.#"; STEAM.PRES(4,1); : LPRINT TAB(G-1);
2110 LPRINT USING "####.#"; STEAM.PRES(4,2); : GOTO 2130
2120 LPRINT TAB(G-1); : LPRINT USING "####.#"; STEAM.PRES(4,1)
2130 LPRINT TAB(B+2); "Feedwater Flow (E6 lb/hr)"; : IF DATA.SETSZ=1 THEN 2160
2140 LPRINT TAB(C-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(3,1);

```

```

2150 LPRINT TAB(D-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(3,2); : GOTO 2170
2160 LPRINT TAB(D-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(3,1);
2170 LPRINT TAB(E+2); "Feedwater Flow (E6 lb/hr)"; : IF DATA.SETSZ=1 THEN 2200
2180 LPRINT TAB(F-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(4,1); : LPRINT TAB(G-1);
2190 LPRINT USING "##.###"; FEEDWATER.FLOW(4,2) : GOTO 2210
2200 LPRINT TAB(G-1); : LPRINT USING "##.###"; FEEDWATER.FLOW(4,1)
2210 LPRINT TAB(H+2); "Feedwater Temperature (F)"; : IF DATA.SETSZ=1 THEN 2240
2220 LPRINT TAB(C); : LPRINT USING "###.#"; FEEDWATER.TEMP(3,1); : LPRINT TAB(D);
2230 LPRINT USING "###.#"; FEEDWATER.TEMP(3,2); : GOTO 2250
2240 LPRINT TAB(D); : LPRINT USING "###.#"; FEEDWATER.TEMP(3,1);
2250 LPRINT TAB(E+2); "Feedwater Temperature (F)"; TAB(F); : IF DATA.SETSZ=1 THEN 2280
2260 LPRINT TAB(F); : LPRINT USING "###.#"; FEEDWATER.TEMP(4,1); : LPRINT TAB(G);
2270 LPRINT USING "###.#"; FEEDWATER.TEMP(4,2); : GOTO 2290
2280 LPRINT TAB(G); : LPRINT USING "###.#"; FEEDWATER.TEMP(4,1);
2290 LPRINT TAB(H+2); "Surface Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 2320
2300 LPRINT TAB(C); : LPRINT USING "###.#"; TOP.BLOW(3,1); : LPRINT TAB(D);
2310 LPRINT USING "###.#"; TOP.BLOW(3,2); : GOTO 2330
2320 LPRINT TAB(D); : LPRINT USING "###.#"; TOP.BLOW(3,1);
2330 LPRINT TAB(E+2); "Surface Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 2360
2340 LPRINT TAB(F); : LPRINT USING "###.#"; TOP.BLOW(4,1); : LPRINT TAB(G);
2350 LPRINT USING "###.#"; TOP.BLOW(4,2) : GOTO 2370
2360 LPRINT TAB(G); : LPRINT USING "###.#"; TOP.BLOW(4,1)
2370 LPRINT TAB(H+2); "Bottom Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 2400
2380 LPRINT TAB(C); : LPRINT USING "###.#"; BOTTOM.BLOW(3,1);
2390 LPRINT TAB(D); : LPRINT USING "###.#"; BOTTOM.BLOW(3,2); : GOTO 2410
2400 LPRINT TAB(D); : LPRINT USING "###.#"; BOTTOM.BLOW(3,1);
2410 LPRINT TAB(E+2); "Bottom Blowdown (gpm)"; : IF DATA.SETSZ=1 THEN 2440
2420 LPRINT TAB(F); : LPRINT USING "###.#"; BOTTOM.BLOW(4,1); : LPRINT TAB(G);
2430 LPRINT USING "###.#"; BOTTOM.BLOW(4,2) : GOTO 2450
2440 LPRINT TAB(G); : LPRINT USING "###.#"; BOTTOM.BLOW(4,1)
2450 LPRINT TAB(H+2); "Water Level (inches)"; : IF DATA.SETSZ=1 THEN 2480
2460 LPRINT TAB(C); : LPRINT USING "###.#"; SG.WATER.LEVEL(3,1); : LPRINT TAB(D);
2470 LPRINT USING "###.#"; SG.WATER.LEVEL(3,2); : GOTO 2490
2480 LPRINT TAB(D); : LPRINT USING "###.#"; SG.WATER.LEVEL(3,1);
2490 LPRINT TAB(E+2); "Water Level (inches)"; : IF DATA.SETSZ=1 THEN 2520
2500 LPRINT TAB(F); : LPRINT USING "###.#"; SG.WATER.LEVEL(4,1); : LPRINT TAB(G);
2510 LPRINT USING "###.#"; SG.WATER.LEVEL(4,2) : GOTO 2540
2520 LPRINT TAB(G); : LPRINT USING "###.#"; SG.WATER.LEVEL(4,1)
2530 '
2540 LPRINT : LPRINT
2550 '
2560 LPRINT TAB(H+1); "LETDOWN LINE"; : LPRINT TAB(E+1); "CHARGING LINE" : LPRINT
2570 LPRINT TAB(H+2); "Flow (gpm)"; : IF DATA.SETSZ=1 THEN GOTO 2600
2580 LPRINT TAB(C); : LPRINT USING "###.#"; LETDOWN.FLOW(1); : LPRINT TAB(D);
2590 LPRINT USING "###.#"; LETDOWN.FLOW(2); : GOTO 2610
2600 LPRINT TAB(D); : LPRINT USING "###.#"; LETDOWN.FLOW(1);
2610 LPRINT TAB(E+2); "Flow (gpm)"; : IF DATA.SETSZ=1 THEN 2640
2620 LPRINT TAB(F); : LPRINT USING "###.#"; CHARGING.FLOW(1); : LPRINT TAB(G);
2630 LPRINT USING "###.#"; CHARGING.FLOW(2) : GOTO 2650
2640 LPRINT TAB(G); : LPRINT USING "###.#"; CHARGING.FLOW(1)
2650 LPRINT TAB(H+2); "Temperature (F)"; : IF DATA.SETSZ=1 THEN 2680
2660 LPRINT TAB(C); : LPRINT USING "###.#"; LETDOWN.TEMP(1); : LPRINT TAB(D);
2670 LPRINT USING "###.#"; LETDOWN.TEMP(2); : GOTO 2690
2680 LPRINT TAB(D); : LPRINT USING "###.#"; LETDOWN.TEMP(1);

```

APPENDIX J: REPORTS.BAS

```

2690 LPRINT TAB(E+2);"Temperature (F)"; : IF DATA.SETSZ=1 THEN 2720
2700 LPRINT TAB(F); : LPRINT USING "###.0"; CHARGING.TEMP(1); : LPRINT TAB(G);
2710 LPRINT USING "###.0"; CHARGING.TEMP(2); GOTO 2740
2720 LPRINT TAB(G); : LPRINT USING "###.0"; CHARGING.TEMP(1)
2730 '
2740 LPRINT : LPRINT : LPRINT TAB(B+1);"PRESSURIZER";TAB(E+1);"REACTOR" : LPRINT
2750 '
2760 LPRINT TAB(B+2);"Pressure (psia)"; : IF DATA.SETSZ=1 THEN 2790
2770 LPRINT TAB(C-1); : LPRINT USING "####.0"; PZR.PRES(1); : LPRINT TAB(D-1);
2780 LPRINT USING "####.0"; PZR.PRES(2); : GOTO 2800
2790 LPRINT TAB(D-1); : LPRINT USING "####.0"; PZR.PRES(1);
2800 LPRINT TAB(E+2);"T ave (F)"; : IF DATA.SETSZ=1 THEN 2830
2810 LPRINT TAB(F); : LPRINT USING "###.0"; T.AVE(1); : LPRINT TAB(G);
2820 LPRINT USING "###.0"; T.AVE(2); GOTO 2840
2830 LPRINT TAB(G); : LPRINT USING "###.0"; T.AVE(1)
2840 LPRINT TAB(B+2);"Water Level (inches)"; : IF DATA.SETSZ=1 THEN 2870
2850 LPRINT TAB(C); : LPRINT USING "###.0"; PZR.WATER.LEVEL(1); : LPRINT TAB(D);
2860 LPRINT USING "###.0"; PZR.WATER.LEVEL(2); : GOTO 2880
2870 LPRINT TAB(D); : LPRINT USING "###.0"; PZR.WATER.LEVEL(1);
2880 LPRINT TAB(E+2);"T cold (F)"; : IF DATA.SETSZ=1 THEN GOTO 2910
2890 LPRINT TAB(F); : LPRINT USING "###.0"; T.COLD(1); : LPRINT TAB(G);
2900 LPRINT USING "###.0"; T.COLD(2); GOTO 2930
2910 LPRINT TAB(G); : LPRINT USING "###.0"; T.COLD(1)
2920 '
2930 IF LOOPSZ=2 THEN FT=19 ELSE IF LOOPSZ=3 THEN FT=11 ELSE IF LOOPSZ=4 THEN FT=10
2940 FOR KX=1 TO FT : LPRINT : NEXT KX
2950 '
2960 LPRINT CHR$(18); : A=0 : GOTO 5210
2970 '
2980 FOR TX=1 TO 2 : TTIME$(TX)=RIGHT$(STR$(TTIME(TX)+10000),4) : NEXT TX : RETURN
2990 '
3000 '
3010 '
3020 ' *****
3030 ' * HEAT BALANCE *
3040 ' *****
3050 FOR IX=20 TO 1 STEP -1
3060 IF MID$(DDATE$,IX,1)<>" * THEN DDATE$=LEFT$(DDATE$,IX) : GOTO 3100
3070 NEXT IX
3080 '
3090 '
3100 ' ** RECALL **
3110 '
3120 LPRINT CHR$(18); : LPRINT CHR$(20); : LPRINT CHR$(27);CHR$(69); : OPEN DOCKET$ AS #1 LEN=127
3130 FIELD #1,32 AS P$,1 AS U$,3 AS D$,2 AS L$,8 AS RP$,8 AS SI$,8 AS RO$,8 AS RN$,8 AS CO$,8 AS PI$,8 AS RA$,
8 AS LC$,8 AS NA$,8 AS IT$,8 AS C$,1 AS PF$
3140 GET #1,1 : LOOPSZ = CVI(L$) : CLOSE #1
3150 '
3160 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=41 : FIELD #1,1 AS RF$,8 AS PH1$,8 AS PH2$,8 AS PF$,8 AS PP$,8 AS RPB$
3170 GET #1,1 : PZR.ENTHALPY(1) = CVD(PH1$) : PZR.ENTHALPY(2) = CVD(PH2$) : PZR.FLOW = CVD(PF$)
3180 PZR.POWER = CVD(PP$) : RC.PUMP.POWER.BTU.PER.HR = CVD(RPB$) : RESULTS.FLAG$ = RF$ : CLOSE #1
3190 '
3200 IF (RESULTS.FLAG$<>"S" AND RESULTS.FLAG$<>"C") THEN GOTO 3500
3210 '

```

```

3220 OPEN "SG-CALC."*DOCKET$ AS #2 LEN=104
3230 FIELD #2,8 AS SH$,8 AS SF$,8 AS SP$,8 AS FH$,8 AS FF$,8 AS FP$,8 AS TBH$,8 AS TBF$,8 AS TBP$,8 AS BBH$,
      8 AS BBF$,8 AS BBP$,8 AS SGP$
3240
3250 FOR TX=1 TO 2 : FOR LZ=1 TO LOOPSX : GET #2, (TX-1) * LOOPSX + LZ
3260 TOP.BLOW.ENTHALPY(LZ,TX) = CVD (TBH$) : STEAM.ENTHALPY(LZ,TX) = CVD (SH$)
3270 TOP.BLOW.LB.PER.HR(LZ,TX) = CVD (TBF$) : STEAM.FLOW(LZ,TX) = CVD (SF$)
3280 TOP.BLOW.POWER(LZ,TX) = CVD (TBP$) : STEAM.POWER(LZ,TX) = CVD (SP$)
3290 BOTTOM.BLOW.ENTHALPY(LZ,TX) = CVD (BBH$) : FEEDWATER.ENTHALPY(LZ,TX) = CVD (FH$)
3300 BOTTOM.BLOW.LB.PER.HR(LZ,TX) = CVD (BBF$) : FEEDWATER.FLOW(LZ,TX) = CVD (FF$)
3310 BOTTOM.BLOW.POWER(LZ,TX) = CVD (BBP$) : FEEDWATER.POWER(LZ,TX) = CVD (FP$)
3320 SG.POWER(LZ,TX) = CVD (SGP$)
3330
3340 NEXT LZ,TX : CLOSE #2
3350
3360 OPEN "OC-CALC."*DOCKET$ AS #3 LEN=88
3370 FIELD #3,8 AS LH$,8 AS LF$,8 AS LP$,8 AS CH$,8 AS CF$,8 AS CP$,8 AS IP$,8 AS OP$,8 AS OM$,8 AS RXP$,8 AS T$
3380
3390 FOR TX = 1 TO 2
3400 GET #3, TX
3410 LETDOWN.ENTHALPY(TX) = CVD (LH$) : INSULATION.POWER(TX) = CVD (IP$)
3420 LETDOWN.FLOW.LB.PER.HR(TX) = CVD (LF$) : OTHER.POWER(TX) = CVD (OP$)
3430 LETDOWN.POWER(TX) = CVD (LP$) : OTHER.POWER.MW(TX) = CVD (OM$)
3440 CHARGING.ENTHALPY(TX) = CVD (CH$) : REACTOR.POWER.MW(TX) = CVD (RXP$)
3450 CHARGING.FLOW.LB.PER.HR(TX) = CVD (CF$) : TTIME(TX) = CVD (T$)
3460 CHARGING.POWER(TX) = CVD (CP$)
3470
3480 NEXT TX : CLOSE #3 : GOTO 3550
3490
3500 PRINT : PRINT : PRINT
3510 PRINT TAB(23);"CHECK DIRECTORY FOR RCS-CALC.;"DOCKET$;" FILE"
3520 PRINT : PRINT : PRINT : STOP
3530
3540
3550
3560
3570 IF RESULTS.FLAG$="S" THEN GOSUB 3630 : GOTO 3760 ELSE GOTO 3590
3580
3590 CLS : LOCATE 11,27 : PRINT "Data have not been analyzed."
3600 PRINT TAB(27);"Go to the ANALYSIS MENU." : GOSUB 3710
3610 CLS : LOCATE 13,21 : PRINT "LOADING INTRODUCTION MODULE INTO MEMORY" : GOTO 3210
3620
3630 CLS : LOCATE 11,26 : PRINT "NOTE: Adjust the paper so that"
3640 PRINT TAB(26);"the top of the page is under"
3650 PRINT TAB(26); "the roller shaft." : GOSUB 3710
3660
3670 CLS : LOCATE 12,26 : PRINT "Make certain that the printer"
3680 LOCATE 13,26 : PRINT "is " ; : COLOR 0,7 : PRINT " ON LINE " ; : COLOR 7,0 : PRINT ".*"
3690 GOSUB 3710 : RETURN
3700
3710 LOCATE 23,22,0 : PRINT " To continue, press the " ; : COLOR 16,7 : PRINT " space bar " ; : COLOR 7,0
3720 PRINT ".*"
3730 I$=INKEY$ : IF I$(<)>CHR$(32) THEN 3730 ELSE CLS : LOCATE 13,23 : PRINT "OUTPUT IS DIRECTED TO THE PRINTER"
3740 RETURN

```

APPENDIX J: REPORTS.BAS

```

3750 '
3760 A=11 : B=A+20 : C=B+15 : D=C+14 : E=D+11
3770 '
3780 OPEN DOCKET$ AS #1 LEN=127 : FIELD #1,36 AS A$,2 AS L$,89 AS B$ : GET #1,1 : LOOP$=CVI(L$)
3790 CLOSE
3800 '
3810 GOSUB 2980          REM Test time
3820 '
3830 FOR TZ=1 TO 2
3840 '
3850 IF LOOP$=2 THEN HD=8 ELSE IF LOOP$=3 THEN HD=5 ELSE IF LOOP$=4 THEN HD=2
3860 FOR KZ=1 TO HD : LPRINT : NEXT KZ
3870 '
3880 LPRINT TAB(38) : "HEAT BALANCE"
3890 LPRINT TAB(A+33-INT((1+LEN(PLANT$+UNIT$))/2)) : PLANT$ : " " : UNIT$
3900 LPRINT TAB(A+33-INT(LEN(DDATE$)/2)) : DDATE$
3910 LPRINT
3920 LPRINT TAB(A) : "DATA SET" : TZ : "OF" : DATA.SETS$ : TAB(B) : "ENTHALPY" : TAB(C) : "FLOW" : TAB(D) : "POWER" : TAB(E) : "POWER"
3930 LPRINT TAB(A) : "TIME" : TZ : " hours" : TAB(B) : "(BTUs/lb)" : TAB(C-3) : "(E6 lb/hr)" : TAB(D-4) : "(E9 BTUs/hr)" :
3940 LPRINT TAB(E) : "(MWt)" : LPRINT
3950 '
3960 FOR LZ=1 TO LOOP$
3970 '
3980 IF LZ=1 THEN Y$="A" ELSE IF LZ=2 THEN Y$="B" ELSE IF LZ=3 THEN Y$="C" ELSE IF LZ=4 THEN Y$="D"
3990 '
4000 LPRINT TAB(A) : "STEAM GENERATOR " : Y$ : LPRINT
4010 '
4020 LPRINT TAB(A) : " Stea" : TAB(B+2) : : LPRINT USING "####.#" : STEAM.ENTHALPY(LZ,TZ) : : LPRINT TAB(C-2) :
4030 LPRINT USING "##.###" : STEAM.FLOW(LZ,TZ) : : LPRINT TAB(D-2) : : LPRINT USING "##.###" : STEAM.POWER(LZ,TZ)
4040 '
4050 LPRINT TAB(A) : " Feedwater" : TAB(B+2) : : LPRINT USING "####.#" : FEEDWATER.ENTHALPY(LZ,TZ) :
4060 LPRINT TAB(C-3) : : LPRINT USING "###.###" : -FEEDWATER.FLOW(LZ,TZ) : : LPRINT TAB(D-2) :
4070 LPRINT USING "##.###" : FEEDWATER.POWER(LZ,TZ)
4080 '
4090 LPRINT TAB(A) : " Surface Blowdown" : TAB(B+2) : : LPRINT USING "####.#" : TOP.BLOW.ENTHALPY(LZ,TZ) :
4100 LPRINT TAB(C-1) : : LPRINT USING "#.#####" : TOP.BLOW.LB.PER.HR(LZ,TZ) : : LPRINT TAB(D-1) :
4110 LPRINT USING "#.#####" : TOP.BLOW.POWER(LZ,TZ)
4120 '
4130 LPRINT TAB(A) : " Bottom Blowdown" : TAB(B+2) : : LPRINT USING "####.#" : BOTTOM.BLOW.ENTHALPY(LZ,TZ) :
4140 LPRINT TAB(C-1) : : LPRINT USING "#.#####" : BOTTOM.BLOW.LB.PER.HR(LZ,TZ) : : LPRINT TAB(D-1) :
4150 LPRINT USING "#.#####" : BOTTOM.BLOW.POWER(LZ,TZ) : LPRINT TAB(D-1) : "-----"
4160 '
4170 LPRINT TAB(A) : " Power Dissipated" : TAB(D-2) : : LPRINT USING "##.###" : SG.POWER(LZ,TZ) :
4180 LPRINT TAB(E-1) : : LPRINT USING "####.#" : 292.875*SG.POWER(LZ,TZ) : LPRINT
4190 '
4200 NEXT LZ
4210 '
4220 LPRINT TAB(A) : "OTHER COMPONENTS" : LPRINT
4230 '
4240 LPRINT TAB(A) : " Letdown Line" : TAB(B+2) : : LPRINT USING "####.#" : LETDOWN.ENTHALPY(TZ) : : LPRINT TAB(C-1) :
4250 LPRINT USING "#.#####" : LETDOWN.FLOW.LB.PER.HR(TZ) : : LPRINT TAB(D-1) :
4260 LPRINT USING "#.#####" : LETDOWN.POWER(TZ)
4270 '

```

```

4280 LPRINT TAB(A); " Charging Line";TAB(B+2); : LPRINT USING "####.#";CHARGING.ENTHALPY(TZ); : LPRINT TAB(C-2);
4290 LPRINT USING "##.#####"; CHARGING.FLOW.LB.PER.HR(TZ); : LPRINT TAB(D-2);
4300 LPRINT USING "##.#####"; CHARGING.POWER(TZ)
4310 '
4320 IF DATA.SETSZ=1 THEN 4360
4330 LPRINT TAB(A); " Pressurizer";TAB(B+2); : LPRINT USING "####.#";PZR.ENTHALPY(TZ); : LPRINT TAB(C-2);
4340 LPRINT USING "##.#####"; PZR.FLOW; : LPRINT TAB(D-2); : LPRINT USING "##.#####"; PZR.POWER
4350 '
4360 LPRINT TAB(A); " Pumps";TAB(D-2); : LPRINT USING "##.#####";RC.PUMP.POWER.BTU.PER.HR
4370 '
4380 LPRINT TAB(A); " Insulation Losses";TAB(D-1); : LPRINT USING "#.#####"; INSULATION.POWER(TZ)
4390 LPRINT TAB(D-1); "-----"
4400 '
4410 LPRINT TAB(A); " Power Dissipated";TAB(D-2); : LPRINT USING "##.#####"; OTHER.POWER(TZ);
4420 LPRINT TAB(E-1); : LPRINT USING "####.#";OTHER.POWER.MW(TZ) : LPRINT TAB(E-1); "-----"
4430 '
4440 LPRINT TAB(A); "REACTOR POWER";TAB(E-1); : LPRINT USING "####.#";REACTOR.POWER.MW(TZ)
4450 '
4460 IF DATA.SETSZ=1 THEN 4520
4470 IF TZ=1 AND LOOPSZ=2 THEN FT=22 ELSE IF TZ=1 AND LOOPSZ=3 THEN FT=16 ELSE IF TZ=1 AND LOOPSZ=4 THEN FT=10
4480 FOR KZ=1 TO FT : LPRINT : NEXT KZ
4490 NEXT
4500 '
4510 IF DATA.SETSZ=2 THEN 4540
4520 IF LOOPSZ=2 THEN FT=23 ELSE IF LOOPSZ=3 THEN FT=17 ELSE IF LOOPSZ=4 THEN FT=11
4530 FOR KZ=1 TO FT : LPRINT : NEXT
4540 LPRINT CHR$(27);CHR$(70); : GOTO 5210
4550 '
4560 '
4570 '
4580 ' *****
4590 ' * DATA SHEET *
4600 ' *****
4610 GOSUB 3630 : A=18 : B=A+1 : C=A+34 : D=A+43 : E=A+63 : F=A+97 : G=A+106 : LPRINT : LPRINT CHR$(15)
4620 '
4630 WIDTH "LPT1:",132
4640 '
4650 LPRINT TAB(A+47); "HEAT BALANCE DATA"
4660 '
4670 LPRINT : LPRINT : LPRINT TAB(B); "PLANT PARAMETERS:" : LPRINT
4680 '
4690 LPRINT TAB(B+1); "REACTOR COOLANT SYSTEM";
4700 LPRINT TAB(E+1); "REFLECTIVE INSULATION"
4710 LPRINT TAB(B+1); " Pump Power (MW each)"; : LPRINT TAB(E+1); " Inside Surface Area (sq ft)"
4720 LPRINT TAB(B+1); " Pump Efficiency (%)" ; : LPRINT TAB(E+1); " Heat Loss Coefficient (BTUs/hr sq ft)"
4730 LPRINT TAB(B+1); " Pressurizer Inside Diameter (inches)"
4740 LPRINT TAB(E+1); "NONREFLECTIVE INSULATION"
4750 LPRINT TAB(B+1); "STEAM GENERATORS"; : LPRINT TAB(E+1); " Inside Surface Area (sq ft)"
4760 LPRINT TAB(B+1); " Dome Inside Diameter (inches)"; : LPRINT TAB(E+1); " Thickness (inches)"
4770 LPRINT TAB(B+1); " Riser Outside Diameter (inches)";
4780 LPRINT TAB(E+1); " Thermal Conductivity (BTUs/hr ft F)";
4790 LPRINT TAB(B+1); " Number of Risers"
4800 LPRINT TAB(B+1); " Moisture Carry-over (%) in A"; : LPRINT TAB(E+1); "LICENSED THERMAL POWER (Mwt)"
4810 LPRINT TAB(B+1); " Moisture Carry-over (%) in B"

```

APPENDIX J: REPORTS.BAS

```

4820 LPRINT TAB(B+1); " Moisture Carry-over (%) in C"
4830 LPRINT TAB(B+1); " Moisture Carry-over (%) in D"
4840 "
4850 LPRINT : LPRINT
4860 "
4870 LPRINT TAB(B); "DATA:";TAB(C);"SET 1";TAB(D);"SET 2";TAB(F);"SET 1";TAB(G);"SET 2" : LPRINT
4880 LPRINT TAB(B+1);"TIME"; : LPRINT TAB(E+1);"TIME" : LPRINT
4890 "
4900 LPRINT TAB(B);"STEAM GENERATOR A";TAB(E);"STEAM GENERATOR B" : LPRINT
4910 "
4920 LPRINT TAB(B+1); "Steam Pressure (psia)";TAB(E+1);"Steam Pressure (psia)"
4930 LPRINT TAB(B+1); "Feedwater Flow (E6 lb/hr)";TAB(E+1);"Feedwater Flow (E6 lb/hr)"
4940 LPRINT TAB(B+1); "Feedwater Temperature (F)";TAB(E+1);"Feedwater Temperature (F)"
4950 LPRINT TAB(B+1); "Surface Blowdown (gpm)";TAB(E+1);"Surface Blowdown (gpm)"
4960 LPRINT TAB(B+1); "Bottom Blowdown (gpm)";TAB(E+1);"Bottom Blowdown (gpm)"
4970 LPRINT TAB(B+1); "Water Level (inches)";TAB(E+1);"Water Level (inches)"
4980 "
4990 "
5000 LPRINT : LPRINT : LPRINT TAB(B);"STEAM GENERATOR C"; : LPRINT TAB(E);"STEAM GENERATOR D" : LPRINT
5010 "
5020 LPRINT TAB(B+1); "Steam Pressure (psia)";TAB(E+1);"Steam Pressure (psia)"
5030 LPRINT TAB(B+1); "Feedwater Flow (E6 lb/hr)";TAB(E+1);"Feedwater Flow (E6 lb/hr)"
5040 LPRINT TAB(B+1); "Feedwater Temperature (F)";TAB(E+1);"Feedwater Temperature (F)"
5050 LPRINT TAB(B+1); "Surface Blowdown (gpm)";TAB(E+1);"Surface Blowdown (gpm)"
5060 LPRINT TAB(B+1); "Bottom Blowdown (gpm)";TAB(E+1);"Bottom Blowdown (gpm)"
5070 LPRINT TAB(B+1); "Water Level (inches)";TAB(E+1);"Water Level (inches)"
5080 "
5090 LPRINT : LPRINT : LPRINT TAB(B);"LETDOWN LINE"; : LPRINT TAB(E);"CHARGING LINE" : LPRINT
5100 "
5110 LPRINT TAB(B+1);"Flow (gpm)";TAB(E+1);"Flow (gpm)"
5120 LPRINT TAB(B+1);"Temperature (F)";TAB(E+1);"Temperature (F)"
5130 "
5140 LPRINT : LPRINT : LPRINT TAB(B);"PRESSURIZER";TAB(E);"REACTOR" : LPRINT
5150 "
5160 LPRINT TAB(B+1);"Pressure (psia)";TAB(E+1);"T ave (F)"
5170 LPRINT TAB(B+1);"Water Level (inches)";TAB(E+1);"T cold (F)"
5180 "
5190 IF X$="INPUT DATA" THEN GOTO 670
5200 FOR K1=1 TO 10 : LPRINT : NEXT K1
5210 CLS: LOCATE 13,20 : PRINT "LOADING INTRODUCTION MODULE INTO MEMORY" : X$="INTRO 1920" : CHAIN "INTRO"
5220 END

```

APPENDIX K

END.BAS


```

10      *****
20      ** END.BAS **
30      *****
40
50      * COMMON *
60
70      DEFDBL A-Z
80
90      REM Parameters
100     COMMON CARRYOVER.A,CARRYOVER.B,CARRYOVER.C,CARRYOVER.D,LOSS.COEFFICIENT,NONREFLECT.AREA,PZR.ID
110     COMMON RISER.NUMBER,RISER.OO,CONDUCTIVITY,INSULATION.THICKNESS,LIC.POWER,LOOPSZ,REFLECT.AREA
120     COMMON RC.PUMP.EFF,RC.PUMP.PWR,SG.ID,DOCKET$
130
140     REM Data
150     COMMON BOTTOM.BLOW(),CHARGING.FLOW(),CHARGING.TEMP(),FEEDWATER.FLOW(),FEEDWATER.TEMP(),LETDOWN.FLOW()
160     COMMON LETDOWN.TEMP(),PZR.PRES(),PZR.WATER.LEVEL(),STEAM.PRES(),T.AVE(),T.COLD(),TTIME(),TOP.BLOW()
170     COMMON SG.WATER.LEVEL(),DATA.SETSZ,DDATE$,PLANT$,UNIT$
180
190     REM Auxiliary Results
200     COMMON A,B,BOTTOM.BLOW.DENSITY(),DELTA.TIME,DRY.STEAM.ENTHALPY(),H,I,MOISTURE.ENTHALPY(),N,P,PSAT$
210     COMMON PZR.DENSITY(),PZR.ENTHALPY(),SG.CROSS.SECTION,T,TSAT,V,MZ,NZ,DATA.FLAG$,DIAGNOSTIC.FLAG$
220     COMMON PARAMETER.FLAG$,PRINT.FLAG$,RESULTS.FLAG$,X$,Y$,Z$
230
240     OPEN DOCKET$ AS #1 LEN=159 : FIELD #1, 32 AS P$, 1 AS U$, 125 AS X$, 1 AS PF$ : GET #1,1
250     PLANT$ = P$ : UNIT$ = U$ : PARAMETER.FLAG$ = PF$ : CLOSE #1
260
270     OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59
280     FIELD #1,32 AS P$, 1 AS U$, 20 AS O$, 3 AS OS$, 1 AS DF$, 2 AS DTS$ : GET #1,1 : DATA.FLAG$ = DF$ : CLOSE #1
290     IF DATA.FLAG$="C" THEN KILL "PLANT-ID."+DOCKET$
300
310     X$ = " "
320     FOR IZ=1 TO 32 : NON.PLANT$=RIGHT$(PLANT$,IZ)
330     IF NON.PLANT$(<>)X$ THEN PLANT$=LEFT$(PLANT$,32-IZ+1) : GOTO 350 ELSE X$=X$+" " : NEXT
340
350     OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=45
360     FIELD #1, 1 AS RF$ : GET #1,1 : RESULTS.FLAG$=RF$ : CLOSE 1 : IF RESULTS.FLAG$(<>)S" THEN RESULTS.FLAG$="C"
370     IF RESULTS.FLAG$="C" THEN KILL "RCS-CALC."+DOCKET$
380
390     CLS
400     IF PARAMETER.FLAG$="C" AND DATA.FLAG$="C" AND RESULTS.FLAG$="C" THEN 460
410     IF PARAMETER.FLAG$="S" AND DATA.FLAG$="C" AND RESULTS.FLAG$="C" THEN 490
420     IF PARAMETER.FLAG$="S" AND DATA.FLAG$="S" AND RESULTS.FLAG$="C" THEN 500
430     IF PARAMETER.FLAG$="S" AND DATA.FLAG$="S" AND RESULTS.FLAG$="S" THEN 510
440     LOCATE 13,34 : PRINT "CHECK FLAGS" : STOP
450
460     LOCATE 13,20 : PRINT "Neither parameters, data, nor results for"
470     LOCATE 14,20 : PRINT "Docket 50-";DOCKET$;" are stored on the diskette." : KILL DOCKET$ : GOTO 560
480
490     LOCATE 11,34 : PRINT "Parameters for" : GOTO 530
500     LOCATE 11,29 : PRINT "Parameters and data for" : GOTO 530
510     LOCATE 10,27 : PRINT "Parameters, data, and results"
520     LOCATE 11,27 : PRINT "for" : GOTO 530
530     LOCATE 13,INT((82-LEN(PLANT$+UNIT$)-11)/2) : PRINT PLANT$;" ";UNIT$ : LOCATE 14,38 : PRINT "50-";DOCKET$
540     LOCATE 16,27 : PRINT "are stored on the diskette."

```

APPENDIX K: END.BAS

```

550 '
560 LOCATE 25,22,0 : PRINT "To continue, press the "; : COLOR 16,7 : PRINT " space bar "; : COLOR 7,0
570 PRINT ".*";
580 Z$=INKEY$ : IF Z$(<)CHR$(32) THEN 580 ELSE 590
590 CLS : KEY 10,"CLS : SYSTEM"+CHR$(13)
600 '
610 IF PARAMETER.FLAG$="C" AND DATA.FLAG$="C" AND RESULTS.FLAG$="C" THEN 1110
620 IF PARAMETER.FLAG$="S" AND DATA.FLAG$="C" AND RESULTS.FLAG$="C" THEN 660
630 IF PARAMETER.FLAG$="S" AND DATA.FLAG$="S" THEN 750
640 '
650 CLS
660 GOSUB 1250
670 LOCATE 10,23 : PRINT "For Docket 50-";DOCKET$;*, you can:
680 LOCATE 12,25 : PRINT "(A) Save the parameters file, or
690 LOCATE 13,25 : PRINT "(B) Erase the parameters file,
700 LOCATE 15,23 : INPUT "Which would you like (A/B)? ",X$
710 '
720 IF X$="A" OR X$="a" THEN 1110
730 IF X$="B" OR X$="b" THEN GOSUB 880 : GOSUB 960 : GOTO 1110 ELSE GOSUB 1290 : GOTO 660
740 '
750 CLS
760 GOSUB 1250
770 LOCATE 10,23 : PRINT "For Docket 50-";DOCKET$;*, you can:
780 IF RESULTS.FLAG$="S" THEN 810 ELSE 790
790 LOCATE 12,25 : PRINT "(A) Save the data files, or"
800 LOCATE 13,25 : PRINT "(B) Erase the data files," : GOTO 830
810 LOCATE 12,25 : PRINT "(A) Save data and results files, or"
820 LOCATE 13,25 : PRINT "(B) Erase data and results files,"
830 LOCATE 15,23 : INPUT "Which would you like (A/B)? ",X$
840 '
850 IF X$="A" OR X$="a" THEN 1110
860 IF X$="B" OR X$="b" THEN GOSUB 880 : GOSUB 1060 : GOTO 1110 ELSE GOSUB 1290 : GOTO 760
870 '
880 CLS
890 GOSUB 1250
900 LOCATE 12,22 : PRINT "You issued a command to erase files."
910 LOCATE 13,22 : INPUT "Do you wish to proceed (Y/N)? ",X$
920 '
930 IF X$="Y" OR X$="y" THEN CLS : LOCATE 13,33 : PRINT "ERASING FILES" : RETURN
940 IF X$="N" OR X$="n" THEN 1110 ELSE GOSUB 1290 : GOTO 900
950 '
960 OPEN "PLANT-ID."+DOCKET$ AS #1 LEN=59 : CLOSE #1 : LOCATE 15,40 : PRINT "7"
970 OPEN "SG-DATA."+DOCKET$ AS #1 LEN=48 : CLOSE #1 : LOCATE 15,40 : PRINT "6"
980 OPEN "OC-DATA."+DOCKET$ AS #1 LEN=72 : CLOSE #1 : LOCATE 15,40 : PRINT "5"
990 OPEN "RCS-CALC."+DOCKET$ AS #1 LEN=33 : CLOSE #1 : LOCATE 15,40 : PRINT "4"
1000 OPEN "SG-CALC."+DOCKET$ AS #1 LEN=104 : CLOSE #1 : LOCATE 15,40 : PRINT "3"
1010 OPEN "OC-CALC."+DOCKET$ AS #1 LEN=88 : CLOSE #1 : LOCATE 15,40 : PRINT "2"
1020 KILL "PLANT-ID."+DOCKET$ : KILL "SG-DATA."+DOCKET$ : KILL "OC-DATA."+DOCKET$ : LOCATE 15,40 : PRINT "1"
1030 KILL "RCS-CALC."+DOCKET$ : KILL "SG-CALC."+DOCKET$ : KILL "OC-CALC."+DOCKET$
1040 KILL DOCKET$ : RETURN
1050 '
1060 KILL "PLANT-ID."+DOCKET$ : KILL "SG-DATA."+DOCKET$ : KILL "OC-DATA."+DOCKET$
1070 IF RESULTS.FLAG$(<)"S" THEN RETURN
1080 KILL "RCS-CALC."+DOCKET$ : KILL "SG-CALC."+DOCKET$ : KILL "OC-CALC."+DOCKET$ : RETURN

```

```
1090 '
1100 '
1110 CLS
1120 GOSUB 1250
1130 LOCATE 10,18 : PRINT "You can now:"
1140 LOCATE 12,20 : PRINT "(A) Return to the beginning of the program, or"
1150 LOCATE 13,20 : PRINT "(B) Exit from TPDWR2."
1160 LOCATE 15,18 : INPUT "Which would you like (A/B)? ",X$
1170 '
1180 IF X$="A" OR X$="a" THEN CHAIN "INTRO"
1190 IF X$="B" OR X$="b" THEN GOTO 1210 ELSE GOSUB 1290 : GOTO 1120
1200 '
1210 CLS : LOCATE 12,30 : PRINT "TPDWR2 is terminated."
1220 LOCATE 14,25 : PRINT "To return to DOS, press "; : COLOR 0,7 : PRINT " F10 "; : COLOR 7,0 : PRINT ".*"
1230 GOTO 1300
1240 '
1250 LOCATE 25,21 : PRINT "After responding, strike the "; : COLOR 16,7
1260 PRINT " *";CHR$(17)+CHR$(196)+CHR$(217);" *";
1270 COLOR 7,0 : PRINT " key."; : RETURN
1280 '
1290 LOCATE 18,29 : COLOR 0,7 : PRINT " Please type Y or N! " : COLOR 7,0 : RETURN
1300 LOCATE 24,1
1310 END
```

APPENDIX L

PROGRAM PERFORMANCE TESTS

TABLE OF CONTENTS

| | Page |
|----------------------------|------|
| | ---- |
| Program Performance Test 1 | J- 1 |
| Program Performance Test 2 | J- 3 |
| Program Performance Test 3 | J- 6 |
| Program Performance Test 4 | J- 8 |
| Program Performance Test 5 | J-11 |
| Program Performance Test 6 | J-13 |
| McGuire 1 | J-16 |

HEAT BALANCE DATA
Program Performance Test 1
September 30, 1985

PLANT PARAMETERS:

REACTOR COOLANT SYSTEM

| | |
|--------------------------------------|------|
| Pump Power (MW each) | 6.1 |
| Pump Efficiency (%) | 90.0 |
| Pressurizer Inside Diameter (inches) | 75.1 |

REFLECTIVE INSULATION

| | |
|---------------------------------------|-------|
| Inside Surface Area (sq ft) | 9,347 |
| Heat Loss Coefficient (BTUs/hr sq ft) | 55.00 |

STEAM GENERATORS

| | |
|---------------------------------|--------|
| Dome Inside Diameter (inches) | 135.90 |
| Riser Outside Diameter (inches) | 19.90 |
| Number of Risers | 12 |
| Moisture Carry-over (%) in A | 0.125 |
| Moisture Carry-over (%) in B | 0.125 |

NONREFLECTIVE INSULATION

| | |
|-------------------------------------|-------|
| Inside Surface Area (sq ft) | 6,328 |
| Thickness (inches) | 4.0 |
| Thermal Conductivity (BTUs/hr ft F) | 0.035 |

LICENSED THERMAL POWER (Mwt)

1950

DATA:

| | |
|------|------|
| TIME | 1:00 |
|------|------|

| | |
|------|------|
| TIME | 1:30 |
|------|------|

STEAM GENERATOR A

| | |
|---------------------------|--------|
| Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.000 |
| Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 10.0 |
| Bottom Blowdown (gpm) | 100.0 |
| Water Level (inches) | 140.0 |

STEAM GENERATOR B

| | |
|---------------------------|-------|
| Steam Pressure (psia) | 900.0 |
| Feedwater Flow (E6 lb/hr) | 4.000 |
| Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 10.0 |
| Bottom Blowdown (gpm) | 100.0 |
| Water Level (inches) | 140.0 |

LETDOWN LINE

| | |
|-----------------|-------|
| Flow (gpm) | 100.0 |
| Temperature (F) | 540.0 |

CHARGING LINE

| | |
|-----------------|-------|
| Flow (gpm) | 50.0 |
| Temperature (F) | 500.0 |

PRESSURIZER

| | |
|----------------------|--------|
| Pressure (psia) | 2300.0 |
| Water Level (inches) | 120.0 |

REACTOR

| | |
|------------|-------|
| T ave (F) | 585.0 |
| T cold (F) | 550.0 |

HEAT BALANCE
Program Performance Test 1
September 30, 1985

| DATA SET 1 OF 1 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (Mwt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 3.957 | 4.717 | |
| Feedwater | 419.5 | -4.000 | -1.678 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.0599 | 896.2 |
| STEAM GENERATOR B | | | | |
| Steam | 1195.6 | 3.956 | 4.730 | |
| Feedwater | 419.4 | -4.000 | -1.678 | |
| Surface Blowdown | 526.7 | 0.00378 | 0.00199 | |
| Bottom Blowdown | 471.4 | 0.03998 | 0.01884 | |
| | | | ----- | |
| Power Dissipated | | | 3.0732 | 900.1 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.03812 | 0.02039 | |
| Charging Line | 487.5 | -0.01997 | -0.00973 | |
| Pumps | | | -0.03749 | |
| Insulation Losses | | | 0.00084 | |
| | | | ----- | |
| Power Dissipated | | | -0.02600 | -7.6 |
| | | | | ----- |
| REACTOR POWER | | | | 1788.6 |

HEAT BALANCE DATA
 Program Performance Test 2
 September 30, 1985

PLANT PARAMETERS:

REACTOR COOLANT SYSTEM

Pump Power (MW each) 6.1
 Pump Efficiency (%) 90.0
 Pressurizer Inside Diameter (inches) 75.1

REFLECTIVE INSULATION

Inside Surface Area (sq ft) 9,347
 Heat Loss Coefficient (BTUs/hr sq ft) 55.00

STEAM GENERATORS

Dome Inside Diameter (inches) 135.90
 Riser Outside Diameter (inches) 19.90
 Number of Risers 12
 Moisture Carry-over (%) in A 0.125
 Moisture Carry-over (%) in B 0.125

NONREFLECTIVE INSULATION

Inside Surface Area (sq ft) 6,328
 Thickness (inches) 4.0
 Thermal Conductivity (BTUs/hr ft F) 0.035

LICENSED THERMAL POWER (Mwt) 1950

DATA:

SET 1 SET 2

SET 1 SET 2

TIME 1300 1320

TIME 1300 1320

STEAM GENERATOR A

Steam Pressure (psia) 1000.0 1000.0
 Feedwater Flow (E6 lb/hr) 4.000 4.000
 Feedwater Temperature (F) 440.0 440.0
 Surface Blowdown (gpm) 10.0 10.0
 Bottom Blowdown (gpm) 100.0 100.0
 Water Level (inches) 140.0 140.0

STEAM GENERATOR B

Steam Pressure (psia) 1000.0 1000.0
 Feedwater Flow (E6 lb/hr) 3.900 3.900
 Feedwater Temperature (F) 440.0 440.0
 Surface Blowdown (gpm) 10.0 10.0
 Bottom Blowdown (gpm) 100.0 100.0
 Water Level (inches) 140.0 70.0

LETDOWN LINE

Flow (gpm) 100.0 100.0
 Temperature (F) 540.0 540.0

CHARGING LINE

Flow (gpm) 50.0 50.0
 Temperature (F) 500.0 500.0

PRESSURIZER

Pressure (psia) 2300.0 2300.0
 Water Level (inches) 120.0 120.0

REACTOR

T ave (F) 585.0 585.0
 T cold (F) 550.0 550.0

HEAT BALANCE
 Program Performance Test 2
 September 30, 1985

| DATA SET 1 OF 2 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (Mwt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 3.957 | 4.717 | |
| Feedwater | 419.5 | -4.000 | -1.678 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.0599 | 896.2 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 3.921 | 4.675 | |
| Feedwater | 419.5 | -3.900 | -1.636 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.0600 | 896.2 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.03812 | 0.02039 | |
| Charging Line | 487.5 | -0.01997 | -0.00973 | |
| Pressurizer | 707.2 | 0.00000 | 0.00000 | |
| Pumps | | | -0.03749 | |
| Insulation Losses | | | 0.00084 | |
| | | | ----- | |
| Power Dissipated | | | -0.02600 | -7.6 |
| | | | | ----- |
| REACTOR POWER | | | | 1784.7 |

HEAT BALANCE
Program Performance Test 2
September 30, 1985

| DATA SET 2 OF 2 1320 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (Mwt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 3.957 | 4.717 | |
| Feedwater | 419.5 | -4.000 | -1.678 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.0599 | 896.2 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 3.921 | 4.675 | |
| Feedwater | 419.5 | -3.900 | -1.636 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.0600 | 896.2 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.03812 | 0.02039 | |
| Charging Line | 487.5 | -0.01997 | -0.00973 | |
| Pressurizer | 707.2 | 0.00000 | 0.00000 | |
| Pumps | | | -0.03749 | |
| Insulation Losses | | | 0.00084 | |
| | | | ----- | |
| Power Dissipated | | | -0.02600 | -7.6 |
| | | | | ----- |
| REACTOR POWER | | | | 1784.7 |

APPENDIX L: PROGRAM PERFORMANCE TESTS

HEAT BALANCE DATA
 Program Performance Test 3
 September 30, 1985

PLANT PARAMETERS:

| | | | |
|--------------------------------------|-------|---------------------------------------|-------|
| REACTOR COOLANT SYSTEM | | REFLECTIVE INSULATION | |
| Pump Power (MW each) | 2.0 | Inside Surface Area (sq ft) | 6,465 |
| Pump Efficiency (%) | 90.0 | Heat Loss Coefficient (BTUs/hr sq ft) | 55.00 |
| Pressurizer Inside Diameter (inches) | 59.6 | | |
| STEAM GENERATORS | | NONREFLECTIVE INSULATION | |
| Dome Inside Diameter (inches) | 94.30 | Inside Surface Area (sq ft) | 4,563 |
| Riser Outside Diameter (inches) | 13.80 | Thickness (inches) | 4.0 |
| Number of Risers | 12 | Thermal Conductivity (BTUs/hr ft F) | 0.035 |
| Moisture Carry-over (%) in A | 0.000 | LICENSED THERMAL POWER (Mwt) | 975 |
| Moisture Carry-over (%) in B | 5.000 | | |
| Moisture Carry-over (%) in C | 9.999 | | |

DATA:

| | | | |
|------|------|------|------|
| TIME | 1300 | TIME | 1300 |
|------|------|------|------|

STEAM GENERATOR A

| | |
|---------------------------|--------|
| Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 1.500 |
| Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 10.0 |
| Bottom Blowdown (gpm) | 100.0 |
| Water Level (inches) | 120.0 |

STEAM GENERATOR B

| | |
|---------------------------|--------|
| Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 1.500 |
| Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 10.0 |
| Bottom Blowdown (gpm) | 100.0 |
| Water Level (inches) | 120.0 |

STEAM GENERATOR C

| | |
|---------------------------|--------|
| Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 1.500 |
| Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 10.0 |
| Bottom Blowdown (gpm) | 100.0 |
| Water Level (inches) | 120.0 |

LETDOWN LINE

| | |
|-----------------|-------|
| Flow (gpm) | 50.0 |
| Temperature (F) | 540.0 |

CHARGING LINE

| | |
|-----------------|-------|
| Flow (gpm) | 25.0 |
| Temperature (F) | 500.0 |

PRESSURIZER

| | |
|----------------------|--------|
| Pressure (psia) | 2300.0 |
| Water Level (inches) | 100.0 |

REACTOR

| | |
|------------|-------|
| T ave (F) | 585.0 |
| T cold (F) | 550.0 |

HEAT BALANCE
Program Performance Test 3
September 30, 1985

| DATA SET 1 OF 1 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (MWt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.9 | 1.457 | 1.738 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1294 | 330.8 |
| STEAM GENERATOR B | | | | |
| Steam | 1160.4 | 1.457 | 1.690 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.0821 | 316.9 |
| STEAM GENERATOR C | | | | |
| Steam | 1127.9 | 1.457 | 1.643 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.0347 | 303.0 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.01906 | 0.01019 | |
| Charging Line | 487.5 | -0.00998 | -0.00487 | |
| Pumps | | | -0.01844 | |
| Insulation Losses | | | 0.00059 | |
| | | | ----- | |
| Power Dissipated | | | -0.01252 | -3.7 |
| | | | | ----- |
| REACTOR POWER | | | | 947.1 |

APPENDIX L: PROGRAM PERFORMANCE TESTS

HEAT BALANCE DATA
 Program Performance Test 4
 September 30, 1985

PLANT PARAMETERS:

REACTOR COOLANT SYSTEM

Pump Power (MW each) 2.0
 Pump Efficiency (%) 90.0
 Pressurizer Inside Diameter (inches) 59.6

REFLECTIVE INSULATION

Inside Surface Area (sq ft) 6,465
 Heat Loss Coefficient (BTUs/hr sq ft) 55.00

STEAM GENERATORS

Down Inside Diameter (inches) 94.30
 Riser Outside Diameter (inches) 13.80
 Number of Risers 12
 Moisture Carry-over (%) in A 0.125
 Moisture Carry-over (%) in B 0.125
 Moisture Carry-over (%) in C 0.125

NONREFLECTIVE INSULATION

Inside Surface Area (sq ft) 4,563
 Thickness (inches) 4.0
 Thermal Conductivity (BTUs/hr ft F) 0.035

LICENSED THERMAL POWER (Mwt) 975

DATA: SET 1 SET 2
 TIME 1300 1320

SET 1 SET 2
 TIME 1300 1320

STEAM GENERATOR A

Steam Pressure (psia) 1000.0 1000.0
 Feedwater Flow (E6 lb/hr) 1.500 1.500
 Feedwater Temperature (F) 440.0 440.0
 Surface Blowdown (gpm) 10.0 10.0
 Bottom Blowdown (gpm) 100.0 100.0
 Water Level (inches) 120.0 120.0

STEAM GENERATOR B

Steam Pressure (psia) 1000.0 1000.0
 Feedwater Flow (E6 lb/hr) 1.500 1.500
 Feedwater Temperature (F) 440.0 440.0
 Surface Blowdown (gpm) 10.0 10.0
 Bottom Blowdown (gpm) 100.0 100.0
 Water Level (inches) 120.0 120.0

STEAM GENERATOR C

Steam Pressure (psia) 1000.0 1000.0
 Feedwater Flow (E6 lb/hr) 1.500 1.500
 Feedwater Temperature (F) 440.0 440.0
 Surface Blowdown (gpm) 10.0 10.0
 Bottom Blowdown (gpm) 100.0 100.0
 Water Level (inches) 120.0 120.0

LETDOWN LINE

Flow (gpm) 50.0 50.0
 Temperature (F) 540.0 540.0

CHARGING LINE

Flow (gpm) 25.0 25.0
 Temperature (F) 500.0 500.0

PRESSURIZER

Pressure (psia) 2300.0 2300.0
 Water Level (inches) 150.0 50.0

REACTOR

T ave (F) 585.0 585.0
 T cold (F) 550.0 500.0

HEAT BALANCE
Program Performance Test 4
September 30, 1985

| DATA SET 1 OF 2 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (MWt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.01906 | 0.01019 | |
| Charging Line | 487.5 | -0.00998 | -0.00487 | |
| Pressurizer | 707.2 | -0.01776 | -0.01256 | |
| Pumps | | | -0.01844 | |
| Insulation Losses | | | 0.00059 | |
| | | | ----- | |
| Power Dissipated | | | -0.02508 | -7.3 |
| | | | | ----- |
| REACTOR POWER | | | | 984.0 |

HEAT BALANCE
 Program Performance Test 4
 September 30, 1985

| DATA SET 2 OF 2 1320 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (Mwt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.1 | 1.457 | 1.736 | |
| Feedwater | 419.5 | -1.500 | -0.629 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 1.1282 | 330.4 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.01906 | 0.01019 | |
| Charging Line | 487.5 | -0.00998 | -0.00487 | |
| Pressurizer | 707.2 | -0.01776 | -0.01256 | |
| Pumps | | | -0.01844 | |
| Insulation Losses | | | 0.00059 | |
| | | | ----- | |
| Power Dissipated | | | -0.02508 | -7.3 |
| | | | | ----- |
| REACTOR POWER | | | | 984.0 |

HEAT BALANCE DATA
 Program Performance Test 5
 September 30, 1985

PLANT PARAMETERS:

| | | | |
|--------------------------------------|--------|---------------------------------------|--------|
| REACTOR COOLANT SYSTEM | | REFLECTIVE INSULATION | |
| Pump Power (MW each) | 6.1 | Inside Surface Area (sq ft) | 4,792 |
| Pump Efficiency (%) | 90.0 | Heat Loss Coefficient (BTUs/hr sq ft) | 55.00 |
| Pressurizer Inside Diameter (inches) | 94.6 | | |
| STEAM GENERATORS | | NONREFLECTIVE INSULATION | |
| Dome Inside Diameter (inches) | 135.90 | Inside Surface Area (sq ft) | 12,657 |
| Riser Outside Diameter (inches) | 19.90 | Thickness (inches) | 4.0 |
| Number of Risers | 12 | Thermal Conductivity (BTUs/hr ft F) | 0.035 |
| Moisture Carry-over (%) in A | 0.125 | | |
| Moisture Carry-over (%) in B | 0.125 | LICENSED THERMAL POWER (Mwt) | 3900 |
| Moisture Carry-over (%) in C | 0.125 | | |
| Moisture Carry-over (%) in D | 0.125 | | |

DATA:

| | | | |
|---------------------------|--------|---------------------------|--------|
| TIME | 1300 | TIME | 1300 |
| STEAM GENERATOR A | | STEAM GENERATOR B | |
| Steam Pressure (psia) | 1000.0 | Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | Feedwater Flow (E6 lb/hr) | 4.450 |
| Feedwater Temperature (F) | 440.0 | Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 0.0 | Surface Blowdown (gpm) | 0.0 |
| Bottom Blowdown (gpm) | 0.0 | Bottom Blowdown (gpm) | 500.0 |
| Water Level (inches) | 140.0 | Water Level (inches) | 140.0 |
| STEAM GENERATOR C | | STEAM GENERATOR D | |
| Steam Pressure (psia) | 1000.0 | Steam Pressure (psia) | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | Feedwater Flow (E6 lb/hr) | 4.450 |
| Feedwater Temperature (F) | 440.0 | Feedwater Temperature (F) | 440.0 |
| Surface Blowdown (gpm) | 500.0 | Surface Blowdown (gpm) | 500.0 |
| Bottom Blowdown (gpm) | 0.0 | Bottom Blowdown (gpm) | 500.0 |
| Water Level (inches) | 140.0 | Water Level (inches) | 140.0 |
| LETDOWN LINE | | CHARGING LINE | |
| Flow (gpm) | 200.0 | Flow (gpm) | 100.0 |
| Temperature (F) | 540.0 | Temperature (F) | 500.0 |
| PRESSURIZER | | REACTOR | |
| Pressure (psia) | 2300.0 | T ave (F) | 585.0 |
| Water Level (inches) | 100.0 | T cold (F) | 550.0 |

HEAT BALANCE
Program Performance Test 5
September 30, 1985

| DATA SET 1 OF 1 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (MWt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 4.450 | 5.305 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.7 | 0.00000 | 0.00000 | |
| | | | ----- | |
| Power Dissipated | | | 3.4384 | 1007.0 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 4.251 | 5.068 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.7 | 0.19871 | 0.09513 | |
| | | | ----- | |
| Power Dissipated | | | 3.2966 | 965.5 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.1 | 4.264 | 5.084 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.18574 | 0.10078 | |
| Bottom Blowdown | 478.7 | 0.00000 | 0.00000 | |
| | | | ----- | |
| Power Dissipated | | | 3.3177 | 971.7 |
| STEAM GENERATOR D | | | | |
| Steam | 1192.1 | 4.066 | 4.847 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.18574 | 0.10078 | |
| Bottom Blowdown | 478.7 | 0.19871 | 0.09513 | |
| | | | ----- | |
| Power Dissipated | | | 3.1760 | 930.2 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.07624 | 0.04077 | |
| Charging Line | 487.5 | -0.03994 | -0.01947 | |
| Pumps | | | -0.07498 | |
| Insulation Losses | | | 0.00091 | |
| | | | ----- | |
| Power Dissipated | | | -0.05277 | -15.5 |
| REACTOR POWER | | | | ----- |
| | | | | 3858.9 |

HEAT BALANCE DATA
 Program Performance Test 6
 September 30, 1985

PLANT PARAMETERS:

REACTOR COOLANT SYSTEM

| | |
|--------------------------------------|------|
| Pump Power (MW each) | 6.1 |
| Pump Efficiency (%) | 90.0 |
| Pressurizer Inside Diameter (inches) | 94.6 |

REFLECTIVE INSULATION

| | |
|---------------------------------------|-------|
| Inside Surface Area (sq ft) | 4,792 |
| Heat Loss Coefficient (BTUs/hr sq ft) | 55.00 |

STEAM GENERATORS

| | |
|---------------------------------|--------|
| Dome Inside Diameter (inches) | 135.90 |
| Riser Outside Diameter (inches) | 19.90 |
| Number of Risers | 12 |
| Moisture Carry-over (%) in A | 0.125 |
| Moisture Carry-over (%) in B | 0.125 |
| Moisture Carry-over (%) in C | 0.125 |
| Moisture Carry-over (%) in D | 0.125 |

NONREFLECTIVE INSULATION

| | |
|-------------------------------------|--------|
| Inside Surface Area (sq ft) | 12,657 |
| Thickness (inches) | 4.0 |
| Thermal Conductivity (BTUs/hr ft F) | 0.035 |

LICENSED THERMAL POWER (Mwt)

3900

| DATA: | SET 1 | SET 2 | | SET 1 | SET 2 |
|-------|-------|-------|------|-------|-------|
| TIME | 1300 | 1320 | TIME | 1300 | 1320 |

STEAM GENERATOR A

| | | |
|---------------------------|--------|--------|
| Steam Pressure (psia) | 1000.0 | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | 4.450 |
| Feedwater Temperature (F) | 460.0 | 460.0 |
| Surface Blowdown (gpm) | 10.0 | 10.0 |
| Bottom Blowdown (gpm) | 100.0 | 100.0 |
| Water Level (inches) | 140.0 | 140.0 |

STEAM GENERATOR B

| | | |
|---------------------------|--------|--------|
| Steam Pressure (psia) | 1000.0 | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | 4.450 |
| Feedwater Temperature (F) | 440.0 | 440.0 |
| Surface Blowdown (gpm) | 10.0 | 10.0 |
| Bottom Blowdown (gpm) | 100.0 | 100.0 |
| Water Level (inches) | 140.0 | 140.0 |

STEAM GENERATOR C

| | | |
|---------------------------|--------|--------|
| Steam Pressure (psia) | 1000.0 | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | 4.450 |
| Feedwater Temperature (F) | 420.0 | 420.0 |
| Surface Blowdown (gpm) | 10.0 | 10.0 |
| Bottom Blowdown (gpm) | 100.0 | 100.0 |
| Water Level (inches) | 140.0 | 140.0 |

STEAM GENERATOR D

| | | |
|---------------------------|--------|--------|
| Steam Pressure (psia) | 1000.0 | 1000.0 |
| Feedwater Flow (E6 lb/hr) | 4.450 | 4.450 |
| Feedwater Temperature (F) | 400.0 | 400.0 |
| Surface Blowdown (gpm) | 10.0 | 10.0 |
| Bottom Blowdown (gpm) | 100.0 | 100.0 |
| Water Level (inches) | 140.0 | 140.0 |

LETDOWN LINE

| | | |
|-----------------|-------|-------|
| Flow (gpm) | 200.0 | 0.0 |
| Temperature (F) | 540.0 | 540.0 |

CHARGING LINE

| | | |
|-----------------|-------|-------|
| Flow (gpm) | 200.0 | 0.0 |
| Temperature (F) | 500.0 | 500.0 |

PRESSURIZER

| | | |
|----------------------|--------|--------|
| Pressure (psia) | 2300.0 | 2300.0 |
| Water Level (inches) | 100.0 | 100.0 |

REACTOR

| | | |
|------------|-------|-------|
| T ave (F) | 585.0 | 585.0 |
| T cold (F) | 550.0 | 550.0 |

APPENDIX L: PROGRAM PERFORMANCE TESTS

HEAT BALANCE
Program Performance Test 6
September 30, 1985

| DATA SET 1 OF 2 1300 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (MWt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 4.407 | 5.254 | |
| Feedwater | 441.7 | -4.450 | -1.966 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 490.5 | 0.03929 | 0.01927 | |
| | | | ----- | |
| Power Dissipated | | | 3.3092 | 969.2 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 4.407 | 5.253 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.4076 | 998.0 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.1 | 4.406 | 5.253 | |
| Feedwater | 397.5 | -4.450 | -1.769 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 467.1 | 0.04017 | 0.01877 | |
| | | | ----- | |
| Power Dissipated | | | 3.5043 | 1026.3 |
| STEAM GENERATOR D | | | | |
| Steam | 1192.1 | 4.406 | 5.252 | |
| Feedwater | 376.0 | -4.450 | -1.673 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 455.7 | 0.04059 | 0.01850 | |
| | | | ----- | |
| Power Dissipated | | | 3.5996 | 1054.2 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.07624 | 0.04077 | |
| Charging Line | 487.5 | -0.07987 | -0.03894 | |
| Pressurizer | 707.2 | 0.00000 | 0.00000 | |
| Pumps | | | -0.07498 | |
| Insulation Losses | | | 0.00091 | |
| | | | ----- | |
| Power Dissipated | | | -0.07224 | -21.2 |
| | | | ----- | |
| REACTOR POWER | | | | 4026.6 |

HEAT BALANCE
Program Performance Test 6
September 30, 1985

| DATA SET 2 OF 2 1320 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (MWt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.1 | 4.407 | 5.254 | |
| Feedwater | 441.7 | -4.450 | -1.966 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 490.5 | 0.03929 | 0.01927 | |
| | | | ----- | |
| Power Dissipated | | | 3.3092 | 969.2 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.1 | 4.407 | 5.253 | |
| Feedwater | 419.5 | -4.450 | -1.867 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 478.7 | 0.03974 | 0.01903 | |
| | | | ----- | |
| Power Dissipated | | | 3.4076 | 998.0 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.1 | 4.406 | 5.253 | |
| Feedwater | 397.5 | -4.450 | -1.769 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 467.1 | 0.04017 | 0.01877 | |
| | | | ----- | |
| Power Dissipated | | | 3.5043 | 1026.3 |
| STEAM GENERATOR D | | | | |
| Steam | 1192.1 | 4.406 | 5.252 | |
| Feedwater | 376.0 | -4.450 | -1.673 | |
| Surface Blowdown | 542.6 | 0.00371 | 0.00202 | |
| Bottom Blowdown | 455.7 | 0.04059 | 0.01850 | |
| | | | ----- | |
| Power Dissipated | | | 3.5996 | 1054.2 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 534.8 | 0.00000 | 0.00000 | |
| Charging Line | 487.5 | 0.00000 | 0.00000 | |
| Pressurizer | 707.2 | 0.00000 | 0.00000 | |
| Pumps | | | -0.07498 | |
| Insulation Losses | | | 0.00091 | |
| | | | ----- | |
| Power Dissipated | | | -0.07407 | -21.7 |
| | | | ----- | |
| REACTOR POWER | | | | 4026.1 |

APPENDIX L: PROGRAM PERFORMANCE TESTS

HEAT BALANCE DATA

McGuire 1

June 6, 1984

PLANT PARAMETERS:

REACTOR COOLANT SYSTEM

Pump Power (MW each) 5.3
 Pump Efficiency (%) 90.0
 Pressurizer Inside Diameter (inches) 90.5

REFLECTIVE INSULATION

Inside Surface Area (sq ft) 15,958
 Heat Loss Coefficient (BTUs/hr sq ft) 55.00

STEAM GENERATORS

Dome Inside Diameter (inches) 130.00
 Riser Outside Diameter (inches) 19.00
 Number of Risers 12
 Moisture Carry-over (%) in A 0.070
 Moisture Carry-over (%) in B 0.070
 Moisture Carry-over (%) in C 0.070
 Moisture Carry-over (%) in D 0.070

NONREFLECTIVE INSULATION

Inside Surface Area (sq ft) 0
 Thickness (inches) 0.0
 Thermal Conductivity (BTUs/hr ft F) 0.000

LICENSED THERMAL POWER (Mwt) 3411

DATA:

TIME 1015

TIME 1015

STEAM GENERATOR A

Steam Pressure (psia) 999.6
 Feedwater Flow (E6 lb/hr) 3.853
 Feedwater Temperature (F) 440.3
 Surface Blowdown (gpm) 0.0
 Bottom Blowdown (gpm) 0.0
 Water Level (inches) 149.1

STEAM GENERATOR B

Steam Pressure (psia) 1001.0
 Feedwater Flow (E6 lb/hr) 3.684
 Feedwater Temperature (F) 440.0
 Surface Blowdown (gpm) 0.0
 Bottom Blowdown (gpm) 22.8
 Water Level (inches) 149.1

STEAM GENERATOR C

Steam Pressure (psia) 1001.3
 Feedwater Flow (E6 lb/hr) 3.864
 Feedwater Temperature (F) 440.2
 Surface Blowdown (gpm) 0.0
 Bottom Blowdown (gpm) 47.0
 Water Level (inches) 149.1

STEAM GENERATOR D

Steam Pressure (psia) 999.5
 Feedwater Flow (E6 lb/hr) 3.666
 Feedwater Temperature (F) 440.3
 Surface Blowdown (gpm) 0.0
 Bottom Blowdown (gpm) 0.0
 Water Level (inches) 149.1

LETDOWN LINE

Flow (gpm) 76.0
 Temperature (F) 558.5

CHARGING LINE

Flow (gpm) 86.0
 Temperature (F) 519.8

PRESSURIZER

Pressure (psia) 2238.5
 Water Level (inches) 120.0

REACTOR

T ave (F) 587.5
 T cold (F) 559.2

HEAT BALANCE
McGuire 1
June 6, 1984

| DATA SET 1 OF 1 1015 hours | ENTHALPY (BTUs/lb) | FLOW (E6 lb/hr) | POWER (E9 BTUs/hr) | POWER (Mwt) |
|-------------------------------|-----------------------|--------------------|-----------------------|----------------|
| STEAM GENERATOR A | | | | |
| Steam | 1192.5 | 3.853 | 4.595 | |
| Feedwater | 419.8 | -3.853 | -1.617 | |
| Surface Blowdown | 542.5 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.9 | 0.00000 | 0.00000 | |
| | | | ----- | |
| Power Dissipated | | | 2.9773 | 872.0 |
| STEAM GENERATOR B | | | | |
| Steam | 1192.4 | 3.675 | 4.382 | |
| Feedwater | 419.5 | -3.684 | -1.545 | |
| Surface Blowdown | 542.7 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.8 | 0.00906 | 0.00434 | |
| | | | ----- | |
| Power Dissipated | | | 2.8412 | 832.1 |
| STEAM GENERATOR C | | | | |
| Steam | 1192.4 | 3.845 | 4.585 | |
| Feedwater | 419.7 | -3.864 | -1.622 | |
| Surface Blowdown | 542.8 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.9 | 0.01867 | 0.00894 | |
| | | | ----- | |
| Power Dissipated | | | 2.9726 | 870.6 |
| STEAM GENERATOR D | | | | |
| Steam | 1192.5 | 3.666 | 4.372 | |
| Feedwater | 419.8 | -3.666 | -1.539 | |
| Surface Blowdown | 542.5 | 0.00000 | 0.00000 | |
| Bottom Blowdown | 478.9 | 0.00000 | 0.00000 | |
| | | | ----- | |
| Power Dissipated | | | 2.8328 | 829.6 |
| OTHER COMPONENTS | | | | |
| Letdown Line | 557.8 | 0.02822 | 0.01574 | |
| Charging Line | 510.6 | -0.03358 | -0.01715 | |
| Pumps | | | -0.06515 | |
| Insulation Losses | | | 0.00088 | |
| | | | ----- | |
| Power Dissipated | | | -0.06567 | -19.2 |
| | | | | ----- |
| REACTOR POWER | | | | 3385.1 |