NOTICE

PORTIONS OF THIS REPORT ARE ILLEGIBLE. It has been reproduced from the best available copy to permit the broadest possible availability.

555
555

THE JOSHUA SYSTEM

VOL. 4.1 LATTICE PHYSICS

Approved by

H. C. Honeck, Research Manager
Computer Applications Division

NOTICE

This document contains descriptions of the JOSHUA Operating System and Applications Modules. The document is prepared as a set of loose-leaf notebooks which will be periodically updated. [REDACTED]

June 1970

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

E. I. DU PONT DE NEMOURS & COMPANY
SAVANNAH RIVER LABORATORY
AIKEN, S. C. 29801

CONTRACT AT(07-2)-1 WITH THE
UNITED STATES ATOMIC ENERGY COMMISSION

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, 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.

EB

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.

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.

JOSHUA Vol. 4.1

LATTICE PHYSICS

Part F	Leakage Modules
Part G	Edit Modules
Part H	Control Modules
Part I	Lattice Depletion Modules
Part J	Lattice Monte Carlo Modules

JOSHUA DPSTM-500	Title Part F - Leakage Modules	Vol.	Page
		IV	F.O.1
		Rev.	Date
			6/70

Contents

F.1 Supercell Homogenization - Module PASTE

F.2 B_L Leakage Calculation - Module BLEAK

JOSHUA
DPSTM-500

Title Part F - Leakage Modules

Vol. IV	Page F.O.2
Rev.	Date 6/70

(Blank)

Work Done By

C. E. Henderson
H. C. Honeck

Report Written By

C. E. Henderson
H. C. Honeck

Contents

	<u>Page</u>
1. <u>Theory</u>	
1.1 Introduction	F.1.3
1.2 Cross Sections	F.1.3
1.3 Resonance Rates	F.1.4
2. <u>Program Logic</u>	
2.1 Interfaces	F.1.5
2.2 Calculation Sequence	F.1.6
2.3 Flux Weighting (FLXWGT)	F.1.6
2.4 Slowing-Down Density Weighting (QWGT)	F.1.6
2.5 Cross Section Homogenization (HØMØGX)	F.1.8
2.6 Scattering Matrix Homogenization (HØMØGS)	F.1.8
3. <u>Coding Details</u>	
3.1 CØMMØN DATA	F.1.9
3.2 CØMMØN SPECS	F.1.9
3.3 CØMMØN GROUPS	F.1.10
3.4 CØMMØN/SCRACH/	F.1.10
3.5 Main Program PASTE	F.1.10
3.6 Subroutine FLXWGT	F.1.10
3.7 Subroutine QWGT	F.1.10
3.8 Subroutine HØMØGX	F.1.10
3.9 Subroutine HØMØGS	F.1.10
4. <u>User Information</u>	
4.1 Input	F.1.11
4.2 Output	F.1.11
4.3 Core Size and Timing	F.1.11
4.4 Error Stops	F.1.11

JOSHUA DPSTM-500	Title Module PASTE	Vol. IV	Page F.1.2
		Rev.	Date 5/70

(Blank)

1. Theory

1.1 Introduction

The PASTE module uses the fine-structure flux and slowing-down density for a super-cell to homogenize the super-cell into a uniform paste. The cross sections and resonance rates for the paste are calculated so that all reaction rates for the paste are the same as the average rates for the supercell.

1.2 Cross Sections

Let $\phi(\underline{r}, E)$ be the fine-structure flux in the super-cell, and $\Sigma^X(\underline{r}, E)$ be the macroscopic cross section of type x . The cell average flux is given by

$$\phi(E) = \frac{1}{V} \int d\underline{r} \phi(\underline{r}, E) \quad (1)$$

where V is the super-cell volume. The total number of reactions of type x in the super-cell is

$$R^X(E) = \int d\underline{r} \Sigma^X(\underline{r}, E) \phi(\underline{r}, E) \quad (2)$$

We want to define the average cross section $\Sigma^X(E)$ such that the total number of reactions are preserved.

$$R^X(E) = \Sigma^X(E) \phi(E) V \quad (3)$$

Thus,

$$\Sigma^X(E) = \frac{\int d\underline{r} \Sigma^X(\underline{r}, E) \phi(\underline{r}, E)}{\int d\underline{r} \phi(\underline{r}, E)} \quad (4)$$

A similar formula will preserve scattering rates.

$$\Sigma_0(E' \rightarrow E) = \frac{\int d\underline{r} \Sigma_0(\underline{r}, E' \rightarrow E)}{\int d\underline{r} \phi(\underline{r}, E')} \quad (5)$$

where the subscript o denotes the P_o angular component. Other angular components should be weighted with their corresponding angular flux components. However, since only the scalar fine-structure is available, equation (5) is used for all angular components.

The cross section defined by equation (4) is the total macroscopic cross section for reaction type x, where x may denote capture, absorption, n^2n , etc. If x denotes fission, cross sections are computed for each isotope (numbered by k).

$$\Sigma_k^f(E) = \int d\underline{r} \Sigma_k^f(\underline{r}, E) \phi(\underline{r}, E) / \int d\underline{r} \phi(\underline{r}, E) \quad (6)$$

In terms of the multigroup cross sections and fluxes in group j, region n, the above equations become

$$\Sigma^x(j) = \sum_n V(n) \Sigma^x(n, j) \phi(n, j) / \sum_n V(n) \phi(n, j) \quad (7)$$

$$\Sigma_\ell(i \rightarrow j) = \sum_n V(n) \Sigma_\ell(n, i \rightarrow j) \phi(n, i) / \sum_n V(n) \phi(n, i) \quad (8)$$

$$\Sigma_k^f(j) = \sum_n V(n) \Sigma_k^f(n, j) \phi(n, j) / \sum_n V(n) \phi(n, j) \quad (9)$$

1.3 Resonance Rates

Let $Q(c, j)$ be the slowing-down density in cell c, group j. Let $R_a(k, n, j)$ and $R_f(k, n, j)$ be the resonance rates for absorption and fission for isotope k in region n and group j. The fine-structure flux ratio is defined by

$$\psi(n, j) = \phi(n, j) V(c) / \sum_{n \in c} V(n) \phi(n, j) \quad (10)$$

that is, the ratio of the flux in region n to the average flux in the cell in which n occurs. The total resonance absorption (or fission) in isotope k , group j , is then

$$\rho_a(k,j) = \sum_n R_a(k,n,j) \psi(n,j) Q(c,j) \quad (11)$$

The total slowing-down density is

$$q(j) = \sum_c Q(c,j) \quad (12)$$

so that the average resonance absorption (or fission) rate, $\bar{R}_a(k,j)$, is given by

$$\bar{R}_a(k,j) = \rho_a(k,j) q(j) \quad (13)$$

Fission rates are kept by isotope. Absorption rates are summed over all isotopes in the super-cell.

$$\bar{R}_a(j) = \sum_k \bar{R}_a(k,j) \quad (14)$$

2. Program Logic

2.1 Interfaces

PASTE reads the following records:

('TEMP'. 'LIDS'. DLAT. VLAT → V)

'INPUT'. 'PASTE'

V. 'NAME'. NCASE.0 .0

V. 'COMP'. NCASE.0 .0

V. 'GEOM'. NCASE.0 .0

V. 'GRPS'. NCASE.0 .0

V. 'RESID'. NCASE.1 .0

V. 'XSEC'. NCASE. NEB.0

V. 'SMAT'. NCASE. NEB.0

V. 'CHI'. NCASE.0 .0

PASTE write the following records:

V.'HOMX'.NCASE.0 .0

V.'PMAT'.NCASE.NEB.0

2.2 Calculation Sequence

The PASTE module consists of a main program (PASTE) which calls subroutines to prepare the weighting flux (FLXWGT), prepare the slowing-down density weight (QWGT), compute average cross sections (HOMOGX), and compute average scattering matrices (HOMOGS). The energy band structure established by the CRISP module is retained for the average scattering matrices (PMAT blocks), but cross sections for all groups are consolidated in a single average cross section block (HOMX). A flow diagram for the PASTE module is shown in figure 1.

2.3 Flux Weighting (FLXWGT)

Subroutine FLXWGT reads the fine-structure flux $\phi(n,j)$ by region n and group j . If the flux cannot be found, a constant flux is set up. A flux weight function $W_f(k,j)$ by isotope k and group j is computed from

$$W_f(k,j) = \frac{\sum_n v(n) C(k,n) \phi(n,j) T(n)}{\sum_n v(n)} \cdot \phi(n,j) T(n) \quad (15)$$

where $T(n)$ is the number of equivalent lattice positions in the supercell for region n and $C(k,n)$ is the concentration of isotope k in region n . Equations like (7) can then be written as

$$\Sigma^x(j) = \sum_k W_f(k,j) \sigma^x(k,j) \quad (16)$$

where $\sigma^x(k,j)$ is the microscopic cross section of type x for isotope k in group j .

2.4 Slowing-Down Density Weighting (QWGT)

If resonance rates are given, subroutine QWGT reads the slowing-down density $Q(c,j)$ by cell c and group j .

figure 1. PASTE Flowdiagram

If $Q(c,j)$ cannot be found, a constant is set up. A Q weight function $W_q(c,j)$ is computed from

$$W_q(c,j) = Q(c,j)T(c) / \sum_c Q(c,j)T(c) \quad (17)$$

where $T(c)$ is the number of equivalent cells in the super-cell for cell c . Equation (13) can then be written as

$$\bar{R}_a(k,j) = \sum_n R_a^1(k,n,j)W_q(c,j) \quad (18)$$

$$R_a^1(k,n,j) = R_a(k,n,j)\psi(n,j) \quad (19)$$

2.5 Cross Section Homogenization (HOMOGX)

Cross section homogenization is performed by the subroutine HOMOGX in the following sequence:

1. Allocate space for the output block HOMX.
2. Process the resonance rates in the RESD block using equation (18) and store results in the HOMX block.
3. Read a XSEC block for one energy band. Process the data using equation (16) and store results in the HOMX block.
4. Repeat step 3 until all energy bands are done.
5. If fissile isotopes are present, read the CHI block and move the data into the HOMX block.

2.6 Scattering Matrix Homogenization (HOMOGS)

Scattering matrix homogenization for one energy band is performed by the subroutine HOMOGS in the following sequence:

1. Allocate space for an output block PMAT.
2. Read the SMAT block for this energy band.
3. Process the data using the equation:

$$\tau(i \rightarrow j) = \sum_k W_f(k, i) \sigma(i \rightarrow j) \quad (20)$$

4. If the SMAT block contains only down-scattering data for the thermal region, compute the up-scattering part from

$$\sigma(i \rightarrow j) = M(j) \sigma(j \rightarrow i) / M(i) \quad (21)$$

$$M(j) = \bar{E}(j) \Delta E(j) \exp[-\bar{E}(j)/T] \quad (22)$$

where T is the isotope temperature in ev.

3. Coding Details

The PASTE module is divided into the following parts:

1. COMMON/DATA/
2. COMMON/SPECS/
3. COMMON/GRUUPS/
4. COMMON/SCRACH/
5. Main program PASTE
6. Subroutine FLXWGT
7. Subroutine QWGT
8. Subroutine HØMØGX
9. Subroutine HØMØGS
10. Subroutine MANAGE (See Manuscript B.1)

3.1 COMMON/DATA/

The LIDS blocks NAME, CØMP, GEØM, XSEC, SMAT, CHI, RESD, FLUX, QCEL, HØMX, PMAT, and the temporary block WGTF are packed in the array DATA defined by

COMMON/DATA/DATA(20000).

Refer to Manuscript B.1 for a description of this array and its usage.

3.2 COMMON/SPECS/

The record 'INPUT'. 'PASTE' contains the data:

TDSN - Temporary data set name = 'TEMP'
DLAT - Lattice name

VLAT - Lattice version
NCASE - Case number
LPRNT - If LPRNT = 1, the maximum length of the
DATA array used will be printed.

These data are stored in /SPECS/.

3.3 COMMON/GROUPS/

This area contains the group structure data written by the CRISP module. The quantities used in PASTE are:

NEG - Number of energy groups
JFTG - Index to first thermal energy group
EBAR(I) - Characteristic energy (EV)
NEBX - Number of energy bands
IEBH(NEB) - Energy band break points

3.4 COMMON/SCRACH/

This is a SCRACH area used by HOMØGS to store intermediate results.

3.5 Main Program PASTE

The flow diagram for PASTE is given in figure 1.

3.6 Subroutine FLXWGT

This routine calculates a flux weight function WGTF as described in section 2.3 and converts the fine-structure region flux to a flux-volume fraction.

3.7 Subroutine QWGT

This routine calculates a slowing-down density weight as described in section 2.4.

3.8 Subroutine HOMØGX

This routine homogenizes cross sections and resonance rates and stores the results in the HOMX block. See section 2.5 for detailed description.

3.9 Subroutine HOMØGS

Scattering matrix homogenization for energy band NEB is performed and stored in the block PMAT. The argument LX is the maximum Legendre order used. (Section 2.6)

4. User Information

4.1 Input

PASTE cannot be used as a stand-alone module since it relies on the NAME, COMP and GEOM records created by ASP and LAGER, as well as the XSEC, SMAT, and CHI blocks and group structure record created by CRISP. The blocks RESD from RESIN and FLUX and QCEL from FIST are also used; however, defaults are provided in case RESIN and FIST have not been executed. In addition to these records, PASTE requires only the record:

Name: 'INPUT'. 'PASTE'

List: DLAT, VLAT, TDSN, NCASE, LPRNT

DLAT - Lattice name

VLAT - Lattice version

TDSN - Temporary data set name = 'TEMP'

NCASE - Case number

LPRNT - If LPRNT = 1, the maximum length of the DATA array used will be printed.

Normally, this record is created by the RAHAB control module. These items are used to establish the names of the input and output records.

4.2 Output

The HØMX and PMAT blocks will be computed and written on the LIDS.

4.3 Core Size and Timing

The PASTE module requires 195K of core, 1 job data set block, and should be executed with a GØTIME = 1 minute. Individual sub-program sizes and timings are given in Table I. The timings were for a uniform lattice with 4 fissionable isotopes.

4.4 Error Stops

All error stops are handled by the JØSHUA operating system and are summarized in Table II.

TABLE I

SIZE AND TIMING FOR PASTE SUBPRØGRAMS

(Size in bytes, time in CPU seconds)

<u>SUBPRØGRAM</u>	<u>SIZE</u>	<u>TØTAL TIME</u>	<u>AVERAGE TIME</u>	<u>ENTRIES</u>
MAIN(PASTE)	1798	.34944	.34944	1
FLXWGT	1714	.14976	.14976	1
QWGT	1028	.01600	.01600	1
HØMØGX	4892	.08320	.08320	1
HØMØGS	3658	.79872	.26624	3
MANAGE*	11850	2.74560**	.00767	358
DATA	80000			
GRØUPS	1624			
SPECS	32			
DATCAT	2440			
SCRACH	2020			
IHCSEXP	432			
JØS***	<u>83072</u>			
	194560	4.14272		

*Includes all subroutines of data manager

**Includes time for JØSHUA operating system

***JØSHUA Operating System, buffers, and timing routines

TABLE II

PASTE ERROR STOPS

Message/Explanation

PASTE.MAIN, PØINTER MISSING

One or more of the blocks NAME, CØMP, or GEØM is not in LIDS.

PASTE.MAIN, CØNTROL RECØRD MISSING

The record 'INPUT'.'PASTE' is missing.

PASTE.MAIN, GRØUPS RECØRD MISSING

The GRPS record is not in LIDS.

PASTE.FLXWGT, PØINTER MISSING

Either the GEØM or CØMP block is not in LIDS.

PASTE.QWGT, PØINTER MISSING

The GEØM block is not in LIDS.

PASTE.HØMØGX, PØINTER MISSING

The GEØM, QCEL, CØMP, or WGTf is not in the DATA array.

PASTE.HØMØGX, DATA ARRAY ØVERFLØW

The data for the HØMX block cannot be contained in the DATA array.

PASTE.HØMØGX, INCØNSISTENT CHI

Either a fissionable isotope in the HØMX block cannot be found in the CHI block or the CHI block is not in LIDS.

PASTE.HØMØGX, XSEC BLØCK MISSING

The XSEC block is not in LIDS.

PASTE.HØMØGS, TØØ MANY LEGENDRE CØMPØNENTS

The number of Legendre components exceeds the maximum number permissible.

JOSHUA DPSTM-500	Title Module PASTE	Vol.IV	Page F.1.14
		Rev.	Date 5/70

TABLE II (Cont'd.)

PASTE ERROR STOPS

PASTE.HØMØGS, PØINTER MISSING

Either of the blocks CØMP or WGTF is not in the data array.

PASTE.HØMØGS, ZERØ TEMPERATURE FØR DETAILED BALANCE

The isotope is temperature independent and detailed balance cannot be used to calculate upscattering terms.

PASTE.HØMØGS, DATA ARRAY ØVERFLØW

The data for the SMAT block cannot be contained in the DATA array.

PASTE.HØMØGS, SMAT BLØCK MISSING

The SMAT block is not in the LIDS.

Work Done By

C. E. Henderson
H. C. Honeck

Report Written By

H. C. Honeck

	<u>Contents</u>	<u>Page</u>
1.	<u>Theory</u>	
1.1	Introduction	F.2.3
1.2	The B ₁ Equations	F.2.3
1.3	The k _{eff} Calculation	F.2.7
1.4	The Critical Buckling Calculation	F.2.9
1.5	Time Eigenvalue Calculation	F.2.12
1.6	The Coefficients R _{1n}	F.2.13
1.7	The Multigroup Approximation	F.2.17
1.8	Resonance Treatment	F.2.17
2.	<u>Program Logic</u>	
2.1	Interfaces	F.2.19
2.2	Module Structure	F.2.19
2.3	Eigenvalue Search (SEARCH)	F.2.21
2.4	Basic B ₁ Calculation (BABL)	F.2.24
2.5	Fission Source Calculation (BLFIS)	F.2.30
2.6	Slowing-Down Source Calculation (SØRCE)	F.2.31
2.7	One-Group B ₁ Equation (BLEQ)	F.2.31
2.8	Storage Considerations	F.2.33
3.	<u>Coding Details</u>	
3.1	CØMMØN/DATA/	F.2.33
3.2	CØMMØN/SPECS/	F.2.34
3.3	CØMMØN/GRØUPS/	F.2.34
3.4	CØMMØN/TEMP/	F.2.34
3.5	CØMMØN/SCRACH/	F.2.35
4.	<u>User Information</u>	
4.1	Input	F.2.35
4.2	Output	F.2.36
4.3	Core Sized Timing	F.2.36
4.4	Error Stops	F.2.36

JOSHUA DPSTM-500	Title Module BLEAK	Vol. IV	Page F.2.2
		Rev.	Date 4/70

(Blank)

1. Theory

1.1 Introduction

The spatial dependence of the neutron flux in a lattice cell is calculated in RAHAB with the restriction that there is no net leakage from the cell. This restriction is necessary to obtain tractable equations. In practice there is net leakage from the cell and, since this leakage rate is different at different energies, the leakage changes the neutron spectrum. If we assume that these leakage distortions are small and that they do not influence the intra-cell flux shapes, we can estimate them using a B_ℓ calculation. This procedure is standard and is used in the existing HAMMER. The basic idea is to use the zero-leakage intra-cell fluxes as weighting functions to homogenize the cell into a paste with uniform spatial properties. The averaged cross sections for this paste are then used in the B_ℓ equations which account for leakage. The results of the B_ℓ calculation are then used to correct the energy shape of the intra-cell fluxes. In the next section we review the B_ℓ theory.

1.2 The B_ℓ Equations

The transport equation in a uniform homogeneous medium is written in the form

$$\left[\mu \frac{\partial}{\partial \chi} + \Sigma(E) \right] \phi(\chi, E, \mu) = H(\chi, E, \mu) \quad (1a)$$

$$H(\chi, E, \mu) = F(\chi, E, \mu) + \int_{-1}^1 d\mu' \int_0^\infty dE' \Sigma(E' \rightarrow E, \mu_0) \phi(\chi, E', \mu') \quad (1b)$$

$$\mu_0 \equiv \mu \mu'$$

The precise meaning of $F(\chi, E, \mu)$ and $\Sigma(E' \rightarrow E, \mu_0)$ will be detailed later. For this section it is sufficient to say that $F(\chi, E, \mu)$ is a known fixed source term, and $\Sigma(E' \rightarrow E, \mu_0)$ includes all processes which transfer neutrons from one energy to another energy and which are not already accounted for in $F(\chi, E, \mu)$.

JOSHUA DPSTM-500	Title Module BLEAK	Vol.	Page
		IV	F.2.4
		Rev.	Date
			4/70

If one takes the Fourier transform of equation (1), the term $\mu \frac{\partial}{\partial x}$ becomes $iB\mu$, and all other terms involving x are transformed quantities as functions of B , e.g., $\phi(x, E, \mu) \rightarrow \tilde{\phi}(B, E, \mu)$. An alternate approach, which leads to the same equation, but is somewhat easier to visualize, is to assume that

$$\begin{aligned}
H(x, E, \mu) &= H(E, \mu) e^{iBx} \\
\phi(x, E, \mu) &= \phi(E, \mu) e^{iBx} \\
F(x, E, \mu) &= F(E, \mu) e^{iBx}
\end{aligned} \tag{2}$$

where the tilde has been suppressed and the dependence of $\phi(E, \mu)$ on B is understood. It will be shown that the sign of B is unimportant so that using e^{-iBx} would give equivalent results. If solutions for e^{iBx} and e^{-iBx} are combined, spatial distributions like $\cos Bx$ are obtained. Thus, the assumption contained in equations (2) is equivalent to assuming a cosine spatial distribution characterized by a "buckling", B^2 .

The equations obtained by inserting equations (2) into (1) and multiplying by e^{-iBx} are

$$\left[iB\mu + \gamma(E) \right] \phi(E, \mu) = H(E, \mu) \tag{3a}$$

$$H(E, \mu) = F(E, \mu) + \int_{-1}^1 d\mu' \int_0^\infty dE' \gamma(E' \rightarrow E, \mu_0) \phi(E', \mu') \tag{3b}$$

The E dependence will later be handled via the multigroup method. The μ dependence is treated using the Legendre expansions:

$$\phi(E, \mu) = \sum_{m=0}^{\infty} \frac{2m+1}{2} P_m(\mu) \phi_m(E) (iB)^m \tag{4a}$$

$$H(E, \mu) = \sum_{n=0}^{\infty} \frac{2n+1}{2} P_n(\mu) H_n(E) (iB)^n \tag{4b}$$

$$F(E, \mu) = \sum_{n=0}^{\infty} \frac{2n+1}{2} P_n(\mu) F_n(E) (iB)^n \quad (4c)$$

$$\Sigma(E' \rightarrow E, \mu_0) = \sum_{k=0}^{\infty} \frac{2k+1}{2} P_k(\mu_0) \tau_k(E' \rightarrow E) \quad (4d)$$

The factors $(iB)^n$ are included so that all of the fluxes will be real. The next step is to insert (4a), (4c), and (4d) into (3b) and examine the second term of (3b). This term will contain a double summation (over m and k), an integral over μ' , and terms like $P_m(\mu)P_k(\mu_0)$, where again $\mu_0 = \mu\mu'$. The "Addition Theorem" for Legendre polynomials expresses $P_k(\mu_0)$ in terms of Legendre polynomials in μ and μ' . When this theorem is inserted the μ' integration can be performed. After considerable manipulation a very simple result is obtained:

$$H_n(E) = F_n(E) + \int_0^{\infty} dE' \tau_n(E' \rightarrow E) \phi_n(E') \quad (5)$$

The next step is to insert (4a) and (4b) into (3a),

$$\begin{aligned} \left[iB\mu + \tau(E) \right] \sum_{m=0}^{\infty} \frac{2m+1}{2} P_m(\mu) \phi_m(E) (iB)^m = \\ = \sum_{n=0}^{\infty} \frac{2n+1}{2} P_n(\mu) H_n(E) (iB)^n \end{aligned} \quad (6)$$

If this equation is now multiplied by $P_\ell(\mu)$ and integrated over all μ , the so-called " P_ℓ equations" will result. To obtain the " B_ℓ equations" we first divide the equations by the term $[iB\mu + \tau(E)]$, multiply by $P_\ell(\mu)$,

and then integrate over all μ

$$\sum_{m=0}^{\infty} \frac{2m+1}{2} \phi_m(E) (iB)^m \int_{-1}^1 d\mu P_m(\mu) P_l(\mu) =$$

$$\sum_{n=0}^{\infty} \frac{2n+1}{2} H_n(E) (iB)^n \int_{-1}^1 d\mu \frac{P_n(\mu) P_l(\mu)}{\gamma(E) + iB\mu} \quad (7)$$

The orthogonality relation for Legendre Polynomials is

$$\int_{-1}^1 d\mu P_m(\mu) P_l(\mu) = \frac{2}{2m+1} \delta_{ml} \quad (8)$$

where δ_{ml} is the Kronecker Delta. The left-hand side of (7) then reduces to just $\phi_l(E)$. Define

$$Q_{ln}(B, E) = \frac{2n+1}{2} (iB)^{n-l} \int_{-1}^1 d\mu \frac{P_n(\mu) P_l(\mu)}{\gamma(E) + iB\mu} \quad (9)$$

Finally using (8) and (9) in (7) we obtained

$$\phi_l(E) = \sum_{n=0}^{\infty} Q_{ln}(B, E) H_n(E) \quad (10)$$

which (together with equation (5)) are the " B_l equations". Note that these are an infinite set of equations ($l = 0, 1, 2, \dots$) and that they are coupled through equation (5).

The importance of these equations lies in the fact that the infinite sum over n in equation (10) can be accurately truncated to a very few terms, usually four or less. This results from the fact that $\Sigma_n(E' \rightarrow E)$ are very small for $n > \sim 3$ for most materials in the important energy ranges.

Then, if we consider that $H_n(E) = 0$ for $n > N$, it is necessary to consider only the equations for $l = 0, 1, \dots, N$.

If we had derived the "P_l equations" from (6), the right-hand side would contain the terms $\phi_{l-1}(E)$, $\phi_l(E)$, and $\phi_{l+1}(E)$. These equations are simpler in form than the "B_l equations", but we cannot truncate the set of equations to only a few equations because it may take many terms in (4a) to adequately represent the angular flux. The number of terms depends on the value of B; small B requires few terms, large B requires many terms.

It is conventional to call a "B_l calculation" for which $N = 3$ a "B₃ calculation". (Strictly speaking, with the notation used in the equations above, we should have called it a B_N calculation). Similarly, a "P₃ calculation" computes only the fluxes $\phi_l(E)$, $l = 0, 1, 2, 3$.

It is convenient to rewrite equation (10) in the form

$$\nabla(E)\phi_l(E) = H_l(E) + \sum_{n=0}^N R_{ln}(B,E) H_n(E) \quad (11)$$

$$R_{ln}(B,E) = \nabla(E) Q_{ln}(B,E) - \delta_{ln} \quad (12)$$

The reason for doing this will become apparent later when the critical buckling search is described.

The "B_l equations" are used in two ways; the k_{eff} calculation, and the critical buckling calculation.

1.3 The k_{eff} Calculation

We next examine the various physical processes which are lumped into the kernel $\nabla(E' \rightarrow E, \mu_0)$ in equation (1b). These processes are elastic scattering, inelastic scattering, (n,2n) reactions, and fission. Denote these reactions with superscripts el, in, n2n, and f. Thus,

$$\begin{aligned} \nabla(E' \rightarrow E, \mu_0) = & \nabla^{el}(E' \rightarrow E, \mu_0) + \nabla^{in}(E' \rightarrow E, \mu_0) + \\ & + 2 \nabla^{n2n}(E' \rightarrow E, \mu_0) + \nu(E') \nabla^f(E' \rightarrow E, \mu_0) \end{aligned} \quad (13)$$

where we have stated explicitly that two neutrons result from an $(n,2n)$ reaction, and that $\nu(E')$ neutrons are produced (on the average) per fission. Expanding these in Legendre polynomials and considering only the n^{th} coefficient, we obtain

$$\begin{aligned} \Sigma_n(E' \rightarrow E) = & \Sigma_n^{el}(E' \rightarrow E) + \Sigma_n^{in}(E' \rightarrow E) + 2 \Sigma_n^{n2n}(E' \rightarrow E) \\ & + \nu(E') \Sigma_n^f(E' \rightarrow E) \end{aligned} \quad (14)$$

It should be noted at this point that the Legendre cross section for process χ is given by

$$\Sigma_n^\chi(E') \equiv \int_0^\infty dE \Sigma_n^\chi(E' \rightarrow E) \quad (15)$$

and that the total cross section, $\Sigma(E)$, used throughout equations (1) - (12) is defined by

$$\Sigma(E) = \Sigma_0^{el}(E) + \Sigma_0^{in}(E) + \Sigma_0^{n2n}(E) + \Sigma^f(E) + \Sigma^c(E) \quad (16)$$

where $\Sigma^c(E)$ is the capture cross section and $\Sigma^f(E) = \Sigma_0^f(E)$.

The fission term in (14) can be further reduced by assuming that neutrons are emitted isotopically and that the emission spectrum for the k^{th} isotope is independent of the energy of the neutron causing fission. Both of these assumptions are customary and quite accurate. The last term in (14) then reduces to

$$\nu(E') \Sigma_n^f(E' \rightarrow E) = \sum_k \nu_k(E') \Sigma_k^f(E') \chi_k(E) \delta_{n0} \quad (17)$$

where $\chi_k(E)$ is the normalized fission spectrum of the k^{th} isotope. We now insert (14) and (17) into (5) and obtain

$$\begin{aligned} H_n(E) = & \delta_{n0} \sum_k \chi_k(E) \int_0^\infty dE' \nu_k(E') \Sigma_k^f(E') \phi_0(E') + \\ & + \int_0^\infty dE' \left[\Sigma_n^{el}(E' \rightarrow E) + \Sigma_n^{in}(E' \rightarrow E) + 2 \Sigma_n^{n2n}(E' \rightarrow E) \right] \phi_n(E') \end{aligned} \quad (18)$$

What has been done here is to specify what is meant by $F_n(E)$ and $\tau_n(E' \rightarrow E)$ in (5). For the k_{eff} calculation

$$F_n(E) \equiv \delta_{n0} \sum_k \nu_k(E) \int_0^\infty dE' \nu_k(E') \tau_k^f(E') \phi_0(E') \quad (19)$$

$$\tau_n(E' \rightarrow E) \equiv \tau_n^{el}(E' \rightarrow E) + \tau_n^{in}(E' \rightarrow E) + 2 \tau_n^{n2n}(E' \rightarrow E) \quad (20)$$

With these definitions the B_l equations are homogeneous and will have a solution only if the value of B is an eigenvalue. If B is fixed then an auxiliary eigenvalue must be introduced to obtain a solution. This eigenvalue is called " k_{eff} " and is introduced in the following way.

The integral in (19) is just the total number of neutrons F_k produced by fission in isotope k . Since the equations are homogeneous we can arbitrarily normalize the flux so that one neutron is produced by fission.

$$\sum_k F_k = \sum_k \int_0^\infty dE' \nu_k(E') \tau_k^f(E') \phi_0(E') = 1 \quad (21)$$

$$F_n(E) = \delta_{n0} \bar{\chi}(E) = \delta_{n0} \sum_k \nu_k(E) F_k \quad (22)$$

If we use (22) in the B_l equations, perform one flux iteration, and calculate the fission neutrons produced by this first flux iterate, we would find that (in general) the number was different than one. We call this number " k_{eff} ", the effective multiplication, and introduce it into equation (22) as the eigenvalue.

$$F_n(E) = \delta_{n0} \bar{\chi}(E) / k_{eff} \quad (23)$$

1.4 The Critical Buckling Calculation

In the k_{eff} calculation we held the buckling, B^2 , fixed and introduced another eigenvalue, k_{eff} , to obtain a solution of the B_l equations. Here we vary B^2 until a solution is obtained. The value of B^2 which satisfies the B_l equations is denoted by B_c^2 and is called the

"critical buckling", that is, the reactor is just critical and $k_{eff} = 1$. Since the eigenvalue is implicit we must use an iterative procedure. Note that k_{eff} was an explicit eigenvalue and we did not have to iterate.

We could use the procedure outlined in the previous section, that is, compute k_{eff} for some fixed B^2 . If k_{eff} is not unity, vary B^2 and recalculate k_{eff} . Repeat this procedure until $k_{eff} = 1$. Fortunately, there is a much faster method.

Let us take equation (11) for $\phi_0(E)$ and integrate over E .

$$\int_0^\infty dE \left[\tau(E) \phi_0(E) - H_0(E) \right] = \sum_{n=0}^N \int_0^\infty dE R_{0n}(B, E) H_n(E) \quad (24)$$

Use equations (5), (14), (17), and set $F_0(E) = 0$, so that

$$H_0(E) = \int_0^\infty dE' \left[\tau_0^{el}(E' \rightarrow E) + \tau_0^{in}(E' \rightarrow E) + 2 \tau_0^{n2n}(E' \rightarrow E) + \sum_k v_k(E') \tau_k^f(E') \chi_k(E) \right] \phi_0(E') \quad (25)$$

When (25) is inserted in (24), the order of integration changed, and definition (16) is used, equation (24) becomes

$$\begin{aligned} \int_0^\infty dE \left\{ \tau^c(E) - \tau_0^{n2n}(E) - \sum_k [v_k(E) - 1] \tau_k^f(E) \right\} \phi_0(E) = \\ = \sum_{n=0}^N \int_0^\infty dE R_{0n}(B, E) H_n(E) \end{aligned} \quad (26)$$

Examination of the $R_{0n}(B, E)$ shows that the leading terms, in an expansion in power of B^2 are B^2 for R_{00} and R_{01} , B^4 for R_{02} , B^6 for R_{03} , etc. Define

$$R'_{0n}(B, E) = R_{0n}(B, E) / B^2 \quad (27)$$

Then R'_{00} and R'_{01} are constants plus higher order terms in B^2 .

$$\int_0^{\infty} dE \left\{ \Sigma^c(E) - \Sigma_0^{n2n}(E) - \sum_k [\nu_k(E) - 1] \Sigma_k^f(E) \right\} \phi_0(E) =$$

$$= B^2 \sum_{n=0}^N \int_0^{\infty} dE R'_{0n}(B, E) H_n(E) \quad (28)$$

Let us examine each term in (28)

$$\int_0^{\infty} dE \Sigma^c(E) \phi_0(E) = \text{total number of neutrons captured}$$

$$\sum_k \int_0^{\infty} dE [\nu_k(E) - 1] \Sigma_k^f(E) \phi_0(E) = \text{net number of neutrons produced by fission (1 in, } \nu(E) \text{ out)}$$

$$\int_0^{\infty} dE \Sigma_0^{n2n}(E) \phi_0(E) = \text{net number of neutrons produced by the (n,2n) reaction. (1 in, 2 out)}$$

The left-hand side of (28) is then the net number of neutrons lost in nuclear reactions. The right-hand side must be the number of neutrons which leak into the system. The diffusion theory analogy of this term is $-M^2 B^2 \Phi$, where Φ is the total scalar flux, and M^2 is the "migration area". Then

$$M^2 = - \sum_{n=0}^N \int_0^{\infty} dE R'_{0n}(B, E) H_n(E) / \int_0^{\infty} dE \phi_0(E) \quad (29)$$

Since $R'_{0n}(B, E)$ is insensitive to B , equation (28) can be used to predict a new value of B^2 in the iteration.

$$B^2 = \frac{\int_0^\infty dE \left\{ \tau^c(E) - \tau_0^{n2n}(E) - \sum_k [\nu_k(E) - 1] \tau_k^f(E) \right\} \phi_0(E)}{\sum_{n=0}^N \int_0^\infty dE R'_{0n}(B, E) H_n(E)} \quad (30)$$

1.5 Time Eigenvalue Calculation

If the time-dependent flux has the asymptotic form,

$$\phi(x, E, \mu, t) = e^{-\alpha t} \phi(x, E, \mu) \quad (31)$$

substitution into the time-dependent transport equation yields an equation like (1) except that the total cross section is reduced by α/v , where v is the neutron velocity. This term is called the "time absorption" cross section and we can use an effective absorption cross section

$$\tau_{eff}^a(E) = \tau^a(E) - \alpha/v \quad (32)$$

$$v = 2.2 \times 10^5 \sqrt{E/0.025290} \text{ cm/sec} \quad (33)$$

in all of the previous equations. If α is given, the k_{eff} or critical buckling calculation is the same as outlined in the previous sections. If k_{eff} and B^2 are given, the problem is to find the eigenvalue α .

Assume that we have some value α , and that a k_{eff} calculation has been done yielding a value k_1 . We wish to estimate a new value α_2 which will yield the desired $k_{eff} = k_2$. If the change in the total absorption A is small, the fractional change in A is equal to the fractional change in k .

$$\delta A/A = \delta k/k \quad (34)$$

If we assume that we have a thermal reactor and that the absorption cross section varies like $v_0 \sigma_0^a v$, then

$$A \sim (v_0 \sigma_0^a - \alpha) / v \quad (35)$$

$$\delta A / A \doteq -\delta \alpha / (v_0 \sigma_0^a - \alpha) \quad (36)$$

$$\delta \alpha = (v_0 \sigma_0^a - \alpha) \delta k / k \quad (37)$$

$$\alpha_2 = \alpha_1 - (v_0 \sigma_0^a - \alpha) (k_2 - k_1) / k_1 \quad (38)$$

The quantity $v_0 \sigma_0^a$ is estimated from the absorption cross section at $E = 0.0253$.

1.6 The Coefficients R_{ln}

It is somewhat easier to work with a real variable than the imaginary quantity (iB). Define

$$\kappa = iB \quad (39)$$

It will turn out that the R_{ln} are even functions of κ . Equation (12) then becomes

$$R_{ln}(\kappa, E) = \kappa^{n-l} \left[\frac{2n+1}{2} \sigma(E) \int_{-1}^1 d\mu \frac{P_l(\mu) P_n(\mu)}{\sigma(E) - \mu \kappa} - \delta_{ln} \right] \quad (40)$$

Define

$$C = C(E) = \kappa / \sigma(E) \quad (41)$$

The orthogonality relation for Legendre polynomials is

$$\frac{2n+1}{2} \int_{-1}^1 d\mu P_l(\mu) P_n(\mu) = \delta_{ln} \quad (42)$$

so that (31) can be written

$$\begin{aligned}
 R_{ln}(\kappa, E) &= \kappa^{n-l} \frac{2n+1}{2} \int_{-1}^1 d\mu P_l(\mu) P_n(\mu) \left[\frac{1}{1 - C\mu} - 1 \right] \\
 &= C \kappa^{n-l} \frac{2n+1}{2} \int_{-1}^1 d\mu \frac{\mu P_l(\mu) P_n(\mu)}{1 - C\mu}
 \end{aligned} \tag{43}$$

Multiply and divide the integrand by $(1 + C\mu)$, and use the fact that $P_n(\mu)$ is an even function of μ if n is even, and an odd function if n is odd.

$$\begin{aligned}
 R_{ln}(\kappa, E) &= \frac{2n+1}{\Sigma(E)} \kappa^{n-l+1} \int_0^1 d\mu \frac{\mu P_l(\mu) P_n(\mu)}{1 - C^2 \mu^2}, \quad l+n \text{ odd} \\
 &= \frac{2n+1}{\Sigma^2(E)} \kappa^{n-l+2} \int_0^1 d\mu \frac{\mu^2 P_l(\mu) P_n(\mu)}{1 - C^2 \mu^2}, \quad l+n \text{ even}
 \end{aligned} \tag{44}$$

The first few Legendre polynomials are:

$$\begin{aligned}
 P_0(\mu) &= 1 & P_2(\mu) &= \frac{1}{2} (3\mu^2 - 1) \\
 P_1(\mu) &= \mu & P_3(\mu) &= \frac{1}{2} (5\mu^3 - 3\mu)
 \end{aligned}$$

The integrals in (35) can all be expressed in terms of the elementary integrals

$$I_n = \int_0^1 d\mu \frac{\mu^m}{1 - C^2 \mu^2}, \quad m = 0, 2, 4, \dots \tag{45}$$

which has the recursion formula

$$C^2 I_m = I_{m-2} - \frac{1}{m-1} \tag{46}$$

and the series expansion for small values of $|c^2| \leq 0.20$

$$I_m = \sum_{k=0}^4 c^{2k} / (2k+m+1) \quad (47)$$

Values for $m=0$ are

$$I_0 = \frac{1}{2C} \ln \left(\frac{1+C}{1-C} \right) , \quad c^2 > 0 \quad (48)$$

$$= \frac{1}{C} \tan^{-1} C , \quad c^2 < 0$$

The resulting expressions for the R_n are given in Table I. It should be noted that they are all even functions of κ and that the dominant dependence on κ has been factored out so that I and T coefficients are non-zero constants at $\kappa=0$.

TABLE I

EXPRESSIONS FOR THE R_{ln} COEFFICIENTS

l	n	R_{ln}	l	n	R_{ln}
0	0	$\kappa^2 I_2 / \Sigma^2$	2	0	T_1 / Σ^2
0	1	$3\kappa^2 I_2 / \Sigma$	2	1	$3T_1 / \Sigma$
0	2	$5\kappa^4 T_1 / \Sigma^2$	2	2	$5\kappa^2 T_4 / \Sigma^2$
0	3	$7\kappa^6 T_2 / \Sigma^3$	2	3	$7\kappa^2 T_3 / \Sigma$
1	0	I_2 / Σ	3	0	T_2 / Σ^3
1	1	$3\kappa^2 I_4 / \Sigma^2$	3	1	$3T_2 / \Sigma^2$
1	2	$5\kappa^2 T_1 / \Sigma$	3	2	$5T_3 / \Sigma$
1	3	$7\kappa^4 T_2 / \Sigma^2$	3	3	$7\kappa^2 T_5 / \Sigma^2$

$$I_m \equiv \int_0^1 d\mu \mu^m / (1 - C^2 \mu^2)$$

$$T_1 \equiv \frac{1}{2} (3I_4 - I_2)$$

$$T_2 \equiv \frac{1}{2} (5I_6 - 3I_4)$$

$$T_3 \equiv \frac{1}{4} (15I_6 - 14I_4 + 3I_2)$$

$$T_4 \equiv \frac{1}{4} (9I_6 - 6I_4 + I_2)$$

$$T_5 \equiv \frac{1}{4} (25I_8 - 30I_6 + 9I_4)$$

1.7 The Multigroup Approximation

The multigroup approximation is made by replacing all integrals over energy by a sum over energy groups.

$$\int_0^{\infty} dE \phi_l(E) = \sum_{j=1}^{NEG} \phi_l(j) \quad (49)$$

Note that $\phi_l(j)$ is the total flux in group j rather than the average flux. Equation (11) becomes

$$\Sigma(j)\phi_l(j) = H_l(j) + \sum_{n=0}^N R_{ln}(B,j) H_n(j) \quad (50)$$

Equations (18 - 23) can then be written as

$$H_n(j) = F_n(j) + \sum_{i=1}^{NEG} \tau_n(i \rightarrow j) \phi_n(i) \quad (51)$$

$$F_n(j) = \delta_{no} \bar{\chi}(j)/k_{eff} \quad (52)$$

$$= \delta_{no} \sum_k \chi_k(j) F_k/k_{eff} \quad (53)$$

$$F_k = \sum_{i=1}^{NEG} \nu_k(i) \Sigma_k^f(i) \phi_o(i) \quad (54)$$

1.8 Resonance Treatment

In using the multigroup approximation the cross sections used in (50) are the "smooth" cross sections. Resonance capture is included by adjusting the slowing-down source terms as follows:

Define the slowing-down density as

$$Q(j) = \sum_{i < j} \phi_o(i) \sum_{m \geq j} \tau_o(i \rightarrow m) \quad (55)$$

that is, the total number of neutrons/sec which slow down past the energy boundary $E(j)$. A fraction of these, $R^a(j)$, are absorbed in group j . The components of the slowing down source into group j are given by

$$H_n^1(j) = \sum_{i=1}^{j-1} \tau_n(i \rightarrow j) \phi_n(i) \quad (56)$$

The scalar source is reduced by $R^a(j)Q(j)$ to account for resonance capture. This corresponds to defining a reduction factor

$$\rho(j) = \left[H_0^1(j) - R^a(j)Q(j) \right] / H_0^1(j) \quad (57)$$

which is then applied to all $H_n^1(j)$.

The $Q(j)$ can be computed recursively from the formula

$$Q(j+1) = Q(j) - H_0^1(j) + \tau^R(j)\phi_0(j) \quad (58)$$

where $\tau^R(j)$ is the removal cross section for group (j) .

2. Program Logic

2.1 Interfaces

BLEAK reads the following records:

```
('TEMP'. 'LIDS'. DLAT. VLAT → V)
'INPUT'. 'BLEAK'
V. 'GRPS'. NCASE. O. O
V. 'HØMX'. NCASE. O. O
V. 'PMAT'. NCASE. NEB. O
```

2.2 Module Structure

The BLEAK module consists of a main program and a hierarchical set of subroutines.

```
BLEAK
  SEARCH
    BABL
      BLFIS
      SØURCE
      BLEQ
      RCALC
```

A flow diagram for the BLEAK modules is shown in Figure 1. The input record specifies what cross section data is to be used and the number and type of searches to be done.

Figure 1. BLEAK Flow Diagram

JOSHUA DPSTM-500	Title Module BLEAK	Vol. IV	Page F.2.21
		Rev.	Date 4/70

2.3 Eigenvalue Search (SEARCH)

Associated with each case to be done is a number, LSØP, indicating the type of eigenvalue search desired.

<u>LSØP</u>	<u>Search Type</u>
1	k_{eff} calculation
2	Buckling (B^2) search.
3	Time eigenvalue (α) search.

Assume that a subroutine (BABL) is available which, given the cross sections, B^2 , and α , will compute an eigenvalue λ . The eigenvalue search consists of varying either B^2 or α until λ equals an input value k_{eff}^* to within a small (10^{-5}) error, that is, until

$$|k_{\text{eff}}^* - \lambda| \leq 10^{-5}$$

We assume that an input value of B^2 or α is available to start the calculation. If LSØP = 3, the cross sections are reduced by the "time absorption cross section".

$$\Sigma^t(j) = \Sigma^t(j) - \alpha/v_j \quad (59)$$

$$\Sigma^a(j) = \Sigma^a(j) - \alpha/v_j \quad (60)$$

$$v_j = 2.2 \times 10^5 \sqrt{E(j)/.02529} \quad (61)$$

Using these values an eigenvalue λ is obtained. If the convergence criterion is satisfied, the search is finished. If not a new value of B^2 or α is predicted from

$$B^2(\text{new}) = B^2 + (B^2 + \frac{1}{M^2}) (\lambda - k_{\text{eff}}^*)/k_{\text{eff}}^* \quad (62)$$

$$\alpha(\text{new}) = \alpha + v \Sigma_0^a (k_{\text{eff}}^* - \lambda)/\lambda \quad (63)$$

where M^2 is the migration area and $v \Sigma_0^a$ is the value of $v \Sigma^a(E)$ at $E = 0.02529$ ev.

The eigenvalue search is done by subroutine SEARCH (see Figure 2). SEARCH also establishes the constant LENR required by the basic calculation BABL.

LENR = 1, Use epithermal groups only
 = 2, Use Thermal groups only
 = 3, Use all energy groups

Figure 2. Subroutine SEARCH Flow Diagram

LENR is determined by LSØP and the constants LXD and LAD.

LXD = 1 Only fast cross sections are given
(JFTG > NEG).

= 2 Only thermal cross sections are given (JFTG = 1).

= 3 Both fast and thermal cross sections are given
(JFTG > 1, JFTG ≤ NEG).

LAD = 0 Active lattice (fission).

= 1 Dead lattice (no fission).

The following table summarizes the allowed combinations.

LXD	LAD	LSØP	LENR	Type of Calculation
1	0	1	1	Fast Reactor k_{eff} .
		2,3	1	Fast Reactor critical buckling or α .
	1	1	1	Fast neutron spectrum, fission source.
		2,3	0	Not allowed.
2	0	1	0	Not allowed.
		2,3	2	Thermal L^2 or decay constant.
	1	1	0	Not allowed.
		2,3	2	Thermal L^2 or decay constant.
3	0	1	3	Thermal Reactor k_{eff} .
		2,3	3	Thermal Reactor critical buckling or α .
	1	1	3	Neutron spectrum, fission source.
		2,3	2	Thermal L^2 or decay constant.

2.4 Basic B_g Calculation (BABL)

Assume that all cross sections and B^2 are given. The BABL subroutine solves the B_g equations and determines the eigenvalue λ and the group fluxes $\phi_g(j)$. Flow diagrams for BABL are shown in Figures 3 - 6.

The calculation starts with a normalized fission spectrum, $\bar{\chi}(j)/k_{eff}$. The fluxes in each epithermal group can then be computed in a step-wise manner using equations (50-51) since the scattering matrix in (51) depends only on higher energy (lower numbered) group fluxes. Subroutine SØRCE

is used to compute the scattering term in (5b) and supply BABL with cross sections and resonance rates. The source is modified for resonance capture according to equation (57). Subroutine BLEQ then solves the one-group B_l equation.

In the thermal region the scattering matrix in (51) contains both up- and down-scattering elements. The calculation consists of two phases. The first phase is the calculation of the thermal source, that is, the number of neutrons entering the thermal region from the epithermal region. If no prior estimate of the thermal flux exists, a guess is established.

$$\phi_0(j) = \bar{E}(j) \Delta E(j) \exp [-\bar{E}(j)/.0253] \quad (64)$$

$$\phi_l(j) = \phi_{l-1}(j) (l+1)/(2l+3) \Sigma^t(j), \quad l = 1, 2, \dots \quad (65)$$

Figure 3. Subroutine BABL Flow Diagram

Figure 4. Flow Diagram of Epithermal Calculation in BABL.

Figure 5. Flow Diagram of Thermal Source Calculation in BABL.

Figure 6. Flow Diagram of Thermal Flux Calculation in BABL.

The fluxes are scaled so that neutrons are conserved, that is, the total slowing-down source equals the total thermal absorptions plus leakage.

$$\sum_{j=JFTG}^{NEG} \left[\tau^a(j) \phi_o(j) + B^2 \phi_1(j) \right] = \sum_{j=JFTG}^{NEG} S_o(j) \quad (66)$$

The second phase is the iteration of the thermal flux. Starting with the first thermal group ($j = JFTG$), the source obtained from subroutine *SORCE*, and the group fluxes obtained from subroutine *BLEQ*. The new fluxes replace the old fluxes and are used in the source calculation for each subsequent group. When all thermal groups have been recalculated, the entire set is scaled to force neutron balance. If the fractional change in the $\phi_o(j)$ in one iteration is less than 10^{-5} , the flux has converged. If not, another iteration is done starting with $j = JFTG$. The iteration continues until either the solution converges or the number of iterations exceeds a number input to *BABL*.

After a complete set of fluxes has been obtained, the number of fission neutrons (λ) and a new fission spectrum ($\bar{\chi}(j)/\lambda$) is computed from equations (52 - 54) using subroutine *BLFIS*. If this fission spectrum is the same (within 10^{-5}) of the fission spectrum used to start the calculation, the fission spectrum has also converged and the *BABL* calculation is finished. If the fission spectrum has not converged, the entire calculation of the flux is repeated once.

2.5 Fission Source Calculation (BLFIS)

The total fission neutrons produced by the k^{th} isotope is

$$F_k = \sum_{i=1}^{NEG} \nu_k(i) \left[\tau_k^f(i) \phi_o(i) + R_k^f(i) Q(i) \right] \quad (67)$$

Since the calculation of the flux was started with one source neutron (normalized fission spectrum), the eigenvalue λ is just the number of fission neutrons produced by that source.

$$\lambda = \sum_k F_k \quad (68)$$

The fission spectrum is computed from

$$F_n(j) = \delta_{no} \sum_k x_k(j) F_k / \lambda \quad (69)$$

If $\lambda = 0$, there is no fission and the flag LAD is set to unity indicating a dead lattice. In this case $F_0(j)$ is computed from the Watts fission spectrum which can be written as

$$F_0(j) = g(x_1) - g(x_2) \quad (70)$$

$$g(x) = 1.1283792 x e^{-x^2} - \text{erf}(x)$$

$$x_1 = \sqrt{E(j+1)/\theta}$$

$$x_2 = \sqrt{E(j)/\theta}$$

$$\theta = 1.273 \times 10^6 \text{ ev}$$

2.6 Slowing-Down Source Calculation (SØRCE)

For a given final group j , and a range of initial groups $IL \leq i \leq IH$, subroutine SØRCE computes the slowing-down source

$$H_n^1(j) = \sum_{i=IL}^{IH} \tau_n(i \rightarrow j) \phi_n(i) \quad (71)$$

The routine also provides the calling program with the cross sections $\Sigma^a(j)$, $\tau^t(j)$, $\tau_n(j \rightarrow j)$, and the resonance absorption rate, $R^a(j)$.

2.7 One-Group B_L Equation (BLEQ)

The subroutine BLEQ solves the one-group B_L equations assuming all other group fluxes are known. We can then decompose (51) into a constant term plus a term proportional to $\phi_n(j)$.

$$H_n(j) = \left[F_n(j) + \sum_{\substack{i=1 \\ i \neq j}}^{\text{NEG}} \Sigma_n(i \rightarrow j) \phi_n(i) \right] + \Sigma_n(j \rightarrow j) \phi_n(j) \quad (72)$$

$$= H_n^1(j) + \Sigma_n(j \rightarrow j) \phi_n(j)$$

If this is inserted into (50), we obtain

$$\Sigma^t \phi_l = \left[H_l^1 + \Sigma_l \phi_l \right] + \sum_{n=0}^N R_{ln}(B) \left[H_n^1 + \Sigma_n \phi_n \right] \quad (73)$$

where the group number j has been suppressed for convenience. Collecting terms containing ϕ_l on the LHS,

$$\phi_l = \beta_l + \sum_{\substack{n=0 \\ n \neq l}}^N \gamma_{ln} \phi_n \quad (74)$$

$$\alpha_l = \Sigma - \Sigma_l \left[1 + R_{ll}(B) \right] \quad (75)$$

$$\beta_l \left[H_l^1 + \sum_{n=0}^N R_{ln}(B) H_n^1 \right] / \alpha_l \quad (76)$$

$$\gamma_{ln} = R_{ln}(B) \Sigma_n / \alpha_l \quad (77)$$

If $N = 0$, the solution is just $\phi_l = \beta_l$. If $N > 0$, an iterative method is used since γ_{ln}/β_l is generally small. The iteration proceeds as follows:

1. Set $\phi_l = \beta_l$.
2. Insert the current ϕ_l into the RHS of (74) and compute a new ϕ_l .
3. Compare the old and new ϕ_l . If the fractional change is greater than 10^{-6} , repeat step 2.
4. If more than 50 iterations are required, the coefficients are probably in error and the calculation is terminated.

Following the calculation of the flux it is convenient to calculate the diffusion coefficient

$$D = - \sum_{n=0}^N R_{on}(B) \left[H_n^1 + \tau_n \phi_n \right] / B^2 \phi_0 \quad (78)$$

which is required for neutron conservation calculations.

2.8 Storage Considerations

The BLEAK module should accommodate:

20 cases
100 groups
4 Legendre components (P_0, P_1, P_2, P_3)

3. Coding Details

The BLEAK module is divided into the following parts:

1. COMMON/DATA/
2. COMMON/SPECS/
3. COMMON/GRUUPS/
4. COMMON/TEMP/
5. COMMON/SCRACH/
6. Main program BLEAK
7. Subroutine SEARCH
8. Subroutine BABL
9. Subroutine BLFIS
10. Subroutine SORCE
11. Subroutine BLEQ
12. Subroutine RCALC
13. Subroutine MANAGE (See Manuscript B.1)

3.1 COMMON/DATA/

The LIDS blocks HØMX and PMAT are packed in the array data defined by

COMMON/DATA/DATA(20000).

Refer to Manuscript B.1 for a description of this array and its usage.

3.2 COMMON/SPECS/

The record 'INPUT'. 'BLEAK' contains the data:

TDSN - Temporary data set name = 'TEMP'
 DLAT - Lattice name
 VLAT - Lattice version
 NCASE - Case number
 LPRNT - If LPRNT = 1, the maximum length of the DATA array used will be printed
 NBSX - Number of buckling searches
 LSØPT - Search option
 KEFFT - The k_{eff} used for a buckling search
 BSQDT - The buckling used for a k_{eff} calculation
 ALPHAT - Time eigenvalue

These data are stored in /SPECS/.

3.3 COMMON/GRØUPS/

This area contains the group structure data written by CRISP on the LIDS, the quantities used in BLEAK are:

NEG - Number of energy groups
 JFTG - Index to first thermal group
 E(I) - Energy break point (EV)
 EBAR(I) - Characteristic energy (EV)
 DE(I) - Delta energy for each energy group
 NEBX - Number of energy bands
 IEBH(NEB) - Energy band break points

3.4 COMMON/TEMP/

This area is used to pass calculations from subroutines SEARCH, BABL, BLFIS, and SØRCE to BLEAK. The items in this area are as follows:

LX - Maximum flux Legendre number
 LPX - Maximum flux Legendre index = LX + 1
 LPXX - Largest value of LPX
 NX - Maximum scattering Legendre number
 NPX - Maximum scattering Legendre index = NX + 1
 DLKG(I) - Diffusion coefficient
 QSD(I) - Slowing-down density
 Q(LP,I) - Source

PHI(LP,I) - Legendre flux
 LAD - Lattice fission flag
 0 - Active lattice
 1 - Dead lattice

3.5 CØMMØN/SCRACH/

This area is used by BABL for temporary storage of the flux in order to test for convergence.

4. User Information

4.1 Input

BLEAK cannot be used as a stand-alone module since it relies on the blocks HØMX and PMAT written by PASTE and the group structure record written by CRISP. If these records have been created, BLEAK requires only the record:

Name: 'INPUT'.'BLEAK'

List: DLAT,VLAT,TDSN,NCASE,LPRNT,NBSX,(LSØPT(N),KEFFT(N),
 BSQDT(N),ALPHAT(N),N=1,NBSX)

DLAT - Lattice name
 VLAT - Lattice version
 TDSN - Temporary data set name = 'TEMP'
 NCASE - Case number
 LPRNT - If LPRNT = 1, the maximum length of the DATA array used will be printed
 NBSX - Number of searches
 LSØPT(N) - Type of eigenvalue search desired
 1 - k_{eff} calculation enter BSQD and ALPHA
 2 - Buckling (B^2) search, enter KEFF and ALPHA
 3 - Time eigenvalue (α) search, enter KEFF and ALPHA
 KEFFT(N) - k_{eff}
 BSQDT(N) - Buckling (B^2)
 ALPHAT(N) - Time eigenvalue (α)

Normally, this record is created by the RAHAB control module. These items are used to establish the names of the input and output records.

4.2 Output

Two records are computed by BLEAK and written on the LIDS. For each search N, a Legendre flux record is written:

Name: 'TEMP'.'LIDS'.DLAT.VLAT.'LFLX'.NCASE.N.O

List: NEG,LPX,LQSD(J),DLKG(J),(PHI(LP,J),LP=1,LPX),J=1,NEG)

NEG - Number of energy groups

LPX - Maximum flux Legendre index

QSD(J) - Slowing-down density

DLKG(J) - Diffusion coefficients

PHI(LP,J) - Legendre flux

Also a search record is written:

NAME: 'TEMP'.'LIDS'.DLAT.VLAT.'SRCH'.NCASE.O.O

List: NBSX,(LSOPT(N),KEFFT(N),BSQDT(N),MSQDT(N),ALPHAT(N),
N=1,NBSX)

NBSX - Number of searches

LSOPT(N) - Search option (see section 4.1)

KEFFT(N) - k_{eff} eigenvalue

BSQDT(N) - Buckling (B^2)

MSQDT(N) - Migration area

ALPHAT(N) - Time eigenvalue

4.3 Core Size and Timing

The BLEAK module requires 193K of core, 1 job data set block, and should be executed with a GØTIME = 1 minute. Individual subprogram sizes and timings are given in Table II. The timings were for a uniform lattice with two searches, a k_{eff} calculation and a buckling search.

4.4 Error Stops

All error stops are handled by the JØSHUA operating system and are summarized in Table III.

TABLE II

SIZE AND TIMING FOR BLEAK SUBPRØGRAMS
(Size in bytes, time in CPU seconds)

<u>SUBPRØGRAM</u>	<u>SIZE</u>	<u>TØTAL TIME</u>	<u>AVERAGE TIME</u>	<u>ENTRIES</u>
MAIN	2934	.44928	.44928	1
SEARCH	1840	.01600	.00800	2
BABL	2640	1.98016	.16501	12
BLFIS	1902	.36608	.02615	14
SØRCE	1664	7.53792	.00224	3372
BLEQ	1654	3.52768	.00120	2952
RCALC	1474	6.92224	.00234	2952
MANAGE*	11,850	5.00864**	.00036	13539
IHCSLØG	442			
IHCFRXPI	321			
IHCSATN2	472			
IHCSERF	456			
IHCSEXP	432			
IHCSSQRT	329			
DATA	80000			
GRØUPS	1624			
SPECS	28			
TEMP	4124			
SCRACH	400			
DATCAT	2440			
JØS***	<u>75,486</u>			
	192,512	<u>25.80800</u>		

* Includes all subroutine of the data manager.

** Includes time for JØSHUA Operating System

*** JØSHUA Operating System and buffers.

TABLE III

BLEAK ERROR STOPS

Message / Explanation

BLEAK.MAIN, HØMX BLØCK MISSING

The homogenized cross section block written by PASTE is not in the LIDS.

BLEAK.MAIN, PMAT BLØCK MISSING

The homogenized scattering matrix blocks written by PASTE are not in the LIDS.

BLEAK.MAIN, CØNTRØL RECØRD MISSING

The record 'INPUT'.'BLEAK' is missing.

BLEAK.MAIN, GRØUPS RECØRD MISSING

The group structure record written by CRISP is not in the LIDS.

BLEAK.SEARCH, ILLEGAL SEARCH ØPTIØN

The input record contains an option which is not allowed.

BLEAK.SEARCH, PØINTER MISSING

The HØMX block is not in the data array.

BLEAK.SEARCH, TØØ MANY CRITICAL BUCKLING SEARCHES

The number of iterations necessary for the buckling to converge exceeds the maximum allowed.

BLEAK.SEARCH, TØØ MANY TIME EIGENVALUE SEARCHES

The number iterations necessary for the time eigenvalue to converge exceeds the maximum allowed.

BLEAK.BLFIS, PØINTER MISSING

The HØMX block is not in the data array.

BLEAK.SØRCE, ENERGY BAND TABLE ERRØR

The table of energy band break points was not set up correctly in CRISP.

BLEAK.SØRCE, PØINTER MISSING

Either PMST or HØMX block is not in the data array.

BLEAK.SØRCE, SMAT ERRØR

A final group number cannot be found in the SMAT block.

BLEAK.BLEQ, TØØ MANY ITERATIØNS

The flux did not converge in the maximum number of allowable iterations.

Contents

- G.1 Intermediate Data Set Print - Module RAPID.
- G.2 RAHAB Edit - Module RABED.
- G.3 Foil Activation Edit - Module FOILED.

JOSHUA
DPSTM-500

Title Part C - Edit Modules

Vol.
IV

Page
G.O.2

Rev.
1

Date
4/72

(Blank)

Work Done By

C. E. Henderson
W. H. Hightower
H. C. Honeck

Report Written By

C. E. Henderson

Contents

	<u>Page</u>
1. Introduction	G.1.3
2. Program Logic	G.1.3
2.1 General Description	G.1.3
2.2 Interfaces	G.1.3
3. Coding Details	G.1.4
3.1 COMMON/DATA/	G.1.4
3.2 COMMON/SPECS/	G.1.4
3.3 COMMON/GRØUPS/	G.1.5
3.4 COMMON/SCRACH/	G.1.5
3.5 Main Program RAPID	G.1.5
3.6 Print Subroutines	G.1.5
3.7 Subroutine NAME	G.1.6
4. User Information	
4.1 Input	G.1.6
4.2 Core Size and Timing	G.1.7

JOSHUA

DPSTM-500

Title Module RAPID

Vol. IV

Page
G.1.2

Rev.

Date
6/70

(BLANK)

1. Introduction

Although the user is generally not concerned with the records written by RAHAB to the LIDS, there may be instances where he wishes to examine these temporary records. For example, if an error stop occurs in one of the calculation modules of RAHAB, the data blocks used by this module might aid in discovering the cause of the error. All such printing of temporary data blocks is done by the print module RAPID.

The RAHAB control program is set up so that RAPID is executed at user option or for error recovery. If the user chooses to print certain records, he merely fills out the input record for RAPID, specifying which blocks he wishes to print. For error recovery, the control module RAHAB decides which blocks need to be printed and writes the input record to RAPID.

2. Program Logic

2.1 General Description

The RAPID module consists of a main program, the RAHAB data manager (see Manuscript B.1), and a group of print subroutines. The main program calls the data manager to read the blocks specified by an option flag in the input record; it then calls either the data manager to dump the block as it is stored on the LIDS or a subroutine to print the block in an easily readable format.

2.2 Interfaces

RAPID is exclusively a printing module so there are no records written by RAPID on the LIDS. RAPID reads the input record

'INPUT'.'RAPID'

as well as records on the LIDS of the form

'TEMP'.'LIDS'.DLAT.VLAT.BLØK.NCASE.N1.N2

where DLAT	-	Lattice name
VLAT	-	Lattice version
BLØK	-	Block name
NCASE	-	Case number
N1,N2	-	Qualifying integers which may be:
		NEB - energy band number
		NZN - spatial zone number
		NBS - buckling search number
		NRG - region number

If RAPID is executed as a stand-alone module or if the user chooses to print certain records, then the user creates the input record. Otherwise, the input record is created by the RAHAB path.

3. Coding Details

The module RAPID is divided into the following parts:

1. CØMMØN/DATA/
2. CØMMØN/SPECS/
3. CØMMØN/GROUPS/
4. CØMMØN/SCRACH/
5. Main program RAPID
6. Subroutine PNTRES
7. Subroutine PXSBRN
8. Subroutine PRRBRN
9. Subroutine PUTMAT
10. Subroutine PUTØUT
11. Subroutine PUTZØN
12. Subroutine PTM
13. Subroutine FLXPNT
14. Subroutine PNTQCL
15. Subroutine PNTJ
16. Subroutine NAME
17. Subroutine MANAGE (See Manuscript B.1)

3.1 CØMMØN/DATA/

The LIDS blocks are packed into the array DATA defined by

CØMMØN/DATA/DATA(10000)

Refer to Manuscript B.1 for a description of this array and its usage.

3.2 CØMMØN/SPECS/

The following data is read in the record 'INPUT'. 'RAPID' and stored in /SPECS/:

TDSN	-	Temporary data set name 'TEMP'
DLAT	-	Lattice name
VLAT	-	Lattice version
NCASE	-	Case number

3.3 COMMON/GRØUPS/

This area contains the group structure data from the MULTIGRP data set. The quantities used in RAPID are:

- NEBX - Number of energy bands
- NEG - Number of energy groups

3.4 COMMON/SCRACH/

This is a scratch area used as temporary storage by the subroutines which print the data blocks.

3.5 Main Program RAPID

As mentioned in section 2.1, the main program consists entirely of calls to the data manager and to the print subroutines. The choice of the records to read and print is made on the basis of an option flag in the input record. In reading these records from the LIDS, certain qualifying integers are sometimes needed; for example, the energy band number NEB, the buckling search number NBS, the zone number NZN, and the region number NRG. In printing or dumping a group of records, RAPID utilizes the fact that all qualifying integers are used in a sequential fashion in naming these records. Starting with one and increasing the integer by one after each successful read, RAPID reads all records qualified by the integer. After the first unsuccessful read, RAPID continues to the next part of the program.

3.6 Print Subroutines

The following is a list of the print subroutines available to the user and a short description of each:

- PNTRES - prints the blocks RESD.NCASE.0.0 and RESD.NCASE.1.0; that is, it prints resonance rates by group, isotope, and region.
- PXSBRN - prints fewgroup microscopic cross sections by region and buckling number as found in the block BURN.NCASE.NBS.NRG
- PRRBRN - prints fewgroup reaction rates as calculated from the block BURN.NCASE.NBS.NRG

- PUTMAT - prints materials, isotopes, temperatures and concentrations as found in the block
COMP.NCASE.0.0
- PUTOUT - prints geometry information as found in
GEOM.NCASE.0.0
- PUTZON - prints the zone block ZONE.NCASE.0.0
- PTM - prints the transfer matrix in matrix form using the blocks TMAT.NCASE.NEB.NZN
- FLXPNT - prints the region and subregion flux by group. Uses the blocks FLUX.NCASE.0.0 and FLUX.NCASE.1.0
- PNTQCL - prints slowing-down density by cell as found in the block QCEL.NCASE.0.0
- PNTJ - prints ingoing and outgoing currents from the records JIN.NCASE.0.0 and JOUT.NCASE.0.0

3.7 Subroutine NAME

Given the number of a cell, region, material or isotope, this routine finds the name in the NAME block and returns it to the calling subroutine. The arguments to subroutine are as follows:

- N1 - Kind of name
 - 1 - cell names (two 8-byte names)
 - 2 - region name
 - 3 - material name
 - 4 - isotope name
- N2 - Absolute number of cell, region, material or isotope
- NAME1 - Returned name (8-bytes)
- NAME2 - Returned name (8-bytes)
- IERR - Error flag

4. User Information

4.1 Input

In addition to the records needed for printing or dumping, RAPID needs only the following input record:

Name: 'INPUT'. 'RAPID'

LIST: TDSN,DLAT,VLAT,NCASE(NBLØK(M),M=1,17)

TDSN - Temporary data set name = 'TEMP'

DLAT - Lattice name

VLAT - Lattice version

NCASE - Case number

NBLØK(M) - Print option flag

NBLØK = 0 - No dump or print

= 1 - Neat print if available

= 2 - Dump

<u>M</u>	<u>NAME ØF BLØCK</u>
1	NAME.NCASE.0 .0
2	CØMP.NCASE.0 .0
3	GEØT.NCASE.0 .0
4	GEØM.NCASE.0 .0
5	XSEC.NCASE.NEB.0
6	SMAT.NCASE.NEB.0
7	PMAT.NCASE.NEB.0
8	CHI .NCASE.0 .0
9	HØMX.NCASE.0 .0
10	BURN.NCASE.NBS.NRG
11	RESN.NCASE.0 .0
	RESN.NCASE.1 .0
12	TMAT.NCASE.NEB.NZN
13	ZØNE.NCASE.0 .0
14	FLUX.NCASE.0 .0
15	FLUX.NCASE.1 .0
16	QCEL.NCASE.0 .0
17	JIN .NCASE.0 .0
	JØUT.NCASE.0 .0

4.2 Core Size and Timing

The RAPID module requires 160K of core, 0 job data set blocks, and should be executed with GØTIME = 1 minute.

Work Done By

C. E. Henderson
H. C. Honeck
M. B. Stroud

Report Written By

H. C. Honeck
M. B. Stroud

Contents

	<u>Page</u>
1. Theory	G.2.3
1.1 Introduction	G.2.3
1.2 Notation	G.2.4
1.3 Leakage Correction	G.2.6
1.4 Reaction Rates and Average Cross Sections	G.2.7
1.5 Average Scattering Matrices	G.2.9
1.6 Diffusion Coefficients	G.2.9
1.7 Special Isotope Edits	G.2.10
2. Program Logic	G.2.11
2.1 Interfaces	G.2.11
2.2 Calculation Sequence	G.2.12
3. Coding Details	G.2.13
3.1 Module Structure	G.2.13
3.2 Main Program RABED	G.2.14
3.3 COMMON/DATA/	G.2.16
3.4 COMMON/SPECS/	G.2.16
3.5 COMMON/GROUPS/	G.2.17
3.6 COMMON/REFPNT/	G.2.18
3.7 COMMON/TEMP/	G.2.19
3.8 Subroutine OUTSET	G.2.20
3.9 Subroutine LFGRP	G.2.22
3.10 Subroutine UPCON	G.2.23
3.11 Subroutine WRLAT	G.2.24
3.12 COMMON/TEMA/	G.2.25
3.13 Subroutine TABSET	G.2.26
3.14 Subroutine FIXFLX	G.2.26
3.15 Subroutine GCENRG	G.2.28
3.16 Subroutine ISO DAT	G.2.30
3.17 Subroutine RESRG	G.2.32
3.18 COMMON/TEMB/	G.2.33
3.19 Subroutine BALED	G.2.34
3.20 Subroutine EDIT1	G.2.36
3.21 COMMON/TEMC/	G.2.38
3.22 Subroutine BURNED	G.2.39
3.23 Subroutine MICAL	G.2.40
3.24 COMMON/TEMD/	G.2.43
4. User Information	G.2.44
4.1 Input	G.2.44
4.2 Output	G.2.45
4.3 Core Size and Timing	G.2.46
4.4 Error Stops	G.2.46

JOSHUA DPSTM-500	Title RAHAB Edit - Module RABED	Vol. IV	Page G.2.2
		Rev.	Date 6/70

(Blank)

1. Theory

1.1 Introduction

The RAHAB edit module (RABED) produces a variety of lattice physics edits. Some of these edits have no value for later computational modules and are simply printed. Other edits are primarily for later computational steps and are written on appropriate data sets. The edits can be summarized as follows:

Neutron Balance Sheet

A printed table of total source, absorption, leakage, fission, fission production, and flux by region and few group.

Lattice Physics Results

A series of records written on the LATTICE Data Set which contain the important physics results needed by other modules. These include:

1. Description of the problem input and abstracted geometry and material description.
2. Buckling search results.
3. Cell-averaged Legendre fluxes.
4. Fission rates by isotope and region.

Cross Sections

A series of records written on the FEWGRP Data Set which contain cell-averaged macroscopic and microscopic cross sections for use in later reactor modules.

Burnup Data

A series of records (BURN blocks) written on the Lattice Intermediate Data Set (LIDS) which contain microscopic reaction rates (all types) and fluxes by isotope, region, and few-group for use by burnup modules.

Although RAHAB performs only one fine-structure calculation for a given lattice, a variety of Bucklings may be specified and a leakage corrected spectrum is computed by the BLEAK module for each Buckling. The above edits are produced for each Buckling.

1.2 Notation

The following notation is used in this manuscript:

- i, j - neutron multigroup indices
- I, J - neutron few-group indices
- ϵ - symbol meaning "contained in". For example, $i \in I$ means the multigroups i contained in the few-group I .
- n - region index
- k - isotope index
- c - cell index
- ℓ - Legendre index
- B^2 - Buckling (cm^{-2})
- $\Delta E(i)$ - Width (ev) of group i
- $V(n)$ - Volume (cc) of region n
- $\phi(n, j)$ - Zero-buckling flux (neutrons/ $\text{cm}^2/\text{sec}/\text{ev}$) in region n and group j
- $\phi^*(n, j)$ - Leakage corrected flux (neutrons/ $\text{cm}^2/\text{sec}/\text{ev}$) in region n and group j .
- $Q^*(c, j)$ - Leakage corrected slowing-down density (neutrons/sec) past energy $E(j)$ in cell c .
- $\psi_\ell(j)$ - Total cell-averaged Legendre flux (neutrons/ cm^2/sec) of order ℓ in group j at a given Buckling.
- $C(k, n)$ - Concentration (atoms/barn/cm) of isotope k in region n .
- $\sigma^x(k, j)$ - Microscopic cross section (barns) of isotope k in group j . Superscript x denotes the reaction type and may be a (absorption), f (fission), $(n, 2n)$, etc.
- $\nu(k, j)$ - Neutrons/fission of isotope k in group j .
- $R^x(k, n, j)$ - Resonance reaction rate per unit slowing-down density (reactions/neutron) in region n for isotope k in group j . Superscript x may be either a for absorption or f for fission.

- $\bar{\phi}^*(n,j)$ - Leakage corrected total neutron flux (neutron-cm/sec) in region n and group j.
- $\sigma_s(k,i \rightarrow j)$ - Microscopic scattering matrix (barns/ev) for isotope k for transfers from group i to j.
- $\Sigma^x(n,j)$ - Macroscopic cross section (1/cm) in region n in group j. Superscript x denotes reaction type.
- $\Sigma_s(n,i \rightarrow j)$ - Macroscopic scattering matrix (1/cm/ev) for region n for transfers from group i to j.
- $r^x(n,J)$ - Reaction rate (events/sec) of type x in region n and few-group J.
- $r^x(k,n,J)$ - Reaction rate (events/sec) of type x for isotope k in region n and few-group J.
- $\sigma^x(k,n,J)$ - Average microscopic cross section (barns) of isotope k in region n in few-group J.
- $\sigma^x(k,J)$ - Supercell averaged microscopic cross section (barns) of isotope k in few-group J.
- $D(J)$ - Diffusion coefficient in few-group J.
- $D_o(j)$ - Supercell-averaged diffusion coefficient in group j from B_L calculation.

1.3 Leakage Correction

Each edit is associated with a given Buckling and it is necessary to compute a "leakage corrected" flux and slowing-down density corresponding to that Buckling. This is done by forcing the supercell total fine-structure flux to have the same energy dependence as the P_0 Legendre flux from the leakage calculation but not changing the spatial flux shape in each group. That is,

$$\phi^*(n,j) = \left[\psi_0(j)/\Delta E(j) \right] \left[\phi(n,j) / \sum_n v(n)\phi(n,j) \right] \quad (1)$$

Since we will be computing reaction rates, we will need the total flux,

$$\bar{\phi}^*(n,j) = \psi_0(j) \frac{v(n)\phi(n,j)}{\sum v(n)\phi(n,j)} \quad (2)$$

The corrected slowing-down density past energy $E(j)$ is then given by

$$Q^*(c,j) = \sum_{n \in c} \sum_{i < j} \bar{\phi}^*(n,i) \sum_{m \geq j} \Delta E(m) \Sigma_s(n,i \rightarrow m) \quad (3)$$

1.4 Reaction Rates and Average Cross Sections

Few-group reaction rates are computed from the leakage corrected flux and slowing-down density.

$$r^X(k,n,J) = C(k,n) \sum_{j \in J} \sigma^X(k,j) \Phi^*(n,j) \quad (4)$$

or

$$r^X(k,n,J) = \sum_{j \in J} R^X(k,n,j) Q^*(c,j) \quad , \quad nec \quad (5)$$

The total reaction rate in region n is

$$r^X(n,J) = \sum_k r^X(k,n,J) \quad (6)$$

The average few-group microscopic cross section in region n is given by

$$\sigma^X(k,n,J) = \frac{r^X(k,n,J)}{C(k,n) \sum_{j \in J} \Phi^*(n,j)} \quad (7)$$

while the supercell average value is given by

$$\sigma^X(k,J) = \frac{\sum_n r^X(k,n,J)}{\sum_n C(k,n) \sum_{j \in J} \Phi^*(n,j)} \quad (8)$$

These equations are used for all reaction types. The average number of neutrons/fission is computed from

$$\nu(k,n,J) = \sigma^P(k,n,J) / \sigma^f(k,n,J) \quad (9)$$

$$\nu(k,J) = \sigma^P(k,J) / \sigma^f(k,J) \quad (10)$$

where σ^P means $\nu \sigma^f$.

Few-group fluxes for a region and supercell are defined by

$$\Phi^*(n, J) = \sum_{j \in J} \Phi^*(n, j) \quad (11)$$

$$\Phi^*(J) = \sum_n \Phi^*(n, J) \quad (12)$$

The average concentration in the supercell is

$$C(k) = \sum_n V(n) C(k, n) / \sum_n V(n) \quad (13)$$

1.5 Average Scattering Matrices

Few-group cell-averaged scattering matrices are defined by

$$\Sigma^S(I \rightarrow J) = \frac{\sum_n \sum_{i \in I} \Phi^*(n, i) \sum_{j \in J} \Delta E(j) \Sigma^S(n, i \rightarrow j)}{\sum_n \sum_{i \in I} \Phi^*(n, i)} \quad (14)$$

The few-group removal cross section is defined by

$$\Sigma^R(I) = \sum_{J \neq I} \Sigma^S(I \rightarrow J) \quad (15)$$

1.6 Diffusion Coefficients

The diffusion coefficients are taken to be the same in each cell and computed from the supercell diffusion coefficients calculation in the B_L approximation.

$$D(J) = \sum_{j \in J} D_o(j) \Psi_o(j) / \sum_{j \in J} \Psi_o(j) \quad (16)$$

1.7 Special Isotope Edits

Some isotopes may not have data on the MULTIGRP data set but have a few cross sections on the NUCPARAM data set. These data are:

- σ_1 - 2200 m/s absorption cross section
- σ_2 - 2200 m/s fission cross section
- R_3 - absorption resonance integral
- χ_4 - ratio of fission/absorption resonance integrals
- σ_5 - average (n,2n) cross section
- σ_6 - 2200 m/s (n, γ) to first excited state cross section
- R_7 - (n, γ) to first excited state resonance integral

For each region we define the fluxes

$$\Phi_{th}^*(n,J) = \sum_{j \in J} \Phi^*(n,j) \left[\frac{2\sqrt{0.02529}}{\sqrt{E(j)} + \sqrt{E(j+1)}} \right] \quad (17)$$

$$\Phi_{epi}^*(n) = \sum_{j \geq j_1} \Phi^*(n,j) \quad , \quad (18)$$

where j_1 is the group containing $E = 0.625$ ev.

$$\Phi_{fast}^*(n) = \sum_{j \geq j_2} \Phi^*(n,j) \quad (19)$$

where j_2 is the group containing $E = 1$ Mev.

We then can compute reaction rates from either

$$r^x(k,n,J) = C(k,n) \sigma_x(k) \Phi_{th}^*(n,J) \quad (20)$$

$$r^x(k,n,J_1) = C(k,n) R_x(k) \Phi_{epi}^*(n)/16.57 \quad (21)$$

$$r^{n2n}(k,n,1) = C(k,n) \sigma_5(k) \Phi_{fast}^*(n) \quad (22)$$

where J_1 is the last epithermal few-group.

2. Program Logic

2.1 Interfaces

RABED reads the following records:

```
'INPUT'. 'RABED'
'INPUT'. 'RAHAB'. 'GEØM'. DLAT. VLAT
'FEWGRP'. VFEW. 'GRØUPS'
('TEMP'. 'LIDS'. DLAT. VLAT→V)
V. 'GRPS'. NCASE. 0. 0
V. 'SRCH'. NCASE. 0. 0
V. 'LFLX'. NCASE. NBS. 0
V. 'NAME'. NCASE. 0. 0
V. 'CØMP'. NCASE. 0. 0
V. 'GEØT'. NCASE. 0. 0
V. 'GEØM'. NCASE. 0. 0
V. 'CHI'. NCASE. 0. 0
V. 'FLUX'. NCASE. 1. 0
V. 'RESD'. NCASE. 1. 0
V. 'XSEC'. NCASE. NEB. 0
V. 'SMAT'. NCASE. NEB. 0
'ASSEMBLY'. DCEL. VCEL. 'RESUME'
```

RABED writes the following records:

```
('LATTICE'. DLAT. VLAT→L)
L. 'INPUT'
L. 'CHANGES'
L. 'FISSION'. NBS
L. 'KEFF'
L. 'LFLUX'. NBS
L. 'GEØMETRY'. DCEL. VCEL
('FEWGRP'. VFEW→F)
'FEWGRP'. 'CØNTENTS'
F. 'GRØUPS'
F. DLAT. DCEL. 'CØNTENTS'
F. DLAT. DCEL. 'MACRØ'. NFGCS
```

JOSHUA DPSTM-500	Title Module RABED	Vol. IV	Page G.2.12
		Rev.	Date 6/70

F.DLAT.DCEL.'MICRØ'.NFGCS
V.'BURN'.NCASE.NBS.NRG

2.2 Calculation Sequence

The over-all calculation sequence for RABED is shown in Figure 1.

3. Coding Details

3.1 Module Structure

The RABED module consists of a main program and five overlay segments. The overlay segments are as follows:

RABED (main program)

NAME (gets name of cell, region, isotope or material)

MANAGE (data manager; see manuscript B.1)

/DATA/ (COMMON area)

/SPECS/ (COMMON area)

/GRØUPS/ (COMMON area)

/REFPNT/ (COMMON area)

/TEMP/ (COMMON area)

OUTSET (sets up fewgroup case numbers, writes lattice and fewgroup records)

LFGRP (load fewgroup structure)

UPCØN (update fewgroup contents records)

WRLAT (write lattice records)

/TEMA/ (COMMON area)

TABSET (set up tables and initialize)

FIXFLX (correct flux, calculate slowing-down density, and write fission distribution on LATTICE library)

RESRG (get resonance rates by region)

GCØNRG (set up group constants for regions)

ISØDAT (get isotope data)

/TEMB/ (COMMON area)

BALED (produce neutron balance table)

EDIT1 (compute reaction rates, fewgroup fluxes, fewgroup scattering matrix)

/TEMC/ (COMMON area)

BURNED (produce BURN block)

MICAL (calculate microscopic fewgroup cross sections)

/TEMD/ (COMMON area)

Additional comments on the subroutines of RABED and definitions of the symbols used in the COMMON areas are provided in the following sections.

JOSHUA DPSTM-500	Title Module RABED	Vol. IV	Page G.2.14
		Rev.	Date 6/70

3.2 Main Program RABED

The main program reads the input records for this module, loads data from the LIDS, and establishes the named COMMON areas /SPECS/, /GROUPS/, and /REFPNT/. The editing overlays are then called as shown in Figure 1.

Figure 1. Main Program RABED

3.3 COMMON/DATA/

The LIDS blocks NAME, COMP, GEOM, GEOT, CHI, FLUX, RESD, XSEC, and SMAT are packed in the array DATA defined by

COMMON/DATA/DATA(25000)

Refer to Manuscript B.1 for a description of this array and its usage.

3.4 COMMON/SPECS/

The data stored in this area is read from the records 'INPUT', 'RABED' and TDSN.'LIDS'.DLAT.VLAT.'SRCH'.NCASE.O.O. The following data are included:

TDSN	-	Temporary data set name = 'TEMP'	(R*8)
DLAT	-	Lattice name	(R*8)
VLAT	-	Lattice version	(R*8)
NCASE	-	Case number	
VFINE	-	FINEGRP data set version name	(R*8)
VMULTI	-	MULTIGRP data set version name	(R*8)
VFEW	-	FEWGRP data set version name	(R*8)
LEDT(NBS,N)	-	Edit flags for buckling search number NBS	
NBSX	-	Number of buckling searches	
NBS	-	Buckling search being edited	
LSOPT(NBS)	-	Buckling search option	
		1 - k_{eff} calculation.	
		2 - Buckling search	
		3 - Time eigenvalue search	
KEFFT(NBS)	-	k_{eff} for buckling search NBS	
BSQDT(NBS)	-	Buckling (cm^{-2})	
MSQDT(NBS)	-	Migration area (cm^2)	
ALPHAT(NBS)	-	Time eigenvalue (sec^{-1})	
NFGCS(NCL,NBS)	-	Fewgroup case number for cell NCL and buckling search NBS.	

3.5 COMMON/GRUUPS/

This area contains the parameters defining and linking the multigroup and few-group structures. The following items refer to the multigroup structure:

- NEG - Number of groups.
- NFTG - First thermal group.
- JCHI - Last group where the fission spectrum is non-zero.
- IDBF - Detailed balance flag. 0 = no; 1 = yes.
- E(I) - Upper energy limit (ev) of group I.
- EBAR(I) - Characteristic energy (ev) of group I.
- DE(I) - Width (ev) of group I.

The following items refer to the few-group structure:

- NFGX - Number of groups.
- FGDE(IFG) - Width (ev) of group IFG.

The following items link the group structures:

- IFGH(IFG) - The highest multigroup in few-group IFG.
- IFGI(I) - The few-group that contains multigroup I.

JOSHUA DPSTM-500	Title Module RABED	Vol. IV	Page G.2.18
		Rev.	Date 6/70

3.6 COMMON/REFPNT/

This area contains the origins of the temporary data blocks in DATA. They are:

KGØRG - 'GEØM' or 'GEOT' origin
KCØRG - 'CØMP' origin
KFØRG - 'FLUX' origin
KRØRG - 'RESD' origin
KBØRG - 'BURN' origin
KQØRG - 'QCEL' origin
KVØRG - 'CHI' origin

3.7 COMMON/TEMP/

This area is in the root segment of the overlay and is used by each overlay segment. TABSET initializes the region tables found in this common.

- NRGX - Number of regions in supercell.
- NCLX - Number of cells in supercell.
- VRGT(NRG) - Volume of region NRG.
- NCLT(NRG) - Cell that contains region NRG.
- KMTT(NRG) - Pointer to area in COMP block describing
 the material in region NRG.
- TLPCL(NCL) - Number of lattice positions occupied by
 cell NCL.
- FS(NRG,J) - Fission source for region NRG and few-group J.
- D(J) - Diffusion coefficient for few-group J.

3.8 Subroutine ØUTSET

ØUTSET is a diver subroutine which performs the following functions by calling the appropriate subroutines:

1. Establishes the few-group structure to be used for all edits (subroutine LFGRP).
2. Update the contents records on the FEWGRP data set if FEWGRP data are to be catalogued by this edit (subroutine UPCØN).
3. If some of the contents records could not be updated because RESUME records were not given on the ASSEMBLY data set, the appropriate edit flags are reset.
4. If data are to be catalogued on the LATTICE data, the appropriate case description records and some results records are written (subroutine WRLAT).

The flow diagram for ØUTSET is shown in Figure 2.

Figure 2. Flow Diagram of OUTSET

JOSHUA DPSTM-500	Title Module RABED	Vol.	Page
		IV	G.2.22
		Rev.	Date
			6/70

3.9 Subroutine LFGRP

An attempt is made to read the record

'FEWGRP'.VFEW.'GRØUPS'

If the record is not found, a 4 group structure with breakpoints at the multigroup boundaries nearest 0.8285 Mev, 5.5308 kev, and 0.6325 ev is established and written.

The linkages between the multigroup and few-group structures (arrays IFGH and IFGI in CØMMØN/GRØUPS/) are then established.

3.10 Subroutine UPCØN

The UPCØN subroutine updates the records

'FEWGRP'.VFEW.'CØNTENTS'

'FEWGRP'.VFEW.DLAT.DCEL.'CØNTENTS'

for the cell whose name and version are DCEL and VCEL.
An attempt is first made to read the record

'ASSEMBLY'.DCEL.VCEL.'RESUME'

If the record is found, the contents records are updated by adding a new case for each buckling search for which a FEWGRP edit was requested. The resulting case numbers are stored in the NFGCS array in CØMMØN/SPECS/ for use by other routines. If the resume record cannot be found, the case numbers are set to zero and no updating of the contents record occurs.

3.11 Subroutine WRLAT

Subroutine WRLAT writes several records on the LATTICE data set. These records and the source of data are:

'LATTICE'.DLAT.VLAT.'INPUT'

'LATTICE'.DLAT.VLAT.'CHANGES'

These records contain selected data from the record

'INPUT'. 'RAHAB'. 'GEØM'.DLAT.VLAT

'LATTICE'.DLAT.VLAT.'KEFF'

A renaming of the record

'TEMP'. 'LIDS'.DLAT.VLAT.'SRCH'.NCASE.O.O

'LATTICE'.DLAT.VLAT.'LFLUX'.NSØ

Selected data from the record

'TEMP'. 'LIDS'.DLAT.VLAT.'LFLX'.NCASE.NSØ.O

'LATTICE'.DLAT.VLAT.'GEØMETRY'.DCEL.VCEL

Selected data from the records

'TEMP'. 'LIDS'.DLAT.VLAT.'GEØM'.NCASE.O.O

'TEMP'. 'LIDS'.DLAT.VLAT.'CØMP'.NCASE.O.O

3.12 COMMON/TEMA/

This area is used by the routines in the overlay segment ØUTSET as temporary storage for large arrays. The following symbols are used:

- VER(N) - Few group version of entry N (R*8) in 'FEWGRP'. 'CØNTENTS' record
- DLAT2(N) - Lattice name of entry N in few-group (R*8) CØNTENTS record
- VLAT2(N) - Lattice version of entry N in few-group (R*8) CØNTENTS record
- DCEL2(N) - Cell name of entry N in few-group (R*8) CØNTENTS records
- VCEL2(N) - Cell version of entry N in few-group (R*8) CØNTENTS records
- IASTY(N) - Subassembly type of entry N
- AA(I) - Case descriptors in assembly RESUME record
- AA(I,N) - Case descriptors for entry N in few-group CØNTENTS record
- NCS(N) - Few-group case number for entry N in CØNTENTS record

3.13 Subroutine TABSET

The TABSET routine extracts frequently used data from the GEOM block and stores it in COMMON/TEMP/.

3.14 Subroutine FIXFLX

This routine corrects the flux and slowing-down density for leakage according to equations (1) and (3), computes a few-group fission source by region, FS(NRG,JFG), and stores it in COMMON/TEMP/. The results are normalized to a unit fission source in the supercell.

Cell-averaged diffusion coefficients are computed according to equation (16) and stored in the array D(NCL) in COMMON/TEMP/.

If the edit option LEDT(NBS,2) is on, the record

 'LATTICE'.DLAT.VLAT.'FISSION'.NSØ

is written.

A flow diagram for FIXFLX is shown in Figure 3.

Figure 3. Flow Diagram of FIXFLX

3.15 Subroutine GCØNRG

This routine is used by FIXFLX to obtain the following group constants by region for one energy group.

- HP(NRG) - Scattering source into the group.
- XABS(NRG) - Absorption cross section.
- XREM(NRG) - Removal cross section.
- NFX - Index used below to denot an isotope/region pair.
- NIST(NFX) - Isotope number.
- NRGT(NFX) - Region number.
- VNT(NFX) - ν (neutrons/fission)
- XFT(NFX) - Smooth fission cross section.

The data are passed to FIXFLX through CØMMØN/TEMB/. A flow diagram for GCØNG is shown in Figure 4.

Figure 4. Flow Diagram of GCØNRG

3.16 Subroutine ISØDAT

Subroutine ISØDAT is called with the statement

CALL ISØDAT(NIS,J,IL,IH,S,XS,KXØRG,KSØRG)

- NIS - isotope number
- J - final group number
- IL,IH - low and high initial group numbers
- S(I) - scattering matrix column, $\sigma^S(I \rightarrow J) * DE(I)$,
I=IL,IH.
- XS(N) - cross section list
 - N=1, Total
 - =2, Absorption
 - =3, Removal (excluding absorption)
 - =4, Transport (including absorption)
 - =5, Fission
 - =6, ν (neutrons/fission)
- KXØRG - origin (first word minus one) of the XSEC
block in the DATA array for the current
energy band
- KSØRG - origin (first word minus one) of the SMAT
block in the DATA array for the current
energy band

Cross sections from the MULTIGRP data are stored in the XSEC
block by data type index, NDT.

<u>NDT</u>	<u>Type</u>
1	Total, $\sigma^t(J)$
2	Capture = $(n,\nu)+(n,p)+(n,d)+etc., \sigma^c(J)$
3	Fission, $\sigma^f(J)$
4	$\nu(J)$
5	Transport (excluding absorption, including n2n), $\sigma^{tr}(J)$
6	$(n,2n), \sigma^{n2n}(J)$

The scattering matrices in the SMAT block are defined by

$$\sigma^s(I \rightarrow J) = \sigma^{el}(I \rightarrow J) + \sigma^{in}(I \rightarrow J) + 2\sigma^{n2n}(I \rightarrow J)$$

where "el" and "in" denote elastic and inelastic. The cross section $\sigma^s(J)$ associated with the matrix is also stored in the SMAT block. These data are moved into the output arrays as follows:

$$\begin{aligned} S(I) &= \sigma^s(I \rightarrow J) \Delta E(I) \quad I \neq J \\ S(J) &= \sigma^s(J \rightarrow J) \Delta E(J) + \sigma^{tr}(J) - \sigma^s(J) \\ XS(1) &= \sigma^t(J) \\ XS(2) &= \sigma^c(J) + \sigma^f(J) - \sigma^{n2n}(J) \\ XS(3) &= \sigma^s(J) - \sigma^s(J \rightarrow J) \Delta E(J) \\ XS(4) &= [\sigma^{tr}(J) - 2\sigma^{n2n}(J)] + [\sigma^c(J) + \sigma^f(J) + \sigma^{n2n}(J)] \\ XS(5) &= \sigma^f(J) \\ XS(6) &= v(J) \end{aligned}$$

3.17 Subroutine RESRG

RESRG extracts the resonance rates for one group from the RESD block and stores them in the tables NIST, NRGT, RAT, RFT in COMMON/TEMB/.

3.18 COMMON/TEMB/

This area is used by the overlay segment FIXFLX as temporary storage. The following items are stored there:

- NFX - Number of fissionable isotope/region pairs.
- NFXX - Maximum number of fissionable isotope/region pairs.
- NIST(N) - Isotope number of entry N in fission production table.
- NRGT(N) - Region number of entry N in fission production table.
- RAT(N) - Resonance absorption rate for entry N.
- RFT(N) - Resonance fission rate for entry N.
- XFT(N) - Macroscopic fission cross section of entry N.
- FISS(N) - Fission for entry N.
- PRØD(N) - Neutron production for entry N.
- VNT(N) - nu, neutrons/fission for entry N.
- XREM(NRG) - Macroscopic removal cross section for region NRG.
- HP(NRG) - Scattering source into the region NRG.
- XABS(NRG) - Macroscopic absorption cross section for region NRG.

3.19 Subroutine BALED

For each cell this overlay computes the following total reaction rates which are printed when the edit option LEDT(NBS,8) is non-zero.

	<u>REGION</u>	<u>CELL</u>	<u>SUPERCELL</u>
Source	C(1,NRG,JFG)	C1(1,JFG)	C2(1,JFG)
Removal	C(2,NRG,JFG)	C1(2,JFG)	C2(2,JFG)
Smooth Absorption	C(3,NRG,JFG)	C1(3,JFG)	C2(3,JFG)
Resonance Absorption	C(4,NRG,JFG)	C1(4,JFG)	C2(4,JFG)
Leakage	C(5,NEG,JFG)	C1(5,JFG)	C2(5,JFG)
Smooth Fission	C(6,NRG,JFG)	C1(6,JFG)	C2(6,JFG)
Resonance Fission	C(7,NRG,JFG)	C1(7,JFG)	C2(7,JFG)
Nu*Fission	C(8,NRG,JFG)	C1(8,JFG)	C2(8,JFG)
Flux*Volume	C(9,NRG,JFG)	C1(9,JFG)	C2(9,JFG)
Volume	C(10,NRG,JFG)	C1(10,JFG)	C2(10,JFG)

The subroutine also computes the cell total fission sources, FI(JFG), by few-group and the scattering matrix, T1(IFG→JFG). It calculates these cell-averaged few-group cross sections:

$$ABS = \text{absorption} = (C1(3,JFG) + C1(4,JFG))/C1(9,JFG)$$

$$FISS = \text{fission} = (C1(6,JFG) + C1(7,JFG))/C1(9,JFG)$$

$$PROD = \text{production} = C1(8,JFG)/C1(9,JFG)$$

D = diffusion coefficient

If the edit option LEDT(NBS,3) is non-zero, the data are written for each cell in the records:

```
'FEWGRP'.VFEW.DLAT.DCEL.'MACRØ'.NFGCS(NCL,NBS)
```

The items generated here are stored in COMMON.

A flow diagram of BALED is shown in Figure 5.

Figure 5. Diagram of BALED

3.20 Subroutine EDIT1

The following reaction rates are computed by few-group (JFG) and reaction type (NXST):

NXST Description

- 1 Total
- 2 Smooth absorption
- 3 Removal
- 4 Transport
- 5 Smooth fission
- 6 Nu* fission

These few-group resonance reaction rates are computed:

- 1 Resonance absorption
- 2 Resonance fission
- 3 Resonance production

In addition the few-group removal cross section, flux-FLX(JFG), the volume of the region, and concentration of the isotope are calculated or recalled from the DATA block.

This data is transmitted from the subroutine through COMMON/TEMPC/.

The argument list contains NRG (region) and MIF (isotope number or if zero, material). A diagram is shown in Figure 6.

Figure 6. Diagram of EDIT1

3.21 COMMON/TEMC/

This area is used by overlay segment BALED to pass results for one region between EDIT1 and BALED and hold cell and supercell totals. The items found in this area are:

- R(N,J) - reaction rate N for few-group J
 - N = 1, Total
 - = 2, Smooth absorption
 - = 3, Removal
 - = 4, Transport
 - = 5, Smooth fission
 - = 6, Nu* smooth fission
- T(I,J) - total scattering from few-group I to
few-group J
- FLX(I) - flux for few-group I
- RK(N,J) - resonance reaction rates by type N and
few-group J
 - N = 1, Absorption
 - = 2, Fission
 - = 3, Nu* Fission
- CØNC - isotopic concentration (atoms/barn-cm)
- Tl(I,J) - total scattering from few-group I to
few-group J for a cell
- Dl(N,J) - few-group cross sections for a cell
 - N = 1, D
 - N = 2, Absorption
 - N = 3, Fission
 - N = 4, Production
- C(N,IRG,J) - reaction rate for region IRG, few-group J,
type N.
- C1(N,J) - cell totals for few-group J, type N
- C2(N,J) - supercell totals for few-group J, type N

The last three arrays are defined in the description of the subroutine BALED.

3.22 Subroutine BURNED

This overlay computes the microscopic reaction rates and fluxes by isotope, region, and few-group. The information is then stored in the BURN blocks for use by the burnup modules. There will be a BURN block for each NCASE/NBS/NRG (case, buckling search, region) combination.

The following reaction rates are computed for each region.

Smooth capture	C(1,JFG)
Resonance capture	C(2,JFG)
Smooth fission	C(3,JFG)
Resonance fission	C(4,JFG)
Nu	C(5,JFG)
Resonance absorption shielding factor.	C(6,JFG)
n, α	C(7,JFG)
n,p	C(8,JFG)
n, γ to first excited state	C(9,JFG)
n,2n	C(10,JFG)

It then calls a subroutine MICAL, which generates the few-group record, MICRØ.

The flow diagram of BURNED is shown in Figure 7.

3.23 Subroutine MICAL

This subroutine performs the average microscopic cross section calculations for region NRG by few-group and isotope for each cell/buckling search combination for all reaction types (equation 7 in section 1.4).

The results are written in the following records:

'FEWGRP'.VFEW.DLAT.DCEL.'MICRØ'.NFGCS(NCL,NBS)

The flow diagram for MICAL is shown in Figure 8.

Figure 7. Diagram of BURNED

Figure 8. Flow Diagram of MICAL

3.24 COMMON/TEMD/

The subroutine MICAL of overlay segment BURNED uses /TEMD/ as temporary storage. The items stored here are as follows:

- SIGMA(M,J,N) - microscopic cross sections of type M
for few-group J and entry N in table
- FLX(J) - flux for few-group J
- SWGT(J,NRG) - accumulation of flux*conc for
few-group J, region NRG
- DISØ(N) - isotope name of entry N (R*8)
- CØNC(N) - concentration of entry N
- ITDC(N) - temperature (°C) of entry N

4. User Information

4.1 Input

RABED reads the following blocks from the Lattice Intermediate Data Set (LIDS):

NAME	GEOM	CHI	RESD	SMAT	SRCH
COMP	GEOT	FLUX	XSEC	GRPS	LFLX

The record 'FEWGRP'.VFEW.'GROUPS' is read to establish the edit group structure. If data are to be catalogued on the FEWGRP data set, the records 'ASSEMBLY'.DCEL.VCEL.'RESUME' are required.

The control record to RABED is:

Name: 'INPUT'. 'RABED'

List: TDSN,DLAT,VLAT,NCASE,VFINE,VMULTI,VFEW,
((LEDT(M,N),M=1,7),N=1,8)

TDSN	-	Temporary data set = 'TEMP'	(R*8)
DLAT	-	Lattice name	(R*8)
VLAT	-	Lattice version	(R*8)
NCASE	-	Case number	
VFINE	-	FINEGRP or NUCPARAM version	(R*8)
VMULTI	-	MULTIGRP version	(R*8)
VFEW	-	FEWGRP version	(R*8)
LEDT(M,N)	-	Edit test (0 - no, 1 - yes) for buckling search M.	

N = 1, write LATTICE LFLUX records?
 = 2, write LATTICE FISSION records?
 = 3, write FEWGRP MACRO records?
 = 4, write FEWGRP MICRO records?
 = 5, write FEWGRP DELAYED records?
 = 6, write FEWGRP ANISOT records?
 = 7, write LIDS BURN records?
 = 8, print neutron balance sheet?

4.2 Output

The following records may be written:

('LATTICE'.DLAT.VLAT→L)

L. 'INPUT'
L. 'CHANGES'
L. 'FISSION'.NBS
L. 'KEFF'
L. 'LFLUX'.NBS
L. 'GEOMETRY'.DCEL.VCEL

('FEWGRP'.VFEW→F)

'FEWGRP'. 'CONTENTS'
F. 'CONTENTS'
F. 'GROUPS'

('FEWGRP'.VFEW.DLAT.DCEL→FC)

FC. 'CONTENTS'
FC. 'MACRO'.NFGCS
FC. 'MICRO'.NFGCS
FC. 'DELAYED'.NFGCS
FC. 'ANISO'.NFGCS

('TEMP'. 'LIDS'.DLAT.VLAT→V)

V. 'BURN'.NCASE.NBS.NRG

where

NBS - Buckling search number

NRG - Region number

NFGCS - Few-group case number

JOSHUA DPSTM-500	Title Module RABED	Vol. IV	Page G.2.46
		Rev.	Date 6/70

4.3 Core Size and Timing

See Table 1.

4.4 Error Stops

The error stops are summarized in Table 2. Error warnings, which let the problem continue, are summarized in Table 3.

TABLE I

SIZE AND TIMINGS FOR RABED SUBPROGRAMS
(Size in bytes, time in CPU seconds.)

Subprogram	Size	Total	Average Time	Entries	Overlay
MAIN	2,892	.44928	.44928	1	RØØT(1)
NAME	1,248	-	-	93	RØØT(1)
MANAGE	7,280 ¹	7.55456 ²	.01019	741	RØØT(1)
/DATA/	100,000				RØØT(1)
/GRØUPS/	2,108				RØØT(1)
/TEMP/	9,608				RØØT(1)
/REFPNT/	108				RØØT(1)
/SPECS/	704				RØØT(1)
/DATCAT/	2,440				RØØT(1)
JØS	76,918 ³				RØØT(1)
LISTDB	556				A(2)
CHECK	1,650				A(2)
/SCRACH/	176				A(2)
ØUTSET	852	-	-	1	A(3)
UPCØN	2,312	.16640	.16640	1	A(3)
WRLAT	4,562				A(3)
LFGRP	2,108	.24960	.24960	1	A(3)
/TEMA/	10,060				A(3)
TABSET	886	-	-	1	A(4)
FIXFIX	3,370	.31616	.31616	1	A(5)
ISØDAT	2,002	.59904	.00102	588	A(5)
GCØNRG	1,968	1.63072	.01941	84	A(5)
RESRG	862	-	-	84	A(5)
/TEMB/	3,808				A(5)
BALED	3,952	.54912	.54912	1	A(6)
EDIT1	3,714	4.22656	.24862	17	A(6)
/TEMC/	18,404				A(6)
BURNED	5,178	1.39776	1.39776	1	A(7)
MICAL	3,278	.19968	.01175	17	A(7)
/TEMD/	16,840				A(7)
TOTALS	229,376 ⁴ (ØVLY)	17.33888		1,632	

- 1 Includes all routines of data manager except CHECK and LISTDB.
- 2 Includes time for JOSHUA Operating System and output routines.
- 3 Includes JOSHUA Operating System, buffers, and system subprograms such as \$SEGTAB, IHCSSQRT, and IHCSEXP.
- 4 Total size is the sum of the size of the root segment and size of the largest overlay segment.

Table 2. RABED Error Stops

Message/Explanation

RABED.MAIN, POINTER MISSING

One or more of the blocks NAME, COMP, GEOT, GEOM, CHI, FLUX,
XSEC, SMAT is not in RAHAB.

RABED.MAIN,GRPS RECORD MISSING

The record TDSN.DLAT.VLAT.'GRPS'.NCASE.0.0 is missing.

RABED.MAIN,CONTROL RECORD MISSING

The record 'INPUT'.'RABED' is missing.

RABED.MAIN,SEARCH OPTION RECORD MISSING

The record TDSN.DLAT.VLAT.'SRCH'.NCASE.0.0 is missing.

RABED.TABSET, TOO MANY REGIONS

The maximum number of regions is 100.

RABED.EDIT1, ISOTOPE NO. NOT IN XSEC TABLES

Self explanatory.

RABED.EDIT1, ISOTOPE NO. NOT IN SMAT TABLES

Self explanatory.

RABED.EDIT1, REGION NOT IN GEOM TABLES

Self explanatory.

RABED.EDIT1, ISOTOPE NOT IN COMP TABLES

Self explanatory.

RABED.EDIT1, INCONSISTANCY IN SMAT TABLES

Calculated origin exceeds the table.

RABED.GCONRG, FISSION TABLES OVERFLOW

Self explanatory.

RABED.RESRG, FISSION TABLES OVERFLOW

Self explanatory.

~~RABED. BURNED~~

BURNED.BRNCAL, ENERGY BAND TABLE NOT SET UP PROPERLY

Self explanatory.

BURNED.BRNCAL, ISOTOPE NOT IN XSEC TABLES

Self explanatory.

BURNED.CRNCAL, DATA ARRAY OVERFLOW

Problem is too large.

RABED.MICAL, CELL NOT IN GEOMETRY TABLE

Self explanatory.

RABED.MICAL, MATERIAL NOT IN COMP TABLE

Self explanatory.

Table 2. (contd)

RABED.MICAL,ERROR IN WRITE
Error in writing 'FEWGRP'.VFEW.DLAT.DCEL.'MICRO'.NFGCS

RABED.NAME,CELL NO. NOT IN GEOM TABLES
Self explanatory.

RABED.NAME,REGION NO. NOT IN GEOM TABLES
Self explanatory.

RABED.NAME,MATERIAL NO. NOT IN COMP TABLES
Self explanatory.

RABED.NAME,ISOTOPE NO. NOT IN COMP TABLES
Self explanatory.

MANAGE,DATA ARRAY OVERFLOWING
Problem is too big for DATA(25000).

RABED.LFGRP, TØØ MANY FEW-GRØUPS
More than 10 few-groups have been specified.

RABED.LFGRP, TØØ MANY FEWGRP VERSION
Only 100 versions have been allowed for.

RABED.UPCØN, CØNTENTS RECØRD IS TØØ LARGE
One of the FEWGRP contents records has more than 100 entries.

RABED.WRLAT, INPUT.RAHAB.GEØM RECØRD MISSING
Self-explanatory.

RABED.WRLAT, LEGENDRE FLUX RECØRD MISSING
Self-explanatory.

RABED.BURNED, TØØ MANY ISØTØPES ØN NUCPARAM
100 isotopes are allowed.

RABED.BURNED, ISØTØPE NOT IN XSEC TABLES
Error in CØMP or XSEC blocks.

JOSHUA DPSTM-500	Title Module RABED	Vol. IV	Page G.2.50
		Rev.	Date 6/70

Table 3. Error Warnings

FEWGRP EDIT FOR CELL xxxx xxxx OMITTED, NO RESUME

RECORD IN ASSEMBLY DATA SET
Self-explanatory.

CAUTION, NO DATA ON NUCPARAM FOR xxxx
An isotope was requested which is neither on MULTIGRP
nor NUCPARAM data sets.

Work Done By

R. L. Currie
M. B. Stroud

Report Written By

R. L. Currie
M. B. Stroud

Contents

	<u>Page</u>
1. Theory	G.3.3
1.1 Introduction	G.3.3
1.2 Notation	G.3.4
1.3 Foils to be Edited	G.3.4
2. Program Logic	G.3.5
2.1 Interfaces	G.3.5
2.2 Calculational Sequence	G.3.5
3. Coding Details	G.3.7
3.1 Module Structure	G.3.7
3.2 Main Program FOILED	G.3.7
3.3 Subroutine ØUTSET	G.3.9
3.4 Subroutine RESING	G.3.9
3.5 Subroutine UNRESØ	G.3.9
3.6 Subroutine TUZ	G.3.9
3.7 Subroutine ØRDER	G.3.9
3.8 Subroutine UNRES	G.3.9
3.9 Subroutine ØUTPUT	G.3.9
3.10 Subroutine LFGRP	G.3.9
3.11 CØMMØN/DATA/	G.3.10
3.12 CØMMØN/SPECS/	G.3.10
3.13 CØMMØN/GROUPS/	G.3.10
3.14 CØMMØN/REFPNT/	G.3.11
3.15 CØMMØN/TUZIN/	G.3.12
3.16 CØMMØN/T/	G.3.12
3.17 CØMMØN/RESØI/	G.3.12
4. User Information	G.3.14
4.1 Input	G.3.14
4.2 Output	G.3.15
4.3 Error Stops	G.3.15
4.4 Core Size	G.3.15

JOSHUA DPSTM-500	Title	Module: FOILED	Vol.	Page.
			IV	G.3.2
			Rev.	Date
				4/71

(Blank)

1. Theory

1.1 Introduction

The FOILED module computes the product of flux and cross section as a function of subregion, few-group and buckling for up to 20 isotopes. This product is proportional to the reaction rate (and therefore the activation) of an infinitely thin foil of the given isotope.

A single input record for each RAHAB lattice name-version (DLATT.VLATT) specifies the foils (up to 20) and reaction types to be edited (capture and/or fission). FOILED computes leakage corrected subregion fluxes $\bar{\phi}^*$ from the zero-buckling subregion fluxes $\bar{\phi}$, Legendre leakage corrected fluxes ψ , and subregion volumes $V(m)$:

$$\bar{\phi}^*(m,j) = \psi_o(j) \frac{\bar{\phi}(m,j)}{\sum_m V(m) \bar{\phi}(m,j)} .$$

Thermal and smooth epithermal cross sections are available in the MULTIGRP data set. The infinitely dilute resonance integrals are computed from parameters stored in NUCPARAM. The infinitely dilute resolved resonance integrals are computed by resonance and assigned to a multigroup j . Infinitely dilute unresolved resonance integrals are computed at selected energy points and the contribution of the unresolved resonance region to multigroup j is evaluated. The total reaction rate of type x in subregion m in few-group J for foil k is then

$$R^X(m,k,J) = \sum_{j \in J} \bar{\phi}^*(m,j) \left[\sigma^X(k,j) + \frac{RI_{\infty}^X(k,j)}{\Delta u(j)} \right] .$$

The reaction rates for each foil are normalized to 1.0 at any arbitrary input subregion m^* (usually $m^*=1$, the center of the center cell) and few-group J^* (usually the thermal group). The absolute reaction rate for each foil at this normalization point is printed out so that different foils may be compared. Finally, the cadmium ratio, defined as the total reaction rate divided by the non-thermal reaction rate, is printed as a function of subregion for each isotope.

1.2 Notation

The following notation is used in this manuscript:

- i, j - Neutron multigroup indices
- J - Neutron few-group indices
- ϵ - Symbol meaning "contained in". For example, $i \in I$ means the multigroups i contained in the few-group I .
- n - Region index
- m - Subregion index
- k - Isotope or foil index
- l - Legendre index
- $\Delta u(j)$ - Lethargy width of group j
- $V(m)$ - Volume (cc) of subregion m
- $\Phi(m, j)$ - Zero-buckling flux (neutrons/cm²/sec/ev) in subregion m and group j
- $\psi_l(j)$ - Total cell-averaged Legendre flux (neutron/cm/sec) of order l in group j at a given buckling.
- $\Phi^*(m, j)$ - Leakage corrected total neutron flux (neutron/cm²/sec) in subregion m and group j
- $\sigma^x(k, j)$ - Microscopic cross section (barns) of isotope k in group j . Superscript x denotes the reaction type and may be a (n, γ) or f (fission).
- $RI_\infty^x(k, j)$ - Infinitely dilute resonance integral of isotope k in multigroup j . Superscript x denotes the reaction type and may be a (n, γ) or f (fission).

1.3 Foils to be Edited

Any isotope or element which appears in NUCPARAM and/or MULTIGRP data sets can be edited as a foil whether or not it appears in the ASSEMBLY records. The MULTIGRP and NUCPARAM versions used in FOILED are the same as those specified in the 'INPUT'. 'RAHAB'. 'SPECS'. DLAT. VLAT. record.

2. Program Logic

2.1 Interfaces

FOILED reads the following records:

```
'INPUT'. 'FOILED'
'INPUT'. 'RAHAB'. 'FOILS'. DLAT. VLAT
('TDSN'. 'LIDS'. DLAT. VLAT → V)
V. 'GRPS'. NCASE. O. O
V. 'SRCH'. NCASE. O. O
V. 'LFLX'. NCASE. NBS. O
V. 'GEOT'. NCASE. O. O
V. 'GEOM'. NCASE. O. O
V. 'FLUX'. NCASE. O. O
('MULTIGRP'. VMULTI. DISØ. ITDC → M)
M. 'XSECTION'
'NUCPARAM'. VMULTI. 'CØNTENTS'
('NUCPARAM'. VMULTI. DISØ → N)
N. 'RESØLVED'. KZAS
N. 'UNRESØLV'. KZAS
'FEWGRP'. VFER. 'GRØUPS'
```

2.2 Calculation Sequence

The sequence of operations is shown in Figure 1.

JOSHUA
DPSTM-500

Title Module: FOILED

Vol.
IV

Page
G.3.6

Rev.

Date
4/71

(Blank)

3. Coding Details

3.1 Module Structure

The FOILED module consists of a main program and five overlay segments. The overlay segments are as follows:

FOILED (main program)

MANAGE (data manager; see manuscript B.1)

OUTSET (sets up geometry)

LFGRP (sets up few-group structure)

RESING (performs infinitely dilute resolved resonance integral calculations)

UNRESØ (coordinates the unresolved resonance integral calculations)

TUZ (determines the number of energy mesh points in the unresolved region and performs the unresolved resonance integral computation at these energies)

ORDER (orders the energies for TUZ)

UNRES (calculates the infinitely dilute unresolved resonance integral for multigroup j following the same procedure used on reaction rates in the RESIN module.

ØUTPUT (writes the edits)

More comments on the subroutines are provided in the next sections.

See Figure 1 for a diagram of the Module FOILED.

3.2 Main Program FOILED

The main program reads the input records for the module, loads data from LIDS, establishes the named COMMON areas /SPECS/, /GRØUPS/, and /REFPNT/. It then leakage-corrects the subregion fluxes and temporarily stores them in 'FOILED'.I.'FLUXSR'. For each foil it convolutes the corrected subregion fluxes with the sum of smooth, resolved resonance and unresolved resonance cross sections. The resulting reaction rates are edited by few-group and printed by the overlay ØUTPUT.

Figure 1. Diagram of Module FOILED.

3.3 Subroutine ØUTSET

This overlay extracts the geometry data from the GEØT block and transmits these data to the driver program. It also calls LFGRP, which generates the few-group structure.

3.4 Subroutine RESING

RESING calculates the infinitely dilute resolved resonance integrals for each resonance, identifies each one with a particular multigroup, and passes this information to the main program. The resonance integrals are calculated as in the CREEP modules.

3.5 Subroutine UNRESØ

This subroutine coordinates the subroutines needed for the unresolved resonance calculations and reads the 'NUCPARAM'.VMULTI.DISØ.'UNRESØLV'.KZAS records.

3.6 Subroutine TUZ

TUZ evaluates the number of energy mesh points in the unresolved region and obtains an infinitely dilute unresolved resonance integral for each.

3.7 Subroutine ØRDER

This subroutine is called by TUZ to order the energy mesh points.

3.8 Subroutine UNRES

UNRES calculates the infinitely dilute unresolved resonance integral by numerically integrating as is done in the RESIN module.

3.9 Subroutine ØUTPUT

This overlay normalizes the foil activations to the specified subregion number and few-group, and computes the cadmium ratios. Then it tabulates the answers.

3.10 Subroutine LFGRP

An attempt is made to read the record

'FEWGRP'.VFEW.'GRØUPS'

If the record is not found, a four-group structure with breakpoints at the multigroup boundaries nearest 0.8285 Mev, 5.5308 kev, and 0.6325 ev is established and written.

The linkages between the multigroup and few-group structures (arrays IFGH and IFGI in COMMON/GRUUPS/) are then established.

3.11 COMMON/GRUUPS/

This area contains the parameters defining and linking the multigroup and few-group structures. The following items refer to the multigroup structure:

- NEG - Number of groups.
- NFTG - First thermal group.
- JCHI - Last group where the fission spectrum is non-zero.
- IDBF - Detailed balance flag. 0 = no; 1 = yes.
- E(I) - Upper energy limit (ev) of group I.
- EBAR(I) - Characteristic energy (ev) of group I.
- DE(I) - Width (ev) of group I.

The following items refer to the few-group structure:

- NFGX - Number of groups.
- FGDE(IFG) - Width (ev) of group IFG.

The following items link the group structures:

- IFGH(IFG) - The highest multigroup in few-group IFG.
- IFGI(I) - The few-group that contains multigroup I.

3.12 COMMON/DATA/

The LIDS blocks GEOM, GEOT, CHI, FLUX, and RESD are packed in the array DATA defined by

COMMON/DATA/DATA(25000)

Refer to Manuscript B.1 for a description of this array and its usage.

3.13 COMMON/SPECS/

The data stored in this area is read from the records 'INPUT'. 'FOILED' and TDSN. 'LIDS'. DLAT. VLAT. 'SRCH'. NCASE. O.O.

The following data are included:

TDSN	-	Temporary data set	(R*8)
DLAT	-	Lattice name	(R*8)
VLAT	-	Lattice version	(R*8)
NCASE	-	Case number	
VFINE	-	FINEGRP data set version name	(R*8)
VMULTI	-	MULTIGRP data set version name	(R*8)
VFEW	-	FEWGRP data set version name	(R*8)
LEDT(NBS,N)	-	Edit flags for buckling search number NBS.	
NBSX	-	Number of buckling searches	
NBS	-	Buckling search being edited	
LSOPT(NBS)	-	Buckling search option	
		1 - k_{eff} calculation	
		2 - Buckling search	
		3 - Time eigenvalue search	
KEFFT(NBS)	-	k_{eff} for buckling search NBS	
BSQDT(NBS)	-	Buckling (cm^{-2})	
MSQDT(NBS)	-	Migration area (cm^2)	
ALPHAT(NBS)	-	Time eigenvalue (sec^{-1})	
NFGCS(NCL,NBS)	-	Few-group case number for cell NCL and buckling search NBS.	

3.14 COMMON/REFPNT/

This area contains the origins of the temporary data blocks in DATA. They are:

KGØRG	-	'GEØM' or 'GEØT' origin
KCØRG	-	'CØMP' origin
KFØRG	-	'FLUX' origin

KFØRG - 'RESD' origin
KBØRB - 'BURN' origin
KQØRG - 'QCEL' origin
KVØRG - 'CHI' origin

3.15 CØMMØN/TUZIN/

The data stored here is generated in TUZ to be used in calculating the unresolved resonances in UNRES. The following data are included:

IRSX - Number of energy bands into which the
 unresolved region is broken.
RRA(IRS) - Infinitely dilute resonance absorption
 integral.
RRF(IRS) - Infinitely dilute resonance fission integral.
RENER(IRS) - Characteristic energy of the integral.

3.16 CØMMØN/T/

The data in this area is obtained by reading 'NUCPARAM'.
VMULTI.DISØ.'UNRESØLV'.KZAS. Refer to Volume III, Manu-
script B for a description of the arrays.

3.17 CØMMØN/RESØI/

These data contain the resolved resonance integrals.

XRFR(J) - resolved infinitely dilute fission resonance
 integral for each energy.
XRAR(J) - resolved infinitely dilute absorption
 resonance integral by energy.

JOSHUA

DPSTM-500

Title Module: FOILED

Vol. IV	Page G.3.13
Rev.	Date 4/71

(Blank)

4. User Information

4.1 Input

FOILED reads the following blocks from the Lattice Intermediate Data Set (LIDS):

GEØT SRCH FLUX GRPS LFLX

The record 'FEWGRP'.VFER.'GRØUPS' is read to establish the edit group structure.

Multigroup fission and capture (n,γ) cross sections are read from

'MULTIGRP'.VMULTI.DISØ.ITDC.'XSECTION'

Parameters from which the infinitely dilute resolved resonance integrals can be computed are read from the record

'NUCPARAM'.FVER.DISØ.'RESØLVED'.KZAS

Parameters for calculating the infinitely dilute unresolved resonance integrals come from the record

'NUCPARAM'.FVER.DISØ.'UNRESØLV'.KZAS

The control records are:

'INPUT'. 'FOILED'

List: TDSN, DLAT, VLAT, VMULTI, NCASE

Where: TDSN - Temporary data set

DLAT - Lattice name

VLAT - Lattice version

VMULTI - Multigroup version

NCASE - Case number

'INPUT'. 'RAHAB'. 'FØILS'. DLAT. VLAT

List: FVER, NØRMSR, NØRMFG, (NBSF(I), I=1,5)
NØØILS, (DISØ(I), NRTYPE(I), I=1, NØØILS)

Where: FVER = few group version

NØRMSR, NØRMFG = subregion and few-group to
which all foil activations
are normalized.

NBSF = buckling search options
0 = no foil edit on this option
1 = edit foil activation

NFØILS = number of foils to be
 edited (≤ 20)

DISØ = isotope constituting the foil

NRTYPE = reaction type to be edited

1 = capture

2 = fission

3 = both capture and fission

4.2 Output

The paper printout states the name of the foil and type of activation, (capture or fission). Then it lists the normalized activations by subregion number and few-group.

4.3 Error Stops

Error Stops are summarized in Table 1.

4.4 Core Size

The required core is 80K plus the system overhead.

TABLE I

ERROR STOPS

(The stop message is self-explanatory.)

FØILED.MAIN,CØNTRØL RECØRD MISSING

FØILED.MAIN,GRPS RECØRD MISSING

FØILED.MAIN,SRCH RECØRD MISSING

FØILED.MAIN,LFLX RECØRD MISSING

FØILED.MAIN,PØINTER MISSING

INPUT.RAHAB.FØILS.DLAT.VLAT RECØRD MISSING

MULTIGRP.VMULTI.DISØ.ITDC.XSECTION RECØRD MISSING

NUCPARAM.FVER.DISØ.RESØLVED.KZAS RECØRD MISSING

NUCPARAM.FVER.DISØ.UNRESØLV.KZAS RECØRD MISSING

Contents

- H.1 Static Lattice Physics - Module RAHAB.
- H.2 Lattice Depletion Physics - Module RAID
- H.3 Cell Response Physics - Module RESPONSE.
- H.4 Cell Synthesis Physics - Module RASCAL.

JOSHUA DPSTM-500	Title Part H - Control Modules	Vol. IV	Page. H.O.2
		Rev. 1	Date 8/71

(Blank)

Work Done By

C. E. Henderson
H. C. Honeck

Report Written By

C. E. Henderson

Contents

	<u>Page</u>
1. Theory	H.1.3
1.1 Introduction	H.1.3
1.2 Theoretical Models	H.1.3
1.3 Searches	H.1.3
1.4 Multiple Cases	H.1.4
2. Logic	H.1.5
2.1 Interfaces	H.1.5
2.2 Calculation Sequence	H.1.6
3. Coding Details	H.1.7
3.1 Typical Module Call	H.1.7
3.2 Error Diagnostics	H.1.7
4. User Information	H.1.8
4.1 Input	H.1.8
4.2 Output	H.1.12
4.3 Core Size and Timings	H.1.12

JOSHUA

DPSTM-500

Title Module RAHAB.

Vol.
IV

Page
H.1.2

Rev.

Date
6/70

(Blank)

1. Theory

1.1 Introduction

For the point of view of a user, the RAHAB module performs a static lattice physics calculation using fuel, target, and control assemblies catalogued on the ASSEMBLY data set, and a brief description of how these assemblies are to be put together into an infinite lattice. The results of the calculation are integral parameters and fluxes catalogued on the LATTICE data set, and few-group cross sections catalogued on the FEWGRP data set.

From the programmer's point of view, RAHAB is a control module which establishes data set names and input records for other modules, and executes these other modules. The specific modules to be executed and the order of execution depend on the user specification for

1. theoretical model to be used,
2. searches to be performed,
3. multiple cases.

1.2 Theoretical Models

The user may select one of several theoretical models for calculating the detailed flux in a lattice, namely

1. Integral Transport Method (implemented).
2. Monte Carlo Method (under development).
3. Discrete Ordinates Method (not planned).

All of these methods employ the multi-group approximation and the user can select from a variety of multi-group structures and sets.

1.3 Searches

The user may elect to perform a search on:

1. Critical Buckling (implemented).
2. Concentration (planned).
3. Dimensions (not planned).

1.4 Multiple Cases

The user may elect to perform:

1. Up to 7 independent cases (implemented).
2. Several dependent cases to obtain quantities
like temperature or density coefficients (planned).

2. Logic

2.1 Interfaces

The RAHAB module reads the following records:

'INPUT'.'RAHAB'.'CASES'.'\$JØB'
'INPUT'.'RAHAB'.'CØNTRØL'
'INPUT'.'RAHAB'.'SPECS'.DLAT.VLAT
'INPUT'.'RAHAB'.'GEØM'.DLAT.VLAT
TDSN.'LIDS'.DLAT.VLAT.'ZØNE'

The RAHAB module writes the following records:

'INPUT'.'ASP'
'INPUT'.'LAGER'
'INPUT'.'CREEP'
'INPUT'.'RESIN'
'INPUT'.'CRISP'
'INPUT'.'TRAMP'
'INPUT'.'FIST'
'INPUT'.'PASTE'
'INPUT'.'BLEAK'
'INPUT'.'RABED'
'INPUT'.'RAPID'

JOSHUA DPSTM-500	Title Module RAHAB	Vol. IV	Page H.1.6
		Rev.	Date 6/70

2.2 Calculation Sequence

RAHAB modules are executed in the following order:

- 1 - ASP - Assembly processor.
- 2 - LAGER - Lattice geometry processor.
- 3 - CREEP - Resonance integral calculation.
- 4 - RESIN - Resonance data input.
- 5 - CRISP - Cross section and scattering matrix processor.
- 6 - TRAMP - Transport matrix processor.
- 7 - FIST - Fine-structure calculation.
- 8 - PASTE - Supercell homogenization.
- 9 - BLEAK - B_L leakage calculation.
- 10 - RABED - Edit program.
- 11 - RAPID - Intermediate data set print.

3. Coding Details

3.1 Typical Module Call

A large part of the RAHAB control module consists of calls to the modules in the path. Each call consists of the following steps:

1. Test execution option to see if the module is to be executed.
2. Set error flag.
3. Write input record of the module on the LIDS.
4. Execute the module, branching to an intermediate data set print if a user error occurs.

3.2 Error Diagnostics

In the event of a user error in one of the modules executed by RAHAB, control is transferred to the intermediate data set print, module RAPID. The record 'INPUT'. 'RAPID' is filled out so that all LIDS blocks written or used by the module are printed or dumped. The blocks to be dumped or printed for each module is shown below:

		(0 - no print, 1 - neat print, 2 - dump)															
	Card & Name	COMP	GEOT	GEOM	XSEC	SMAT	PMAT	CHI	HOMX	BURN	RESD	TMAT	ZONE	SUBR. FLUX	REG. FLUX	QCEL	JIN & JOUT
ASP	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LAGER	1	1	1	2	0	0	0	0	0	0	0	0	1	0	0	0	0
CREEP	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RESIN	1	1	1	2	0	0	0	0	0	0	1	0	0	0	0	0	0
CRISP	1	1	1	2	2	2	0	2	0	0	0	0	0	0	0	0	0
TRAMP	0	1	1	0	2	2	0	0	0	0	0	2	1	0	0	0	0
FIST	0	1	1	2	2	2	0	2	0	0	1	2	1	1	2	1	1
PASTE	0	1	1	2	2	2	2	2	2	0	1	0	0	0	0	0	0
BLEAK	0	0	0	0	0	0	2	0	2	0	0	0	0	0	0	0	0
RABED	1	1	1	2	2	2	0	0	0	1	1	0	1	1	2	2	0

4. User Information

4.1 Input

In order to execute RAHAB, certain input records must be filled out by the user. A description of these records follows:

1. Name: 'INPUT'. 'RAHAB'. 'CASES'. \$JOB

List: NCSX, (DLAT(N), VLAT(N), N=1,8)

NCSX - Number of cases.

DLAT(N) - Lattice name of Nth case.

VLAT(N) - Lattice version of Nth case.

In the name string, \$JOB is a JOSHUA system variable corresponding to the user's job name.

2. Name: 'INPUT'. 'RAHAB'. 'CONTROL'

List: TDSN, (LEXEC(M), LPRNT(M), M=1,10),
(NBLØK(1,K), K=1,17)

TDSN - Temporary data set name (SCRACH or TEMP)

LEXEC(M) - Execution flag for module M.

1 - execute module M

0 - do not execute module M

M = 1	ASP	= 6	TRAMP
= 2	LAGER	= 7	FIST
= 3	CREEP	= 8	PASTE
= 4	RESIN	= 9	BLEAK
= 5	CRISP	=10	RABED

LPRNT(M) - print flag for module M

0 - no print

1 - print maximum number of words used in data array in module M

NBLØK(1,M) - intermediate data set print flag for records generated by RAHAB
(0 - no print, 1 - neat print, 2 - dump)

<u>K</u>	<u>Block</u>	<u>K</u>	<u>Block</u>	<u>K</u>	<u>Block</u>
1	'NAME' 'CARD'	7	'PMAT'	13	'ZØNE'
2	'CØMP'	8	'CHI'	14	'FLUX' (sub-region)
3	'GEØT'	9	'HØMX'	15	'FLUX' (region)
4	'GEØM'	10	'BURN'	16	'QCEL'
5	'XSEC'	11	'RESD'	17	'JIN'
6	'SMAT'	12	'TMAT'		'JØUT'

See Manuscript B.2 for a description of these records.

If this record is not filled out by the user, the temporary data set name defaults to 'SCRACH' and none of the RAHAB records are saved, all modules are executed, and there is no intermediate data set print.

For each lattice given in the CASES record, we have the following records.

3. Name: 'INPUT'. 'RAHAB'. 'SPECS'. DLAT. VLAT

List: XVER, VFIN, VFEW, (LSOPT(M), KEFFT(M),
BSQDT(M), ALPHAT(M), M=1,5), ((LEDT(M,K),
M=1,7), K=1,8), ANEWD, TPFIT

XVER - Cross section data set version.
Defaults to 'STANDARD'.

VFIN - NUCPARAM data set version.
Defaults to 'ØPENHAMR'

VFEW - Few-group data set version.
Defaults to standard 4-group
version 'STD4GRP'.

LSØPT(M) - Type of eigenvalue search desired

- 1 - k_{eff} calculation, enter
BSQD and ALPHA.
- 2 - Buckling (B^2) search, enter
KEFF and ALPHA.
- 3 - Time eigenvalue (α) search,
enter KEFF and ALPHA.

KEFFT(M) - k_{eff}

BSQDT(M) - Buckling (B^2)

ALPHAT(M) - Time eigenvalue (α).

LEDT(NBS,K) - Edit flag for buckling search NBS.
(0 - no, 1 - yes).

K=1 Create LATTICE LFLUX record.

=2 Create LATTICE FISSIØN record.

=3 Create FEWGRP MACRØ record.

=4 Create FEWGRP MICRØ record.

=5 Create FEWGRP DELAYED record.

=6 Create FEWGRP ANISØ record.

=7 Create LIDS BURN blocks.

=8 Write neutron balance

ANEWD - Factor by which the number of NIT mesh
points is reduced. Defaults to 2.00.

TPFIT - Interpolation option for collision
probabilities. Defaults to 'YES'.

The last two parameters are used by the CREEP
module. If the user does not create this record,
the defaults described above are used. The user
gets a critical buckling search and a neutron
balance sheet.

4. Name: 'INPUT'.'RAHAB'.'GEOM'.DLAT.VLAT

List: PATRN,(DCEL(N),VCEL(N),N=1,7),
(NCLA(N),RGNEW(N),(RGOLD(K,N),K=1,4),
N=1,8),(NCLB(N),REG(N),NSUB(N),
NAZM(N),N=1,24),(NCLC(N),ISA(N),
NZN(N),N=1,24),(IDØ(M),ICYL(M),
METH(M),DHT(M),DAT(M),M=1,8)

PATRN	-	Pattern name.	(R*8)
DCEL(N)	-	Cell name.	(R*8)
VCEL(N)	-	Cell version.	(R*8)
NCLA(N)	-	Number of the cell containing the region to be homogenized.	
RGNEW(N)	-	New name of regions to be homogenized.	(R*8)
RGOLD(N)	-	Name of regions to be homogenized.	(R*8)
NCLB(N)	-	Number of cell containing region to be subdivided.	
REG(N)	-	Region to be subdivided.	(R*8)
NSUB(N)	-	Number of radial subdivisions in region REG.	
NAZM(N)	-	Number of azimuthal subdivisions in region REG.	
NCLC(N)	-	Cell number containing sub-assembly ISA.	
ISA(N)	-	Relative sub-assembly number.	
NZN(N)	-	Zone number containing sub-assembly ISA.	
IDØ(N)	-	For each zone N, IDØ is the number of the preceding zone identical to this zone; 0 otherwise.	
ICYL(N)	-	Cylindrical cell approximation flag for zone N. (0 - no, 1 - yes).	
METH(N)	-	Collision probability generator method. 1 - Direct integration. 2 - Cosine currents.	
DHT(N)	-	H-integration mesh spacing in cm.	
DAT(N)	-	Alpha integration mesh spacing in degrees. Defaults to 10 degrees.	

JOSHUA DPSTM-500	Title Module RAHAB	Vol. IV	Page H.1.12
		Rev.	Date 6/70

4.2 Output

The only records written by RAHAB to the LIDS are the input records for the modules it executes. They are all of the form 'INPUT'. 'NAME' where NAME is the name of the module. For descriptions of the lists, see the Volume IV writeups on the individual modules of RAHAB.

4.3 Core Size and Timings

The RAHAB driver itself takes 12K of core. The driver together with each module takes a maximum of 240K of core. For a mixed lattice with 3 cells, 7 isotopes, 8 materials, 2 sub-assemblies, 33 regions, and 44 sub-regions, RAHAB takes 18 minutes CPU time and 10 job data set blocks.

Work Done By

R. J. Pryor

Report Written By

R. J. Pryor

Contents

	<u>Page</u>
1. Introduction	H.2.3
2. Interfaces	H.2.3
3. Calculation Sequence	H.2.4
4. User Information	H.2.6
4.1 Input	H.2.6
4.2 Error Messages	H.2.9
4.3 Core Size and Timing	H.2.9

JOSHUA
DPSTM-500

Title Lattice Depletion Physics - Module RAID

Vol.
IV

Page
H.2.2

Rev.

Date
8/71

(Blank)

Introduction

The purpose of the RAID module is to execute the various modules within JOSHUA needed for a lattice depletion calculation, viz., the RAHAB modules and the modules RADCØ and RADLE. Execution of a RAID job begins with RAID reading a user-supplied specification of the depletion calculation to be performed. For static RAHAB calculations, the user may elect to perform: 1) a search on critical buckling or 2) a KEFF calculation. For the depletion calculations, the user inputs the lattice or assembly power and desired depletion criteria. From the information supplied the RAID module will execute the appropriate computational modules in sequence and write appropriate input records for the modules executed.

Interfaces

The RAID module reads the following records: *

```
'INPUT'.'RAID'.'CASE'.'$JØB'
'INPUT'.'RAID'.'EDIT'.'$JØB'
'INPUT'.'RAID'.'CØNTROL'.'$JØB'
TDSN.'LIDS'.'DLAT.VLAT'.'ZØNE'
'INPUT'.'RAHAB'.'CEØM'.'DLAT.VLAT.'
'ASSEMBLY'.'DCEL.VCEL'.'RESUME'
```

The RAID module writes the following records:

```
'INPUT'.'ASP'
'INPUT'.'LAGER'
'INPUT'.'CREEP'
'INPUT'.'RESIN'
'INPUT'.'CRISP'
'INPUT'.'TRAMP'
'INPUT'.'FIST'
'INPUT'.'PASTE'
```

* A description of the record names is given in User Information section.

'INPUT'. 'BLEAK'

'INPUT'. 'RABED'

'INPUT'. 'RADCO'

'INPUT'. 'RADLE'

'ASSEMBLY'. DCEL. VCEL. 'RESUME'

Calculation Sequence

During a depletion calculation many calls are made to the RAHAB modules. These calls are made at points in the calculation where new cell parameters are needed. In a depletion calculation the following order of calls is always preserved:

1	-	ASP	-	assembly processor*
2	-	LAGER	-	lattice geometry processor*
3	-	CREEP	-	resonance integral calculation**
4	-	RESIN	-	resonance data input
5	-	CRISP	-	cross section and scattering
matrix processor*
6	-	TRAMP	-	transport matrix processor
7	-	FIST	-	fine-structure calculation
8	-	PASTE	-	supercell homogenization
9	-	BLEAK	-	B _L leakage calculation
10	-	RABED	-	RAHAB edit program
11	-	RADCO	-	depletion program
12	-	RADLE	-	depletion edit program***

* Executed only once during depletion.

** Execution depends on input criteria.

*** Executes at end of depletion.

The following flow diagram shows the simplified calculational sequence of the RAID module.

User Information

1. Input

To execute a RAID job, certain input records must be completed by the user. A description of these records follows:

a) NAME: 'INPUT'.'RAID'.'CASE'.\$JOB

LIST: DLAT, VLAT, VFIN, XVER, FVER,
HEIGHT, POWER, PDCL, PVCL, BSQDT,
KEFF, ANEWD, TPFIT, DISØ, DCL, VCL,
RGN, PERCT1, PERCT2, LIFETM,
ØPT2, ØPT1, DTIME, DUNIT

\$JOB	-	JOB name
DLAT	-	lattice name
VLAT	-	lattice version
VFIN	-	version of NUCPARAM library
XVER	-	version of MULTIGRP library
FVER	-	version of FEWGRP library
POWER	-	power of cell/supercell (MW)
PDCL	-	cell name for which power is specified
PVCL	-	cell version for which power is specified
HEIGHT	-	height of cell/supercell (CM)
LIFETM	-	end of life
BSQDT	-	input buckling
KEFF	-	KEFF (a buckling search is done if KEFF≠0)

ANEWD - NIT reduction factor (CREEP)

TPFIT - collision probability option (CREEP)

DISØ - name of isotope in which concentration is to be checked for spectrum calculations

DCL - cell name where isotope is found

VCL - cell version where isotope is found

RGN - region name where isotope is found

PERCT1 - percent change of isotope before spectrum calculation is repeated

PERCT2 - percent change of isotope before CREEP calculation is repeated

OPT1 - decay option? (YES or NO)

DTIME - decay time (If YES is specified above.)

DUNIT - time units (SEC., HRS., DAYS)

OPT2 - cell calculation at end of life option? (YES or NO)

b) NAME: 'INPUT'.'RAID'.'EDIT'.\$JØB

LIST: (LPRINT(I),I=1,9), (LEDT(I),I=1,7), (LPRINT(I), I=1,9), (LFDT(I),I=1,7), (EDIT(I),I=1,10)

LPRINT(I) - Print options for the various modules at spectrum calculations (see RAHAB).

LEDT(I) - Edit options at spectrum calculations.

(0 - No; 1 - Yes)

I=1 Create LATTICE LFLUX record

=2 Create LATTICE FISSION record

- =3 Create FEWGRP MACRØ record.
- =4 Create FEWGRP MICRØ record.
- =5 Create FEWGRP DELAYED record.
- =6 Create FEWGRP ANISØ record.
- =7 Write neutron balance table.

LFPRINT(I) - Same as above, except used in end of life cell calculation (optional).

LFDI(I) - Same as above except used in end of life cell calculation (optional).

EDIT(I) - Edit times (MWD)
(special option - if EDIT(1) = -1, then EDIT(2) = edit frequency).

c) NAME: 'INPUT'. 'RAID'. 'CØNTRØL'. \$JØB*

LIST: NRGX, (CELNAM(I), CELVSN(I),

RGNAM(I), I=1, NRGX), (ISØNAM(I), I=1, 30)

NRGX - number of regions not depleted

CELNAM(I) - name of cell

CELVSN(I) - cell version

RGNAM(I) - region, not to be depleted

ISONAM(I) - name of isotope to be considered in equilibrium

d) NAME: 'INPUT'. 'RAHAB'. 'GEØM'. DLAT. VLAT
(See RAHAB)

e) NAME: 'ASSEMBLY'. DCEL. VCEL. 'RESUME'*
(See RAHAB)

The following records are not read by the RAID module but need to be completed for the RAHAB calculations:

- f) NAME: 'ASSEMBLY'.DECL.VCEL.'MATERIAL'. 'DENSITY'
 - g) 'ISOTOPES'
 - h) . . . 'GEOMETRY'.GEOMTYPE
 - i) 'INPUT'. 'RAHAB'. 'ISOTOPES'.DLAT.VLAT*
- (See RAHAB)

2. Error Messages

The following error messages are written by the RAID module.

<u>Message</u>	<u>Response</u>
a) RAID - INPUT RECORD MISSING	Complete missing input record.
b) RAID - ZONE RECORD MISSING	Problem in program; see programmer.

3. Core Size and Timing

The RAID module together with each module it executes requires at least 240K of core. Because of the use of dynamic storage allocation, the exact amount of core needed will depend on the size of the problem. The total execution time depends strongly on the depletion criteria supplied by the user. Average timings for a depletion calculation are approximately equal to (number of cell calculations) times (average time per cell calculation)**.

The number of job data blocks needed to execute RAID depends on the number of regions in the lattice and the number of edits wanted. The following relationship appears to hold over a wide range of parameters.

$$\begin{aligned}
 (\text{number of JOB data blocks}) &= .02 (\text{number of edits} \\
 &\quad \text{times (number of regions} \\
 &\quad \text{depleted)}) \\
 &+ (\text{number of job data} \\
 &\quad \text{blocks needed for one} \\
 &\quad \text{cell calculation})
 \end{aligned}$$

* This record is optional and need not be completed by the user.

** See RAHAB.

JOSHUA
DPSTM-500

Title Module: RAID

Vol.
IV

Page
H.2.10

Rev.

Date
8/71

(Blank)

Work Done By

R. J. Pryor

Report Written By

R. J. Pryor

Contents

	<u>Page</u>
1. Introduction	H.3.3
2. Interfaces	H.3.3
3. Calculational Sequence	H.3.3
4. Coding Details	H.3.4
5. User Information	H.3.4
5.1 Input	H.3.4
5.2 Core Size and Timings	H.3.5

JOSHUA

DPSTM-500

Title Module RESPONSE

Vol.

IV

Page

H.3.2

Rev.

Date

1/72

(Blank)

1. Introduction

RESPONSE is a control module designed to execute the various modules needed in the computation of response functions for individual cells. The response functions are calculated using integral transport methods and are defined simply as a characteristic reaction (or response) of a cell that has an inward-directed neutron current imposed on one of its external faces. The RESPONSE modules compute four types of response functions. The types are: CURRENT, FLUX, REACTION, and SCATTERING. JOSHUA records containing the results of these calculations are written to the CELL data set and are used later by the ASCAL modules.

2. Interfaces

The RESPONSE module reads the following records:

'INPUT'. 'RESPONSE'. 'CASES'. \$JOB

'INPUT'. 'RESPONSE'. 'SPECS'. DCEL.VCEL

The RESPONSE module writes the following records.

'INPUT'. 'RAHAB'. 'CONTROL'. \$JOB

'INPUT'. 'RAHAB'. 'CASES'. \$JOB

'INPUT'. 'RAHAB'. 'GEOM'. DCEL.VCEL

'INPUT'. 'RAHAB'. 'SPECS'. DCEL.VCEL

'INPUT'. 'RESPOND'. \$JOB

3. Calculational Sequence

The RESPONSE module executes the following modules in the sequence shown:

1	-	ASP	] RAHAB Modules
2	-	LAGER	
3	-	CREEP	
4	-	RESIN	
5	-	CRISP	
6	-	TRAMP	
7	-	FIST	
8	-	RESPOND	

4. Coding Details

RESPONSE uses the RAHAB modules to compile cross-section and geometry information, and to compute the spatial transport matrices. All input needed to run the RAHAB modules is generated by the RESPONSE control module. After RAHAB has completed its calculations, RESPONSE executes the module RESPOND. The RESPOND module solves the integral transport equations and edits the cell response functions. Errors encountered by either RAHAB or RESPOND are handled by their respective error routines.

5. User Information

5.1 Input

In order to execute RESPONSE, certain input records must be filled out by the user. A description of these records follows:

1. Name: 'INPUT'. 'RESPONSE'. 'CASES'. \$JOB

List: NCLX, (DCEL(I), VCEL(I), I=1,NCLX)

NCLX - Number of cells.

DCEL - Cell name.

VCEL - Cell version.

\$JOB - is a JOSHUA system variable corresponding to the user's job name.

2. Name: 'INPUT'. 'RESPONSE'. 'SPECS'. DCEL.VCEL

List: TDSN, XVER, VFIN, VGRPS, VSHAPE, VASY,
LPRNT, NBLOK, LPCNTR, LEXEC, ANEWD,
TPFIT, (RGNEW(N), (RGOLD(K,N), K=1,4), N=1,8),
(REG(N), NSUB(N), NAZN(N), N=1,24),
(ISA(N), NZN(N), N=1,24),
(IDØ(N), METH(N), DHT(N), DAT(N), N=1,8)

TDSN - Temporary data set name.

XVER - MULTIGRP version.

VFIN - NUCPARAM version.

VGRPS - CELL data set version.

VSHAPE - CELL shape version.

VASY - Assembly version.

LPRNT - RESPOND print option (0 - NONE; 1 - PRINT)

NBLØK - LIDS print option (0 - NONE; 1 - PRINT)

- LPRNTR - RAHAB print option (0 - NONE; 1 - PRINT)
- LEXEC - Execution option.
 - 1 - Execute both RAHAB and RESPOND
 - 2 - Execute only RAHAB.
 - 3 - Execute only RESPOND
- ANEWD - Resonance Interpolation Option (Yes or No)
- TPFIT - NIT Reduction Factor
- RGNEW(N) - New homogenized region name.
- RCOLD(N) - Old region name.
- REG(N) - Region to be subdivided.
- NSUB(N) - Number of radial subdivisions.
- NAZN(N) - Number of azimuthal subdivisions.
- ISA(N) - Subassembly number.
- NZN(N) - Associated zone number.
- IDØ(N) - Zone that is the same as previous defined zone.
- METH(N) - Method of transport calculation for zone.
 - 1 - direct integration
 - 2 - cosine currents
- DHT(N) - dh integration mesh spacing
- DAT(N) - alpha angular mesh spacing

5.2 Core Size and Timings

The RESPONSE control module itself takes 9K of core. The control module together with each computational module takes a minimum of 260K of core. Execute times depend strongly on the amount of spatial subdivision in the cell. A cell containing 100 space points takes about 75 minutes. The above timings are for a 37 group cross-section library.

Work Done By

R. J. Pryor

Report Written By

R. J. Pryor

Contents

	<u>Page</u>
1. Introduction	H.4.3
2. Interfaces	H.4.3
3. Calculational Sequence	H.4.4
4. User Information	H.4.4
4.1 Input	H.4.4
4.2 Core Size and Timings	H.4.5

JOSHUA
DPSTM-500

Title Module RASCAL

Vol.
IV

Page
H.4.2

Rev.

Date
1/72

(Blank)

1. Introduction

The RASCAL modules are members of the RESPONSE-RASCAL package of modules. Together these modules provide a means of efficiently obtaining integral transport solutions of the neutron flux over regions of space containing many hundreds of cells. The solution of the integral transport equations is obtained by splitting the total calculation into two separate, independent steps. In the first step, the integral transport equations are solved for each different cell in the lattice. The results of these calculations are the cell response functions. These functions provide detailed information about the properties of the individual cells and do not depend on surrounding or neighboring cells. The response functions are calculated by the RESPONSE modules (discussed elsewhere) and are stored as permanent data in the CELL data set. In the second step of the calculation, the individual cells are coupled together for a lattice or reactor calculation and the solution of the cell boundary-neutron current equations is obtained. The neutron current equations are derived by conserving neutrons at cell boundaries and include the cell response functions which are retrieved from the CELL data set. This second series of calculations is done by the RASCAL modules. The RASCAL module is itself the control module and it executes the various computational modules required to carry out step two. The RASCAL modules are highly versatile and provide the user with a variety of input options. For example, the user may select either a simple lattice calculation containing a few cells or a full reactor calculation containing several hundred cells; and he may do either calculation with any desired energy group structure. Furthermore, for reactor calculations, the required input is greatly simplified by the allowed use of the RM (Reactor Map) terminal function. The RASCAL modules can perform either buckling searches or k_{eff} calculations and edit cell total fluxes and reaction rates, and cell averaged cross-sections.

2. Interfaces

The RASCAL module reads the following records:

```
'INPUT'.'RASCAL'.'CASES'.$JOB  
'INPUT'.'RASCAL'.'SPECS'.DCASE
```

The RASCAL module writes the following records:

```
'INPUT'.'GAS'  
'INPUT'.'JITTER'  
'INPUT'.'BLAST'  
'INPUT'.'GIN'
```

3. Calculational Sequence

The RASCAL modules are executed in the following order:

	<u>Module</u>	<u>Purpose</u>
1 -	GAS	Lattice and reactor geometry processor.
2 -	JITTER	Current calculation.
3 -	BLAST	Cross-section generator.
4 -	GIN	Edit module.

4. User Information

4.1 Input

In order to execute RASCAL, certain input records must be completed by the user. A description of these records follows:

1. Name: 'INPUT'. 'RASCAL'. 'CASES'. \$JOB

List: NCSX, (DCASE(I), I=1, NCSX)

NCSX - Number of cases to be run.

DCASE(I) - Case name.

2. Name: 'INPUT'. 'RASCAL'. 'SPECS'. DCASE

List: DNAM, VNAME, CVER, PATTN, TDSN,

FVER, EFFG, NCLX,

(RM(NCL), DCEL(NCL), VCEL(NCL), NCL=1, NCLX),

(LEXEC(I), LPRNT(I), I=1, 4),

(LSRCH(I), EFFK(I), BSQ(I), I=1, 5),

(LEDIT(I), I=1, 50)

DNAM - Reactor or lattice name.

VNAME - Reactor or lattice version.

CVER - Version of CELL data set.

PATTN	-	Reactor level or lattice pattern name.
TDSN	-	Temporary data set name.
FVER	-	Fewgroup edit version.
EFFG	-	k_{eff} guess (defaults to 1.0)
NCLX	-	Number of different cells.
RM(NCL)	-	Reactor Map symbol.
DCEL(ML)	-	Cell name.
VCEL(NCL)	-	Cell version.
LEXEC(I)	-	Module execute options (0 - No; 1 - Yes)
LPRNT(I)	-	Module print options (0 - No; 1 - Yes)
LSRCH(I)	-	Search options: 1 - k_{eff} calculation 2 - buckling search
EFFK(I)	-	Enter k_{eff} for buckling search.
BSQ(I)	-	Enter buckling for k_{eff} calculation.
LEDIT(I)	-	Enter reactor positions (x,y) or lattice positions (I) for which edits will be given.

4.2 Core Size and Timings

Core size and timings depend strongly on the size of the problem. Below is a table of some sample calculations. These calculations used a six-group energy structure.

JOSHUA

DPSTM-500

Title Module RASCAL

Vol. IV Page H.4.6

Rev. Date 1/72

<u>Number of Lattice Positions</u>	<u>Number of Different Cells</u>	<u>Core Size (K)</u>	<u>Time in Seconds</u>
3	3	150	11
21	3	180	75
113	15	210	240
225	15	280	370
703	15	380	1800

JOSHUA
DPSTM-500

Title Lattice Depletion Modules

Vol.
IV

Page
I.O.1

Rev.

Date
5/71

Contents

I.1 Solution of Depletion Equations - Module RADCO.

JOSHUA
DPSTM-500

Title Lattice Depletion Modules

Vol.
IV

Page
I.O.2

Rev.

Date
5/71

(Blank)

Work Done By

R. J. Pryor

Report Written By

R. J. Pryor

Contents

	<u>Page</u>
A. Theory	
1. Introduction	I.1.3
2. Depletion Equations	I.1.3
3. Depletion Criteria	I.1.5
4. Defining Materials	I.1.5
B. Program Logic	
1. Interfaces	I.1.6
2. Computational Sequence	I.1.7
3. λ Matrices	I.1.9
4. Flux Normalization	I.1.9
5. Numerical Integration	I.1.10
C. Modular Structure	I.1.10
D. User Information	
1. Input	I.1.12
2. Core Size and Timing	I.1.14
3. Error Stops	I.1.14

JOSHUA
DPSTM-500

Title Module RADCO

Vol.
IV

Page
I.1.2

Rev.

Date
5/71

(Blank)

A. Theory

1. Introduction

Few-group cross sections and fluxes stored in the LIDS data set, along with depletion parameters stored in the NUCPARAM data set are read by the module RADCO and used to calculate the space- and time-dependent, isotopic composition of different materials in a cell or supercell. Each material is defined in the ASSEMBLY data set and is assigned to one or more spatial REGIONS in the cell. The computation is done on a REGION-wise basis, each spatial REGION being depleted individually and results saved for editing and future lattice calculations.

2. Depletion Equations

The concentration of each isotope in a material is governed by the following differential equation,

$$\frac{d}{dt} C_i(X,t) = -L\text{LOSS}_i(X,t) + PR\text{OD}_i(X,t) \quad (1)$$

where $C_i(X,t)$ is the concentration of the i^{th} isotope at a position X and a time t .

$L\text{LOSS}_i(X,t)$ is the rate of loss of isotope i at a position X and a time t , per unit volume.

$PR\text{OD}_i(X,t)$ is the rate of production of isotope i at a position X and a time t , per unit volume.

In a nuclear reactor the loss or production of any isotope is primarily due to nuclear reactions with neutrons and nuclear decay. The reaction rate for a particular nuclear reaction j can be written as:

$$C_i(X,T) \int \sigma_{i,j}(X,E) \phi(X,E) dE \quad (2)$$

where $\sigma_{i,j}(X,E)$ is the microscopic cross section for isotope i , j and reaction type j , and $\phi(X,E)$ is the neutron flux. In a few-group formalism equation (2) becomes

$$C_i(X,t) \sum_g \sigma_{i,j}^g(X) \phi^g(X) \quad (3)$$

where g is the few-group index. The nuclear decay reaction rate can be simply written

$$\lambda^i C_i(X,t)$$

where λ^i is the total decay constant of isotope i . Before defining the production and loss terms we define the following:

$$\lambda_{i,j}(X,t) \equiv \sum_g \sigma_{i,j}^g(X,t) \phi^g(X,t), \quad (4)$$

where $\lambda_{i,j}(X,t)$ is the "reaction time constant" for a nuclear reaction type j and isotope i . We now define the loss term for isotope i as simply the sum over j of all $\lambda_{i,j}$, plus the total decay constant for nuclear decay, times the concentration of isotope i . Thus we write,

$$LOSS_i = \left(\sum_j \lambda_{i,j} + \lambda^i \right) C_i.$$

The production term for isotope i can be similarly constructed. It is the sum of all nuclear reactions plus the sum of all possible nuclear decays leading to isotope i , from all possible k isotopes. That is,

$$PROD_i = \sum_k \left(\sum_j \lambda_{k,j} + \lambda^{k \rightarrow i} \right) C_k,$$

where $\lambda^{k \rightarrow i}$ is a partial decay constant, and the reaction type j is restricted to those reactions connecting isotopes k to isotope i . Since there is usually only one such reaction type that can couple isotopes k and i , we remove the sum over j and write the production term,

$$PROD_i = \sum_k \left(\lambda_{k,j(k \rightarrow i)} + \lambda^{k \rightarrow i} \right) C_k.$$

The depletion equations (1) are solved for each REGION in the cell. Thus, the spatial variable X in equations (1) refers to a particular region. Few-group cross sections and fluxes for that region are read from the LIDS data set (generated by RAHAB) and a set of "time constants" is constructed. Time variation of the constants is treated by fitting each one to a time dependent quadratic function. Points needed for the fit are saved from preceding RAHAB calculations. Solutions to the depletion equations are then obtained by the Runge-Kutta-Gill numerical technique.

OSHUA DPSTM-500	Title Module RADCØ	Vol. IV	Page I.1.5
		Rev.	Date 5/71

3. Depletion Criteria

RAHAB calculations are done at varying times in the depletion calculation depending on a criterion supplied by the user. The criterion allows the user to select an isotope that RADCØ will monitor during depletion and to specify a depletion percentage. When the concentration of this isotope changes by this percentage, the depletion calculation is halted and information is passed to RAID for another spectrum calculation. Control is returned to RADCØ once the spectrum calculation is completed; and the depletion is then continued until another spectrum calculation is required. The cycling between RADCØ and RAHAB is repeated until the end of life as specified by the user is reached.

4. Defining Materials

The ASSEMBLY data set is used by both RAHAB and RADCØ to define the composition of each material in an assembly. In the RAHAB calculation, the composition of each material determines the characteristics of the few-group spectrum; and in the RADCØ calculation, the composition of each material defines those isotopes that are to be treated in the depletion equations. To properly describe the depletion or the production of any isotope, a systematic isotopic composition must be built into each material. First, all isotopes necessary to accurately predict the few-group spectrum must be included in the material. Secondly, all precursors of a "viewed" isotope must be included in the material (by "viewed" we mean an isotope of interest). The first point is obvious and need not be discussed. The second point perhaps needs explanation. The depletion code is so designed that the LOSS term for each isotope is treated exactly. However, the production term, PRØD, depends on the list of isotopes included in the material; and to accurately calculate this term, every isotope that either decays to, or has a nuclear reaction that leads to a "viewed" isotope must be included. In most cases, several isotopes are needed to describe the production of any one isotope. For example, if we started with U^{235} and wished to describe the U^{238} production, we will need to include U^{236} and U^{237} in the calculation. The reason for including these isotopes is to account for the (n,v) couplings that occur. U^{236} was added to the calculation to feed U^{237} which in turn feeds U^{238} . If (3n,v) reactions were much more important than (n,v) reactions, U^{236} and U^{237} would not need to be included. However, it is well known that reactions of this type seldom occur and are certainly much less important than (n,v) reactions. It is therefore necessary to include these extra isotopes in this example.

It becomes evident that many intermediate isotopes would be needed if the "viewed" isotope is many reactions away from the initial isotope. To predict the Cf^{252} production, for

JOSHUA DPSTM-500	Title Module RADCO	Vol. IV	Page. I.1:6
		Rev.	Date 5/71

example, about 30 isotopes need to be included in the material. We can determine what isotopes need to be considered in the calculation if we remember to include all those isotopes that physically link the starting isotope with the "viewed" isotope. The linking is defined by the set of possible reaction types we are able to consider. Omission of an isotope may cause significant errors in production; but no errors can ever result from adding another isotope to the material.

B. Program Logic

1. Interfaces

RADCO reads the following records:

```
( 'TEMP'. 'LIDS'. DLAT. VLAT → V )
V. 'NAME'. NCASE. O. O
V. 'COMP'. NCASE. O. O
V. 'GEOM'. NCASE. O. O
V. 'BURN'. NCASE. NBS. NRG
'INPUT'. 'RADCO'
'INPUT'. 'RAID'. 'CONTROL'. $JOB
'NUCPARAM'. VERSN. 'DPDATA'. 'QVALUE'
'NUCPARAM'. VERSN. 'DPDATA'. 'FISSPRD'
'NUCPARAM'. VERSN. 'DPDATA'. 'XSECTION'
'NUCPARAM'. VERSN. 'DECAY'
```

RADCO writes the following records:

```
'ASSEMBLY'. DCEL. VCEL. 'MATERIAL'. 'ISOTOPES'
V. 'BURNT'. NEDIT. NBURN. NRG
V. 'COMP'. NCASE. O. O
V. 'BURN'. NCASE. NPASS. NRG
```

2. Computational Sequence

A flow diagram of the module RADCO is shown in Figure 1. The basic computational steps are:

1. Read and store NUCPARAM data.
2. Read and store NAME, COMP, CARD, GEOM and BURN blocks from LIDS data set.
3. Set up initial concentration arrays and store.
4. Normalize flux for power specified.
5. Using most recent cell calculation parameters, calculate λ matrices, store, and write them in the SCRATCH data set.
6. Attempt to read and store λ 's from the preceding two cell calculations, establish λ time dependence.
7. Start depletion in region specified by user and monitor special isotope. When the concentration of this isotope changes by the given percentage or the lifetime is reached, stop.
8. Deplete all other regions to the same point in time, writing edit information and updating COMP block in LIDS data set.
9. Write COMP block if another cell calculation is needed; or if lifetime is reached, determine and write new ASSEMBLY records.

3. Matrices

Because only certain nuclear reactions are possible or significant for a given isotope, it would be fruitless to treat every reaction type in the depletion equations. To limit the amount of core storage and calculation time, the module RADCO considers only the following reaction types:

1. (n,v)
2. (n,p)
3. (n, α)
4. (n,2n)
5. Fission
6. Fission Yield
7. Nuclear Decay

Linking between different isotopes in the material is determined from their respective ISOTOPE number difference and the seven different reaction types above. An array is determined for each isotope which includes a value for each reaction type and a pointer to a corresponding linking isotope. The depletion equations are solved by looping through the arrays, calculating production and loss contributions at each time step.

4. Flux Normalization

Few-group fluxes are normalized according to the cell/supercell power supplied by the user. Every isotope that has a fission cross section is included in the normalization calculation. Fission Q-values for each isotope are obtained from the NUCPARAM data set. A default value of 200 mev/fission is used for those isotopes not included in the Q-value record. The normalized flux is related to the unnormalized flux by the relation

$$\phi^g(n) = f \phi_R^g(n)$$

where $\phi_R^g(n)$ is the few-group flux obtained from the RAHAB calculations, n is a region number, and f is the normalization factor. The factor f is determined from the expression

$$\text{POWER} = f \sum_n \sum_g \sum_j Q(j) \phi_R^g(n) \sigma_{j,\text{fission}}^g(n) C_n(n)$$

where $Q(j)$ is the Q-value for isotope j .

5. Numerical Integration

The depletion equations are solved using a procedure designed by Gill. The procedure is a modified Runge-Kutta integration giving fourth-order accuracy and requiring a minimum amount of storage. To reduce round-off errors, all arithmetic in the depletion calculation is done in double precision. The integration step width is determined for the calculation from the relationship

$$\Delta t = \frac{.1}{|\lambda_{\max}|}$$

where $\lambda_{\max}$ is the maximum absolute value in the full λ matrix. Because some isotopes have very large time constants, a provision has been made to exclude these from the expression above. All isotopes of this category are assumed to be in equilibrium and their concentration is calculated from the expression

$$C_i(t + \Delta t) = \frac{\text{PROD}_i(t + \Delta t)}{(\text{LOSS})_i(t + \Delta t)} C_i(t)$$

In order that all points in time be treated equally and as accurately as possible, a retracing technique is used which will correct the isotopic concentrations calculated by the preceding RADCO calculations. Each time RADCO is called it reads a set of corrected concentrations determined at the time of the second preceding cell calculation and begins iterating from that point in time to the point in time where a new cell calculation is needed. During this retracing all saved concentrations are over-written with new values. This form of retracing guarantees that all concentrations are calculated with an interpolated quadratic fit of the reaction decay constants which will reduce the errors that can be contributed to the interpolation of fast varying reaction rates.

C. Modular Structure

The module RADCO consists of a main routine and a number of subroutines and COMMON areas. A brief description of these follows:

MAIN ROUTINE RADCO

This routine reads the RADCO input record and sets up a calling sequence for the different subroutines in the program. All subroutines are called from this main routine.

SUBROUTINE RLIDS

This routine reads records from the LIDS and NUCPARAM data sets. It sets up a concentration array for each region and writes the arrays into the SCRATCH data set. A table is also constructed that specifies which regions and zones are not depleted and specifies isotopes that are to be treated as in equilibrium.

SUBROUTINE FCALC

This routine calculates the relative fission power for a region. The main program calls FCALC for each region in the lattice.

SUBROUTINE FIXUP

This routine modifies the few-group cross section structure of LIDS BURN block. A (n,γ) cross section is calculated and is written over the capture cross section in the block. This is done for each region and isotope.

SUBROUTINE SETUP

This routine determines the order in which regions are depleted. It selects a region and sets up λ matrices for that region. λ matrices are read from preceding calculations and a λ time-dependence is determined.

SUBROUTINE DELT

This routine computes each time step used in the numerical integration of the depletion equations. The routine adjusts the time step to conform with edit times and the end of life.

SUBROUTINE SOLVE

This routine integrates the depletion equations for the time step specified by the DELT subroutine. SOLVE monitors the special isotope and sets a control flag when its concentration changes the prescribed maximum amount.

SUBROUTINE WLIDS1

This routine updates the COMP block with the latest concentrations calculated at the end of life. WLIDS1 is called for each region depleted.

SUBROUTINE WLIDS2

This routine writes the information that is later read by the edit module. WLIDS2 is called at each edit time during the depletion.

SUBROUTINE DECAY

This routine integrates the depletion equations with a zero power flux normalization. This routine is called only if a decay option is asked. At the end of the decay period, new ASSEMBLY records and edit records are written using the latest concentrations.

COMMON/SPECS/

This storage area is used to contain the various control flags and depletion parameters.

COMMON/DATA/

This area contains the LIDS data blocks read by subroutine RLIDS.

COMMON/DEEQ/

This area contains the λ matrices and concentration array for the region being depleted.

COMMON/QVAL/

This area contains the decay and fission yield information read from the NUCPARAM data set.

COMMON/DØIT/

This area contains the tables of regions and zones not to be depleted, and isotopes considered in equilibrium.

D. User Information

1. Input

The module RADCO is normally executed in a path by RAID; however, it can be executed alone if the LIDS records already exist and if the following record is completed by the user:

NAME: 'INPUT'..'RADCO' *

LIST: DLAT, VLAT, VFIN, XVER, FVER, ZNAM, DCEL,
VCEL, RGC, PERCT1, PERCT2, NCASE, POWER,
HEIGHT, LIFETM, DØPT, DTIME, DUNIT, PDCL,
PVCL, LPRINT, (EDIT(I), I=1,10)

* The record is normally created in the path by RAID.

DLAT	-	lattice name
VLAT	-	lattice version
VFIN	-	NUCPARAM version name
XVER	-	MULTIGRP version name
FVER	-	FEWGRP version name
ZNAM	-	special isotope name
DCEL	-	cell name where special isotope is found
VCEL	-	cell version name where special isotope is found
RGC	-	region name where special isotope is found
PERCT1	-	percent change in concentration of special isotope before spectrum calculation is repeated
PERCT2	-	percent change in concentration of special isotope before CREEP calculation is repeated
NCASE	-	case number (set equal to 1)
POWER	-	power of cell/supercell (MW)
HEIGHT	-	height of cell/supercell (CM)
LIFETM	-	lifetime of core (MWD)
DØPT	-	decay option (YES or NO)
DTIME	-	decay time
DUNIT	-	decay time units (HRS, SEC, DAYS)
PDCL	-	cell name in which power is specified (may be left blank)
PVCL	-	cell version name in which power is specified (may be left blank)
LPRINT	-	print option
	= 0	only fission power printed
	= 1	fission power and λ matrices printed
	= 2	full dump, including time step iterations

JOSHUA DPSTM-500	Title Module RADCO	Vol. IV	Page I.1.14
		Rev.	Date 5/71

EDIT - edit time (MWD)

2. Core Size and Timing

The module RADCO requires at least 240K of core and executes in less than one minute. TIMINGS for the various subroutines are given in Table I. The number of JOB data blocks needed can be determined from the relation

$$(\# \text{ JOB data blocks}) = .02 (\text{number of regions depleted}) \\ \text{times } (\text{number of edits})$$

3. Error Stops

The error stops issued by RADCO are summarized in the following table:

MESSAGE/EXPLANATION

- a) RADCO.RLIDS, LIDS NAME RECORD MISSING
Name record missing. Check lattice name and version specified.
- b) RADCO.RLIDS, LIDS COMP RECORD MISSING
(Same as above.)
- c) RADCO.RLIDS, LIDS GEOM RECORD MISSING
(Same as above.)
- d) RADCO.RLIDS, LIDS CARD RECORD MISSING
(Same as above.)
- e) RADCO.RLIDS, LIDS BURN RECORD MISSING
(Same as above.)
- f) RADCO.RLIDS, TOO MANY ISOTOPES SPECIFIED IN CHAIN
More than 70 different isotopes have been specified in a material.
- g) RADCO.RLIDS, DATA ARRAY OVERFLOW
Increase gosize and execute module again.
- h) RADCO.RLIDS, POWER IN CELLS FOUND ZERO
No fissionable isotopes were found.

- i) RADCO.RLIDS, SPECIAL ISOTOPE NOT FOUND
Special isotope was not found in cell specified.
- j) RADCO.RLIDS, DECAY RECORD MISSING
Decay record was not found. Check NUCPARAM
version name.
- k) RADCO.RLIDS, Q-VALUE RECORD MISSING
(Same as above.)
- l) RADCO.RLIDS, YIELD RECORD MISSING
(Same as above.)
- m) RADCO.RLIDS, REGION NRGD ASKED NOT TO BE DEPLETED
You have asked that the region in which the special
isotope is found not be depleted. Omit this region
in the RAID control record.

TABLE I
TIMINGS FOR RADCO SUBROUTINES*

<u>SUBROUTINE</u>	<u>TOTAL TIME</u>	<u>AVERAGE TIME</u>	<u>ENTRIES</u>
MAIN	1.79988	1.79988	1
CLEAR	0.13332	0.06666	2
RLIDS	1.01660	1.01660	1
READDB	1.23325	0.02681	46
DELEDB	0.06666	0.00196	34
FCALC	0.01667	0.00556	3
FINDDB	0.03333	0.00062	54
LENDB	0.01667	0.00238	7
FIXUP2	0.05000	0.05000	1
SETUP	5.24966	0.74995	7
DELT	0.69996	0.00132	532
SOLVE	26.51497	0.04984	532
WLIDS2	0.96660	0.02301	42
WLIDS1	0.05000	0.00714	7
DECAY	5.94962	0.84995	7
WRITDB	0.14999	0.07500	2
FIXUP3	0.05000	0.05000	1
COUNT	<u>0.01667</u>	0.01667	<u>1</u>
TOTALS	44.01385		1,280

* In seconds.

JOSHUA
DPSTM-500

Title Part J - Lattice Monte Carlo Modules

Vol.
IV

Page
J.O.1

Rev.

Date
12/70

Contents

J.1 Module STERIL.

JOSHUA DPSTM-500	Title Part J - Lattice Monte Carlo Modules	Vol. IV	Page J.O.2
		Rev.	Date 12/70

(Blank)

7

Work Done By

C. E. Henderson
H. C. Honeck

Report Written By

C. E. Henderson
H. C. Honeck

Contents

	<u>Page</u>
1. <u>Theory</u>	J.1.3
1.1 <u>Introduction</u>	J.1.3
1.1.1 Purpose	J.1.3
1.1.2 The Monte Carlo Method	J.1.4
1.1.3 Coordinate System	J.1.5
1.1.4 Random Numbers	J.1.7
1.1.5 Probability Distributions	J.1.7
1.1.6 Error Analysis	J.1.8
1.1.7 Batches and Histories	J.1.9
1.2 <u>Neutron Processes</u>	J.1.10
1.2.1 Fission Source	J.1.10
1.2.2 Neutron Weight	J.1.12
1.2.3 Collision Point	J.1.13
1.2.4 Scattering Type	J.1.14
1.2.5 Scattering Isotope	J.1.15
1.2.6 Post-scattering Energy Group	J.1.16
1.2.7 Scattering Angle	J.1.16
1.2.8 Post-scattering Direction	J.1.22
1.2.9 Resonance Treatment	J.1.22
1.2.10 Cross Section Conventions	J.1.25
1.3 <u>Edits</u>	J.1.28
1.3.1 Eigenvalue	J.1.28
1.3.2 Neutron Flux	J.1.28
1.3.3 Slowing-down Density	J.1.28
1.3.4 Spatial Moments	J.1.28

	<u>Page</u>
2. <u>Program Logic</u>	J.1.33
2.1 Interfaces	J.1.33
2.2 Over-all Structure	J.1.34
2.3 Data Preparation	J.1.36
2.4 Distribution Function for Fission Production	J.1.38
2.5 Fission Source Probabilities	J.1.42
2.6 Scattering, Elastic and Transfer Probabilities	J.1.46
2.7 Cross Sections and Resonance Absorption Rates	J.1.50
2.8 Coordinate Systems and Tracking Methods	J.1.51
2.9 Moments Calculation	J.1.53
2.10 Preparation of Output	J.1.59
3. <u>Coding Details</u>	J.1.60
3.1 Module Structure	J.1.60
3.2 Common Areas	J.1.61
3.3 Subprograms	J.1.65
4. <u>User Information</u>	J.1.70
4.1 Input	J.1.70
4.2 Output	J.1.71
4.3 Core Size and Timing	J.1.71
4.4 Error Stops	J.1.71

1. Theory

1.1 Introduction

1.1.1 Purpose

The STERIL module provides a Statistical Estimation of Reaction Rates in Lattices using the Monte Carlo method. STERIL is not a general-purpose Monte Carlo code. It was written for several specific purposes:

1. To test the adequacy and the anisotropic scattering approximation (the "transport approximation") used in the RAHAB integral transport modules.
2. To evaluate the accuracy of the current coupled zone approximation used in the RAHAB integral transport modules and provide criterion for selection of zone boundaries.
3. To provide a method for analysis of lattices whose geometry is so complex that integral transport matrices are not appropriate.
4. To provide information about anisotropic migration in a lattice.
5. To provide information about neutron angular distribution to guide future theoretical improvements in the integral transport method.
6. To provide a framework for later extensions to the resonance capture treatment.

To accomplish these purposes (in particular, the first two), it is essential that all features of the integral transport calculation (except the ones we are trying to test) be treated in an identical manner. Thus, the Monte Carlo calculation must be capable of using the same cross sections, scattering matrices, multigroup formalism, resonance treatment, (n,2n) treatment, and fission treatment that is used in the RAHAB integral transport modules. In this way we can vary one treatment at a time and evaluate its adequacy.

The advantage of a modular code system is that, if modules perform well-defined functions, they can be used in a variety of applications. STERIL makes use

JOSHUA DPSTM-500	Title Module STERIL	Vol. IV	Page J.1.4
		Rev.	Date 12/70

of all of the RAHAP modules which:

1. Prepare and organize the material and geometry data (modules ASP and LACER).
2. Provide resonance rates (modules CREEP and RESIN).
3. Organize cross sections and scattering matrices (module CRISP).
4. Estimate leakage corrections to the spectrum (modules PASTE and BLEAK).
5. Edit the results (module RABED).

STERIL then replaces only the integral transport modules TRAMP and FIST. It is essential that STERIL accept the same input records available to TRAMP and FIST and produce the same output records that are produced by FIST. The FIST output records are the flux by region and group and the slowing-down density by cell and group. These quantities are not the normal output of a Monte Carlo calculation so that the scoring techniques used in STERIL are a bit different than in many Monte Carlo codes.

Finally, we recognize that a great deal of work has been done by many researchers to improve the speed and accuracy of Monte Carlo calculations. We have adopted a few of those techniques here (those with which the author feels comfortable). The adoption of others would take longer to implement. These will be deferred to a later version because the immediate need is to evaluate the integral transport method approximations as soon as possible.

1.1.2 The Monte Carlo Method

The Monte Carlo method substitutes brute force computational power for theoretical elegance. It assumes only that we know the distribution functions for each neutron interaction process. Let us consider the history of a single fission neutron in a lattice. Several questions arise:

1. Where did the fission occur?
2. What is the fission neutron's energy?

3. In what direction is it traveling?
4. How far does it go before it collides with an atom?
5. What type of atom did it collide with?
6. Was it captured, scattered, or did it cause fission?
7. If it was scattered, what is its new energy and direction?

(Continue asking questions 4 - 7 until capture or fission occurs to end the history.)

Each of these questions pertains to an event in the neutron's history. Each event may have a variety of outcomes and the relative occurrence (distribution function) of each outcome must be known. In the Monte Carlo method a specific outcome for each event is selected from the distribution function for that event each time a question is posed. Having given a definite answer to one question, the calculation proceeds to the next question and again a specific answer (outcome) is drawn from the appropriate distribution function for that event. The process is repeated until the history is terminated.

The result is the detailed history of one fission neutron. If we repeat the process, we will follow another neutron history which will be different from the first one because we would have selected an entirely different set of outcomes for each event. Modern digital computers allow us to follow many neutron histories and in so doing compute the expected average nuclear reaction rate which we would observe in a lattice with a large neutron population.

1.1.3 Coordinate System

We assume that the lattice is uniform and infinite in the axial (z) direction. It is then convenient to choose the coordinate system shown in Figure 1.

Figure 1. Coordinate System

1.1.4 Random Numbers

The Monte Carlo method requires a set of numbers, ξ_n , randomly distributed in the interval (0, 1). The random numbers used here are calculated by the congruence method

$$\xi_{n+1} = \left[\xi_n B \right] \text{ low order 32 bits}$$

where $B = 5^{15}$ in the base and is the largest power of 5 which will fit into 32 bits. ξ_0 is the starting random number.

Obviously this set of numbers is not truly random so that it should be called a set of "pseudo-random" numbers. It is fortunate that the set is not truly random or we would never be able to reproduce a calculation.

The user can control the sequence by inputting the starting random number, ξ_0 . If the program prints out the last random number it used in the calculation, this number can then be input to resume the calculation.

1.1.5 Probability Distributions

The outcome, x , of an event is governed by a distribution function, $f(x) > 0$, $a \leq x \leq b$, for that event. If $f(x)$ is normalized,

$$\int_a^b f(x) dx = 1$$

then the function

$$P(x_1) = \int_a^{x_1} f(x) dx$$

is the probability that x will be less than x_1 . If we draw a random number, ξ , from the range (0,1), then

$$P(x) = \xi, \quad x = P^{-1}(\xi)$$

selects a value of x randomly drawn from the distribution $f(x)$.

If the distribution function consists of a discrete set of N outcomes, x_n , with probabilities, f_n , then if the f_n are normalized,

$$\sum_{n=1}^N f_n = 1$$

the probability function is

$$P_n = \sum_{m=1}^n f_m$$

If we draw a random number, ξ , from the range (0,1), then

$$P_{n-1} \leq \xi < P_n$$

selects a value of n randomly drawn from the distribution of f_n .

1.1.6 Error Analysis

Assume that x is a property to be estimated by the Monte Carlo calculation, and that x_n , $n=1, N$, are individual estimates which have weights, W_n . The estimate (or average) of x is

$$\langle x \rangle = \sum_n W_n x_n / \sum_n W_n$$

The x_n have some distribution whose weighted standard deviation is

$$\begin{aligned} \sigma &= \left\{ \sum_n W_n (x_n - \langle x \rangle)^2 / \sum_n W_n \right\}^{1/2} \\ &= \left\{ \frac{\sum_n W_n x_n^2 - 2 \langle x \rangle \sum_n W_n x_n + \langle x \rangle^2 \sum_n W_n}{\sum_n W_n} \right\}^{1/2} \\ &= \sqrt{\langle x^2 \rangle - \langle x \rangle^2} \end{aligned}$$

where

$$\langle x^2 \rangle = \frac{\sum_n W_n x_n^2}{\sum_n W_n} .$$

The probable error in $\langle x \rangle$ is

$$P.E. = 0.6745 \sigma / \sqrt{N} .$$

These quantities can be computed if we tally the running sums for

$$\sum_n 1 , \quad \sum_n W_n , \quad \sum_n W_n x_n , \quad \sum_n W_n x_n^2 .$$

Where convenient, these sums will be retained in core. In other cases, the error analysis will be done on the results of each batch so that n refers to the batch number, W_n refers to the number of histories in a batch, and x_n is the average x computed in the batch.

1.1.7 Patches and Histories

A history is all those events which occur from the time a fission neutron is selected to the time it leaves the system by leakage or absorption (see the section on neutron weights for a description of how a history is terminated). A collection of successive histories is called a batch. For example, we might elect to run 10 batches of 100 histories each.

The batch is used to provide a convenient interruption in the calculation to perform edits and auxiliary calculations. For example, the following calculations are done following the completion of a batch.

1. A new fission source distribution is computed.
2. New flux shape factors are computed to correct resonance rates.
3. Statistical estimates of the error in k_{eff} are calculated.

1.2 Neutron Processes

1.2.1 Fission Source

Four questions must be answered to select a neutron from a fission source distribution.

1. At what point in the lattice did the fission occur?
2. What isotope fissioned?
3. What is the fission neutron energy?
4. What is the direction of motion of the fission neutron?

The distribution function for fission production by sub-region and isotope is $F(n,k)$ normalized so that

$$\sum_n \sum_k F(n,k) = 1 \quad .$$

It is easier to think of this as the function, $F(\rho)$, where ρ numbers the sub-region/isotope pairs. The cumulative probability is then

$$P_f(\rho) = \sum_{\rho'=1}^{\rho} F(\rho') \quad .$$

If we draw a random number, ξ , in the range (0,1), the inequality

$$P_f(\rho-1) \leq \xi < P_f(\rho)$$

selects a value of ρ which designates an isotope and a sub-region.

Since we keep no information on the shape of the fission distribution within a sub-region, we will select a starting point (x,y) from a uniform distribution in the sub-region. The particular method depends on the shape of the sub-region and is described in the section on the tracking sub-routines.

The neutron energy is selected from the fission spectrum, $\chi(k,i)$, for the isotope selected above. The fission spectrum is normalized such that

$$\sum_i \chi(k,i) \Delta E(i) = 1$$

The cumulative probability distribution is then

$$P_c(i) = \sum_{i'=1}^i \chi(k,i') \Delta E(i')$$

If we draw a random number, ξ , in the range (0,1), the inequality

$$P_c(i-1) \leq \xi < P_c(i)$$

selects a value of i , the initial group number.

We assume that the fission neutrons are generated isotropically, and draw from a distribution of isotropic directions. The direction is specified by the angles θ , ϕ . The differential element of solid angle is

$$\begin{aligned} d\Omega &= d\theta \sin \phi d\phi \\ &= d\theta d\eta, \quad \eta = \cos \phi \end{aligned}$$

$$\int d\Omega = \int_0^{2\pi} d\theta \int_{-1}^1 d\eta = 4\pi$$

Since θ and η are not correlated, we can draw from the independent distributions $f(\theta) = 1$, $g(\eta) = 1$.

$$P_1(\theta) = \frac{1}{2\pi} \int_0^\theta d\theta = \theta/2\pi$$

$$P_2(\eta) = \frac{1}{2} \int_{-1}^\eta d\eta = \frac{1}{2} (\eta+1)$$

If we draw two random numbers, ξ_1 and ξ_2 , in the

range (0,1), then

$$\xi_1 = \theta/2\pi, \quad \theta = 2\pi\xi_1$$

$$\xi_2 = \frac{1}{2}(n+1), \quad n = 2\xi_2 - 1$$

select values of θ and n randomly drawn from an isotropic distribution.

1.2.2 Neutron Weight

A full analog Monte Carlo calculation would make a decision at each collision point as to whether the neutron was captured, caused fission, or was scattered. A more efficient method is to visualize the neutron not as a single neutron but as an ensemble of neutrons all of which have the same trajectories and collisions. If W is the number of neutrons in the ensemble prior to a collision, then at a collision

$$\frac{\Sigma_c}{\Sigma_t} W \text{ are captured,}$$

$$\frac{\Sigma_f}{\Sigma_t} W \text{ cause fission, and}$$

$$\frac{\Sigma_s}{\Sigma_t} W \text{ are scattered.}$$

We call W the neutron "weight" and arbitrarily assign a fission neutron a unit weight, $W = 1$. At each collision the weight is reduced by the fraction absorbed so that the weight following a collision, W' , is

$$W' = \frac{\Sigma_s}{\Sigma_t} W.$$

After many collisions the weight becomes quite small and there is little point in following this neutron further. We select a cut-off weight, W_c , and if $W' < W_c$, we play the game of "Russian Roulette". Assume some average weight, $W_a > W_c$, has been specified. If a random number, ξ , is drawn from the range (0,1),

then the two outcomes, W'' , of the game are

$$\xi > W'/W_a, \quad W'' = 0 \quad (\text{death})$$

$$\xi < W'/W_a, \quad W'' = W_a \quad (\text{life})$$

In the first case the history is ended. In the second case the history is continued with a higher weight. Typical values might be $W_c = .01$ and $W_a = 0.1$. Then for $W' \approx .01$, $W'/W_a = 0.1$ and 90% of the time the history ends. 10% of the time the history is continued with a new weight $W'' = 0.1$. The average survival weight is $W' = 0.01$.

1.2.3 Collision Point

The direction of travel and starting point define a track through the lattice which intersects many sub-regions. We must answer the questions -

1. How far will the neutron go before it collides with an atom?
2. Where in the lattice is this collision point?

The distribution function for the length of the flight in mean-free paths is

$$p_c(\lambda) = e^{-\lambda}$$

The probability distribution is

$$P_c(\lambda) = \int_0^{\lambda} e^{-x} dx = 1 - e^{-\lambda}$$

If we draw a number, ξ , randomly from the range (0,1), then

$$\xi = P_c(\lambda) = 1 - e^{-\lambda}$$

$$\lambda = -\ln(1-\xi)$$

selects a value of λ . Since $1 - \xi$ is just as random in the range (0,1) as ξ itself, we might as well use just

$$\lambda = -\ln \xi$$

The λ selected in this way is called the "flight allowance".

The tracking subroutines, given the starting point and direction of travel, produce a table of track segments, $d(t)$, and associated sub-region numbers, $n(t)$, where t is an index of track segments. Associated with each sub-region, $n(t)$, is a material, $m(t)$, and a total cross section, $\Sigma(t,i)$, where i is the current energy group number. The mean-free-path length of the track segments are $\Sigma(t,i)d(t)$. The collision occurs in the segment, τ , defined by the inequality

$$\sum_{t=1}^{\tau-1} \Sigma(t,i)d(t) < \lambda < \sum_{t=1}^{\tau} \Sigma(t,i)d(t)$$

The distance, D , to the collision point is

$$D = \sum_{t=1}^{\tau-1} d(t) + \left[\lambda - \sum_{t=1}^{\tau-1} \Sigma(t,i)d(t) \right] / \Sigma(\tau,i)$$

where the first term is the distance to segment τ and the second term is the distance in segment τ .

If the track started at the point (x_1, y_1, z_1) , the collision occurs at the point (x_2, y_2, z_2) , given by

$$x_2 = x_1 + D \cos \theta \sin \phi$$

$$y_2 = y_1 + D \sin \theta \sin \phi$$

$$z_2 = z_1 + D \cos \phi$$

1.2.4 Scattering Type

When a neutron makes a scattering collision in a region, four questions must be answered.

1. With which isotope did the scattering occur?
2. In what energy group is the neutron following the scattering?
3. What is the change in the direction of motion as a result of the scattering?
4. What are the direction coordinates following the scattering?

We will consider two types of scattering events. The first type is called "elastic" and will imply that the neutron does not change energy and that it scatters isotropically. The second type is called "scattering" and will imply both a change in energy and anisotropic scattering. We will use superscripts "e" and "s" to denote these two types.

If m denotes the material in the sub-region in which the collision occurs, and i denotes the energy group of the neutron prior to the collision, the elastic and scattering cross sections for the material are:

$$\tau^e(m,i) = \sum_k C(k,m) \sigma^e(k,i)$$

$$\tau^s(m,i) = \sum_k C(k,m) \sigma^s(k,i) \quad .$$

The probability that an elastic event occurs is

$$P_e(m,i) = \tau^e(m,i) / [\tau^e(m,i) + \tau^s(m,i)] \quad .$$

If we draw a random number ξ in the range $(0,1)$, and $\xi < P_e(m,i)$, then an elastic event occurs. In this case we quickly answer the four questions above.

1. (answer not needed)
2. The same group (i) as before the collision.
3. Initial and final directions are not correlated.
4. New direction coordinates drawn from a random isotropic distribution.

1.2.5 Scattering Isotope

If $\xi > P_e(m,i)$, a scattering event occurs. The probability that it occurs with isotope k is

$$p_s(k,m,i) = C(k,m) \sigma^s(k,i) / \tau^s(m,i) \quad .$$

The cumulative probability for this material is

$$P_s(k, m, i) = \sum_{k'=1}^k p_s(k', m, i) \quad .$$

If we draw a random number ξ in the range (0,1), and

$$P_s(k-1, m, i) \leq \xi < P_s(k, m, i)$$

then scattering occurs with isotope k .

1.2.6 Post-Scattering Energy Group

Once we know the isotope number k , we can answer questions 2 and 3 from the Legendre moments of the scattering matrix, $\sigma_o^s(k, i \rightarrow j)$. The isotropic component ($l = 0$) is used to select the final group number j . The probability of transfer into group j is

$$p_t(k, i \rightarrow j) = \sigma_o^s(k, i \rightarrow j) \Delta E(j) / \sigma_o^s(k, i)$$

$$\sigma_o^s(k, i) = \sum_j \sigma_o^s(k, i \rightarrow j) \Delta E(j) \quad .$$

The cumulative probability is

$$P_t(k, i \rightarrow j) = \sum_{j'=1}^j p_t(k, i \rightarrow j') \quad .$$

If we draw a random number ξ in the range (0,1), and

$$P_t(k, i \rightarrow j-1) \leq \xi < P_t(k, i \rightarrow j)$$

then j is the final group number.

For the particular set of (k, i, j) selected above, the angle of scattering is specified by the Legendre coefficients

$$f_e(k, i \rightarrow j) = \sigma_l^s(k, i \rightarrow j) / \sigma_o^s(k, i \rightarrow j) \quad .$$

1.2.7 Scattering Angle

Since (k, i, j) are now fixed, they will be understood and not explicitly represented in the following discussion.

Prior to the scattering event the neutron was traveling in the direction (θ_1, ϕ_1) . The mechanics of the collision process are described in terms of the change in direction of motion, that is, the angle θ_0 and ϕ_0 in Figure 2. If the medium is isotropic (no crystalline or polarization effects), all values of θ_0 are equally probable, so that we can draw a random number ξ in the range $(0,1)$ and compute

$$\theta_0 = 2\pi\xi$$

The distribution of values of ϕ_0 is specified by the function $(\mu_0 = \cos \phi_0)$

$$f(\mu_0) = \sum_{l=0}^L \frac{2l+1}{2} f_l P_l(\mu_0)$$

where the $P_l(\mu_0)$ are the Legendre polynomials and the f_l are the $e_{f,0}(k,i \rightarrow j)$ above. Note that $f \equiv 1$. We will, at a later time, consider $L > 1$. For the moment we will restrict ourselves to $L \leq 1$, so that

$$f(\mu_0) = \frac{1}{2} + \frac{3}{2} f_1 \mu_0, \quad -1 \leq \mu_0 \leq 1$$

$$= \frac{1}{2} (1 + a_1 \mu_0), \quad a_1 \equiv 3f_1$$

If $|a_1| \leq 1$, then $f(\mu_0)$ is everywhere positive. We note that

$$\int_{-1}^1 du_0 f(\mu_0) = 1$$

so that the probability distribution can be written as

$$\begin{aligned} P(\mu_0) &= \int_{-1}^{\mu_0} dx f(x) \\ &= \frac{1}{2} (\mu_0 + 1) + \frac{1}{4} a_1 (\mu_0^2 - 1) \end{aligned}$$

If we draw a random number ξ in the range $(0,1)$, then we can compute a μ_0 from the equation

$$\xi = \frac{1}{2} (\mu_0 + 1) + \frac{1}{4} a_1 (\mu_0^2 - 1)$$

Figure 2. Scattering Angles

which has the solution

$$\mu_o = \frac{1}{a_1} \left[-1 + \sqrt{1 + 2a_1(2\xi - 1) + a_1^2} \right], \quad 0 < |a_1| \leq 1.$$

If $a_1 > 1$, then $f(\mu_o)$ is negative for $\mu_o < -1/a_1$ and f is an improper distribution function. This results from truncating the Legendre expansion too soon ($L = 1$ in this case). There are several ways to get around the problem and we use a very simple one here. We replace $f(\mu_o)$ with another distribution (see Figure 3).

$$f^*(\mu_o) = b (\mu_o - \mu_o^*) \quad , \quad \mu_o^* \leq \mu_o < 1$$

$$= 0 \quad , \quad \mu_o < \mu_o^*$$

and select b and μ_o^* so that the first two moments of $f^*(\mu_o)$ are the same as those of $f(\mu_o)$

$$\int_{\mu_o^*}^1 d\mu_o \, b(\mu_o - \mu_o^*) = \frac{1}{2} b(1 - \mu_o^{*2}) - b \mu_o^* (1 - \mu_o^*) = 1$$

$$\int_{\mu_o^*}^1 d\mu_o \, \mu_o \, b(\mu_o - \mu_o^*) = \frac{1}{3} b(1 - \mu_o^{*3}) - \frac{1}{2} b \mu_o^* (1 - \mu_o^{*2}) = \frac{1}{3} a_1$$

After much algebra we obtain

$$\mu_o^* = a_1 - 2$$

$$b = 2/(1 - \mu_o^*)^2$$

Since μ_o^* must be ≤ 1 , a_1 must be ≤ 3 . If $a_1 > 3$, the distribution with $L = 1$ is hopeless and we will simply take $\mu_o = 1$. If $a_1 \leq 3$, the probability distribution is

$$P(\mu_o) = \int_{\mu_o^*}^{\mu_o} d\mu_o \, b(\mu_o - \mu_o^*)$$

$$= (\mu_o - \mu_o^*) / (1 - \mu_o^*)^2.$$

Figure 3. Distribution Function for u .

If we draw a random number ξ in the range (0,1), then we can compute a μ_o from the equation

$$\xi = (\mu_o - \mu_o^*)^2 / (1 - \mu_o^*)^2$$

which has the solution

$$\mu_o = \mu_o^* + (1 - \mu_o^*) \sqrt{\xi}$$

If $a_1 < -1$, then in a similar manner we replace $f(\mu_o)$ with the distribution

$$\begin{aligned} f^*(\mu_o) &= b(\mu_o^* - \mu_o) \text{ for } -1 < \mu_o < \mu_o^* \\ &= 0 & \mu_o^* < \mu_o \end{aligned}$$

Selecting b and μ_o^* so that the first two moments of $f^*(\mu_o)$ are the same as those of $f(\mu_o)$, we get

$$\mu_o^* = a_1 + 2$$

$$b = 2 / (1 + \mu_o^*)^2$$

Since μ_o^* must be ≥ -1 , a_1 must be ≥ -3 . If $a_1 < -3$, we will simply take $\mu_o = -1$. If $a_1 \geq -3$, the probability distribution is

$$\begin{aligned} P(\mu_o) &= \int_{\mu_o}^{\mu_o^*} d\mu_o b(\mu_o^* - \mu_o) \\ &= \frac{(\mu_o - \mu_o^*)^2}{(1 + \mu_o^*)^2} \end{aligned}$$

Drawing a random number ξ from the range (0,1), we compute μ_o from the equation

$$\xi = \frac{(\mu_o - \mu_o^*)^2}{(1 + \mu_o^*)^2}$$

to get the solution

$$\mu_o = \mu_o^* + (1 + \mu_o^*) \sqrt{\xi}$$

1.2.8 Post-Scattering Direction

If (θ_1, ϕ_1) were the direction coordinates before the scattering, and (θ_0, ϕ_0) are the coordinates after scattering measured from the initial direction, the coordinates after scattering measured from the (x, y, z) axis are (θ_2, ϕ_2) . The relationship between these angles is expressed somewhat more compactly by using the direction cosines α, β, γ as shown in Figure 4. The formulas are:

$$|\gamma_1| \neq 1, \alpha_2 = \mu_0 \alpha_1 + g(\alpha_1 \gamma_1 \cos \theta_0 - \beta_1 \sin \theta_0)$$

$$\beta_2 = \mu_0 \beta_1 + g(\beta_1 \gamma_1 \cos \theta_0 + \alpha_1 \sin \theta_0)$$

$$\gamma_2 = \mu_0 \gamma_1 - g(1 - \gamma_1^2 \cos \theta_0)$$

$$g \equiv \sqrt{(1 - \mu_0^2)(1 - \gamma_1^2)}$$

$$|\gamma| = 1, \alpha_2 = \sqrt{1 - \mu_0^2} \cos \theta_0$$

$$\beta_2 = \sqrt{1 - \mu_0^2} \sin \theta_0$$

$$\gamma_2 = \mu_0 \gamma_1 / |\gamma_1|$$

$$\text{where } \alpha_1 = \cos \theta_1 \sin \phi_1 \quad \alpha_2 = \cos \theta_2 \sin \phi_2$$

$$\beta_1 = \sin \theta_1 \sin \phi_1 \quad \beta_2 = \sin \theta_2 \sin \phi_2$$

$$\gamma_1 = \cos \phi_1 \quad \gamma_2 = \cos \phi_2$$

1.2.9 Resonance Treatment

The resonance treatment in STERIL duplicates the treatment in the RAHAB integral transport modules. Resonance absorption and fission reaction rates per unit slowing-down density, $R_a(k, n, i)$ and $R_f(k, n, i)$, are calculated by CREEP and RESIN for isotope k , sub-region n , and group i . These rates are to be corrected by the flux ratios

$$f(n, i) = \phi(n, i-1) / \bar{\phi}(c, i-1), \quad n \in c$$

where $\bar{\phi}(c, i-1)$ is the volume average flux in cell c , group $i-1$. Corrected rates are given by

Figure 4. Direction Cosines.

$$R_a^*(k,n,i) = f(n,i) R_a(k,n,i)$$

$$R_f^*(k,n,i) = f(n,i) R_f(k,n,i)$$

If a RAHAB integral transport calculation has been done for this lattice, the corrected rates are available in the Lattice Intermediate Data Set (LIDS). If corrected results are not available, the $f(n,i)$ are taken to be unity for the first batch, and computed from the current flux profiles for subsequent batches.

The total reaction rates are computed from

$$R_a(k,n,i) = R_a^*(k,n,i) Q(c,i) \quad , \quad nec$$

$$R_f(k,n,i) = R_f^*(k,n,i) Q(c,i) \quad , \quad nec$$

where $Q(c,i)$ is the slowing-down density in cell c , group i . By definition, $Q(c,i)$ is the number of neutrons which cross the energy boundary $E(i)$ in cell c . We can compute this quantity by accumulating the neutron weight at each boundary crossing.

Resonance absorption decreases the neutron weight. If $R_a^*(c,i)$ is given by

$$R_a^*(c,i) = \sum_k \sum_{nec} R_a^*(k,n,i)$$

then the weight following a boundary crossing is

$$W' = W \left[1 - R_a^*(c,i) \right]$$

If j is the final group following a collision, and i is the initial group, then boundaries $j, j+1, j+2, \dots, i+1$, were crossed so that the post-collision weight is given by

$$W' = W \prod_{l=i+1}^j \left[1 - R_a^*(c,l) \right]$$

For the calculation of spatial moments to absorption, we assume (for want of more detailed information) that all of the resonance absorption occurs at the collision point.

1.2.10 Cross Section Conventions

The use of the "transport approximation" and treating the (n,2n) reaction as scattering requires several modifications to the cross sections. To understand these, we will manipulate the one-dimensional multigroup transport equation,

$$\left[\mu \frac{\partial}{\partial x} + \Sigma^t(j) \right] \phi(x, j, \mu) = \int_{-1}^1 d\mu' \sum_i \Delta E(i) \Sigma^m(i \rightarrow j, \mu_0) \phi(x, i, \mu')$$

$$\Sigma^m(i \rightarrow j, \mu_0) = \Sigma^{el}(i \rightarrow j, \mu_0) + \Sigma^{in}(i \rightarrow j, \mu_0) + 2\Sigma^{n2n}(i \rightarrow j, \mu_0)$$

$$\Sigma^t(j) = \Sigma^c(j) + \Sigma^f(j) + \Sigma^{el}(j) + \Sigma^{in}(j) + \Sigma^{n2n}(j)$$

$$\mu_0 = \mu \cdot \mu'$$

where we have ignored the fission source and suppressed the x argument in the cross sections.

The scattering matrix is represented on the MULTIGRP data set as the microscopic expansion coefficient matrices, $\sigma_\ell^m(i \rightarrow j)$,

$$\sigma^m(i \rightarrow j, \mu_0) = \sum_{\ell=0}^L \frac{2\ell+1}{2} \sigma_\ell^m(i \rightarrow j) P_\ell(\mu_0)$$

and the associated cross sections, $\sigma_\ell^m(i)$,

$$\begin{aligned} \sigma_\ell^m(i) &= \sum_j \sigma_\ell^m(i \rightarrow j) \\ &= \sigma_\ell^{el}(i) + \sigma_\ell^{in}(i) + 2\sigma_\ell^{n2n}(i) \end{aligned}$$

(Note that the $\Delta E(j)$ is included in $\sigma_\ell^m(i \rightarrow j)$ on MULTIGRP.)

In the "transport approximation" we construct an artificial scattering matrix composed of

1. an isotropic matrix, plus
2. a diagonal matrix with a forward-directed ($\mu_0=1$) angular distribution.

If these terms are selected to match (on the average) the first two Legendre moments of the true matrix, we obtain

$$\Sigma^{m*}(i \rightarrow j, \mu_0) = \frac{1}{2} \Sigma_0^{m*}(i \rightarrow j) + \Sigma_1^m(i) \delta_{ij} \delta(\mu_0 - 1) / \Delta E(i)$$

$$\Sigma_0^{m*}(i \rightarrow j) = \Sigma_0^m(i \rightarrow j) - \Sigma_1^m(i) \delta_{ij} / \Delta E(i) .$$

Inserting this into the transport equation and performing the μ' integration,

$$\left[\mu \frac{\partial}{\partial x} + \Sigma^t(j) - \Sigma_1^m(j) \right] \phi(x, j, \mu) = \frac{1}{2} \sum_i \Delta E(i) \Sigma_0^{m*}(i \rightarrow j) \phi_0(x, i)$$

The transport cross section on the MULTIGRP data set is defined by

$$\sigma^{tr}(j) = \sigma^t(j) - \sigma_1^m(j)$$

and is then used in place of the total cross section in integral transport calculations.

The (n,2n) reaction is treated like scattering and it is necessary to remove the (n,2n) production from the scattering matrix. This is done by subtracting $\Sigma^{n2n}(j) \phi_0(j)$ from both sides. The RHS then contains the matrix

$$\Sigma_0^{m**}(i \rightarrow j) = \Sigma_0^{m*}(i \rightarrow j) - 2\Sigma^{n2n}(j) \delta_{ij} / \Delta E(j) .$$

The LHS is

$$\left[\mu \frac{\partial}{\partial x} + \Sigma^{tr}(j) \right] \phi(x, j, \mu) - 2\Sigma^{n2n}(j) \frac{1}{2} \phi_0(x, j) .$$

For the purposes of this approximation we can replace $\frac{1}{2} \phi_0(x, j)$ with $\phi(x, j, \mu)$ and define a corrected transport cross section

$$\Sigma^{tr*}(j) = \Sigma^{tr}(j) - 2\Sigma^{n2n}(j)$$

which corresponds to an absorption cross section

$$\sigma^{a*}(j) = \sigma^c(j) + \sigma^f(j) - \sigma^{n2n}(j) .$$

These corrections (particularly the transport approximation) frequently lead to negative diagonal elements of the scattering matrix. These negative elements

cause problems when we convert the matrices to probabilities. The present version of STERIL stops with a diagnostic message if a negative diagonal element is encountered.

JOSHUA DPSTM-500	Title Module STERIL	Vol. IV	Page J.1.28
		Rev.	Date 12/70

1.3 Edits

1.3.1 Eigenvalue

Let us suppose that B batches of H histories each have been completed. The total number of fission neutrons produced can then be computed from the flux and slowing-down density. If we denote this number by F, denote the total absorptions by A, and total leakage by L, then, since each starting neutron has a unit weight,

$$A + L = BH$$

the eigenvalue, k_{eff} , is the total number of neutrons produced per fission source neutron,

$$k_{eff} = F / (A + L) = F / BH$$

1.3.2 Neutron Flux

If d is the length of a track segment in a sub-region and W is the weight of a neutron entering the sub-region, then we expect to observe

$$\int_0^d dx W \Sigma e^{-\Sigma x} = W (1 - e^{-\Sigma d})$$

collisions in the sub-region. The accumulation of this quantity by group, i, and sub-region, n, for all histories is the total collision density which is equal to $\Sigma(n,i) \phi(n,i) V(n) \Delta E(i)$.

1.3.3 Slowing-Down Density

The slowing-down density by cell, c, and energy group, i, is the total number of neutrons which cross the energy boundary $E(i)$ in cell c. Whenever a collision occurs, and the neutron changes groups, the post-collision weight is accumulated for each energy boundary crossed.

1.3.4 Spatial Moments

There are many integral properties of the slowing-down distribution that can be computed during the Monte Carlo calculation. The most useful are the mean-squared-distances along the x, y, and z axes that a neutron travels between some starting point and some end point such as absorption or slowing-down

below some given energy. These moments are of the form

$$\langle x^2 \rangle = \frac{\sum_n W_n (x_n - x_0)^2}{\sum_n W_n}$$

$$\langle y^2 \rangle = \frac{\sum_n W_n (y_n - y_0)^2}{\sum_n W_n}$$

$$\langle z^2 \rangle = \frac{\sum_n W_n (z_n - z_0)^2}{\sum_n W_n}$$

where n counts those collisions which contribute to the moments, W_n is the appropriate weight, (x_n, y_n, z_n) are the coordinates of a collision, and (x_0, y_0, z_0) are the coordinates of the collision where the moments calculation was started. Since the formulas for $\langle x^2 \rangle$ in the remaining discussion.

The precise definition of these quantities depends on the type of moments calculation. We consider three types here. For each type there is an energy range (E_0, E_1) over which the moment is to be calculated. The calculation starts when a neutron is born with an energy E which lies in the range $(E_0 > E > E_1)$ or enters the range as a result of a collision which reduces the neutron energy from an $E > E_0$ to an $E < E_0$ (see Figure 5). The point where this occurs is the starting point (x_0, y_0, z_0) , and the post-collision weight as it enters the range is W_0 .

In the first two types of moments calculation, we want the mean-squared-distance to the point where a collision removes the neutron from the range, that is, the post-collision energy $E < E_1$. The coordinates of this point are (x_1, y_1, z_1) and the post-collision weight is W_1 . If we want the moments of the distribution of those neutrons which actually escape the range, we would use (Type 1)

$$\langle x^2 \rangle = \frac{\sum_h W_1 (x_1 - x_0)^2}{\sum_h W_1}$$

and we would get one contribution from each history h .

If we want to ignore absorptions in the range, we would use (Type 2)

$$\langle x^2 \rangle = \frac{\sum_h W_0 (x_1 - x_0)^2}{\sum_h W_0} .$$

We note that the escape probability for the range is

$$p = \frac{\sum_h W_1}{\sum_h W_0} .$$

In the third type of moments calculation we want the mean-squared-distance to the point where neutrons are absorbed in the range. Let (x_n, y_n, z_n) denote a collision point, and W_n^a the weight absorbed at that collision. The index n will count collisions in which the pre-collision energy lies in the range for one history. The formula is (Type 3),

$$\langle x^2 \rangle = \frac{\sum_h \sum_n W_n^a (x_n - x_0)^2}{\sum_h \sum_n W_n^a}$$

Figure 5. Energy Range for Moments Calculation.

JOSHUA
DPSTM-500

Title

Module STERIL

Vol.

IV

Page

J.1.32

Rev.

Date

12/70

(Blank)

2. Program Logic

2.1 Interfaces

STERIL reads the following records:

('TEMP'.'LIDS'.DLAT.VLAT → V)

'INPUT'.'STERIL'
V.'NAME'.NCASE.O .O
V.'CØMP'.NCASE.O .O
V.'GEØM'.NCASE.O .O
V.'GEØT'.NCASE.O .O
V.'CRPS'.NCASE.O .O
V.'RESØ'.NCASE.1 .O
V.'XSEC'.NCASE.NER.O
V.'SMAT'.NCASE.NER.O
V.'CHI'.NCASE.O .O

STERIL creates the following records to be used internally:

V.'PRØD'.NCASE.O .O
V.'IFSP'.NCASE.O .O
V.'ISCT'.NCASE.O .O
V.'XTØT'.NCASE.O .O
V.'XABS'.NCASE.O .O
V.'RABS'.NCASE.O .O

STERIL writes the following records to the LIDS:

V.'FLUX'.NCASE.O .O
V.'FLUX'.NCASE.1 .O
V.'QCEL'.NCASE.O .O

2.2 Over-all Structure

The STERIL module consists basically of two segments: the data setup and the Monte Carlo calculations. A detailed flow diagram of the main program is given in Figure 6. The basic steps are:

1. Set up initial fission production distribution, probabilities, cross sections, and resonance reaction rates.
2. Start batch counter.
3. Start history counter.
4. Set up the initial parameters for the neutron-weight, position, and energy.
5. Clear flight path and compute flight allowance.
6. Track neutrons until a collision is made.
7. Compute relative and absolute coordinates of collision point.
8. Compute weight absorbed at collision and reduce weights.
9. Determine type of collision.
10. Tally moments.
11. If scattering event, calculate slowing-down density and resonance absorption for each boundary crossed.
12. Decide if history is to continue. If so, select direction coordinates and repeat from Step 5.
13. Repeat from Step 3 until all histories are completed.
14. Update fission production distribution and fission production probabilities and repeat from Step 2 until all batches are completed.
15. Prepare output for the LIDS.

Figure 6. Subroutine MAIN.

JOSHUA DPSTM-500	Title Module STERIL	Vol. IV	Page J.1.36
		Rev.	Date 12/70

2.3 Data Setup

The Monte Carlo calculations require that certain data be kept in core at all times; such as cross sections and scattering data for all energy groups, the fission spectrum, and resonance reaction rates. These data are available to STERIL in the records used by the integral transport modules TRAMP and FIST.

In the subroutine SETUP, these records from the LIDS are read in and are organized into compact intermediate data sets for use in STERIL alone. Then these intermediate data sets are packed into the DATA array and are kept in core during the calculation. Also, space is allocated for accumulation of the collision density and slowing-down density, and tables relating materials to the geometry of the lattice are constructed.

A flow chart of SETUP is provided in Figure 7, and the succeeding four sections describe the subroutines called by SETUP to construct the intermediate data sets.

Figure 7. Subroutine SETUP.

2.4 Preparation of the Distribution for Fission Production

The total number of fission neutrons produced by the k^{th} isotope in the n^{th} sub-region which is in cell c is given by

$$F(k,n) = \sum_{i=1}^{\text{NEG}} v_k(i) \left[\sigma_k^f(i) \phi(i,n) + R_k^f(i,n) * Q(i,c) \right] .$$

Following the completion of a batch, the total flux by sub-region $\phi(i,n)$ and the slowing-down density $Q(i,c)$ are available for calculating $F(k,n)$. This function $F(k,n)$ and the additional quantities needed for its calculation are packed into the DATA array in the PRØD block. The structure of this block is shown in Figure 8 .

Initially, the flux and slowing-down density are not available; therefore, a distribution must be guessed. If a fission distribution from RAHAB is catalogued in the LATTICE library, it is used to calculate the production which is then stored in the PRØD block by sub-region and isotope. If there are no RAHAB results available, the function $F(k,n)$ is calculated as follows:

$F(k,n) = V(n) * v(\text{ITH}) * \sigma_k^f(\text{ITH})$ where $V(n)$ is the volume of sub-region n and ITH is the index of the group containing $E = 0.0253$ ev. Since this function will be used to calculate probabilities, it is always normalized so that

$$\sum_n \sum_k F(k,n) = 1 .$$

If the lattice is a dead lattice for which there are no fission neutrons, then there is no PRØD block.

Initially, the PRØD block is set up in the subroutine SPRØD. The logic for this subroutine is shown in Figure 9 . Once a batch has been completed, the fission production can be computed using the flux and slowing-down density tabulated during the tracking. This updated fission production is then stored in the PRØD block by the subroutine PRØDUP. (Fig. 10) Also, since the total production and the total absorption are available at this time, k -effective is calculated for this batch. At the completion of all batches, an edit of k -effective statistics is printed.

PRØD(K)
KPPT(2,K)

KSR	+0	NSR	KNXTSR	Sub-region No., pointer to next sub-region.
KIS	+0	NIS	KNXTIS	Isotope number, pointer to next isotope.
	+1		KXNF	Pointer to $\sigma^f(i)$ for this isotope.
	+2	C(NIS,NSR)		Concentration of isotope in sub-region.
	+3	F(NIS,NSR)		Total production for this isotope in this sub-region.
	+4	IA	IB	Upper and lower group limits for resonance fission production rates, RNF. If IA = 0, no rates are given.
	+5	RNF(NIS,NSR,IA)		

		RNF(NIS,NSR,IB)		Upper and lower limits of data for $\sigma^f(I)$, I = IA,IB for this isotope.

KXNF	+0	IA	IB	Upper and lower limits of data for $\sigma^f(I)$, I = IA,IB for this isotope.
	+1	XNF(NIS,IA)		

		XNF(NIS,IB)		

Figure 8. Schematic of the PRØD Block.

Figure 9. Subroutine SPRØD.

Figure 10. Subroutine PRØDUP.

2.5 Preparation of Fission Probabilities and Chi Probabilities

To select the starting sub-region/isotope pair and the initial energy of the isotopes, cumulative probability functions for the fission source and for the fission spectrum are needed. In the case of an active lattice, the fission source probability P_f is calculated from the discrete normalized function $F(\rho)$ that is stored in the PRØD block. For each sub-region/isotope pair ρ , the function P_f is calculated as follows:

$$P_f(\rho) = \sum_{\rho'=1}^{\rho} F(\rho').$$

In the case of a dead lattice, there is no production; so we use a uniform distribution of sub-regions; that is,

$$F(\rho) = \frac{\text{VSR}(\rho) * \text{NLP}(\rho)}{\sum_{\rho \in \text{NCL}} \text{VSR}(\rho) * \text{NLP}(\rho)}$$

where VSR is the volume of the sub-region of the pair ρ and NLP is the number of lattice positions of the sub-region in the supercell.

For each isotope k of the above pair, there is a fission spectrum $\chi(k,i)$ if the lattice is active. This spectrum is read in from the MULTIGRP data set. Since this function has been normalized so that

$$\sum_{i=1}^{\text{NEC}} \chi(k,i) \Delta E(i) = 1,$$

we can calculate a cumulative Chi probability function as follows:

$$P_c(i) = \sum_{i'=1}^i \chi(k,i') \Delta E(i').$$

For a dead lattice, there is no fission spectrum on MULTIGRP; so a spatially uniform fission spectrum is calculated as follows:

$$F(j) = g(\chi_1) - g(\chi_2)$$

$$g(x) = 1.1283792 \, x e^{-x^2} - \operatorname{erf}(x)$$

$$x_1 = \sqrt{E(j+1)/\theta}$$

$$x_2 = \sqrt{E(j)/\theta}$$

$$\theta = 1.273 \times 10^6 \text{ ev} \quad .$$

These probabilities must be kept in core during the course of the Monte Carlo calculations. Thus, since storage is critical, we store all probabilities P as half-words by expressing each one as an integer $IP < IBASE$; that is IP is the integer part of $IBASE \cdot P$. These Integer probabilities are packed in the block IFSP with the structure shown in Figure 11. The logic of the subroutine FISPRØ that initially sets up the block IFSP is shown in Figure 12. After each batch the fission production probabilities are updated by the subroutine FISSUP.

INTEGER*2 IFSP(K)

* Probabilities are expressed as integers $< \text{IBASE}$. If P is a probability ($0 \leq P \leq 1.0$), then $\text{IP} \equiv \text{integer part of } \text{IBASE} * P$.

** Ordered so that the largest contributor appears first in the list and smallest last.

Figure 11. Schematic of IFSP Block.

Figure 12. Subroutine FISPRØ

2.6 Preparation of Scattering, Elastic, and Transfer Probabilities

To determine the fate of a neutron following a collision, we use the scattering matrices and cross sections from the MULTIGRP data set to construct a probability matrix. For each initial group I , we calculate the following probabilities:

1. P_s - the probability that a scattering event occurs
2. P_t - the probability of transfer of energy in scattering event
3. P_e - the probability that an elastic event occurs.

Recall that in a scattering event, there is both a change in energy and anisotropic scattering. Since the scattering matrix in MULTIGRP includes the $(n,2n)$ matrix, we compute the effective scattering cross section for each material m and group i as follows:

$$\sigma^S(m,i) = \sum_k \sigma_o^m(k,i) - 2\sigma^{n2n}(k,i)$$

(See Section 1.2.10 for details.)

Then

$$\Sigma^S(m,i) = \sum_k C(k,m) \sigma^S(m,i)$$

and our cumulative probability density function P_s is

$$P_s(k,m,i) = \sum_{k'=1}^k p_s(k',m,i)$$

where

$$p_s = \frac{C(k,m) \sigma^S(k,i)}{\Sigma^S(m,i)}$$

Title	Module STERIL	Vol.	Page
		IV	J.1.47
Rev.			Date
			12/70

For each isotope k , the probability of transfer from group i to group j is

$$p_t(k, i \rightarrow j) = \sigma_O^S(k, i \rightarrow j) \Delta E(j) / \sigma_O^S(k, i)$$

where

$$\sigma_O^S(k, i \rightarrow j) = \left[\sigma_O^m(k, i \rightarrow j) - 2\sigma^{n2n}(k, i) \delta_{ij} \right] .$$

(Refer to Section 1.2.10 for details.)

Then the cumulative transfer probability function is

$$P_t(k, i \rightarrow j) = \sum_{j'=1}^j p_t(k, i \rightarrow j') .$$

In an elastic event the neutron does not change energy and it scatters isotropically. In this instance, the scattering matrix of the isotope is diagonal and hence is not stored in the MULTIGRP data set. We calculate the elastic cross section for material m and group i as follows:

$$\sum^e (m, i) = \sum_k C(k, m) \sigma^e(k, i)$$

where $\sigma^e(k, i) = \sigma^t(k, i) - \sigma^c(k, i) - \sigma^f(k, i) - \sigma^{n2n}(k, i) -$

$$- (\sigma_O^m(k, i) - 2\sigma^{n2n}(k, i)) .$$

(Refer to Section 1.2.10 for details.)

Since these probability matrices must be held in core during the entire calculation, it is important to store the probabilities in a compact form. This is accomplished by converting the probability to an integer and storing it in a half-word; that is, if P is a probability, then we store P as the integer $IP = IBASE * P$ where $IBASE$ is any number less than or equal to $32K$, the maximum value that can be stored in a half word. These integer probabilities are packed into the DATA array in the block ISCT. The structure is shown in Figure 13. In Figure 14, the logic of the subprogram STEP which sets up the matrices, is shown.

* Probabilities and fractions are expressed as integers $< \text{IBASE}$. If F is a fraction ($0 \leq F \leq 1.0$), then $\text{IF} = \text{Integer part of } \text{IBASE} * F$.

** Ordered so that largest contributor appears first in list and smallest contributor last.

Figure 13. Schematic of ISCT Block.

2 pages missing from origiinal document

2.8 Coordinate Systems and Tracking Methods

Since the lattice we are considering is uniform and infinite in the z direction, it is convenient to use a coordinate system as illustrated in Figure 1. The Monte Carlo tracking routine assumes that this coordinate system is centered in each cell of the lattice. Thus, as a particle leaves one cell and enters another, its coordinates are changed to refer to the coordinate system of the cell entered. However, for the purpose of calculating the moments to absorption and the moments to the energy boundary, the neutron must be tracked in reference to the entire lattice, rather than the cell. For this reason we have two coordinate systems: a relative coordinate system and an absolute coordinate system.

The Monte Carlo tracking subroutine tracks the neutron in the x-y plane to a sub-assembly boundary. Thus when tracking a neutron using the relative coordinate system, all calculations of its position must be done on a sub-assembly-to-sub-assembly basis. For example, suppose a track of a neutron begins at the point (x_0, y_0, z_0) with direction (θ_0, ϕ_0) . It is followed until it escapes a sub-assembly at the point (x_1, y_1, z_1) . If no collision was made in this sub-assembly, the tracking continues. However, the tracking routine first calculates the position and direction of the neutron relative to the coordinate system of the cell containing the adjacent sub-assembly. If the adjacent sub-assembly is in the same cell, the track continues from the point (x_1, y_1, z_1) with direction (θ_0, ϕ_0) . But if the sub-assembly from which the neutron leaves is the last sub-assembly of the cell, then the point (x_1, y_1, z_1) and direction (θ_0, ϕ_0) are translated and notated to refer to the coordinate system of the cell just entered.

The neutron is followed in this manner until a collision is made. Then, if the neutron's flight path traverses a sub-assembly, (x_1, y_1, z_1) and $(\theta_1, \phi_0)^*$ are the position and direction of the neutron at the point of entrance to the sub-assembly, and (x_1, y_1, z_1) lies on the sub-assembly face. If the neutron remains in the same sub-assembly for the entire flight, then (x_1, y_1, z_1) and (θ_1, ϕ_0) are the position and direction at the last collision. Thus, the tracking routine provides us with the necessary information to calculate the collision position (x_2, y_2, z_2) :

$$x_2 = x_1 + D \cos \theta_1 \sin \phi_0$$

$$y_2 = y_1 + D \sin \theta_1 \sin \phi_0$$

$$z_2 = z_1 + D \cos \phi_0$$

* Note that angle ϕ does not change when a cell boundary is crossed.

where D is the distance traveled from (x_1, y_1, z_1) in the x-y-z plane.

(See Section 1.2.3 for details.) Calculation of the post collision direction (θ_2, ϕ_2) is discussed in Section 1.2.8.

Now consider a track of the neutron relative to the absolute coordinate system. Suppose a track begins at (x_{1A}, y_{1A}, z_{1A}) with direction (θ_{1A}, ϕ_0) . Since there is no change in direction until a collision is made, tracking can be done on a collision-to-collision basis, rather than on a sub-assembly-to-sub-assembly basis. Thus, when a neutron collides, its absolute position is given by

$$x_{2A} = x_{1A} + D \cos \theta_{1A} \sin \phi_0$$

$$y_{2A} = y_{1A} + D \sin \theta_{1A} \sin \phi_0$$

$$z_{2A} = z_{1A} + D \sin \phi_0$$

where D is the total distance in the x-y-z plane that the neutron has traveled since the last collision. After a collision occurs, the new direction is given by (θ_{2A}, ϕ_2) where

$$\theta_{2A} = \theta_{1A} + (\theta_2 - \theta_1)$$

and ϕ_2 is the angle discussed in Section 1.2.8.

2.9 Moments Calculation

The spatial moments editing is done in a separate subroutine MØMENT. Presently, it will calculate the mean-square distances along the x, y, and z axes that a neutron travels between some starting point and absorption or slowing-down below some given energy. Before any calculations are begun, the routine MØMENT is called to clear all internal summation tables and to adjust the boundaries of the energy range over which the moment calculation is to be done. After the calculation is begun, the routine is called again to store the initial weight and position of the neutron as it enters the energy range of the moments calculation. Then each time the neutron makes a collision, MØMENT is given the post-collision weight, the weight absorbed, the absolute collision point coordinates, and the post-collision energy. If the neutron changes energies upon collision and if the initial energy is not in the thermal range, then each boundary crossed is treated as a separate collision. At this time, the absorption due to resonances are tabulated.

A detailed explanation of the logic used in MØMENT follows along with a flow chart shown in Figure 15.

Moments Calculation

Input Tables (in CØMMØN)

M	-	Moments case number (allow for about 20 cases).	
MX	-	Number of cases.	
MTYPE(M)	-	Type of moment	
	= 1	Moments to energy boundary (including absorption).	
	= 2	Moments to energy boundary (excluding absorption).	
	= 3	Moments to absorption.	
EO(M)	-	Starting energy (ev)	} Adjusted by code to coincide with nearest group boundary.
E1(M)	-	Stopping energy (ev)	

Internal Tables

- IST(M) - Status flag for this history.
 = 0 talley has not been started
 = 1 talley has been started
 = 2 talley has been stopped
- A(N,I,M) - Table of results
 N = 1, refers to x
 = 2, refers to y
 = 3, refers to z
 I = 1, value of x_o , y_o , or z_o
 = 2, sum of $w(x-x_o)^2$, $w(y-y_o)^2$, or
 $w(z-z_o)^2$
 = 3, sum of $w(x-x_o)^4$, $w(y-y_o)^4$, or
 $w(z-z_o)^4$
- WO(M) - w_o for this history
- SUMW(M) - Sum of w
- ICNT(M) - Count of entries in sum

Arguments

- WT - Post-collision total weight
- WA - Weight absorbed
- x,y,z - Collision point coordinates
- EJ - Post-collision energy or source energy
- L - Entry flag
 = 1, start of calculation, clear summation
 tables
 = 2, start of history entry
 = 3, collision entry
 = 4, calculate results to date and print
 report

L=1, Start of Calculation

1. Count cases and set MX.
2. Adjust EO and EI to lie on nearest group boundary.
3. Clear tables A(N,I,M), SUMW(M), ICNT(M), IST(M), WO(M)

L=2, Start of a History

(Assume WT, X, Y, Z, EJ are given; ignore WA)

1. Do the following steps for each M
2. Test EJ

EO(M) < EJ , Set IST(M)=0.
 EI(M) < EJ < EO(M) , Set IST(M)=1, move x,y,z
 into A(N,1,M),
 move WT into WO(M).

EJ < EI(M) , Set IST(M)=2.

L=3, Collision Point

1. Do the following steps for each M.
2. If IST(M) = 2, ignore this case.
3. If IST(M) = 0, EJ > EO(M), ignore this case.
 EJ < EO(M), set IST(M)=1,
 move X,Y,Z into A(N,1,M),
 move WT into WO(M).
4. If IST(M) = 1, MTYPE(M) = 1 or 2, test EJ.
 EJ > EI(M), ignore this case.
 EJ ≤ EI(M), accumulate W, $W(x-x_0)^2$, $W(x-x_0)^4$, etc.
 and CNT
 using W = WT if MTYPE(M) = 1
 = WO(M) if MTYPE(M) = 2
 Set IST(M) = 2

5. If $IST(M) = 1$, $MTYPE(M) = 3$, accumulate, W , $W(x-x_0)^2$, $W(x-x_0)^4$, etc.
and $ICNT(M)$ using $W=WA$.
If $EJ < E1(M)$, set $IST(M) = 2$

L=4 Calculate Results and Print

1. Print headings, batch number, and total histories.
2. Repeat the following steps for each case $M=1, MX$
3. Repeat the following steps for each coordinate $N=1,2,3$
4. Calculate

$$\langle x^2 \rangle = \sum_n W_n (x_n - x_0)^2 / \sum_n W_n = A(N,2,M) / SUMW(M)$$

$$\langle x^4 \rangle = \sum_n W_n (x_n - x_0)^4 / \sum_n W_n = A(N,3,M) / SUMW(M)$$

$$\sigma = \sqrt{|\langle x^4 \rangle - \langle x^2 \rangle^2|}$$

$$P.E. = 0.873 \sigma / \sqrt{\sum_n 1} = 0.873 \sigma / \sqrt{ICNT(M)}$$

5. If $N=1$ print $M, MTYPE(M), ICNT(M), 'x', \sigma, \langle x^2 \rangle, P.E.$
 $=2$ print $'y', \sigma, \langle y^2 \rangle, P.E.$
 $=3$ print $'z', \sigma, \langle z^2 \rangle, P.E.$

Figure 15. Subroutine MOMENT

Figure 15a.

2.10 Preparation of Output

Since STERIL replaces the TRAMP and FIST modules, it must produce the same output records as FIST. These output records are the flux by region and the slowing-down densities. We have available the total collision density by sub-region and the total slowing-down density. The collision density is converted to flux by region by subroutine CRF, and in the case of an active lattice both the flux and slowing-down density are scaled to unit total production. These records are then written to the LIDS.

JOSHUA DPSTM-500	Title Module STERIL	Vol. IV	Page J.1.60
		Rev.	Date 12/75

3. Coding Details

3.1 Module Structure

The STERIL module consists of a main program and two overlay segments. The subprograms of each one are as follows:

ROOT SEGMENT

1. All common areas.
2. MAIN
3. MANAGE
4. ERRØR
5. FISSUP
6. BINSCH
7. RANDIN
8. RNDØUT
9. FLTRNF
10. CRF

OVERLAY SEGMENT 1 (Data Setup)

1. SETUP
2. SPRØD
3. FISPRØ
4. STEP
5. RESABS

OVERLAY SEGMENT 2 (Monte Carlo Calculations)

1. FISR
2. USSR
3. TRACK
4. MØMENT
5. PRØDUP

Additional comments on the subroutines of STERIL and the definitions of the symbols used in CØMMØN areas are provided in the following sections.

3.2 Common Areas

1. ~~C~~OMM~~O~~N/DATA - This common area holds the intermediate data sets created in the first overlay segment. Its size is determined at execution time and is dependent on the core size specified in the procedure step. Refer to manuscript B.1 for a description of this array and its usage.

2. ~~C~~OMM~~O~~N/M~~O~~MNT/ - This common area contains the input data for the moments calculation. The following data are included:

MX - Number of cases to be run.
MTYPE(M) - Type of moment
 = 1 Moments to energy boundary
 (including absorption).
 = 2 Moments to energy boundary
 (excluding absorption).
 = 3 Moments to absorption.

EO(M) - Starting energy in ev.

E1(M) - Stopping energy in ev.

3. ~~C~~OMM~~O~~N/GR~~O~~UPS/ - This area contains the parameters defining the multigroup structure. The following items refer to the multigroup structure:

NEG - Number of groups.
JFTG - Index of first thermal group.
JCHI - Index of the last group where the
 fission spectrum is non-zero.
E(I) - Upper energy limit (ev) of group I.
EBAR(I) - Characteristic energy (ev) of group I.
DE(I) - Width of group I.
NEBX - Number of energy bands (set to three
 in CRISP).
IEBH(N) - Largest group number contained in
 energy band n.

4. ~~COMMON~~/TEMP/ - This common area contains miscellaneous data needed for the Monte Carlo calculations. The following items are included:

NBCH	-	The current batch number.
NBCHX	-	The total number of batches.
NHST	-	The current history number in batch NBCH.
NHSTX	-	The total number of histories in one batch.
NSRX	-	Number of sub-regions in supercell.
NSRXX	-	Maximum number of sub-regions allowed.
NCLX	-	Number of cells in supercell.
NMTX	-	Number of materials.
TWABS	-	Total weight absorbed in a batch.
TPRØD	-	Total production for batches 1 through NBCH.
IBASEF	-	Fission probability base.
IBASEC	-	Chi probability base.
IBASES	-	Scattering probability base.
IBASEJ	-	Transfer probability base.
VSRT(N)	-	Volume of sub-region n.
NCLSR(N)	-	Cell containing sub-region n.
NMTSR(N)	-	Material in sub-region n.
TLPCL(N)	-	Number of lattice positions of sub-region n.

5. ~~COMMON~~/SPECS/ - Specification information is contained in this area. The following data are included:

TDSN	-	Temporary data set name = 'TEMP' (R*8)
DLAT	-	Lattice name. (R*8)
VLAT	-	Lattice version. (R*8)
NCASE	-	Case number.
LPRNT(M)	-	Print options.

6. COMMON/TRAK/ - This common area is used to pass tracking data between the main program, the subroutine TRACK, and the routines used to calculate the initial parameters. The following data are included:

NLP	-	Starting lattice position.
NFC	-	Starting face; this parameter is changed by the tracking routine as a neutron crosses a sub-assembly boundary.
NSR	-	Starting sub-region; this parameter is changed by the tracking routine as it crosses sub-region boundaries.
I	-	Starting energy group.
(XO,YO,ZO)	-	Starting position of neutron; it is changed by the tracking routine to the coordinates of the point where the neutron crosses a sub-assembly boundary. If it lies on a face, the tracking routine translates and rotates the point to refer to the coordinate system of the cell being entered.
(THO,PHO)	-	Starting direction of neutron; THO is rotated by tracking routine when face is crossed.
STHO	-	sin (THO)
CTHO	-	cos (THO)
SPHO	-	sin (PHO)
CPHO	-	cos (PHO)
(X1,Y1,Z1)	-	Coordinates of the last collision or of the position where the last sub-assembly boundary was crossed. The tracking routine rotates and translates it so that it is always relative to the cell where the collision is made.
TH1	-	Direction of neutron traveling from the point (X1,Y1,Z1).
KGØX	-	Number of track lengths.
KGØXX	-	Maximum number of track lengths.
NSRT(N)	-	Sub-region of track length n.
DXT(N)	-	Length of track length n in x-y plane (centimeters).

7. ~~COMMON~~/REFPNT/ - This area contains the origins of the temporary data blocks in DATA. They are as follows:

KGØRG	-	'GEØT' origin
KAØRG	-	'XABS' origin
KTØRG	-	'XTØT' origin
KFØRG	-	'FLUX' origin
KQØRG	-	'QCEL' origin
KPØRG	-	'PRØD' origin
KSØ	-	'ISCT' origin
KFØ	-	'IFSP' origin
KCØRG	-	'CØMP' origin
KXØRG	-	'XSEC' origin
KSØRG	-	'SMAT' origin

8. ~~COMMON~~/ERR/ - This common is used to pass variables that can be printed by the error routine to help determine the cause of an error stop. It includes the following:

R	-	Random number used to start current history. (R*8)
IFLAG	-	Error flag
	= 0	- no print
	= 1	- print track edit for collisions between NCØLL and NCØLH
NSIX	-	Number of scattering isotopes
NIRX	-	Number of isotope/sub-region pairs.
NIS	-	Isotope number.
JA,JB	-	Final group numbers.
KMT	-	Pointer to material in block ISCT.
KPT	-	Pointer to transfer data in block ISCT.
KPC	-	Pointer to Chi data in block IFSP.
NCØL	-	Counter of the number of collisions in current history.
NCØLL } NCØLH }	-	If error flag is on, print track edit for all collisions NCØL such that NCØLL ≤ NCØL ≤ NCØLH.
IXI	-	Integer random number. (I*2)

3.3 Subprograms

Additional comments on the subroutines of STERIL are provided in this section.

1. MAIN - The basic Monte Carlo calculations are done in the main program.

2. Subroutine SETUP.

This routine reads in all data from LIDS and calls the appropriate subroutine to construct the intermediate data sets. Also, the variables in COMMON/TEMP/ are initialized.

3. Subroutine SPRØD.

This routine sets up the production distribution and stores it in the PRØD block.

4. Subroutine FISPRØ.

The block IFSP is set up and Chi probabilities are stored.

5. Subroutine FISSUP.

This routine stores the fission probabilities in the block IFSP.

6. Subroutine STEP.

The scattering, transfer, and elastic probabilities and macroscopic total and absorption cross sections are set up in blocks ISCT, XTØT, and XABS.

7. Subroutine RESABS.

Resonance absorption rates by cell are set up in the block RABS.

8. Function BINSCH(IXI,LEN,DATA(K),L,NEL).

This library function does a binary search on the probability tables. The arguments are as follows:

- IXI - random number to be searched for in tables
- LEN - length in bytes of the random number
- DATA(K) - array to be searched
- L - number of bytes in the elements of the array to be searched
- NEL - number of elements to be searched.

Title	Module STERIL	Vol.	Page
		IV	J.1.66
Rev.		Date	12/78

The value of the function is the position of the random number in the tables. If the value is not found, it is flagged with a negative sign and is the position of the first entry in the table greater than the random number. The value 0 is returned in case of an error.

9. Function FLTRNF (0).

The value of the function is a random number on the interval (0,1).

10. Subroutine RANDIN(R).

Store the last 6 bytes of R (an 8-byte word) as a 48 bit starting random number. Care should be taken to insure that R is unnormalized.

11. Subroutine RANDOUT(R).

Store the current random number in R. R may be treated as a REAL*8 number (unnormalized) or the last 6 bytes as a 48 bit integer.

12. Subroutine FISR.

FISR selects the starting sub-region, isotope, lattice position, energy, and direction. These values are passed to the main program through common /TRAK/. It is called once at the start of each history.

13. Subroutine USSR.

This routine is a uniform spatial source routine. It selects a starting position in the sub-region from a uniform distribution. A detailed description of this routine is included in manuscript J.2.

14. Subroutine TRACK(L,IERR).

This is a general purpose tracking routine. If $L = 1, 2, 3$, the tracking begins at (X_0, Y_0, Z_0) with direction (TH_0, PH_0) and is followed until it escapes sub-assembly NSA. If $L = 1$, the starting point is on face NFC of an adjacent sub-assembly. Coordinates and direction refer to the adjacent sub-assembly. If $L = 2$, the starting point is on a side of sub-region NSR. If $L = 3$, the starting point is inside sub-region NSR. After tracking is complete, tables of track segments $DX(K)$ and associated sub-region numbers $NSRT(K)$ are returned through common /TRAK/. If error occurs, $IERR = 1$. Manuscript J.2 includes a detailed description of the tracking routine.

15. Subroutine PRØDUP.

This routine updates the production distribution.

16. Subroutine CRF.

This routine computes region flux.

17. Subroutine MØMENT(WT,WA,X,Y,Z,EJ,L)

This routine calculates mean square distances along the x,y, and z axes that a neutron travels between some starting point and absorption or slowing-down below some given energy. The arguments are:

WT	-	Post-collision total weight.
WA	-	Weight absorbed.
X,Y,Z	-	Collision point coordinates.
EJ	-	Post-collision energy or source energy.
L	-	Entry flag -
		= 1 start of calculation; clear tables
		= 2 start of history entry
		= 3 collision entry
		= 4 calculate results and print.

18. Subroutine ERRØR(I,*).

This routine handles all errors that call for additional printing to determine reason for error. Its arguments are an index I which indicates which error occurred and a statement number to which control should be returned if a tracking edit is necessary. Presently, the error routine handles tracking errors or errors in searching probability tables. If a tracking error occurs, the error flag IFLAG is set to one and the history is re-run with a tracking edit for the collisions immediately preceding the error call. All errors handled by this routine generate a message with a similar format:

STERIL.NAME, STØP I, MESSAGE

where NAME is the name of the routine in which the error occurred and I is an index for error control. Table I indicates the values of I and corresponding messages.

JOSHUA DPSTM-500	Title Module STERIL	Vol. IV	Page J.1.68
		Rev.	Date 12/7

TABLE I
ERROR STOPS FROM SUBROUTINE ERROR

<u>I</u>	<u>Message</u>
1-2	Binary search of table ISCT failed. Pointers are printed and the table ISCT is dumped. This type of error can result from negative arguments in the routine BINSCH or from negative elements in table you are trying to search. Error occurred in sub-routine MAIN.
3-4	Binary search of table IFSP failed. Pointers are printed and the table IFSP is dumped. Error occurs in subroutine FISR.
5	Error occurs in tracking routine when called from subroutine MAIN. No track lengths were generated. The history is re-run with a detailed track edit.
6	Error occurs in TRACK when the face NFC could not be located in geometry table. The face number is printed, geometry tables are dumped, and history is re-run with detailed track edit.
7	Error occurs in TRACK when lattice position number NLP could not be found in the geometry tables. The lattice position NLP is printed, geometry tables are dumped, and the history is re-run with a detailed track edit.
8	Error occurs in TRACK when the point (XO,YO) could not be located on a face of the sub-assembly. Tracking variables are printed and the history is re-run with detailed track edit.
9	Error occurs in TRACK when sub-region NSR could not be located in geometry tables. The sub-region number is printed, geometry tables are dumped, and the history is re-run with detailed track edit.
10	Error occurs in TRACK when the point (XO,YO) cannot be located in sub-region NSR. Tracking variables are printed, and the history is re-run with a detailed track edit.
11	Material in sub-region NSR could not be found in ISCT block. Pointers and variables are printed and history is re-run.

2 pages missing from origiinal document

Error stops are handled by the subroutine ERRØR or the JOSHUA Operating System. They are summarized in Tables I and III.

JOSHUA DPSTM-500	Title Module STERIL	Vol.	Page
		IV	J.1.72
		Rev.	Date
			12/76

TABLE III

JOSHUA ERROR STOP MESSAGES

MESSAGE/EXPLANATION

STERIL.MAIN, INPUT RECORD MISSING.

The record INPUT.STERIL is missing.

STERIL.MAIN, GRPS RECORD MISSING.

The group structure record is not in the LIDS.

STERIL.SETUP, DATA BLOCK MISSING.

One of records from LIDS is missing.

STERIL.SPRØD, DATA ØVERFLØW.

Core size specified in procedure step is too small.

STERIL.SPRØD, ISØTØPE NØT IN CØMP BLØCK.

An isotope in the RAHAB record 'LATTICE'.DLAT.VLAT.
'FISSION' is not in the composition block.

STERIL.SPRØD, ISØTØPE NØT IN XSEC BLØCK.

An isotope from the composition block has no cross
sections.

STERIL.SPRØD, TØØ MANY ISØTØPE/SUBREGION PAIRS.

Number of isotope/subregion pairs exceeds maximum
allowable.

STERIL.SPRØD, TØØ MANY FISSIONABLE ISØTØPES.

Number of fissionable isotopes exceeds 50.

STERIL.FISPRØ, CHI BLØCK MISSING.

Fission spectrum is missing from the LIDS.

STERIL.FISPRØ, DATA ARRAY ØVERFLØW.

Core size specified in procedure step is too small.

STERIL.FISPRØ, ISØTØPE NØT IN CHI BLØCK.

A fissionable isotope has no fission spectrum.

TABLE III (CONT.)

STERIL.STEP, DATA ARRAY ØVERFLØW.

Core size specified in procedure step is too small.

STERIL.STEP, ISØTØPE NØT IN XSEC BLØCK.

Isotope has no cross sections.

STERIL.STEP, ISØTØPE NØT IN CØMP BLØCK.

Isotope in scattering matrix is not in the composition block.

STERIL.STEP, TØØ MANY ISØTØPES IN MATERIAL.

Number of isotopes in material exceeds 50.

STERIL.RESABS, REGION NØT IN GEØM TABLE.

Region number cannot be found in the geometry table.

STERIL.FISSUP, SUBREGION NØT IN PRØD BLØCK.

Self-explanatory.

STERIL.FISSUP, ISØTØPE NØT IN PRØD BLØCK.

Self-explanatory.

STERIL.USSR, BAD VALUE OF NSR.

Starting sub-region is not in the geometry block.

STERIL.CRF, DATA ARRAY ØVERFLØW.

Core size specified in procedure step is too small.

STERIL.CRF, PØINTER MISSING.

The CØMP, FLUX, or GEØT block is not in LIDS.

STERIL.TRACK, NØ SIDE FØUND.

Entry 2, the starting point (X0,Y0) is not on the side of sub-region NSR.

STERIL.TRACK, TØØ MANY TRACK LENGTHS.

Number of track lengths exceeds 200.

DISTRIBUTION

1. F. P. Baranowski	51. File Point
2. J. Lewellyn	52. R. M. Satterfield
5. T. B. Niland	53. M. B. Stroud
6. R. L. Smith	54. C. E. Parker
8. C. W. Farmer, Jr.	55. W. H. Brotherton, Jr.
9. C. A. McCharen	56. J. P. Church
10. J. W. Croach	57. F. D. Knight
11. F. E. Kruesi	58. F. D. King
13. R. F. Gambill, Jr.	59. W. V. Baxter
14. R. L. Hellens, CEI	60. R. J. Pryor
15. L. W. Patrick	61. M. R. Buckner
17. O. A. Towler	62. C. K. Paulson
18. E. O. Kiger	63. B. Toppel, ANL
19. D. A. Ward	64. Richard Ehrlich, KAPL
20. P. S. Tennant	65. D. Vondy, ORNL
21. J. N. Wilson	66. C. W. Craven, Jr., ORNL
23. Kent F. Hansen, MIT	67. R. L. Howell, ORNL
24. J. W. Wade	68. P. Michael, BNL
25. H. C. Honeck	69. Neal Carter, BMI
26. L. M. Arnett	70. G. H. Best, LASL
27. W. R. Hartshorn	71. R. A. Dannels, WAPD
28. J. C. Jensen	72. D. Roy, B&W
29. R. S. Wood	73. P. G. Aline, GE
30. C. E. Bailey	74. B. Hutchens, GE
31. D. R. Finch	75. Alleen Thompson, GE
32. W. H. Hightower	76. H. Alter, AI
33. J. H. Maddox	77. R. F. DeAngelis, GU
34. R. C. Howard	78. Martin Becker, RPD
35. J. W. Stewart	79-80. D. S. Craig, AECL
36. B. P. Compton	82. George Minton, PWR
37. W. E. Graves	83. D. E. Barnes
38. D. A. Sharp	84. M. M. Anderson
39. F. J. McCrosson	85. H. R. Reeve
40. P. L. Ames	86. R. W. Guy, TN
42. J. E. Beach	87. H. L. Dodds
43. R. M. Wallace	89. Mrs. Margaret Butler, ANL
44. M. L. Hyder	91. Bruce M. Beardsley, IN
45. F. R. Field	94. B. G. Gingery
46. D. H. Knoebel	95. L. R. Chandler
47. I. M. Macafee	98. David S. Selengut, KAPL
48. C. E. Parker	99. TIS Record
49. J. R. Taylor	100. Vital Records
50. Jerry Robertson	

TIS File

3, 4, 7, 12, 16, 22, 41, 81,
88, 90, 92-93, 96-97.