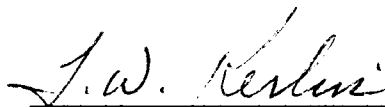


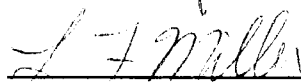
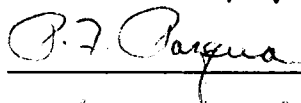
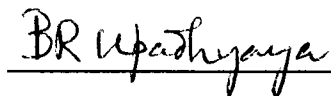
To the Graduate Council:

I am submitting herewith a thesis written by Mounir Jamil Mneimneh entitled "Modular Modeling." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Nuclear Engineering.



T. W. Kerlin, Major Professor

We have read this thesis
and recommend its acceptance:



Accepted for the Council:



The Graduate School

MODULAR MODELING

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Mounir Jamil Mneimneh

March 1985

ACKNOWLEDGMENTS

The author would like to thank Dr. T. W. Kerlin, the major professor, for his guidance and valuable suggestions during the development of this thesis.

The help of Dr. P. F. Pasqua, Head of the Nuclear Engineering Department, in obtaining a Teaching Assistantship during the graduate studies at The University of Tennessee is greatly appreciated.

The author would like to thank his wife, Susanne, for her support during the graduate studies.

ABSTRACT

The Modular Modeling Program is a computer package that was developed to simulate operational transients in a typical pressurized water reactor. The program is made up of a simulator, a library and a driver. The simulator functions as a simulation package. The library contains theoretical models for several components in a typical pressurized water reactor coolant loop. The lumped parameter modeling approach was used to develop the theoretical models. The driver functions to merge the simulator and theoretical models in the library into one large file.

The Modular Modeling Program provides the user with an opportunity to select models of various complexity from the library, construct a system of interest and execute the case for a given perturbation. The formulation of these models allows proper interconnection.

The Modular Modeling Program is adequate for mini and microcomputers. The fact that mini and microcomputers have limited memory makes the program an attractive alternative to the nonmodular approach.

TABLE OF CONTENTS

CHAPTER	PAGE
I. INTRODUCTION	1
General Consideration	1
Purpose and Importance of This Study	4
Organization of Thesis	5
II. THE MODULAR MODELING SYSTEM AND THE MODULAR MODELING PROGRAM	7
Introduction	7
The Modular Modeling System	8
The Modular Modeling Program	11
Summary	13
III. DESCRIPTION OF THE PHYSICAL SYSTEM	15
The System	15
Reactor	15
Steam Generator	19
Pressurizer	19
Pressure control system	23
IV. THEORETICAL MODELS	27
Introduction	27
Core	27
Pressurizer Theoretical Model	38
Steam Generator Theoretical Models	41
Pressurizer Pressure Control System	43
Summary	47
V. METHOD OF SOLUTION AND THE BASIC APPROACH FOR DEVELOPING A MODULAR MODELING PROGRAM	48
Introduction	48
Method of Solution	48
Summary	58
VI. THE MODULAR MODELING PROGRAM	60
Introduction	60
The Simulator	60
The Library	63
The Driver	79
Summary	85
VII. CONCLUSIONS AND RECOMMENDATIONS	98
Conclusions	98
Recommendations	99
BIBLIOGRAPHY	100

CHAPTER	PAGE
APPENDIXES	103
APPENDIX A	104
APPENDIX B	122
APPENDIX C	133
APPENDIX D	158
VITA	220

LIST OF TABLES

TABLE	PAGE
3.1 Reactor Design Data	17
3.2 Steam Generator Design Data	21
3.3 Pressurizer Design Data	24
6.1 List of the Library Files	64
A.1 List of Variables Used in the Core Theoretical Models	120
C.1 Variables Used in the Steam Generator Simple Model	151
C.2 Variables Used in the Steam Generator Detailed Model . . .	154

LIST OF FIGURES

FIGURE	PAGE
1.1 Pressurized water reactor coolant loop	3
2.1 Model schematic	9
2.2 A typical data sheet for the MMS	10
2.3 MMS structure	12
2.4 MMP structure	14
3.1 Schematic diagram for the reactor core and the vessel external	16
3.2 Schematic diagram for a typical U-tube steam generator . .	20
3.3 Pressurizer	22
3.4 Schematic Diagram for a Typical Pressurizer Pressure Controller	25
4.1 A schematic diagram for the five lump core model	29
4.2 A schematic diagram for the seventeen lump core model	30
4.3 A schematic diagram for the forty seven lump core model	31
4.4 Mann's model	33
4.5 Core power distribution as approximated for the theoretical core models	36
4.6 Schematic diagram for the pressurizer model	40
4.7 Schematic diagram for the simple U-tube steam generator model	42
4.8 Schematic diagram for the detailed steam generator model	44
4.9 Block diagram for the pressurizer heater control system	45
4.10 Block diagram for the pressurizer sprayer control system	46

FIGURE	PAGE
5.1 Schematic diagram for a possible infinite loop that may occur when using the Runge-Kutta-Fehlberg and Adams-Bashforth-Moulton algorithms	52
5.2 Schematic diagram for a typical algorithm that utilizes the Runge-Kutta-Fehlberg method and the Adams-Bashforth-Moulton method	53
6.1 Response of the core theoretical model to a +10¢ step change in the external reactivity	81
6.2 Response of the hot leg theoretical model to a +10°F step change in the inlet temperature	86
6.3 Response of the cold leg temperature to a +10°F step change in the inlet temperature	87
6.4 Response of the simple steam generator model to a +10% step change in the steam valve	88
6.5 Response of the primary loop to a +10% step change in the steam valve coefficient	91
A.1 Response of the five lump core model to a +10°F step change in the inlet coolant temperature	114
A.2 Response of the seventeen lump core model to a +10°F step change in the inlet coolant temperature	116
A.3 Response of the forty seven lump core model to a +10°F step change in the inlet coolant temperature	118
B.1 Response of the pressurizer theoretical model to a +1.0 lbm/sec step change in the surge line flow rate	131
C.1 Response of the simple steam generator model to a +10°F step change in the inlet temperature	140
C.2 Response of the detailed steam generator model to a +10°F step change in the inlet temperature	143

CHAPTER I

INTRODUCTION

General Consideration

Dynamic simulation and testing have become important techniques for system design studies and performance analysis. Any attempt to design a system should begin with prediction of its performance before the system itself can be designed in detail or actually built. Such prediction is based on a mathematical description of the system's dynamic characteristics. This mathematical description is called a mathematical model. Mathematical models may be described in terms of differential equations. A distributed parameter mathematical model consists of a set of partial differential equations. A partial differential equation is an equality involving one or more dependent and two or more independent variables, together with partial derivatives of the dependent with respect to the independent variables. A mathematical model is called a lumped parameter model if it consists of a set of ordinary differential equations. An ordinary differential equation is an equality involving one or more dependent variables, one independent variable, and one or more derivatives of the dependent variable with respect to the independent variable. From the system engineering viewpoint, a distributed parameter model will account for the spatial dependence of the dependent variable while the lumped parameter model will assume average system parameters over defined regions or lumps.

A typical pressurized water reactor consists of two main subsystems, the Nuclear Steam Supply System and the Balance of Plant. Each of the above subsystems is further composed of many subsystems. Figure 1.1. shows a schematic diagram for a pressurized water reactor. Each subsystem has a vital role in the overall performance of the power plant. Therefore, for reliable and safe performance, the dynamic behavior of each component as well as the overall system should be fully understood.

Modeling is the key tool for understanding the dynamics of a nuclear power plant. Its importance extends beyond building a relation between an input and output signals to include and assure safety of the public and the investment.

The concept of modular modeling is a new concept in the modeling world. It was initiated by the need for a more convenient, flexible, and economical tool for analyzing the dynamics of a steam generating plant. As the name implies, each component must be modeled so that it accurately describes the processes taking place and is still computationally compatible with other models. Theoretical models for the components are stored in a library and they can be easily coupled to each other based on the user request. The prime goal of this work is to develop, document and validate a modular modeling code to simulate operational transients of a typical nuclear steam supply system. This work will also provide and describe a methodology for adding more components to the code. In the past several years, lumped parameter dynamic models for several pressurized water reactor components were developed, documented and validated by many former graduate students of

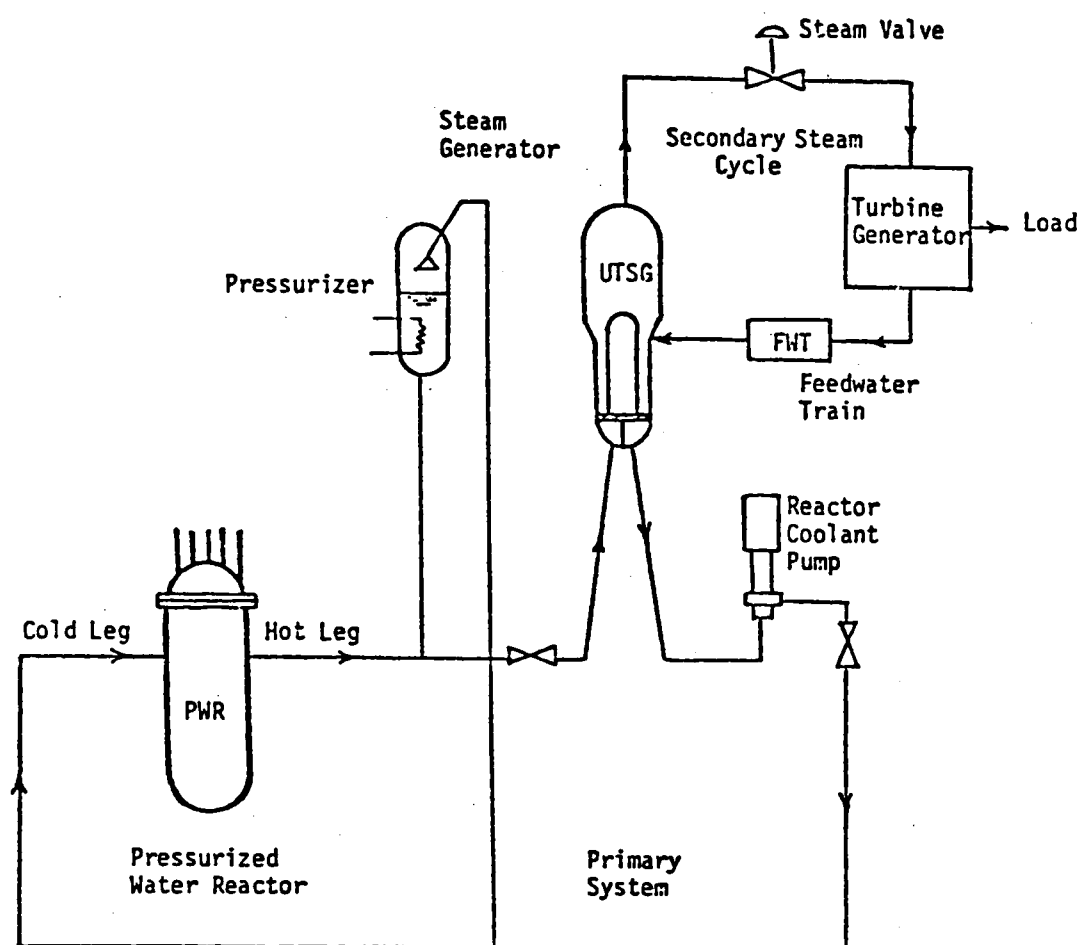


Figure 1.1. Pressurized water reactor coolant loop.

The University of Tennessee. These models will be used in this work to establish a working library for the modular modeling code. This will also serve as a tool to test the accuracy of the code by comparing the code outputs to previously obtained outputs.

Purpose and Importance of This Study

The purpose of this study is to develop a modular modeling system to simulate operational transients in a typical nuclear steam supply system using the lumped parameter modeling approach. The study will also provide a methodology for expanding the code to include a complete nuclear power plant. The code is needed to make modeling work easier and more efficient.

The primary drawback in the conventional modeling approach is the time consuming repetition process which is necessary to account for changes in the physical structure of the system to be modeled or to account for changes in the mathematical formulation of one of the previously developed models. A computer program to simulate transients in a system consisting of a nuclear core and steam generator is not useful to someone who is developing a computer program to simulate transients in a system consisting of a nuclear core, a steam generator and a pressurizer. This is due to the fact that computer codes, once written, are hard to modify by anyone but the author. Complex codes are sometimes impossible to modify. As a result, it is necessary to repeat the process of modeling the same core and the same steam generator in addition to the new pressurizer model. The program mentioned above is not useful to someone who wants to add a secondary loop to

simulate the whole power plant. In this case, the primary loop as well as the secondary loop should be modeled, programmed and loaded into the computer. The picture gets darker when the complexity of the physical system, the theoretical models or both increases.

The modular modeling concept provides a means to eliminate the time consuming process described above. Theoretical models of varying complexity for different components in a nuclear steam generating system are stored in a library. Users select subsystem models of complexity appropriate for the job and use them easily. Theoretical models for large systems can be obtained by combining smaller theoretical models into one large model.

Organization of Thesis

Following this introductory chapter, a review of the Modular Modeling System (MMS) developed by Babcock and Wilcox is presented in Chapter II along with a brief description of the Modular Modeling Program (MMP) developed at The University of Tennessee. Description of the H.B. Robinson nuclear steam supply is presented in Chapter III. Description of the theoretical models is presented in Chapter IV. Theoretical models make up the working library for the MMP. A description of the numerical scheme used in this work to solve a set of differential equations is presented in Chapter V along with a description of a basic modular structure. Application of the MMP to simulate operational transients in a pressurized water reactor is presented in Chapter VI. The conclusion is presented in Chapter VII with some recommendations for future work. The Appendices contain materials

that otherwise interfere with the smooth continuity of the discussion in the main body of this thesis.

CHAPTER II

THE MODULAR MODELING SYSTEM AND THE MODULAR MODELING PROGRAM

Introduction

The Modular Modeling System (MMS) was developed under Electric Power Research Institute (EPRI) sponsorship by the Bechtel Corporation and by the Babcock and Wilcox Company. The project was jointly funded by the EPRI Fossil Power Division and the Nuclear Power Division. The project began in 1978 and made its first public appearance in 1982. The MMS is capable of performing rapid and accurate analysis of operational transients and accidents in pressurized water reactors and fossil plants. It consists of a library of modules representing various fossil and nuclear power plant components. The modules are formulated using the lumped parameter approach. The consistency of the modules formulation allows their interconnection to construct a simulation model of any plant subsystem or system.

The MMS uses advanced simulation languages such as the Advanced Continuous Simulation Language (ACSL) and the Continuous System Modeling Program (CSMP). The system requires large computers such as the IBM 370 and the CDC 6600 and it is available through remote terminal access to commercial time share computers for users who do not have large computer facilities.

The Modular Modeling Program (MMP) is a computer program that was developed at The University of Tennessee. The purpose of this program is to demonstrate the modular modeling concept. The program consists

of three parts, the driver, the simulator and the library. The driver functions to merge several modules in the library into one model based on the user instructions. The simulator functions as a substitute to the ACSL and CSMP simulation languages. The library consists of the modules representing various nuclear steam supply system components. The modules are formulated using the lumped parameter approach and they are computationally compatible with each other.

In order to develop a clear picture of the modular modeling concept, application of the MMS and the MMP to system simulation is discussed below. For more information about the MMS, the reader is advised to refer to the references listed at the end of this thesis (1-3). The MMS methodology is presented using an example (3). The purpose of this example is to create a picture of how the MMS works rather than focusing on specific simulation goals for a system of interest.

The Modular Modeling System

The MMS methodology requires that the modeler draw a one line schematic diagram of his process showing each module and flow as shown in Figure 2.1. He then assigns a unique identifier for each module and each flow stream using any alphanumeric combination with up to three characters for the modules and four characters for the flow streams. For each module he identifies the macroname that will call the appropriate set of equations as shown in the parenthesis in the figure. The macroname is a system command. Commands such as COPY, PRINT and DELETE are considered system commands. This diagram provides a basis for filling out a user worksheet. Figure 2.2. presents an example of a

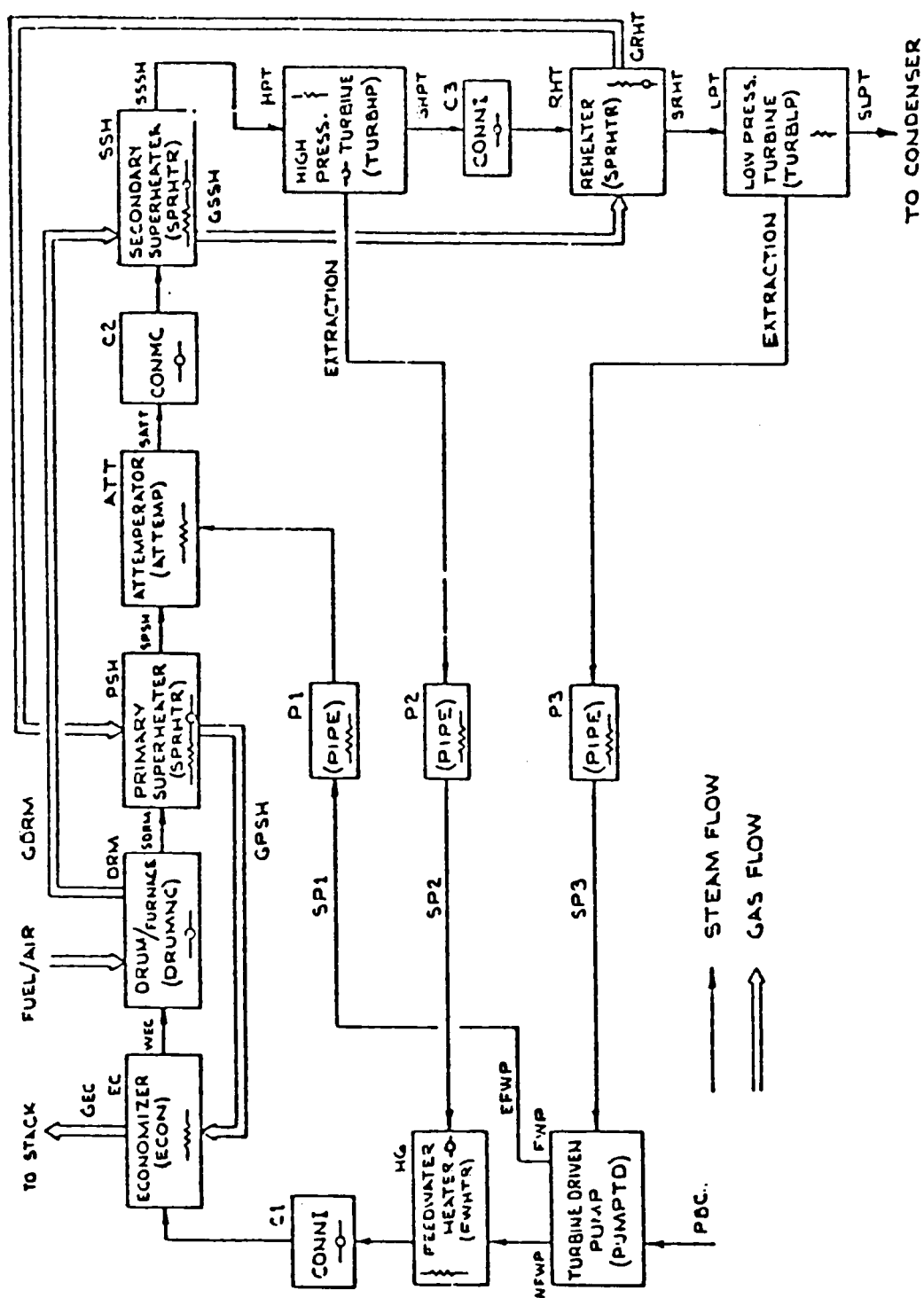


Figure 2.1. Model schematic.

ID: _ _ _

MACRO Invocation: SPRHTR (' _ ID _ ', ' _ GE _ ', ' _ SE _ ',
 ' _ GL _ ', ' _ SL _ ')

CONSTANT PARAMETERS: (All Parameters are Appended by ID)

Flow Area of Stream	(ft ²)	: KAP	_ _ _	(ID)=
Flow Area (Minimum) of Gas	(ft ²)	: KAG	_ _ _	(ID)=
Flow Conductance	(pph/ pcf-psi)	: KCP	_ _ _	(ID)=
Moisture Fraction of Flue Gas		: KMP	_ _ _	(ID)=
Mass of Heated Metal	(lbm)	: KMM	_ _ _	(ID)=
Slag Coefficient Parameter		: KSG	_ _ _	(ID)=

Gas to Metal Convective H-T Parameter		: KUC	_ _ _	(ID)=
Gas to Metal Radiant H-T Parameter		: KUR	_ _ _	(ID)=
Metal to Steam H-T Parameter		: KUS	_ _ _	(ID)=
Volume of Gas	(ft ³)	: KVG	_ _ _	(ID)=
Volume of Steam	(ft ³)	: KVS	_ _ _	(ID)=

INITIAL CONDITIONS:

Enthalpy of Flue Gas Leaving	(Btu/lbm)	: IH	_ _ _	(GL)=
Pressure of Steam Leaving	(psia)	: IP	_ _ _	(SL)=
Temperature of Metal	(P)	: ZIT	_ _ _	(ID)=
Spec Internal Energy of Steam Leaving	(Btu/lbm)	: IU	_ _ _	(SL)=

Figure 2.2. A typical data sheet for the MMS.

worksheet for a module. The block diagram at the top shows all the input and output parts of the module. All upstream ports are on the left side of the block while downstream ports are on the right. The variables listed show the inputs and outputs required to be compatible with other modules. The direction of the arrow attached to the box distinguishes a computational input from an output. Below the box is shown the one line coding required to invoke the macro; that is, call the set of equations that represents the module. The list of parameters following the macro invocation statement are the required parameters for the module. A schematic diagram for the MMS is shown in Figure 2.3. A data sheet for each module is required and when fed to the computer, several worksheets generate the system of interest.

The Modular Modeling Program

The MMP methodology is fundamentally different from the MMS. The MMS is modular in the microscopic sense. That is, a single equation is coupled to a single equation. the MMP is modular in the macroscopic sense. That is, a single theoretical model is coupled to another theoretical model. However, this is no drawback to the MMP since a single equation may define a theoretical model. The MMP methodology requires that each component in a system be modeled so that it accurately describes the process taking place and is still computationally compatible with other modules. Each module requires a data sheet. The data sheet is a file that consists of the necessary parameters for the module. Each module is provided with a reference data sheet which contains necessary parameters for a specific module as obtained from

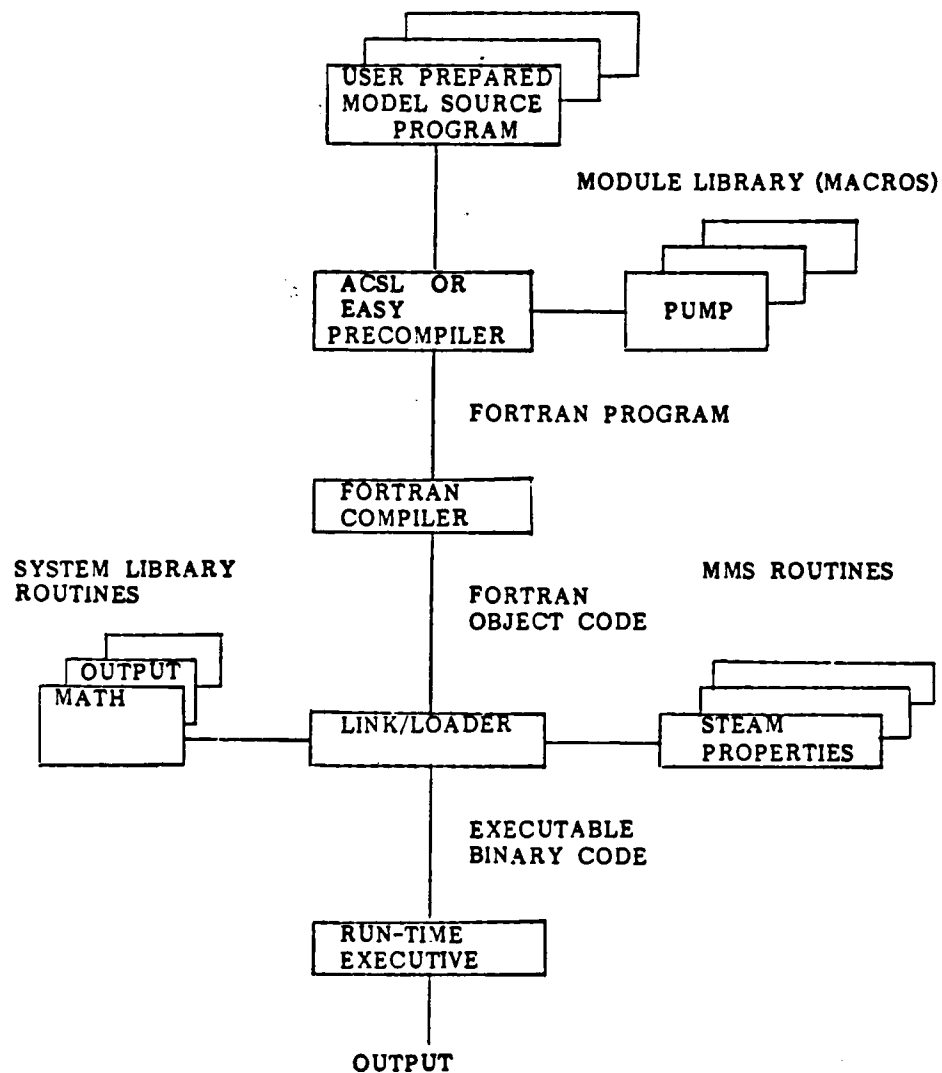


Figure 2.3. MMS structure.

the H.B. Robinson nuclear plant design data. A different set of parameters may be used by creating an adequate data file. Several data files are presented in Appendix D. A schematic diagram for the MMP is shown in Figure 2.4.

Summary

The MMS and the MMP are two modular modeling programs that are designed to simulate transients in a nuclear power plant. The MMS depends on advanced simulation language packages and requires large computers to operate. The user has no access to the basic structure of the system and is limited to the modules in the MMS library. The MMP (as will be discussed later) provides its own package to solve a set of nonlinear first order differential equations and is adequate for medium size microcomputers. The user of the MMP has the option of adding new modules to the library with minimum effort provided that the theoretical models are ready. The MMP will be discussed in more detail in the coming chapters.

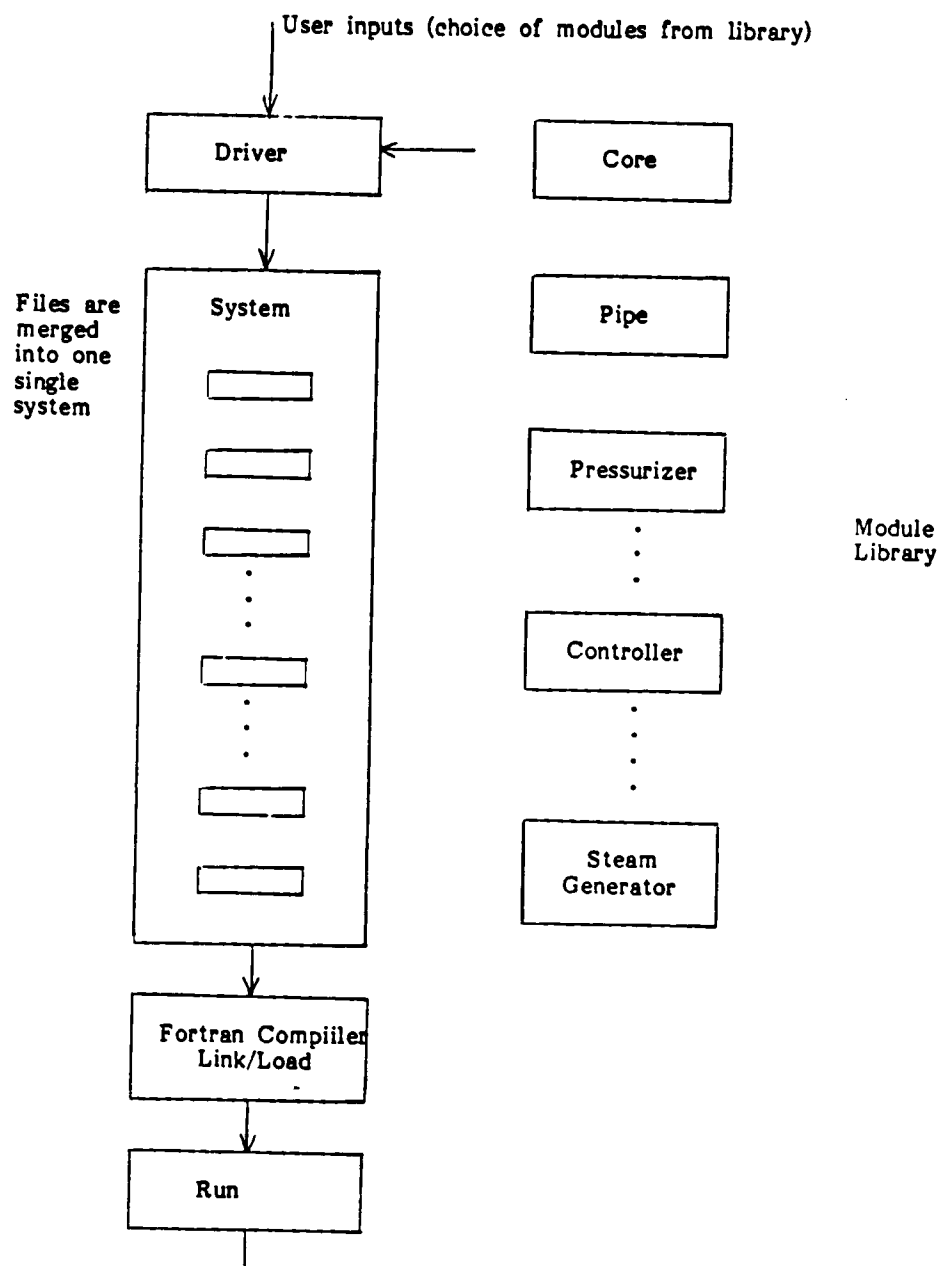


Figure 2.4. MMP structure.

CHAPTER III

DESCRIPTION OF THE PHYSICAL PLANT

The System

In order to check the accuracy of a model, theoretical results should be compared to experimental results. In this study, the H.B. Robinson Unit 2 nuclear power plant was used as a reference plant. This choice was based on the fact that experimental results for this plant are available in documents written by former graduate students of The University of Tennessee. The H.B. Robinson Unit 2 is a 739 Mwe. (2200 MWt) pressurized water reactor. The Nuclear Steam Supply System consists of three reactor coolant loops. Each coolant loop consists of a U-tube and shell type steam generator, a circulating pump, a hot leg and a cold leg. The system contains one pressurizer attached to one of the hot legs.

Reactor

The H.B. Robinson nuclear plant contains a Westinghouse pressurized water reactor. The reactor core and the vessel externals are shown in Figure 3.1. Reactor coolant enters the reactor vessel through nozzles near the top of the core and, constrained by a "core barrel" between the vessel and the core, flows to the bottom of the core. The water then flows up through the core and out exit nozzles to the steam generators. From there, the coolant is recirculated to the core large primary coolant pump. A list of some important design parameters for the core is shown in Table 3.1.

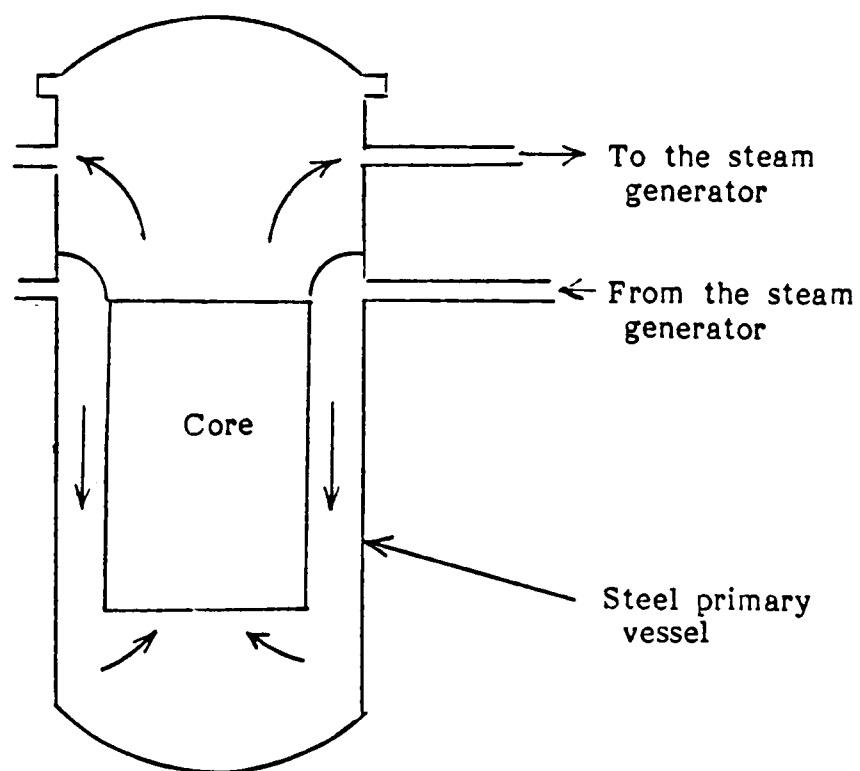


Figure 3.1. Schematic diagram for the reactor core and the vessel external.

TABLE 3.1
REACTOR DESIGN DATA

1. Fuel Weight (UO ₂), lbs	176,200
2. Cladding Weight (Zircaloy), lbs	36,300
3. Core Diameter, inches	119.7
4. Core Height, inches	144.
5. H ₂ O/U Volume Ratio (cold)	4.18
6. Number of Fuel Assemblies	157
7. UO ₂ Rods per Assemble	204
8. Doppler Coefficient ($\Delta K/K$)/°F	-1.1×10^{-5}
9. Moderator Temperature Coefficient ($\Delta K/K$)/°F	-2.0×10^{-4}
10. Moderator Pressure Coefficient ($\Delta K/K$)/psi	-1.0×10^{-6}
11. Prompt Neutrol Lifetime, sec	1.79×10^{-5}
12. Delayed Neutron Franction, %	6.45×10^{-3}
13. Delayed Neutron Constants	
Decay Constant λ_i , sec ⁻¹	Fraction of Total Neutrons
0.0125	0.00021
0.0305	0.00140
0.111	0.00125
0.301	0.00253
1.14	0.00074
3.01	0.00027
14. Total Primary Heat Output, MWt	2,200
15. Nominal System Pressure, psi	2,250
16. Total Coolant Flow Rate, lbs/hr	101.5×10^6
17. Average Coolant Velocity along Fuel Rods, ft/sec	14.3

TABLE 3.1 (continued)

18. Total Mass of Primary Coolant, lbs	406,060
19. Average Coolant Mass Velocity, lbs/hr.ft ²	2.32×10^6
20. Nominal Coolant Inlet Temperature, °F	546.2
21. Nominal Coolant Outlet Temperature, °F	602.1
22. Active Heat Transfer Surface Area, ft ²	42,460
23. Average Heat Flux, BTU/hr.ft ²	171,600
24. Maximum Thermal Output, Kw/ft	17.9
25. Maximum Average Clad Temperature, °F	715
26. Maximum Fuel Central Temperature at 100% Power, °F	4,030

Steam Generator

Steam generators are employed to transfer heat energy from the primary loop to the secondary loop. The Westinghouse reactor design uses the U-tube steam generators similar to the one shown in Figure 3.2. Primary coolant enters at the bottom left, flows inside a bundle of tubes (of which only two are shown in the figure) in a U-shaped path and leaves on the opposite side. Feedwater is introduced at a level just above the U-tubes through a Feedwater ring. It mixes with the recirculated water from the steam drum section. The slightly subcooled mixture passes through the downcomer and flows upward outside the U-tubes. Heat is transferred from the pressurized primary coolant to the secondary coolant through the tubes and a steam-water mixture is formed. The steam-water mixture from the tube bundle passes through the steam swirl vane assembly which imparts centrifugal motion to the steam-water mixture and removes most of the water from the mixture. The recirculated water flows down through the downcomer while the steam-water mixture flows upward and passes through the secondary separator to limit its moisture to 0.25%. A list of some important design parameters is shown in Table 3.2.

Pressurizer

The pressurizer supplies and controls the system pressure in a pressurized water reactor. The unit is a vertical vessel as shown in Figure 3.3. The relief valves and the spray nozzle are at the top in the vapor zone; the heaters are in the liquid zone. Pressurizers have a volume of 1500 to 1700 cubic feet, about half of which is filled with

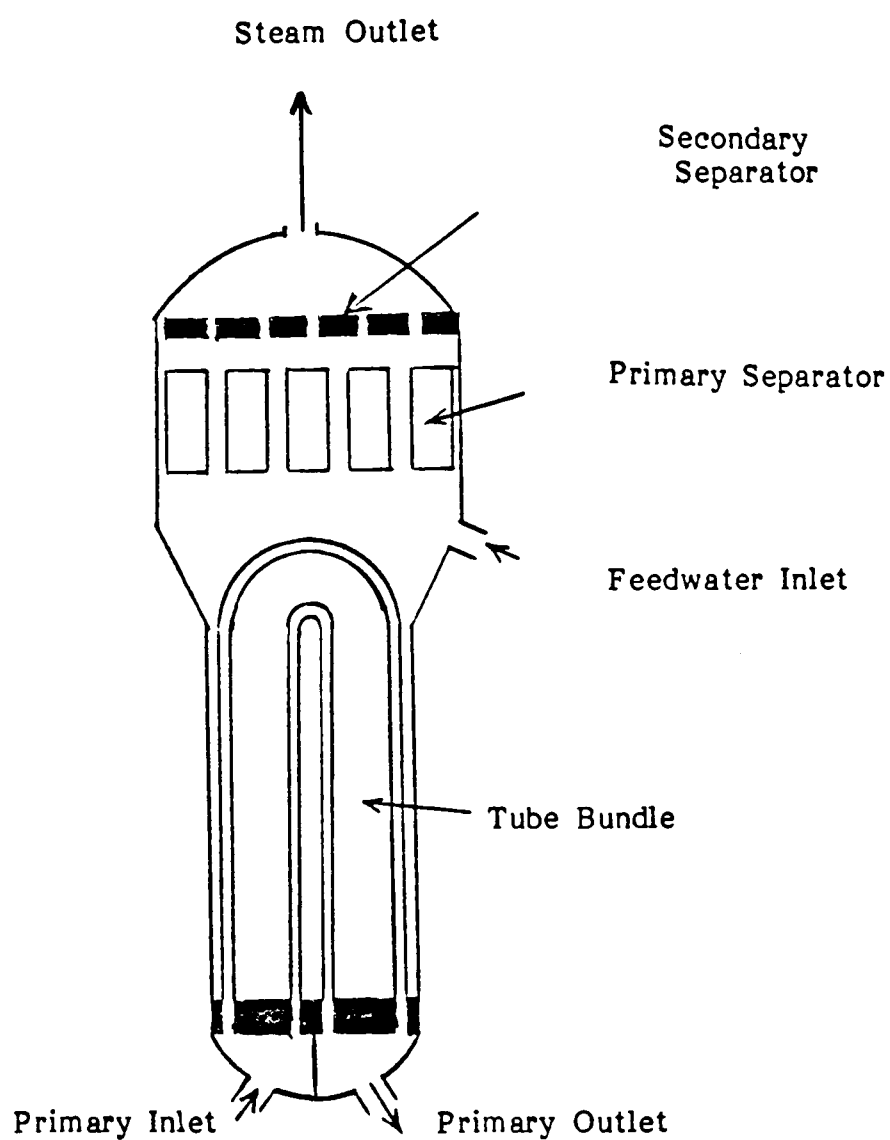


Figure 3.2. Schematic diagram for a typical U-tube steam generator.

TABLE 3.2
STEAM GENERATOR DESIGN DATA

1. Design Pressure, Reactor Coolant/steam, psig	2485/1085
2. Design Temperature, Reactor Coolant/steam, °F	650/556
3. Overall Height, ft-in	63-1.6
4. Shell OD, Upper/Lower, in.	166/127.5
5. Shell Thickness, Upper/lower, in.	3.5/2.63
6. Number of U-tubes	3260
7. U-tubes Diameter, in.	0.875
8. Tube Wall Thickness, in.	0.050
9. Mass of U-Tube Metal, lbs	91800
10. total Heat Transfer Area, ft ²	44430
11. Steam Flow, lb/hr	3.169×10^6
12. Steam Temperature, °F	516
13. Steam Pressure, psig	770
14. Reactor Coolant Flow, lbs/hr	33.93×10^6
15. Reactor Coolant Water Volume, ft ³	928
16. Feedwater Temperature, °F	425
17. Recirculation Ratio	9
18. Secondary Side Water Volume, (full power), ft ³	1526
19. Secondary Side Steam Volume, (full power), ft ³	3203

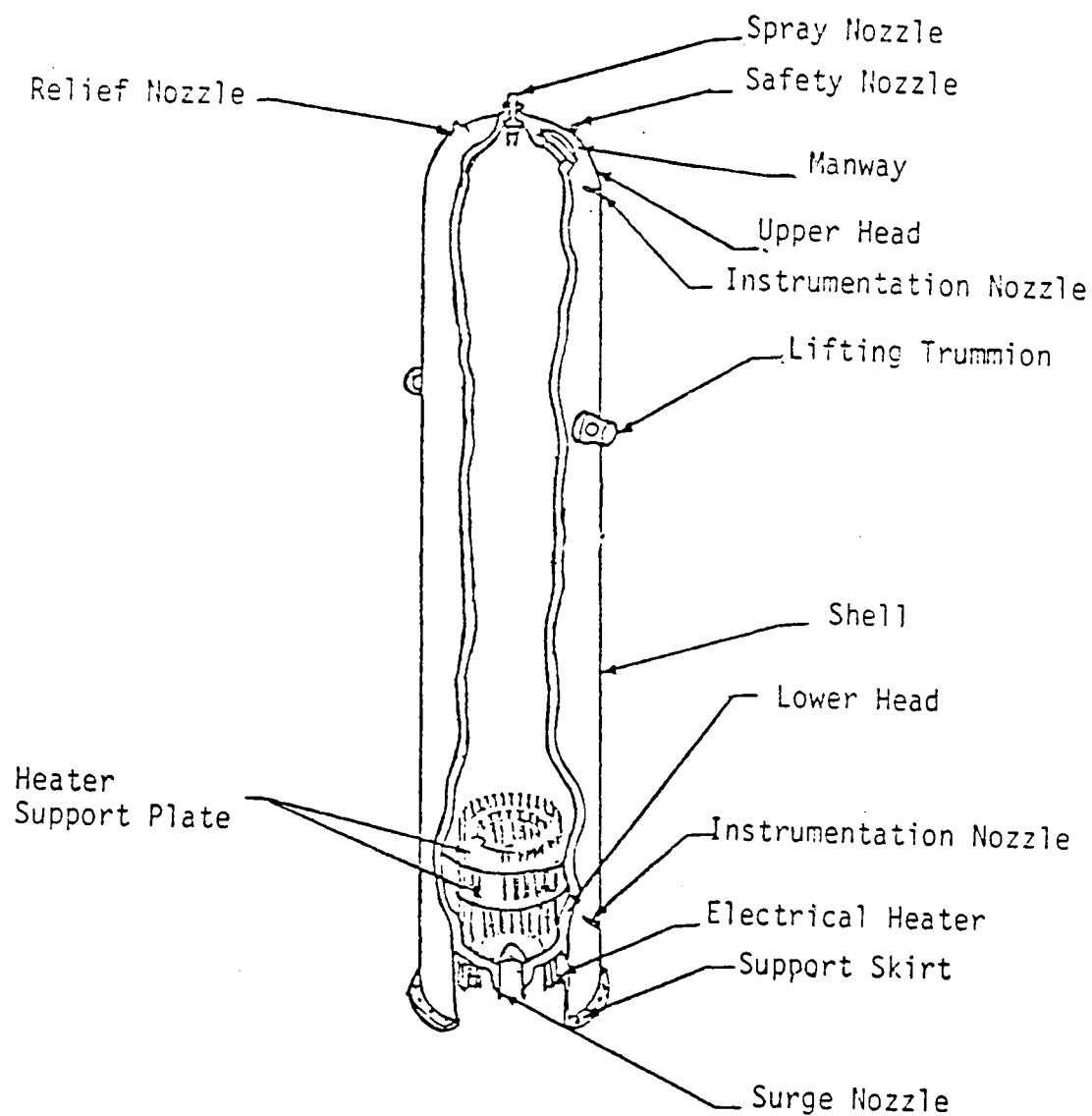


Figure 3.3. Pressurizer.

liquid. Safety valves and power operated relief valves are connected to the vapor space nozzles and the surge line is connected to the bottom of the vessel. A typical pressurizer has an inside diameter of about seven feet, six inches and an overall length of about forty-five feet. Heat energy is supplied by multiple banks of replaceable electric heaters; typical total heater load is 1800 kw. On start up, all heater banks are used in order to minimize pressurization time. As the unit comes up to pressure, heaters are turned off until finally only one small group of heaters operates continuously to replace heat losses. A second small group of heaters responds to the demands of the pressure control system as required. Necessary design data for the pressurizer are listed in Table 3.3.

Pressure control system. A schematic diagram for the pressurizer pressure control system is shown in Figure 3.4. Pressurizer spray comes on for high pressures to condense steam in the pressurizer steam phase and reduce high pressures. The pressurizer power operated relief valves at pressures above the pressure at which spray is turned on fully but at pressures below the high pressure at which the reactor is tripped or the self-actuated safety are activated.

The pressurizer heaters are divided into heaters with proportional control and heaters with ON-OFF control (backup heaters). During steady state operation, the pressurizer control system normally controls only the proportional heaters to compensate for minor pressure fluctuations. The proportional heater will continuously operate at low level to compensate for the continuous spray rate (approximately 1 gpm).

TABLE 3.3
PRESSURIZER DESIGN DATA

1.	Design/Operating Pressure, psig	2485/2235
2.	Design/Operating Temperature, °F	680/653
3.	Surge Line Maximum Flow Rate, gpm	19390
4.	Water Volume, Full Power, ft ³	780
5.	Steam Volume, Full Power, ft ³	520
6.	Electric Heater Capacity, KW (total)	1800
7.	Heat Up Rate of Pressurizer, °F/hr	55
8.	Continuous Spray Rate, gpm	1.0
9.	Design Spray Rate, Full Open, gpm	840

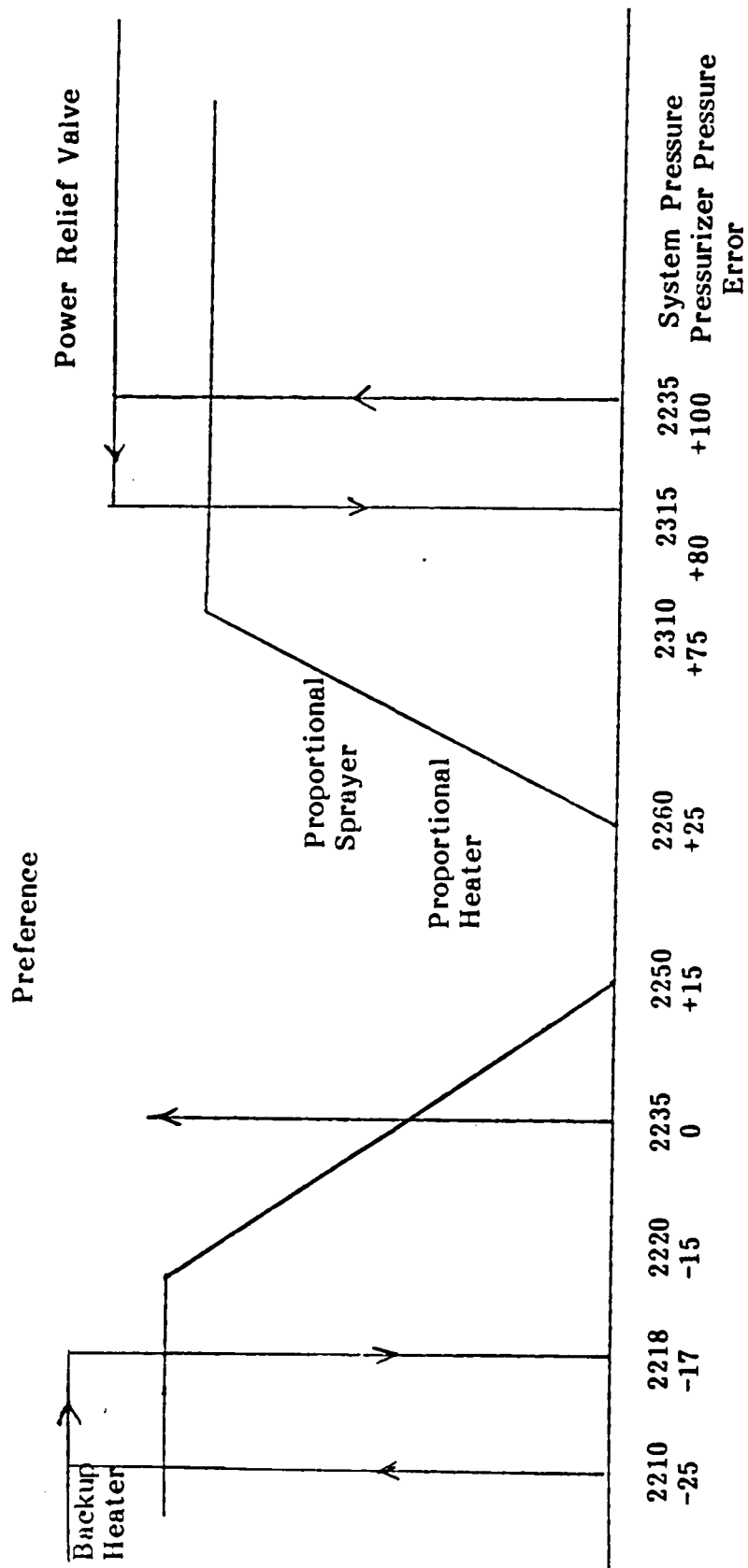


Figure 3.4. Schematic diagram for a typical pressurizer pressure controller.

A dead band between the initiation of the proportional spray and turn off of the proportional heaters prevents frequent operation of the proportional spray valves during minor system pressure variations.

CHAPTER IV

THEORETICAL MODELS

Introduction

Theoretical models for the H.B. Robinson Unit 2 primary loop are discussed in this chapter. The models consist of sets of first order ordinary differential and algebraic equations. For each theoretical model, a list of the governing equations is presented in this chapter. All theoretical models that are included in this work are derived in Appendices A through C. For more information, interested readers are referred to the references listed at the end of this thesis.

Core

Three theoretical models for the reactor core are included in this work as a part of the MMP library. These models were developed and tested previously by former graduate students (4-10) of The University of Tennessee. The reason behind using three core theoretical models is to demonstrate the ability to select subsystem models from a library of models in the MMP library. For each core model, two different formulations were developed. The difference between these submodels is in the formulation of the reactor power equation. In one case, the power equation is a first order ordinary differential equation and in the other case, the power equation is an algebraic equation (the prompt jump approximation). In the first case, the time step for the numerical solution must be very small and therefore requires a large amount of computer time. In the second case, the time step is large compared

to the previous case and still produces virtually the same response as in the first case for normal operational transients.

Schematic diagrams for the core models are shown in Figure 4.1 through 4.3. The first core model consists of one lump for the fuel, two lumps for the coolant, one lump for the lower plenum and one lump for the upper plenum. The second core model consists of five axial lumps for the fuel, ten axial lumps for the coolant, one lump for the upper plenum and one lump for the lower plenum. The third core model consists of a set of twenty fuel lumps (five axial and four radial), forty coolant lumps (ten axial and four radial), one lump for the upper plenum and one lump for the lower plenum. As shown in Figures 4.1 through 4.3, two coolant lumps are associated with each fuel lump. Each coolant lump is considered to be a well-stirred tank (outlet coolant temperature equals the average temperature of the lump). This modeling procedure is called Mann's method. The driving force for heat transfer from the fuel to the coolant is taken as the difference between the fuel temperature and the average temperature of the first coolant section. Mann's model is a more realistic representation for the heat transfer process from the fuel to the coolant than representing the heat as being transferred from single fuel section to a single well-stirred tank coolant section.

When using a single coolant lump, a common assumption for the average temperature of the lump is the mean value of the inlet and outlet temperatures as given by:

$$TAV = \frac{TIN + TOUT}{2} \quad 4.1$$

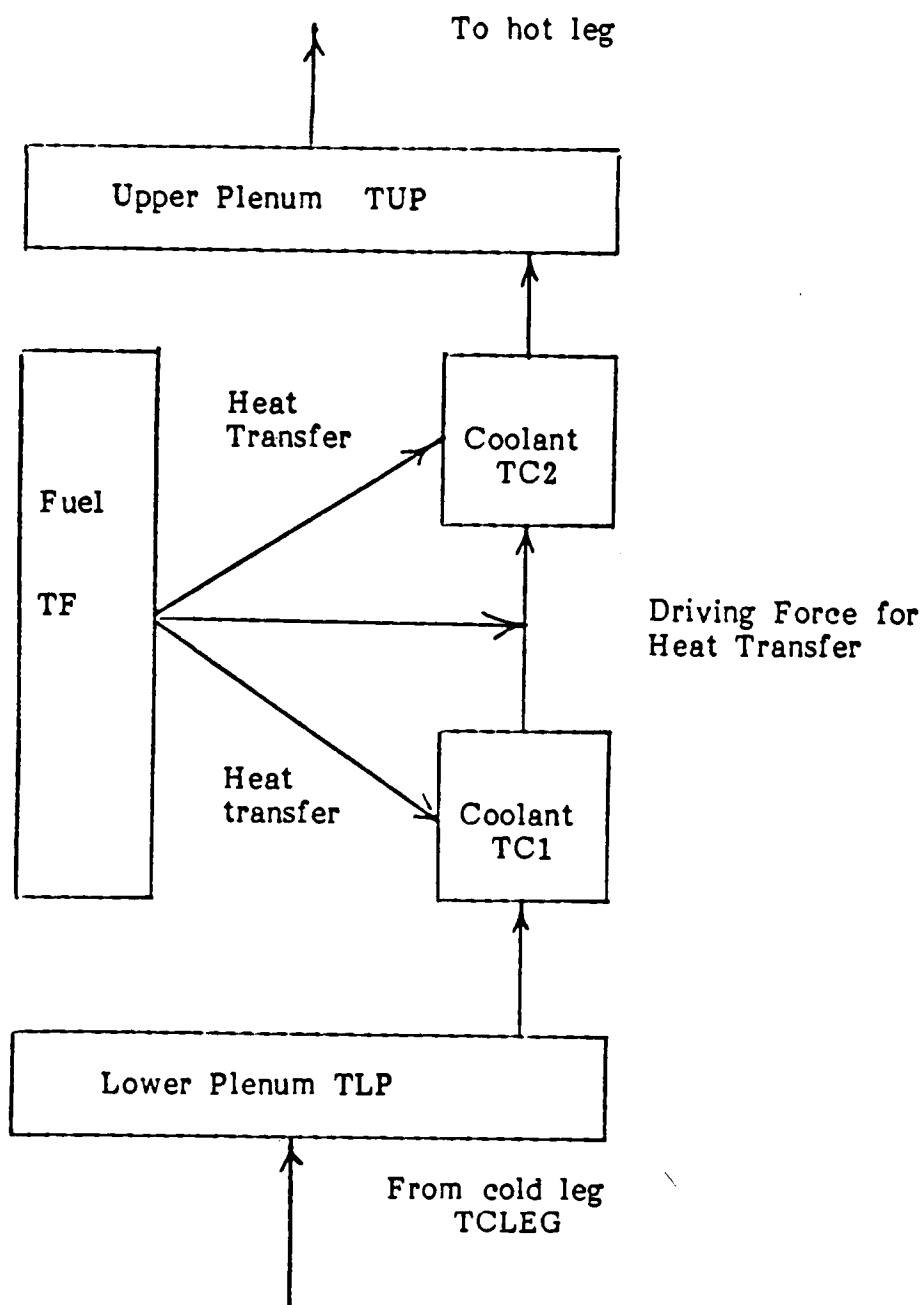


Figure 4.1. A schematic diagram for the five lump core model.

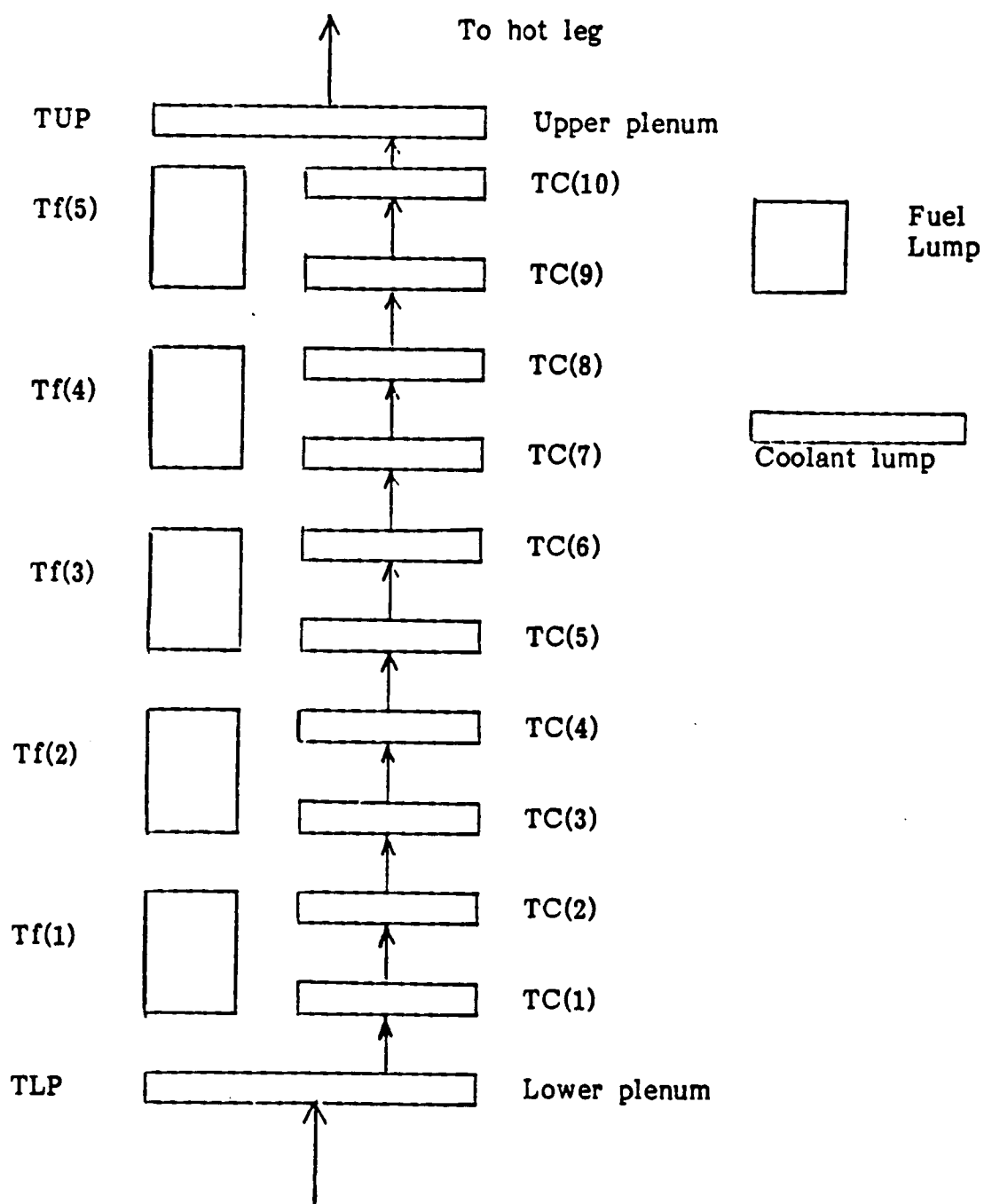


Figure 4.2. A schematic diagram for the seventeen lump core model.

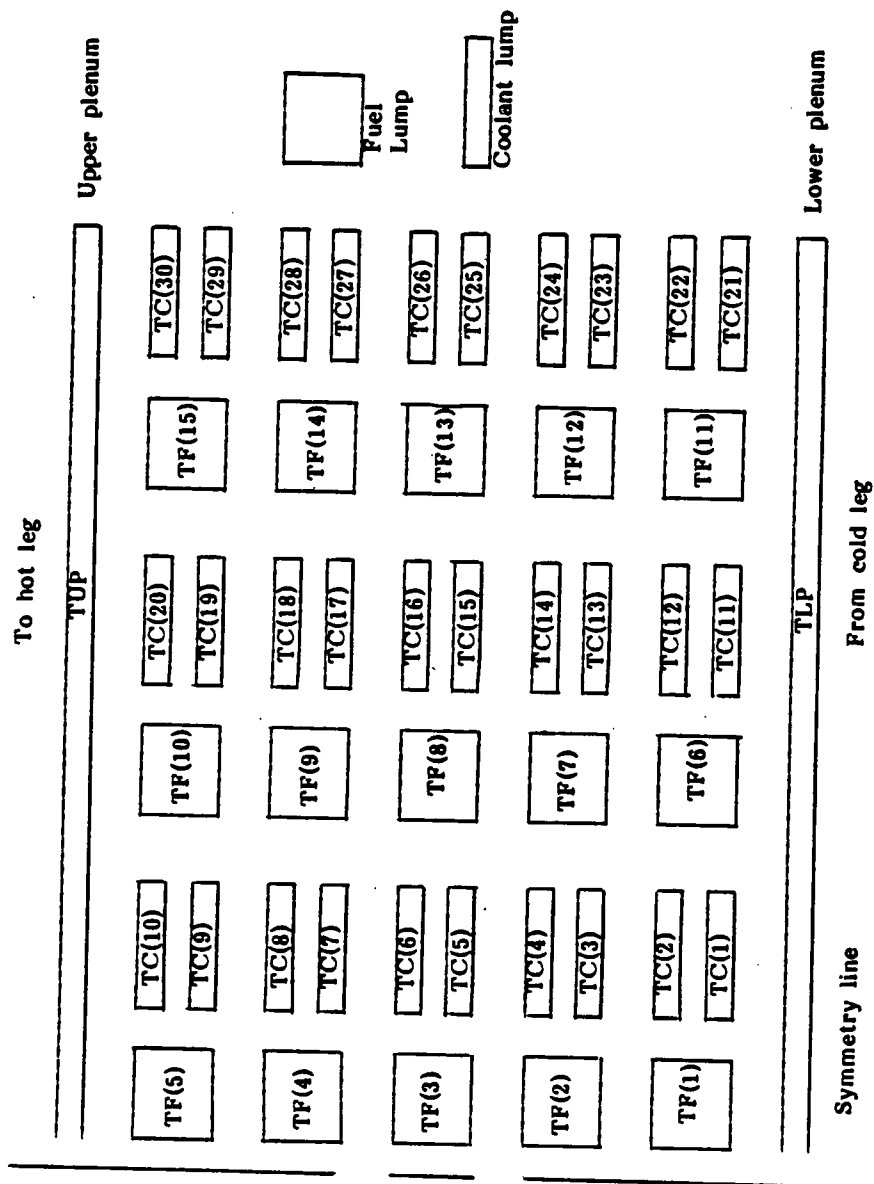


Figure 4.3. A schematic diagram for the forty seven lump core model.

where

TAV = average temperature of the coolant in the section, °F

TIN = inlet coolant temperature, °F

TOUT = outlet coolant temperature, °F

Rearranging equation 4.1 results in

$$TOUT = 2TAV - TIN \quad 4.2$$

A rapid step increase in the coolant inlet temperature will lead to a rapid decrease in the coolant outlet temperature (based on equation 4.2). This is due to the fact the TAV cannot change instantaneously in a medium which has a heat capacity.

As mentioned before, three theoretical models for the core are used in this work. Each of the described core models consists of a Mann's model node that is axially coupled to another Mann's model node in cases where more than one fuel lump is used. A schematic diagram for a typical Mann's model is shown in Figure 4.4. Using the point kinetics equation with six groups of delayed neutrons and reactivity feedback, the governing set of equations for the core (using one Mann's model section to describe the heat transfer process between one fuel lump and the adjacent coolant lumps) are listed below (see Appendix A for details).

Power Equation:

$$\frac{d}{dt} \frac{\delta P}{P_0} = \frac{\delta \rho}{\Lambda} + \frac{1}{\Lambda} (\delta \rho - \beta t) \frac{\delta P}{P_0} + \sum_{i=1}^6 \lambda_i \frac{\delta C_i}{P_0} \quad 4.3$$

Power Equation with the Prompt Jump Assumption:

$$\frac{\delta P}{P_0} = (\delta \rho + \Lambda \sum_{i=1}^6 \lambda_i \frac{\delta C_i}{P_0}) / (\beta t - \delta \rho) \quad 4.4$$

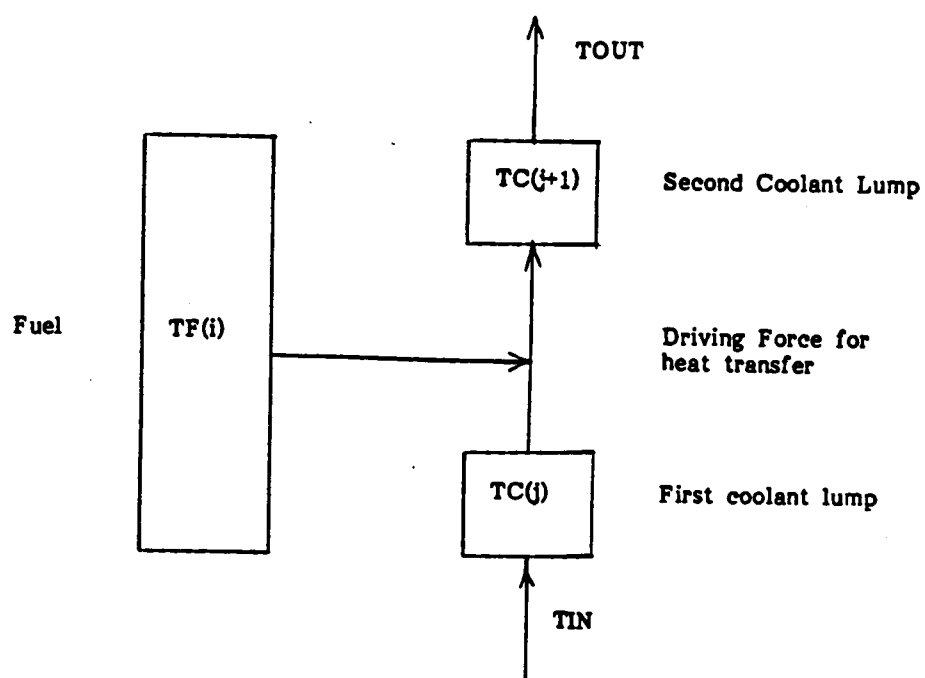


Figure 4.4. Mann's model.

Delayed Neutrons Precursors:

$$\frac{d}{dt} \frac{\delta C_i}{P_0} = \frac{\beta_i}{\Lambda} \frac{\delta P}{P_0} - \lambda_i \frac{\delta C_i}{P_0} \quad i = 1, 2, \dots, 6 \quad 4.5$$

where

δP = deviation of the reactor power from its steady state value, BTU/sec

δC_i = deviation of the concentration of the i^{th} precursor from its steady state value

β_t = total delayed neutron fraction

β_i = delayed neutron fraction for the i^{th} group

λ_i = delayed neutron decay constant for the i^{th} group, sec^{-1}

Λ = prompt neutron generation time, sec

P_0 = steady state reactor power, BTU/sec

$\delta \rho$ = change in reactivity due to changes in the fuel temperature, moderator temperature, primary pressure and the automatic control rod position

The reactivity term $\delta \rho$ may be expressed as follows:

$$\delta \rho = \delta \rho_1 + \delta \rho_2 + \delta \rho_3 \quad 4.6$$

where

$\delta \rho_1$ = reactivity changes due to changes in the position of the manual control rod

$\delta \rho_2$ = reactivity changes due to changes in the position of the automatic control rod

$\delta \rho_3$ = reactivity changes due to changes in the fuel temperature, coolant temperature, and the primary side pressure

The reactivity term $\delta \rho_3$ may be expressed as follows:

$$\delta \rho_3 = \sum_{i=1}^N \text{ALFAF IF}(i) (TF(i) - TF_0(i)) + \sum_{j=1}^M \text{ALFAC IC}(j) (TC(j) - TCo(j)) + \text{ALFAPR} (PR - PRO) \quad 4.7$$

where

ALFAF = Doppler coefficient, $(\Delta K/K)/^{\circ}\text{F}$

ALFAC = moderator temperature coefficient of reactivity, $(\Delta K/K)/^{\circ}\text{F}$

IF(i) = reactivity importance for temperature changes in the i^{th} fuel lump

IC(j) = reactivity importance for temperature changes in the j^{th} coolant lump

TF(i) = average fuel temperature in the i^{th} fuel lump, $^{\circ}\text{F}$

TFo(i) = average steady state fuel temperature in the i^{th} fuel lump, $^{\circ}\text{F}$

TC(j) = average coolant temperature in the j^{th} coolant lump, $^{\circ}\text{F}$

TCo(j) = average steady state coolant temperature in the j^{th} coolant lump, $^{\circ}\text{F}$

Pr = average pressure in the primary loop, psi

Pro = average steady state pressure in the primary loop, psi

N = total number of the fuel lumps

M = total number of the coolant lumps

The reactivity importance for each lump (fuel plus coolant) may be obtained by considering a cosine power shape in the axial and the horizontal planes as shown in Figure 4.5. Based on this power distribution, the reactivity importance may be obtained by setting the following loop in analogy to a typical computer loop (see Appendix A for details).

i = 0

a = 1 to A

i = i + 1

h = 1 to H

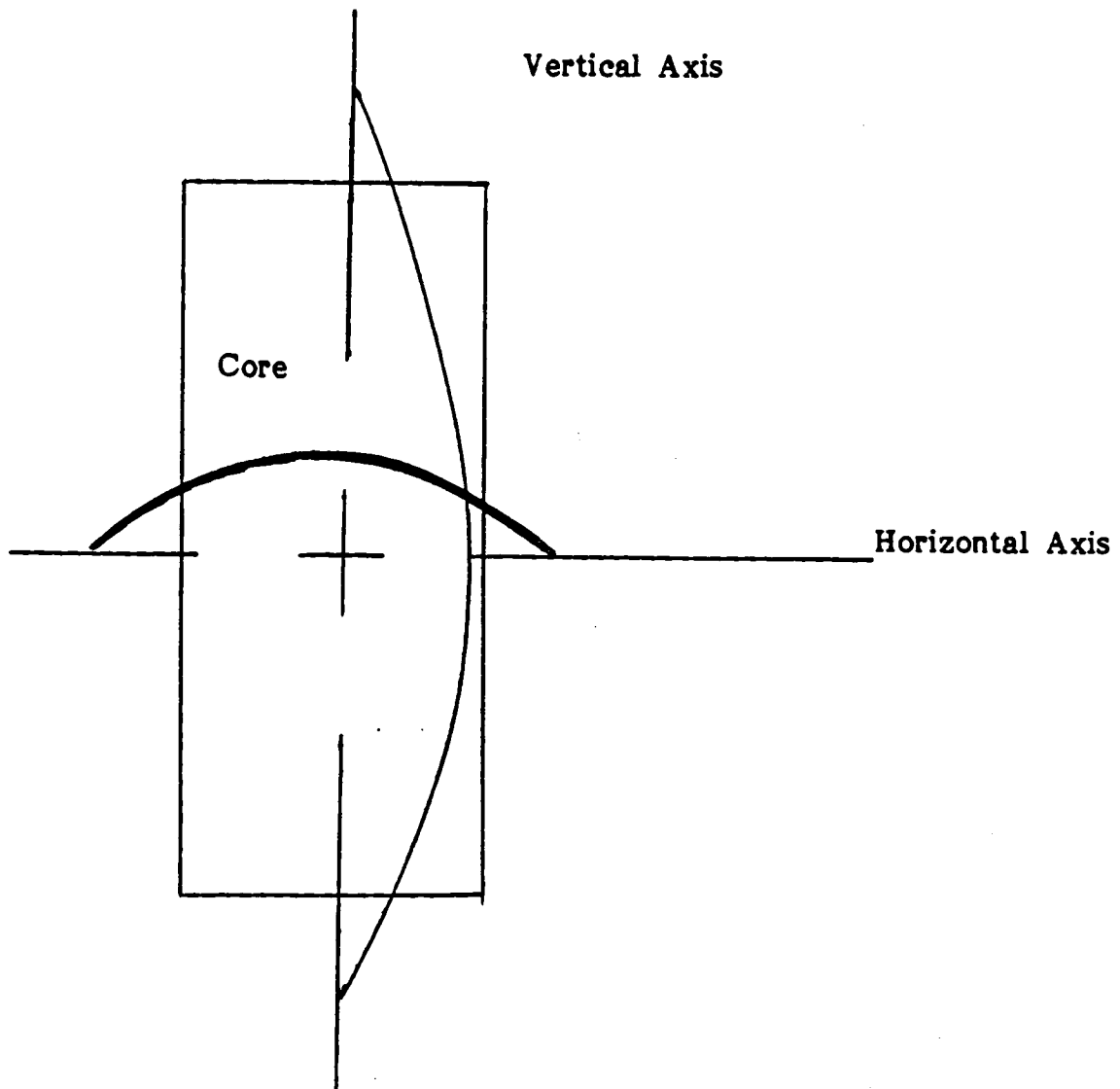


Figure 4.5. Core power distribution as approximated for the theoretical core models.

$$IF(1) = \frac{\int_{\frac{\pi}{A}(a-1)}^{\frac{\pi}{A} * a} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{\frac{\pi}{H}(h-1)}^{\frac{\pi}{H} * h} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad 4.8$$

$$h = h + 1$$

$$a = a + 1$$

where

A = total number of the fuel or coolant lumps in the axial direction

H = total number of the fuel or coolant lumps in the horizontal direction

a = counter for the axial lumps

h = counter for the horizontal lumps

i = denotes the i^{th} lump

Fuel Lump Average Temperature:

$$\frac{d}{dt} TF(i) = \frac{IF(i)}{MF \, CPF} P_o \left(1 + \frac{P}{P_o}\right) - \frac{HFC \, A}{MF \, CPF} (TF(i) - TC(j)) \quad 4.9$$

First Coolant Lump Average Temperature:

$$\frac{d}{dt} TC(j) = \frac{WP}{MC} (TIN - TC(j)) + \frac{HFC \, A}{2MC \, CP1} (TF(i) - TC(j)) \quad 4.10$$

Second Coolant Lump Average Temperature:

$$\frac{d}{dt} TC(j+1) = \frac{WP}{MC} (TC(j) - TC(j+1)) + \frac{HFC \, A}{2MC \, CP1} (TF(i) - TC(j)) \quad 4.11$$

where

MF = mass of the fuel in the fuel lump, lbm

CPf = specific heat for the fuel, BTU/lbm-°F

MC = mass of the coolant in the coolant lump, lbm

CP1 = specific heat for the coolant, BTU/lbm-°F

WP = primary coolant mass flow rate, lbm/sec

A = effective heat transfer area between the fuel and the coolant, ft²

HFC = overall heat transfer coefficient between the fuel and the coolant, BTU/hr-ft²-°F

Core Lower Plenum:

$$\frac{d}{dt} TLP = \frac{WP}{MLP} (TCLEG - TLP) \quad 4.12$$

Core Upper Plenum:

$$\frac{d}{dt} TUP = \frac{WP}{MLP} (TCOR - TUP) \quad 4.13$$

where

TLP = average coolant temperature in the core lower plenum, °F

TUP = average coolant temperature in the core upper plenum, °F

TCLEG = average coolant temperature in the cold leg, °F

TCOR = average coolant temperature at the core exist, °F

MLP = mass of the coolant in the lower plenum, °F

MUP = mass of the coolant in the upper plenum, °F

Pressurizer Theoretical Model

The MMP library contains one pressurizer model. The pressurizer theoretical model consists of a set of two lumps, one lump for the

liquid zone and the other for the steam zone. A schematic diagram for the model is shown in Figure 4.6. The pressurizer theoretical model was developed by applying the mass, energy and volume balances to each lump using the following assumptions.

1. Both the water and the steam in the pressurizer are always at the saturation temperature corresponding to the reactor coolant pressure.
2. Heat loss through the pressurizer wall is not a function of time and therefore remains constant during the transient.

The governing set of equations for the pressurizer theoretical model (see Appendix B for details) is

Pressurizer Pressure:

$$\frac{d}{dt} PR = \frac{1}{b1} (Q + b2 WWI + b3 WSP) \quad 4.14$$

Pressurizer Water Level:

$$\frac{d}{dt} LW = c1 Q + c2 WWI + c3 WSP \quad 4.15$$

where

PR = average pressure in the primary side, psi

LW = pressurizer water level, in.

Q = pressurizer heater load, KW

WWI = surge line flow rate, lbm/sec

WSP = pressurizer sprayer flow rate, lbm/sec

The coefficients b1, b2, b3, c1, c2 and c3 are derived in Appendix

B.

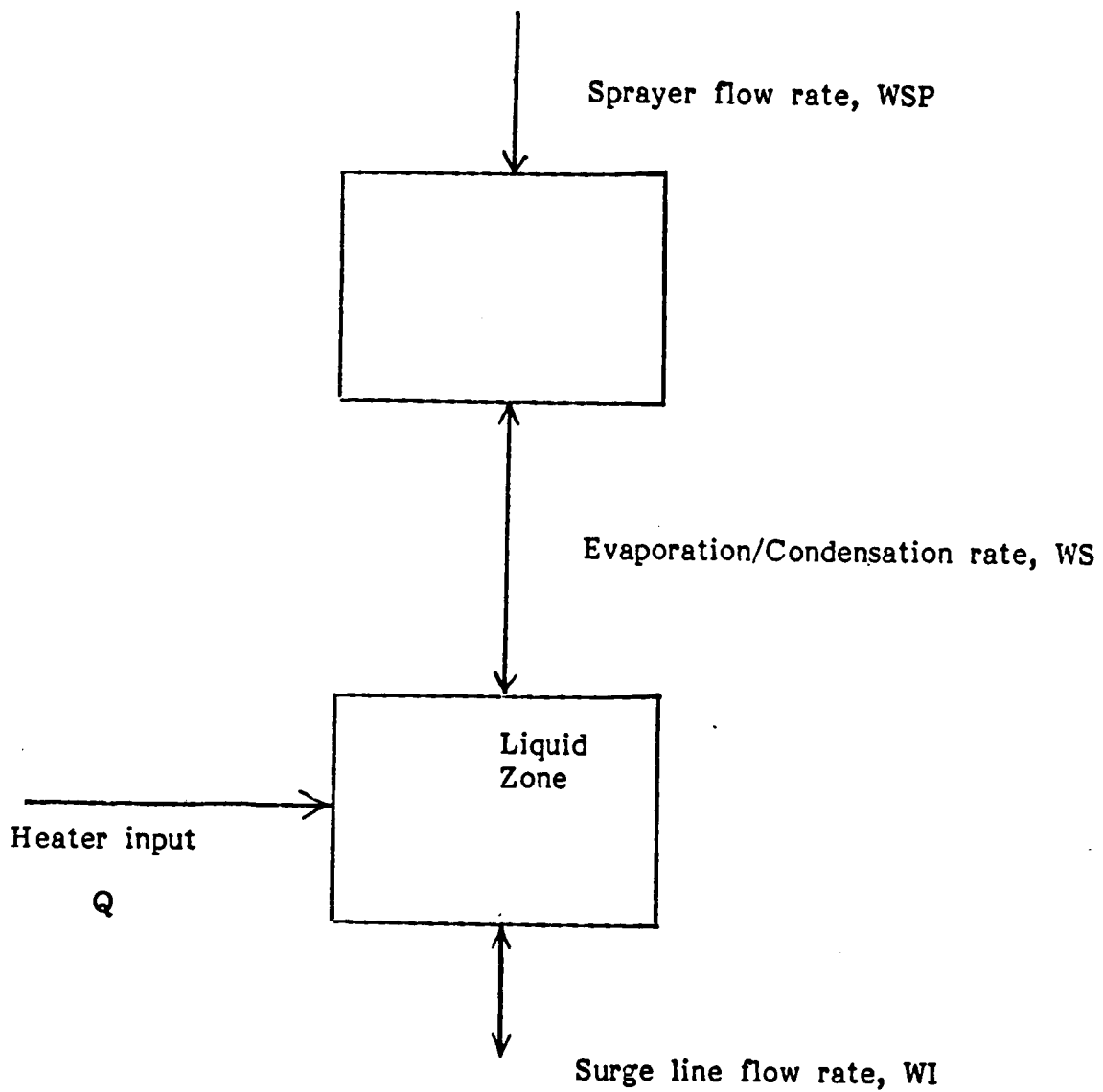


Figure 4.6. Schematic diagram for the pressurizer model.

Steam Generator Theoretical Models

Several theoretical models for the U-tube steam generator were developed by Ali in 1976. Two of these theoretical models are used in this work. Both are in the MMP library. The simple model consists of one lump for the steam generator inlet plenum, one lump for the steam generator outlet plenum, two lumps for the primary coolant inside the tubes, one lump for the tube metal and one lump for the secondary side. A schematic diagram for the model is shown in Figure 4.7. The governing set of equations is derived in Appendix C. The detailed model consists of the following lumps:

C. The detailed model consists of the following lumps:

A. Primary side lumps

1. Primary inlet plenum (PRIN)
2. Parallel flow branch subcooled heat transfer section, primary lump 1 (PRL1)
3. Parallel flow branch boiling heat transfer section, primary lump 2 (PRL2)
4. Counter flow branch boiling heat transfer section, primary lump 3 (PRL3)
5. Counter flow branch subcooled heat transfer section, primary lump 4 (PRL4)
6. Primary outlet plenum (PROUT)

B. Conducting metal lumps

7. Parallel flow branch subcooled heat transfer section, tube metal lump 1 (MTL1)

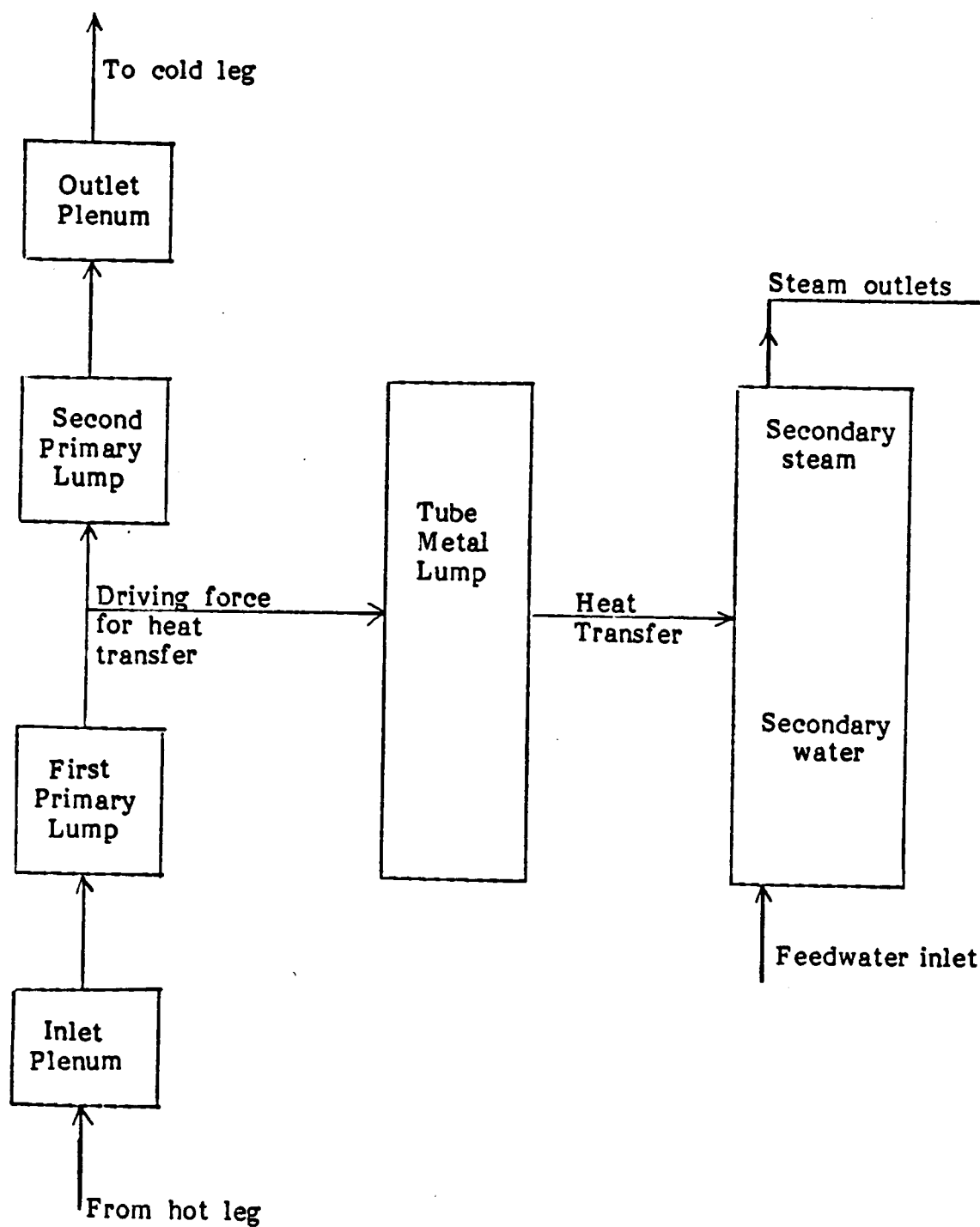


Figure 4.7. Schematic diagram for the simple U-tube steam generator model.

8. Parallel flow branch boiling transfer section, tube metal lump 2 (MTL2)
 9. Counter flow branch subcooled heat transfer section, tube metal lump 4 (MTL4)
- C. Secondary side lumps
10. Subcooled secondary fluid lump (SFSL)
 11. Boiling secondary fluid lump (SFBL)
 12. Drum equivalent secondary fluid lump (SFDRL)
 13. Down comer secondary fluid lump (SFDCL)

A schematic diagram for the model is shown in Figure 4.8. The governing set of equations is derived in Appendix C. Interested readers are referred to Ali's dissertation.

Pressurizer Pressure Control System

The pressurizer pressure control system model is taken from Westinghouse documentation. In this work, only the heater control system and the spray control system are included. Block diagrams for the heater control system and the spray control system are shown in Figures 4.9 and 4.10, respectively. The governing set of equations is as follows (see Appendix B for details):

$$\delta Q = \text{GAIN1} \left(\delta \text{PR} + \frac{1}{\text{TPR1}} \delta \text{D1} + \text{TPR2} \frac{d}{dt} \delta \text{PR} \right)$$

$$\frac{d}{dt} \delta \text{D1} = \delta \text{PR}$$

$$\delta \text{WSP} = \text{GAIN2} \left(\delta \text{PR} + \frac{1}{\text{TPR3}} \delta \text{D2} + \text{TPR4} \frac{d}{dt} \delta \text{PR} \right)$$

$$\frac{d}{dt} \delta \text{D2} = \delta \text{PR}$$

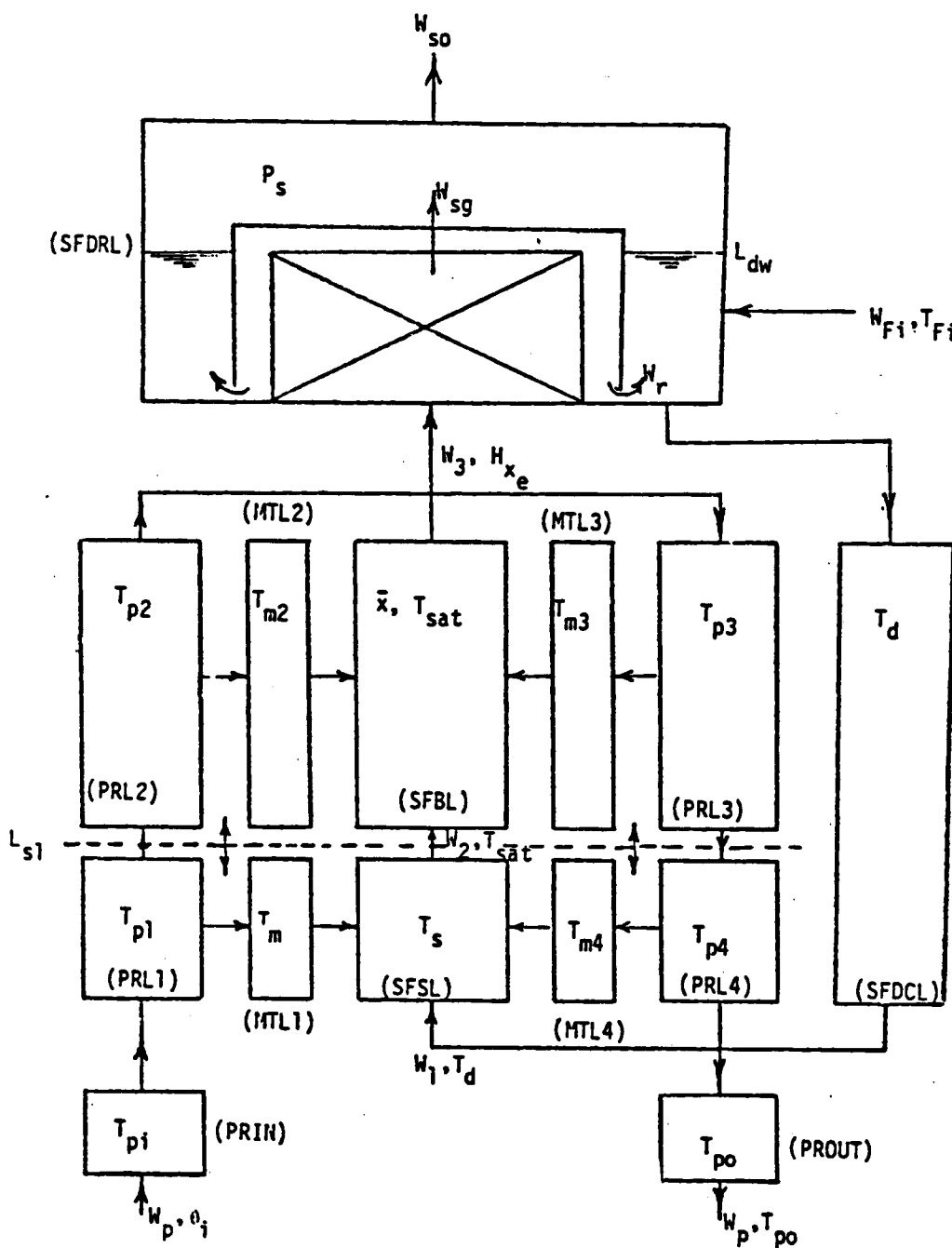


Figure 4.8. Schematic diagram for the detailed steam generator model.

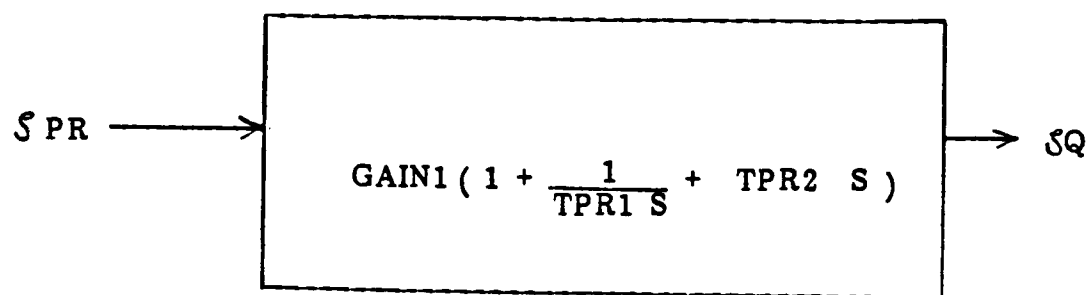


Figure 4.9. Block diagram for the pressurizer heater control system.

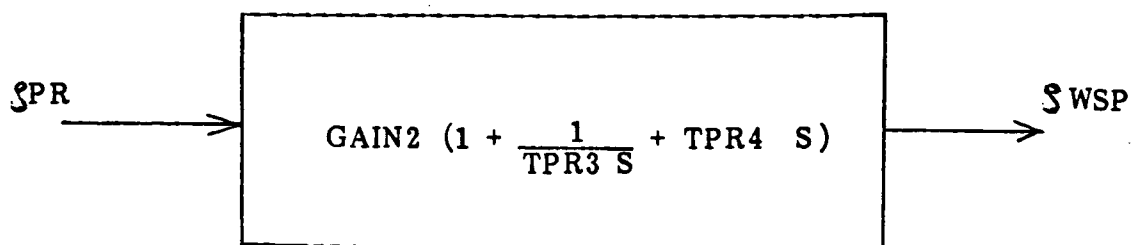


Figure 4.10. Block diagram for the pressurizer sprayer control system.

Summary

Theoretical models for the reactor core, pressurizer, steam generator, and the pressurizer controller are presented in this chapter. As noticed, for each model, a set of first order differential and algebraic equations was obtained. These sets of equations will be used in later chapters to establish a working library for the MMP. Additional theoretical models may be added to the library if necessary.

CHAPTER V

METHOD OF SOLUTION AND THE BASIC APPROACH FOR DEVELOPING A MODULAR MODELING PROGRAM

Introduction

The Modular Modeling System developed by Babcock and Wilcox Company utilizes advanced simulation languages such as ASCL, EASY and CSMP to establish computational compatibility between modules that represent physical systems of interest. Advanced simulation languages such as the ones listed above require large computer systems. In this work, efforts were made to develop a modular modeling program that does not rely on large computer facilities and is adequate for moderate size micro or minicomputers.

In this chapter, a description of the numerical scheme that was adopted in this work to solve a set of first order nonlinear and/or linear differential equations is presented. The basic approach for developing a modular modeling program is also presented. In the next chapter, the main parts that make up the MMP (i.e., the driver, simulator and library) are described.

Method of Solution

The lumped parameter modeling approach results in a set of linear and/or nonlinear first order differential and algebraic equations. The time response of the theoretical model to a forcing function can be obtained by solving the governing set of equations numerically. In this work, the variable step size Runge-Kutta-Fehlberg Algorithm

(VSSRKFA) and the Variable step size Adams-Bashforth-Moulton Algorithm (VSSABMA) are used to solve the sets of equations of interest numerically. The Runge-Kutta-Fehlberg method requires evaluation of the following functions:

$$F1 = hf(X) \quad 5.1$$

$$F2 = hf(X + \frac{1}{4} F1) \quad 5.2$$

$$F3 = hf(X + \frac{3}{32} F1 + \frac{9}{32} F2) \quad 5.3$$

$$F4 = hf(X + \frac{1932}{2197} F1 - \frac{7200}{2197} F2 + \frac{7296}{2197} F3) \quad 5.4$$

$$F5 = hf(X + \frac{439}{216} F1 - 8 F2 + \frac{3680}{513} F3 - \frac{845}{4104} F4) \quad 5.5$$

$$F6 = hf(X - \frac{8}{27} F1 + 2F2 - \frac{3544}{2565} F3 + \frac{1859}{4104} F4 - \frac{11}{40} F5) \quad 5.6$$

where

h = time step

f = function of interest

Using the above formulas for $F1$ through $F6$, two estimates for future value of $X(t)$ may be obtained using the Runge-Kutta Fourth order procedure:

$$X1(t+h) = X(t) + \frac{25}{216} F1 + \frac{1408}{2565} F3 + \frac{2197}{4104} F4 - \frac{1}{5} F5 \quad 5.7$$

and the Runge-Kutta fifth order procedure:

$$X2(t+h) = X(t) + \frac{16}{135} F1 + \frac{6656}{12825} F3 + \frac{38561}{56430} F4 - \frac{9}{50} F5 + \frac{2}{55} F6 \quad 5.8$$

The difference between $X_1(t+h)$ and $X_2(t-h)$ is an estimate of the local truncation error in the fourth order procedure. The local truncation error is the error that occurs when the Taylor series is truncated after a finite number of terms.

$$E(X+h) = X_1(t+h) - X_2(t+h) \quad 5.9$$

The Runge-Kutta-Fehlberg method is a single step numerical method. In other words, if the solution $X(t)$ is known at a particular point t , then $X(t+h)$ can be computed without knowing the solution at points earlier than t . A more efficient method is the Adams-Bashforth-Moulton method which utilizes the values of X at $X(t-h)$, $X(t-2h)$, . . . to compute the new value $X(t+h)$. Such a method is called the multistep method. It has the obvious drawback that at the beginning of the numerical solution no prior values for X are available. In order to solve this problem, the numerical solution scheme is started with a single step method and then transferred to a multistep method for more efficiency as soon as enough starting values have been computed.

The Adams-Bashforth-Moulton method is a multistep method. It consists of a predictor formula and a corrector formula. The predictor formula (Adams-Bashforth Formula) is

$$\begin{aligned} X(t+h) = X(t) + \frac{h}{24} (55F(X(t)) - 59F(X(t-h)) + 37F(X(t-2h)) \\ - 9F(X(t-3h))) \end{aligned} \quad 5.10$$

while the corrector formula (Adams-Moulton Formula) is

$$\begin{aligned} X(t+h) = X(t) + \frac{h}{24} (9F(X(p)) + 19F(X(t)) - 5F(X(t-h)) \\ + F(X(t-2h))) \end{aligned} \quad 5.11$$

and the single step error at $t+h$ is given by

$$E(t+h) = \frac{19}{270} \frac{X(c) - X(p)}{X(c)} \quad 5.12$$

where

$X(p)$ = Future value of X as predicted by the predictor formula.

$X(c)$ = Future value of X as obtained from the corrector formula.

The error per step is a necessary parameter to initiate an action to change the time step size in order to insure accuracy of the solution. An error range between 0.5×10^{-5} and 0.5×10^{-10} is considered an acceptable error. If the error per step is less than 0.5×10^{-10} then the step size is multiplied by 2 while if the error per step is greater than 0.5×10^{-5} then the step size is divided by 2. This approach for controlling the step size has the obvious drawback that the algorithm may end up lost in an infinite loop as shown in Figure 5.1. As the figure illustrates, the loop began with a very small error (i.e., the error is less than 0.5×10^{-10}), multiplying the step size by 2 yields large error (i.e., the error is greater than 0.5×10^{-5}), dividing the step size by 2 yields the same small error mentioned above, multiplying the step size by 2 yields the same large error as above and so on. In order to avoid such a possibility, an algorithm to monitor for this kind of infinite loop, pick up the smallest step size and proceed to the next step was developed.

The Runge Kutta-Fehlberg Formula and the Adams-Bashforth-Moulton Formula are easy to program on a digital computer. A typical structure for such a program is shown in Figure 5.2. The main body of the program consists of the necessary formulas and the error control scheme

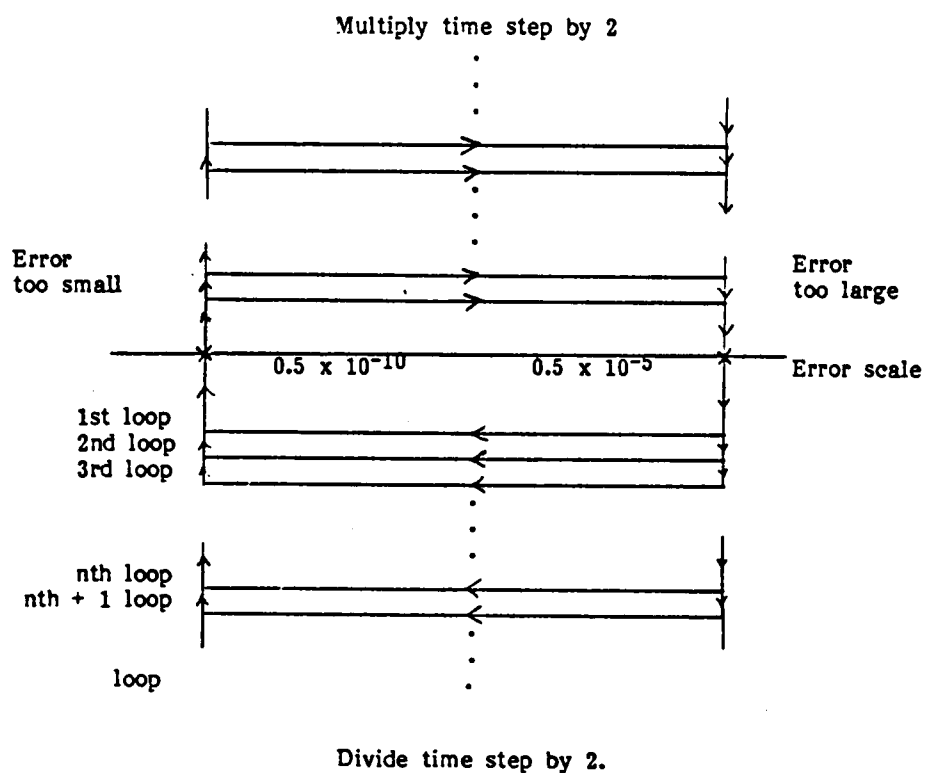


Figure 5.1. Schematic diagram for a possible infinite loop that may occur when using the Runge-Kutta-Fehlburg and Adams-Bashforth-Moulton algorithms.

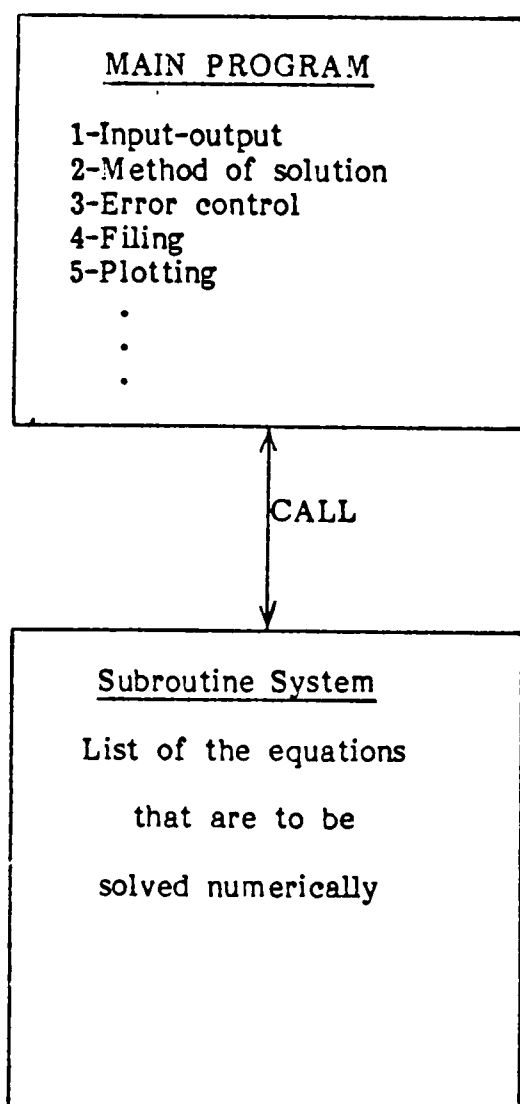


Figure 5.2. Schematic diagram for a typical algorithm that utilizes the Runge-Kutta-Fehlberg method and the Adams-Bashforth-Moulton method.

plus several other options. The subroutine consists of a list of the equations to be solved. Let us assume that the following set of equations represents a theoretical model of interest.

$$\frac{d}{dt} X1 = 3*X1*X2 + X2**2 - X3 + 1$$

$$\frac{d}{dt} X2 = X1**2 + X3**2 - 2*X2$$

$$\frac{d}{dt} X3 = X2 + X1$$

A typical subroutine for the above set of equations is the following (with F replacing the derivative term).

Subroutine statement

Dimension statement

F(1) = 3*X1*X2 + X2**2 - X3 + 1

F(2) = X1**2 + X3**2 - 2*X2

F(3) = X2 + X1

Return statement

End statement.

The above subroutine consists of three main sections as shown below.

Section 1: Subroutine statement

Dimension statement

Section 2: F(1) = 3*X1*X2 + X2**2 - X3 + 1

F(2) = X1**2 + X3**2 - 2*X2

F(3) = X2 + X1

Section 3: Return statement

End statement

A slightly modified version for the above subroutine is the following:

Section 1: Subroutine statement

Dimension statement

N = 0

Section 2: N = N + 1

$F(N) = 3X1 \cdot X2 + X2^2 - X3 + 1$

$F(N+1) = X1^2 + X3^2 - 2 \cdot X2$

$F(N+2) = X2 + X1$

N = N + 2

Section 3: Return statement

End statement

Section 2 in the above subroutine consists of a list of equations that govern the dynamic behavior of the assumed theoretical model. For a system that is composed of a set of coupled components, each component can be modeled separately and an overall model for the whole system can be obtained by merging theoretical models of different components (note that section 2 in the above example subroutine is considered to be the theoretical model for the assumed system) into one model representing the whole system. The governing set of equations for each model representing a single system can be programmed on a digital computer in the same manner as shown in the above subroutine example. Consider the following two sets of equations:

Set 1: $\frac{dX1}{dt} = X1^2 + X2^2 - 3 \cdot X3 + X4 - 2$

$$\frac{dX_2}{dt} = X_1 + X_2 - X_3$$

$$\frac{dX_3}{dt} = X_3^2 + X_1 - 4X_2 - 3X_4$$

and

$$\text{Set 2: } \frac{dX_4}{dt} = X_4^2 - 3X_5 + 1$$

$$\frac{dX_5}{dt} = X_4 - X_5$$

Assuming that each set of equations represents a theoretical model for a physical system that is coupled to another physical system (note the coupling between the first and the second equation through the X_4 term) then, the above set of equations may be programmed as follows:

Set 1: $N = N + 1$

$$F(N) = X_1^2 + X_2^2 - 3X_3 + X_4 - 2$$

$$F(N+1) = X_1 + X_2 - X_3$$

$$F(N+2) = X_3^2 + X_1 - 4X_2 - 3X_4$$

$$N = N + 2$$

Set 2: $N = N + 1$

$$F(N) = X_4^2 - 3X_5 + 1$$

$$F(N+1) = X_4 - X_5$$

$$N = N + 1$$

The above sets of equations can be merged into one larger set to obtain (including section 1 and 3 mentioned earlier in this chapter):

Subroutine statement

Dimension statement

$$N = 0$$

$$N = N + 1$$

$$F(N) = X1^{**2} + X2^{**2} - 3*X3 + X4 - 2$$

$$F(N+1) = X1 + X2 - X3$$

$$F(N+2) = X3^{**2} + X1 - 4*X2 - 3*X4$$

$$N = N + 2$$

$$N = N + 1$$

$$F(N) = X4^{**2} - 3*X5 + 1$$

$$F(N+1) = X4 - X5$$

$$N = N + 1$$

Return statement

End statement.

The procedure listed above provides a means to couple one set of equations to another (Given the fact that the examples used above do not represent real physical systems, the procedures used to couple one set to another are, in principle, valid to apply to physical systems. Some modifications are necessary and will be explained in the next chapter). Once the set of equations that govern the dynamic behavior of the system is known, it is easy to program it as done in the previous examples. If the system of interest is made up of several components, each component will have its own theoretical model that can be programmed using the same procedures mentioned above. These programs will form a library of models for the system of interest. Based on the user request, models (or sets of equations) can be called from the library, merged into one large model and executed. Once this is done, the system subroutine will have the following form:

Subroutine statement

Dimension statement

$N = 0$

$N = N + 1$

$F(N) = . .$

$F(N+1) = . .$

$F(N+2) = . .$

$N = N + 2$

$N = N + 1$

$F(N) = . .$

$F(N+1) = . .$

$F(N+2) = . .$

$F(N+3) = . .$

$N = N + 3$

$N = N + 1$

.

.

.

$N = N + 1$

.

.

.

$N = N + 1$

.

.

.

.

Return statement

End statement.

Summary

In this chapter, Runge-Kutta-Fehlberg and Adams-Bashforth-Moulton formulas for solving a set of first order ordinary differential equations were presented as well as the procedures for coupling a set of

equations to another set of equations. The numerical techniques mentioned above require that the set of equations be placed in a subroutine that can be called by a call statement in the main body of the program. Using this approach the subroutine was divided into three main sections (section 1, section 2 and section 3). Section 2 contains the list of equations that are required to be solved for future values. Each set of equations may be written in a separate program. Programs (or lists of equations) of several components will form a library. Programs in the library (note each program has a structure similar to the section called section 2 in the examples in this chapter) can be merged together, merged with the main program and then executed.

CHAPTER VI

THE MODULAR MODELING PROGRAM

Introduction

The Modular Modeling Program (MMP) is a computer package that was developed to perform accurate analysis of transients in dynamic systems. In this work, the application of this package is limited to analysis of operational transients in a typical pressurized water reactor. However, this should not limit the application of this package to nuclear systems.

The MMP consists of three major components, the simulator, the library and the driver. The simulator and the library are written using the FORTRAN language. The driver program is written using the MIC facilities on the DEC-10 system. The driver program consists of several system commands. Commands such as COPY, SAVE, DELETE and PROTECT are considered system commands. These commands may be programmed to form a program to perform a specific task.

In this chapter, each of the above programs will be discussed separately. The programs are listed in Appendix D.

The Simulator

The computer program SMLTOR·FOR was developed as a part of this work. The purpose of this program is to function as a substitute to more advanced simulation languages. As mentioned in the previous chapters, advanced simulation languages require large computers. The author made every effort to make the MMP package suitable for mini or

microcomputers. SMLTOR·FOR provides the user with a means to obtain the response of his/her theoretical model with minimal programming effort. Given the fact that the computer program SMLTOR·FOR does not contain all the options that are available in a more advanced simulation package (i.e., options such as block diagram algebra are not available as a part of the SMLTOR·FOR program), it is still comparable as far as solving a set of differential equations is of concern. Roughly speaking, when using an advanced simulation language package, the user provides the package with his/her list of equations (certain rules should be followed) and the package generates the response as a function of time. The same thing may be said about the SMLTOR·FOR program. The user presents his/her set of equations (again certain rules should be followed) and SMLTOR·FOR generates the response as a function of time. This program is considered the cornerstone of this work.

The computer program SMLTOR·FOR made the idea of developing a modular modeling package that is suitable for mini and microcomputers an attainable goal. This is due to the fact that SMLTOR·FOR is capable of solving a set of equations stored in a file (file structure will be discussed in the next section) in the library. The library may contain several related or unrelated files (i.e., files or theoretical models need not be restricted to nuclear systems). The user may choose one file, merge it with SMLTOR·FOR and execute the case. The user may also choose several files, merge them with SMLTOR·FOR and execute the case.

A list of the computer program SMLTOR·FOR is shown in Appendix D. The program consists of the following sections.

- a. a section that makes use of the Runge-Kutta-Fehlberg method to predict the future values of the variable of interest (see equations 5.1 through 5.8 in Chapter V). This method is used for the first four time steps.
- b. an error control section that makes use of equation 5.9 in Chapter V to limit the error per step to a value between 0.5×10^{-5} and 0.5×10^{-10} .
- c. a section that makes use of the Adams-Bashforth-Moulton method to predict the future values of the variable of interest (see equations 5.10 and 5.11 in Chapter V).
- d. an error control section that makes use of equation 5.12 in Chapter V to limit the error per step to a value between 0.5×10^{-5} and 0.5×10^{-10} .
- e. a filing section that places the response of each variable in the theoretical model(s) as a function of time in a separate file. Initially, all data are stored in one file in order to reduce the required computer time to execute the case of interest.

The computer program SMLTOR-FOR requires two inputs, the execution time and the sampling rate. The execution time is the total time to be spanned in the simulation. This input should be in seconds and is decided by the user. The second input, the sampling rate, is the rate at which data are to be saved on the computer disk. This option is necessary when the theoretical model contains a variable that requires a very short time step. The power equation in the theoretical reactor core model is an example. The power equation (with no prompt jump

assumption) requires a time step as small as 0.25×10^{-2} sec. Assuming that data are to be collected for 100 seconds for 50 variables that make up a theoretical model, then 2×10^6 data points should be stored on the user disk. When using the sampling rate option, user specifies the interval at which data should be stored on the disk. If the input is one second then data will be stored on the disk every one second. Compared to the previous example, only 5×10^3 data points will be saved on the user disk. SMLTOR·FOR is also designed to handle sets of coupled first order differential equations that represent physical systems of interest. When dealing with physical systems (as compared to the pure mathematical cases discussed in the previous chapter), special attention should be given to the interfaces between one system and another. This topic will be postponed to the end of this chapter.

The Library

The Modular Modeling Program library consists of several files that represent theoretical models of several components in a typical PWR. A list of the files that are included in the library is shown in Table 6.1. SMLTOR·FOR recognizes the file only if it is written in a specific format (to be discussed later). Therefore, all files in the library have a common structure.

Let us assume that the following equation represents a theoretical model of interest (please note that there is no theoretical or physical reasoning behind the formulation of this equation).

$$\frac{d}{dt} X(t) = (3 \sin (X) + \cos (X))^2/9 + \tan (X) + \ln (X) \quad 6.1$$

TABLE 6.1
LIST OF THE LIBRARY FILES

System	File Name
Core	NCORE5.FOR: A lumped parameter model for a pressurized water reactor core (5 lumps).
	PCORE5.FOR: A lumped parameter model for a PWR core with the prompt jump assumption (5 lumps).
	NCORE17.FOR: A lumped parameter model for a PWR core (17 lumps).
	PCOR17.FOR: A lumped parameter model for a PWR core with the prompt jump assumption (17 lumps).
	NCOR47.FOR: A lumped parameter model for a PWR core (47 lumps).
	PCOR47.FOR: A lumped parameter model for a PWR core with the prompt jump assumption (47 lumps).
Hot Leg Piping	HOTLEG.FOR: A lumped parameter model for a pressurized water reactor hot leg piping.
Cold Leg Piping	CLDLEG.FOR: A lumped parameter model for a pressurized water reactor cold leg piping.
Pressurizer	PRESS.FOR: A lumped parameter model for a pressurized water reactor pressurizer.
Steam Generator	SIMSGR.FOR: A lumped parameter model for a pressurized water reactor U-tube steam generator (6 lumps).
	DETSGR.FOR: A lumped parameter model for a pressurized water reactor U-tube steam generator (15 lumps).

In order to establish a file for the above equation in the library, the user creates the file and gives it a name. The file will include the following lines:

```
ICOUNT = IPRIME + 1
```

```
X = X2(M,ICOUNT)
```

```
F(M,ICOUNT) = (3*sin(X) + cos(X))*2/9 + tan(X) + ln(X)
```

```
IPRIME = ICOUNT
```

The variable M serves as a counter for the time step. When M is four, SMLTOR·FOR switches from the Runge-Kutta-Fehlberg scheme to the Adams-Bashforth-Moulton scheme. The variable ICOUNT preserves the order of the variables. When ICOUNT is 1, then the function F(M,ICOUNT) is the first differential equation in a set of equations and the function F(M,ICOUNT + 10) is the eleventh differential equation in a set of equations (assuming that ICOUNT is one). IPRIME preserves the coupling to the next set of equations (see the first and the last lines in the above example). The variable F(M,ICOUNT) represents the derivative of the variable X with respect to time.

Another example is the following (again there is no theoretical or physical reasoning behind the following set of equations).

$$\frac{dX_1}{dt} = \sin(X_2) + 3X_1X_2 + 5 \quad 6.2$$

$$\frac{dX_2}{dt} = X_1^2 + 3\sin(X_1X_2) + 4X_3 \quad 6.3$$

$$\frac{dX_3}{dt} = X_2 + X_1 - X_3 \quad 6.4$$

In order to establish a file for the above set of equations, the user creates a file and gives it a name. The file will contain the following lines.

$$\text{ICOUNT} = \text{IPRIME} + 1$$

$$X1 = X2(\text{M}, \text{ICOUNT})$$

$$X2 = X2(\text{M}, \text{ICOUNT} + 1)$$

$$X3 = X2(\text{M}, \text{ICOUNT} + 2)$$

$$F(\text{M}, \text{ICOUNT}) = \sin(X2) + 3*X1**2*X2 + 5$$

$$F(\text{M}, \text{ICOUNT} + 1) = X1**2 + 3*\sin(X1*X2) + 4*X3$$

$$F(\text{M}, \text{ICOUNT} + 2) = X2 + X1 - X3$$

$$\text{IPRIME} = \text{ICOUNT} + 2$$

A slightly different case occurs when a set of mixed algebraic and differential equations is to be solved by SMLTOR·FOR. The difference between this case and the cases discussed above is present only if the responses of the algebraic variables are to be saved or sampled on a file along with the differential variable. The following set of equations will be used to demonstrate the above case (please note that there is no physical or theoretical reasoning behind the following set of equations).

$$X1 = X2 + X3 \quad 6.5$$

$$\frac{dX2}{dt} = X1^2 + 4*X2^2 + 4*X3*X1 \quad 6.6$$

$$\frac{dX3}{dt} = X2^2 - 4*X1*X3 \quad 6.7$$

In order to establish a file for the above case in the library, the user creates the file and gives it a name. This file will include the

following lines (assuming that the algebraic variable is not to be saved on an output file as a function of time).

```

ICOUNT = IPRIME + 1
X2 = X2(M,ICOUNT)
X3 = X2(M,ICOUNT + 1)
X1 = X2 + X3
F(M,ICOUNT) = X1**2 + 4*X2**2 - 4*X3*X1
F(M,ICOUNT) = X2**2 - 4*X1*X3
IPRIME = ICOUNT + 1

```

If the algebraic variable is to be saved on an output file as a function of time then the file will include the following:

```

ICOUNT = IPRIME + 1
X2 = X2(M,ICOUNT)
X3 = X2(M,ICOUNT + 1)
X1 = X2 + X3
F(M,ICOUNT) = X1**2 + 4*X2**2 - 4*X3*X1
F(M,ICOUNT) = X2**2 - 4*X1*X3
IF(IFINE.EQ.0) write (23,*) X1
IPRIME = ICOUNT + 1

```

The only new statement is the statement in line 7. IFINE is a parameter that can have a value of 0 or 1. When IFINE is 0, this indicates that the future values of the differential variables (in the above example X2 and X3 are considered differential variables) have been estimated for the current time step. Once IFINE is 0, SMLTOR·FOR sweeps the above set of equations one more time to determine the numerical values of the algebraic variables (in the above example X1

is considered an algebraic variable) based on the estimated values of the differential variable and saves them on an output file (write (23,*) X1 means that X1 is to be saved on disk 23 using the free format).

The only missing element in the above development is the specification of the output files' names. Given a set of mixed algebraic and differential equations, estimate future values of the variables as a function of time are to be saved on appropriate output files. This problem can be solved by expanding the procedures mentioned above to include the output files name. To present the procedures the following set of equations will be used.

$$X1 = X2 + X3 \quad 6.8$$

$$X5 = X4 + X6 \quad 6.9$$

$$\frac{dX2}{dt} = X1 - X5 + X4 \quad 6.10$$

$$\frac{dX3}{dt} = X2 - X3 + X5 \quad 6.11$$

$$\frac{dX4}{dt} = X5 * X6 - X6 * X1 \quad 6.12$$

$$\frac{dX6}{dt} = X1 * X2 - X3 * X4 \quad 6.13$$

A complete file for the above case is the following

```
ICOUNT = IPRIME + 1
```

```
IJ = IK + 1
```

```
IJA = IJKA + 1
```

```
AFRI(IJA) = 'X1.DAT'
```

```
AFRI(IJA+1) = 'X5.DAT'
```

files names for the algebraic variables are stored in a different array than the differential variables

```
FR1(IJ) = 'X2.DAT'
```

```
FR1(IJ+1) = 'X3.DAT'
```

```
FR1(IJ+2) = 'X4.DAT'
```

```
FR1(IJ+3) = 'X6.DAT'
```

```
X1 = X2 + X3
```

```
X5 = X4 + X6
```

```
F(M,ICOUNT) = X1 - X5 + X4
```

```
F(M,ICOUNT) = X1 - X5 + X4
```

```
F(M,ICOUNT + 1) = X2 - X3 + X5
```

```
F(M,ICOUNT + 2) = X5*X6 - X6*X1
```

```
F(M,ICOUNT + 3) = X1*X2 - X3*X4
```

```
IF(IFINE.EQ.0) write (23,*) X1
```

```
IF(IFINE.EQ.0) write (23,*) X5
```

```
IPRIME = ICOUNT + 3
```

```
IK = IJ + 3
```

```
IJKA = IJA + 1
```

FR1 is an array that contains names of the output files for the differential variables

The variables AFR1 and FR1 are alphanumeric arrays that hold names of the output files for the algebraic and differential variables respectively. The variables IJ, IK, IJKA and IJA are integer variables that provide proper connection with other files in the library (compare the first three lines and the last three lines in the above example). Files in the MMP library are similar in structure to the examples presented earlier. These examples lead to the following conclusions:

- a. Given a set of mixed algebraic and differential equations, an appropriate file can be constructed in a very short time.

- b. The structure of the file is inherently modular so that several files may be merged together (will be discussed later on) to form a single larger file. The counters ICOUNT, IJ, IK, IJKA and IJA provide an easy contact between several files in the library.

The last two statements will be proved by considering a physical system (such as the primary side in a typical pressurized water reactor), present the governing sets of equations and create an appropriate file for each set.

Consider a primary loop in a typical pressurized water reactor that consists of a nuclear core, hot leg, cold leg and a steam generator (for simplicity the above system was limited to the mentioned components). A simple theoretical model for each of the core and the steam generator will be used. The governing set of equations for each component is presented below. Associated with each set of equations an appropriate computer file is presented. In the next section these files will be merged into one large file. The governing set of equations for the core theoretical model with the prompt jump assumption (see Figure 4.1):

Reactivity Term:

$$\begin{aligned} \text{REACT} = & \text{RH01} + \text{RH02} + \text{ALFAF}(\text{TFUEL} - \text{TFUEL1}) + \text{ALFAC}(\text{TC1} \\ & - \text{TC11}) + \text{ALFAPR}(\text{PR} - \text{PRO}) \end{aligned} \quad 6.14$$

Power Equation:

$$\begin{aligned} \text{POWER} = & \frac{1}{(\text{BTOTAL} - \text{REACT})} (\text{REACT} + \text{OMEGA} * (\text{DECAY1} * \text{PREC1} \\ & + \text{DECAY2} * \text{PREC2} + \text{DECAY3} * \text{PREC3} + \text{DECAY4} * \text{PREC4} \\ & + \text{DECAY5} * \text{PREC5} + \text{DECAY6} * \text{PREC6}) \end{aligned} \quad 6.15$$

Delayed Neutron Group Equations:

$$\frac{d}{dt} \text{PREC1} = \text{BETA1} * \text{POWER} / \text{OMEGA} - \text{DECAY1} * \text{PREC1} \quad 6.16$$

$$\frac{d}{dt} \text{PREC2} = \text{BETA2} * \text{POWER} / \text{OMEGA} - \text{DECAY2} * \text{PREC2} \quad 6.17$$

$$\frac{d}{dt} \text{PREC3} = \text{BETA3} * \text{POWER} / \text{OMEGA} - \text{DECAY3} * \text{PREC3} \quad 6.18$$

$$\frac{d}{dt} \text{PREC5} = \text{BETA5} * \text{POWER} / \text{OMEGA} - \text{DECAY5} * \text{PREC5} \quad 6.20$$

$$\frac{d}{dt} \text{PREC6} = \text{BETA6} * \text{POWER} / \text{OMEGA} - \text{DECAY6} * \text{PREC6} \quad 6.21$$

Fuel Temperature:

$$\frac{d}{dt} \text{TFUEL} = ((\text{POWER} + 1) * \text{POWER1} - \text{HFC1} * \text{AREAFC} * (\text{TFUEL} - \text{TC1})) / (\text{FUELM} * \text{CPFUEL}) \quad 6.22$$

Lower Plenum:

$$\frac{d}{dt} \text{TLP} = (\text{TCLEG} - \text{TLP}) * \text{WP} / \text{PLRM} \quad 6.23$$

Coolant Lumps:

$$\frac{d}{dt} \text{TC1} = (\text{TLP} - \text{TC1}) * \text{WP} / \text{CMC} + \text{HFC1} * \text{AREAFC} * (\text{TFUEL} - \text{TC1}) / (2 * \text{CMC1} * \text{CP1}) \quad 6.24$$

$$\frac{d}{dt} \text{TC2} = (\text{TC1} - \text{TC2}) * \text{WP} / \text{CMC} + \text{HFC1} * \text{AREAFC} * (\text{TFUEL} - \text{TC1}) / (2 * \text{CMC} * \text{CP1}) \quad 6.25$$

Upper Plenum:

$$\frac{d}{dt} \text{TUP} = (\text{TC2} - \text{TUP}) * \text{WP} / \text{UPM} \quad 6.27$$

where

REACT = total reactivity in the reactor

RH01 = reactivity due to the manual control rod

RH02 = reactivity due to the automatic control rod

ALFAF = fuel temperature coefficient of reactivity

TFUEL = fuel temperature

TFUEL1 = fuel temperature at steady state

ALFAC = coolant temperature coefficient of reactivity

TC1 = coolant temperature in the core first coolant lump

TC2 = coolant temperature in the core first coolant lump at steady
state

ALFAPR = coolant pressure coefficient of reactivity

PR = primary side pressure

PRo = primary side pressure at steady state

POWER = normalized reactor power

POWER1 = reactor power at steady state

BTOTAL = total delayed neutron fraction

OMEGA = neutron generation time

DECAY1 - DECAY6 = decay constants for delayed neutron groups 1-6

PREC1 - PREC6 = delayed neutron concentrations for groups 1-6

BETA1 - BETA6 = fractions of delayed neutrons for groups 1-6

HFC1 = overall fuel to coolant heat transfer coefficient

AREAFC = overall fuel to coolant heat transfer area

FUELM = total fuel mass

CPFUEL = specific heat for the fuel

TLP = coolant temperature in the core lower plenum

TCLEG = coolant temperature in the cold leg piping

WP = coolant flow rate inside the core

PLRM = mass of the coolant in the core lower plenum

CMC = mass of the coolant in the coolant lumps

CP1 = specific heat for the coolant

TC2 = coolant temperature in the second coolant lump

TUP = coolant temperature in the core upper plenum

UPM = mass of the coolant in the core upper plenum

An appropriate file for the above set of equations is the following.

ICOUNT = IPRIME + 1

IJ = IK + 1

IJA = IJKA + 1

AFR1(IJA) = 'REACT.DAT'

AFR1(IJA) = 'POWER.DAT'

FR1(IJ) = 'PREC1.DAT'

FR1(IJ+1) = 'PREC2.DAT'

FR1(IJ+2) = 'PREC3.DAT'

FR1(IJ+3) = 'PREC4.DAT'

FR1(IJ+4) = 'PREC5.DAT'

FR1(IJ+5) = 'PREC6.DAT'

FR1(IJ+6) = 'TFUEL.DAT'

FR1(IJ+7) = 'TLP.DAT'

FR1(IJ+8) = 'TC1.DAT'

FR1(IJ+9) = 'TC2.DAT'

FR1(IJ+10) = 'TUP.DAT'

PREC1 = X2(M,ICOUNT)

```

PREC2 = X2(M,ICOUNT+1)
PREC3 = X2(M,ICOUNT+2)
PREC4 = X2(M,ICOUNT+3)
PREC5 = X2(M,ICOUNT+4)
PREC6 = X2(M,ICOUNT+5)
TFUEL = X2(M,ICOUNT+6)
TLP = X2(M,ICOUNT+7)
TC1 = X2(M,ICOUNT+8)
TC2 = X2(M,ICOUNT+9)
TUP = X2(M,ICOUNT+10)
REACT = RH01+RH02+ALFAF*(TFUEL-TFUEL1)+ALFAC*(TC1-TC11) +
        ALFAPR*(PRESS-PRESS1)
POWER = (REACT+OMEGA*(DECA1*PREC1+DECAY2*PREC2+DECAY3*
        PREC3+DECAY4*PREC4+DECAY5*PREC5+decay6*PREC6)/
        (BTOTAL - REACT)
F(M,ICOUNT) = BETA1*POWER/OMEGA - DECAY1*PREC1
F(M,ICOUNT+1) = BETA2*POWER/OMEGA - DECAY2*PREC2
F(M,ICOUNT+2) = BETA3*POWER/OMEGA - DECAY3*PREC3
F(M,ICOUNT+3) = BETA4*POWER/OMEGA - DECAY4*PREC4
F(M,ICOUNT+4) = BETA5*POWER/OMEGA - DECAY5*PREC5
F(M,ICOUNT+5) = BETA6*POWER/OMEGA - DECAY6*PREC6
F(M,ICOUNT+6) = ((POWER+1)*POWER1 - HFC1*AREAFC*(TFUEL
        - TC1))/(FUELM*CPFUEL)
F(M,ICOUNT+7) = (TCLEG - TPLR)*WP*PLRM
F(M,ICOUNT+8) = (TPLR-TC1)*WP.CMC1 + HFC1*AREAFC(TFUEL
        - TFUEL1)/(2*CMC1*CP1)

```

$$F(M, I\text{COUNT}+9) = (TC1=TC2)*WP/CMC1 + HFC1*AREAFC (TFUEL- \\ TFUEL1)/(2*CMC2*CP1)$$

$$F(M, I\text{COUNT}+10) = (TC2-TUP)*WP/UPM$$

IF(IFINE.EQ.0)WRITE(23,*) REACT

IF(IFINE.EQ.0)WRITE(23,*) POWER

IPRIME = ICOUNT + 10

IK = IJ + 10

IJKA = IJA + 1

The governing equations for the cold leg model is listed below:

$$\frac{d}{dt} TCLEG = \frac{WP}{CLEGM} (TSGOP - TCLEG) \quad 6.27$$

where

WP = coolant mass flow rate in the primary loop

CLEGM = mass of the coolant in the cold leg piping

TSGOP = coolant temperature in the steam generator outlet plenum

TCLEG = coolant temperature in the cold leg piping

An appropriate library file for the above equation is the following:

ICOUNT = IPRIME + 1

IJ = IK + 1

FR1(IJ) = 'CLEG.DAT'

TCLEG = X2 (M, ICOUNT)

F(M, ICOUNT) = (TSGOP - TCLEG)*WP/CLEGM

IPRIME = ICOUNT

IK = IJ

The governing equation for the hot leg model is listed below:

$$\frac{d}{dt} THLEG = \frac{WP}{HLEGM} (TUP - THLEG) \quad 6.28$$

where

TUP = coolant temperature in the core upper plenum

THLEG = coolant temperature in the hot leg piping

HLEGM = mass of the coolant in the hot leg piping

An appropriate library file of the above equation is the following:

ICOUNT = IPRIME + 1

IJ = IK + 1

FR1(IJ) = 'HLEG.DAT'

THLEG = X2 (M,ICOUNT)

F(M,ICOUNT) = (TUP - THLEG)*WP/HLEGM

IPRIME = ICOUNT

IK = IJ + 1

The governing set of equations for the simple steam generator theoretical model is listed below (see Appendix D for details and definition of variables).

Steam Generator Primary Inlet Plenum:

$$\frac{d}{dt} TSGIP = (THLEG - TSGIP) * WSG / MSGIP \quad 6.29$$

First Primary Coolant Lump:

$$\frac{d}{dt} TP1 = (TSGIP - TP1) * \frac{WSG}{MP} - \frac{UPM}{MP} \frac{SPM}{CP1} * (TP1 - TMETAL) \quad 6.30$$

Second Primary Coolant Lump:

$$\frac{d}{dt} TP2 = (TP1 - TP2) * \frac{WSG}{MP} - \frac{UPM}{MP} \frac{SPM}{CP1} * (TP1 - TMETAL) \quad 6.31$$

Steam Generator Primary Outlet Plenum:

$$\frac{d}{dt} TSGOP = (TP2 - TSGOP) * \frac{WSG}{MSGOP} \quad 6.32$$

Tube Metal Lump:

$$TS = 0.1453 \cdot PS + 401.1 \quad 6.33$$

$$\frac{d}{dt} T_{METAL} = \frac{UPM \cdot SPM \cdot (TP1 - T_{METAL}) - UMS \cdot SMS \cdot (T_{METAL} - TS)}{MMETAL \cdot CM} \quad 6.34$$

Secondary Side Lump:

Steam enthalpy:

$$HG = -0.029 \cdot PS + 1222.45 \quad 6.35$$

Secondary side water mass:

$$MSW = SWV \cdot (54.63 - 8.36E-3 \cdot PS) \quad 6.36$$

Secondary side steam mass:

$$MSS = SSV \cdot (2.359E-3 \cdot PS - 0.125) \quad 6.37$$

Slope of the water enthalpy in relation with pressure at saturated condition:

$$DHF = 0.1173 \quad 6.38$$

Latent heat of vaporization:

$$HFG = 854.55 - 0.205 \cdot PS \quad 6.39$$

Difference between the liquid and steam specific volumes at saturated condition:

$$VFG = 1.2 - 7.98 E-4 \cdot PS \quad 6.40$$

Slope of the steam specific volume in relation with pressure at saturated condition:

$$DVG = -7.953 E-4 \quad 6.41$$

Slope of steam enthalpy in relation with pressure at saturated conditions

$$DHG = -0.029$$

Dummy variable:

$$K = (MSW \text{ DHF} + MSS \text{ DHG} - MSS \text{ HFG} \text{ DVG/VFG}) \quad 6.42$$

Secondary pressure:

$$\frac{d}{dt} PS = (UMS \text{ SMS} (TMETAL-TS) - WSO (HG-CP2 \text{ TF1}))/K \quad 6.43$$

An appropriate file for the above set of equations is the following:

$$ICOUNT = IPRIME + 1$$

$$IJ = IK + 1$$

$$FR1(IJ) = 'INLET.DAT'$$

$$FR1(IJ+1) = 'TP1.DAT'$$

$$FR1(IJ+2) = 'TP2.DAT'$$

$$FR1(IJ+3) = 'OUTLET.DAT'$$

$$FR1(IJ+4) = 'TM.DAT'$$

$$FR1(IJ+5) = 'SECON.DAT'$$

$$TSGIP = X2(M, ICOUNT)$$

$$TP1 = X2(M, ICOUNT + 1)$$

$$TP2 = X2(M, ICOUNT + 2)$$

$$TSGOP = X2(M, ICOUNT + 3)$$

$$TM = X2(M, ICOUNT + 4)$$

$$PS = X2(M, ICOUNT + 5)$$

$$F(M, ICOUNT) = (THLEG - TSGIP) * WSG / MSGIP$$

$$F(M, ICOUNT+1) = (TSGIP - TP1) * \frac{WDG}{MP} - \frac{UPM \text{ SPM}}{MP \text{ CP1}} (TP1 - TMETAL)$$

$$F(M, ICOUNT+2) = (TP1 - TP2) * \frac{WSG}{MP} - \frac{UPM \text{ SPM}}{MP \text{ CP1}} (TP1 - TMETAL)$$

$$F(M, ICOUNT+3) = (TP2 - TSGOP) * \frac{WSG}{MSGOP}$$

```

TS = 0.1453*PS + 401.1
F(M,ICOUNT+4) = (UPM SPM*(TP1-TMETAL) - UMS SMS*(TMETAL-TS))/
                (MMETAL CM)
HG = -0.029*PS + 1222.45
SWM = SWV*(54.63-8.36E-3*PS)
SSM = SSV*(2.359E-3*PS-0.125)
DHF = 0.1173
DHG = -0.029
HFG = 854.55 - 0.205*PS
VFG = 1.2 - 7.98E-4*PS
DVG = -7.953E-4
K = (MSW*DHF + MSS*DHG - MSS*HFG*DVG/VFG)
F(M,ICOUNT+5) = (UMS*SMS*(TMETAL-TS) - WSO(HG-CP2*TF1))/K
IPRIME = ICOUNT + 5
IK = IJ + 5

```

The Driver

The computer program DRIVER.MIC is written using the MIC facilities provided by the DEC.10 system at The University of Tennessee. The MIC facilities are similar to the BATCH system on the IBM PC system. Therefore, usage of the MIC facilities does not violate the fact that the MMP is adequate for mini and microcomputer systems. DRIVER.MIC functions to merge several files in the library into one large file. The files in the MMP library (as discussed and shown in the previous section) are structured such that they are computationally compatible with each other. The variables ICOUNT, IJ, IK, IJKA, IJ (see previous

section) are the only variables that are necessary to insure compatibility of the files. DRIVER.MIC consists of several system commands such as COPY, TYPE, DELETE and SAVE. These commands and several others may be programmed to form a program to perform a certain task. The inputs to DRIVER.MIC are the names of the file in the library. In the previous section the following files were discussed:

PCORE5.FOR: Core theoretical model

HOTLEG.FOR: Hot leg theoretical model

CLDLEG.FOR: Cold leg theoretical model

SIMPSG.FOR: Steam generator theoretical model

These files will be used to demonstrate the functions of DRIVER.MIC.

Consider a case in which the core model PCORE5.FOR is to be called from the library and tested for a 10¢ step change in reactivity. This can be accomplished by issuing the following command.

```
DO DRIVER PCORE5.FOR, STOP.FOR
```

The above command will bring PCORE5.FOR from the library, merge it with SMLTOR.FOR and execute the case. The user inputs the duration time, the sampling rate and the forcing term. The output of the test will be several data files that contain the responses of the theoretical model variables as a function of time. Figure 6.1 is a graphical presentation of the response.

Consider a case in which the hot leg model HOTLEG.FOR is to be called from the library and tested for a +10°F step change in the core outlet temperature. This can be accomplished by issuing the following command.

```
DO DRIVER HOTLEG.FOR, STOP.FOR
```

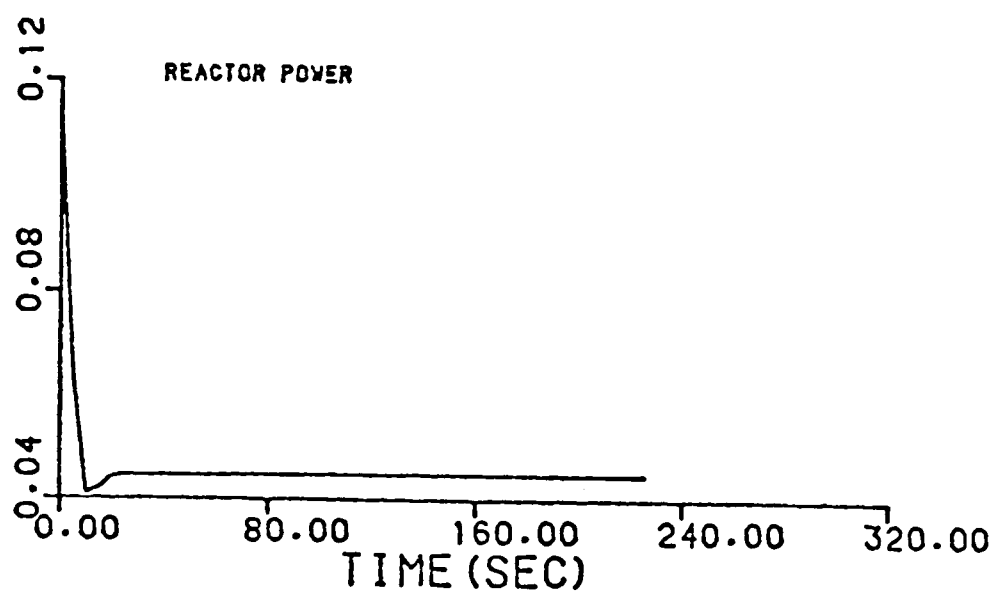
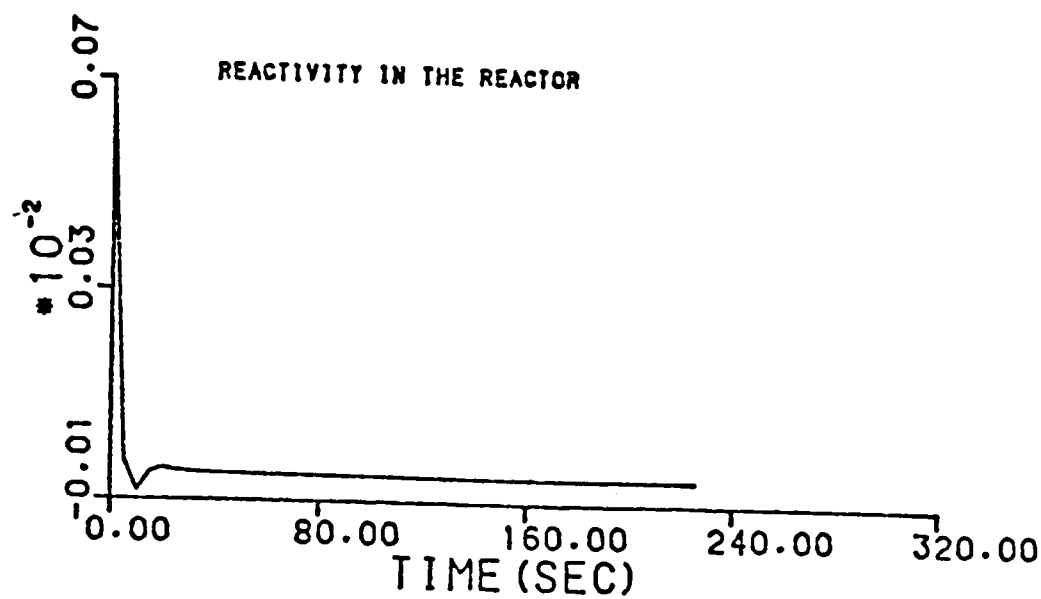


Figure 6.1. Response of the core theoretical model to a +10¢ step change in the external reactivity.

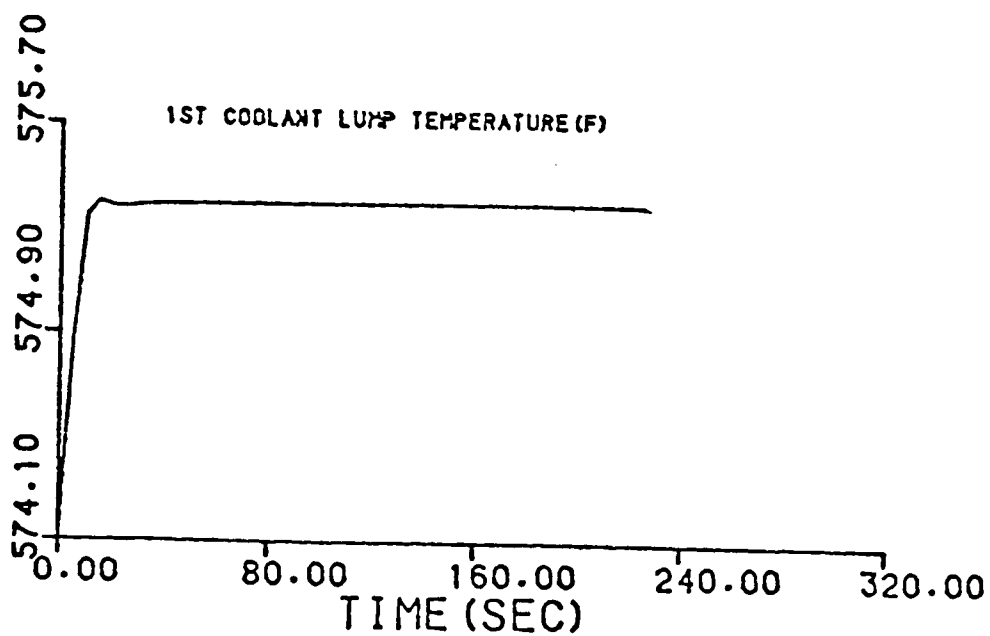
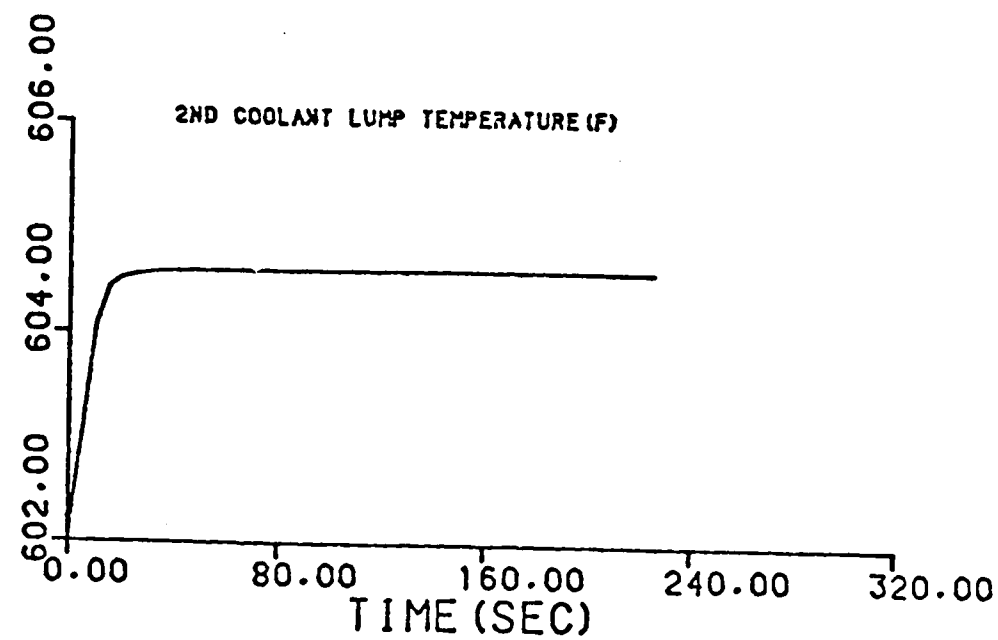


Figure 6.1. (Continued)

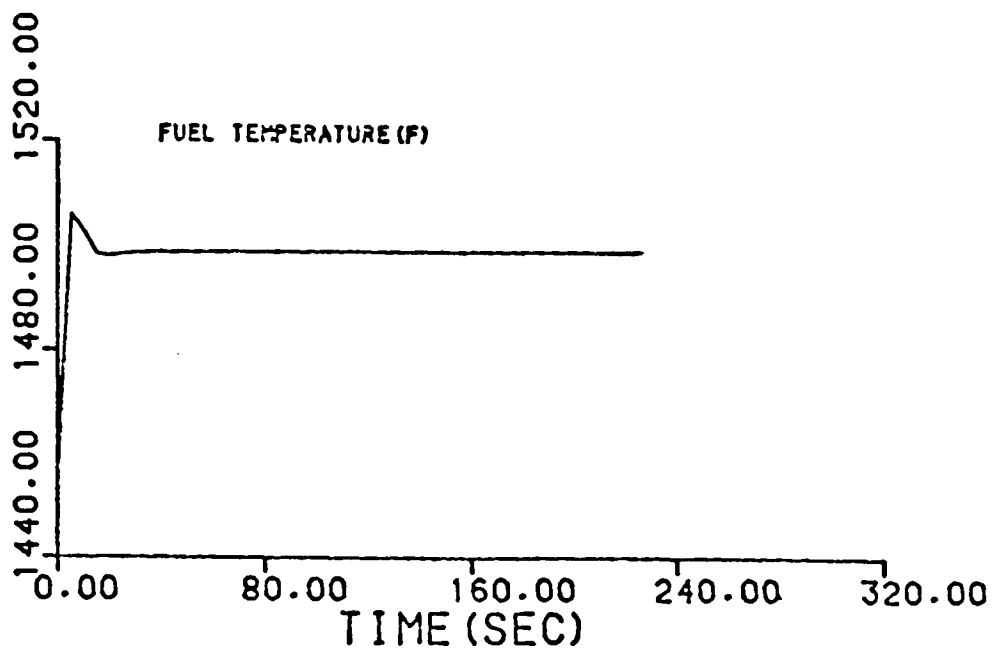
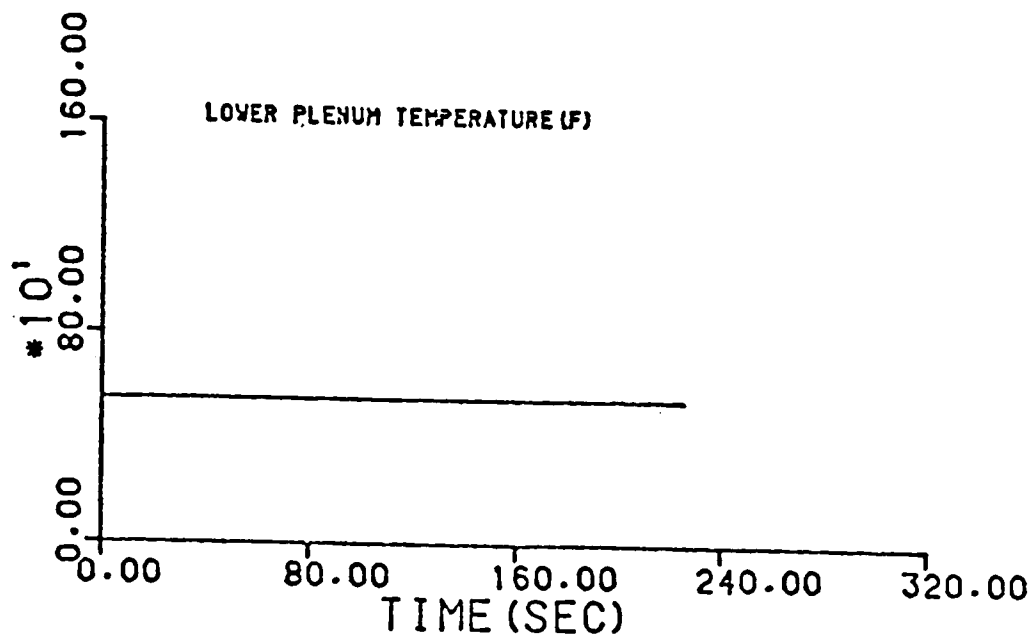


Figure 6.1. (Continued)

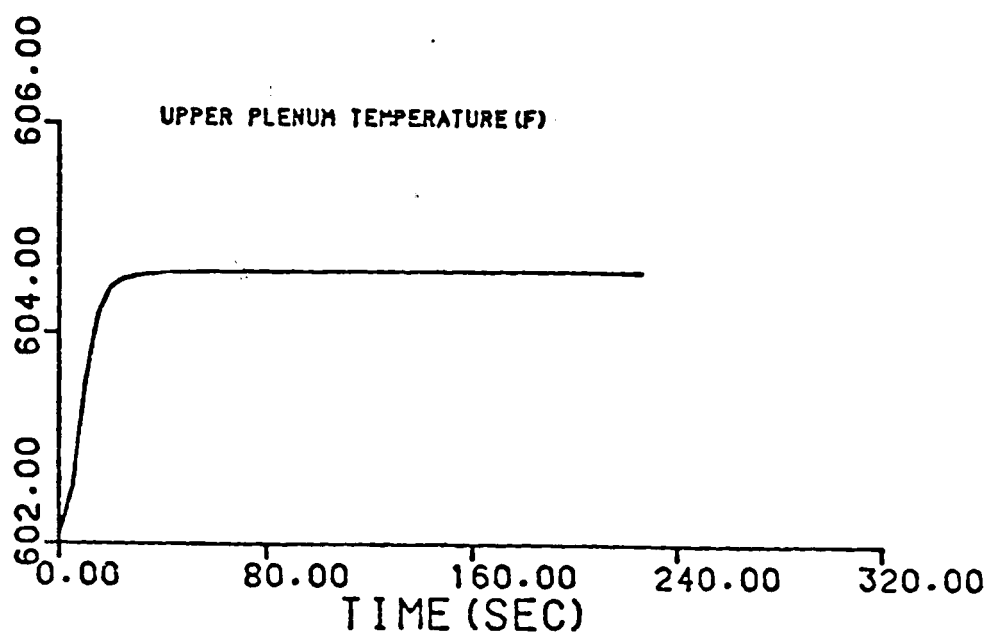


Figure 6.1. (Continued)

Figure 6.2 is a graphical representation of the response as a function of time. The commands

```
DO DRIVER CLDLEG.FOR, STOP.FOR
```

```
DO DRIVER SIMPSG.FOR, STOP.FOR
```

will bring the files CLDLEG.FOR/SIMPSG.FOR from the library, merge it with SMLTOR.FOR and execute the case. Figures 6.3 and 6.4 represent the responses of the cold leg temperature to a +10°F step change in the inlet temperature and the steam generator to a +10% in the steam valve position.

Consider a case in which the core model, the hot leg model, the cold leg model and the steam generator are to be merged into one system and executed the case for a 10¢ step change in the reactivity. This can be accomplished by issuing the following command:

```
DO DRIVER PCORE5.FOR, HOTLEG.FOR, SIMPSG.FOR, CLDLEG.FOR,  
STOP.FOR
```

The above command will merge PCORE5.FOR, HOTLEG.FOR, SIMPSG.FOR, CLDLEG.FOR and SMLTOR.FOR into one large file and execute the case. The user inputs the duration time, the sampling rate and the forcing term. Figure 6.5 is a graphical presentation of the responses.

Summary

The Modular Modeling Program consists of the simulator (SMLTOR.FOR), the library (set of files that represent theoretical models) and the driver (DRIVER.MIC). The simulator functions as a substitute for an advanced simulation language. The library contains system files that are written in a certain format to insure computational

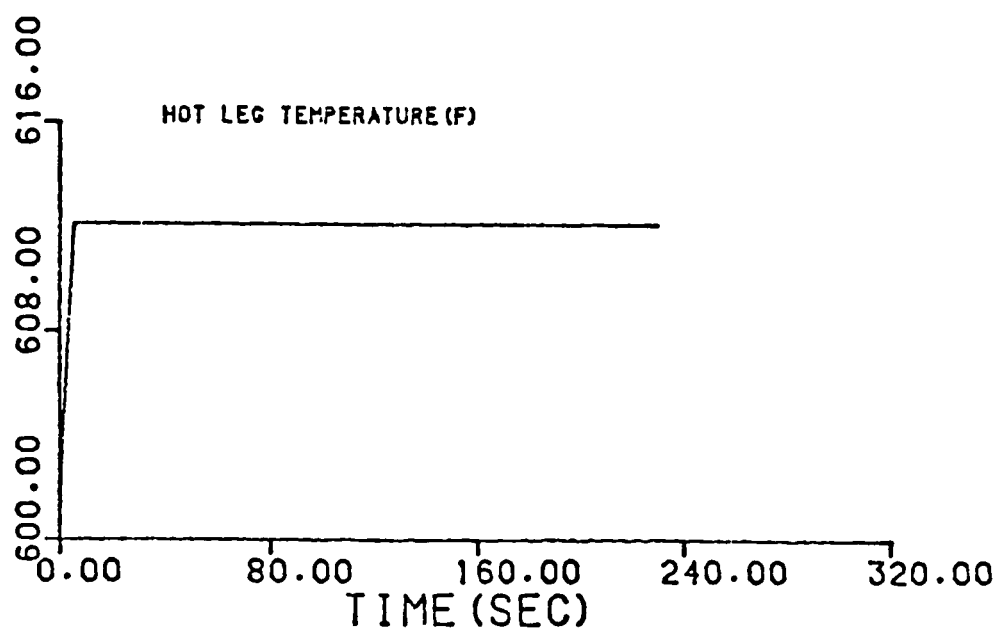


Figure 6.2. Response of the hot leg theoretical model to a +10°F step change in the inlet temperature.

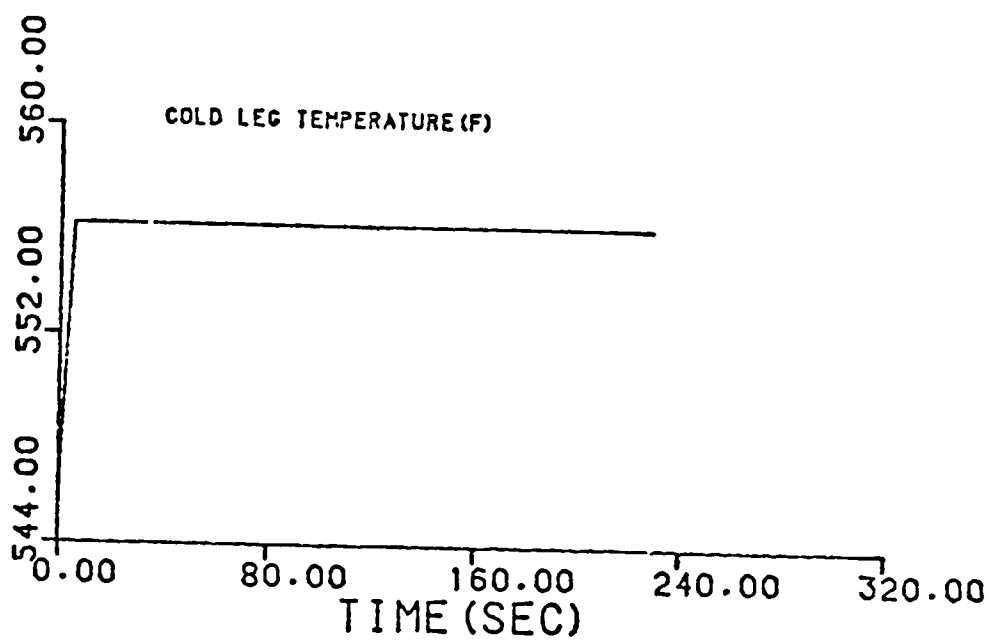


Figure 6.3. Response of the cold leg temperature to a +10°F step change in the inlet temperature.

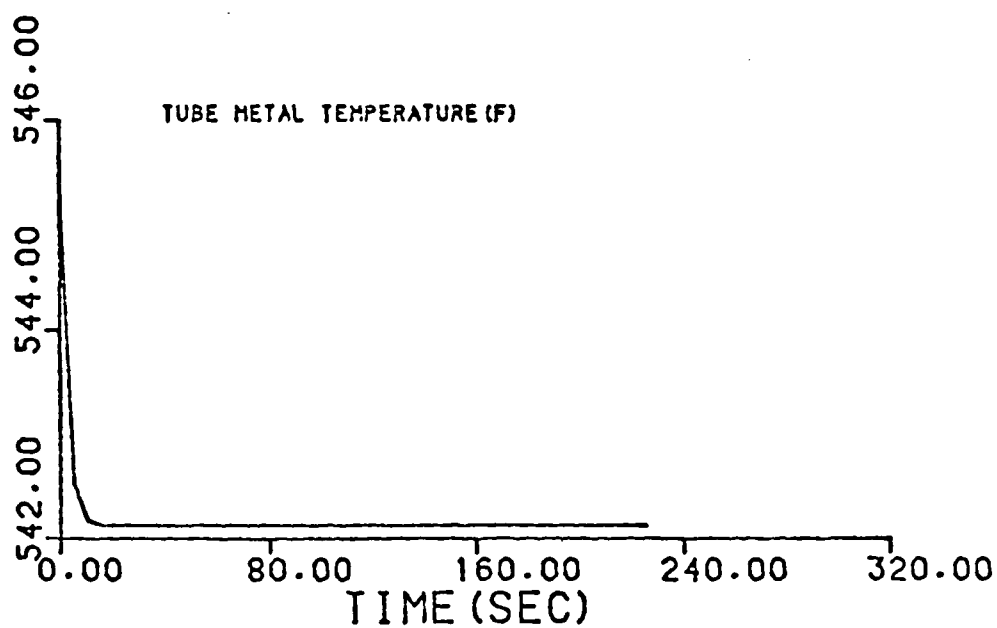
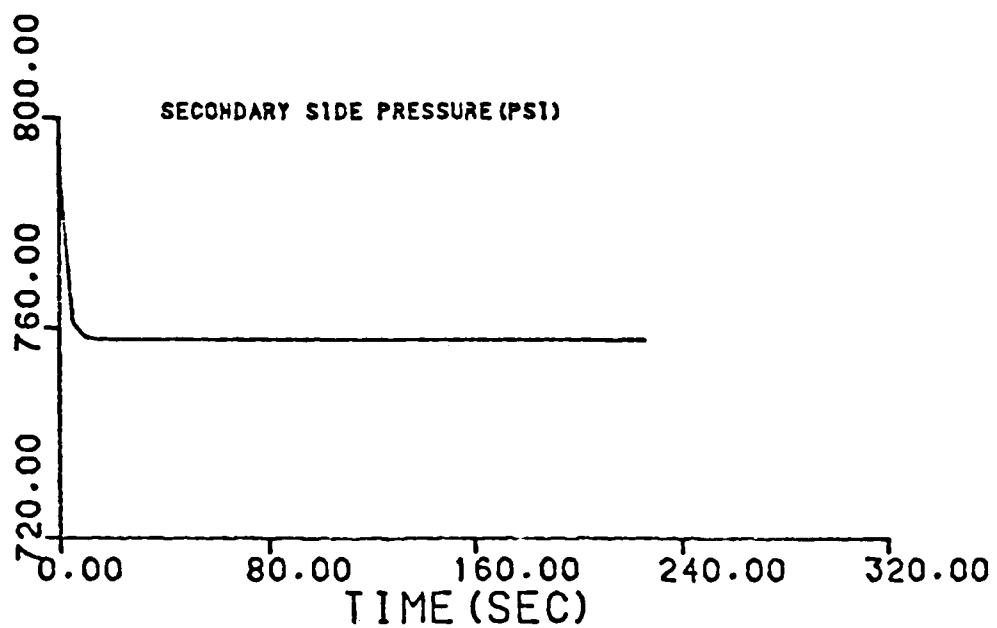


Figure 6.4. Response of the simple steam generator model to a +10% step change in the steam valve.

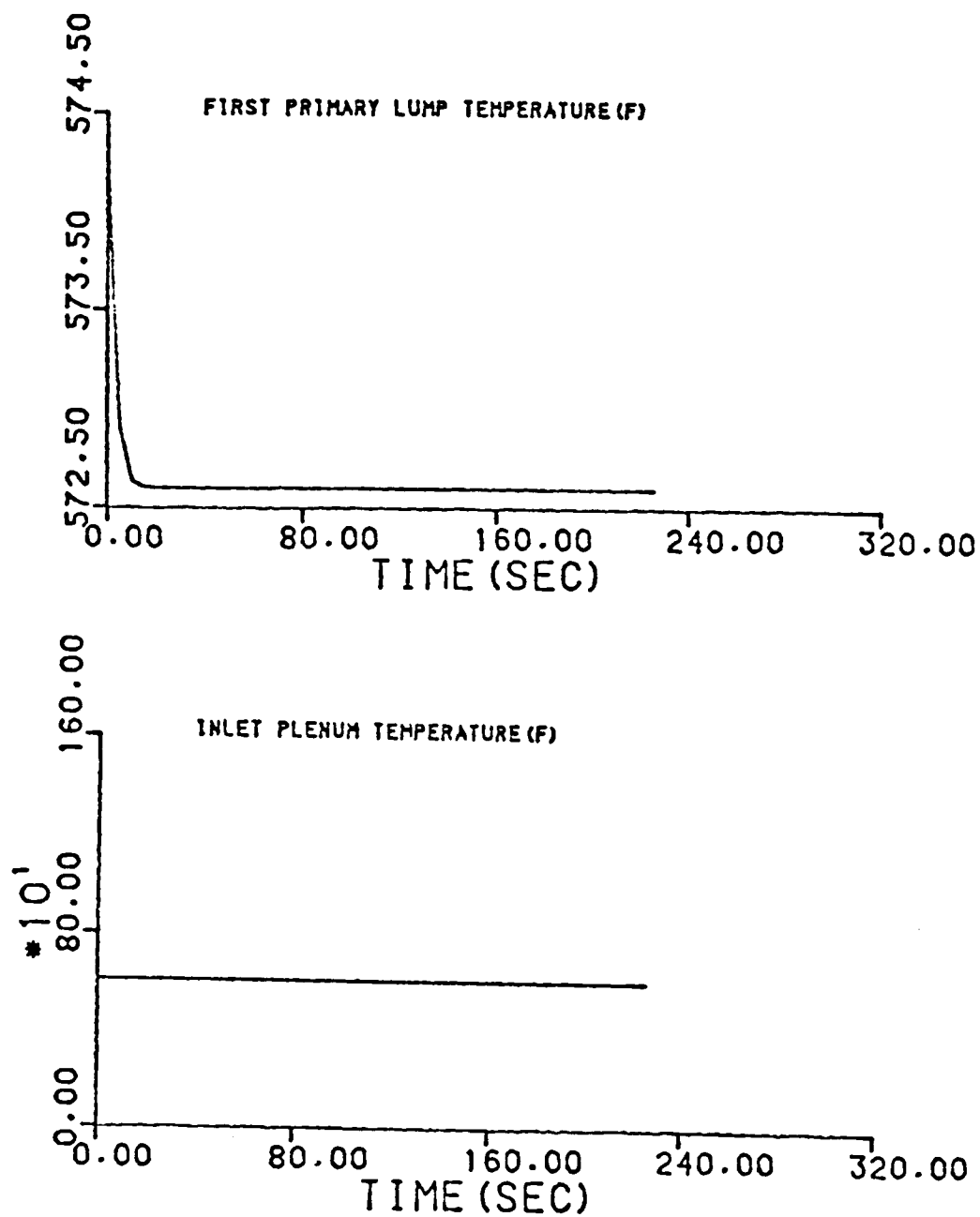


Figure 6.4. (Continued).

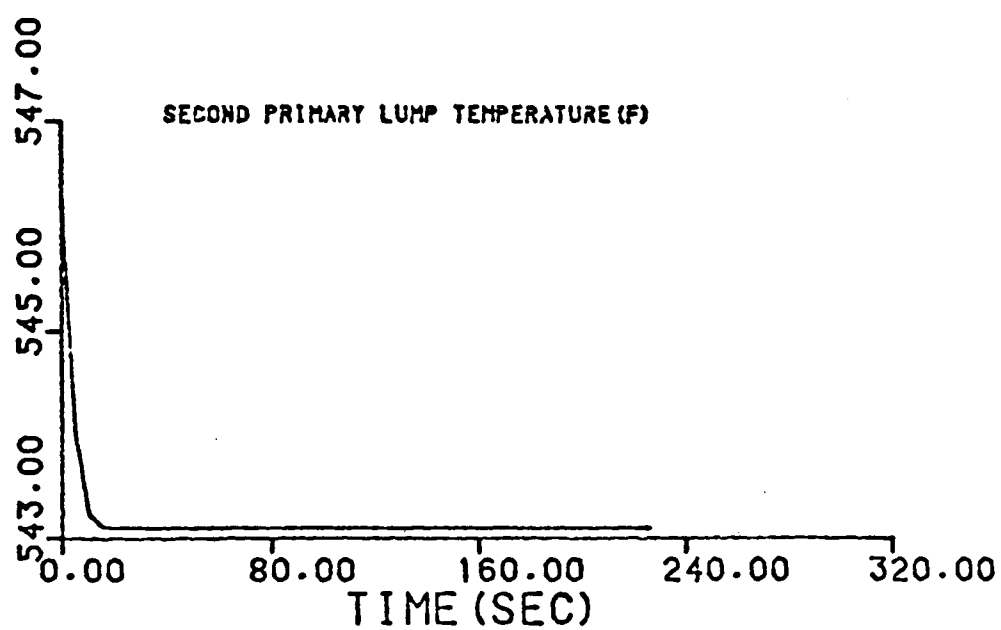
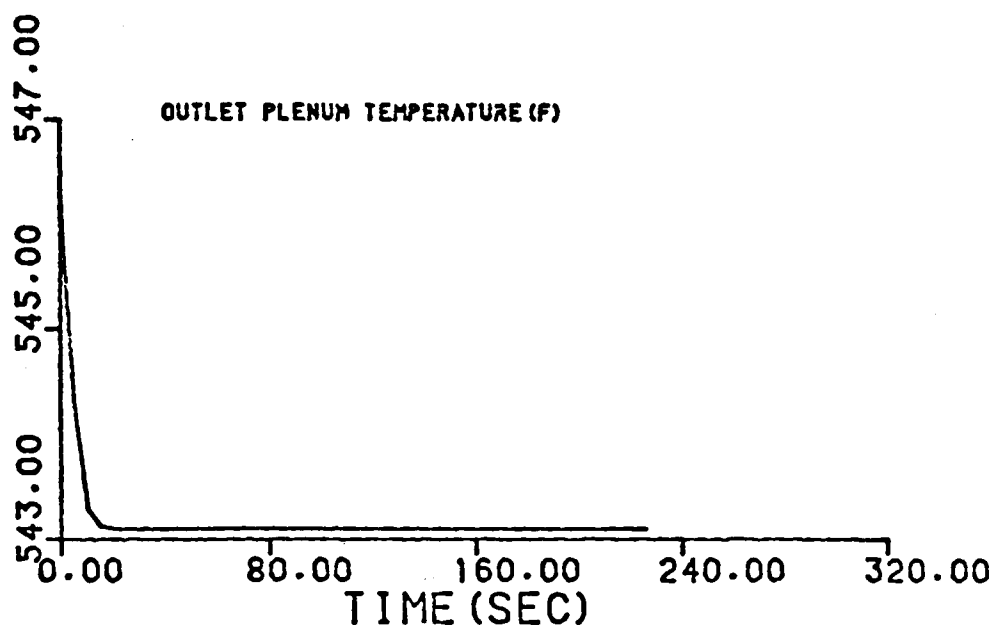


Figure 6.4. (Continued).

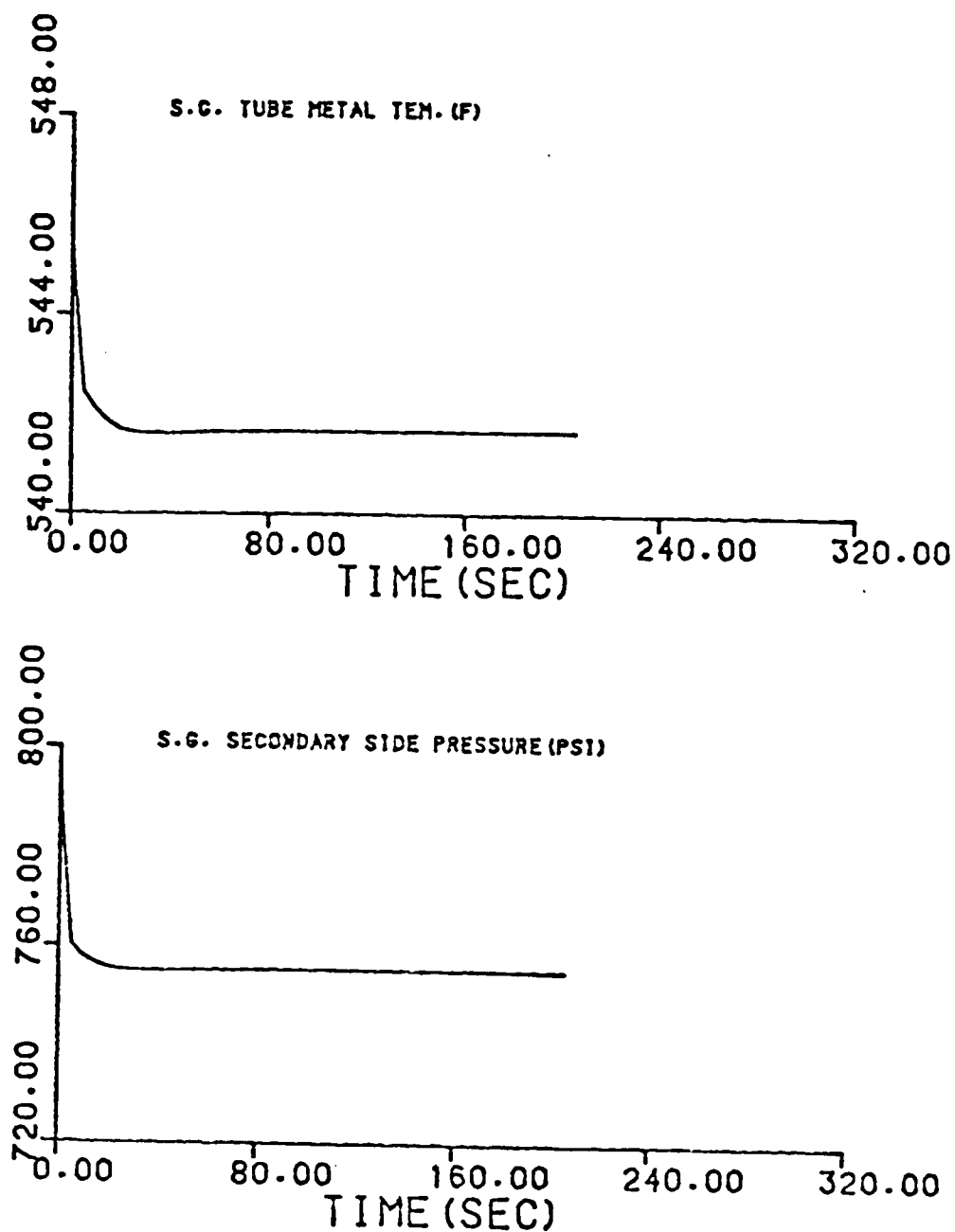


Figure 6.5. Response of the primary loop to a +10% step change in the steam valve coefficient.

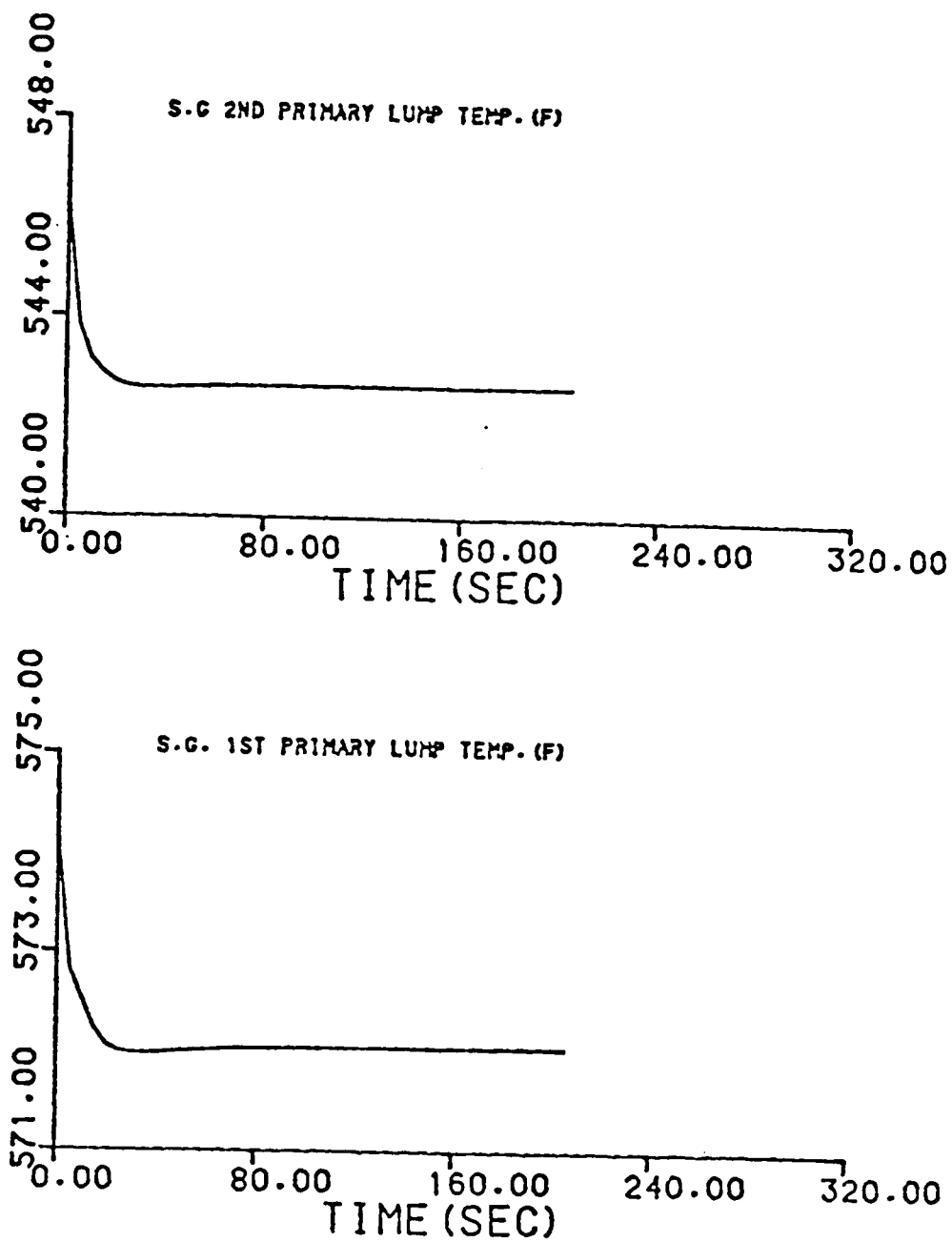


Figure 6.5. (Continued).

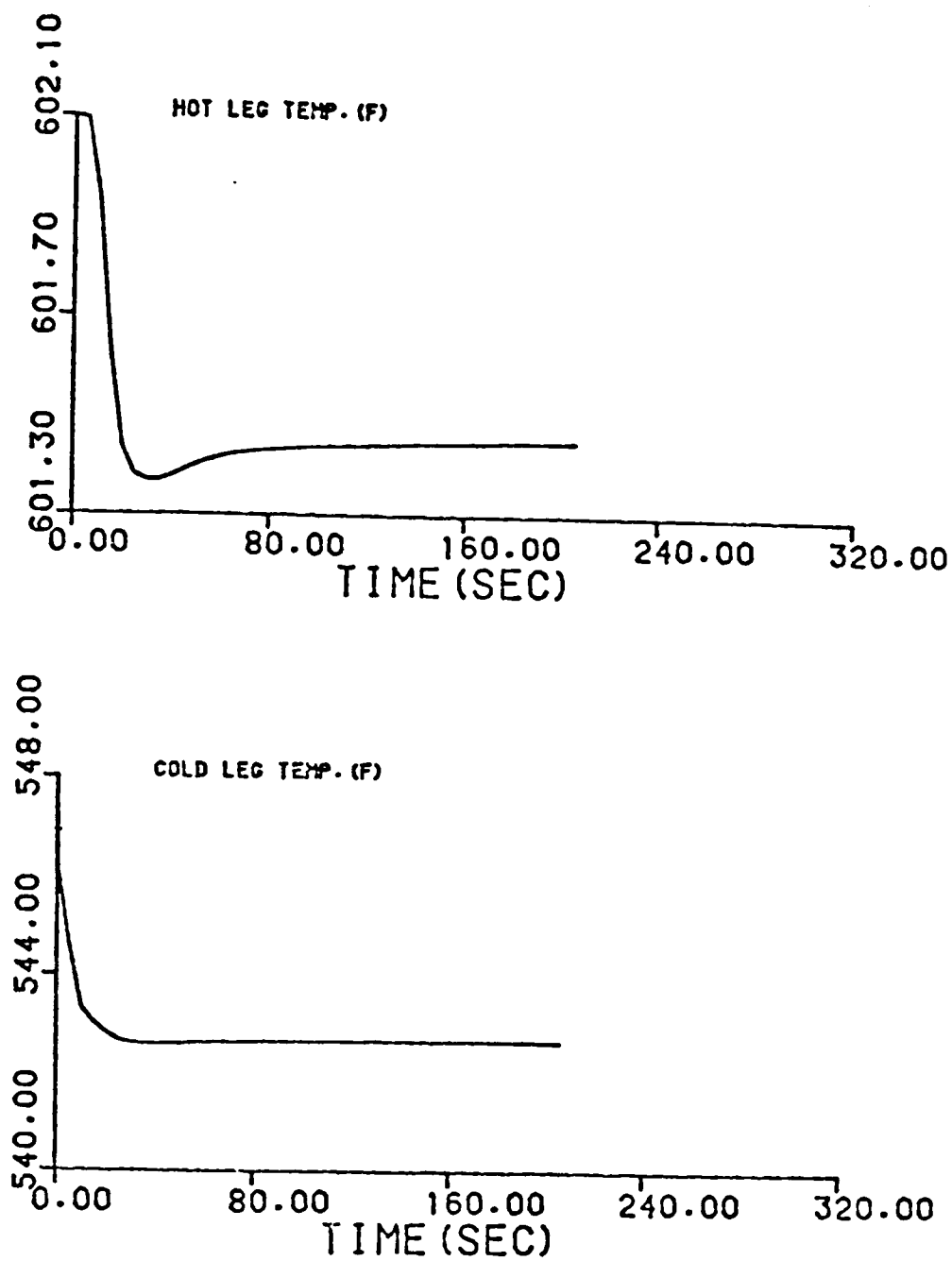


Figure 6.5. (Continued).

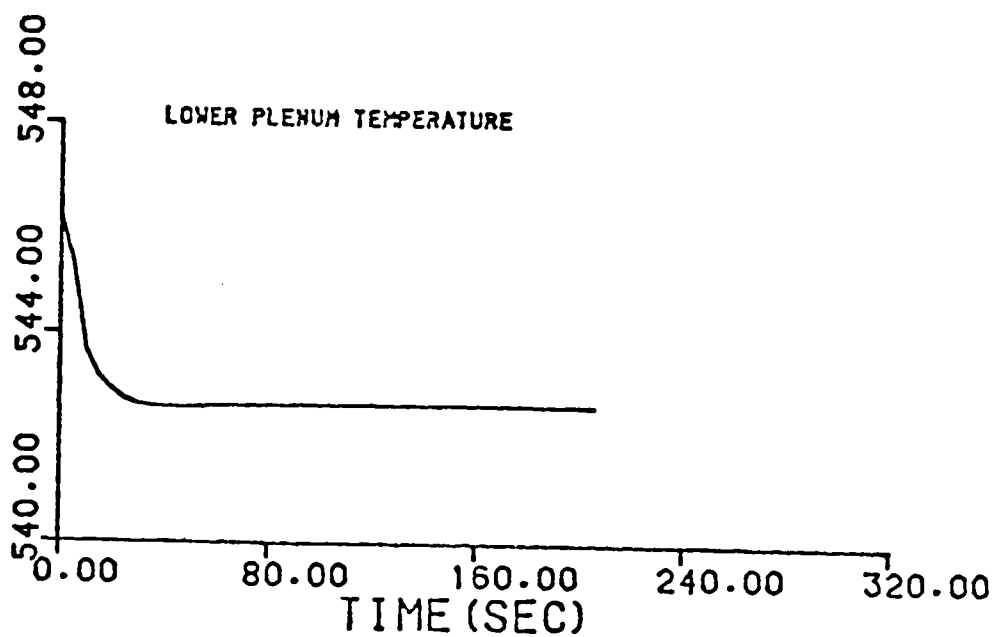
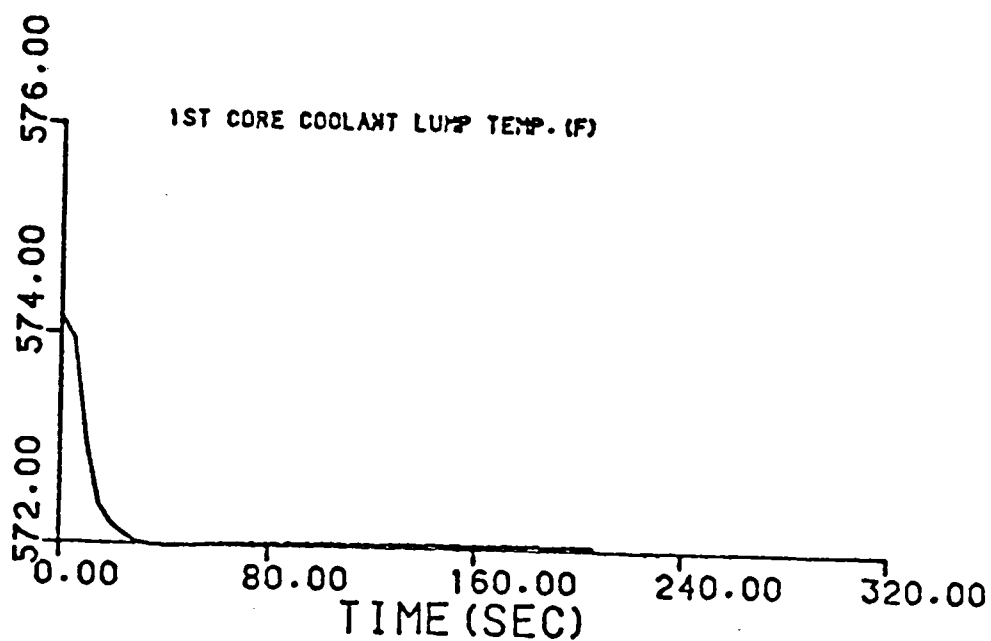


Figure 6.5. (Continued).

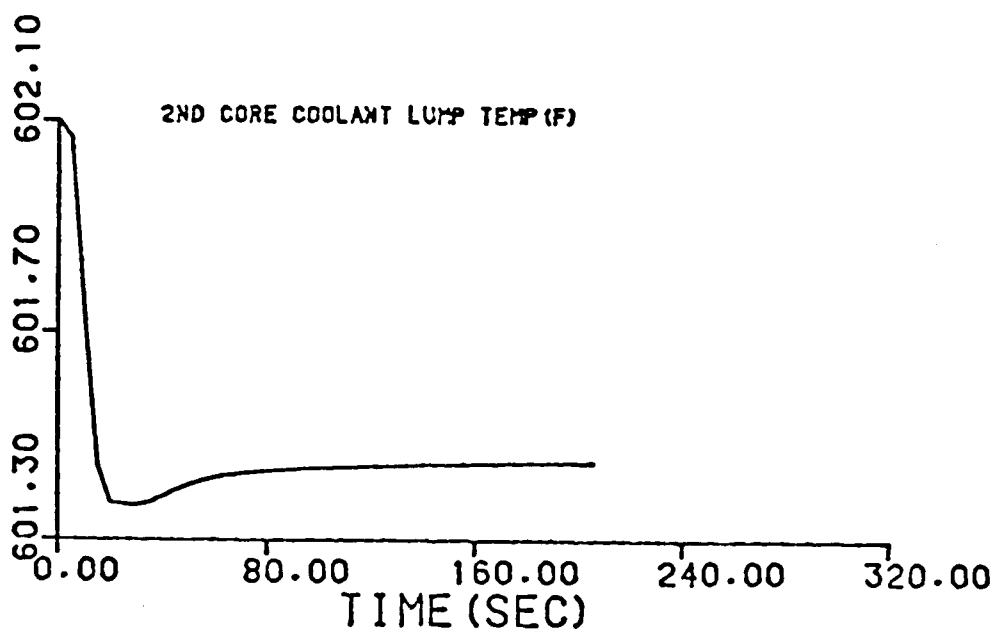
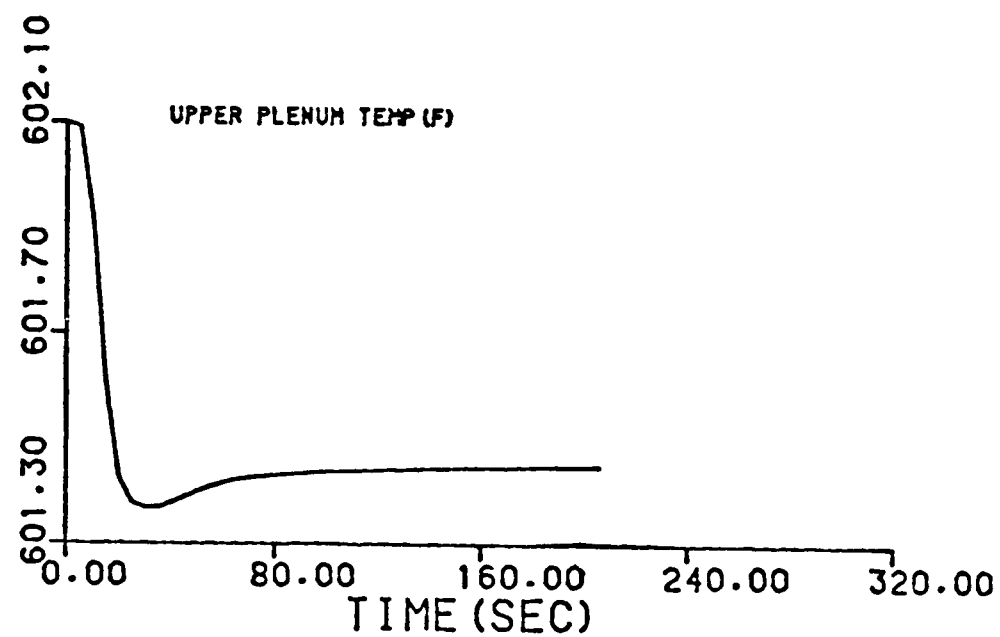


Figure 6.5. (Continued).

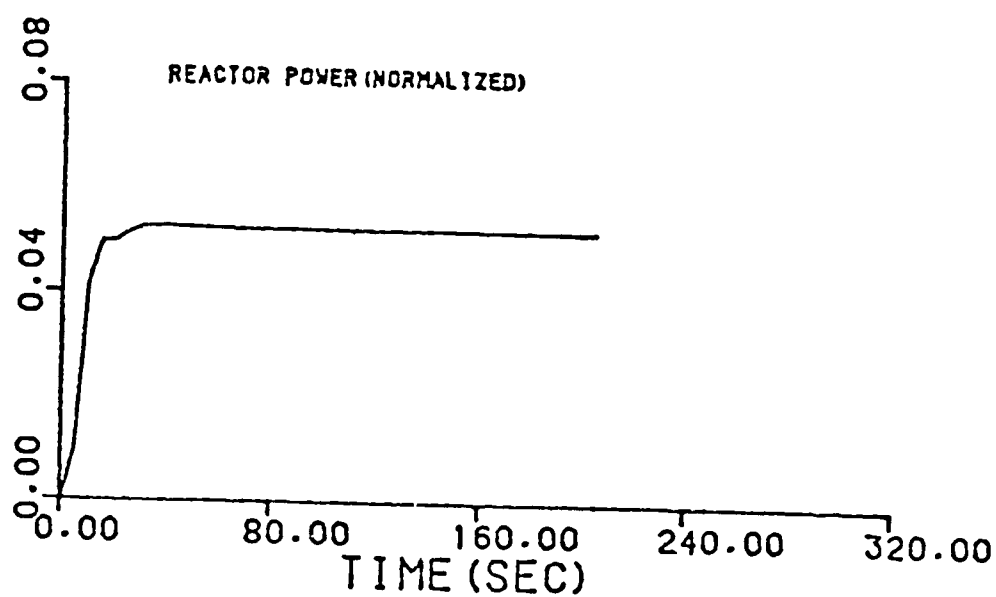
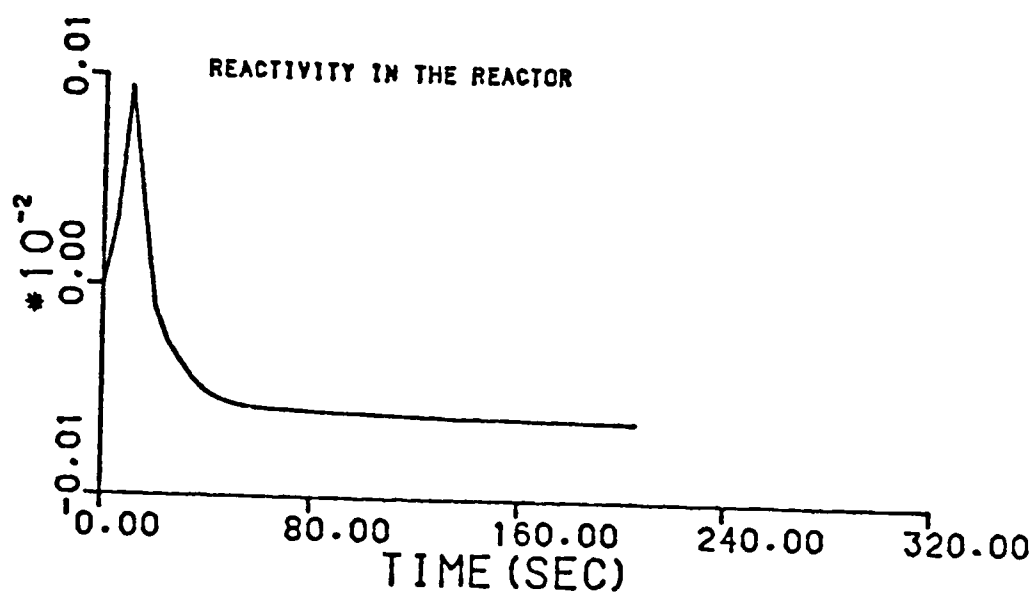


Figure 6.5. (Continued).

compatibility between one file and another. The driver functions to merge files in the library with SMLTOR.FOR and execute the case. It was shown that the MMP package is easy to use and reduces the programming time to minimum.

CHAPTER VII

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

The following conclusions can be drawn from the results of this work:

(1) The Variable Step Size Runge-Kutta-Fehlberg method and the variable step size Adams-Brashforth-Moulton method are adequate to generate a simulation package that is useful for studying the dynamic characteristics of a dynamic system.

(2) The Modular Modeling Program is a simulation package that utilizes the numerical methods mentioned above to provide the modeler with a tool that makes the programming part of his/her work easier and faster.

(3) The Modular Modeling Program is capable of solving a set of coupled mixed linear/nonlinear first order differential equations. This feature eliminates the linearization process that are necessary when the user is limited to a numerical package that can handle linear systems only.

(4) The accuracy of the package was tested by executing several cases that represent theoretical models for the primary loop in a typical pressurized water reactor. The results obtained using the code were in agreement with the results obtained by former graduate students of the University of Tennessee.

(5) The Modular Modeling Program is a self erected program. Advance simulation programs such as the CSMP, ACSL can be used but never seen by the user. In contrast, the Modular Modeling Program may be constructed by interested users. The only requirements are a computer system and a knowledge of the numerical scheme used in this work.

(6) Files in the Modular Modeling Program library have an inherent modular structure. Therefore, theoretical models of various complexity may be called from the library, merged into one file and list the case for a given forcing function.

(7) The Modular Modeling Program is written using a standard FORTRAN language. This feature makes the program adequate for mini and microcomputer applications.

Recommendations

(1) The Modular Modeling Program package at its current status is simply a program that generates the responses of a set of equation subjects to a forcing function. The package may be expanded to include several subroutines that perform numerical tasks. These subroutines may include fitting algorithms so that a parameter such as the thermal conductivity may be represented as a variable instead of a constant.

(2) The steam tables may be added to the package for cases where they may be needed.

(3) The Modular Modeling Program package should be loaded on a mini or microcomputer system where its application can be most useful.

BIBLIOGRAPHY

BIBLIOGRAPHY

1. R.A. Smoak and R.R. Dixon, "Modular Modeling of Boilers," Proceeding of the Third Power Plant Dynamics, Control and Testing Symposium, Knoxville, Tennessee, September 7-9, 1977.
2. R.R. Dixon, "The Modular Modeling System for Power Plant Dynamic Analysis," Proceeding of the Fourth Power Plant Dynamics, Control and Testing Symposium, Knoxville, Tennessee, March 17-19, 1980.
3. P.K. Jain, S. Oh, S.M. Divakaruni, "Simulation of Overcooling Transients with the Modular Modeling System (MMS)," Proceeding of the Fifth Power Plant Dynamics, Control and Testing Symposium, Knoxville, Tennessee, March 21-23, 1983.
4. J.G. Thakkar, "Correlation of Theory and Experiment for the Dynamics of a Pressurized Water Reactor," M.S. Thesis, Nuclear Engineering Department, University of Tennessee (March 1975).
5. E.M. Katz, "Planning, Performing and Interpreting Dynamic Measurements in Pressurized Water Reactors," Ph.D. Dissertation, Nuclear Engineering Department, University of Tennessee (June 1976).
6. M.R.A. Ali, "Lumped Parameter, State Variable Dynamic Models for U-Tube Recirculation Type Nuclear Steam Generators," Ph.D. Dissertation, Nuclear Engineering Department, University of Tennessee, (August 1976).
7. I. Anastasiou, "Model Validation and Control System Analysis for Pressurized Water Reactor Components," Ph.D. Dissertation, Nuclear Engineering Department, University of Tennessee (August 1979).
8. J.D. Freels, "An Investigation of High Order and Low Order Dynamic Modeling of a Complete Pressurized Water Reactor Nuclear Power Plant," M.S. Thesis, Nuclear Engineering Department, University of Tennessee (June 1970).
9. T.W. Kerlin, Frequency Response Testing in Nuclear Reactors, Academic Press, New York (1974).
10. T.W. Kerlin, Nuclear Engineering 5210, 5220, 5230 class notes. Nuclear Engineering Department, University of Tennessee (1982-1983).
11. W. Cheney and D. Kincaid, Numerical Mathematics and Computing, Brooks/Cole Publishing Company, Monterey, California (1980).
12. R.L. Burden, J.D. Faires and A.C. Reynolds, Numerical Analysis, second edition, Prindle, Weber and Schmidt, Boston, Massachusetts (1981).

13. Westinghouse Electric Corporation, PWR System Division, "Precaution Limitations, and Setpoints for Nuclear Steam Supply Systems," Revision 5, June 1976.
14. A.F. Henry, Nuclear-Reactor Analysis, The MIT Press, Cambridge, Massachusetts, and London, England (1975).
15. J.J. Duderstadt and L.J. Hamilton, Nuclear Reactor Analysis, John Wiley and Sons, Inc. (1975).

APPENDIXES

APPENDIX A

REACTOR CORE THEORETICAL MODELS

The theoretical models representing a typical pressurized water reactor core are discussed in this Appendix. Schematic diagrams for the models are shown in Figures 4.1 through 4.3. Each theoretical model consists of a set of ordinary differential equations. The equations represent the reactor kinetics, the core heat transfer and the transport of the coolant in the piping connecting the core to the steam generator and the pressurizer.

The point kinetics equation is the following:

$$\frac{d}{dt} P(t) = \frac{\rho(t) - \beta t}{\Lambda} P(t) + \sum_{i=1}^6 \lambda_i C_i(t) \quad A.1$$

where

$P(t)$ = reactor power

$\rho(t)$ = reactivity in the reactor

βt = total fraction of delayed neutron

= neutron generation time

λ_i = decay constant for delayed neutrons of group i

$C_i(t)$ = group i delayed neutron concentration

Equation A.1 may be written as

$$\frac{d}{dt} (P_o + \delta P(t)) = \frac{(\rho_o + \delta \rho(t) - \beta t)}{\Lambda} (P_o + \delta P(t)) + \sum_{i=1}^6 \lambda_i (C_{oi} + \delta C_i(t)) \quad A.2$$

where

δ denotes deviations from steady state

o denotes steady state conditions.

Equation A.2 may be written as (after collecting steady state terms and equating them to zero):

$$\frac{d}{dt} \delta P(t) = \frac{\delta P(t)}{\Lambda} P_o + \frac{(\delta \rho(t) - \beta t)}{\Lambda} \delta P(t) + \sum_{i=1}^6 \lambda_i \delta C_i(t) \quad A.3$$

or

$$\frac{d}{dt} \delta P(t)/P_o = \frac{\delta \rho(t)}{\Lambda} + \frac{(\delta \rho(t) - \beta t)}{\Lambda} \delta \rho(t)/P_o + \sum_{i=1}^6 \lambda_i \delta C_i(t)/P_o \quad A.4$$

The prompt jump assumption to equation A.4 yields the following:

$$\delta P(t)/P_o = \frac{1}{(\beta_t - \delta \rho(t))} (\delta \rho(t) + \Lambda \sum_{i=1}^6 \lambda_i \delta C_i(t)/P_o) \quad A.5$$

The governing set of equations for the delayed neutrons is the following:

$$\frac{d}{dt} C_i(t)/P_o = \frac{\beta_i}{\Lambda} \delta P/P_o - \lambda_i \delta C_i(t)/P_o \quad i = 1, 2, \dots, 6 \quad A.6$$

The governing set of equations that describes the heat transfer process from the fuel to the coolant is derived by applying an energy balance to each lump in the model. An energy balance on the fuel lump yields the following equation (using Mann's model as discussed in Chapter IV).

$$\frac{d}{dt} E_f = \text{Energy generated in fuel} - \text{Energy leaving fuel} \quad A.7$$

where

$$E_f = \text{energy stored in the fuel.}$$

Equation A.7 may be written as

$$\frac{d}{dt} (M_f C_{pf} T_f) = f P - H A (T_f - T_{c1}) \quad A.8$$

where

- M_f = Mass of the fuel
- C_{pf} = specific heat for the fuel
- T_f = average temperature in the fuel lump
- f = fraction of power generated in the fuel lump
- P = reactor power
- H = overall heat transfer coefficient between the fuel and the coolant
- A = overall heat transfer area between the fuel and the coolant
- T_{c1} = average temperature in the first coolant lump

Equation A.8 may be written as follows:

$$\frac{d}{dt} T_f = \frac{f}{M_f C_f} (1 + \delta P/P_o) P_o - \frac{H A}{M_f C_f} (T_f - T_{c1}) \quad A.9$$

An energy balance on the coolant lump yields the following:

$$\frac{d}{dt} E_{c1} = \begin{array}{l} \text{Energy transferred} \\ \text{from the fuel} \end{array} + \begin{array}{l} \text{Energy in the} \\ \text{incoming coolant} \end{array} - \begin{array}{l} \text{Energy in the} \\ \text{outgoing coolant} \end{array} \quad A.10$$

where

E_{c1} = stored energy in the fuel lump.

Equation A.10 may be written as

$$\frac{d}{dt} (M_{c1} C_{p1} T_{c1}) = \frac{H A}{2} (T_f - T_{c1}) + W C_{p1} (T_{cin} - T_{c1}) \quad A.11$$

where

M_{c1} = mass of the coolant in the coolant lump

C_{p1} = specific heat for the coolant

W = coolant mass flow rate

T_{cin} = average temperature of the coolant entering the lump.

Equation A.11 may be written as follows:

$$\frac{d}{dt} T_{c1} = \frac{H A}{2 M_{c1} C_{p1}} (T_f - T_{c1}) + \frac{W}{M_{c1}} (T_{cin} - T_{c1}) \quad A.12$$

The governing equations for the five lumps core model (see Figure 4.1) are the following:

$$\delta p = \delta p_1 + \delta p_2 - \delta p_3 \quad A.13$$

$$\delta p_3 = ALFAT(T_f - T_{f0}) + ALFAC(T_{c1} - T_{c10}) + ALFAPR(PR - PR_0) \quad A.14$$

$$\frac{d}{dt} T_f(i) = \frac{f}{M_f C_f} (1 + \beta P/P_0) P_0 - \frac{H A}{M_f C_f} (T_f(i) - T_{c(i)}) \quad A.15$$

$$i = 1$$

$$\frac{d}{dt} T_{LP} = \frac{W}{MLP} (T_{CLEG} - T_{LP}) \quad A.16$$

$$\frac{d}{dt} T_{c(i)} = \frac{H A}{2 M C C_{p1}} (T_f(i) + T_{c(i)}) + \frac{W}{MC} (T_{LP} - T_{c(i)}) \quad A.17$$

$$i = 1$$

$$\frac{d}{dt} T_{c(i)} = \frac{H A}{2 M C C_{p1}} (T_f(i) - T_{c(i-1)}) + \frac{W}{MC} (T_{c(i-1)} - T_{c(i)}) \quad A.18$$

$$i = 2$$

$$\frac{d}{dt} TUP = \frac{W}{MUP} (TC(2) - TUP) \quad A.19$$

$$TCLEG = \frac{W}{MCLEG} (TSGOP - TCLEG) \quad A.20$$

$$THLEG = \frac{W}{MHLEG} (TUP - THLEG) \quad A.21$$

Variables for equations A.13 through A.21 are defined in Table A.1

The governing equations for the seventeen lumps core model are the following (see Figure 4.2).

$$\delta p = \delta p_1 + \delta p_2 + \delta p_3 \quad A.22$$

$$\begin{aligned} \delta p_3 = & \text{ALFAF} \left(\sum_{i=1}^5 IF(i) (TF(i) - TFO(i)) \right) \\ & + \text{ALFAC} \left(\sum_{i=1}^{10} IC(i) (TC(i) - TCO(i)) \right) \\ & + \text{ALFAPR} (PR - PRO) \end{aligned} \quad A.23$$

$$IF(i) = \frac{\int_{(i-1)\pi/5}^{i\pi/5} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad i = 1, 2, \dots, 5 \quad A.24$$

$$IC(i) = \frac{\int_{(i-1)\pi/10}^{i\pi/10} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad i = 1, 2, \dots, 10 \quad A.25$$

$$\frac{d}{dt} T_f(i) = \frac{IF(i)}{M_f C_{pf}} (1 - \delta P/P_o) P_o - \frac{H A}{M_f C_f} (T_f(i) - TC(2i - 1)) \quad A.26$$

$$i = 1, 2, \dots, 5$$

$$\frac{d}{dt} TLP = \frac{W}{MLP} (TCLEG - TLP) \quad A.27$$

$$\frac{d}{dt} TC(i) = \frac{H A}{2 MC CP1} (T_f(i) - TC(i)) + \frac{W}{MC} (TLP - TC(i)) \quad A.28$$

$$i = 1$$

$$\frac{d}{dt} TC(i) = \frac{H A}{2 MC CP1} (T_f(\frac{i+1}{2}) - TC(i)) + \frac{W}{MC} (TC(i-1) - TC(i)) \quad A.29$$

$$i = 3, 5, 7, 9$$

$$\frac{d}{dt} (TC(i)) = \frac{H A}{2 MC CP1} (T_f(\frac{i}{2}) - TC(i-1)) + \frac{W}{MC} (TC(i-1) - TC(i)) \quad A.30$$

$$i = 2, 4, 6, 8, 10$$

$$\frac{d}{dt} TUP = \frac{W}{MUP} (TC(10) - TUP) \quad A.31$$

$$\frac{d}{dt} TCLEG = \frac{W}{MCLEG} (TSGOP - TCLEG) \quad A.32$$

$$\frac{d}{dt} THLEG = \frac{W}{MHLEG} (TUP - THLEG) \quad A.33$$

Variables for equations A.22 through A.33 are defined in Table A.1.

The governing equations for the forty five lump core model are the following (see Figure 4.3).

$$\delta\rho = \delta\rho_1 + \delta\rho_2 + \delta\rho_3 \quad \text{A.34}$$

$$\delta\rho_3 = \text{ALFAF} \left(\sum_{i=1}^{15} \text{IF}(i) (\text{TF}(i) - \text{TFo}(i)) \right) \quad \text{A.35}$$

$$+ \text{ALFAC} \left(\sum_{i=1}^{30} \text{IC}(i) (\text{TC}(i) - \text{TCo}(i)) \right) \quad \text{A.35}$$

$$+ \text{ALFAPR} (\text{PR} - \text{PRo})$$

$$\text{IF} = \frac{\int_{(i-1)\pi/5}^{i\pi/5} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{3\pi/6}^{4\pi/6} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad \text{A.36}$$

$$i = 1, 2, \dots, 5$$

$$\text{IF}(i) = \frac{\int_{(i-6)\pi/5}^{(i-5)\pi/5} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{4\pi/6}^{5\pi/6} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad \text{A.37}$$

$$i = 6, 7, \dots, 10$$

$$\text{IF}(i) = \frac{\int_{(i-11)\pi/5}^{(i-10)\pi/5} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{5\pi/6}^{\pi} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad \text{A.38}$$

$$i = 11, 12, \dots, 15$$

$$IC(i) = \frac{\int_{(i-1)\pi/10}^{i\pi/10} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{3\pi/6}^{4\pi/6} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad A.39$$

$$i = 1, 2, \dots, 10$$

$$IC(i) = \frac{\int_{(i-11)\pi/10}^{(i-20)\pi/10} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{4\pi/6}^{5\pi/6} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad A.40$$

$$i = 11, 12, 13, \dots, 20$$

$$IC(i) = \frac{\int_{(i-21)\pi/10}^{(i-30)\pi/10} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} * \frac{\int_{5/6}^{\pi} \sin x \, dx}{\int_0^{\pi} \sin x \, dx} \quad A.41$$

$$l = 21, 22, 23, \dots, 30$$

$$\frac{d}{dt} T_f(i) = \frac{IF(i)}{M_f C_f} (1 - \delta P/P_o) - \frac{H A}{M_f C_f} (T_f(i) - TC(2i-1)) \quad A.42$$

$$i = 1, 2, 3, \dots, 15$$

$$\frac{d}{dt} TLP = \frac{W}{MLP} (TCLEG) - TLP$$

$$\frac{d}{dt} (TC(i)) = \frac{H A}{2 MC CP1} (T_f(i) - TC(i)) + \frac{W}{MC} (TLP - TC(i)) \quad A.43$$

$$i = 1$$

$$\frac{d}{dt} TC(i) = \frac{H A}{2 MC CP1} (T_f(\frac{i+2}{2}) - TC(i)) + \frac{W}{MC} (TC(i-1) - TC(i)) \quad A.44$$

$$i = 3, 5, 7, \dots, 29$$

$$\frac{d}{dt} TC(i) = \frac{H A}{2 MC CP1} (T_f(\frac{i}{2}) - TC(i-1)) + \frac{W}{MC} (TC(i-1) - TC(i)) \quad A.45$$

$$i = 2, 4, 6, \dots, 30$$

$$\frac{d}{dt} TUP = \frac{W}{MUP} ((TC(10) + TC(20) + TC(30))/3 - TUP) \quad A.46$$

$$\frac{d}{dt} TCLEG = \frac{W}{MCLEG} (TSGOP - TCLEG) \quad A.47$$

$$\frac{d}{dt} THLEG = \frac{W}{MHLEG} (TUP - THLEG) \quad A.48$$

Variables for equations A.34 through A.48 are defined in Table A.1.

Responses of the five lump core model, the seventeen lump core model and the forty-seven lumps core model to a +10°F step change in the coolant inlet temperature are shown in Figures A.1 through A.3, respectively.

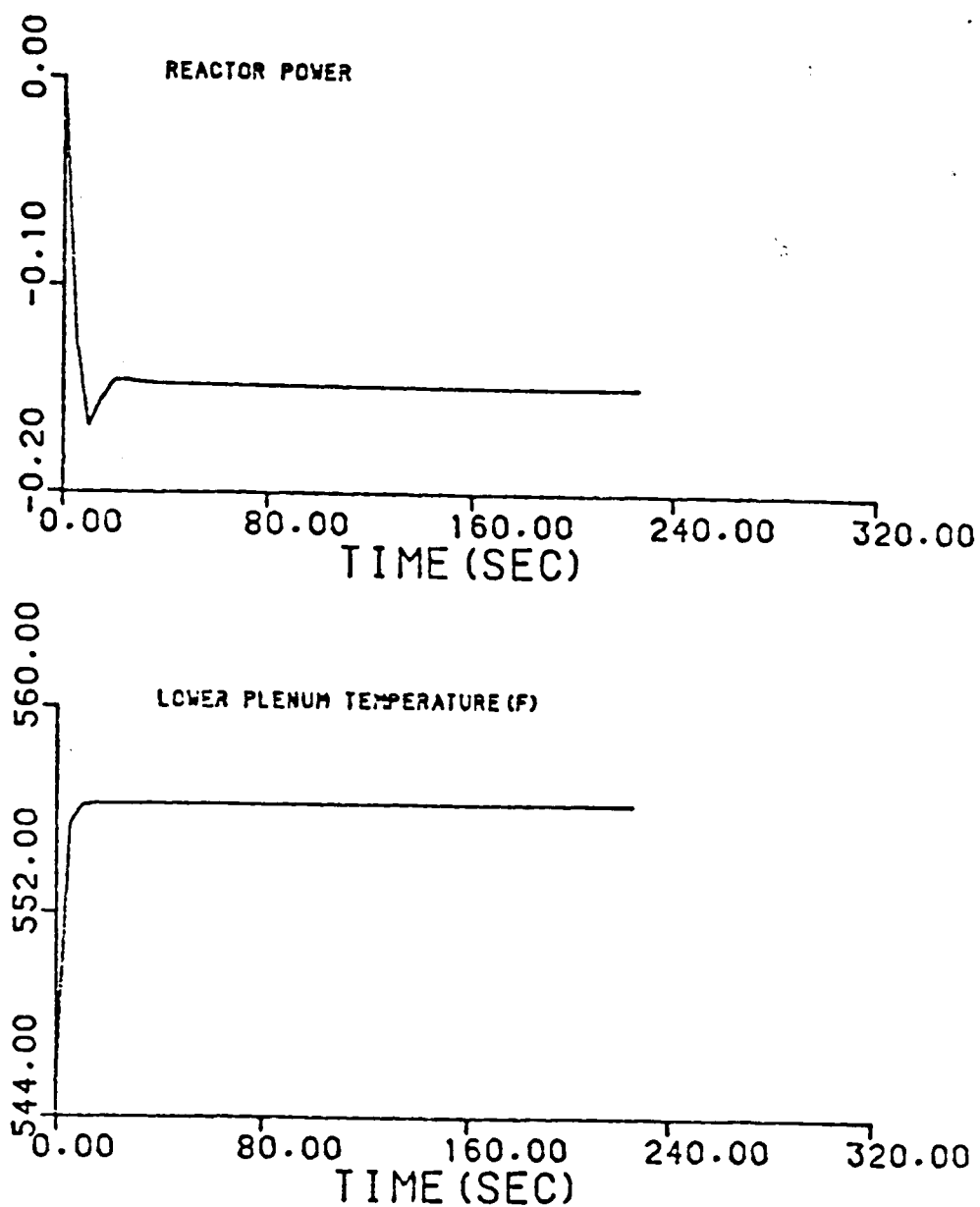


Figure A.1. Response of the five lump core model to a +10°F step change in the inlet coolant temperature.

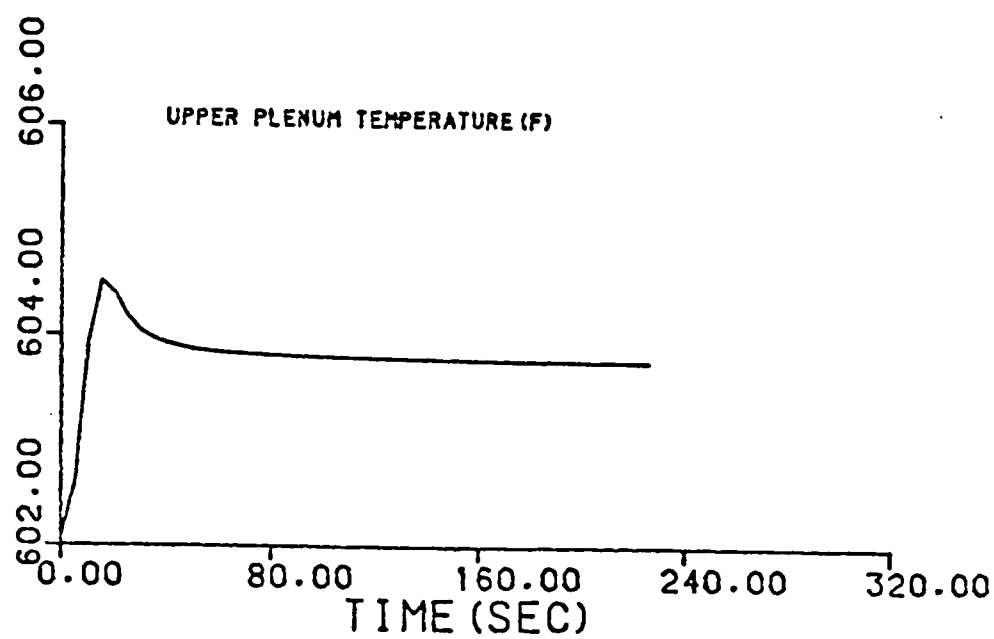


Figure A.1. (Continued).

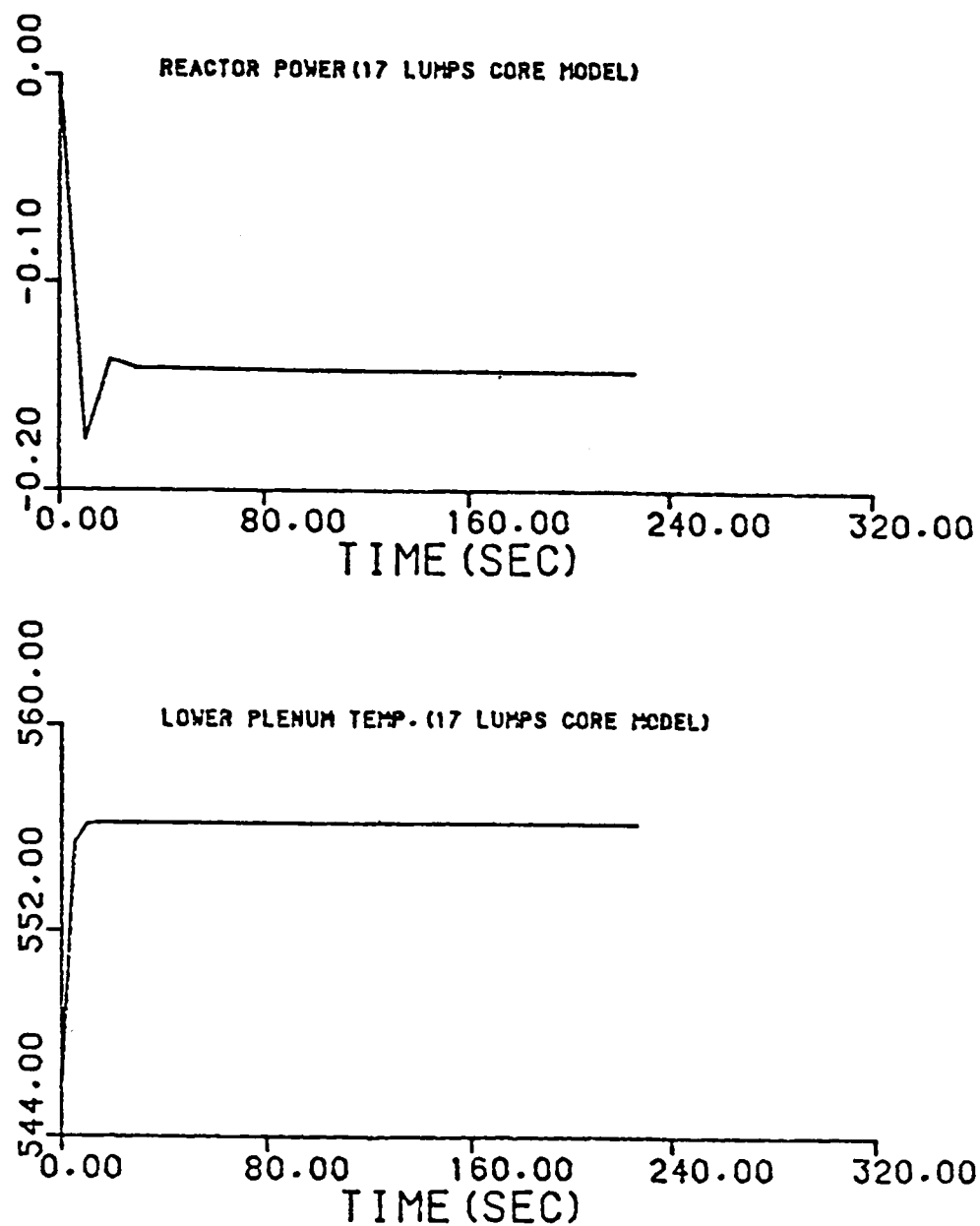


Figure A.2. Response of the seventeen lump core model to a +10°F step change in the inlet coolant temperature.

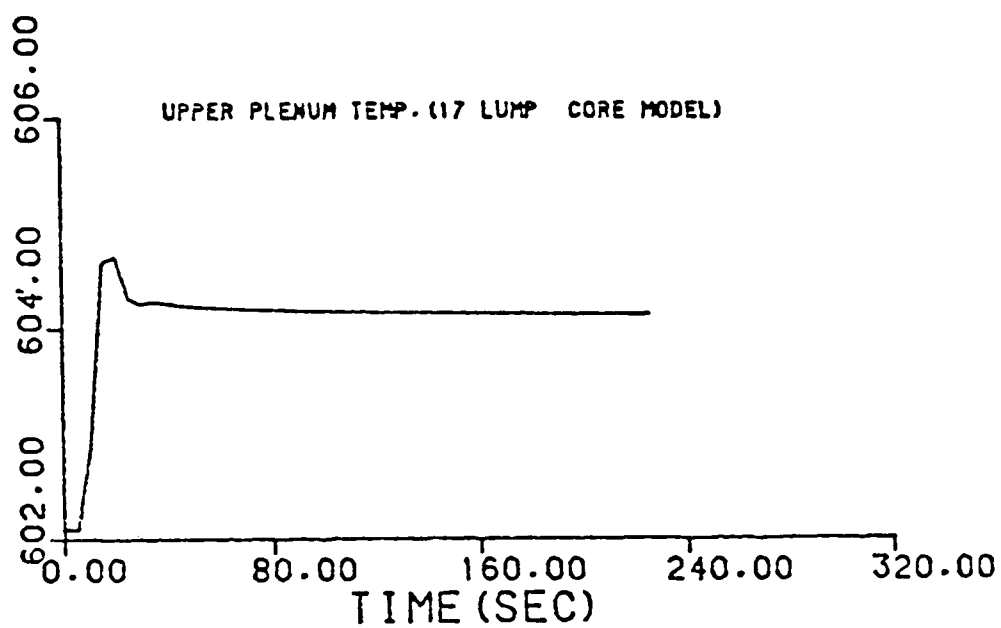


Figure A.2. (Continued).

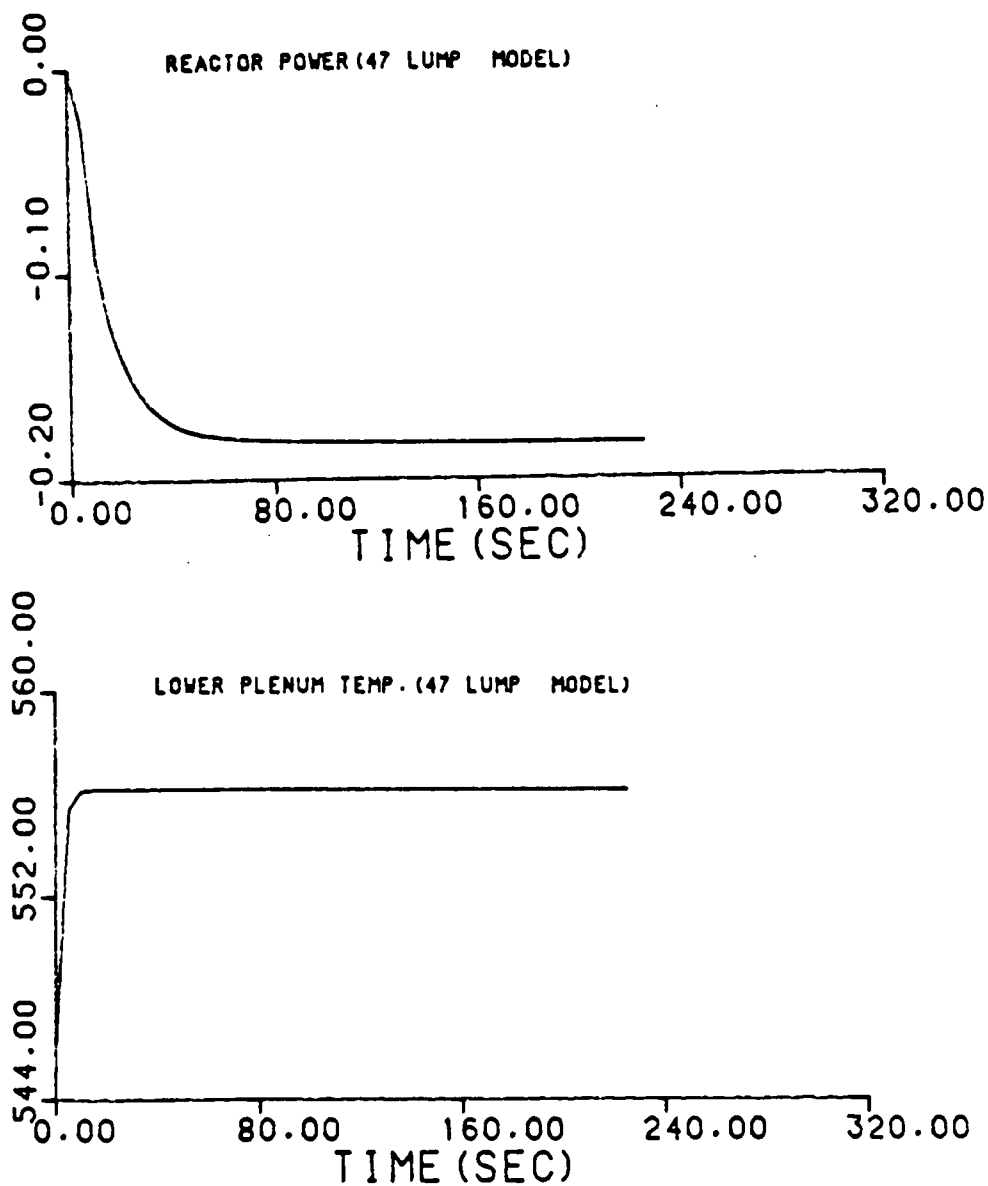


Figure A.3. Response of the forty seven lump core model to a +10°F step change in the inlet coolant temperature.

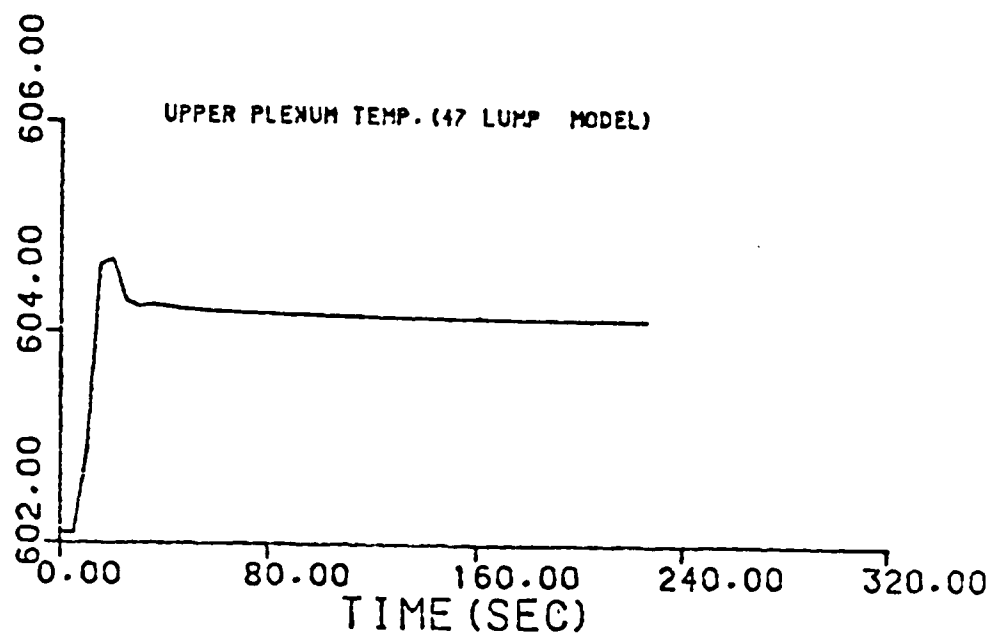


Figure A.3. (Continued).

TABLE A.1

LIST OF VARIABLES USED IN THE CORE THEORETICAL MODELS

$\delta\rho_1$	Reactivity due to the temperature and pressure feedback effects
$\delta\rho_2$	Reactivity due to the manual control rod (forcing function)
$\delta\rho_3$	Reactivity due to the automatic control rod
ALFAF	Fuel coefficient of reactivity
ALFAC	Coolant coefficient of reactivity
ALFAPR	Pressure coefficient of reactivity
IF(i)	Importance of the reactivity introduced in the reactor due to temperature changes in fuel lump i
IC(i)	Importance of the reactivity introduced in the reactor due to temperature changes in coolant lump i
TF(i)	Average fuel temperature in fuel lump i
TFo(i)	Average steady state temperature in fuel lump i
TC(i)	Average coolant temperature in coolant lump i
TCo(i)	Average steady state temperature in coolant lump i
PR	Average primary side pressure
PRo	Average steady state primary side pressure
M_f	Mass of the fuel per fuel lump
C_{pf}	Specific heat for the fuel
H	Overall heat transfer coefficient between the fuel and the coolant
A	Overall heat transfer area between the fuel and the coolant (per fuel lump)
W	Primary coolant mass flow rate

TABLE A.1 (continued)

MC	Mass of the coolant (per coolant lump)
CP1	Specific heat for the coolant
TUP	Average coolant temperature in the core upper plenum
TLP	Average coolant temperature in the core lower plenum
MUP	Mass of the coolant in the core upper plenum
MLP	Mass of the coolant in the core lower plenum
TCLEG	Average coolant temperature in the hot leg piping
MCLEG	Mass of the coolant in the hot leg piping
THLEG	Average coolant temperature in the cold leg piping
MHLEG	Mass of the coolant in the cold leg piping
TSGOP	Average coolant temperature in the steam generator outlet plenum

APPENDIX B

PRESSURIZER THEORETICAL MODEL

The theoretical model representing a typical pressurized water reactor pressurizer is discussed in this Appendix. The governing equations for the model are derived by applying the energy and mass balance equations to the liquid and steam phases in the pressurizer.

Mass Balance:

$$\text{Liquid Phase} \quad \frac{d}{dt} MW = WW1 + WSP - WS \quad B.1$$

$$\text{Steam Phase} \quad \frac{d}{dt} MS = WS \quad B.2$$

where

MW = Mass of the liquid in the pressurizer

MS = Mass of the steam in the pressurizer

WW1 = Pressurizer surge line flow rate

WSP = Pressurizer sprayer flow rate

WS = Evaporation/condensation rate.

Substitute

$$MW = \rho_{H_2O} A L_W \quad B.3$$

$$M_s = \rho_{H_2O} A (L - L_W) \quad B.4$$

into equation B.1 and B.2 to obtain

$$\frac{d}{dt} L_W = \frac{1}{\rho_{H_2O} A} (WW1 + WSP - WS - A L_W \frac{d}{d PR} \rho_{H_2O} \frac{d}{dt} PR) \quad B.5$$

$$\frac{d}{dt} L_W = \frac{1}{\rho_{H_2O} A} (A(L - L_W) \frac{d}{d PR} \rho_{H_2O} \frac{d}{dt} PR - WS) \quad B.6$$

where

ρ_{H_2O} = Density of the water at saturated condition

- $RHOS$ = Density of the steam at saturated condition
 A = Cross sectional area for the pressurizer
 LW = Height of the liquid water in the pressurizer
 L = Total height of the pressurizer
 PR = Pressurizer pressure

Equations B.5 and B.6 may be solved for WS to obtain

$$WS = \frac{1}{\frac{RHOW}{RHOS} - 1} (A(L - LW) \frac{RHOW}{RHOS} \frac{d RHOS}{d PR} + A LW \frac{d RHOS}{d PR} \frac{d PR}{dt} - WW1 - WSP) \quad B.7$$

or

$$WS = \frac{1}{a1} (a2 \frac{d PR}{dt} - WW1 - WSP) \quad B.8$$

where

$$a1 = \left(\frac{RHOW}{RHOS} - 1 \right) \quad B.9$$

$$a2 = (A (L - LW) \frac{RHOW}{RHOS} \frac{d RHOS}{d PR} + A LW \frac{d RHOW}{d PR}) \quad B.10$$

Substitute equation B.8 into equation B.6 to obtain

$$\frac{d LW}{dt} = \frac{1}{RHOS A} \left((A(L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1} \frac{d PR}{dt} + \frac{WW1}{a1} + \frac{WSP}{a1}) \right) \quad B.11$$

Energy Balance:

$$\text{Liquid Phase: } \frac{d EW}{dt} = Q + WW1 HW1 + WSP HSP - WS HS - \frac{PR}{J} \frac{d VW}{dt} \quad B.12$$

where

- EW = Energy stored in the water

- Q = Pressurizer heater output
 $HW1$ = Specific enthalpy of the water flowing through the surge line
 HSP = Specific enthalpy of the sprayer water
 HS = Specific steam enthalpy
 VW = Volume of the pressurizer water
 J = Joule's constant

Substitute

$$EW = MW UW \quad B.13$$

$$VW = MW \tilde{VW} \quad B.14$$

where

UW = Specific internal energy of the pressurizer water

$\tilde{VW}$ = Specific volume of the pressurizer water

into equation B.12 to obtain

$$\begin{aligned}
 MW \frac{d}{dt} UW + UW (WW1 + WSP - WS) &= Q + WW1 HW1 + WSP HSP - WS HS \\
 &- \frac{PR}{J} MW \frac{d}{dt} \tilde{VW} \\
 &- \frac{PR}{J} \tilde{VW} (WW1 + WSP - WS) \quad B.15
 \end{aligned}$$

Equation B.15 may be reduced to

$$\begin{aligned}
 (MW \frac{d}{dt} UW + \frac{PR}{J} MW \frac{d}{dt} \tilde{VW}) \frac{d PR}{dt} &= Q + (HW1 - UW - \frac{PR}{J} \tilde{VW}) WW1 \\
 &+ (HSP - UW - \frac{PR}{J} \tilde{VW}) WSP
 \end{aligned}$$

$$- (HS + UW + \frac{PR}{J} \tilde{VW}) WS \quad B.16$$

Substitute equation B.8 into equation B.16 to obtain

$$\begin{aligned} & (MW \frac{d UW}{d PR} + \frac{PR}{J} MW \frac{d \tilde{VW}}{d PR} + (HS + UW + \frac{PR}{J} \tilde{VW}) \frac{a2}{a1}) \frac{d PR}{dt} \\ = & Q + (HW1 - UW - \frac{PR}{J} \tilde{VW} + \frac{1}{a1} (HS + UW + \frac{PR}{J} \tilde{VW})) WW1 \\ & + (HSP - UW - \frac{PR}{J} \tilde{VW} + \frac{1}{a1} (HS + UW + \frac{PR}{J} \tilde{VW})) WSP \end{aligned} \quad B.17$$

or

$$\frac{d PR}{dt} = \frac{1}{b1} (Q + b2 WW1 + b3 WSP) \quad B.18$$

where

$$b1 = (MW \frac{d U}{d PR} + \frac{PR}{J} MW \frac{d \tilde{VW}}{d PR} + (HS + UW + \frac{PR}{J} \tilde{VW}) \frac{a2}{a1}) \quad B.19$$

$$b2 = (HW1 - UW - \frac{PR}{J} \tilde{VW} + \frac{1}{a1} (HS + UW + \frac{PR}{J} \tilde{VW})) \quad B.20$$

$$b3 = (HSP - UW - \frac{PR}{J} \tilde{VW} + \frac{1}{a1} (HS + UW + \frac{PR}{J} \tilde{VW})) \quad B.21$$

Equation B.18 may be substituted into equation B.11 to obtain

$$\frac{d LW}{dt} = \frac{1}{RHOS A b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) Q$$

$$\begin{aligned}
& + \left(\frac{b2}{RHOS \ A \ b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) + \frac{1}{RHOS \ A \ a1} \right) WWI \\
& + \left(\frac{b3}{RHOS \ A \ b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) + \frac{1}{RHOS \ A \ a1} \right) WSP
\end{aligned}
\tag{B.22}$$

or

$$\frac{d}{dt} LW = c1 \ Q + c2 \ WWI + c3 \ WSP \tag{B.23}$$

where

$$c1 = \frac{1}{RHOS \ A \ b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) \tag{B.24}$$

$$c2 = \left(\frac{b2}{RHOS \ A \ b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) + \frac{1}{RHOS \ A \ a1} \right) \tag{B.25}$$

$$c3 = \left(\frac{b3}{RHOS \ A \ b1} (A (L - LW) \frac{d RHOS}{d PR} - \frac{a2}{a1}) - \frac{1}{RHOS \ A \ a1} \right) \tag{B.26}$$

Therefore, the governing equations for the pressurizer model are

$$\frac{d}{dt} PR = \frac{1}{b1} (Q + b2 \ WWI + b3 \ WSP) \tag{B.27}$$

$$\frac{d}{dt} LW = c1 \ Q + c2 \ WS1 + C3 \ WSP \tag{B.28}$$

The surge line flow rate, WWI, may be expressed in terms of the changes in the primary water volume. In this case,

$$WWI = \frac{d}{dt} MPRIME \tag{B.29}$$

where

MPRIME = Mass of the primary water in the primary loop.

If we assume that changes in the hot leg and cold leg primary water masses are adequate to represent equation B.29, then equation B.29 can be written as follows:

$$WW1 = \frac{d}{dt} MHLEG + \frac{d}{dt} MCLEG \quad B.30$$

where

MHLEG = Mass of the primary water in the hot leg piping

MCLEG = Mass of the primary water in the cold leg piping.

Substitute

$$MHLEG = RHOP \quad VH \quad B.31$$

$$MCLEG = RHOP \quad VC \quad B.32$$

where

RHOP = average density of the primary coolant

VH = volume of the coolant in the hot leg

VC = volume of the coolant in the cold leg

into equation B.30 to obtain

$$WW1 = VH \frac{d RHOP}{d THLEG} \frac{d}{dt} THLEG + VC \frac{d RHOP}{d TCLEG} \frac{d}{dt} TCLEG \quad B.33$$

or

$$WW1 = VH \quad BETA \left(\frac{d}{dt} THLEG + \frac{d}{dt} TCLEG \right) \quad B.34$$

where

$$BETA = \frac{d RHOP}{d THLEG} = \frac{d RHOP}{d TCLEG} \quad B.35$$

THLEG = Average temperature in the hot leg

TCLEG = Average temperature in the cold leg

The pressurizer heater output and the pressurizer sprayer flow rate are controlled by a proportional plus integral plus derivative (PID) controllers. The block diagrams for the controllers are shown in Figure 4.9 and 4.10. The governing set of equations for the heater controller and sprayer controller can be obtained by transforming from the Laplace domain to the time domain. Consider the following transfer function for the heater controller.

$$\frac{\delta Q}{\delta PR} = \text{GAIN1} \left(1 + \frac{1}{\text{TPR1} S} + \text{TPR2} S \right) \quad \text{B.36}$$

and the sprayer controller

$$\frac{\delta WSP}{\delta PR} = \text{GAIN2} \left(1 + \frac{1}{\text{TPR3} S} + \text{TPR4} S \right) \quad \text{B.37}$$

where

δQ	=	heater output expressed as deviation from steady state
δWSP	=	sprayer output expressed as deviation from steady state
δPR	=	primary side pressure expressed as a deviation from steady state
GAIN1	=	heater controller gain
TPR1	=	time constant for the integral control (heater)
TPR2	=	time constant for the derivative control (heater)
GAIN2	=	spray controller gain
TPR3	=	time constant for the integral control (sprayer)
TPR4	=	time constant for the derivative control (sprayer)
S	=	denotes laplace domain

let us cross multiply equations B.36 and B.37 and take the inverse laplace transform of the resulting equations to obtain

$$\delta Q = \text{GAIN2} \left(\delta \text{PR} + \frac{1}{\text{TPR1}} \delta \text{PR} \, dt + \text{TPR2} \frac{d}{dt} \delta \text{PR} \right) \quad \text{B.38}$$

$$\delta \text{WSP} = \text{GAIN2} \left(\delta \text{PR} + \frac{1}{\text{TPR3}} \delta \text{PR} \, dt + \text{TPR4} \frac{d}{dt} \delta \text{PR} \right) \quad \text{B.39}$$

let

$$\frac{d}{dt} \delta \text{D1} = \delta \text{PR}$$

where

$$\delta \text{D1} = \text{a dummy variable}$$

then equations B.38 and B.39 can be written as

$$\delta Q = \text{GAIN1} \left(\delta \text{PR} + \frac{1}{\text{TPR1}} \delta \text{D1} + \text{TPR2} \frac{d}{dt} \delta \text{PR} \right) \quad \text{B.40}$$

$$\delta \text{WSP} = \text{GAIN2} \left(\delta \text{PR} + \frac{1}{\text{TPR3}} \delta \text{D1} + \text{TPR4} \frac{d}{dt} \delta \text{PR} \right) \quad \text{B.41}$$

$$\frac{d}{dt} \delta \text{D1} = \delta \text{PR}$$

Response of the pressurizer model to a + 1.0 IBM/SEC step change in the surge line flow rate is shown in figure B.1.

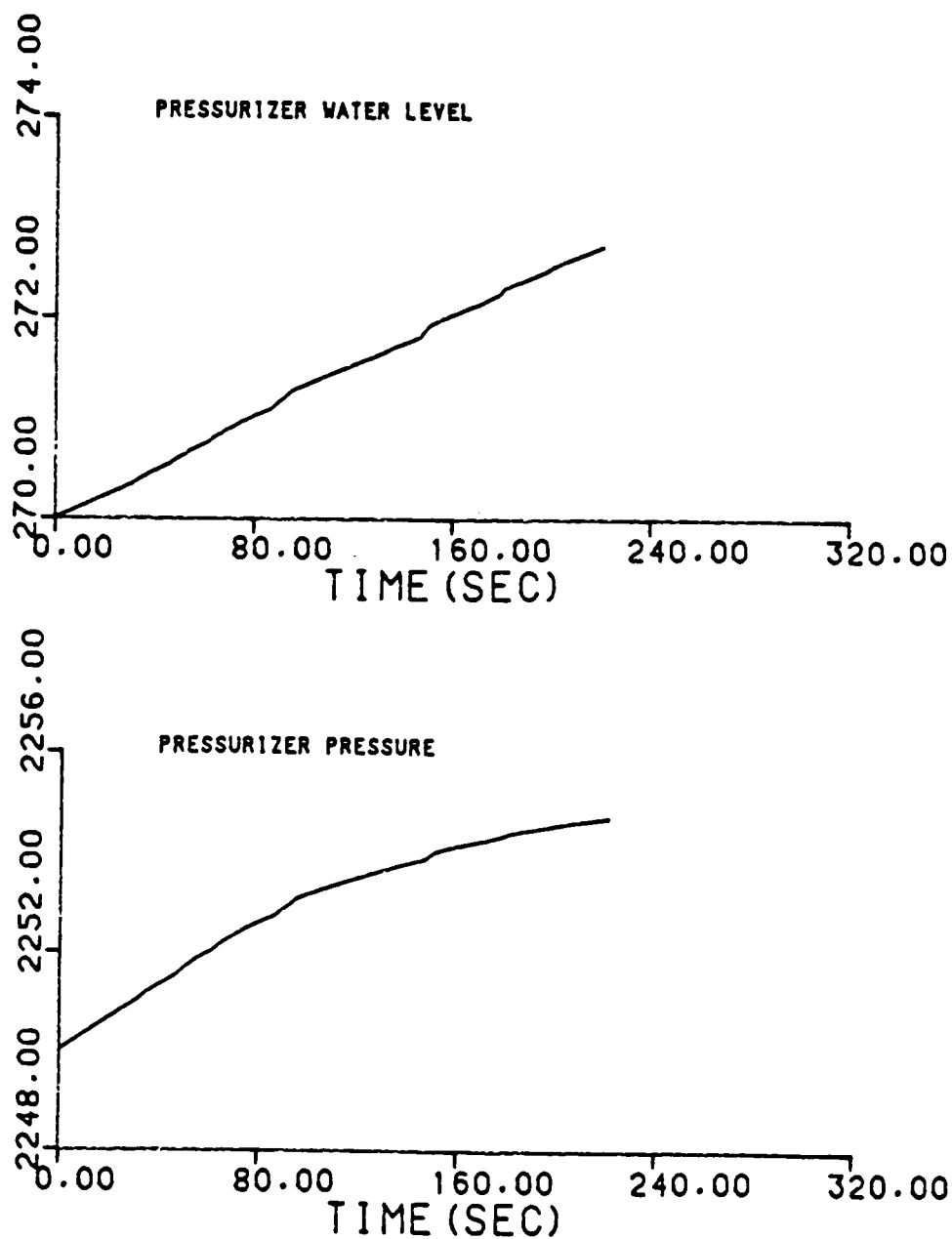


Figure B.1. Response of the pressurizer theoretical model to a +1.0 lbm/sec step change in the surge line flow rate.

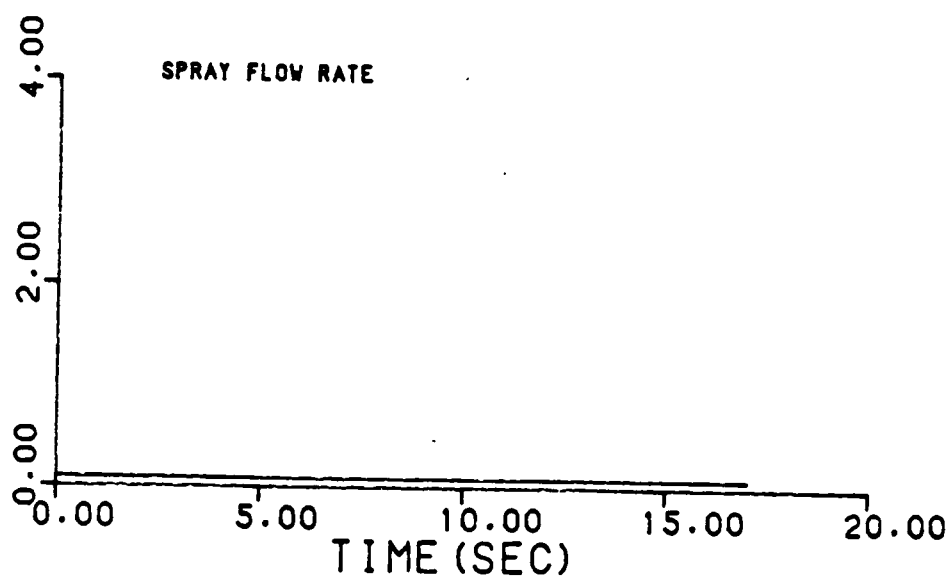
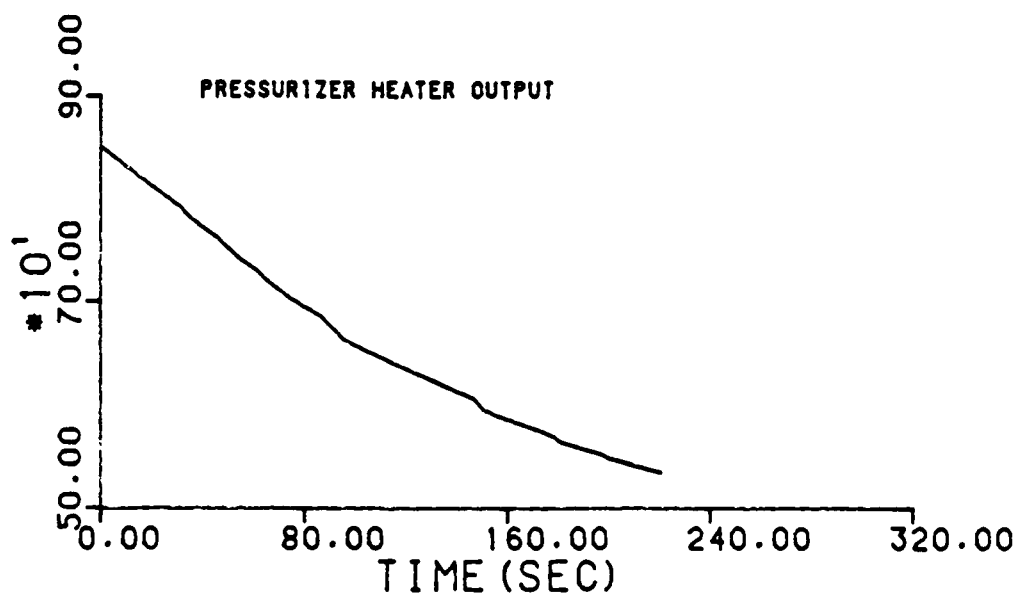


Figure B.1. (Continued).

APPENDIX C

STEAM GENERATOR THEORETICAL MODELS

Theoretical models for the U-tube steam generator are derived in this Appendix. The theoretical models that are presented in the Appendix are derived in detail by Ali in 1976. The models are rederived in this Appendix to provide the reader with an easy access to the sets of equations that represent the U-tube steam generator theoretical models.

A schematic diagram for the simple steam generator model is shown in Figure 4.7. The governing equations for the model are derived below (see Table C.1 for definition of variables).

Inlet Plenum:

$$\frac{d}{dt} (\text{MSG1P CP1 TSG1P}) = \text{WSG CP1 (THLEG - TSG1P)} \quad \text{C.1}$$

or

$$\frac{d}{dt} \text{TSG1P} = \frac{\text{WSG}}{\text{MSG1P}} (\text{THLEG - TSG1P}) \quad \text{C.2}$$

1st Primary Lump:

$$\frac{d}{dt} (\text{MP CP1 TP1}) = \text{WSG CP1 (TSG1P - TP1) - UPM SPM (TP1 - TMETAL)} \quad \text{C.3}$$

or

$$\frac{d}{dt} \text{TP1} = \frac{\text{WSG}}{\text{MP}} (\text{TSG1P - TP1}) - \frac{\text{UPM SPM}}{\text{MP CP1}} (\text{TP1 - TMETAL}) \quad \text{C.4}$$

2nd Primary Lump:

$$\frac{d}{dt} \text{TP2} = \frac{\text{WSG}}{\text{MP}} (\text{TP1 - TP2}) - \frac{\text{UPM SPM}}{\text{MP CP1}} (\text{TP1 - TMETAL}) \quad \text{C.5}$$

Outlet Plenum:

$$\frac{d}{dt} (\text{MSGOP CP1 TSGOP}) = \text{WSG CP1 (TP2 - TSGOP)} \quad \text{C.6}$$

or

$$\frac{d}{dt} \text{TSGOP} = \frac{\text{WSG}}{\text{MSGOP}} (\text{TP2 - TSGOP}) \quad \text{C.7}$$

Tube Metal Lump:

$$\frac{d}{dt} (\text{MMETAL CM TMETAL}) = \text{UPM SPM (TP1 - TMETAL) - UMS SMS (TMETAL - TS)} \quad \text{C.8}$$

or

$$\frac{d}{dt} \text{TMETAL} = \frac{\text{UPM SPM}}{\text{MMETAL CM}} (\text{TP1 - TMETAL}) - \frac{\text{UMS SMS}}{\text{MMETAL CM}} (\text{TMETAL - TS}) \quad \text{C.9}$$

TS = steam temperature, is related to PS, the steam pressure by

$$\text{TS} = \text{Constant} + \text{slope PS} \quad \text{C.10}$$

Secondary Fluid Lump:

$$k = (\text{MSW} \frac{\text{DHF}}{\text{PS}} + \text{MSS} \frac{\text{HG}}{\text{PS}} - \text{MSS} \frac{\text{HFG}}{\text{VFG}} \cdot \frac{\text{VG}}{\text{P}}) \quad \text{C.11}$$

$$\frac{d}{dt} \text{PS} = \frac{1}{k} (\text{UMS SMS (TMETAL - TS) - WSO (HG - CP2 TF1)}) \quad \text{C.12}$$

A schematic diagram for the detailed steam generator model is shown in Figure 4.8. The governing equations for the model are derived below (see Table C.2 for definition of variables).

Inlet Plenum:

$$\text{Energy balance: } \frac{d}{dt} (\text{MP1 CP1 TP1}) = \text{WP1 CP1 (THLEG - TP1)} \quad \text{C.13}$$

1st Primary Lump (PRL1):

$$\text{Mass balance: } \frac{d}{dt} (MP1) = WP1 - WP1 \quad C.14$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (MP1 \text{ CP1 TP1}) &= WP1 \text{ CP1 TP1} - WP1 \text{ CP1 TP1} \\ &- UPM \text{ PR1 LS1 (TP1 - TM1)} \end{aligned} \quad C.15$$

2nd Primary Lump (PRL2)

$$\text{Mass balance: } \frac{d}{dt} (MP2) + WP1 - WP2 \quad C.16$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (MP2 \text{ CP1 TP2}) &= WP1 \text{ CP1 TP1} - WP2 \text{ CP1 TP2} \\ &- UPM \text{ PR1 LS2 (TP2 - TM2)} \end{aligned} \quad C.17$$

3rd Primary Lump (PRL3)

$$\text{Mass balance: } \frac{d}{dt} (MP3) = WP2 - WP3 \quad C.18$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (MP3 \text{ CP1 TP3}) &= WP2 \text{ CP1 TP2} - WP3 \text{ CP1 TP3} \\ &- UPM \text{ PR1 LS2 (TP3 - TM3)} \end{aligned} \quad C.19$$

4th Primary Lump (PRL4)

$$\text{Mass balance: } \frac{d}{dt} (MP4) = WP3 - WP4 \quad C.20$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (MP4 \text{ CP1 TP4}) &= WP4 \text{ CP1 TP3} - WP4 \text{ CP1 TP4} \\ &- UPM \text{ PR1 LS1 (TP4 - TM4)} \end{aligned} \quad C.21$$

Outlet Plenum (PROUT):

$$\text{Energy balance: } \frac{d}{dt} (MP0 \text{ CP1 TP0}) = WP1 \text{ CP1 (TP4 - TP0)} \quad C.22$$

1st Metal Lump (MTL1):

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (\text{MM1 CM TM1}) &= \text{UPM PR1 LS1 (TP1 - TM1)} \quad \text{C.23} \\ &- \text{UMS1 PR2 LS1 (TM1 - TS1)} \\ &+ \text{ROM AM CM } \frac{(\text{TM1}+\text{TM2})}{2} \frac{d}{dt} \text{LS1} \end{aligned}$$

2nd Metal Lump (MTL2):

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (\text{MM2 CM TM2}) &= \text{UPM PR1 LS2 (TP2 - TM2)} \quad \text{C.24} \\ &- \text{UMS2 PR2 LS2 (TM2 - TSAT)} \\ &- \text{ROM AM CM } \frac{(\text{TM1}+\text{TM2})}{2} \frac{d}{dt} \text{LS1} \end{aligned}$$

3rd Metal Lump (MTL3):

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (\text{MM3 CM TM3}) &= \text{UPM PR1 LS2 (TP3 - TM3)} \quad \text{C.25} \\ &- \text{UMS2 PR2 LS2 (TM3 - TSAT)} \\ &- \text{ROM AM CM } \frac{(\text{TM3}+\text{TM4})}{2} \frac{d}{dt} \text{LS1} \end{aligned}$$

4th Metal Lump (MTL4):

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (\text{MM4 CM TM4}) &= \text{UPM PR1 LS1 (TP4 - TM4)} \quad \text{C.26} \\ &- \text{UMS1 PR2 LS1 (TM4 - TS1)} \\ &+ \text{ROM AM CM } \frac{(\text{TM3}+\text{TM4})}{2} \frac{d}{dt} \text{LS1} \end{aligned}$$

Subcooled Secondary Fluid Lump (SFSL):

$$\text{Mass balance: } \frac{d}{dt} \text{MS1} = \text{S1} - \text{S2} \quad \text{C.27}$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (\text{MS1 CP2 TS1}) &= \text{UMS1 PR2 LS1 (TM1 - TS1)} \quad \text{C.28} \\ &+ \text{UMS1 PR2 LS1 (TM4 - TS1)} \\ &+ \text{W1 CP2 TD} - \text{W2 CP2 TSAT} \end{aligned}$$

Boiling Secondary Fluid Lump (SFBL):

$$\bar{X} = \frac{XE}{2} \quad C.29$$

$$ROB = \frac{1}{VF - \bar{X} VFG} \quad C.30$$

$$HB = HF + \bar{X} HFG \quad C.31$$

$$HXE = HF + XE HFG \quad C.32$$

$$\text{Mass balance: } \frac{d}{dt} (ROB \text{ AFS LS2}) = W2 - W3 \quad C.33$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (ROB \text{ AFS LS2 HB}) &= UMS2 \text{ PR2 LS2 } (TM2 - TSAT) \\ &+ UMS2 \text{ PR2 LS2 } (TM3 - TSAT) \\ &+ W2 HF - W3 HXE \end{aligned} \quad C.34$$

Riser/Separator Volume:

$$ROR = \frac{1}{VF + XE VFG} \quad C.35$$

$$\text{Mass balance: } \frac{d}{dt} (VR \text{ ROR}) = W2 - W3 \quad C.36$$

Drum Water Volume:

$$WR = (1 \times \bar{X}E) W3 \quad C.37$$

$$\text{Mass balance: } \frac{d}{dt} (MDW) = WF1 + WR - WDW \quad C.38$$

$$\begin{aligned} \text{Energy balance: } \frac{d}{dt} (MDW \text{ CP2 TDW}) &= WF1 \text{ CP2 TF1} + WR \text{ CP2 TSAT} \\ &- WDW \text{ CP2 TDW} \end{aligned} \quad C.39$$

Drum Steam Volume:

$$\text{Mass balance: } \frac{d}{dt} MDS = XE W3 - WSO \quad C.40$$

Downcomer Secondary Fluid Lump (SFDCL):

$$\text{Mass balance: } W1 = WDW \quad \text{C.41}$$

$$\text{Energy balance: } \frac{d}{dt} (MD \text{ CP2 TD}) = WDW \text{ CP2 TDW} - W1 \text{ CP2 TD} \quad \text{C.42}$$

The Recirculation Loop Equation:

$$W1^2 = C1 \Delta PD \quad \text{C.43}$$

Responses of the simple steam generator model and the detailed steam generator model to a +10°F step change in the inlet temperature are shown in Figures C.1 and C.2, respectively.

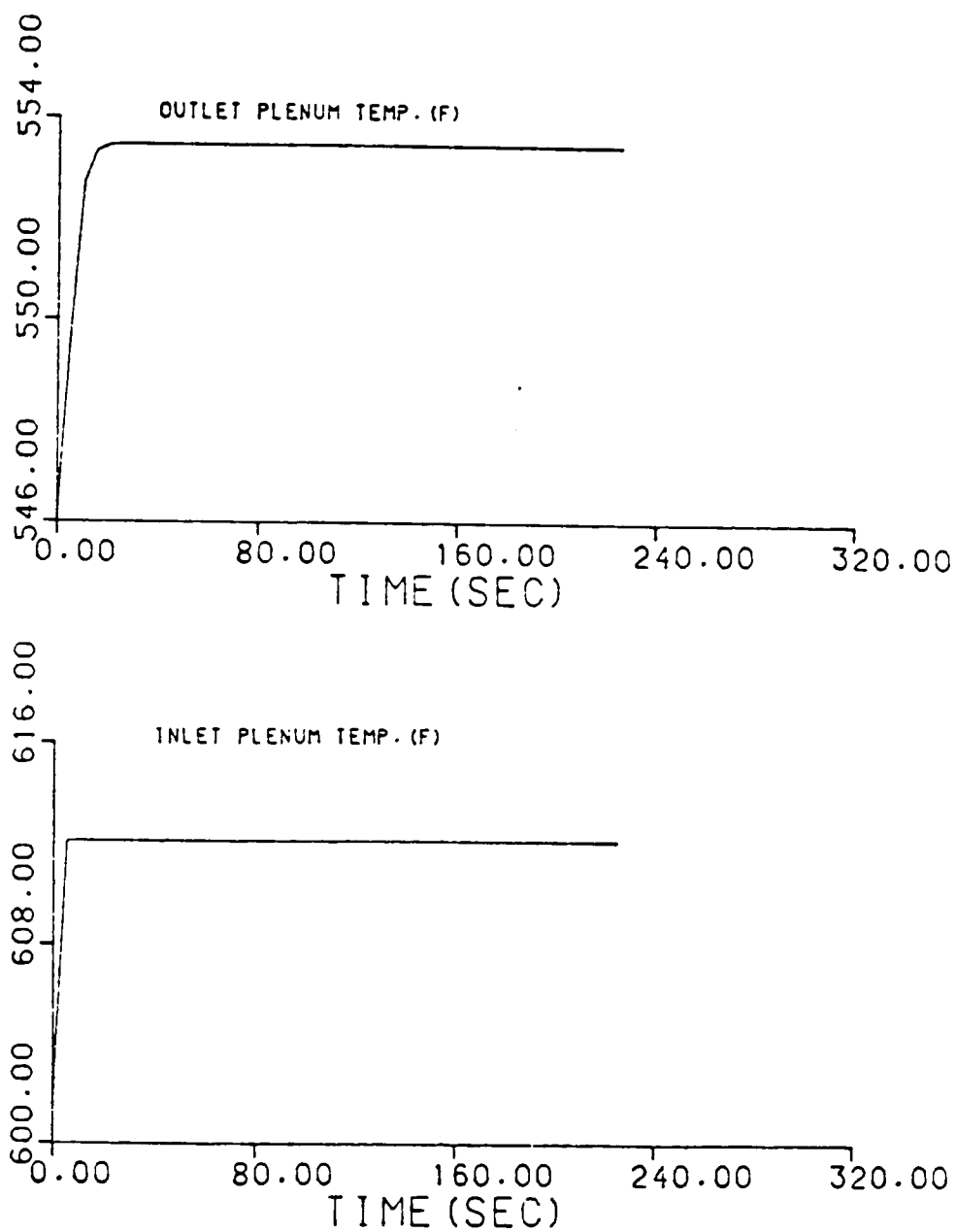


Figure C.1. Response of the simple steam generator model to a +10°F step change in the inlet temperature.

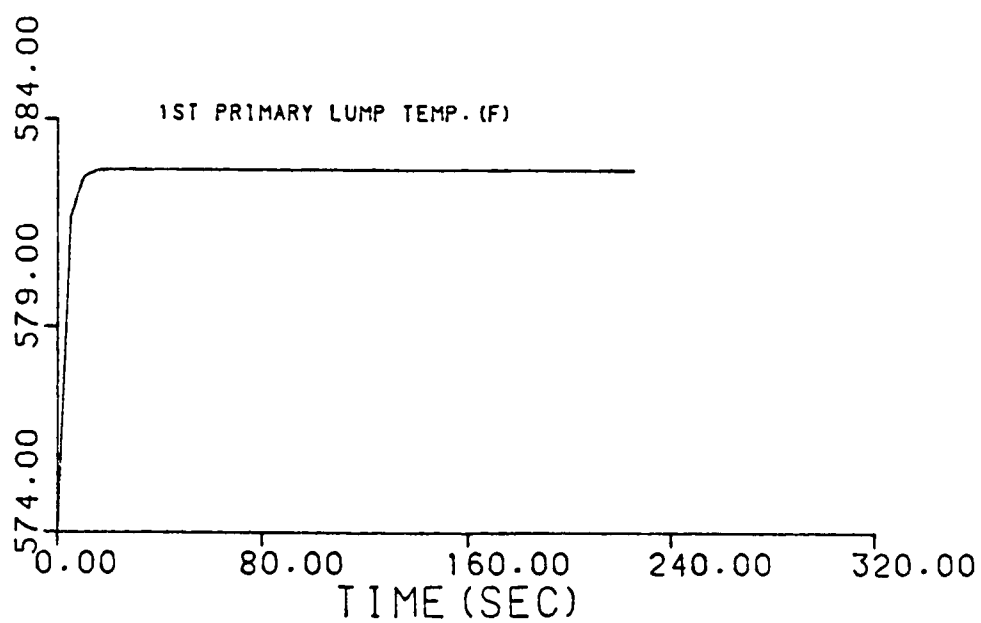
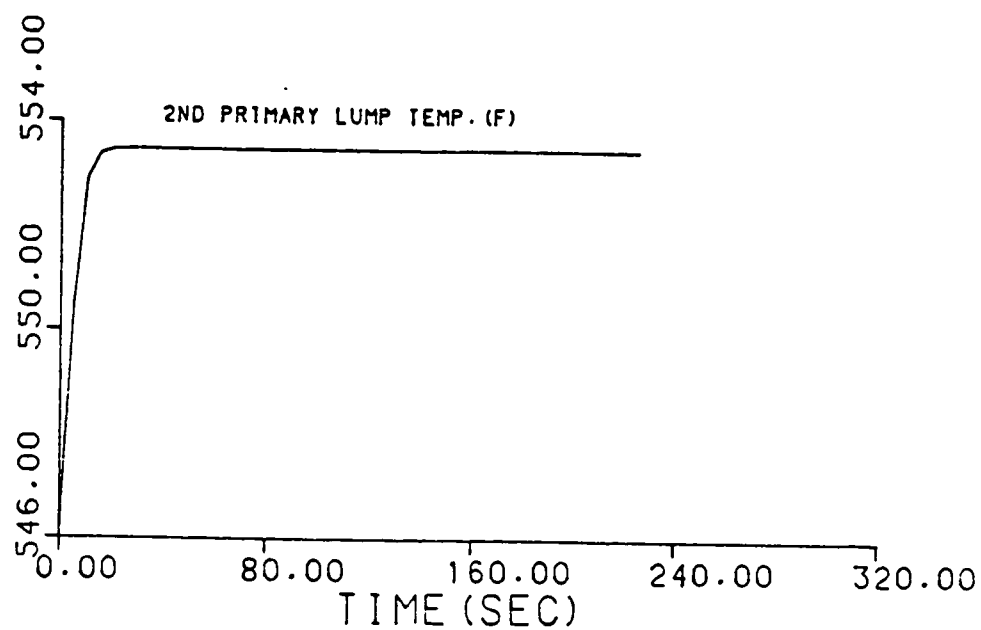


Figure C.1. (Continued).

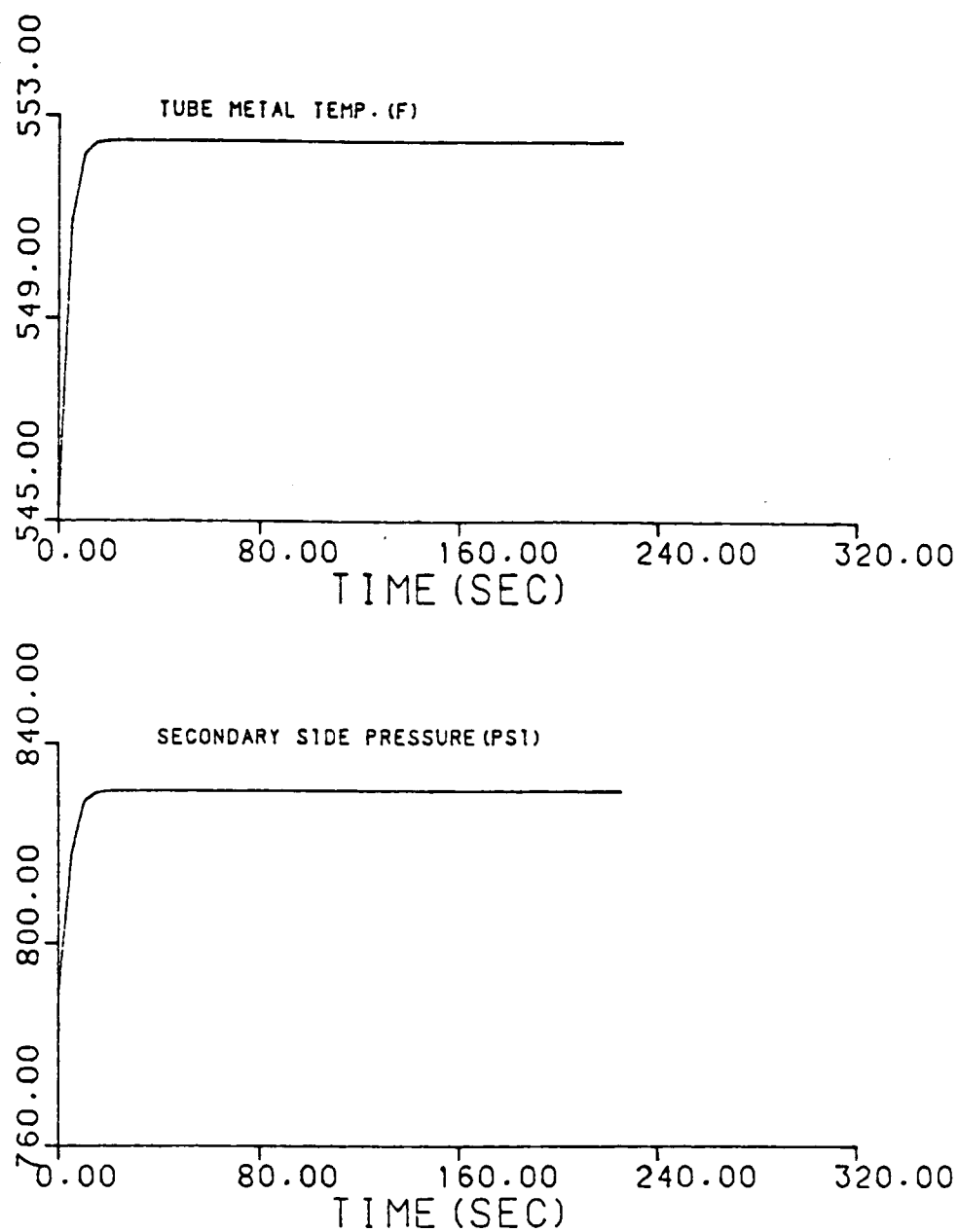


Figure C.1. (Continued).

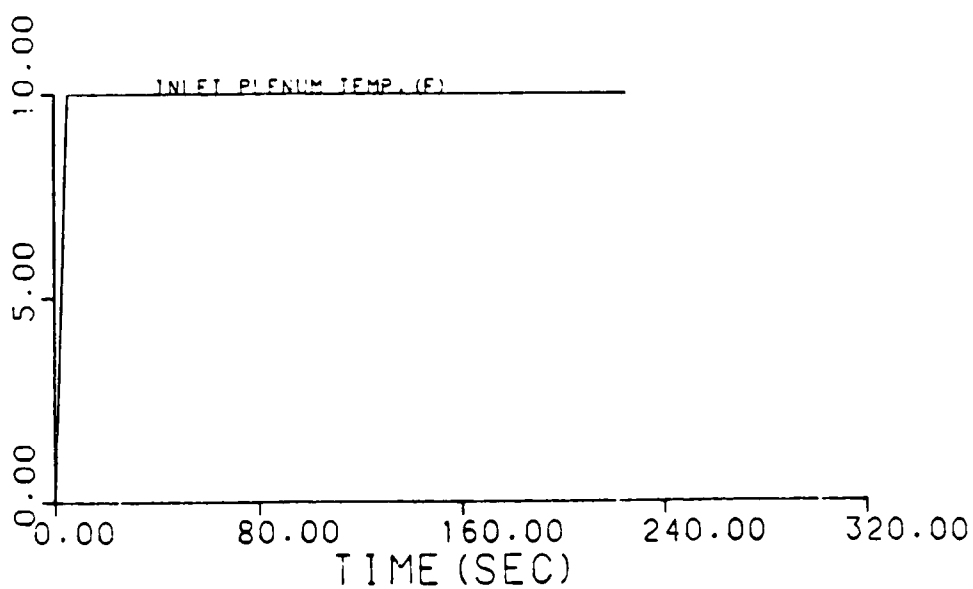
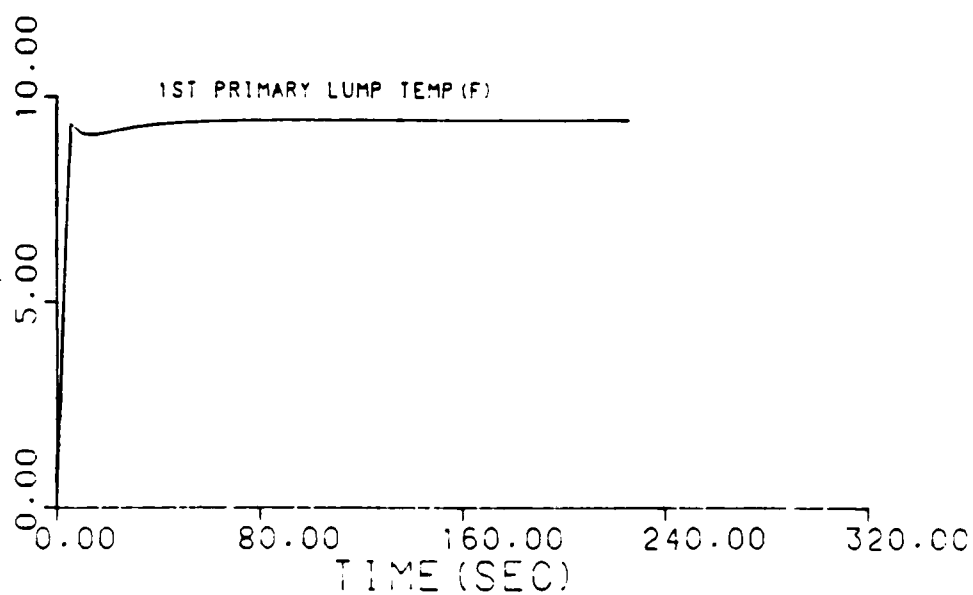


Figure C.2. Response of the detailed steam generator model to a +10°F step change in the inlet temperature.

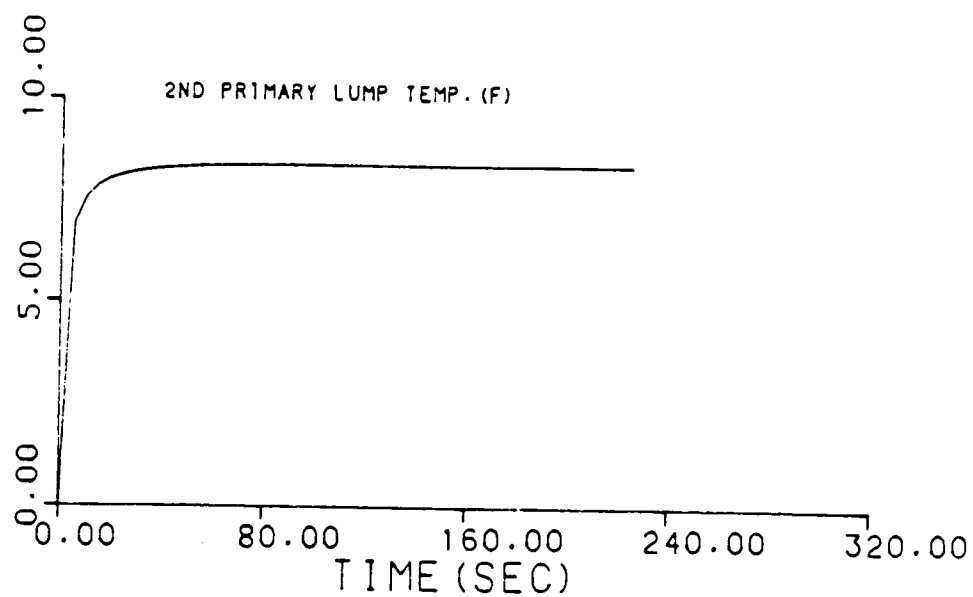
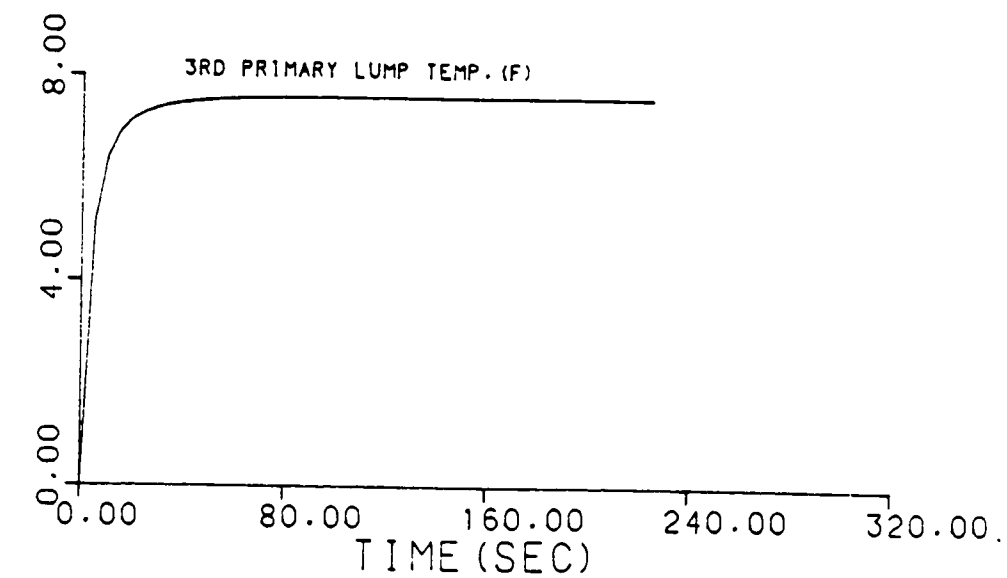


Figure C.2. (Continued).

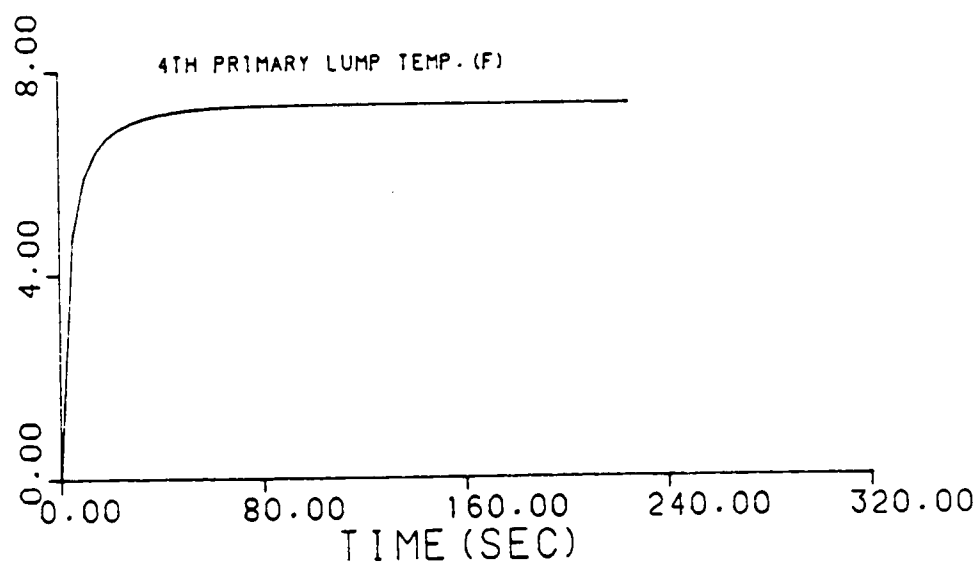
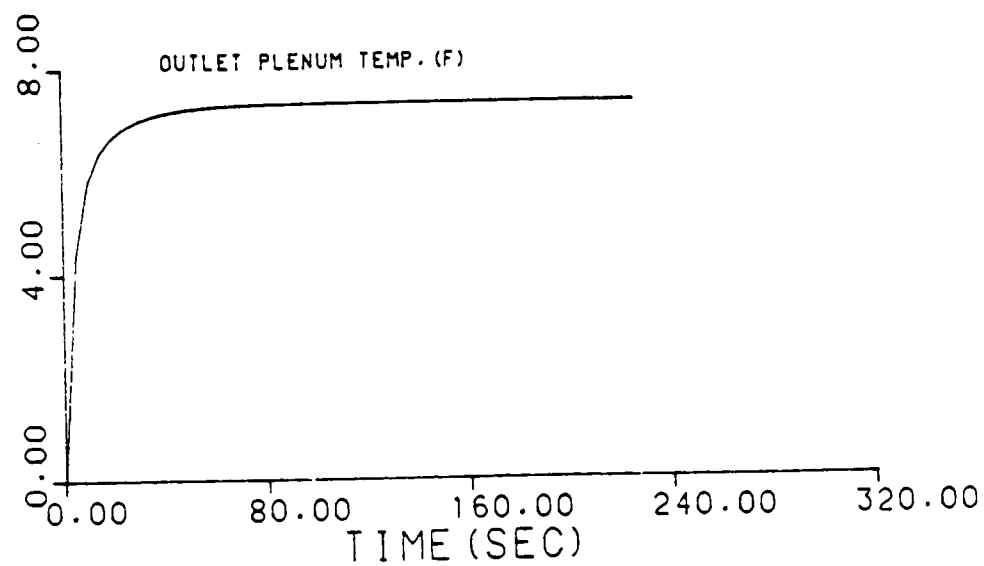


Figure C.2. (Continued).

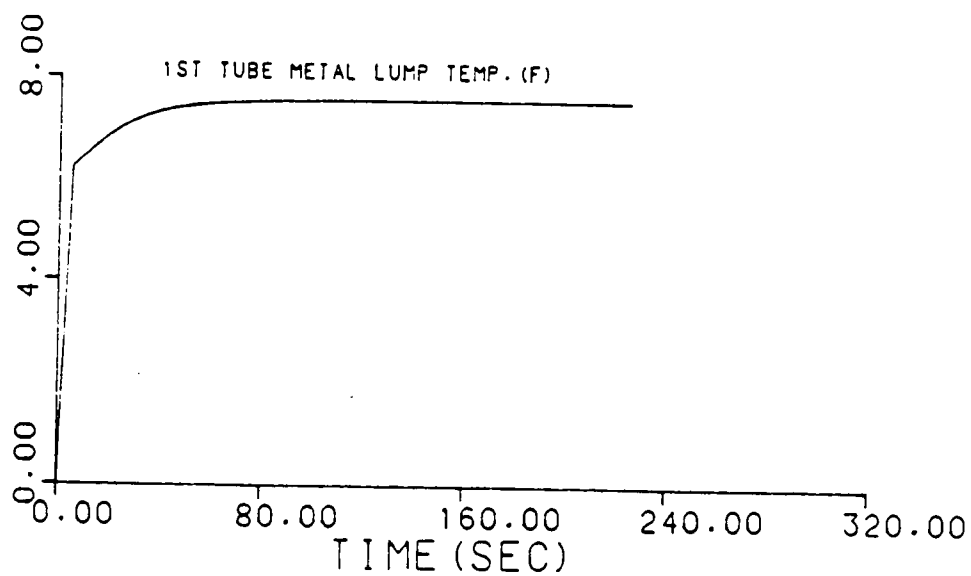
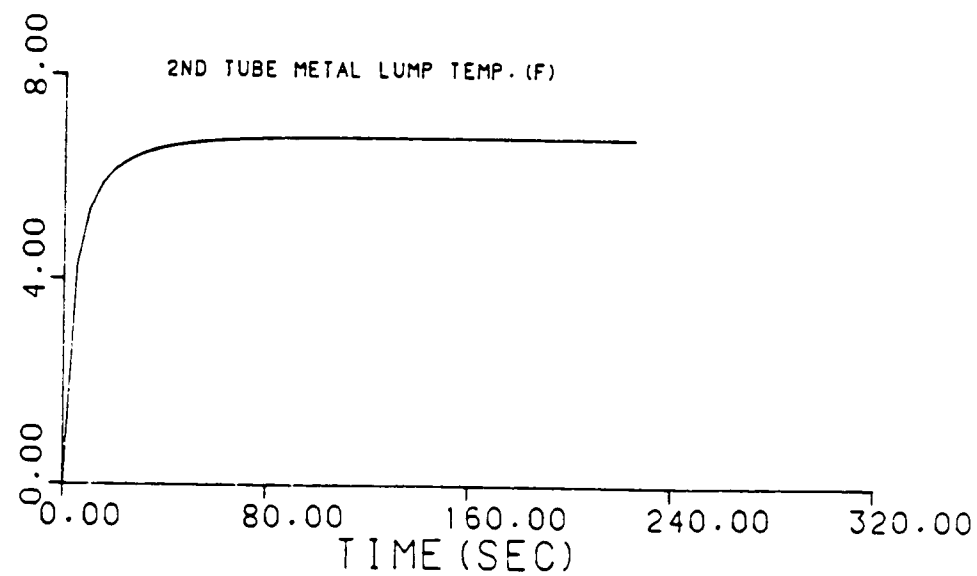


Figure C.2. (Continued).

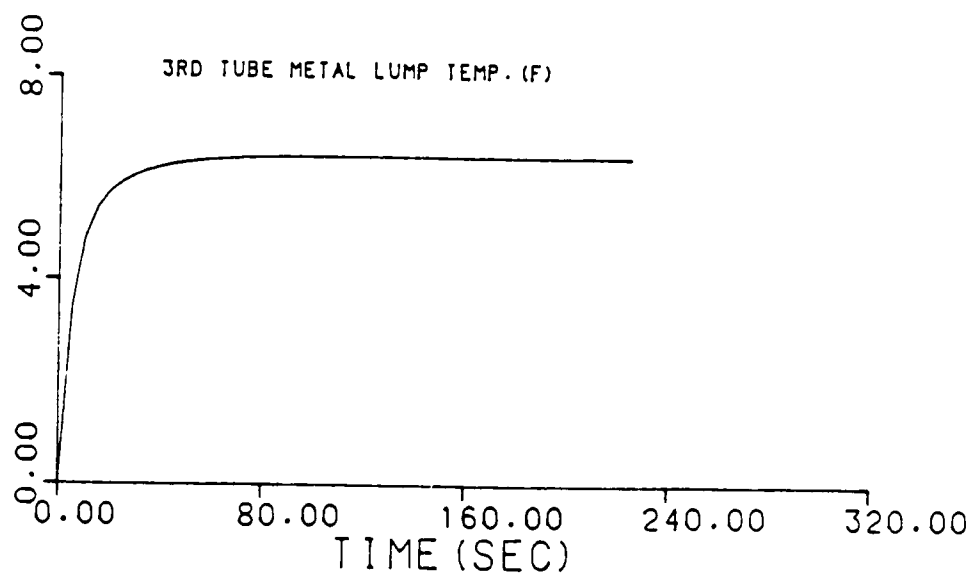
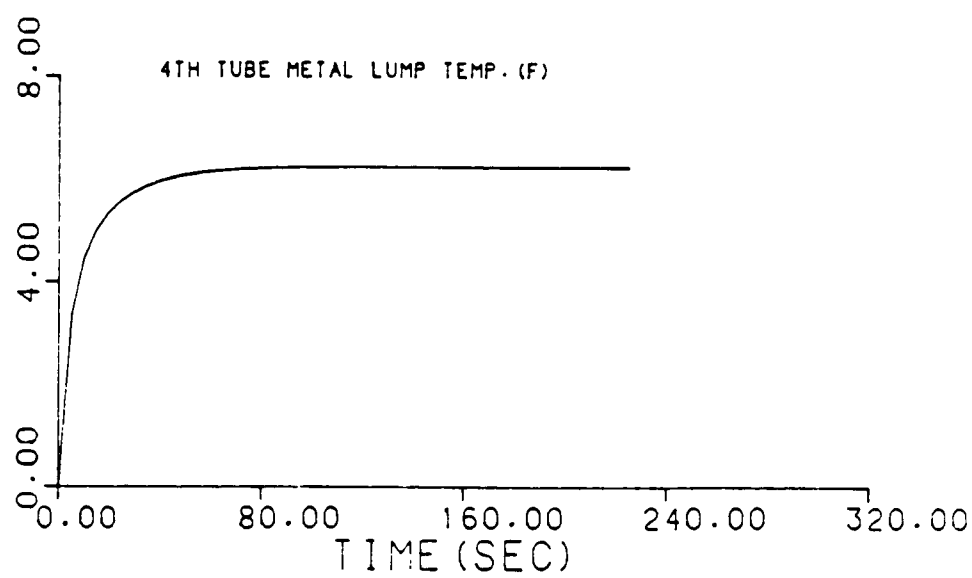


Figure C.2. (Continued).

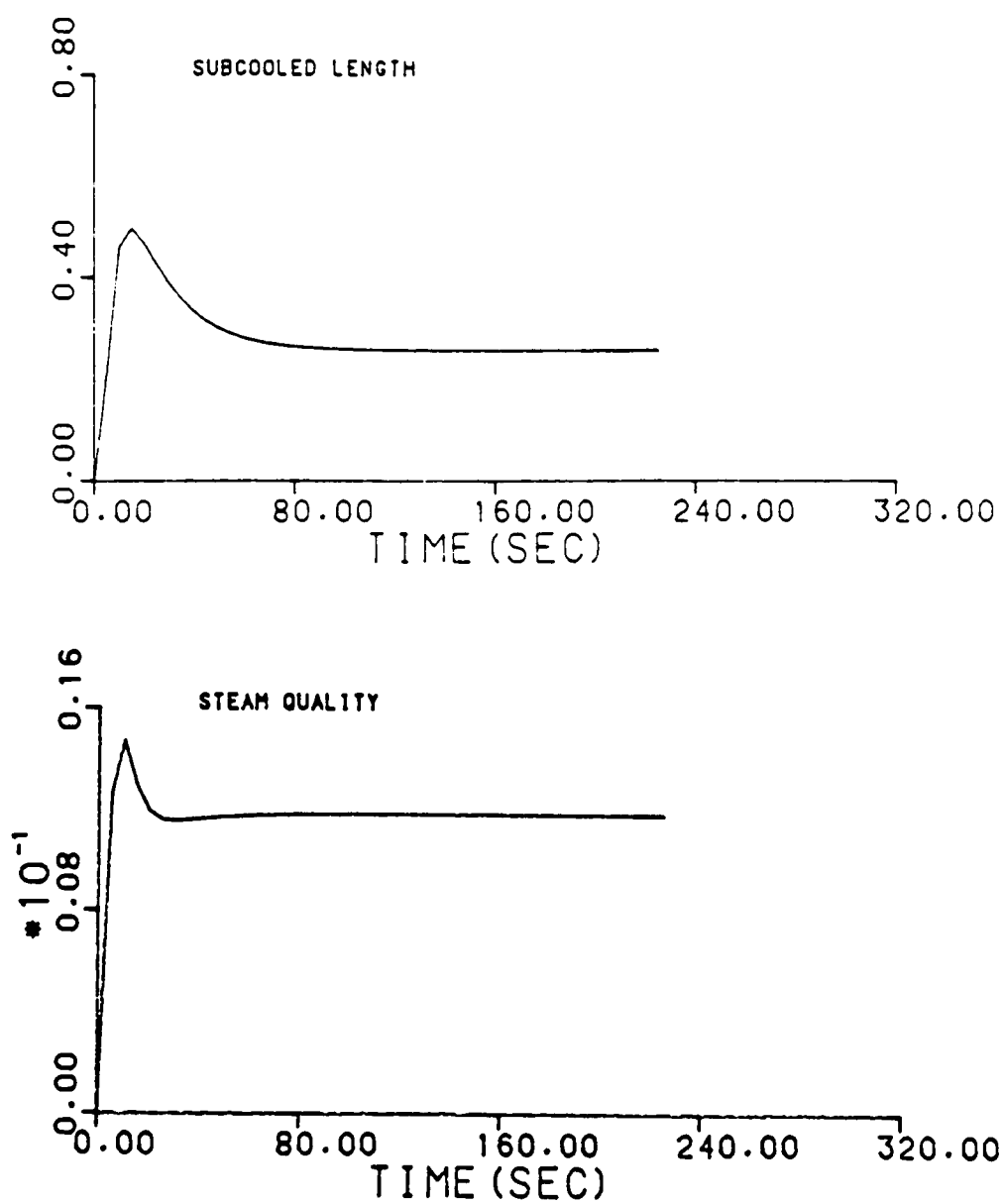


Figure C.2. (Continued).

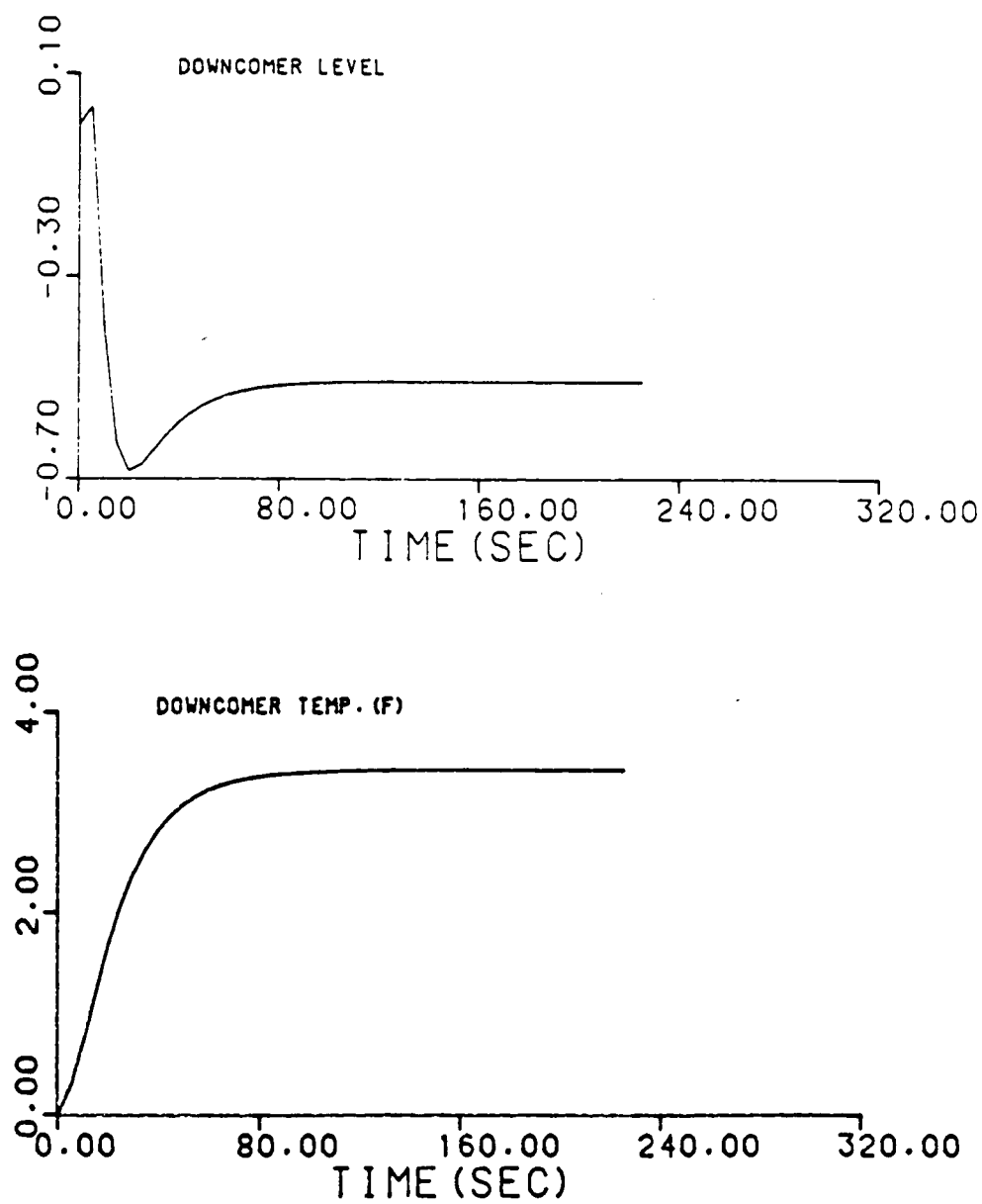


Figure C.2. (Continued).

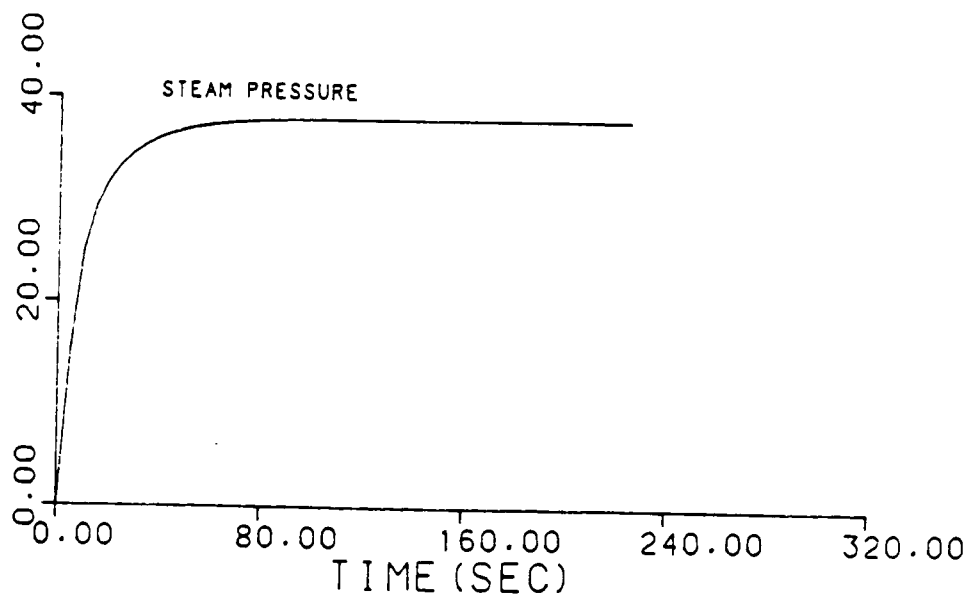


Figure C.2. (Continued).

TABLE C.1
VARIABLES USED IN THE STEAM GENERATOR SIMPLE MODEL

Variable	Definition
MSG1P	mass of the coolant in the steam generator inlet plenum
CP1	specific heat for the primary coolant
TSG1P	average temperature of the coolant in the steam generator inlet plenum
WSG	primary coolant flow rate inside the tubes
THLEG	average temperature of the coolant in the hot leg piping
MP	mass of the coolant in the primary lump per lump
TP1	average temperature of the coolant in the 1st primary lump
UPM	overall heat transfer coefficient between the primary coolant and the tube metal
SPM	overall heat transfer area between the primary coolant and the tube metal
TMETAL	average temperature of the tube metal
TP2	average temperature of the coolant in the second primary lump
MSGOP	mass of the coolant in the steam generator outlet plenum
TSGOP	average temperature of the coolant in the steam generator outlet plenum
MMETAL	total mass of the tube metal
UMS	overall heat transfer coefficient between the tube metal and the secondary side

TABLE C.1 (continued)

Variable	Definition
SMS	overall heat transfer area between the tube metal and the secondary side
CM	specific heat for the tube metal
TS	secondary side average temperature
PS	secondary side pressure
K	constant defined by equation C.11
MSW	mass of the secondary side water
HP	enthalpy of the fluid in the secondary side
HG	enthalpy of the steam in the secondary side
MSS	mass of the steam in the secondary side
HFG	latent heat of vaporization
VFG	the difference between the water specific volume and the steam specific volume at saturated temperature
VG	specific volume of the steam at saturated temperature
WSO	steam flow rate out of the steam generator
TF1	average temperature of the feed water
CP2	specific heat for the feed water
SWV	secondary side water volume
SSV	secondary side steam volume
DHF	derivative of the water enthalpy in relation to pressure at saturated condition

TABLE C.1 (continued)

Variable	Definition
DVG	derivative of the steam specific volume in relation to pressure at saturated condition
DHG	derivative of the steam enthalpy in relation to pressure at saturated condition

TABLE C.2
VARIABLES USED IN THE STEAM GENERATOR DETAILED MODEL

Variable	Definition
MS1	mass of the primary coolant in the steam generator inlet plenum
CP1	specific heat for the primary coolant
TP1	average temperature in the steam generator inlet plenum
WP1	primary water mass flow rate in the inlet plenum
THLEG	average temperature of the coolant in the hot leg piping
MP1	mass of the coolant in the first primary lump
WP1	coolant mass flow rate out of the first primary loop
TP1	average coolant temperature in the first coolant lump
UPM	overall heat transfer coefficient between the primary coolant and the tube metal
PR1	heat transfer area/unit length between primary water lump and tube metal
LS1	length of the secondary subcooled lump
TM1	average temperature in the first metal lump
MP2	mass of the primary coolant in the second coolant lump
WP2	coolant mass flow rate out of the second coolant lump
TP2	average coolant temperature in the second primary lump

TABLE C.2 (continued)

Variable	Definition
LS2	length of the boiling lump
TM2	average temperature in the second metal lump
MP3	mass of the primary coolant in the third coolant lump
WP3	coolant mass flow rate out of the third coolant lump
TP3	average coolant temperature in the third coolant lump
TM3	average temperature in the third coolant lump
MP4	mass of the primary coolant in the fourth coolant lump
WP4	primary mass flow rate out of the fourth coolant lump
TP4	average temperature in the fourth primary lump
TM4	average temperature in the fourth metal lump
MP0	mass of the primary coolant in the steam generator outlet plenum
TP0	average coolant temperature in the steam generator outlet plenum
MM1	mass of the tube metal in the first metal lump
CM	specific heat for the metal
UMS1	overall heat transfer coefficient between the tube metal and the subcooled secondary side

TABLE C.2 (continued)

Variable	Definition
PR2	heat transfer area/unit length between the tube metal and the secondary lump
TS1	average coolant temperature in the subcooled secondary lump
ROM	average density in the tube metal lump
AM	tube metal cross sectional area
MM2	mass of the tube metal in the second tube metal lump
UMS2	overall heat transfer coefficient between the tube metal and the boiling secondary lump
TSAT	secondary side saturated temperature
MM3	mass of the tube metal in the third metal lump
MM4	mass of the tube metal in the fourth metal lump
MS1	mass of the subcooled secondary water
W1	secondary fluid mass flow rate into the subcooled lump
W2	secondary fluid mass flow rate out of the subcooled lump
TD	average secondary water temperature in the down comer
XE	quality of the steam leaving the boiling lump
ROB	density of the secondary fluid in the boiling lump
VF	specific volume of the subcooled water in the secondary side

TABLE C.2 (continued)

Variable	Definition
VFG	
HB	enthalpy of the fluid in the boiling secondary lump
HF	enthalpy of the liquid water in the secondary side
HFG	latent heat of vaporization
HXE	enthalpy of the exit steam
AFS	flow area in the second lump
W3	secondary fluid mass flow rate out of the boiling lump
VR	volume of the riser separator section
WR	recirculated water flow rate
MDW	mass of the secondary fluid in the drum water section
WF1	feedwater flow rate
WDW	secondary water flow rate out of the drum water
TDW	average temperature in the secondary drum water section
MDS	mass of the steam in the secondary drum steam section
WSO	outlet steam mass flow rate
CD	effective dynamic pressure coefficient
ΔPD	driving pressure due to static head difference

APPENDIX D

```

C*****
C*
C*      PROGRAMMER : MOUNIR JAMIL MNEIMNEH
C*      SUBJECT    : COMPUTER PROGRAM SMLTOR.FOR
C*      DATE       : SEPT 83
C*
C*****
C*****
C*SPECIFY SET DIMENSIONS
C*****
      DOUBLE PRECISION FR(75),TIMER,HEAD,AFR(75),AFILE,CHOICE(75)
      REAL*8 X(6,75),X1(75),X5(6,75),EST(75),CO(75)
      REAL*8 F1(75),F2(75),F3(75),F4(75),F5(75),F6(75)
      REAL*8 F7(75),FF(6,75),E,ERROR(75),STEP(75),A(75),B(75)
      INTEGER ICASE1(75)
      COMMON ISTEP1,M,X,JJ,FF,FR,IFINE1,IEQO,AFR,AFILE,IEQOA1
      COMMON CHOICE,ICASE1,STIME
1     FORMAT(I5)
2     FORMAT(F12.6)
3     FORMAT(A10)
      IFINE1=0
      JJ=1
      ISTEP1=0
      IPOINT=1
      IERROR=0
      H=0.5
C*****
C* START RK4 METHOD
C*****
      TIMER='TIME.DAT'
      HEAD='DUMPER.DAT'
      AFIL='ALGEB.DAT'
C*****
C* TIME.DAT IS A FILE WHERE THE CURRENT TIME WILL BE SAVED.
C* DUMPER.DAT IS A FILE WHERE ALL ESTIMATED X(I) WILL BE DUMPED.
C* THIS WILL BE DONE IN ORDER X(1),X(2)...X(N)-X(1),X(2)...X(N)-
C*
C*      -----
C*      1ST STEP          2ND STEP
C*
C* ....TO THE FINAL STEP.
C* ALGEB.DAT IS A FILE WHERE RESULTS OF IMPORTANT ALGEBRAIC EQUA-
C* TIONS MAY BE STORED. EXAMPLE WHEN USING THE PROMPT JUMP ASSUM-
C* PTION FOR THE REACTOR POWER EQUATION, THE POWER EQUATION WILL
C* CEASED TO BE A DIFFERENTIAL EQUATION ANY FURTHER. BUT WE STILL
C* WOULD LIKE TO SEE THE POWER RESPONSE. IN THIS CASE THE POWER
C* RESPONSE ALONG MANY OTHER ALGEBRAIC RESULTS(IF DESIRED)MAY BE
C* STORED IN THIS FILE.
C*****
      OPEN(UNIT=21,DEVICE='DSKD',FILE=HEAD)
      OPEN(UNIT=22,DEVICE='DSKD',FILE=TIMER)
      OPEN(UNIT=23,DEVICE='DSKD',FILE=AFIL)
      STIME=0.0
      WRITE(5,673)
      READ(5,*) TTIME
673  FORMAT(1X,'ENTER TERMINATING TIME,TTIME='$)
C*****
C* TERMINATING TIME IS THE TIME WHERE THE RUN WILL BE TERMINATED.*
C*****
      WRITE(5,9823)
      READ(5,*) SAMPLE
9823  FORMAT(1X,'ENTER THE SAMPLING TIME,SAMPLE='$)
C*****

```

```

C* SAMPLING FEATURE ENABLES THE USER TO SPECIFY THE RATE AT *
C* WHICH THE DATA WILL BE SAVED THE USER STORAGE DISK. THIS *
C* WILL PREVENT USING UP ALL THE STORAGE ROOM TRYING TO STORE *
C* ALL THE DATA POINTS WHEN THE TIME STEP IS VERY SMALL. *
C*****
      SK1=SAMPLE
      ISTEP1=0
      M=1
      JJ=1
      CALL PWRSUB
      JJ=1
      DO 4389 I=1,IEQ0
      X1(I)=X(1,I)
      WRITE(21,*) X(1,I),STIME
4389  CONTINUE
      IFINE1=1
      WRITE(22,*) STIME
      ISTEP1=ISTEP1+1
      DO 79 LM=1,4
      M=LM
5913  STIME=STIME+H
      DO 72 I=1,IEQ0
      X(M,I)=X1(I)
72    CONTINUE
      JJ=0
      CALL PWRSUB
      JJ=1
      CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F1 IN RUNGE-KUTTA-FEHLBERG SCHEME. *
C*****
105   DO 73 I=1,IEQ0
      F1(I)=H*FF(M,I)
      X(M,I)=X1(I)+F1(I)/4.0
73    CONTINUE
      JJ=0
      CALL PWRSUB
      JJ=1
      CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F2 IN RUNGE-KUTTA-FEHLBERG SCHEME. *
C*****
      DO 74 I=1,IEQ0
      F2(I)=H*FF(M,I)
      X(M,I)=X1(I)+(3.0*F1(I)+9.0*F2(I))/32.0
74    CONTINUE
      JJ=0
      CALL PWRSUB
      JJ=1
      CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F3 IN RUNGE-KUTTA-FEHLBERG SCHEME. *
C*****
      DO 75 I=1,IEQ0
      F3(I)=H*FF(M,I)
      X(M,I)=X1(I)+(1932.0*F1(I)-7200.0*F2(I)
      * +7296.0*F3(I))/2197.0
75    CONTINUE
      JJ=0
      CALL PWRSUB
      JJ=1

```

```

      CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F4 IN RUNGE-KUTTA-FEHLBERG SCHEME.  *
C*****
      DO 76 I=1,IEQO
        F4(I)=H*FF(M,I)
        X(M,I)=X1(I)+439.0*F1(I)/216.0-8.0*F2(I)
        * +3680.0*F3(I)/513.0-845.0*F4(I)/4104.0
76      CONTINUE
        JJ=0
        CALL PWRSUB
        JJ=1
        CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F5 IN RUNGE-KUTTA-FEHLBERG SCHEME.  *
C*****
      DO 77 I=1,IEQO
        F5(I)=H*FF(M,I)
        X(M,I)=X1(I)-8.0*F1(I)/27.0+2.0*F2(I)
        * -3544*F3(I)/2565.0+1859.0*F4(I)/4104.0
        * -11.0*F5(I)/40.0
77      CONTINUE
        JJ=0
        CALL PWRSUB
        JJ=1
        CALL PWRSUB
C*****
C* NEXT DO LOOP IS TO FIND F6 IN RUNGE-KUTTA-FEHLBERG SCHEME.  *
C*****
      DO 78 I=1,IEQO
        F6(I)=H*FF(M,I)
C*****
C* X IS THE ESTIMATED VALUE FOR X(T+H).X5 IS ANOTHER ESTIMATION *
C* FOR X(T+H).THE ERROR IS THE DIFFERENCE BETWEEN THESE TWO EST- *
C* IMATIONS. *
C*****
        X(M,I)=X1(I)+(25.0*F1(I)/216.0
        * +1408.0*F3(I)/2565.0+2197.0*F4(I)/4104.0
        * -F5(I)/5.0)
        X5(M,I)=X1(I)+(16.0*F1(I)/135.0
        * +6656.0*F3(I)/12825.0+28561.0*F4(I)/56430.0
        * -9.0*F5(I)/50.0+2.0*F6(I)/55.0)
        EST(I)=ABS(X(M,I)-X5(M,I))
78      CONTINUE
C*****
C* NEXT FEW LINES ARE ERROR CONTROL STATMENTS.THE ERROR IS LIMITED*
C* TO 0.5E-10<=ERROR<=0.5E-3.IF THE ERROR IS OUTSIDE THIS BAND H *
C* WILL BE MULTIPLIED BY 2 OR DIVIDED BY 2. *
C*****
        E=EST(1)
        IF(IEQO.EQ.1)GOTO 508
        DO 108 II=1,IEQO-1
          IF(E.GE.EST(II+1)) GOTO 108
107      E=EST(II+1)
108      CONTINUE
508      CONTINUE
          IFLAG1=0
          IFLAG2=0
          IF(0.5E-10.LE.E) GOTO 101
          IFLAG1=1
          H1=H*2

```

```

101  IF(E.LT.0.5E-5) GOTO 102
      IFLAG2=1
      H1=H/2
102  IF(IFLAG2.EQ.1) GOTO 1273
      IF(H.GE.SAMPLE) H=SAMPLE
      IF(H.GE.SAMPLE) GOTO 103
      IF(IFLAG1.EQ.0.AND.IFLAG2.EQ.0) GOTO 103
1273  STIME=STIME-H
      H=H1
      GOTO 5913
103  CONTINUE
7485      DO 501 I=1,IEQO
      X1(I)=X(M,I)
501  CONTINUE
C*****
C* IF IT IS TIME TO SAMPLE THE DATA, THE PROGRAM WILL PASS THRO- *
C* UGH THE NEXT FEW LINES. OTHERWISE SKIP TO 79. THIS SAMPLING *
C* FEATURE IS NECESSARY TO AVOID USING UP THE DISK STORAGE AREA *
C* TRYING TO SAVE THE RESULTS AT EACH TIME STEP WHICH MIGHT BE *
C* VERY SMALL. *
C*****
      IF(SK1.GE.STIME) GOTO 79
      WRITE(5,*)H,STIME,E
      WRITE(22,*) STIME
C*****
C* IT TIME TO STORE THE DATA. GOTO TO PWRSUB AND SEE IF ANY ALG- *
C* EBRAIC RESULTS IS TO BE STORED IN ALGEB.DAT DATA FILE. *
C*****
      JJ=1
      IFINE1=0
      CALL PWRSUB
      IFINE1=1
C*****
C* TIME FOR NEXT STEP. *
C*****
      IPOINT=IPOINT+1
      SK1=SK1+SAMPLE
      DO 983 I=1,IEQO
      WRITE(21,*) X(M,I),STIME
983  CONTINUE
79  CONTINUE
C*****
C* ONCE THE FIRST FOUR POINTS ARE ESTIMATED, THE PROGRAM WILL MOVE *
C* TO THE PREDICTOR-CORRECTOR SCHEME. *
C*****
C* START A/M
C*****
      DO 80 LM=1,4
      M=LM
      JJ=0
      CALL PWRSUB
      JJ=1
      CALL PWRSUB
80  CONTINUE
C*****
C* ABOVE DO LOOP IS TO FIND THE VALUE OF THE FUNCTION AT STEPS *
C* 1,2,3,AND 4. *
C*****
5061  CONTINUE
506  STIME=STIME+H

```

```

      IF(IERROR.GT.4) IERROR=0
      IERROR=IERROR+1
C*****
C* NEXT DO LOOP IS THE PREDICTOR EQUATION .
C*****
      81 DO 82 I=1,IEQO
          X(6,I)=X(4,I)+H*(55*FF(4,I)-59*FF(3,I)
          * +37*FF(2,I)-9*FF(1,I))/24
      82 CONTINUE
          JJ=0
          M=6
          CALL PWRSUB
          JJ=1
          CALL PWRSUB
C*****
C* NEXT DO LOOP IS THE CORRECTOR EQUATION.
C*****
      DO 83 I=1,IEQO
          X(5,I)=X(4,I)+H*(9*FF(6,I)+19*FF(4,I)
          * -5*FF(3,I)+FF(2,I))/24.0
          A(I)=X(5,I)
          B(I)=X(6,I)
          IF(ABS(A(I)).EQ.0) GOTO 9843
          IF(A(I).LT.0.5E-15) B(I)=1.0
          EST(I)=(19.0*ABS(A(I)-B(I)))/(270*ABS(A(I)))
          GOTO 83
      9843 EST(I)=0.0
      83 CONTINUE
          M=5
          JJ=0
          CALL PWRSUB
          JJ=1
          CALL PWRSUB
C*****
C* NEXT FEW LINES ARE ERROR CONTROL STATMENTS.
C*****
      E=EST(1)
      IF(IEQO.EQ.1) GOTO 7023
      DO 109 II=1,IEQO-1
          IF(E.GE.EST(II+1)) GOTO 109
      111 E=EST(II+1)
      109 CONTINUE
      7023 CONTINUE
          ERROR(IERROR)=E
          IF(ERROR(1).EQ.ERROR(3).AND.ERROR(2).EQ.ERROR(4)) ITIP=1
          IF(ITIP.EQ.1)GOTO 7433
          ITIP=0
      7433 IFLAG1=0
          IFLAG2=0
          IF(0.5E-10.LE.E) GOTO 113
          IFLAG1=1
          H1=H*2
          STEP(IERROR)=H1
      113 IF(E.LE.0.5E-5) GOTO 114
          IFLAG2=1
          H1=H/2
          STEP(IERROR)=H1
      114 IF(IFLAG2.EQ.1) GOTO 5432
          IF(H.GE.SAMPLE) H=SAMPLE
          IF(H.GE.SAMPLE) GOTO 115
          IF(IFLAG1.EQ.0.AND.IFLAG2.EQ.0) GOTO 115

```

```

5432 STIME=STIME-H
      IF(ITIP.EQ.0) GOTO 7451
      IF(STEP(1).OR.STEP(3).LE.STEP(2).OR.STEP(4)) H=STEP(1)
      IF(STEP(2).OR.STEP(4).LE.STEP(1).OR.STEP(3)) H=STEP(2)
      IOUT=1
7451 CONTINUE
      H=H1
      IF(IOUT.EQ.1)GOTO 115
      GOTO 506
115  CONTINUE
      IERROR=0
      IOUT=0
C*****
C* IF IT TIME TO SAMPLE GO THROUGH NEXT LINES. OTHERWISE GOTO 1325 *
C*****
      IF(SK1.GE.STIME) GOTO 1325
      WRITE(5,*) H,STIME,E,IERROR
      WRITE(22,*) STIME
      DO 9041 I=1,IEQO
      WRITE(21,*) X(M,I),STIME
9041 CONTINUE
      SK1=SK1+SAMPLE
      JJ=1
      IFINE1=0
      CALL PWRSUB
      IFINE1=1
      IPOINT=IPOINT+1
C*****
C* SET X(I-1)=X(I) AND SOLVE FOR THE NEXT X(I). *
C*****
1325 DO 84 LM=1,4
      M=LM
      DO 85 I=1,IEQO
      X(M,I)=X(M+1,I)
      FF(M,I)=FF(M+1,I)
85  CONTINUE
84  CONTINUE
C*****
C* IF CURRENT TIME(STIME) IS STILL LESS THAN TERMINATING(TTIME) *
C* DO NEXT STEP. OTHERWISE CONTINUE TO THE FILING SECTION. *
C*****
      IF(STIME.LT.TTIME-H) GOTO 5061
      CLOSE(UNIT=21,DEVICE='DSKD',FILE=HEAD)
      CLOSE(UNIT=22,DEVICE='DSKD',FILE=TIMER)
      CLOSE(UNIT=23,DEVICE='DSKD',FILE=AFILE)
C*****
C*DIVIDE FILE DUMPER.DAT INTO SUBFILES. *
C*****
      DO 303 I=1,IEQO
      ITIME=1
      OPEN(UNIT=21,DEVICE='DSKD',FILE=HEAD)
      OPEN(UNIT=22,DEVICE='DSKD',FILE=FR(I))
      WRITE(22,*) IPOINT
301  DO 302 K=1,IEQO
      READ(21,*) CO(K),STIME
302  CONTINUE
      WRITE(22,*) CO(I),STIME
      ITIME=ITIME+1
      IF(ITIME.LT.IPOINT-1) GOTO 301
      CLOSE(UNIT=22,DEVICE='DSKD',FILE=FR(I))
      CLOSE(UNIT=21,DEVICE='DSKD',FILE=HEAD)

```

303 CONTINUE

C*****
 C* IF OUTPUT OF ANY ALGEBRAIC EQUATION IS SAMPLED THEN DIVIDE FILE*
 C* ALGEB.DAT INTO SUBFILES. *

C*****

```

      IF(IEQOA1.LT.1) GOTO 5000
      DO 361 I=1,IEQOA1
      ITIME=1
      OPEN(UNIT=23,DEVICE='DSKD',FILE=AFILE)
      OPEN(UNIT=22,DEVICE='DSKD',FILE=AFR(I))
      WRITE(22,*) IPOINT
351   DO 352 K=1,IEQOA1
      READ(23,*) CO(K),STIME
352   CONTINUE
      WRITE(22,*) CO(I),STIME
      ITIME=ITIME+1
      IF(ITIME.LT.IPOINT-1) GOTO 351
      CLOSE(UNIT=22,DEVICE='DSKD',FILE=AFR(I))
      CLOSE(UNIT=23,DEVICE='DSKD',FILE=AFILE)
361   CONTINUE
5000  END

```

```

C*****
C* TOP.FOR: THIS TOP SECTION OF THE SUBPROGRAM PWRSUB IS DESIGNED TO*
C*     TRANSFER SOME PARAMETERS FROM THE MAIN PROGRAM TO THE SUB*
C*     ROUTINE ALSO IN THIS SECTION THE USER SPECIFY NAME OF  *
C*     THE SYSTEM/SUBSYSTEM WHERE PERTURBATION IS TO APPLIED.  *
C*****
      SUBROUTINE PWRSUB
      DOUBLE PRECISION FR1(75),HEAD1,AFR1(75),AFILE1,CHOICE(75)
      DOUBLE PRECISION YTITLE(75,6),TITLE,YTITL(75,6)
      REAL*8 X2(6,75),F(6,75),EF(500),FORC(500)
      INTEGER ICASE(75)
      COMMON ISTEP,M,X2,JJ1,F,FR1,IFINE,IEQ01,AFR1,AFILE1,IEQ0A
      COMMON CHOICE,ICASE,STIME
3     FORMAT(A10)
15    FORMAT(6A10)
9     FORMAT(1X,'*****')
      IF(ISTEP.NE.0) GOTO 5
      WRITE(5,9);WRITE(5,6)
      READ(5,*) IPER
6     FORMAT(1X,'* ENTER # OF THE SYSTEMS TO BE PERTURBED,IPER='$)
      WRITE(5,9)
      DO 7 IY=1,IPER
      WRITE(5,4) IY,IY
      READ(5,3) CHOICE(IY)
4     FORMAT(1X,'* ENTER NAME OF THE SYSTEM FILE WHERE',/,
* 1X,'* PERTURBATION(',I3,2X,') IS TO BE APPLIED',/,
* 1X,'* FILENAME(',I3,2X,')='$)
7     CONTINUE
      WRITE(5,9);WRITE(5,10);WRITE(5,13);WRITE(5,11);WRITE(5,12)
      WRITE(5,9)
      WRITE(5,14);READ(5,*) IREFF
10    FORMAT(1X,'* CHOOSE A CASE BY ENTERING THE # TO THE  ')
13    FORMAT(1X,'* LEFT  LEFT')
11    FORMAT(1X,'*          1- REFERENCE CASE          ')
12    FORMAT(1X,'*          2- NEW CASE                ')
14    FORMAT(1X,'CASE='$)
5     ICOUNT=1
      IPRIME=0
      IJ=1
      IK=0
      IJA=1
      IJKA=0
      IAM=1

```

```

C*****
C* PCORE5.FOR: A NONLINEAR LUMPED PARAMETER MODEL FOR A TYPICAL *
C* PRESSURIZED WATER REACTOR CORE.FIVE LUMPS ARE USED *
C* TO REPRESENT THE CORE.ONE LUMP FOR THE FUEL,TWO LUMPS *
C* FOR THE COOLANT,ONE LUMP FOR THE LOWER PLENUM AND ONE *
C* LUMP FOR THE UPPER PLENUM.PROMPT JUMP ASSUMPTION IS *
C* ASSUMED FOR THE POWER EQUATION. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IJA=IJK+1
      IF(ISTEP.NE.0)GOTO 4016
      DO 4000 IY=1,IPEP
      IF(CHOICE(IY).EQ.'PCORE5.FOR') GOTO 4001
4000  CONTINUE
      GOTO 4011
C*****
C* ENTER THE FORCING TERM. *
C*****
4001  DO 4002 ICLEAR=1,24
      WRITE(5,*)
4002  CONTINUE
      WRITE(5,4003);WRITE(5,4004);WRITE(5,4005);WRITE(5,4006)
      WRITE(5,4003);WRITE(5,4007);READ(5,*) ICASE(IAM)
4003  FORMAT(1X,'*****')
4004  FORMAT(1X,'* CHOOSE A CASE BY ENTERING THE # TO THE LEFT. *')
4005  FORMAT(1X,'* 1- ISOLATED CORE CASE. *')
4006  FORMAT(1X,'* 2- CORE COUPLED TO A STEAM GENERATOR. *')
4007  FORMAT(1X,'CASE=')$)
      WRITE(5,4003);WRITE(5,4008);READ(5,*) DROH
4008  FORMAT(1X,'*ENTER DEVIATION IN EXTERNAL REACTIVITY($),DROH=$)
      IF(ICASE(IAM).EQ.2) GOTO 4010
      WRITE(5,4009);READ(5,*) DTCLEG
4009  FORMAT(1X,'*ENTER DEVIATION IN THE COLDLEG TEMPP.(F),DTCLED=$)
      WRITE(5,4003)
4010  IAM=IAM+1
C*****
C* OPEN A FILE AND READ MODEL'S PARAMERTERS. *
C*****
4011  DO 4012 ICLEAR=1,24
      WRITE(5,*)
4012  CONTINUE
      IF(IREFF.EQ.1) GOTO 4014
      WRITE(5,4003);WRITE(5,4013);WRITE(5,4003);READ(5,3)HEAD1
4013  FORMAT(1X,'* ENTER A FILE NAME FOR THE CORE DATA,FILENAME=$)
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 4015
4014  HEAD1='CORE5.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
4015  READ(20,3)TITLE;READ(20,3) AFR1(IJA)
      READ(20,3)TITLE;READ(20,3) AFR1(IJA+1)
      READ(20,3)TITLE;READ(20,3) FR1(IJ)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+1)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+2)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+3)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+4)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+5)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+6)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+7)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+8)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+9)

```

```

READ(20,3)TITLE;READ(20,3) FR1(IJ+10)
READ(20,3)TITLE;READ(20,15) (YTITL(IJA,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJA+1,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+1,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+2,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+3,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+4,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+5,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+6,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+7,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+8,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+9,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,15) (YTITL(IJ+10,IOP),IOP=1,6)
READ(20,3)TITLE;READ(20,*) ROM1
READ(20,3)TITLE;READ(20,*) OMEGA
READ(20,3)TITLE;READ(20,*) ALFAF
READ(20,3)TITLE;READ(20,*) ALFAC
READ(20,3)TITLE;READ(20,*) ALFAPR
READ(20,3)TITLE;READ(20,*) BTOTAL
READ(20,3)TITLE;READ(20,*) DECAY1
READ(20,3)TITLE;READ(20,*) DECAY2
READ(20,3)TITLE;READ(20,*) DECAY3
READ(20,3)TITLE;READ(20,*) DECAY4
READ(20,3)TITLE;READ(20,*) DECAY5
READ(20,3)TITLE;READ(20,*) DECAY6
READ(20,3)TITLE;READ(20,*) BETA1
READ(20,3)TITLE;READ(20,*) BETA2
READ(20,3)TITLE;READ(20,*) BETA3
READ(20,3)TITLE;READ(20,*) BETA4
READ(20,3)TITLE;READ(20,*) BETA5
READ(20,3)TITLE;READ(20,*) BETA6
READ(20,3)TITLE;READ(20,*) POWER1
READ(20,3)TITLE;READ(20,*) FUELM
READ(20,3)TITLE;READ(20,*) CPFUEL
READ(20,3)TITLE;READ(20,*) HFC1
READ(20,3)TITLE;READ(20,*) AREAFC
READ(20,3)TITLE;READ(20,*) WP
READ(20,3)TITLE;READ(20,*) PLRM
READ(20,3)TITLE;READ(20,*) CMC1
READ(20,3)TITLE;READ(20,*) CP1
READ(20,3)TITLE;READ(20,*) CMC2
READ(20,3)TITLE;READ(20,*) UPM
READ(20,3)TITLE;READ(20,*) POWER
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+1)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+2)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+3)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+4)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+5)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+6);TFUEL1=X2(M,ICOUNT+6)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+7)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+8);TC11=X2(M,ICOUNT+8)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+9)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+10)
READ(20,3)TITLE;READ(20,*) PRESS1;PRESS=PRESS1
READ(20,3)TITLE;READ(20,*) TCLEG1
CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE FORCING TERM.
C*****

```

```

      TCLEG=DTCLEG+TCLEG1
      ROH=ROH1+DROH
C*****
C* EXCHANGE THE VARIABLES.
C*****
4016  PREC1=X2(M,ICOUNT)
      PREC2=X2(M,ICOUNT+1)
      PREC3=X2(M,ICOUNT+2)
      PREC4=X2(M,ICOUNT+3)
      PREC5=X2(M,ICOUNT+4)
      PREC6=X2(M,ICOUNT+5)
      TFUEL=X2(M,ICOUNT+6)
      TPLR=X2(M,ICOUNT+7)
      TC1=X2(M,ICOUNT+8)
      TC2=X2(M,ICOUNT+9)
      TUP=X2(M,ICOUNT+10)
      IPRIME=ICOUNT+10
      IK=IJ+10
      IJKA=IJA+1
      IF(JJ1.EQ.0)GOTO 4066
C*****
C* MODEL'S GOVERNING EQUATIONS ARE LISTED BELOW.
C*****
C
C
C REACTIVITY TERM
      REACT=ROH+ALFAP*(TFUEL-TFUEL1)+ALFAC*(TC1-TC11)+ALFAPR*(PRESS-
* PRESS1)
C POWER EQUATION
      POWER=(REACT+OMEGA*(DECAY1*PREC1+DECAY2*PREC2+DECAY3*PREC3+
* DECAY3*PREC3+DECAY4*PREC4+DECAY5*PREC5+DECAY6*PREC6))/(BTOTAL-
* REACT)
C DELAYED NEUTRONS EQUATION (GROUP ONE)
      F(M,ICOUNT)=BETA1*POWER/OMEGA-DECAY1*PREC1
C DELAYED NEUTRONS EQUATION (GROUP TWO)
      F(M,ICOUNT+1)=BETA2*POWER/OMEGA-DECAY2*PREC2
C DELAYED NEUTRONS EQUATION (GROUP THREE)
      F(M,ICOUNT+2)=BETA3*POWER/OMEGA-DECAY3*PREC3
C DELAYED NEUTRONS EQUATION (GROUP FOUR)
      F(M,ICOUNT+3)=BETA4*POWER/OMEGA-DECAY4*PREC4
C DELAYED NEUTRONS EQUATION (GROUP FIVE)
      F(M,ICOUNT+4)=BETA5*POWER/OMEGA-DECAY5*PREC5
C DELAYED NEUTRONS EQUATION (GROUP SIX)
      F(M,ICOUNT+5)=BETA6*POWER/OMEGA-DECAY6*PREC6
C FUEL TEMPERATURE EQUATION
      F(M,ICOUNT+6)=((1+POWER)*POWER1-HFC1*AREAFC*(TFUEL-TC1))/
* (FUELM*CPFUEL)
C COOLANT TEMPERATURE IN THE LOWER PLENUM LUMP
      F(M,ICOUNT+7)=WP*(TCLEG-TPLR)/PLRM
C COOLANT TEMPERATURE IN THE FIRST COOLANT LUMP
      F(M,ICOUNT+8)=WP*(TPLR-TC1)/CMC1+HFC1*AREAFC*(TFUEL-TC1)/(2*
* CMC1*CP1)
C COOLANT TEMPERATURE IN THE SECOND COOLANT LUMP
      F(M,ICOUNT+9)=WP*(TC1-TC2)/CMC2+HFC1*AREAFC*(TFUEL-TC1)/(2*
* CMC2*CP1)
C COOLANT TEMPERATURE IN THE CORE UPPER PLENUM LUMP
      F(M,ICOUNT+10)=(WP*(TC2-TUP))/UPM
      IF(IFINE.EQ.0) WRITE(23,*) REACT,STIME
      IF(IFINE.EQ.0) WRITE(23,*) POWER,STIME
4066  CONTINUE

```

RESPONSE OF THE REACTIVITY IN THE REACTOR IS PLACED IN REACT.DAT
 REACT.DAT
 RESPONSE OF THE REACTOR POWER IS PLACED IN POWER.DAT
 POWER.DAT
 RESPONSE OF THE 1ST PRECURSOR GROUP IS PLACED IN PREC1.DAT
 PRE1.DAT
 RESPONSE OF THE 2ND PRECURSOR GROUP IS PLACED IN PREC2.DAT
 PRE2.DAT
 RESPONSE OF THE 3RD PRECURSOR GROUP IS PLACED IN PREC3.DAT
 PRE3.DAT
 RESPONSE OF THE 4TH PRECURSOR GROUP IS PLACED IN PREC4.DAT
 PRE4.DAT
 RESPONSE OF THE 5TH PRECURSOR GROUP IS PLACED IN PREC5.DAT
 PRE5.DAT
 RESPONSE OF THE 6TH PRECURSOR GROUP IS PLACED IN PREC6.DAT
 PRE6.DAT
 RESPONSE OF THE FUEL TEMPERATURE IS PLACED IN FUEL.DAT
 FUEL.DAT
 RESPONSE OF THE LOWER PLENUM TEMPERATURE IS PLACED IN LOWER.DAT
 LOWER.DAT
 RESPONSE OF THE 1ST COOLANT LUMP TEMPERATURE IS PLACED IN 1ST.DAT
 1ST.DAT
 RESPONSE OF THE 2ND COOLANT LUMP TEMPERATURE IS PLACED IN 2ND.DAT
 2ND.DAT
 RESPONSE OF THE UPPER PLENUM TEMPERATURE IS PLACED IN 2ND.DAT
 UPPER.DAT
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR REACT. VS TIME
 REACTIVITY IN THE REACTOR.
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR POWER VS TIME
 NORMALIZED POWER
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 1ST PREC. GROUP
 1ST PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 2ND PREC. GROUP
 2ND PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 3RD PREC.GROUP
 3RD PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 4TH PREC. GROUP
 4TH PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 5TH PREC. GROUP
 5TH PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 6TH PREC. GROUP
 6TH PRECURSOR GROUP CONCENTRATION
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR THE FUEL TEMP.
 FUEL LUMP TEMPERATUE(F)
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR LOWER PLEN.TEMP.
 LOWER PLENUM TEMPERATURE(F)
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 1ST COOLANT LUMP
 1ST COOLANT LUMP TEMPERATURE(F)
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR 2ND COOLANT LUMP
 2ND COOLANT LUMP TEMPERATURE(F)
 NEXT LINE IS THE Y TILTE AS IT APPEAR ON THE PLOT FOR UPPER PLEN.TEMP.
 FIRST COOLANT LUMP TEMPERATURE(F)
 EXTERNAL REACTIVITY AT STEADY STATE(\$)
 0.00000000
 NEUTRON GENERATION TIME(SEG)
 1.790000E-05
 FUEL REACTIVITY COEFFICIENT(/F)
 -1.100000E-05
 COOLANT REACTIVITY COEFFICIENT(/F)
 -2.000000E-04
 PRESSURE REACTIVITY COEFFICIENT(/F)

-3.000000E-07
TOTAL DELAYED NEUTRON FRACTION
6.450000E-03
DECAY CONSTANT FOR GROUP 1
1.240000E-02
DECAY CONSTANT FOR GROUP 2
0.3050000
DECAY CONSTANT FOR GROUP 3
0.1110000
DECAY CONSTANT FOR GROUP 4
0.3010000
DECAY CONSTANT FOR GROUP 5
1.100000
DECAY CONSTANT FOR GROUP 6
3.000000
1ST GROUP DELAYED NEUTRON FRACTION
2.100000E-04
2ND GROUP DELAYED NEUTRON FRACTION
1.410000E-03
3RD GROUP DELAYED NEUTRON FRACTION
1.270000E-03
4TH GROUP DELAYED NEUTRON FRACTION
2.550000E-03
5TH GROUP DELAYED NEUTRON FRACTION
7.400000E-04
6TH GROUP DELAYED NEUTRON FRACTION
2.700000E-04
STEADY STATE REACTOR POWER AT 100%(BTU/SEC)
2.0851722E+06
FUEL MASS(IBM)
212500.0
FUEL SPECIFIC HEAT(BTU/IBM-F)
5.900000E-02
OVERALL HEAT TRANSFER COFF. BETWEEN FUEL AND COOLANT(BTU/FT2-F-SEC)
5.55555E-02
OVERALL HEAT TRANSFER AREA BETWEEN FUEL AND COOLANT(FT2)
42460.00
COOLANT FLOW RATE INSIDE THE CORE(IBM/SEC)
28194.44
MASS OF THE COOLANT INSIDE THE CORE INLET PLENUM(IBM)
54640.00
MASS OF THE COOLANT INSIDE THE FIRST COOLANT LUMP(IBM)
133805.0
SPECIFIC HEAT OF THE COOLANT(BTU/IBM-F)
1.323000
MASS OF THE COOLANT INSIDE THE SECOND COOLANT LUMP(IBM)
133805.0
MASS OF THE COOLANT INSIDE THE CORE UPPER PLENUM(IBM)
83800.00
DEVIATION OF THE REACTOR POWER FROM STEADY STATE AT STEADY STATE
0.0000000000000000
DEVIATION OF THE 1ST PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 2ND PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 3RD PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 4TH PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 5TH PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000

DEVIATION OF THE 6TH PRECURSOR GROUP FROM S.S AT S.S
0.0000000000000000
STEADY STATE TEMPERATURE IN THE FUEL(F)
1458.1141640000000
STEADY STATE TEMPERATURE IN THE LOWER PLENUM(F)
546.200000000000000
STEADY STATE TEMPERATURE IN THE 1ST COOLANT SECTION(F)
574.15043360000000
STEADY STATE TEMPERATURE IN THE 2ND COOLANT SECTION(F)
602.10086720000000
STEADY STATE TEMPERATURE IN THE CORE UPPER PLENUM(F)
602.10086720000000
STEADY STATE PRESSURE IN THE PRIMARY SIDE(Psi)
2250.0000000000000
STEADY STATE TEMPERATURE IN THE COLDLEG(F)
546.20000000000000

```

C*****
C* PCOR17.FOR: A NONLINERAR LUMPED PARAMETER MODEL FOR A TYPICAL *
C* PRESSURIZED WATER REACTOR CORE.SEVENTEEN LUMPS ARE *
C* USED TO REPRESENT THE CORE.FIVE LUMPS FOR THE FUEL, *
C* TEN LUMPS FOR THE COOLANT,ONE LUMP FOR THE LOWER PLE- *
C* NUM AND ONE LUMP FOR THE UPPER PLENUM.THE PROMPT JUMP *
C* IS USED FOR THE POWER EQUATION. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IJA=IJKA+1
      IF(ISTEP.NE.0)GOTO 4016
      DO 4000 IY=1,IPER
      IF(CHOICE(IY).EQ.'PCOR17.FOR') GOTO 4001
      WRITE(5,*)
4000  CONTINUE
      GOTO 4011
C*****
C* ENTER THE FORCING TERM. *
C*****
4001  DO 4002 ICLEAR=1,24
      WRITE(5,*)
4002  CONTINUE
      WRITE(5,4003);WRITE(5,4004);WRITE(5,4005);WRITE(5,4006)
      WRITE(5,4003);WRITE(5,4007);READ(5,*) ICASE(IAM)
4003  FORMAT(1X,'*****')
4004  FORMAT(1X,'* CHOOSE A CASE BY ENTERING THE # TO THE LEFT. *')
4005  FORMAT(1X,'* 1- ISOLATED CORE CASE. *')
4006  FORMAT(1X,'* 2- CORE COUPLED TO A STEAM GENERATOR. *')
4007  FORMAT(1X,'CASE=')$
      WRITE(5,4003);WRITE(5,4008);READ(5,*) DROH
4008  FORMAT(1X,'*ENTER DEVIATION IN EXTERNAL REACTIVITY($),DROH=$')
      IF(ICASE(IAM).EQ.2) GOTO 4010
      WRITE(5,4009);READ(5,*) DTCLEG
4009  FORMAT(1X,'*ENTER DEVIATION IN THE COLDLEG TEMPP.(F),DTCLED=$')
      WRITE(5,4003)
4010  IAM=IAM+1
C*****
C* OPEN A FILE AND READ MODEL'S PARAMERTERS. *
C*****
4011  DO 4012 ICLEAR=1,24
      WRITE(5,*)
4012  CONTINUE
      IF(IREFF.EQ.1) GOTO 4014
      WRITE(5,4003);WRITE(5,4013);WRITE(5,4003);READ(5,3)HEAD1
4013  FORMAT(1X,'* ENTER A FILE NAME FOR THE CORE DATA,FILENAME=$')
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 4015
4014  HEAD1='CORE17.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
4015  READ(20,3)TITLE;READ(20,3) AFR1(IJA)
      READ(20,3)TITLE;READ(20,3) FR1(IJ)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+1)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+2)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+3)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+4)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+5)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+6)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+7)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+8)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+9)

```

```
READ(20,3)TITLE;READ(20,3) FR1(IJ+10)
READ(20,3)TITLE;READ(20,3) FR1(IJ+11)
READ(20,3)TITLE;READ(20,3) FR1(IJ+12)
READ(20,3)TITLE;READ(20,3) FR1(IJ+13)
READ(20,3)TITLE;READ(20,3) FR1(IJ+14)
READ(20,3)TITLE;READ(20,3) FR1(IJ+15)
READ(20,3)TITLE;READ(20,3) FR1(IJ+16)
READ(20,3)TITLE;READ(20,3) FR1(IJ+17)
READ(20,3)TITLE;READ(20,3) FR1(IJ+18)
READ(20,3)TITLE;READ(20,3) FR1(IJ+19)
READ(20,3)TITLE;READ(20,3) FR1(IJ+20)
READ(20,3)TITLE;READ(20,3) FR1(IJ+21)
READ(20,3)TITLE;READ(20,3) FR1(IJ+22)
READ(20,3)TITLE;READ(20,*) ROH1
READ(20,3)TITLE;READ(20,*) OMEGA
READ(20,3)TITLE;READ(20,*) ALFAF
READ(20,3)TITLE;READ(20,*) ALFAC
READ(20,3)TITLE;READ(20,*) ALFAPR
READ(20,3)TITLE;READ(20,*) BTOTAL
READ(20,3)TITLE;READ(20,*) DECAY1
READ(20,3)TITLE;READ(20,*) DECAY2
READ(20,3)TITLE;READ(20,*) DECAY3
READ(20,3)TITLE;READ(20,*) DECAY4
READ(20,3)TITLE;READ(20,*) DECAY5
READ(20,3)TITLE;READ(20,*) DECAY6
READ(20,3)TITLE;READ(20,*) BETA1
READ(20,3)TITLE;READ(20,*) BETA2
READ(20,3)TITLE;READ(20,*) BETA3
READ(20,3)TITLE;READ(20,*) BETA4
READ(20,3)TITLE;READ(20,*) BETA5
READ(20,3)TITLE;READ(20,*) BETA6
READ(20,3)TITLE;READ(20,*) FI1
READ(20,3)TITLE;READ(20,*) FI2
READ(20,3)TITLE;READ(20,*) FI3
READ(20,3)TITLE;READ(20,*) FI4
READ(20,3)TITLE;READ(20,*) FI5
READ(20,3)TITLE;READ(20,*) CI1
READ(20,3)TITLE;READ(20,*) CI2
READ(20,3)TITLE;READ(20,*) CI3
READ(20,3)TITLE;READ(20,*) CI4
READ(20,3)TITLE;READ(20,*) CI5
READ(20,3)TITLE;READ(20,*) CI6
READ(20,3)TITLE;READ(20,*) CI7
READ(20,3)TITLE;READ(20,*) CI8
READ(20,3)TITLE;READ(20,*) CI9
READ(20,3)TITLE;READ(20,*) CI10
READ(20,3)TITLE;READ(20,*) POWER1
READ(20,3)TITLE;READ(20,*) FM1
READ(20,3)TITLE;READ(20,*) FM2
READ(20,3)TITLE;READ(20,*) FM3
READ(20,3)TITLE;READ(20,*) FM4
READ(20,3)TITLE;READ(20,*) FM5
READ(20,3)TITLE;READ(20,*) CPFUEL
READ(20,3)TITLE;READ(20,*) HFC
READ(20,3)TITLE;READ(20,*) AREAFC
READ(20,3)TITLE;READ(20,*) WP
READ(20,3)TITLE;READ(20,*) PLRM
READ(20,3)TITLE;READ(20,*) CMC1
READ(20,3)TITLE;READ(20,*) CMC2
READ(20,3)TITLE;READ(20,*) CMC3
READ(20,3)TITLE;READ(20,*) CMC4
```

```

READ(20,3)TITLE;READ(20,*) CMC5
READ(20,3)TITLE;READ(20,*) CMC6
READ(20,3)TITLE;READ(20,*) CMC7
READ(20,3)TITLE;READ(20,*) CMC8
READ(20,3)TITLE;READ(20,*) CMC9
READ(20,3)TITLE;READ(20,*) CMC10
READ(20,3)TITLE;READ(20,*) CP1
READ(20,3)TITLE;READ(20,*) UPM
READ(20,3)TITLE;READ(20,*) POWER
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+1)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+2)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+3)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+4)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+5)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+6);TF10=X2(M,ICOUNT+6)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+7);TF20=X2(M,ICOUNT+7)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+8);TF30=X2(M,ICOUNT+8)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+9);TF40=X2(M,ICOUNT+9)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+10);TF50=X2(M,ICOUNT+10)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+11)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+12);TC10=X2(M,ICOUNT+12)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+13);TC20=X2(M,ICOUNT+13)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+14);TC30=X2(M,ICOUNT+14)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+15);TC40=X2(M,ICOUNT+15)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+16);TC50=X2(M,ICOUNT+16)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+17);TC60=X2(M,ICOUNT+17)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+18);TC70=X2(M,ICOUNT+18)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+19);TC80=X2(M,ICOUNT+19)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+20);TC90=X2(M,ICOUNT+20)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+21);T100=X2(M,ICOUNT+21)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+22)
READ(20,3)TITLE;READ(20,*) PRESS1;PRESS=PRESS1
READ(20,3)TITLE;READ(20,*) TCLEG1
CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE FORCING TERM. *
C*****
      TCLEG=DTCLLEG+TCLEG1
      ROH=ROH1+DROH
C*****
C* EXCHANGE THE VARIABLES. *
C*****
4016  PREC1=X2(M,ICOUNT)
      PREC2=X2(M,ICOUNT+1)
      PREC3=X2(M,ICOUNT+2)
      PREC4=X2(M,ICOUNT+3)
      PREC5=X2(M,ICOUNT+4)
      PREC6=X2(M,ICOUNT+5)
      TF1=X2(M,ICOUNT+6)
      TF2=X2(M,ICOUNT+7)
      TF3=X2(M,ICOUNT+8)
      TF4=X2(M,ICOUNT+9)
      TF5=X2(M,ICOUNT+10)
      TLP=X2(M,ICOUNT+11)
      TC1=X2(M,ICOUNT+12)
      TC2=X2(M,ICOUNT+13)
      TC3=X2(M,ICOUNT+14)
      TC4=X2(M,ICOUNT+15)
      TC5=X2(M,ICOUNT+16)
      TC6=X2(M,ICOUNT+17)

```

```

TC7=X2(M,ICOUNT+18)
TC8=X2(M,ICOUNT+19)
TC9=X2(M,ICOUNT+20)
T10=X2(M,ICOUNT+21)
TUP=X2(M,ICOUNT+22)
IPRIME=ICOUNT+22
IK=IJ+22
IJK=IJA
IF(JJ1.EQ.0)GOTO 4066
C*****
C* MODEL'S GOVERNING EQUATIONS ARE LISTED BELOW.
C*****
C REACTIVITY TERM
  REACT=ROH+ALFAF*(FI1*(TF1-TF10)+FI2*(TF2-TF20)+FI3*(TF3-TF30)+
  * IF4*(TF4-TF40)+IF5*(TF5-TF50))+ALFAC*(CI1*(TC1-TC10)
  * +CI2*(TC2-TC20)+CI3*(TC3-TC30)+CI4*(TC4-TC40)+CI5*(TC5-TC50)
  * +CI6*(TC6-TC60)+CI7*(TC7-TC70)+CI8*(TC8-TC80)+CI9*(TC9-TC90)
  * +CI10*(T10-T100))+ALPAPR*(PRESS-PRESS1)
C POWER EQUATION
  POWER=(REACT+OMEGA*(DECAY1*PREC1
  * +DECAY2*PREC2+DECAY3*PREC3+DECAY4*PREC4+DECAY5*PREC5
  * +DECAY6*PREC6))/(BTOTAL-REACT)
C DELAYED NEUTRONS EQUATION (GROUP ONE)
  F(M,ICOUNT)=BETA1*POWER/OMEGA-DECAY1*PREC1
C DELAYED NEUTRONS EQUATION (GROUP TWO)
  F(M,ICOUNT+1)=BETA2*POWER/OMEGA-DECAY2*PREC2
C DELAYED NEUTRONS EQUATION (GROUP THREE)
  F(M,ICOUNT+2)=BETA3*POWER/OMEGA-DECAY3*PREC3
C DELAYED NEUTRONS EQUATION (GROUP FOUR)
  F(M,ICOUNT+3)=BETA4*POWER/OMEGA-DECAY4*PREC4
C DELAYED NEUTRONS EQUATION (GROUP FIVE)
  F(M,ICOUNT+4)=BETA5*POWER/OMEGA-DECAY5*PREC5
C DELAYED NEUTRONS EQUATION (GROUP SIX)
  F(M,ICOUNT+5)=BETA6*POWER/OMEGA-DECAY6*PREC6
C FUEL TEMPERATURE EQUATION (FIRST FUEL LUMP)
  F(M,ICOUNT+6)=(FI1*(POWER1+POWER1*POWER)-HFC*AREAFC*(TF1-TC1)
  * )/(FM1*CPFUEL)
C FUEL TEMPERATURE EQUATION (SECOND FUEL LUMP)
  F(M,ICOUNT+7)=(FI2*(POWER1+POWER1*POWER)-HFC*AREAFC*(TF2-TC3)
  * )/(FM2*CPFUEL)
C FUEL TEMPERATURE EQUATION (THIRD FUEL LUMP)
  F(M,ICOUNT+8)=(FI3*(POWER1+POWER1*POWER)-HFC*AREAFC*(TF3-TC5)
  * )/(FM3*CPFUEL)
C FUEL TEMPERATURE EQUATION (FORTH FUEL LUMP)
  F(M,ICOUNT+9)=(FI4*(POWER1+POWER1*POWER)-HFC*AREAFC*(TF4-TC7)
  * )/(FM4*CPFUEL)
C FUEL TEMPERATURE EQUATION (FIFTH FUEL LUMP)
  F(M,ICOUNT+10)=(FI5*(POWER1+POWER1*POWER)-HFC*AREAFC*(TF5-TC9)
  * )/(FM5*CPFUEL)
C COOLANT TEMPERATURE IN THE LOWER PLENUM LUMP
  F(M,ICOUNT+11)=WP*(TCLEG-TLP)/PLRM
C COOLANT TEMPERATURE IN THE FIRST COOLANT LUMP
  F(M,ICOUNT+12)=WP*(TLP-TC1)/CMC1+HFC*AREAFC*(TF1-TC1)/(2*CMC1
  * *CP1)
C COOLANT TEMPERATURE IN THE SECOND COOLANT LUMP
  F(M,ICOUNT+13)=(WP*(TC1-TC2)+HFC*AREAFC*(TF1-TC1)/(2*CP1))/CMC2
C COOLANT TEMPERATURE IN THE THIRD COOLANT LUMP
  F(M,ICOUNT+14)=(WP*(TC2-TC3)+HFC*AREAFC*(TF2-TC3)/(2*CP1))/CMC3
C COOLANT TEMPERATURE IN THE FORTH COOLANT LUMP
  F(M,ICOUNT+15)=(WP*(TC3-TC4)+HFC*AREAFC*(TF2-TC3)/(2*CP1))/CMC4
C COOLANT TEMPERATURE IN THE FIFTH COOLANT LUMP

```

```

      F(M,ICOUNT+16)=(WP*(TC4-TC5)+HFC*AREAFC*(TF3-TC5)/(2*CP1))/CNC5
C COOLANT TEMPERATURE IN THE SIXTH COOLANT LUMP
      F(M,ICOUNT+17)=(WP*(TC5-TC6)+HFC*AREAFC*(TF3-TC5)/(2*CP1))/CNC6
C COOLANT TEMPERATURE IN THE SEVENTH COOLANT LUMP
      F(M,ICOUNT+18)=(WP*(TC6-TC7)+HFC*AREAFC*(TF4-TC7)/(2*CP1))/CNC7
C COOLANT TEMPERATUER IN THE EIGHTH COOLANT LUMP
      F(M,ICOUNT+19)=(WP*(TC7-TC8)+HFC*AREAFC*(TF4-TC7)/(2*CP1))/CNC8
C COOLANT TEMPERATURE IN THE NINTH COOLANT LUMP
      F(M,ICOUNT+20)=(WP*(TC8-TC9)+HFC*AREAFC*(TF5-TC9)/(2*CP1))/CNC9
C COOLANT TEMPERATURE IN THE TENTH COOLANT LUMP
      F(M,ICOUNT+21)=(WP*(TC9-T10)+HFC*AREAFC*(TF5-TC9)/(2*CP1))/CNC10
C COOLANT TEMPERATURE IN THE CORE UPPER PLENUM LUMP
      F(M,ICOUNT+22)=(WP*(T10-TUP))/UPM
      IF(IFINE.EQ.0) WRITE(23,*)POWER,STIME
4066 CONTINUE

```

RESPONSE OF THE REACTOR POWER IS PLACED IN POWER.DAT
POWER.DAT
RESPONSE OF THE 1ST PRECURSOR GROUP IS PLACED IN PRE1.DAT
PRE1.DAT
RESPONSE OF THE 2ND PRECURSOR GROUP IS PLACED IN PRE2.DAT
PRE2.DAT
RESPONSE OF THE 3RD PRECURSOR GROUP IS PLACED IN PRE3.DAT
PRE3.DAT
RESPONSE OF THE 4TH PRECURSOR GROUP IS PLACED IN PRE4.DAT
PRE4.DAT
RESPONSE OF THE 5TH PRECURSOR GROUP IS PLACED IN PRE5.DAT
PRE5.DAT
RESPONSE OF THE 6TH PRECURSOR GROUP IS PLACED IN PRE6.DAT
PRE6.DAT
RESPONSE OF THE 1ST FUEL LUMP TEMPERATURE IS PLACED IN FUEL1.DAT
FUEL1.DAT
RESPONSE OF THE 2ND FUEL LUMP TEMPERATURE IS PLACED IN FUEL2.DAT
FUEL2.DAT
RESPONSE OF THE 3RD FUEL LUMP TEMPERATURE IS PLACED IN FUEL3.DAT
FUEL3.DAT
RESPONSE OF THE 4TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL4.DAT
FUEL4.DAT
RESPONSE OF THE 5TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL5.DAT
FUEL5.DAT
RESPONSE OF THE LOWER PLENUM TEMPERATURE IS PLACED IN LOWER.DAT
LOWER.DAT
RESPONSE OF THE 1ST COOLANT LUMP TEMPERATURE IS PLACED IN 1ST.DAT
1ST.DAT
RESPONSE OF THE 2ND COOLANT LUMP TEMPERATURE IS PLACED IN 2ND.DAT
2ND.DAT
RESPONSE OF THE 3RD COOLANT LUMP TEMPERATURE IS PLACED IN 3RD.DAT
3RD.DAT
RESPONSE OF THE 4TH COOLANT LUMP TEMPERATURE IS PLACED IN 4TH.DAT
4TH.DAT
RESPONSE OF THE 5TH COOLANT LUMP TEMPERATURE IS PLACED IN 5TH.DAT
5TH.DAT
RESPONSE OF THE 6TH COOLANT LUMP TEMPERATURE IS PLACED IN 6TH.DAT
6TH.DAT
RESPONSE OF THE 7TH COOLANT LUMP TEMPERATURE IS PLACED IN 7TH.DAT
7TH.DAT
RESPONSE OF THE 8TH COOLANT LUMP TEMPERATURE IS PLACED IN 8TH.DAT
8TH.DAT
RESPONSE OF THE 9TH COOLANT LUMP TEMPERATURE IS PLACED IN 9TH.DAT
9TH.DAT
RESPONSE OF THE 10TH COOLANT LUMP TEMPERATURE IS PLACED IN 10TH.DAT
10TH.DAT
RESPONSE OF THE UPPER PLENUM TEMPERATURE IS PLACED IN UPPER.DAT
UPPER.DAT
EXTERNAL REACTIVITY AT STEADY STATE(\$)
0.0000000
NEUTRON GENERATION TIME(SEC)
1.790000E-05
FUEL REACTIVITY COEFFICIENT(/F)
-1.100000E-05
COOLANT REACTIVITY COEFFICIENT(/F)
-2.000000E-04
PRESSURE REACTIVITY COEFFICIENT(/F)
3.000000E-03
TOTAL DELAYED NEUTRONS FRACTION
6.450000E-03
DECAY CONSTANT FOR GROUP 1

1.240000E-02
DECAY CONSTANT FOR GROUP 2
0.3050000
DECAY CONSTANT FOR GROUP 3
0.1110000
DECAY CONSTANT FOR GROUP 4
0.3010000
DECAY CONSTANT FOR GROUP 5
1.100000
DECAY CONSTANT FOR GROUP 6
3.000000
1ST GROUP DELAYED NEUTRONS FRACTION
2.100000E-04
2ND GROUP DELAYED NEUTRONS FRACTION
1.410000E-03
3RD GROUP DELAYED NEUTRONS FRACTION
1.270000E-03
4TH GROUP DELAYED NEUTRONS FRACTION
2.550000E-03
5TH GROUP DELAYED NEUTRONS FRACTION
7.400000E-04
6TH GROUP DELAYED NEUTRONS FRACTION
2.700000E-04
IMPORTANCE OF THE REACTIVITY INTRODUCED IN 1ST FUEL LUMP
9.549000E-02
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 2ND FUEL LUMP
2.500000E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 3RD FUEL LUMP
3.090000E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 4TH FUEL LUMP
2.500000E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 5TH FUEL LUMP
9.549000E-02
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 1ST COOLANT LUMP
2.447000E-02
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 2ND COOLANT LUMP
7.102000E-02
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 3RD COOLANT LUMP
1.106100E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 4TH COOLANT LUMP
1.393800E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 5TH COOLANT LUMP
1.545000E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 6TH COOLANT LUMP
1.545000E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 7TH COOLANT LUMP
1.393800E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 8TH COOLANT LUMP
1.106100E-01
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 9TH COOLANT LUMP
7.102000E-02
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 10TH COOLANT LUMP
2.447000E-02
STEADY STATE REACTOR POWER AT 100%(BTU/SEC)
2.0851722E+06
MASS OF THE FUEL IN THE FIRST FUEL LUMP(IBM)
42500
MASS OF THE FUEL IN THE SECOND FUEL LUMP(IBM)
42500
MASS OF THE FUEL IN THE THIRD FUEL LUMP(IBM)
42500

MASS OF THE FUEL IN THE FOURTH FUEL LUMP(IBM)
42500
MASS OF THE FUEL IN THE FIFTH FUEL LUMP(IBM)
42500
FUEL SPECIFIC HEAT(BTU/IBM-F)
5.90000E-02
OVERALL HEAT TRANSFER COEFF. BETWEEN FUEL AND COOLANT(BTU/FT2-F-SEC)
5.55555E-02
OVERALL HEAT TRANSFER AREA PER FUEL LUMP BETWEEN FUEL AND COOLANT(FT2)
8492.00
COOLANT FLOW RATE(IBM/SEC)
28194.44
MASS OF THE COOLANT INSIDE THE REACTOR LOWER PLENUM(IBM)
54640
MASS OF THE COOLANT INSIDE THE 1ST COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 2ND COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 3RD COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 4TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 5TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 6TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 7TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 8TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 9TH COOLANT LUMP(IBM)
26761
MASS OF THE COOLANT INSIDE THE 10TH COOLANT LUMP(IBM)
26761
SPECIFIC HEAT FOR THE COOLANT(BTU/IBM-F)
1.323
MASS OF THE COOLANT IN THE REACTOR UPPER PLENUM(IBM)
83800
DEVIATION OF THE REACTOR POWER FROM STEADY STATE AT STEADY STATE
0.0000000000000000
DEVIATION OF THE 1ST PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 2ND PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 3RD PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 4TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 5TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
DEVIATION OF THE 6TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
0.0000000000000000
STEADY STATE TEMPERATURE IN THE FIRST FUEL LUMP(F)
970.917470200000000
STEADY STATE TEMPERATURE IN THE SECOND FUEL LUMP(F)
1663.480246000000000
STEADY STATE TEMPERATURE IN THE THIRD FUEL LUMP(F)
1939.873841000000000
STEADY STATE TEMPERATURE IN THE FOURTH FUEL LUMP(F)
1694.728835000000000
STEADY STATE TEMPERATURE IN THE FIFTH FUEL LUMP(F)

1021.47925200000000
STEADY STATE TEMPERATURE IN THE CORE LOWER PLENUM
546.20000000000000
STEADY STATE TEMPERATURE IN THE FIRST COOLANT LUMP(F)
548.86898730000000
STEADY STATE TEMPERATURE IN THE SECOND COOLANT LUMP(F)
551.53797460000000
STEADY STATE TEMPERATURE IN THE THIRD COOLANT LUMP(F)
558.52558400000000
STEADY STATE TEMPERATURE IN THE FOURTH COOLANT LUMP(F)
565.51319340000000
STEADY STATE TEMPERATURE IN THE FIFTH COOLANT LUMP(F)
574.14987860000000
STEADY STATE TEMPERATURE IN THE SIXTH COOLANT LUMP(F)
582.78656380000000
STEADY STATE TEMPERATURE IN THE SEVENTH COOLANT LUMP(F)
589.77417280000000
STEADY STATE TEMPERATURE IN THE EIGHTH COOLANT LUMP(F)
596.76178180000000
STEADY STATE TEMPERATURE IN THE NINTH COOLANT LUMP(F)
599.43076910000000
STEADY STATE TEMPERATURE IN THE TENTH COOLANT LUMP(F)
602.10086720000000
STEADY STATE TEMPERATURE IN THE REACTOR UPPER PLENUM(F)
602.10086720000000
STEADY STATE PRESSURE IN THE PRIMARY SIDE(P51)
2250.000000000000
STEADY STATE TEMPERATURE IN THE COLD LEG(F)
546.20000000000000

```

C*****
C* PCOR47.FOR: A NONLINEAR LUMPED PARAMETER MODEL FOR A TYPICAL *
C* PRESSURIZED WATER REACTOR CORE.FORTY SEVEN LUMPS ARE *
C* USED TO REPRESENT THE CORE.FIFTEEN LUMPS FOR THE FUEL,*
C* THIRTY LUMPS FOR THE COOLANT,ONE LUMP FOR THE LOWER *
C* PLENUM AND ONE LUMP FOR THE UPPER PLENUM.THE PROMPT *
C* JUMP ASSUMPTION IS USED FOR THE POWER EQUATION. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IJA=IJKA+1
      IF(ISTEP.NE.0)GOTO 4016
      DO 4000 IY=1,IPER
      IF(CHOICE(IY).EQ.'PCOR47.FOR') GOTO 4001
      WRITE(5,*)
4000  CONTINUE
      GOTO 4011
C*****
C* ENTER THE FORCING TERM. *
C*****
4001  DO 4002 ICLEAR=1,24
      WRITE(5,*)
4002  CONTINUE
      WRITE(5,4003);WRITE(5,4004);WRITE(5,4005);WRITE(5,4006)
      WRITE(5,4003);WRITE(5,4007);READ(5,*) ICASE(IAM)
4003  FORMAT(1X,'*****')
4004  FORMAT(1X,'* CHOOSE A CASE BY ENTERING THE # TO THE LEFT. *')
4005  FORMAT(1X,'*      1- ISOLATED CORE CASE. *')
4006  FORMAT(1X,'*      2- CORE COUPLED TO A STEAM GENERATOR. *')
4007  FORMAT(1X,'CASE='$)
      WRITE(5,4003);WRITE(5,4008);READ(5,*) DROH
4008  FORMAT(1X,'*ENTER DEVIATION IN EXTERNAL REACTIVITY($),DROH='$)
      IF(ICASE(IAM).EQ.2) GOTO 4010
      WRITE(5,4009);READ(5,*) DTCLEG
4009  FORMAT(1X,'*ENTER DEVIATION IN THE COLDLEG TEMPP.(F),DTCLED='$)
      WRITE(5,4003)
4010  IAM=IAM+1
C*****
C* OPEN A FILE AND READ MODEL'S PARAMERTERS. *
C*****
4011  DO 4012 ICLEAR=1,24
      WRITE(5,*)
4012  CONTINUE
      IF(IREFF.EQ.1) GOTO 4014
      WRITE(5,4003);WRITE(5,4013);WRITE(5,4003);READ(5,3)HEAD1
4013  FORMAT(1X,'* ENTER A FILE NAME FOR THE CORE DATA,FILENAME='$)
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 4015
4014  HEAD1='CORE47.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
4015  READ(20,3)TITLE;READ(20,3) AFR1(IJA)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+1)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+2)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+3)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+4)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+5)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+6)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+7)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+8)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+9)
      READ(20,3)TITLE;READ(20,3) FR1(IJ+10)

```

```

READ(20,3)TITLE;READ(20,3) FR1(IJ+11)
READ(20,3)TITLE;READ(20,3) FR1(IJ+12)
READ(20,3)TITLE;READ(20,3) FR1(IJ+13)
READ(20,3)TITLE;READ(20,3) FR1(IJ+14)
READ(20,3)TITLE;READ(20,3) FR1(IJ+15)
READ(20,3)TITLE;READ(20,3) FR1(IJ+16)
READ(20,3)TITLE;READ(20,3) FR1(IJ+17)
READ(20,3)TITLE;READ(20,3) FR1(IJ+18)
READ(20,3)TITLE;READ(20,3) FR1(IJ+19)
READ(20,3)TITLE;READ(20,3) FR1(IJ+20)
READ(20,3)TITLE;READ(20,3) FR1(IJ+21)
READ(20,3)TITLE;READ(20,3) FR1(IJ+22)
READ(20,3)TITLE;READ(20,3) FR1(IJ+23)
READ(20,3)TITLE;READ(20,3) FR1(IJ+24)
READ(20,3)TITLE;READ(20,3) FR1(IJ+25)
READ(20,3)TITLE;READ(20,3) FR1(IJ+26)
READ(20,3)TITLE;READ(20,3) FR1(IJ+27)
READ(20,3)TITLE;READ(20,3) FR1(IJ+28)
READ(20,3)TITLE;READ(20,3) FR1(IJ+29)
READ(20,3)TITLE;READ(20,3) FR1(IJ+30)
READ(20,3)TITLE;READ(20,3) FR1(IJ+31)
READ(20,3)TITLE;READ(20,3) FR1(IJ+32)
READ(20,3)TITLE;READ(20,3) FR1(IJ+33)
READ(20,3)TITLE;READ(20,3) FR1(IJ+34)
READ(20,3)TITLE;READ(20,3) FR1(IJ+35)
READ(20,3)TITLE;READ(20,3) FR1(IJ+36)
READ(20,3)TITLE;READ(20,3) FR1(IJ+37)
READ(20,3)TITLE;READ(20,3) FR1(IJ+38)
READ(20,3)TITLE;READ(20,3) FR1(IJ+39)
READ(20,3)TITLE;READ(20,3) FR1(IJ+40)
READ(20,3)TITLE;READ(20,3) FR1(IJ+41)
READ(20,3)TITLE;READ(20,3) FR1(IJ+42)
READ(20,3)TITLE;READ(20,3) FR1(IJ+43)
READ(20,3)TITLE;READ(20,3) FR1(IJ+44)
READ(20,3)TITLE;READ(20,3) FR1(IJ+45)
READ(20,3)TITLE;READ(20,3) FR1(IJ+46)
READ(20,3)TITLE;READ(20,3) FR1(IJ+47)
READ(20,3)TITLE;READ(20,3) FR1(IJ+48)
READ(20,3)TITLE;READ(20,3) FR1(IJ+49)
READ(20,3)TITLE;READ(20,3) FR1(IJ+50)
READ(20,3)TITLE;READ(20,3) FR1(IJ+51)
READ(20,3)TITLE;READ(20,3) FR1(IJ+52)
READ(20,3)TITLE;READ(20,3) FR1(IJ+53)
READ(20,3)TITLE;READ(20,*) ROH1
READ(20,3)TITLE;READ(20,*) OMEGA
READ(20,3)TITLE;READ(20,*) ALFAF
READ(20,3)TITLE;READ(20,*) ALFAC
READ(20,3)TITLE;READ(20,*) ALFAPR
READ(20,3)TITLE;READ(20,*) BTOTAL
READ(20,3)TITLE;READ(20,*) DECAY1
READ(20,3)TITLE;READ(20,*) DECAY2
READ(20,3)TITLE;READ(20,*) DECAY3
READ(20,3)TITLE;READ(20,*) DECAY4
READ(20,3)TITLE;READ(20,*) DECAY5
READ(20,3)TITLE;READ(20,*) DECAY6
READ(20,3)TITLE;READ(20,*) BETA1
READ(20,3)TITLE;READ(20,*) BETA2
READ(20,3)TITLE;READ(20,*) BETA3
READ(20,3)TITLE;READ(20,*) BETA4
READ(20,3)TITLE;READ(20,*) BETA5
READ(20,3)TITLE;READ(20,*) BETA6

```

```
READ(20,3)TITLE;READ(20,*) FI1
READ(20,3)TITLE;READ(20,*) FI2
READ(20,3)TITLE;READ(20,*) FI3
READ(20,3)TITLE;READ(20,*) FI4
READ(20,3)TITLE;READ(20,*) FI5
READ(20,3)TITLE;READ(20,*) FI6
READ(20,3)TITLE;READ(20,*) FI7
READ(20,3)TITLE;READ(20,*) FI8
READ(20,3)TITLE;READ(20,*) FI9
READ(20,3)TITLE;READ(20,*) FI10
READ(20,3)TITLE;READ(20,*) FI11
READ(20,3)TITLE;READ(20,*) FI12
READ(20,3)TITLE;READ(20,*) FI13
READ(20,3)TITLE;READ(20,*) FI14
READ(20,3)TITLE;READ(20,*) FI15
READ(20,3)TITLE;READ(20,*) CI1
READ(20,3)TITLE;READ(20,*) CI2
READ(20,3)TITLE;READ(20,*) CI3
READ(20,3)TITLE;READ(20,*) CI4
READ(20,3)TITLE;READ(20,*) CI5
READ(20,3)TITLE;READ(20,*) CI6
READ(20,3)TITLE;READ(20,*) CI7
READ(20,3)TITLE;READ(20,*) CI8
READ(20,3)TITLE;READ(20,*) CI9
READ(20,3)TITLE;READ(20,*) CI10
READ(20,3)TITLE;READ(20,*) CI11
READ(20,3)TITLE;READ(20,*) CI12
READ(20,3)TITLE;READ(20,*) CI13
READ(20,3)TITLE;READ(20,*) CI14
READ(20,3)TITLE;READ(20,*) CI15
READ(20,3)TITLE;READ(20,*) CI16
READ(20,3)TITLE;READ(20,*) CI17
READ(20,3)TITLE;READ(20,*) CI18
READ(20,3)TITLE;READ(20,*) CI19
READ(20,3)TITLE;READ(20,*) CI20
READ(20,3)TITLE;READ(20,*) CI21
READ(20,3)TITLE;READ(20,*) CI22
READ(20,3)TITLE;READ(20,*) CI23
READ(20,3)TITLE;READ(20,*) CI24
READ(20,3)TITLE;READ(20,*) CI25
READ(20,3)TITLE;READ(20,*) CI26
READ(20,3)TITLE;READ(20,*) CI27
READ(20,3)TITLE;READ(20,*) CI28
READ(20,3)TITLE;READ(20,*) CI29
READ(20,3)TITLE;READ(20,*) CI30
READ(20,3)TITLE;READ(20,*) POWER1
READ(20,3)TITLE;READ(20,*) FM1
READ(20,3)TITLE;READ(20,*) FM2
READ(20,3)TITLE;READ(20,*) FM3
READ(20,3)TITLE;READ(20,*) FM4
READ(20,3)TITLE;READ(20,*) FM5
READ(20,3)TITLE;READ(20,*) FM6
READ(20,3)TITLE;READ(20,*) FM7
READ(20,3)TITLE;READ(20,*) FM8
READ(20,3)TITLE;READ(20,*) FM9
READ(20,3)TITLE;READ(20,*) FM10
READ(20,3)TITLE;READ(20,*) FM11
READ(20,3)TITLE;READ(20,*) FM12
READ(20,3)TITLE;READ(20,*) FM13
READ(20,3)TITLE;READ(20,*) FM14
READ(20,3)TITLE;READ(20,*) FM15
```

```

READ(20,3)TITLE;READ(20,*) CPFUEL
READ(20,3)TITLE;READ(20,*) HFC
READ(20,3)TITLE;READ(20,*) AREAFC
READ(20,3)TITLE;READ(20,*) WP
READ(20,3)TITLE;READ(20,*) PLRM
READ(20,3)TITLE;READ(20,*) CMC1
READ(20,3)TITLE;READ(20,*) CMC2
READ(20,3)TITLE;READ(20,*) CMC3
READ(20,3)TITLE;READ(20,*) CMC4
READ(20,3)TITLE;READ(20,*) CMC5
READ(20,3)TITLE;READ(20,*) CMC6
READ(20,3)TITLE;READ(20,*) CMC7
READ(20,3)TITLE;READ(20,*) CMC8
READ(20,3)TITLE;READ(20,*) CMC9
READ(20,3)TITLE;READ(20,*) CMC10
READ(20,3)TITLE;READ(20,*) CMC11
READ(20,3)TITLE;READ(20,*) CMC12
READ(20,3)TITLE;READ(20,*) CMC13
READ(20,3)TITLE;READ(20,*) CMC14
READ(20,3)TITLE;READ(20,*) CMC15
READ(20,3)TITLE;READ(20,*) CMC16
READ(20,3)TITLE;READ(20,*) CMC17
READ(20,3)TITLE;READ(20,*) CMC18
READ(20,3)TITLE;READ(20,*) CMC19
READ(20,3)TITLE;READ(20,*) CMC20
READ(20,3)TITLE;READ(20,*) CMC21
READ(20,3)TITLE;READ(20,*) CMC22
READ(20,3)TITLE;READ(20,*) CMC23
READ(20,3)TITLE;READ(20,*) CMC24
READ(20,3)TITLE;READ(20,*) CMC25
READ(20,3)TITLE;READ(20,*) CMC26
READ(20,3)TITLE;READ(20,*) CMC27
READ(20,3)TITLE;READ(20,*) CMC28
READ(20,3)TITLE;READ(20,*) CMC29
READ(20,3)TITLE;READ(20,*) CMC30
READ(20,3)TITLE;READ(20,*) CP1
READ(20,3)TITLE;READ(20,*) UPM
READ(20,3)TITLE;READ(20,*) POWER
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+1)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+2)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+3)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+4)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+5)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+6);TF10=X2(M,ICOUNT+6)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+7);TF20=X2(M,ICOUNT+7)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+8);TF30=X2(M,ICOUNT+8)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+9);TF40=X2(M,ICOUNT+9)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+10);TF50=X2(M,ICOUNT+10)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+11);TF60=X2(M,ICOUNT+11)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+12);TF70=X2(M,ICOUNT+12)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+13);TF80=X2(M,ICOUNT+13)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+14);TF90=X2(M,ICOUNT+14)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+15);TF100=X2(M,ICOUNT+15)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+16);TF110=X2(M,ICOUNT+16)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+17);TF120=X2(M,ICOUNT+17)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+18);TF130=X2(M,ICOUNT+18)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+19);TF140=X2(M,ICOUNT+19)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+20);TF150=X2(M,ICOUNT+20)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+21)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+22);TC10=X2(M,ICOUNT+22)

```

```

READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+23);TC20=X2(M,ICOUNT+23)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+24);TC30=X2(M,ICOUNT+24)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+25);TC40=X2(M,ICOUNT+25)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+26);TC50=X2(M,ICOUNT+26)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+27);TC60=X2(M,ICOUNT+27)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+28);TC70=X2(M,ICOUNT+28)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+29);TC80=X2(M,ICOUNT+29)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+30);TC90=X2(M,ICOUNT+
* 30)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+31);TCC100=X2(M,ICOUNT+
* 31)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+32);TC110=X2(M,ICOUNT+32)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+33);TC120=X2(M,ICOUNT+33)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+34);TC130=X2(M,ICOUNT+34)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+35);TC140=X2(M,ICOUNT+35)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+36);TC150=X2(M,ICOUNT+36)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+37);TC160=X2(M,ICOUNT+37)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+38);TC170=X2(M,ICOUNT+38)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+39);TC180=X2(M,ICOUNT+39)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+40);TC190=X2(M,ICOUNT+40)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+41);TC200=X2(M,ICOUNT+41)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+42);TC210=X2(M,ICOUNT+42)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+43);TC220=X2(M,ICOUNT+43)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+44);TC230=X2(M,ICOUNT+44)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+45);TC240=X2(M,ICOUNT+45)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+46);TC250=X2(M,ICOUNT+46)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+47);TC260=X2(M,ICOUNT+47)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+48);TC270=X2(M,ICOUNT+48)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+49);TC280=X2(M,ICOUNT+49)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+50);TC290=X2(M,ICOUNT+50)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+51);TC300=X2(M,ICOUNT+51)
READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT+52)
READ(20,3)TITLE;READ(20,*) PRESS1;PRESS=PRESS1
READ(20,3)TITLE;READ(20,*) TCLEG1
CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE FORCING TERM.
C*****
      TCLEG=DTTCLEG+TCLEG1
      ROH=ROH1+DROH
C*****
C* EXCHANGE THE VARIABLES.
C*****
4016  PREC1=X2(M,ICOUNT)
      PREC2=X2(M,ICOUNT+1)
      PREC3=X2(M,ICOUNT+2)
      PREC4=X2(M,ICOUNT+3)
      PREC5=X2(M,ICOUNT+4)
      PREC6=X2(M,ICOUNT+5)
      TF1=X2(M,ICOUNT+6)
      TF2=X2(M,ICOUNT+7)
      TF3=X2(M,ICOUNT+8)
      TF4=X2(M,ICOUNT+9)
      TF5=X2(M,ICOUNT+10)
      TF6=X2(M,ICOUNT+11)
      TF7=X2(M,ICOUNT+12)
      TF8=X2(M,ICOUNT+13)
      TF9=X2(M,ICOUNT+14)
      TFF10=X2(M,ICOUNT+15)
      TF11=X2(M,ICOUNT+16)
      TF12=X2(M,ICOUNT+17)

```

```

TF13=X2(M,ICOUNT+18)
TF14=X2(M,ICOUNT+19)
TF15=X2(M,ICOUNT+20)
TLP=X2(M,ICOUNT+21)
TC1=X2(M,ICOUNT+22)
TC2=X2(M,ICOUNT+23)
TC3=X2(M,ICOUNT+24)
TC4=X2(M,ICOUNT+25)
TC5=X2(M,ICOUNT+26)
TC6=X2(M,ICOUNT+27)
TC7=X2(M,ICOUNT+28)
TC8=X2(M,ICOUNT+29)
TC9=X2(M,ICOUNT+30)
TCC10=X2(M,ICOUNT+31)
TC11=X2(M,ICOUNT+32)
TC12=X2(M,ICOUNT+33)
TC13=X2(M,ICOUNT+34)
TC14=X2(M,ICOUNT+35)
TC15=X2(M,ICOUNT+36)
TC16=X2(M,ICOUNT+37)
TC17=X2(M,ICOUNT+38)
TC18=X2(M,ICOUNT+39)
TC19=X2(M,ICOUNT+40)
TCC20=X2(M,ICOUNT+41)
TC21=X2(M,ICOUNT+42)
TC22=X2(M,ICOUNT+43)
TC23=X2(M,ICOUNT+44)
TC24=X2(M,ICOUNT+45)
TC25=X2(M,ICOUNT+46)
TC26=X2(M,ICOUNT+47)
TC27=X2(M,ICOUNT+48)
TC28=X2(M,ICOUNT+49)
TC29=X2(M,ICOUNT+50)
TCC30=X2(M,ICOUNT+51)
TUP=X2(M,ICOUNT+52)
IPRIME=ICOUNT+52
IK=IJ+52
IJK=IJA
IF(JJ1.EQ.0)GOTO 4066

```

C*****

C* MODEL'S GOVERNING EQUATIONS ARE LISTED BELOW. *

C*****

C REACTIVITY TERM

```

REACT=ROH+ALFAF*(FI1*(TF1-TF10)+FI2*(TF2-TF20)+FI3*(TF3-TF30)
* +FI4*(TF4-TF40)+FI5*(TF5-TF50)+FI6*(TF6-TF60)+FI7*(TF7-TF70)
* +FI8*(TF8-TF80)+FI9*(TF9-TF90)+FI10*(TF10-TF100)+FI11*(TF11-
* TF110)+FI12*(TF12-TF120)+FI13*(TF13-TF130)+FI14*(TF14-TF140)
* +FI15*(TF15-TF150))+ALFAC*(CI1*(TC1-TC10)+CI2*(TC2-TC20)
* +CI3*(TC3-TC30)+CI4*(TC4-TC40)+CI5*(TC5-TC50)+CI6*(TC6-TC60)
* +CI7*(TC7-TC70)+CI8*(TC8-TC80)+CI9*(TC9-TC90)+CI10*(TCC10
* -TCC100)+CI11*(TC11-TC110)+CI12*(TC12-TC120)+CI13*(TC13-TC130)
* +CI14*(TC14-TC140)+CI15*(TC15-TC150)+CI16*(TC16-TC160)+CI17
* *(TC17-TC170)+CI18*(TC18-TC180)+CI19*(TC19-TC190)+CI20*(TCC20-
* TC200)+CI21*(TC21-TC210)+CI22*(TC22-TC220)+CI23*(TC23-TC230)
* +CI24*(TC24-TC240)+CI25*(TC25-TC250)+CI26*(TC26-TC260)+CI27
* *(TC27-TC270)+CI28*(TC28-TC280)+CI29*(TC29-TC290)+CI30*(TCC30
* -TC300))+ALPAPR*(PRESS-PRESS1)

```

C POWER EQUATION

```

POWER=(REACT+OMEGA*(+DECAY1*PREC1
* +DECAY2*PREC2+DECAY3*PREC3+DECAY4*PREC4+DECAY5*PREC5
* +DECAY6*PREC6))/(BTOTAL-REACT)

```

```

C DELAYED NEUTRONS EQUATION (GROUP ONE)
  F(M,ICOUNT)=BETA1*POWER/OMEGA-DECAY1*PREC1
C DELAYED NEUTRONS EQUATION (GROUP TWO)
  F(M,ICOUNT+1)=BETA2*POWER/OMEGA-DECAY2*PREC2
C DELAYED NEUTRONS EQUATION (GROUP THREE)
  F(M,ICOUNT+2)=BETA3*POWER/OMEGA-DECAY3*PREC3
C DELAYED NEUTRONS EQUATION (GROUP FOUR)
  F(M,ICOUNT+3)=BETA4*POWER/OMEGA-DECAY4*PREC4
C DELAYED NEUTRONS EQUATION (GROUP FIVE)
  F(M,ICOUNT+4)=BETA5*POWER/OMEGA-DECAY5*PREC5
C DELAYED NEUTRONS EQUATION (GROUP SIX)
  F(M,ICOUNT+5)=BETA6*POWER/OMEGA-DECAY6*PREC6
C FUEL TEMPERATURE EQUATION (1ST FUEL LUMP)
  F(M,ICOUNT+6)=(FI1*(POWER1+POWER1*POWER)-HFC*AREAF*(TF1-TC1)
    *)/(FM1*CPFUEL)
C FUEL TEMPERATURE EQUATION (2ND FUEL LUMP)
  F(M,ICOUNT+7)=(FI2*(POWER1+POWER1*POWER)-HFC*AREAF*(TF2-TC3)
    *)/(FM2*CPFUEL)
C FUEL TEMPERATURE EQUATION (3RD FUEL LUMP)
  F(M,ICOUNT+8)=(FI3*(POWER1+POWER1*POWER)-HFC*AREAF*(TF3-TC5)
    *)/(FM3*CPFUEL)
C FUEL TEMPERATURE EQUATION (4TH FUEL LUMP)
  F(M,ICOUNT+9)=(FI4*(POWER1+POWER1*POWER)-HFC*AREAF*(TF4-TC7)
    *)/(FM4*CPFUEL)
C FUEL TEMPERATURE EQUATION (5TH FUEL LUMP)
  F(M,ICOUNT+10)=(FI5*(POWER1+POWER1*POWER)-HFC*AREAF*(TF5-TC9)
    *)/(FM5*CPFUEL)
C FUEL TEMPERATURE EQUATION (6TH FUEL LUMP)
  F(M,ICOUNT+11)=(FI6*(POWER1+POWER1*POWER)-HFC*AREAF*(TF6-TC11)
    *)/(FM6*CPFUEL)
C FUEL TEMPERATURE EQUATION (7TH FUEL LUMP)
  F(M,ICOUNT+12)=(FI7*(POWER1+POWER1*POWER)-HFC*AREAF*(TF7-TC13)
    *)/(FM7*CPFUEL)
C FUEL TEMPERATURE EQUATION (8TH FUEL LUMP)
  F(M,ICOUNT+13)=(FI8*(POWER1+POWER1*POWER)-HFC*AREAF*(TF8-TC15)
    *)/(FM8*CPFUEL)
C FUEL TEMPERATURE EQUATION (9TH FUEL LUMP)
  F(M,ICOUNT+14)=(FI9*(POWER1+POWER1*POWER)-HFC*AREAF*(TF9-TC17)
    *)/(FM9*CPFUEL)
C FUEL TEMPERATURE EQUATION (10TH FUEL LUMP)
  F(M,ICOUNT+15)=(FI10*(POWER1+POWER1*POWER)-HFC*AREAF*(TF10-
    * TC19))/(FM10*CPFUEL)
C FUEL TEMPERATURE EQUATION (11TH FUEL LUMP)
  F(M,ICOUNT+16)=(FI11*(POWER1+POWER1*POWER)-HFC*AREAF*(TF11-
    * TC21))/(FM11*CPFUEL)
C FUEL TEMPERATURE EQUATION (12TH FUEL LUMP)
  F(M,ICOUNT+17)=(FI12*(POWER1+POWER1*POWER)-HFC*AREAF*(TF12-
    * TC23))/(FM12*CPFUEL)
C FUEL TEMPERATURE EQUATION (13TH FUEL LUMP)
  F(M,ICOUNT+18)=(FI13*(POWER1+POWER1*POWER)-HFC*AREAF*(TF13-
    * TC25))/(FM13*CPFUEL)
C FUEL TEMPERATURE EQUATION (14TH FUEL LUMP)
  F(M,ICOUNT+19)=(FI14*(POWER1+POWER1*POWER)-HFC*AREAF*(TF14-
    * TC27))/(FM14*CPFUEL)
C FUEL TEMPERATURE EQUATION (15TH FUEL LUMP)
  F(M,ICOUNT+20)=(FI15*(POWER1+POWER1*POWER)-HFC*AREAF*(TF15-
    * TC29))/(FM15*CPFUEL)
C COOLANT TEMPERATURE IN THE LOWER PLENUM LUMP
  F(M,ICOUNT+21)=WP*(TCLEG-TLP)/PLRM
C COOLANT TEMPERATURE IN THE 1ST COOLANT LUMP
  F(M,ICOUNT+22)=WP*(TLP-TC1)/CMC1+HFC*AREAF*(TF1-TC1)/(2*CMC1

```

```

* *CP1)
C COOLANT TEMPERATURE IN THE 2ND COOLANT LUMP
  F(M,ICOUNT+23)=(WP*(TC1-TC2)+HFC*AREAF* (TF1-TC1))/(2*CP1))/CMC2
C COOLANT TEMPERATURE IN THE 3RD COOLANT LUMP
  F(M,ICOUNT+24)=(WP*(TC2-TC3)+HFC*AREAF* (TF2-TC3))/(2*CP1))/CMC3
C COOLANT TEMPERATURE IN THE 4TH COOLANT LUMP
  F(M,ICOUNT+25)=(WP*(TC3-TC4)+HFC*AREAF* (TF2-TC3))/(2*CP1))/CMC4
C COOLANT TEMPERATURE IN THE 5TH COOLANT LUMP
  F(M,ICOUNT+26)=(WP*(TC4-TC5)+HFC*AREAF* (TF3-TC5))/(2*CP1))/CMC5
C COOLANT TEMPERATURE IN THE 6TH COOLANT LUMP
  F(M,ICOUNT+27)=(WP*(TC5-TC6)+HFC*AREAF* (TF3-TC5))/(2*CP1))/CMC6
C COOLANT TEMPERATURE IN THE 7TH COOLANT LUMP
  F(M,ICOUNT+28)=(WP*(TC6-TC7)+HFC*AREAF* (TF4-TC7))/(2*CP1))/CMC7
C COOLANT TEMPERATURE IN THE 8TH COOLANT LUMP
  F(M,ICOUNT+29)=(WP*(TC7-TC8)+HFC*AREAF* (TF4-TC7))/(2*CP1))/CMC8
C COOLANT TEMPERATURE IN THE 9TH COOLANT LUMP
  F(M,ICOUNT+30)=(WP*(TC8-TC9)+HFC*AREAF* (TF5-TC9))/(2*CP1))/CMC9
C COOLANT TEMPERATURE IN THE 10TH COOLANT LUMP
  F(M,ICOUNT+31)=(WP*(TC9-TC10)+HFC*AREAF* (TF5-TC9))/(2*CP1))/
  * CMC10
C COOLANT TEMPERATURE IN THE 11TH COOLANT LUMP
  F(M,ICOUNT+32)=(WP*(TLP-TC11))/CMC11+HFC*AREAF* (TF6-TC11)/(2
  * *CMC11*CP1)
C COOLANT TEMPERATURE IN THE 12TH COOLANT LUMP
  F(M,ICOUNT+33)=(WP*(TC11-TC12)+HFC*AREAF* (TF6-TC11))/(2*CP1))
  * /CMC12
C COOLANT TEMPERATURE IN THE 13TH COOLANT LUMP
  F(M,ICOUNT+34)=(WP*(TC12-TC13)+HFC*AREAF* (TF7-TC13))/(2*CP1))
  * /CMC13
C COOLANT TEMPERATURE IN THE 14TH COOLANT LUMP
  F(M,ICOUNT+35)=(WP*(TC13-TC14)+HFC*AREAF* (TF7-TC13))/(2*CP1))
  * /CMC14
C COOLANT TEMPERATURE IN THE 15TH COOLANT LUMP
  F(M,ICOUNT+36)=(WP*(TC14-TC15)+HFC*AREAF* (TF8-TC15))/(2*CP1))
  * /CMC15
C COOLANT TEMPERATURE IN THE 16TH COOLANT LUMP
  F(M,ICOUNT+37)=(WP*(TC15-TC16)+HFC*AREAF* (TF8-TC15))/(2*CP1))
  * /CMC16
C COOLANT TEMPERATURE IN THE 17TH COOLANT LUMP
  F(M,ICOUNT+38)=(WP*(TC16-TC17)+HFC*AREAF* (TF9-TC17))/(2*CP1))
  * /CMC17
C COOLANT TEMPERATURE IN THE 18TH COOLANT LUMP
  F(M,ICOUNT+39)=(WP*(TC17-TC18)+HFC*AREAF* (TF9-TC17))/(2*CP1))
  * /CMC18
C COOLANT TEMPERATURE IN THE 19TH COOLANT LUMP
  F(M,ICOUNT+40)=(WP*(TC18-TC19)+HFC*AREAF* (TF10-TC19))/(2*CP1))
  * /CMC19
C COOLANT TEMPERATURE IN THE 20TH COOLANT LUMP
  F(M,ICOUNT+41)=(WP*(TC19-TC20)+HFC*AREAF* (TF10-TC19))/(2*CP1))
  * /CMC20
C COOLANT TEMPERATURE IN THE 21TH COOLANT LUMP
  F(M,ICOUNT+42)=(WP*(TLP-TC21)+HFC*AREAF* (TF11-TC21))/(2*CP1))
  * /CMC21
C COOLANT TEMPERATURE IN THE 22TH COOLANT LUMP
  F(M,ICOUNT+43)=(WP*(TC21-TC22)+HFC*AREAF* (TF11-TC21))/(2*CP1))
  * /CMC22
C COOLANT TEMPERATURE IN THE 23TH COOLANT LUMP
  F(M,ICOUNT+44)=(WP*(TC22-TC23)+HFC*AREAF* (TF12-TC23))/(2*CP1))
  * /CMC23
C COOLANT TEMPERATURE IN THE 24TH COOLANT LUMP
  F(M,ICOUNT+45)=(WP*(TC23-TC24)+HFC*AREAF* (TF12-TC23))/(2*CP1))

```

```

* /CMC24
C COOLANT TEMPERATURE IN THE 25TH COOLANT LUMP
  F(M,ICOUNT+46)=(WP*(TC24-TC25)+HFC*AREAFC*(TF13-TC25))/(2*CP1))
* /CMC25
C COOLANT TEMPERATURE IN THE 26TH COOLANT LUMP
  F(M,ICOUNT+47)=(WP*(TC25-TC26)+HFC*AREAFC*(TF13-TC25))/(2*CP1))
* /CMC26
C COOLANT TEMPERATURE IN THE 27TH COOLANT LUMP
  F(M,ICOUNT+48)=(WP*(TC26-TC27)+HFC*AREAFC*(TF14-TC27))/(2*CP1))
* /CMC27
C COOLANT TEMPERATURE IN THE 28TH COOLANT LUMP
  F(M,ICOUNT+49)=(WP*(TC27-TC28)+HFC*AREAFC*(TF14-TC27))/(2*CP1))
* /CMC28
C COOLANT TEMPERATURE IN THE 29TH COOLANT LUMP
  F(M,ICOUNT+50)=(WP*(TC28-TC29)+HFC*AREAFC*(TF15-TC29))/(2*CP1))
* /CMC29
C COOLANT TEMPERATURE IN THE 30TH COOLANT LUMP
  F(M,ICOUNT+51)=(WP*(TC29-TCC30)+HFC*AREAFC*(TF15-TC29))/(2*CP1))
* /CMC30
C COOLANT TEMPERATURE IN THE CORE UPPER PLENUM
  F(M,ICOUNT+52)=WP*((TCC10+TCC20+TCC30)/3-TUP)/UPM
  IF(1FINE.EQ.0) WRITE(23,*)POWER
4066 CONTINUE

```

RESPONSE OF THE REACTOR POWER IS PLACED IN POWER.DAT
POWER.DAT
RESPONSE OF THE 1ST PRECURSOR GROUP IS PLACED IN PRE1.DAT
PRE1.DAT
RESPONSE OF THE 2ND PRECURSOR GROUP IS PLACED IN PRE2.DAT
PRE2.DAT
RESPONSE OF THE 3RD PRECURSOR GROUP IS PLACED IN PRE3.DAT
PRE3.DAT
RESPONSE OF THE 4TH PRECURSOR GROUP IS PLACED IN PRE4.DAT
PRE4.DAT
RESPONSE OF THE 5TH PRECURSOR GROUP IS PLACED IN PRE5.DAT
PRE5.DAT
RESPONSE OF THE 6TH PRECURSOR GROUP IS PLACED IN PRE6.DAT
PRE6.DAT
RESPONSE OF THE 1ST FUEL LUMP TEMPERATURE IS PLACED IN FUEL1.DAT
FUEL1.DAT
RESPONSE OF THE 2ND FUEL LUMP TEMPERATURE IS PLACED IN FUEL2.DAT
FUEL2.DAT
RESPONSE OF THE 3RD FUEL LUMP TEMPERATURE IS PLACED IN FUEL3.DAT
FUEL3.DAT
RESPONSE OF THE 4TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL4.DAT
FUEL4.DAT
RESPONSE OF THE 5TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL5.DAT
FUEL5.DAT
RESPONSE OF THE 6TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL6.DAT
FUEL6.DAT
RESPONSE OF THE 7TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL7.DAT
FUEL7.DAT
RESPONSE OF THE 8TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL8.DAT
FUEL8.DAT
RESPONSE OF THE 9TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL9.DAT
FUEL9.DAT
RESPONSE OF THE 10TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL10.DAT
FUEL10.DAT
RESPONSE OF THE 11TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL11.DAT
FUEL11.DAT
RESPONSE OF THE 12TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL12.DAT
FUEL12.DAT
RESPONSE OF THE 13TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL13.DAT
FUEL13.DAT
RESPONSE OF THE 14TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL14.DAT
FUEL14.DAT
RESPONSE OF THE 15TH FUEL LUMP TEMPERATURE IS PLACED IN FUEL15.DAT
FUEL15.DAT
RESPONSE OF THE LOWER PLENUM TEMPERATURE IS PLACED IN LOWER.DAT
LOWER.DAT
RESPONSE OF THE 1ST COOLANT LUMP TEMPERATURE IS PLACED IN 1ST.DAT
1ST.DAT
RESPONSE OF THE 2ND COOLANT LUMP TEMPERATURE IS PLACED IN 2ND.DAT
2ND.DAT
RESPONSE OF THE 3RD COOLANT LUMP TEMPERATURE IS PLACED IN 3RD.DAT
3RD.DAT
RESPONSE OF THE 4TH COOLANT LUMP TEMPERATURE IS PLACED IN 4TH.DAT
4TH.DAT
RESPONSE OF THE 5TH COOLANT LUMP TEMPERATURE IS PLACED IN 5TH.DAT
5TH.DAT
RESPONSE OF THE 6TH COOLANT LUMP TEMPERATURE IS PLACED IN 6TH.DAT
6TH.DAT
RESPONSE OF THE 7TH COOLANT LUMP TEMPERATURE IS PLACED IN 7TH.DAT
7TH.DAT
RESPONSE OF THE 8TH COOLANT LUMP TEMPERATURE IS PLACED IN 8TH.DAT

8TH.DAT
RESPONSE OF THE 9TH COOLANT LUMP TEMPERATURE IS PLACED IN 9TH.DAT
9TH.DAT
RESPONSE OF THE 10TH COOLANT LUMP TEMPERATURE IS PLACED IN 10TH.DAT
10TH.DAT
RESPONSE OF THE 11TH COOLANT LUMP TEMPERATURE IS PLACED IN 11TH.DAT
11TH.DAT
RESPONSE OF THE 12TH COOLANT LUMP TEMPERATURE IS PLACED IN 12TH.DAT
12TH.DAT
RESPONSE OF THE 13TH COOLANT LUMP TEMPERATURE IS PLACED IN 13TH.DAT
13TH.DAT
RESPONSE OF THE 14TH COOLANT LUMP TEMPERATURE IS PLACED IN 14TH.DAT
14TH.DAT
RESPONSE OF THE 15TH COOLANT LUMP TEMPERATURE IS PLACED IN 15TH.DAT
15TH.DAT
RESPONSE OF THE 16TH COOLANT LUMP TEMPERATURE IS PLACED IN 16TH.DAT
16TH.DAT
RESPONSE OF THE 17TH COOLANT LUMP TEMPERATURE IS PLACED IN 17TH.DAT
17TH.DAT
RESPONSE OF THE 18TH COOLANT LUMP TEMPERATURE IS PLACED IN 18TH.DAT
18TH.DAT
RESPONSE OF THE 19TH COOLANT LUMP TEMPERATURE IS PLACED IN 19TH.DAT
19TH.DAT
RESPONSE OF THE 20TH COOLANT LUMP TEMPERATURE IS PLACED IN 20TH.DAT
20TH.DAT
RESPONSE OF THE 21TH COOLANT LUMP TEMPERATURE IS PLACED IN 21TH.DAT
21TH.DAT
RESPONSE OF THE 22TH COOLANT LUMP TEMPERATURE IS PLACED IN 22TH.DAT
22TH.DAT
RESPONSE OF THE 23TH COOLANT LUMP TEMPERATURE IS PLACED IN 23TH.DAT
23TH.DAT
RESPONSE OF THE 24TH COOLANT LUMP TEMPERATURE IS PLACED IN 24TH.DAT
24TH.DAT
RESPONSE OF THE 25TH COOLANT LUMP TEMPERATURE IS PLACED IN 25TH.DAT
25TH.DAT
RESPONSE OF THE 26TH COOLANT LUMP TEMPERATURE IS PLACED IN 26TH.DAT
26TH.DAT
RESPONSE OF THE 27TH COOLANT LUMP TEMPERATURE IS PLACED IN 27TH.DAT
27TH.DAT
RESPONSE OF THE 28TH COOLANT LUMP TEMPERATURE IS PLACED IN 28TH.DAT
28TH.DAT
RESPONSE OF THE 29TH COOLANT LUMP TEMPERATURE IS PLACED IN 29TH.DAT
29TH.DAT
RESPONSE OF THE 30TH COOLANT LUMP TEMPERATURE IS PLACED IN 30TH.DAT
30TH.DAT
RESPONSE OF THE UPPER PLENUM TEMPERATURE IS PLACED IN UPPER.DAT
UPPER.DAT
EXTERNAL REACTIVITY AT STEADY STATE(\$)
0.00000000
NEUTRON GENERATION TIME(SEC)
1.790000E-05
FUEL REACTIVITY COEFFICIENT(/F)
-1.100000E-05
COOLANT REACTIVITY COEFFICIENT(/F)
-2.000000E-04
PRESSURE REACTIVITY COEFFICIENT(/F)
3.000000E-03
TOTAL DELAYED NEUTRONS FRACTION
6.450000E-03
DECAY CONSTANT FOR GROUP 1
1.240000E-02

DECAY CONSTANT FOR GROUP 2
0.3050000
DECAY CONSTANT FOR GROUP 3
0.1110000
DECAY CONSTANT FOR GROUP 4
0.3010000
DECAY CONSTANT FOR GROUP 5
1.100000
DECAY CONSTANT FOR GROUP 6
3.000000
1ST GROUP DELAYED NEUTRONS FRACTION
2.100000E-04
2ND GROUP DELAYED NEUTRONS FRACTION
1.410000E-03
3RD GROUP DELAYED NEUTRONS FRACTION
1.270000E-03
4TH GROUP DELAYED NEUTRONS FRACTION
2.550000E-03
5TH GROUP DELAYED NEUTRONS FRACTION
7.400000E-04
6TH GROUP DELAYED NEUTRONS FRACTION
2.700000E-04
IMPORTANCE OF THE REACTIVITY INTRODUCED IN 1ST FUEL LUMP
0.04862
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 2ND FUEL LUMP
0.12727
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 3RD FUEL LUMP
0.1573
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 4TH FUEL LUMP
0.12727
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 5TH FUEL LUMP
0.04862
IMPORTANCE OF THE REACTIVITY INTRODUCED IN 6TH FUEL LUMP
0.03433
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 7TH FUEL LUMP
0.08989
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 8TH FUEL LUMP
0.11111
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 9TH FUEL LUMP
0.08989
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 10TH FUEL LUMP
0.03433
IMPORTANCE OF THE REACTIVITY INTRODUCED IN 11TH FUEL LUMP
0.01254
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 12TH FUEL LUMP
0.03284
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 13TH FUEL LUMP
0.04059
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 14TH FUEL LUMP
0.03284
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 15TH FUEL LUMP
0.01254
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 1ST COOLANT LUMP
0.01246
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 2ND COOLANT LUMP
0.03617
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 3RD COOLANT LUMP
0.05631
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 4TH COOLANT LUMP
0.07096
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 5TH COOLANT LUMP

0.07866
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 6TH COOLANT LUMP
0.07866
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 7TH COOLANT LUMP
0.07096
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 8TH COOLANT LUMP
0.05631
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 9TH COOLANT LUMP
0.03617
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 10TH COOLANT LUMP
0.01246
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 11TH COOLANT LUMP
0.0088
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 12TH COOLANT LUMP
0.0255
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 13TH COOLANT LUMP
0.03977
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 14TH COOLANT LUMP
0.05011
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 15TH COOLANT LUMP
0.05555
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 16TH COOLANT LUMP
0.05555
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 17TH COOLANT LUMP
0.05011
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 18TH COOLANT LUMP
0.03977
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 19TH COOLANT LUMP
0.0255
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 20TH COOLANT LUMP
0.0088
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 21ST COOLANT LUMP
0.00321
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 22ND COOLANT LUMP
0.00932
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 23RD COOLANT LUMP
0.01453
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 24TH COOLANT LUMP
0.01831
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 25TH COOLANT LUMP
0.02029
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 26TH COOLANT LUMP
0.02029
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 27TH COOLANT LUMP
0.01831
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 28TH COOLANT LUMP
0.01453
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 29TH COOLANT LUMP
0.00932
IMPORTANCE OF THE REACTIVITY INTRODUCED IN THE 30TH COOLANT LUMP
0.00321
STEADY STATE REACTOR POWER AT 100%(BTU/SEC)
1.0425861E+06
MASS OF THE FUEL IN THE 1ST FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 2ND FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 3RD FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 4TH FUEL LUMP(IBM)
14166.7

MASS OF THE FUEL IN THE 5TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 6TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 7TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 8TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 9TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 10TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 11TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 12TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 13TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 14TH FUEL LUMP(IBM)
14166.7
MASS OF THE FUEL IN THE 15TH FUEL LUMP(IBM)
14166.7
FUEL SPECIFIC HEAT(BTU/IBM-F)
5.90000E-02
OVERALL HEAT TRANSFER COEFF. BETWEEN FUEL AND COOLANT(BTU/FT²-F-SEC)
5.55555E-02
OVERALL HEAT TRANSFER AREA PER FUEL LUMP BETWEEN FUEL AND COOLANT(FT²)
5661.3
COOLANT FLOW RATE(IBM/SEC)
4699.073333
MASS OF THE COOLANT INSIDE THE REACTOR LOWER PLENUM(IBM)
27320
MASS OF THE COOLANT INSIDE THE 1ST COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 2ND COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 3RD COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 4TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 5TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 6TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 7TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 8TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 9TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 10TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 11TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 12TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 13TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 14TH COOLANT LUMP(IBM)
4460.17
MASS OF THE COOLANT INSIDE THE 15TH COOLANT LUMP(IBM)

4460.17
 MASS OF THE COOLANT INSIDE THE 16TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 17TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 18TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 19TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 20TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 21ST COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 22ND COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 23RD COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 24TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 25TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 26TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 27TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 28TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 29TH COOLANT LUMP(IBM)
 4460.17
 MASS OF THE COOLANT INSIDE THE 30TH COOLANT LUMP(IBM)
 4460.17
 SPECIFIC HEAT FOR THE COOLANT(BTU/IBM-F)
 1.323
 MASS OF THE COOLANT IN THE REACTOR UPPER PLENUM(IBM)
 41900
 DEVIATION OF THE REACTOR POWER FROM STEADY STATE AT STEADY STATE
 0.0000000000000000
 DEVIATION OF THE 1ST PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 DEVIATION OF THE 2ND PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 DEVIATION OF THE 3RD PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 DEVIATION OF THE 4TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 DEVIATION OF THE 5TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 DEVIATION OF THE 6TH PRECURSOR GROUP CONCENTRATION FROM S.S AT S.S
 0.0000000000000000
 STEADY STATE TEMPERATURE IN THE 1ST FUEL LUMP(F)
 711.44464640000000
 STEADY STATE TEMPERATURE IN THE 2ND FUEL LUMP(F)
 986.91109970000000
 STEADY STATE TEMPERATURE IN THE 3RD FUEL LUMP(F)
 1110.318718
 STEADY STATE TEMPERATURE IN THE 4TH FUEL LUMP(F)
 1034.634236
 STEADY STATE TEMPERATURE IN THE 5TH FUEL LUMP(F)
 788.6669976
 STEADY STATE TEMPERATURE IN THE 6TH FUEL LUMP(F)
 662.878679

STEADY STATE TEMPERATURE IN THE 7TH FUEL LUMP(F)
857.469799
STEADY STATE TEMPERATURE IN THE 8TH FUEL LUMP(F)
944.6658105
STEADY STATE TEMPERATURE IN THE 9TH FUEL LUMP(F)
891.1745865
STEADY STATE TEMPERATURE IN THE 10TH FUEL LUMP(F)
717.4189269
STEADY STATE TEMPERATURE IN THE 11TH FUEL LUMP(F)
588.8201758
STEADY STATE TEMPERATURE IN THE 12TH FUEL LUMP(F)
659.9175501
STEADY STATE TEMPERATURE IN THE 13TH FUEL LUMP(F)
691.7651247
STEADY STATE TEMPERATURE IN THE 14TH FUEL LUMP(F)
672.231954
STEADY STATE TEMPERATURE IN THE 15TH FUEL LUMP(F)
608.7449248
STEADY STATE TEMPERATURE IN THE CORE LOWER PLENUM
546.20000000000000
STEADY STATE TEMPERATURE IN THE 1ST COOLANT LUMP(F)
550.27485080000000
STEADY STATE TEMPERATURE IN THE 2ND COOLANT LUMP(F)
554.35370170000000
STEADY STATE TEMPERATURE IN THE 3RD COOLANT LUMP(F)
565.02545830000000
STEADY STATE TEMPERATURE IN THE 4TH COOLANT LUMP(F)
575.69721490000000
STEADY STATE TEMPERATURE IN THE 5TH COOLANT LUMP(F)
588.88702640000000
STEADY STATE TEMPERATURE IN THE 6TH COOLANT LUMP(F)
602.07683800000000
STEADY STATE TEMPERATURE IN THE 7TH COOLANT LUMP(F)
612.74859460000000
STEADY STATE TEMPERATURE IN THE 8TH COOLANT LUMP(F)
623.42035120000000
STEADY STATE TEMPERATURE IN THE 9TH COOLANT LUMP(F)
627.49720200000000
STEADY STATE TEMPERATURE IN THE 10TH COOLANT LUMP(F)
631.57405290000000
STEADY STATE TEMPERATURE IN THE 11TH COOLANT LUMP(F)
549.07861560000000
STEADY STATE TEMPERATURE IN THE 12TH COOLANT LUMP(F)
551.95723110000000
STEADY STATE TEMPERATURE IN THE 13TH COOLANT LUMP(F)
559.49462570000000
STEADY STATE TEMPERATURE IN THE 14TH COOLANT LUMP(F)
567.03202020000000
STEADY STATE TEMPERATURE IN THE 15TH COOLANT LUMP(F)
576.34873960000000
STEADY STATE TEMPERATURE IN THE 16TH COOLANT LUMP(F)
585.66545900000000
STEADY STATE TEMPERATURE IN THE 17TH COOLANT LUMP(F)
593.20285350000000
STEADY STATE TEMPERATURE IN THE 18TH COOLANT LUMP(F)
600.74024800000000
STEADY STATE TEMPERATURE IN THE 19TH COOLANT LUMP(F)
603.61886350000000
STEADY STATE TEMPERATURE IN THE 20TH COOLANT LUMP(F)
606.49747910000000
STEADY STATE TEMPERATURE IN THE 21ST COOLANT LUMP(F)

547.251495500000000
STEADY STATE TEMPERATURE IN THE 22ND COOLANT LUMP(F)
548.302990900000000
STEADY STATE TEMPERATURE IN THE 23RD COOLANT LUMP(F)
551.056668000000000
STEADY STATE TEMPERATURE IN THE 24TH COOLANT LUMP(F)
553.810345200000000
STEADY STATE TEMPERATURE IN THE 25TH COOLANT LUMP(F)
557.213870000000000
STEADY STATE TEMPERATURE IN THE 26TH COOLANT LUMP(F)
560.617394800000000
STEADY STATE TEMPERATURE IN THE 27TH COOLANT LUMP(F)
563.371071000000000
STEADY STATE TEMPERATURE IN THE 28TH COOLANT LUMP(F)
566.124749000000000
STEADY STATE TEMPERATURE IN THE 29TH COOLANT LUMP(F)
567.176244500000000
STEADY STATE TEMPERATURE IN THE 30TH COOLANT LUMP(F)
568.227740000000000
STEADY STATE TEMPERATURE IN THE REACTOR UPPER PLENUM(F)
602.100867200000000
STEADY STATE PRESSURE IN THE PRIMARY SIDE(Psi)
2250.000000000000000
STEADY STATE TEMPERATURE IN THE COLD LEG(F)
546.200000000000000

```

C*****
C* CLDLEG.FOR: A LUMPED PARAMETERS DYNAMIC MODEL FOR A TYPICAL PRESS- *
C*          URIZED WATER REACTOR COLDLEG PIPING. THIS SECTION IS *
C*          MODELED AS A PURE TIME DELAY BETWEEN THE STEAM GENERA- *
C*          TOR EXIT AND THE REACTOR CORE ENTRANCE. ONE LUMP IS USED*
C*          TO MODEL THIS PIPING SYSTEM. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IF(ISTEP.NE.0) GOTO 5012
      DO 5080 IY=1,IPER
      IF(CHOICE(IY).EQ.'CLDLEG.FOR') GOTO 5090
5080  CONTINUE
      GOTO 5002
C*****
C* ENTER THE FORCING TERM. *
C*****
      DO 5001 ICLEAR=1,20
      WRITE(5,*)
5001  CONTINUE
5090  WRITE(5,5006)
      WRITE(5,5000)
      READ(5,*) DTSGOP
5000  FORMAT(1X,'*ENTER THE CHANGE IN THE STEAM GENERATOR OUTLET'/
      * 1X,'*PLENUM TEMPERATURE,DTSGOP='$)
      WRITE(5,5006)
5006  FORMAT(1X,'*****')
C*****
C* OPEN AN EXISTING FILE AND READ MODEL'S PARAMETERS. *
C*****
5002  IF(IREFF.EQ.1)GOTO 5004
5009  WRITE(5,5006);WRITE(5,5010);WRITE(5,5006)
      READ(5,3) HEAD1
5010  FORMAT(1X,'ENTER INPUT DATA FILE NAME, DATA FILENAME='$)
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 5005
5004  HEAD1='COLD.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
5005  READ(20,3)TITLE;READ(20,3) FR1(IJ)
      READ(20,3)TITLE;READ(20,15) (YTITLE(IJ,IOP),IOP=1,6)
      READ(20,3)TITLE;READ(20,*) WPCLEG
      READ(20,3)TITLE;READ(20,*) CLEGM
      READ(20,3)TITLE;READ(20,*) TSGOP1
      READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT)
      CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE FORCING TERM. *
C*****
5011  TSGOP=TSGOP1+DTSGOP
C*****
C* EXCHANGE THE VARIABLES. *
C*****
5012  TCLEG=X2(M,ICOUNT)
      IPRIME=ICOUNT
      IK=IJ
      IF(JJ1.EQ.0) GOTO 5013
C*****
C* MODEL GOVERNING EQUATION. *
C*****
C COLDLEG PIPING.
      F(M,ICOUNT)=(TSGOP-TCLEG)*WPCLEG/CLEGM

```

5013 CONTINUE.

**** This is the end of the program cldleg.for .

RESPONSE OF THE COLD LEG TEMPERATURE IS IN CLEG.DAT
CLEG.DAT
NEXT LINE IS THE TITLE AS IT APPEAR ON THE PLOT FOR CLEG.DAT VS TIME
COLD LEG TEMPERATURE(F)
COOLANT FLOW RATE IN THE COLD LEG (IBM/SEC)
9398.148
MASS OF THE COOLANT IN THE COLD LEG (IBM)
6350.1
COOLANT AVERAGE TEMP. IN THE COLD LEG AT STEADY STATE (F)
546.2
COOLANT AVERAGE TEMP. IN THE COLD LEG AT STEADY STATE (F)
546.2

```

C*****
C* HOTLEG.FOR: A LUMPED PARAMETERS DYNAMIC MODEL FOR A TYPICAL PRESS- *
C*          URIZED WATER REACTOR HOTLEG PIPING. THIS SECTION IS      *
C*          MODELED AS A PURE TIME DELAY BETWEEN THE CORE OUTLET      *
C*          AND THE STEAM GENERATOR ENTRANCE. ONE LUMP IS USED TO      *
C*          MODEL THIS PIPING SYSTEM.                                  *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IF(ISTEP.NE.0) GOTO 6012
      DO 6060 IY=1,IPER
      IF(CHOICE(IY).EQ.'HOTLEG.FOR') GOTO 6070
6060  CONTINUE
      GOTO 6002
C*****
C* ENTER THE FORCING TERM. *
C*****
      DO 6001 ICLEAR=1,20
      WRITE(5,*)
6001  CONTINUE
6070  WRITE(5,6006)
      WRITE(5,6000)
      READ(5,*) DTUP
6000  FORMAT(1X,'*ENTER THE CHANGE IN THE REACTOR CORE OUTLET',/,
* 1X,'*PLENUM TEMPERATURE,DTUP='$')
      WRITE(5,6006)
6006  FORMAT(1X,'*****')
C*****
C* OPEN AN EXISTING FILE AND READ MODEL'S PARAMETERS. *
C*****
6002  IF(IREFF.EQ.1) GOTO 6004
6009  WRITE(5,6010)
      READ(5,3) HEAD1
6010  FORMAT(1X,'ENTER INPUT DATA FILE NAME, DATA FILENAME='$')
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 6005
6004  HEAD1='HOT.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
6005  READ(20,3)TITLE;READ(20,3) FR1(IJ)
      READ(20,3)TITLE;READ(20,15) (YTITLE(IJ,IOP),IOP=1,6)
      READ(20,3)TITLE;READ(20,*) WPHOT
      READ(20,3)TITLE;READ(20,*) HLEGM
      READ(20,3)TITLE;READ(20,*) TUP1
      READ(20,3)TITLE;READ(20,*) X2(M,ICOUNT)
      CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE PARAMETERS. *
C*****
6011  TUP=TUP1+DTUP
C*****
C* EXCHANGE THE VARIABLES. *
C*****
6012  THLEG=X2(M,ICOUNT)
      IPRIME=ICOUNT
      IK=IJ
      IF(JJ1.EQ.0) GOTO 6013
C*****
C* MODEL GOVERNING EQUATION. *
C*****
C HOTLEG PIPING.
      F(M,ICOUNT)=(TUP-THLEG)*WPHOT/HLEGM

```

6013 CONTINUE

**** NOTE : This is the end of the program HOTLEG.FOR .

RESPONSE OF THE HOTLEG TEMPERATURE WILL BE IN HLEG.DAT
HLEG.DAT
NEXT LINE IS THE Y TITLE ON THE PLOT FOR HOTLEG RESPONSE VS TIME
HOT LEG TEMPERATURE(F).
COOLANT FLOW RATE(IBM/SEC)
9398.148
MASS OF THE COOLANT IN THE HOTLEG(IBM)
3759.250
AVERAGE COOLANT TEMPERATURE IN THE HOT LEG AT STEADY STATE(F)
602.100867200000000
AVERAGE COOLANT TEMPERATURE IN THE HOT LEG AT STEADY STATE(F)
602.100867200000000

```

C*****
C* SIMSGR.FOR : A LUMPED PARAMETERS DYNAMIC MODEL FOR A TYPICAL U- *
C*      TUBE STEAM GENERATOR SYSTEM.THIS SIMPLE STEAM GENE- *
C*      RATOR MODEL IS MADE UP OF THE FOLLOWING LUMPS. *
C*      -1 ONE LUMPE FOR THE INLET PLENUM. *
C*      -2 TWO LUMPS FOR THE PRIMARY WATER. *
C*      -3 ONE LUMP FOR THE OUTLET PLENUM. *
C*      -4 ONE LUMP FOR THE METAL LUMP. *
C*      -5 ONE LUMP FOR THE SECONDARY SIDE. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IF(ISTEP.NE.0)GOTO 7039
      DO 7080 IY=1,IPER
      IF(CHOICE(IY).EQ.'SIMSGR.FOR') GOTO 7090
7080  CONTINUE
      GOTO 7003
C*****
C* ENTER THE FORCING TERM. *
C*****
      DO 7020 ICLEAR=1,20
      WRITE(5,*)
7020  CONTINUE
7090  WRITE(5,7021)
      WRITE(5,7000)
      READ(5,*) DTHLEG
7000  FORMAT(1X,'*ENTER THE CHANGE IN THE HOTLEG TEMPERATURE',/,
* 1X,'*DTHLEG=')$)
      WRITE(5,7001)
      READ(5,*) DTFW
7001  FORMAT(1X,'*ENTER THE CHANGE IN THE FEEDWATER TEMPERATURE',/,
* 1X,'*DTFW=')$)
      WRITE(5,7002)
      READ(5,*) DVALVE
7002  FORMAT(1X,'*ENTER THE % CHANGE IN THE STEAM VALVE COEFFICIENT'/
* 1X,'*DVALVE=')$)
      WRITE(5,7021)
7021  FORMAT(1X,'*****')
C*****
C* OPEN AN EXISTING FILE AND READ MODEL'S PARAMETERS. *
C*****
7003  IF(IREFF.EQ.1)GOTO 7005
7036  WRITE(5,7037)
      READ(5,3) HEAD1
7037  FORMAT(1X,'ENTER INPUT DATA FILE NAME,FILE NAME=')$)
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 7006
7005  HEAD1='SIMPSG.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
7006  READ(20,3) TITLE;READ(20,3) FR1(IJ)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+1)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+2)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+3)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+4)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+5)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ,IOP),IOP=1,6)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ+1,IOP),IOP=1,6)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ+2,IOP),IOP=1,6)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ+3,IOP),IOP=1,6)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ+4,IOP),IOP=1,6)
      READ(20,3) TITLE;READ(20,15) (YTITLE(IJ+5,IOP),IOP=1,6)

```

```

READ(20,3) TITLE;READ(20,*) WPSG
READ(20,3) TITLE;READ(20,*) SGIPM
READ(20,3) TITLE;READ(20,*) PM
READ(20,3) TITLE;READ(20,*) HPM
READ(20,3) TITLE;READ(20,*) APM
READ(20,3) TITLE;READ(20,*) CP1
READ(20,3) TITLE;READ(20,*) SGOPM
READ(20,3) TITLE;READ(20,*) SMETAL
READ(20,3) TITLE;READ(20,*) CM
READ(20,3) TITLE;READ(20,*) HMS
READ(20,3) TITLE;READ(20,*) AMS
READ(20,3) TITLE;READ(20,*) SWV
READ(20,3) TITLE;READ(20,*) SSV
READ(20,3) TITLE;READ(20,*) CP2
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+1)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+2)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+3)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+4)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+5)
READ(20,3) TITLE;READ(20,*) THLEG1
READ(20,3) TITLE;READ(20,*) WSO1
READ(20,3) TITLE;READ(20,*) TFW1
CLOSE(UNIT=20,FILE=HEAD1)
C*****
C* INITIALIZE THE FORCING TERM.
C*****
7038 THLEG=THLEG1+DTTHLEG
      CVALVE=WSO1/X2(M,ICOUNT+5)+DVALVE*WSO1/X2(M,ICOUNT+5)
      TFW=TFW1+DTFW
C*****
C* EXCHANGE THE VARIABLES.
C*****
7039 TSGIP=X2(M,ICOUNT)
      TP1=X2(M,ICOUNT+1)
      TP2=X2(M,ICOUNT+2)
      TSGOP=X2(M,ICOUNT+3)
      TM=X2(M,ICOUNT+4)
      SPRES=X2(M,ICOUNT+5)
      IPRIME=ICOUNT+5
      IK=IJ+5
7040 IF(JJ1.EQ.0) GOTO 7041
C*****
C* MODEL'S EQUATIONS ARE LISTED BELOW.
C*****
C INLET PLENUM LUMP
      F(M,ICOUNT)=(THLEG-TSGIP)*WPSG/SGIPM
C 1ST PRIMARY COOLANT LUMP.
      F(M,ICOUNT+1)=(TSGIP-TP1)*2*WPSG/PM-HPM*APM*(TP1-TM)/(PM*CP1)
C 2ND PRIMARY COOLANT LUMP.
      F(M,ICOUNT+2)=(TP1-TP2)*2*WPSG/PM-HPM*APM*(TP1-TM)/(PM*CP1)
C OUTLET PLENUM LUMP.
      F(M,ICOUNT+3)=(TP2-TSGOP)*WPSG/SGOPM
C TUBE METAL LUMP.
      TSTEAM=0.1453*SPRES+401.1
      F(M,ICOUNT+4)=(HPM*APM*(TP1-TM)-HMS*AMS*(TM-TSTEAM))/
      * (SMETAL*CM)
C SECONDARY SIDE LUMP.
      HG=-0.029*SPRES+1222.45
      SWM=SWV*(54.63-8.36E-3*SPRES)
      SSM=SSV*(2.359E-3*SPRES-0.125)

```

```
DHF=-0.029
HFG=854.55-0.205*SPRES
VFG=1.2-7.98E-4*SPRES
DVG=-7.953E-4
SCON=(SWM*DHF+SSM*DHG-SSM*HFG*DVG/VFG)
F(M,ICOUNT+5)=(HMS*AMS*(TM-TSTEAM)-CVALVE*SPRES*(HG-CP2*TFW))/
* SCON
7041 CONTINUE
```

RESPONSE OF THE INLET PLENUM TEMPERATURE IS IN INLET.DAT
 INLET.DAT
 RESPONSE OF THE 1ST PRIMARY LUMP TEMP. IS IN PRIM1.DAT
 PRIM1.DAT
 RESPONSE OF THE 2ND PRIMARY LUMP TEMP. IS IN PRIM2.DAT
 PRIM2.DAT
 RESPONSE OF THE OUTLET PLENUM LUMP TEMP. IS IN OUTLET.DAT
 OUTLET.DAT
 RESPONSE OF THE TUBE METAL LUMP TEMP. IS IN METAL.DAT
 METAL.DAT
 RESPONSE OF THE SECONDARY SIDE PRESSURE IS IN SEC.DAT
 SEC.DAT
 NEXT LINE IS A TITLE FOR THE PLOT OF INLET.DAT VS TIME
 INLET PLENUM TEMPERATURE(F)
 NEXT LINE IS A TITLE FOR THE PLOT OF PRIM1.DAT VS TIME
 FIRST PRIMARY LUMP TEMPERATURE(F)
 NEXT LINE IS A TITLE FOR THE PLOT OF PRIM2.DAT VS TIME
 SECOND PRIMARY LUMP TEMPERATURE(F)
 NEXT LINE IS A TITLE FOR THE PLOT OF OUTLET.DAT VS TIME
 OUTLET PLENUM TEMPERATURE(F)
 NEXT LINE IS A TITLE FOR THE PLOT OF METAL.DAT VS TIME
 TUBE METAL TEMPERATURE(F)
 NEXT LINE IS A TITLE FOR THE PLOT OF SEC.DAT VS TIME
 SECONDARY SIDE PRESSURE(PSI)
 PRIMARY WATER FLOW RATE IN THE SG(IBM/SEC)
 9398.148
 PRIMARY COOLANT MASS INSIDE THE INLET PLENUM(IBM)
 6757.700
 MASS OF THE COOLANT INSIDE THE SG TUBES(IBM)
 29356.70
 OVERALL HEAT TRANSFER COEFFICIENT BETWEEN PRIMARY @ METAL(BTU/FT2-F-SE)
 0.611729
 HEAT TRANSFER AREA BETWEEN PRIMARY AND METAL(FT2)
 39355.50
 SPECIFIC HEAT FOR THE PRIMARY COOLANT(BTU/IBM-F)
 1.323000
 PRIMARY WATER MASS IN THE SG OUTLET PLENUM(IBM)
 6757.700
 TUBE METAL MASS(IBM)
 91800.00
 SPECIFIC HEAT FOR THE METAL TUBE(BTU/IBM-F)
 0.1090000
 HEAT TRANSFER COEFFICIENT BETWEEN METAL AND SEC. SIDE(BTU/FT2-F-SEC)
 0.5342885
 AREA FOR HEAT TRANSFER FROM THE METAL SIDE TO THE SEC SIDE(FT2)
 44430.00
 SECONDARY WATER VOLUME(FT3)
 1536.000
 SECONDARY STEAM VOLUME(FT3)
 3203.000
 SPECIFIC HEAT FOR THE SECONDARY WATER(BTU/IBM-F)
 0.980234112
 STEADY STATE TEMPERATURE IN THE SG INLET PLENUM(F)
 602.1008672
 S.S. TEMPERATURE IN THE SG 1ST COOLANT LUMP(F)
 574.1504336
 S.S. TEMPERATURE IN THE SG 2ND COOLANT LUMP(F)
 546.2
 S.S. TEMPERATURE IN THE SG OUTLET PLENUM(F)
 546.2
 S.S. TEMPERATURE IN THE SG TUBE METAL LUMP(F)

545.2798192
S.S PRESSURE IN THE SECONDARY SIDE(PSI)
790.7777013
S.S TEMPERATURE IN THE HOT LEG(F)
602.1008672
STEAM FLOW RATE FROM THE SG AT S.S(IBM/SEC)
887.7778
FEED WATER TEMPERATURE AT S.S(F)
425.0000000

```

C*****
C* PRZER2.FOR: A NONLINEAR LUMPED PARAMETER MODEL FOR A TYPICAL *
C* PRESSURIZED WATER REACTOR PRESSURIZER.TWO LUMPS *
C* ARE USED TO REPRESENT THE FLUID AND THE STEAM *
C* PHASES IN THE PRESSURIZER.THE PID HEATER CONTR- *
C* OLLER AND THE PID PROPORTIONAL SPRAYER ARE INCLU- *
C* IN THE MODEL *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IJA=IJKA+1
      IF(ISTEP.NE.0)GOTO 8017
      DO 8000 IY=1,IPER
      IF(CHOICE(IY).EQ.'PRZER2.FOR')GOTO 8001
8000  CONTINUE
      GOTO 8012
C*****
C* ENTER THE FORCING TERM. *
C*****
8001  DO 8002 ICLEAR=1,24
      WRITE(5,*)
8002  CONTINUE
      WRITE(5,8003);WRITE(5,8004);WRITE(5,8005);WRITE(5,8006)
      WRITE(5,8003);WRITE(5,8007);READ(5,*) ICASE(IAM)
8003  FORMAT(1X,'*****')
8004  FORMAT(1X,'* CHOOSE A CASE BY ENTERING THE # TO THE LEFT. *')
8005  FORMAT(1X,'* 1-ISOLATED PRESSURIZER CASE. *')
8006  FORMAT(1X,'* 2-PRESSURIZER COUPLED TO A PRIMARY LOOP. *')
8007  FORMAT(1X,'CASE=')$)
      IF(ICASE(IAM).EQ.2) GOTO 8011
      WRITE(5,8003)
      WRITE(5,8008);READ(5,*) DWI
8008  FORMAT(1X,'* ENTER DEVIATION IN THE SURGE LINE FLOW RATE',/,
* 1X,'*(IBM/SEC),DWI=')$)
      WRITE(5,8003)
8011  IAM=IAM+1
C*****
C* OPEN A FILE AND READ MODEL'S PARAMETERS. *
C*****
8012  DO 8013 ICLEAR=1,24
      WRITE(5,*)
8013  CONTINUE
      IF(IREFF.EQ.1) GOTO 8015
      WRITE(5,8003);WRITE(5,8014);WRITE(5,8003);READ(5,3) HEAD1
8014  FORMAT(1X,'ENTER A FILE NAME FOR THE PRESSURIZER DATA',/,
* 1X,'FILENAME=')$)
      OPEN(UNIT=20,FILE=HEAD1)
      GOTO 8016
8015  HEAD1='PRZ2.DAT'
      OPEN(UNIT=20,FILE=HEAD1)
8016  READ(20,3) TITLE;READ(20,3) AFR1(IJA)
      READ(20,3) TITLE;READ(20,3) AFR1(IJA+1)
      READ(20,3) TITLE;READ(20,3) FR1(IJ)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+1)
      READ(20,3) TITLE;READ(20,3) FR1(IJ+2)
      READ(20,3) TITLE;READ(20,*) PAREA
      READ(20,3) TITLE;READ(20,*) TLEN
      READ(20,3) TITLE;READ(20,*) HWI1
      READ(20,3) TITLE;READ(20,*) HWI2
      READ(20,3) TITLE;READ(20,*) HSP
      READ(20,3) TITLE;READ(20,*) SJQW

```

```

READ(20,3) TITLE;READ(20,*) WCLEG
READ(20,3) TITLE;READ(20,*) WHLEG
READ(20,3) TITLE;READ(20,*) TSGOP
READ(20,3) TITLE;READ(20,*) GAIN1
READ(20,3) TITLE;READ(20,*) GAIN2
READ(20,3) TITLE;READ(20,*) TOW1
READ(20,3) TITLE;READ(20,*) TOW2
READ(20,3) TITLE;READ(20,*) TCLEG
READ(20,3) TITLE;READ(20,*) THLEG
READ(20,3) TITLE;READ(20,*) TUP
READ(20,3) TITLE;READ(20,*) VHLEG
READ(20,3) TITLE;READ(20,*) VCLEG
READ(20,3) TITLE;READ(20,*) BTP
READ(20,3) TITLE;READ(20,*) HEATO
READ(20,3) TITLE;READ(20,*) HLOST
READ(20,3) TITLE;READ(20,*) WSP0
READ(20,3) TITLE;READ(20,*) HLEGM
READ(20,3) TITLE;READ(20,*) CLEGM
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT);PRESS1=X2(M,ICOUNT)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+1)
READ(20,3) TITLE;READ(20,*) X2(M,ICOUNT+2)
C*****
C* EXCHANGE THE VARIABLES.
C*****
8017 PRESS=X2(M,ICOUNT)
      WLEN=X2(M,ICOUNT+1)
      DUMMY=X2(M,ICOUNT+2)
      IPRIME=ICOUNT+2
      IK=IJ+2
      IJKA=IJA+1
      IF(JJ1.EQ.0) GOTO 8066
C*****
C* MODEL'S GOVERNING EQUATIONS ARE LISTED BELOW.
C*****
C
C
      WI=DWI-VHLEG*BTP*WCLEG*(TUP-THLEG)/HLEGM-VCLEG*BTP*WHLEG*
* (TSGOP-TCLEG)/CLEGM
      IF(WI.GT.0) HWI=HWI1
      IF(WI.LT.0) HWI=HWI2
      WPSLOP=-5.599091435E-06
      SPSLOP=2.675771597E-06
      VWATER=0.01009152*PRESS+23.939212
      VSTEAM=-0.19668096*PRESS+714.021696
      HWATER=0.1174*PRESS+436.99
      HSTEAM=-0.089*PRESS+1316.01
      UWATER=0.11*PRESS+442.4
      USTEAM=-0.07*PRESS+1207.94
      CONA1=((VSTEAM/VWATER)-1)
      CONA2=(PAREA*(TLEN-WLEN)*VSTEAM*SPSLOP/VWATER+PAREA*WLEN*WPSLOP)
      CONB1=(PAREA*WLEN*.11/VWATER+PRESS*PAREA*WLEN*0.01009152/
* (SJQW*VWATER)+(HSTEAM+UWATER+PRESS*VWATER/SJQW)*CONA2/CONA1)
      CONB2=(HWI-UWATER-PRESS*VWATER/SJQW+(HSTEAM+UWATER+PRESS*
* VWATER/SJQW)/CONA1)
      CONB3=(HISP-UWATER-PRESS*VWATER/SJQW+(HSTEAM+UWATER+PRESS*
* VWATER/SJQW)/CONA1)
      CONB4=-1*(HSTEAM+UWATER+PRESS*VWATER/SJQW)*VSTEAM/(CONA1*
* VWATER)
      CONC1=(PAREA*(TLEN-WLEN)*SPSLOP-CONA2/CONA1)*VSTEAM/(PAREA*
* CONB1)
      CONC2=((PAREA*(TLEN-WLEN)*SPSLOP-CONA2/CONA1)*CONB2*VSTEAM/

```

```

* (PAREA*CONB1)+VSTEAM/(PAREA*CONA1))
CONC3=((PAREA*(TLEN-WLEN)*SPSLOP-CONA2/CONA1)*CONB3*VSTEAM/
* (PAREA*CONB1)+VSTEAM/(PAREA*CONA1))
CONC4=((PAREA*(TLEN-WLEN)*SPSLOP-CONA2/CONA1)*CONB4*VSTEAM/
* (PAREA*CONB1)+(1-VSTEAM/(CONA1*VWATER))*VSTEAM/PAREA)
COND1=1/(CONB1-GAIN1*TOW2)
COND2=1/(CONB1-GAIN2*TOW2)
CONE1=(CONC1+CONC1*GAIN1*TOW2*COND1)
CONE2=(CONC1*GAIN1+CONC1*GAIN1*TOW2*COND1*GAIN1)
CONE3=(CONC1*GAIN1/TOW1+CONC1*GAIN1*TOW2*COND1*GAIN1/TOW1)
CONE4=(CONC1*GAIN1*TOW2*COND1*CONB2+CONC2)
CONE5=(CONC1*GAIN1*TOW2*COND1*CONB3+CONC3)
TESTER=PRESS-PRESS1
IF(TESTER.GT.15.0.AND.TESTER.LT.25.0) GOTO 8019
IF(TESTER.LE.15.0)GOTO 8020
IF(TESTER.GE.25.0) GOTO 8028
C*****
C* DEAD BAND.
C*****
8019 WSP=8.259E-02
HEAT=0
F(M,ICOUNT)=(HEAT-HLOST+CONB2*WI+CONB3*WSP)/CONB1
F(M,ICOUNT+1)=CONC1*(HEAT-HLOST)+CONC2*WI+CONC3*WSP
F(M,ICOUNT+2)=0
RHEAT=0
IF(IFINE.EQ.0) WRITE(23,*) RHEAT,STIME
IF(IFINE.EQ.0) WRITE(23,*) WSP,STIME
GOTO 8066
C*****
C* HEATERS ON
C*****
8020 IF(TESTER.LE.15.0.AND.TESTER.GE.-15.0) GOTO 8023
IF(TESTER.LT.-15.0.AND.TESTER.GE.-17.0) GOTO 8024
WRITE(5,8021);WRITE(5,8022)
8021 FORMAT(1X,'PRESSURE DEVIATION IS TOO LOW')
8022 FORMAT(1X,'PRESSURE DEVIATION IS OUT OF CONTROL');STOP
C*****
C* PID CONTROLLER
C*****
8023 WSP=8.259E-02
F(M,ICOUNT)=COND1*(HEAT-HLOST+(PRESS-PRESS1)*GAIN1+GAIN1*
* DUMMY/TOW1+CONB2*WI+CONB3*WSP+DHEAT)
F(M,ICOUNT+1)=CONE1*(HEAT-HLOST+DHEAT)+CONE2*(PRESS-PRESS1)+
* CONE3*DUMMY+CONE4*WI+CONE5*WSP
F(M,ICOUNT+2)=PRESS-PRESS1
RHEAT=HEAT+GAIN1*(PRESS-PRESS1)+GAIN1*DUMMY/TOW1+GAIN1*
* F(M,ICOUNT)
IF(RHEAT.LT.0.0) RHEAT=0.0
IF(IFINE.EQ.0)WRITE(23,*) RHEAT,STIME
IF(WSP.LT.0.0) WSP=0.0
IF(IFINE.EQ.0)WRITE(23,*) WSP,STIME
GOTO 8066
C*****
C* PID AT MAXIMUM.
C*****
8024 IFPR2=0
WSP=8.259E-02
HEAT=1706.05
F(M,ICOUNT)=(HEAT-HLOST+DHEAT+CONB2*WI+CONB3*WSP)/CONB1
F(M,ICOUNT+1)=CONC1*(HEAT-HLOST+DHEAT)+CONC2*WI+CONC3*WSP
F(M,ICOUNT+2)=0

```

```

RHEAT=HEAT
IF(RHEAT.LT.0.0) RHEAT=0.0
IF(WSP.LT.0.0) WSP=0.0
IF(IFINE.EQ.0) WRITE(23,*) RHEAT,STIME
IF(IFINE.EQ.0) WRITE(23,*) WSP,STIME
GOTO 8066
C*****
C* SPRAYER AND POWER RELIEF VALVE RANGE.
C*****
8028 IF(TESTER.GE.25.0.AND.TESTER.LE.75.0) GOTO 8031
      IF(TESTER.GT.75.0.AND.TESTER.LE.80.0) GOTO 8032
      WRITE(5,8029);WRITE(5,8030)
8029 FORMAT(1X,'PRESSURE DEVIATION IS TOO HIGH')
8030 FORMAT(1X,'PRESSURE DEVIATION IS OUT OF CONTROL');STOP
C*****
C* PROPORTIONAL SPRAYER ON.
C*****
8031 F(M,ICOUNT)=(DHEAT-HLOST+CONB2*WI+CONB3*(8.259E-02+
      * (PRESS-PRESS1)*GAIN2+GAIN2*DUMMY/TOW1))/(CONB1*(1-CONB3*
      * GAIN2*TOW2/CONB1))
      F(M,ICOUNT+1)=CONC1*(DHEAT-HLOST)+CONC2*WI+CONC3*(8.259E-02+
      * (PRESS-PRESS1)*GAIN2+GAIN2*DUMMY/TOW1+GAIN2*TOW2*F(M,ICOUNT))
      F(M,ICOUNT+2)=PRESS-PRESS1
      WSP=8.259E-02+(PRESS-PRESS1)*GAIN2+GAIN2*DUMMY/TOW1+GAIN2*
      * F(M,ICOUNT)
      RHEAT=0
      IF(IFINE.EQ.0)WRITE(23,*) RHEAT,STIME
      IF(WSP.LT.0.0) WSP=0.0
      IF(IFINE.EQ.0)WRITE(23,*) WSP,STIME
      GOTO 8066
C*****
C* PROPORTIONAL SPRAYER AT MAXIMUM.
C*****
8032 IFPR5=0
      WSP=870
      F(M,ICOUNT)=(DHEAT-HLOST+CONB2*WI+CONB3*WSP)/CONB1
      F(M,ICOUNT+1)=CONC1*(DHEAT-HLOST)+CONC2*WI+CONC3*WSP
      F(M,ICOUNT+2)=0
      RHEAT=0
      IF(IFINE.EQ.0) WRITE(23,*) RHEAT,STIME
      IF(WSP.LT.0.0) WSP=0.0
      IF(IFINE.EQ.0) WRITE(23,*) WSP,STIME
      GOTO 8066
8066 CONTINUE

```

OUTPUT DATA FOR THE HEATER RESPONSE ARE PLACED IN HEATER.DAT
HEATER.DAT
OUTPUT DATA FOR THE SPRAYER RESPONSE ARE PLACED IN SPRYER.DAT
SPRYER.DAT
OUTPUT DATA FOR THE PRESSURE RESPONSE ARE PLACED IN PRESSR.DAT
PRESSR.DAT
OUTPUT DATA FOR THE PRESSURIZER WATER LEVEL RESPONSE ARE IN WHIGHT.DAT
WHIGHT.DAT
OUTPUT DATA FOR THE RESPONSE OF THE DUMMY VARIABLE ARE IN DUMMY.DAT
DUMMY.DAT
PRESSURIZER CROSS SECTION(IN-2)
6361.725124
TOTAL HIGHT IF THE PRESSURIZER VESSEL(IN)
540.0
ENTHALPY OF THE WATER IN THE HOTLEG(BTU/IBM)
672.81
ENTHALPY OF THE WATER LEAVING THE PRESSURIZER(BTU/IBM)
701.1
ENTHALPY OF THE SPRAY WATER(BTU/IBM)
574.36
JOUL'S CONSTANT(IN-IBF)/BTU
9336
COOLANT FLOW RATE IN THE COLD LEG(IBM/SEC)
9398.148
COOLANT FLOW RATE IN THE HOTLEG(IBM/SEC)
9398.148
COOLANT TEMPERATURE AS IT LEAVES THE STEAM GENERATOR(F)
546.2
PRESSURIZER PID HEATER CONTROLLER GAIN(BTU/(SEC-PSI))
-56.811465
PRESSURIZER SPRAYER CONTROLLER GAIN(IBM/(SEC-PSI))
11.6
PID HEATER RATE TIME CONSTANT(SEC)
900.0
PID HEATER RESET TIME CONSTANT(SEC)
1.0
COLD LEG TEMPERATURE AT STEADY STATE(F)
546.2
HOT LEG TEMPERATURE AT STEADY STATE(F)
602.100867
COOLANT TEMPERATURE IN THE CORE UPPER PLENUM AT STEADY STATE(F)
602.100867
PRIMARY WATER VOLUME IN THE HOTLEG PIPING(IBM)
241.9085371
PRIMARY WATER VOLUME IN THE COLDLEG PIPING(IBM)
408.6292856
SLOPE OF THE PRIMARY WATER DENSITY W.R.T TEMPERATURE
-0.0687
HEATER OUTPUT AT STEADY STATE(BTU/SEC)
853.025
HEAT LOST TO THE SURROUNDING AT STEADY STATE(BTU/SEC)
842.557543
SPRAY FLOW RATE AT STEADY STATE(IBM/SEC)
8.259E-02
MASS OF THE COOLANT IN THE HOT LEG(IBM)
3759.25
MASS OF THE COOLANT IN THE COLD LEG(IBM)
6350.1
STEADY STATE PRESSURE IN THE PRIMARY LOOP(PSI)
2250
STEADY STATE WATER LEVEL IN THE PRESSURIZER(IN)

270.0
STEADY STATE VALUE FOR THE VARIABLE DUMMY(PSI-SEC)
0.

```

C*****
C* DETSGR.FOR : A LUMPED PARAMETERS DYNAMIC MODEL FOR A TYPICAL U- *
C* TUBE STEAM GENERATOR SYSTEM. *
C*****
      ICOUNT=IPRIME+1
      IJ=IK+1
      IF(ISTEP.NE.0)GOTO 7039
      OPEN(UNIT=24,FILE='SUE.DAT')
      DO 783 IMARK1=1,225
      READ(24,*) EF(IMARK1)
      EF(IMARK1)=EF(IMARK1)*-1
783  CONTINUE
      DO 784 IMARK1=1,60
      READ(24,*) FORC(IMARK1)
784  CONTINUE
      THLEG=602.1008672
      FR1(IJ)='SGTPI.DAT'
      FR1(IJ+1)='SGTP1.DAT'
      FR1(IJ+2)='SGTP2.DAT'
      FR1(IJ+3)='SGTP3.DAT'
      FR1(IJ+4)='SGTP4.DAT'
      FR1(IJ+5)='SGTPO.DAT'
      FR1(IJ+6)='SGTM1.DAT'
      FR1(IJ+7)='SGTM2.DAT'
      FR1(IJ+8)='SGTM3.DAT'
      FR1(IJ+9)='SGTM4.DAT'
      FR1(IJ+10)='SGLDW.DAT'
      FR1(IJ+11)='SGLS1.DAT'
      FR1(IJ+12)='SGPR.DAT'
      FR1(IJ+13)='SGXE.DAT'
      FR1(IJ+14)='SGTDW.DAT'
      DO 7080 IY=1,IPER
      IF(CHOICE(IY).EQ.'DETSGR.FOR') GOTO 7090
7080 CONTINUE
      GOTO 7039
C*****
C* ENTER THE FORCING TERM. *
C*****
      DO 7020 ICLEAR=1,20
      WRITE(5,*)
7020 CONTINUE
7090 WRITE(5,7021)
      WRITE(5,7000)
      READ(5,*) DTHLEG
7000 FORMAT(1X,'*ENTER THE CHANGE IN THE HOTLEG TEMPERATURE',/,
* 1X,'*DTHLEG='$)
      WRITE(5,7845)
      READ(5,*) DTFW
7845 FORMAT(1X,'*ENTER THE CHANGE IN THE FEEDWATER TEMPERATURE',/,
* 1X,'*DTFW='$)
      WRITE(5,7846)
      READ(5,*) DFWI
7846 FORMAT(1X,'*ENTER THE CHANGE IN THE FEEDWATER FLOW RATE',/,
* 1X,'*DFWI='$)
      WRITE(5,7847)
      READ(5,*) DVALVE
7847 FORMAT(1X,'*ENTER THE CHANGE IN THE STEAM VALVE OPENING',/,
* 1X,'*DVALVE='$)
      WRITE(5,7021)
7021 FORMAT(1X,'*****')
C*****

```

C* EXCHANGE THE VARIABLES. *

C*****

```

7039  SGRTPI=X2(M,ICOUNT)
      SGRTPI=X2(M,ICOUNT+1)
      SGRTPI=X2(M,ICOUNT+2)
      SGRTPI=X2(M,ICOUNT+3)
      SGRTPI=X2(M,ICOUNT+4)
      SGRTPI=X2(M,ICOUNT+5)
      SGRTPI=X2(M,ICOUNT+6)
      SGRTPI=X2(M,ICOUNT+7)
      SGRTPI=X2(M,ICOUNT+8)
      SGRTPI=X2(M,ICOUNT+9)
      SGRLDW=X2(M,ICOUNT+10)
      SGRLS1=X2(M,ICOUNT+11)
      SGRPR=X2(M,ICOUNT+12)
      SGRXE=X2(M,ICOUNT+13)
      SGRTDW=X2(M,ICOUNT+14)
      TSGOP=546.2+SGRTPO
      IPRIME=ICOUNT+14
      IK=IJ+14

```

```

7040  IF(JJ1.EQ.0) GOTO 7041

```

C*****

C* MODEL'S EQUATIONS ARE LISTED BELOW. *

C*****

```

      IV=1
      IB=1
      F(M,ICOUNT)=EF(IV)*SGRTPI+EF(IV+1)*SGRTPI+EF(IV+2)*SGRTPI2
* +EF(IV+3)*SGRTPI3+EF(IV+4)*SGRTPI4+EF(IV+5)*SGRTPO
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALUE
      IV=IV+15
      IB=IB+4
      F(M,ICOUNT+1)=EF(IV)*SGRTPI+EF(IV+1)*SGRTPI+EF(IV+2)*SGRTPI2
* +EF(IV+3)*SGRTPI3+EF(IV+4)*SGRTPI4+EF(IV+5)*SGRTPO
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALUE
      IV=IV+15
      IB=IB+4
      F(M,ICOUNT+2)=EF(IV)*SGRTPI+EF(IV+1)*SGRTPI+EF(IV+2)*SGRTPI2
* +EF(IV+3)*SGRTPI3+EF(IV+4)*SGRTPI4+EF(IV+5)*SGRTPO
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALUE
      IV=IV+15
      IB=IB+4
      F(M,ICOUNT+3)=EF(IV)*SGRTPI+EF(IV+1)*SGRTPI+EF(IV+2)*SGRTPI2
* +EF(IV+3)*SGRTPI3+EF(IV+4)*SGRTPI4+EF(IV+5)*SGRTPO
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALUE
      IV=IV+15

```

```

IB=IB+4
F(M,ICOUNT+4)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+5)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+6)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+7)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+8)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+9)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW
* +FORC(1B+3)*DVALVE
IV=IV+15
IB=IB+4
F(M,ICOUNT+10)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(1B)*DTHLEG+FORC(1B+1)*DWFI+FORC(1B+2)*DTFW

```

```

* +FORC(IB+3)*DVALVE
  IV=IV+15
  IB=IB+4
  F(M,ICOUNT+11)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALVE
  IV=IV+15
  IB=IB+4
  F(M,ICOUNT+12)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALVE
  IV=IV+15
  IB=IB+4
  F(M,ICOUNT+13)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALVE
  IV=IV+15
  IB=IB+4
  F(M,ICOUNT+14)=EF(IV)*SGRTP1+EF(IV+1)*SGRTP1+EF(IV+2)*SGRTP2
* +EF(IV+3)*SGRTP3+EF(IV+4)*SGRTP4+EF(IV+5)*SGRTP0
* +EF(IV+6)*SGRTM1+EF(IV+7)*SGRTM2+EF(IV+8)*SGRTM3
* +EF(IV+9)*SGRTM4+EF(IV+10)*SGRLDW+EF(IV+11)*SGRLS1
* +EF(IV+12)*SGRPR+EF(IV+13)*SGRXE+EF(IV+14)*SGRTDW
* +FORC(IB)*DTHLEG+FORC(IB+1)*DWFI+FORC(IB+2)*DTFW
* +FORC(IB+3)*DVALVE
7041 CONTINUE

```

VITA

Mounir Jamil Mneimneh was born in Beirut, Lebanon on September 24, 1958. His parents are Jamil Mneimneh and former Fatmeh Haddad. He graduated from Rass Al Nabeh High School in 1977. In 1979 he entered the nuclear engineering curriculum at The University of Tennessee and obtained a B.S. degree in Nuclear Engineering in 1982. In 1985 he obtained the M.S. degree in Nuclear Engineering.

Mounir is married to the former Miss Sussane Selinda Case of Greenville, South Carolina and has two children named Jamil and Ahmad.