OPTIMUM SPATIAL FLUX CONTROL PROJECT: PHASE I

MASTER

A
THREE-DIMENSIONAL
NODAL METHOD
FOR
ANALYSIS OF
XENON OSCILLATIONS
AND CONTROL

By
Richard Kern
David C. Wright
W. Carlton Coppersmith
Zoltan R. Rosztoczy

Q & C W.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

OPTIMUM SPATIAL FLUX CONTROL PROJECT: PHASE I

CEXE: A THREE-DIMENSIONAL NODAL METHOD FOR
ANALYSIS OF XENON OSCILLATIONS AND CONTROL

By

Richard Kern
David C. Wright
W. Carlton Coppersmith
Zoltan R. Rosztoczy

Prepared for
THE UNITED STATES ATOMIC ENERGY COMMISSION
Under
Contract No. AT(30-1)3932

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Approved by:

M. F. Valerino
M. F. Valerino, Manager,
Reactor Theory and Analysis Projects

NUCLEAR POWER DEPARTMENT
COMBUSTION ENGINEERING, INC.

Windsor, Connecticut

May 31, 1969

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

pet

EXTERNAL DISTRIBUTION

CEND-3932-1-349

	<u>No. of Copies</u>
USAEC	
Division of Reactor Development & Technology	
Washington, D. C. 20545	
Attention: Chief, HWR Branch (A. N. Tardiff)	1
Chief, Desalting Branch	1
Assistant Director, Plant Engineering	1
Chief, Inst. & Control Branch	1
Assistant Director, Reactor Engineering	1
Chief, Core Design Branch	1
Assistant Director, Reactor Technology	1
Chief, Reactor Physics Branch	1
Assistant Director, Nuclear Safety	1
USAEC	
New York Operations Office	
376 Hudson Street	
New York, New York 10014	
Attention: Leonard S. Hartman	1
Attention: Reports Library	1
USAEC	
New York Patent Group	1
Upton, Long Island, N.Y. 11973	
USAEC	
Division of Technical Information Extension	1
P. O. Box 62	
Oak Ridge, Tennessee 37830	
USAEC	
RDT Site Office	
P. O. Box 550	
Richland, Washington 99352	1
USAEC	
RDT Site Office	
Idaho Operations Office	
P. O. Box 2108	
Idaho Falls, Idaho 83401	1
USAEC	
Oak Ridge National Laboratory	
P. O. Box X	
Oak Ridge, Tennessee 37830	
Attention: RDT Senior Site Representative	1
Director, Studies & Eval. - Reactor Program	1
Dep. Director, Nuclear Desalting	1
Representative, Lucas Heights	1

	<u>No. of Copies</u>
CSAEC Scientific Representative Chalk River Liaison Office c/o Atomic Energy of Canada Limited Chalk River, Ontario, Canada	3
Pacific Northwest Laboratory P. O. Box 999 Richland, Washington 99352 Attention: HWRPO, Manager (H. Harty)	3
E. I. du Pont de Nemours and Company Savannah River Laboratory P. O. Box 117 Aiken, South Carolina 29802 Attention: Manager, Exp. Phy. Research	1
Supervisor, Technical Information Services	1
E. I. du Pont de Nemours and Company Wilmington, Delaware 29802 Attention: Director, Reactor Eng. Section	1
The Babcock & Wilcox Company Atomic Energy Division P. O. Box 1260 Lynchburg, Virginia 24505 Attention: HWR Project Manager	1
Mr. M. F. Ross	1
Massachusetts Institute of Technology Cambridge, Massachusetts 02139 Attention: Head, Dept. of Nuclear Engineering	1
Phillips Petroleum Company Atomic Energy Division P. O. Box 2067 Idaho Falls, Idaho 83401 Attention: Manager, Water Reactors Safety	1
Dr. C. H. Millar Atomic Energy of Canada Limited Chalk River Nuclear Laboratory Chalk River, Ontario, Canada	1
Dr. J. L. Crandall E. I. du Pont de Nemours & Company Savannah River Laboratory Aiken, South Carolina 29801	1

No. of Copies

r. Liberty Pease
Atomic Energy of Canada Limited
P. O. Box 905
Toronto 18, Ontario, Canada

1

Mr. L. C. Schmid
Battelle Memorial Institute
Pacific Northwest Laboratory
P. O. Box 999
Richland, Washington 99352

1

Mr. A. Pasanen
Atomic Energy of Canada Limited
Sheridan Park, Ontario, Canada

1

Mr. M. J. O'Boyle
Westinghouse Electric Corporation
Atomic Power Division
P. O. Box 355
Pittsburg, Penn. 15230

1

Dr. A. M. Perry
Oak Ridge National Laboratory
P. O. Box X
Oak Ridge, Tennessee 37830

1

Dr. Weston M. Stacey, Jr.
General Electric Company
Knolls Atomic Power Laboratory
P. O. Box 1072
Schenectady, New York 12301

1

Mr. P. S. Lacy
Nuclear Associates International
12601 Twin Brook Parkway
Rockville, Maryland 20852

1

Mr. R. L. Haueter
Consumers Power Company
212 West Michigan Avenue
Jackson, Michigan 49201

1

Dr. S. Turner
Southern Nuclear Engineering
P. O. Box 10
Dunedin, Florida 33528

1

TABLE OF CONTENTS

<u>Section No</u>		<u>Page No</u>
I.	INTRODUCTION AND SUMMARY	1
	A. Purpose	1
	B. Code Description and Summary	1
II.	BASIC METHODS AND FEATURES	4
	A. Nodal Source Method	4
	1. Description of Method	4
	2. Convergence Scheme	5
	B. Feedback Features	5
	1. Calculation of Power and Thermal Flux	5
	2. Moderator Temperature Effects	6
	3. Doppler Effects	7
	4. Xenon Concentration Effects	7
	5. Calculation of the Xenon 135 Absorption Cross Sectional Dependence on Moderator Temperature	8
	C. Control Features	8
	1. Soluble Poison Concentration	8
	2. Control Rod Simulation	9
	D. General Features	9
	1. Node and Node Size Limitations	9
	2. Transport Kernels	9
	3. Convergence Criteria	10
	4. Single or Bidirectional Flow	11
	5. Load Following	12
III.	MATHEMATICAL MODEL	13
	A. Neutron Source and Transport Kernel Calculations	13
	B. Infinite Multiplication Factor Calculation	16
	1. Reference Value	17
	2. Doppler and Moderator Temperature Effects	17
	3. Control Effect	18
	4. Soluble Poison Effect	18
	5. Eigenvalue Search Effect	18
	6. Xenon Effect	19

TABLE OF CONTENTS (Cont.)

<u>Section No.</u>	<u>Page No.</u>
C. Migration Area Calculation	20
1. Reference Value	20
2. Doppler and Moderator Temperature Effects	21
3. Control, Soluble Poison, Xenon, and Eigenvalue Search Effects	21
D. Moderator Temperature Calculations	21
1. Specific Enthalpy Calculations	22
2. Moderator Temperature Calculations	22
E. Thermal Flux and Power Calculations	23
1. Thermal Flux Calculation	23
2. Power Calculation	24
F. Iodine and Xenon Concentration Calculations	24
1. Iodine Calculations	25
2. Xenon Calculations	25
IV. APPLICATION AND LOGISTICS OF CEXE	26
A. Description of INCXE	26
B. Steady - State Calculations	27
C. Perturbations	28
D. Restart Features	28
E. Time Dependent Input Features	28
F. Overall Program Execution Logic	29
1. New Problems	29
2. Restart Problems	30
G. Subroutine Descriptions	31
1. Subroutine Main	31
2. Subroutine INITAL	31
3. Subroutine INPUT1	31
4. Subroutine INPUT2	32
5. Subroutine OUTIN	32
6. Subroutine TEMPER	32
7. Subroutine SOURCE	33
8. Subroutine CONVRG	34

TABLE OF CONTENTS (Cont.)

<u>Section No.</u>		<u>Page No.</u>
	9. Subroutine NWTIME	34
	10. Subroutine XENON	35
	11. Subroutine NWSRCH	35
	12. Subroutine SORCIO	35
	13. Subroutine KANDM2	35
	14. Subroutine OUTPUT	36
	15. Subroutine STORE	36
	16. Subroutine FNTERP	36
	17. Subroutine POWER	36
	H. Estimates of Computational Times	36
V.	INPUT DESCRIPTIONS	38
	A. Input Description for INCEXE	38
	B. Input Description for CEEXE	42
	1. INPUT1 Description	43
	2. INPUT2 Description	52
VI.	OUTPUT DESCRIPTIONS	61
	A. Output Description for INCEXE	61
	B. Output Description for CEEXE	61
	1. New Problem	62
	2. Restart Problem	63
VII.	FORTRAN IV LISTINGS	64
	A. FORTRAN Listing for INCEXE	64
	B. FORTRAN Listing for CEEXE	68
VIII.	REFERENCES	100
	FIGURES	
	APPENDICES	
	A. Derivation of an Alternate Transport Kernel	A-1
	B. IBM-360 Hardware and Software Requirements for CEEXE	B-1
	C. IBM-360 Job Control Language Setups for INCEXE and CEEXE	C-1
	D. Logical I/O Units for CEEXE	D-1

LIST OF ILLUSTRATIONS

Figure No.

Title

Following Page 100

1. Simplified Flow Diagram of the Logistics of the
CEXE Code Calculations
2. Detailed Flow Diagram of the Logistics of the CEXE
Code
(Sheet 1 of 5)
(Sheet 2 of 5)
(Sheet 3 of 5)
(Sheet 4 of 5)
(Sheet 5 of 5)
3. Input Listing for a Sample INCEXE Problem
4. Input Listing for a Sample CEXE Problem
(Sheet 1 of 5)
(Sheet 2 of 5)
(Sheet 3 of 5)
(Sheet 4 of 5)
(Sheet 5 of 5)
5. Output from a Sample INCEXE Problem
(Sheet 1 of 6)
(Sheet 2 of 6)
(Sheet 3 of 6)
(Sheet 4 of 6)
(Sheet 5 of 6)
(Sheet 6 of 6)
6. Output from a Sample CEXE Problem
(Sheet 1 of 4)
(Sheet 2 of 4)
(Sheet 3 of 4)
(Sheet 4 of 4)

I. INTRODUCTION AND SUMMARY

A. PURPOSE

Because of economic considerations, the current designs for commercial power reactors require: (1) high power densities and low fuel enrichments; (2) low peak-to-average power distributions; and (3) high total thermal power output. The first of these requirements results in a high thermal neutron flux level; the second leads to flattened power shapes; and the last encourages the design of large reactors. Each of these conditions leads toward reactors in which it is possible to sustain xenon-induced spatial power oscillations.

For such reactor systems it is necessary to design a detection and control system that will minimize deviations from the design power distribution in order to maintain the required thermal margins at rated power. The need for a relatively fast three-dimensional method for performing design analyses provided the incentive for the Optimum Spatial Flux Control Project.

The objectives of this project were: Phase I, to develop a simple three-dimensional calculational method which can be used to study xenon oscillations and control; Phase II, to test the calculational model by comparing the results obtained from it against those obtained by one- and two-dimensional diffusion theory methods and to study several selected problems associated with the Pickering Reactor (Pickering is a heavy water moderated and cooled Canadian reactor). This report covers the results of the efforts carried out under Phase I above, while Reference 1 describes the Phase II work.

B. CODE DESCRIPTION AND SUMMARY

A basic objective in the development of CEXE was to develop a code that is practical for use in design studies; consequently, techniques were used to shorten computational times to a minimum without compromising on the maximum number of nodes or on the representations of the desired feedback mechanisms. To facilitate the use of CEXE, another code, INCEXE, was written for the purpose of preparing a portion of the CEXE input. Both codes are written in FORTRAN IV for the IBM 360/65 machine but can be converted to other computers.

In contrast to commonly used multigroup diffusion theory techniques, the CEXE Code is based on modified one-group theory for the calculation of the neutron source distribution. The equivalent one-group constants which are used in this theory are based on two-group considerations. This method is justified since in most power reactors the local fast-to-thermal flux ratio does not change appreciably during xenon oscillations. Holding this flux ratio constant at its initial value provides a scheme for computing the thermal flux and power distributions from the one-group source distribution. The one-group approach is further justified since the fast and thermal neutron leakages are accounted for properly through judicious selection of input leakage and transport parameters. Computational simplicity is further enhanced by employing nodal analysis (Reference 2) rather than a finite difference method. Nodal analysis has the advantage of yielding reasonably accurate results for a relatively small number of nodes in contrast to the large number of mesh points required in the finite difference method.

In addition to the above techniques, major feedback mechanisms associated with xenon oscillations (spatial effects due to local Doppler, local moderator temperature, and local xenon concentration) are explicitly represented in CEXE. The code also allows for the simulation of arbitrary control rod position changes during a transient. Core criticality can be maintained with the use of an automatic homogeneous thermal poison search. The code is also designed to simulate load-following.

Figure 1 shows a simplified flow diagram of the logistics used in CEXE to perform the power distribution and feedback calculations.

Reference 1 discusses the results of comparisons between diffusion theory and CEXE calculations for one- and two-dimensional transient problems and the three-dimensional equilibrium problem for the Pickering Reactor. References 3 and 4 discuss the comparisons between diffusion theory and CEXE calculations for one-dimensional transient problems for both long and short time step intervals for a pressurized water reactor. The good agreement that was found between the diffusion theory and CEXE calculations is encouraging and recommends the code as a practical design tool since: (1) one- and two-dimensional methods may not be adequate for certain problems as pointed out

in Reference 1, and (2) computing time is greatly reduced relative to three-dimensional finite difference solutions.

It should be pointed out that although CEXE has been developed and tested primarily for heavy water reactors, the code has the necessary flexibility for analyzing other types of reactors including pressurized light water reactors; the code can be adapted to handle boiling light water reactors.

II. BASIC METHODS AND FEATURES

This section outlines and describes the basic methods and features that have been programmed into CEXE. Section III describes the mathematical foundations of the methods and features.

A. NODAL SOURCE METHOD

The nodal source method is the basis for calculating three-dimensional flux and power distributions in CEXE. This method is essentially the one employed in the FLARE Code as described in Reference 2.

1. Description of Method

The reactor core is divided into a finite number of nodes in XYZ geometry and the source distribution is computed at each node by a modified one-group theory method. The formulation of this method consists of expressing the multiplication properties within each node in terms of k_{∞} (the infinite multiplication factor) and the leakage properties between adjacent nodes in terms of a transport kernel which depends directly on M^2 (the migration area). Net leakage out of peripheral nodes is specified by albedo values at the boundary surfaces of these nodes. The albedos are input quantities that are determined by a trial-and-error scheme in which the albedos are varied until a reference power distribution, obtained from either a detailed diffusion theory calculation or from experimental data, is matched. These statically fitted values are then held constant for the transient calculations.

By assuming that the fast-to-thermal flux ratio at each node remains constant during the transient, it is possible to calculate the thermal flux and power distributions from the source distribution. This assumption is justified since in most power reactors the fast-to-thermal flux ratio at a given node does not change significantly during xenon oscillations, especially when the oscillations are controlled to avoid large deviations from the design power distribution. Because of the validity of this assumption, the modified one-group theory will yield correct fast and thermal flux leakages during the transient.

2. Convergence Scheme

Because of the sensitivity of the xenon oscillation phenomenon to the power distribution, CEXE employs a stringent convergence criterion. The convergence test that is applied to the source calculations consists of: (1) comparing the source calculated during a specific iteration to the source computed in the previous iteration at each node in the reactor, and (2) comparing the eigenvalues of these two iterations. For convenience, CEXE has been programmed so that a maximum number of iterations per source calculation can be specified.

B. FEEDBACK FEATURES

CEXE includes the major feedback effects associated with xenon oscillations. Both the spatial and time dependence of each feedback effect is included explicitly.

1. Calculation of Power and Thermal Flux

In order to properly represent the physical processes that are involved in xenon oscillations, it is necessary to compute the three-dimensional power and thermal flux distributions in CEXE. The power distribution is used in the Doppler and moderator temperature feedbacks as well as in the iodine and xenon equations. The thermal flux distribution is only required for the iodine and xenon concentration calculations.

As pointed out in the previous section, CEXE computes the neutron source distribution rather than power or neutron flux. Thus it is necessary to be able to determine these latter distributions from the source distribution. An analysis was undertaken to determine the most accurate technique for accomplishing this (Reference 5). The results showed that the local fast-to-thermal neutron flux ratio remains nearly constant throughout the entire core during a xenon oscillation. This assumption is not valid for those nodes in which the control poison is varied and thus provisions have been made in the input to allow for changes in this variable when the control poison is changed. In order to perform these calculations, CEXE requires the fast and thermal macroscopic fission and nu-fission cross sections in each node, and these are supplied as part of the input which is prepared by the INCXE Code (refer to

section IV.A for a description of INCEXE). The magnitudes of nodal power and thermal flux are obtained through normalization procedures which preserve the total core power.

2. Moderator Temperature Effects

For the calculation of the average moderator temperature within each node, the temperature difference between the centers of two vertically adjacent nodes is calculated from the enthalpy rise and the specific heat of the moderator at constant pressure. The value of the temperature at the center of a node is then calculated by summing the temperature differences up the channel* to that node. It is assumed that the average moderator temperature of a node is given by the temperature at the center of the node. Allowances have been made in the code for variation of the specific heat with temperature. Provisions have also been made for bidirectional flow (refer to section II.D.4). The inlet coolant temperatures are assumed to be the same for all channels with coolant flow in the same direction. Provisions, however, have been made to allow for orificing of the channel flow rates. For pressure tube type reactors, such as Pickering⁽¹⁾, the moderator temperature effect is associated with changes in the coolant temperature.

The relationship between " Δk_{∞} versus moderator temperature" and " ΔM^2 versus moderator temperature" can be specified by tabular input or by a linear relationship (e.g., a multiplier with units of $\Delta k_{\infty}/^{\circ}\text{F}$). Tabular input allows a more accurate representation for specification over a broader range of the independent variable. The use of this feedback mechanism is optional. In addition, an option has been provided whereby, if the user feels that it is not necessary to update the moderator temperature feedback at each time, the effect can be held constant over any number of time steps that are desired. When this option is exercised, the power distribution that is used to compute changes in the xenon and iodine concentrations during a time step still corresponds to the latest source distribution. It must be noted that if the similar option is used to keep the Doppler feedback effects constant (see II.B.3 below), then the moderator temperature effects must also be kept constant. As noted in the Introduction and Summary, CEXE can be used for a variety of reactor types. As an example, the application of CEXE to boiling water reactors

* A channel is defined as all the nodes which have the same XY(I,J) location coordinates.

could be made by redefining (through the proper choice of input parameters) the moderator temperature to be the coolant enthalpy and then inserting tables of Δk_{∞} and ΔM^2 as functions of enthalpy rather than as function of the moderator temperature. If this is done, however, the moderator temperature dependence of the Xenon 135 absorption cross section (as described in II.B.5 below) cannot be used. This technique and others similar to it have not been tested.

3. Doppler Effects

The Doppler effect is represented by assuming that the changes in k_{∞} and M^2 within each node are proportional to the local power density. The effects are calculated in the same manner as the moderator temperature effects, i.e., either from a set of tables or by using linear coefficients. The use of this feedback mechanism is optional. An option similar to the one provided for the moderator temperature effects is available to suppress the updating of the Doppler feedback effect.

4. Xenon Concentration Effects

The iodine and xenon equations are solved simultaneously at each node in the core. The source term in each equation, which represents the direct production from the fission process, is proportional to the total nodal power. In solving the iodine and xenon equations, it is assumed that the power and thermal flux distributions remain constant during each time step interval and the values that are used correspond to the converged distributions at the beginning of the time step. Hence, the iodine and xenon equations are programmed in their integrated forms using the constant power and thermal flux assumptions. The use of these assumptions will lead to errors, which are dependent on the time step length, in the computed stability index and period for oscillations with no control action; however the similarity between CEXE and diffusion theory (Reference 1) indicates that the procedures used to correct diffusion theory results (Reference 6) can also be applied to CEXE results. The error introduced by the constant power and flux assumption should be less severe for problems in which control studies are being investigated, where deviations from the steady state power distribution are minimized.

5. Calculation of the Xenon 135 Absorption Cross Sectional Dependence on Moderator Temperature

In pressured water reactors, the Xenon 135 average thermal absorption cross section has a significant dependence on the average moderator temperature (Reference 7). In CEXE, the dependence of the xenon cross section on the moderator temperature is assumed to be the same in all regions of the core, since the dependence on fuel enrichment and exposure is negligible. The functional dependence is represented by a quadratic fit between the cross section and the average local moderator temperature.

C. CONTROL FEATURES

There are two types of control available in CEXE and each is independent of the other.

1. Soluble Poison Concentration

There are two types of soluble poison concentrations in CEXE. The first type corresponds to the value used as part of the input specification. It is used to represent the poison concentration which actually exists in the reactor. The effects on k_{∞} and M^2 are treated explicitly.

The remaining soluble poison concentration represents the variable on which the criticality search can be made, which is optional. In a transient calculation, changes in the xenon distribution and in control rod configurations (which are discussed in the following section) will result in changes in the core eigenvalue. Hence CEXE has been provided with an automatic feedback search whereby the code under requestor option will calculate the value of the soluble poison macroscopic absorption cross section required to maintain a constant core eigenvalue. This constant eigenvalue is defined as the reference value.

In utilizing this homogeneous thermal poison option, two possibilities are available for defining the reference value. The first is to input the desired eigenvalue. The second is to allow the code to calculate the equilibrium eigenvalue which is then used as the reference value during the transient (refer to section IV.B for the description of the equilibrium calculations).

In using the poison search option, the updating of the macroscopic poison

cross section is performed after each source iteration by relating the desired change in eigenvalue to a change in cross section through the core volume weighted average value of (ρ_{nf2}/Σ_{a2}) . These symbols are defined in III.B.4.

2. Control Rod Simulation

The second type of control feature available in CEXE is the simulation of control rods and/or local poison absorber strengths. Arbitrary changes in k_{∞} and M^2 within each node can be specified as a function of time as part of the input at each time point in the transient calculations. No options have been provided for control rod position or strength searches. Thus, in regard to the poison location, the control rod simulation is highly flexible but the control rod changes must be specified prior to running the problem.

D. GENERAL FEATURES

1. Node and Node Size Limitations

The maximum node configuration which is permissible in CEXE is $19 \times 19 \times 10$ for the X,Y,Z directions, respectively. The node length for the X and Y dimensions must be identical but the Z length may be different.

The nodal configuration is usually chosen on the basis of the core loading pattern (such as in Reference 1) in which each XY node represents one, four, or more fuel channels or fuel assemblies. This, however, is not a restriction of the code. Satisfactory results have been found in comparisons between fine-mesh diffusion theory and CEXE transient calculations in which the nodal dimensions were in the range of 3 to 8 migration lengths (References 1, 3 and 4).

2. Transport Kernels

There are three different transport kernels available for use in the source calculations in CEXE. Two of these are identical to those formulated in Reference 2 and the third one is a new type that was developed during the course of this project (see Appendix A). It is possible to use an arbitrary linear combination of the two FLARE-type kernels for calculation of the horizontal leakage probabilities and a different linear combination for the vertical direction. If the new type kernel is used for the calculation of

horizontal leakages then it must also be used for the calculation of the vertical leakages, and vice versa. For computing the leakage probabilities with any of these kernels, it is possible to specify two arbitrary multipliers for the kernel, one for the calculation of the horizontal and another for the calculation of the vertical leakages. These multipliers have been found at times to be advantageous in forcing the nodal calculations to match diffusion theory results. The type of kernel, the multiplier, and the albedo values for peripheral nodes are selected by matching a reference neutron source distribution, obtained either from fine-mesh diffusion theory calculations or experimental data.

The specification of the horizontal (XY) and vertical (Z) albedo values, which govern the calculated net leakages from the peripheral nodes, is limited in the following manner. For a particular (XY) location at the boundary, the same horizontal albedo value is applied for all Z locations. The vertical values are specified by the use of one value for all of the bottom nodes and one value for all of the top nodes. This amount of flexibility has proven to be adequate for the Pickering Reactor on the basis of the good agreement that was found between the three-dimensional diffusion theory and CEXE results for the equilibrium thermal flux distribution.

3. Convergence Criteria

As noted previously in section II.A.2, the source convergence test consists of checking both the source distribution at each node and the core eigenvalue. However, in running a new CEXE problem (see section IV.B for the definition of a new problem) the code is required to calculate the equilibrium core conditions before introducing the perturbation. The convergence test for the equilibrium core condition is made on the nodalwise source distribution (for further discussions see section IV.B).

For the homogeneous thermal poison search option, CEXE checks the convergence of the value of the homogeneous poison cross section by comparing the core eigenvalue to the reference eigenvalue. The absolute value of the difference between these two numbers must be less than the input convergence criterion.

The iterated nodal method for computing source distribution may lead to nodal sources which oscillate about some mean value after a certain number

of iterations. This causes a reduction in the rate of convergence that could add a large number of iterations to the calculations in order to obtain convergence. CESE has been programmed with an option that accelerates the convergence in this case; however, experience has shown that care must be taken if the technique is implemented (Reference 3). The essence of this technique is the use of a modified source guess. In the course of the usual iteration scheme, the source guess for the (i)th iteration is the computed source distribution from the (i-1)th iteration. The modified source guess consists of averaging the value of the source from the (i-1)th iteration and its corresponding source guess and then using this value as the source guess for the (i)th iteration. The usual iteration scheme is employed until the modified procedure is invoked. The modified procedure does not take over until the code exceeds a certain number of iterations which is an input variable. Hence, the user must know at which iteration the calculational oscillations will begin before running the problem in order to use this option properly. The use of the option prior to this time will decrease the rate of convergence. Once the modified procedure begins, source convergence is checked at the end of each iteration by comparing the source computed in the iteration with its corresponding guess rather than with the source distribution from the previous iteration (see section IV.G.7).

In connection with the inclusion of the feedback mechanisms, the Doppler feedback, the moderator temperature feedback and the homogeneous thermal poison search effects are part of the source iteration loop. This means that these feedback effects are updated during each source iteration in order to eliminate the need to solve for a converged source distribution and then update the feedback effects. Results of test cases have shown that this is a fast means for obtaining a totally converged solution (Reference 8).

4. Single or Bidirectional Flow

An option has been provided in which bidirectional flow can be specified (this option allows simulation of the Pickering Reactor coolant system). Arbitrary specification for upward and downward (i.e., either Z - direction) flow channels is available. In addition, the inlet coolant temperatures at the top and bottom of the core, which are assumed constant across the core

cross section, can be specified independently. The option is only used in connection with the moderator temperature feedback effects and the code calculates the proper nodal moderator temperature by summing the moderator temperature differences either up or down the channel depending on the coolant flow direction.

5. Load Following

In the present context, the load following simulation capability of the CEXE Code means that the code can handle a problem in which the core power level is a variable. However, this type of load following is restrictive in the following sense. CEXE has been set up so that it is possible to change the total core power, the average coolant flow rate per channel, the inlet moderator temperature, the control rod positions, and the soluble poison concentration at any time point in the problem. Hence, when simulating power level changes, these system parameters can be changed by manual input and must be known prior to running the problem from some other source of information. Once these system parameters have been changed, the code will automatically compute the proper changes in the Doppler feedback, the moderator temperature feedback, and the xenon and iodine concentrations as the problem progresses. It should be noted again at this point that CEXE has no provision for performing control rod position searches and that a criticality search can only be made on the homogeneous thermal poison.

III. MATHEMATICAL MODEL

This section covers the mathematical foundations of the model programmed into the CESE Code and describes the equations used in the code.

A. NEUTRON SOURCE AND TRANSPORT KERNEL CALCULATIONS

The details of the basic theory of the nodal model for an effective one-group theory are discussed in Reference 2. The main points of the model are summarized below. The initial assumption is that the neutron source in node $\ell = (ijk)$ is given by

$$S_{\ell} = k_{\infty\ell} A_{\ell} / \lambda$$

where $k_{\infty\ell}$ is the infinite multiplication factor in node ℓ , A_{ℓ} is the absorption rate in node ℓ , and λ is the core eigenvalue.

Defining the probability that a neutron born in node ℓ is absorbed in node m as $W_{\ell m}$ (from now on, $W_{\ell m}$ will be denoted as the transport kernel), we get

$$A_{\ell} = \sum' S_m W_{m\ell} + S_{\ell} W_{\ell\ell}$$

Using the assumption that only six nearest neighbors interact, the summation is always taken only over these six nodes which is indicated by the prime on the summation symbol.

Thus we can write

$$S_{\ell} = \frac{\frac{k_{\infty\ell}}{\lambda} \sum' S_m W_{m\ell}}{1 - \frac{k_{\infty\ell}}{\lambda} W_{\ell\ell}}$$

This is the fundamental equation which is solved for interior points. At the periphery of the core, account must be taken for neutrons returning from the

reflectors. The net neutron leakage probability from the exposed surfaces in a peripheral node is given by

$$\mathcal{L}_\ell = W_{\ell m}^H (n_\ell^H - \alpha_\ell^H) + W_{\ell m}^V (n_\ell^V - \alpha_\ell^V)$$

where n_ℓ is the number of exterior faces and α_ℓ is the albedo. Note that the specification of a pure reflecting external surface is accomplished by specifying $\alpha = 1.0$ for that surface. The superscripts H and V refer to the horizontal and vertical direction, respectively. Since the nonleakage probability is $W_{\ell\ell}$, we obtain

$$W_{\ell\ell} = 1 - \mathcal{L}_\ell - (4 - n_\ell^H) W_{\ell m}^H - (2 - n_\ell^V) W_{\ell m}^V$$

where the last two terms represent the leakages to the adjacent nodes. This equation can be reduced to

$$W_{\ell\ell} = 1 - (4 - \alpha_\ell^H) W_{\ell m}^H - (2 - \alpha_\ell^V) W_{\ell m}^V$$

Thus the most general expression for the source becomes

$$S_\ell = \frac{\frac{k_{\infty\ell}}{\lambda} \sum_m S_m W_{m\ell}}{1 - \frac{k_{\infty\ell}}{\lambda} \left[1 - (4 - \alpha_\ell^H) W_{\ell m}^H - (2 - \alpha_\ell^V) W_{\ell m}^V \right]}$$

The eigenvalue is obtained by a simple neutron balance over the core and is given by

$$\lambda = \frac{\sum_\ell S_\ell - \sum_\ell S_\ell \left[W_{\ell m}^H (n_\ell^H - \alpha_\ell^H) + W_{\ell m}^V (n_\ell^V - \alpha_\ell^V) \right]}{\sum_\ell \frac{S_\ell}{k_{\infty\ell}}}$$

where each of the summations is over all nodes in the core.

In order to compute an appropriate W_{lm}^H and W_{lm}^V at each node in the reactor, two types of kernels have been developed for the FLARE code (Reference 2).

They are

$$W_{lm}^{(i)}(1) = \frac{\sqrt{M_l^2}}{2\Delta X^{(i)}} \quad \text{and} \quad W_{lm}^{(i)}(2) = \frac{M_l^2}{\left[\Delta X^{(i)}\right]^2}$$

where $i = H, V$ and $\Delta X^{(H)} = \Delta X$ and $\Delta X^{(V)} = \Delta Z$; ΔX and ΔZ are the lengths of the node in the horizontal and vertical directions, respectively. These two kernels can be mixed together by specification of a mixing parameter $g^{(i)}$ and the total kernel is given by

$$W_{lm}^{(i)} = (1 - g^{(i)}) W_{lm}^{(i)}(1) + g^{(i)} W_{lm}^{(i)}(2)$$

with $i = H, V$

This last expression has been adopted for CEXE with an additional modification. The results of a node size study conducted during this project (refer to Reference 7 for example) indicate that in order to obtain satisfactory matches in power distributions between CEXE and diffusion theory calculations it may be necessary to introduce a multiplier on the kernels. Hence the expression for the kernels in CEXE is

$$W_{lm}^{(i)} = \mu^{(i)} \left[(1 - g^{(i)}) W_{lm}^{(i)}(1) + g^{(i)} W_{lm}^{(i)}(2) \right]$$

with $i = H, V$

In addition to this transport kernel, a new kernel has been developed during the course of this project. The derivation of this kernel is given in Appendix A and the final result is

$$W_{lm}^H = \frac{g^H P_l}{4 + 2 \left(\frac{\Delta X}{\Delta Z} \right)}$$

$$W_{lm}^V = g^V W_{lm}^H \left(\frac{\Delta X}{\Delta Z} \right)$$

where

$$P_{\ell} = \frac{4\pi}{\kappa_{\ell}^3 V} (1 + R'_{\ell}) \exp \{-R'_{\ell}\} \sinh R'_{\ell} [R'_{\ell} \coth R'_{\ell} - 1]$$

with

$$\kappa_{\ell} = 1/M_{\ell} \quad R'_{\ell} = \kappa_{\ell} R$$

and

$$V = \frac{4\pi}{3} R^3 = (\Delta X)^2 \Delta Z$$

This last kernel has also been programmed into CEXE and it should be noted that here g^V and g^H play the role of μ^V and μ^H of the former kernels. Because of the definitions used for the kernels, care must be taken when specifying the values of g^H and g^V when these parameters are selected on the basis of decoupled one- and two-dimensional calculations.

Thus once the proper physics parameters (k_{∞} and M^2) and the appropriate leakage parameters (α , $g^{(i)}$ and $\mu^{(i)}$) are specified, the neutron source distribution can be found by iterative techniques. This is exactly the prescription used in CEXE; however, the source distribution that is used as a guess for a particular iteration may be either the source from the previous iteration or the average of two source distributions, as was explained in section II.D.3.

B. INFINITE MULTIPLICATION FACTOR CALCULATION

In order to compute the appropriate values of the physics parameters in each node accounting for all feedback effects, the following equations are used. For k_{∞} we have at time = t:

$$\begin{aligned} [k_{\infty}(t)]_{\ell} = & (k_{\infty} \text{ reference})_{\ell} + \{\Delta k_{\infty} [P_{\ell}(t)]\}_{\text{Doppler}} \\ & + \{\Delta k_{\infty} [T_{\ell}(t)]\}_{\text{Moderator}} \\ & + [\Delta k_{\infty}(t)]_{\text{control},\ell} + [\Delta k_{\infty}(t)]_{\text{HTP},\ell} \\ & + [\Delta k_{\infty}(t)]_{\text{search},\ell} + [\Delta k_{\infty}(t)]_{\text{xenon},\ell} \end{aligned}$$

Each term in this expression is discussed in the following paragraphs 1 through 6.

1. Reference Value

$(k_{\infty} \text{ reference})_{\ell}$ is the reference value as input from the INCXE tape (section IV.A). These values must correspond to the reference nodal power and reference moderator temperature which are compatible with the tables of changes in k_{∞} with nodal power and moderator temperature as noted in paragraph 2 below. It should also be noted that the reference values correspond to the xenon-free condition since the code automatically accounts for the xenon effect.

2. Doppler and Moderator Temperature Effects

$\{\Delta k_{\infty} [P_{\ell}(t)]\}_{\text{Doppler}}$ and $\{\Delta k_{\infty} [T_{\ell}(t)]\}_{\text{Moderator}}$ are the changes in k_{∞} due to the Doppler and moderator temperature feedback effects, respectively. These parameters are calculated from one of the following two prescriptions. An option has been provided whereby the Doppler and moderator temperature effects can be specified as being linear functions of the nodal power and moderator temperature, respectively. When this option is implemented, the following equations are used in CEXE:

$$\{\Delta k_{\infty} [P_{\ell}(t)]\}_{\text{Doppler}} = \left(\frac{\partial k_{\infty}}{\partial P} \right) [P_{\ell}(t) - P_{\text{reference}}]$$

where $\left(\frac{\partial k_{\infty}}{\partial P} \right)$ is the input power coefficient and $P_{\text{reference}}$ is the input reference nodal power level.

$$\{\Delta k_{\infty} [T_{\ell}(t)]\}_{\text{Moderator}} = \left(\frac{\partial k_{\infty}}{\partial T} \right) [T_{\ell}(t) - T_{\text{reference}}]$$

with $\left(\frac{\partial k_{\infty}}{\partial T} \right)$ and $T_{\text{reference}}$ being the input values of the temperature coefficient and reference temperature, respectively. $P_{\ell}(t)$ and $T_{\ell}(t)$ are the power and average moderator temperature in node ℓ at time t .

If this option is not used, then CEXE performs interpolations and extrapolations in a set of Δk_{∞} versus nodal power and Δk_{∞} versus moderator temperature tables which are input from the INCXE tape.

It should be noted that all input variables which are used to obtain the Doppler and moderator temperature effects must be consistent with the input specifications for the reference k_{∞} values.

3. Control Effect

$[\Delta k_{\infty}(t)]_{\text{control},\ell}$ is the input value for nodalwise external control which corresponds to the control rod or local poison absorbers simulation. These values may be changed at any node and at any time point in the transient calculations.

4. Soluble Poison Effect

$[\Delta k_{\infty}(t)]_{\text{HTP},\ell}$ is the term which corresponds to the actual homogeneous thermal poison concentration in the core which is part of the system input specification. This effect is computed from the formula:

$$[\Delta k_{\infty}(t)]_{\text{HTP},\ell} = -(\text{pnf}_2)_{\ell} \cdot \frac{\beta_{\ell}(t)}{[1 + \beta_{\ell}(t)]}$$

where

$$\beta_{\ell}(t) = \frac{\Sigma_{a_2}^{\text{HTP}}(t)}{(\Sigma_{a_2})_{\ell}} \quad \text{and} \quad (\text{pnf}_2)_{\ell} = \frac{(\Sigma_{r_1})_{\ell}}{(\Sigma_{r_1})_{\ell} + (\Sigma_{a_1})_{\ell}} \cdot \frac{(\nu \Sigma_{f_2})_{\ell}}{(\Sigma_{a_2})_{\ell}}$$

$(\Sigma_{r_1})_{\ell}$ and $(\Sigma_{a_1})_{\ell}$ are the node average values of the fast neutron removal and absorption cross sections, respectively, in node ℓ . $(\nu \Sigma_{f_2})_{\ell}$ and $(\Sigma_{a_2})_{\ell}$ are the node average values of the thermal neutron nu-fission and absorption cross sections, respectively, in node ℓ . These cross sections must correspond to the xenon-free reference nodal power and moderator temperature and are in units of cm^{-1} . $\Sigma_{a_2}^{\text{HTP}}$ is the input value of the homogeneous thermal poison cross section in units of cm^{-1} . Σ_{a_2} and pnf_2 are input from the INCEXE tape.

5. Eigenvalue Search Effect

$[\Delta k_{\infty}(t)]_{\text{search},\ell}$ is the parameter which results from forcing the core eigenvalue to remain constant at some reference value.

$$[\Delta k_{\infty}(t)]_{\text{search},\ell} = -(\text{pnf}_2)_{\ell} \cdot \frac{\gamma_{\ell}(t)}{[1 + \gamma_{\ell}(t)]}$$

where

$$\gamma_{\ell}(t) = \frac{\Sigma_{a_2}^{\text{search}}(t)_{\ell}}{(\Sigma_{a_2})_{\ell}}$$

$\Sigma_{a_2}^{\text{search}}(t)$ is the node averaged absorption cross section of the homogeneous thermal poison and is found by iteration. The value used for iteration K is computed by the following formula:

$$\left[\Sigma_{a_2}^{\text{search}}(t) \right]^{(K)} = \left[\Sigma_{a_2}^{\text{search}}(t) \right]^{(K-1)} + \frac{\left[\lambda(t) \right]^{(K-1)} - \lambda_{\text{reference}}}{\left(\frac{\text{pnf}_2}{\Sigma_{a_2}} \right)_{\text{average}}}$$

where $\lambda^{(K-1)}(t)$ is the eigenvalue of the K-1 iteration at time t and $\lambda_{\text{reference}}$ is the reference eigenvalue.

$$\left(\frac{\text{pnf}_2}{\Sigma_{a_2}} \right)_{\text{average}} = \frac{\sum_l \frac{(\text{pnf}_2)_l}{(\Sigma_{a_2})_l}}{N}$$

where N is the total number of nodes in the core and the summation is over the entire core.

Note that $\left[\Sigma_{a_2}^{\text{search}}(t) \right]^{(0)}$ which is the value used for the first iteration for a given time, t, is given by the equation

$$\left[\Sigma_{a_2}^{\text{search}}(t) \right]^{(0)} = \left[\Sigma_{a_2}^{\text{search}}(t - \Delta t) \right]^{(K_f)} + \frac{\left[\lambda(t - \Delta t) \right]^{(K_f)} - \lambda_{\text{reference}}}{\left(\frac{\text{pnf}_2}{\Sigma_{a_2}} \right)_{\text{average}}}$$

where K_f is the final iteration index number at time $(t - \Delta t)$ and Δt is the time step interval. $\Sigma_{a_2}^{\text{search}}$ has units of cm^{-1} .

6. Xenon Effect

$[\Delta k_{\infty}(t)]_{\text{xenon},l}$ is the term corresponding to the local xenon concentration effect and is given by:

$$[\Delta k_{\infty}(t)]_{\text{xenon},l} = -(\text{pnf}_2)_l \frac{\alpha_l(t)}{[1 + \alpha_l(t)]}$$

where

$$\alpha_l(t) = \frac{[\Sigma_{a_2}^{\text{xenon}}(t)]_l}{(\Sigma_{a_2})_l}$$

The following equations are used to calculate the xenon absorption cross section:

$$[\Sigma_{a_2}^{\text{xenon}}(t)]_l = X_l(t) [\sigma_{a_2}^{\text{Xe}}(t)]_l$$

where

$$[\sigma_{a_2}^{\text{Xe}}(t)]_l = (\sigma_{a_2}^{\text{Xe}})_l [1 + aT_l(t) + bT_l^2(t)]$$

$\sigma_{a_2}^{\text{Xe}}$ is the reference value of the microscopic Xenon 135 absorption cross section which is input by the INCXEXE tape. It is the spatial and spectrum averaged thermal absorption cross section of Xenon 135 for a node in units of cm^2 . Here a and b are corewise input parameters used to compute the moderator temperature effect on $\sigma_{a_2}^{\text{Xe}}$. $X_l(t)$ is the node average xenon concentration in units of $(\text{atoms}/\text{cm}^3)$. Thus $[\Sigma_{a_2}^{\text{xenon}}(t)]_l$ is the node average xenon cross section in units of cm^{-1} . The calculation of $X_l(t)$ is given in section III.F.2.

C. MIGRATION AREA CALCULATION

The calculation of M^2 proceeds along the same lines as for k_∞ and is given by:

$$\begin{aligned} [M^2(t)]_l = & (M^2_{\text{reference}})_l + \{\Delta M^2[P_l(t)]\}_{\text{Doppler}} \\ & + \{\Delta M^2[T_l(t)]\}_{\text{Moderator}} \\ & + [\Delta M^2(t)]_{\text{control},l} + [\Delta M^2(t)]_{\text{HTP},l} \\ & + [\Delta M^2(t)]_{\text{search},l} + [\Delta M^2(t)]_{\text{xenon},l} \end{aligned}$$

Each term is discussed in the following paragraphs (1 through 3) and each one is in units of cm^2 .

1. Reference Value

$(M^2_{\text{reference}})_l$ is the reference value as input by the INCXEXE tape. These values must correspond to the reference nodal power and reference moderator temperature which are compatible with the changes in M^2 with power and

temperature as noted in paragraph 2. These reference values must also correspond to the xenon-free condition.

2. Doppler and Moderator Temperature Effects

$\{\Delta M^2[P_\ell(t)]\}_{\text{Doppler}}$ and $\{\Delta M^2[T_\ell(t)]\}_{\text{Moderator}}$ are found either by interpolating and extrapolating of tables that come from the INCXE tape or from the following expressions when linear coefficients are being used.

$$\{\Delta M^2[P_\ell(t)]\}_{\text{Doppler}} = \left(\frac{\partial M^2}{\partial P}\right) [P_\ell(t) - P_{\text{reference}}]$$

and

$$\{\Delta M^2[T_\ell(t)]\}_{\text{Moderator}} = \left(\frac{\partial M^2}{\partial T}\right) [T_\ell(t) - T_{\text{reference}}]$$

where $\left(\frac{\partial M^2}{\partial P}\right)$ and $\left(\frac{\partial M^2}{\partial T}\right)$ are the input power and temperature coefficients, respectively.

3. Control, Soluble Poison, Xenon, and Eigenvalue Search Effects

$[\Delta M^2(t)]_{\text{control},\ell}$ is the input value for nodalwise external control.

The remaining terms are given by:

$$[\Delta M^2(t)]_{\text{HTP},\ell} = -(L^2)_\ell \frac{\bar{\beta}_\ell(t)}{[1 + \beta_\ell]}$$

$$[\Delta M^2(t)]_{\text{search},\ell} = -(L^2)_\ell \frac{\gamma_\ell(t)}{[1 + \gamma_\ell(t)]}$$

$$[\Delta M^2(t)]_{\text{xenon},\ell} = -(L^2)_\ell \frac{\alpha_\ell(t)}{[1 + \alpha_\ell(t)]}$$

where $(L^2)_\ell$ is the square of the migration length for thermal neutrons. $(L^2)_\ell$ is the node averaged value corresponding to the average core power and average moderator temperature with no xenon and comes from the INCXE tape. Note that L^2 also has units of cm^2 . $\alpha_\ell(t)$, $\beta_\ell(t)$ and $\gamma_\ell(t)$ are described in the preceding section which covers the k_∞ effects.

D. MODERATOR TEMPERATURE CALCULATIONS

The moderator temperature is calculated on the basis of a heat balance. The enthalpy difference between two adjacent nodes within a flow channel

is calculated first. Then this difference is used to compute the moderator temperature differential between the two nodes. Finally, the moderator temperature is determined by adding up the temperature differentials. It should be noted that all calculations are based on the assumption that the central point within each node represents the average of the node.

1. Specific Enthalpy Calculations

The specific enthalpy difference between two nodes is given by:

$$\Delta h_{\ell}(t) = \frac{C}{F_{ij}} \frac{1}{W(t)} \frac{1}{2} [P_{ijk}(t) + P_{ijk}^*(t)]$$

Here, $\Delta h_{\ell}(t)$ is the difference of the average specific enthalpy of node ℓ and either the node below or the node above, depending on which direction the coolant flows, and has units of (BTU/lb). C is the factor for converting Mw to BTU/hr and equals 3.413×10^6 (BTU/Mwh). $W(t)$ is the average channel coolant flow rate in units of (lb/hr) and is an input variable. F_{ij} is an input array of channel relative flow factors which are normalized to a core average value of unity.

$P_{ijk}(t)$ is the total power in node ℓ in units of Mw, and its determining equation is given in section III.E.2. P_{ijk}^* is defined as:

$$P_{ijk}^*(t) = \begin{cases} P_{ijk-l}(t) & \text{(for upward coolant flow)} \\ P_{ijk+l}(t) & \text{(for downward coolant flow)} \end{cases}$$

Note that in the calculation of $\Delta h_{\ell}(t)$, the program defines $P_{ij0}(t) = P_{ij K_M+1}(t) = 0$ where K_M is the maximum node index in the K-direction.

2. Moderator Temperature Calculations

$\Delta T_{\ell}(t)$ is the coolant temperature difference between the center of node ℓ and the center of the node above or below it in units of °F and is given by:

$$\Delta T_{\ell}(t) = \frac{\Delta h_{\ell}(t)}{\left[\frac{\partial h [T_{\ell}^*(t)]}{\partial T} \right]_p}$$

$\left[\frac{\partial h(T)}{\partial T}\right]_p$ is found either by interpolation or extrapolation from an input table of the specific heat of the coolant at constant pressure as a function of temperature or it is equal to a single input value for all nodes in the core when the option to use constant coefficients for calculating $\{\Delta k_\infty[P_\ell(t)]\}_{\text{Doppler}}$, $\{\Delta k_\infty[T_\ell(t)]\}_{\text{Moderator}}$, $\{\Delta M^2[P_\ell(t)]\}_{\text{Doppler}}$, and $\{\Delta M^2[T_\ell(t)]\}_{\text{Moderator}}$ is in use (refer to sections III.B.2 and III.C.2). The specific heat is in units of (BTU/lb°F). Here, $T_\ell^*(t)$ is the temperature of the coolant at the center of the node below (above) node ℓ when the coolant flow is upward (downward). Thus the temperature in node ℓ is given by:

$$T_\ell(t) = T_\ell^*(t) + \Delta T_\ell(t)$$

For channels with upward coolant flow

$$T_{i,j_0}(t) = [T_{\text{inlet}}(t)]_1$$

and for channels with downward coolant flow

$$T_{i,j_{K_M+1}}(t) = [T_{\text{inlet}}(t)]_2$$

where $[T_{\text{inlet}}(t)]_m$ are the inlet coolant temperatures which are input in °F.

E. THERMAL FLUX AND POWER CALCULATIONS

In order to compute the power and thermal flux distributions from the source distribution, the following procedures (paragraphs 1 and 2) are utilized.

1. Thermal Flux Calculation

The unnormalized thermal flux is defined as follows:

$$\psi_{2,\ell}(t) = \frac{\lambda(t) S_\ell(t)}{(\nu \Sigma_{f_1})_\ell \Gamma_\ell(t) + (\nu \Sigma_{f_2})_\ell}$$

where $S_\ell(t)$ and $\lambda(t)$ are the converged source distribution and core eigenvalue respectively, at time t . $(\nu \Sigma_{f_1})_\ell$ and $(\nu \Sigma_{f_2})_\ell$ are node

averaged values corresponding to the average core power and average moderator temperature which are input from the INCXE tape and are in units of cm^{-1} . $\Gamma_l(t)$ is the ratio of the node average fast-to-thermal flux which can be a time dependent input parameter.

The normalization constant for the thermal flux is given by:

$$M(t) = \frac{P_{\text{core}}(t)}{V\kappa \sum_j [(\Sigma_{f1})_j \Gamma_j(t) + (\Sigma_{f2})_j] \psi_{2,j}(t)}$$

where $P_{\text{core}}(t)$ is the total core power in Mw which can be a time dependent input variable and must be consistent with the total core volume in one- and two-dimensional as well as three-dimensional problems to yield the proper power density. V is the volume of a node in cm^3 and κ is the Mwsec/fission constant which is input. $(\Sigma_{f1})_l$ and $(\Sigma_{f2})_l$ are node averaged values corresponding to the average core power and average moderator temperature which are input from the INCXE tape and are in units of cm^{-1} .

The normalized thermal flux in absolute units (neutrons/ cm^2 sec) is then given by:

$$\phi_{2,l}(t) = M(t) \psi_{2,l}(t)$$

Note that this is the node averaged value of the flux.

2. Power Calculation

$P_l(t)$ is the total power in Mw produced in node l at time t and is given by:

$$P_l(t) = \kappa [(\Sigma_{f1})_l \Gamma_l(t) + (\Sigma_{f2})_l] \phi_{2,l}(t) V$$

F. IODINE AND XENON CONCENTRATION CALCULATIONS

The calculations of the xenon and iodine concentrations are based on the assumption of constant flux and power throughout the duration of a time step. With this assumption, the xenon and iodine equations can be integrated over the time step interval Δt .

1. Iodine Calculations

$I_{\ell}(t)$ is the node average iodine concentration in units of (atom/cm³) and is calculated by:

$$I_{\ell}(t) = I_{o,\ell}(t) - [I_{o,\ell}(t) - I_{\ell}(t - \Delta t)] e^{-\lambda_I \Delta t}$$

where

$$I_{o,\ell}(t) = \gamma_I \frac{P_{\ell}(t - \Delta t)}{\kappa V \lambda_I}$$

γ_I is the yield of Iodine-135 per fission, λ_I is the decay constant of I-135 in sec⁻¹, and Δt is given in seconds. Note that Δt is input into CESE in units of hours and the code converts it to seconds automatically.

2. Xenon Calculations

For the xenon concentrations, we have:

$$X_{\ell}(t) = X_{o,\ell}(t) - \lambda_I \frac{[I_{o,\ell}(t) - I_{\ell}(t - \Delta t)]}{\lambda_{\ell}(t) - \lambda_I} e^{-\lambda_I \Delta t} - \left[X_{o,\ell}(t) - X_{\ell}(t - \Delta t) - \lambda_I \frac{[I_{o,\ell}(t) - I_{\ell}(t - \Delta t)]}{\lambda_{\ell}(t) - \lambda_I} \right] e^{-\lambda_{\ell}(t) \Delta t}$$

where

$$X_{o,\ell}(t) = (\gamma_I + \gamma_X) \frac{P_{\ell}(t - \Delta t)}{\kappa V \lambda_{\ell}(t)}$$

$$\lambda_{\ell}(t) = \lambda_X + [\sigma_{a2}^{Xe}(t - \Delta t)]_{\ell} \phi_{2,\ell}(t - \Delta t)$$

Here, $X_{\ell}(t)$ is the node average xenon concentration in units of (atoms/cm³), γ_X is the direct yield of Xenon-135 per fission, and λ_X is the decay constant of Xenon-135 in sec⁻¹.

IV. APPLICATION AND LOGISTICS OF CEXE

A. DESCRIPTION OF INCEXE

The INCEXE code is the auxiliary program that generates tape input for CEXE. This code was written in order to eliminate the preparation of card input for the nodal dependent physics parameters that CEXE uses. The use of INCEXE to prepare CEXE input is particularly of value when calculations are required at various times during core life for which the fuel exposure distribution in the reactor may be complex.

INCEXE input requires, for each fuel type, tables of all of the exposure-dependent physics parameters along with the three-dimensional geometry, composition, and exposure distribution. The code then performs linear interpolations and extrapolations to determine the values of the various parameters in each node. In addition, INCEXE also inputs other information required for the CEXE calculations.

The exposure-dependent physics parameters that are handled with INCEXE are:

Σ_{f1} ; Σ_{f2} ; $v\Sigma_{f1}$; $v\Sigma_{f2}$; σ_{a2}^{Xe} ; Σ_{a2} ; L^2 ; M^2 reference; k_{∞} reference; and pnf_2 .

The definitions of each of these variables are given in sections III and V.A.

The equations utilized for each parameter are identical, and they are as follows. Define $(Z)_\ell$ as the value of parameter Z in node ℓ , where Z is any of the above list of variables, and E_ℓ as the corresponding exposure of that node. Assume that composition I resides in node ℓ . The input table of each parameter is given as a function of exposure at the points E_i which are arbitrarily spaced apart by the variable intervals, $\Delta E_i = E_{i+1} - E_i$. E_M is the maximum exposure value for which table input has been supplied. Then for $E_\ell < E_M$ and $E_i < E_\ell < E_{i+1}$,

$$(Z)_\ell = \frac{[Z(E_{i+1})]^I - [Z(E_i)]^I}{\Delta E_i} (E_\ell - E_i) + [Z(E_i)]^I$$

For $E > E_M$

$$(Z)_\ell = \frac{[Z(E_M)]^I - [Z(E_{M-1})]^I}{\Delta E_{M-1}} (E_\ell - E_M) + [Z(E_M)]^I$$

In addition to the nodal-dependent physics parameters, INCEXE supplies:

(1) the core geometry by indicating which nodes contain fuel; (2) tables of the specific heat of the moderator; (3) tables of the feedback effect of the moderator temperature on k_∞ and M^2 ; and (4) tables of the Doppler feedback effect on k_∞ and M^2 .

It should be noted that the physics parameters specified for the INCEXE tape must correspond to the xenon-free condition, since the xenon density is set equal to zero at the beginning of a new CESE problem as discussed in the following section.

B. STEADY-STATE CALCULATIONS

CESE has been set up to study xenon transients in the following manner. For a new problem (sometimes referred to as a normal problem), the core is assumed to be xenon- and iodine-free and the source distribution is iterated on until convergence is met and balances exist in the power, moderator temperature and source distributions. This power distribution is then used to build in the "equilibrium" xenon concentrations by the use of a large initial time step; somewhere on the order of 40 hours. The source distribution is again iterated on until convergence is met and a balance exists between the power, moderator temperature and source distributions. This new converged distribution is checked against the first converged distribution and if these two distributions are such that the sources at each node meet an input convergence criterion, then the problem is considered to be at equilibrium core conditions. If any node fails this convergence test, then another large time step is taken and new xenon and iodine concentrations are computed. Another converged source distribution is obtained and checked against the one from the previous time step. This procedure is repeated until two successive converged source distributions pass the convergence test at each node in the core at which time equilibrium conditions are assumed.

C. PERTURBATIONS

Once having obtained equilibrium core conditions, CEXE proceeds to introduce the perturbation that will give rise to the transient. The types of perturbation that can be studied are quite varied since any of the time-dependent input parameters as noted in section IV.E below can be changed. For example, the change of the xenon and/or iodine concentrations at any node in the reactor, the insertion or withdrawal of control rods, replacement of exposed fuel with fresh fuel, etc., can be simulated. Once the perturbation has been introduced, the calculation continues until it reaches the end of the input data.

D. RESTART FEATURES

Two restart tapes can be created by CEXE during the course of a new problem. The first of these is generated automatically at the equilibrium core condition and can be used to study different types of perturbations for the same initial conditions without having to establish the equilibrium conditions each time. The second, which can be generated at the end of the problem, is optional. The restart tape at the end of the problem can be used to continue the problem further in time and it is possible to change any of the time-dependent input variables as noted in the following section in the restarted problem.

E. TIME DEPENDENT INPUT FEATURES

The time dependent input for CEXE has been set up to allow for arbitrary changes in the iodine and xenon concentrations in any node at any time during the transient. In addition, arbitrary changes of control in any node at any time can be simulated by input changes in $(\Delta k_{\infty})_{\text{control}}$ and $(\Delta M^2)_{\text{control}}$. Account can be taken for nodal-dependent changes in the fast-to-thermal flux ratio which result from these changes in control. Time-dependent input parameters for load following are: (1) the total core power, (2) the average nodal coolant flow, (3) the inlet coolant temperature, and (4) the soluble poison cross section and control rod positions. In addition, the size of the time step can be varied.

F. OVERALL PROGRAM EXECUTION LOGIC

The essential features of the overall logistic flow path of the input/output and the calculations of CEXE are now discussed. Figure 2 shows the logical flow path of CEXE in detail. In this figure each box represents a specific operation and the name of the subroutine in which the operation is carried out is indicated below the box.

1. New Problems

In executing a new problem, the problem is set up by reading in the data, which will remain constant throughout the entire problem, from cards and from the INCXE tape. Then the first set of time-dependent data is read in (time-dependent data are read in at each time point after equilibrium is reached, see section V.B). These data will be used to calculate the equilibrium core condition and will remain constant until equilibrium is attained. The code then computes the thermal flux, the power, and the moderator temperature distributions based on the initial source and eigenvalue guesses. These values are then used in the feedback equations to compute the k_{∞} and M^2 values at each node in the core, and the code proceeds to calculate the first iterated source distribution and eigenvalue which are checked against the initial guesses for convergence. If the convergence test fails, CEXE continues until the source distribution and eigenvalue between two successive iterations pass the convergence tests. It should be noted that if the problem requires a poison search such that the core eigenvalue be equal to some reference, the poison cross section is readjusted after each iteration.

After converging at the initial time point, i.e., time = 0.0, a large time step Δt_0 which is input in the first set of time-dependent data, is taken to build in the "equilibrium" xenon and iodine concentrations. The source distribution is then recomputed by the iteration procedure and during each iteration new power and moderator temperature feedback effects are calculated. After convergence, the code checks for equilibrium core conditions by a comparison of the nodalwise source distributions between the converged values at this time point and those at time zero. If the equilibrium test fails, the code then takes another large time step (Δt_0) and recomputes the xenon and iodine values at the end of the new time step. A new converged source distribution is calculated and this is checked against the last

converged distribution to ascertain whether or not equilibrium conditions have been obtained. This is continued until the equilibrium test is passed at which point equilibrium conditions are assumed.

If an eigenvalue search is required in any problem, it is possible to either set the reference value equal to the value computed at equilibrium conditions or input a reference value. Under the former condition, the reference value is automatically set equal to the last converged value at the equilibrium point. Also, the code automatically generates a restart tape at this point.

Once the problem has reached equilibrium core conditions, it is possible to change any one of the system variables, as discussed in section IV.C, in order to introduce a perturbation. This is accomplished by making the necessary changes in the second set of time-dependent data. Once these changes have been introduced, the code computes a new converged source distribution and updates the power and moderator feedback effects during each iteration. This converged source distribution is the one that is used to proceed through the next time step Δt , which is input in the second set of time-dependent data, for the computation of new xenon and iodine concentrations. Once these new concentrations have been calculated, new system parameter changes may be input via the next time-dependent data set. This procedure is continued until the last set of time-dependent data has been read in and then, if the option to generate a restart tape is exercised, the tape will be created and the problem will terminate. It should be noted that the time step length Δt beyond the equilibrium time may be varied since a new value is read in with each set of time-dependent data.

2. Restart Problems

In executing a restart problem, care must be exercised in preparation of the card input. The reason is that all of the data that are input through cards must be input again at the beginning of the restart problem even though these data may be identical to that which were in the machine at the time the restart tape was created. This is because essentially only the source, the xenon, and the iodine distributions are stored on the restart tape. It should be noted that it is thus possible to change any of the input variables at the beginning of a restart problem so that the code

will utilize these new data in obtaining a new converged source distribution that will be used to proceed through the first time step of the restart problem. For example, the control rod configuration can be changed at the time that the original problem was stopped. Note also that if all parameters remain the same at the time of restart, the code would require one additional source iteration before it can proceed on to the next time step, provided it was properly converged at the time the original problem was terminated.

G. SUBROUTINE DESCRIPTIONS

The following paragraphs (1 through 17) describe briefly each of the subroutines in the CEXE Code.

1. Subroutine MAIN

MAIN sets up the logic of the major portion of CEXE by governing the overlay calls. A clear definition of the three divisions of the program is implicit in MAIN. These divisions are the initialization and input; the source-eigenvalue iteration loop; and the xenon and iodine concentration calculations.

2. Subroutine NITIAL

NITIAL, which is executed only at the beginning of the problem, initializes CEXE constants. Examples are: time, eigenvalue, flow rate, top and bottom core temperatures, transport kernels, sources, and albedos.

3. Subroutine INPUT1

All time independent input variables, whether they are read from cards, the restart tape, or the INCXE tape, are read by INPUT1. Thus INPUT1 is called only once in each problem.

In addition, INPUT1 performs some minor calculations; the nodal volume is computed, total number of fuel-containing channels and nodes is determined, input albedos are subtracted from either 2.0 or 4.0 (for the vertical or horizontal, respectively) since only this result is pertinent to the source calculations, the number of exposed surfaces for each of the fuel-containing nodes are calculated.

4. Subroutine INPUT2

INPUT2 reads all time-dependent input quantities appearing on cards. The number of calls to INPUT2 during a problem depends on the problem type. For a new problem there are M+2 sets of INPUT2 data in a problem with M time steps beyond equilibrium. The first set applies to the equilibrium core condition calculations. The next set applies at the equilibrium time and is used to input the perturbation. The subsequent sets apply sequentially to the time points beyond equilibrium. For restart problems, there are M+1 sets of INPUT2 data in a problem with M time steps beyond the point of restart. The procedures for setting up INPUT2 data cards are explained in detail in section V.B.2.

Since the INCEXE tape and the optional restart tape both contain information pertinent to time-dependent quantities, this type of information is transferred by INPUT2 to scratch units. This facilitates the rewinding of such units and eliminates the need for performing lengthy tape searches.

Minor calculations are performed by INPUT2 which include the transformation of the input time from hours to seconds and the calculation of $[(\Sigma_{f_1})_l \Gamma_l + (\Sigma_{f_2})_l]$ and $[(v\Sigma_{f_1})_l \Gamma_l + (v\Sigma_{f_2})_l]$ which are required in the thermal flux and power calculations.

5. Subroutine OUTIN

Every INPUT1 and INPUT2 call is followed by a call to OUTIN. This subroutine is for the purpose of writing input information on the standard output unit and no calculations are performed here. Only parameters new to the input stream are displayed. If the problem is a restart type, then information contained on the restart tape will be displayed on the first call to OUTIN.

6. Subroutine TEMPER

The primary function of TEMPER is the calculation of the nodal moderator temperature distribution in each fuel containing node. The power distribution which is required in these calculations is supplied by POWER which is described in section IV.G.17. The moderator temperature calculations proceed one node at a time, working either up or down a channel, depending on the direction of flow.

If the option to bypass both the moderator temperature and the Doppler feedbacks throughout a problem is used, TEMPER is completely bypassed.

If Doppler, but not moderator temperature feedback, is required in a problem TEMPER must be called since the call to POWER for the computation of the power distribution is made from TEMPER. If a temporary bypass to either the Doppler and/or the moderator temperature feedback is issued for a particular time step, the power and/or the moderator temperature used in the preceding time step are retained. Note that if the Doppler feedback is bypassed at any time, the moderator temperature feedback must be bypassed at the same time. The converse is not true.

7. Subroutine SOURCE

SOURCE calculates the neutron source distribution and the core eigenvalue by iteration techniques. The calculations of the source distribution are performed on a slab-by-slab basis proceeding from the bottom of the reactor core (i.e., from $K=1$) to the top. A slab is defined as a two-dimensional horizontal (I,J) cross section of the reactor core with one node in the K-direction. Since the source in any one node is dependent on sources within its six nearest neighboring nodes, it is necessary to have three slabs in the computer memory at any one time (the slab of interest and its two adjacent neighbors). Neighboring nodes are defined as those six nodes adjacent to the six faces of the cubic node of interest. Diagonal internodal effects are neglected. The neutron source within a particular node is a function of its k_{∞} , the core eigenvalue, nearest neighboring node sources, the transport kernels (vertical and horizontal) and the albedos (vertical and horizontal).

When the calculations of the sources in the nodes of the first (bottom) slab are being performed it is necessary to have available parameters for the first two slabs. Thus information pertinent to the first slab is obtained from secondary memory (INCEXE tape and various scratch units). Refer to paragraph 12 for the description of SORCIO. Next, k_{∞} and M^2 are calculated for each node of the first slab (refer to paragraph 13 for the description of KANDM2) and the vertical and horizontal transport kernels are calculated for each node of this slab. Then the above-described procedure is repeated for the second slab. The code then proceeds to calculate

the source distribution in the first slab. Having completed these calculations, CEXE continues the above cyclic procedure for each of the higher slabs in sequential order until the top slab calculations have been completed.

After the source distribution has been calculated, the core eigenvalue is obtained and the sources are normalized to an average value of unity.

When using the option of the modified source guess in the iteration scheme as described in section II.D.3, the following calculation precedes the source calculation:

$$S_{old}^{(i)} = (S_{old}^{(i-1)} + S^{(i-1)})/2$$

where $S_{old}^{(i-1)}$ is the source guess used for the (i-1)th iteration and $S^{(i-1)}$ is the source computed from the (i-1)th iteration. Thus the source guess for the (i)th iteration is the average value of these two distributions.

8. Subroutine CONVRG

CONVRG performs all of the convergence tests. The converged source distributions from successive time points are compared in this subroutine for new problems when checks for equilibrium core conditions are made. Both the source distribution and the core eigenvalue convergence tests during the iteration procedures are made in CONVRG. The test for convergence of the homogeneous thermal poison search is also made here by comparing the computed core eigenvalue after each iteration with the reference eigenvalue.

9. Subroutine NWTIME

The primary function of NWTIME is to add the time increment to the time variable at the end of the calculations at each time point. This subroutine also calls POWER which performs the calculation of the normalization constant $M(t)$ used in the thermal flux calculations (refer to section III.E.1). This normalization constant is used for the xenon and iodine calculations.

An indicator (IND) is used in NWTIME for new problems at the equilibrium time since the program logic is set up to perform two source calculations at this time point. The first source corresponds to the equilibrium unperturbed core condition and the second corresponds to the perturbed

core condition. IND is utilized to suppress the incrementation of the time variable and the calculation of the xenon and iodine concentrations at the end of the first time step beyond the equilibrium time until after the second source calculation has been performed.

10. Subroutine XENON

XENON calculates iodine and xenon concentrations. The iodine concentrations are retained in out-of-core storage. This subroutine contains its own calculation of the power and flux distributions.

11. Subroutine NWSRCH

NWSRCH calculates the node-averaged absorption cross section of the homogeneous thermal poison. This cross section is used to maintain the core eigenvalue at some reference value. This cross section appears in the calculation of k_{∞} and M^2 .

12. Subroutine SORCIO

Each neutron source iteration requires nodally dependent parameters supplied by the INCEXE tape. FORTRAN IV direct access statements allow referencing of these parameters on the slab-by-slab scheme, described under SOURCE, from out-of-core storage locations with considerable savings in execution time. SORCIO performs the retrieval of such parameters from the disk and is totally subsidiary to SOURCE.

13. Subroutine KANDM2

KANDM2 calculates k_{∞} and M^2 and is called by SOURCE each time these variables are required in the source distribution calculations. The various feedback options are incorporated in these calculations. Among these are: k_{∞} and M^2 input control values; homogeneous thermal poison absorption cross section; the iterated, node-averaged, homogeneous thermal poison-absorption cross section described in NWSRCH; reference k_{∞} and M^2 from INCEXE; xenon feedback; and Doppler and moderator temperature feedback. When Doppler and moderator temperature feedbacks are found through interpolation (or extrapolation) of tables from INCEXE a call to FNTERP is made. Various option switches are used throughout KANDM2 in reference to the above feedbacks.

14. Subroutine OUTPUT

OUTPUT is a subroutine which writes calculated quantities on the standard output unit and is called from NWTIME. The nodally dependent output quantities are the xenon and iodine concentrations, the thermal neutron fluxes, and the powers.

Separate calculations of the powers and the thermal fluxes are contained in OUTPUT.

15. Subroutine STORE

STORE generates the two restart tapes, the second of which is optional. This subroutine is called from NWTIME.

16. Subroutine FNTERP

FNTERP performs linear interpolations and linear extrapolations on one-dimensional tables. These tables are the specific heat versus moderator temperature, k_{∞} and M^2 moderator feedback versus moderator temperature, and k_{∞} and M^2 Doppler feedback versus power. These tables are input through INCEXE. TEMPER uses FNTERP to calculate the nodal-dependent specific heat of the moderator. KANDM2 uses FNTERP to determine the effects of the moderator temperature and the Doppler feedbacks on k_{∞} and M^2 .

17. Subroutine POWER

The primary purpose of POWER is the calculation of power for each fuel-containing node. The powers calculated here are used solely for Doppler and moderator temperature feedbacks.

The parameter $M(t)$ used in the thermal flux and power calculations is also calculated in POWER. The calculation of $M(t)$ in this subroutine is independent of the calculation of the power distribution. XENON and OUTPUT make use of this by utilizing only the $M(t)$ calculational part of this subroutine.

H. ESTIMATES OF COMPUTATIONAL TIMES

The estimated running time for an INCEXE problem with 2600 nodes is approximately 2.0 minutes on the IBM 360/65.

For CEXE problems, the total running time is dependent on the number of nodes, the number of source iterations at each time point, the type of feedbacks incorporated into the calculation (i.e., Doppler and moderator temperature) and the number of time steps. Typical estimates are that CEXE requires, on the IBM 360/65, 4.0 seconds per source iteration for a problem containing about 1000 nodes with no feedback and 7.5 seconds per source iteration for a problem with 2600 nodes and both Doppler and moderator temperature feedbacks. In the three-dimensional transient analysis of the Pickering Reactor with no control (see Reference 2), the total running time of the problem was 4 hours. The core geometry of this problem contained 970 nodes. There were no Doppler or moderator temperature feedbacks. The problem was run for 25 time steps (with 4 hour time step intervals) and required 150 iterations at each time point to obtain a properly converged source distribution.

It should be noted that substantial savings in computing times can be obtained with CEXE relative to equivalent diffusion theory calculations in multidimensional problems. Comparisons between CEXE and two-group, two-dimensional diffusion theory calculations have shown that CEXE is about four times faster. Larger savings are expected for three-dimensional problems.

V. INPUT DESCRIPTIONS

The input formats for both CEEXE and INCEXE are fixed and the usual FORTRAN descriptions are used here to specify these formats. Because of the tape labeling and identification involved in running CEEXE and INCEXE, a vital portion of the input to these codes consists of the Job Control Language (JCL). A sample of the JCL used on the IBM 360/65 is given in Appendix C along with listings of these cards. The units for all physical parameter input variables are given in parentheses following the respective input descriptions.

A. INPUT DESCRIPTION FOR INCEXE

	<u>Columns</u>	<u>Format</u>
1. <u>Tape Label Card</u>		
This card contains a four-character word consisting of any alphanumeric data which are used to label the output tape	1-4	A4
2. <u>Data Specification Card</u>		
This card contains the following information:		
IMAX, maximum number of nodes in the I direction of the core (≤ 19)	1-5	I5
JMAX, maximum number of nodes in the J direction of the core (≤ 19)	6-10	I5
KMAX, maximum number of nodes in the K direction of the core (≤ 10)	11-15	I5
NTBL, number of values of exposure for which the exposure dependent physics parameters will be input (≤ 100)	16-20	I5
NFUELS, total number of fuel types for which physics parameters will be input (≤ 20)	21-25	I5
NTTBL, number of values of temperature for which the moderator temperature tables will be input (≤ 20)	26-30	I5
NPTBL, number of values of power for which the Doppler tables will be input (≤ 20)	31-35	I5

	<u>Columns</u>	<u>Format</u>
NOEXP		
If NOEXP = 0, no nodalwise exposures are input and $E_0 = 0.0$ in all nodes of the core.		
If NOEXP $\neq$ 0, all nodalwise exposures must be input.		
	36-40	I5
3. <u>Fuel Type Specification Cards</u>		
This set of cards gives the fuel type contained within each node in the (I,J) plane of the core, where 0 denotes nonfueled nodes.		
In the first subset, the first card contains the first 16 nodes for I = 1 and the next contains the last 3 nodes for I = 1, if JMAX > 16.		
In the second subset, the first card contains the first 16 nodes for I = 2 and so on.		
There is a total of IMAX such subsets of cards.		
	1-5, 6-10, 11-15, etc.	16I5
4. <u>Moderator Temperature Cards</u>		
This set is omitted if NTTBL = 0. The values of moderator temperature for which values of the specific heat, $(\Delta k_{\infty})_{\text{moderator}}$, and $(\Delta M^2)_{\text{moderator}}$ are to be supplied. There are NTTBL values on these cards. ($^{\circ}\text{F}$)		
	1-10, 11-20, etc.	8E10.0
5. <u>Specific Heat Table Cards</u>		
This set is omitted if NTTBL = 0. The values of specific heat that correspond to the temperatures given on card type 4. (Btu/lb $^{\circ}\text{F}$)		
	1-10, 11-20, etc.	8E10.0
6. <u>$(\Delta k_{\infty})_{\text{moderator}}$ Table Cards</u>		
Omit this set if NTTBL = 0.		
The values of $(\Delta k_{\infty})_{\text{moderator}}$ that correspond to the temperatures given on card type 4. (Dimensionless)		
	1-10, 11-20, etc.	8E10.0

		<u>Columns</u>	<u>Format</u>
7.	<u>$(\Delta M^2)_{\text{moderator}}$ Table Cards</u>		
	Omit this set if NTTBL = 0.	1-10,	8E10.0
	The values of $(\Delta M^2)_{\text{moderator}}$ that correspond to the temperatures given on card type 4. (cm^2)	11-20, etc.	
8.	<u>Power Cards</u>		
	Omit this set if NPTBL = 0. The values of nodal power for which values of $(\Delta k_{\infty})_{\text{Doppler}}$, and $(\Delta M^2)_{\text{Doppler}}$ are to be supplied. There are NPTBL values on these cards. (Mwt)	1-10, 11-20, etc.	8E10.0
9.	<u>$(\Delta k_{\infty})_{\text{Doppler}}$ Table Cards</u>		
	Omit this set if NPTBL = 0. The values of $(\Delta k_{\infty})_{\text{Doppler}}$ that correspond to the powers given on card type 8. (Dimensionless)	1-10, 11-20, etc.	8E10.0
10.	<u>$(\Delta M^2)_{\text{Doppler}}$ Table Cards</u>		
	Omit this set if NPTBL = 0. The values of $(\Delta M^2)_{\text{Doppler}}$ that correspond to the powers given on card type 8. (cm^2)	1-10, 11-20, etc.	8E10.0
11.	<u>Nodalwise Exposure Cards</u>		
	This set is omitted if NOEXP = 0. The values of the exposure in each node of the core which contains fuel. (Mwd/MTU)		
	In the first subset, the first card contains the exposures in channel ($I = 1, J = 1$) for $K = 1$ to 8 and the next card contains the exposures in channel ($I = 1, J = 1$) for $K = 9, 10$ if KMAX > 8.		

	<u>Columns</u>	<u>Format</u>
--	----------------	---------------

In the second subset, the first card contains the exposures in channel ($I = 1, J = 2$) for $K = 1$ to 8 and so on. This procedure is repeated until $I = 1, J = JMAX$ has been reached. Then the values for $I = 2, J = 1$ appear, and so on.

Note: The subsets for all channels in which there is no fuel are omitted.

1-10,	8E10.0
11-20,	
etc.	

12. Exposure Table Cards

The values of exposure for which values of the exposure-dependent physics parameters are to be supplied. There are a total of NTBL values on these cards. (Mwd/MTU)

1-10,	8E10.0
11-20,	
etc.	

13. Fast Sigma-Fission Table Cards (Σ_{f1})

The values of Σ_{f1} that correspond to the exposure given on card type 12. (cm^{-1})

In the first subset, the first card contains the first 8 values of Σ_{f1} for fuel type 1, the next card contains the next 8 values for fuel type 1, and so on until the NTBL values of Σ_{f1} for fuel type 1 have been supplied.

Following this are the subset cards of Σ_{f1} values for fuel type 2. There are a total NFUELS such subsets.

1-10,	8E10.0
11-20,	
etc.	

The following sets of cards have the identical format as set 13 with the following correspondence between the set identification and the parameter.

14. Thermal Sigma-Fission Table Cards (Σ_{f2}). (cm^{-1})

15. Fast nu-Sigma-Fission Table Cards ($\nu\Sigma_{f1}$). (cm^{-1})

16. Thermal nu-Sigma-Fission Table Cards ($\nu\Sigma_{f2}$). (cm^{-1})

17. Effective Xenon Thermal Absorption Cross Section Table Cards (σ_{a2}^{Xe}). (cm^2)

18. Thermal Absorption Cross Section Table Cards (Σ_{a2}). (cm^{-1})
19. Square of the Migration Length for Thermal Neutrons Table Cards (L^2). (cm^2)
20. Reference M^2 Table Cards ($M^2_{\text{reference}}$). (cm^2)
21. Reference k_{∞} Table Cards ($k_{\infty \text{ reference}}$). (Dimensionless)
22. pnf_2 Table Cards (pnf_2). (Dimensionless)

Figure 3 gives a listing of the input cards for INCEXE which were used to generate the PC3D tape used in the three-dimensional analyses of the Pickering Reactor. The column numbers for the card punching are given at the top of the figure. A portion of the corresponding output from this problem is shown in figure 5.

B. INPUT DESCRIPTION FOR CEXE

The input data on cards is supplied to CEXE through two different input routines called INPUT1 and INPUT2. The former routine is used for the time-independent parameters while the latter inputs the parameters that can be varied at any time point. Hence, for each problem, there is only one set of INPUT1 data and there are numerous sets of INPUT2 data, the total of which depends on the type of problem which is being executed. Note that for both a new problem and a restart problem, all INPUT1 data must always be specified. For a new problem, there are $M + 2$ sets of INPUT2 data where M is the total number of time steps after equilibrium time is established. This number is arrived at from the following considerations. The first set of INPUT2 data defines the system variables which are pertinent to the description of equilibrium core parameters. Hence, these variables are held constant while the code proceeds to establish the equilibrium core. Once this has been attained, the code reads the next set of INPUT2 data which introduces the perturbation that will cause the transient. These new parameters are then held constant through the first time step beyond equilibrium time. After completion of this time step, a new set of INPUT2 data is read in and the code proceeds in a similar fashion until it reaches the last set of INPUT2 data for this transient. The code is set up to perform a source calculation at this last time point and this is the reason for supplying a final set of INPUT2 data there.

For restart problems, there are $M + 1$ sets of INPUT2 data where M is the number of time steps beyond the restart time of the problem. This total can be arrived at from the above considerations along with the fact that CEXE is set up to perform a new source calculation at the initial time and, hence, requires a set of INPUT2 data at that point. This source calculation at initial time permits the user to change any of the problem input variables prior to performing the first time step. As an example, consider a problem has been run out to time t , and a restart tape has been generated. If the user desires to change the control rod configuration at t , before proceeding to time $(t + \Delta t)$ in his restart problem, he may do this in the first set of INPUT2 data. The comments concerning input preparation for restart problems as given in section IV.F.2 should be noted.

No provision has been made for stacking successive cases.

1. INPUT1 DESCRIPTION

	<u>Columns</u>	<u>Format</u>
1. <u>Title Card</u>		
Any alphanumeric data in any column.	1-80	20A4
This is the main title which is used as a heading for the printout of each time point.		
2. <u>Option Card</u>		
NSW1:		
NSW1 = 0 if moderator temperature feedback is never required.		
NSW1 = 1 otherwise	1-4	I4
NSW2:		
NSW2 = 0 if no reference eigenvalue is to be input		
NSW2 = 1 otherwise	5-8	I4
Note that if the $\lambda_{\text{reference}}$ is to be set equal to the equilibrium eigenvalue, then NSW2 = 0.		

	<u>Columns</u>	<u>Format</u>
NSW4:		
NSW4 = 0 if Doppler feedback is never required.		
NSW4 = 1 otherwise	9-12	I4
RSTRT:		
RSTRT = 0 if this is a new problem.		
RSTRT = 1 if this is a restart problem.	13-16	I4
RST2:		
RST2 = 0 if there is to be no restart tape created at the end of the problem.		
RST2 = 1 otherwise	17-20	I4
LAMSW:		
LAMSW = 0 if there is to be no poison search in order to maintain the eigenvalue equal to some reference value		
LAMSW = 1 otherwise	21-24	I4
SOURSW:		
SOURSW = Number of initial time source guesses to be input on cards.		
Note: SOURSW = 0 for restart problems.	25-28	I4
When SOURSW = 0 for new problems the initial sources are set equal to unity.		
ITRMAX:		
ITRMAX = maximum number of source iterations at all time points.	29-32	I4
ITRAVG:		
ITRAVG = the number of the first iteration for which the source averaging technique is to be applied.		
Note: if source averaging is not required, set ITRAVG > ITRMAX.	33-36	I4

	<u>Columns</u>	<u>Format</u>
UPDNSW:		
UPDNSW = 0 if there is no bidirectional flow		
UPDNSW = 1 otherwise	37-40	I4
NSW6:		
NSW6 = 1 if the new type of transport kernel (Appendix A) is to be used.		
NSW6 = 2 if the FLARE-type transport kernels are to be used	41-44	I4
NFIN:		
NFIN = 0 if all flow factors are unity as utilized by the code. This allows for the omission of the flow factor cards in this case.		
NFIN = 1 requires the input of all flow factors.	45-48	I4
INTROP:		
INTROP = 0 if interpolation is to be performed in the Δk_{∞} and ΔM^2 tables for moderator temperature and Doppler feedback effects.		
INTROP = 1 if interpolation routines are to be replaced by linear equations.	49-52	I4
LABEL:		
Four-character word which matches the label given to the INCEXE tape that is to be used for this problem.	53-56	A4
3. <u>Node Size Specification and Convergence Criteria</u>		
<u>Card</u>		
DELTX:		
Length of the node in the horizontal direction ($\Delta X = \Delta Y$). (cm)	1-10	E10.0

	<u>Columns</u>	<u>Format</u>
DELTZ:		
Length of the node in the vertical direction (ΔZ). (cm)	11-20	E10.0
EPSREF:		
Nodalwise source convergence criterion. If $ S_{\ell}^{(i)}(t) - S_{\ell}^{(i-1)}(t) \leq \text{EPSREF}$ for all nodes in the core at iteration (i) and time (t), then the source distribution is considered to be converged for time (t).	21-30	E10.0
EPLREF:		
Eigenvalue convergence criterion. If $ \lambda^{(i)}(t) - \lambda^{(i-1)}(t) \leq \text{EPLREF}$ then the eigenvalue is considered to be converged at iteration (i) and time (t).	31-40	E10.0
Note that both of the preceding convergence criteria must be met in order to have total convergence.		
EPSO:		
Nodalwise source convergence criterion for equilibrium. If $ S_{\ell}(t) - S_{\ell}(t - \Delta t_0) \leq \text{EPSO}$ for all nodes in the core, where $S_{\ell}(t)$ is the converged or last source distribution at time (t), Δt_0 is the time step increment before the equilibrium time (t_0) [see IV.F.1], and $t < t_0$; then equilibrium conditions have been attained.		
Note: the code initially sets $t_0 = 10^{10}$ seconds for a new problem. When the above convergence is met, the code sets $t_0 = t$.	41-50	E10.0

	<u>Columns</u>	<u>Format</u>
EPSLRF:		
Eigenvalue convergence criterion for use with the search option. If $ \lambda^{(i)}(t) - \lambda_{\text{reference}} \leq \text{EPSLRF}$ then the poison search cross section is converged.	51-60	E10.0
4. <u>Coefficients and Related Parameters for Moderator Temperature and Doppler Feedback</u>		
Omit this card if INTROP = 0.		
CP:		
Constant specific heat used in all nodes of the core. (Btu/lb/°F)	1-10	E10.0
DKDT:		
Linear reactivity coefficient for moderator temperature feedback. ($\Delta k/^\circ\text{F}$)	11-20	E10.0
DMDT:		
Linear migration area coefficient for moderator temperature feedback. ($\text{cm}^2/^\circ\text{F}$)	21-30	E10.0
DKDP:		
Linear reactivity coefficient for Doppler feedback. ($\Delta k/\text{Mwt of nodal power}$)	31-40	E10.0
DMDP:		
Linear migration area coefficient for Doppler feedback. ($\text{cm}^2/\text{Mwt of nodal power}$)	41-50	E10.0
TREF:		
Reference moderator temperature for computing linear feedback effects. ($^\circ\text{F}$)	51-60	E10.0
PREF:		
Reference nodal power for computing linear feedback effects. (Mwt)	61-70	E10.0

Columns Format

5. Transport Kernel Parameters, Coefficients for
Temperature Dependence of the Xenon Cross
Section, and Reference Eigenvalue Card

The format of this card depends on the values
of NSW2 and NSW6.

GH:

The value of g^H as used in either the
FLARE-type transport kernel or the new
type kernel. (Dimensionless)

1-10 E10.0

GV:

The value of g^V as used in either the
FLARE or the new type transport kernels.
(Dimensionless)

11-20 E10.0

HMU:

The value of μ^H as required by the
FLARE-type kernels. Note that if
 $\mu^H = 1.0$, this value *must* still be
input. The value is omitted if NSW6 = 1
(Dimensionless)

If NSW6 = 1 - - - - - (omitted)

If NSW6 = 2 - - - - - 21-30 E10.0

VMU:

The value of μ^V as required by the
FLARE kernels. This value is omitted
if NSW6 = 1. (Dimensionless)

If NSW6 = 1 - - - - - (omitted)

If NSW6 = 2 - - - - - 31-40 E10.0

A: The value of a, the coefficient of
the linear term for the fit of the
temperature dependence of the xenon
absorption cross section. ($^{\circ}\text{F}$)⁻¹

If NSW6 = 1 - - - - - (21-30 E10.0)

If NSW6 = 2 - - - - - (41-50 E10.0)

	<u>Columns</u>	<u>Format</u>
B: The value of b, the coefficient of the quadratic term for the fit of the temperature dependence of the xenon absorption cross section. $(^{\circ}\text{F})^{-2}$.		
If NSW6 = 1 - - - - -	(31-40)	E10.0)
If NSW6 = 2 - - - - -	(51-60)	E10.0)
LAMREF:		
The value of $\lambda_{\text{reference}}$. This value is omitted if NSW2 = 0		
If NSW6 = 1 - - - - -	(41-50)	E10.0)
If NSW6 = 2 - - - - -	(61-70)	E10.0)
6. <u>Eigenvalue Guess and Energy per Fission</u>		
LAMBDA:		
Input eigenvalue used as the guess for the initial source iteration.		
For restart problems, this value is overridden by the last computed value from the previous case which is transferred to the code by the restart tape.	1-10	E10.0
KAPPA:		
Energy per fission. (Mw sec/fission)	11-20	E10.0
7. <u>Xenon and Iodine Decay and Yield Parameters</u>		
DECAYI:		
Decay constant for I-135. $(\text{seconds})^{-1}$	1-10	E10.0
GAMMAI:		
I-135 yield per fission. (Dimensionless)	11-20	E10.0
DECAYX:		
Decay constant for Xe-135. $(\text{seconds})^{-1}$	21-30	E10.0
GAMMAX:		
Xe-135 direct yield per fission. (Dimensionless)	31-40	E10.0

Columns Format

8. Matrix Identifying the Direction of Flow of the Moderator

Omit this set if there is no bi-directional flow, i.e. UPDNSW = 0. This set of cards identifies the flow direction of the moderator in each channel in the (I,J) plane, where 1 identifies the channel as having upward flow and 2 as having downward flow. The first card contains the values corresponding to I = 1 with J going from 1 to JMAX. The second card contains the JMAX values for I = 2, and so on. There are a total of IMAX cards in this set

UPDOWN (1,1)	1-3	19I3
UPDOWN (1,2)	4-6	
UPDOWN (1,3)	7-9	
etc.	etc.	

9. Flow Factors

If there is no input of flow factors, i.e. NFIN = 0, all flow factors are automatically set to unity and this set of cards is omitted. This set of cards gives the flow factors in each channel in the (I,J) plane. The first card of the first subset contains the values for I = 1 and J = 1 to 8. The second card contains the values for I = 1 and J = 9 to 16 and the last card contains the remaining values for I = 1. This procedure is then repeated for each value of I in increasing order up to and including IMAX. (Dimensionless)

F (1,1)	1-10	8E10.0
F (1,2)	11-20	
F (1,3)	21-30	
etc.	etc.	

	<u>Columns</u>	<u>Format</u>
10. <u>Source Guesses</u>		
<p>If there is no input of a source guess, i.e. SOURSW = 0, the initial source guess is automatically set to unity. In this case, omit this set. The format used here is set up to allow for input of source guesses in arbit- rary nodes without the need of specifying values in all nodes. Nodes for which no source guess is input have a unit value guess. The total number of source guesses appearing in this set must equal SOURSW. (Dimensionless)</p>		
I ₁	1-3	I3
J ₁	4-6	I3
K ₁	7-9	I3
S(I ₁ ,J ₁ ,K ₁)	10-20	E11.0
I ₂	21-23	I3
J ₂	24-26	I3
K ₂	27-29	I3
S(I ₂ ,J ₂ ,K ₂)	30-40	E11.0
etc.	etc.	
11. <u>Vertical Albedos</u>		
ALPHAV(1):		
<p>Vertical albedo used in all of the bottom nodes, i.e. K = 1. (Dimensionless)</p>		
	1-10	E10.0
ALPHAV(KMAX):		
<p>Vertical albedo used in all of the top nodes, i.e. K = KMAX. (Dimensionless)</p>		
	11-20	E10.0
12. <u>Number of Horizontal Albedos</u>		
NOALFA:		
<p>The total number of nodes in which horizontal albedos are to be input.</p>		
	1-4	I4

Columns Format

13. Horizontal Albedos

The horizontal albedos are input only in those channels for which the value is different from zero. Horizontal albedos are specified by channel only with the same value being used in corresponding nodes along the axial direction. Each card contains five sets of data with each set specifying the I,J channel coordinates and the horizontal albedo. There is a total of NOALFA sets of data. (Dimensionless)

I ₁	1-3	I3
J ₁	4-6	I3
ALPHAH(I ₁ ,J ₁)	7-16	E10.0
I ₂	17-19	I3
J ₂	20-22	I3
ALPHAH(I ₂ ,J ₂)	23-32	E10.0
etc.	etc.	

2. INPUT2 DESCRIPTION

1. Title Card

The format of this card depends on the time point for which the data is being specified. For all time points other than the last one, the format consists of any alphanumeric data in any column.

1-80 20A4

For the last time point, the following format must be used:

LAST	1-4	A4
Any alphanumeric data in columns 5 to 80.	5-8	19A4

	<u>Columns</u>	<u>Format</u>
2. <u>Option Card</u>		
NSW3:		
NSW3 = 0 if the moderator temperature feedback effects are to be the same as the previous time point.		
NSW3 = 1 otherwise	1-4	I4
Note: NSW3 must be 0 if NSW5 is 0.		
XSW:		
XSW = 0 if no xenon concentrations are to be input.		
XSW = 1 otherwise	5-8	I4
ISW:		
ISW = 0 if no iodine concentrations are to be input.		
ISW = 1 otherwise	9-12	I4
NKINFC:		
NKINFC = 0 if no new (Δk_{∞}) control values are to be input.		
NKINFC = 1 otherwise	13-16	I4
NDLM2C:		
NDLM2C = 0 if no new (ΔM^2) control values are to be input.		
NDLM2C = 1 otherwise	17-20	I4
NGAMIN:		
NGAMIN = number of Γ values to be input.	21-24	I4
NSW5:		
NSW5 = 0 if the Doppler feedback effects are to be the same as the previous time point.		
NSW5 = 1 otherwise	25-28	I4
PRXEN:		
PRXEN = 0 if the printing of the nodal-wise xenon concentrations are to be bypassed for this time point.		

	<u>Columns</u>	<u>Format</u>
PRXEN = 1 if the xenon concentrations are to be printed.	29-32	I4
PRIOD:		
PRIOD = 0 if the printing of the nodalwise iodine concentrations are to be bypassed for this time point.		
PRIOD = 1 if the iodine concentrations are to be printed.	33-36	I4
PRFLUX:		
PRFLUX = 0 if the printing of the nodal- wise thermal fluxes are to be bypassed for this time point.		
PRFLUX = 1 if the thermal fluxes are to be printed.	37-40	I4
PRPOW:		
PRPOW = 0 if the printing of the nodalwise powers are to be bypassed for this time point.		
PRPOW = 1 if the powers are to be printed.	41-44	I4

3. System Variables

DTIME:

Time step increment used to proceed through the subsequent time step. For a new problem, the value appearing in the first INPUT2 set is Δt_0 which is used for all time step increments in getting to equilibrium. The values appearing in the remaining INPUT2 sets are the various values of Δt , which can be varied for each time step (see section IV.F.1). For restart problems, Δt_0 is not used and the value in the first INPUT2 set, as well as the values in subsequent sets, is Δt . (hours)

1-10 E10.0

	<u>Columns</u>	<u>Format</u>
PCORE:		
The total core thermal power output.		
Note: For one- and two-dimensional problems this value must be consistent with the core volume in order to yield the proper power density. (Mwt)	11-20	E10.0
SGA2H:		
The value of the homogeneous thermal poison cross section Σ_{a2}^{HTP} . (cm^{-1})	21-30	E10.0
W:*		
The average channel coolant flow rate. (lb/hr)	31-40	E10.0
TMPBOT:*		
Moderator temperature at inlet of core bottom, i.e. for K=1. ($^{\circ}\text{F}$)	41-50	E10.0
TMPTOP:**		
Moderator temperature at inlet of core top, i.e. for K=KMAX. ($^{\circ}\text{F}$)	51-60	E10.0
4. <u>Γ Values</u>		
The basic feature of the format used for these cards consists of an overlay structure and this is accomplished by the input of values which are used in all nodes contained within three-dimensional blocks. Hence, the data is divided into sets in which the minimum and maximum I,J,K coordinates of the nodes within the block are specified along with the Γ value which is to be used for all nodes therein. The last value specified for		

* These values can be omitted when moderator temperature feedback effects are not included in the calculations, i.e. NSW1 = 0.

** This value can be omitted when there is not bidirectional flow, i.e. UPDNSW = 0.

Columns Format

each node is the one used in the calculations. It is possible to change the value of Γ in any node at any time point; however, when this is done, a complete set of Γ values must be supplied for all nodes in the core. There is one set of data per card and hence there must be NGAMIN cards in this card set. It should also be noted that this card set must appear in the first INPUT2 data set for both a new and a restart problem. (Dimensionless)

I_1	1-5	I5
J_1	6-10	I5
K_1	11-15	I5
I_2	21-25	I5
J_2	26-30	I5
K_2	31-35	I5
GAM(I_1 to I_2 , J_1 to J_2 , K_1 to K_2)	41-50	E10.0

As a simple example of the use of the overlay structure, consider a cubic reactor with 27 nodes, i.e. $IMAX = JMAX = KMAX = 3$.

Suppose that Γ is a constant value throughout the core ($\Gamma = x$) with the exception of the center node ($\Gamma = y$). This data specification could most easily be accomplished with two cards; the first would contain $I_1 = J_1 = K_1 = 1$; $I_2 = J_2 = K_2 = 3$; and $GAM = x$. The second would contain $I_1 = J_1 = K_1 = I_2 = J_2 = K_2 = 2$ and $GAM = y$.

5. Input Xenon Concentrations

Omit this set if $XSW = 0$. This set is divided into subsets (denoted by Greek letters) for each K level in the following manner:

(5.a.1) Number of input values

	<u>Columns</u>	<u>Format</u>
NOX:		
Number of input xenon concentrations which are to be changed for $K = 1$.	1-4	I4
(5.α.2) Xenon Concentrations		
If NOX = 0 on card (5.α.1) omit this part of the subset.		
This part of the subset consists of groups of data in which I, J, and X(I, J) are specified. There are NOX [as input on card (5.α.1)] of these groups. (atoms/cm ³)		
I ₁	1-3	I3
J ₁	4-6	I3
X(I ₁ , J ₁)	7-16	E10.0
I ₂	17-19	I3
J ₂	20-22	I3
X(I ₂ , J ₂)	23-32	E10.0
etc.	etc.	
(5.β.1) Number of input values		
NOX:		
Number of input xenon concentrations which are to be changed for $K = 2$	1-4	I4
(5.β.2) Xenon Concentrations		
If NOX = 0 on card (5.β.1), omit this part of the subset.		
This part of the subset has the same format as (5.α.2) and there are NOX [as input on card (5.β.1)] of these groups.		
I ₃	1-3	I3
J ₃	4-6	I3
X(I ₃ , J ₃)	7-16	E10.0
etc.	etc.	

This sequence is continued until all K levels
up to and including KMAX have been exhausted.

Note that in each subset, the first card (5._.1) must always appear and that only those values which are to be changed are input. CEXE initializes all xenon concentration to zero at the beginning of new problems.

6. Input Iodine Concentrations

Omit this set if ISW = 0. The format for this set is identical to that of card set (5) with the iodine concentrations replacing the xenon values. Note that the iodine and xenon inputs are entirely independent. (atoms/cm³)

7. Input of Control Variables

If NKINFC = NDLM2C = 0 this set is omitted. If either one of these input options is set to 1, the entire set must appear in the input deck. This set is divided into subsets (denoted by Greek letters) for each K level in the following manner:

(7.α.1) Number of input values

NOC:

Number of new values of $(\Delta k_{\infty})_{\text{control}}$
which are to be input for K = 1

1-4 I4

(7.α.2) $(\Delta k_{\infty})_{\text{control}}$ Values

If NOC on card 7.α.1 is zero, omit this part of the subset.

This part of the subset consists of groups of data with the same format as (5.α.2). There are NOC [as inputed on card (7.α.1)] of these groups.
(Dimensionless)

	<u>Columns</u>	<u>Format</u>
I_1	1-3	I3
J_1	4-6	I3
KINFC (I_1, J_1)	7-16	E10.0
I_2	17-19	I3
J_2	20-22	I3
KINFC (I_2, J_2)	23-32	E10.0
etc.	etc.	

(7.α.3) Number of input values

NOC:

Number of new values of $(\Delta M^2)_{\text{control}}$
which are to be input for $K = 1$

1-4 I4

(7.α.4) $(\Delta M^2)_{\text{control}}$ Values

If NOC is zero on card (7.α.3), omit
this part of the subset.

The format of this part of the
subset is identical to that of
(7.α.2) with $(\Delta M^2)_{\text{control}}$ replacing
 $(\Delta k_{\infty})_{\text{control}}$ (cm^2)

(7.β.1) Number of input values

NOC:

Number of new values of $(\Delta k_{\infty})_{\text{control}}$
which are input for $K = 2$

1-4 I4

(7.β.2) $(\Delta k_{\infty})_{\text{control}}$ Values

Same format as (7.α.2)

etc.

etc.

This sequence is continued until all K
levels up to and including K_{MAX} have been
exhausted.

Note that in each subset, the odd number
portions of the subset, (7._.1) and (7._.3),
must always appear and that only those
values which are to be changed are input.

CEXE initializes all values of $(\Delta k_{\infty})_{\text{control}}$
and $(\Delta M^2)_{\text{control}}$ to zero at the beginning
of both new and restart problems.

Figure 4 gives a listing of the input cards for CEXE which were used for a new problem in the three-dimensional Pickering Reactor analyses. The column numbers for the card punching are given at the top of the sheets. A portion of the corresponding output from this problem is shown in figure 6.

VI. OUTPUT DESCRIPTIONS

A. OUTPUT DESCRIPTION FOR INCEXE

All the data that is written on the INCEXE tape that is used as input for CEXE is printed as part of the output.

The initial data consists of the tape label, the maximum number of nodes in the I, J, and K directions, and the number of entries in the moderator temperature and Doppler feedback tables. This is then followed by a print-out of these latter tables. In the moderator temperature tables, the first set of data consists of the values of the moderator temperature which is followed by the values of the specific heat of the moderator, the $(\Delta k_{\infty}(T))$ moderator and $(\Delta M^2(T))$ moderator corresponding to the above values of the temperature. In the Doppler tables, the first set of information contains the values of the nodal powers which are followed by the corresponding values of $(\Delta k_{\infty}(P))$ Doppler and $(\Delta M^2(P))$ Doppler.

The next set of output displays the core loading pattern in the (I,J) plane in which the fuel type identification numbers for each channel are given.

The remainder of the output data consists of the nodally dependent physics parameters. The output is grouped into two sets, the first of which gives the values of Σ_{f1} , Σ_{f2} , $\nu \Sigma_{f1}$, and $\nu \Sigma_{f2}$ and the second gives values of σ_{a2}^{Xe} , Σ_{a2} , L^2 , M^2 reference, k_{∞} reference, and pnf_2 . Each of these sets is further grouped according to K levels with the values for all parameters within a set given for $K = 1$ and then for $K = 2$, and so on.

Figure 5 shows a portion of the output from a typical INCEXE problem for the Pickering Reactor whose input is shown in Figure 3.

B. OUTPUT DESCRIPTION FOR CEXE

The essential features of the output from CEXE consists of printing the xenon and iodine concentrations, the thermal flux and the power for each node at each time point in the transient. The output is divided into time-independent and time-dependent sections. The specifics of the output depend on the problem type.

1. New Problem

The time independent output appears only once per problem. The first set in this output consists of INPUT1 data. The first piece of output is the main title card which is followed by messages that show the choices for the various options. This is followed by the values of ΔX , ΔZ , and the convergence criteria. Then the values of the parameters used in the transport kernel calculations, the values of the coefficients used for the moderator temperature effects on σ_{a2}^{Xe} , and the input reference eigenvalue appear. The initial eigenvalue value guess, the energy per fission and the values of the iodine and xenon decay constants and direct yields per fission are printed next. This is followed by a printout of data related to the horizontal and vertical albedos; specifically $(4 - \alpha^H)$ for each I, J location and $(2 - \alpha^V)$ for each K level.

The time dependent output is repeated for each time point in the calculation. The first set of this output relates to INPUT2 data. First the main title appears which is followed by the title card from INPUT2. Again messages appear which show the choices used for the options available in INPUT2. Next, a line is printed showing the time increment, the total core power, Σ_{a2}^{HTP} , the average coolant flow rate per channel, and the inlet temperatures at the top and bottom of the core.

If any of the iodine or xenon concentrations, $(\Delta k_{\infty})_{\text{control}}$ or $(\Delta M^2)_{\text{control}}$, are changed manually through INPUT2 data, the entire three-dimensional array for the variable in which the change is made is the next set of output data. Each set of the changed variables is printed by K level and is labeled as being "new" in the case of the iodine and xenon concentrations.

The format of the next set of data depends on the problem. If no poison search is required for maintaining the eigenvalue equal to a reference value, the code prints the source iteration number and corresponding eigenvalue for each source calculation. If a poison search is requested, the code prints the iteration number followed by the searched poison cross section, $\lambda - \lambda_{\text{reference}}$, λ , and $\lambda_{\text{reference}}$ for each source iteration.

After the code either converges on a source distribution or reaches the maximum number of iterations, it then repeats the two titles, the time

in hours corresponding to this time point, the total core power, and the final eigenvalue. This data is then followed by a printout of the xenon concentrations in units of atoms/cc., the iodine concentrations in units of atoms/cc., the thermal flux in units of neutrons/cm² sec., and finally the power in units of Mw in each node. Each one of these variables is printed by K level with all the values of each variable grouped together. Figure 6 shows portions of the output from a typical CEXE problem for the Pickering Reactor whose input is shown in Figure 4.

2. Restart Problem

The first set of time-independent output for this type of problem is identical to that of a new problem; that is, it consists of a title, messages concerning selection of options, dimensions and convergence criteria, transport kernel variables, coefficients used to give moderator temperature effects on σ_{a2}^{Xe} , $\lambda_{reference}$, λ_0 , κ , and the decay constants and direct yields of iodine and xenon.

This is followed by a printout of the nodalwise values of the source which come from the restart tape and which are used as the initial source guess. Then follow the values of $(4 - \alpha^H)$ and $(2 - \alpha^V)$ with the same format used in new problems.

The first set of time-dependent output relates to INPUT2 data and it also has the same format as in new problems.

The first time that the time dependent output appears in the problem it contains a printout of the xenon and iodine concentrations as input from the restart tape and they are labeled as "new" concentrations. At subsequent time points this portion is deleted. This is followed by a printout of the $(\Delta k_{\infty})_{control}$ and $(\Delta M^2)_{control}$ values.

The remainder of the output is identical to that described for new problems.

VII. FORTRAN IV LISTINGS

This section displays the FORTRAN IV Listings for the CEXE and INCEXE Codes.

A. FORTRAN LISTING FOR INCEXE

```

C      PROGRAM INCEXE
C      THE CEXE INPUT LIBRARY GENERATOR
C
      IMPLICIT INTEGER*2 (I-N)
      INTEGER*2 FUEL
      INTEGER*4 LABEL
      DIMENSION FUEL(19,19), DAT(20), EXPOSE(19,19,10), EXPTBL(100),
1 DATA(100,20), DATAUT(19,19,10,6), JMAXX(2)
      DO 4 I = 1, 19
      DO 4 J = 1, 19
      DO 4 K = 1, 10
      EXPOSE(I,J,K) = 0.0
      DO 4 III = 1, 6
4 DATAUT(I, J, K, III) = 0.0
      READ (5,1002) LABEL
1002 FORMAT (A4)
      WRITE (10) LABEL
      READ (5,1000) IMAX, JMAX, KMAX, NTBL, NFUELS, NTTBL, NPTBL, NOEXP
1000 FORMAT (16I5)
      WRITE (10) IMAX, JMAX, KMAX, NTTBL, NPTBL
      IF (JMAX .GT. 10) GO TO 60
      JMAXX(1) = JMAX
      NMAX = 1
      GO TO 61
60 JMAXX(1) = 10
      JMAXX(2) = JMAX
      NMAX = 2
61 WRITE (6,2000) LABEL
2000 FORMAT ('1', ///, ' INCEXE TAPE LABELED ', A4 )
      WRITE (6,2001) IMAX, JMAX, KMAX, NTTBL, NPTBL
      DO 10 I = 1, IMAX
      READ (5,1000) (FUEL(I,J), J = 1, JMAX)
10 WRITE (10) (FUEL(I,J), J = 1, JMAX)
2001 FORMAT ('NO.OF NODES I ', I2, / ' NO.OF NODES J ', I2, / ' NO.OF
      INODES K ', I2, / ' NO.OF MODERATOR ENTRIES ', I2, / ' NO.OF DOPPLE
      IR ENTRIES ', I2 )
2002 FORMAT ( ///, ' IN THE FOLLOWING TABLES THE ABSCISSA CORRESPONDS T
      O NODES 1 THROUGH J MAX AND', / ' THE ORDINATE CORRESPONDS TO NODE
      IS 1 THROUGH I MAX, AXIAL INCREMENTS ARE ON SUCCEEDING PAGES')
      IF (NTTBL .EQ. 0) GO TO 7
      DO 5 III = 1, 4
      READ (5,1001) (DAT(II), II = 1, NTTBL)
      WRITE (10) (DAT(II), II = 1, NTTBL)
      GO TO (100, 110, 120, 130), III
100 WRITE (6,2005)
2005 FORMAT ('1', ///, ' TEMPERATURES (DEGREES F)', / )
      WRITE (6,2006) (DAT(II), II = 1, NTTBL)
2006 FORMAT (1X, 10F12.4)

```

```

GO TO 5
110 WRITE (6,2007)
2007 FORMAT ('OSPECIFIC HEAT (BTU/DEGREES F)' )
115 WRITE (6,2008) (DAT(II), II = 1, NTTBL)
008 FORMAT ('0', I3, 2X, 1P10E12.4)
GO TO 5
120 WRITE (6,2009)
2009 FORMAT ('ODELTA K INF.(MODERATOR)' )
GO TO 115
130 WRITE (6,2011)
2011 FORMAT ('ODELTA M**2(MODERATOR)' )
GO TO 115
5 CONTINUE
7 IF (NPTBL .EQ. 0) GO TO 8
DO 6 III = 1, 3
READ (5,1001) (DAT(II), II = 1, NPTBL)
WRITE (10) (DAT(II), II = 1, NPTBL)
GO TO (140, 150, 160), III
140 WRITE (6,2012)
2012 FORMAT ('1', ///, ' POWERS (MW)' )
145 WRITE (6,2008) (DAT(II), II = 1, NPTBL)
GO TO 6
150 WRITE (6,2013)
2013 FORMAT ('ODELTA K INF.(DOPPLER)' )
GO TO 145
160 WRITE (6,2014)
2014 FORMAT ('ODELTA M**2(DOPPLER)' )
GO TO 145
6 CONTINUE
8 WRITE (6,2002)
WRITE (6,2004)
2004 FORMAT ('1', ///, ' FUEL TYPES', // )
WRITE (6,3000) (J, J = 1, JMAX)
3000 FORMAT (8X, 19I4 )
WRITE (6,3002)
3002 FORMAT ('0')
DO 11 I = 1, IMAX
11 WRITE (6,2003) I, (FUEL(I,J), J = 1, JMAX)
2003 FORMAT (I4, 4X, 19I4)
IF (NOEXP .EQ. 0) GO TO 141
DO 15 I = 1, IMAX
DO 15 J = 1, JMAX
IF (FUEL(I,J) .EQ. 0) GO TO 15
READ (5,1001) (EXPOSE(I,J,K), K = 1, KMAX)
15 CONTINUE
1001 FORMAT (8E10.0)
141 READ (5,1001) (EXPTBL(IJK), IJK = 1, NTBL)
NTABS = 4

```

```

19 DO 50 III = 1, NTABS
   DO 20 N = 1, NFUELS
20 READ (5,1001) (DATA(IJK,N), IJK = 1, NTBL)
   DO 30 I = 1, IMAX
   DO 30 J = 1, JMAX
   IF (FUEL(I,J) .EQ. 0) GO TO 30
   DO 30 K = 1, KMAX
   IF (EXPOSE(I,J,K) - EXPTBL(1)) 21, 22, 23
21 DATAUT(I,J,K,III) = DATA(1,FJEL(I,J)) - (DATA(2,FUEL(I,J)) -
1 DATA(1,FUEL(I,J))) /
1 (EXPTBL(2) - EXPTBL(1)) *
1 (EXPTBL(1) - EXPOSE(I,J,K))
   GO TO 30
22 DATAUT(I,J,K,III) = DATA(1,FUEL(I,J))
   GO TO 30
23 IF (EXPOSE(I,J,K) - EXPTBL(NTBL)) 24, 27, 28
24 DO 25 II = 2, NTBL
   IF (EXPOSE(I,J,K) .LE. EXPTBL(II)) GO TO 26
25 CONTINUE
   WRITE (6,1003)
1003 FORMAT ('OMACHINE ERROR AT INCXEX STATEMENT 25 ... SORRY')
   STOP
26 DATAUT(I,J,K,III) = DATA(II-1,FUEL(I,J)) + (DATA(II,FJEL(I,J)) -
1 DATA(II-1,FUEL(I,J))) /
1 (EXPTBL(II) - EXPTBL(II-1)) *
1 (EXPOSE(I,J,K) - EXPTBL(II-1))
   GO TO 30
27 DATAUT(I,J,K,III) = DATA(NTBL,FUEL(I,J))
   GO TO 30
28 DATAUT(I,J,K,III) = DATA(NTBL,FUEL(I,J)) + (DATA(NTBL,FUEL(I,J)) -
1 DATA(NTBL-1,FUEL(I,J))) /
1 (EXPTBL(NTBL) - EXPTBL(NTBL-1)) *
1 (EXPOSE(I,J,K) - EXPTBL(NTBL))
30 CONTINUE
50 CONTINUE
   IF (NTABS - 4) 200, 200, 300
200 DO 250 K = 1, KMAX
   DO 250 III = 1, 4
   MIN = 1
   DO 250 M = 1, NMAX
   GO TO (210, 220, 230, 240), III
210 WRITE (6,2015) K
2015 FORMAT ('1', ///, ' SIGMA F1   K = ', I2, // )
213 MAX = JMAXX(M)
   WRITE (6,3001) (J, J = MIN, MAX)
   DO 215 I = 1, IMAX
215 WRITE (6,2008) I, (DATAUT(I,J,K,III), J = MIN, MAX)
   MIN = MAX + 1

```

```

      GO TO 250
3001 FORMAT (6X, 10(I6, 6X))
220 WRITE (6,2016) K
2016 FORMAT ('1', ///, ' SIGMA F2   K = ', I2, // )
      GO TO 213
230 WRITE (6,2017) K
2017 FORMAT ('1', ///, ' NU SIGMA F1   K = ', I2, // )
      GO TO 213
240 WRITE (6,2018) K
2018 FORMAT ('1', ///, ' NU SIGMA F2   K = ', I2, // )
      GO TO 213
250 CONTINUE
      GO TO 55
300 DO 370 K = 1, KMAX
      DO 370 III = 1, 6
      MIN = 1
      DO 370 M = 1, NMAX
      GO TO (310, 320, 330, 340, 350, 360), III
310 WRITE (6,2019) K
2019 FORMAT ('1', ///, ' NODE AVERAGED FLUX WEIGHTED THERMAL ABSORPTION
1CROSS SECTION OF XENON   K = ', I2, // )
313 MAX = JMAXX(M)
      WRITE (6,3001) (J, J = MIN, MAX)
      DO 315 I = 1, IMAX
315 WRITE (6,2008) I, (DATAUT(I,J,K,III), J = MIN, MAX)
      MIN = MAX + 1
      GO TO 370
320 WRITE (6,2020) K
2020 FORMAT ('1', ///, ' SIGMA A2   K = ', I2, // )
      GO TO 313
330 WRITE (6,2021) K
2021 FORMAT ('1', ///, ' SQUARE OF MIGRATION LENGTH FOR THERMAL NEUTRON
1S   K = ', I2, // )
      GO TO 313
340 WRITE (6,2022) K
2022 FORMAT ('1', ///, ' M**2 (REFERENCE)   K = ', I2, // )
      GO TO 313
350 WRITE (6,2023) K
2023 FORMAT ('1', ///, ' K INF.(REFERENCE)   K = ', I2, // )
      GO TO 313
360 WRITE (6,2024) K
2024 FORMAT ('1', ///, ' PNF2   K = ', I2, // )
      GO TO 313
370 CONTINUE
55 DO 70 K = 1, KMAX
      DO 70 III = 1, NTABS
70 WRITE (10) ((DATAUT(I,J,K,III), J = 1, JMAX), I = 1, IMAX)
      IF (NTABS .EQ. 6) STOP
      NTABS = 6
      GO TO 19
END

```

B. FORTRAN LISTING FOR CEXE

```

C     CEXE
C     A THREE DIMENSIONAL PROGRAM FOR OPTIMUM SPATIAL FLUX CONTROL
C
C     IMPLICIT INTEGER * 2 (I-N)
C     COMMON /HCOM/ LAST, ITR, ITRMAX, IL, TIME, TIME0, SWCONL, IND,
1     ITMIN, RST2
C     COMMON /KCOM/ N22, L22
C     INTEGER * 2 SWCONL, RST2
C     DEFINE FILE 22(51,361,U,N22)
C     FROID IS A ROUTINE FOR OVERRIDING LABEL INFO ON INCXE TAPE UNIT 10
C     CALL FROID2(10)
C     NITIAL INITIALIZES STARTING QUANTITIES
C     CALL NITIAL
C     INPUT1 READS TIME INDEPENDENT INPUT
C     CALL INPUT1
C     INPUT2 READS TIME DEPENDENT INPUT
C     CALL INPUT2
C     OUTIN PRINTS INPUT
C     CALL OUTIN(1)
C     INITIALIZE ITERATION COUNTER
45    ITR = 0
C     TEMPER CALCULATES TEMPERATURES
50    CALL TEMPER
C     SOURCE CALCULATES TRANSPORT KERNELS, SOURCES, EIGENVALUE, ETC.
70    CALL SOURCE
C     CONVRG CHECKS FOR VARIOUS CONVERGENCES
C     CALL CONVRG(IRETUR)
C     IRETUR = 1 MEANS EIGENVALUE SEARCH HAS FAILED
C     IRETUR = 2 MEANS CONVERGENCE TESTS HAVE PASSED
C     IRETUR = 3 MEANS SOURCE ITERATION HAS FAILED TO CONVERGE
C     GO TO(70,100,75) ,IRETUR
75    IF (ITR .LT. ITRMAX) GO TO 50
C     NWTIME UPDATES TIME
100   CALL NWTIME
C     TIME0 IS EQUILIBRIUM TIME
C     TIMIN IS INITIAL TIME, WHETHER 0.0 OR RESTART
C     IF ((TIME .EQ. TIME0) .OR. (TIME .EQ. TIMIN)) GO TO 120
C     XENON CALCULATES XENON AND IODINE CONCENTRATIONS
C     CALL XENON
C     IF(TIME.LT.TIME0) GO TO 45
120   CALL INPUT2
C     CALL OUTIN(0)
C     GO TO 45
C     END

```

```

SUBROUTINE SORCIO(N)
C  SORCIO CONTROLS DIRECT ACCESS READING OF INCON INFO
COMMON/FCDM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
COMMON /ICOM/ KINFC(19,19,10),DELM2C(19,19,10),SGA2(19,19),
1 L2(19,19),M2IN(19,19),KINFIN(19,19),PNF2(19,19)
COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /KCOM/ N22, L22
INTEGER*2 FUEL, IMAX, JMAX, KMAX,          LOOP, SUMCHN
REAL KINFC, L2, M2IN, KINFIN, IOD
DEFINE FILE 22(51,361,U,N22)
GO TO (10,20),N
10 L22 = 1
DO 13 K = 1, KMAX
READ (10) ((SA2XEF(I,J,K), J = 1, JMAX), I = 1, IMAX)
READ (10) ((SGA2(I,J), J = 1, JMAX), I = 1, IMAX)
READ (10) ((L2(I,J), J = 1, JMAX), I = 1, IMAX)
READ (10) ((M2IN(I,J), J = 1, JMAX), I = 1, IMAX)
READ (10) ((KINFIN(I,J), J = 1, JMAX), I = 1, IMAX)
READ (10) ((PNF2(I,J), J = 1, JMAX), I = 1, IMAX)
WRITE (22'L22) ((SGA2(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
WRITE (22'L22) ((L2(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
WRITE (22'L22) ((M2IN(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
WRITE (22'L22) ((KINFIN(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
WRITE (22'L22) ((PNF2(I,J), J = 1, JMAX), I = 1, IMAX)
13 L22 = L22 + 1
RETURN
20 IF ((K .EQ. 1) .AND. (LOOP .EQ. 1)) L22 = 1
READ (22'L22) ((SGA2(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
READ (22'L22) ((L2(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
READ (22'L22) ((M2IN(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
READ (22'L22) ((KINFIN(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
READ (22'L22) ((PNF2(I,J), J = 1, JMAX), I = 1, IMAX)
L22 = L22 + 1
IF (L22 .GT. (5 * KMAX)) FIND (22'1)
RETURN
END

```



```

SUBROUTINE ENTERP (V,N,XIN,YINK,YINM,N1,ANSW1,ANSW2)
C  ENTERP INTERPOLATES TABLES FOR SPECIFIC HEAT, DOPPLER AND MODERATOR
C  FEEDBACKS
  IMPLICIT INTEGER*2 (I-N)
  INTEGER*4 N1
  DIMENSION XIN(20), YINK(20), YINM(20)
  GO TO (5, 15, 15), V1
5  ANSW1 = 0.0
  ANSW2 = 0.0
  RETURN
15 IF (V - XIN(1)) 20, 25, 30
C  EXTRAPOLATE DOWNWARD
20 ANSW1 = YINK(1) - (YINK(2) - YINK(1)) / (XIN(2) - XIN(1)) *
1   (XIN(1) - V)
  IF (N1 .EQ. 3) RETURN
  ANSW2 = YINM(1) - (YINM(2) - YINM(1)) / (XIN(2) - XIN(1)) *
1   (XIN(1) - V)
  RETURN
25 ANSW1 = YINK(1)
  ANSW2 = YINM(1)
  RETURN
30 IF (V - XIN(N)) 35, 50, 55
C  INTERPOLATE
35 DO 40 I = 2, N
  IF (V .LE. XIN(I)) GO TO 45
40 CONTINUE
45 ANSW1 = YINK(I-1) + (YINK(I) - YINK(I-1)) / (XIN(I) - XIN(I-1)) *
1   (V - XIN(I-1))
  IF (N1 .EQ. 3) RETURN
  ANSW2 = YINM(I-1) + (YINM(I) - YINM(I-1)) / (XIN(I) - XIN(I-1)) *
1   (V - XIN(I-1))
  RETURN
50 ANSW1 = YINK(N)
  ANSW2 = YINM(N)
  RETURN
C  EXTRAPOLATE UPWARD
55 ANSW1 = YINK(N) + (YINK(N) - YINK(N-1)) / (XIN(N) - XIN(N-1)) *
1   (V - XIN(N))
  IF (N1 .EQ. 3) RETURN
  ANSW2 = YINM(N) + (YINM(N) - YINM(N-1)) / (XIN(N) - XIN(N-1)) *
1   (V - XIN(N))
  RETURN
END

```

```

SUBROUTINE INITAL
INITAL INITIALIZES STARTING QUANTITIES
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 I, J, K
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/ECOM/WH(19,19),WV(19,19),WVBLD(19,19),WVABV(19,19),
1 WHABV(19,19)
COMMON /HCOM/ LAST, ITR, ITRMAX, I1, TIME, TIME0, SWCONL, IND,
1 TIMIN, RST2
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
COMMON/NCOM/ X(19,19,10),IDD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /OCOM/ SGA2H, SGA2S, PNSAAV
REAL KAPPA, LAMBDA, LOLD, IDD, M, NUSGGM
INTEGER * 2 UPDOWN,SWCONL,RST2,SUMCHN,FJEL
IND = 0
TIME = 0.0
TIME0 = 1.0E+10
LAMBDA = 1.0
TIMIN = 0.0
PNSAAV = 0.0
W = 0.0
TMPTOP = 0.0
TMPBOT = 0.0
DO 10 I=1,19
DO 10 J=1,19
UPDOWN(I,J) = 1
ALPHAH(I,J) = 0.0
WV(I,J) = 0.0
WH(I,J) = 0.0
F(I,J) = 1.0
DO 10 K = 1, 10
S(I,J,K) = 1.0
T(I,J,K) = 0.0
10 CONTINUE
DO 11 K = 1, 10
11 ALPHAV(K) = 0.0
RETURN
END

```

```

SUBROUTINE POWER(NRETRN)
POWER CALCULATES NODALLY DEPENDENT POWERS
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
COMMON /PCOM/ DELTX,DELTZ,V,DTIME
COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEQ,SWCONL,IND,
1 TIMIN, RST2
INTEGER*2          IMAX, JMAX, KMAX, FUEL, NSW4, NSW5, LOOP,
1 LAST, ITR, ITRMAX, I1, SWCONL, IND, RST2, I3, NSW1, NSW3, NSW6,
1 ITRSRC, UPDOWN
REAL NUSGGM, M, KAPPA, LAMBDA, LOLD
SUMPHI = 0.0
DO 10 I = 1, IMAX
DO 10 J = 1, JMAX
IF (FUEL(I,J) .EQ. 0) GO TO 10
DO 9 K = 1, KMAX
9 SUMPHI = SUMPHI + LAMBDA * S(I,J,K) * SGGM(I,J,K) / NUSGGM(I,J,K)
10 CONTINUE
M = PCORE / (KAPPA * SUMPHI * V)
IF (NRETRN .EQ. 0) RETURN
12 DO 13 I = 1, IMAX
DO 13 J = 1, JMAX
IF (FUEL(I,J) .EQ. 0) GO TO 13
DO 11 K = 1, KMAX
11 P(I,J,K) = PCORE * LAMBDA * S(I,J,K) * SGGM(I,J,K) / NUSGGM(I,J,K)
1 / SUMPHI
13 CONTINUE
RETURN
END

```

```

SUBROUTINE TEMPER
C   TEMPER CALCULATES TEMPERATURES
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 I, J, K, INTRDP
COMMON /HCOM/ LAST, ITR, ITRMAX, IL, TIME, TIMEO, SWCONL, IND,
1 TIMIN, RST2
COMMON /FCOM/ FUEL(19,19), IMAX, JMAX, KMAX, I, J, K, LOOP
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON /MCOM/ S(19,19,10), ALPHAV(10), ALPHAH(19,19),
1 SOLD(19,19,10), SOLDT(19,19),
1 LAMBDA, LOLO, GV, GH, ITRSRC, I3, HMU, VMU
COMMON /BCOM/ F(19,19), PCORE, W, KAPPA, SGGM(19,19,10), NUSGGM(19,19,10)
1, UPDOWN(19,19), NSW1, NSW3, NSW4, NSW5, NSW6, NFIN
COMMON /PCOM/ DELTX, DELTZ, V, DTIME
COMMON /QCOM/ CP, DKDT, DMDT, DKDP, DMDP, TREF, PREF, INTRDP
COMMON /UCOM/ TIN(20), CPIN(20), KNMIN(20), M2MIN(20), PIN(20),
1 M2DIN(20), KNDIN(20), NTIN, NPIN
REAL M, LAMBDA, LOLO, KAPPA, NUSGGM, KNMIN, M2MIN, M2DIN, KNDIN
INTEGER * 2 SWCONL, FUEL, UPDOWN, RST2
IF ((NSW4 .EQ. 0) .OR. (NSW5 .EQ. 0)) RETURN
C   POWER CALCULATES NODALLY DEPENDENT POWERS
CALL POWER(1)
C   IF NEITHER MODERATOR FEEDBACK NOR UPDATE ARE REQUIRED DON'T BOTHER
C   WITH TEMPERATURES
12 IF (NSW1 .EQ. 0) RETURN
IF (NSW3 .EQ. 0) RETURN
DO 29 I = 1, IMAX
DO 29 J = 1, JMAX
IF (FUEL(I,J) .EQ. 0) GO TO 29
IF (UPDOWN(I,J) .EQ. 2) GO TO 25
C   UPWARD FLOW
DO 24 K = 1, KMAX
IF (K .GT. 1) GO TO 22
DELH = 3.413E6 / (F(I,J) * W) * 0.5 * P(I,J,K)
IF (INTRDP .EQ. 1) GO TO 10
C   ENTERP INTERPOLATES SPECIFIC HEAT TABLES
CALL ENTERP(TMPBOT, NTIN, TIN, CPIN, M2MIN, 3, CP, DUMMY)
10 DELT = DELH / CP
T(I,J,K) = TMPBOT + DELT
GO TO 24
22 DELH = 3.413E6 / (F(I,J) * W) * 0.5 * (P(I,J,K) + P(I,J,K-1))
IF (INTRDP .EQ. 1) GO TO 11
CALL ENTERP(T(I,J,K-1), NTIN, TIN, CPIN, M2MIN, 3, CP, DUMMY)
11 DELT = DELH / CP
T(I,J,K) = T(I,J,K-1) + DELT
24 CONTINUE
GO TO 29

```

```

C      DOWNWARD FLOW
25 K = KMAX
26 IF (K .LT. KMAX) GO TO 27
   DELH = 3.413E6 / (F(I,J) * W) * 0.5 * P(I,J,K)
   IF (INTROP .EQ. 1) GO TO 13
   CALL ENTERP(TMPTOP,NTIN,TIN,CPIN,M2MIN,3,CP,DUMMY)
13 DELT = DELH / CP
   T(I,J,K) = TMPTOP + DELT
   GO TO 28
27 DELH = 3.413E6 / (F(I,J) * W) * 0.5 * (P(I,J,K) + P(I,J,K+1))
   IF (INTROP .EQ. 1) GO TO 14
   CALL ENTERP(T(I,J,K+1),NTIN,TIN,CPIN,M2MIN,3,CP,DUMMY)
14 DELT = DELH / CP
   T(I,J,K) = T(I,J,K+1) + DELT
28 K = K - 1
   IF (K .GT. 0) GO TO 26
29 CONTINUE
   RETURN
   END

```

```

SUBROUTINE SOURCE
C SOURCE CALCULATES PRIMARILY SOURCES AND EIGENVALUE
  IMPLICIT INTEGER * 2 (I-N)
  INTEGER * 4 I, J, K
  COMMON /DCOM/ LAMSW, RSTRT, NSW2, I4, EPSREF, EPLREF, EPSO,
1 EPSLRF, LAMREF, EPSLAM
  COMMON/ECOM/WH(19,19),WV(19,19),WVBLO(19,19),WVABV(19,19),
1 WHABV(19,19)
  COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
  COMMON /GCOM/ ITRAVG
  COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEO,SWCONL,IND,
1 TIMIN, RST2
  COMMON /JCOM/ NH(19,19), NV(10), R, DXDZ
  COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
  COMMON /OCOM/ SGA2H, SGA2S, PNSAAV
  COMMON /PCOM/ DELTX,DELTZ,V,DTIME
  COMMON/SCOM/M2(19,19),M2ABV(19,19)
  COMMON/TCOM/KINF(19,19),KNFABV(19,19)
  COMMON /ICOM/ KINFC(19,19,10),DELM2C(19,19,10),SGA2(19,19),
1 L2(19,19),M2IN(19,19),KINFIN(19,19),PNF2(19,19)
  COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
  COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
  COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
  REAL NV, NH, LAMBDA, LOLD, M2, KINF, KAY, M2ABV, KINFC, L2, M2IN,
1 KINFIN, IOD, M, LAMREF, KAPPA, NUSGGM, KINFC, KNFABV
  INTEGER * 2 FUFL,SWCONL,RST2,SUMCHN,UPDOWN,RSTRT
  ITR = ITR + 1
  SUM3 = 0.0
  SUM4 = 0.0
  SUM5 = 0.0
C NWSRCH CALCULATES NODE AVERAGED ABSORPTION CROSS SECTION OF A
C HOMOGENEOUS THERMAL POISON
  CALL NWSRCH
  IF (TIME .GT. TIMIN) GO TO 12
  IF (ITR .GT. 1) GO TO 12
C SORCIO HANDLES INCON INFO
  CALL SORCIO(1)
12 K = 1
C SOURCE CALCULATIONS ARE MADE ONE HORIZONTAL SLAB AT A TIME. LOOP IS
C A CONTROL VARIABLE USED ONLY ON THE BOTTOM SLAB.
  LOOP = 0
1 LOOP = LOOP + 1
  IF (KMAX .EQ. 1) GO TO 3

```

```

2 IF (K .GE. KMAX) GO TO 15
3 CALL SORCIO(2)
  IF ((TIME .NE. 0.0) .OR. (ITR .NE. 1)) GO TO 15
  DO 14 I = 1, IMAX
  DO 14 J = 1, JMAX
  IF (FUEL(I,J) .EQ. 0) GO TO 14
  PNSAAV = PNSAAV + PNF2(I,J) / SGA2(I,J)
14 CONTINUE
15 DO 5 I = 1, IMAX
  DO 5 J = 1, JMAX
  IF (FUEL(I,J) .EQ. 0) GO TO 5
C   KANDM2 CALCULATES K INFINITY AND M**2
16 CALL KANDM2
C   CALCULATE TRANSPORT KERNELS
  Z=M2(I,J)
  IF (LOOP.NE.1) Z=M2ABV(I,J)
  IF ((K.EQ.1) .AND. (LOOP.EQ.1)) GO TO 558
  WVBLO(I,J)=WV(I,J)
  WV(I,J)=WVABV(I,J)
  WH(I,J)=WHABV(I,J)
558 IF ((K .GE. KMAX) .AND. (KMAX .NE. 1)) GO TO 5
  IF (NSW6.EQ.1) GO TO 560
  Z1=HMJ*((1.-GH)*SQRT(Z)/(2.*DELTX)+GH*Z/DELTX**2)
  Z2=VMU*((1.-GV)*SQRT(Z)/(2.*DELTZ)+GV*Z/DELTZ**2)
  GO TO 570
560 RP=R/SQRT(Z)
  Z3=EXP(-RP)
  PT=6.2831853*Z*SQRT(Z)/V*(RP*RP-1.+73*Z3*(RP+1.)*(RP+1.))
  Z1=GH*PT/(4.+2.*DXDZ)
  Z2=Z1*GV*DXDZ
570 IF (KMAX .GT. 1) GO TO 571
  WH(I,J) = Z1
  WV(I,J) = Z2
  GO TO 5
571 WHABV(I,J)=Z1
  WVABV(I,J)=Z2
  5 CONTINUE
  IF (KMAX .EQ. 1) GO TO 572
  IF ((K .EQ. 1) .AND. (LOOP .EQ. 1)) GO TO 1
572 IF ((ITR .LT. ITRAVG) .AND. (K .EQ. 1)) GO TO 103
  IF ((ITR .GE. ITRAVG) .AND. (K .EQ. 1)) GO TO 106
  GO TO 108
C   FIND SOURCES FROM PREVIOUS ITERATION
103 DO 105 I = 1, IMAX
  DO 105 J = 1, JMAX
  DO 105 KK = 1, KMAX
105 SOLD(I,J,KK) = S(I,J,KK)
  GO TO 108

```

```

106 DO 107 I = 1, IMAX
    DO 107 J = 1, JMAX
        DO 107 KK = 1, KMAX
107 SOLD(I,J,KK) = 0.5 * (S(I,J,KK) + SOLD(I,J,KK))
108 DO 20 I = 1, IMAX
    DO 20 J = 1, JMAX
104 IF (FUEL(I,J) .EQ. 0) GO TO 20
C   CALCULATE SOURCES
    SUM1 = 0.0
    IF (I .EQ. 1) GO TO 6
    SUM1 = SOLD(I-1,J,K) * WH(I-1,J)
  6 IF (I .EQ. IMAX) GO TO 7
    SUM1 = SUM1 + SOLD(I+1,J,K) * WH(I+1,J)
  7 IF (J .EQ. 1) GO TO 8
    SUM1 = SUM1 + SOLD(I,J-1,K) * WH(I,J-1)
  8 IF (J .EQ. JMAX) GO TO 9
    SUM1 = SUM1 + SOLD(I,J+1,K) * WH(I,J+1)
  9 IF (K .EQ. 1) GO TO 10
    SUM1 = SUM1 + SOLD(I,J,K-1) * WVBLO(I,J)
 10 IF (K .EQ. KMAX) GO TO 11
    SUM1 = SUM1 + SOLD(I,J,K+1) * WVABV(I,J)
 11 SUM2 = 1.0 - (ALPHAH(I,J) * WH(I,J) + ALPHAV(K) * WV(I,J))
    S(I,J,K) = KINF(I,J) / LAMBDA * SUM1 /
  1      (1.0 - KINF(I,J) / LAMBDA * SUM2)
C   CALCULATE EIGENVALUE
    SUM3 = SUM3 + S(I,J,K)
    SUM4 = SUM4 + S(I,J,K) * (WV(I,J) * NV(K) + WH(I,J) * NH(I,J))
    SUM5 = SUM5 + S(I,J,K) / KINF(I,J)
20 CONTINUE
52 IF (K .GE. KMAX) GO TO 30
    K = K + 1
    GO TO 2
30 LOLD = LAMBDA
    IF ((TIME .EQ. 0.0) .AND. (ITR .EQ. 1)) PNSAAV = PNSAAV / SUMNOD
    LAMBDA = (SUM3 - SUM4) / SUM5
    IF ((LAMSW .EQ. 1) .OR. (TIME .LT. TIME0)) GO TO 51
    IF (ITR .GT. 1) GO TO 50
    WRITE ( 6,1000)
1000 FORMAT ('ITERATION      EIGENVALUE' / )
  50 WRITE (6,1001) ITR, LAMBDA
1001 FORMAT (1X, I6, F17.6)
C   NORMALIZE SOURCES
  51 DO 54 K = 1, KMAX
    DO 54 I = 1, IMAX
        DO 54 J = 1, JMAX
            IF (FUEL(I,J) .EQ. 0) GO TO 54
            S(I,J,K) = S(I,J,K) / SUM3 * SUMNOD
  54 CONTINUE
    RETURN
    END

```



```

SUBROUTINE KANDM2
KANDM2 CALCULATES K INFINITY AND M**2
IMPLICIT INTEGER * 2 (I-N)
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
COMMON /ICOM/ KINFC(19,19,10),DELM2C(19,19,10),SGA2(19,19),
1 L2(19,19),M2IN(19,19),KINFIN(19,19),PNF2(19,19)
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMVTD, SA2XEF(19,19,10), SUMCHN
COMMON /DCOM/ SGA2H, SGA2S, PNSAAV
COMMON /QCOM/ CP, DKDT, DMDT, DKDP, DMDP, TREF, PREF, INTRJP
COMMON/SCOM/M2(19,19),M2ABV(19,19)
COMMON/TCOM/KINF(19,19),KNFABV(19,19)
COMMON /UCOM/ TIV(20), CPIN(20), KNMIN(20), M2MIN(20), PIN(20),
1 M2DIN(20), KNDIN(20), NTIN, NPIN
REAL KINFC, IOD, KINF, KNMIN, M2MIN, M2DIN, KNDIN, M, L2, M2IN,
1 KINFIN, KNFABV, KINFH, KINF5, KINFX, KAPPA, NUSGGM, M2, M2ABV,
1 LL, MODERK, MODERM
INTEGER*2 FUEL, SUMCHN, UPDOWN
INTEGER*4 NOPT1, NOPT2, I, J, K, INTRDP
IF ((K .GE. KMAX) .AND. (KMAX .NE. 1)) GO TO 7
KK = K + 1
IF ((K .EQ. 1) .AND. (LOOP .EQ. 1)) KK = K
TEMP = T(I,J,KK)
SGA2X = X(I,J,KK) * SA2XEF(I,J,KK) * (1.0 + A*TEMP + B*TEMP*TEMP)
SGA = SGA2(I,J)
ALFA = SGA2X / SGA
BETA = SGA2H / SGA
GAMA = SGA2S / SGA
PNF = -PNF2(I,J)
LL = -L2(I,J)
HTPK = PNF * BETA / (1.0 + BETA)
SRCHK = PNF * GAMA / (1.0 + GAMA)
XENK = PNF * ALFA / (1.0 + ALFA)
TRANSF = LL / PNF
HTPM = HTPK * TRANSF
SRCHM = SRCHK * TRANSF
XENM = XENK * TRANSF
NOPT1 = 2
IF (NSW4 .EQ. 0) NOPT1 = 1
NOPT2 = 2
IF (NSW1 .EQ. 0) NOPT2 = 1
7 IF (K .EQ. 1) GO TO 10
KINF(I,J) = KNFABV(I,J)
M2(I,J) = M2ABV(I,J)

```

```

8 IF (K .GE. KMAX) RETURN
10 IF (INTROP .EQ. 1) GO TO 9
    ENTERP INTERPOLATES THE MODERATOR FEEDBACK TABLES
    CALL ENTERP(TEMP,NTIN,TIN,KNMIN,M2MIN,NOPT2,MODERK,MODERM)
    ENTERP INTERPOLATES THE DOPPLER FEEDBACK TABLES
    CALL ENTERP(P(I,J,KK),NPIN,PIN,KNDIN,M2DIN,NOPT1,DOPLRK,DOPLRM)
    GO TO 15
9 IF (NSW1 .GT. 0) GO TO 11
    MODERK = 0.0
    MODERM = 0.0
    GO TO 12
11 MODERK = DKDT * (TEMP - TREF)
    MODERM = DMDT * (TEMP - TREF)
12 IF (NSW4 .GT. 0) GO TO 14
    DOPLRK = 0.0
    DOPLRM = 0.0
    GO TO 15
14 DOPLRK = DKDP * (P(I,J,KK) - PREF)
    DOPLRM = DMDP * (P(I,J,KK) - PREF)
15 KNFABV(I,J) = KINF(I,J) + DOPLRK + MODERK + KINFC(I,J,KK) +
1     HTPK + SRCHK + XENK
    M2ABV(I,J) = M2IN(I,J) + DOPLRM + MODERM + DELM2C(I,J,KK) +
1     HTPM + SRCHM + XENM
    IF (LOOP .GT. 1) RETURN
    KINF(I,J) = KNFABV(I,J)
    M2(I,J) = M2ABV(I,J)
    RETURN
    END

```

```

SUBROUTINE NWSRCH
NWSRCH CALCULATES THE NODE AVERAGED ABSORPTION CROSS SECTION OF A
HOMOGENEOUS THERMAL POISON WHICH IS USED TO MAINTAIN THE CORE
EIGENVALUE EQUAL TO A SPECIFIC VALUE (LAMREF)
C IMPLICIT INTEGER * 2 (I-N)
COMMON/DOM/LAMSW,RSTRT,NSW2,I4,EPSREF,EPLREF,EPSO,EPSLRF,
1 LAMREF, EPSLAM
COMMON /HCOM/ LAST, ITR, ITRMAX, I1, TIME, TIMEO, SWCONL, IND,
1 TIMIN, RST2
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
COMMON /OCOM/ SGA2H, SGA2S, PNSAAV
REAL LAMBDA, LOLD, LAMREF
INTEGER * 2 RSTRT,SWCONL,RST2
IF ((TIME .EQ. 0.0) .AND. (ITR .EQ. 1)) GO TO 30
IF (LAMSW .EQ. 0) GO TO 30
IF ((NSW2 .EQ. 0) .AND. (TIME .LT. TIMEO)) GO TO 30
SGA2S = SGA2S + (LAMBDA - LAMREF) / PNSAAV
RETURN
30 SGA2S = 0.0
RETURN
END

```

```

SUBROUTINE CONVRG(IRETUR)
C   CONVRG PERFORMS THE VARIOUS CONVERGENCE TESTS
    IMPLICIT INTEGER * 2 (I-N)
    INTEGER * 4 I, J, K
    COMMON/DCOM/LAMSW,RSTRT,NSW2,I4,EPSREF,EPLREF,EPSO,EPSLRF,
1   LAMREF, EPSLAM
    COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
    COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEO,SWCONL,IND,
1   TIMIN, RST2
    COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1   SOLD(19,19,10),SOLDT(19,19),
1   LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
    COMMON /OCOM/ SGA2H, SGA2S, PNSAAV
    REAL LAMREF, LAMBDA, LOLD
    INTEGER * 2 RSTRT,FUEL,SWCONL,RST2
C   TEST FOR SOURCE AND EIGENVALUE CONVERGENCE
    EPSL = LAMBDA - LOLD
    SWCONL = 0
    IF (ABS(EPSL) .LE. EPLREF) GO TO 9
    IRETUR = 3
    GO TO 14
9   DO 10 K = 1, KMAX
    DO 10 I = 1, IMAX
    DO 10 J = 1, JMAX
    IF (FUEL(I,J) .EQ. 0) GO TO 10
    EPSS = S(I,J,K) - SOLD(I,J,K)
    IF ((ABS(EPSS) .LE. EPSREF)) GO TO 10
    IRETUR=3
    GO TO 14
10  CONTINUE
    SWCONL = 1
C   LAMSW IS THE EIGENVALUE SEARCH INDICATOR
14  IF (LAMSW .EQ. 0) GO TO 13
    IF (NSW2 .EQ. 1) GO TO 11
    IF (TIME .LT. TIMEO) GO TO 13
C   TEST FOR EIGENVALUE SEARCH CONVERGENCE ON LAMREF
11  EPSLAM = ABS(LAMBDA - LAMREF)
    IF (ITR .GT. 1) GO TO 12
    WRITE (6,1000)
1000 FORMAT ('1', /// ' SEARCH ITERATION', // ' NO.      SIGMA SEARCH
1   ABS(LAMBDA-LAMREF)      LAMBDA      LAMREF')
    12 WRITE (6,1001) ITR, SGA2S, EPSLAM, LAMBDA, LAMREF
1001 FORMAT ('0', I3, 5X, E14.7, 3X, E14.7, 9X, E14.7, 4X, E14.7)
    IF (SWCONL .EQ. 0) RETURN
    IF (EPSLAM .LE. EPSLRF) GO TO 13
    IRETUR=1
    RETURN
13  IF (SWCONL .EQ. 0) RETURN

```

```

C      RSTRT IS THE RESTART PROBLEM INDICATOR
      IF (RSTRT .EQ. 0) GO TO 20
      IRETUR=2
      RETURN
20     IF ((TIME .LT. TIME0) .AND. (TIME .NE. TIMIN)) GO TO 30
      IRETUR=2
      RETURN
30     REWIND 23
C      TEST FOR EQUILIBRIUM
      DO 35 K = 1, KMAX
      READ (23) ((SOLDT(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 35 I = 1, IMAX
      DO 35 J = 1, JMAX
      IF (FUEL(I,J) .EQ. 0) GO TO 35
      FPSPRM = ABS(S(I,J,K) - SOLDT(I,J))
      IF(EPSPRM.LE.EPS0) GO TO 35
      IRETUR=2
      RETURN
35     CONTINUE
C      EQUILIBRIUM REACHED. MARK TIME, AND EIGENVALUE IF NOT INPUT
40     TIME0 = TIME
      IF(NSW2 .NE. 1) GO TO 400
      IRETUR=2
      RETURN
400    LAMREF=LAMBDA
      IRETUR=2
      RETURN
      END

```

```

SUBROUTINE NWTIME
C NWTIME INCREMENTS TIME AND TERMINATES PROGRAM AT END TIME
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 I, J, K
COMMON /ACOM/ TITLE(20), TITLE2(20)
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIME0,SWCONL,IND,
1 TIMIN, RST2
COMMON /PCOM/ DELTX,DELTZ,V,DTIME
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
REAL LAMBDA, LOLD
INTEGER * 2 FUEL,SWCONL,RST2
DATA XLAST / 'LAST' /
C POWER CALCULATES M FOR USE IN OUTPUT AND XENON
CALL POWER(0)
IF (TIME .EQ. TIMIN) REWIND 10
IF ((TIME .GE. TIME0) .AND. (IND .EQ. 1)) GO TO 7
REWIND 23
IF (TIME .EQ. TIME0) GO TO 7
C SET UP SOURCES FORM PREVIOUS TIME STEP
DO 6 K = 1, KMAX
DO 5 I = 1, IMAX
DO 5 J = 1, JMAX
SOLDT(I,J) = S(I,J,K)
5 CONTINUE
6 WRITE (23) ((SOLDT(I,J), J = 1, JMAX), I = 1, IMAX)
7 IF (TIME - TIME0) 10,20,30
10 TIME = TIME + DTIME
RETURN
20 IF (IND .EQ. 1) GO TO 30
C OUTPUT PRINTS OUTPUT
CALL OUTPUT
IND = 1
C STORE WRITES RESTART TAPES
CALL STORE
RETURN
30 CALL OUTPUT
TIME = TIME + DTIME
C CHECK FOR END TIME
IF (TITLE2(1) .NE. XLAST) RETURN
IF (RST2 .EQ. 1) GO TO 35
STOP
35 TIME = TIME - DTIME
CALL STORE
STOP
END

```

```

SUBROUTINE XENON
XENON CALCULATES XENON AND IODINE CONCENTRATIONS
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 I, J, K
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LODP
COMMON /PCOM/ DELTX,DELTZ,V,DTIME
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITR SRC,I3,HMU,VMU
COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEO,SWCONL,IND,
1 TIMIN, RST2
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
REAL LAMREF, LAMBDA, LOLD
REAL IOD0
REAL KAPPA, IOD, M, NUSGGM
INTEGER * 2 UPDOWN,FUEL,SWCONL,RST2,SUMCHN
IF (TIME .EQ. TIMEO) RETURN
DT = DTIME
REWIND 16
REWIND 15
DO 30 K = 1, KMAX
READ (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
DO 25 I = 1, IMAX
DO 25 J = 1, JMAX
IF (FUEL(I,J) .EQ. 0) GO TO 25
PHI2P = LAMBDA * S(I,J,K) / NUSGGM(I,J,K)
PHI2 = M * PHI2P
POW = KAPPA * SGGM(I,J,K) * PHI2 * V
IOD0 = GAMMAI * POW / (KAPPA * V * DECAYI)
DELIOD = IOD0 - IOD(I,J)
IOD(I,J) = IOD0 - DELIOD * EXP(-DECAYI * DT)
DCY = DECAYX + SA2XEF(I,J,K) * (1.0+A*T(I,J,K)+B*T(I,J,K)**2)*PHI2
X0 = (GAMMAI + GAMMAX) * POW / (KAPPA * V * DCY)
X(I,J,K) = X0 - DECAYI * DELIOD / (DCY - DECAYI) *
1 EXP(-DECAYI * DT) - (X0 - X(I,J,K) - DECAYI *
1 DELIOD / (DCY - DECAYI)) * EXP(-DCY * DT)
25 CONTINUE
WRITE (16) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
30 CONTINUE
REWIND 15
REWIND 16
DO 40 K = 1, KMAX
READ (16) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
40 WRITE (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
50 RETURN
END

```

```

SUBROUTINE INPUT1
C INPUT1 READS TIME INDEPENDENT INPUT INFO
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 LABEL, LABLIN, I, J, K, INTROP
COMMON /ACOM/ TITLE(20), TITLE2(20)
COMMON /BCOM/ F(19,19), PCORE, W, KAPPA, SGGM(19,19,10), NUSGGM(19,19,10)
1, UPDOWN(19,19), NSW1, NSW3, NSW4, NSW5, NSW6, NFIN
COMMON /CCOM/ SOURSW, UPDNSW, LABEL, XSW, ISW, NKINFC,
1 NDLM2C, NGAMIN
COMMON /DCOM/ LAMSW, RSTRT, NSW2, I4, EPSREF, EPLREF, EPS0, EPSLRF,
1 LAMREF, EPSLAM
COMMON /ECOM/ FUEL(19,19), IMAX, JMAX, KMAX, I, J, K, LOOP
COMMON /GCOM/ ITRAVG
COMMON /HCOM/ LAST, ITR, ITRMAX, I1, TIME, TIME0, SWCONL, IND,
1 TIMIN, RST2
COMMON /JCOM/ NH(19,19), NV(10), R, DXDZ
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON /MCOM/ S(19,19,10), ALPHAV(10), ALPHAH(19,19),
1 SOLD(19,19,10), SOLDT(19,19),
1 LAMBDA, LOLD, GV, GH, ITRSRC, I3, HMU, VMU
COMMON /NCOM/ X(19,19,10), IOD(19,19), DECAYX, DECAYI, GAMMAX, GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /QCOM/ SGA2H, SGA2S, PNSAAV
COMMON /PCOM/ DELTX, DELTZ, V, DTIME
COMMON /QCOM/ CP, DKDT, DMDT, DKDP, DMDP, TREF, PREF, INTROP
COMMON /UCOM/ TIN(20), CPIN(20), KNMIN(20), M2MIN(20), PIN(20),
1 M2DIN(20), KNDIN(20), NTIN, NPIN
REAL KAPPA, KINFC, IOD, LAMBDA, LAMREF, NH, LOLD, KNMIN, NV,
1 M2MIN, KNDIN, M2DIN, NUSGGM, L2, M2IN, KINFIN, M
INTEGER * 2 UPDOWN, FUEL, RSTRT, SWCONL, UPDNSW, SOURSW,
1 XSW, RST2, SUMCHN
EPSLRF = 0.0
LAMREF = 0.0
HMU = 0.0
VMU = 0.0
READ (5,1000) (TITLE(I), I = 1, 20)
1000 FORMAT (20A4)
READ (5,1001) NSW1, NSW2, NSW4, RSTRT, RST2, LAMSW, SOURSW,
1 ITRMAX, ITRAVG, UPDNSW, NSW6, NFIN, INTROP, LABEL
1001 FORMAT (13I4, A4)
IF (RSTRT .EQ. 1) CALL FROID2(20)
C UNIT 10 IS INCON TAPE
READ (10) LABLIN
IF (LABEL .EQ. LABLIN) GO TO 3
WRITE (6,1007) LABLIN
1007 FORMAT ('1 YOU REQUESTED TAPE LABELED ', A4, ' WHICH IS WRONG')
STOP

```



```

3 READ (10) IMAX, JMAX, KMAX, NTIN, NPIN
  READ (5,1002) DELTX, DELTZ, EPSREF, EPLREF, EPSO,
1      EPSLRF
1002 FORMAT (8E10.0)
  V = DELTX**2 * DELTZ
  R=(0.2387326*V)**0.3333333
  DXDZ = (DELTZ / DELTX)
  IF (INTRDP .EQ. 0) GO TO 1
  READ (5,1002) CP, DKDT, DMDT, DKDP, DMDP, TREF, PREF
1 IF (NSW2 .EQ. 1) GO TO 5
  IF (NSW6 .EQ. 2) GO TO 2
  READ (5,1002) GH, GV, A, B
  GO TO 10
2 READ (5,1002) GH, GV, HMU, VMU, A, B
  GO TO 10
5 IF (NSW6 .EQ. 2) GO TO 4
  READ (5,1002) GH, GV, A, B, LAMREF
  GO TO 10
4 READ (5,1002) GH, GV, HMU, VMU, A, B, LAMREF
10 READ (5,1002) LAMBDA, KAPPA
  READ (5,1002) DECAYI, GAMMAI, DECAYX, GAMMAX
  IF (UPDNSW .EQ. 0) GO TO 24
  DO 20 I = 1, IMAX
20 READ (5,1011) (UPDOWN(I,J), J = 1, JMAX)
1011 FORMAT(24I3)
24 IF (NFIN .EQ. 0) GO TO 28
  DO 25 I = 1, IMAX
25 READ (5,1002) (F(I,J), J = 1, JMAX)
28 IF (SOURSW .EQ. 0) GO TO 31
  READ (5,1003) (I, J, K, S(I,J,K), II = 1, SOURSW)
1003 FORMAT (4(3I3, E11.0))
31 DO 32 I = 1, IMAX
32 READ (10) (FUEL(I,J), J = 1, JMAX)
  IF (NTIN .EQ. 0) GO TO 34
  READ (10) (TIN(I), I = 1, NTIN)
  READ (10) (CPIN(I), I = 1, NTIN)
  READ (10) (KNMIN(I), I = 1, NTIN)
  READ (10) (M2MIN(I), I = 1, NTIN)
34 IF (NPIN .EQ. 0) GO TO 36
  READ (10) (PIN(I), I = 1, NPIN)
  READ (10) (KNDIN(I), I = 1, NPIN)
  READ (10) (M2DIN(I), I = 1, NPIN)
36 SUMCHN = 0
  DO 40 I = 1, IMAX
  DO 40 J = 1, JMAX
  IF (FUEL(I,J) .EQ. 0) GO TO 40
  SUMCHN = SUMCHN + 1
40 CONTINUE

```

```

SUMNOD = SUMCHN * KMAX
READ (5,1002) ALPHAV(1), ALPHAV(KMAX)
DO 42 K = 1, KMAX
42 ALPHAV(K) = 2.0 - ALPHAV(K)
READ (5,1001) NOALFA
IF (NOALFA .EQ. 0) GO TO 43
READ (5,1004) (I,J,ALPHAH(I,J)), II = 1, NOALFA)
1004 FORMAT (5(2I3, E10.0))
43 CONTINUE
DO 44 I = 1, IMAX
DO 44 J = 1, JMAX
44 ALPHAH(I,J) = 4.0 - ALPHAH(I,J)
IF (RSTRT .EQ. 0) GO TO 53
C UNIT 20 IS RESTART TAPE
READ (20) TIME, TIME0, LAMBDA, LAMREF, SGA2S, PNSAAV
TIMIN = TIME
READ (20) ((S(I,J,K), K = 1, KMAX), J = 1, JMAX), I = 1, IMAX)
NSW2 = 1
IND = 1
53 DO 54 K = 1, KMAX
54 WRITE (23) ((S(I,J,K), J = 1, JMAX), I = 1, IMAX)
C INITIALIZE ALL SOURCES IN NODES CONTAINING NO FUEL TO ZERO
DO 59 I = 1, IMAX
DO 59 J = 1, JMAX
IF (FUEL(I,J) .NE. 0) GO TO 59
DO 59 K = 1, KMAX
S(I,J,K) = 0.0
59 CONTINUE
DO 224 K = 1, KMAX
IF (KMAX .GT. 1) GO TO 221
NV(K) = 2.0 + ALPHAV(K) - 2.0
GO TO 224
221 IF (K .GT. 1) GO TO 222
NV(K) = 1.0 + ALPHAV(K) - 2.0
GO TO 224
222 IF (K .LT. KMAX) GO TO 223
NV(K) = 1.0 + ALPHAV(K) - 2.0
GO TO 224
223 NV(K) = ALPHAV(K) - 2.0
224 CONTINUE
DO 268 I = 1, IMAX
DO 268 J = 1, JMAX
SUMN = 0.0
IF (I .GT. 1) GO TO 261
SUMN = SUMN + 1.0
IF (I .LT. IMAX) GO TO 262
SUMN = SUMN + 1.0
GO TO 264

```

```

261 IF (I .LT. IMAX) GO TO 263
    SUMN = SUMN + 1.0
    IF (FUEL(I-1,J) .GT. 0) GO TO 264
    SUMN = SUMN + 1.0
    GO TO 264
262 IF (FUEL(I+1,J) .GT. 0) GO TO 264
    SUMN = SUMN + 1.0
    GO TO 264
263 IF (FUEL(I-1,J) .EQ. 0) SUMN = SUMN + 1.0
    IF (FUEL(I+1,J) .EQ. 0) SUMN = SUMN + 1.0
264 IF (J .GT. 1) GO TO 265
    SUMN = SUMN + 1.0
    IF (J .LT. JMAX) GO TO 266
    SUMN = SUMN + 1.0
    GO TO 268
265 IF (J .LT. JMAX) GO TO 267
    SUMN = SUMN + 1.0
    IF (FUEL(I,J-1) .GT. 0) GO TO 268
    SUMN = SUMN + 1.0
    GO TO 268
266 IF (FUEL(I,J+1) .GT. 0) GO TO 268
    SUMN = SUMN + 1.0
    GO TO 268
267 IF (FUEL(I,J-1) .EQ. 0) SUMN = SUMN + 1.0
    IF (FUEL(I,J+1) .EQ. 0) SUMN = SUMN + 1.0
268 NH(I,J) = SUMN + ALPHA(I,J) - 4.0
60 RETURN
    END

```

```

SUBROUTINE INPUT2
C INPUT2 READS TIME DEPENDENT INPUT INFO
IMPLICIT INTEGER * 2 (I-N)
INTEGER * 4 LABEL, I, J, K
C TIME DEPENDENT PARAMETERS
DIMENSION SGF(19,19)
COMMON /ACOM/ TITLE(20), TITLE2(20)
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/CCOM/ SOURSW,UPDNSW,LABEL,XSW,ISW,NKINFC,
1 NDLM2C, NGAMIN
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, R, TMPBOT,
1 TMPTOP, M
COMMON /DCOM/ SGA2H, SGA2S, PNSAAV
COMMON /HCOM/ LAST, ITR, ITRMAX, I1, TIME, TIMEO, SWCONL, IND,
1 TIMIN, RST2
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LOOP
COMMON/DCOM/LAMSW,RSTRT,NSW2,I4,EPSREF,EPLREF,EPSO,EPSLRF,
1 LAMREF, EPSLAM
COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /ICOM/ KINFC(19,19,10),DELM2C(19,19,10),SGA2(19,19),
1 L2(19,19),M2IN(19,19),KINFIN(19,19),PNF2(19,19)
COMMON /PCOM/ DELTX,DELTZ,V,DTIME
COMMON /RCOM/ NSWXIO, NSWIIQ, NSWFIQ, NSWPIO
INTEGER*2 UPDOWN,SOURSW,UPDNSW,XSW,SWCONL,RST2,FUEL,
1 RSTRT, SUMCHN
REAL KAPPA, NUSGGM, M, LAMREF, IOD, KINFC, L2, M2IN, <INFIN
READ (5,1000) (TITLE2(I), I = 1, 20)
1000 FORMAT (20A4)
READ (5,1001) NSW3, XSW, ISW, NKINFC, NDLM2C, NGAMIN, NSW5,
1 NSWXIO, NSWIIQ, NSWFIQ, NSWPIO
IF ((NSW5 .EQ. 0) .AND. (NSW3 .EQ. 1)) GO TO 150
IF (TIME .GT. TIMIN) GO TO 74
IF ((NSW1 .EQ. 1) .AND. (NSW3 .EQ. 0)) GO TO 151
IF ((NSW4 .EQ. 1) .AND. (NSW5 .EQ. 0)) GO TO 152
74 IF (NSW3 .EQ. 0) GO TO 75
IF (UPDNSW .GT. 0) GO TO 77
READ (5,1002) DTIME, PCORE, SGA2H, W, TMPBOT
GO TO 81
75 READ (5,1002) DTIME, PCORE, SGA2H
GO TO 81
77 READ (5,1002) DTIME, PCORE, SGA2H, W, TMPBOT, TMPTOP
81 DTIME = DTIME * 3600.0
IF (TIME .GT. TIMIN) GO TO 130
REWIND 18
IF (NGAMIN .EQ. 0) WRITE (6,1010)
DO 82 K = 1, KMAX

```

```

      DO 82 II = 1, 4
      READ (10) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
82  WRITE (18) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
1010 FORMAT ('O GAMMA MUST BE INPUT AT INITIAL TIME')
      GO TO 140
130  IF (NGAMIN .EQ. 0) GO TO 138
140  REWIND 18
      DO 131 II = 1, NGAMIN
      READ (5,1006) I1, J1, K1, I2, J2, K2, GAM
1006  FORMAT (3I5, 5X, 3I5, 5X, E10.0)
1001  FORMAT (11I4)
1002  FORMAT (8E10.0)
1005  FORMAT (5(2I3, E10.0))
      DO 131 I = I1, I2
      DO 131 J = J1, J2
      DO 131 K = K1, K2
      SGGM(I,J,K) = GAM
131  NUSGGM(I,J,K) = GAM
132  DO 137 K = 1, KMAX
      READ (18) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 133 I = 1, IMAX
      DO 133 J = 1, JMAX
133  SGGM(I,J,K) = SGF(I,J) * SGGM(I,J,K)
      READ (18) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 134 I = 1, IMAX
      DO 134 J = 1, JMAX
134  SGGM(I,J,K) = SGGM(I,J,K) + SGF(I,J)
      READ (18) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 135 I = 1, IMAX
      DO 135 J = 1, JMAX
135  NUSGGM(I,J,K) = SGF(I,J) * NUSGGM(I,J,K)
      READ (18) ((SGF(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 136 I = 1, IMAX
      DO 136 J = 1, JMAX
136  NUSGGM(I,J,K) = NUSGGM(I,J,K) + SGF(I,J)
137  CONTINUE
138  IF ((TIME .GT. TIMIN) .OR. (RSTRT .EQ. 0)) GO TO 139
      REWIND 15
      DO 145 K = 1, KMAX
145  READ (20) ((X(I,J,K), J = 1, JMAX), I = 1, IMAX)
      DO 146 K = 1, KMAX
      READ (20) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
146  WRITE (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
139  IF ((TIME .GT. TIMIN) .OR. (RSTRT .EQ. 1)) GO TO 85
      DO 83 K = 1, KMAX
      DO 83 I = 1, IMAX
      DO 83 J = 1, JMAX
83  X(I,J,K) = 0.0

```

```

85 IF (XSW .EQ. 0) GO TO 90
   DO 88 K = 1, KMAX
   READ (5,1001) NOX
   IF (NOX .EQ. 0) GO TO 88
87 READ (5,1005) (I,J,X(I,J,K), II = 1, NOX)
88 CONTINUE
90 IF ((TIME .GT. TIMIN) .AND. (ISW .EQ. 0)) GO TO 110
   REWIND 16
   IF ((TIME .GT. TIMIN) .OR. (RSTRT .EQ. 1)) REWIND 15
   DO 98 K = 1, KMAX
   IF ((TIME .GT. TIMIN) .OR. (RSTRT .EQ. 1)) GO TO 94
   DO 93 I = 1, IMAX
   DO 93 J = 1, JMAX
93 IOD(I,J) = 0.0
   GO TO 95
94 READ (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
95 IF (ISW .EQ. 0) GO TO 97
   READ (5,1001) NOI
   IF (NOI .EQ. 0) GO TO 97
   READ (5,1005) (I, J, IOD(I,J), II = 1, NOI)
97 WRITE (16) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
98 CONTINUE
   REWIND 15
   REWIND 16
   DO 99 K = 1, KMAX
   READ (16) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
99 WRITE (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
110 IF (TIME .GT. TIMIN) GO TO 102
   DO 100 K = 1, KMAX
   DO 100 I = 1, IMAX
   DO 100 J = 1, JMAX
   KINFC(I,J,K) = 0.0
100 DELM2C(I,J,K) = 0.0
102 IF((NKINFC.EQ.0).AND.(NDLM2C.EQ.0)) GO TO 1500
   DO 103 K=1,KMAX
   IF (NKINFC.EQ.0) GO TO 104
   READ (5,1001) NOC
   IF (NOC.EQ.0) GO TO 104
   READ (5,1005) (I,J,KINFC(I,J,K),II=1,NOC)
104 IF(NDLM2C.EQ.0) GO TO 103
   READ (5,1001) NOC
   IF (NOC.EQ.0) GO TO 103
   READ (5,1005) (I,J,DELM2C(I,J,K),II=1,NOC)
103 CONTINUE
1500 IF ((TIME .EQ. TIMIN) .AND. (RSTRT .EQ. 1)) REWIND 20
   RETURN
150 WRITE (6,1011)
1011 FORMAT ('YOU MAY NOT HAVE MODERATOR FEEDBACK WITHOUT A DOPPLER PD
   1WER UPDATE')
   STOP
151 WRITE (6,1012)
1012 FORMAT ('IF MODERATOR FEEDBACK IS REQUESTED THERE MUST BE AN UPDA
   1TE AT INITIAL TIME')
   STOP
152 WRITE (6,1013)
1013 FORMAT ('IF DOPPLER FEEDBACK IS REQUESTED THERE MUST BE AN UPDATE
   1 AT INITIAL TIME')
   STOP
END

```

C

```

SUBROUTINE OUTIN(INPUTX)
OUTIN PRINTS INPUT INFO
IMPLICIT INTEGER*2 (I-N)
INTEGER*4 LABEL, I, J, K, INPUTX, INTRO
REAL KAPPA, NUSGGM, LAMREF, LAMBDA, LOLD, M, IOD, KINFC, L2, M2IN,
1 KINFIN
INTEGER*2 UPDOWN, RSTRT, SWCONL, RST2, SOURSW, UPDNSW, XSW, FUEL,
1 SUMCHN
COMMON /ACOM/ TITLE(20), TITLE2(20)
COMMON/BCOM/F(19,19),PCORE,W,KAPPA,SGGM(19,19,10),NUSGGM(19,19,10)
1,UPDOWN(19,19),NSW1,NSW3,NSW4,NSW5,NSW6,NFIN
COMMON/CCOM/ SOURSW,UPDNSW,LABEL,XSW,ISW,NKINFC,
1 NDLM2C, NGAMIN
COMMON/DCOM/LAMSW,RSTRT,NSW2,I4,EPSREF,EPLREF,EPSO,EPSLRF,
1 LAMREF, EPSLAM
COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LODP
COMMON /GCOM/ ITRAVG
COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEO,SWCONL,IND,
1 TIMIN, RST2
COMMON /ICOM/ KINFC(19,19,10),DELM2C(19,19,10),SGA2(19,19),
1 L2(19,19),M2IN(19,19),KINFIN(19,19),PNF2(19,19)
COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1 TMPTOP, M
COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1 SOLD(19,19,10),SOLDT(19,19),
1 LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMJ
COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1 SUMNOD, SA2XEF(19,19,10), SUMCHN
COMMON /QCOM/ SGA2H, SGA2S, PNSAAV
COMMON /PCOM/ DELTX,DELTZ,V,DTIME
COMMON /QCOM/ CP, DKDT, DMDT, DKDP, DMOP, TREF, PREF, INTRO
DIMENSION RST(4), HOLL(100), NOPT(2,10), JMAXX(3), HJL(70)
DATA RST /'STAR','TUP','REST','ART' /
DATA HJL / 'MODE','RATO','R UP','DATE',' ',' ',' ',' ',' ',' ',
1 ' ',' ',' ',' ',' ','DOPP','LER','UPDA','TE',' ',' ',' ',
1 ' ',' ',' ',' ',' ',' ',' ','XEND','N CO','NCEN',
1 'TRAT','IONS','INP','UT',' ',' ',' ',' ',' ','IODI',
1 'NE C','ONCE','NTRA','TION','S IN','PUT',' ',' ',' ',
1 ' ',' ','K IN','FINI','TE C','ONTR','OL I','NPUT',' ',' ',
1 ' ',' ',' ',' ',' ','M**2','CON','TROL','INP','UT',
1 ' ',' ',' ',' ',' ',' ',' ','NEW','GAMM','A IN',
1 'PUT',' ',' ',' ',' ',' ',' ',' ',' ',' '/
DATA HOLL /'A RE','STAR','T TA','PE W','ILL','BE G','ENER',
1 'ATED','AT','END','MODE','RATO','R FE','EDBA','CK',
1 ' ',' ',' ',' ',' ',' ',' ','DOPP','LER','FEED',
1 'BACK',' ',' ',' ',' ',' ',' ',' ','EIGE',
1 'NVAL','UE S','EARC','H',' ',' ',' ',' ',' ',
1 ' ','REFE','RENC','E EI','GEN','VALU','E IN','PUT'

```

```

1      'DOWN', 'WARD', 'FLO', 'W PR', 'ESEN',
1      'T', 'INTE', 'RPOL', 'ATIO',
1      'N OF', 'FEE', 'DBAC', 'K TA', 'BLES', 'MU I',
1      'INPUT', 'FOR', 'TRA', 'NSPO', 'RT K', 'ERNE', 'LS',
1      'SOUR', 'CE G', 'UESS', 'INP', 'UT',
1      'FLOW', 'FAC', 'TORS', 'INP', 'UT',
1      '/'

      IF (JMAX .GT. 8) GO TO 1
      JMAXX(1) = JMAX
      NMAX = 1
      GO TO 3
1     JMAXX(1) = 8
      IF (JMAX .GT. 16) GO TO 2
      JMAXX(2) = JMAX
      NMAX = 2
      GO TO 3
2     JMAXX(2) = 16
      JMAXX(3) = JMAX
      NMAX = 3
3     IF (INPUTX .EQ. 0) GO TO 100
      WRITE (6,1000) (TITLE(I), I = 1, 20)
1000  FORMAT ('1', // 1X, 'INPUT DATA', // 1X, 20A4)
      I1 = 2 * RSTRT + 1
      I2 = I1 + 1
      WRITE (6,1001) LABEL, RST(I1), RST(I2)
1001  FORMAT ('OINCEXE TAPE LABELED ', A4, ' TO BE USED      THIS IS A ',
1     2A4, 'PROBLEM')
      NOPT(1,1) = RST2
      NOPT(1,2) = NSW1
      NOPT(1,3) = NSW4
      NOPT(1,4) = LAMSW
      NOPT(1,5) = NSW2
      NOPT(1,6) = UPONSW
      NOPT(1,7) = -INTROP + 1
      NOPT(1,8) = NSW6 - 1
      NOPT(1,9) = SOURS
      NOPT(1,10) = NFIN
      NOPT(2,1) = 1
      NOPT(2,2) = 11
      NOPT(2,3) = 21
      NOPT(2,4) = 31
      NOPT(2,5) = 41
      NOPT(2,6) = 51
      NOPT(2,7) = 61
      NOPT(2,8) = 71
      NOPT(2,9) = 81
      NOPT(2,10) = 91
      NCOUNT = 0

```



```

DO 11 NO = 1, 10
IF (NOPT(1,NO) .EQ. 0) GO TO 11
NCOUNT = NCOUNT + 1
IF (NCOUNT .EQ. 1) GO TO 10
I3 = NOPT(2,NO)
I4 = I3 + 9
WRITE (6,1002) (HOLL(I), I = 11, I2), (HOLL(II), II = I3, I4)
NCOUNT = 0
GO TO 11
10 I1 = NOPT(2,NO)
I2 = I1 + 9
11 CONTINUE
1002 FORMAT (1X, 20A4)
IF (NCOUNT .NE. 1) GO TO 12
WRITE (6,1003) (HOLL(I), I = 11, I2), ITRMAX
1003 FORMAT (1X, 10A4, 'MAX ITERATIONS PER SOURCE CALC =', I3)
IF (ITRAVG .LE. ITRMAX) WRITE (6,1004) ITRAVG
1004 FORMAT (1X, 'SOURCE AVERAGING STARTS ON ITERATION', I3)
GO TO 14
12 IF (ITRAVG .LE. ITRMAX) GO TO 13
WRITE (6,1005) ITRMAX
1005 FORMAT (1X, 'MAX ITERATIONS PER SOURCE CALC =', I3)
GO TO 14
13 WRITE (6,1006) ITRMAX, ITRAVG
1006 FORMAT (1X, 'MAX ITERATIONS PER SOURCE CALC =', I3, 5X, 'SOURCE A
1VERAGING STARTS ON ITERATION', I3)
14 WRITE (6,1007) DELTX, DELTZ, EPSREF, EPLREF, EPSO, EPSLRF
1007 FORMAT ('0 DELTA X DELTA Z DEL S DEL EIG DE
1L EQUIL DEL EIG SRCH' / 1X, 1P6E13.5)
WRITE (6,1008) GH, GV, A, B, LAMREF, HMJ, VMU
1008 FORMAT ('0 GH GV A B EI
1G REF MU HORIZ MU VERT' / 1X, 1P7E13.5)
WRITE (6,1009) LAMBDA, KAPPA, DECAYI, GAMMAI, DECAYX, GAMMAX
1009 FORMAT ('0 EIGEN KAPPA IOD DECAY IOD GAMMA XE
1N DECAY XEN GAMMA' / 1X, 1P6E13.5)
IF (UPDNSW .EQ. 0) GO TO 16
WRITE (6,1010)
1010 FORMAT('1', '///// ' UP OR DOWN FLOW' / )
DO 15 I = 1, IMAX
15 WRITE (6,1011) (UPDOWN(I,J), J = 1, JMAX)
1011 FORMAT (1X, 19I3)
16 IF (NFIN .EQ. 0) GO TO 19
MIN = 1
DO 18 N = 1, NMAX
MAX = JMAXX(N)
WRITE (6,1012) (II, II = MIN, MAX)
DO 17 I = 1, IMAX
17 WRITE (6,1013) I, (F(I,J), J = MIN, MAX)

```

```

18 MIN = MAX + 1
1012 FORMAT ('1' // 'FLOW FACTORS' / '0', 8X, 12, 7I15)
1013 FORMAT ('0', 12, 1P8E15.6)
19 IF (RSTRT .EQ. 1) GO TO 20
   IF (SOURSW .EQ. 0) GO TO 23
20 DO 22 K = 1, KMAX
   MIN = 1
   DO 22 N = 1, NMAX
   MAX = JMAXX(N)
   WRITE (6,1014) K, (II, II = MIN, MAX)
   DO 21 I = 1, IMAX
21 WRITE (6,1013) I, (S(I,J,K), J = MIN, MAX)
22 MIN = MAX + 1
1014 FORMAT ('1' // 'SOURCE GUESS' / '0', 8X, 12, 7I15)
23 MIN = 1
   DO 25 N = 1, NMAX
   MAX = JMAXX(N)
   WRITE (6,1015) (II, II = MIN, MAX)
   DO 24 I = 1, IMAX
24 WRITE (6,1013) I, (ALPHAH(I,J), J = MIN, MAX)
25 MIN = MAX + 1
1015 FORMAT ('1' // '4.0 - ALBEDO HORIZONTAL' / '0', 8X, 12, 7I15)
   WRITE (6,1016)
1016 FORMAT ('1' // '2.0 - ALBEDO VERTICAL' //)
   DO 26 K = 1, KMAX
26 WRITE (6,1017) K, ALPHAV(K)
1017 FORMAT (40X, 13, 2X, 1P8E15.6)
100 NOPT(1,1) = NSW3
   NOPT(1,2) = NSW5
   NOPT(1,3) = XSW
   NOPT(1,4) = ISW
   NOPT(1,5) = NKINFC
   NOPT(1,6) = NDLM2C
   NOPT(1,7) = NGAMIN
   NOPT(2,1) = 1
   NOPT(2,2) = 11
   NOPT(2,3) = 21
   NOPT(2,4) = 31
   NOPT(2,5) = 41
   NOPT(2,6) = 51
   NOPT(2,7) = 61
   WRITE (6,1000) (TITLE(I), I = 1, 20)
   WRITE (6,1018) (TITLE2(I), I = 1, 20)
1018 FORMAT ('0', 1X, 20A4, /)
   NCOUNT = 0
   DO 102 NO = 1, 7
   IF (NOPT(1,NO) .EQ. 0) GO TO 102
   NCOUNT = NCOUNT + 1

```

```

      IF (NCOUNT .EQ. 1) GO TO 101
      I3 = NOPT(2,NO)
      I4 = I3 + 9
      WRITE (6,1002) (HOL(I), I = I1, I2), (HOL(II), II = I3, I4)
      NCOUNT = 0
      GO TO 102
101  I1 = NOPT(2,NO)
      I2 = I1 + 9
102  CONTINUE
      IF (NCOUNT .EQ. 1) WRITE (6,1002) (HOL(I), I = I1, I2)
      DUMMY = DTIME / 3600.0
      WRITE (6,1019) DUMMY, PCORE, SGA2H, W, TMPROT, TMPTOP
1019  FORMAT ('0' DEL TIME CORE POWER THERMAL POISON FLOW RA
      ITE TEMP BOTTOM TEMP TOP / 1X, 1P6E15.6 )
106  IF ((PSTRT .EQ. 1) .AND. (TIME .EQ. TIMIN)) GO TO 107
      IF (XSW .EQ. 0) GO TO 110
107  DO 109 K = 1, KMAX
      MIN = 1
      DO 109 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1021) K, (II, II = MIN, MAX)
      DO 108 I = 1, IMAX
108  WRITE (6,1013) I, (X(I,J,K), J = MIN, MAX)
109  MIN = MAX + 1
1021  FORMAT ('1' // 'NEW XENON CONCENTRATIONS' K =', I2 /
1      '0', 8X, I2, 7I15)
110  IF ((RSTRT .EQ. 1) .AND. (TIME .EQ. TIMIN)) GO TO 111
      IF (ISW .EQ. 0) GO TO 114
111  REWIND 15
      DO 113 K = 1, KMAX
      READ (15) (IOD(I,J), J = 1, JMAX), I = 1, IMAX)
      MIN = 1
      DO 113 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1022) K, (II, II = MIN, MAX)
      DO 112 I = 1, IMAX
112  WRITE (6,1013) I, (IOD(I,J), J = MIN, MAX)
113  MIN = MAX + 1
1022  FORMAT ('1' // 'NEW IODINE CONCENTRATIONS' K =', I2 /
1      '0', 8X, I2, 7I15)
114  IF (NKINFC .EQ. 0) GO TO 117
      DO 116 K = 1, KMAX
      MIN = 1
      DO 116 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1023) K, (II, II = MIN, MAX)
      DO 115 I = 1, IMAX
115  WRITE (6,1013) I, (KINFC(I,J,K), J = MIN, MAX)
116  MIN = MAX + 1
1023  FORMAT ('1' // 'K INFINITY CONTROL' K =', I2 /
1      '0', 8X, I2, 7I15)
117  IF (NDLM2C .EQ. 0) GO TO 120
      DO 119 K = 1, KMAX
      MIN = 1
      DO 119 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1024) K, (II, II = MIN, MAX)
      DO 118 I = 1, IMAX
118  WRITE (6,1013) I, (DELM2C(I,J,K), J = MIN, MAX)
119  MIN = MAX + 1
1024  FORMAT ('1' // 'M**2 CONTROL' K =', I2 /
1      '0', 8X, I2, 7I15)
120  RETURN
      END

```

```

SUBROUTINE OUTPUT
C  OUTPUT PRINTS CALCULATED INFO
  IMPLICIT INTEGER * 2 (I-N)
  INTEGER*4 I, J, K
  COMMON /ACOM/ TITLE(20), TITLE2(20)
  COMMON/BCOM/F(19,19), PCORE, W, KAPPA, SGGM(19,19,10), NUSGGM(19,19,10)
1  ,UPDOWN(19,19), NSW1, NSW3, NSW4, NSW5, NSW6, NFIN
  COMMON /HCOM/ LAST, ITR, ITRMAX, I1, TIME, TIME0, SWCONL, IND,
1  TIMIN, RST2
  COMMON /PCOM/ DELTX, DELTZ, V, DTIME
  COMMON /OCOM/ SGA2H, SGA2S, PNSAAV
  COMMON/MCOM/ S(19,19,10), ALPHAV(10), ALPHAH(19,19),
1  SOLD(19,19,10), SOLDT(19,19),
1  LAMBDA, LOLD, GV, GH, ITRSRC, I3, HMU, VMU
  COMMON/FCOM/FUEL(19,19), IMAX, JMAX, KMAX, I, J, K, LOOP
  COMMON/NCOM/ X(19,19,10), IOD(19,19), DECAYX, DECAYI, GAMMAX, GAMMAI,
1  SUMNOD, SA2XEF(19,19,10), SUMCHN
  COMMON /LCOM/ T(19,19,10), PHI2P, P(19,19,10), A, B, TMPBOT,
1  TMPTOP, M
  COMMON /RCOM/ NSWXIO, NSWIIO, NSWFIO, NSWPIO
  REAL KAPPA, NUSGGM, LAMBDA, LOLD, IOD, M
  INTEGER * 2      UPDOWN, SWCONL, RST2, FUEL, SUMCHN
  DIMENSION DUM(19)
  DIMENSION PLG1(6), PLG2(6), PLG3(6), PLG4(6)
  DIMENSION JMAXX(3)
  DATA PLG1/'XENO', 'N CO', 'NCEN', 'TRAT', 'IONS', ' ' //
  DATA PLG2/'IODI', 'NE C', 'ONCE', 'NTRA', 'TION', 'S ' //
  DATA PLG3/'THER', 'MAL ', 'FLUX', 'ES ', ' ' //
  DATA PLG4/'POWE', 'RS ', ' ', ' ', ' ', ' ' //
  REWIND 15
  WRITE (6,1000) (TITLE(I), I = 1, 20)
1000 FORMAT ('1', // 1X, 20A4)
  WRITE (6,1005) (TITLE2(I), I = 1, 20)
1005 FORMAT ('0', 20A4, //)
  DUMMY = TIME / 3600.0
  IF (NSW3 .EQ. 0) GO TO 9
  WRITE (6,1001) DUMMY, W, PCORE, LAMBDA
1001 FORMAT (' TIME =', E10.5, // ' AVERAGE COOLANT FLOW RATE =', E15.7,
1  // ' TOTAL CORE POWER =', E15.7, // ' EIGENVALUE =', E15.7)
  GO TO 11
  9 WRITE (6,1002) DUMMY, PCORE, LAMBDA
1002 FORMAT (' TIME =', F10.5, // ' TOTAL CORE POWER =', E15.7, //
1  ' EIGENVALUE =', F15.7 )
  11 IF (JMAX .GT. 8) GO TO 10
  JMAXX(1) = JMAX
  IMAX = 1
  GO TO 20
  10 JMAXX(1) = 8

```

```

      IF (JMAX .GT. 16) GO TO 12
      JMAXX(2) = JMAX
      NMAX = 2
      GO TO 20
12  JMAXX(2) = 16
      JMAXX(3) = JMAX
      NMAX = 3
20  IF (NSWXIO .EQ. 0) GO TO 26
      DO 25 K = 1, KMAX
      MIN = 1
      DO 25 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1003) PLG1, K, DUMMY, (II, II = MIN, MAX)
      DO 24 I = 1, IMAX
24  WRITE (6,1004) I, (X(I,J,K), J = MIN, MAX)
25  MIN = MAX + 1
26  IF (NSWIIIO .EQ. 0) GO TO 31
      DO 30 K = 1, KMAX
      MIN = 1
      READ (15) ((IIO(I,J), J = 1, JMAX), I = 1, IMAX)
      DO 30 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1003) PLG2, K, DUMMY, (II, II = MIN, MAX)
      DO 29 I = 1, IMAX
29  WRITE (6,1004) I, (IIO(I,J), J = MIN, MAX)
30  MIN = MAX + 1
31  IF (NSWFIO .EQ. 0) GO TO 36
      DO 35 K = 1, KMAX
      MIN = 1
      DO 35 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1003) PLG3, K, DUMMY, (II, II = MIN, MAX)
      DO 34 I = 1, IMAX
      DO 33 J = MIN, MAX
      IF (FUEL(I,J) .NE. 0) GO TO 32
      DUM(J) = 0.0
      GO TO 33
32  PHI2P = LAMBDA * S(I,J,K) / NUSGGM(I,J,K)
      DUM(J) = M * PHI2P
33  CONTINUE
34  WRITE (6,1004) I, (DUM(J), J = MIN, MAX)
35  MIN = MAX + 1
36  IF (NSWPID .EQ. 0) RETURN
      DO 40 K = 1, KMAX
      MIN = 1
      DO 40 N = 1, NMAX
      MAX = JMAXX(N)
      WRITE (6,1003) PLG4, K, DUMMY, (II, II = MIN, MAX)
      DO 39 I = 1, IMAX
      DO 38 J = MIN, MAX
      IF (FUEL(I,J) .NE. 0) GO TO 37
      DUM(J) = 0.0
      GO TO 38
37  PHI2P = LAMBDA * S(I,J,K) / NUSGGM(I,J,K)
      DUM(J) = M * PHI2P * KAPPA * SGGM(I,J,K) * V
38  CONTINUE
39  WRITE (6,1004) I, (DUM(J), J = MIN, MAX)
40  MIN = MAX + 1
1003 FORMAT('1', //20X, 6A4, ' K=', I2, 10X, 'TIME=', E10.5, // 9X, I2,
1      7I15)
1004 FORMAT('0', I2, 1P8E15.6)
      RETURN
      END

```

```

SUBROUTINE STORE
C  STORE GENERATES RESTART TAPES
  IMPLICIT INTEGER * 2 (I-N)
  INTEGER *4 NT      , I, J, K
  COMMON/DCOM/LAMSW,RSTRT,NSW2,I4,EPSREF,EPLREF,EPSO,EPSLRF,
1  LAMREF, EPSLAM
  COMMON /HCOM/ LAST,ITR,ITRMAX,I1,TIME,TIMEO,SWCONL,IND,
1  TIMIN, RST2
  COMMON/MCOM/ S(19,19,10),ALPHAV(10),ALPHAH(19,19),
1  SOLD(19,19,10),SOLDT(19,19),
1  LAMBDA,LOLD,GV,GH,ITRSRC,I3,HMU,VMU
  COMMON /QCOM/ SGA2H, SGA2S, PNSAAV
  COMMON/FCOM/FUEL(19,19),IMAX,JMAX,KMAX,I,J,K,LODP
  COMMON/NCOM/ X(19,19,10),IOD(19,19),DECAYX,DECAYI,GAMMAX,GAMMAI,
1  SUMNDD, SA2XEF(19,19,10), SUMCHN
  REAL LAMBDA, LOLD, IOD, LAMREF
  INTEGER * 2 SWCONL,RST2,FUEL,SUMCHN,RSTRT
  REWIND 15
C  UNIT 21 IS GENERATED AT EQUILIBRIUM TIME
  NT = 21
C  UNIT 24 IS GENERATED AT END TIME
  IF (TIME .GT. TIMEO) NT = 24
  WRITE (NT) TIME, TIMEO, LAMBDA, LAMREF, SGA2S, PNSAAV
  WRITE (NT) ((S(I,J,K), K = 1, KMAX), J = 1, JMAX), I = 1, IMAX)
  DO 10 K = 1, KMAX
10  WRITE (NT) ((X(I,J,K), J=1,JMAX), I=1,IMAX)
  DO 20 K = 1, KMAX
  READ (15) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
20  WRITE (NT) ((IOD(I,J), J = 1, JMAX), I = 1, IMAX)
  RETURN
  END

```

VIII. REFERENCES

1. R. Kern, M. Bonaca, Z. R. Rosztoczy, and W. C. Coppersmith, "Optimum Spatial Flux Control Project: Phase II - Xenon Oscillation Studies for the Pickering Reactor, "CEND-3932-2-350, Combustion Engineering, Inc., Nuclear Power Department, May 31, 1969.
2. D. L. Delp, D. L. Fischer, J. M. Harriman, and M. J. Stedwell, "FLARE: A Three-Dimensional Boiling Water Reactor Simulator," GEAP-4598, General Electric Co., 1964.
3. M. F. Valerino and R. C. Kern, "Optimum Spatial Flux Control Project - Monthly Progress Letter No. 9," RTA-A-73, Combustion Engineering, Inc., October 23, 1968.
4. M. F. Valerino and R. C. Kern, "Optimum Spatial Flux Control Project - Monthly Progress Letter No. 10," RTA-A-76, Combustion Engineering, Inc., November 11, 1968.
5. M. F. Valerino and R. C. Kern, "Optimum Spatial Flux Control Project - Monthly Progress Letter No. 6," RTA-A-50, Combustion Engineering, Inc., July 10, 1968.
6. C. G. Poncelet and A. M. Christie, "Xenon-Induced Spatial Instabilities in Large Pressurized Water Reactors," WCAP-3680-20, Westinghouse Electric Corp., March 1968.
7. M. F. Valerino and R. C. Kern, "Optimum Spatial Flux Control Project - Monthly Progress Letter No. 3," RTA-A-28, Combustion Engineering, Inc., April 16, 1968.
8. M. F. Valerino and R. C. Kern, "Optimum Spatial Flux Control Project - Monthly Progress Letter No. 8," RTA-A-63, Combustion Engineering, Inc., September 10, 1968.

Figure 1
SIMPLIFIED FLOW DIAGRAM OF THE LOGISTICS
OF THE CEXE CODE

Figure 2 - Sheet 1 of 5
DETAILED FLOW DIAGRAM OF THE LOGISTICS OF THE CEXE CODE

Figure 2 - Sheet 2 of 5

[illegible]

ITR_{avg} Is Iteration No.
For Which Source
Calculations Become
Based on Source
Averaging Technique

Figure 2 - Sheet 4 of 5

Figure 2 - Sheet 5 of 5

Figure 3

INPUT LISTING FOR A SAMPLE INCXE PROBLEM

COLUMN NUMBERS

0000000001111111112222222222333333333344444444445555555555666666666677777777778
12345678901234567890123456789012345678901234567890123456789012345678901234567890

PC3D

11	11	10	1	1	0	0	0				
0	0	0	1	1	1	1	1	0	0	0	
0	0	1	1	1	1	1	1	1	0	0	
0	1	1	1	1	1	1	1	1	1	0	
1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	
0	1	1	1	1	1	1	1	1	1	0	
0	0	1	1	1	1	1	1	1	0	0	
0	0	0	1	1	1	1	1	0	0	0	

0.0

0.0

0.0018604

0.0

0.0048929

3.5000E-18

0.0040878

227.555

374.926

1.07343

1.07342

Figure 4
INPUT LISTING FOR A SAMPLE CEXE PROBLEM
Sheet 1 of 5

COLUMN NUMBERS

00000000111111112222222233333333444444445555555566666666777777778
1234567890123456789012345678901234567890123456789012345678901234567890

CEXE 3 DIMENSIONAL PICKERING REACTOR - ESTABLISHING EQUILIBRIUM CORE CONDITIONS

0	0	0	0	0	0	0	60	155	0	2	0	OPC3D
57.15		59.4		.000010		.000010		.000010		.000010		
1.0		1.0		1.0		1.0		0.0		0.0		
1.0		3.250E-17										
2.9390E-05		0.0618		2.100E-05		0.0048						
-0.8		-0.8										
28												
1 4	1.19	1 5	0.45	1 6	0.45	1 7	0.45	1 8	1.19			
2 3	1.28	2 9	1.28	3 2	1.28	3 10	1.28	4 1	1.22			
4 11	1.22	5 1	0.45	5 11	0.45	6 1	0.46	6 11	0.46			
7 1	0.45	7 11	0.45	8 1	1.18	8 11	1.18	9 2	1.25			
9 10	1.25	10 3	1.25	10 9	1.25	11 4	1.15	11 5	0.45			
11 6	0.46	11 7	0.45	11 8	1.15							

Figure 4
INPUT LISTING FOR A SAMPLE CEXE PROBLEM
Sheet 2 of 5

COLUMN NUMBERS

0000000001111111112222222223333333334444444445555555556666666667777777778
1234567890123456789012345678901234567890123456789012345678901234567890

ESTABLISHING EQUILIBRIUM

0 0 0 1 1 1 0 1 1 1 1
40. 1621.6 0.0 0.0 0.0 0.0
1 1 1 11 11 10 1.0

0
0
0
0
45

3 4 -0.01131	3 5 -0.02261	3 6 -0.02261	3 7 -0.02261	3 8 -0.01131
4 3 -0.01131	4 4 -0.02261	4 5 -0.02261	4 6 -0.02261	4 7 -0.02261
4 8 -0.02261	4 9 -0.01131	5 3 -0.01131	5 4 -0.02261	5 5 -0.02261
5 6 -0.02261	5 7 -0.02261	5 8 -0.02261	5 9 -0.01131	6 3 -0.01131
6 4 -0.02261	6 5 -0.02261	6 6 -0.02261	6 7 -0.02261	6 8 -0.02261
6 9 -0.01131	7 3 -0.01131	7 4 -0.02261	7 5 -0.02261	7 6 -0.02261
7 7 -0.02261	7 8 -0.02261	7 9 -0.01131	8 3 -0.01131	8 4 -0.02261
8 5 -0.02261	8 6 -0.02261	8 7 -0.02261	8 8 -0.02261	8 9 -0.01131
9 4 -0.01131	9 5 -0.02261	9 6 -0.02261	9 7 -0.02261	9 8 -0.01131

45

3 4 -2.384	3 5 -4.769	3 6 -4.769	3 7 -4.769	3 8 -2.384
4 3 -2.384	4 4 -4.769	4 5 -4.769	4 6 -4.769	4 7 -4.769
4 8 -4.769	4 9 -2.384	5 3 -2.384	5 4 -4.769	5 5 -4.769
5 6 -4.769	5 7 -4.769	5 8 -4.769	5 9 -2.384	6 3 -2.384
6 4 -4.769	6 5 -4.769	6 6 -4.769	6 7 -4.769	6 8 -4.769
6 9 -2.384	7 3 -2.384	7 4 -4.769	7 5 -4.769	7 6 -4.769
7 7 -4.769	7 8 -4.769	7 9 -2.384	8 3 -2.384	8 4 -4.769
8 5 -4.769	8 6 -4.769	8 7 -4.769	8 8 -4.769	8 9 -2.384
9 4 -2.384	9 5 -4.769	9 6 -4.769	9 7 -4.769	9 8 -2.384

45

3 4 -0.02119	3 5 -0.04237	3 6 -0.04237	3 7 -0.04237	3 8 -0.02119
4 3 -0.02119	4 4 -0.04237	4 5 -0.04237	4 6 -0.04237	4 7 -0.04237
4 8 -0.04237	4 9 -0.02119	5 3 -0.02119	5 4 -0.04237	5 5 -0.04237
5 6 -0.04237	5 7 -0.04237	5 8 -0.04237	5 9 -0.02119	6 3 -0.02119
6 4 -0.04237	6 5 -0.04237	6 6 -0.04237	6 7 -0.04237	6 8 -0.04237
6 9 -0.02119	7 3 -0.02119	7 4 -0.04237	7 5 -0.04237	7 6 -0.04237
7 7 -0.04237	7 8 -0.04237	7 9 -0.02119	8 3 -0.02119	8 4 -0.04237
8 5 -0.04237	8 6 -0.04237	8 7 -0.04237	8 8 -0.04237	8 9 -0.02119
9 4 -0.02119	9 5 -0.04237	9 6 -0.04237	9 7 -0.04237	9 8 -0.02119

Figure 4
INPUT LISTING FOR A SAMPLE CEXE PROBLEM
Sheet 3 of 5

COLUMN NUMBERS

00000000011111111122222222233333333334444444445555555556666666667777777778
1234567890123456789012345678901234567890123456789012345678901234567890

45				
3 4 -4.468	3 5 -8.936	3 6 -8.936	3 7 -8.936	3 8 -4.468
4 3 -4.468	4 4 -8.936	4 5 -8.936	4 6 -8.936	4 7 -8.936
4 8 -8.936	4 9 -4.468	5 3 -4.468	5 4 -8.936	5 5 -8.936
5 6 -8.936	5 7 -8.936	5 8 -8.936	5 9 -4.468	6 3 -4.468
6 4 -8.936	6 5 -8.936	6 6 -8.936	6 7 -8.936	6 8 -8.936
6 9 -4.468	7 3 -4.468	7 4 -8.936	7 5 -8.936	7 6 -8.936
7 7 -8.936	7 8 -8.936	7 9 -4.468	8 3 -4.468	8 4 -8.936
8 5 -8.936	8 6 -8.936	8 7 -8.936	8 8 -8.936	8 9 -4.468
9 4 -4.468	9 5 -8.936	9 6 -8.936	9 7 -8.936	9 8 -4.468
45				
3 4 -0.02119	3 5 -0.04237	3 6 -0.04237	3 7 -0.04237	3 8 -0.02119
4 3 -0.02119	4 4 -0.04237	4 5 -0.04237	4 6 -0.04237	4 7 -0.04237
4 8 -0.04237	4 9 -0.02119	5 3 -0.02119	5 4 -0.04237	5 5 -0.04237
5 6 -0.04237	5 7 -0.04237	5 8 -0.04237	5 9 -0.02119	6 3 -0.02119
6 4 -0.04237	6 5 -0.04237	6 6 -0.04237	6 7 -0.04237	6 8 -0.04237
6 9 -0.02119	7 3 -0.02119	7 4 -0.04237	7 5 -0.04237	7 6 -0.04237
7 7 -0.04237	7 8 -0.04237	7 9 -0.02119	8 3 -0.02119	8 4 -0.04237
8 5 -0.04237	8 6 -0.04237	8 7 -0.04237	8 8 -0.04237	8 9 -0.02119
9 4 -0.02119	9 5 -0.04237	9 6 -0.04237	9 7 -0.04237	9 8 -0.02119
45				
3 4 -4.468	3 5 -8.936	3 6 -8.936	3 7 -8.936	3 8 -4.468
4 3 -4.468	4 4 -8.936	4 5 -8.936	4 6 -8.936	4 7 -8.936
4 8 -8.936	4 9 -4.468	5 3 -4.468	5 4 -8.936	5 5 -8.936
5 6 -8.936	5 7 -8.936	5 8 -8.936	5 9 -4.468	6 3 -4.468
6 4 -8.936	6 5 -8.936	6 6 -8.936	6 7 -8.936	6 8 -8.936
6 9 -4.468	7 3 -4.468	7 4 -8.936	7 5 -8.936	7 6 -8.936
7 7 -8.936	7 8 -8.936	7 9 -4.468	8 3 -4.468	8 4 -8.936
8 5 -8.936	8 6 -8.936	8 7 -8.936	8 8 -8.936	8 9 -4.468
9 4 -4.468	9 5 -8.936	9 6 -8.936	9 7 -8.936	9 8 -4.468
45				
3 4 -0.02119	3 5 -0.04237	3 6 -0.04237	3 7 -0.04237	3 8 -0.02119
4 3 -0.02119	4 4 -0.04237	4 5 -0.04237	4 6 -0.04237	4 7 -0.04237
4 8 -0.04237	4 9 -0.02119	5 3 -0.02119	5 4 -0.04237	5 5 -0.04237
5 6 -0.04237	5 7 -0.04237	5 8 -0.04237	5 9 -0.02119	6 3 -0.02119
6 4 -0.04237	6 5 -0.04237	6 6 -0.04237	6 7 -0.04237	6 8 -0.04237
6 9 -0.02119	7 3 -0.02119	7 4 -0.04237	7 5 -0.04237	7 6 -0.04237
7 7 -0.04237	7 8 -0.04237	7 9 -0.02119	8 3 -0.02119	8 4 -0.04237
8 5 -0.04237	8 6 -0.04237	8 7 -0.04237	8 8 -0.04237	8 9 -0.02119
9 4 -0.02119	9 5 -0.04237	9 6 -0.04237	9 7 -0.04237	9 8 -0.02119

Figure 4
INPUT LISTING FOR A SAMPLE CEXE PROBLEM
Sheet 4 of 5

COLUMN NUMBERS

00000000011111111122222222233333333344444444455555555566666666677777777778
1234567890123456789012345678901234567890123456789012345678901234567890

45														
3	4	-4.468	3	5	-8.936	3	6	-8.936	3	7	-8.936	3	8	-4.468
4	3	-4.468	4	4	-8.936	4	5	-8.936	4	6	-8.936	4	7	-8.936
4	8	-8.936	4	9	-4.468	5	3	-4.468	5	4	-8.936	5	5	-8.936
5	6	-8.936	5	7	-8.936	5	8	-8.936	5	9	-4.468	6	3	-4.468
6	4	-8.936	6	5	-8.936	6	6	-8.936	6	7	-8.936	6	8	-8.936
6	9	-4.468	7	3	-4.468	7	4	-8.936	7	5	-8.936	7	6	-8.936
7	7	-8.936	7	8	-8.936	7	9	-4.468	8	3	-4.468	8	4	-8.936
8	5	-8.936	8	6	-8.936	8	7	-8.936	8	8	-8.936	8	9	-4.468
9	4	-4.468	9	5	-8.936	9	6	-8.936	9	7	-8.936	9	8	-4.468
45														
3	4	-0.02119	3	5	-0.04237	3	6	-0.04237	3	7	-0.04237	3	8	-0.02119
4	3	-0.02119	4	4	-0.04237	4	5	-0.04237	4	6	-0.04237	4	7	-0.04237
4	8	-0.04237	4	9	-0.02119	5	3	-0.02119	5	4	-0.04237	5	5	-0.04237
5	6	-0.04237	5	7	-0.04237	5	8	-0.04237	5	9	-0.02119	6	3	-0.02119
6	4	-0.04237	6	5	-0.04237	6	6	-0.04237	6	7	-0.04237	6	8	-0.04237
6	9	-0.02119	7	3	-0.02119	7	4	-0.04237	7	5	-0.04237	7	6	-0.04237
7	7	-0.04237	7	8	-0.04237	7	9	-0.02119	8	3	-0.02119	8	4	-0.04237
8	5	-0.04237	8	6	-0.04237	8	7	-0.04237	8	8	-0.04237	8	9	-0.02119
9	4	-0.02119	9	5	-0.04237	9	6	-0.04237	9	7	-0.04237	9	8	-0.02119
45														
3	4	-4.468	3	5	-8.936	3	6	-8.936	3	7	-8.936	3	8	-4.468
4	3	-4.468	4	4	-8.936	4	5	-8.936	4	6	-8.936	4	7	-8.936
4	8	-8.936	4	9	-4.468	5	3	-4.468	5	4	-8.936	5	5	-8.936
5	6	-8.936	5	7	-8.936	5	8	-8.936	5	9	-4.468	6	3	-4.468
6	4	-8.936	6	5	-8.936	6	6	-8.936	6	7	-8.936	6	8	-8.936
6	9	-4.468	7	3	-4.468	7	4	-8.936	7	5	-8.936	7	6	-8.936
7	7	-8.936	7	8	-8.936	7	9	-4.468	8	3	-4.468	8	4	-8.936
8	5	-8.936	8	6	-8.936	8	7	-8.936	8	8	-8.936	8	9	-4.468
9	4	-4.468	9	5	-8.936	9	6	-8.936	9	7	-8.936	9	8	-4.468
45														
3	4	-0.01131	3	5	-0.02261	3	6	-0.02261	3	7	-0.02261	3	8	-0.01131
4	3	-0.01131	4	4	-0.02261	4	5	-0.02261	4	6	-0.02261	4	7	-0.02261
4	8	-0.02261	4	9	-0.01131	5	3	-0.01131	5	4	-0.02261	5	5	-0.02261
5	6	-0.02261	5	7	-0.02261	5	8	-0.02261	5	9	-0.01131	6	3	-0.01131
6	4	-0.02261	6	5	-0.02261	6	6	-0.02261	6	7	-0.02261	6	8	-0.02261
6	9	-0.01131	7	3	-0.01131	7	4	-0.02261	7	5	-0.02261	7	6	-0.02261
7	7	-0.02261	7	8	-0.02261	7	9	-0.01131	8	3	-0.01131	8	4	-0.02261
8	5	-0.02261	8	6	-0.02261	8	7	-0.02261	8	8	-0.02261	8	9	-0.01131
9	4	-0.01131	9	5	-0.02261	9	6	-0.02261	9	7	-0.02261	9	8	-0.01131

Figure 4
INPUT LISTING FOR A SAMPLE CEXE PROBLEM
Sheet 5 of 5

COLUMN NUMBERS

0000000001111111112222222222333333333344444444445555555555666666666677777777778
1234567890123456789012345678901234567890123456789012345678901234567890

45				
3 4 -2.384	3 5 -4.769	3 6 -4.769	3 7 -4.769	3 8 -2.384
4 3 -2.384	4 4 -4.769	4 5 -4.769	4 6 -4.769	4 7 -4.769
4 8 -4.769	4 9 -2.384	5 3 -2.384	5 4 -4.769	5 5 -4.769
5 6 -4.769	5 7 -4.769	5 8 -4.769	5 9 -2.384	6 3 -2.384
6 4 -4.769	6 5 -4.769	6 6 -4.769	6 7 -4.769	6 8 -4.769
6 9 -2.384	7 3 -2.384	7 4 -4.769	7 5 -4.769	7 6 -4.769
7 7 -4.769	7 8 -4.769	7 9 -2.384	8 3 -2.384	8 4 -4.769
8 5 -4.769	8 6 -4.769	8 7 -4.769	8 8 -4.769	8 9 -2.384
9 4 -2.384	9 5 -4.769	9 6 -4.769	9 7 -4.769	9 8 -2.384

0
0
0
0

DUMMY TIME STEP

0	0	0	0	0	0	0	1	0	1	0	
4.0		1621.6		0.0		0		0.0		0.0	0.0

Figure 5
 OUTPUT FROM A SAMPLE INCEXE PROBLEM
 Sheet 1 of 6

INCEXE TAPE LABELED PC3D

NO.OF NODES I 11
 NO.OF NODES J 11
 NO.OF NODES K 10
 NO.OF MODERATOR ENTRIES 0
 NO.OF DOPPLER ENTRIES 0

IN THE FOLLOWING TABLES THE ABSCISSA CORRESPONDS TO NODES 1 THROUGH J MAX AND
 THE ORDINATE CORRESPONDS TO NODES 1 THROUGH I MAX, AXIAL INCREMENTS ARE ON SUCCEEDING PAGES

FUEL TYPES

	1	2	3	4	5	6	7	8	9	10	11
1	0	0	0	1	1	1	1	1	0	0	0
2	0	0	1	1	1	1	1	1	1	0	0
3	0	1	1	1	1	1	1	1	1	1	0
4	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1
9	0	1	1	1	1	1	1	1	1	1	0
10	0	0	1	1	1	1	1	1	1	0	0
11	0	0	0	1	1	1	1	1	0	0	0

Figure 5

OUTPUT FROM A SAMPLE INCXEXE PROBLEM

Sheet 2 of 6

SIGMA F1 K = 1

[illegible]

SIGMA F2 K = 1

[illegible]

Sheet 3 of 6

NU SIGMA F1 K = 1

[illegible]

NU SIGMA F2 K = 1

[illegible]

Sheet 4 of 6

NODE AVERAGED FLUX WEIGHTED THERMAL ABSORPTION CROSS SECTION OF XENON K = 1

[illegible]

SIGMA A2 K = 1

[illegible]

Sheet 5 of 6

SQUARE OF MIGRATION LENGTH FOR THERMAL NEUTRONS $K = 1$

[illegible]

M**2 (REFERENCE) K = 1

[illegible]

Sheet 6 of 6

K INF.(REFERENCE) K = 1

[illegible]

PNF2 K = 1

[illegible]

Figure 6
 OUTPUT FROM A SAMPLE CEXE PROBLEM
 Sheet 1 of 4

INPUT DATA

CEXE - 3 DIMENSIONAL PICKERING REACTOR - ESTABLISHING EQUILIBRIUM CORE CONDITIONS

INCEXE TAPE LABELED PC3D TO BE USED
 INTERPOLATION OF FEEDBACK TABLES
 MAX ITERATIONS PER SOURCE CALC = 60

THIS IS A STARTUP PROBLEM
 MU INPUT FOR TRANSPORT KERNELS

DELTA X	DELTA Z	DEL S	DEL EIG	DEL EQUIL	DEL EIG SRCH
5.71500E 01	5.94000E 01	1.00000E-05	1.00000E-05	1.00000E-05	1.00000E-05
GH	GV	A	B	EIG REF	MU HORIZ
1.00000E 00	1.00000E 00	0.0	0.0	0.0	1.00000E 00
					MU VERT
					1.00000E 00
EIGEN	KAPPA	IOD DECAY	IOD GAMMA	XEN DECAY	XEN GAMMA
1.00000E 00	3.25000E-17	2.93900E-05	6.18000E-02	2.10000E-05	4.80000E-03

4.0 - ALBEDO HORIZONTAL

	1	2	3	4	5	6	7	8
1	4.000000E 00	4.000000E 00	4.000000E 00	2.810000E 00	3.549999E 00	3.549999E 00	3.549999E 00	2.810000E 00
2	4.000000E 00	4.000000E 00	2.720000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
3	4.000000E 00	2.720000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
4	2.780001E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
5	3.549999E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
6	3.540000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
7	3.549999E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
8	2.820001E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
9	4.000000E 00	2.750000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
10	4.000000E 00	4.000000E 00	2.750000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00	4.000000E 00
11	4.000000E 00	4.000000E 00	4.000000E 00	2.850000E 00	3.549999E 00	3.540000E 00	3.549999E 00	2.850000E 00

2.0 - ALBEDO VERTICAL

1	2.799999E 00
2	2.000000E 00
3	2.000000E 00
4	2.000000E 00
5	2.000000E 00
6	2.000000E 00
7	2.000000E 00
8	2.000000E 00
9	2.000000E 00
10	2.799999E 00

Sheet 2 of 4

ESTABLISHING EQUILIBRIUM

M**2 CONTROL INPUT.

DEL TIME	CORE POWER	THERMAL POISON	FLOW RATE	TEMP BOTTOM	TEMP TOP
4.000000E 01	1.621600E 03	0.0	0.0	0.0	0.0

$$K = 3$$
[illegible]

K = 3

[illegible]

Figure 6
OUTPUT FROM A SAMPLE CEXE PROBLEM
Sheet 3 of 4

CEXE - 3 DIMENSIONAL PICKERING REACTOR - ESTABLISHING EQUILIBRIUM CONDITIONS

ESTABLISHING EQUILIBRIUM

TIME =.20000E 03

TOTAL CORE POWER = 0.1621600E 04

EIGENVALUE = 0.1000767E 01

XENON CONCENTRATIONS			K= 1	TIME=.20000E 03				
1	2	3	4	5	6	7	8	
1	0.0	0.0	0.0	2.921521E 13	2.979733E 13	2.994948E 13	2.979733E 13	2.921521E 13
2	0.0	0.0	3.016619E 13	3.096457E 13	3.128568E 13	3.137354E 13	3.128568E 13	3.096457E 13
3	0.0	3.026476E 13	3.127583E 13	3.171157E 13	3.189080E 13	3.194053E 13	3.189080E 13	3.171157E 13
4	2.955142E 13	3.112663E 13	3.178608E 13	3.206363E 13	3.217627E 13	3.220685E 13	3.217627E 13	3.206363E 13
5	3.009536E 13	3.147892E 13	3.201663E 13	3.222761E 13	3.230881E 13	3.233003E 13	3.230881E 13	3.222761E 13
6	3.023779E 13	3.156741E 13	3.207529E 13	3.226884E 13	3.234136E 13	3.235991E 13	3.234136E 13	3.226884E 13
7	3.003004E 13	3.143807E 13	3.198618E 13	3.220214E 13	3.228589E 13	3.230795E 13	3.228589E 13	3.220214E 13
8	2.937481E 13	3.103238E 13	3.171994E 13	3.201011E 13	3.212911E 13	3.216176E 13	3.212911E 13	3.201011E 13
9	0.0	3.008603E 13	3.116029E 13	3.162326E 13	3.181608E 13	3.187022E 13	3.181608E 13	3.162326E 13
10	0.0	0.0	2.995901E 13	3.082206E 13	3.117495E 13	3.127327E 13	3.117495E 13	3.082206E 13
11	0.0	0.0	0.0	2.894867E 13	2.962705E 13	2.981197E 13	2.962705E 13	2.894867E 13

IODINE CONCENTRATIONS			K= 2	TIME=.20000E 03				
1	2	3	4	5	6	7	8	
1 0.0	0.0	0.0	2.888266E 14	3.245838E 14	3.350899E 14	3.245838E 14	2.888266E 14	
2 0.0	0.0	3.512843E 14	4.232371E 14	4.586158E 14	4.691374E 14	4.586158E 14	4.232371E 14	
3 0.0	3.599271E 14	4.586402E 14	5.122267E 14	5.351929E 14	5.419446E 14	5.351929E 14	5.122267E 14	
4 3.103938E 14	4.437056E 14	5.263383E 14	5.641678E 14	5.796276E 14	5.838565E 14	5.796276E 14	5.641678E 14	
5 3.483798E 14	4.882210E 14	5.637515E 14	5.926558E 14	6.031981E 14	6.058349E 14	6.031981E 14	5.926558E 14	
6 3.596114E 14	5.006203E 14	5.740457E 14	6.002265E 14	6.091536E 14	6.112750E 14	6.091536E 14	6.002265E 14	
7 3.433260E 14	4.824569E 14	5.580655E 14	5.873462E 14	5.982229E 14	6.009889E 14	5.982229E 14	5.873462E 14	
8 2.993772E 14	4.326245E 14	5.155596E 14	5.540897E 14	5.702127E 14	5.747088E 14	5.702127E 14	5.540897E 14	
9 0.0	3.455682E 14	4.442000E 14	4.985311E 14	5.225053E 14	5.296972E 14	5.225053E 14	4.985311E 14	
10 0.0	0.0	3.353417E 14	4.078700E 14	4.447213E 14	4.559951E 14	4.447213E 14	4.078700E 14	
11 0.0	0.0	0.0	2.741586E 14	3.130062E 14	3.251518E 14	3.130062E 14	2.741586E 14	

Figure 6
OUTPUT FROM A SAMPLE CEXE PROBLEM
Sheet 4 of 4

THERMAL FLUXES								
K= 1 TIME=.20000E 03								
1	2	3	4	5	6	7	8	
1	0.0	0.0	0.0	2.833897E 13	3.190644E 13	3.296428E 13	3.190644E 13	2.833897E 13
2	0.0	0.0	3.457746E 13	4.188053E 13	4.561650E 13	4.674287E 13	4.561650E 13	4.188053E 13
3	0.0	3.535609E 13	4.549345E 13	5.157546E 13	5.451567E 13	5.538505E 13	5.451567E 13	5.157546E 13
4	3.031269E 13	4.369594E 13	5.276215E 13	5.764957E 13	5.987295E 13	6.050401E 13	5.987295E 13	5.764957E 13
5	3.403592E 13	4.815975E 13	5.676573E 13	6.093852E 13	6.269694E 13	6.317162E 13	6.269694E 13	6.093852E 13
6	3.514026E 13	4.941004E 13	5.787227E 13	6.182038E 13	6.342739E 13	6.385155E 13	6.342739E 13	6.182038E 13
7	3.354891E 13	4.760146E 13	5.620596E 13	6.040549E 13	6.219160E 13	6.267748E 13	6.219160E 13	6.040549E 13
8	2.924834E 13	4.262337E 13	5.170595E 13	5.664486E 13	5.892338E 13	5.957776E 13	5.892338E 13	5.664486E 13
9	0.0	3.396541E 13	4.409022E 13	5.022916E 13	5.325400E 13	5.416251E 13	5.325400E 13	5.022916E 13
10	0.0	0.0	3.303246E 13	4.038994E 13	4.426398E 13	4.546134E 13	4.426398E 13	4.038994E 13
11	0.0	0.0	0.0	2.692020E 13	3.078852E 13	3.200559E 13	3.078852E 13	2.692020E 13

POWERS								
K= 1 TIME=.20000E 03								
1	2	3	4	5	6	7	8	
1	0.0	0.0	0.0	3.324238E-01	3.742713E-01	3.866799E-01	3.742713E-01	3.324238E-01
2	0.0	0.0	4.056029E-01	4.912701E-01	5.350940E-01	5.483065E-01	5.350940E-01	4.912701E-01
3	0.0	4.147364E-01	5.336506E-01	6.049942E-01	6.394837E-01	6.496818E-01	6.394837E-01	6.049942E-01
4	3.555762E-01	5.125654E-01	6.189144E-01	6.762452E-01	7.023262E-01	7.097286E-01	7.023262E-01	6.762452E-01
5	3.992506E-01	5.649271E-01	6.658775E-01	7.148255E-01	7.354522E-01	7.410204E-01	7.354522E-01	7.148255E-01
6	4.122050E-01	5.795933E-01	6.788574E-01	7.251700E-01	7.440207E-01	7.489962E-01	7.440207E-01	7.251700E-01
7	3.935379E-01	5.583782E-01	6.593114E-01	7.085730E-01	7.295244E-01	7.352241E-01	7.295244E-01	7.085730E-01
8	3.430910E-01	4.999838E-01	6.065251E-01	6.644598E-01	6.911873E-01	6.988634E-01	6.911873E-01	6.644598E-01
9	0.0	3.984234E-01	5.171903E-01	5.892017E-01	6.246840E-01	6.353409E-01	6.246840E-01	5.892017E-01
10	0.0	0.0	3.874797E-01	4.737850E-01	5.192286E-01	5.332739E-01	5.192286E-01	4.737850E-01
11	0.0	0.0	0.0	3.157813E-01	3.611578E-01	3.754342E-01	3.611578E-01	3.157813E-01

6

APPENDIX A DERIVATION OF AN ALTERNATE TRANSPORT KERNEL

Since the transport kernels which were developed for the FLARE Code were based on one-dimensional considerations (Reference 2), an alternate transport kernel has been formulated by accounting for the three-dimensional effects involved in the calculation of neutron migration between adjacent nodes and this kernel has been programmed into CEXE. The derivation is based on a one-group theory in which diffusion theory is assumed to be valid.

To begin, the absorption probability at the point $\vec{r}$ due to a unit source at $\vec{r}_0$ is denoted as $W(\vec{r}, \vec{r}_0)$. Thus $W(\vec{r}, \vec{r}_0) = \phi(\vec{r}, \vec{r}_0) \Sigma_a(\vec{r})$ where $\phi(\vec{r}, \vec{r}_0)$ is the flux at $\vec{r}$ due to a unit source at $\vec{r}_0$ and Σ_a is the absorption cross section. Assuming that the medium is uniform and infinite and the source is isotropic, then diffusion theory gives

$$\phi(\vec{r}, \vec{r}_0) = \frac{\exp\{-\kappa |\vec{r} - \vec{r}_0|\}}{4\pi D |\vec{r} - \vec{r}_0|}$$

where

$$\kappa^2 = \frac{\Sigma_a}{D} = \frac{1}{M^2}$$

and D is the diffusion coefficient. Substituting this into the previous expression for $W(\vec{r}, \vec{r}_0)$ yields

$$W(\vec{r}, \vec{r}_0) = \frac{\kappa^2}{4\pi} \frac{\exp\{-\kappa |\vec{r} - \vec{r}_0|\}}{|\vec{r} - \vec{r}_0|}$$

Define $W_{\ell\ell}$ as the average total absorption probability in a volume V_ℓ due to a uniform distribution of sources in V_ℓ . Then

$$W_{\ell\ell} = \frac{\int_{V_\ell} d\vec{r}_0 \int_{V_\ell} d\vec{r} W(\vec{r}, \vec{r}_0)}{V_\ell}$$

Dropping the ℓ subscripts and substituting in the expression for $W(\vec{r}, \vec{r}_0)$ gives

$$W = \frac{\kappa^2}{4\pi V} \int_V \int_V d\vec{r}_0 d\vec{r} \frac{e^{-\kappa|\vec{r} - \vec{r}_0|}}{|\vec{r} - \vec{r}_0|}$$

Define the dimensionless variables $\vec{\rho}$ and $\vec{\rho}_0$ by $\vec{\rho}_0 = \kappa\vec{r}_0$, $\vec{\rho} = \kappa\vec{r}$ and the dimensionless volume by $V^1 = \kappa^3 V$. Then

$$W = \frac{1}{4\pi V^1} \int_{V^1} \int_{V^1} d\vec{\rho}_0 d\vec{\rho} \frac{\exp\{-|\vec{\rho} - \vec{\rho}_0|\}}{|\vec{\rho} - \vec{\rho}_0|}$$

In applying this formalism to CEXE, V^1 is a rectangular parallelepiped and the integrations over $\vec{\rho}_0$ and $\vec{\rho}$ can only be carried out numerically. Thus the volume V^1 is replaced by a sphere of equal volume and the integration variables are transformed into spherical polar coordinates. Taking μ as the cosine of the angle between $\vec{\rho}_0$ and $\vec{\rho}$ we then have

$$W = \frac{2\pi}{V^1} \int_0^{R^1} \rho_0^2 d\rho_0 \int_0^{R^1} \rho^2 d\rho \int_{-1}^1 d\mu \frac{\exp\{-\sqrt{\rho^2 + \rho_0^2 - 2\rho\rho_0\mu}\}}{\sqrt{\rho^2 + \rho_0^2 - 2\rho\rho_0\mu}}$$

where $R^1 = (\frac{3V^1}{4\pi})^{1/3}$ is the radius of the equivalent dimensionless sphere.

But

$$\frac{\exp[-\sqrt{\rho^2 + \rho_0^2 - 2\rho\rho_0\mu}]}{\sqrt{\rho^2 + \rho_0^2 - 2\rho\rho_0\mu}} = \left\{ \frac{1}{\rho_0\rho} \frac{\partial}{\partial \mu} \left(\exp[-\sqrt{\rho^2 + \rho_0^2 - 2\rho\rho_0\mu}] \right) \right\}$$

Thus the integration over μ is trivial and yields

$$W = \frac{2\pi}{Vl} \int_0^{R^1} \rho_0 d\rho_0 \int_0^{R^1} \rho d\rho \{ e^{-|\rho - \rho_0|} - e^{-(\rho + \rho_0)} \}$$

The integration over ρ is performed first. Making use of the fact that

$$e^{-|\rho - \rho_0|} = \begin{cases} e^{-(\rho - \rho_0)} & \rho > \rho_0 \\ e^{-(\rho_0 - \rho)} & \rho < \rho_0 \end{cases}$$

we get

$$I_1 = \int_0^{R^1} \rho d\rho e^{-|\rho - \rho_0|} = 2\rho_0 + e^{-\rho_0} - e^{-(R^1 - \rho_0)} (1 + R^1)$$

and

$$I_2 = \int_0^{R^1} \rho d\rho e^{-(\rho + \rho_0)} = e^{-\rho_0} [1 - e^{-R^1} (1 + R^1)]$$

Thus

$$W = \frac{2\pi}{Vl} \int_0^{R^1} \rho_0 d\rho_0 [I_1 - I_2]$$

The integration over ρ_0 is straightforward and yields

$$W = 1 - P$$

where

$$P = \frac{2\pi}{Vl} (1 + R^1) e^{-R^1} \{R^1[e^{R^1} + e^{-R^1}] - [e^{R^1} - e^{-R^1}]\}$$

$$= \frac{4\pi}{Vl} (1 + R^1) e^{-R^1} \sinh R^1 [R^1 \coth R^1 - 1]$$

Since W is the self-absorption probability, P is the escape probability. Thus in adapting these results to CEXE

$$P_{\ell} = 4W_{\ell m}^H + 2W_{\ell m}^V$$

Assuming that $W_{\ell m}^{(i)}$ is proportional to the transverse area yields

$$\frac{W_{\ell m}^H}{W_{\ell m}^V} = \frac{\Delta Z}{\Delta X}$$

Solving for the W^H and W^V gives

$$W_{\ell m}^H = \frac{P_{\ell}}{4 + 2 \frac{\Delta X}{\Delta Z}}$$

and

$$W_{\ell m}^V = W_{\ell m}^H \frac{\Delta X}{\Delta Z}$$

Since these results were developed on the basis of a spherical node and no account has been made for leakage beyond the six nearest neighbors, it is expected that the use of this transport kernel will require modifying constants. Thus the form of the transport kernel as programmed into CEXE is:

$$W_{\ell m}^H = \frac{g^H P_{\ell}}{4 + 2(\frac{\Delta X}{\Delta Z})} \quad \text{and} \quad W_{\ell m}^V = g^V W_{\ell m}^H \left(\frac{\Delta X}{\Delta Z}\right)$$

APPENDIX B

IBM-360 HARDWARE AND SOFTWARE REQUIREMENTS FOR CEXE

CEXE is written in FORTRAN IV. All but two subroutines, MAIN and SORCIO, were compiled using the IBM FORTRAN Level H compiler, OPT=2. MAIN and SORCIO, which are the only routines utilizing Direct Access I/O FORTRAN statements, were compiled using the IBM FORTRAN Level G compiler. The majority of fixed point variables are half-word (2-bytes). Many of the fixed point input numbers which come from the INCEXE tape are half-word. There are no double precision floating point numbers. The CEXE overlay structure is as follows:

MAIN, FNTERP, POWER

Overlay One

NITIAL

Overlay One

INPUT1

Overlay One

INPUT2

Overlay One

OUTIN

Overlay One

TEMPER, SOURCE, NWSRCH, KANDM2, SORCIO, CONVRG

Overlay One

NWTIME, OUTPUT, STORE

Overlay One

XENON

The only subroutine called which is not contained in the supplied package is FROID2. FROID2 merely replaces an operator reply which overrides the label information on standard label tapes.

Core storage requirements for the above overlay structure and associated IHC routines should not exceed $(33,000)_{16}$ bytes. Due to the large amount of I/O, physical separation of disk drives was found desirable although not mandatory.

The installation on which the code was developed utilizes six IBM-2314 disk drives, three IBM-2400 tape drives and one IBM-2540 card reader/punch during execution of a maximum size problem.

APPENDIX C

IBM-360 JOB CONTROL LANGUAGE SETUPS FOR INCEXE AND CEXE

As mentioned in section V, the Job Control Language (JCL) is a vital portion of the input. The JCL cards are strongly installation-dependent and the setups in this appendix only typify the hardware requirements for the installation on which INCEXE and CEXE were developed.

The JCL Setup for the INCEXE code is as follows:†

```
/*SETUP DDNAME=FT10F001,DEVICE=2400-9,ID=(SCRATCH,,SL)
//JOBLIB DD DSN=NUC.PROD.LIB1,DISP=(OLD,PASS)
// EXEC PGM=NUC00135
//FT10F001 DD DSN=XXX.XXXXXX,UNIT=2400,DISP=(NEW,CATLG),
//          LABEL=(,SL,EXPDT=XXYY),
//          DCB=(,RECFM=VB,LRECL=1448,BLKSIZE=1452)
//SYSABEND DD SYSOUT=A
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
```

(INCEXE Data Deck)

/*

XXX.XXXXXX is the INCEXE library tape catalog name. XXYY is the expiration date in year and day of year.

The JCL setup for CEXE depends on the type of problem which is being executed. In the following example setups, the JOB card is deleted due to installation dependence, i.e. JOB is assumed. ASP (Asynchronous Support Processor) is also assumed.

† All statements begin in column 1; indented statements begin in column 16; and all X's at the end of the cards are in column 72.

For new problems in which a restart tape is to be generated at end time, the following setup might be used.†

Card No.

```

1  /*SETUP DDNAME=FT10F001,DEVICE=2400-9,ID=(XXXXXX,,,SL)
2  /*SETUP DDNAME=FT21F001,DEVICE=2400-9,ID=(SCRTCH,,,SL)
3  /*SETUP DDNAME=FT24F001,DEVICE=2400-9,ID=(SCRTCH,,,SL)
4  //JOB LIB DD DSNAME=NUC.PROD.LIB1,DISP=(OLD,PASS)
5  // EXEC PGM=NUC00136
6  //FT22F001 DD UNIT=2314,VOLUME=REF=PACK4,SPACE=(1444,(51)),      X
7  //          DCB=(BUFNO=2,DSORG=DA)
8  //FT18F001 DD UNIT=SYSDA,VOLUME=REF=PACK6,SPACE=(CYL,1),          X
9  //          DCB=(BUFNO=1,LRECL=1448,BLKSIZE=1452)
10 //FT15F001 DD UNIT=SYSDA,VOLUME=REF=PACK4,SPACE=(CYL,1),          X
11 //          DCB=(BUFNO=1,LRECL=1448,BLKSIZE=1452)
12 //FT23F001 DD UNIT=SYSDA,VOLUME=REF=PACK6,SPACE=(CYL,1),          X
13 //          DCB=(BUFNO=1,LRECL=1448,BLKSIZE=1452)
14 //FT16F001 DD UNIT=SYSDA,VOLUME=REF=PACK1,SPACE=(CYL,1),          X
15 //          DCB=(BUFNO=1,LRECL=1448,BLKSIZE=1452)
16 //FT10F001 DD DSNAME=XXX.XXXXXX,UNIT=2400,DISP=(OLD,KEEP),DCB=(BUFNO=1)
17 //FT21F001 DD DSNAME=XXX.XXXXXX,UNIT=2400,DISP=(NEW,CATLG),      X
18 //          LABEL=(,SL,EXPDT=XXYY),                               X
19 //          DCB=(RECFM=VB,LRECL=1448,BLKSIZE=1452,BUFNO=1)
20 //FT24F001 DD DSNAME=XXX.XXXXXX,UNIT=2400,DISP=(NEW,CATLG),      X
21 //          DCB=(RECFM=VB,LRECL=1448,BLKSIZE=1452,BUFNO=1),      X
22 //          LABEL=(,SL,EXPDT=XXYY).
23 //SYSABEND DD SYSOUT=A
24 //FT06F001 DD SYSOUT=A
25 //FT05F001 DD *
```

(CEXE Data Deck)

/*

XXXXXX on card one is serial number of INCXE library tape. XXX.XXXXXX on card 16 is catalog name of INCXE library tape. XXX.XXXXXX on card 17 is a unique catalog name for the equilibrium time restart tape to be generated. XXYY on card 18 is the expiration date of the equilibrium time restart tape: year, day of year. XXX.XXXXXX on card 20 is a unique catalog name for the end time restart tape to be generated. XXYY on card 22 is the expiration date of the end time restart tape: year, day of year.

† All statements begin in column 1; indented statements begin in column 16; and all X's at the end of the cards are in column 72.

For a new problem in which a restart tape is not to be generated at the end time, cards 3, 20, 21 and 22 should be omitted from the above set.

For restart problems, card two should be replaced in the above set by:

```
/*SETUP DDNAME=FT20F001,DEVICE=2400-9,ID=(XXXXXX,,SL)
```

Cards 17, 18 and 19 should be deleted and the following inserted in their place:

```
//FT20F001 DD DSNAME=XXX.XXXXXX,UNIT=2400,DISP=(OLD,KEEP),DCB=(BUFNO=1)
```

On the above XXXXXX is the serial number of the restart tape generated by a previous CEXE run. XXX.XXXXXX is the catalog name of that tape.

For restart problems, the inclusion or omission of cards 3, 20, 21 and 22 is the same as for new problems depending on whether or not a restart tape is to be generated at the end time.

APPENDIX D
LOGICAL I/O UNITS FOR CEXE

A brief description of CEXE logical I/O units is as follows.

- Unit 5: standard input.
- Unit 6: standard output.
- Unit 10: INCEXE tape used as input.
- Unit 15: scratch disk which contains iodine concentrations.
- Unit 16: scratch disk which is used when iodine concentrations are updated.
- Unit 18: scratch disk which contains several arrays copied from the INCEXE tape.
- Unit 20: input restart tape for restart problems only.
- Unit 21: output restart tape generated at equilibrium time for new problems only.
- Unit 22: Direct Access scratch disk which contains arrays copied from the INCEXE tape.
- Unit 24: output restart tape generated at the End Time.

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

- A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or
- B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.