NASA TECHNICAL NOTE

NASA TN D-5906

c. 1

NASA TN D-5906

PERTRAN - A TRANSPORT-PERTURBATION PROGRAM

by John L. Anderson

Lewis Research Center
Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • JULY 1970

1. Report No.
NASA TN D-5906

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
PERTRAN - A TRANSPORT-PERTURBATION PROGRAM

5. Report Date
July 1970

6. Performing Organization Code

7. Author(s)
John L. Anderson

8. Performing Organization Report No.
E-5359

9. Performing Organization Name and Address

Lewis Research Center
National Aeronautics and Space Administration
Cleveland, Ohio 44135

10. Work Unit No.
120-27

11. Contract or Grant No.

12. Sponsoring Agency Name and Address

National Aeronautics and Space Administration
Washington, D.C. 20546

13. Type of Report and Period Covered

Technical Note

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

PERTRAN is written to be used primarily with the two-dimensional transport program TDSN. The equations of first-order transport perturbation theory are cast into three discrete forms consistent with the P-1, transport-corrected P-0, and diffusion approximations. The FORTRAN IV program calculates the neutron lifetime and effective delayed neutron fraction and the reactivity contributions of various reactions for each of the three approximations. Sample problems compare the three perturbation approximations with two-dimensional transport spatial calculations. Input instructions and listings of the program and a sample problem are provided.

17. Key Words *(Suggested by Author(s))*
Perturbation
Computer program
Nuclear reactor
Reactivity

18. Distribution Statement
Unclassified - unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	65	$3.00

*For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151

PERTRAN - A TRANSPORT-PERTURBATION PROGRAM

by John L. Anderson

Lewis Research Center

SUMMARY

PERTRAN is written to be used primarily with the two-dimensional transport program TDSN. The principal steps in the derivation of the equations of first-order transport perturbation theory from the Boltzmann transport equation are given. These equations are cast into three discrete forms consistent with the P-1, transport-corrected P-0, and diffusion approximations to transport theory. The equations have been incorporated into a FORTRAN IV program which will calculate the neutron lifetime, the effective delayed neutron fraction, and the reactivity contributions of various reactions for the three approximations.

One advantage of having three approximations is that by comparing the various approximations and their dependence on parameters such as mesh spacing one may choose an approximation for which the associated transport calculations require the least computer storage and computational time. Sample problems compare the three perturbation approximations with two-dimensional transport spatial calculations. The input and output features of the program are described, and listings of the program and a sample problem are provided.

INTRODUCTION

The design of a nuclear reactor requires many computer calculations to determine the nuclear characteristics. These calculations are usually made by one of the several multidimensional, multigroup transport and diffusion theory programs that are available. However, these programs often require large amounts of computational time. Furthermore, the number of calculations needed is large because many design alternatives must be compared. Also, the effect of manufacturing tolerances and experimental uncertainty in the input cross sections must be determined.

However, these design problems can be solved with greatly reduced computational time through perturbation theory. Somewhat fewer transport calculations, each of which is an iterative time-consuming process, are needed when the perturbation method is used. In using the method one first chooses a certain assembly as an unperturbed base. Then by using the transport theory solutions for this single unperturbed assembly one can determine the reactivity effect of small perturbations in the assembly. Perturbation theory for fast-neutron critical systems is described in reference 1.

This report describes the computer program PERTRAN, which uses first-order transport perturbation theory to compute reactivity, neutron life-time, and effective delayed neutron fractions. PERTRAN is written to be used primarily with the TDSN transport program (ref. 3). Many perturbation programs have been written for both diffusion and transport theory (e.g., ref. 2). PERTRAN differs from other transport perturbation programs in that it provides three approximations to the perturbation calculations. The best approximation uses P-1 cross sections and unperturbed real and adjoint fluxes and currents from the transport theory spatial calculation. The next best approximation is provided by transport-corrected P-0 cross sections and fluxes and currents. The diffusion approximation, which is least accurate, uses the transport-corrected cross sections and the fluxes from a transport calculation.

These approximations allow one to choose the accuracy of the transport calculations to be commensurate with the importance of the perturbation.

SYMBOLS

B	buckling factor
E	neutron energy
H	buckling dimension
$\hat{i},\hat{j},\hat{k}$	unit vectors in the x-, y-, and z-directions for rectangular coordinates (fig. 1)
$\vec{J}(\vec{r},E)$	neutron current (directional) at position $\vec{r}$ with energy E
k	neutron multiplication factor; 1/k is the eigenvalue of Boltzmann equation
l	neutron lifetime
N_o	atomic density, atoms/(b)(cm)
NG	number of discrete energy groups
NIJ	number of discrete volume elements (V)
$P_l(\mu)$	Legendre polynomial
$P_l^m(\mu)$	associated Legendre polynomial
P-0	zero-order ($l = 0$) cross sections

P-0*	transport-corrected P-0 cross sections
P-1	first-order (l = 1) cross sections
ΔQ	incremental change in quantity Q
$\vec{r}$	position variable representing three-dimensional coordinates of neutron
V	volume element, cm^3
$x(E)$	fission spectrum; probability that neutron released through fission will have a particular energy E: $\int x(E)dE = 1$
β_j	delayed neutron fraction of jth delayed group
γ	extrapolation distance constant (0.71045608)
Σ	macroscopic cross section, cm^{-1}
θ	angle between $\hat{k}$ and $\hat{\Omega}$ (fig. 1)
μ	cosine of θ
μ_o	angle between $\hat{\Omega}$ and $\hat{\Omega}'$
$\nu(E)$	average number of neutrons with energy E released per fission
$\Phi(\vec{r}, E, \hat{\Omega})$	directional neutron flux; number of neutrons of energy E at position $\vec{r}$ flowing through a unit solid angle and unit area in direction $\hat{\Omega}$
$\varphi(\vec{r}, E)$	scalar (nondirectional) neutron flux of energy E at position $\vec{r}$
ψ	angle between $\hat{i}$ and projection of $\hat{\Omega}$ in the plane perpendicular to $\hat{k}$
$\hat{\Omega}$	direction of neutron flow at position $\hat{r}$ (fig. 1)

Subscripts:

g	index of energy groups
i	index of spatial position
l	order of Legendre polynomial

Superscripts:

p	perturbed quantity, $Q^p \equiv Q + \Delta Q$
0, 1	order of Legendre polynomial for cross-section expansion
†	adjoint quantity

Cross-section definitions:

(Macroscopic definitions are presented in discrete form for energy group g; the equivalent continuous form is shown for the second definition only.)

D_g diffusion coefficient, $D_g = \left(\dfrac{1}{3\Sigma_{tr_g}}\right)$; used in $\vec{J}_g = -D_g \vec{\nabla}\varphi_g$ (Fick's law)

Σ_{a_g} absorption (includes capture and fission), $\Sigma_a(E)$

Σ_{f_g} fission

$\Sigma^{N2N}_{g\to g'}$ n → 2n scattering from group g to group g'

$\Sigma^{(0)}_{g\to g'}$ P-0 scattering from group g to group g' (includes elastic, inelastic, and twice the n → 2n scattering)

$\Sigma^{(1)}_{g\to g'}$ P-1 scattering from group g to group g'

$\Sigma^{(0)}_{gg}$ within group scattering (used to provide neutron balance),

$$\Sigma^{(0)}_{gg} \equiv \left[\Sigma_{t_g} \text{ or } \Sigma_{tr_g}\right] - \Sigma_{a_g} - \Sigma_{r_g} \quad \left\{\begin{matrix} \Sigma_t \text{ for P-1} \\ \\ \Sigma_{tr} \text{ for P-0}^* \end{matrix}\right.$$

Σ_{lk_g} transverse leakage (buckling-loss) cross section

Σ_{r_g} removal or outscatter, $\Sigma_{r_g} = \sum\limits_{\substack{g' \\ g\neq 0}} \left(\Sigma^{(0)}_{g\to g'} - \Sigma^{N2N}_{g\to g'}\right)$

$\Sigma_{s_{g\to g'}}$ total scattering, $\Sigma_{s_{g\to g'}} = \Sigma^{0}_{g\to g'} + \Sigma^{1}_{g\to g'}$

Σ_{t_g} total, $\Sigma_{t_g} = \Sigma_{a_g} + \Sigma_{r_g} + \Sigma^{(0)}_{gg}$

Σ_{tr_g} transport, $\Sigma_{tr_g} = \Sigma_{t_g} - \dfrac{\sum\limits_{g} \left(\Sigma^{(1)}_{g\to g'} J_{g'}\right)}{J_g}$

TRANSPORT PERTURBATION EQUATIONS

The time independent Boltzmann transport equation may be written

$$[\hat{\Omega} \cdot \vec{\nabla} + \Sigma_t(\vec{r},E)]\Phi(\vec{r},E,\hat{\Omega}) = \frac{1}{4\pi k} \iint dE'\, d\hat{\Omega}'[\Phi(\vec{r},E',\hat{\Omega}')\nu(\vec{r},E')\Sigma_f(\vec{r},E')x(E)]$$

$$+ \iint dE'\, d\hat{\Omega}'\, \Phi(\vec{r},E,\hat{\Omega}')\Sigma_s(\vec{r},E' \to E,\hat{\Omega}' \to \hat{\Omega}) \qquad (1)$$

The continuous variables $\vec{r}$, E, and $\hat{\Omega}$ represent the dependence on position, energy, and direction; $\Phi(\vec{r},E,\hat{\Omega})$ is the real flux.

Two other equations, nearly identical to equation (1), are needed to develop the perturbation equations. One equation provides the adjoint flux $\Phi^{\dagger}(\vec{r},E,\hat{\Omega})$:

$$[-\hat{\Omega} \cdot \vec{\nabla} + \Sigma_t(\vec{r},E)]\Phi^{\dagger}(\vec{r},E,\hat{\Omega}) = \frac{1}{4\pi k^{\dagger}} \iint dE'\, d\hat{\Omega}'[\Phi^{\dagger}(\vec{r},E',\hat{\Omega}')\nu(\vec{r},E)\Sigma_f(\vec{r},E)x(E')]$$

$$+ \iint dE'\, d\hat{\Omega}'\, \Phi^{\dagger}(\vec{r},E',\hat{\Omega}')\Sigma_s(\vec{r},E \to E',\hat{\Omega} \to \hat{\Omega}') \qquad (2)$$

where $k^{\dagger} = k$. The other equation provides the perturbed flux $\Phi^p(\vec{r},E,\hat{\Omega})$:

$$[\hat{\Omega} \cdot \vec{\nabla} + \Sigma_t^p(\vec{r},E)]\Phi^p(\vec{r},E,\hat{\Omega}) = \frac{1}{4\pi k^p} \iint dE'\, d\hat{\Omega}'\, \Phi^p(\vec{r},E',\hat{\Omega}')$$

$$\cdot\ [\nu(\vec{r},E')\Sigma_f(\vec{r},E')]^p\, x(E) + \iint dE'\, d\hat{\Omega}'\, \Phi^p(\vec{r},E',\hat{\Omega}')\Sigma_s^p(\vec{r},E' \to E,\hat{\Omega}' \to \Omega) \qquad (3)$$

where each perturbed quantity Q^p is equal to the unperturbed quantity Q plus its perturbation increment ΔQ (not necessarily small).

Derivation

The following derivation of the transport perturbation equations is taken from reference 2. Some intermediate steps omitted in this report may be found in that reference.

The transport perturbation equations may be obtained by multiplying equation (2) by $\Phi^p(\vec{r},E,\hat{\Omega})$ and equation (3) by $\Phi^{\dagger}(\vec{r},E,\hat{\Omega})$, integrating the two equations over all space, energy, and direction, and then subtracting the resulting equations. The exact equation for an eigenvalue increment resulting from a perturbation is then

$$\frac{1}{k^p} - \frac{1}{k} \approx \Delta\left(\frac{1}{k}\right) = \frac{T + F + S}{p} \tag{4}$$

where

$$T = \iiint d\vec{r}\, dE\, d\hat{\Omega}\; \Delta\Sigma_t(\vec{r}, E)\Phi^{\dagger}(\vec{r}, E, \hat{\Omega})\Phi^{p}(\vec{r}, E, \hat{\Omega}) \tag{5}$$

$$F \approx -\frac{1}{4\pi k} \iiint d\vec{r}\, dE\, dE'\; \Delta[\nu(\vec{r}, E')\Sigma_f(\vec{r}, E')]\; x(E)\Phi^{\dagger}(\vec{r}, E)\Phi^{p}(\vec{r}, E') \tag{6}$$

$$S = -\iiiint\!\!\int d\vec{r}\, dE\, dE'\, d\hat{\Omega}\, d\hat{\Omega}'\; \Delta[\Sigma_s(\vec{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega})]\Phi^{\dagger}(\vec{r}, E, \hat{\Omega})\Phi^{p}(\vec{r}, E', \hat{\Omega}') \tag{7}$$

$$P = \frac{1}{4\pi} \iiint d\vec{r}\, dE\, dE'[\nu(\vec{r}, E')\Sigma_f(\vec{r}, E)]^{p}\; x(E)\varphi^{\dagger}(\vec{r}, E)\varphi^{p}(\vec{r}, E') \tag{8}$$

Equation (4) determines the eigenvalue change resulting from a perturbation. The reactivity, however it is defined, can be subsequently obtained from the eigenvalue change.

A first-order approximation to perturbation theory is obtained by replacing the perturbed flux Φ^p by the unperturbed flux Φ. This now restricts the perturbation to one which causes only a negligible change in the unperturbed flux.

The unperturbed angular flux is then expanded in terms of spherical harmonics before performing the solid angle integrations in equations (5) and (7):

$$\Phi(\vec{r}, E, \hat{\Omega}) = \frac{1}{4\pi} \sum_{l=0}^{\infty} \left[\Phi_l P_l{'}(\mu) + \sum_{m=1}^{l} (\varphi_l^{m} \sin m\psi + \varphi_l^{-m} \cos m\psi) P_l^{m}(\mu)\right] \tag{9}$$

where

$$\hat{\Omega} = \hat{i} \sin\theta \cos\psi + \hat{j} \sin\theta \sin\psi + \hat{k} \cos\theta$$

and

$$\mu = \cos\theta$$

(see fig. 1). If terms are retained only through the P-1 approximation,

$$\Phi(\vec{r}, E, \hat{\Omega}) \cong \frac{1}{4\pi} [\varphi(\vec{r}, E) + 3\hat{\Omega} \cdot \vec{J}(\vec{r}, E)] \tag{10}$$

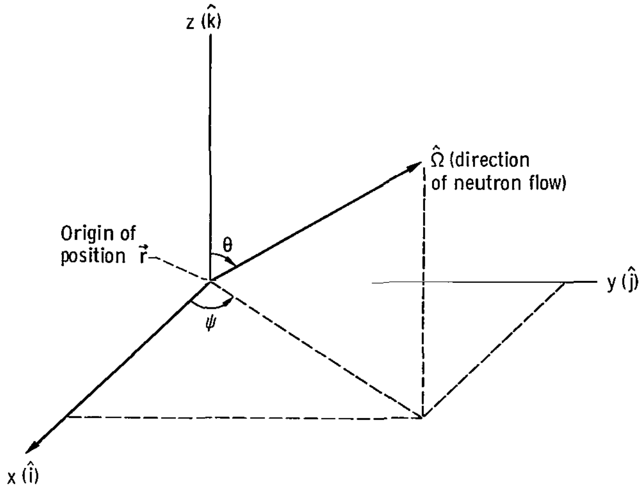
z ($\hat{k}$)
$\hat{\Omega}$ (direction of neutron flow)
Origin of position $\vec{r}$
θ
y ($\hat{j}$)
ψ
x ($\hat{i}$)

Figure 1. - Coordinate systems.

where $\varphi(\vec{r}, E)$ is the neutron scalar flux and $\vec{J}(\vec{r}, E)$ is the neutron current. A similar expansion for the adjoint yields

$$\Phi^{\dagger}(\vec{r}, E, \hat{\Omega}) \cong \frac{1}{4\pi} [\varphi^{\dagger}(\vec{r}, E) - 3\hat{\Omega} \cdot \vec{J}^{\dagger}(\vec{r}, E)] \tag{11}$$

Furthermore, the scattering is assumed to be dependent only on the angle between directions $\hat{\Omega}$ and $\hat{\Omega}'$, and the incremental scattering cross section is expanded in terms of Legendre polynomials. The expansion is truncated after the P-1 term to yield

$$\Delta\Sigma_s(\vec{r}, E' \rightarrow E, \hat{\Omega}' \rightarrow \Omega) \cong \frac{1}{4\pi} \left[\Delta\Sigma^{(0)}(\vec{r}, E' \rightarrow E) + 3\mu_0 \, \Delta\Sigma^{(1)}(\vec{r}, E' \rightarrow E)\right] \tag{12}$$

Substituting equations (10) and (11) in equation (5) and equations (10) to (12) in equation (7) yields after integrating over the solid angle

$$T = \iint dr \, dE \, \frac{1}{4\pi} \Delta\Sigma_t(\vec{r}, E)[\varphi(\vec{r}, E)\varphi^{\dagger}(\vec{r}, E) - 3\vec{J}(\vec{r}, E) \cdot \vec{J}(\vec{r}, E)] \tag{13}$$

and

$$S = \iiint d\vec{r} \, dE \, dE' \, \frac{1}{4\pi} \left[\Delta\Sigma_s^{(0)}(\vec{r}, E' \rightarrow E)\varphi(\vec{r}, E')\varphi^{\dagger}(\vec{r}, E) \right.$$

$$\left. - 3\Delta\Sigma_s^{(1)}(\vec{r}, E' \rightarrow E)J(\vec{r}, E') \cdot J(\vec{r}, E)\right] \tag{14}$$

Discrete Form

The components of equation (4), which are equations (6), (8), (13), and (14), may be written in discrete form by replacing the continuous variables $\vec{r}$ and E by the discrete indices i and g. Hence, the equations may now be written as summations over volume increments and energy groups:

$$T = \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \frac{1}{4\pi} \Delta\Sigma_{t_{gi}} \left[\varphi_g \varphi_g^\dagger - 3\vec{J}_g \cdot \vec{J}_g^\dagger\right]_i V_i \tag{15}$$

$$F = -\sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \frac{1}{4\pi} \frac{1}{k} \Delta \left[(\nu\Sigma_f)_{g'}\right]_i x_g \varphi_{g'i} \varphi_{gi}^\dagger V_i \tag{16}$$

$$S = -\sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \frac{1}{4\pi} \left[\Delta\Sigma_{g'\to g}^{(0)} \varphi_{g'} \varphi_g^\dagger - 3\Delta\Sigma_{g'\to g}^{(1)} \vec{J}_{g'} \cdot \vec{J}_{g'}^\dagger\right]_i V_i \tag{17}$$

$$P = \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \frac{1}{4\pi} \left[\nu\Sigma_{fg'}\right]_i^p x_g \varphi_{g'i} \varphi_{gi}^\dagger V_i \tag{18}$$

This last equation is the product of the perturbed real power and the equivalent quantity from the adjoint calculation (an adjoint power).

Approximations

Equations (15) to (18) constitute the P-1 approximation to first-order transport perturbation theory. Further approximations to equations (15) and (17) can provide a transport-corrected P-0 and a diffusion approximation. But before proceeding to these approximations, it is useful to isolate the reactivity contributions due to various reactions and regroup the P-1 equations (eqs. (15) to (17)) as reactivity sources or losses.

Transport - P-1

The incremental total cross section may be written

$$\Delta\Sigma_{t_g} = \Delta\Sigma_{a_g} + \sum_{g'}^{NG} \Delta\Sigma^{(0)}_{g\to g'} = \Delta\Sigma_{a_g} + \Delta\Sigma_{r_g} + \Delta\Sigma^{(0)}_{gg} \qquad (19)$$

The reactivity contributions are as follows:

Source:

$$(\text{Fission})\ F = - \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \frac{1}{4\pi k} \Delta\left[\left(\nu\Sigma_f\right)_{g'}\right]_i x_g \left[\varphi_{g'}\varphi^\dagger_g\right]_i V_i \qquad (20)$$

$$(\text{Scattering})\ S = - \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \left[\Delta\Sigma^{(0)}_{g\to g'}\varphi_{gi}\varphi^\dagger_{g'} - 3\Delta\Sigma^{(1)}_{g\to g'}\vec{J}_g \cdot \vec{J}^\dagger_{g'}\right]_i V_i \qquad (21)$$

Loss:

$$(\text{Absorption})\ A = \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \Delta\Sigma_{a_{gi}} \left[\varphi_g\varphi^\dagger_g - 3\vec{J}_g \cdot \vec{J}^\dagger_g\right]_i V_i \qquad (22)$$

$$(\text{Removal})\ R = \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \sum_{g'=1}^{NG} \left(\Delta\Sigma^{(0)}_{g\to g'}\right)_i \left[\varphi_g\varphi^\dagger_g - 3J_g \cdot J^\dagger_g\right]_i V_i \qquad (23)$$

By collecting the current weighted terms in equations (20) to (23) we can estimate the contribution of the nontransverse leakage out of the system:

$$(\text{Leakage})\ L = - \frac{3}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \left[\Delta\Sigma_{t_g}\vec{J}_g \cdot \vec{J}^\dagger_g - \sum_{g'=1}^{NG} \Delta\Sigma^{(1)}_{g'\to g}\vec{J}_{g'} \cdot \vec{J}^\dagger_g\right]_i V_i \qquad (24)$$

Note that the $\Delta\Sigma_t$ term includes the within group scattering; it does not cancel when current weighted as it does when flux weighted (eqs. (21) and (23)).

There is another reactivity loss mechanism which has not been accounted for - the transverse leakage or buckling loss. The cross section for the buckling loss, as used in reference 3, is

$$\Sigma_{1k_g} = \frac{B^2 \Sigma_{tr_g}}{\left(\Sigma_{tr_g} H + 2\gamma\right)^2} \tag{25}$$

The buckling factor B is $\pi/\sqrt{3}$ for plane boundaries; for cylindrical boundaries, B is $2(2.405/\sqrt{3})$ and the buckling dimension H is the diameter. Within PERTRAN the increment $\Delta\Sigma_{1k_g}$ may be obtained by changing the transport cross section Σ_{tr_g} or the buckling dimension H.

The contribution to the eigenvalue increment is

$$TL = \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \frac{1}{4\pi} \Delta\Sigma_{1k_{g_i}} \left[\varphi_g \varphi_g^{\dagger} - 3\vec{J}_g \cdot \vec{J}_g^{\dagger}\right]_i V_i \tag{26}$$

which is identical in form to the equation of the total cross section (eq. (15)). If transverse leakage does occur, its contribution to the eigenvalue increment must be included in equation (4), which becomes

$$\Delta\left(\frac{1}{k}\right) = \frac{F + S + A + R + TL}{P} \tag{27}$$

Transport - Transport-Corrected P-0 (P-0*)

If equations (15) and (17) are combined,

$$
\begin{aligned}
T + S &= \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \left[\Delta\Sigma_{tg} \left(\varphi_g \varphi_g - 3\vec{J}_g \cdot \vec{J}_g^{\dagger} \right) - \sum_{g'=1}^{NG} \Delta\Sigma^{(0)}_{g' \to g} \varphi_{g'} \varphi_g^{\dagger} \right. \\
&\qquad \left. + 3 \sum_{g'=1}^{NG} \Delta\Sigma^{(1)}_{g' \to g} \vec{J}_{g'} \cdot \vec{J}_g^{\dagger} \right]_i V_i \\
&= \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \left[\left(-3\vec{J}_g \cdot \vec{J}_g^{\dagger} \right) \left(\Delta\Sigma_{t_g} - \frac{\sum_{g'=1}^{NG} \Delta\Sigma^{(1)}_{g' \to g} \vec{J}_{g'}}{\vec{J}_g} \right) \right. \\
&\qquad \left. + \left(\Delta\Sigma_{a_g} + \Delta\Sigma_{r_g} \right) \left(\varphi_g \varphi_g^{\dagger} \right) - \sum_{g'=1}^{NG} \Delta\Sigma^{(0)}_{g' \to g} \varphi_{g'} \varphi_g^{\dagger} \right]_i V_i \\
&= \frac{1}{4\pi} \sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \left[-3\Delta\Sigma_{tr_g} \vec{J}_g \cdot \vec{J}_g^{\dagger} + \left(\Delta\Sigma_{a_g} + \Delta\Sigma_{r_g} \right) \varphi_g \varphi_g^{\dagger} \right. \\
&\qquad \left. - \sum_{g'=1}^{NG} \Delta\Sigma^{(0)}_{g' \to g} \varphi_{g'} \varphi_g^{\dagger} \right]_i V_i \qquad (28)
\end{aligned}
$$

where the incremental transport cross section is defined as

$$
\Delta\Sigma_{tr_g} = \Delta\Sigma_{t_g} - \frac{\sum_{g'=1}^{NG} \Delta\Sigma^{(1)}_{g' \to g} \vec{J}_{g'}}{\vec{J}_g} \qquad (29)
$$

Note that currents are needed from the spatial calculation in order to weight the transport contribution to the reactivity. The TDSN program calculates currents from the angular fluxes and, hence, can provide currents when using only P-0* cross sections.

Diffusion - P-0* and Fick's Law

It requires a considerable amount of computer storage to provide both fluxes and currents for a spatial calculation. One further approximation is to use the diffusion theory definition of current, that is, Fick's law:

$$\vec{J}_g = -D_g \vec{\nabla}\varphi_g \tag{30}$$

where D_g is the diffusion coefficient for the group g. With this approach the currents can be calculated within the perturbation program from the gradients of the fluxes.

The transport cross section term in equation (28) becomes

$$3\Delta\Sigma_{tr_g}\vec{J}_g \cdot \vec{J}_g^\dagger = 3D_g^2\,\Delta\Sigma_{tr_g}\,\vec{\nabla}\varphi_g \cdot \vec{\nabla}\varphi_g^\dagger \tag{31}$$

If the fluxes φ_g come from a transport calculation, then equation (31) can be written in terms of the transport cross section:

$$3\Delta\Sigma_{tr_g}\vec{J}_g \cdot \vec{J}_g^\dagger = \left[\frac{1}{3}\,\frac{\Delta\Sigma_{tr_g}}{\left(\Sigma_{tr_g}\right)^2}\right]\vec{\nabla}\varphi_g \cdot \vec{\nabla}\varphi_g^\dagger \tag{32}$$

Equation (31) can also be written in terms of the diffusion coefficient:

$$3\Delta\Sigma_{tr_g}\vec{J}_g \cdot \vec{J}_g^\dagger = \left(\frac{-\Delta D_g}{1 + \frac{\Delta D_g}{D_g}}\right)\vec{\nabla}\varphi_g \cdot \vec{\nabla}\varphi_g^\dagger \tag{33}$$

The current terms in the transverse leakage contribution (eq. (26)) are also calculated using Fick's law.

The program uses curve fitting techniques to determine the flux shape and subsequently the gradients of the fluxes. The two particular techniques are presented in appendix A as they apply to fitting the flux profile to a second-degree polynomial.

Prompt Neutron Lifetime

The lifetime (l) of prompt neutrons (from ref. 2) is given by

$$l = \frac{4\pi k}{N}\iiint d\vec{r}\; dE\; d\hat{\Omega}\left(\overline{\frac{1}{v}}\right)\Phi(\vec{r}, E, \hat{\Omega})\Phi^{\dagger}(\vec{r}, E, \hat{\Omega}) \tag{34}$$

where

$$N = \iiint d\vec{r}\; dE\; dE'\; x(E)\nu(\vec{r}, E')\Sigma_f(\vec{r}, E')\varphi(\vec{r}, E')\varphi^{\dagger}(\vec{r}, E) \tag{35}$$

and $\overline{(1/v)}$ is the spectrum averaged inverse of the neutron speed. In discrete form these equations become

$$l = \frac{k}{N}\sum_{i=1}^{NIJ}\sum_{g=1}^{NG}\left(\overline{\frac{1}{v}}\right)_g\left(\varphi_g\varphi_g^{\dagger} - 3\vec{J}_g \cdot \vec{J}_g\right)_i V_i \tag{36}$$

$$N = \sum_{i=1}^{NIJ}\sum_{g=1}^{NG}\sum_{g'=1}^{NG} x_g\left(\nu\Sigma_{f_{g'}}\right)_i\left(\varphi_{g'}\varphi_g^{\dagger}\right)_i V_i \tag{37}$$

Equation (37) for N is just the product of the unperturbed real and adjoint power - analogous to equation (18). In fact, in the absence of a perturbation to the fission cross section, the equation for the lifetime (eq. (36)) is identical to the equation for the reactivity contribution of an absorption perturbation (eq. (22) divided by eq. (18)) with $\Delta\Sigma_{a_g}$ replaced by $\overline{(1/v)}_g$.

Hence, the absorption cross section from a pure 1/v absorber can be treated simply as an absorption perturbation. The perturbation in this case must extend over the whole region that established the flux spectrum A normalization factor such as an atom density N_o can be used so that the cross sections supplied are $N_o\overline{(1/v)}$ and thus the lifetime obtained is $N_o l$.

Effective Delayed Neutron Fraction

The ratio of the effective delayed neutron fraction β_{eff} to the true delayed neutron fraction β for a delayed neutron group j is given (ref. 4) by

$$\frac{\beta_{j(eff)}}{\beta_j} = \frac{\iiint d\vec{r}\; dE\; dE'\; \nu\Sigma_f(\vec{r}, E)\varphi(\vec{r}, E)\beta_j(E')\varphi^{\dagger}(\vec{r}, E')}{\iiint d\vec{r}\; dE\; dE'\; \nu\Sigma_f(\vec{r}, E)\varphi(\vec{r}, E)x(E')\varphi^{\dagger}(\vec{r}, E')} \tag{38}$$

where $\beta_j(E)$ is the delayed neutron spectrum for delayed group j (normalized to β_j) and the denominator is simply the N of equation (35).

In multigroup notation for $\beta_{j(\text{eff})}$,

$$\beta_{j(\text{eff})} = \beta_j \left(\frac{\sum_{i=1}^{NIJ} \sum_{g=1}^{NG} \left[\left(\nu\Sigma_{f_g}\varphi_g\right)_i \sum_{g'=g_l}^{g_h} \left(\beta_{j_{g'}}\varphi^{\dagger}_{g'}\right)_i \right] V_i}{N} \right) \tag{39}$$

where g_h and g_l are the inclusive high- and low-energy groups that bound the particular delayed spectrum $\beta_{j_{g'}}$.

DISCUSSION OF SAMPLE PROBLEMS

In order to determine the accuracy of the various approximations within PERTRAN, two-dimensional spatial calculations in x-y geometry were performed with the TDSN pro-

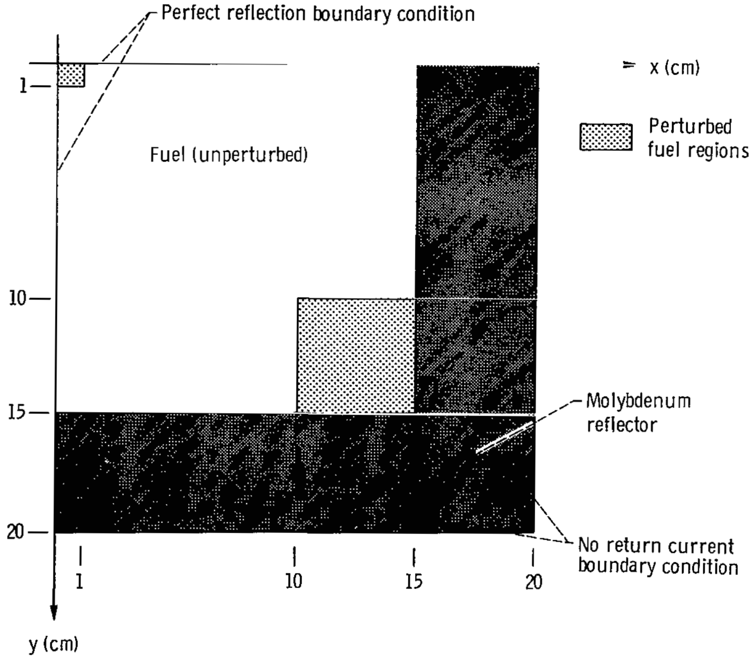

Figure 2. - Sample problem geometry with boundary conditions and material regions (height = 45 cm). Perturbed regions for both sample problems are shown.

gram. These spatial calculations provided both the change in eigenvalue $\Delta(1/k)$ resulting from a perturbation and the fluxes and currents which PERTRAN uses to determine this eigenvalue change.

A quadrant of the fast spectrum reactor model is shown in figure 2. The fueled region is 30 by 30 centimeters in cross section; the annular molybdenum reflector raises the overall cross section to 40 by 40 centimeters. The reactor height is 45 centimeters. Four group cross sections from the GAM program (ref. 5) were used; the energy group structure is shown in table I.

TABLE I. - ENERGY GROUP STRUCTURE

Group	Low energy boundary
1	[a]0.821 MeV
2	.183 MeV
3	40.87 keV
4	.414 eV

[a] Upper energy boundary is 14.9 MeV.

Two sample problems were considered: (1) the perturbed region is small, at the center of the reactor, with a small transverse leakage rate; and (2) the perturbed region is larger, with a high transverse leakage, at the corner of the reactor. The perturbation in all cases was a 1-percent increase in material density in the perturbed region. The output listing of the center-perturbed sample problem is given in appendix B.

For each of the sample problems the perturbed and unperturbed eigenvalues were calculated with TDSN using both P-1 and P-0^* cross sections. The three approximations to the change in eigenvalue $\Delta(1/k)$ that PERTRAN provides are compared to the TDSN eigenvalue changes.

In the absence of P-1 cross sections, TDSN will compute the current directly from the angular fluxes. Hence, the P-0^* spatial calculation provided fluxes and currents for the corresponding approximation in PERTRAN as well as the fluxes for the diffusion approximation in PERTRAN. Furthermore, P-1 cross sections were used only for the perturbed material - not for the entire assembly.

Center Perturbation

For the smaller center region, the perturbation changed the eigenvalue $1/k$ by about 0.005 percent (table II). All the PERTRAN approximations gave values of $\Delta(1/k)$

TABLE II. - COMPARISON OF TRANSPORT AND VARIOUS PERTURBATION CALCULATIONS

[Perturbation is 1-percent increase in material density; perturbed regions are corner and center (fig. 2); K_{eff} is nearly unity.]

Program	Quantity	Approximations							
		Corner[a]			Center (1-point)[b]			Center (3-point)[c]	
		P-1[d]	P-0[*e]	Diffusion[f]	P-1	P-0[*]	Diffusion[f]	P-0[*]	Diffusion[f]
TDSN[g]	k(regular)	0.999267	0.999508	0.999508	1.006531	1.006541	1.006541	1.006262	1.006262
	k(adjoint)	.999266	.999507	.999507	1.006531	1.006541	1.006541	1.006258	1.006258
	$k^{perturbed}$	.999609	.999854	.999854	1.006586	1.006595	1.006595	1.006317	1.006317
	Δ(1/k)	$-.342\times10^{-3}$	$-.346\times10^{-3}$	$-.346\times10^{-3}$	$-.55\times10^{-4}$	$.54\times10^{-4}$	$-.54\times10^{-4}$	$-.55\times10^{-4}$	$-.55\times10^{-4}$
PERTRAN	Δ(1/k)	-0.352×10^{-3}	-0.351×10^{-3}	-0.276×10^{-3}	-0.547×10^{-4}	-0.548×10^{-4}	-0.551×10^{-4}	-0.550×10^{-4}	-0.551×10^{-4}
	Leakage[h]	$-.999\times10^{-4}$	$-.985\times10^{-4}$	$-.199\times10^{-4}$	$-.102\times10^{-7}$	$-.102\times10^{-7}$	$-.391\times10^{-6}$	$-.140\times10^{-7}$	$-.621\times10^{-7}$

[a] Refers to perturbation of large corner region (see fig. 2).

[b] Refers to perturbation of small center region (see fig. 2) using 1 mesh point in that region.

[c] Same as footnote b but using 3 mesh points in that region.

[d] P-1 cross sections used in TDSN spatial calculations; corresponding approximation used in PERTRAN.

[e] P-0 transport corrected cross sections used in TDSN; currents obtained and used in corresponding PERTRAN approximation.

[f] Same TDSN calculation as in footnote e; TDSN currents not used in PERTRAN (KD=-1 option used).

[g] Convergence criterion used in TDSN was 10^{-6}.

[h] Leakage is in direction of calculation (not a buckling loss).

that are within 2 percent of the TDSN value. The leakage (in the direction of the calculation - in the xy-plane) was a negligible part of the total Δ(1/k) - about a thousand times smaller.

The P-1 and P-0* approximations provided nearly identical results for Δ(1/k) and the leakage. However, the leakage calculated in the diffusion approximation is a factor of 40 times greater than the P-1 and P-0* leakage. In this particular problem only one mesh interval was used in the 1-centimeter perturbed region. The same problem was also examined with three mesh intervals in the 1-centimeter region; the total number of mesh intervals remained the same (12 by 12).

The leakage in the xy-plane was strongly affected by this mesh change. The P-0* leakage increased in absolute value by about 40 percent; the diffusion leakage decreased to one-sixth of the value obtained when using only one mesh interval. These changes brought the diffusion leakage to a factor of 4 greater than the P-1 or P-0* leakage. The eigenvalue increments Δ(1/k) in the P-0* and diffusion approximations were essentially not affected by this mesh change.

In order to further improve the leakage calculation (in the xy-plane) in the diffusion approximation, one would probably have to further increase the number of mesh intervals in the perturbed region, which in this case would then require more total mesh intervals.

Corner Perturbation

The corner perturbation produced an eigenvalue change of about 0.035 percent. The P-1 and P-0* approximations were the same, about 3 percent different from the TDSN values of $\Delta(1/k)$. But the diffusion approximation to $\Delta(1/k)$ was only about 75 percent of the TDSN value. The P-1 and P-0^* leakages differed by about 2 percent; the diffusion leakage was about one-fifth of the P-1 or P-0^* leakage.

It appears that for the same accuracy the diffusion approximation will require more mesh intervals than the P-1 or P-0^* approximations. However, if nontransverse leakage is an important part of the total eigenvalue change, then the diffusion approximation may not be adequate regardless of the number of mesh intervals.

The PERTRAN input instructions and notes on the input and output features are presented in appendix C. Appendix D is a FORTRAN IV listing of the entire PERTRAN program.

CONCLUDING REMARKS

The most important characteristic of PERTRAN, compared to other perturbation programs, is that it offers three approximations to the perturbation calculation. The approximations, P-1, P-0^*, and diffusion, are in order of decreasing accuracy, but the associated transport calculations also require decreasing amounts of computer storage and computational time.

The most accurate (P-1) approximation requires the use of P-0 and P-1 cross sections and, hence, the calculation of currents as well as fluxes. The P-0^* approximation in PERTRAN requires only a P-0^* cross section set but it also requires the currents from the spatial calculation. The diffusion approximation (least accurate) uses only P-0^* cross sections and fluxes. (The fluxes, however, may come from a P-0^* spatial calculation and so they do have that accuracy.) The leakage is obtained from the flux gradient (using Fick's law). Since only fluxes from the spatial calculation are required, this approximation could treat problems with many more mesh points and groups than the approximations requiring currents.

The relative accuracy of the approximations and the dependence of accuracy on number and spacing of mesh intervals may be determined for a specific problem. Thus, an approximation which gives a certain accuracy for the least amount of computer storage and computational time for the spatial calculations may be used.

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, April 28, 1970,
120-27.

APPENDIX A

CURVE-FITTING TECHNIQUES FOR FLUX PROFILE

The first technique, the Vandermonde matrix method, which will fit a polynomial of degree n-1 through n given points, is used in PERTRAN to obtain the flux shapes and then the gradients. Three points or fluxes at three consecutive mesh intervals are used to fit a second-degree polynomial. The following derivation of the equations is conducted in generalized form in which y will represent the flux and x will represent the spatial variable. The polynomial is

$$y = a_o + a_1 x + a_2 x^2 \tag{A1}$$

which is $Xa = y$ in matrix form. The column matrices (a and y) are

$$a = \begin{pmatrix} a_o \\ a_1 \\ a_2 \end{pmatrix} \tag{A2}$$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \tag{A3}$$

and the Vandermonde matrix is

$$X = \begin{pmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \end{pmatrix} \tag{A4}$$

If X is written as the product of a lower triangular matrix L and an upper triangular matrix U, then

$$X = LU \tag{A5}$$

and the inverse of X is given by

$$X^{-1} = U^{-1}L^{-1} \tag{A6}$$

with

$$U^{-1} = \begin{pmatrix} 1 & -x_1 & x_1x_2 \\ 0 & 0 & -x_1-x_2 \\ 0 & 0 & 1 \end{pmatrix} \tag{A7}$$

and

$$L^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ \dfrac{1}{x_1 - x_2} & \dfrac{1}{x_2 - x_1} & 0 \\ \dfrac{1}{(x_1 - x_2)(x_1 - x_3)} & \dfrac{1}{(x_2 - x_1)(x_2 - x_3)} & \dfrac{1}{(x_3 - x_1)(x_3 - x_2)} \end{pmatrix} \tag{A8}$$

Now the matrix of the coefficients a_i is given by

$$\left.\begin{aligned} a &= X^{-1}y \\ a &= U^{-1}L^{-1}y \end{aligned}\right\} \tag{A9}$$

Evaluating a gives

$$L_y^{-1} = \begin{pmatrix} y_1 \\ \dfrac{y_1}{x_1 - x_2} + \dfrac{y_2}{x_2 - x_1} \\ \dfrac{y_1}{(x_1 - x_2)(x_1 - x_3)} + \dfrac{y_2}{(x_2 - x_1)(x_2 - x_3)} + \dfrac{y_3}{(x_3 - x_1)(x_3 - x_2)} \end{pmatrix} \tag{A10}$$

Let the lower element of the L_y^{-1} matrix be z. Then,

$$U^{-1}L_y^{-1} = \begin{pmatrix} y_1 - \dfrac{x_1y_1}{(x_1 - x_2)} - \dfrac{x_1y_2}{(x_2 - x_1)} + x_1x_2(z) \\ \\ \dfrac{y_1}{x_1 - x_2} + \dfrac{y_2}{x_2 - x_1} - (x_1 + x_2)(z) \\ \\ z \end{pmatrix} \tag{A11}$$

This is the matrix of the coefficients a_i.

The gradient of y at any location x_j within the range of validity of the three-point fitted polynomial is given by the derivative of equation (A1):

$$\nabla y \Big|_{x_j} = \frac{dy}{dx}\Big|_{x_j} = y'_{xj} = a_1 + 2a_2x_j \tag{A12}$$

Substituting the coefficients a_1 and a_2 from equation (A11) gives

$$\nabla y \Big|_{x_j} = \left[\frac{1}{x_1 - x_2} + \frac{2x_j - (x_1 + x_2)}{(x_1 - x_2)(x_1 - x_3)}\right] \cdot y_1 + \left[\frac{1}{x_2 - x_1} + \frac{2x_j - (x_1 + x_2)}{(x_2 - x_1)(x_2 - x_3)}\right] \cdot y_2$$
$$+ \left[\frac{2x_j - (x_1 + x_2)}{(x_3 - x_1)(x_3 - x_2)}\right] \cdot y_3 \tag{A13}$$

This is the desired flux gradient $(\vec{\nabla}\varphi)$ expression. Through Fick's law the flux gradient determines the neutron current for a particular energy and location:

$$\vec{J} = -D\, \vec{\nabla}\varphi \tag{A14}$$

However, because TDSN provides scalar fluxes for the midpoints of mesh intervals, the flux gradient at these midpoints does not necessarily provide a good representation of the current through that mesh interval. More information can be incorporated into the current calculation by using the net current through the parallel faces of the mesh interval to represent that interval. Thus, a difference in gradients taken at boundary points of mesh intervals is used as the net flux gradient $\vec{\nabla}\varphi$:

$$\nabla y_{j+1} - \nabla y_j = 2a_2(x_{j+1} - x_j) \tag{A15}$$

$$\nabla y_{j+1} - \nabla y_j = \frac{2(x_{j+1} - x_j)}{(\Delta x_1 \cdot \Delta x_2)(\Delta x_1 + \Delta x_2)} \left[(\Delta x_2)y_1 - (\Delta x_1 + \Delta x_2)y_2 + (\Delta x_1)y_3\right] \quad \text{(A15a)}$$

with $\Delta x_1 = x_2 - x_1$ and $\Delta x_2 = x_3 - x_2$.

Frequently, a calculational configuration has boundaries that are perfectly reflecting (i.e., the cell condition). In such case there is no net current across the boundary and, therefore, the flux gradient is zero. If a flux profile fitted to three mesh points is extrapolated to the boundary, it will not necessarily have zero slope at the boundary. Therefore, if the cell condition exists, the PERTRAN program fits a second-degree polynomial to the slope (identically zero) at the boundary and the two closest flux points. Equation (A1) is the generalized polynomial for the flux profile. The flux gradient is then

$$y' = a_1 + 2a_2x \quad \text{(A16)}$$

The zero gradient restriction

$$y'_b = 0 = a_1 + 2a_2x_b \quad \text{(A17)}$$

at any interior or exterior boundary x_b determines a_1 in terms of a_2 and x_b. Therefore, using the flux (y_1 and y_2) at two adjacent mesh points (x_1 and x_2) we obtain from equation (A16)

$$a_2 = \frac{y_2 - y_1}{(x_2 - x_1)\left[(x_2 - x_b) + (x_1 - x_b)\right]} \quad \text{(A18)}$$

Now to determine the net flux gradient across such a bounding mesh interval with a cell condition on one side, we need merely to evaluate equation (A16) for y' at the boundary between x_1 and x_2. This particular gradient is given by

$$y' = 2a_2(x - x_b) \quad \text{(A19)}$$

and is used as $\vec{\nabla}\varphi$ in equation (A14).

APPENDIX B

SAMPLE PROBLEM OUTPUT

This sample of PERTRAN output is the corner perturbation problem (P-1 approximation) discussed earlier.

```
SAMPLE PROBLEM CORNER P-1 APPROXIMATION
        NMAT          NG          NFAST          NDOWN           NUP          NR          NZ
           1           4              4              3             0          12          12
       KCTYP         KIN           KPAP             KP            KS         KAR          KD
           0           1              0              1             1           1           1
      NZONE1      NZONE2         LISTFX           NMID        KAPROX       KFLUX       KBEFF
           3           3              1              3             2           1          -0
        LAST       LAST1          LAST2          XKEFF          NDSP        IGHI        IGLD
        4136        4136           4018     0.99926700            -0          -0          -0


V(I,J)
  0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01
  0.2000000E 01   0.2000000E 01   0.5000000E 01   0.5000000E 01
  0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01
  0.2000000E 01   0.2000000E 01   0.5000000E 01   0.5000000E 01
  0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01
  0.2000000E 01   0.2000000E 01   0.5000000E 01   0.5000000E 01
  0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01
  0.2000000E 01   0.2000000E 01   0.5000000E 01   0.5000000E 01
  0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.4000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01
  0.2000000E 01   0.2000000E 01   0.5000000E 01   0.5000000E 01
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1000000E 01   0.2500000E 01   0.2500000E 01
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1000000E 01   0.2500000E 01   0.2500000E 01
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1000000E 01   0.2500000E 01   0.2500000E 01
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1000000E 01   0.2500000E 01   0.2500000E 01
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1000000E 01   0.2500000E 01   0.2500000E 01
  0.5000000E 01   0.5000000E 01   0.5000000E 01   0.5000000E 01   0.5000000E 01   0.2500000E 01   0.2500000E 01   0.2500000E 01
  0.2500000E 01   0.2500000E 01   0.6250000E 01   0.6250000E 01
  0.5000000E 01   0.5000000E 01   0.5000000E 01   0.5000000E 01   0.5000000E 01   0.2500000E 01   0.2500000E 01   0.2500000E 01
  0.2500000E 01   0.2500000E 01   0.6250000E 01   0.6250000E 01


R(I)
  0.              0.2000000E 01   0.4000000E 01   0.6000000E 01   0.8000000E 01   0.1000000E 02   0.1100000E 02   0.1200000E 02
  0.1300000E 02   0.1400000E 02   0.1500000E 02   0.1750000E 02   0.2000000E 02


Z(J)
  0.              0.2000000E 01   0.4000000E 01   0.6000000E 01   0.8000000E 01   0.1000000E 02   0.1100000E 02   0.1200000E 02
  0.1300000E 02   0.1400000E 02   0.1500000E 02   0.1750000E 02   0.2000000E 02


DELR(I)
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1500000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1750000E 01   0.2500000E 01


DELZ(J)
  0.2000000E 01   0.2000000E 01   0.2000000E 01   0.2000000E 01   0.1500000E 01   0.1000000E 01   0.1000000E 01   0.1000000E 01
  0.1000000E 01   0.1750000E 01   0.2500000E 01


CHI(IG)
    0.7538200E 00      0.2136700E 00      0.2963000E-01      0.2880000E-02

DELTAS COMPUTED INTERNALLY FROM DIFFERENCE BETWEEN PERTURBED AND UNPERTURBED CROSS SECTIONS
TDSN FORMAT CROSS SECTIONS. TRANSPORT APPROXIMATION. WITH P-0 AND P-1 CS.

                                          MATERIAL  1
      0.1739999E-03      0.4360001E-03      0.1271000E-02      0.8730004E-03      0.                0.                0.
      0.1959999E-03      0.3960002E-03      0.1495000E-02      0.1245999E-02      0.2019999E-03     0.                0.
      0.2959999E-03      0.5300003E-03      0.1968000E-02      0.1651000E-02      0.4499999E-04     0.2000001E-04     0.
      0.5319999E-03      0.8439999E-03      0.2429001E-02      0.1897000E-02      0.2000001E-04     0.5000002E-05     0.4000001E-05

 TDSN FORMAT. P-1 CROSS SECTIONS WITH FACTOR OF 3 INCLUDED

                                          MATERIAL  1
      0.                 0.                 0.8410001E-03      0.1294000E-02      0.                0.                0.
      0.                 0.                 0.1135000E-02      0.1023000E-02      0.                0.                0.
      0.                 0.                 0.1718000E-02      0.7260004E-03    -0.1999999E-05     0.                0.
      0.                 0.                 0.2329001E-02      0.2020001E-03    -0.2000001E-05     0.                0.


 DELTA OUTSCATTER CROSS SECTIONS
    0.2251100E-03      0.5066000E-04      0.2002000E-04     -0.

 BUCKLING LOSS INFORMATION ****
 FIRST DIMENSION  45.000000 SECOND DIMENSION  -0.
 FIRST DIMENSION PERTURBED  -0.        SECOND DIMENSION PERTURBED  -0.
 BUCKLING FACTOR     1.8137994

 C(N,IG,ITY) FOR ITY OF SIG A, NU SIG F,(D* OR TR OR TOT), SIG OUTSCATTER, NUP VALUES OF SIG UP, AND NDOWN VALUES OF SIG DOWN.

                                           MATERIAL  1
       0.1739999E-03      0.4360001E-03      0.1271000E-02      0.2251100E-03      0.                0.                0.
       0.1959999E-03      0.3960002E-03      0.1495000E-02      0.5066000E-04      0.2019999E-03     0.                0.
       0.2959999E-03      0.5300003E-03      0.1968000E-02      0.2002000E-04      0.4499999E-04     0.2000001E-04     0.
       0.5319999E-03      0.8439999E-03      0.2429001E-02     -0.                 0.2000001E-04     0.5000002E-05     0.4000001E-05
```

```
C(N,IG,ITY) FOR ITY OF 0.0 , P-0 G-TO-G, TRANSPORT, P-1 G-TO-G, NUP VALUES OF SIG UP, AND NDOWN VALUES OF SIG DOWN.

                                             MATERIAL  1
    0.                  0.8730004E-03    0.8410001E-03    0.1294000E-02    0.                0.                 0.
    0.                  0.1246999E-02    0.1135000E-02    0.1023000E-02    0.                0.                 0.
    0.                  0.1651000E-02    0.1718000E-02    0.7260004E-03   -0.1999999E-05     0.                 0.
    0.                  0.1897000E-02    0.2329001E-02    0.2020001E-03   -0.2000001E-05     0.                 0.

 DELTA BUCKLING LOSS CROSS SECTIONS
      FIRST DIRECTION
  -0.4621013E-04    -0.4918664E-04    -0.4613533E-04    -0.4080136E-04


THE NON-ZERO MATERIALS ARE PERTURBED
   0     0     1


MAP MA(I,J)
 1  1  1  1  1  1  1  1  1  1  2  2
 1  1  1  1  1  1  1  1  1  1  2  2
 1  1  1  1  1  1  1  1  1  1  2  2
 1  1  1  1  1  1  1  1  1  1  2  2
 1  1  1  1  1  1  1  1  1  1  2  2
 1  1  1  1  1  3  3  3  3  3  2  2
 1  1  1  1  1  3  3  3  3  3  2  2
 1  1  1  1  1  3  3  3  3  3  2  2
 1  1  1  1  1  3  3  3  3  3  2  2
 1  1  1  1  1  3  3  3  3  3  2  2
 2  2  2  2  2  2  2  2  2  2  2  2
 2  2  2  2  2  2  2  2  2  2  2  2


MAP MA(I,J) WITH ONLY PERTURBED ENTRIES
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  1  1  1  1  1  0  0
 0  0  0  0  0  1  1  1  1  1  0  0
 0  0  0  0  0  1  1  1  1  1  0  0
 0  0  0  0  0  1  1  1  1  1  0  0
 0  0  0  0  0  1  1  1  1  1  0  0
 0  0  0  0  0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0  0  0  0  0


THE FLUXES XN(I,J)
IG=  1
 J=  1
  0.[illegible]E-01 . . .
  0.[illegible]E-01 . . .
 J=  2, . . .
IG=  2
 J=  1, . . .
IG=  3, . . .

THE ADJOINT FLUXES XNA(I,J)
IG=  1, . . .
 J=  1, . . .

THE FIRST DIRECTION CURRENTS XJ(I,J)
IG=  1, . . .
 J=  1, . . .

THE FIRST DIRECTION ADJOINT CURRENTS XJA(I,J)
IG=  1, . . .
 J=  1, . . .

THE SECOND DIRECTION CURRENTS YJ(I,J)
IG=  1, . . .
 J=  1, . . .

THE SECOND DIRECTION ADJOINT CURRENTS YJA(I,J)
IG=  1, . . .
 J=  1, . . .

THE POWER DENSITY P(I,J)
 J=  1
 J=  [illegible], . . .

THE ADJOINT POWER DENSITY PA(I,J)
 J=  1, . . .


ADJOINT POWER TIMES POWER EQUALS 0.[illegible]E-[illegible]

                                       ((1/KEFF PERTURBED)-(1/KEFF UNPERTURBED))
                                                P-1 TRANSPORT APPROXIMATION

     PRODUCTION      SCATTERING        ABSORPTION       REMOVAL        G-TO-G***J-WGT     LEAKAGE
 -0.3102826E-03    0.               0.4416316E-04    0.5713548E-04   -0.2488113E-04   -0.4742149E-04
 -C.9094846E-C4   -0.6478360E-04    0.7559187E-04    0.1953820E-04   -0.2581540E-04   -0.3283457E-04
 -C.1307562E-04   -0.2598379E-04    0.5467171E-04    0.3697730E-05   -0.1285647E-04   -0.1616039E-04
 -C.1202218E-05   -0.6957652E-05    0.1309543E-04   -0.               -0.2604322E-05   -0.3498791E-05
  GROUP  TOTALS
 -C.4155089E-03   -0.9772504E-04    0.1875222E-03    0.8037140E-04   -0.6615832E-04   -0.9991522E-04

    TRANSVERSE        TRANSVERSE                                                               GROUP
    LEAKAGE  1        LEAKAGE  2
 -0.1172866E-04    0.                                                                   -0.2455937E-03
 -0.1896996E-04    0.                                                                   -0.1053884E-03
 -0.8521278E-05    0.                                                                   -0.2067713E-05
 -C.1004345E-C5    0.                                                                    0.1326893E-05
  GROUP  TOTALS
 -0.4022424E-04    0.                                                                   -0.3517229E-03


UNPERTURBED MULTIPLICATION FACTOR ****  0.99926700
 PERTURBED MULTIPLICATION FACTOR  ****  0.99961832
```

APPENDIX C

PROGRAM INFORMATION

Input Instructions

This section contains the input instructions and explanations of the input parameters. The symbol * after a card number means to use as much of the card or as many cards as necessary.

Card	Format	Variable	Description
1^*	I1, 1X, 14A5	TITLE	Title cards. Number in card column 1 signifies the last title card.
2	7I10	NMAT	Number of materials for which perturbed cross sections will be provided.
		NG	Number of energy groups.
		NFAST	Number of fast groups (to be used if KCTYP = 0 or 1 and NUP > 0).
		NDOWN	Maximum number of groups down-scattered to.
		NUP	Maximum number of groups up-scattered to.
		NR	Number of first direction mesh intervals (horizontal direction on map - left to right).
		NZ	Number of second direction mesh intervals (vertical direction on map - top to bottom).
3	7I10	KCTYP	Format for cross sections (see card 15). = 0 Perturbed then unperturbed cross sections in TDSN format. Increments are obtained within the program: $\Delta\Sigma \equiv \Sigma^p - \Sigma$. (This option must be used if a buckling loss is to be determined - card 16.) = 1 Increments $\Delta\Sigma$ in TDSN format.
		KIN	Format for geometric specifications. =1 Use mesh from TDSN binary dump (obtained from TDSN option KBCDUP = -2). See also cards 6 to 8. = 2 Use binary dump of Δr and Δz instead of r and z. See cards 9 and 10. = 3 Not binary (see cards 11 to 13).
		KPAP	= -1 Read in perturbed ν-fission cross sections, and then the adjoint production (P_i^+).

Card	Format	Variable	Description
			= 0 Read in real (P_i) and adjoint ($P_i^\dagger$) production = 1 Read in PAP, which is $\sum_{i}^{NIJ} P_i P_i^\dagger V_i$ (cannot be used if KBEFF = 1).
		KP	= 1 Contribution to $\Delta(1/k)$ from production increments $\Delta(\nu\Sigma_f)$ will be calculated.
		KS	= 1 Contribution to $\Delta(1/k)$ from incremental scattering into a group will be calculated.
		KAR	= 1 Contributions to $\Delta(1/k)$ from transport, absorption, scattering removal, and buckling loss increments will be calculated. = 2 Lifetime will be calculated.
		KD	= 1 Contributions to $\Delta(1/k)$ from diffusion coefficient increment will be calculated. Use only with KAPROX = 0. If KCTYP $\neq$ 0, ΔD will be calculated internally as $\Delta D = \Delta\left[1/(3\Sigma_{tr})\right]$. Restricted to NR $\geq$ 3 and NZ $\neq$ 2. = -1 Same as for +1 except that cell boundary condition exists which will be specified on card 30.
4	7I10	NZONE1	Number of material zones in first direction (corresponding to material map in TDSN but not restricted to that map).
		NZONE2	Number of material zones in second direction (corresponding to material map in TDSN but not restricted to that map).
		LISTFX	= 1 List flux input and production rate input as part of output.
		NMID	Number of materials in identification map (IDM) (hence, TDSN map can be used directly).
		KAPROX	Type of approximation. = 0 Diffusion theory. = 1 Transport theory - (P-0*) transport corrected P-0 cross sections. = 2 Transport theory - P-1 cross sections.
		KFLUX	Format for fluxes and currents (see card 21*). = 1 From TDSN binary dump. = 2 5(I5, E10.6). = 3 7E10.6.

Card	Format	Variable	Description
		KBEFF	= 1 Perform β-effective calculation; read in delayed spectra information at cards 32 and 33.
5	F10.8, 3I10	XKEFF	Multiplication factor (k) from unperturbed calculation. Must be included for all problems, even if KBEFF = 1.
		NDSP	Number of delayed spectra to be read in.
		IGHI	Highest energy group in which any of NDSP spectra contribute.
		IGLO	Lowest energy group in which any of NDSP spectra contribute.

If KIN = 1 and KBEFF = 0 read in cards 6 to 8. If KIN = 1 and KBEFF = 1 read in only card 6.

6*	Binary	V(IJ)	Volumes from TDSN binary dump.
7*	Binary	R(I)	Mesh boundaries in first direction (NZONE1 values).
8*		Z(I)	Mesh boundaries in second direction if NZ > 2 (NZONE2 values).

If KIN = 2, read in cards 9 and 10.

9*	Binary	DELR(I)	Mesh increments in first direction.
10*	Binary	DELZ(I)	Mesh increments in second direction.

If KIN = 3, read in cards 11 to 13.

11	I10	KGEO	Geometry. = 1 Slab (x - y). = 2 Cylinder (r - z). = 3 Sphere (r).
12*	5(I5, E10.6)	NM(I), RM(I)	First direction mesh. NM is the number of mesh intervals to include between the preceding value of RM and the value of RM that immediately follows NM. If NM = 0, associated RM is ignored. If NM < 0, associated RM is the last value to be used. SUM(NM) = NR.
13*	5(I5, E10.6)	NM(I), RM(I)	Second direction mesh if NZ > 1. Same as for card 12 except that SUM(NM) = NZ.

Card	Format	Variable	Description
If KBEFF = 1 read in only cards 22, 28, 29, 32, and 33.			
14*	7E10.6	CHI(IG)	Fission spectrum (NG values).
15*	TDSN 7E10.6	C	Cross sections (see Input Notes, p. 29). If KAPROX = 2 and KCTYP = 0, then the order of cross sections for each of NMAT materials is as follows: Perturbed P-0 c.s. Unperturbed P-0 c.s. Perturbed P-1 c.s. Unperturbed P-1 c.s. Perturbed removal c.s. Unperturbed removal c.s. If KAPROX < 2, do not include P-1 cross sections. If KCTYP = 1, each set of perturbed and unperturbed cross sections is replaced by one set of cross section increments.
If KCTYP = 1, skip card 16.			
16	7E10.6	H1	Buckling dimension in first direction; zero only if no buckling loss is considered.
		H2	Buckling dimension in second direction; zero except for one-dimensional slabs.
		BF	Buckling factor $\left(\sqrt{B^2}\right)$: $\pi/\sqrt{3}$ for plane boundaries; $2(2.405/\sqrt{3})$ for cylindrical boundaries.
		HP1	Perturbed buckling dimension (1st direction); zero if H1 is not to be perturbed.
		HP2	Perturbed buckling dimension (2nd direction); zero if H2 is not to be perturbed.
17*	14I5	MATCHG(I)	= 0 Particular material in map is not to be perturbed. = material number In compacted sequence beginning with 1 if it is to be perturbed. There will be NMID entries of which NMAT will be nonzero running from 1 to NMAT.

Card	Format	Variable	Description

Cards 18 to 20 form the material identification map. For convenience, the TDSN map may be used but it is not required.

18*	7I10	NMRA(I)	Number of mesh intervals per zone in first direction. NZONE1 values.
19*	7I10	NMZA(I)	Number of mesh intervals per zone in second direction if NZONE2 $>$ 0. NZONE2 values.
20*	14I5	IDM(IJ)	Material identification number to include in each zone. NZONE2 sets of cards (1 if NZONE2 = 0) with NZONE1 values per card. IDM = 0 if no cross sections are read in for the zone (i.e., no perturbation). However, through MATCHG (card 17) the map from TDSN can be used here without having to zero any IDM entry).
21*	Binary	XN	Use if KFLUX = 1; real fluxes (NIJ values for group 1, then NIJ values for group 2, etc.).
	5(I5, E10.6)	NM, XN	Use if KFLUX = 2; NM is the number of mesh intervals with the flux level XN. The NM entry completing the NIJ total must be $<$0.
	7E10.6	XN	Use if KFLUX = 3; fluxes for all intervals for group 1, then for group 2, etc.
22*		XNA	Adjoint flux. Same format as on card 21.

If KAPROX $>$ 0, read in currents on cards 23 and 24; if NZ $>$ 1, read in cards 25 and 26 as well. The format must be the same as for card 21. The adjoint currents may require reversal, just as adjoint fluxes.

23*		XJ	First direction real current.
24*		XJA	First direction adjoint current.

Read cards 25 and 26 only if KAPROX $>$ 0 and NZ $>$ 1.

25*		YJ	Second direction real current.
26*		YJA	Second direction adjoint current.

Read in card 27 only if KPAP = -1.

27*	7F10.8	PNUF(IG)	Use if KPAP = -1. Read in perturbed ν-fission cross sections (to be combined with the unperturbed fluxes to obtain the real production F(IJ)).

Card	Format	Variable	Description
27* (Cont.)			See equation (18). NMAT card sets, each with NG entries.
28*	Binary	F(IJ)	Use if KPAP ≤ 0. Read in the real production, either perturbed or unperturbed (see Input Notes, p. 29). Binary format from TDSN. NIJ values. (If KBEFF = 1 or if KPAP = -1 this must be the unperturbed production.)
29	Binary	FA(IJ)	Unperturbed adjoint production. Binary format from TDSN. NIJ values.

Read in card 30 only if KPAP = 1.

30	E10.6	PAP	$\sum_{IJ}^{NIJ}$ F(IJ)*FA(IJ)*V(IJ) as given from a previous PERTRAN problem.

Read in card 31 only if KD = -1.

31	7I10	KRBC	= 0 Not a cell condition; that is, no return current across the right boundary (I = NR). = 1 Perfect reflection exists across the right boundary (I = NR).
		KLBC	Same options for left boundary (I = 1).
		KTBC	Same options for top boundary (J = NZ).
		KBBC	Same options for bottom boundary (J = 1).

Read in cards 32 and 33 only if KBEFF = 1.

32	7E10.6	BETA(I)	Delayed neutron fractions; NDSP entries.
33*	7E10.6	DELSP(J)	Delayed spectra; NDSP sets of cards, each set with (IGLO-IGHI+1) entries (high to low energy).

Input Notes

Cross sections are required in the TDSN (ref. 3) format. The TDSN cross sections for each group are absorption, ν-fission, transport if P-0* (total if P-0 of P-1), up-scattering into the group, within-group scattering, and down-scattering into the group in a 7E10.6 format. The P-1 cross sections (P-1 of P-1) do not occupy the first two fields; the rest of the fields contain total, up-scattering, within-group scattering, and down-scattering cross sections. The P-1 scattering cross sections already contain the $2l + 1$ multiplier. The removal cross sections from all energy groups are listed consecutively, 7 to a card. The removal cross sections and the TDSN group sets are ordered from high energy to low.

Within PERTRAN the format is changed to a slight modification of the TDSN format. Perturbed cross sections may be used and the increments then calculated within PERTRAN. The P-1 cross sections are not considered separate materials and must be provided immediately after the appropriate P-0 cross sections for each of the NMAT materials (see card 15).

The real and adjoint fluxes and currents should be obtained from TDSN transport calculations - the real and adjoint solutions having been converged to the same multiplication factor $k = k^+$. All fluxes and currents from TDSN are punched in binary form continuously for all mesh intervals for each group.

The identification map for a perturbation problem may be identical to that used in TDSN. PERTRAN thus requires information as to which material regions in this map are to be perturbed.

The normalization factor determined by equation (18) requires the perturbed ν-fission cross sections and the unperturbed real fluxes. The KPAP = -1 options forms the production quantity $\sum_g \left(\nu\Sigma_{f_g}\right)^p \varphi_g$ from this information. However, the TDSN program provides in convenient form the quantities $\sum_g \nu\Sigma_{f_g} \varphi_g$ and $\sum_g \left(\nu\Sigma_{f_g}\right)^p \varphi_g^p$ from the unperturbed and perturbed calculations. Under certain conditions these quantities, which can be more conveniently handled, may be satisfactory.

For example, if $\nu\Sigma_{f_g}$ is not perturbed or if all $\nu\Sigma_{f_g}$ are perturbed by the same factor, then the unperturbed production may be used (KPAP = 0). In the latter case this factor must then be removed from the printed perturbation results. On the other hand, if all the $\nu\Sigma_{f_g}$ are not perturbed by the same factor but the perturbed fluxes φ_g^p are not significantly different from the unperturbed fluxes, then the perturbed production may be used (KPAP = 0).

The lifetime is calculated as an absorption perturbation (KP=KS=KD=0; KAR=2) in which (1/v) cross sections are supplied rather than absorption cross sections (KCTYP may be 0 or 1). Furthermore, the unperturbed real production rate should be used and any approximation (KAPROX) may be used. However, if KAPROX = 0, then KCTYP must be 1 in order to provide transport cross sections for the Fick's law current approximation. Any lifetime "perturbation" should extend over the whole region that produced the particular flux spectrum and averaged 1/v cross sections. The fission spectrum and the P-1 and removal cross sections are not used so that blank cards may be read in for them.

The calculation of beta effective requires that the unperturbed production be used both separately and combined with the adjoint production; hence, the KPAP=0 option must be used. Furthermore, only the adjoint fluxes are needed so KAPROX=0 should also be used. In this case KCTYP may be either 0 or 1; no current approximation is made. Each of the variables, NMAT, KP, KS, KAR, KD, NZONE1, NZONE2, and NMID should be equal to 0 when KBEFF=1.

Within TDSN the normalization of the fission source in an adjoint calculation results in an adjoint flux containing a factor of k. If the prompt fission spectrum is treated as the delayed spectrum and a beta effective calculation is performed, then the unadjusted adjoint flux provides a β_{eff} equal to k. Within PERTRAN this factor of k is removed (in a β_{eff} calculation) from the adjoint flux so that an absolute β_{eff} is obtained. If a delayed neutron fraction of 1.0 is read in, the ratio β_{eff}/β is obtained.

Because it is difficult to determine $\Delta\Sigma_{lk}$ explicitly from the perturbed and unperturbed forms of equation (25), the calculation of the buckling loss requires an unperturbed Σ_{tr} as well as the increment $\Delta\Sigma_{tr}$; thus, the KCTYP=0 option must be used for a buckling loss calculation. The perturbation of a buckling dimension H may be treated separately or in combination with a perturbation of Σ_{tr}.

Output Notes

The input parameters are listed and the computer storage required for the problem is listed under LAST. The incremental cross sections are labeled and listed in the internal modified TDSN form.

If LISTFX = 1, the regular and adjoint flux, current, and production are given in the output. The normalization factor in equation (18) is printed out.

The contribution to $\Delta(1/k)$ is given by group for production and inscattering sources and for absorption, outscatter (removal), and leakage losses. Subtotals provide the total contribution by group and type of process. Each GROUP entry is the sum of all the listed contributions, with one exception. In the P-1 approximation (KAPROX=2) the LEAKAGE contribution has already been included in other categories (see eq. (24)).

The eigenvalue increment resulting from a change in the buckling loss is listed as TRANSVERSE LEAKAGE in the output. The nontransverse leakage out of a reactor system is given by the current-weighted contribution to $\Delta(1/k)$. For the P-1 approximation it is listed as LEAKAGE in the output; for the P-0* approximation it is listed as TRANSPORT; and for the diffusion approximation it is the sum of the two DIFFUSION COEFFICIENT entries.

The perturbed and the unperturbed multiplications factors (k^p and k) are also provided. If the lifetime has been calculated it appears as the absorption contribution to $\Delta(1/k)$. If a β_{eff} calculation has been performed the delayed spectra are provided as output, and for each delayed spectrum j considered, the corresponding spectral sums β_j and $\beta_{j(\mathrm{eff})}$ are given.

Programming Notes

Basically PERTRAN provides one large storage array (X) with a length of 25 000. Within this array the number of groups, types of cross sections, materials, and mesh intervals are variable. (The amount of this 25 000 storage actually used is listed in the output as LAST.) The dimension of this one large array could be easily changed to accommodate modifications to the program.

Three other easily changed storage constraints occur in PERTRAN: (1) the PNUF array (in subroutine INPUT), which must contain (NG*NMAT) locations, is set at 50; (2) the STOR array (in subroutine CREAD) must contain the larger of (NTYPS+2*NG) or (3*NG) locations and is presently set at 70; (3) the MATCHG array (in subroutine IDACAV) must contain NMID locations and is now set at 25. For further convenience in modification, an extra common block CALL3 (which is not used in the present version of PERTRAN) has been included.

APPENDIX D

PROGRAM LISTING

This appendix contains the listing (in FORTRAN IV) of the program, overlay information, and a memory map.

```
$IBFTC PERSN
C
C
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON       X
      COMMON       /CALL1/                                 NG,       NTYPS,
     1   NTYNG,    NFAST,    NUP,      NR,       NZ,       NIJ,
     2             KFLUX,   KBEFF,     KCTYP,    KIN,
     3             KP,       KS,       KAR,      KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON       /CALL2/   PAP
      COMMON       /CALL3/   KXTRA1,   KXTRA2,   KXTRA3,   KXTRA4,
     1   KXTRA5,   EXTRA1,   EXTRA2,   EXTRA3,   EXTRA4,   EXTRA5
      COMMON       /CINPT/   LMA,      LV,       LDELR,    LDELZ,
     1                       LC,       LCHI,     LF,       LN,
     2   LNA, LXJ,LXJA,LYJ,LYJA,       LP,       LS,       LA,
     3   LR,       LD1,      LD2,      LTL1,     LTL2,     LS1X,
     4   LR1,      LZ1,      LC1,      LFA,      LNM,      LRM,
     5   LID1,     LID2,     LIDM,     XKEFF,
     6   LAST1,    LAST2,    LAST
C
C     THE DIMENSION STATEMENTS
      DIMENSION    X(25000)
C
C
C
 1000 CALL INPUT
      IF (LMA) 1000,1000,1025
C
C
 1025 CALL PERTUR ( X(LMA), X(LV), X(LDELR), X(LDELZ),    X(LC), X(LCHI),
     1XKEFF,       X(LN), X(LNA), X(LF),   X(LP), X(LS), X(LA), X(LR),
     2   X(LD1), X(LD2),  X(LXJ),  X(LXJA),  X(LYJ),  X(LYJA),
     3   X(LTL1),X(LTL2), X(LS1X))
C
C
      IF(KAPROX.GT.0) GO TO 1050
      IF(KD.EQ.0.AND.KAR.NE.2) GO TO 1050
      CALL DIFUSE (X(LMA),X(LV),X(LDELR),X(LDELZ),X(LC),X(LN),X( LNA),
     1 X(LD1),X(LD2), X(LTL1), X(LTL2), X(LA) )
```

```
  1050 CALL OUTPUT ( X(LP), X(LS), X(LA), X(LR), X(LD1), X(LD2),
     1X(LYJA),XKEFF, X(LTL1), X(LTL2))
      GO TO 1000
C
C
C
      END
```

```
$IBFTC PER1

      SUBROUTINE INPUT
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON        X
      COMMON        /CALL1/                         NG,       NTYPS,
     1   NTYNG,     NFAST,    NUP,      NR,         NZ,       NIJ,
     2              KFLUX,   KBEFF,     KCTYP,      KIN,
     3              KP,       KS,       KAR,        KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON        /CALL2/   PAP
      COMMON        /CALL3/   KXTRA1,   KXTRA2,     KXTRA3,   KXTRA4,
     1   KXTRA5,    EXTRA1,   EXTRA2,   EXTRA3,     EXTRA4,   EXTRA5
      COMMON        /CINPT/   LMA,      LV,         LDELR,    LDELZ,
     1                        LC,       LCHI,       LF,       LN,
     2   LNA, LXJ,LXJA,LYJ,LYJA,        LP,         LS,       LA,
     3   LR,        LD1,      LD2,      LTL1,       LTL2,     LS1X,
     4   LR1,       LZ1,      LC1,      LFA,        LNM,      LRM,
     5   LID1,      LID2,     LIDM,     XKEFF,
     6   LAST1,     LAST2,    LAST
      COMMON        /CHANG/             NDOWN,
     1   NGM1,      NGP1,               N
C
C     THE DIMENSION STATEMENTS
      DIMENSION     X(25000)
      DIMENSION     TITLE(14)
      DIMENSION PNUF(50)
C     IF KPAP.LT.0, PNUF IS USED AND MUST CONTAIN NG*NMAT LOCATIONS
C
C     THE FORMAT STATEMENTS
  100 FORMAT (1H1)
  101 FORMAT(7F10.8)
  102 FORMAT(F10.8,3I10)
  103 FORMAT(1HL,78H THE PERTURBED NU*FISSION CROSS SECTIONS ARE (BY GRO
     1UP AND PERTURBED MATERIAL))
  104 FORMAT(8E16.7)
  105 FORMAT (I1,1X,14A5)
  106 FORMAT (2X,14A5)
  110 FORMAT (7I10)
  111 FORMAT(7I16)
```

```
   112 FORMAT(3I16,F16.8,3I16)
   120 FORMAT (1HJ,11X,4HNMAT,14X,2HNG,11X,5HNFAST,11X,5HNDOWN,13X,3HNUP,
      1    14X,2HNR,14X,2HNZ)
   121 FORMAT (1HJ,10X,5HKCTYP,13X,3HKIN,12X,4HKPAP,14X,2HKP,14X, 2HKS,
      1    13X,3HKAR,14X,2HKD)
   122 FORMAT (1HJ,9X,6HNZONE1,10X,6HNZONE2,10X,6HLISTFX,10X,6H  NMID,10X
      1,6HKAPROX,10X,6H KFLUX,10X,6H KBEFF)
   130 FORMAT (1HL,3HNR=,I2,20H IS LESS THAN THREE.)
   131 FORMAT (1HL,14HNZ EQUALS TWO.)
   135 FORMAT (1HJ,11X,4HLAST,11X,5HLAST1,11X,5HLAST2,11X,5HXKEFF,12X,
      14HNDSP,12X,4HIGHI,12X,4HIGLO)
C
C
C
  1000 WRITE (6,100)
  1005 READ (5,105) ITEMP,(TITLE(I), I=1,14)
       WRITE (6,106) (TITLE(I), I=1,14)
       IF (ITEMP) 1005,1005,1010
C
  1010 READ (5,110) NMAT,NG,NFAST,NDOWN,NUP,NR,NZ
       WRITE (6,120)
       WRITE (6,111) NMAT,NG,NFAST,NDOWN,NUP,NR,NZ
       READ (5,110) KCTYP,KIN,KPAP,KP,KS,KAR,KD
       WRITE (6,121)
       WRITE (6,111) KCTYP,KIN,KPAP,KP,KS,KAR,KD
       READ (5,110) NZONE1,NZONE2,LISTFX,NMID,KAPROX,KFLUX,KBEFF
       WRITE (6,122)
       WRITE (6,111) NZONE1,NZONE2,LISTFX,NMID,KAPROX,KFLUX,KBEFF
       READ(5,102) XKEFF,NDSP,IGHI,IGLO
C
C
  1050 IF (NZ) 1055,1055,1060
  1055 NZ=1
  1060 IF (KD) 1085,1085,1065
  1065 IF (NR-3) 1070,1075,1075
  1070 WRITE (6,130) NR
       RETURN
  1075 IF (NZ-2) 1085,1080,1085
  1080 WRITE (6,131)
       RETURN
  1085 NIJ=NR*NZ
       NTYPS=NDOWN+NUP+4
       NTYNG=NTYPS*NG
       NGP1=NG+1
       NGM1=NG-1
       IBSTOR=NIJ+NDSP*(IGLO-IGHI+1)+NDSP
C
C
  1100 LMA=1
       LV=LMA+NIJ
       LDELR=LV+NIJ
       LDELZ=LDELR+NR-1
       IF (NZ-1) 1105,1105,1110
  1105 LC=LDELZ+1
       GO TO 1115
  1110 LC=LDELZ+NZ-1
  1115 LP1CS=LC
       IF(KAPROX.EQ.2) LP1CS=LC+NMAT*NTYNG
       LCHI=LP1CS+NMAT*NTYNG
```

```
C     BUCKLING LOSS STORAGE (BELOW)
      LCHI=LCHI +3*NG*NMAT
      LF=LCHI+NG
      LN=LF+NIJ
      LNA=LN +NG*NIJ
      LXJ =LNA+NG*NIJ
      IADD=1
      IF(KAPROX.GT.0) IADD=NG*NIJ
      LXJA=LXJ+IADD
      LYJ=LXJA+IADD
      IF(NZ.LE.1) IADD=1
      LYJA=LYJ+IADD
      LP=LYJA+IADD
      IF(KBEFF.GT.0) LP=LYJA+IBSTOR
      LS=LP +NG
      LA=LS+NG
      LR=LA +NG
      LD1=LR +NG
      LD2=LD1+NG
      LTL1=LD2+NG
      LTL2=LTL1+NG*NMAT
      LS1X=LTL2+NG*NMAT
      LAST2=LS1X+NG-1
C
C     LAST1 = STORAGE REQUIRED BY OVERLAY 011
C     LAST2 = STORAGE REQUIRED BY OVERLAY 001
C
 1125 LNM=LP
      LRM=LNM+5
      LR1=LRM+5
      GO TO (1126,1127,1126), KIN
 1126 LZ1=LR1+NR+1
      LAST1=LZ1+NZ
      GO TO 1128
 1127 LZ1=LR1+1
      LAST1=LZ1+1
 1128 LC1=LP
      ITEMP=LC1+NTYNG-1
      IF (ITEMP-LAST1) 1135,1135,1130
 1130 LAST1=ITEMP
 1135 LFA=LR1
      IF (KPAP) 1136,1136,1145
 1136 ITEMP=LFA+NIJ-1
      IF (ITEMP-LAST1) 1145,1145,1140
 1140 LAST1=ITEMP
 1145 LID1=LP
      LID2=LID1+NZONE1
      LIDM=LID2+NZONE2
      ITEMP=LIDM+NZONE1
      IF (ITEMP-LAST1) 1175,1175,1150
 1150 LAST1=ITEMP
C
C
 1175 LAST=LAST2
      IF (LAST1-LAST) 1185,1185,1180
 1180 LAST=LAST1
 1185 WRITE (6,135)
 1186 WRITE (6,112) LAST,LAST1,LAST2,XKEFF,NDSP,IGHI,IGLO
      IF (LAST-25000) 1200,1200,1190
 1190 LMA=0
      GO TO 1400
C
C
 1200 CALL VRZ ( X(LNM), X(LRM),  X(LV), X(LR1), X(LZ1),  X(LDELR),
     1   X(LDELZ)    )
C
 1225 IF(KBEFF.EQ.1) GO TO 1275
      CALL CREAD ( X(LCHI), X(LC), X(LC1)    )
C
 1250 CALL IDACAV ( NR, NZ,  NZONE1, NZONE2,NMID,X(LID1),X(LID2),X(LIDM)
     1,X(LMA))
C
 1275 CALL NREAD ( LISTFX,  X(LNM), X(LRM),  X(LN), X(LNA), X(LXJ),
     1 X(LXJA),  X(LYJ),  X(LYJA),  X(LV)   )
C
      IF (KPAP.GE.0) GO TO 1300
      IGO=0
      WRITE(6,103)
      DO 2300 IN=1,NMAT
      ISP=IGO+NG
      IGO=NG*(IN-1)+1
      READ(5,101) (PNUF(IG),IG=IGO,ISP)
 2300 WRITE(6,104) (PNUF(IG),IG=IGO,ISP)
      LFEND=LF+NIJ-1
      DO 2200 IJ=LF,LFEND
 2200 X(IJ)=0.0
      IJG=LN-1
      DO 2000 IG=1,NG
      DO 2000 IJ=1,NIJ
      IJG=IJG+1
      K=X(IJ)
      IF(K.EQ.0) GO TO 2000
      LA=LF+IJ-1
      LB=NG*(K-1)+IG
      X(LA)=X(LA)+ X(IJG)*PNUF(LB)
 2000 CONTINUE
C     STORING IN FISSION POWER ARRAY F OF SUBROUTINE PAPCAL
 1300 CALL PAPCAL ( KPAP,  LISTFX,  X(LNM), X(LRM),  X(LV), X(LF),
     1   X(LFA)    )
C
C
C
 1400 RETURN
C
      END
```

```
$IBFTC PER2
      SUBROUTINE VRZ ( NM, RM,  V, R, Z,  DELR, DELZ   )
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON        /CALL1/                              NG,        NTYPS,
     1  NTYNG,      NFAST,    NUP,       NR,             NZ,        NIJ,
     2              KFLUX,   KBEFF,      KCTYP,          KIN,
     3              KP,       KS,        KAR,            KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
C
C     THE DIMENSION STATEMENTS
      DIMENSION     NM(1),             RM(1)
      DIMENSION     V(1),              R(1),              Z(1)
      DIMENSION     DELR(1),           DELZ(1)
C
C     THE FORMAT STATEMENTS
  110 FORMAT (7I10)
  115 FORMAT (8E16.7)
  120 FORMAT (1HL,7HDELR(I))
  121 FORMAT (1HL,7HDELZ(J))
  122 FORMAT (1HL,32HINCORRECT NUMBER OF MESH POINTS=,I5,52H INCLUDED TO
     1 OBTAIN EITHER R OR Z.  SHOULD HAVE BEEN,I5)
  130 FORMAT (1HL,5HKGEO=,I3)
  131 FORMAT (1HL,6HV(I,J))
  132 FORMAT (1HL,4HR(I))
  133 FORMAT (1HL,4HZ(J))
C
C     THE FUNCTION STATEMENTS
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C
 1000 NRP1=NR+1
      NZP1=NZ+1
      GO TO (1005,1005,1050), KIN
 1005 CALL BCREAD ( V(1), V(NIJ))
      IF(KBEFF.EQ.1) GO TO 1100
      GO TO (1006,1100,1050), KIN
 1006 CALL BCREAD (R(1),R(NRP1))
      IF (NZ-2) 1100,1100,1010
 1010 CALL BCREAD (Z(1),Z(NZP1))
      GO TO 1100
C
 1050 READ (5,110) KGEO
      WRITE (6,130) KGEO
      CALL RZREAD ( NM, RM,  ITEMP, R    )
      IF (ITEMP-NR) 1055,1060,1055
 1055 WRITE (6,122) ITEMP,NR
      GO TO 1200
 1060 IF (NZ-1) 1075,1075,1065
 1065 CALL RZREAD ( NM, RM,  ITEMP, Z    )
      IF (ITEMP-NZ) 1070,1080,1070
 1070 WRITE (6,122) ITEMP,NZ
      GO TO 1200
```

```
C
 1075 TEMP=1.0
      J=1
      GO TO 1086
 1080 J=1
 1085 TEMP=Z(J+1)-Z(J)
 1086 IJ=INDEX(NR,J,0)
      GO TO (1087,1088,1089), KGEO
 1087 AVE=1.C
      GO TO 1090
 1088 AVE=3.14159265
      GO TO 1090
 1089 AVE=3.14159265*4.0/3.0
 1090 DO 1091 I=1,NR
      IJ=IJ+1
 1091 V(IJ)=TEMP*(R(I+1)**KGEO-R(I)**KGEO)
      IF (NZ-J) 1100,1100,1092
 1092 J=J+1
      GO TO 1085
C
C
 1100 WRITE (6,131)
      DO 1105 J=1,NZ
      I=INDEX(NR,J,1)
      ITEMP=INDEX(NR,J,NR)
 1105 WRITE (6,115) (V(IJ), IJ=I,ITEMP)
      IF(KBEFF.EQ.1) GO TO 1200
      GO TO (1110,1150,1110), KIN
 1110 WRITE (6,132)
      WRITE (6,115) (R(I), I=1,NRP1)
      XIR=R(2)/2.
      IF (NZ-2) 1150,1115,1120
 1115 GO TO (1150,1150,1120), KIN
 1120 WRITE (6,133)
      WRITE (6,115) (Z(J), J=1,NZP1)
      XIZ=Z(2)/2.
C
 1150 ITEMP=NR-1
      GO TO (1155,1165,1155), KIN
 1155 DO 116C I=1,ITEMP
 1160 DELR(I)=0.5*(R(I+2)-R(I))
      GO TO 1170
 1165 CALL BCREAD (DELR(1),DELR(ITEMP))
 1170 WRITE (6,120)
      WRITE (6,115) (DELR(I), I=1,ITEMP)
      IF (NZ-2) 1200,1200,1175
 1175 ITEMP=NZ-1
      GO TO (1180,1190,1180), KIN
 1180 DO 1185 I=1,ITEMP
 1185 DELZ(I)=0.5*(Z(I+2)-Z(I))
      GO TO 1195
 1190 CALL BCREAD (DELZ(1),DELZ(ITEMP))
 1195 WRITE (6,121)
      WRITE (6,115) (DELZ(I), I=1,ITEMP)
C
C
C
 1200 RETURN
C
      END
```

```
$IBFTC PER21
      SUBROUTINE RZREAD ( NM, RM,   KOUNT,   R    )
C
C
C
C
C
C     THE DIMENSION STATEMENTS
      DIMENSION    NM(1),          RM(1)
      DIMENSION    R(1)
C
C     THE FORMAT STATEMENTS
  116 FORMAT (5(I5,E10.6))
C
C
C
 1000 R(1)=0.0
      KOUNT=0
      KSTOP=1
 1005 READ (5,116) (NM(I),RM(I), I=1,5)
      DO 1025 I=1,5
      IF (NM(I)) 1010,1025,1015
 1010 NM(I)=-NM(I)
      KSTOP=2
 1015 K1=KOUNT+1
      KOUNT=KOUNT+NM(I)
      DELL=RM(I )-R(K1)
      TEMP=NM(I )
      DELL=DELL/TEMP
      DO 1020 K=K1,KOUNT
 1020 R(K+1)=R(K)+DELL
      GO TO (1025,1050), KSTOP
 1025 CONTINUE
      GO TO 1005
C
C
C
 1050 RETURN
C
      END
```

```
$IBFTC PER3

       SUBROUTINE CREAD ( CHI, C, C1   )
C
C
C      READ IN CROSS SECTIONS
C
C
C      THE COMMON STATEMENTS
       COMMON        /CALL1/                                 NG,        NTYPS,
      1   NTYNG,     NFAST,     NUP,       NR,          NZ,        NIJ,
      2              KFLUX,   KBEFF,       KCTYP,       KIN,
      3              KP,        KS,        KAR,         KD,
      4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
       COMMON        /CHANG/                 NDOWN,
      1   NGM1,      NGP1,                   N
C
C      THE DIMENSION STATEMENTS
       DIMENSION     C(1),            CHI(1),          C1(1)
       DIMENSION  STOR(70)
C      STOR MUST CONTAIN THE LARGER OF (NTYPS+2*NG) OR (3*NG) LOCATIONS
C
C      THE FORMAT STATEMENTS
  101 FORMAT (1H )
  102 FORMAT(1HK,36H DELTA BUCKLING LOSS CROSS SECTIONS )
  103 FORMAT(1HJ,5X,18H  FIRST DIRECTION /(7E18.7))
  104 FORMAT(1HJ,5X,18H SECOND DIRECTION /(7E18.7))
  105 FORMAT(1HK,40X,10H MATERIAL ,I2)
  106 FORMAT(1HK,31H BUCKLING LOSS INFORMATION ****/18H  FIRST DIMENSION
     1 ,F10.6,18H SECOND DIMENSION ,F10.6/28H  FIRST DIMENSION PERTURBED
     2 ,F10.6,28H SECOND DIMENSION PERTURBED ,F10.6/18H  BUCKLING FACTOR
     3 ,F12.7)
  112 FORMAT (7E10.6)
  113 FORMAT (7E18.7)
  120 FORMAT (1HL,7HCHI(IG))
  121 FORMAT (1HL,26HRP1 FORMAT CROSS SECTIONS.)
  122 FORMAT (1HL,125HC(N,IG,ITY) FOR ITY OF SIG A, NU SIG F,(D* OR TR O
     1R TOT), SIG OUTSCATTER, NUP VALUES OF SIG UP, AND NDOWN VALUES OF
     2SIG DOWN.)
  123 FORMAT (1HL,80HTDSN FORMAT CROSS SECTIONS. TRANSPORT APPROXIMATION
     1. P-0 TRANSPORT CORRECTED CS.)
  124 FORMAT (1HJ,80HTDSN FORMAT CROSS SECTIONS WITH D* IN PLACE OF SIG
     1 TR. DIFFUSION APPROXIMATION./83H D*=(DELTA SIGTR/(3.*(SIGTR**2)))
     2 OR D*=(-DELTA DIFCOEF/(1.+DELTA DIFCOEF/DIFCOEF)))
  125 FORMAT(1HL,91HDELTAS COMPUTED INTERNALLY FROM DIFFERENCE BETWEEN P
     1ERTURBED AND UNPERTURBED CROSS SECTIONS)
  126 FORMAT (1HJ,73HTDSN FORMAT CROSS SECTIONS. TRANSPORT APPROXIMATION
     1. WITH P-0 AND P-1 CS.)
  127 FORMAT(1HL,15HDELTAS READ IN )
  128 FORMAT(1HK,59H TDSN FORMAT. P-1 CROSS SECTIONS WITH FACTOR OF 3 IN
     1CLUDED )
  129 FORMAT(1HK,33H DELTA OUTSCATTER CROSS SECTIONS )
  130 FORMAT(1HL,116H C(N,IG,ITY) FOR ITY OF 0.0 , P-0 G-TO-G, TRANSPORT
     1, P-1 G-TO-G, NUP VALUES OF SIG UP, AND NDOWN VALUES OF SIG DOWN.)
C
C      THE FUNCTION STATEMENTS
```

```
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C
 1000 READ (5,112) (CHI(IG), IG=1,NG)
      WRITE (6,120)
      WRITE (6,113) (CHI(IG), IG=1,NG)
      NPOCS=NTYNG*NMAT
C     JOG=0   P-0 CROSS SECTIONS
C     JOG=1   P-1 CROSS SECTIONS
      DO 290 N=1,NMAT
      NOWP1=0
      JOG=0
      IF(KCTYP.NE.0) GO TO 1025
  279 JDOUBL=1
C     JDOUBL=1  READ IN PERTURBED CROSS SECTIONS
C     JDOUBL=2  READ IN UNPERTURBED CROSS SECTIONS
  281 DO 280 JIG=1,NG
      JK1=INDEX(NTYPS,JIG,1)
      JK2=INDEX(NTYPS,JIG,NTYPS)
      IF(JDOUBL.EQ.2) GO TO 282
      READ(5,112) (C1(JK),JK=JK1,JK2)
      GO TO 280
  282 READ(5,112) (STOR(JIK),JIK=1,NTYPS)
      HOLD=STOR(3)
      LEND=70-NG+JIG
      STOR(LEND)=STOR(3)
      KONT=0
      DO 270 KK=JK1,JK2
      KONT=KONT+1
      C1(KK)=C1(KK)-STOR(KONT)
      IF(ABS(C1(KK)).LT..0000005) C1(KK)=0.0
  270 CONTINUE
      IF(KAPROX.GT.0) GO TO 280
      C1(JK1+2)= C1(JK1+2)/(3.0*HOLD*HOLD)
C     SEE EQ. 32 IN WRITEUP
  280 CONTINUE
      IF(JDOUBL.EQ.2) GO TO 210
      JDOUBL=2
      GO TO 281
C
C
 1025 GO TO (1031,1100), KCTYP
  210 IF(JOG.EQ.1) GO TO 1032
      JOG=1
      WRITE(6,125)
      IF(KAPROX.EQ.0) WRITE(6,124)
      IF(KAPROX.EQ.1) WRITE(6,123)
      IF(KAPROX.EQ.2) WRITE(6,126)
      GO TO 1032
 1031 WRITE (6,127)
 1032 WRITE(6,105) N
      DO 1055 IG=1,NG
      K1=INDEX(NTYPS,IG,1)
      K2=INDEX(NTYPS,IG,NTYPS)
      IF(KCTYP.EQ.0) GO TO 300
      READ (5,112) (C1(K), K=K1,K2)
  300 WRITE(6,113) (C1(K),K=K1,K2)
C
C     TRANSFER TDSN CROSS SECTIONS FROM INPUT ARRAY C1 TO PERMANENT
C     STORAGE (ALTERED FORMAT) IN C
      L=INDEX(NTYNG,N,K1)
      LL=L+1
      IF(NOWP1.EQ.1) L=L+NPOCS
      DO 1055 K=K1,K2
      LLL=NPOCS+LL
      IF(NOWP1.EQ.1.AND.K.EQ.(K1+1)) GO TO 1055
      IF(K.GT.(K1+2)) GO TO 1045
 1035 C(L)=C1(K)
      GO TO 1055
 1045 IF(NOWP1.EQ.1.AND.K.EQ.(K1+NUP+3)) C(LLL+2)=C1(K)
      IF(NOWP1.EQ.1) GO TO 1046
      IF(KAPROX.EQ.2.AND.K.EQ.(K1+NUP+3)) C(LLL)=C1(K)
 1046 IF (K-(K1+NUP+3)) 1050,1055,1035
 1050 C(L+1)=C1(K)
 1055 L=L+1
      IF(KAPROX.LE.1) GO TO 1054
      IF(NOWP1.EQ.1) GO TO 1054
      WRITE(6,128)
      NOWP1=1
      GO TO 279
 1054 WRITE(6,101)
C     READ OUTSCATTER
      WRITE(6,129)
      READ (5,112) (C1(K), K=1,NG)
      IF(KCTYP.NE.0) GO TO 250
      READ(5,112) (STOR(K),K=1,NG)
      DO 240 JX=1,NG
  240 C1(JX)=C1(JX)-STOR(JX)
  250 WRITE(6,113) (C1(K),K=1,NG)
      DO 1060 IG=1,NG
      K1=INDEX(NTYPS,IG,4)
      L=INDEX(NTYNG,N,K1)
 1060 C(L)=C1(IG)
      H1=0.0
      IF(KCTYP.EQ.1) GO TO 290
C     READ TRANSVERSE LEAKAGE (BUCKLING) LOSS INFORMATION
C     CALCULATE BUCKLING LOSS CROSS SECTIONS AND STORE IN C
C     GAMMA = 0.71045608
      ITAG=0
      IF(N.GT.1) GO TO 1200
      READ(5,112) H1,H2,BF,HP1,HP2
      WRITE(6,106) H1,H2,HP1,HP2,BF
      IF(H1.EQ.0.0) GO TO 1150
      H=H1
      HH=H2
      IST=70-NG
      NS=NG*NMAT*NTYPS
      IF(KAPROX.EQ.2) NS=2*NS
      IA=NS+1
      IB=3*NG*NMAT+IA-1
      DO 1201 I=IA,IB
 1201 C(I)=0.0
 1200 DO 1202 ILK=1,NG
      ITAG=ITAG+1
      INOW=IST+ILK
      IK=NG*(N-1)+NS+ILK
      HANGON=(H*STOR(INOW)+1.42091216)**2
      C(IK)=(STOR(INOW)*BF*BF/HANGON)-C(IK)
      IF(H2.EQ.0.0) GO TO 1203
      HANGON=(HH*STOR(INOW)+1.42091216)**2
      IKK=NG*NMAT+IK
      C(IKK)=(STOR(INOW)*BF*BF/HANGON)-C(IKK)
 1203 JJ=INDEX(NTYPS,ILK,3)
      JJJ=INDEX(NTYNG,N,JJ)
      IF(KAPROX.EQ.2) JJJ=JJJ+NPOCS
      IF(KAPROX.GE.1) FACTOR=1.0
      IF(KAPROX.EQ.0) FACTOR=3.*(STOR(INOW)**2)
      IKKK=2*NG*NMAT+IK
C
C     STORE 1/(3*(SIGTR**2)) IN C -- USED FOR CURRENT (FICK,S LAW)
C     APPROXIMATION FOR KAPROX=0.   J = -DELPHI/(3*SIGTR)
      C(IKKK)=1.0/FACTOR
      STOR(INOW)=STOR(INOW)+FACTOR*C(JJJ)
 1202 CONTINUE
      IF(HP1.NE.0.0) H=HP1
      IF(HP2.NE.0.0) HH=HP2
      IF(ITAG.EQ.NG) GO TO 1200
  290 WRITE(6,101)
      GO TO 1150
C
C     PROVISION FOR DIFFUSION THEORY CROSS SECTIONS
 1100 WRITE (6,121)
      DO 1105 N=1,NMAT
      CALL XCHANG (C1, C   )
 1105 WRITE (6,101)
C
 1150 WRITE (6,122)
      JT=0
 1151 DO 1160 N=1,NMAT
      WRITE(6,105) N
      DO 1155 IG=1,NG
      K1=INDEX (NTYPS,IG,1)
      K2=INDEX (NTYPS,IG,NTYPS)
      K1=INDEX (NTYNG,N,K1)
      K2=INDEX (NTYNG,N,K2)
      IF(JT.EQ.0) GO TO 1156
      K1=K1+NPOCS
      K2=K2+NPOCS
 1156 WRITE(6,113) (C(K),K=K1,K2)
 1155 CONTINUE
      IF(KAPROX.EQ.2.AND.JT.EQ.0) GO TO 1160
      IF(H1.EQ.0.0) GO TO 1160
      WRITE(6,102)
      IST=IA
      ISP=IST+NG-1
      IIST=NG*NMAT+IST
      IISP=IIST+NG-1
      WRITE(6,103) (C(IG),IG=IST,ISP)
      IF(H2.NE.0.0) WRITE(6,104) (C(IG),IG=IIST,IISP)
 1160 WRITE (6,101)
      IF(KAPROX.LE.1.OR.JT.EQ.1) GO TO 1161
      WRITE(6,130)
      JT=1
      GO TO 1151
C
C
C
 1161 RETURN
C
      END
```

```
$IBFTC PER4

      SUBROUTINE XCHANG ( C1, C   )
C
C     PROVISION FOR DIFFUSION THEORY CROSS SECTIONS
C
 1400 RETURN
      END
```

```
$IBFTC PER5    DECK

      SUBROUTINE IDACAV (NR,NZ,NZONRA,NZONZA,NMID,    NMRA,NMZA,IDM,    ID
     1MAP )
C
C     SUBROUTINE IDACAV OBTAINS THE IDENIFICATION NUMBER MAP.
C
C
C
C
C
C     THE DIMENSION STATEMENTS
      DIMENSION    NMRA(1),      NMZA(1),      IDM(1)
      DIMENSION    IDMAP(1),  MATCHG(25)
C     MATCHG MUST CONTAIN NMID LOCATIONS
C
C     THE FORMAT STATEMENTS
  110 FORMAT (7I10)
  115 FORMAT (14I5)
  116 FORMAT (43I3)
  120 FORMAT (1HL,11HMAP MA(I,J))
  121 FORMAT(1HL,39HMAP MA(I,J) WITH ONLY PERTURBED ENTRIES )
  126 FORMAT(14I5)
  127 FORMAT(1HK,37HTHE NON-ZERO MATERIALS ARE PERTURBED /24I5)
```

```
C
C     THE FUNCTION STATEMENTS
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C
      NIJ=NR*NZ
      READ(5,126)(MATCHG(I),I=1,NMID)
      WRITE(6,127)(MATCHG(I),I=1,NMID)
 1000 READ (5,110) (NMRA(I), I=1,NZONRA)
      IF (NZONZA) 1005,1005,1010
 1005 NZONZA=1
      NMZA(1)=1
      GO TO 1025
 1010 READ (5,110) (NMZA(J), J=1,NZONZA)
C
 1025 K=0
      J=0
      DO 1050 JJ=1,NZONZA
      READ (5,115) (IDM(II), II=1,NZONRA)
      K=K+NMZA(JJ)
 1030 J=J+1
      I=0
      DO 1045 II=1,NZONRA
      L=NMRA(II)
      DO 1035 KK=1,L
      I=I+1
      IJ=INDEX(NR,J,I)
 1035 IDMAP(IJ)=IDM(II)
 1045 CONTINUE
      IF (J-K) 1030,1050,1050
 1050 CONTINUE
C
C
 1075 WRITE (6,120)
      JDOG=0
C     JDOG=0  REGULAR MAP WITH NMID MATERIALS
C     JDOG=1  MAP WITH NMAT PERTURBED MATERIALS
 1076 DO 1080 J=1,NZ
      K=INDEX(NR,J,1)
      L=INDEX(NR,J,NR)
 1080 WRITE (6,116) (IDMAP(IJ), IJ=K,L)
      IF(JDOG.EQ.1) GO TO 1100
      DO 1200 IK=1,NIJ
      KKK=IDMAP(IK)
 1200 IDMAP(IK)=MATCHG(KKK)
      JDOG=1
      WRITE(6,121)
      GO TO 1076
C
C
C
 1100 RETURN
C
      END
```

```
$IBFTC PER61
      SUBROUTINE READIT ( NMR, RMI, IJGS, NIJ, SV, KEND  )
C
C
C
C
C
C     THE DIMENSION STATEMENTS
      DIMENSION    NMR(1),        RMI(1),        SV(1)
C
C     THE FORMAT STATEMENTS
  118 FORMAT (I5,E10.6,I5,E10.6,I5,E10.6,I5,E10.6,I5,E10.6)
  119 FORMAT (1HL,51HTOO MANY VALUES FOR SUBSCRIPTED VARIABLE SV READ IN
     1   )
C
C
C
 1000 L=0
      KEND=0
 1005 READ (5,118) (NMR(I),RMI(I), I=1,5)
      I=1
 1010 IF (NMR(I)) 1050,1050,1015
 1015 L1=L+1
      L=L+NMR(I)
      IF (L-NIJ) 1025,1025,1020
 1020 WRITE (6,119)
      KEND=1
      GO TO 1100
 1025 ITEMP1=L1+IJGS
      ITEMP2=L+IJGS
      DO 1030 IJG=ITEMP1,ITEMP2
 1030 SV(IJG)=RMI(I)
      IF (L-NIJ) 1035,1100,1100
 1035 IF (I-5) 1040,1005,1005
 1040 I=I+1
      GO TO 1010
 1050 IF (L-NIJ) 1055,1100,1100
 1055 IF (L) 1060,1060,1065
 1060 ITEMP1=IJGS+1
      GO TO 1070
 1065 ITEMP1=ITEMP2+1
 1070 ITEMP2=IJGS+NIJ
      DO 1075 IJG=ITEMP1,ITEMP2
 1075 SV(IJG)=0.0
C
C
C
 1100 RETURN
C
      END
```

```
$IBFTC PER6

      SUBROUTINE NREAD ( LISTFX,  NM, RM,  XN, XNA, XJ,XJA,YJ,YJA,V)
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON       /CALL1/                              NG,       NTYPS,
     1  NTYNG,     NFAST,    NUP,       NR,       NZ,       NIJ,
     2             KFLUX,   KBEFF,      KCTYP,    KIN,
     3             KP,      KS,         KAR,      KD,
     4 KAPRUX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
C
C     THE DIMENSION STATEMENTS
      DIMENSION    NM(1), V(1),    RM(1),    XJ(1),     XJA(1)
      DIMENSION    XN(1),          XNA(1),  YJ(1),  YJA(1)
C
C     THE FORMAT STATEMENTS
  115 FORMAT (8E16.7)
  116 FORMAT (4H IG=,I3)
  117 FORMAT (4H  J=,I3)
  120 FORMAT (1HL,18HTHE FLUXES XN(I,J))
  121 FORMAT (1HL,27HTHE ADJOINT FLUXES XNA(I,J))
  122 FORMAT(1HL,36HTHE FIRST DIRECTION CURRENTS XJ(I,J))
  123 FORMAT(1HL,45HTHE FIRST DIRECTION ADJOINT CURRENTS XJA(I,J))
  124 FORMAT(1HL,37HTHE SECOND DIRECTION CURRENTS YJ(I,J))
  125 FORMAT(1HL,46HTHE SECOND DIRECTION ADJOINT CURRENTS YJA(I,J))
C
C     THE FUNCTION STATEMENTS
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C     READ IN THE REGULAR THEN THE ADJOINT 1) FLUX 2) FIRST DIRECTION
C     CURRENT 3) SECOND DIRECTION CURRENT
C
      IF(KBEFF.EQ.0) GO TO 190
      CALL REED(XNA)
      KRITE=2
      GO TO 1106
  190 CALL REED(XN)
      CALL REED(XNA)
      IF(KAPRUX.EQ.0) GO TO 1100
      CALL REED(XJ)
      CALL REED(XJA)
      NIT=NIJ*NG
      DO 200 J=1,NIT
      YJ(J)=XJ(J)
  200 YJA(J)=XJA(J)
      IF(NZ.LE.1) GO TO 1100
      CALL REED(YJ)
      CALL REED(YJA)
C
C
C
 1100 IF (LISTFX) 1200,1200,1104
 1104 KRITE=1
 1105 WRITE(6,120)
      GO TO 1300
 1106 WRITE(6,121)
      GO TO 1300
 1107 WRITE(6,122)
      GO TO 1300
 1108 WRITE(6,123)
      GO TO 1300
 1109 WRITE(6,124)
      GO TO 1300
 1111 WRITE(6,125)
 1300 DO 1110 IG=1,NG
      WRITE (6,116) IG
      DO 1110 J=1,NZ
      WRITE (6,117) J
      K1=INDEX(NR,J,1)
      K2=INDEX(NR,J,NR)
      K1=INDEX(NIJ,IG,K1)
      K2=INDEX(NIJ,IG,K2)
      GO TO (1201,1202,1203,1204,1205,1206), KRITE
 1201 WRITE(6,115) (XN (IJ),IJ=K1,K2)
      GO TO 1110
 1202 WRITE(6,115) (XNA(IJ),IJ=K1,K2)
      GO TO 1110
 1203 WRITE(6,115) (XJ (IJ),IJ=K1,K2)
      GO TO 1110
 1204 WRITE(6,115) (XJA(IJ),IJ=K1,K2)
      GO TO 1110
 1205 WRITE(6,115) (YJ (IJ),IJ=K1,K2)
      GO TO 1110
 1206 WRITE(6,115) (YJA(IJ),IJ=K1,K2)
 1110 CONTINUE
      KRITE=KRITE+1
      IF(KRITE.EQ.2) GO TO 1106
      IF(KAPROX.EQ.0) GO TO 1200
      IF(KRITE.EQ.3) GO TO 1107
      IF(KRITE.EQ.4) GO TO 1108
      IF(NZ.LE.1) GO TO 1200
      IF(KRITE.EQ.5) GO TO 1109
      IF(KRITE.EQ.6) GO TO 1111
C
C
C
 1200 RETURN
C
      END
```

```
$IBFTC PER666

      SUBROUTINE REED(XYZ)
      COMMON        /CALL1/                                   NG,        NTYPS,
     1   NTYNG,     NFAST,     NUP,       NR,       NZ,        NIJ,
     2              KFLUX,    KBEFF,      KCTYP,    KIN,
     3              KP,        KS,        KAR,      KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      DIMENSION XYZ(1)
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
  100 FORMAT(7E10.6)
C
C
      GO TO (200,200,160,150), KFLUX
  200 DO 100C IG=1,NG
      K1=INDEX(NIJ,IG,1)
      K2=INDEX(NIJ,IG,NIJ)
 1000 CALL BCREAD(XYZ(K1),XYZ(K2))
      RETURN
  150 DO 200C IG=1,NG
      K1=INDEX(NIJ,IG,1)
      K2=INDEX(NIJ,IG,NIJ)
 2000 READ(5,100) (XYZ(I),I=K1,K2)
      RETURN
  16C K1=0
      DO 300C IG=1,NG
      CALL READIT(NM,RM,K1,NIJ,XYZ,KEND)
      IF(KEND) 3000,3000,1700
 300C K1=K1+NIJ
 1700 RETURN
      END
```

```
$IBFTC PER7

      SUBROUTINE PAPCAL ( KPAP,  LISTFX,  NM, RM,  V, F, FA  )
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON        /CALL1/                                   NG,        NTYPS,
     1   NTYNG,     NFAST,     NUP,       NR,       NZ,        NIJ,
     2              KFLUX,    KBEFF,      KCTYP,    KIN,
     3              KP,        KS,        KAR,      KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON        /CALL2/    PAP
C
C     THE DIMENSION STATEMENTS
      DIMENSION     NM(1),          RM(1)
      DIMENSION     V(1)
      DIMENSION     F(1),           FA(1)
```

```
C
C       THE FORMAT STATEMENTS
  112 FORMAT (7E10.6)
  115 FORMAT (8E16.7)
  117 FORMAT (4H  J=,I3)
  120 FORMAT (1HL,24HTHE POWER DENSITY F(I,J))
  121 FORMAT (1HL,33HTHE ADJOINT POWER DENSITY FA(I,J))
  150 FORMAT (1HL,32HADJOINT POWER TIMES POWER EQUALS,E16.8)
C
C       THE FUNCTION STATEMENTS
        INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C
 1000 IF (KPAP) 1015,1C10,1150
 1015 DO 1016 II=1,NIJ
 1016 FA(II)=F(II)
C
 1010 CALL BCREAD (F(1),F(NIJ))
      IF(KPAP.GE.0) GO TO 1018
      DO 1017 JJ=1,NIJ
      IF(FA(JJ).NE.0.0) F(JJ)=FA(JJ)
 1017 CONTINUE
 1018 CALL BCREAD (FA(1),FA(NIJ))
C
C
C
 1100 IF (LISTFX) 1125,1125,1105
 1105 WRITE (6,120)
      DO 1110 J=1,NZ
      WRITE (6,117) J
      K1=INDEX(NR,J,1)
      K2=INDEX(NR,J,NR)
 1110 WRITE (6,115) (F(IJ), IJ=K1,K2)
      WRITE (6,121)
      DO 1115 J=1,NZ
      WRITE (6,117) J
      K1=INDEX(NR,J,1)
      K2=INDEX(NR,J,NR)
 1115 WRITE (6,115) (FA(IJ), IJ=K1,K2)
C
C
 1125 PAP=0.C
      DO 113C IJ=1,NIJ
 1130 PAP=PAP+F(IJ)*FA(IJ)*V(IJ)
      GO TO 1175
C
C
 1150 READ (5,112) PAP
C
C
 1175 WRITE (6,150) PAP
C
C
C
 1200 RETURN
C
      END
```

```
$IBFTC PER8
      SUBROUTINE PERTUR ( MA, V, DELR, DELZ,C,CHI,XKEFF,XN, XNA, F,
     1    P, S, A, R, D1, D2   , XJ,  XJA,  YJ,  YJA, TL1, TL2, S1X)
C
C     TO COMPUTE PERTURBATION SOURCES AND LOSSES.
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON        /CALL1/                             NG,       NTYPS,
     1   NTYNG,     NFAST,    NUP,      NR,       NZ,       NIJ,
     2              KFLUX,   KBEFF,     KCTYP,    KIN,
     3              KP,       KS,       KAR,      KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON        /CALL2/   PAP
      COMMON        /CALL3/   KXTRA1,   KXTRA2,   KXTRA3,   KXTRA4,
     1   KXTRA5,    EXTRA1,   EXTRA2,   EXTRA3,   EXTRA4,   EXTRA5
C
C     THE DIMENSION STATEMENTS
      DIMENSION     MA(1),         V(1),
     1              DELR(1),       DELZ(1)
      DIMENSION     C(1),          CHI(1)
      DIMENSION     XN(1),         XNA(1),        F(1)
      DIMENSION     P(1),          S(1),          A(1),
     1              R(1),          D1(1),         D2(1), TL1(1), TL2(1)
      DIMENSION XJ(1), XJA(1), YJ(1), YJA(1), S1X(1)
C
C     THE FUNCTION STATEMENTS
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
  101 FORMAT(7E10.6)
  102 FORMAT(1H1,5H THE ,I2, 40H DELAYED SPECTRA ARE (FROM ENERGY GROUP
     1,I2, 19H THRU ENERGY GROUP ,I2,5H )***)
  103 FORMAT(1HK,18H DELAYED SPECTRUM ,I2,/(7E18.7))
  104 FORMAT(7I10)
C
      IF(KBEFF.EQ.0) GO TO 1000
      NIT=NIJ*NG
      DO 900 IJK=1,NIT
      XNA(IJK)=XNA(IJK)/XKEFF
  900 CONTINUE
      DO 1006 KJ=1,NIJ
C     THIS PORTION OF YJA IS USED TO STORE REGULAR PRODUCTION FOR
C     BETA EFFECTIVE CALCULATION
 1006 YJA(KJ)=F(KJ)
      GO TO 1400
 1000 DO 1005 IG=1,NG
      P(IG)=0.0
      S(IG)=0.0
      A(IG)=0.0
      R(IG)=0.0
      S1X(IG)=0.0
      TL1(IG)=0.0
      TL2(IG)=0.0
      D1(IG)=0.0
 1005 D2(IG)=0.0
      NOWP1=C
C
C
C     THE FISSION PERTURBATION SOURCE
C
C
      IF (KP) 1100,1100,1010
 1010 DO 1025 IJ=1,NIJ
 1025 F(IJ)=C.0
      DO 105C IG=1,NG
      K1=INDEX(NTYPS,IG,2)
      IJG=INDEX(NIJ,IG,0)
      DO 105C IJ=1,NIJ
      IJG=IJG+1
      K=MA(IJ)
      IF (K) 1050,1050,1030
 1030 ITEMP=INDEX(NTYNG,K,K1)
      F(IJ)=F(IJ)+C(ITEMP)*XN(IJG)
 1050 CONTINUE
C
C
      DO 1075 IG=1,NG
      IJG=INDEX(NIJ,IG,0)
      DO 107C IJ=1,NIJ
      IJG=IJG+1
 1070 P(IG)=P(IG)+XNA(IJG)*F(IJ)*V(IJ)
 1075 P(IG)=-CHI(IG)*P(IG)/(PAP*XKEFF)
C
C
C     THE SCATTERING PERTURBATION SOURCE
C
C
C
C     NOWP1=-2      P-1 CROSS SECTIONS**NZ.GT.1
C     NOWP1=-1      P-1 CROSS SECTIONS**NZ.LE.1
C     NOWP1= 0      DIFFUSION APPROXIMATION
C     NOWP1=1       P-0 TRANSPORT CORRECTED CROSS SECTIONS**NZ.LE.1
C     NOWP1=2       P-0 TRANSPORT CORRECTED CROSS SECTIONS**NZ.GT.1
C
 1100 IF (KS) 1350,1350,1105
 1105 IF (NG-1) 1350,1350,1110
 1110 DO 130C IG=1,NG
      KSET=1
      DO 1115 IJ=1,NIJ
 1115 F(IJ)=C.0
C
C     UP SCATTERING
C
      IF (NUP) 1200,1200,1125
 1125 IF (IG-NFAST) 1200,1130,1130
 1130 IF (IG-NG) 1135,1200,1200
 1135 KSET=2
```

```
      IGG=IG
      ITY=NUP+5
 1140 IGG=IGG+1
      ITY=ITY-1
      K1=INDEX(NTYPS,IG,ITY)
      IJG=INDEX(NIJ,IGG,0)
C     IJG IS THE GROUP SCATTERED FROM
      IJT=INDEX(NIJ,IG,0)
      DO 115C IJ=1,NIJ
      IJG=IJG+1
      K=MA(IJ)
      IF (K) 1150,1150,1145
 1145 ITEMP=INDEX(NTYNG,K,K1)
      F(IJ)=F(IJ)+C(ITEMP)*XN(IJG)
      IF(NOWP1.GE.0) GO TO 1150
      IJT=IJT+1
      JMPU=NTYNG*NMAT+ITEMP
      IF(NOWP1.EQ.-1) F(IJ)=F(IJ)+C(JMPU)*XJ(IJG)
      IF(NOWP1.EQ.-2) F(IJ)=F(IJ)+C(JMPU)*
     1(XJ(IJG)*XJA(IJT)+YJ(IJG)*YJA(IJT))
 1150 CONTINUE
      IF (IGG-NG) 1155,1200,1200
 1155 IF (ITY-5) 1200,1200,1140
C
C     DOWN SCATTERING
C
 1200 CONTINUE
      IF(IG.LE.1) GO TO 1275
 1205 KSET=2
      IGG=0
      ITY=(NUP+4)+IG
 1210 IGG=IGG+1
      ITY=ITY-1
      IF (NTYPS-ITY) 1210,1215,1215
 1215 K1=INDEX(NTYPS,IG,ITY)
      IJG=INDEX(NIJ,IGG,0)
      IJT=INDEX(NIJ,IG,0)
      DO 125C IJ=1,NIJ
      IJG=IJG+1
      IJT=IJT+1
      K=MA(IJ)
      IF (K) 1250,1250,1245
 1245 ITEMP=INDEX(NTYNG,K,K1)
      IF(NOWP1.LT.0) GO TO 1246
      F(IJ)=F(IJ)+C(ITEMP)*XN(IJG)
      GO TO 1250
 1246 JMPD=NTYNG*NMAT+ITEMP
      IF(NOWP1.EQ.-1) F(IJ)=F(IJ)+C(JMPD)*XJ(IJG)
      IF(NOWP1.EQ.-2) F(IJ)=F(IJ)+C(JMPD)*
     1(XJ(IJG)*XJA(IJT)+YJ(IJG)*YJA(IJT))
 1250 CONTINUE
      IF(IGG-(IG-1)) 1210,1280,1280
C
 1275 IF(KSET.EQ.1) GO TO 1300
C
C
```

```
 1280 IJG=INDEX(NIJ,IG,0)
C     IJG IS THE GROUP SCATTERED TO
      IF(NOWP1+1) 1283,1282,1281
 1281 DO 1285 IJ=1,NIJ
      IJG=IJG+1
      S(IG)=S(IG)+XNA(IJG)*F(IJ)*V(IJ)
 1285 CONTINUE
      GO TO 1301
 1282 DO 1286 IJ=1,NIJ
      IJG=IJG+1
 1286 S1X(IG)=S1X(IG)+XJA(IJG)*F(IJ)*V(IJ)
      GO TO 1301
 1283 DO 1287 IJ=1,NIJ
 1287 S1X(IG)=S1X(IG)+F(IJ)*V(IJ)
C
 1301 IF(KAPROX.LE.1.OR.NOWP1.LT.0) GO TO 1299
      IF(IG.NE.NG) GO TO 1300
      NOWP1=-1
      IF(NZ.GT.1) NOWP1=-2
      GO TO 1105
 1299 S(IG)=-(S(IG)-    S1X(IG))/PAP
C     TDSN USES P-1 CS WITH A FACTOR OF 3 INCLUDED
 1300 CONTINUE
C
C
C     ABSORPTION                ***
C     SCATTERING(REMOVAL)       ***                         PERTURBATION
C     LEAKAGE (TRANSVERSE)      ***
C     LEAKAGE(TRANSPORT)        ***                            LOSSES
C     WITHIN GROUP SCATTERING***
C
C
 1350 IF (KAR) 1400,1400,1355
 1355 DO 1380 IG=1,NG
      K1=INDEX(NTYPS,IG,1)
      IJG=INDEX(NIJ,IG,0)
      DO 1375 IJ=1,NIJ
      IJG=IJG+1
      K=MA(IJ)
      IF (K) 1375,1375,1370
 1370 ITEMP=INDEX(NTYNG,K,K1)
      LTEM=NTYNG*NMAT+NG*(K-1)+IG
      IF(KAPROX.EQ.2) LTEM=LTEM+NTYNG*NMAT
      LLTEM=LTEM+NG*NMAT
      TEMP=XNA(IJG)*XN(IJG)*V(IJ)
      IF(KAPROX.EQ.0) GO TO 1371
      TEMPJ=3.*XJA(IJG)*XJ(IJG)*V(IJ)
      IF(NZ.LE.1) GO TO 1374
      TEMPJ=TEMPJ+3.*YJA(IJG)*YJ(IJG)*V(IJ)
 1374 CONTINUE
      IF(KAPROX.EQ.2) TEMP=TEMP-TEMPJ
 1371 A(IG)=A(IG)+TEMP*C(ITEMP)
      R(IG)=R(IG)+TEMP*C(ITEMP+3)
      TEMB=TEMP
      IF(KAPROX.EQ.1) TEMB=TEMP-TEMPJ
      TL1(IG)=TL1(IG)+TEMB*C(LTEM)
      TL2(IG)=TL2(IG)+TEMB*C(LLTEM)
      SIGGG=C(ITEMP+2)-C(ITEMP)-C(ITEMP+3)
      JTEMP=ITEMP+NTYPS*NG*NMAT
      SIGGG=SIGGG-(C(JTEMP+3)/3.0)
      IF(KAPROX.EQ.1) D1(IG)=D1(IG)-TEMPJ*C(ITEMP+2)
      IF(KAPROX.EQ.2) D1(IG)=D1(IG)-TEMPJ*SIGGG
C     D1 IS THE TRANSPORT CROSS SECTION FOR KAPROX.EQ.1
C     D1 IS THE J-WGT GTOG CROSS SECTION FOR KAPROX.EQ.2
C     D2 IS THE LEAKAGE  CROSS SECTION FOR KAPROX.EQ.2
      IF(KAPROX.LE.1) GO TO 1375
      D2(IG)=D2(IG)-TEMPJ*(C(ITEMP+2)-(C(JTEMP+3)/3.0))
      D2(IG)= D2(IG)+SIX(IG)
 1375 CONTINUE
      D1(IG)= D1(IG)/PAP
      D2(IG)= D2(IG)/PAP
      TL1(IG)=TL1(IG)/PAP
      TL2(IG)=TL2(IG)/PAP
      A(IG)=A(IG)/PAP
 1380 R(IG)=R(IG)/PAP
C
C
C
C
C
C
 1400 IF(KBEFF.EQ.0) GO TO 1800
C     NOTE THE DELAYED NEUTRON INFORMATION IS READ INTO OTHER ARRAYS
C     ( S AND YJA ) TO SAVE STORAGE
      READ(5,101)(S(I),I=1,NDSP)
C     S(I) CONTAINS THE DELAYED NEUTRON FRACTION
      ISPAN=IGLO-IGHI+1
      DO 1710 IJK=1,NDSP
      LST=INDEX(ISPAN,IJK,1)+ NIJ
      LSP=LST+ISPAN-1
C     THIS PORTION OF YJA CONTAINS THE DELAYED SPECTRA
 1710 READ(5,101) (YJA(K),K=LST,LSP)
      WRITE(6,102) NDSP,IGHI,IGLO
      DO 1720 IJK=1,NDSP
      LST=INDEX(ISPAN,IJK,1)+ NIJ
      LSP=LST+ISPAN-1
 1720 WRITE(6,103) IJK,(YJA(K),K=LST,LSP)
C
C     CALCULATE THE IMPORTANCE OF THE DELAYED SPECTRA
C
      NST=NIJ+NDSP*ISPAN
      DO 1750 ID=1,NDSP
      NST=NST+1
      YJA(NST)=0.0
C     THIS PORTION OF YJA CONTAINS BETA
      DO 1730 IJ=1,NIJ
      TEEM=0.0
      IGA=INDEX(ISPAN,ID,1)
      IGA=IGA+NIJ
      DO 1740 IG=IGHI,IGLO
      LOCXN=INDEX(NIJ,IG,IJ)
      TEEM=TEEM+YJA(IGA)*XNA(LOCXN)*S(ID)
 1740 IGA=IGA+1
      TEEM=TEEM*YJA(IJ)*V(IJ)
 1730 YJA(NST)=YJA(NST)+TEEM
      YJA(NST)=YJA(NST)/PAP
 1750 CONTINUE
C
 1800 RETURN
C
      END
```

```
$IBFTC PER81

      SUBROUTINE DIFUSE (MA,V,DELR,DELZ,C,XN,XNA,D1,D2,TL1,TL2,A)
      COMMON        /CALL1/                             NG,       NTYPS,
     1   NTYNG,     NFAST,    NUP,       NR,        NZ,       NIJ,
     2              KFLUX,   KBEFF,      KCTYP,     KIN,
     3              KP,       KS,        KAR,       KD,
     4 KAPRUX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON        /CALL2/   PAP
      DIMENSION MA(1),V(1),DELR(1),DELZ(1),C(1),XN(1),XNA(1),D1( 1),D2(1)
      DIMENSION TL1(1),TL2(1),A(1)
      INDEX(LENGTH,INDEXO,INDEXL)=LENGTH*(INDEXO-1)+INDEXL
C
C
C     THE DIFFUSION PERTURBATION LOSS
C
C
  104 FORMAT(7I10)
  105 FORMAT(1HK,69H  BOUNDARY CONDITICNS FOR RIGHT, LEFT, TOP, AND BOTT
     1OM BOUNDARIES ** ,4I2)
 1405 DO 1595 IG=1,NG
      K1=INDEX(NTYPS,IG,3)
      IF( IG.NE.1.OR.KD.NE.-1) GO TO 1409
      READ(5,104) KRBC,KLBC,KTBC,KBBC
      WRITE(6,105) KRBC,KLBC,KTBC,KBBC
C
C     FOR THE FIRST DIRECTION (MUST BE AT LEAST THREE MESH INTERVALS)
C        THREE POINT VANDERMONDE SOLUTION FOR FLUX DERIVATIVE
C     THE DIFFERENCE IN GRADIENT ACROSS PARALLEL FACES OF A MESH
C     INTERVAL IS USED AS THE NET GRADIENT OF THE INTERVAL
C
 1409 XI=XIR
      DO 149C I=1,NR
      IF( I.EQ.1) DXI=2.*XI
      IF (I-1) 1410,1410,1415
 1410 I1=I+1
      GO TO 1420
 1415 I1=I-1
      DXI=2.*DELR(I1)-DXI
      IF (I-NR) 1420,1425,1425
 1420 DEL1=DELR(I)+DELR(I1)
      DEL2=DELR(I)*DELR(I1)
      DEL3=DEL1*DEL2
```

```
C
 1425 DO 1490 J=1,NZ
      IJ=INDEX(NR,J,I)
      K=MA(IJ)
      IF (K) 1490,1490,1430
 1430 ITEMP=INDEX(NTYNG,K,K1)
      IJG=INDEX(NIJ,IG,IJ)
      IF (I-1) 1435,1435,1440
 1435 IJG4=IJG
      IJG5=IJG+1
      IJG6=IJG+2
      GO TO 1475
 1440 IF (I-NR) 1445,1450,1450
 1445 IJG4=IJG-1
      IJG5=IJG
      IJG6=IJG+1
      GO TO 1475
 1450 IJG4=IJG-2
      IJG5=IJG-1
      IJG6=IJG
C
 1475 I2=I
      IF(I.EQ.NR) I2=I-1
      IF(I.NE.1.AND.I.NE.NR) GO TO 1485
      IF(KD.GE.0) GO TO 1485
      IF(I2.NE.1.OR.KLBC.EQ.0) GO TO 1479
      SIGN=1.0
      NONE=IJG4
      NTWO=IJG5
      GO TO 1480
 1479 IF(KRBC.EQ.0) GO TO 1480
      SIGN=-1.0
      NONE=IJG5
      NTWO=IJG6
 1480 DENOM=SIGN*DELR(I2)*(DXI+DELR(I2))
      A2=(XN(NTWO)-XN(NONE))/DENOM
      A2A=(XNA(NTWO)-XNA(NONE))/DENOM
      DELP=2.*SIGN*A2*DXI
      DELPA=2.*SIGN*A2A*DXI
      GO TO 1486
 1485 IF(I.EQ.1) I2=I+1
      DELP=2.*(DELR(I2)*XN(IJG4)-DEL1*XN(IJG5)+DELR(I2-1)*XN(IJG6))*DXI/
     1DEL3
      DELPA=2.*(DELR(I2)*XNA(IJG4)-DEL1*XNA(IJG5)+DELR(I2-1)*XNA(IJG6))*
     1DXI/DEL3
 1486 D1(IG)=D1(IG)-DELPA*C(ITEMP)*DELP*V(IJ)
C     BUCKLING LOSS  - FLUX GRADIENT CONTRIBUTION
      NGM=NG*NMAT
      LTEM=NTYNG*NMAT+NG*(K-1)+IG
      LLTEM=LTEM+NGM
      LLL=LTEM+2*NGM
C     LLL IS LOCATION OF FICK,S LAW CURRENT FACTOR 1/(3*(SIGTR**2))
C     LTEM AND LLTEM ARE LOCATIONS OF THE BUCKLING LOSS CROSS SECTIONS
      TL1(IG)=TL1(IG)-C(LTEM)*DELP*DELPA*C(LLL)*V(IJ)/PAP
      TL2(IG)=TL2(IG)-C(LLTEM)*DELP*DELPA*C(LLL)*V(IJ)/PAP
      IF(KAR.NE.2) GO TO 1490
      ITM1=ITEMP-2
      A(IG)=A(IG)-C(ITM1)*DELP*DELPA*C(LLL)*V(IJ)/PAP
```

```
C
 1490 CONTINUE
      D1(IG)=D1(IG)/PAP
C     SEE EQ. 32 IN WRITEUP
C
C     FOR THE SECOND DIRECTION (MUST BE EITHER ONE OR THREE OR MORE MESH
C        INTERVALS) THREE POINT VANDERMONDE SOLUTION FOR FLUX DERIVATIVE
C     THE DIFFERENCE IN GRADIENT ACROSS PARALLEL FACES OF A MESH
C     INTERVAL IS USED AS THE NET GRADIENT OF THE INTERVAL
C
      IF(NZ.LE.1) GO TO 1595
      XII=XIZ
 1500 DO 1590 J=1,NZ
      IF(J.EQ.1) DXII=2.*XII
      IF (J-1) 1505,1505,1510
 1505 J1=J+1
      GO TO 1515
 1510 J1=J-1
      DXII=2.*DELZ(J1)-DXII
      IF (J-NZ) 1515,1525,1525
 1515 DEL1=DELZ(J)+DELZ(J1)
      DEL2=DELZ(J)*DELZ(J1)
      DEL3=DEL1*DEL2
C
 1525 IJ=INDEX(NR,J,0)
      IJG=INDEX(NIJ,IG,IJ)
      DO 1590 I=1,NR
      IJ=IJ+1
      IJG=IJG+1
      K=MA(IJ)
      IF (K) 1590,1590,1530
 1530 ITEMP=INDEX(NTYNG,K,K1)
      IF (J-1) 1535,1535,1540
 1535 IJG4=IJG
      IJG5=IJG+NR
      IJG6=IJG5+NR
      GO TO 1575
 1540 IF (J-NZ) 1545,1550,1550
 1545 IJG4=IJG-NR
      IJG5=IJG
      IJG6=IJG+NR
      GO TO 1575
 1550 IJG5=IJG-NR
      IJG4=IJG5-NR
      IJG6=IJG
C
 1575 J2=J
      IF(J.EQ.NZ) J2=J-1
      IF(J.NE.1.AND.J.NE.NZ) GO TO 1585
      IF(KD.GE.0) GO TO 1585
      IF(J2.NE.1.OR.KBBC.EQ.0) GO TO 1579
      SIGN=1.0
      NONE=IJG4
      NTWO=IJG5
      GO TO 1580
 1579 IF(KTBC.EQ.0) GO TO 1580
      SIGN=-1.0
      NONE=IJG5
      NTWO=IJG6
 1580 DENOM=SIGN*DELZ(J2)*(DXII+DELZ(J2))
      A2=(XN(NTWO)-XN(NONE))/DENOM
      A2A=(XNA(NTWO)-XNA(NONE))/DENOM
      DELP=2.*SIGN*A2*DXII
      DELPA=2.*SIGN*A2A*DXII
      GO TO 1586
 1585 IF(J.EQ.1) J2=J+1
      DELP=2.*(DELZ(J2)*XN(IJG4)-DEL1*XN(IJG5)+DELZ(J2-1)*XN(IJG6))*DXII
     1/DEL3
      DELPA=2.*(DELZ(J2)*XNA(IJG4)-DEL1*XNA(IJG5)+DELZ(J2-1)*XNA(IJG6))*
     1DXII/DEL3
 1586 D2(IG)=D2(IG)-DELPA*C(ITEMP)*DELP*V(IJ)
      LTEM=NTYNG*NMAT+NG*(K-1)+IG
      LLL=LTEM+2*NGM
      TL1(IG)=TL1(IG)-C(LTEM)*DELP*DELPA*C(LLL)*V(IJ)/PAP
      IF(KAR.NE.2) GO TO 1590
      ITM1=ITEMP-2
      A(IG)=A(IG)-C(ITM1)*DELP*DELPA*C(LLL)*V(IJ)/PAP
C
 1590 CONTINUE
      D2(IG)=D2(IG)/PAP
C
C
 1595 CONTINUE
      RETURN
      END




$IBFTC PER9

      SUBROUTINE OUTPUT ( P, S, A, R, D1, D2, YJA,XKEFF,TL1,TL2)
C
C
C
C
C
C     THE COMMON STATEMENTS
      COMMON       /CALL1/                              NG,       NTYPS,
     1   NTYNG,    NFAST,    NUP,       NR,             NZ,       NIJ,
     2             KFLUX,    KBEFF,     KCTYP,          KIN,
     3             KP,       KS,        KAR,            KD,
     4 KAPROX,NMAT,NMID,NDSP,IGHI,IGLO,XIR,XIZ
      COMMON       /CALL2/   PAP
      COMMON       /CALL3/   KXTRA1,    KXTRA2,         KXTRA3,   KXTRA4,
     1   KXTRA5,   EXTRA1,   EXTRA2,    EXTRA3,         EXTRA4,   EXTRA5
C
C     THE DIMENSION STATEMENTS
      DIMENSION    P(1),          S(1),                 A(1),
     1TL1(1),TL2(1), R(1),          D1(1),                D2(1),YJA(1)
C
C     THE FORMAT STATEMENTS
  100 FORMAT(1H1)
  101 FORMAT(1HK,25H DELAYED SPECTRUM NUMBER ,I2,21H **** SPECTRAL SUM =
     1 ,F10.7/42X,7HBETA = ,F10.7/37X,12HBETA(EFF) = ,F10.7)
  113 FORMAT (7E16.7)
  114 FORMAT(1HL,38HUNPERTURBED MULTIPLICATION FACTOR ****,F12.8/
     139H  PERTURBED MULTIPLICATION FACTOR   ****,F12.8)
  115 FORMAT(1HK,4X,10HTRANSVERSE,6X,10HTRANSVERSE,56X,6HGROUP /
     15X,10HLEAKAGE  1,6X,10HLEAKAGE  2)
  116 FORMAT(2E16.7,48X,E16.7)
  119 FORMAT(35X,41H((1/KEFF PERTURBED)-(1/KEFF UNPERTURBED)))
  120 FORMAT(44X,23HDIFFUSION APPROXIMATION//5X,
     111H PRODUCTION,5X,10HSCATTERING,6X,10HABSORPTION,6X,7HREMOVAL,5X,
     216H DIFF. COEFF. 1 ,16H DIFF. COEFF. 2 )
  121 FORMAT (1HJ,14H  GROUP TOTALS)
  122 FORMAT(37X,37HTRANSPORT CORRECTED P-0 APPROXIMATION//5X,
     111H PRODUCTION,5X,10HSCATTERING,6X,10HABSORPTION,6X,7HREMOVAL,9X,
     29HTRANSPORT,18H    DIFF. COEFF. 2 )
  123 FORMAT(42X,27HP-1 TRANSPORT APPROXIMATION//5X,
     111H PRODUCTION,5X,10HSCATTERING,6X,10HABSORPTION,6X,7HREMOVAL,6X,
     214HG-TO-G***J-WGT,5X,7HLEAKAGE)
C
C
C
 1000 WRITE (6,100)
      IF(KBEFF.EQ.0) GO TO 1010
      ISPAN=IGLO-IGHI+1
      NST=NIJ+NDSP*ISPAN
      LOB=NIJ
      DO 1015 ID=1,NDSP
      NST=NST+1
      SUM=0.0
      DO 1020 IGD=1,ISPAN
      LOB=LOB+1
 1020 SUM=SUM+YJA(LOB)
 1015 WRITE(6,101) ID,SUM,S(ID),YJA(NST)
      GO TO 1100
 1010 WRITE(6,119)
      IF(KAPROX.EQ.0) WRITE(6,120)
      IF(KAPROX.EQ.1) WRITE(6,122)
      IF(KAPROX.EQ.2) WRITE(6,123)
      RHO=0.0
      RHOP=0.0
      RHOS=0.0
      RHOA=0.0
      RHOR=0.0
      RHOD1=0.0
      RHOD2=0.0
      RHOTL1=0.0
      RHOTL2=0.0
      DO 1005 IG=1,NG
      RHOA=RHOA+A(IG)
      IF(KAR.EQ.2) GO TO 1005
      RHOP=RHOP+P(IG)
      RHOS=RHOS+S(IG)
      RHOR=RHOR+R(IG)
      RHOD1=RHOD1+D1(IG)
      RHOD2=RHOD2+D2(IG)
 1005 WRITE (6,113) P(IG),S(IG),A(IG),R(IG),D1(IG),D2(IG)
      WRITE (6,121)
      WRITE (6,113) RHOP,RHOS,RHOA,RHOR,RHOD1,RHOD2
      WRITE(6,115)
      DO 1050 IG=1,NG
      RHOG=A(IG)+R(IG)+P(IG)+S(IG)+TL1(IG)+TL2(IG)
      IF(KAPROX.EQ.0) RHOG=RHOG+D1(IG)+D2(IG)
      IF(KAPROX.GE.1) RHOG=RHOG+D1(IG)
      RHOTL1=RHOTL1+TL1(IG)
      RHOTL2=RHOTL2+TL2(IG)
      RHO=RHO+RHOG
 1050 WRITE(6,116) TL1(IG),TL2(IG),RHOG
      WRITE(6,121)
      WRITE(6,116) RHOTL1,RHOTL2,RHO
      XKPERT=XKEFF/(1.0+XKEFF*RHO)
      WRITE(6,114) XKEFF,XKPERT
C
C
C
 1100 RETURN
C
      END
```

```
                         OVERLAY ORIGIN CARDS AND ASSIGNED LINK NUMBERS

$ORIGIN       001                            IS LINK    1, PARENT LINK IS    0

$ORIGIN       011                            IS LINK    2, PARENT LINK IS    1

$ORIGIN       011                            IS LINK    3, PARENT LINK IS    1

$ORIGIN       011                            IS LINK    4, PARENT LINK IS    1

$ORIGIN       011                            IS LINK    5, PARENT LINK IS    1

$ORIGIN       011                            IS LINK    6, PARENT LINK IS    1

$ORIGIN       001                            IS LINK    7, PARENT LINK IS    0

$ORIGIN       001                            IS LINK    8, PARENT LINK IS    0

$ORIGIN       001                            IS LINK    9, PARENT LINK IS    0
```

```
                                     * MEMORY MAP *

SYSTEM                                          00000 THRU  02717
FILE BLOCK ORIGIN                               02720
FILES         1.      UNIT06                    (NO BUFF POOL ATTACHED)
              2.      UNIT05                    (NO BUFF POOL ATTACHED)
PRE-EXECUTION INITIALIZATION                    02750
CALL ON OBJECT PROGRAM                          02757
OBJECT PROGRAM                                  02764 THRU  16004

LINK  DECK   ORIGIN        CONTROL SECTIONS  (/NAME/=NON 0 LENGTH, (LOC)=DELETED, *=NOT REFERENCED)

  0  PERSN   02764        ///     /(17130)    /CALL1 / 02765     /CALL2 / 03015     /CALL3 / 03016     /CINPT / 03030
                           ......  03470 *
     .LINK   03514        /.LDT  / 03514      /.LRECT/ 03526     /.LVEC / 03550
     .LXCON  03572         .LXSTR  03572       .LXSTP  03575      .LKOUT  03617 *    .LXERR  03626      .LXCAL  03531
                           .LXRTN  03631 *     IBEXIT  03631      .DBCLS  04011 *    .LXARG  04100 *    .LO     04107 *
                           .CLSE   04115 *     .LFBL   04116 *    .LJNB   04117 *    .DFOUT  04120
     .LOVRY  04124         .LOVRY (04124)      .LDT   (03514)     .LRECT (03526)     .LVEC  (03550)
     TGDM    04635         TAGDUM  04635
     .LXSL   04636         .LXSEL  04636       .LXSL1  04637      .LXTST  04642      .LXOVL  04702      .LXMOD  04744
                           .LXIND  04770       .LXDIS  04773      .LXFLG  04774      .LTCH   04775
     .FPTRP  05003         .FFPT.  05003 *     .FXEM.  05005      .FXEM   05005 *    .ETAG   05234 *    .FMCRT  05350 *
                           .CGOTO  05363 *     .FXOUT  05367 *    .FXARG  05401 *    .EXIT.  05403 *    EXIT    05403 *
                           .OVCHK  05410 *     OVFLOW  05411 *    SYSONE  05436      .NOP    05437      .BLANK  05452
     .ERAS.  05467         E.1     05467       E.2     05470      E.3     05471      F.4     05472
     .XCC.   05473         CC.1    05473       CC.2    05474      CC.3    05475      CC.4    05476
     FCNV    05477         .FCON.  05477       .DDPRE  05507 *    .FCNV.  05515      .ENDFS  05526      .CNVSW  05530
                           .FDX1   05534       .FDX2   05535      .DBC    05537      .DBC14  05602 *    .STOP4  05605 *
                           .DBC20  05621       .DBC10  05635      .DBC99  05650 *    .DDSW   05673 *    .FIXSW  05702
                           .DXPSE  05716 *     .DDZET  06023 *    .ICIO   06064 *    .STOPJ  06101 *    .FCOUT  05111 *
                           .FCARG  06117 *     .ALCOD  06121 *    .ANPT   06132      .ONPT   06147      .LNTP   06223
                           .AOUT   06262       .OOUT   06274      .LOUT   06323      .GOUT   06333      .TGOUT  05342 *
                           .FLT    06442       .FXFL1  06577      .FXD    06603      .FXFL2  06607      .FXFL3  06613
                           .INTG   06617       .TUPAC  06635      .WIDTH  06641      .FPACK  06646      .TEST   05547
                           .DDSF   06700 *     .KOUNT  06701      .LIST   06704      .DONE   06713      .OUTBF  07047
                           .CHAR   07301       .DEXP   07310 *    .TEN    07314 *    .FBDBF  07316      .DATUM  07324 *
                           .WORD   07337       .MQD    07340      .PEX    07341      .FEXP   07342      .DIG    07343
     FIOH    07356         .FIOH.  07356       .DDIO   07403 *    .DDLET  07544 *    .DCPT1  07747 *    .FXPT   10145 *
                           .FFIL.  10176       .DDFIN  10214 *    .FRTN.  10223      .DDRTN  10227 *
     FIOS.   10332         ..FIOS  10332       .FCLS   10354 *    ..FIOC  10401      ..FSFL  10422      ..FBCK  10441 *
                           ..FTCK  10443       .IOEF   10527 *    ..XEM   10535 *    ..FCHK  10552 *
     FRDD.   10655         ..FRDD  10655
     FWRD.   10702         ..FWRD  10702
     FBCD.   10730         ..FBCD  10730       ..FBCW  10732      ..FBCB  10743
     UNIT06  10773         ..UN06  10773
     UN05    10774         .UN05.  10774
     .IOE.   10775
     .IOE56  11007
     .RWDOE  11015
     .BCREA  11035         BCREAD  11035       BCREAD (11035)
     .BCRWD  11115         ..BCRD  11115       ..BCWD  11117 *    ..BRDB  11131
     FXP2    11167         .XP2.   11167
     .XEXP.  11262         .NODEF  11262       .XEXP   11266 *    .ZJDEF  11271
     //      17130

  1  PER1    11274        ///     /(17130)    /CALL1 /(02765)    /CALL2 /(03015)    /CALL3 /(03016)    /CINPT /(03030)
                          /CHANG / 11275       INPUT   13222

  2  PER2    13246        /CALL1 /(02765)      VRZ     14241
     PER21   14341         RZREAD  14500

  3  PER3    13246        /CALL1 /(02765)     /CHANG /(11275)     CREAD   15673
     PER4    15766         XCHANG  15776

  4  PER5    13246         IDACAV  13720

  5  PER61   13246         READIT  13450
     PER6    13516        /CALL1 /(02765)      NREAD   14330
     PER666  14413        /CALL1 /(02765)      REED    14637

  6  PER7    13246        /CALL1 /(02765)     /CALL2 /(03015)     PAPCAL  13624

  7  PER8    11274        /CALL1 /(02765)     /CALL2 /(03015)    /CALL3 /(03016)     PERTUR  13454

  8  PER81   11274        /CALL1 /(02765)     /CALL2 /(03015)     DIFUSE  13112

  9  PER9    11274        /CALL1 /(02765)     /CALL2 /(03015)    /CALL3 /(03016)     OUTPUT  12200

UNUSED CORE                                     16005 THRU  17127

   BEGIN EXECUTION.
```

REFERENCES

1. Hansen, Gordon E.; and Maier, Clifford: Perturbation Theory of Reactivity Coefficients for Fast-Neutron Critical Systems. Nucl. Sci. Eng., vol. 8, no. 6, Dec. 1960, pp. 532-542.
2. Sargis, D. A.: GAPER: A Transport Perturbation Theory Program. Rep. GA-8667, Gulf General Atomic, Apr. 26, 1968.
3. Barber, Clayton E.: A FORTRAN IV Two-Dimensional Discrete Angular Segmentation Transport Program. NASA TN D-3573, 1966.
4. Gross, E. E.; and Marable, J. H.: Static and Dynamic Multiplication Factors and Their Relation to the Inhour Equation. Nucl. Sci. Eng., vol. 7, no. 4, Apr. 1960, pp. 281-291.
5. Joanou, G. D.; and Dudek, J. S.: GAM-II. A B_3 Code for the Calculation of Fast-Neutron Spectra and Associated Multigroup Constants. Rep. GA-4265, General Dynamics Corp., Sept. 16, 1963.

NASA-Langley, 1970 —— 22

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Notes, and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546