Report No. IITRI-A6088-21
(Final Report, Volume I)

PREDICTION OF NEUTRON INDUCED ACTIVATION
VOLUME I - NAP CODE MANUAL

May 14, 1964 through January 31, 1966

Contract No. NAS8-11160
Control No. DCN 1-4-50-01014-01 & S1(1F)
CPB 02-1058064
IITRI Project A6088

Prepared by
David A. Klopp

IIT RESEARCH INSTITUTE
Technology Center
Chicago, Illinois 60616

for

George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Huntsville, Alabama 35812

Attention: PR-EC

Report No. IITRI-A6088-21

(Final Report, Volume I)

PREDICTION OF NEUTRON INDUCED ACTIVATION
VOLUME I - NAP CODE MANUAL

May 14, 1964 through January 31, 1966

Contract No. NAS8-11160
Control No. DCN 1-4-50-01014-01 & S1(1F)
CPB 02-1058064
IITRI Project A6088

Prepared by

David A. Klopp

IIT RESEARCH INSTITUTE
Technology Center
Chicago, Illinois 60616

for

George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Huntsville, Alabama 35812

Attention: PR-EC

This report was prepared by the IIT Research Institute under Contract No. NAS8-11160 entitled Development and Validation of a Method for Predicting Neutron Induced Activation in Materials for the George C. Marshall Space Flight Center of the National Aeronautics and Space Administration. The work was administered under the technical direction of the Propulsion and Vehicle Engineering Laboratory, Materials Division of the George C. Marshall Space Flight Center with Lowell K. Zoller acting as project manager.

FOREWORD

This is Report No. IITRI-A6088-21 of IITRI Project A6088, Contract No. NAS8-11160, entitled "Prediction of Neutron Induced Activation, Volume I - NAP Code Manual." The report covers the period from May 14, 1964, through January 31, 1966.

Personnel who made significant contributions to the research reported here include Dr. Robert B. Moler, who programmed the major portion of the computer subroutine dealing with radioisotope decay chains, and Dr. Gerald Hardie, who programmed the computer subroutine for the computation of the cross section for compound nucleus formation due to neutron bombardment.

Respectfully submitted,

IIT RESEARCH INSTITUTE

D. A. Klopp
Research Physicist
Nuclear and Radiation Physics Section

Approved by:

T. G. Stinchcomb
Acting Manager
Nuclear and Radiation Physics Section

C. W. Terrell
Director
Physics Research

/jh

ABSTRACT

PREDICTION OF NEUTRON INDUCED ACTIVATION
VOLUME I - NAP CODE MANUAL

An IBM 7094 computer program was written for the prediction of neutron induced activation. This report describes the preparation of input data and the interpretation of output data. Flow charts for the main program and each subroutine are given, as is a complete program listing in Fortran IV. A sample problem is also presented.

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

I. INTRODUCTION

A computer program for the prediction of neutron induced gamma ray radioactivity was required for use in the design of facilities for developing a nuclear space vehicle. Neutron induced activation calculations are required not only in the selection of structural materials for such a facility, but also for the scheduling of the static tests and the maintenance and repair of equipment, once the facility is constructed. A neutron induced activation computer program could also be utilized to determine the gamma ray activity of reactor coolants, or any other material exposed to a neutron flux. The analysis of foil activation data for the determination of neutron energy spectra, and other activation analyses, could be assisted by a neutron activation computer program.

The objectives of the research program reported here were:

1. The development of a flexible and comprehensive analytical method for computing neutron induced activation, providing for items such as:
   a. Inclusion of at least (n,γ), (n,p), (n,α), and $(n,2n)$ reactions which result in radionuclei for all stable and long-lived metastable isotopes.
   b. Consideration of at least first-generation daughter radioisotopes.
   c. Variation of the neutron energy spectrum, not necessarily limited to fission spectra.
   d. Consideration of non-uniform cyclic irradiation.
   e. Consideration of self-shielding.
   f. Determination of radiation dose rates from activated materials for a variety of geometric conditions, at least by use of one or more of the commonly used radiation shielding or reactor computer codes.

g. Adaption for primary, charged particle induced reactions.

2. The programming of the analytical method for use on a digital computer.
3. The validation of the analytical method through an experimental program.

Computer programs developed previously throughout industry for calculating neutron induced activation were generally limited to a relatively few neutron reactions by lack of data, or were restricted in utility by failure to incorporate one or more of the critical items listed above.

This volume of the final report constitutes a manual for the use of the NAP (Neutron Activation Prediction) computer program. The NAP program is a FORTRAN-IV IBM-7094 computer program which computes neutron induced activation gamma ray source strengths as a function of time, space, and energy given an incident neutron flux and region material compositions. Simplified dose and dose rate calculations, which do not account for gamma ray attenuation or buildup, are also performed by the program. More accurate dose rate calculations may be performed by utilizing results obtained from the NAP code as input data for any of the more popular gamma ray shielding computer codes. All the critical items listed in the previous paragraph are treated adequately by the NAP program. This code manual consists of a general description of the NAP code, instructions for preparing input and interpreting output, operating instructions, flow charts, and a complete program listing.

The experimental validation of the NAP computer program is discussed in Volume II of this final report series. Volume II also describes in detail those portions of the NAP code which required extensive physical analysis. The description in Volume II emphasizes the theoretical aspects and the experimental validation of the program, rather than the

programming aspects which are emphasized in this volume. A description and listing of the NAP Cross Section Library is given in Volume III of this series. A similar description and listing of the NAP Gamma Radiation Library is given in Volume IV.

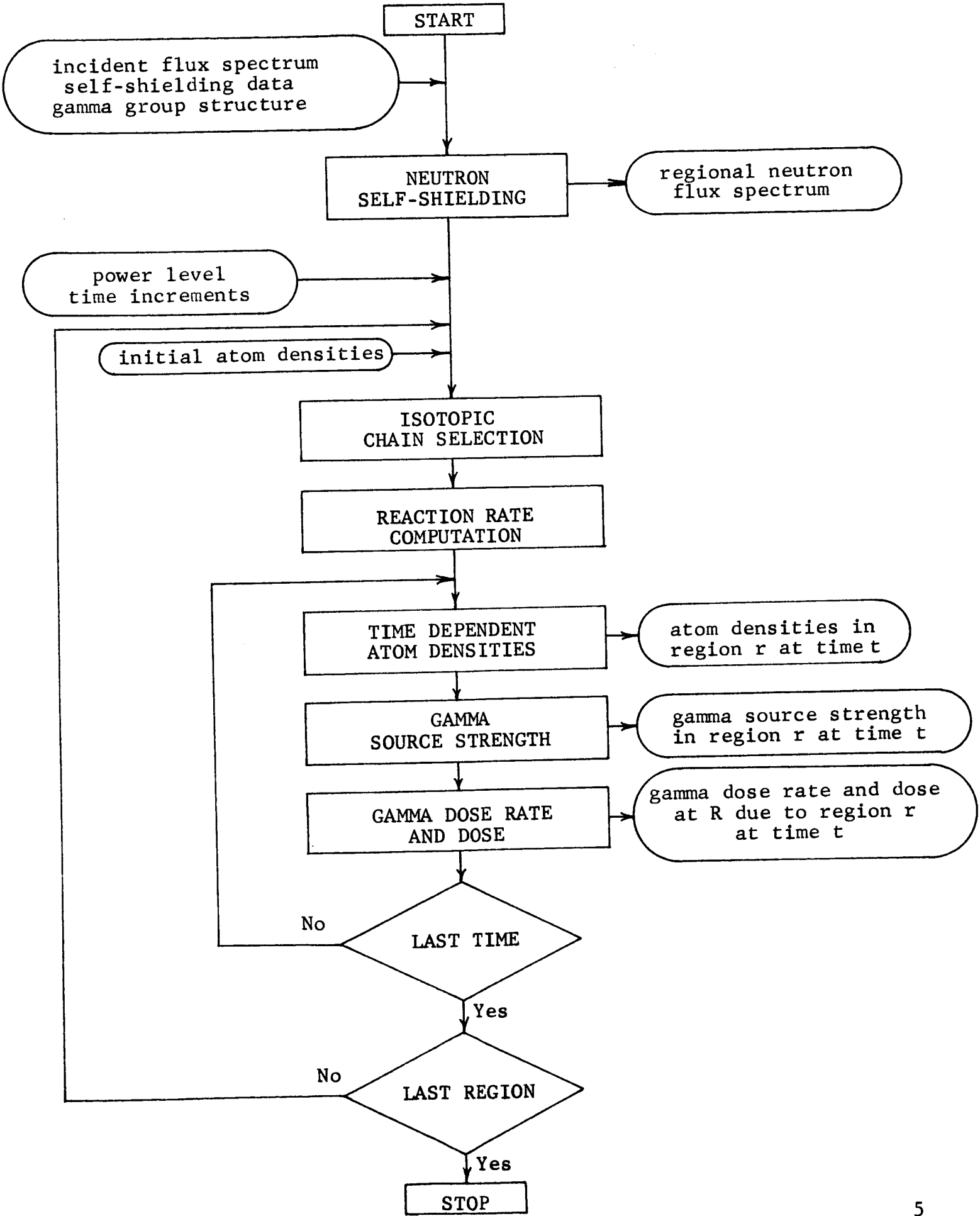
II. GENERAL DESCRIPTION OF THE NAP CODE

The NAP (Neutron Activation Prediction) computer program is a comprehensive and flexible tool for the computation of neutron induced activation gamma ray source strengths. A simplified method, ignoring gamma ray attenuation and buildup, is provided for the calculation of gamma ray dose and dose rate due to these activation sources. The source strengths may also be utilized as input data for any of the more popular gamma ray shielding codes to provide more sophisticated shielding calculations. The NAP code may be used to provide activation calculations for structural materials, reactor coolants, or any material exposed to a neutron flux. The NAP code may also be used as an analytical tool in the interpretation of activation data, such as that obtained in attempting to measure neutron flux spectra or isotopic composition.

The basic logical flow during execution of a typical NAP problem is shown in Figure 1. Only the most important phases of a NAP calculation are depicted. Input data are indicated in abreviated form by the oval boxes on the left hand side of the figure, output data on the right hand side.

The typical NAP program operation commences by reading in the neutron energy flux spectrum in terms of the number of neutrons per square centimeter per second incident upon the activated material, or some other specified spatial region, having energies within various neutron energy bands or groups, whose selection is specified by the program user. Angular incident neutron flux information may also be presented. The energy bands or groups into which the discrete gamma ray energies are to be sorted are also specified by the program user. If the NAP program is to calculate neutron flux distributions within the activated material regions, or other regions, various neutron cross sections must be provided by energy groups and regions, along with geometrical data. If requested by the problem originator, the NAP program then

BLOCK DIAGRAM OF NAP PROGRAM



computes the neutron flux distribution as a function of neutron energy group and spatial position within the various regions specified by the program user. This neutron self-shielding calculation is performed in one-dimensional slab or spherical geometry. The basic result of this computation is the spatially-averaged neutron flux in each neutron energy group and each spatial region.

The time-dependence of the incident neutron flux is accounted for by reading in the power levels for a series of time increments. The energy group and region dependent flux values are multiplied by the power level to obtain fluxes which rise or fall at specific times. The different time values at which gamma source strengths are desired must also be supplied to the NAP program, in addition to those time values which form integration end points for the computation of gamma ray dose, i.e., the gamma ray dose rate integrated over a given time interval.

The NAP calculation now proceeds region by region. A region is defined as that spatial volume throughout which the neutron flux is treated as spatially constant and which has a uniform, homogeneous material composition at zero time. Because the neutron flux is regarded as spatially constant in each region, the gamma source strength is also treated as spatially constant in each region. Starting with the first region, the isotopic (or elemental) atom densities initially present are read into the program. Using the first isotope whose initial atom density has been supplied as a target nucleus, the NAP program discovers the identity of the isotope resulting from (n, γ) reactions. The rate at which the residual nucleus is produced is computed, and the NAP Gamma Radiation Library is searched to discover the identity of the daughter isotope resulting from radioactive decay of the reaction product nucleus. If the daughter decays, the gamma library is searched for the identity of the second-generation

daughter, etc. This process continues through four generations of decay, or until a stable daughter isotope is found.

Having established a specific decay chain, the NAP program computes and prints out the atom density of each chain member at each of the specific times supplied previously to the program. This is followed by a computation of the emission rate of gamma rays, grouped according to the previously selected gamma ray energy bands, from each of the chain members at each time value. These gamma source strengths are also printed out by the program. At each time value, the gamma ray emission rate is converted to a gamma ray dose rate, using the average gamma ray energies of the selected energy bands and the source-detector distance. These dose rates are printed out, and are integrated over time to provide gamma ray dose values. Finally, these dose values are printed out.

The NAP program now returns to the reaction product nucleus to discover if another decay chain exists. If so, the process described briefly above is repeated. Having exhausted the possibilities of decay following the (n, γ) reaction, the NAP program returns to the isotope originally present in the region. Now the (n, p) reaction is followed through in the manner described above. The process is repeated for the (n, α) and then the (n, 2n) reaction. After all these reaction possibilities and the subsequent decay chains have been investigated, the NAP program considers the second isotope originally present in the region. Using the second isotope as a base, the entire calculation described above is repeated. This process is continued until all the isotopes originally present have been used as a base for the calculation. At this point, the NAP program proceeds to the second region, and the computation begins anew.

This brief description of a typical NAP problem shows that the problem may be conceptually divided into four main sub-calculations. These are calculations of:

1. Neutron flux as a function of space, time, and energy,
2. Isotopic atom densities as functions of space and time,
3. Gamma ray source strengths as functions of space, time, and energy,
4. Gamma ray dose and dose rate as functions of time for a single detector position.

The following sections A through D discusses each of the calculations in greater detail.

A. Calculation of Neutron Flux

The major protion of the neutron flux calculation is optional and consists of a multigroup discrete ordinate transport calculation. It should be performed if significant spatial variations of the flux are anticipated in the regions for which the gamma ray source strengths are desired, or in the regions between the position where the flux is known and the positions where the source strengths will be calculated.

In any case, an incident neutron flux must be specified. The neutron energy range of interest is divided into contiguous neutron energy groups such that the upper and lower energy limits of energy group g are E_{g-1} and E_g, respectively. If x_o is the position where the incident flux is given, then the flux as a function of time and energy is taken as

$$\emptyset(t,E,x_o) = P(t)F\emptyset_g(x_o)$$

where P(t) is a time-dependent reactor power level, F is an arbitrary normalization factor, and $\emptyset_g$ is the flux in energy group g. The incident flux is thus completely specified by giving the power level as a function of time, the flux normalization factor, the incident flux in each energy group, and the group energy limits.

The time dependence of the flux is contained in the "power level" P(t), which is simply a dimensionless flux intensity normalization factor. The entire time span of interest is divided into as many as 50 time periods, each of arbitrary length. The factor P(t) must be given for each time period, and is assumed constant throughout the duration of each time period. An arbitrary time-dependent flux is thus approximated by a series of power levels, each of arbitrary time duration but having a constant magnitude throughout the duration of each time period. For example, a known time dependence of the incident flux might be approximated by

P(t) as shown in Figure 2. Non-cyclic irradiations are handled easily in this framework.

The time-independent flux spectrum is specified by giving the group fluxes ϕ_g and the energy limits E_g. A maximum of 43 energy groups is permitted. The group fluxes may be input in any one of three forms, where:

1. ϕ_g is the average flux per unit lethargy in energy group g,
2. ϕ_g is the average flux per unit energy in energy group g,
3. ϕ_g is the integral of $\phi(E)$ over energy from E_g to E_{g-1}.

The input group energy limits are adjusted by the NAP program, if they are not consistent with the group energy limits used in the NAP Cross Section Library, which are listed in Table I. Similarly, input group fluxes ϕ_g are adjusted to be consistent with the library energy limits. A simple check calculation is automatically performed to insure that the total flux is invariant to this energy limit and group flux adjustment. Thus, in many cases, the group energy limits and group fluxes appearing on the output will be equivalent but not identical to those supplied as input by the problem originator.

If desired, this group adjustment procedure can be bypassed by appropriate selection of an input parameter. The program user must then insure that the input group fluxes are consistent either with the NAP Cross Section Library or with cross sections supplied by the user. In any case, the quantity $P(t)F\ \phi_g\ \sigma_g$ integrated over energy, where σ_g is a group g cross section supplied by the NAP library or by the user, must have dimensions of neutron-barn/cm^2-sec.

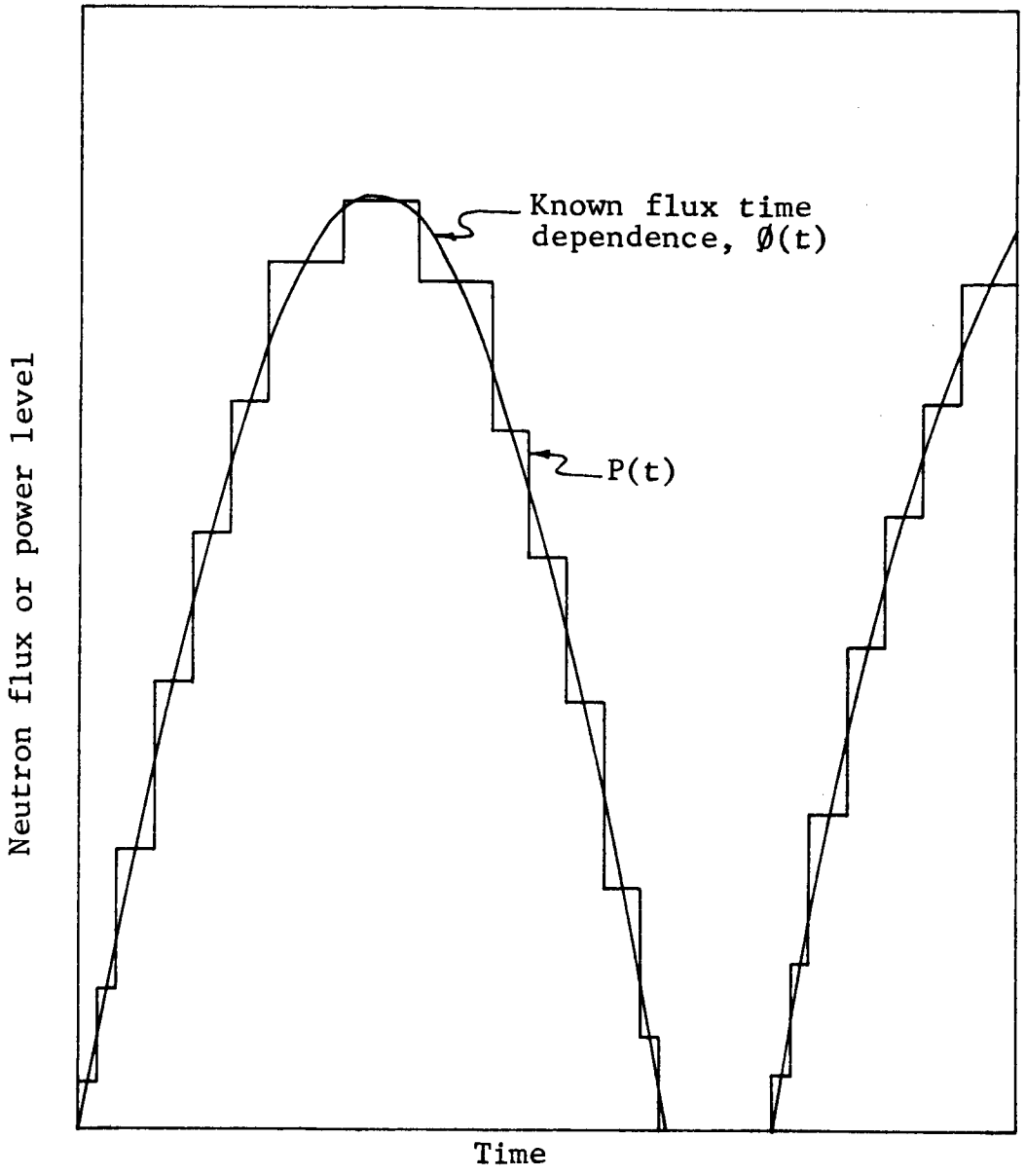


Figure 2

APPROXIMATION OF NEUTRON FLUX TIME DEPENDENCE

Table 1

NAP LIBRARY ENERGY GROUP STRUCTURE

Group Number	Lower Energy Limit	Lower Lethargy Limit
0	21.17 MeV	-0.75
1	16.49 MeV	-0.50
2	12.84 MeV	-0.25
3	10.00 MeV	0.00
4	7.788 MeV	0.25
5	6.065 MeV	0.50
6	4.724 MeV	0.75
7	3.679 MeV	1.00
8	2.865 MeV	1.25
9	2.231 MeV	1.50
10	1.738 MeV	1.75
11	1.353 MeV	2.00
12	1.054 MeV	2.25
13	820.8 keV	2.50
14	497.9 keV	3.00
15	302.0 keV	3.50
16	183.2 keV	4.00
17	111.1 keV	4.50
18	67.38 keV	5.00
19	40.87 keV	5.50
20	24.79 keV	6.00
21	15.03 keV	6.50
22	9.119 keV	7.00
23	5.531 keV	7.50
24	3.355 keV	8.00
25	2.035 keV	8.50
26	1.234 keV	9.00
27	748.5 eV	9.50
28	454.0 eV	10.00
29	275.4 eV	10.50
30	167.0 eV	11.00
31	101.3 eV	11.50
32	61.44 eV	12.00
33	37.24 eV	12.50
34	22.60 eV	13.00
35	13.71 eV	13.50
36	8.315 eV	14.00
37	5.043 eV	14.50
38	3.059 eV	15.00
39	1.855 eV	15.50
40	1.125 eV	16.00
41	0.6826 eV	16.50
42	0.4140 eV	17.00
43	0.0010 eV	23.03

Spatial variation of the neutron flux is provided by the problem originator or by the neutron transport subroutines of the NAP program. After specifying the incident neutron flux at the position x_o, or in the first spatial region, the program user may

1. stipulate that the group fluxes are spatially-independent,
2. supply spatially-dependent group fluxes as input, or
3. utilize the NAP neutron transport subroutines to provide spatially-dependent group fluxes.

The NAP neutron transport calculation is based on the discrete ordinate method in one-dimensional slab or spherical geometry. In the absence of internal neutron sources, the monoenergetic steady-state Boltzmann transport equation may be written[1]

$$\mu \frac{\partial \psi(x,\mu)}{\partial x} + \Sigma_t \psi(x,\mu) = \frac{1}{2} \Sigma_s \int_{-1}^{1} \psi(x,\mu')d\mu'$$

where plane geometry and isotropic scattering have been assumed. Here $\psi(x,\mu)$ is the monoenergetic neutron flux at x traveling in the direction $\cos^{-1}\mu$ with respect to the positive x-axis, and Σ_t and Σ_s are the total and scattering cross sections, respectively, both being step functions of position. In the discrete ordinate method, the integral in the equation above is approximated by a numerical integration formula of the type:

$$\int_{-1}^{1} \psi(x,\mu')d\mu' = \sum_{j=1}^{N} a_j \psi(x,\mu_j)$$

[1]Davison, B.: Neutron Transport Theory. Oxford University Press, 1957.

where the discrete ordinates μ_j and the weights a_j are given by the integration formula used, and are independent of the integrand. The transport equation need then be solved only for μ equal to each of the μ_j. That is, the transport equation is replaced by the system of N differential equations

$$\mu_j \frac{\partial \psi(x,\mu_j)}{\partial x} + \Sigma_t \psi(x,\mu_j) = \frac{1}{2} \Sigma_s \sum_{k=1}^{N} a_k \psi(x,\mu_k)$$

The technique is easily extended in the multigroup formalism. The equations solved by the NAP neutron transport subroutine are, in slab geometry,

$$(\mu_j \frac{d}{dx} + \Sigma_g^t)\phi_g(x,\mu_j) = \frac{1}{2} \Sigma_{g,g} \sum_{k=1}^{N} a_k \phi_g(x,\mu_k)$$

$$+ \frac{1}{2} \Sigma_{g-1,g} \sum_{k=1}^{N} a_k \phi_{g-1}(x,\mu_k)$$

where $\phi_g(x,\mu_j)$ is the neutron flux at x traveling in the direction $\cos^{-1}\mu_j$ with energy in energy group g, Σ_g^t is the total cross section for energy group g, and $\Sigma_{h,g}$ is the scattering cross section for transfer from energy group h into energy group g. Similar, but more complicated equations, are solved for problems in spherical geometry. One of the restrictions of the current transport subroutine is that neutron slowing-down is assumed to be from one energy group only to the adjacent energy group of lower energy. These equations are solved by an iterative procedure.

The boundary conditions used in the NAP transport subroutine are that the incident flux is specified at the left-most, or first, value of x and that there is no return current (vacuum boundary condition) at the right-most, or largest value of x. That is,

$$\phi_g(x_o,\mu_j) = \text{given for all g and } \mu_j > 0$$
$$\phi_g(x_L,\mu_j) = 0 \text{ for all } \mu_j < 0$$

As many as 20 slab or spherical shell regions, of different materials, may be used in the NAP neutron transport subroutine. A maximum of ten ordinates may be used to describe the angular dependence of the flux, and 43 neutron energy groups may be used. Either Legendre-Gauss or Lobatto quadrature is available. The appropriate ordinates, i.e. values of cos θ where θ is the angle between the neutron velocity and the normal to the surface, are given in Table II. The program user must specify the forward components of each group flux at the left-most surface. A maximum of 100 spatial mesh points may be used in the problem. Figure 3 illustrates a one-group, two-region problem using eleven mesh points. Macroscopic total, scattering, and group transfer cross sections must be supplied for each energy group and each region. Isotropic scattering is assumed and group transfer is permitted from group g to g+1 only, i.e. only down-scatter is permitted to the next lower energy group. The convergence criterion ϵ is supplied by the problem originator. Convergence is assumed when, for each energy group, the fractional change in the flux at the right-most spatial point is less than ϵ between iterations. The flux in each group is then spatially averaged in each region for later use in computing activation. Further details of the physics aspects of the NAP neutron transport subroutines are given in Volume II of this report series.

In summary, the incident neutron flux must be specified by the program user. The time-dependence is contained in the power level factors P(t), the energy-dependence is contained in the group fluxes ϕ_g, and the spatial-dependence is either specified as input data or computed by the NAP program in one-dimensional slab or spherical geometry.

Table II

VALUES OF DISCRETE ORDINATES

Number of Ordinates	Value of Ordinate ($\cos \theta$) Legendre-Gauss	Lobatto
2	$\pm$ 0.5773503	——
4	$\pm$ 0.3399810	$\pm$ 0.447214
	$\pm$ 0.8611363	$\pm$ 1.0
6	$\pm$ 0.2386192	$\pm$ 0.285232
	$\pm$ 0.6612094	$\pm$ 0.765055
	$\pm$ 0.9324695	$\pm$ 1.0
8	$\pm$ 0.1834346	——
	$\pm$ 0.5255324	
	$\pm$ 0.7966665	
	$\pm$ 0.9602899	
10	$\pm$ 0.1488743	——
	$\pm$ 0.4333954	
	$\pm$ 0.6794097	
	$\pm$ 0.8650634	
	$\pm$ 0.9739065	

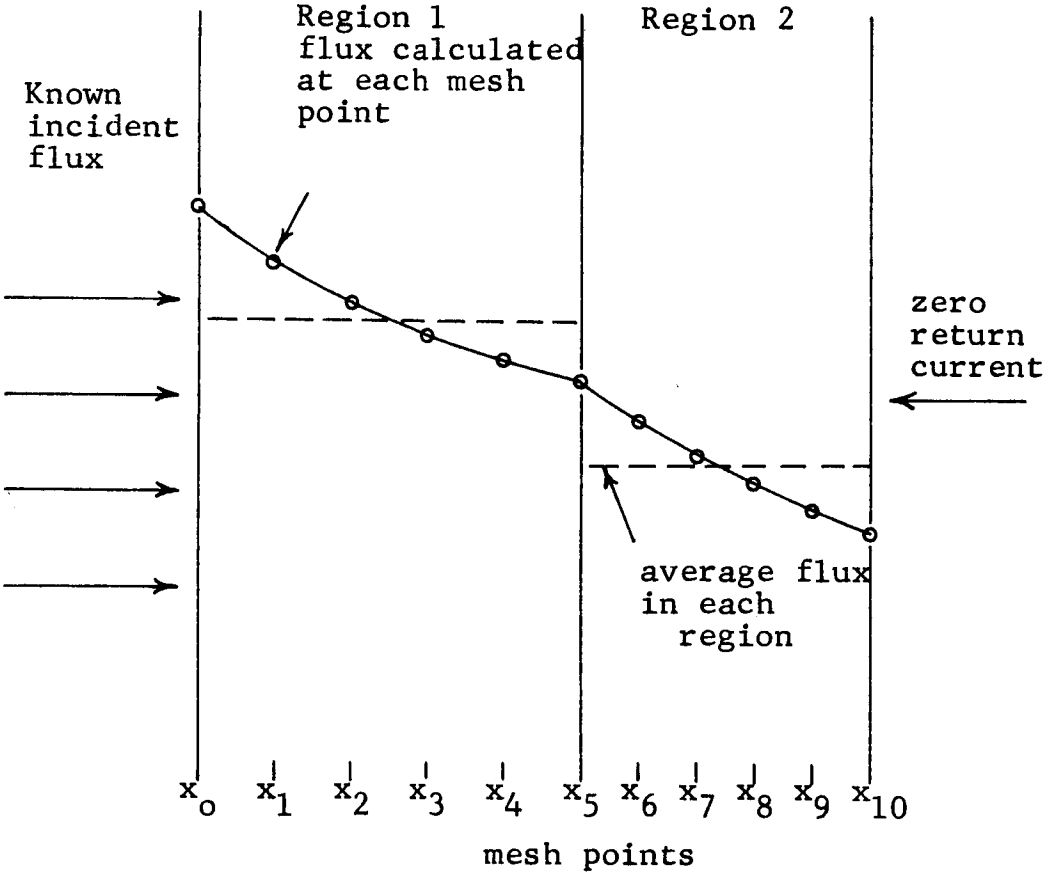


Figure 3

SPATIAL DEPENDENCE OF NEUTRON FLUX

B. Calculation of Radioisotopic Atom Densities

The neutron produced radioisotopic atom densities are computed as a function of time in each region, region by region. A maximum of 20 regions may be used. The initial isotopic or elemental atom densities in each region must be provided as input data. If elemental densities are specified, the isotopic composition is assumed to be that of the naturally occuring element. A maximum of 20 initial isotopes may be utilized in each region. The geometry of each region is arbitrary, but the neutron flux is taken as spatially constant in each region.

Each time period for which the power level is constant is divided into a set of equal time intervals. The total number of time intervals summed over the time periods is limited to 199. That is, the isotopic atom densities in each region may be computed at 200 different times, including the initial time. The time periods and time intervals are taken identically in each spatial region.

In each region, each initially present isotope is examined for possible (n,γ), (n,p), (n,α), and $(n,2n)$ reactions ending with a ground state or an isomeric state. Only one ground state and one isomeric state is permitted for each isotope. Each isotope initially present leads to the possible production of eight new "isotopes." Four of these result from (n,x) reactions, where x is γ, p, α, or 2n, leading to a ground state; the other four result from (n,x) reactions leading to an isomeric state. The labeling of the (n,x) reaction cross section contained in the NAP Cross Section Library, or in the cross section set supplied by the program user, indicates whether the cross section of the (n,x) reaction is to be associated with a ground state product or an isomeric state product. If no tabulated cross section for the (n,x) reaction is found by the NAP program, the code will either

set the cross section equal to zero or will calculate the cross section as indicated in the input options by the problem originator. This cross section calculation will be described briefly below, and applies only in the case of reactions leading to ground state products. If no cross section is found for reactions leading to an isomeric state, the cross section is automatically taken as zero. In addition, the cross sections for (n,γ) reactions are corrected to account for resonance self-shielding. This correction is also discussed below.

The members of each radioisotope decay chain are determined by searching the NAP Gamma Radiation Library for the appropriate isotopes and their mode of decay. If an isotope is not found listed in the library, it is regarded as stable and the chain is terminated. It may be noted that if the decay of a radioactive isotope can in no way lead to photon emission, the isotope may be regarded as stable without error in computing gamma ray source strengths. The NAP Gamma Radiation Library was constructed to be as complete as possible and contains decay data for over 800 isotopes. A complete listing of the library is given in Volume IV of this report series. The maximum length of each chain is five members, including the isotope originally present. The formulation of the chain is discussed further in Volume II.

Having formulated a decay chain, the NAP program computes the atom density of each chain member at each time step, i.e., at the end of each time interval. For example, at the time $t + \Delta t$, where the atom densities are known at the time t and Δt is the duration of the next time interval,

$$n_1(t + \Delta t) = n_1(t)\exp(Q_1\Delta t)$$

$$n_2(t + \Delta t) = n_2(t)\exp(-Q_2\Delta t) + \frac{n_1(t)S_1}{Q_2-Q_1} \cdot [\exp(-Q_1\Delta t)-\exp(-Q_2\Delta t)]$$

where n_i is the atom density of chain member i, Q_i is the loss

rate, and S_j is the production rate due to chain member j. The Q and S pertaining to the isotope initially present are

$$Q_1 = P(\Delta t)F \int \phi_g \sigma_g^t \, dE$$

$$S_1 = P(\Delta t)F \int \phi_g \sigma_g^x \, dE$$

where $P(\Delta t)$ is the power level during the time interval Δt, F is the flux normalization factor, ϕ_g is the neutron flux for energy group g and the appropriate spatial region, σ_g^x is the cross section for neutron energy group g and reaction (n,x), and σ_g^t is the sum of σ_g^x over all x, x being γ, p,α, or 2n. The loss and production rates for the remaining members of the chain involve only decay rates and branching ratios. The expressions for the atom densities of the other chain members are similar to those quoted above. Further discussion is provided in Volume II of this report series.

The reaction rate per nucleus, i.e., the quantity $\phi_g \sigma_g$ integrated over energy, is computed by one of the NAP subroutines. The total reaction rate is simply the sum of the partial rates. Neutron reactions for other than the first chain member are ignored. The microscopic group cross sections are tabulated in the NAP Cross Section Library, or they may be supplied by the problem originator. The library cross sections are tabulated for 43 neutron energy groups using the energy structure given in Table I. A complete library listing is given in Volume III of this report series.

Because the program user will frequently desire to use less than 43 neutron energy groups, the program will collapse the 43 group cross section set to produce a cross section set desired by the user. The 43 group cross sections are collapsed by assuming one of the following flux spectra:

1. constant flux per unit energy
2. constant flux per unit lethargy
3. fission spectrum above 183 keV, constant flux per unit lethargy below 183 keV.

That is, if the problem originator specifies less than 43 neutron energy groups and also selects use of the NAP Cross Section Library, narrow group library cross sections are weighted with one of the above spectra to produce broad group cross sections consistent with the problem originator's given group structure.

The library cross sections for the 43rd group are multiplied by a thermal averaging parameter and a non-1/v factor, if given for the isotope of interest. A maximum of ten isotopic non-1/v factors may be specified in a NAP problem. The group 43 library cross section is the 2200 m/sec value of the cross section. The thermal averaging parameter is that factor such that the 2200 m/sec value of the cross section multiplied by the thermal averaging parameter would yield the value of the thermal cross section averaged over the thermal neutron flux spectrum thought to be appropriate to the problem, if the cross section had a 1/v neutron energy dependence. The thermal averaging parameter is regarded as spatially independent, and is applied to all isotopic cross sections.

Because of the possibility of appreciable spatial and energy self-shielding in large (n,γ) resonances, effective (n,γ) cross sections are computed automatically if resonance parameters for the isotope of interest are found in the cross section library, or in the cross section set supplied by the program user. Resonance self-shielding is accounted for by computation of effective resonance integrals for as many as nine resolved resonances for each isotope. Effective resonance integrals are computed using either the NR or NRIA approximation[2] and the rational approximation for the escape probability. Further details of the physical model utilized by the NAP program in the computation of effective resonance integrals are given in Volume II of this report series.

---

[2]Dresner, L.: Resonance Absorption in Nuclear Reactors. Pergamon Press, 1960.

In an effort to make the NAP program as comprehensive as feasible, cross section calculations are provided for use when any of the required cross sections are poorly known. By appropriate choice of input control parameters, the NAP program will compute any (n,γ),(n,p),(n,α), or $(n,2n)$ reaction cross sections for ground state product nuclei, if the required cross section is not found by the program in either the library or in the cross section set supplied by the program user. If desired, computed cross sections can be used in place of the library cross sections. The thermal (n,γ) cross sections are crudely estimated from known systematics in measured thermal (n,γ) cross sections and least squares fits to the measured data. Epithermal (n,γ) cross sections are estimated using statistical resonance theory and least squares fits to measured resonance statistical parameters. Only s-wave resonances are considered. The (n,p),(n,α), and $(n,2n)$ reaction cross sections are estimated by computing compound nucleus formation cross sections due to neutron bombardment and using the statistical model to compute compound nucleus decay probabilities. Direct nuclear reactions are ignored. Further details of the physical models used in cross section computation are reported in Volume II. Finally, it may be emphasized that no additional input data is required by the NAP program for the computation of cross sections. All necessary data, such as reaction Q-values, are generated automatically.

In summary, isotopic atom densities are computed region by region at up to 200 different specified times. Only isotopes initially present are assumed to undergo nuclear reactions. The transmutation chains resulting from each isotope initially present are formulated by examining the product of flux and cross section to find possible nuclear reaction products. The decay chain resulting from each reaction product is followed until a stable isotope is encountered or a maximum chain length of five members is

attained. Required reaction cross sections may be obtained from the NAP library, computed internally, or supplied by the program user. Resonance self-shielding of the (n,γ) reactions is accounted for if resonance parameters are available to the program.

C. Calculation of Gamma Ray Source Strengths

After each time interval, the gamma ray source strength density due to each chain member is computed. The gamma ray spectrum is obtained by dividing the entire gamma ray energy range of interest into contiguous energy groups. The gamma ray energy group limits must be specified by the program user. A maximum of 20 gamma ray energy groups is permitted. The isotopic source strength density is given by

$$a_{i,j}(t) = n_j(t)\lambda_j f_{i,j}$$

where $a_{i,j}(t)$ is the number of photons in gamma ray energy group i emitted by chain member j at the time t per unit volume per unit time, $n_j(t)$ is the atom density of chain member j at the time t, λ_j is the decay constant of chain member j, and $f_{i,j}$ is the number of photons emitted in gamma energy group i by chain member j per disintegration.

The $f_{i,j}$ above are obtained by using the gamma ray energy group structure supplied by the problem originator and data contained in the NAP Gamma Radiation Library. The library contains, for each isotope, the half-life, a listing of gamma ray energies, and a fractional emission probability for each listed photon energy. A complete description and listing of the NAP Gamma Radiation Library is given in Volume IV of this report series.

Regional source strength densities are obtained by summing the $a_{i,j}(t)$ in a given region over the index j. The region source strength densities are printed out for each region, each time interval, and each gamma ray energy group. In a similar manner, the regional energy source strength densities are printed out. The energy source strengths are simply the gamma ray source strengths multiplied by the average energy of each gamma ray energy group.

D. Calculation of Gamma Ray Dose Rate and Dose

For each source region, the distance R from the center of the region to a single detector position must be given. After each time interval, the gamma ray dose rate due to each source region is computed according to

$$DR(t) = \frac{1}{4\pi R^2} \sum_{i,j} K_i a_{i,j}(t)$$

where K_i is a conversion factor from photons/cm^2sec to tissue dose rate in rads/hr for gamma ray energy group i.

The gamma ray dose due to each source region is obtained by trapezoidal integration of the dose rate over time. The initial and final times for each dose calculation must be supplied as input data.

III. INPUT DATA PREPARATION

This section defines the input variables and specifies the input format for each input card necessary in using the NAP program. Except for the first input card, all input data to the NAP program are integer numbers or floating point numbers. The first card, which contains the problem title, may contain any of the alphabetic and numerical characters A to Z and 0 to 9.

An integer number is written without a decimal point, using the decimal digits 0, 1,...,9. A floating point number is written with a decimal point, using the decimal digits 0, 1,...9. Any unsigned number (without a preceeding + or - sign) is assumed to be positive. A floating point number may include an integer exponent preceeded by an E. Thus the floating point number 5.0E+03 means 5.0×10^3. An unsigned exponent is assumed to be positive.

The manner in which the value of an input variable is to be entered on an input card is specified below in the form Iw or Ew.d. Here I indicates that the value should be entered as an integer number, while E indicates that the value should be entered as a floating point number. The value of w is the number of columns on the input card which may be used to specify the value of the input variable, while the value of d is the number of columns to the right of the decimal (excluding an integer exponent). For example, the specification I6 indicates that six columns on the input card are reserved for entering an integer number. The integer number should be punched in this six-column field right-justified (i.e., the units position is at the extreme right). Thus the largest number which can be entered using an I6 format is 999999. Similarly, the specification E12.5 indicates that 12 columns on the input card are reserved for entering a floating point number. The integer exponent, if any, must be entered as right-justified.

Succeeding input format specifications as used here are separated by commas, and each specification is repeated as many times as shown by the integer preceeding the specification. If no integer preceeds the specification, it is used only once. Thus the specifications 2E12.5, 3I2 as used with the input variables EPS, XO, IGEON, IOUT, and NOANG mean that the value of EPS is punched as a floating point number on the input card in columns 1-12, the value of XO is punched as a floating point number in columns 13-24, and the values of IGEON, IOUT, and NOANG are punched as integer numbers in columns 25-26, 27-28, and 29-30, respectively.

Blank columns on any input card (except the first) are interpreted as a zero by the program. Floating point numbers need not have 4 columns devoted to the exponent field. The start of the exponent field must be marked by an E, or, if that is omitted, by a + or -. Thus E2, E+2, +02, E02, and E+02 are all permissible exponent fields. The decimal point in a floating point number, as punched on the input card, overrides the position indicated in the input format specification.

Card Type 1, format 12A6; problem title. This card is simply a title card and may contain any 72 alphanumeric characters. The title will appear at the top of each page of output data.

Card Type 2, format E12.5, 7I6; FLUXN, NOBG, NOREG, NOGG, NOSS, NISO, IFLX, IWT.

FLUXN is a neutron flux normalization factor (p. 9). It is used to avoid repeated entries of powers of ten in describing the neutron flux on card type 6 or 11. All fluxes input to the NAP code are multiplied by this factor. The dimensional units are not fixed, but must be consistent with the dimensions used on card type 6 or 11.

NOBG is the number of neutron energy groups (p. 9) used in the specification of the neutron flux on card type 6 or 11. NOBG must be less than 44.

NOREG is the total number of spatial regions (p. 15) in the problem. If the NAP neutron transport subroutine is used, neutron transport cross sections and geometrical data must be supplied for each of the NOREG regions. Similarly, if region-dependent neutron fluxes are input to the program (card type 11), the code expects the group fluxes to be specified for each of the NOREG regions. Finally, the program expects initial isotopic atom densities (card type 22) to be specified in each of the NOREG regions. In any case, NOREG must be less than 21.

NOGG is the number of gamma ray energy groups (p. 24) to be used in the description of the gamma ray energy spectrum. NOGG must be less than 21.

NOSS is an integer number used as a control option. If NOSS is entered as zero, the NAP program assumes that the neutron flux spectrum specified by card type 6 is region-

independent, i.e., the fluxes input on card type 6 will be used in each region. If NOSS is entered as a positive integer (> 0), the NAP neutron transport subroutine (p. 13) will be used to compute region-dependent fluxes. Finally, if NOSS is entered as a negative integer (< 0), the NAP program expects region-dependent fluxes to be supplied as input data using card type 11.

NISO is meaningful only if the neutron transport subroutine is used (NOSS > 0). If NISO is entered as zero, the neutron transport subroutine assumes that the incident neutron flux (p. 14) specified on card type 6 is isotropic. If NISO is entered as any positive integer (> 0), the neutron transport subroutine expects that the angular dependence of the incident flux is specified by data on card type 9.

IFLX is a control integer indicating the interpretation (p. 10) to be given to the incident neutron group fluxes specified by card type 6 or 11. IFLX = 0 signifies that the group fluxes input on card type 6 or 11 are integrals of $\emptyset(E)$ over energy using the group energy limits specified by card type 3. IFLX = 1 signifies that the input group fluxes are average fluxes per unit lethargy. IFLX = 2 signifies that the input group fluxes are average fluxes per unit energy. IFLX = 5 signifies that no interpretation is required for the input group fluxes, and further that the neutron energy group limits (card type 3) should <u>not</u> be adjusted to be consistent with the NAP Cross Section Library energy limits. This last option is used only when the problem originator desires to rely exclusively on his own cross section set (card types 19 and 20). The use of IFLX is summarized in Table III.

IWT is a control integer indicating the type of neutron flux spectrum weighting (p. 20) given to the reaction cross sections if NOBG is less than 43. If IWT = 0, a fission flux spectrum is used to weight the cross sections above 183 keV and a 1/E flux spectrum below 183 keV. If IWT = 1, a 1/E flux spectrum is used; if IWT = 2, a constant flux per unit energy spectrum is used. The use of IWT is summarized in Table IV.

Table III

EFFECT OF IFLX OPTION

Value of IFLX	Meaning of Group Fluxes (card type 6)
0	$\int_{E_g}^{E_{g-1}} \phi(E)\, dE$
1	$\int_{E_g}^{E_{g-1}} \phi(E)\, \frac{dE}{E} \Big/ \int_{E_g}^{E_{g-1}} \frac{dE}{E}$
2	$\int_{E_g}^{E_{g-1}} \phi(E)\, dE \Big/ \int_{E_g}^{E_{g-1}} dE$
5	arbitrary

Table IV

EFFECT OF IWT OPTION

Value of IWT	Group Reaction Cross Sections calculated from library according to:
0	$\dfrac{\int_{E_g}^{183\ keV} \sigma(E)\,\frac{dE}{E} + \int_{183\ keV}^{E_{g-1}} \sigma(E)\phi_F(E)\,dE}{\int_{E_g}^{183\ keV} \frac{dE}{E} + \int_{183\ keV}^{E_{g-1}} \phi_F(E)\,dE}$
1	$\int_{E_g}^{E_{g-1}} \sigma(E)\frac{dE}{E} \Big/ \int_{E_g}^{E_{g-1}} \frac{dE}{E}$
2	$\int_{E_g}^{E_{g-1}} \sigma(E)\,dE \Big/ \int_{E_g}^{E_{g-1}} dE$

Note: $\phi_F(E)$ denotes a fission flux spectrum.

Card type 3, format 6E12.5; ELIM(I).

ELIM(I) is the lower energy limit of neutron energy group I in units of electron-volts. That is, ELIM(I) corresponds to the energy limit E_I on p. 9. Here I is a running index such that I = 1, 2, 3, ..., NOBG, where NOBG is the total number of neutron energy groups entered on card type 2. The values of ELIM(I) are specified six values per card in order of decreasing neutron energy. The upper energy limit of the first neutron energy group is not specified, but is programmed to be 21.17 MeV. The total number of energy limits entered is thus equal to NOBG. As many as seven cards of this type may be required. If IFLX (card type 2) is 5, NOBG energy limits must be given but are not actually used. However, the energy limits will be printed out as part of the output data.

Card type 4, format 24I3; NLIM(I). (Used if IFLX = 5).

NLIM(I) are integers which specify the manner in which the 43-group cross section sets provided by the program user should be collapsed into a set of NOBG groups. This type of card must be submitted if and only if IFLX is 5. Here I is a running index such that I = 1, 2, 3, ..., NOBG. Each value of NLIM specifies the largest group number of each of the broad neutron energy groups. For example, if five neutron energy groups (NOBG = 5) are used to specify the flux such that the first group consists of groups 1-10 of the 43-group structure used to describe the cross sections, the second group consists of groups 11-20, the third of 21-30, the fourth of 31-42, and the fifth of 43 only, then the values of NLIM entered on this card should be 10, 20, 30, 42, 43. The values of NLIM are entered 24 to a card arranged in order of increasing group number. Two cards of this type may be required. The broad group cross section set is obtained by an arithmetic average of the appropriate narrow group cross sections.

Card type 5, format 6E12.5; EGG(I).

EGG(I) are floating point numbers which are the gamma ray energy group limits (p. 24) desired in the NAP computation. Here I is a running index such that I = 1, 2, 3, ..., NOGG+1 where NOGG is the number of gamma ray energy groups entered on card type 2. The EGG(I) are expressed in MeV and must be listed in order of decreasing value, six values to a card. Unlike the specification of the neutron energy group limits, the uppermost gamma ray energy group limit must be specified. Thus EGG(1) is the upper energy limit of the first gamma ray energy group, EGG(2) is the lower energy limit of the first gamma ray energy group, EGG(3) is the lower energy limit of the second gamma ray energy group, etc. All photons of energy above the largest energy limit or below the smallest energy limit are ignored. Up to four cards of this type may be required.

Card type 6, format 6E12.5; FLXIN(I).

FLXIN(I) are floating point numbers which specify the magnitude of the incident neutron flux (p. 9) using the energy group structure previously entered on card type 3. Here I is a running index such that I = 1, 2, 3, ..., NOBG where NOBG is the number of neutron energy groups entered on card type 2. Thus, FLXIN(I) is the magnitude of the incident neutron flux in energy group I. The values of the FLXIN array are listed in order of decreasing energy (increasing group number) with six values to a card. Eight cards of this type may be required. The physical units used depend upon the value entered for IFLX on card type 2. The values of the FLXIN(I) entered on this card are all multiplied by the value of FLUXN (card type 2) during operation of the program. Thus if IFLX is entered as zero, the product of FLUXN and FLXIN(I) should have units of neutrons/cm^2-sec. If IFLX is 1, the product of FLUXN and FLXIN(I) should have units of neutrons per unit lethargy/cm^2-sec. If IFLX is 2, the product of FLUXN and FLXIN(I) should have units

of neutrons/eV-cm^2-sec. If IFLX is 5, the units are arbitrary but must be consistent with the units of the cross sections entered on card type 20.

Card type 7, format 2E12.5, 3I2; EPS, XO, IGEON, IOUT, NOANG. Must be submitted if and only if NOSS (card type 2) is greater than zero.

EPS is a floating point number specifying the convergence criterion (p. 15) used in the NAP neutron transport subroutine. In each neutron energy group, convergence is satisfied when the fractional change in the group flux at the right-most boundary is less than EPS between successive iterations.

XO is the spatial coordinate in centimeters of the left-most boundary (p. 14). An error stop will occur if XO is zero and spherical geometry is used.

IGEON is an integer number indicating the geometry for the neutron transport subroutine. If IGEON is zero, spherical geometry is assumed; if IGEON is 1, slab geometry is assumed.

IOUT is an integer number specifying the quantity of data output generated by the neutron transport subroutine. If IOUT is entered as 3, the angular flux, i.e. $\psi(x,\mu_j)$ on p. 14, obtained from the neutron transport solution is printed out as a function of space (mesh point), energy group, and angle (μ_j). If IOUT is entered as 2, the group flux, i.e. $\psi(x,\mu_j)$ integrated over μ, is printed as a function of space and energy group. If IOUT is entered as 1, the spatially-averaged group flux in each spatial region is printed as a function of energy group.

NOANG is an integer specifying the type and number of discrete ordinates (p. 13) to be used in the neutron transport subroutine. If NOANG is 2, 4, 6, 8, or 10, Legendre-Gauss quadrature is used employing the 2, 4, 6, 8, or 10 angular

ordinates given in Table II (p. 16). If NOANG is 24 or 26, Lobatto quadrature is used employing 4 or 6 ordinates. If spherical geometry is used (IGEON = 0), Legendre-Gauss quadrature will be employed, even if Lobatto quadrature is selected.

Card type 8, format 4E12.4, I2; SIGS(I), SIGT(I), SIGSL(I), DX(I), NINT(I). Must be submitted if and only if NOSS (card type 2) is greater than zero.

SIGS(I) is a floating point number giving the value of the macroscopic neutron scattering cross section (p. 13) for neutron energy group one and region I. Here I is a running index such that I = 1, 2, 3, ..., NOREG where NOREG is the number of regions specified on card type 2. There are as many cards of this type as there are regions in the problem. Thus the first type 8 card gives data pertaining to region one, the second to region two, etc. Note however that SIGS(I) on this card pertains only to the first (highest) neutron energy group. Cross sections for other energy groups are given on card type 10. The physical units of SIGS(I) must be consistent with the units used for FLUXN (card type 2) and FLXIN(I) (card type 6). The product of SIGS, FLUXN, and FLXIN should have units of neutrons/cm^3-sec.

SIGT(I) is analogous to SIGS(I), but is the total cross section.

SIGSL(I) is also analogous to SIGS(I), but is the cross section for neutron scattering from energy group one to energy group two. The SIGSL values, when multiplied by the group one flux values, act as a slowing down source term for the second energy group.

DX(I) is the mesh point spacing (p. 15) in centimeters for region I.

NINT(I) is an integer equal to the number of mesh point intervals in region I. The thickness of region I is thus DX(I) times NINT(I). The total number of mesh intervals summed over <u>all</u> of the regions must be less than 100.

Card type 9, format 5E12.4; AFLX(J). Must be submitted if and only if both NOSS and NISO (card type 2) are greater than zero.

AFLX(J) is a floating point number which specifies the value of the incident neutron flux in energy group one at the discrete ordinate μ_J (p. 14). The number of AFLX values entered should be one-half the number of discrete ordinates indicated by NOANG on card type 7, inasmuch as only the forward components of the angular flux are to be specified. The AFLX values should be entered in order of increasing θ (decreasing μ), and must be normalized such that the sum of the values times the angular weights pertaining to both forward and backward components is unity. That is, the value of AFLX(J) must be equal to one-half the fractional neutron flux in the first neutron energy group incident upon the left-most boundary directed at an angle $\theta_J = \cos^{-1} \mu_J$ to the normal. Only positive values of μ_J are considered. An example is given in the following paragraph. If NISO is entered as zero, this card is not submitted and the AFLX values are internally programmed to be 0.5.

As an example, suppose that the neutron flux incident upon the left-most boundary of the system is known as a function of energy and angle to the normal. A typical angular dependence is shown in Figure 4. An isotropic flux is also shown for comparison. If Legendre-Gauss quadrature with four ordinates are used, two values of AFLX(J) should be entered on card type 9. For the example shown in Figure 4, these two values should be 0.795 and 0.550 in that order.

Card type 10, format 3E12.4; SIGS(I), SIGT(I), SIGSL(I). Must be submitted if and only if NOSS is greater than zero and NOBG is greater than one. This card is analogous to card type 8, except that the mesh point spacing and number of mesh intervals is omitted. The cross sections now refer to the second neutron energy group. As before, I is a running index

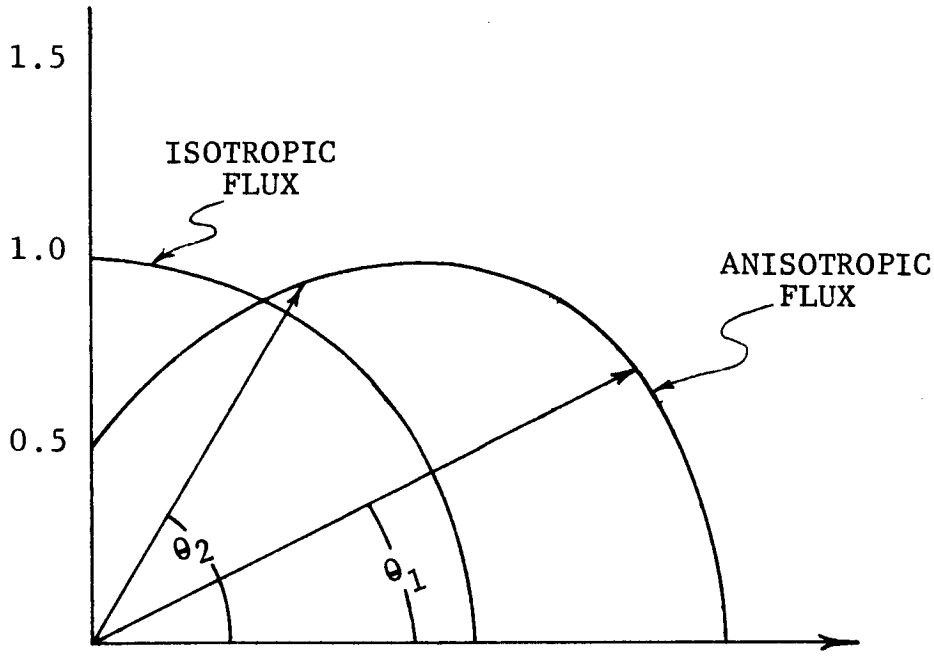


FIGURE 4

EXAMPLE OF ANISOTROPIC FLUX

which indicates the region number, I = 1, 2, 3, ..., NOREG. These cards are ordered by increasing region number, the region numbers increasing from left to right.

If both NOSS and NISO are greater than zero and NOBG is greater than one, a card of type 9 now follows the above set of card type 10. The AFLX values now refer to the anisotropy of the incident neutron flux in the second neutron energy group. Card types 10 and 9 are repeated until data for all the neutron energy groups have been submitted.

To clarify the ordering of card types 8, 9, and 10, the following example may be considered. Suppose a NAP problem is set up with three neutron energy groups, two spatial regions, the incident neutron flux is anisotropic in a known manner, and it is desired to use the NAP neutron transport subroutine. On card type 2, NOBG should be given as 3, NOREG as 2, NOSS and NISO as some integer greater than zero, say 1. Card types 8, 9, and 10 should be submitted in the following order:

1. card type 8 for region one and neutron energy group one
2. card type 8 for region two and energy group one
3. card type 9 for neutron energy group one
4. card type 10 for region one and neutron energy group two
5. card type 10 for region two and energy group two
6. card type 9 for neutron energy group two
7. card type 10 for region one and neutron energy group three
8. card type 10 for region two and energy group three
9. card type 9 for neutron energy group three

If the incident neutron flux were known to be isotropic, then NISO on card type 2 should be given as zero. The ordering of the cards is the same as that just given, except that all cards of type 9 must be omitted.

Card type 11, format 6E12.5; FLX(I). Must be submitted if and only if NOREG (card type 2) is greater than one and NOSS (card type 2) is less than zero.

FLX is an array of floating point numbers used to describe the average neutron fluxes in each spatial region as a function of neutron energy group. If NOSS is greater than zero, the NAP neutron transport subroutine is used to generate the FLX array. If NOSS is zero, the FLXIN array submitted on card type 6 is assumed to apply in each spatial region, i.e. the neutron flux is independent of spatial region. If NOSS is less than zero, the neutron flux in each energy group and each spatial region (other than the first) must be supplied on this type input card. Here I is a running index such that I = NOBG+1, NOBG+2, ..., (NOBG) x (NOREG). For example, if NOSS is less than zero, NOREG is 3, and NOBG is 4, then the group fluxes in region one should be entered on card type 6 for energy groups one through four, in order of increasing group number (decreasing energy). The cards of type 11 should then contain eight numbers, six on the first card and two on the second. The first four numbers should be the group fluxes in region two, in order of increasing group number; the second set of four numbers should be the group fluxes in region three, in order of increasing group number.

Card type 12, format E12.5; TFAC, NONV.

TFAC is a floating point number which specifies the thermal averaging parameter (p. 21). All group 43 neutron reaction cross sections are multiplied by TFAC. Thus TFAC is that factor such that multiplication of the 2200 m/sec cross section value of a 1/v cross section by TFAC yields the effective thermal cross section. For example, if the thermal neutron flux is assumed to have a Maxwellian velocity distribution with a most probable velocity of 2200 m/sec, then TFAC is the ratio of the most probable velocity to the average velocity and should be entered as 1/1.128 = 0.8862.

NONV is an integer equal to the number of isotopes in the problem (in all spatial regions for which gamma ray source strengths are desired) whose thermal cross sections are not 1/v. The maximum value of NONV is ten.

Card type 13, format 3(2I6, E12.5); NZ(I), NA(I), VFAC(I). Must be submitted if and only if NONV (card type 12) is greater than zero.

NZ(I) is an integer equal to the atomic number of the Ith isotope having a non-1/v thermal cross section. Here I is a running index such that I = 1,2,..., NONV.

NA(I) is an integer equal to the atomic mass number of the Ith isotope having a non-1/v thermal cross section.

VFAC(I) is a floating point number equal to the non-1/v factor (p. 21) for the Ith isotope having a non-1/v thermal cross section. In computing neutron reaction rates for isotopes originally present in the problem, the NAP program searches the NZ, NA table provided by cards of type 13. If both the atomic number and the mass number of the isotope under consideration are found in the table, the thermal (Cross Section Library neutron energy group 43) cross section is multiplied by both TFAC and VFAC(I) to obtain an effective thermal cross section for use in computing the reaction rate. Three sets of NZ, NA, and VFAC are permitted on each card of type 13 in the order NZ(1), NA(1), VFAC(1), NZ(2), NA(2), VFAC(2), NZ(3), NA(3), VFAC(3). Since NONV must be equal to or less than ten, a maximum of four cards of type 13 is permitted, the last card having only one set of values.

Card type 14, format 2I6; NOPER, NODOS.

NOPER is an integer equal to the total number of time periods in which the power level (p. 9) is constant. NOPER is limited to a value of 50 or less. For example, if P(t) is given by 22 values as indicated in Figure 2, NOPER must be entered as 22.

NODOS is an integer equal to the number of initial and final times to be used in the dose calculations (p. 25).

Each dose calculation consists of a numerical integration of the dose rate from some initial time to some final time. These times are supplied on card type 18. NODOS must be equal to the total number of such times and must be 50 or less. For example, suppose three dose calculations are desired, the first from time t_1 to time t_2, the second from t_3 to t_4, and the third from t_4 to t_5. Six values for the time must be supplied on card type 18, the fourth such value being repeated, and NODOS must be entered as six. It may be noted that NODOS is twice the number of dose calculations desired.

Card Type 15, format 6E12.5; POW(I).

POW(I) is a floating point number which gives the value of the power level P(t) (p. 9) during the Ith time period. Here I is a running index such that I = 1,2,..., NOPER. The physical units are arbitrary, but must be consistent with the units used for FLUXN on card type 2 and FLXIN(I) on card type 6. It is essential that the product of POW, FLUXN, FLXIN, and the neutron cross sections, whether taken from the NAP Cross Section Library or supplied by the user, has dimensions of neutrons/sec. The number of values of POW(I) submitted, six to a card, must be equal to the value of NOPER (card type 14). Thus, for the example shown in Figure 2, 22 values of POW(I) must be entered. A value of zero entered for any POW(I) implies that there is no incident neutron flux during the Ith time period.

Card Type 16, format 6E12.5; TI(I).

TI(I) is a floating point number equal to the duration in hours, of the Ith time period. Here I is a running index such that I = 1,2,..., NOPER. The power level POW(I) is constant throughout the time period of length TI(I). There must be NOPER values of TI(I) given, six per card.

Card Type 17, format 12I6, NINT(I).

NINT(I) is an integer equal to the number of equal time intervals contained in the time period of length TI(I). Again I is a running index such that I = 1,2,..., NOPER. Thus the Ith time period of length TI(I), during which the power level is constant at the value POW(I), is divided into NINT(I) time intervals, each of length TI(I)/NINT(I). The radioisotopic atom densities, gamma ray source strengths, and gamma ray dose rates are computed at the end of each time interval. There must be NOPER values of NINT(I) given, twelve per card. The total number of time intervals used in all the time periods is limited to 200. That is, the sum of the values of all the NINT(I) must be equal to or less than 200.

As an example, suppose the power level is unity for one hour, and then is zero. Then NOPER is 2, POW(1) is 1.0, and POW(2) is 0.0. If it is desired to compute the gamma ray dose rate every ten minutes during the irradiation, and every five minutes for three hours after the irradiation, then TI(1) is 1.0, TI(2) is 3.0, NINT(1) is 6, and NINT(2) is 36.

Card Type 18, format 6E12.5; TS(I).

TS(I) is a floating point number equal to an initial or final time (in hours) used in calculating the gamma ray dose. Here I is a running index such that I = 1,2,..., NODOS. This card must be submitted even if NODOS is zero, in which case a blank card may be used. There must be NODOS values of TS(I) entered, six per card. For the example discussed above under card type 14, one card of type 18 should be submitted containing the values t_1, t_2, t_3, t_4, t_4, t_5.

Card Type 19, format I6; NX.

NX is an integer which specifies the number of neutron reaction cross section sets supplied by the program user, to be used in preference to the NAP Cross Section Library. Each reaction type for each isotope is counted as a single set. If no cross sections are supplied, NX should be entered as zero.

Card Type 20, format 10E8.1; X(M).

X(M), M = 1,2,...,50, is an array of floating point numbers, ten per card, giving the microscopic cross section set supplied by the program user. Five cards of this type constitutes a single cross section set. These cards must not be submitted if NX (card type 19) is zero.

X(1) = atomic number (Z) of isotope described by this set of cross sections.
X(2) = mass number (A) of isotope.
X(3) = fractional abundance of naturally occuring isotope.
X(4) = a number describing the type of cross section given. Acceptable values of X(4), and their meanings, are given in Table V.
X(5) = microscopic potential scattering cross section (barns).

If X(4) is less than 100,

X(6) = cross section (barns) for neutron energy group 1.
X(7) = cross section (barns) for neutron energy group 2.
⋮ ⋮ ⋮
X(48) = cross section (barns) for neutron energy group 43.

If X(4) is 100 or 200,

X(6) = neutron resonance energy (eV) for first resonance.
X(7) = resonance statistical factor g for first resonance.

Table V

IDENTIFICATION OF CROSS SECTION TYPE

X(4)	type of reaction	product nucleus ground state	product nucleus isomeric state
1	(n,γ)	x	
2	(n,p)	x	
3	(n,α)	x	
4	(n,2n)	x	
11	(n,γ)		x
12	(n,p)		x
13	(n,α)		x
14	(n,2n)		x
100	(n,γ) resolved resonance	x	
200	(n,γ) resolved resonance		x

X(8) = resonance capture width (eV) for first resonance.
X(9) = resonance neutron width (eV) for first resonance.
X(10) = resonance parasitic width (eV) for first resonance.
X(11) = neutron resonance energy (eV) for second resonance.
⋮ ⋮ ⋮ ⋮
X(50) = resonance parasitic width (eV) for ninth resonance.

If X(4) is less than 15, the NAP program assumes that the cross section set is supplied, using a 43-group neutron energy structure, in the order of decreasing energy. The cross section for group 43 is assumed to be a 2200 m/sec value, i.e., it is multiplied by TFAC and by VFAC, if appropriate, in computing the reaction rate. If IFLX (card type 2) is five, the 43-group energy structure need not be identical to that used in the NAP Cross Section Library. Five cards of type 20 must be supplied for each cross section type given. Additional cross section sets are placed in order of increasing Z, increasing A, and increasing value of X(4). For example, cross section sets for $_{30}Zn^{70}$ must preceed those for $_{31}Ga^{69}$.

Card type 21, format I12,4E12.5; ISOR, R, TEMP, VOL, RD. This card type and the following are grouped together in order of increasing region number. There must be NOREG cards of this type.

ISOR is an integer specifying the number of isotopic or elemental atom densities to be specified as initially present in the region. ISOR must not be greater than 20. If ISOR is zero, the region is not considered in the computation of gamma ray source strengths, but is considered in the neutron transport problem if NOSS (card type 2) is greater than zero.

R is a floating point number giving one-half the radius or one-half the thickness of the region, in centimeters. For infinitely-long cylinders, R is the radius; for infinite slabs, R is the thickness. R is used only in the computation of effective resonance integrals (see Volume II). A machine overflow will

occur if R is entered as zero and an effective resonance integral calculation is attempted.

TEMP is a floating point number giving the temperature (degrees Fahrenheit) of the region, and is used only in effective resonance integral calculations.

VOL is a floating point number giving the volume (cm^3) of the region.

RD is a floating point number giving the distance (cm) from the center of the region to the position where gamma ray dose rate and dose information is desired. If RD is entered as zero, no such information will be obtained, and the NAP program will advance to the next region, after computing the gamma ray source strengths.

Card Type 22, format 3(2I3, I6, E12.5); IZ(I), IA(I), IKEY(I), DEN(I).

IZ(I) is an integer giving the atomic number of the Ith isotope initially present in the region. On this type card, I is a running index such that I = 1,2,..., ISOR.

IA(I) is an integer equal to the atomic mass number of the Ith isotope initially present in the region. If IA(I) is entered as zero, the NAP program assumes that the element specified by IZ(I) is present in its naturally occurring isotopic composition, provided that composition is available in either the NAP Cross Section Library or in the Cross Section sets supplied by the program user (see card type 20).

IKEY(I) is an integer which controls the origin of all neutron reaction cross sections for the Ith isotope or element according to the following scheme:

IKEY = 0: cross sections will be calculated if not found in the NAP Cross Section Library.

IKEY = 1: cross sections will be set equal to zero if not found in the library.

IKEY = 2: cross sections will be calculated even if in the library.

IKEY = 3: cross sections are supplied (card type 20) by the user. If not found in the supplied data, they will be calculated.

DEN(I) is a floating point number giving the isotopic (or elemental) atom density (10^{24} atoms/cc) of the $I^{\underline{th}}$ isotope.

Each card of this type may contain data for three isotopes. As an example of the ordering of card types 21 and 22, consider a two-region problem with five isotopes initially present in the first region and four in the second. The proper order is then:

card type 21 for first region
card type 22 for first region (three isotopes)
card type 22 for first region (two isotopes)
card type 21 for second region
card type 22 for second region (three isotopes)
card type 22 for second region (one isotope)

Finally, a summary of program options available to the problem originator is given in Table VI, a dictionary of all input variables in Table VII, and a summary of input card formats in Table VIII.

TABLE VI

NAP PROGRAM OPTIONS

Card Type	Variable	Suggested Value	Effect
2	NOSS	-1	Regional fluxes are input
		0	Flux is spatially uniform
		-1	Perform neutron transport calculation
2	NISO	0	Incident flux is isotropic
		1	Incident flux is anisotropic
2	IFLX	0	Input group fluxes are integrals over energy
		1	Input group fluxes are averages per unit lethargy
		2	Input group fluxes are averages per unit energy
		5	Input group fluxes are undefined and energy limits are not adjusted
2	IWT	0	Cross Sections are weighted by fission flux spectrum and 1/E flux spectrum
		1	Cross sections are weighted by 1/E flux spectrum
		2	Cross sections are weighted by constant flux spectrum
7	IGEON	0	Spherical geometry
		1	Plane geometry
7	IOUT	1	Print average group flux by region
		2	Print average group flux and group flux by mesh point

TABLE VI (CONT'D.)

NAP PROGRAM OPTIONS

Card Type	Variables	Suggested Value	Effect
		3	Print average group flux, group flux, and angular flux
22	IKEY(I)	0	Calculate cross sections (for isotope I) if not found in library
		1	Set cross sections equal to zero if not found
		2	Calculate cross sections
		3	Cross sections are supplied as input

TABLE VII

NAP INPUT DICTIONARY

Name	Card Type	Meaning
AFLX(J)	9	Normalized incident angular flux (used only if NOSS >0 and NISO $\neq 0$) (No more than 5 values)
DEN(I)	22	Initial isotopic atom density (10^{24}/cc) for isotope I.
DX(I)	8	Mesh spacing (cm) in region I (used only if NOSS >0)
EGG(I)	5	Gamma energy group limits (MeV)
ELIM(I)	3	Neutron flux energy group limits (eV)
EPS	7	Convergence criterion for neutron transport calculation
FLUXIN(I)	6	Incident neutron energy group fluxes
FLUXN	2	Neutron flux normalization factor
FLX(I)	11	Regional neutron energy group fluxes (used only if NOSS <0)
IA(I)	22	Atomic mass number of initial isotope I (if IA = 0, natural abundance is used)
IFLX	2	0,1,2 if FLUXIN(I) is integral flux, flux per unit lethargy, or flux per unit energy, respectively. If IFLX = 5, FLUXIN(I) is arbitrary.
IGEON	7	0 for spherical geometry, 1 for slab geometry (used only if NOSS >0)
IKEY(I)	22	Supplied for isotope I: 0: σ calculated if not in library 1: $\sigma = 0$ if not in library 2: σ always calculated 3: σ applied as input
IOUT	7	Used if and only if NOSS >0: 3: print angular flux 2: print group flux 1: print average group flux

TABLE VII (CONT'D.)

NAP INPUT DICTIONARY

Name	Card Type	Meaning
ISOR	21	Number of initial isotopes in region I ($\leq$ 20)
IWT	2	Indicates weighting for averaging of group σ's: 0,1,2 indicates fission and 1/E weighting, 1/E weighting, and constant weighting, respectively
IZ(I)	22	Atomic number of initial isotope I
NA(I)	13	Atomic number of I^{th} non-1/v isotope (used if and only if NONV $>$ 0)
NINT(I)	8	Number of mesh intervals in region I (used if and only if NOSS $>$ 0) (Sum over I $<$ 100)
NINT(I)	17	Number of time intervals in time period I (Sum over I $\leq$ 200)
NISO	2	$>$ 0 if incident flux is anisotropic (used only if NOSS $>$ 0)
NLIM(I)	4	Used if and only if IFLX = 5. Specific group structure of incident flux for σ weighting.
NOANG	7	Number of flux angular ordinates (used if and only if NOSS $>$ 0) (Restricted to 2,4,6,8, 10,24, or 26)
NOBG	2	Number of neutron energy groups ($\leq$ 43)
NODOS	14	Number of initial and final dose times ($\leq$ 50)
NOGG	2	Number of gamma energy groups ($\leq$ 20)
NONV	12	Number of non-1/v isotopes ($\leq$ 10)
NOPER	14	Number of constant flux time periods ($\leq$ 50)
NOREG	2	Number of spatial regions ($\leq$ 20)
NOSS	2	$<$ 0, regional fluxes are input = 0, flux is spatially uniform $>$ 0, perform neutron transport calculation
NX	19	Number of σ sets given as input
NZ(I)	13	Atomic number of I^{th} non-1/v isotope (used if and only if NONV $>$ 0)

TABLE VII (CONT'D.)

NAP INPUT DICTIONARY

Name	Card Type	Meaning
POW(I)	15	Neutron flux intensity level for time period I
R	21	One-half times mean chord length (cm) ($\neq 0$)
RD	21	Distance (cm) from source to detector
SIGS(I)	8,10	Scattering cross section (cm^{-1}) for region I (used if and only if NOSS > 0)
SIGSL(I)	8,10	Slowing down cross section (cm^{-1}) for region I (used if and only if NOSS > 0)
SIGT(I)	8,10	Total cross section (cm^{-1}) for region I (used if and only if NOSS > 0)
TEMP	21	Temperature (degrees F)
TFAC	12	Thermal cross section averaging parameter
TI(I)	16	Length (hrs) of time period I
TS(I)	18	Time (hrs) at which dose starts or stops
VFAC(I)	13	Non-1/v factor for non-1/v isotope I
VOL	21	Volume (cm^3) of region
X(M)	20	Input values of "cross section"
XO	7	Coordinate (cm) of system left hand boundary ($\neq 0$ if IGEON = 0)

TABLE VIII

INPUT DATA FORMAT SUMMARY

Card Type	Variables	Format	Comment
1	-----	12A6	problem title
2	FLUXN, NOBG, NOREG, NOGG, NOSS, NISO, IFLX, IWT	E12.5, 7I6	
3	ELIM(I)	6E12.5	must provide NOBG values
4	NLIM(I)	24I3	used if and only if IFLX = 5, must provide NOBG values
5	EGG(I)	6E12.5	must provide NOGG+1 values
6	FLUXIN(I)	6E12.5	must provide NOBG values
7	EPS, XO, IGEON, IOUT, NOANG	2E12.5, 3I2	used if and only if NOSS >0
8	SIGS(I), SIGT(I), SIGSL(I), DX(I), NINT(I)	4E12.4, I2	used if and only if NOSS >0
9	AFLX(J)	5E12.4	used if and only if NOSS >0 and NISO $\neq 0$
10	SIGS(I), SIGT(I), SIGSL(I)	3E12.4	used if and only if NOSS >0 and NOBG >1
11	FLX(I)	6E12.5	used if and only if NOSS <0, must provide NOBG x (NOREG-1) values
12	TFAC, NONV	E12.5, I6	
13	NZ(I), NA(I), VFAC	3(2I6, E12.5)	used if and only if NONV >0, must provide NONV values
14	NOPER, NODOS	2I6	
15	POW(I)	6E12.5	must provide NOPER values
16	TI(I)	6E12.5	must provide NOPER values
17	NINT(I)	12I6	must provide NOPER values
18	TS(I)	6E12.5	must provide NODOS values
19	NX	I6	
20	X(M)	10E8.1	used if and only if NX >0, must provide 5 times NX cards

TABLE VIII (CONT'D.)

INPUT DATA FORMAT SUMMARY

Card Type	Variables	Format	Comment
21	ISOR,R,TEMP VOL,RD	I12, 4E12.5	must provide NOREG cards
22	IZ(I),IA(I) IKEY(I),DEN(I)	3(2I3,I6, E12.5)	must provide ISOR sets for each region

IV. OPERATING INSTRUCTIONS

The NAP program has been written entirely in the IBM 7090/7094 FORTRAN IV language, and is designed for operation under the IBSYS Operating System (Version 13). The program binary decks currently available are for use on an IBM 7094.

Correct operation of the NAP program requires placement of the NAP Cross Section Library tape on FORTRAN tape unit 8 and the NAP Gamma Radiation Library tape on FORTRAN tape unit 10. FORTRAN tape unit 1 is used to store any cross section data submitted by the program user. FORTRAN tape unit 2 is used for temporary storage during program operation.

As an aid in determining the rapidity of convergence when the neutron transport subroutine is used to calculate the spatial dependence of the neutron fluxes, sense lights 1 and 2 may be observed. Both sense lights are off prior to the use of the neutron transport subroutine. At the start of the iterations for the first group flux solution, sense light 1 turns on and remains on until a converged solution for the first group flux is attained. Sense light 1 remains off until a converged solution for the second group flux is attained. Thus, as convergence is achieved for each energy group flux, the status of sense light 1 will change. When a converged solution has been obtained for all of the energy group fluxes, sense light 2 turns on. Depressing sense switch 1 results in a bypassing of the convergence test after five iterations are completed in each group.

Sense switches 2 and 3 may be depressed to obtain supplemental output data, as explained in the following section. All output is written on FORTRAN tape unit 6; input data is assumed to be on FORTRAN tape unit 5.

V. OUTPUT DATA

NAP program output is largely self-explanatory. Typical input and output are illustrated by the sample problem in Section VIII of this report. If region-dependent neutron group fluxes are input to the problem, the group fluxes will appear in the output in order of increasing region number, but the regions are not identified on the flux output. In many cases, two or more atom densities will be printed for the same radioisotope. This will occur whenever the radioisotope is produced in more than one manner. Atom densities, gamma ray source strengths, energy source strengths, dose rates, and doses are printed as a function of time, region by region.

Sense switch 3 may be depressed to produce additional output. This consists of the results of effective resonance integral calculations and the broad group cross section sets resulting from condensation of the NAP Cross Section Library sets.

Sense switch 2 may be depressed to produce some "debugging" output. This includes cross section sets and resolved resonance parameters as obtained directly from the NAP Cross Section Library. Additional informtion pertaining to effective resonance integral calculations is printed. Cross section sets calculated by the NAP program are also printed using the NAP library 43-group energy structure. The groups are not identified on the output, but the first column contains group cross sections for groups 1-15, the second column groups 16-30, and the third column 31-43. Each decay chain is described by printing out the isotopic identification of each chain member. Isomeric states are identified by mass numbers in excess of 500, e.g., mass number 683 indicates an isomeric state of mass number 183. The Q_i and S_j defined in Section II-B, but without the $P(\Delta t)$ factor, are printed for each chain member. In addition, the chain member atom densities (in units of 10^{24} atoms/cc) are printed, prior to dumping on magnetic tape for later use. Here the time

in hours is identified by T and the atom density of chain member I by N(I).

VI. FLOW CHARTS

This section presents a brief description and a flow chart for each subprogram of the NAP code. Arguments, dimensioned variables, common variables, subroutines called, and subroutines entered from are listed for each subprogram. The flow charts show explicitly the operation of the input program control options available to the program user. The main program is described first, followed by the subprograms in alphabetical order.

A. MAIN PROGRAM

1. Purpose: The main program reads in most of the input data and links together most of the NAP subroutines. It sets up the fixed neutron energy group limits consistent with the NAP cross section library, sets up the time intervals used in computing isotopic concentrations, dose rate, and dose, and computes the regional gamma source strength densities.
2. Arguments: None
3. Dimensioned Variables:

A(5)	IKEY(20)	REGS(20,200)
BX(12)	IZ(20)	SOR(20,5)
D(5)	KMAX(20)	SPOT(20)
DEN(20)	NA(10)	T(200)
EGG(21)	NINT(50)	TI(50)
ELIM(43)	NLIM(43)	TS(50)
FELIM(44)	NZ(10)	VFAC(10)
FLX(860)	POW(50)	VOL(20)
FLXIN(43)	POWR(200)	Z(5)
IA(20)	RD(20)	

4. Common Variables:

BS*: EGG, LASTT, NOGG, POWR, SOR, T
CF: ELIM, FELIM, FLXIN, IFLX, IWT, NA, NOBG, NONV, NZ, TFAC, VFAC
CP: BX,IC,LEAF
CQ: NLIM
R: DEN,IA,ISOR,IZ,R,TEMP,SPOT

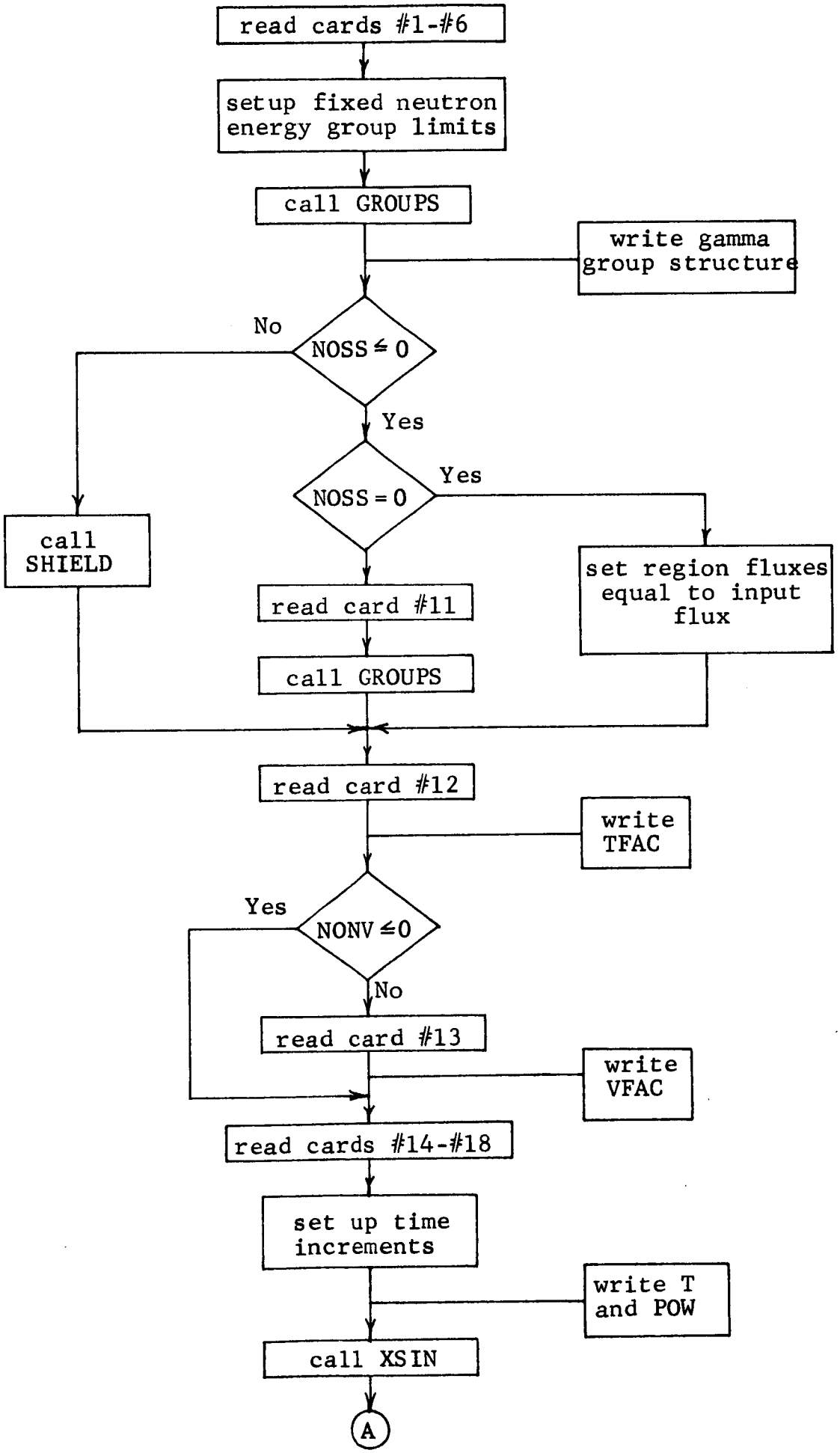
* name of labeled common

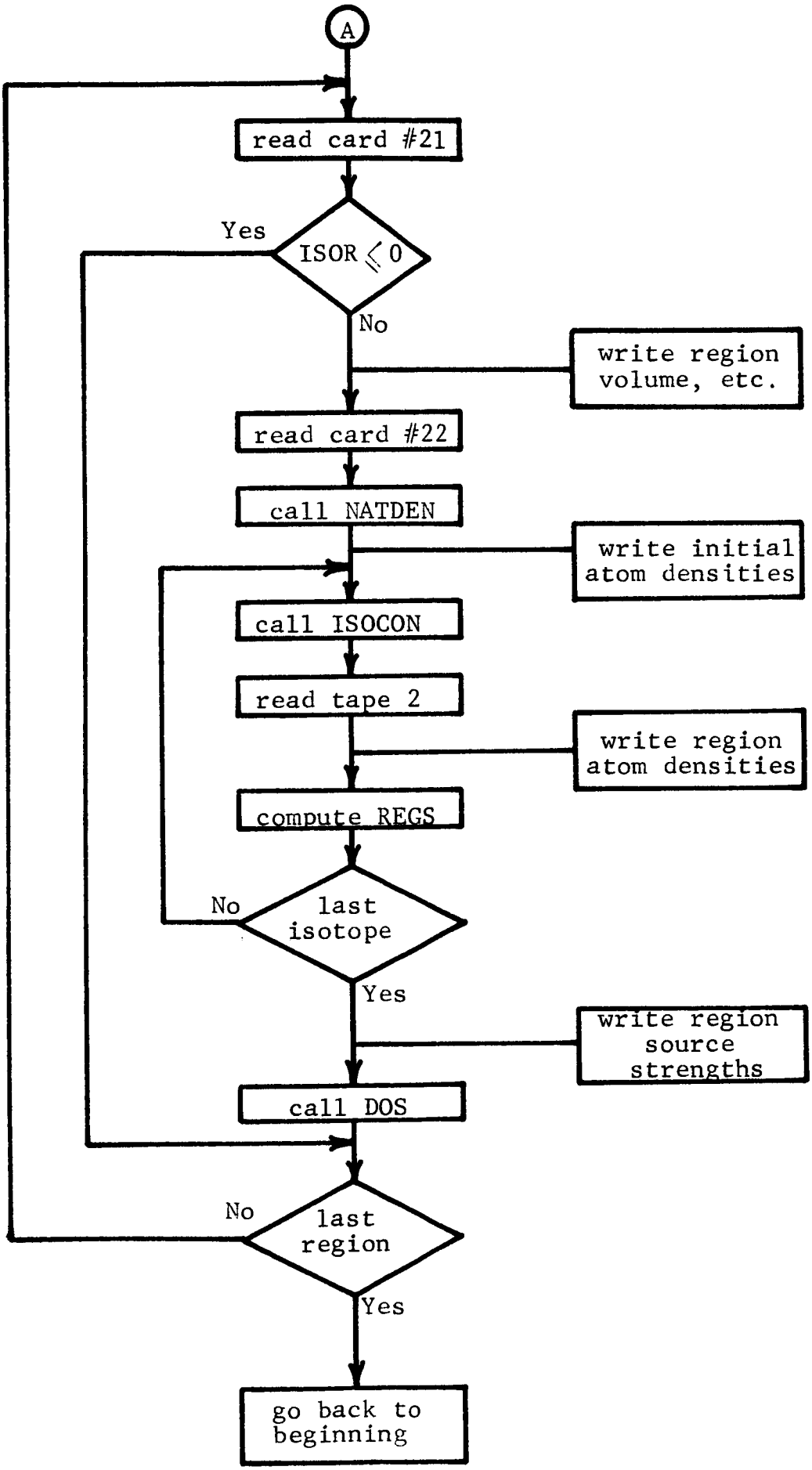
5. Called Subprograms:

   DOS, GROUPS, ISOCON, NATDEN, PAGE, RLIM,
   SHIELD, XSIN.

6. Calling Subprograms: None.

7. Comments: The position in the program of various write statements are indicated on the right hand side of the logical flow chart to aid in malfunction analysis. The array name REGS, referred to after calling ISOCON, signifies REGS(I,J), the regional gamma source strength density for gamma energy group I and time step J. It is computed from data dumped on tape 2 by subroutine ISOCON and is used later by subroutine DOS. Although not indicated on the flow chart, the main program calls RLIB and PAGE. The first statement of the main program calls RLIB to initiate reading of part of the NAP Radiation Library into an allocated portion of core storage. PAGE is utilized with most of the write statements to provide page numbering of output data.

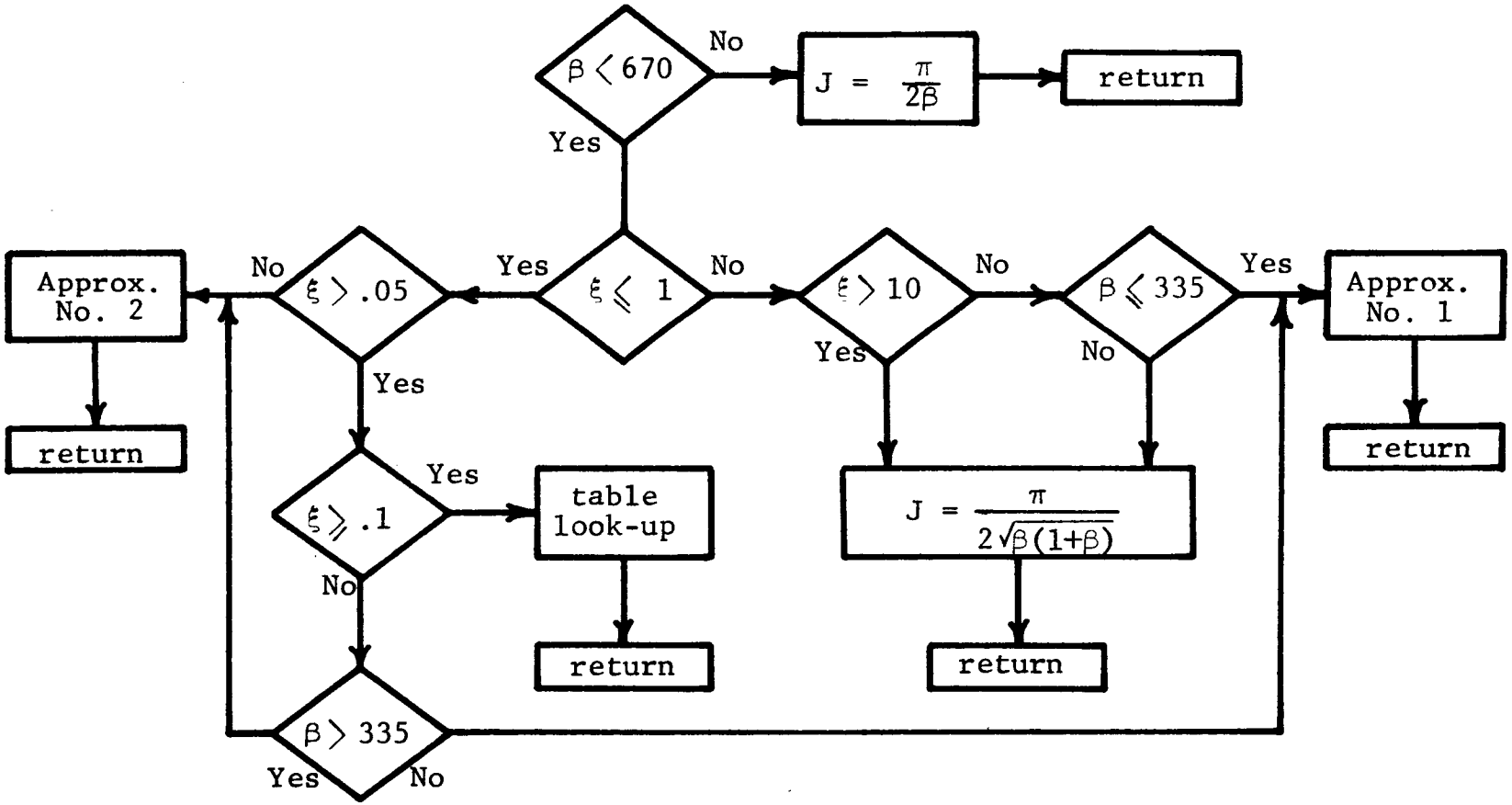

read cards #1-#6
setup fixed neutron energy group limits
call GROUPS
write gamma group structure
No
NOSS ≤ 0
Yes
NOSS = 0
Yes
call SHIELD
set region fluxes equal to input flux
read card #11
call GROUPS
read card #12
write TFAC
Yes
NONV ≤ 0
No
read card #13
write VFAC
read cards #14-#18
set up time increments
write T and POW
call XSIN
A


A
read card #21
Yes
ISOR ⩽ 0
No
write region volume, etc.
read card #22
call NATDEN
write initial atom densities
call ISOCON
read tape 2
write region atom densities
compute REGS
No
last isotope
Yes
write region source strengths
call DOS
No
last region
Yes
go back to beginning

B. Function AJ

1. Purpose: Computes Dresner's J-function with ξ and β asn input. See Volume II.
2. Arguments: XI, BETA.
3. Dimensioned Variables: BJ(27,10)
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: RESINT
7. Comments: Tabular data are contained in the array BJ which is set up by three data statements. Approximation No. 1 is given by equation (87) of Volume II, approximation No. 2 by equation (84).

FUNCTION AJ FLOW CHART



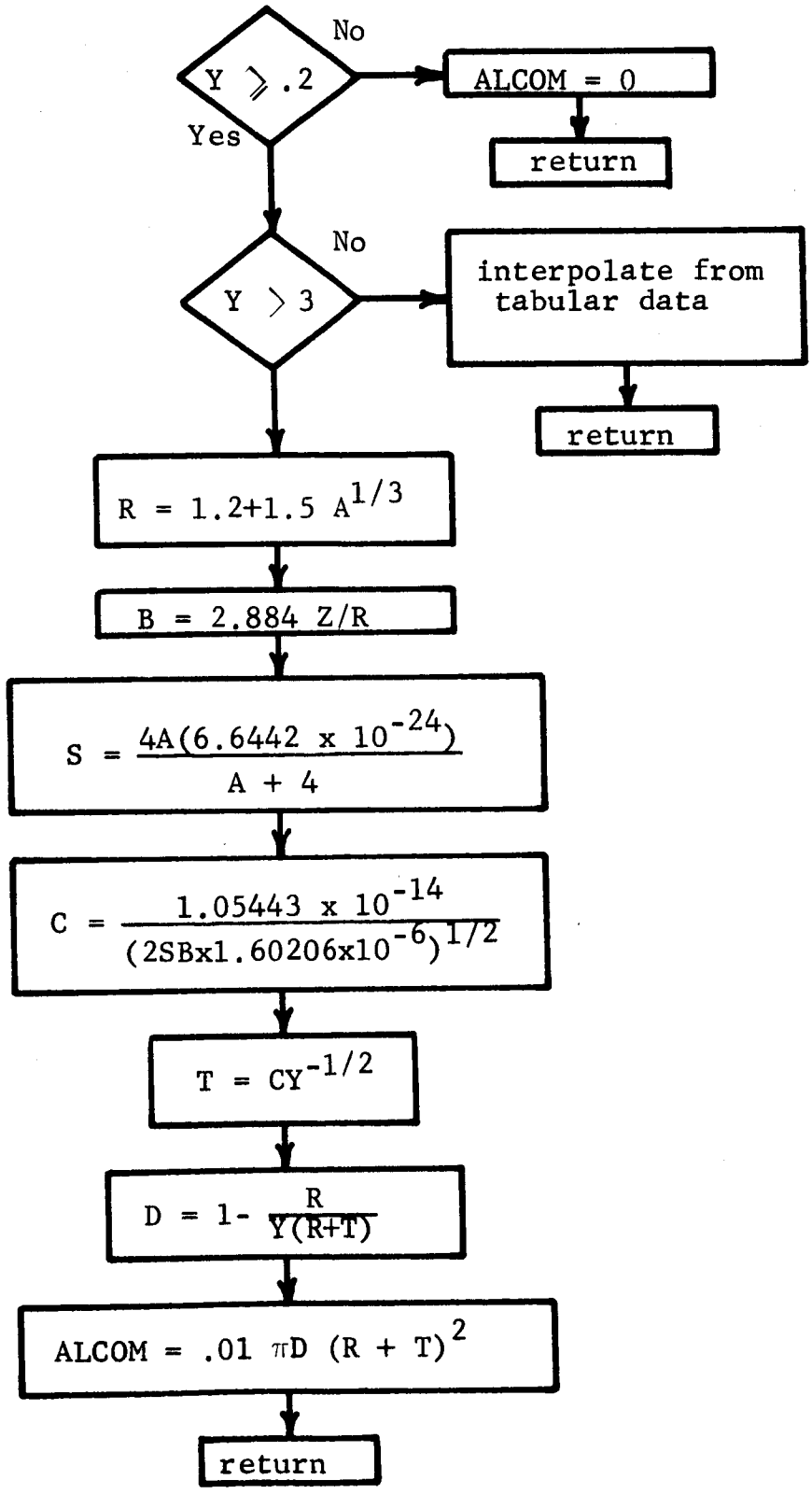
C. Function ALCOM

1. Purpose: Obtain compound nucleus formation cross section of nucleus, specified by Z and A, due to incident alpha particle of energy specified by parameter Y.
2. Arguments: Y, Z, A.
3. Dimensioned Variables:

FLIS(4)	FFLIST(4)	YL(29)
FLIST(29,6)	XLIST(4)	ZL(6)

4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: FFUN.
7. Comments: Tabular data are contained in the array FLIST as a function of YL and ZL. See Volume II for other notation.

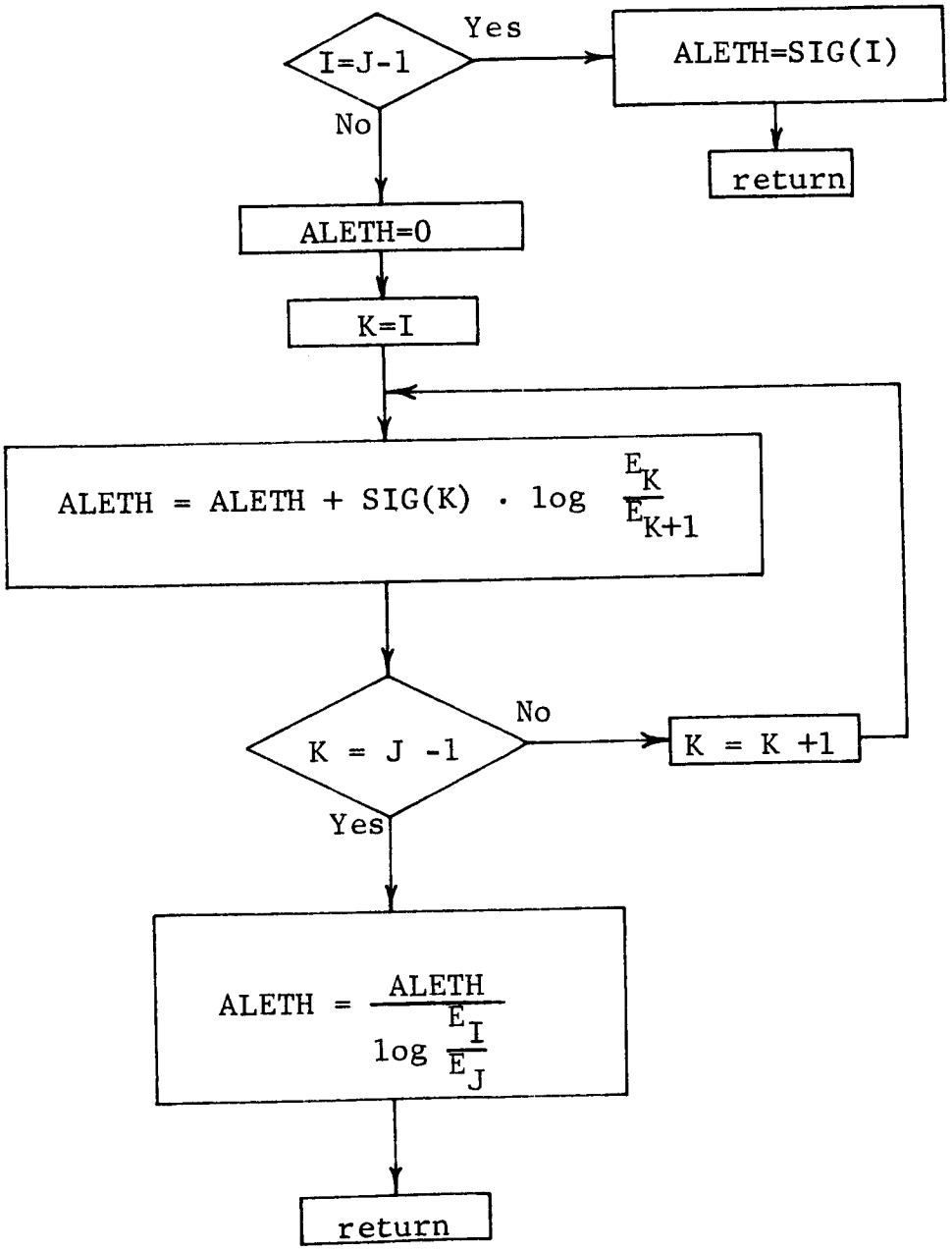
FUNCTION ALCOM FLOW CHART



D. Function ALETH

1. Purpose: Averages the cross sections in the array SIG over a 1/E flux to obtain cross sections in the group structure specified by the program user.
2. Arguments: FELIM, SIG, I, J.
3. Dimensioned Variables: FELIM(44), SIG(43).
4. Common Variables: None
5. Called Subprograms: None
6. Calling Subprograms: FS
7. Comments: SIG is averaged from energy group I to energy group J, inclusive. FELIM(K) is the lower energy limit of energy group K-1 and is indicated in the flow chart by E_K.

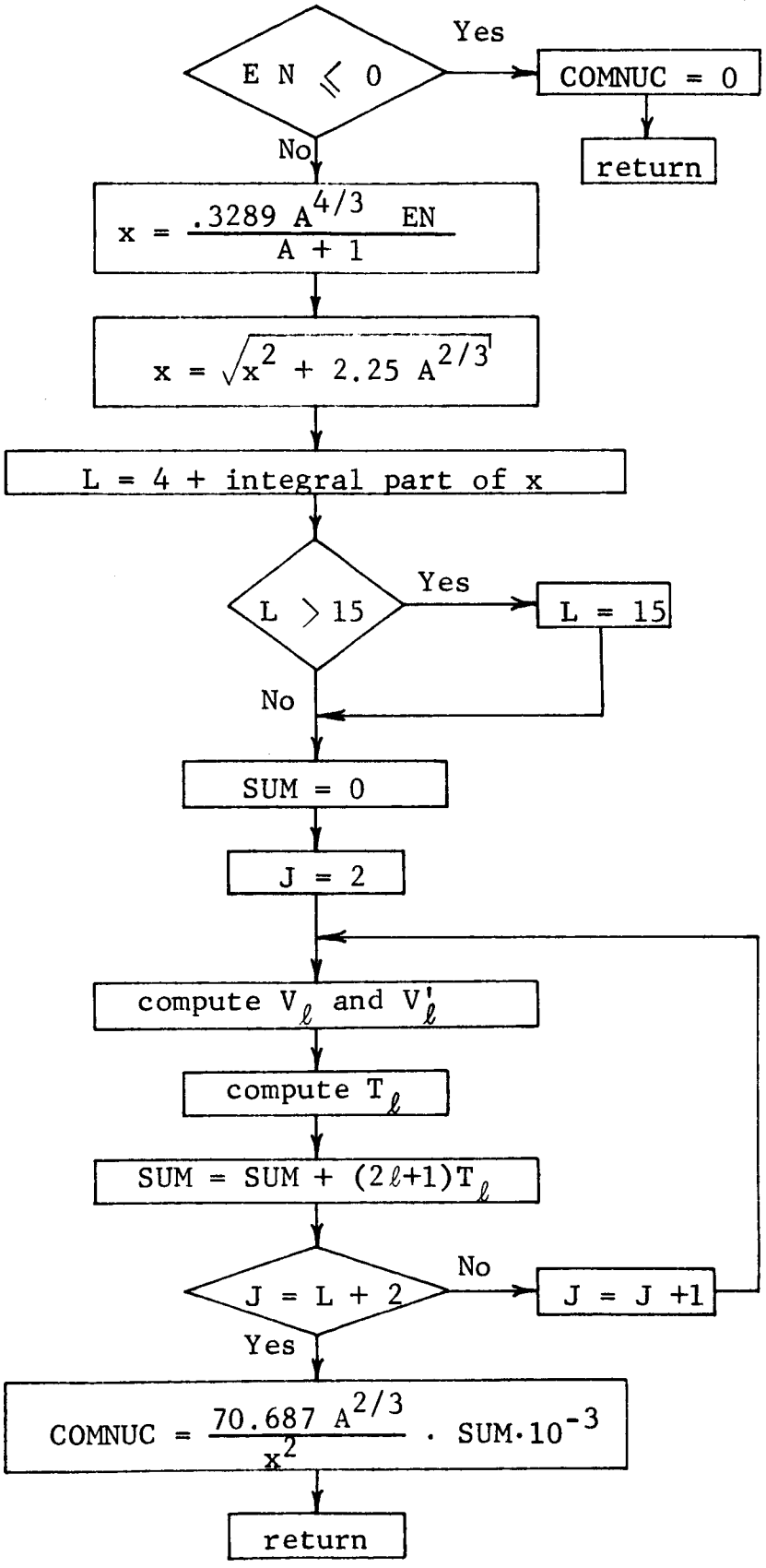
FUNCTION ALETH FLOW CHART



E. Function COMNUC

1. Purpose: Computes compound nucleus formation cross section for isotope of mass A due to incident neutron of energy EN.
2. Arguments: A, EN.
3. Dimensioned Variables: F(22), G(22).
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: FFUN, SIGGAL.
7. Comments: See Volume II for notation.

FUNCTION COMNUC FLOW CHART



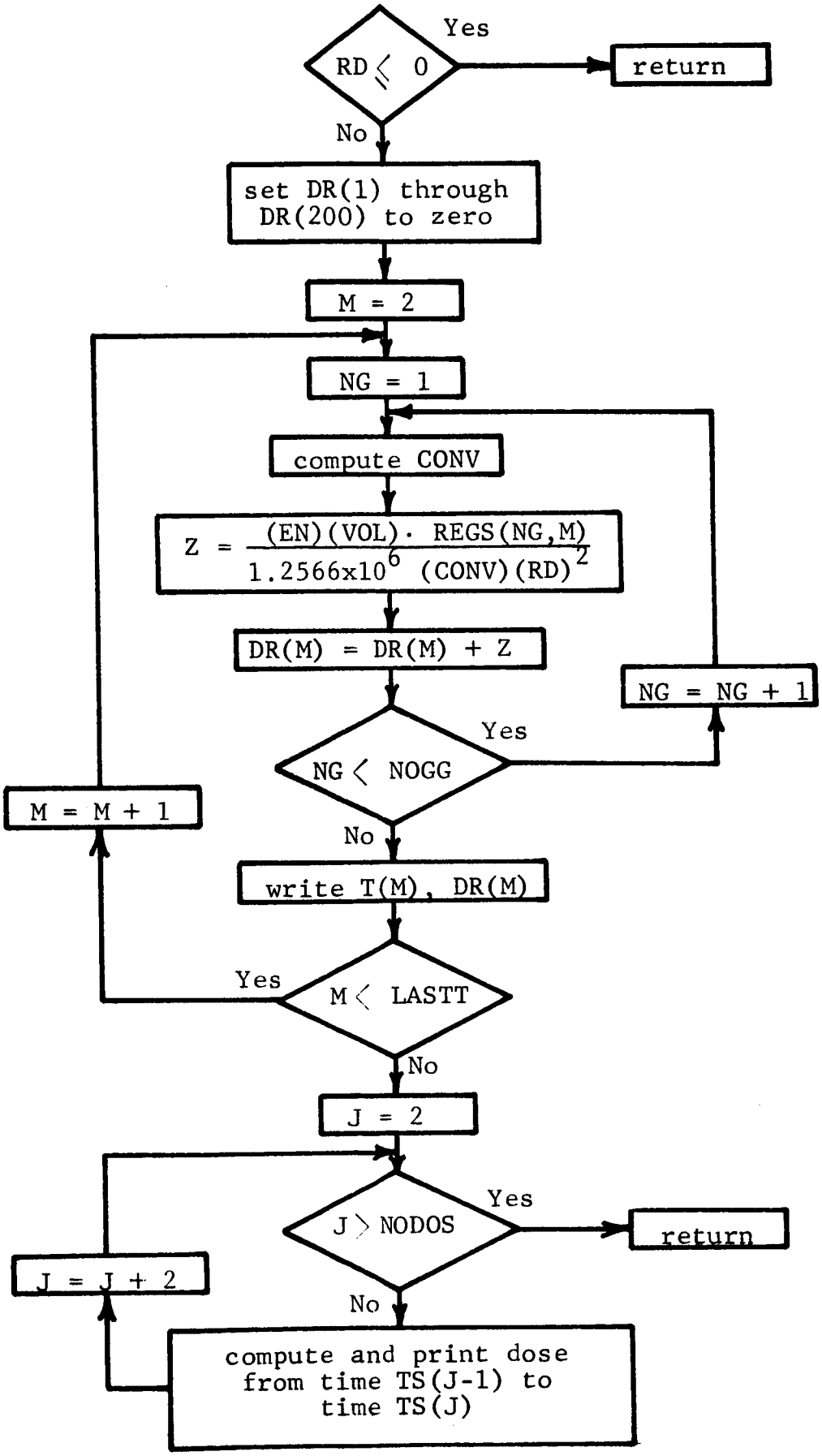
F. SUBROUTINE DOS

1. Purpose: To compute the gamma ray dose and dose rate at the distance RD from an activation source.
2. Arguments: RD,VOL,IR,NODOS,TS,REGS,NOGG, LASTT,EGG,T
3. Dimensioned Variables:

C(18)	E(18)	T(200)
D(25)	EGG(21)	TS(50)
DR(200)	REGS(20,200)	BX(12)

4. Common Variables:
   CP: BX,IC,LEAF
5. Called Subprograms: PAGE
6. Calling Subprograms: MAIN
7. Comments: CONV is a conversion factor from gamma ray energy flux (10^5 photons-MeV/cm^2-sec) to gamma ray absorbed dose rate (rad/hr) in tissue. It is computed from the conversion factor array C, which is ordered according to the photon energies listed in the array E. The arrays C and E are provided internally in this subprogram by data statements. Different units for the dose rate can be provided only by rewriting a few of the FORTRAN statements in this subroutine and changing the C and E arrays. The array D is not used and could be eliminated.

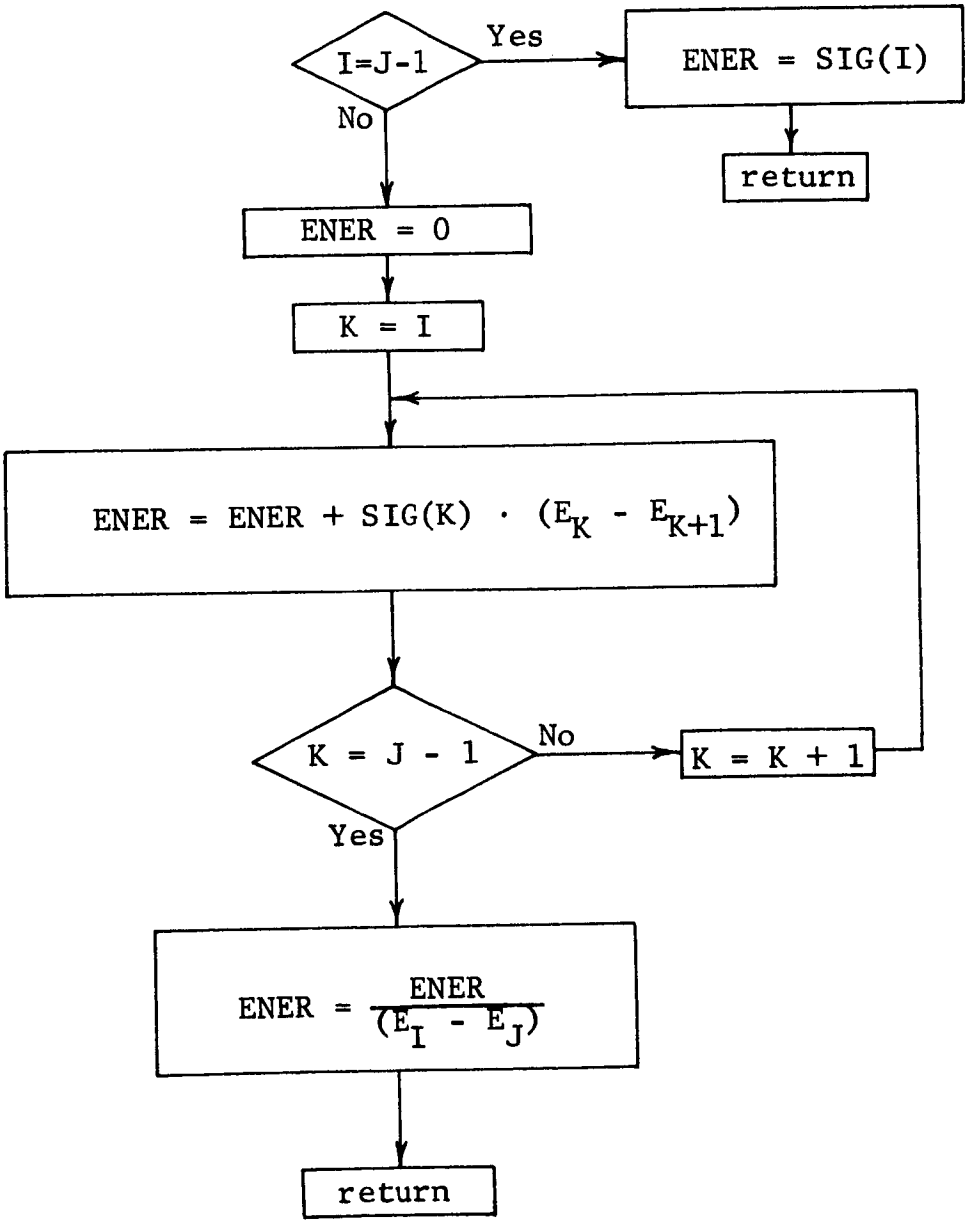
SUBROUTINE DOS FLOW CHART



G. Function ENER

1. Purpose: Averages the cross sections in the array SIG over a constant flux per unit energy to obtain group cross sections in the group structure specified by the program user.
2. Arguments: FELIM, SIG, I, J.
3. Dimensioned Variables: FELIM(44), SIG(43).
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: FS
7. Comments: SIG is averaged from energy group I to energy group J, inclusive. FELIM(K) is the lower energy limit of energy group K-1 and is indicated in the flow chart by E_K.

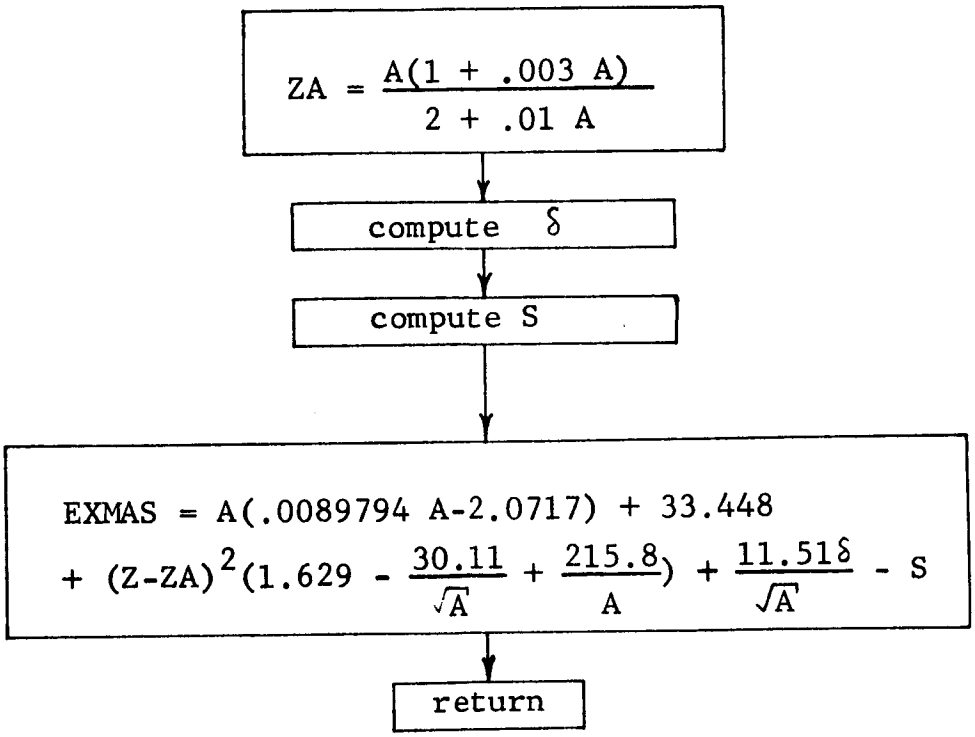
FUNCTION ENER FLOW CHART



H. Function EXMAS

1. Purpose: Computes mass excess using Wing-Fong formula for nucleus specified by Z and A.
2. Arguments: Z,A.
3. Dimensioned Variables: B(5), C(5), D(5), E(5).
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: XSCAL
7. Comments: Refer to Volume II for notation.

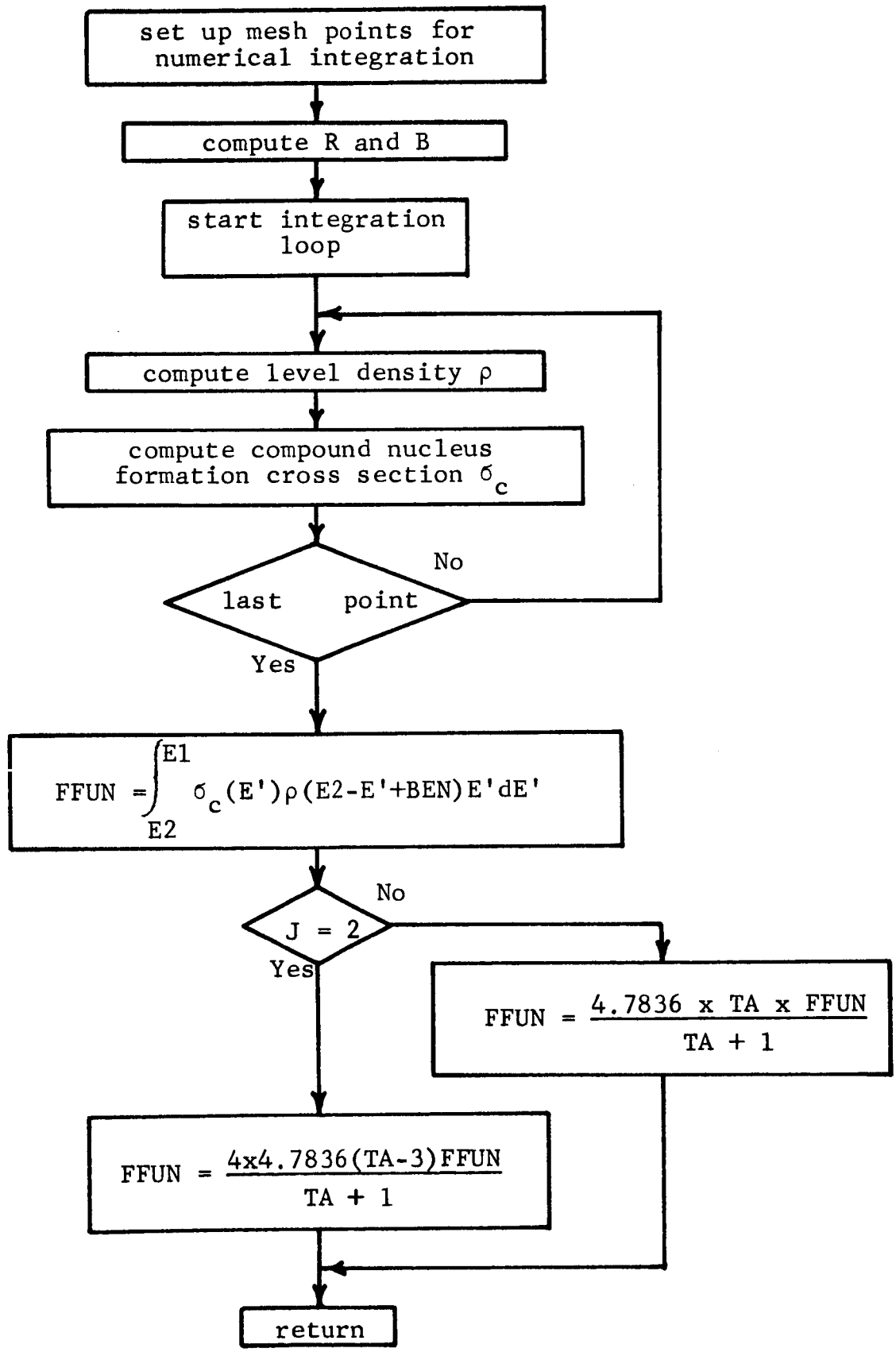
FUNCTION EXMAS FLOW CHART



I. Function FFUN

1. Purpose: Performs integration of product of compound nucleus cross section and level density over energy as required by equations such as equation (46) of Volume II.
2. Arguments: J, TZ, TA, E2, E1, C, A, BEN.
3. Dimensioned Variables: None.
4. Common Variables: None.
5. Called Subprograms: ALCOM, COMNUC, PROCOM.
6. Calling Subprograms: SIGCAL.
7. Comments: J indicates the type of compound nucleus formation cross section required, TZ and TA the Z and A of the target isotope, E2 and E1 the upper and lower limits of integration, C and A the level density parameters, BEN the neutron binding energy.

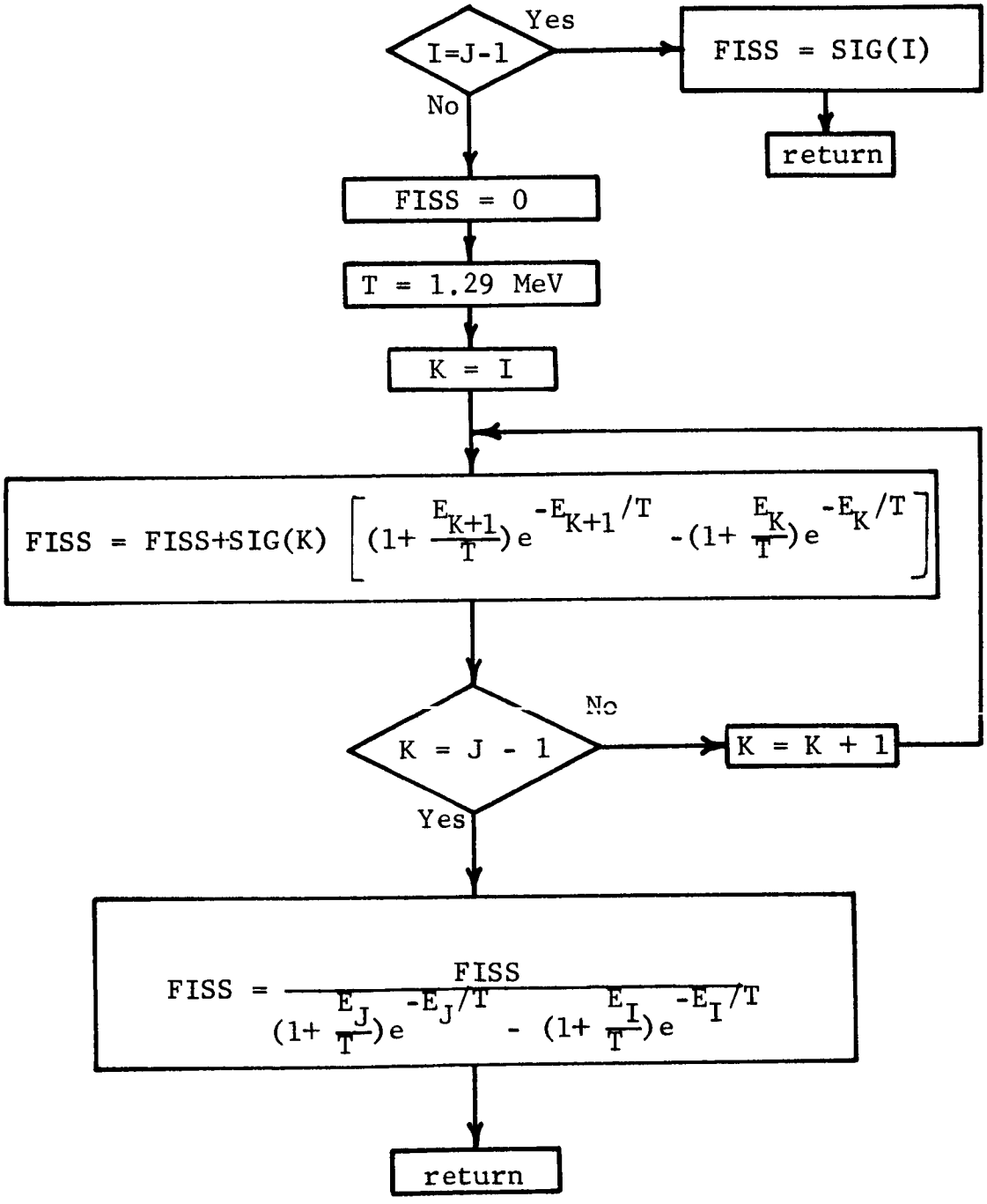
FUNCTION FFUN FLOW CHART



J. Function FISS

1. Purpose: Averages the cross sections in the array SIG over a fission flux spectrum to obtain cross sections in the group structure specified by the program user.
2. Arguments: FELIM, SIG, I, J.
3. Dimensioned Variables: FELIM(44), SIG(43).
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: FS.
7. Comments: SIG is averaged from energy group I to energy group J, inclusive. FELIM(K) is the lower energy limit of energy group K-1 and is indicated in the flow chart by E_K.

FUNCTION FISS FLOW CHART



K. Function FS

1. Purpose: Integrates the flux times the group cross sections over energy to obtain reaction rates.
2. Arguments: Z, A, INRE, LKEY, ISO.
3. Dimensioned Variables:

		NLIM(43)
BX(12)	FLXIN(43)	NZ(10)
ELIM(43)	GSIG(43)	SIG(43)
FELIM(44)	NA(10)	VFAC(10)

4. Common Variables:

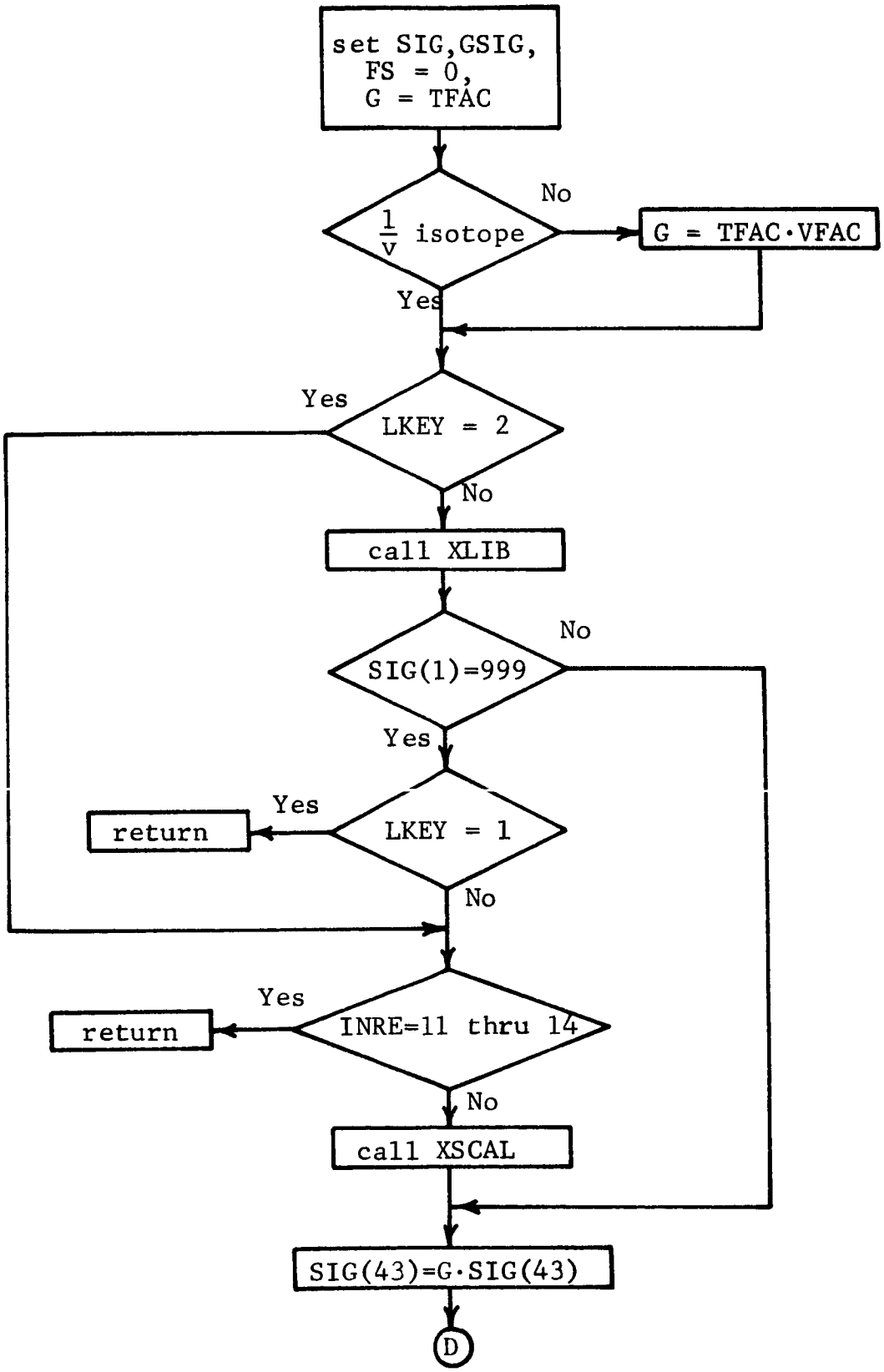
   CF: ELIM, FELIM, FLXIN, IFLX, IWT, NA, NOBG, NONV, NZ, TFAC, VFAC.

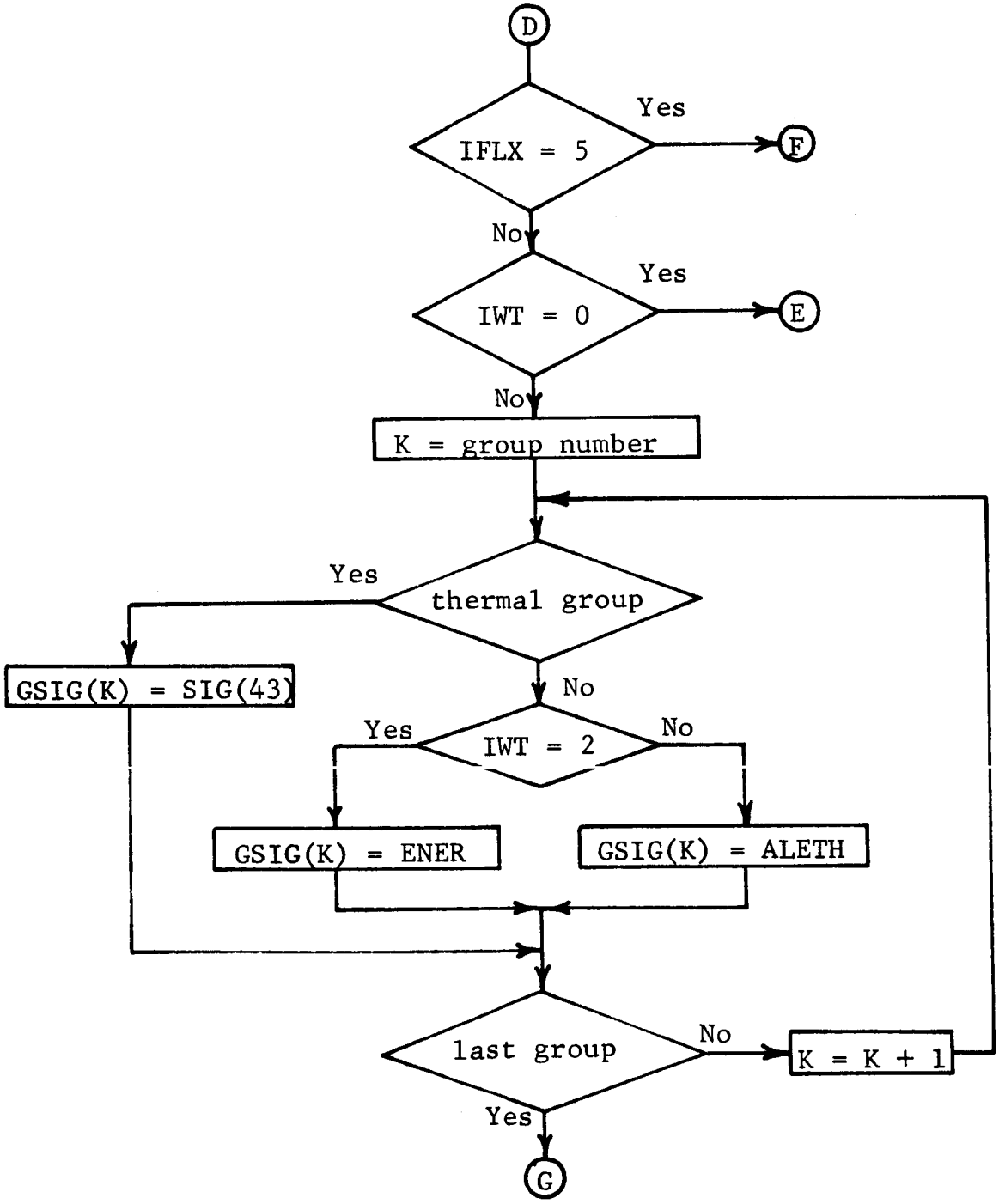
   CP: BX, IC, LEAF.

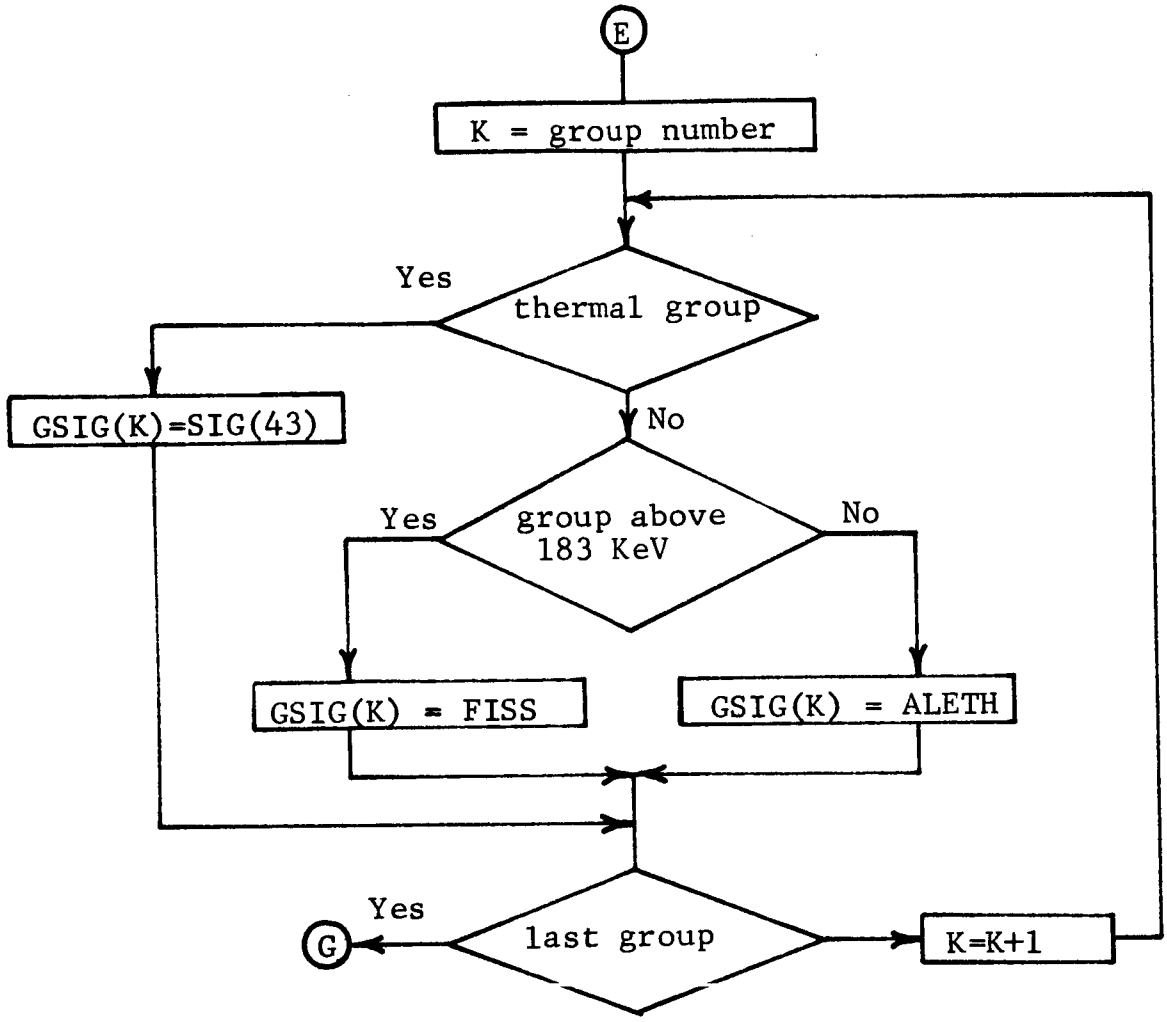
   CQ: NLIM.

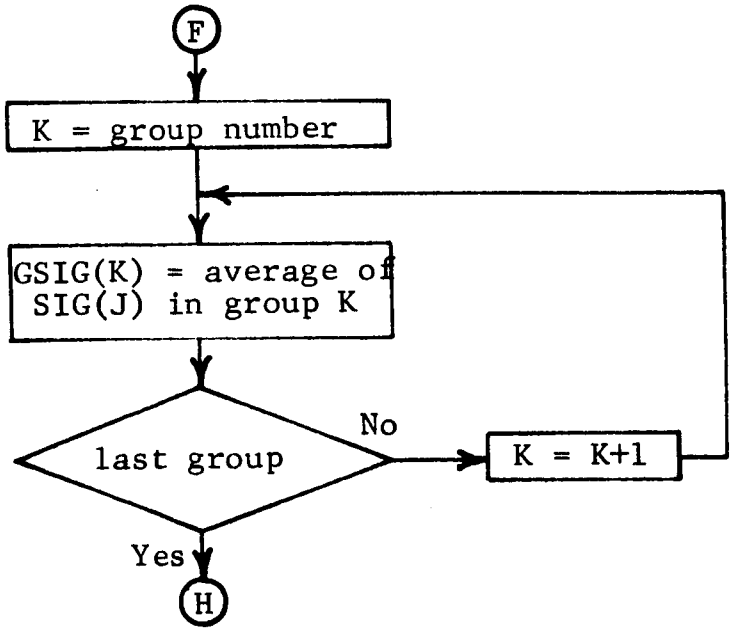
5. Called Subprograms: ALETH, ENER, FISS, PAGE, XLIB, XSCAL.
6. Calling Subprograms: ISOCON
7. Comments: Z and A identify the isotope for which the reaction rate is desired, INRE the type of reaction, LKEY the cross section option, ISO the isotopic index of the material. The array SIG is the 43-group cross section set as obtained from the library or the input. The array GSIG is the cross section set in the group structure specified by the program user.

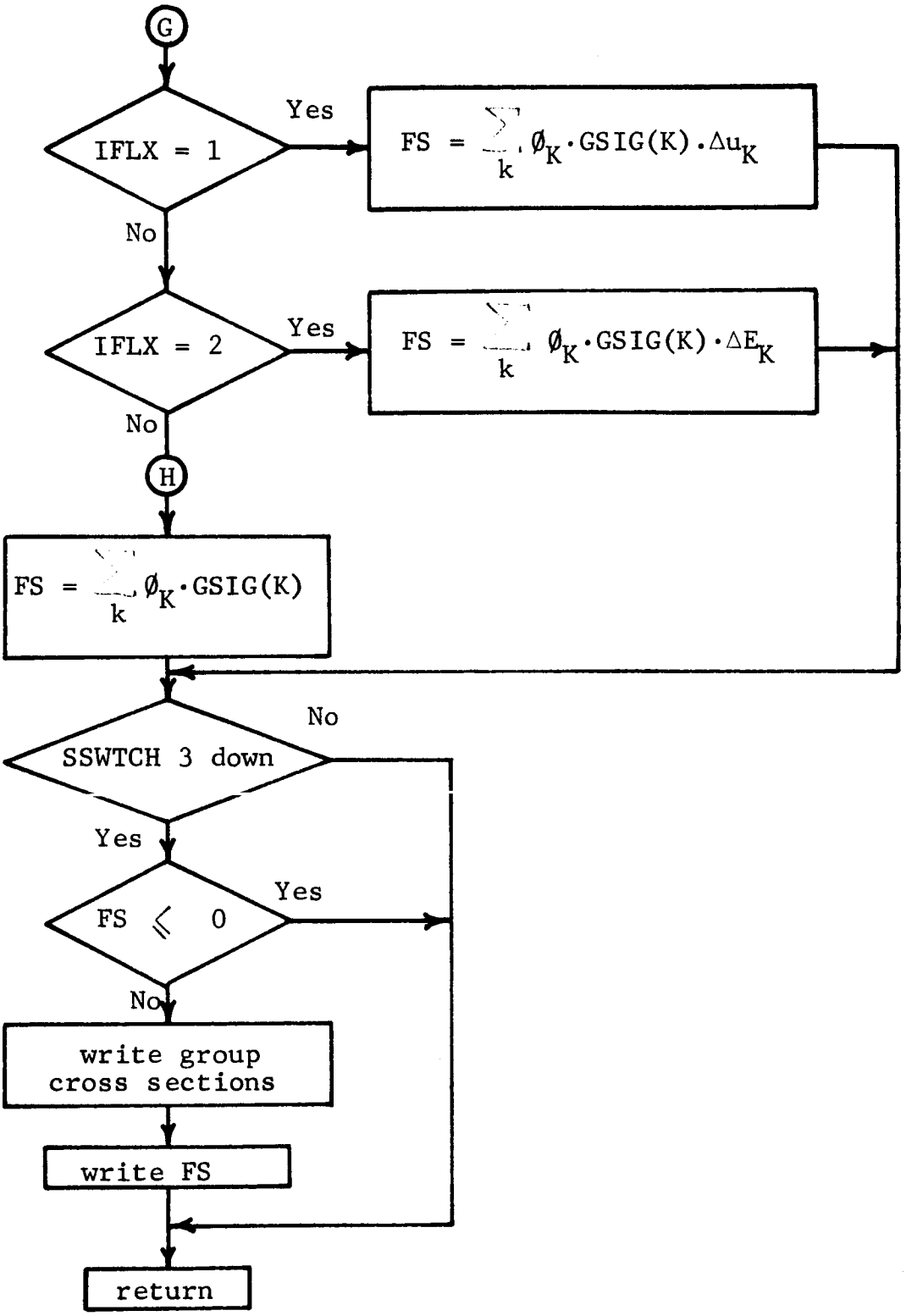
FUNCTION FS FLOW CHART




D
IFLX = 5
Yes
F
No
IWT = 0
Yes
E
No
K = group number
thermal group
Yes
GSIG(K) = SIG(43)
No
IWT = 2
Yes
No
GSIG(K) = ENER
GSIG(K) = ALETH
last group
No
K = K + 1
Yes
G


E
K = group number
Yes
thermal group
No
GSIG(K)=SIG(43)
Yes
group above
183 KeV
No
GSIG(K) = FISS
GSIG(K) = ALETH
Yes
G
last group
K=K+1


F
K = group number
GSIG(K) = average of
SIG(J) in group K
No
last group
K = K+1
Yes
H


G
IFLX = 1
Yes
FS = $\sum_k \phi_K \cdot GSIG(K) \cdot \Delta u_K$
No
IFLX = 2
Yes
FS = $\sum_k \phi_K \cdot GSIG(K) \cdot \Delta E_K$
No
H
FS = $\sum_k \phi_K \cdot GSIG(K)$
SSWTCH 3 down
No
Yes
FS $\leqslant$ 0
Yes
No
write group cross sections
write FS
return

L. SUBROUTINE GROUPS

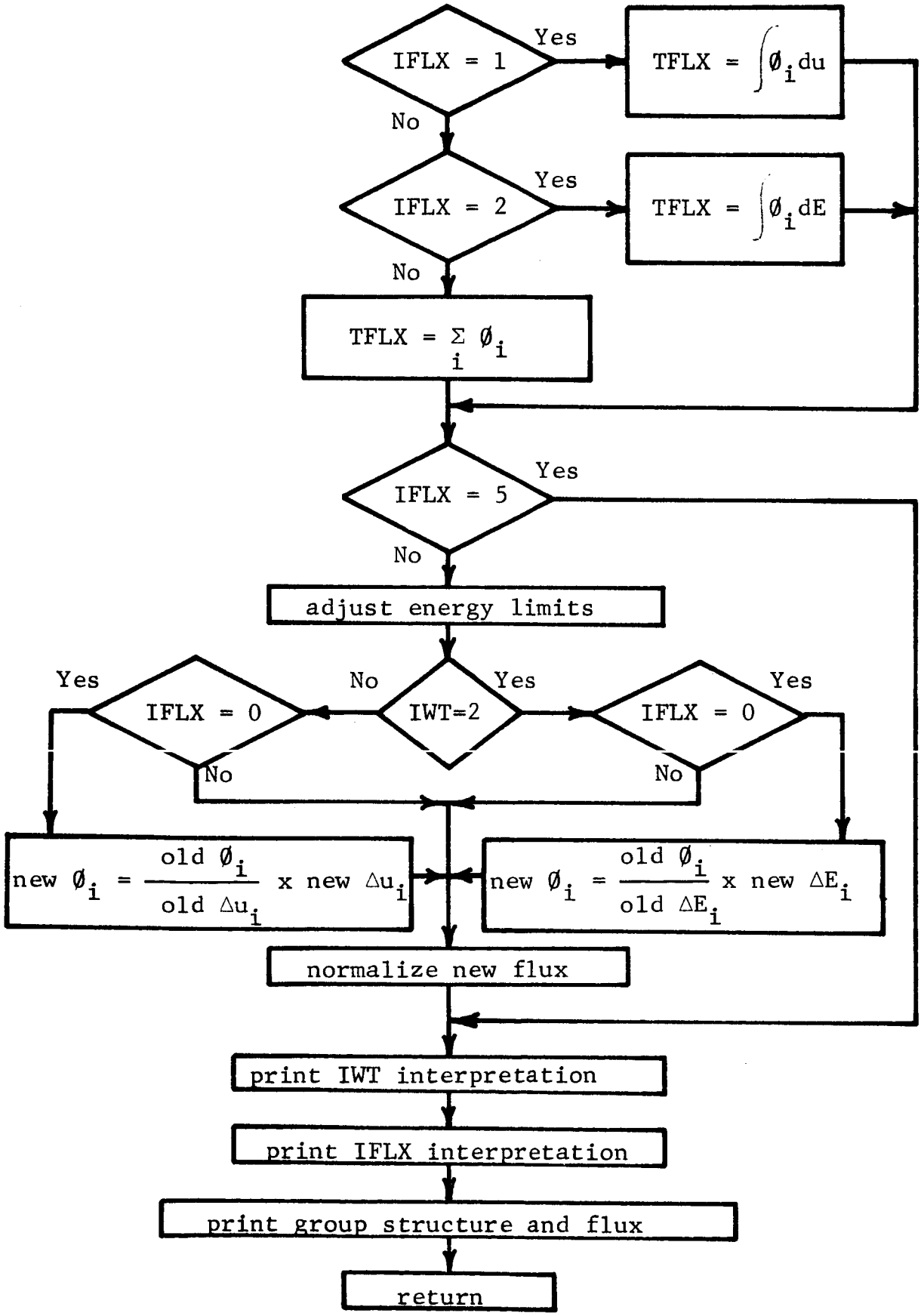
1. Purpose: Computes total neutron flux integrated over energy, adjusts the input energy limits and group fluxes to be consistent with the NAP Cross Section Library, and prints the resulting group energy limits and group fluxes.
2. Arguments: TFLX, which is the total neutron flux integrated over energy.
3. Dimensioned Variables:

BX(12)	FELIM(44)	NZ(10)
ELIM(43)	FLXIN(43)	VFAC(10)
	NA(10)	

4. Common Variables:

   CF: ELIM, FELIM, FLXIN, IFLX, IWT, NA, NOBG, NONV, NZ, TFAC, VFAC.
5. Called Subprograms: PAGE.
6. Calling Subprograms: MAIN.
7. Comments: None.

SUBROUTINE GROUPS FLOW CHART



M. SUBROUTINE ISOCON

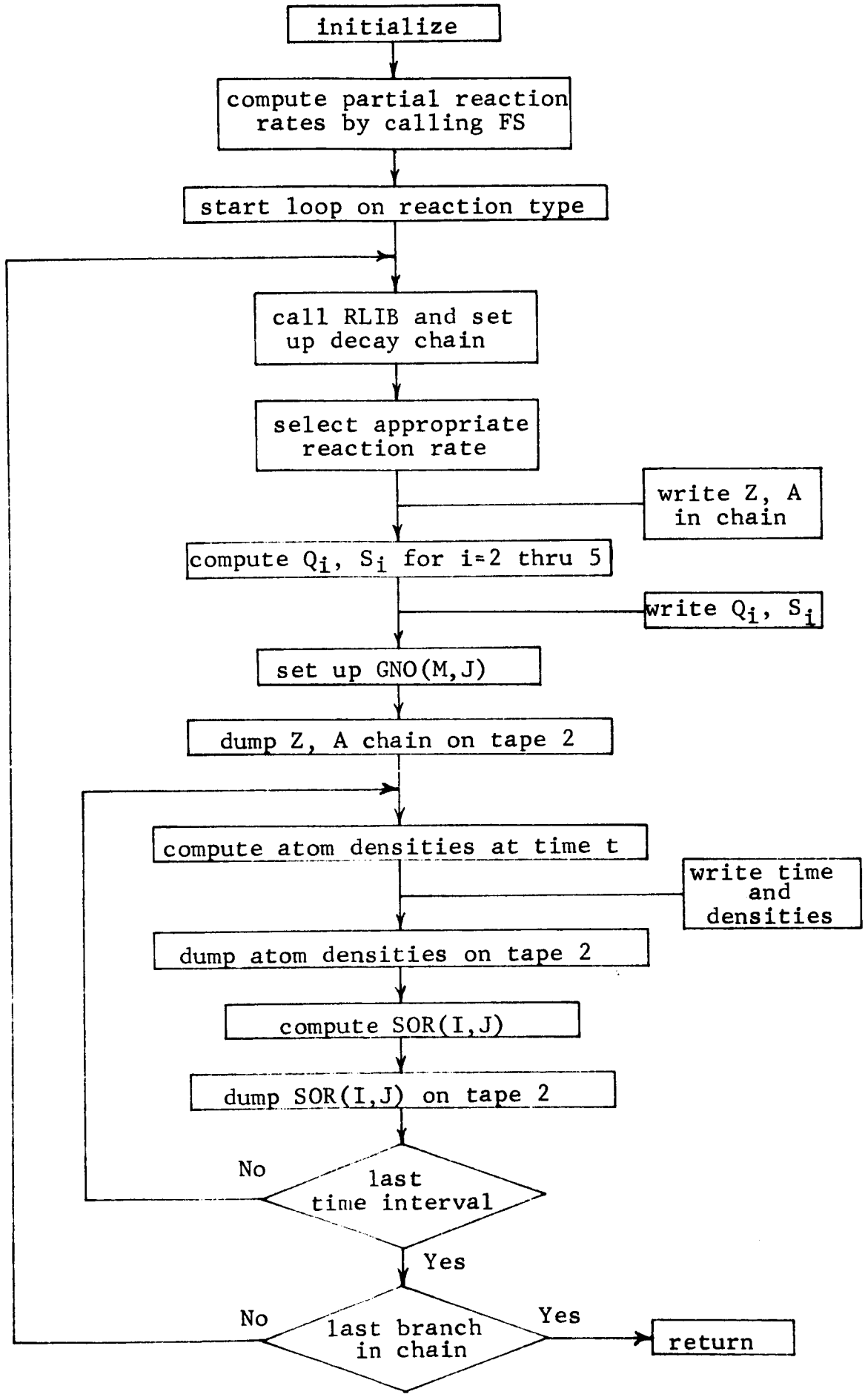
1. Purpose: Sets up isotopic decay chains, computes isotopic densities and gamma source strengths as function of time.
2. Arguments: ZISO, AISO, ATD, LKEY, KM, ISO.
3. Dimensioned Variables:

A(5)	EGG(21)	RATE(5)
BX(12)	E1(5,7)	SOR(20,5)
BR(6)	F1(5,7)	T(200)
CS(8)	GAM(21)	Z(5)
D(5)	GNO(20,5)	
DENS(5)	POWR(200)	

4. Common Variables:

   BS: EGG, LASTT, NOGG, POWR, SOR, T
   CP: BX, IC, LEAF
5. Called Subprograms: FS, PAGE, RLIB
6. Calling Subprograms: MAIN
7. Comments: The write statements indicated on the flow chart are performed only if sense switch 3 is depressed. GNO(M,J) is the number of gammas per decay from chain member J in gamma energy group M. SOR(I,J) is the number of gammas per unit volume per unit time emitted in energy group I from chain member J, and is time-dependent. ZISO and AISO are the Z and A for the chain parent, ATD is its initial atom density, and LKEY is its cross section option. KM is a running counter of the number of chains for which data are dumped on tape 2. ISO is the parent chain member's isotopic index in the region.

SUBROUTINE ISOCON FLOW CHART



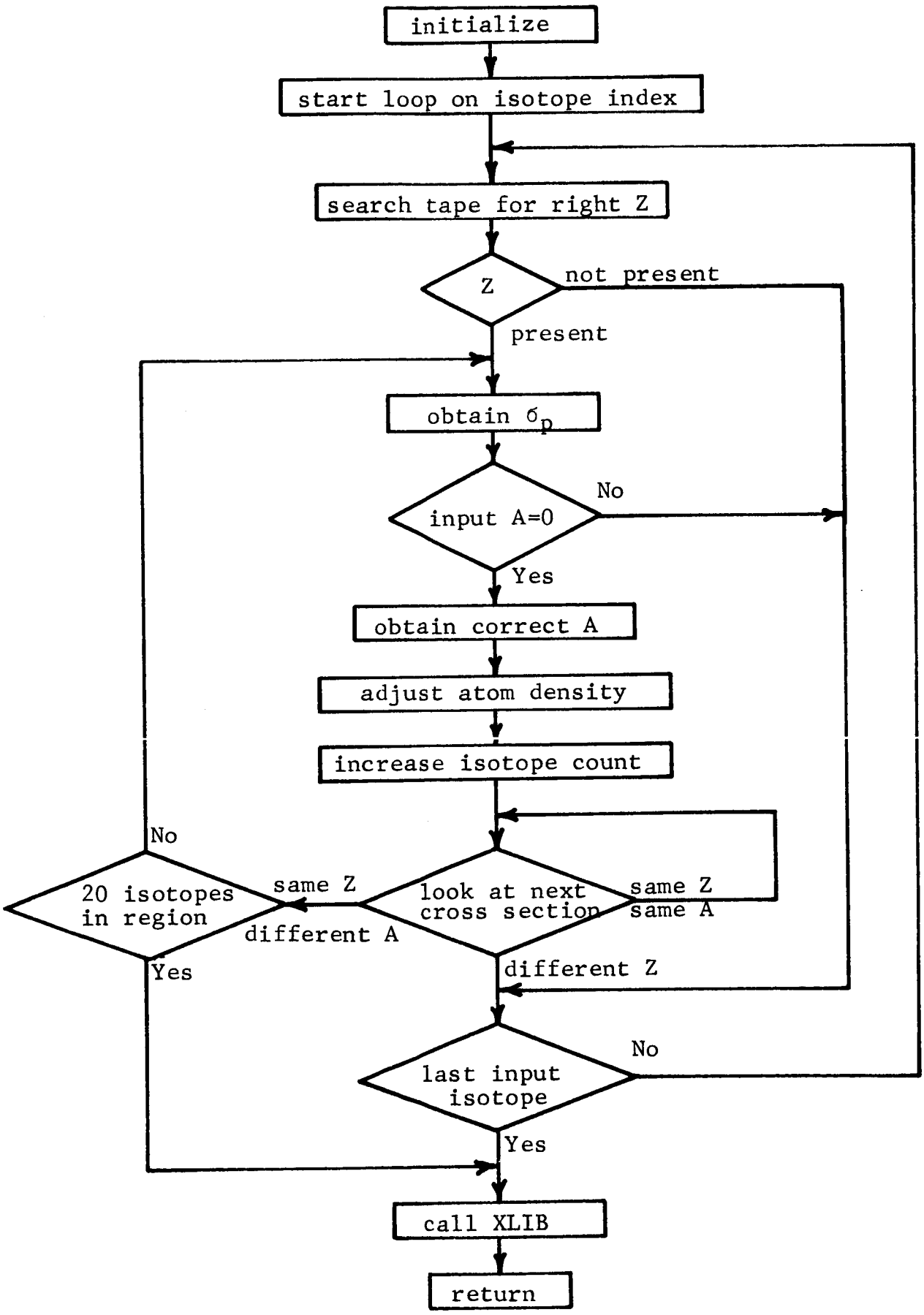
N. Subroutine NATDEN

1. Purpose: Obtains isotopic composition of naturally occurring isotopes and potential scattering cross section of all isotopes in region from NAP Cross Section Library.
2. Arguments: LKEY
3. Dimensioned Variables:

AO(20)	IZ(20)	SPOT(20)
D(20)	LK(20)	X(250)
DEN(20)	LKEY(20)	ZO(20)
IA(20)	SIG(43)	BX(12)

4. Common Variables:
   CP: BX, IC, LEAF
   R: DEN, IA, ISOR, IZ, R, SPOT, TEMP
5. Called Subprograms: PAGE, XLIB
6. Calling Subprograms: MAIN
7. Comments: LKEY indicates whether the material cross sections searched for are input by the program user or are in the Cross Section Library. XLIB is called to initiate reading of data from the Cross Section Library into allocated core storage.

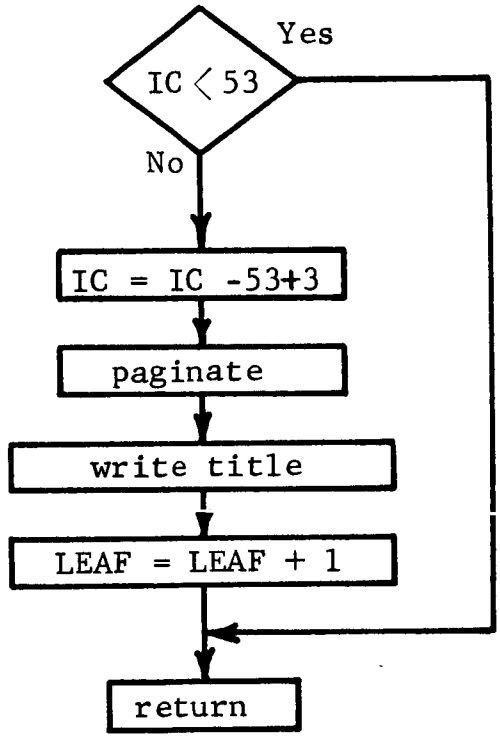
SUBROUTINE NATDEN FLOW CHART



0. Subroutine PAGE

1. Purpose: To provide pagination of output data.
2. Arguments: None.
3. Dimensioned Variables: BX(12)
4. Common Variables:
   CP: BX, IC, LEAF.
5. Called Subprograms: None.
6. Calling Subprograms: MAIN, GROUPS, SHIELD, SETUP, NATDEN, ISOCON, FS, RESINT, XSCAL.
7. Comments: The problem title is contained in the array BX, which is printed at the top of each page of output. IC is a line counter, LEAF is a page counter. Use of sense switch settings to provide optional output will usually result in some output pages without title or page number.

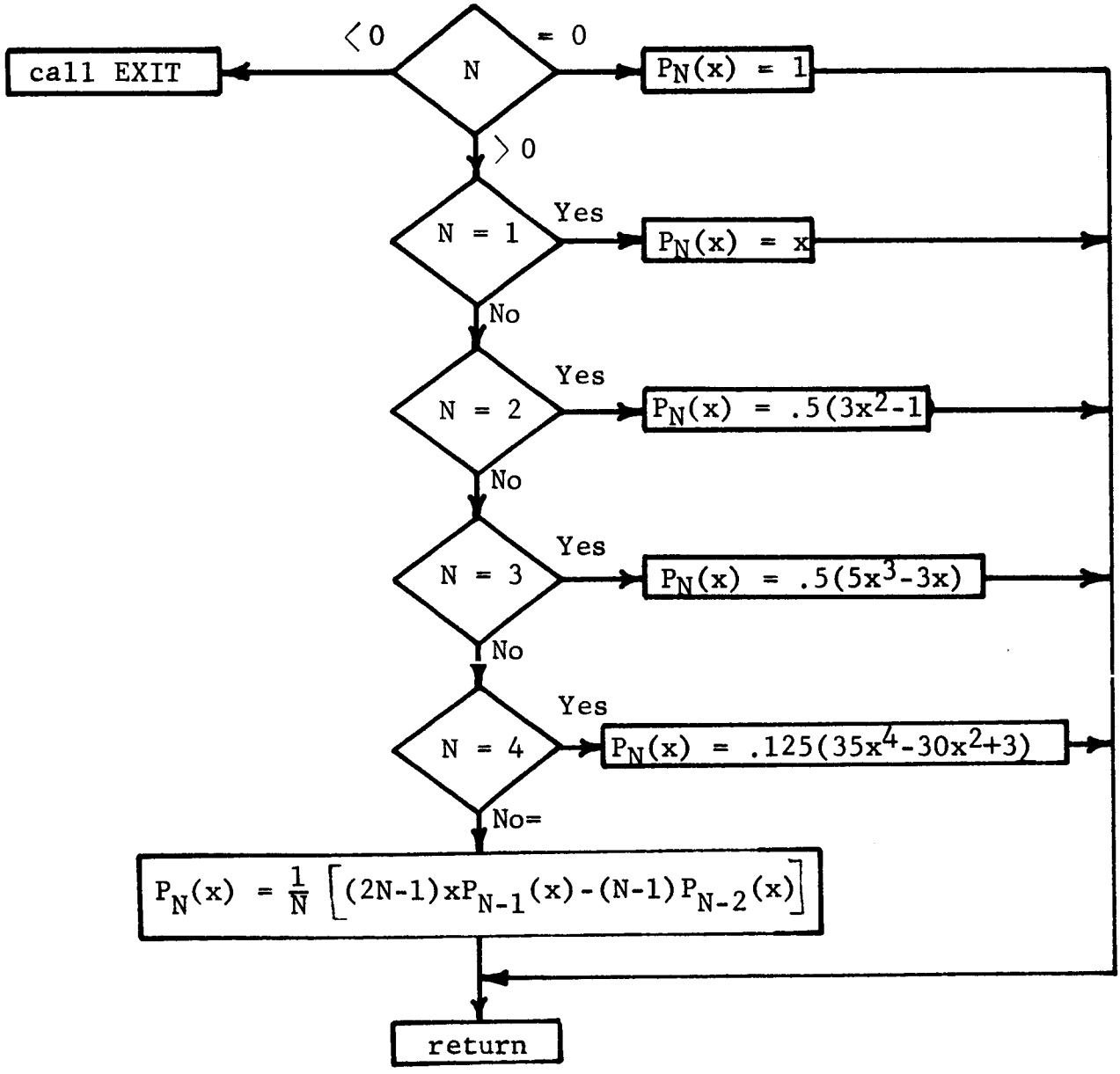
SUBROUTINE PAGE FLOW CHART



P. Function POLY

1. Purpose: Computes Legendre polynomials required by subroutine SETUP.
2. Arguments: N,X
3. Dimensioned Variables: None.
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: SETUP
7. Comments: N is the order of the Legendre polynomial required, X is the argument of the polynomial.

FUNCTION POLY FLOW CHART



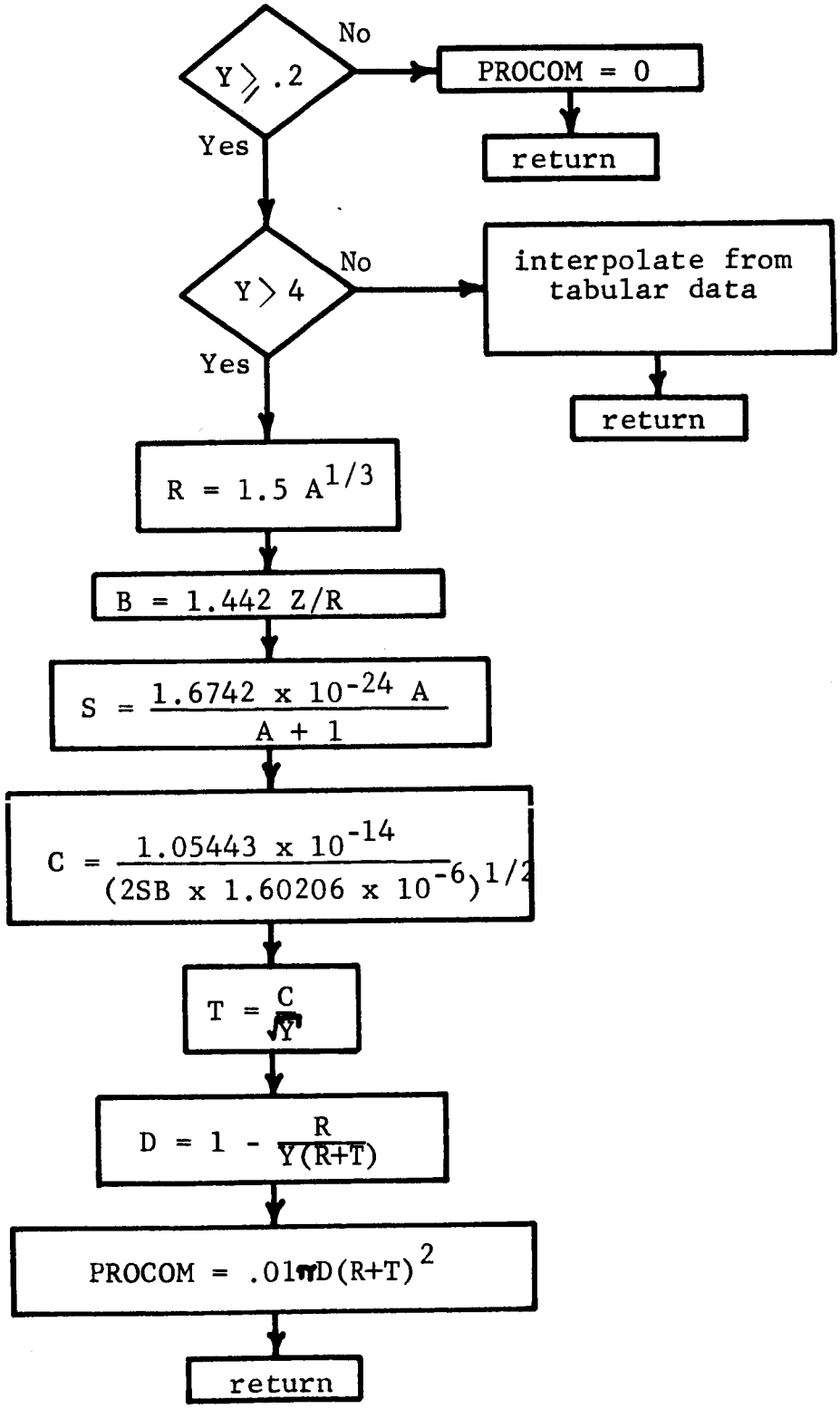
Q. Function PROCOM

1. Purpose: Obtain compound nucleus formation cross section of nucleus, specified by Z and A, due to incident protons of energy specified by parameter Y.
2. Arguments: Y, Z, A.
3. Dimensioned Variables:

FLIX(4)	FFLIST(4)	YL(39)
FLIST(39,9)	XLIST(4)	ZL(9)

4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: FFUN.
7. Comments: Tabular data is contained in the array FLIST as a function of YL and ZL. See Volume II for other notation.

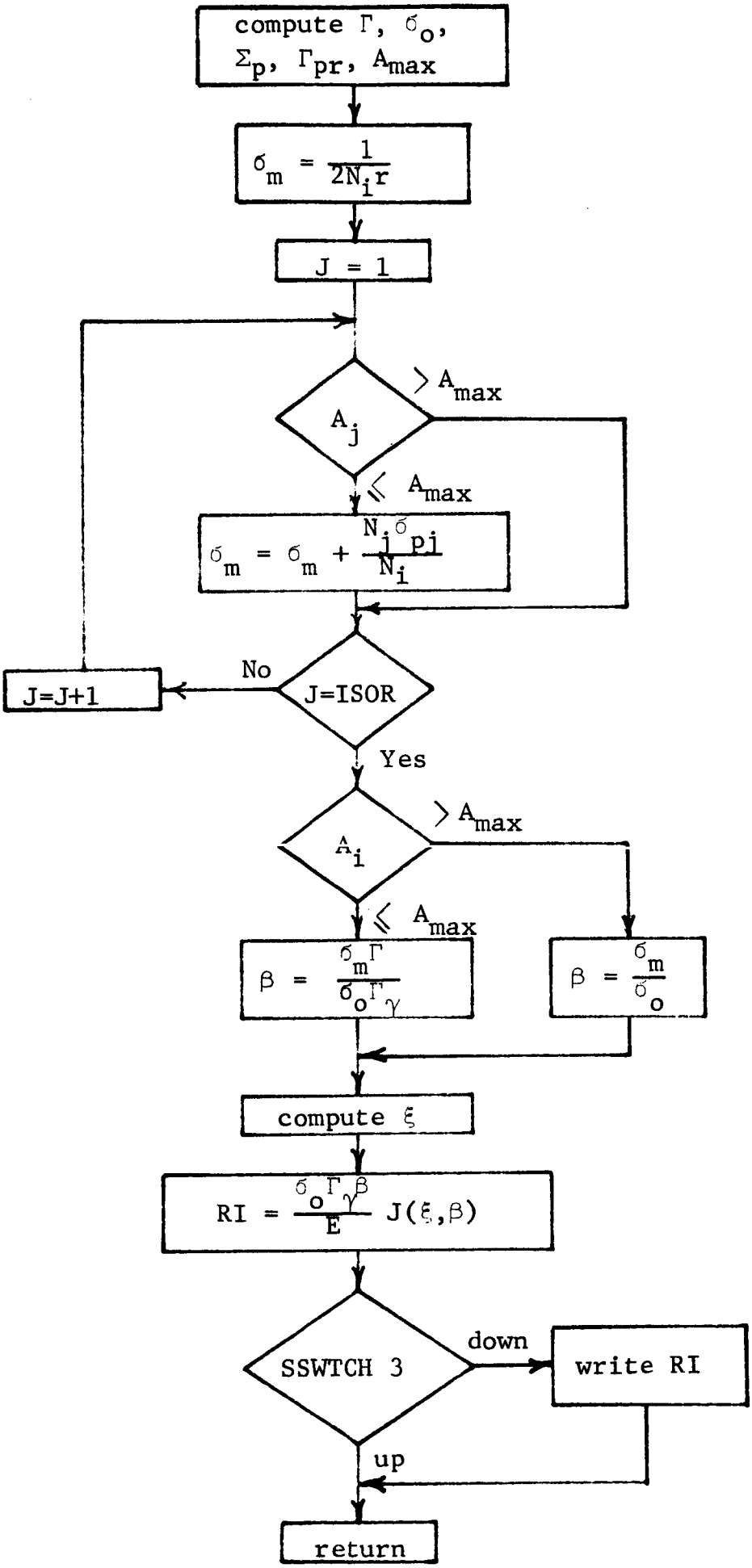
FUNCTION PROCOM FLOW CHART



R. Subroutine RESINT

1. Purpose: Computes effective resonance integrals from resonance parameters.
2. Arguments: E, G, GAMG, GAMN, GAMF, RI, I.
3. Dimensioned Variables: BX(12), IA(20), DEN(20), SPOT(20).
4. Common Variables:
   CP: BX, IC, LEAF
   R: IA, ISOR, IZ, DEN, R, SPOT, TEMP
5. Called Subprograms: AJ, PAGE
6. Calling Subprograms: XLIB
7. Comments: See Volume II for notation.

SUBROUTINE RESINT FLOW CHART



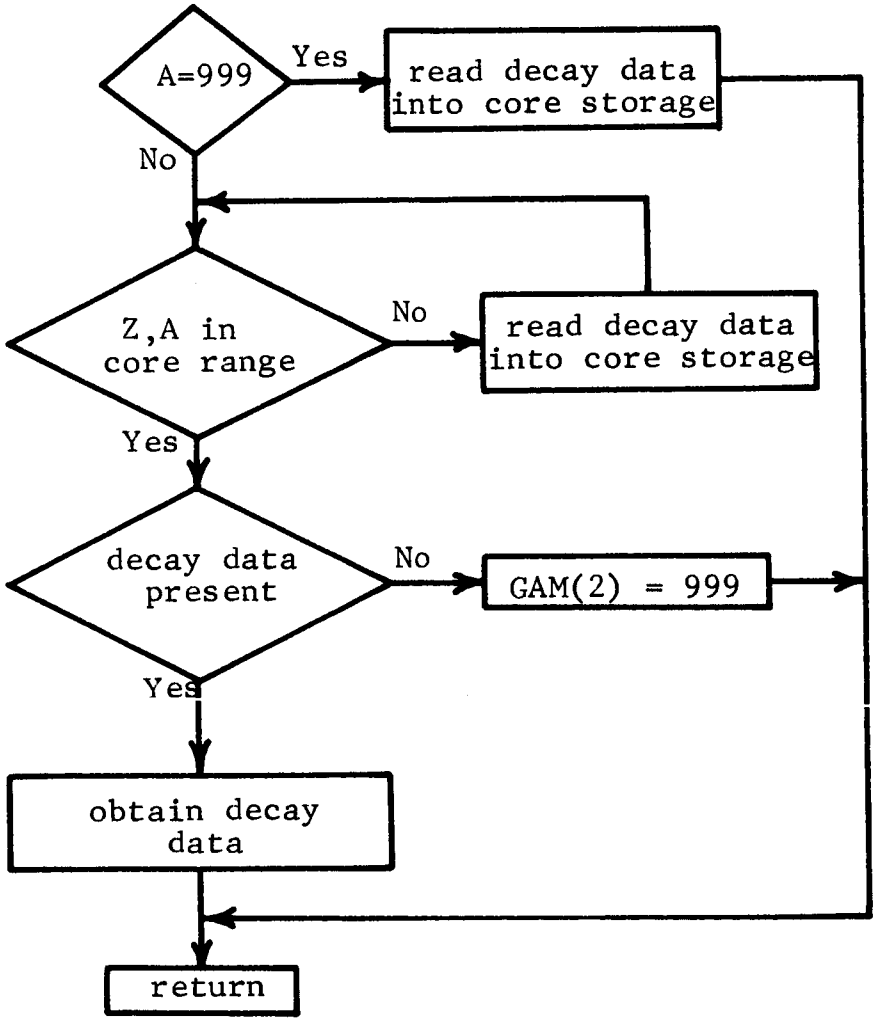
S. Subroutine REVLIB

1. Purpose: To provide for future internally calculated cross sections to the NAP Cross Section Library.
2. Arguments: Z, A, IN, SIG.
3. Dimensioned Variables: SIG(43).
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: XSCAL.
7. Comments: This subroutine was never programmed and consists solely of a return statement. Therefore, no flow chart is included here.

T. Subroutine RLIB

1. Purpose: Reads decay data from FORTRAN tape unit 10.
2. Arguments: J3, Z, A, GAM.
3. Dimensioned Variables: GAM(21), X(252)
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: MAIN, ISOCON.
7. Comments: Z and A identify the isotope for which decay data are required. J3 is a decay chain branch indicator. The decay data are transferred to ISOCON by the array GAM. If the decay data are not found, GAM(2) is set equal to 999. To avoid excessive tape reading time, twelve sets of decay data are stored in core at any one time. The initial reading of this data is accomplished by setting A equal to 999.

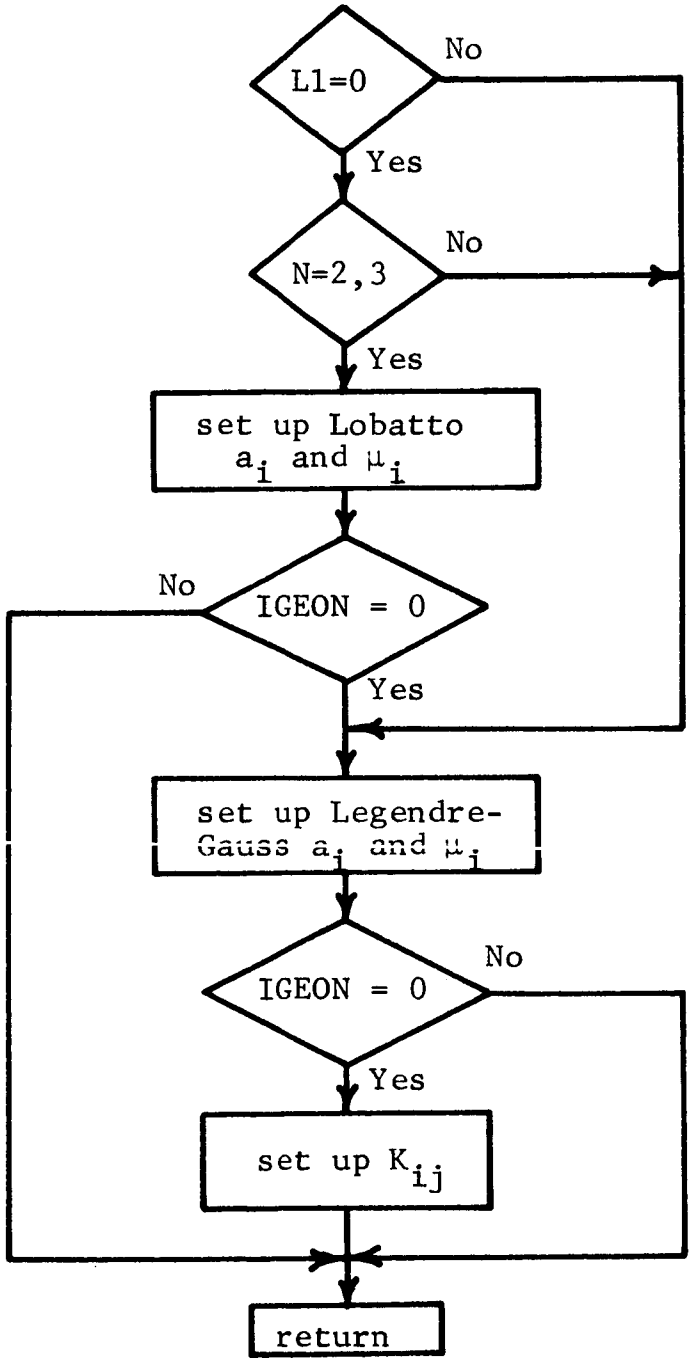
SUBROUTINE RLIB FLOW CHART



U. Subroutine SETUP

1. Purpose: Set up angular ordinates and quadrature weights required by SHIELD.
2. Arguments: IGEON, NOANG, L1
3. Dimensioned Variables:
   A(10), AK(10,10), AL(40), AMU(10), BX(12)
4. Common Variables:
   CP: BX, IC, LEAF
   SH: A, AK, AMU
   L: AL
5. Called Subprograms: BLOCK DATA, POLY, PAGE
6. Calling Subprograms: SHIELD
7. Comments: The array A consists of the quadrature weights, AMU of the angular ordinates, AL of the set of possible weights and ordinates, and AK of the K matrix defined by equations (25) and (26) of Volume II. L1 designates the desired quadrature scheme. The subprogram BLOCK DATA is simply a single data instruction inputing the set of possible ordinates and weights.

SUBROUTINE SETUP FLOW CHART



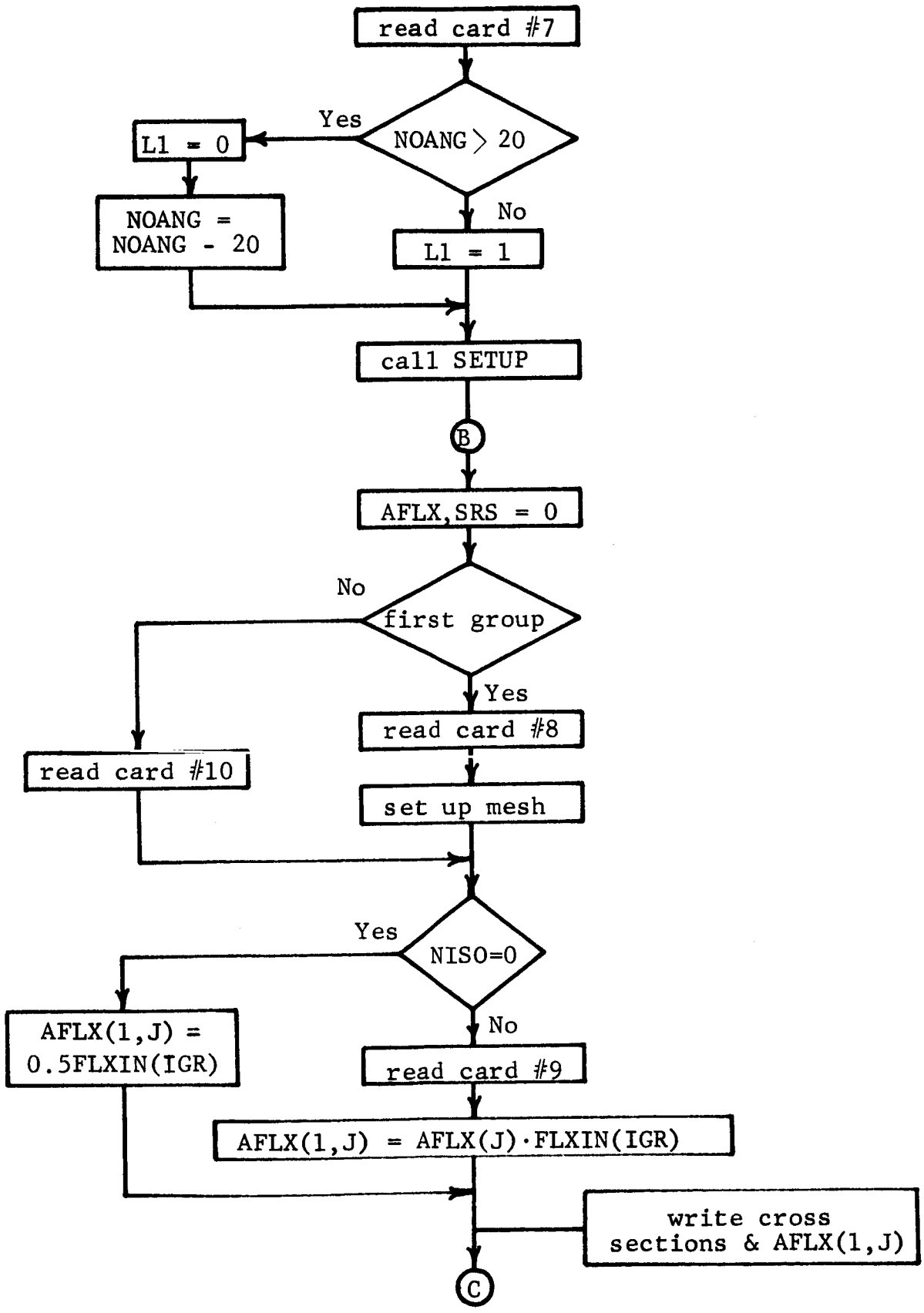
V. Subroutine SHIELD

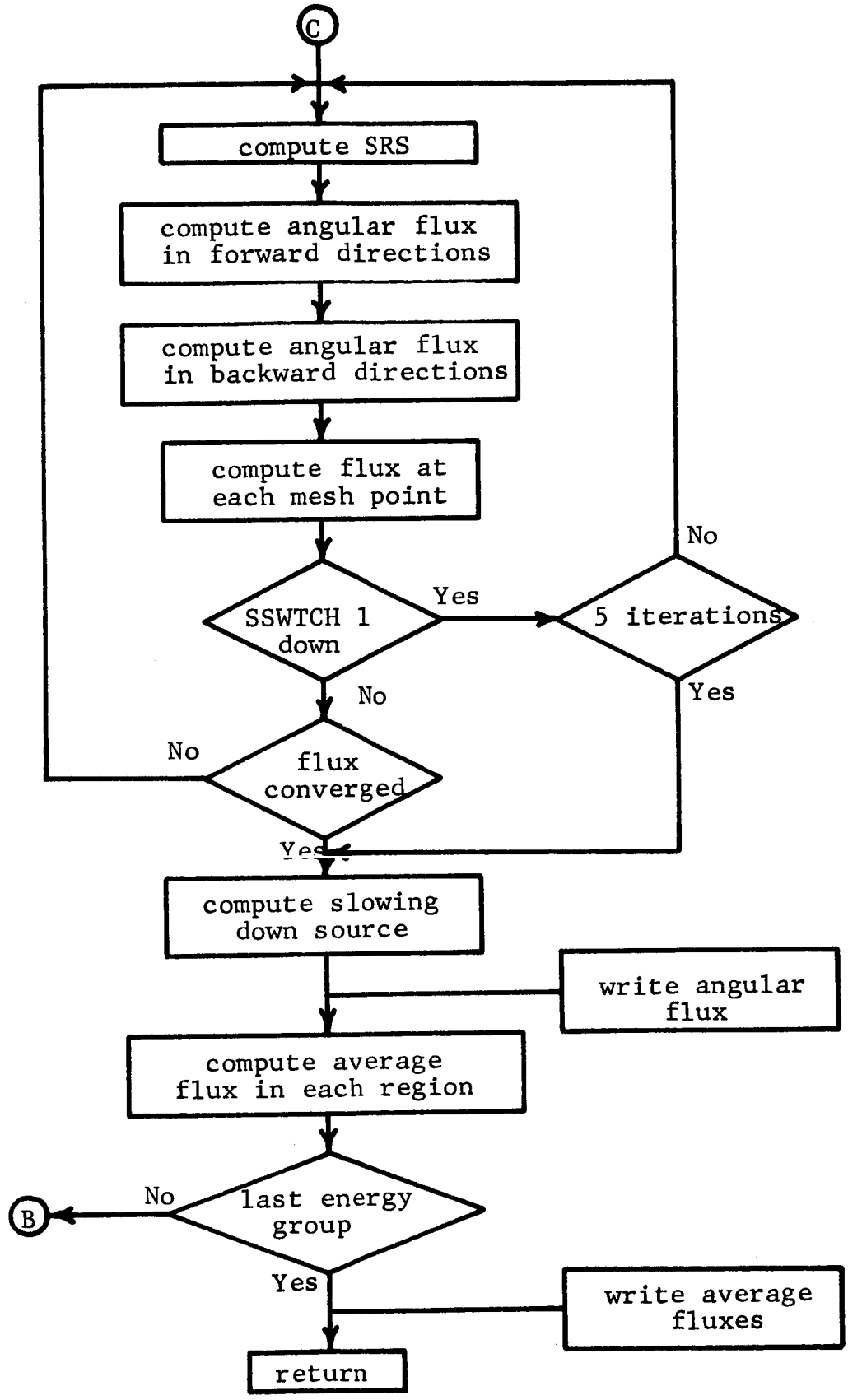
1. Purpose: Perform neutron transport calculation, set up average regional group fluxes, and print flux.
2. Arguments: NOBG, NOREG, FLXIN, FLX, NISO
3. Dimensioned Variables:

A(10)	BX(12)	SIGS(20)
AFLX(100,10)	BUM(10)	SIGSL(20)
AK(10,10)	DX(20)	SIGT(20)
AMU(10)	FLX(860)	SLD(100)
ANINT(20)	FLXIN(43)	SRS(100,10)
BLFX(100)	NINT(20)	X(101)

4. Common Variables:
   CP: BX, IC, LEAF
   SH: A, AK, AMU
5. Called Subprograms: PAGE, SETUP
6. Calling Subprograms: MAIN
7. Comments: AFLX(I,J) is the angular flux at the mesh point I and angular ordinate J. The quantity AFLX(J) referred to in the input data is really AFLX(1,J). SRS(I,J) is the source term at mesh point I and angular ordinate J defined by equation (16) of Volume II. FLX is a one-dimensional array which specifies the average flux in each group and each region. The array is ordered by increasing region number. Within each region, the array is ordered by decreasing neutron energy. Output is dependent upon the value of input variable IOUT.

SUBROUTINE SHIELD FLOW CHART

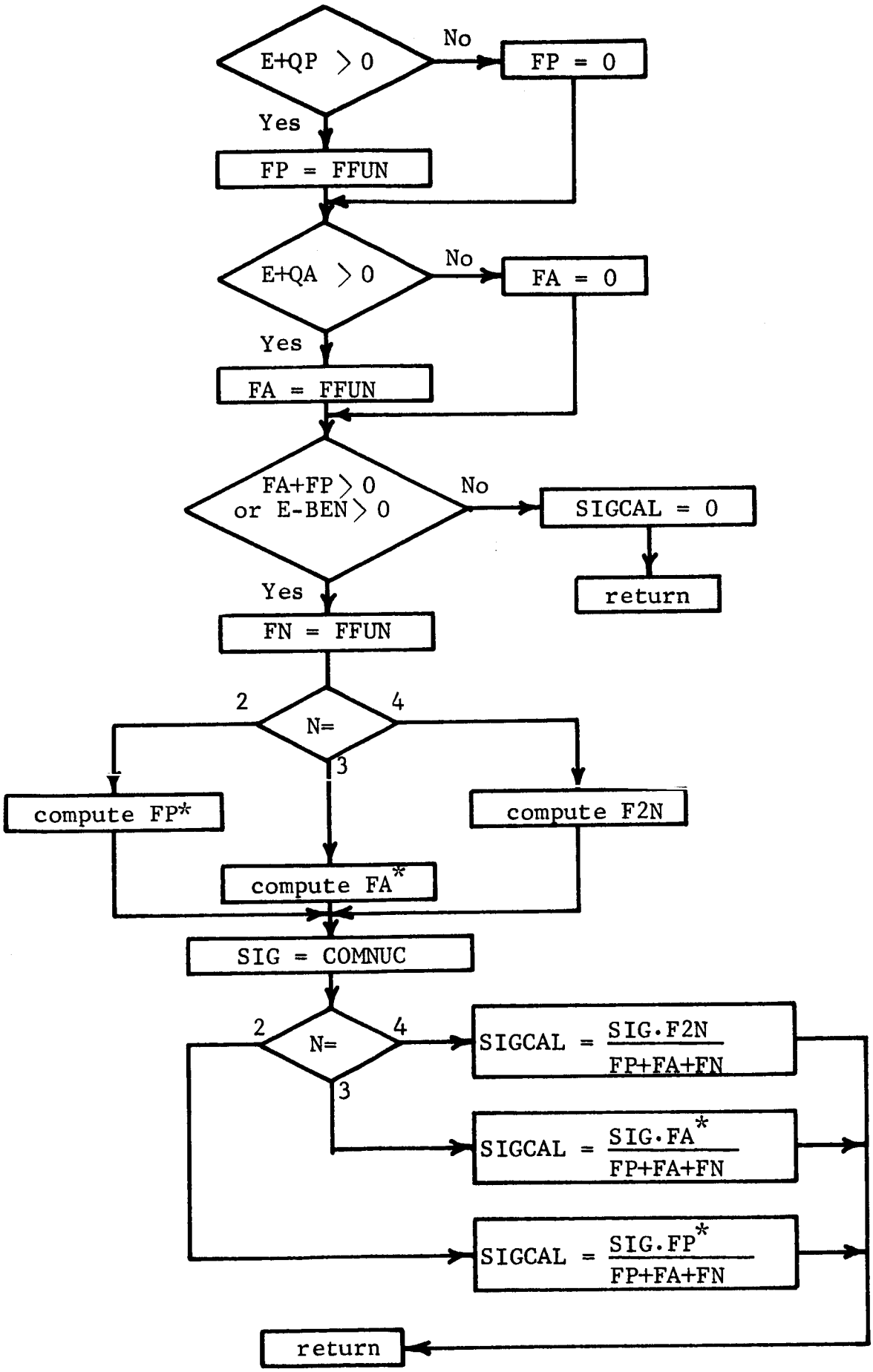



C
compute SRS
compute angular flux in forward directions
compute angular flux in backward directions
compute flux at each mesh point
SSWTCH 1 down
Yes
5 iterations
No
Yes
No
flux converged
No
Yes
compute slowing down source
write angular flux
compute average flux in each region
B
No
last energy group
Yes
write average fluxes
return

W. Function SIGCAL

1. Purpose: Computes $\sigma(n,\alpha)$, $\sigma(n,p)$, or $\sigma(n,2n)$ at energy E given level density parameters, binding energies, and Q values.
2. Arguments: Z, A, E, N.
3. Dimensioned Variables: None.
4. Common Variables:
   CX: AA, AN, AP, BEA, BEN, BEP, CA, CN, CP, QA, QN, QP.
5. Called Subprograms: FFUN, COMNUC.
6. Calling Subprograms: XSCAL.
7. Comments: FP, FA, FN, FP*, FA*, and F2N are computed using equations (46) through (51) of Volume II, respectively. The argument N identifies the reaction type.

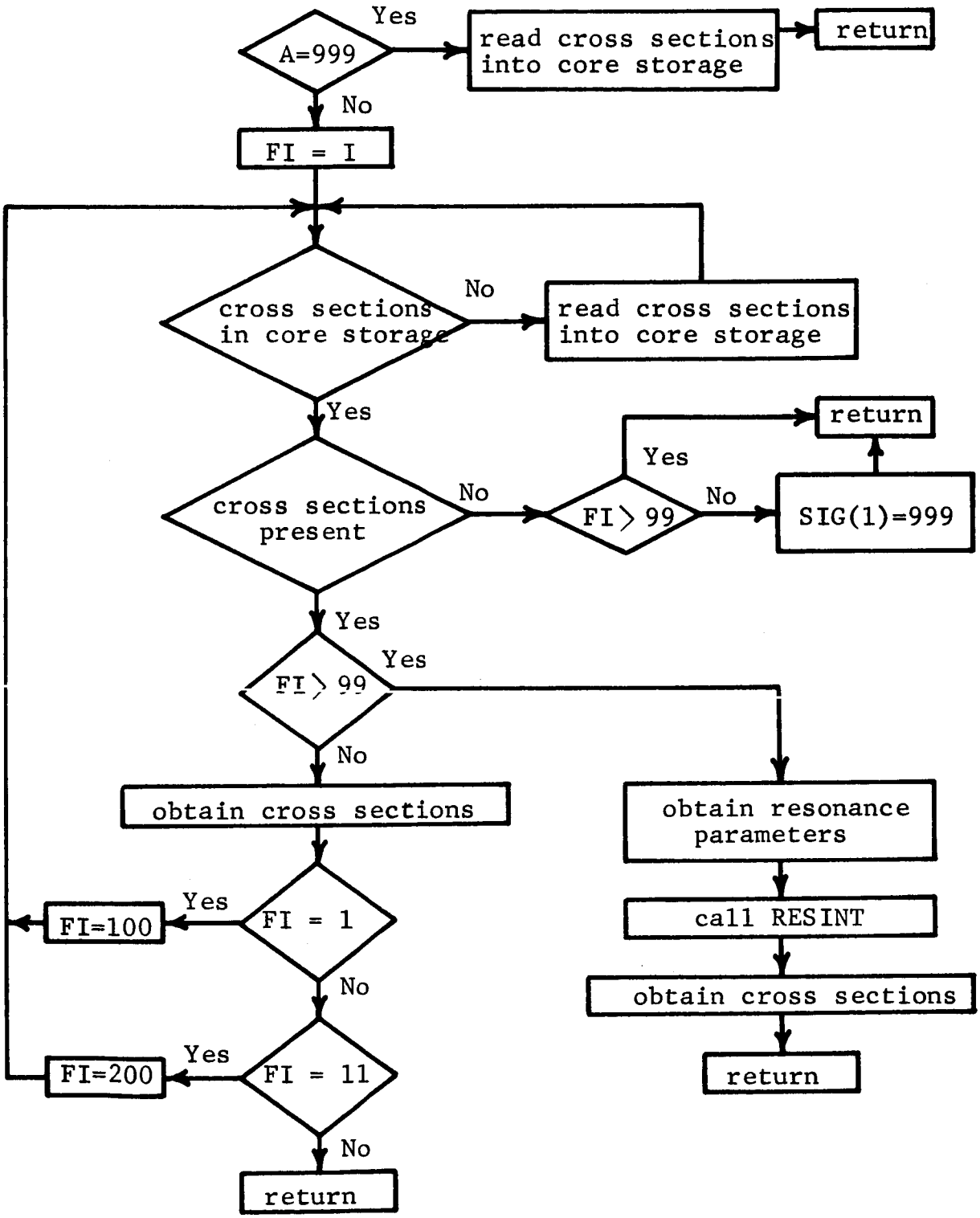
FUNCTION SIGCAL FLOW CHART



X. Subroutine XLIB

1. Purpose: Reads cross sections and resonance parameters from FORTRAN tape unit 1 (data supplied by program user) or tape unit 8 (NAP Cross Section Library), and initiates resonance integral calculations.
2. Arguments: Z, A, I, SIG, LKEY, ISO.
3. Dimensioned Variables: SIG(43), X(250).
4. Common Variables: None.
5. Called Subprograms: RESINT
6. Calling Subprograms: FS, NATDEN
7. Comments: Z and A identify the isotope for which the cross section is desired, I the cross section type, SIG the cross section array, LKEY the cross section option, and ISO the isotopic index. The initial reading of data from the library into core storage is accomplished by setting A equal to 999. If the desired cross sections are not found, SIG(1) is set equal to 999.

SUBROUTINE XLIB FLOW CHART



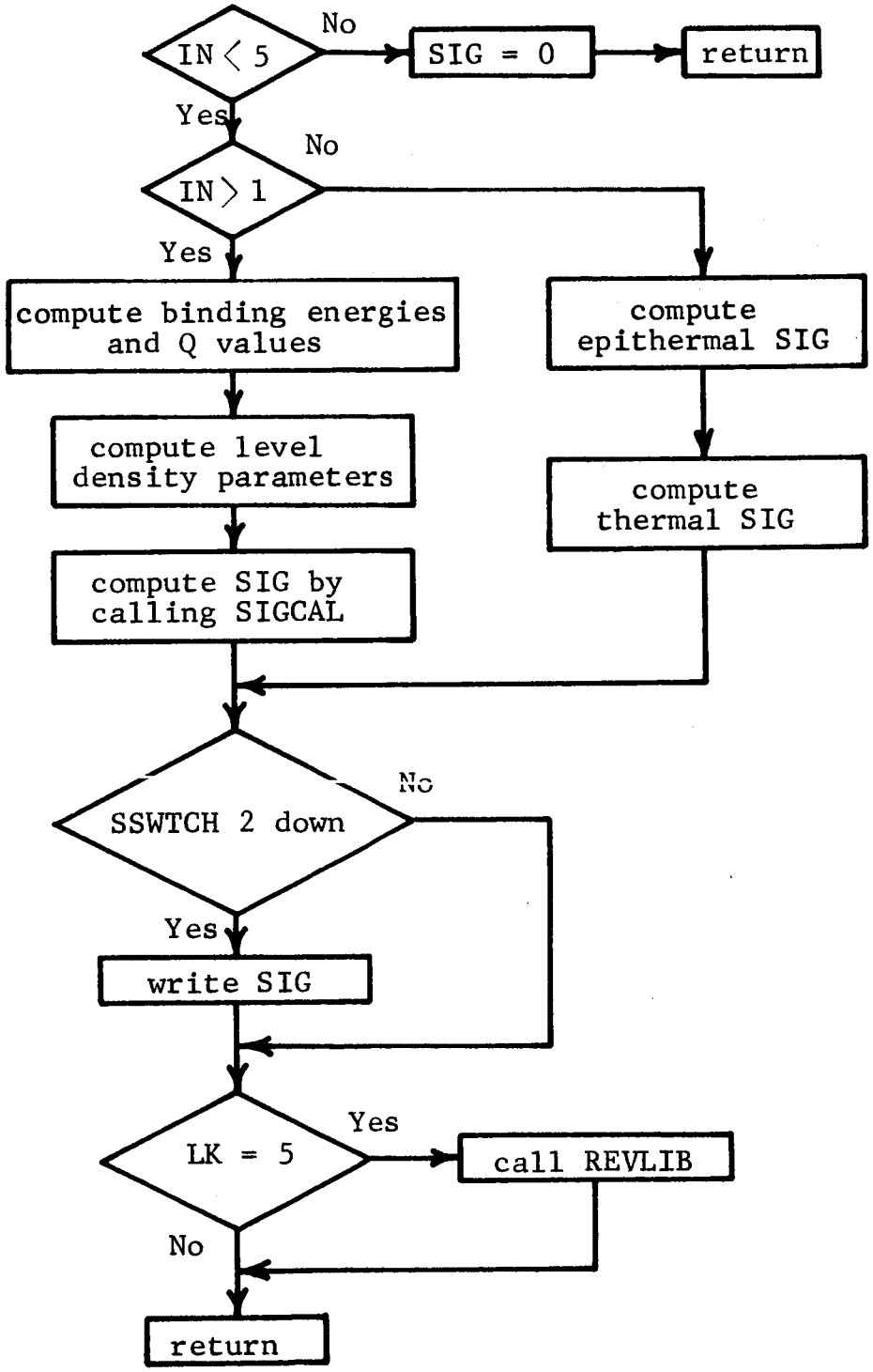
Y. Subroutine XSCAL

1. Purpose: Controls calculation of cross sections and contains $\sigma(n,\gamma)$ calculation.
2. Arguments: Z, A, IN, SIG, LK, ISO.
3. Dimensioned Variables:

BX(12)	FLXIN(43)	XSIG(10)
ELIM(43)	NA(10)	VFAC(10)
EN(10)	NZ(10)	
FELIM(44)	SIG(43)	

4. Common Variables:
   CF: ELIM, FELIM, FLXIN, IFLX, IWT, NA, NOBG, NONV, NZ, TFAC, VFAC.
   CP: BX, IC, LEAF
   CX: AA, AN, AP, BEA, BEN, BEP, CA, CN, CP, QA, QN, QP.
5. Called Subprograms: EXMAS, PAGE, SIGCAL, REVLIB.
6. Calling Subprograms: FS
7. Comments: Z and A identify the isotope for which the cross section is desired, IN the reaction type, SIG the cross section array, LK the cross section option, and ISO the isotopic index.

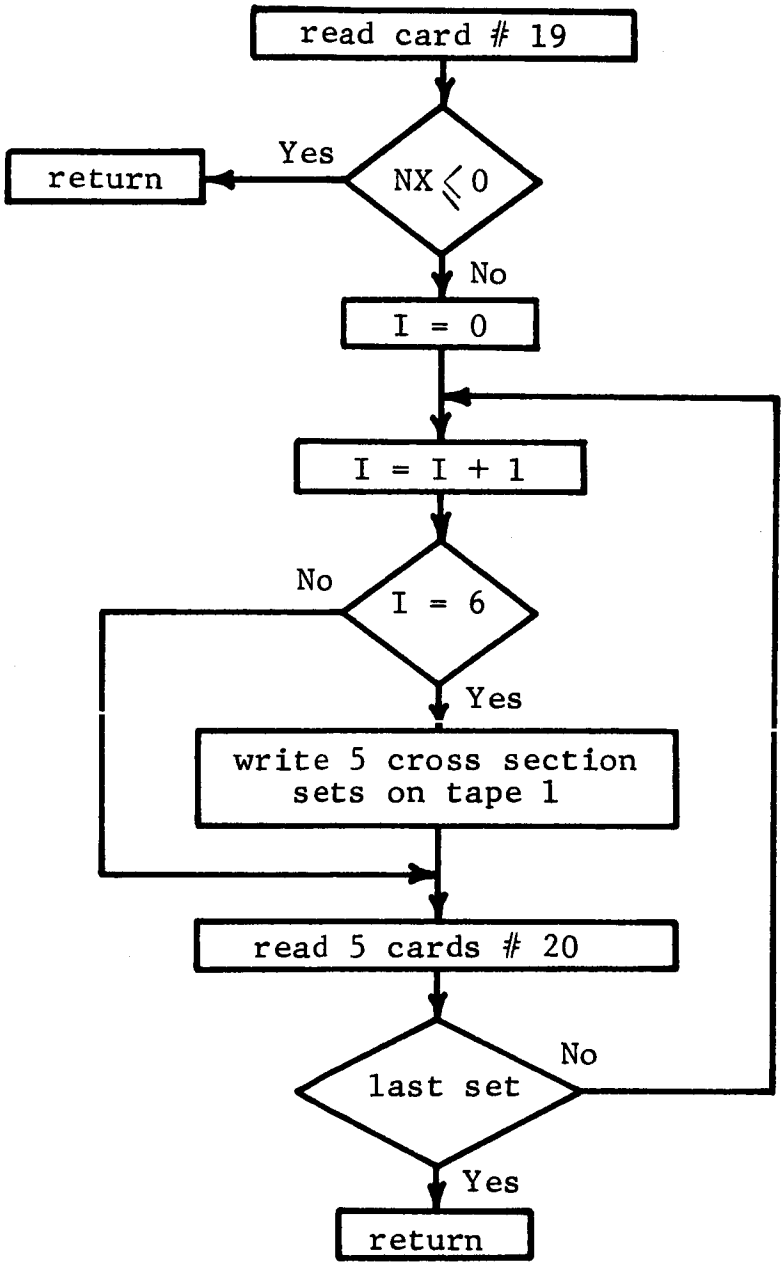
SUBROUTINE XSCAL FLOW CHART



Z. Subroutine XSIN

1. Purpose: Places cross sections input by program user on FORTRAN tape unit 1 in format required by subroutine XLIB.
2. Arguments: None.
3. Dimensioned Variables: X(250)
4. Common Variables: None.
5. Called Subprograms: None.
6. Calling Subprograms: MAIN
7. Comments: None.

SUBROUTINE XSIN FLOW CHART



VII. PROGRAM LISTING

This section gives a complete program listing of the main program and each subroutine. The entire program is written in the FORTRAN IV language, version 13. The listing of the main program is given first, followed by the listings of the subroutines in alphabetical order.

```
      DIMENSION BX(12),ELIM(43),EGG(21),FLXIN(43),FLX(860),TI(50),TS(50)
     1,POW(50),NINT(50),T(200),POWR(200),FELIM(44),IZ(20),IA(20),DEN(20)
     2,IKEY(20),REGS(20,200),Z(5),A(5),D(5),SOR(20,5),NZ(10),NA(10),
     3VFAC(10),KMAX(20),VOL(20),RD(20),NLIM(43),SPOT(20)
      COMMON /BS/ NOGG,LASTT,EGG,T,POWR,SOR
      COMMON /CF/ NOBG,ELIM,FELIM,FLXIN,NZ,NA,VFAC,NONV,TFAC,IFLX,IWT
      COMMON /CP/IC,LEAF,BX
      COMMON /CQ/ NLIM
      COMMON /R/ ISOR, IZ, IA, DEN, SPOT, R, TEMP
C               FLUXN = CONSTANT FLUX NORMALIZATION FACTOR
C               NOBG = NO OF NEUTRON ENERGY GROUPS, LESS THAN 44
C               NOREG = NO OF SPATIAL REGIONS, LESS THAN 21
C               NOGG = NO OF GAMMA ENERGY GROUPS, LESS THAN 21
C               NOSS = NON-ZERO IF SELF-SHIELDING IS REQUIRED
C               NISO = NON-ZERO IF FLUX IS ANISOTROPIC AND NOSS NON-ZERO
      CALL RLIB(1,1,,999,,EGG)
      TFLX = 0.
    1 READ (5,901) (BX(I),I=1,12)
  901 FORMAT(12A6)
      LEAF = 1
      IC = 70
      CALL PAGE
      READ (5,902) FLUXN,NOBG,NOREG,NOGG,NOSS,NISO,IFLX,IWT
  902 FORMAT (E12,5,7I6)
      IC = IC+3
      CALL PAGE
      WRITE (6,903) NOBG,NOREG,NOGG
  903 FORMAT(//10H THERE ARE,I5,24H  NEUTRON ENERGY GROUPS,,I5,14H  REGI
     1ONS, AND,I5,21H  GAMMA ENERGY GROUPS )
      IF (NOSS.LE.0) GO TO 2
      IC = IC+1
      WRITE (6,904)
  904 FORMAT(43H NEUTRON SELF-SHIELDING WILL BE CALCULATED )
      IF (NISO.LE.0) GO TO 2
      WRITE (6,905)
  905 FORMAT(34H THE INCIDENT FLUX IS ANISOTROPIC  )
      IC = IC+1
    2 READ (5,906) (FLIM(I),I=1,NOBG)
  906 FORMAT(6E12,5)
      IF (IFLX.EQ.5) READ (5,933) (NLIM(I),I=1,NOBG)
  933 FORMAT(24I3)
      K = NOGG+1
      IC = IC + 3
      WRITE (6,920) FLUXN
  920 FORMAT(/34H  THE FLUX NORMALIZATION FACTOR IS ,1PE15.8/)
      READ (5,906) (EGG(I),I=1,K)
      READ (5,906) (FLXIN(I),I=1,NOBG)
C
CC        SET UP THE CROSS SECTION ENERGY LIMITS FELIM
C
      FELIM(4) = 1.E7
      FELIM(1) = FELIM(4)*EXP(.75)
      FELIM(2) = FELIM(4)*EXP(.5)
      FELIM(3) = FELIM(4)*EXP(.25)
      I = 5
```

```
      U = .25
   12 EELIM(I) = EELIM(4)*EXP(-U)
      U = U+.25
      I = I+1
      IF (I.LT.16) GO TO 12
      U = 3.0
   13 EELIM(I) = EELIM(4)*EXP(-U)
      U = U+.5
      I = I+1
      IF (I.LT.44) GO TO 13
      EELIM(44) = .001
      CALL GROUPS(TFLX)
      IC = IC+NOGG+3
      CALL PAGE
      WRITE (6,909)
  909 FORMAT(//26H     GAMMA ENERGY GROUP NO.,10X,25H LOWER ENERGY LIMIT
     1(MEV)    )
      DO 4 I=1,K
      J = I - 1
    4 WRITE (6,908) J,EGG(I)
  908 FORMAT (14X, I5, 18X, 1PE14.7 )
      IF (NOSS.LE.0) GO TO 35
      CALL SHIELD(NORG,NOREG,FLXIN,FLX,NISO)
      GO TO 37
   35 IF (NOSS.EQ.0) GO TO 36
      DO 3 I=1,NORG
    3 FLX(I) = FLXIN(I)
      KP = NORG*NOREG
      MP = NORG + 1
      READ (5,906) (FLX(I),I=MP,KP)
      TFLUX = TFLX
      DO 42 J=2,NOREG
      DO 41 I=1,NORG
      KP = (J-1)*NORG + I
   41 FLXIN(I) = FLX(KP)
      CALL GROUPS(TFLX)
   42 CONTINUE
      TFLX = TFLUX
      GO TO 37
   36 DO 40 J=1,NOREG
      DO 40 I=1,NORG
      MP = (J-1)*NORG + I
      FLX(MP) = FLXIN(I)
   40 CONTINUE
   37 KP = NORG*NOREG
      DO 5 I=1,KP
    5 FLX(I) = FLUXN*FLX(I)
      READ (5,924) TFAC,NONV
  924 FORMAT (E12.5,I6)
      IC = IC + 3
      CALL PAGE
      WRITE (6,926) TFAC
  926 FORMAT (/ 9H UNIT A =,1PE14.7)
      IF (NONV.LE.0) GO TO 109
      READ (5,925) (N7(I),NA(I),VFAC(I),I=1,NONV)
  925 FORMAT (3(2I6,E12.5))
```

```
      IC = IC+2+NONV
      CALL PAGE
      WRITE (6,927)
  927 FORMAT (//10X,2H Z,5X,2H A,5X,15H NON 1/V FACTOR  )
      DO 38 I=1,NONV
   38 WRITE (6,928) NZ(I),NA(I),VFAC(I)
  928 FORMAT (8X,I3,3X,I3,8X,1PE14.7)
  109 READ (5,910) NOPER,NODOS
  910 FORMAT(2I6)
C
C               NOPER = NO OF PERIODS OF EQUAL TIME INTERVALS, LESS THAN 51
C               NODOS = NO OF TIMES DOSE CALC. STARTS AND STOPS, LESS THAN 51
C               TI = LENGTH OF EACH TIME PERIOD IN HOURS
C               TS = TIME (IN HOURS) AT WHICH DOSE CALC. STARTS OR STOPS
C
      READ (5,906) (POW(I),I=1,NOPER)
      READ (5,906) (TI(I),I=1,NOPER)
      READ (5,911) (NINT(I),I=1,NOPER)
  911 FORMAT (12I6)
      READ (5,906) (TS(I),I=1,NODOS)
      LASTT = 1
      DO 6 I=1,NOPER
    6 LASTT = LASTT+NINT(I)
      I = 1
      J = 2
      K = 1
      T(1) = 0.
      POWR(1) = POW(1)
    7 DT = TI(I)/FLOAT(NINT(I))
    8 T(J) = DT + T(J-1)
      POWR(J) = POW(I)
      K = K + 1
      J = J+1
      IF (K.LE.NINT(I)) GO TO 8
      I = I+1
      K = 1
      IF (I.LE.NOPER) GO TO 7
C
C     CALCULATE THE TOTAL FLUX
C
   10 TFLX = FLUXN*TFLX
      IC = IC + 5
      CALL PAGE
      WRITE (6,912) TFLX
  912 FORMAT(//90H ISOTOPIC CONCENTRATIONS ARE CALCULATED AT THE FOLLOWI
     1NG TIMES AND NORMALIZED POWER LEVELS /44H THE POWER IS NORMALIZED
     2TO A TOTAL FLUX OF ,1PE14.7/10X,14H TIME INTERVAL,5X,13H TIME (HOU
     3RS),10X,12H POWER LEVEL )
      J = 0
      WRITE (6,913) J,T(1)
  913 FORMAT(13X,I5,10X,1PE14.7)
      DO 11 K=2,LASTT
      J = J+1
      IC = IC + 1
      CALL PAGE
   11 WRITE (6,914) J,T(K),POWR(K)
```

```
  914 FORMAT(13X,I5,10X,1PE14.7,9X,E14.7)
      CALL XSIN
C
C         THE LOOP ON THE REGION NUMBER IR STARTS HERE
C          ISOR = THE NO. OF INPUT ISOTOPES IN THIS REGION
C         IZ(I) = Z FOR THE ITH ISOTOPE
C         IA(I) = A FOR THE ITH ISOTOPE
C         DEN(I) = ATOM DENSITY OF THE ITH ISOTOPE IN UNITS OF E+24
C         IKEY(I) = 0 IF CROSS SECTION NOT IN LIBRARY SHOULD BE CALCULATED
C                 = 1 IF CROSS SECTION SHOULD NOT BE CALCULATED
C                 = 2 IF CROSS SECTION SHOULD ALWAYS BE CALCULATED
C                 = 3 IF CROSS SECTION IS SUPPLIED
C
      IR = 1
   14 READ (5,915) ISOR,R,TEMP,VOL(IR),RD(IR)
  915 FORMAT(I12,4F12.5)
      IF (ISOR.LE.0)  GO TO 160
      IC = 61
      CALL PAGE
      WRITE (6,907) ISOR, IR, VOL(IR), R, TEMP
  907 FORMAT(///10H THERE ARE,I5,31H ISOTOPES OR ELEMENTS IN REGION,I4 /
     1 21H THE REGION VOLUME IS,1PE15.8,3H CC/30H HALF THE MEAN CHORD LE
     2NGTH IS,E15.8,3H CM/19H THE TEMPERATURE IS,E15.8,7H DEG F,/)
      READ (5,916) (IZ(I),IA(I),IKEY(I),DEN(I),I=1,ISOR)
  916 FORMAT(3(2I3,I6,F12.5))
      CALL NATDEN(IKEY)
      IC = IC+4+ISOR
      CALL PAGE
      WRITE (6,917) IR
  917 FORMAT(//37H THE INITIAL ATOM DENSITIES IN REGION,I5,5H  ARE/14X,
     12H Z,8X,2H A,6X,20H ATOM DENSITY (E+24),5X,21H CROSS SECTION OPTIO
     2N )
      DO 15 I=1,ISOR
   15 WRITE (6,918) IZ(I),IA(I),DEN(I),IKEY(I)
  918 FORMAT(10X,I5,5X,I5,7X,1PE14.7,12X,I5)
      DO 16 I=1,NORG
      KP = I+NORG*(IR-1)
   16 FLXIN(I) = FLX(KP)
      I = 1
   17 ZISO = IZ(I)
      AISO = IA(I)
      ATD = DEN(I)
      LKEY = IKEY(I)
      KM = 0
      CALL ISOCON(ZISO,AISO,ATD,LKEY,KM,I)
      KMAX(I) = KM
      I = I+1
      IF (I.LE.ISOR) GO TO 17
      END FILE 2
      REWIND 2
      IC = 53
      CALL PAGE
      WRITE (6,919) IR
  919 FORMAT(//45H THE ATOM DENSITIES (E+24 ATOMS/CC) IN REGION,I4,5H  A
     1RE )
      IC = IC + 4
```

```
      DO 19 J=1,200
      DO 18 NG=1,NOGG
   18 REGS(NG,J) = 0.
   19 CONTINUE
      I = 1
  120 KK = KMAX(I)
      IF (KK.LE.0)  GO TO 145
      DO 140 K=1,KK
      READ (2) IMIN,IMAX,(Z(M),A(M),M=1,IMAX)
      IC = IC + 5
      CALL PAGE
      WRITE (6,930) K
  930 FORMAT(//14H  CHAIN NUMBER,I3//13H TIME (HOURS),5X,25H    Z    A
     1ATOM DENSITY    )
      DO 135 J=2,LASTT
      READ (2) (D(M),M=1,5)
      READ (2) ((SOR(M,MM),M=1,NOGG),MM=IMIN,IMAX)
      IC = IC+3
      CALL PAGE
      WRITE (6,921) T(J)
  921 FORMAT(/1PE15.7/)
      DO 130 M=IMIN,IMAX
      IZ(M) = IFIX(Z(M)+.5)
      IA(M) = IFIX(A(M)+.5)
      IC = IC+1
      CALL PAGE
      WRITE (6,929) IZ(M),IA(M),D(M)
  929 FORMAT(19X,2I4,1PE15.7)
      DO 125 NG=1,NOGG
  125 REGS(NG,J) = REGS(NG,J) + SOR(NG,M)
  130 CONTINUE
  135 CONTINUE
  140 CONTINUE
  145 I = I + 1
      IF (I.LE.ISOR)  GO TO 120
  147 IC = 70
      CALL PAGE
      WRITE (6,922) IP
  922 FORMAT(//37H THE PHOTON SOURCE STRENGTH IN REGION,I4,3H IS//31X,
     1 15HSOURCE STRENGTH,36X,5HGAMMA,14X,6HENERGY/35X, 7HDENSITY,12X,
     2 14HENERGY DENSITY,8X,15HSOURCE STRENGTH,6X,15HSOURCE STRENGTH/
     3 115H  TIME(HOURS)   ENERGY GROUP   (PHOTONS/CC-SEC)     (PHOTONS-
     4MEV/CC-SEC)      (PHOTONS/SEC)      (PHOTONS-MEV/SEC)    )
      IC = IC + 8
      DO 155 J=2,LASTT
      DUM = .5*REGS(1,J)*(EGG(1)+EGG(2))
      DUM1 = VOL(IR)*REGS(1,J)
      DUM2 = VOL(IR)*DUM
      IC = IC + 2
      CALL PAGE
      WRITE (6,923) T(J),REGS(1,J),DUM,DUM1,DUM2
  923 FORMAT(/1PE14.7,5X,2H 1,11X,E14.7,9X,E14.7,8X,E14.7,7X,E14.7)
      IF (NOGG.LE.1)  GO TO 155
      DO 150 NG=2,NOGG
      DUM = .5*REGS(NG,J)*(EGG(NG)+EGG(NG+1))
      DUM1 = VOL(IR)*REGS(NG,J)
```

```
      DUM2 = VOL(IR)*DUM
      IC = IC + 1
      CALL PAGE
  150 WRITE (6,935) NG,REGS(NG,J),DUM,DUM1,DUM2
  935 FORMAT(14X,I7,11X,1PE14.7,9X,E14.7,8X,E14.7,7X,E14.7)
  155 CONTINUE
      CALL DOS(RD(IR),VOL(IR),IR,NODUS,TS,REGS,NOGG,LASTT,EGG,T)
  160 IR = IR + 1
      REWIND 2
      REWIND 8
      REWIND 10
      IF (IR.LE.NOREG)  GO TO 14
      REWIND 1
      GO TO 1
      END
```

```
      FUNCTION AJ(XI,BETA)
      DIMENSION BJ(27,10)
      DATA ((BJ(I,J),J=1,10),I=1,12)/407.9,407.,496.9,3*496.8,4*496.7,
     1 353.2,351.7,351.4,7*351.3,251.4,249.1,248.7,2*248.5,5*248.4,180.1
     2 ,176.7,176.1,175.9,175.8,5*175.7,130.7,125.7,124.8,124.5,124.4,
     3 3*124.3,2*124.2,96.67,89.93,86.72,88.31,88.12,88.02,87.96,87.92,
     4 87.9,87.88,73.55,65.01,63.35,62.73,62.52,62.39,62.3,62.25,62.21,
     5 62.18,57.73,47.77,45.62,44.85,44.5,44.3,44.19,44.12,44.07,44.03,
     6 46.47,35.89,33.28,32.3,31.83,31.58,31.43,31.33,31.26,31.21,37.81,
     7 27.59,24.71,23.54,22.97,22.65,22.45,22.32,22.23,22.17,30.45,21.53
     8  ,18.67,17.41,16.75,16.38,16.14,15.98,15.87,15.79,23.67,16.76,
     9 14.23,13.01,12.35,11.94,11.63,11.51,11.38,11.29/
      DATA ((BJ(I,J),J=1,10),I=13,23)/17.3,12.68,10.74,9.718,9.119,8.739
     1 ,8.484,8.304,8.174,8.077,11.54,8.081,7.815,7.027,6.629,6.322,
     2 6.107,5.93,5.783,5.744,7.172,6.014,5.342,4.914,4.624,4.419,4.268,
     3 4.154,4.066,3.997,4.088,3.658,3.371,3.169,3.022,2.911,2.826,2.759
     4 ,2.706,2.663,2.204,2.067,1.966,1.889,1.829,1.781,1.743,1.712,
     5 1.687,1.666,1.148,1.109,1.078,1.053,1.033,1.016,1.002,.9904,.9805
     6 ,.9722,.5582,.5757,.5671,.5599,.5539,.5488,.5445,.5408,.5376,
     7 .5348,.2943,.2936,.2913,.2894,.2877,.2863,.2851,.284,.2831,.2823,
     8 .1469,.1463,.1477,.1472,.1466,.1464,.1461,.1459,.1455,.1453,.07468
     9 ,.07452,.07437,.07424,.07413,.07403,.07395,.07388,.07381,.07375/
      DATA ((BJ(I,J),J=1,10),I=23,27)/.03739,.03735,.03732,.03728,.03726
     1 ,.03723,.03721,.03719,.03718,.03716,.01871,.01870,.01869,2*.01868
     2 ,2*,01867,2*,01864,,01865,,009358,,009356,,009355,,009352,,009350
     3 ,,009348,,009348,,009345,,009345,,009344,2*,004688,2*,004679,3*
     4 .004677,3*.004677,10*.002347/
      IF (BETA.LT..670.) GO TO 5
      AJ = 1.3708/BETA
      RETURN
    5 IF (XI.LE.1.) GO TO 15
      IF (XI.GT.10.) GO TO 30
      IF (BETA.LE.335.) GO TO 10
   30 AJ = 1.570P/SQRT(BETA*(1.+BETA))
      RETURN
   10 Z = ALOG(XI)
      AY = (11.513+ALOG(BETA))/0.69315
      Y = AY**1.4508
      DUM = -2.824-1.9579*Z+.036905*Z*Z-.0025594*Z*Z*Z+Y*(.29494-.027824
     1       *Z+.0010257*Z*Z+.0032990*Z*Z*Z)+Y*Y*(-.0027388+.0012878*Z-
     2       .0002865*Z*Z-.0001168*Z*Z*Z)+Y*Y*Y*1.E-06*(-1.7541-9.6663*Z+
     3       2.7506*Z*Z+.8618*Z*Z*Z)
      ETA = 1. + EXP(DUM)
      AJ = 1.570*BETA/SQRT(BETA*(1.+BETA))
      RETURN
   15 IF (XI.GT.0.05) GO TO 20
   18 DUM = 2.*BETA + 1.77245*XI
      AJ = 1.77245*(1.77245+XI/DUM)/DUM
      RETURN
   20 IF (XI.GE.0.1) GO TO 25
      IF (BETA.GT.335.) GO TO 18
      GO TO 10
   25 AK =(ALOG(BETA)+11.513)/0.69315
      IF (AK.LT.0.) GO TO 10
      IK = AK
```

```
      IXI = 10.0*XI
      AAK = IK
      RAK = AK - AAK
      AXI = IXI
      RXI = 10.0*XI - AXI
      AJ1 = BJ(IK+1,IXI)+RXI*(BJ(IK+1,IXI+1)-BJ(IK+1,IXI))
      AJ2 = BJ(IK+2,IXI)+RXI*(BJ(IK+2,IXI+1)-BJ(IK+2,IXI))
      AJ = AJ1 + RAK*(AJ2-AJ1)
      RETURN
      END
```

```
      FUNCTION ALCOM(Y,Z,A)
      DIMENSION YL(29),ZL(6),FLIST(29,6),FFLIST(4),XLIST(4),FLIS(4)
      BLIF(P,Q,R,S,T) = ((Q-P)*(S-T)/(R-Q)+S)
      IF (Y.GE.(.2)) GO TO 202
  201 ALCOM = 0.
      RETURN
  202 IF (Z.LT.(0.)) GO TO 201
      IF (Y.GT.3.) GO TO 300
      YL(1) = .2
      DO 203 I=2,29
  203 YL(I) = YL(I-1)+.1
      ZL(1) = 10.
      ZL(2) = 20.
      ZL(3) = 30.
      ZL(4) = 50.
      ZL(5) = 70.
      ZL(6) = 90.
      DATA ((FLIST(I,J),I=1,29),J=1,3)/79.E-7.,.00179.,.061.,.59,2.47,6.7,
     1 12.5,19.2,25.6,31.6,37.,42.,46.,50.,53.,55.,57.,58.,59.,59.5,60.4
     2 ,60.8,61.3,61.6,62.1,62.5,63.,63.5,64.,28.6E-11,54.E-6,.00156,
     3 .051,.56,2.79,8.1,15.9,24.5,33.,40.,47.,53.,58.,63.,68.,72.,75.,
     4 77.,79.5,81.,82.5,84.4,85.6,87.,88.3,89.6,90.8,91.9,76.E-15,
     5 97.E-8,59.E-6,.0063,.156,1.38,5.8,14.3,24.8,36.,46.,55.,63.,70.,
     6 76.,81.,86.,89.7,93.7,97.,100.2,103.,105.5,107.7,110.,112.3,114.5
     7 ,116.5,118.3/
      DATA ((FLIST(I,J),I=1,29),J=4,6)/64.E-21,62.E-13,12.3E-8,.000148,
     1 .0173.,39.3,23.9,4.25.1,40.,53.,65.,76.,86.,95.,103.,111.,118.5,
     2 125.5,131.5,137.5,143.,148.,152.,155.6,159.,162.,164.9,167.6,0.,
     3 36.E-16,63.E-10,81.E-7.,.00249.,132,1.94,10.3,25.4,42.,60.,76.,89.
     4 ,101.,113.,123.,131.,140.,148.,155.5,163.3,168.9,173.9,178.6,
     5 182.9,186.9,190.6,194.,197.2,0.,35.E-19,42.E-12,46.E-8.,.00042,
     6 .046,1.21,12.3,25.7,45.,66.,85.,103.,117.,130.,142.,153.,161.,
     7 171.,182.,191.,200.,206.5,212.5,217.7,222.5,226.9,231.1,234.9/
      IF (Y.LT.YL(29)) GO TO 1
      K = 27
      GO TO 2
    1 IF (Y.GT.YL(1)) GO TO 3
      K = 1
    2 KK = 2
      GO TO 9
    3 DO 4 I=1,29
      IF (Y.LE.YL(I)) GO TO 5
    4 CONTINUE
      I = 29
    5 I = I-1
      IF (I.LE.37) GO TO 6
      K = I-1
      GO TO 7
    6 IF (I.GE.2) GO TO 8
      K = I
    7 KK = 3
      GO TO 9
    8 K = I-1
      KK = 4
    9 IF (Z.LT.ZL(6)) GO TO 10
```

```
      L = 5
      GO TO 11
   10 IF (Z.GT.ZL(1)) GO TO 12
      L = 1
   11 LL = 2
      GO TO 19
   12 DO 13 I=1,6
      IF (Z.LE.ZL(I)) GO TO 14
   13 CONTINUE
      I = 6
   14 I = I-1
      IF (I.GT.4) GO TO 15
      IF ((I-1)-1) 16,18,18
   15 L = I-1
      GO TO 17
   16 L = I
   17 LL = 3
      GO TO 19
   18 L = I-1
      LL = 4
   19 J1 = 1
      J3 = LL+L-1
      J2 = L
      X = Y
      N = KK
      I = 1
      KA = K+KK-1
      DO 20 IA=K,KA
      XLIST(I) = YL(IA)
   20 I = I+1
      I = I-1
      KB = 0
      GO TO 22
   21 KB = KB+1
      FFLIST(KB) = DUMMY
   22 I = 1
      DO 23 IA = K,KA
      FLIS(I) = FLIST(IA,J2)
   23 I = I+1
      I = I-1
      J2 = J2+1
      IF (J2-J3) 24,24,25
   24 ASSIGN 21 TO MAIN
      GO TO 30
   25 ASSIGN 26 TO MAIN
      GO TO 30
   26 KB = KB+1
      FFLIST(KB) = DUMMY
      N = KB
      X = Z
      I = 1
      DO 27 KC=L,J3
      XLIST(I) = ZL(KC)
   27 I = I+1
      DO 28 I=1,KB
   28 FLIS(I) = FFLIST(I)
```

```
      ASSIGN 29 TO MAIN
      GO TO 30
   29 ALCOM = 0.01*DUMMY
      RETURN
   30 IF (X-XLIST(N)) 32,31,31
   31 I = N-1
      GO TO 37
   32 IF (X-XLIST(1)) 33,33,34
   33 I = 1
      GO TO 37
   34 DO 35 I=1,N
      IF (X-XLIST(I)) 36,36,35
   35 CONTINUE
      I = N
   36 I = I-1
   37 DUMMY = BLIF(X,XLIST(I),XLIST(I+1),FLIS(I),FLIS(I+1))
      GO TO MAIN,(26,21,29)
  300 R = 1.2 + 1.5*(A**.333333)
      B = 2.884*Z/R
      S = 4.*A*6.6442E-24/(A+4.)
      C = 1.05443E-14/(SQRT(2.*S*B*1.60206E-06))
      Y = C/SQRT(Y)
      D = 1.-R/(Y*(R+T))
      ALCOM = .031416*D*(R+T)*(R+T)
      RETURN
      END
```

```
      FUNCTION ALETH(FELIM,SIG,I,J)
C
C               THIS SUBROUTINE AVERAGES CROSS SECTIONS OVER A 1/E FLUX
C
      DIMENSION FELIM(44), SIG(43)
      M = J-1
      IF (I.EQ.M)  GO TO 10
      ALETH = 0.
      DO 5  K=I,M
      ALETH = ALETH+SIG(K)*ALOG(FELIM(K)/FELIM(K+1))
    5 CONTINUE
      ALETH = ALETH/ALOG(FELIM(I)/FELIM(J))
      RETURN
   10 ALETH = SIG(I)
      RETURN
      END
```

```
      BLOCK DATA
      COMMON /L/AL
      DIMENSION AL(40)
      DATA (AL(I),I=1,40)/,57735027,,86113631,,33998104,,93246951,,66120
     1939,,23861919,,96028986,,79666648,,52553241,,18343464,,97390653,,8
     26506337,,67940957,,43339539,,14887434,1,,,447214,1,,,765055,,28523
     32,1,,,34785485,,65214515,,20740065,,36076157,,46791393,,10122854,,
     422238103,,31370665,,36268378,,06667134,,14945135,,21908636,,269266
     572,,29552422,,16666667,,83333333,,066667,,378475,,554858/
      END
```

```
      FUNCTION COMNUC(A,EN)
C     SUBROUTINE FOR COMPOUND NUCLEUS CROSS-SECTION
      DIMENSION F(22),G(22)
      AREA= (70.687)* (A**0.66667)
      IF (EN.LE.0.) GO TO 206
      SEX = .3289*((A**1.33333)/(1.+A))*SQRT(EN)
      BEX = SQRT(SEX**2  +2.25*(A**0.66667))
      L = 4+IFIX(SEX)
      IF (L.GT.15) L=15
      K=L+2
      SUM = 0.
      D=0.
      B=1.
      C=5.
      DO 202  J=2,K
      G(1) =0.
      G(2) =0.
      G(3) =1.
      F(1) =1.
      F(2) =1.
      F(3) =1./SEX
      AJ=J
      IF(J-3) 203,203,204
  203 V=1./(F(J)**2 +G(J)**2 )
      VP=(1./(SEX**2 ))*((SEX**2 )*(F(J-1)**2 +G(J-1)**2 )+((AJ-2.)**2 )
     1*(F(J)**2 +G(J)**2 )-2.*SEX*(AJ-2.)*(G(J)*G(J-1)+F(J)*F(J-1)))
      T=((4.*SEX*BEX*V)/(BEX**2+(2.*BEX+SEX*VP)*SEX*V))*B
      B=3.
      GO TO 205
  204 F(J)=((3.+D) /SEX)*F(J-1)-F(J-2)
      G(J) =((3.+D)/SEX)*G(J-1)-G(J-2)
      D=D+2.
      V=1./(F(J)**2 +G(J)**2 )
      VP=(1./(SEX**2 ))*((SEX**2 )*(F(J-1)**2 +G(J-1)**2 )+((AJ-2.)**2 )
     1*(F(J)**2 +G(J)**2 )-2.*SEX*(AJ-2.)*(G(J)*G(J-1)+F(J)*F(J-1)))
      T=C*(4.*SEX*BEX*V)/(BEX**2+(2.*BEX+SEX*VP)*(SEX*V))
      C=C+2.
  205 SUM=SUM+T
  202 CONTINUE
      COMNUC = (AREA/(SEX*SEX))*SUM*1.E-3
      RETURN
  206 COMNUC = 0.
      RETURN
      END
```

```
      SUBROUTINE DOS(RD,VOL,IR,NODOS,TS,REGS,NOGG,LASTT,EGG,T)
      DIMENSION TS(50), REGS(20,200), T(200), EGG(21), E(18), C(18),
     1          DR(200),D(25)
      COMMON /CP/ IC,LEAF,BX(12)
      DATA (E(I),I=1,18)/.1,.15,.2,.3,.4,.5,.6,.9,1.,1.25,1.5,2.,3.,4.,
     1 5.,6.,8.,10./
      DATA (C(I),I=1,18)/5.9,5.82,5.82,5.54,5.43,5.425,5.44,5.58,5.77,
     1 5.96,6.27,6.81,7.89,8.41,9.,9.46,10.22,10.67/
      IF (RD.LE.0.)  RETURN
      DO 5 J=1,200
    5 DR(J) = 0.
      IC = 70
      CALL PAGE
      WRITE (6,901) RD,IR
  901 FORMAT(//28H THE GAMMA DOSE RATE AT R = ,1PE15.7,15H CM FROM REGIO
     1N,I5,3H IS/36H  TIME(HOURS)      DOSE RATE (RAD/HR)    )
      IC = IC + 5
      DO 33 M=2,LASTT
      DO 30 NG=1,NOGG
      EN = .5*(EGG(NG)+EGG(NG+1))
      IF (EN.GE.0.1)  GO TO 10
      Z = EN*VOL*REGS(NG,M)/(7.4142E+06*RD*RD)
      GO TO 30
   10 IF (EN.LE.10.) GO TO 15
      Z = EN*VOL*REGS(NG,M)/(1.3408E+07*RD*RD)
      GO TO 30
   15 J = 0
      DO 20 K=1,18
      J = J+1
      IF (EN.GT.E(K))  GO TO 20
      GO TO 25
   20 CONTINUE
   25 J = J - 1
      CONV = C(J) + (EN-E(J))*(C(J+1)-C(J))/(E(J+1)-E(J))
      Z = EN*VOL*REGS(NG,M)/(1.2566E+06*CONV*RD*RD)
   30 DR(M) = DR(M) + Z
      IC = IC+1
      CALL PAGE
   33 WRITE (6,902) T(M),DR(M)
  902 FORMAT(1PE15.7,3X,E15.7)
      IC = 70
      CALL PAGE
      WRITE (6,903) RD,IR
  903 FORMAT(//23H THE GAMMA DOSE AT R = ,1PE15.7,23H  CM FROM SOURCE RE
     1GION,I5,4H  IS/58H  INITIAL TIME(HOURS)      FINIAL TIME(HOURS)
     2DOSE(RAD)    )
      IC = IC + 5
      J = 2
   35 IF (J.GT.NODOS)  GO TO 80
      M = 2
      K = J - 1
   40 IF (TS(K).LT.T(M))  GO TO 45
      M = M + 1
      GO TO 40
   45 MB = M - 1
```

```
      M = MB
   50 IF (TS(J).LE.T(M))  GO TO 55
      M = M + 1
      GO TO 50
   55 ME = M
      IF (ME.LE.(MB+1))  GO TO 65
      DRX = DR(MB) + (TS(K)-T(MB))*(DR(MB+1)-DR(MB))/(T(MB+1)-T(MB))
      AB = .5*(T(MB+1)-TS(K))*(DR(MB+1)+DRX)
      DRX = DR(ME-1) + (TS(J)-T(ME-1))*(DR(ME)-DR(ME-1))/(T(ME)-T(ME-1))
      AC = .5*(TS(J)-T(ME-1))*(DRX+DR(ME-1))
      IB = MB + 1
      IE = ME - 2
      AR = 0.
      DO 60 I=IB,IE
   60 AR = .5*(T(I+1)-T(I))*(DR(I+1)+DR(I)) + AR
      DOSE = AR + AB + AC
      GO TO 70
   65 DRX = DR(MB)+(TS(K)-T(MB))*(DR(ME)-DR(MB))/(T(ME)-T(MB))
      DRY = DR(MB)+(TS(J)-T(MB))*(DR(ME)-DR(MB))/(T(ME)-T(MB))
      DOSE = .5*(TS(J)-TS(K))*(DRY+DRX)
   70 WRITE (6,904) TS(K), TS(J), DOSE
  904 FORMAT(3X,1PE15.7,8X,E15.7,6X,E15.7)
      IC = IC + 1
      CALL PAGE
      J = J + 2
      GO TO 35
   80 RETURN
      END
```

```
      FUNCTION ENER(FELIM,SIG,I,J)
C
C               THIS SUBROUTINE AVERAGES CROSS SECTIONS OVER A CONSTANT
C               FLUX PER UNIT ENERGY
C
      DIMENSION FELIM(44), SIG(43)
      M = J-I
      IF (I.EQ.M) GO TO 10
      ENER = 0.
      DO 5 K=I,M
      ENER = ENER+SIG(K)*(FELIM(I)-FELIM(K+1))
    5 CONTINUE
      ENER = ENER/(FELIM(I)-FELIM(J))
      RETURN
   10 ENER = SIG(I)
      RETURN
      END
```

```
      FUNCTION EXMAS(Z,A)
C A SUBPROGRAM FOR COMPUTING MASS EXCESS FROM THE WING-FONG FORMULA
C SEE ANL-6886
      DIMENSION B(5),C(5),D(5),E(5)
      ZA = A*(1.+.003*A)/(2.+.01*A)
      IF (AMOD(A,2.)) 4,1,4
    1 IF (AMOD(Z,2.)) 3,2,3
    2 DELTA = -1.
      GO TO 5
    3 DELTA = 1.
      GO TO 5
    4 DELTA = 0.
    5 S1 = 0.
      S2 = 0.
      D(1) = 3.49
      D(2) = 5.99
      D(3) = 5.75
      D(4) = 7.76
      D(5) = 5.02
      E(1) = A-Z-28.
      E(2) = A-Z-50.
      E(3) = A-Z-82.
      E(4) = A-Z-126.
      E(5) = A-Z-152.
      IF (E(1)) 7,6,6
    6 B(1) = 4.04
      C(1) = 0.
      GO TO 8
    7 B(1) = 0.
      C(1) = 1.44
    8 IF (E(2)) 10,9,9
    9 B(2) = 5.96
      C(2) = 0.
      GO TO 11
   10 B(2) = 0.
      C(2) = 2.88
   11 IF (E(3)) 13,12,12
   12 B(3) = 2.49
      C(3) = 0.
      GO TO 14
   13 B(3) = 0.
      C(3) = 5.32
   14 IF (E(4)) 16,15,15
   15 B(4) = 2.9
      C(4) = 0.
      GO TO 17
   16 B(4) = 0.
      C(4) = 5.36
   17 IF (E(5)) 19,18,18
   18 B(5) = 6.88
      C(5) = 0.
      GO TO 20
   19 B(5) = 0.
      C(5) = 5.29
   20 I = 1
```

```
21 S1 = S1 + D(I)*(B(I)*B(I)+C(I)*C(I))/(E(I)*E(I)+B(I)*B(I)+C(I)*C(I
  1       ))
   IF (I-5) 22,23,23
22 I = I+1
   GO TO 21
23 D(1) = 3,07
   D(2) = 2,74
   D(3) = 4,22
   E(1) = Z-28,
   E(2) = Z-50,
   E(3) = Z-82,
   IF (E(1)) 25,24,24
24 B(1) = 2,27
   C(1) = 0,
   GO TO 26
25 B(1) = 0,
   C(1) = 2,77
26 IF (E(2)) 28,27,27
27 B(2) = 4,31
   C(2) = 0,
   GO TO 29
28 B(2) = 0,
   C(2) = 3,1
29 IF(E(3)) 31,30,30
30 B(3) = 1,51
   C(3) = 0,
   GO TO 32
31 B(3) = 0,
   C(3) = 2,35
32 I = 1
33 S2 = S2+D(I)*(B(I)*B(I)+C(I)*C(I))/(E(I)*E(I)+B(I)*B(I)+C(I)*C(I))
   IF (I-3) 34,35,35
34 I = I+1
   GO TO 33
35 S = S1 + S2
   AS = SQRT(A)
   EXMAS = A*(,0089794*A-2,0717)+33,448+(Z-ZA)*(Z-ZA)*(1,629-30,11/AS
  1        +215,8/A)+11,51*DELTA/AS-S
   RETURN
   END
```

```
      FUNCTION FFUN(J,TZ,TA,E2,E1,C,A,BEN)
C
C                SET UP NO. OF POINTS (=K) FOR INTEGRATION
C
      IF (E2.GT.E1) GO TO 1
      FFUN = 0.
      RETURN
    1 K = 10.*(E2-E1)
      IF (K.GT.50) K = 50
      IF (K.LT.10) K = 10
      DE = (E2-E1)/(FLOAT(K-1))
      M = 2
      IF (J.NE.1) GO TO 2
      R = 1.5*(TA**.333333)
      B = 1.442*(TZ-1.)/R
      GO TO 3
    2 IF (J.NE.2) GO TO 3
      R = 1.2 + 1.5*((TA-3.)**.333333)
      B = 1.442*2.*(TZ-2.)/R
    3 I = 1
      FFUN = 0.
      FUN = 0.
      EE = E1
C
C                INTEGRATION LOOP STARTS HERE
C
    4 IF (EE.LE.0.) GO TO 8
      Y = E2 - EE
      RHO = 0.
      IF (Y.GT.0.)  RHO = C*EXP(2.*SQRT(A*(Y+BEN)))
      IF (J.LT.3) GO TO 5
      EN = EE*(TA+1.)/TA
      SIG = COMNUC(TA,EN)
      GO TO 7
    5 X = EE/B
      IF (J.NE.1) GO TO 6
      Z = TZ - 1.
      SIG = PROCOM(X,Z,TA)
      GO TO 7
    6 IF (J.NE.2) GO TO 7
      Z = TZ - 2.
      PA = TA - 3.
      SIG = ALCOM(X,Z,PA)
    7 FUN = EE*SIG*RHO
      IF ( I.EQ.1) FUN = .5*FUN
      IF (I.EQ.K) FUN = .5*FUN
      FFUN = FFUN + DE*FUN
      IF (M.EQ.1) WRITE (6,901) EE,SIG,RHO
  901 FORMAT(20X,4H EE=,E13.6,6H  SIG=,E13.6,6H  RHO=,E13.6)
    8 IF (I.GE.K) GO TO 9
      I = I + 1
      EE = EE + DE
      GO TO 4
    9 IF (J.EQ.2) GO TO 10
      FFUN = 4.7836*TA*FFUN/(TA+1.)
```

```
      GO TO 11
   10 FFUN = 4.*4.7836*(TA-3.)*FFUN/(TA+1.)
   11 IF (M.EQ.1) WRITE (6,902) J,TZ,TA,E2,E1,A,FFUN
  902 FORMAT(20X,3H J=,I3,6(3X,E13.6))
      RETURN
      END
```

```
      FUNCTION FISS(FELIM,SIG,I,J)
C
C               THIS SUBROUTINE AVERAGES CROSS SECTIONS OVER A FISSION
C               FLUX SPECTRUM
C
      DIMENSION FELIM(44), SIG(43)
      M = J-1
      IF (I.EQ.M)  GO TO 10
      FISS = 0.
      T = 1.29E+06
      DO 5  K=I,M
      FISS = FISS+SIG(K)*((1.+FELIM(K+1)/T)*EXP(-FELIM(K+1)/T)-(1.+FELIM
     1       (K)/T)*EXP(-FELIM(K)/T))
    5 CONTINUE
      FISS = FISS/((1.+FELIM(J)/T)*EXP(-FELIM(J)/T)-(1.+FELIM(I)/T)*EXP
     1       (-FELIM(I)/T))
      RETURN
   10 FISS = SIG(I)
      RETURN
      END
```

```
      FUNCTION FS(Z,A,INRE,LKEY,ISO)
C
C               THIS SUBROUTINE INTEGRATES THE FLUX TIMES THE GROUP CROSS
C               SECTIONS OVER ENERGY AND PRINTS THE GROUP CROSS SECTIONS
C               IF SENSE SWITCH 3 IS DOWN
C
      DIMENSION BX(12), FLIM(43), FELIM(44), FLXIN(43), NZ(10), NA(10),
     1          VFAC(10), GSIG(43) , SIG(43)
      COMMON /CF/ NORG, FLIM, FELIM, FLXIN, NZ, NA, VFAC, NONV, TFAC,
     1           IFLX, IWT
      COMMON /CP/ IC, LEAF, BX
      COMMON /CG/ XLIM(43)
      FS = 0.
      DO 5 I=1,43
      SIG(I) = 0.
    5 GSIG(I) = 0.
      G = TFAC
      IF (IFLX.EQ.5)  GO TO 25
      IF (NONV.EQ.0)  GO TO 12
      DO 10 I=1,NONV
      IZ = Z
      IA = A
      IF (IZ.NE.NZ(I))  GO TO 10
      IF (IA.NE.NA(I))  GO TO 10
      G = TFAC*VFAC(I)
   10 CONTINUE
   12 IF (LKEY.EQ.2)  GO TO 15
      CALL XLIB(Z,A,INRE,SIG,LKEY,ISO)
      IF (SIG(1).NE.999.)  GO TO 20
      IF (LKEY.NE.1)  GO TO 15
      RETURN
   15 IF (INRE.EQ.11)  RETURN
      IF (INRE.EQ.12)  RETURN
      IF (INRE.EQ.13)  RETURN
      IF (INRE.EQ.14)  RETURN
      CALL XSCAL(Z,A,INRE,SIG,LKEY,ISO)
   20 SIG(43) = G*SIG(43)
      CALL SSWTCH(3,IS2)
      IF (IS2.NE.1)  GO TO 23
      IC = IC+12
      CALL PAGE
      WRITE (6,920) Z,A,INRE,(SIG(II),II=1,43)
  920 FORMAT(/29H THE FINE GROUP XSECTS FOR Z=,F5.1,3H A=,F6.1,6H INRE=,
     113,4H ARE/(5E20.8))
   23 IF (IFLX.EQ.5)  GO TO 165
      IF (IWT.EQ.0)  GO TO 45
      I = 1
      DO 40 K=1,NORG
      DO 25 J=I,44
      IF (ELIM(K).GE.FELIM(J))  GO TO 28
   25 CONTINUE
   28 IF (J.EQ.44)  GO TO 35
      IF (IWT.NE.2)  GO TO 30
      GSIG(K) = FNER(FELIM,SIG,I,J)
      GO TO 40
```

```
   30 GSIG(K) = ALETH(FELIM,SIG,I,J)
      GO TO 40
   35 GSIG(K) = SIG(43)
   40 I = J
      GO TO 75
   45 I = L
      DO 72 K=1,NORG
      DO 50 J=I,44
      IF (FLIM(K).GE.FELIM(J))  GO TO 53
   50 CONTINUE
   53 IF (J.EQ.44)  GO TO 55
      IF (J.LE.17)  GO TO 60
      IF (I.GE.17)  GO TO 65
      X = (17-I)/(J-I)
      IF (X.LT.0.5) GO TO 65
      GO TO 60
   55 GSIG(K) = SIG(43)
      GO TO 70
   60 GSIG(K) = FISS(FELIM,SIG,I,J)
      GO TO 70
   65 GSIG(K) = ALETH(FELIM,SIG,I,J)
   70 I = J
   72 CONTINUE
   75 IF (IFLX.NE.1)  GO TO 85
      FS = FLXIN(1)*GSIG(1)*ALOG(FELIM(1)/ELIM(1))
      DO 80 K=2,NORG
   80 FS = FS+FLXIN(K)*GSIG(K)*ALOG(ELIM(K-1)/ELIM(K))
      GO TO 105
   85 IF (IFLX.NE.2)  GO TO 95
      FS = FLXIN(1)*GSIG(1)*(FELIM(1)-ELIM(1))
      DO 90 K=2,NORG
   90 FS = FS+FLXIN(K)*GSIG(K)*(ELIM(K-1)-ELIM(K))
      GO TO 105
   95 DO 100 K=1,NORG
  100 FS = FS + FLXIN(K)*GSIG(K)
  105 CALL SSWTCH(3,IS)
      IF (IS.NE.1) GO TO 155
      IF (FS.LE.0.)   GO TO 160
      IC = IC+NORG+4
      CALL PAGE
      IZ = Z
      IA = A
      IF (INRE.GT.4) GO TO 108
      GO TO (110,115,120,125), INRE
  108 INR = INRE-10
      GO TO (130,135,140,145), INR
  110 WRITE (6,901) IZ, IA
      GO TO 150
  115 WRITE (6,902) IZ, IA
      GO TO 150
  120 WRITE (6,903) IZ, IA
      GO TO 150
  125 WRITE (6,904) IZ, IA
      GO TO 150
  130 WRITE (6,911) IZ, IA
      GO TO 150
```

```
  135 WRITE (6,912) IZ, IA
      GO TO 150
  140 WRITE (6,913) IZ, IA
      GO TO 150
  145 WRITE (6,914) IZ, IA
  150 WRITE (6,905)
      WRITE (6,906) (J,GSIG(J),J=1,NOBG)
C
C                 CONVERT UNITS ON FS
  155 FS = 1.E-24*FS
      IF (IS.NE.1) GO TO 160
      WRITE (6,907) FS
  160 RETURN
  165 I = 0
      J = 1
      K = 1
  170 IF (K.EQ.44)  GO TO 95
      I = I + 1
      GSIG(J) = GSIG(J) + SIG(K)
      IF (K.EQ.NLIM(J))  GO TO 175
      K = K + 1
      GO TO 170
  175 GSIG(J) = GSIG(J)/FLOAT(I)
      I = 0
      J = J + 1
      K = K + 1
      GO TO 170
C
C
  901 FORMAT(/46H  THE (N,GAMMA) GROUP CROSS SECTIONS FOR Z OF ,I3,10H A
     1ND A OF ,I3,5H ARE )
  902 FORMAT(/42H  THE (N,P) GROUP CROSS SECTIONS FOR Z OF ,I3,10H AND A
     1 OF ,I3,5H ARE  )
  903 FORMAT(/46H  THE (N,ALPHA) GROUP CROSS SECTIONS FOR Z OF ,I3,10H A
     1ND A FO ,I3,5H ARE  )
  904 FORMAT(/43H  THE (N,2N) GROUP CROSS SECTIONS FOR Z OF ,I3,10H AND
     1A OF ,I3,5H ARE  )
  905 FORMAT(20X,7H GROUP ,20X,22H CROSS SECTION (BARNS)  )
  906 FORMAT(22X,I3,23X,1PE20.4)
  907 FORMAT(42H  THE PRODUCT OF CROSS SECTION AND FLUX IS ,1PE20.8,19H
     1   NEUTS/CM-CM-SEC  )
  911 FORMAT(/55H  THE ISOMERIC (N,GAMMA) GROUP CROSS SECTIONS FOR Z OF
     1,I3,10H AND A OF ,I3,5H ARE  )
  912 FORMAT(/51H  THE ISOMERIC (N,P) GROUP CROSS SECTIONS FOR Z OF ,I3,
     110H AND A OF ,I3,5H ARE  )
  913 FORMAT(/55H  THE ISOMERIC (N,ALPHA) GROUP CROSS SECTIONS FOR Z OF
     1,I3,10H AND A OF ,I3,5H ARE  )
  914 FORMAT(/52H THE ISOMERIC (N,2N) GROUP CROSS SECTIONS FOR Z OF ,I3
     1,10H AND A OF ,I3,5H ARE  )
      END
```

```
      SUBROUTINE GROUPS(TFLX)
C
C                THIS SUBROUTINE ADJUSTS THE ORIGINAL GROUP BOUNDARIES TO
C                THE FIXED LIBRARY GROUP BOUNDARIES
C
      DIMENSION ELIM(43), FELIM(44), FLXIN(43), NZ(10), NA(10), VFAC(10)
     1          , BX(12)
      COMMON /CF/ NOBG, ELIM, FELIM, FLXIN, NZ, NA, VFAC, NONV, TFAC,
     1           TFLX, IWT
      COMMON /CP/ TC, LEAF, BX
      TFLX = 0.
      IF (IFLX.EQ.1)  GO TO 140
      IF (IFLX.EQ.2)  GO TO 150
      DO 135 I=1,NOBG
  135 TFLX = TFLX + FLXIN(I)
      GO TO 100
  140 TFLX = FLXIN(1)*ALOG(2.117E+07/ELIM(1))
      DO 145 I=2,NOBG
  145 TFLX = TFLX + FLXIN(I)*ALOG(ELIM(I-1)/ELIM(I))
      GO TO 100
  150 TFLX = FLXIN(1)*(2.117E+07-ELIM(1))
      DO 155 I=2,NOBG
  155 TFLX = TFLX + FLXIN(I)*(ELIM(I-1)-ELIM(I))
C                IF IFLX=5 DO NOT ADJUST GROUP BOUNDARIES
  100 IF (IFLX.EQ.5)  GO TO 42
      J = NOBG - 1
      DO 40 I=1,J
      DO 5  K=2,44
      IF (ELIM(I).GT.FELIM(K)) GO TO 10
    5 CONTINUE
C                IF ELIM LESS THAN 0.4EV THIS GROUP IS THERMAL
   10 IF (ELIM(I).LT.0.4) GO TO 40
      IF (ELIM(I).EQ.FELIM(K-1))GO TO 40
C                IWT=2 IMPLIES GROUP ADJUSTING BY ENERGY, NOT LETHARGY
      IF (IWT.EQ.2)  GO TO 25
      IF (ALOG(FELIM(K-1)/ELIM(I)).GT.ALOG(ELIM(I)/FELIM(K)))  GO TO 15
      K = K-1
C                IFLX=3 IMPLIES INPUT FLUX MUST BE ADJUSTED
   15 IF (IFLX.NE.3)  GO TO 20
      DEL = ALOG(ELIM(I)/FELIM(K))
      X = FELIM(I)
      IF (I.NE.1)  X = ELIM(I-1)
      FLXIN(I) = FLXIN(I)*((DEL+ALOG(X/ELIM(I)))/ALOG(X/ELIM(I)))
      IF (I.EQ.(NOBG-1))  GO TO 18
      FLXIN(I+1) = FLXIN(I+1)*(ALOG(ELIM(I)/ELIM(I+1))-DEL)/
     1             ALOG(ELIM(I)/ELIM(I+1))
      GO TO 20
   18 FLXIN(I+1)  =  FLXIN(I+1) - DEL*FLXIN(I)/
     1             ALOG(ELIM(I-1)/ELIM(I))
   20 ELIM(I) = FELIM(K)
      GO TO 40
   25 IF ((ELIM(I)-FELIM(K)).LT.(FELIM(K-1)-ELIM(I)))  GO TO 30
      K = K-1
   30 IF (IFLX.NE.3)  GO TO 35
      DEL = ELIM(I)-FELIM(K)
```

```
      X = FELIM(I)
      IF (I.NE.1)  X = ELIM(I-1)
      FLXIN(I) = FLXIN(I)*((DEL+X-ELIM(I))/(X-ELIM(I)))
   35 ELIM(I) = FELIM(K)
   40 CONTINUE
   90 B = 0.
      DO 92 I=1,NOPG
   92 IF (FLXIN(I).LE.0.)  FLXIN(I)=0.0
      IF (IFLX.EQ.1)  GO TO 105
      IF (IFLX.EQ.2)  GO TO 115
      DO 95 I=1,NOPG
   95 B = B + FLXIN(I)
      GO TO 125
  105 B = FLXIN(1)*ALOG(2.117E+07/ELIM(1))
      DO 110 I=2,NOPG
  110 B = B + FLXIN(I)*ALOG(ELIM(I-1)/ELIM(I))
      GO TO 125
  115 B = FLXIN(1)*(2.117E+07-ELIM(1))
      DO 120 I=2,NOPG
  120 B = B + FLXIN(I)*(ELIM(I-1)-ELIM(I))
  125 DO 130 I=1,NOPG
  130 FLXIN(I) = TFLX*FLXIN(I)/B
   42 IC = IC+4
      CALL PAGE
      IF (IWT.EQ.0) GO TO 45
      IF (IWT.EQ.1) GO TO 50
      WRITE (6,901)
  901 FORMAT(//75H  CROSS SECTIONS WITHIN GROUPS WEIGHTED WITH CONSTANT
     1FLUX PER UNIT ENERGY    )
      GO TO 55
   45 WRITE (6,902)
  902 FORMAT(//  87H  CROSS SECTIONS WITHIN GROUPS WEIGHTED WITH FISSION
     1FLUX ABOVE 183KEV, 1/E FLUX BELOW    )
      GO TO 55
   50 WRITE (6,903)
  903 FORMAT(//75H  CROSS SECTIONS WITHIN GROUPS WEIGHTED WITH CONSTANT
     1FLUX PER UNIT LETHARGY    )
   55 IF (IFLX.EQ.5) GO TO 60
      IF (IFLX.EQ.2) GO TO 65
      IF (IFLX.EQ.1) GO TO 70
      WRITE (6,907)
  907 FORMAT(52H  INTEGRATED FLUX BETWEEN GROUP BOUNDARIES IN INPUT    )
      GO TO 75
   60 WRITE (6,904)
  904 FORMAT(74H  GROUP BOUNDARIES NOT ADJUSTED, ALL CROSS SECTIONS MUST
     1 BE INPUT OR ZERO    )
      GO TO 75
   65 WRITE (6,905)
  905 FORMAT(40H  AVERAGE FLUX PER UNIT ENERGY IS INPUT    )
      GO TO 75
   70 WRITE (6,906)
  906 FORMAT(42H  AVERAGE FLUX PER UNIT LETHARGY IS INPUT    )
   75 IC = IC+NOPG+2
      CALL PAGE
      WRITE (6,908)
  908 FORMAT(/26H  NEUTRON ENERGY GROUP NO.,10X,24H LOWER ENERGY LIMIT (
```

```
     1EV) ,10X,22H INCIDENT NEUTRON FLUX )
          DO 80 I=1,KGPG
       80 WRITE(6,909) I, FLUX(I), FLXIN(I)
      909 FORMAT(14X,I5,1X,1PE14.7,24X,E14.7)
          RETURN
          END
```

```
      SUBROUTINE ISOCON(ZISO,AISO,ATP,LKEY,KM,TSO)
C     PROGRAM FOR CALCULATING THE SOURCE STRENGTH OF ALL GAMMA TRANSITIONS OF
C     DAUGHTER PRODUCTS FROM THE DECAY OF SOME NEUTRON REACTION PRODUCED PARENT
      DIMENSION B(5),Z(5),A(5),BR(6),E1(5,7),F1(5,7),GND(20,5),SOR(20,5)
     1,EGG(21),T(200),POWR(200),DENS(5),BX(12),RATE(5),GAM(21),CS(8)
      COMMON /BS/ NOGG,LAST1,EGG,T,POWR,SOR
      COMMON /CI/ IC,LEAF,BX
      J3 = 1
      RATE(1) = 0.
      DO 7 I=1,5
      Z(I) = 0.
    4 B(I) = 0.
    7 A(I) = 0.
      DO 8 I=1,21
    8 GAM(I) = 0.
      DO 200 I=1,4
  200 CS(I) = FS(ZISO,AISO,I,LKEY,TSO)
      DO 205 I=5,8
      J = I + 6
  205 CS(I) = FS(ZISO,AISO,J,LKEY,TSO)
      Z(1)=ZISO
      A(1)=AISO
      DO 170 ITYPE=1,6
   10 GO TO (11,14,17,20,23,26,29,32),ITYPE
   11 Z(2)=Z(1)
      A(2)=A(1)+501.0
      GO TO 47
   14 Z(2)=Z(1)
      A(2)=A(1)+1.0
      GO TO 47
   17 Z(2)=Z(1)
      A(2)=A(1)+499.0
      GO TO 47
   20 Z(2)=Z(1)
      A(2)=A(1)-1.0
      GO TO 47
   23 Z(2)=Z(1)-1.0
      A(2)=A(1)+500.0
      GO TO 47
   26 Z(2)=Z(1)-1.0
      A(2)=A(1)
      GO TO 47
   29 Z(2)=Z(1)-2.0
      A(2)=A(1)+497.0
      GO TO 47
   32 Z(2)=Z(1)-2.0
      A(2)=A(1)-3.0
   47 ASSIGN 58 TO JTRAN
      GO TO 50
   49 ASSIGN 59 TO JTRAN
   50 CALL RLIB(J3,Z(2),A(2),GAM)
      IF (ABS(GAM(2)-999.).LT.(.05)) GO TO 170
      HT = GAM(3)
      II = IFIX(GAM(4)+.005)
      BR(3) = GAM(5)
```

```
      IZD = IFIX(GAM(6)+.005)
      IAD = IFIX(GAM(7)+.05)
      DO 54 I1=1,7
      E1(2,I1) = GAM(2*I1+6)
   54 F1(2,I1) = GAM(2*I1+7)
   56 IF(IT.EQ.0) GO TO 60
      GOTOJTRAN,(58,59,67,68,77,78)
   58 IF(IT.EQ.1)GOTO 588
      GO TO 50
  588 ASSIGN 49 TO ITRAN1
      IMIN=1
      GO TO 612
   59 IF(IT.NE.2)GO TO 50
      IMIN = 3
      GO TO 611
   60 IMIN = 1
  611 ASSIGN 170 TO ITRAN1
  612 RATE(2) = .0115525/HT
      Z(3)=IZL
      A(3)=IAL
      ASSIGN 67 TO JTRAN
      GO TO 62
   61 ASSIGN 68 TO JTRAN
   62 CALL RLIB(J3,Z(3),A(3),GAM)
      IF (ABS(GAM(2)-999.).GT.(.05)) GO TO 66
   65 IMAX=2
      ASSIGN 169 TO LTRAN
      GO TO 90
   66 HT = GAM(3)
      IT = IFIX(GAM(4)+.05)
      BR(4) = GAM(5)
      IZD = IFIX(GAM(6)+.5)
      IAD = IFIX(GAM(7)+.5)
      DO 64 I1=1,7
      E1(3,I1) = GAM(2*I1+6)
   64 F1(3,I1) = GAM(2*I1+7)
      IF (IT.EQ.0) GO TO 69
      GO TO JTRAN, (58,59,67,68,77,78)
   67 IF(IT.EQ.1)GO TO 677
      GO TO 62
  677 ASSIGN 61 TO ITRAN2
      GO TO 70
   68 IF(IT.NE.2) GO TO 62
      IMIN=4
   69 ASSIGN 169 TO ITRAN2
   70 RATE(3) = .0115525/HT
      Z(4)=IZD
      A(4)=IAL
      ASSIGN 77 TO JTRAN
      GO TO 72
   71 ASSIGN 78 TO JTRAN
   72 CALL RLIB(J3,Z(4),A(4),GAM)
      IF (ABS(GAM(2)-999.).GT.(.05)) GO TO 76
   75 IMAX=3
      ASSIGN 168 TO LTRAN
      GO TO 90
```

```
 76 HT = GAM(3)
    IT = IFIX(GAM(4)+.5)
    BR(5) = GAM(5)
    IZD = IFIX(GAM(6)+.5)
    IAD = IFIX(GAM(6)+.5)
    DO 73 I1=1,7
    E1(4,I1) = GAM(2*I1+6)
 73 F1(4,I1) = GAM(2*I1+7)
    IF (IT.EQ.0) GO TO 79
    GO TO JTRAN,(58,59,67,68,77,78)
 77 IF(IT.EQ.1)GO TO 771
    GO TO 72
771 ASSIGN 71 TO ITRAN3
    GO TO 80
 78 IF(IT.NE.2)GO TO 72
    IMIN=5
 79 ASSIGN 168 TO ITRAN3
 80 RATE(4) = .0115525/HT
    Z(5)=IZD
    A(5)=IAD
 81 CALL RLIB(J3,Z(5),A(5),GAM)
    IF (ABS(GAM(2)-999.).GT.(.05)) GO TO 85
 84 ASSIGN 167 TO ITRAN
    IMAX=4
    GO TO 90
 85 HT = GAM(3)
    IT = IFIX(GAM(4)+.5)
    BR(6) = GAM(5)
    IZD = IFIX(GAM(6)+.5)
    IAD = IFIX(GAM(6)+.5)
    DO 83 I1=1,7
    E1(5,I1) = GAM(2*I1+6)
 83 F1(5,I1) = GAM(2*I1+7)
    ASSIGN 167 TO ITRAN
    IMAX=5
    RATE(5) = .0115525/HT
 90 IF(IMIN.GT.IMAX)GO TO 166
    GO TO (1,2,3,6,109,110,111,112),ITYPE
  1 FS1 = CS(5)
    GO TO 113
  2 FS1 = CS(1)
    GO TO 113
  3 FS1 = CS(8)
    GO TO 113
  6 FS1 = CS(4)
    GO TO 113
109 FS1 = CS(6)
    GO TO 113
110 FS1 = CS(2)
    GO TO 113
111 FS1 = CS(7)
    GO TO 113
112 FS1 = CS(3)
113 IF (FS1.LE.0.)  GO TO 166
    FST = CS(1)+CS(2)+CS(3)+CS(4)+CS(5)+CS(6)+CS(7)+CS(8)
    CALL SSWTCH(3,IS)
```

```
      CALL SSWTCH(2,IS)
      IF (IS.NE.1)  GO TO 175
      IC = IC+IMAX+2
      WRITE (6,509) (Z(J5),A(J5),J5=1,IMAX)
  509 FORMAT(/14H THIS CHAIN IS / (3X,3H Z=,F5.1,3X,3H A=,F6.1))
  175 A2 = A(2)
      Z2 = Z(2)
      Q5 = 0.
      Q4 = 0.
      Q3 = 0.
      Q2 = 0.
      Q1 = 0.
      S4 = 0.
      S3 = 0.
      S2 = 0.
      S1 = 0.
      RATE(1) = 0.
      GO TO (95,94,93,92,91),IMAX
   91 Q5 = RATE(5)
      S4 = RATE(4)*BR(5)
   92 Q4 = RATE(4)
      S3 = RATE(3)*BR(4)
   93 Q3 = RATE(3)
      S2 = RATE(2)*BR(3)
   94 Q2 = RATE(2)
      S1 = FSI
   95 Q1 = FST
      D1 = ATU
      D2 = 0.
      D3 = 0.
      D4 = 0.
      D5 = 0.
      OD1 = ATU
      OD2 = 0.
      OD3 = 0.
      OD4 = 0.
      OD5 = 0.
      IF (IS.EQ.1)  IC=IC+6
      IF (IS.EQ.1)  CALL PAGE
      IF (IS.EQ.1) WRITE (6,502) Q1,Q2,S1,Q3,S2,Q4,S3,Q5,S4
  502 FORMAT(/4H Q1=,1PE14.7/4H Q2=,E14.7,5X,4H S1=,E14.7/4H Q3=,E14.7,5
     1X,4H S2=,E14.7/4H Q4=,E14.7,5X,4H S3=,E14.7/4H Q5=,E14.7,5X,4H S4=
     2,E14.7)
C
C               GNO(I,J)=NO. OF PHOTONS PER DECAY IN ENERGY GROUP I FROM
C                        CHAIN MEMBER J
C               SOR(I,J)=NO. OF PHOTONS PER UNIT VOL PER UNIT TIME EMITTED
C                        IN ENERGY GROUP I FROM CHAIN MEMBER J
C
      DO 101 I=1,NOGG
      DO 101 J=1,IMAX
      SOR(I,J) = 0.
  101 GNO(I,J) = 0.
      DO 108 J=1,IMAX
      DO 103 K=1,7
      M = 1
```

```
105 IF (E1(J,K).GT.EGG(M)) GO TO 103
    IF (E1(J,K)-EGG(M+1)) 102,102,104
102 M = M+1
    IF (M-NOGG) 105,105,103
104 GNO(M,J) = GNO(M,J)+E1(J,K)
103 CONTINUE
108 CONTINUE
    KM = KM+1
    WRITE( 2) IMIN,IMAX,(Z(I),A(I),I=1,IMAX)
    DO 107 IME=2,LASTT
    D1 = 3.6E+03*(T(IME)-T(IME-1))
    Q1 = Q1*POWR(IME)
    S1 = FS1*POWR(IME)
    EQ1 = EXP(-Q1*DT)
    EQ2 = EXP(-Q2*DT)
    EQ3 = EXP(-Q3*DT)
    EQ4 = EXP(-Q4*DT)
    EQ5 = EXP(-Q5*DT)
    GO TO (100,99,98,97,96),IMAX
 96 D5 = OD5*EQ5+OD1*(S1*S2*S3*S4*(EQ1-EQ5)/((Q5-Q1)*(Q2-Q1)*(Q3-Q1)*
   1(Q4-Q1))+S1*S2*S3*S4*(EQ2-EQ5)/((Q5-Q2)*(Q1-Q2)*(Q3-Q2)*(Q4-Q2))+
   2S1*S2*S3*S4*(EQ3-EQ5)/((Q5-Q3)*(Q1-Q3)*(Q2-Q3)*(Q4-Q3))+S1*S2*S3*
   3S4*(EQ4-EQ5)/((Q5-Q4)*(Q1-Q4)*(Q2-Q4)*(Q3-Q4)))+OD2*S2*S3*S4*((EQ2
   4-EQ5)/((Q5-Q2)*(Q3-Q2)*(Q4-Q2))+(EQ3-EQ5)/((Q5-Q3)*(Q2-Q3)*(Q4-Q3)
   5)+(EQ4-EQ5)/((Q5-Q4)*(Q2-Q4)*(Q3-Q4)))+S3*S4*OD3*((EQ3-EQ5)/((Q5-
   6Q3)*(Q4-Q3))+(EQ4-EQ5)/((Q5-Q4)*(Q3-Q4)))+OD4*S4*(EQ4-EQ5)/(Q5-Q4)
 97 D4 = OD4*EQ4+OD1*S1*S2*S3*((EQ1-EQ4)/((Q4-Q1)*(Q2-Q1)*(Q3-Q1))+
   1(EQ2-EQ4)/((Q4-Q2)*(Q1-Q2)*(Q3-Q2)))+(EQ3-EQ4)/((Q4-Q3)*(Q1-Q3)*(Q2
   2-Q3)))+OD2*S2*S3*((EQ2-EQ4)/((Q4-Q2)*(Q3-Q2)))+(EQ3-EQ4)/((Q4-Q3)*
   3(Q2-Q3)))+OD3*S3*(EQ3-EQ4)/(Q4-Q3)
 98 D3 = OD3*EQ3+OD1*S1*S2*((EQ1-EQ3)/((Q3-Q1)*(Q2-Q1))+(EQ2-EQ3)/((Q3
   1-Q2)*(Q1-Q2)))+OD2*S2*(EQ2-EQ3)/(Q3-Q2)
 99 D2 = OD2*EQ2+OD1*S1*(EQ1-EQ2)/(Q2-Q1)
100 D1 = OD1*EQ1
    IF (IS.EQ.1)  IC=IC+3
    IF (IS.EQ.1)  CALL PAGE
    IF (IS.EQ.1) WRITE(6,503) T(IME),D1,D2,D3,D4,D5
503 FORMAT(/5H T=,1PE14.7,5X,6H N(1)=,E14.7,5X,6H N(2)=,E14.7/6H N(3)=
   1,E14.7,5X,6H N(4)=,E14.7,5X,6H N(5)=,E14.7)
    WRITE ( 2) D1,D2,D3,D4,D5
    OD1 = D1
    OD2 = D2
    OD3 = D3
    OD4 = D4
    OD5 = D5
    DENS(1) = D1
    DENS(2) = D2
    DENS(3) = D3
    DENS(4) = D4
    DENS(5) = D5
    DO 106 J=IMIN,IMAX
    DO 106 I=1,NOGG
106 SOR(I,J) = DENS(J)*1.0E+24*RATE(J)*GNO(I,J)
    WRITE ( 2) ((SOR(K,M),K=1,NOGG),M=IMIN,IMAX)
    IF (IS.NE.1)  GO TO 107
    IC = IC+IMAX-IMIN+3
```

```
      CALL PAGE
      WRITE (6,505) ((SOR(K,M),M=IMIN,IMAX),K=1,NOGG)
  505 FORMAT(/31H THE SOURCE STRENGTH DENSITY IS /(5E20.8))
  107 CONTINUE
  165 CONTINUE
  166 J3 = 2
      GO TO LTRAN,(169,168,167)
  167 GO TO ITRAN3,(71,168)
  168 GO TO ITRAN2,(61,169)
  169 GO TO ITRAN1, (49,170)
  170 J3 = 1
      RETURN
      END
```

```
      SUBROUTINE NATDEN(LKEY)
C             THIS SUBPROGRAM EXTRACTS SCATTERING CROSS SECTIONS AND
C             ELEMENTAL COMPOSITIONS FROM THE CROSS SECTION LIBRARY
      DIMENSION SIG(43), X(250), SPOT(20), ZO(20), IZ(20), AO(20),
     1          IA(20), D(20), DEN(20), LKEY(20), LK(20)
      COMMON /R7/ ISOR, IZ, IA, DEN, SPOT, R, TEMP
      COMMON /CP/ IC, LEAF, BX(12)
C
      DO 5 I=1,ISOR
      SPOT(I) = 0.
      ZO(I) = IZ(I)
      AO(I) = IA(I)
      LK(I) = LKEY(I)
    5 D(I) = DEN(I)
      NOIS = ISOR
      J = 0
      REWIND 8
      REWIND 1
C
      DO 40 I=1,ISOR
      READ (8) (X(M),M=1,250)
      J = J + 1
      NT = 8
      IF (LKEY(I).EQ.3)  NT=1
      IF (LKEY(I).NE.3)  GO TO 12
   10 READ (NT) (X(M),M=1,250)
   12 IF (X(201).LT.ZO(I))  GO TO 10
      DO 15 K=1,201,50
      IF (X(K).EQ.ZO(I))  GO TO 20
   15 CONTINUE
      NOIS = NOIS - 1
      IC = IC + 3
      CALL PAGE
      WRITE (6,901) ZO(I)
  901 FORMAT (/38H  NO CROSS SECTION DATA FOUND FOR Z = ,E13.6,38H  IT H
     1AS BEEN OMITTED FROM THE PROBLEM  /)
      J = J - 1
      GO TO 40
C
   20 SPOT(J) = X(K+4)
      IZ(J) = ZO(I)
      IF (AO(I).NE.0.)  GO TO 35
      IA(J) = X(K+1)
      LK(J) = LKEY(I)
      DEN(J) = X(K+2)*D(I)
   25 K = K + 50
      IF (K.LE.250)  GO TO 30
      K = 1
      READ (NT) (X(M),M=1,250)
      IF (X(2).EQ.999.)  GO TO 45
   30 IF (X(K).NE.ZO(I))  GO TO 40
      IF (X(K+1).EQ.FLOAT(IA(J)))  GO TO 25
      J = J + 1
      NOIS = NOIS + 1
      IF (NOIS.NE.20)  GO TO 20
```

```
      IC = IC + 4
      CALL PAGE
      WRITE (6,902) IZ(J), IA(J)
  902 FORMAT (/21H  20 ISOTOPES REACHED/23H  EVERYTHING AFTER Z = ,I5,6H
     1  A = ,I5, 9H  OMITTED  /)
      GO TO 45
C
   35 IA(J) = AO(I)
      DEN(J) = D(I)
   40 CONTINUE
   45 ISOR = NOIS
      IF (ISOR.GT.20)  ISOR = 20
      DO 42 I=1,ISOR
   42 LKEY(I) = LK(I)
      REWIND 8
      REWIND 1
      DO 50 K=1,43
   50 SIG(K) = 0.
      CALL XLIB(1.,.999,.1,SIG,1,1)
      RETURN
      END
```

```
      SUBROUTINE PAGE
      DIMENSION BX(12)
      COMMON /CP/ IC,LEAF,BX
      IF (IC.LT.53) RETURN
      IC = IC-53+3
      WRITE (6,901) (BX(I),I=1,12),LEAF

  901 FORMAT (1H1,//10X,12A6,10X,5H PAGE,I5//)
      LEAF = LEAF + 1
      RETURN
      END
```

```
      FUNCTION POLY(N,X)
C     THIS SUBPROGRAM COMPUTES THE NTH ORDER LEGENDRE POLYNOMIAL OF X
      IF (N) 2,1,3
    1 POLY = 1.
      RETURN
    2 WRITE (6,901)
  901 FORMAT ( 45HONEGATIVE ORDER LEGENDRE POLYNOMIAL ASKED FOR )
      CALL EXIT
    3 IF (N-1) 4,4,5
    4 POLY = X
      RETURN
    5 IF (N-2) 6,6,7
    6 POLY = 0.5*(3.*X*X-1.)
      RETURN
    7 IF (N-3) 8,8,9
    8 POLY = 0.5*(5.*X*X*X-3.*X)
      RETURN
    9 IF (N-4) 10,10,11
   10 POLY = 0.125*(35.*X*X*X*X-30.*X*X+3.)
      RETURN
   11 M = 5
      EM = M
      POLY2 = 0.5*(5.*X*X*X-3.*X)
      POLY1 = 0.125*(35.*X*X*X*X-30.*X*X+3.)
   12 POLY = ((2.*EM-1.)*X*POLY1-(EM-1.)*POLY2)/EM
      IF (N-M) 13,13,14
   13 RETURN
   14 M = M+1
      POLY2 = POLY1
      POLY1 = POLY
      EM = M
      GO TO 12
      END
```

```
      FUNCTION PROCOM(Y,Z,A)
      DIMENSION YL(39),ZL(9),FLIST(39,9),FFLIST(4),XLIST(4),FLIS(4)
      BLIF(P,Q,R,S,T) = ((Q-P)*(S-T)/(R-Q)+S)
      IF (Y.GE.(.2)) GO TO 202
  201 PROCOM = 0.
      RETURN
  202 IF (Z.LT.(0.)) GO TO 201
      IF (Y.GT.4.) GO TO 300
      YL(1) = .2
      DO 203 I=2,39
  203 YL(I) = YL(I-1)+.1
      ZL(1) = 10.
      DO 204 I=2,9
  204 ZL(I) = ZL(I-1)*10.
      DATA ((FLIST(I,J),I=1,39),J=1,3)/.74,3.7,10.2,16.3,22.,27.1,31.3,
     1 35.,39.,42.,46.,49.,52.,56.,59.,61.,64.,65.5,67.,68.,69.,69.8,
     2 70.2,70.8,71.,71.1,71.2,71.3,71.4,71.5,71.5,71.5,71.5,71.5,71.5,
     3 71.5,71.4,71.4,71.4,.0323,.59,2.81,7.6,12.6,20.7,27.4,33.6,39.,
     4 45.,50.,55.,58.,62.,65.5,69.,72.,74.5,77.,79.,81.,83.,84.5,85.7,
     5 87.,87.7,88.6,89.3,89.9,90.3,90.8,91.1,91.5,91.8,92.1,92.4,92.6,
     6 92.9,93.1,.00179,.106,.96,3.85,9.1,17.2,25.4,33.8,42.,50.,56.,62.
     7 ,67.,72.,75.5,79.5,83.5,87.,90.,93.,96.,98.5,101.,103.,105.,106.5
     8 ,108.,109.,110.5, 111.4,112.2,112.9,113.5,114.1,114.7,115.2,
     9 115.6,116.1,116.5/
      DATA ((FLIST(I,J),I=1,39),J=4,6)/14.7E-05,.0232,.385,2.13,6.6,14.3
     1 ,24.,33.4,43.,52.,60.,67.,74.,79.,84.,89.5,95.,99.5,104.,108.,
     2 111.5,115.,118.,120.5,123.,125.,127.,128.5,129.8,131.,132.,133.,
     3 133.9,134.7,135.5,136.3,137.,137.6,138.2,14.7E-06,.0057,.154,1.23
     4 ,4.8,12.,22.3,33.1,44.,55.,65.,74.,81.,87.,93.,97.,101.,105.,111.
     5 ,116.,120.,125.,128.5,132.5,135.5,138.5,141.,144.,146.,148.,150.,
     6 152.,154.,155.5,157.,158.,159.5,160.5,161.5,20.3E-07,.0015,.07,
     7 .76,3.54,10.2,20.9,32.6,45.,58.,68.,78.,88.,97.,103.,108.,112.,
     8 118.,124.,129.,134.,138.5,142.5,146.,150.,153.,156.,158.5,161.,
     9 163.,165.,167.,169.,170.5,172.,173.5,174.5,176.,177./
      DATA ((FLIST(I,J),I=1,39),J=7,9)/29.2E-08,.00047,.032,.47,2.6,8.7,
     1 19.3,32.2,46.,61.,72.,83.,94.,103.,111.,117.,122.,128.,134.,140.,
     2 145.,150.,154.,158.,162.,166.,169.,172.,175.,178.,181.,183.,185.,
     3 187.,189.,191.,193.,194.,195.,38.3E-09,.00014,.0152,.286,2.2,7.5,
     4 18.3,32.,48.,63.,76.,88.,100.,110.,119.,128.,136.,143.,150.,156.,
     5 162.,167.,172.,177.,181.,185.,188.,192.,195.,198.,200.,203.,205.,
     6 207.,209.,211.,212.,213.,215.,59.E-10,.000045,.0075,.175,1.72,6.5
     7 ,16.9,32.8,46.,64.,79.,93.,105.,116.,127.,137.,145.,152.,160.,
     8 168.,174.,180.,185.,190.,195.,199.,202.,206.,209.,212.,215.,217.,
     9 220.,222.,224.,225.5,227.1,228.6,230./
      IF (Y.LT.YL(39)) GO TO 1
      K = 38
      GO TO 2
    1 IF (Y.GT.YL(1)) GO TO 3
      K = 1
    2 KK = 2
      GO TO 9
    3 DO 4 I=1,39
      IF (Y.LE.YL(I)) GO TO 5
    4 CONTINUE
      I = 39
```

```
    5 I = I-1
      IF (I.LE.37) GO TO 6
      K = I-1
      GO TO 7
    6 IF (I.GE.2) GO TO 8
      K = I
    7 KK = 3
      GO TO 9
    8 K = I-1
      KK = 4
    9 IF (Z.LT.ZL(9)) GO TO 10
      L = 8
      GO TO 11
   10 IF (Z.GT.ZL(1)) GO TO 12
      L = 1
   11 LL = 2
      GO TO 19
   12 DO 13 I=1,9
      IF (Z.LE.ZL(I)) GO TO 14
   13 CONTINUE
      I = 9
   14 I = I-1
      IF (I.GT.7) GO TO 15
      IF ((I-1)-1) 16,18,18
   15 L = I-1
      GO TO 17
   16 L = I
   17 LL = 3
      GO TO 19
   18 L = I-1
      LL = 4
   19 J1 = 1
      J3 = LL+L-1
      J2 = L
      X = Y
      N = KK
      I = 1
      KA = K+KK-1
      DO 20 IA=K,KA
      XLIST(I) = YL(IA)
   20 I = I+1
      I = I-1
      KB = 0
      GO TO 22
   21 KB = KB+1
      FFLIST(KB) = DUMMY
   22 I = 1
      DO 23 IA = K,KA
      FLIS(I) = FLIST(IA,J2)
   23 I = I+1
      I = I-1
      J2 = J2+1
      IF (J2-J3) 24,24,25
   24 ASSIGN 21 TO MAIN
      GO TO 30
   25 ASSIGN 26 TO MAIN
```

```
      GO TO 30
   26 KB = KB+1
      FFLIST(KB) = DUMMY
      N = KB
      X = Z
      I = 1
      DO 27 KC=L,J3
      XLIST(I) = ZL(KC)
   27 I = I+1
      DO 28 I=1,KB
   28 FLIS(I) = FFLIST(I)
      ASSIGN 29 TO MAIN
      GO TO 30
   29 PROCOM = 0.01*DUMMY
      RETURN
   30 IF (X-XLIST(N)) 32,31,31
   31 I = N-1
      GO TO 37
   32 IF (X-XLIST(1)) 33,33,34
   33 I = 1
      GO TO 37
   34 DO 35 I=1,N
      IF (X-XLIST(I)) 36,36,35
   35 CONTINUE
      I = N
   36 I = I-1
   37 DUMMY = BLIF(X,XLIST(I),XLIST(I+1),FLIS(I),FLIS(I+1))
      GO TO MAIN,(26,21,29)
  300 R = 1.5*(A**.333333)
      B = 1.442*Z/R
      S = A*1.6742E-24/(A+1.)
      C = SQRT(2.*S*B*1.60206E-06)
      C = 1.05443E-14/C
      T = C/SQRT(Y)
      D = 1.-R/(Y*(R+T))
      PROCOM = .031416*D*(R+T)*(R+T)
      RETURN
      END
```

```
      SUBROUTINE RESINT(E,G,GAMG,GAMN,GAMF,RI,I)
      DIMENSION IZ(20), IA(20),  DEN(20), SPOT(20) ,BX(12)
      COMMON /R/ ISOR, IZ, IA, DEN,  SPOT, R, TEMP
      COMMON /CP/ IC,LEAF,BX
      GAM = GAMG + GAMN + GAMF
      SIGO = 2.62E+06*G*GAMN/(E*GAM)
      BIGP = 0.
      DO 2 J=1,ISOR
    2 BIGP = BIGP + DEN(J)*SPOT(J)
      DUM = 1. + 2.*R*BIGP
      DUM = 2.*R*SIGO*DEN(I)/DUM
      IF (DUM.GT.1.) GO TO 4
      IAM = 700
      GO TO 6
    4 GAMPR = GAM*SQRT(DUM-1.)
      AMAX = 4.*E/GAMPR
      IAM = AMAX
    6 SIGM = 1./(2.*R*DEN(I))
      DO 10 J=1,ISOR
      IF (IA(J).GT.IAM) GO TO 10
      SIGM = SIGM + DEN(J)*SPOT(J)/DEN(I)
   10 CONTINUE
      IF (IA(I)-IAM) 20, 20, 15
   15 BETA = SIGM/SIGO
      GO TO 25
   20 BETA = SIGM*GAM/(SIGO*GAMG)
   25 A = IA(I)
      XI = GAM*SQRT(A/(1.9148E-04*E*(TEMP+459.69)))
      RI = SIGO*GAMG*BETA*AJ(XI,BETA)/E
      CALL SSWTCH(3,IS)
      CALL SSWTCH(2,IS2)
      IF (IS2.NE.1)  GO TO 28
      IC = IC+3
      CALL PAGE
      WRITE (6,902) E,G,GAMG,GAMN,GAMF,SIGO,BIGP,SIGM,BETA,XI
  902 FORMAT(/3H E=,E12.5,3H G=,E12.5,6H GAMG=,F12.5,6H GAMN=,E12.5,6H G
     1AMF=,E12.5/6H SIGO=,1PE12.5,6H BIGP=,E12.5,6H SIGM=,E12.5,6H BETA=
     2,E12.5,4H XI=,E12.5)
   28 IF (IS.NE.1)  GO TO 30
      IC = IC+2
      CALL PAGE
      WRITE (6,901) E,IZ(I),IA(I),RI
  901 FORMAT(/5H THE ,F8.3,29H EV RESONANCE INTEGRAL FOR Z=,I3,3H A=,I3,
     14H IS ,1PE14.7,6H BARNS )
   30 RETURN
      END
```

```
SUBROUTINE REVLIB(Z,A,IN,SIG)
DIMENSION SIG(43)
RETURN
END
```

```
      SUBROUTINE RLIB(J3,Z,A,GAM)
      DIMENSION GAM(21),X(252)
      IF (A.NE.999.) GO TO 13
      READ (10)  (X(M),M=1,252)
      RETURN
   13 J5 = 1
    1 IF (X(1).GT.(Z+.05)) GO TO 5
      IF (X(232).LT.(Z-.05)) GO TO 5
      IF (ABS(X(1)-Z).GT.(.05)) GO TO 3
      IF (X(2).GE.500.) GO TO 2
      IF (A.GE.500.) GO TO 12
      IF (A.LT.(X(2)-.05)) GO TO 5
      GO TO 6
    2 IF (A.LT.(X(2)-500.05)) GO TO 5
      GO TO 6
    3 IF (ABS(X(232)-Z).GT.(.05)) GO TO 6
      IF (X(233).GE.500.) GO TO 4
      IF (A.GT.(X(233)+.05)) GO TO 5
      GO TO 6
    4 IF (A.GT.(X(233)-499.95)) GO TO 5
      GO TO 6
    6 DO 7 I=1,232,21
      IF (ABS(X(I)-Z).GT.(.05)) GO TO 7
      IF (ABS(X(I+1)-A).LT.(.05)) GO TO 8
    7 CONTINUE
      GAM(2) = 999.
      RETURN
    8 IF(J3.NE.2) GO TO 15
      J3 = 1
      IF (ABS(X(I+3)-2.).LT.(.05)) GO TO 10
      IF (I.EQ.232) GO TO 11
      I = I+21
   15 DO 9 J = 1,21
      K = I+J
    9 GAM(J) = X(K-1)
      RETURN
    5 IF (Z.LE.(X(1)+.5)) GO TO 14
      IF (J5.EQ.2)  J5 = 3
      GO TO 11
   14 IF (ABS(Z-X(1)).LT.(.05)) GO TO 10
      IF (ABS(Z-X(1)).LE.30.) GO TO 10
      REWIND 10
   11 READ (10)  (X(M),M=1,252)
      GO TO 1
   10 BACKSPACE 10
      BACKSPACE 10
      IF (J5.EQ.1)  J5 = 2
      IF (J5.EQ.3)  GO TO 7
      GO TO 11
   12 IF ((A-500.).LT.(X(2)-.5)) GO TO 5
      GO TO 6
      END
```

```
      SUBROUTINE SETUP(IGEON,NOANG,L1)
C THIS SUBROUTINE SETS UP THE ANGLES,WEIGHTS,AND K MATRIX FOR SHIELD
      COMMON /SH/A,AK,AMU/L/AL
      COMMON /CP/ IC,LEAF
      DIMENSION A(10),AK(10,10),AMU(10),AL(40)
      N = NOANG/2
      IF (L1) 1,15,1
C LEGENDRE-GAUSS QUADRATURE
    1 GO TO (2,3,4,5,6),N
    2 A(1) = AL(21)
      A(2) = AL(21)
      AMU(1) = AL(1)
      AMU(2) = -AL(1)
      GO TO 7
    3 A(1) = AL(22)
      A(2) = AL(23)
      A(3) = AL(23)
      A(4) = AL(22)
      AMU(1) = AL(2)
      AMU(2) = AL(3)
      AMU(3) = -AL(3)
      AMU(4) = -AL(2)
      GO TO 7
    4 DO 81 I=1,3
      A(I) = AL(I+23)
      AMU(I) = AL(I+3)
      A(I+3) = A(I)
   81 AMU(I+3) = -AMU(I)
      GO TO 7
    5 DO 82 I=1,4
      A(I) = AL(I+26)
      AMU(I) = AL(I+6)
      A(I+4) = A(I)
   82 AMU(I+4) = -AMU(I)
      GO TO 7
    6 DO 83 I=1,5
      A(I) = AL(I+30)
      AMU(I) = AL(I+10)
      A(I+5) = A(I)
   83 AMU(I+5) = -AMU(I)
    7 IF (IGEON) 8,10,8
    8 DO 9 J=1,NOANG
      DO 9 I=1,NOANG
    9 AK(I,J) = 0.
      RETURN
   10 N1 = NOANG-1
      ANOANG = NOANG
      DO 13 J=1,NOANG
      AMUJ = AMU(J)
      IU = J-1
      DO 13 I=1,IU
      AMUI = AMU(I)
      IF (I-J) 11,12,11
   11 AK(J,I) = 0.5*ANOANG*ANOANG*A(I)*POLY(N1,AMUJ)*POLY(N1,AMUI)/(AMUJ
     1             -AMUI)
```

```
      GO TO 13
   12 AK(J,I) = AMUJ
   13 CONTINUE
      DO 14 I=2,NOANG
      JU = I-1
      DO 14 J=1,JU
   14 AK(J,I) = -A(I)*AK(I,J)/A(J)
      RETURN
C LOBATTO QUADRATURE
   15 GO TO (16,17,18,16,16),N
   16 CALL PAGE
      WRITE (6,901)
  901 FORMAT ( 33HOLOBATTO QUADRATURE NOT PERMITTED / 40HOLEGENDRE-GAUSS
     1 QUADRATURE WILL BE USED )
      GO TO 1
   17 A(1) = AL(36)
      A(2) = AL(37)
      A(3) = AL(36)
      A(4) = AL(37)
      AMU(1) = AL(16)
      AMU(2) = AL(17)
      AMU(3) = -AMU(1)
      AMU(4) = -AMU(2)
      GO TO 19
   18 DO 84 I=1,3
      A(I) = AL(I+37)
      AMU(I) = AL(I+17)
      A(I+3) = A(I)
   84 AMU(I+3) = -AMU(I)
   19 IF (IGEON) 21,20,21
   20 CALL PAGE
      WRITE (6,902)
  902 FORMAT ( 55HOSPHERICAL GEOMETRY NOT PERMITTED IN LOBATTO QUADRATUR
     1E / 40HOLEGENDRE-GAUSS QUADRATURE WILL BE USED )
      GO TO 1
   21 RETURN
      END
```

```
      SUBROUTINE SHIELF(NOBG,NOREG,FLXIN,FLX,NISO)
C THIS SUBROUTINE HANDLES NEUTRON SELF-SHIELDING
      COMMON /SH/ A,AK,AMU
      COMMON /CP/ IC,LEAF,PX
      DIMENSION FLXIN(3),FLX(960),SLD(100),AFLX(100,10),SRS(100,10),
     1DX(20),NINT(20),SIGS(20),SIGT(20),SIGSL(20),X(101),A(10),AUM(10),
     2AK(10,10),AMU(10),BFLX(100),ANINT(20),BX(12)
      READ (5,901) EPS,XO,IGEON,IOUT,NOANG
  901 FORMAT (2E12.4,3I2)
      LITE = 2
      CALL SLITE(0)
      IF (NOANG-20) 2,2,1
    1 NOANG = NOANG - 20
      L1 = 0
      GO TO 3
    2 L1 = 1
    3 IC = 56
      CALL PAGE
      WRITE (6,902)
  902 FORMAT(/32H  NAP SELF-SHIELDING CALCULATION /)
      CALL SETUP(IGEON,NOANG,L1)
      DO 4 K=1,100
    4 SLD(K) = 0.
      DO 100 IGR = 1,NOBG
      IF (LITE.EQ.2)  CALL SLITE(1)
      IF (LITE.NE.2)  CALL SLITE(0)
      DO 5 K=1,100
      DO 5 J=1,10
      AFLX(K,J) = 0.
    5 SRS(K,J) = 0.
      IF (IGR-1) 11,6,11
    6 READ (5,903) (SIGS(I),SIGT(I),SIGSL(I),DX(I),NINT(I),I=1,NOREG)
  903 FORMAT (3E12.4,I2)
      DO 7 K=1,101
    7 X(K) = 0.
      X(1) = XO
      J3 = 1
      DO 10 J1=1,NOREG
      J2 = J3 + NINT(J1) - 1
      DO 9  J4=J3,J2
      X(J4+1) = ABS(X(J4))+DX(J1)
      IF (J4-J3) 9,8,9
    8 X(J4+1) = -X(J4+1)
    9 CONTINUE
   10 J3 = J3 + NINT(J1)
      MAXPT = J3
      JANG = NOANG/2
      IF (IGR-1) 11,92,11
   11 READ (5,904) (SIGS(I),SIGT(I),SIGSL(I),I=1,NOREG)
  904 FORMAT (3E12.4)
   92 IF (NISO.NE.0)  GO TO 12
   13 DO 14 J=1,JANG
   14 AFLX(1,J) = 0.5*FLXIN(IGR)
      GO TO 17
   12 READ (5,905) (AFLX(1,J),J=1,5)
```

```
  905 FORMAT (5E12.4)
   15 DO 16 J=1,JANG
   16 AFLX(1,J) = AFLX(1,J)*FLXIN(IGR)
   17 IF (IOUT-1) 26,18,18
   18 J2 = 1
      IC = IC +5
      IF (IGR.NE.1)  IC = 58
      CALL PAGE
      WRITE (6,906) IGR
  906 FORMAT(/30X,5H GROUP,I4//60X,27H MACROSCOPIC CROSS SECTIONS/10X,
     111H MESH POINT,4X,6H X(CM),4X,7H REGION,9X,11H SCATTERING,13X,6H T
     2OTAL,10X,13H SLOWING DOWN  )
      DO 25 K=1,MAXPT
      IC = IC + 1
      CALL PAGE
      IF (X(K)) 21,22,22
   21 AX = X(K)
      AX = ABS(AX)
      WRITE (6,908) K,AX,J2,SIGS(J2),SIGT(J2),SIGSL(J2)
  908 FORMAT (15X,I5,F12.5,I7,6X,E19.6,5X,E19.6,1X,E19.6)
      J2 = J2 + 1
      GO TO 25
   22 J3 = J2 - 1
   24 AX = X(K)
      AX = ABS(AX)
      WRITE (6,909) K,AX,J3
  909 FORMAT (15X,I5,F12.5,I7)
   25 CONTINUE
      IC = IC+JANG+3
      CALL PAGE
      WRITE (6,910) IGR
  910 FORMAT(/50H  THE ANGULAR FLUX INCIDENT FROM THE LEFT IN GROUP13,
     113H IS / 15X,3H MU,17X,5H FLUX  )
      DO 150 J2=1,JANG
  150 WRITE (6,907) AMU(J2),AFLX(1,J2)
  907 FORMAT(10X,F13.9,4X,E15.8)
   26 IF (IGFON) 95,27,95
   27 DO 29 K=1,MAXPT
      IF (X(K)) 29,28,29
   28 WRITE (6,911) K
  911 FORMAT (16H0 ERROR STOP  X(,I3,10H ) IS ZERO )
      STOP
   29 CONTINUE
   95 ELFLX = 0.0
      ITER=0
C
C     THIS STARTS THE FLUX ITERATIONS
C
   30 J2=1
      ITER=ITER+1
C     THIS LOOP SETS UP THE SOURCE TERM AT EACH POINT
      DO 35  K=1,MAXPT
      DUM=0.
      DO 31  J=1,NOANG
   31 DUM = DUM+(0.5*SIGS(J2)*A(J)*AFLX(K,J))
      DO 33  J=1,NOANG
```

```
      BUM(J)=0.
      DO 32  L=1,NOANG
   32 BUM(J) = BUM(J)+(AK(J,L)*AFLX(K,L)/ABS(X(K)))
   33 SRS(K,J) = SL(K)+BUM(J)+DUM
      IF (K-1) 80,35,80
   80 IF (X(K)) 34,35,35
   34 J2 = J2+1
   35 CONTINUE
C     THIS LOOP COMPUTES ANGULAR FLUX IN FORWARD DIRECTION
      DO 37 J=1,JANG
      J2 = 0
      KU = MAXPT-1
      DO 37  K=1,KU
      IF (X(K+1)) 36,507,507
   36 J2 = J2+1
      SIG = EXP(-DX(J2)*SIGT(J2)/AMU(J))
      W0=(AMU(J)*(1.-SIG))/(DX(J2)*SIGT(J2))
      W0=(W0-SIG)/SIGT(J2)
      W1 = ((1.-SIG)/SIGT(J2))-W0
  507 AFLX(K+1,J)=SIG*AFLX(K,J)+W0*SRS(K,J)+W1*SRS(K+1,J)
   37 CONTINUE
C     THIS LOOP COMPUTES ANGULAR FLUX IN BACKWARD DIRECTION
      JL = JANG+1
      DO 39 J=JL,NOANG
      J2 = NORES
      SIG = EXP(-DX(J2)*SIGT(J2)/AMU(J))
      W0=((AMU(J)*(1.-SIG)/(DX(J2)*SIGT(J2)))-SIG)/SIGT(J2)
      W1 = ((1.-SIG)/SIGT(J2))-W0
      DO 39  K=1,KU
      KM = MAXPT - K + 2
      KN = MAXPT - K + 1
      MK = MAXPT - K
      IF (X(KM)) 38,509,509
   38 J2 = J2-1
      SIG = EXP(-DX(J2)*SIGT(J2)/AMU(J))
      W0=((AMU(J)*(1.-SIG)/(DX(J2)*SIGT(J2)))-SIG)/SIGT(J2)
      W1 = ((1.-SIG)/SIGT(J2))-W0
  509 AFLX(MK,J) = (AFLX(KN,J)-W0*SRS(MK,J)-W1*SRS(KN,J))/SIG
   39 CONTINUE
C     OBTAIN THE INTEGRATED FLUX AT EACH POINT
      DO 41  K=1,MAXPT
      DUM = 0.
      DO 40  J=1,NOANG
   40 DUM = DUM+AFLX(K,J)*A(J)
   41 BFLX(K) = DUM
C     CONVERGE ON FLUX AT LAST POINT
      CALL SSWTCH(1,K5)
      IF (K5.NE.1)  GO TO 85
      IF (ITER.GE.6)  GO TO 44
   85 DUM = BFLX(MAXPT)
      ERR = ABS((DUM-ELFLX)/DUM)
      IF (ERR-EPS) 44,43,43
   43 ELFLX = DUM
      GO TO 30
C
C     THIS COMPLETES ITERATIONS ON FLUX--NEXT COMPUTE SLOWING DOWN SOURCE
```

```
C
   44 SLD(1) = STSL(1)*BFLX(1)
      J2 = 1
      DO 46 K=2,MAXPT
      SLD(K) = STSL(J2)*BFLX(K)
      IF (X(K+1)) 45,45,46
   45 J2 = J2+1
   46 CONTINUE
C     NOW PRINT OUT THE FLUX
      IF (IOUT-3) 62,47,47
   47 IC = 55
      CALL PAGE
      WRITE (6,912) ITER,IGR,IGR
  912 FORMAT                    (20X,11H THERE WERE,I5,20H ITERATIONS IN
     1 GROUP,I5/20X,10H THE GROUP,I5,16H ANGULAR FLUX IS)
      IF (NOANG-5) 48,48,54
   48 IC = IC + 1
      CALL PAGE
      WRITE (6,913) (ANGJ(J),J=1,NOANG)
  913 FORMAT (6X,16H POINT      X(CM),5X,5(5H MU =,F11.8,3X))
   49 DO 50 K=1,MAXPT
      IC = IC + 1
      CALL PAGE
   50 WRITE (6,914) K,X(K),(AFLX(K,J),J=1,NOANG)
  914 FORMAT(6X,I4,1PE14.6,5E19.8)
      GO TO 62
   54 IC = IC + 1
      CALL PAGE
      WRITE (6,913) (ANGJ(J),J=1,JANG)
   55 DO 56 K=1,MAXPT
      IC = IC + 1
      CALL PAGE
   56 WRITE (6,914) K,X(K),(AFLX(K,J),J=1,JANG)
      IC = IC + 1
      CALL PAGE
      WRITE (6,913) (ANGJ(J),J=JL,NOANG)
      DO 61 K= 1,MAXPT
      IC = IC + 1
      CALL PAGE
   61 WRITE (6,914) K,X(K),(AFLX(K,J),J=JL,NOANG)
   62 IF (IOUT-2) 56,63,63
   63 IC = 55
      CALL PAGE
      WRITE (6,915) IGR
  915 FORMAT (20X,10H THE GROUP,I5, 9H  FLUX IS )
      WRITE (6,916)
  916 FORMAT (20X,16H POINT      X(CM) )
      DO 64 K=1,MAXPT
      IC = IC + 1
      CALL PAGE
      AXP = ABS(X(K))
   64 WRITE (6,917) K,AXP,BFLX(K)
  917 FORMAT (20X,I5,1PE14.6,0PE20.8)
C
C     COMPUTE AVERAGE FLUX IN EACH REGION
C
```

```
   66 J1 = 1
      DO 68 J2=1,NOREG
      DUM = -0.5*FLX(J1)
      J3 = J1+NINT(J2)
      DO 67 K=J1,J3
   67 DUM = DUM + FLX(K)
      DUM = DUM - FLX(J3)*0.5
      J1 = J3
      KP = IGR+NOBG*(J2-1)
      ANINT(J2) = NINT(J2)
   68 FLX(KP) = DUM/ANINT(J2)
      IF (LITE.EQ.2)  LIT = 1
      IF (LITE.NE.2)  LIT = 2
      LITE = LIT
  100 CONTINUE
C     PRINT OUT AVERAGE FLUX IN EACH REGION
      IF (IOUT-1) 72,69,69
   69 IC = 56
      CALL PAGE
      WRITE (6,918)
  918 FORMAT (20X,35H THE AVERAGE FLUX IN EACH REGION IS //20X,6H GROUP,
     1 10X, 7H REGION,10X,13H AVERAGE FLUX )
      DO 71 J2=1,NOREG
      DO 71 IGR=1,NOBG
      J4 = (J2-1)*NOBG+IGR
      IC = IC + 1
      CALL PAGE
      WRITE (6,919) IGR,J2,FLX(J4)
  919 FORMAT (20X,I6,10X,I6,E17.8 )
   71 CONTINUE
   72 RETURN
      END
```

```
      SUBROUTINE XLIB(Z,A,I,SIG,LKEY,ISC)
      DIMENSION SIG(43), X(250)
      IF (A.NE.999.) GO TO 1
C
C                INITIALIZE STORING OF XLIB IN CORE
      READ (8)  (X(M),M=1,250)
      RETURN
C
C                START SEARCH HERE, J5 USED TO PREVENT REPEATED SEARCHES
    1 FI = I
      IF (LKEY.EQ.3)  BACKSPACE 1
    2 J5 = 1
      NT = 8
      N4 = 2
      IF (LKEY.EQ.3)  NT=1
      IF (LKEY.EQ.3)  GO TO 12
    3 IF (X(1).GT.(Z+.05)) GO TO 9
      IF (X(201).LT.(Z-.05)) GO TO 9
C                IN Z BALLPARK
      IF (ABS(X(1)-Z).GT.0.05)  GO TO 4
      IF (A.LT.(X(2)-.05)) GO TO 9
      IF (A.GT.(X(202)+.05)) GO TO 9
      IF (FI.LT.(X(4)-.05)) GO TO 9
      IF (FI.GT.(X(204)+.05)) GO TO 9
      GO TO 5
C                PIN DOWN A
    4 IF (ABS(X(201)-Z).GT.0.05) GO TO 5
      IF (A.GT.(X(202)+.05)) GO TO 9
      IF (FI.GT.(X(204)+.05)) GO TO 9
C                RIGHT A AND TYPE BALLPARK
    5 DO 6 M=1,201,50
      IF (ABS(X(M)-Z).GT.0.05) GO TO 6
      IF (ABS(X(M+1)-A).GT.0.05) GO TO 6
      IF (ABS(X(M+3)-FI).LT.0.05) GO TO 7
    6 CONTINUE
C                CROSS SECTION NOT PRESENT
      IF (FI.GT.99.) RETURN
      SIG(1) = 999.
      RETURN
C                CORRECT CROSS SECTION OBTAINED
    7 IF (FI.GT.99.) GO TO 15
      DO 8 J=1,43
      K = M+J+4
    8 SIG(J) = X(K)
      IF (ABS(FI-1.).LT.0.05) GO TO 13
      IF (ABS(FI-11.).LT.0.05) GO TO 14
      IF (N4.EQ.1)  GO TO 17
      RETURN
C                WRONG BALLPARK
    9 IF (Z.LE.(X(1)-.5)) GO TO 10
      IF (J5.EQ.2)  J5=3
      GO TO 12
C                WHICH WAY DO WE GO
   10 IF (ABS(X(1)-Z).GT.40.) GO TO 12
      IF (ABS(Z-X(1)).LT.0.05) GO TO 11
```

```
      IF (ABS(Z-X(1)).LT.30.) GO TO 11
      REWIND NT
      GO TO 12
   11 BACKSPACE NT
      BACKSPACE NT
      IF (IS.EQ.1) JS=2
      IF (IS.EQ.3) GO TO 6
   12 READ (NT) (X(M),M=1,250)
      GO TO 3
C                SEARCH FOR RESONANCE PARAMETERS
   13 FI = 100.
      GO TO 2
   14 FI = 200.
      GO TO 2
C                CALCULATE RESONANCE INTEGRALS
   15 DO 16 J=1,9
      JJ = N + 50J
      E = X(JJ)
      IF (E.LE.0.)  RETURN
      G = X(JJ+1)
      GAMG = X(JJ+2)
      GAMN = X(JJ+3)
      GAMF = X(JJ+4)
      RI = 0.
      CALL RESINT(E,G,GAMG,GAMN,GAMF,RI,ISO)
      U = ALOG(1.0E+07/E)
      K = 0. + 2.*U
      SIG(K) = SIG(K) + 2.*RI
   16 CONTINUE
      IF (N4.EQ.1)  GO TO 17
      RETURN
   17 WRITE (6,901) Z,A,I,LKEY,ISO,(SIG(J),J=1,43)
  901 FORMAT (//2E20.8,3I12/9(5E20.8/))
      RETURN
      END
```

```
      SUBROUTINE XSCAL(Z,A,IN,SIG,LK,ISO)
C
C                THIS SUBROUTINE COMPUTES CROSS SECTIONS
C
      DIMENSION BX(12),ELIM(43), FELIM(44), FLXIN(43), NZ(10), NA(10),
     1          VFAC(10),SIG(43), EN(10), XSIG(10)
      COMMON /CF/ NOBG, ELIM, FELIM, FLXIN, NZ, NA, VFAC, NONV, TFAC,
     1            IFLX, IWT
      COMMON /CP/ IC, LEAF, BX
      COMMON /CX/ QP, QA, QN, BEP, BEA, BEN, CP, CA, CN, AP, AA, AN
C
C                SET ISOMERIC CROSS SECTIONS EQUAL TO ZERO
      IF (IN.LT.5)  GO TO 8
      DO 5 I=1,43
    5 SIG(I) = 0.
      RETURN
C
C                FIND NATURE OF TARGET NUCLEUS
C                IT = 1 MEANS EVEN Z EVEN N
C                   = 2 MEANS ODD  Z EVEN N
C                   = 3 MEANS EVEN Z  ODD N
C                   = 4 MEANS ODD  Z  ODD N
    8 IZ = Z
      IA = A
      N = IA-IZ
      IF (MOD(N,2).EQ.0)  GO TO 15
      IF (MOD(IZ,2),EQ.0)  GO TO 10
      IT = 4
      GO TO 25
   10 IT = 3
      GO TO 25
   15 IF (MOD(IZ,2),EQ.0)  GO TO 20
      IT = 2
      GO TO 25
   20 IT = 1
   25 IF (IN.GT.1)  GO TO 100
C
C                COMPUTE (N,GAMMA) CROSS SECTIONS
C
C                  COMPUTE LEVEL SPACING D
      D = 5.336-.04596*A+.0001103*A*A
      D = 10.**D
      IF (IT.EQ.4)  GO TO 30
      IF (IT.EQ.1)  GO TO 35
      GO TO 40
   30 D = .5*D
      GO TO 40
   35 D = 5.*D
C
C                COMPUTE STRENGTH FUNCTION F
   40 F = -201.1+A*(12.94-A*(.2559-A*(.002217-A*(8.696E-06-1.264E-08*
     1      A))))
      F = .0001*F
      IF (F.LT.0.)  F = 1.E-05
C
```

```
C               COMPUTE CAPTURE WIDTH GG
      GG = 1.802-Z*(.0765-Z*(.001152-5.73E-06*Z))
      IF (Z.LT.30.)  GG = .01297*Z
C
C               CALCULATION ASSUMES G=.5
      B = F*D/GG
      C = 1.31E+06*3.1416*F
      DO 45 I=1,42
      DU = ALOG(FELIM(I)/FELIM(I+1))
      XU = SQRT(1./FELIM(I))
      XL = SQRT(1./FELIM(I+1))
      SIG(I) = C*(XL-XU+B*ALOG((B+XU)/(B+XL)))/DU
      IF (SIG(I).LT.0.)  SIG(I) = 0.
   45 CONTINUE
C
C               THERMAL (N,GAMMA)
      V  = FLOAT(N)
      IF (IT.GT.1)  GO TO 50
      B = 1520.-29.91*V+.1399*V*V
      IF (N.LE.132)  B = .1
      IF (N.LE.122)  B = -6503.+131.6*V-.6401*V*V
      IF (N.LE.82)   B = -104.9+3.195*V-.02278*V*V
      IF (N.LE.54)   B = -17.92+.3383*V
      IF (N.LT.50)   B = -34.61+2.371*V-.03016*V*V
      IF (N.LE.20)   B = 10.**(3.*(.05*V-1.))
      GO TO 65
   50 IF (IT.GT.2)  GO TO 55
      B = -2624.+33.64*V-.09556*V*V
      IF (N.LT.124)  B = 2919.-23.75*V
      IF (N.LT.82)   B = -875.2+28.48*V-.2176*V*V
      IF (N.LT.50)   B = -33.26+2.756*V-.04243*V*V
      IF (N.LE.20)   B = .05*V-.79
      IF (N.LE.10)   B = 10.**(-.46032*V-.15873)
      GO TO 65
   55 IF (IT.GT.3)  GO TO 60
      B = 115441.-1647.4*V+5.988*V*V
      IF (N.LE.133)  B = 10.
      IF (N.LE.121)  B = -30474.+619.86*V-3.0243*V*V
      IF (N.LE.81)   B = 148.5-.7464*V
      IF (N.LE.49)   B = 181.4-12.95*V+.2302*V*V
      IF (N.LE.29)   B = 10.**(3.*(.05*V-1.))
      GO TO 65
   60 B = -2809304.+37936.*V-127.7*V*V
      IF (N.LE.139)  B = 2376.-10.27*V
      IF (N.LT.81)   B = -9897.+317.7*V-2.431*V*V
      IF (N.LE.51)   B = 100.
      IF (N.LT.21)   B = 10.**(2.*(.1*V+1.))
   65 IF (B.LT.0.)   B = 0.
      SIG(43) = B
   70 DO 75 I=1,42
   75 SIG(I)=SIG(I)+SIG(43)*.31812*(1./SQRT(FELIM(I+1))-1./SQRT
     1       (FELIM(I)))/ALOG(FELIM(I)/FELIM(I+1))
      GO TO 200
C
C               COMPUTE Q VALUES AND BINDING ENERGIES
C               QP = TARGET Q VALUE FOR (N,P)
```

```
C               QA = TARGET Q VALUE FOR (N,ALPHA)
C               QN = TARGET Q VALUE FOR (N,2N)
C               BEP = NEUTRON BINDING ENERGY IN (N,P) RESIDUAL NUCLEUS
C               BEA = NEUTRON BINDING ENERGY IN (N,ALPHA) RESIDUAL NUCLEUS
C               BEN = NEUTRON BINDING ENERGY IN TARGET NUCLEUS
C
  100 ZP = Z-1.
      AP = A
      ZA = Z-2.
      AA = A-3.
      ZN = Z
      AN = A-1.
      B = EXMAS(Z,A)
      QP = .7822 + B - EXMAS(ZP,AP)
      QA = 5.6474 + B - EXMAS(ZA,AA)
      QN = B - EXMAS(ZN,AN) - 8.071
      AB = A-4.
      BEP = 7.289 + QP - B + EXMAS(ZP,AN)
      BEA = 2.424 + QA - B + EXMAS(ZA,AB)
      BEN = -QN
C
C               COMPUTE LEVEL DENSITY PARAMETERS.
C               CP = C FOR (N,P) RESIDUAL NUCLEUS
C               CA = C FOR (N,ALPHA) RESIDUAL NUCLEUS
C               CN = C FOR TARGET NUCLEUS
C               SIMILARLY FOR AX
C
      CP = .6441 - .0054*A
      IF (A.GT.115.)  CP = .3459 - .00013*A
      IF (IT.EQ.1)  CP = 2.*CP
      IF (IT.EQ.4)  CP = .2*CP
      CA = .6441 - .0054*(A-2.)
      IF (A.GT.117.)  CA = .3459 - .00013*(A-2.)
      IF (IT.EQ.2)    CA = 2.*CA
      IF (IT.EQ.3)    CA = .2*CA
      CN = .6441 - .0054*A
      IF (A.GT.115.)  CN = .3459 - .00013*A
      IF (IT.EQ.4)    CN = 2.*CN
      IF (IT.EQ.1)    CN = .2*CN
      AP = .03*A
      IF (A.LE.60.)  AP = .1825 - .0033*A + .0005*A*A
      AA = .03*(A-3.)
      IF (A.LE.63.)  AA = .1825 - .0033*(A-3.) + .0005*(A-3.)*(A-3.)
      AN = AP
C
C               CALCULATE EPITHERMAL CROSS SECTIONS
C
      DO 120 I=1,42
      SIG(I) = 0.
      EN(1) = FELIM(I)*1.E-06
      XSIG(1) = 0.
      DE = (FELIM(I)-FELIM(I+1))/9.E+06
      DO 105 J=2,10
      XSIG(J) = 0.
  105 EN(J) = EN(J-1) - DE
      DO 110 J=1,10
```

```
      EPSN = EN(J)*A/(A+1.)
  110 XSIG(J) = SIGCAL(Z,A,EPSN,IN)
      DO 115 J=2,9
  115 SIG(I) = SIG(I) + XSIG(J)
      SIG(I) = (SIG(I) + .5*XSIG(1) + .5*XSIG(10))/(FELIM(I)-FELIM(I+1))
      SIG(I) = SIG(I)*1.E+06
  120 CONTINUE
C
C               CALCULATE 2200 M/SEC CROSS SECTION
C
      EPSN = .0253E-06*A/(A+1.)
      SIG(43) = SIGCAL(Z,A,EPSN,IN)
C
C               PRINT OUT CROSS SECTIONS IF SSW2 DOWN
C               ADD TO LIBRARY IF LKEY = 5
C
  200 CALL SSWTCH(2,IS)
      IF (IS.NE.1)  GO TO 240
      IC = IC+18
      CALL PAGE
      GO TO (205, 210, 215, 220), IN
  205 WRITE (6,901)
  901 FORMAT(/38H  CALCULATED (N,GAMMA) CROSS SECTIONS  )
      GO TO 225
  210 WRITE (6,902)
  902 FORMAT(/33H  CALCULATED (N,P) CROSS SECTIONS  )
      GO TO 225
  215 WRITE (6,903)
  903 FORMAT(/38H  CALCULATED (N,ALPHA) CROSS SECTIONS  )
      GO TO 225
  220 WRITE (6,904)
  904 FORMAT(/34H  CALCULATED (N,2N) CROSS SECTIONS  )
  225 R = 0.
      ES = 0.
      WRITE (6,905) Z, A, R, ES
  905 FORMAT(15X,3H Z=,F7.2,5H   A=,F7.2,5H   R=,F7.5,6H   ES=,F7.3)
      DO 230 I=1,13
      J = I+15
      K = I+30
  230 WRITE (6,906) SIG(I), SIG(J), SIG(K)
  906 FORMAT(17X,1PE13.6,5X,E13.6,5X,E13.6)
      WRITE (6,906) SIG(14), SIG(29)
      WRITE (6,906) SIG(15), SIG(30)
  240 IF (LK.NE.5)  GO TO 400
      CALL REVLIB(Z,A,IN,SIG)
  400 RETURN
      END
```

```
      SUBROUTINE XSIN
C
C               THIS SUBROUTINE READS IN CROSS SECTIONS
C               NX = NO. OF CROSS SECTION SETS READ IN
C
      DIMENSION X(250)
      READ (5,901) NX
  901 FORMAT (I6)
      IF (NX.LE.0 ) RETURN
      REWIND 1
      DO 5 I=1,250
    5 X(I) = 0.
      I = 0
      DO 15 J=1,NX
      I = I+1
      IF (I.NE.6)  GO TO 10
      WRITE (1)  (X(M),M=1,250)
      DO 8 K=1,250
    8 X(K) = 0.
      I = 1
   10 K = 50*(I-1)+1
      L = K+49
      READ (5,902) (X(M),M=K,L)
   15 CONTINUE
  902 FORMAT (5(10E8.1/))
      DO 20 I=1,250
   20 X(I) = 0.
      X(1) = 99.
      X(2) = 999.
      X(3) = 999.
      WRITE (1)  (X(M),M=1,250)
      END FILE 1
      REWIND 1
      RETURN
      END
```

VIII. SAMPLE PROBLEM

This section presents input and output for a sample NAP problem. This sample problem consists of a time-dependent neutron flux incident upon the left face of a five centimeter thick slab containing 0.00254 atoms of sodium per cm^3. A ten centimeter thick slab containing 0.00848 atoms of iron per cm^3 lies behind the sodium-containing slab. An isotropic neutron flux of 10^{14} neutrons per square centimeter per second is incident upon the system for one hour. The neutron source is then turned off for one hour, and then turned on for one more hour at twice the original intensity. The total gamma ray dose nine hours from the start of the first irradiation is calculated 100 centimeters from the rear of the iron-containing slab.

Input data are shown on the next page on a standard form which is submitted to a key punch operator. The input was punched exactly as shown. This particular sample problem is illustrative only. The incident neutrons are treated in two neutron energy groups, while the gamma rays produced from neutron activation are divided into three energy groups. The NAP neutron transport subroutine is used to compute the neutron flux distribution in the two slabs. Output data are shown on the following pages. Gamma ray source strengths, dose rates, and dose are tabulated. This sample problem took approximately 1.5 minutes of operating time on an IBM 7094.

CODING CONVENTIONS	
Letter	Number
ϕ	0
I	1
Z	2
BLANK COLUMN	
b or • or □	
BUT BE CONSISTENT!	

PROGRAMMER D.A. Klopp

DATE

PAGE 1 OF 1

PROBLEM CODE

TITLE NAP Sample Problem

```
NAP  SAMPLE  PRϕBLEM
1.0       E+13      2       2       3       1       0       0       0
0.414               0.001
6.0                 3.0             1.0             0.1
2.0                 8.0
0.01                0.0             1 2 4
0.529               0.536           0.006448        0.5             10
0.300               0.304           0.00305         0.5             20
0.694               0.717           0.0
0.370               0.387           0.0
0.8862                      1
    26          58  .99
    4           4
1.0                 0.0             2.0             0.0
1.0                 1.0             1.0             6.0
    4           4       4       6
1.0                 2.0             3.0             9.0
    0
                15.0                150.0           500.0           112.5
  11  23            10.00254
                    110.0           150.0           1000.0          105.0
  26    0           10.00848
```

THERE ARE 2 NEUTRON ENERGY GROUPS, 2 REGIONS, AND 3 GAMMA ENERGY GROUPS
NEUTRON SELF-SHIELDING WILL BE CALCULATED

THE FLUX NORMALIZATION FACTOR IS 1.00000000E 13

CROSS SECTIONS WITHIN GROUPS WEIGHTED WITH FISSION FLUX ABOVE 183KEV, 1/E FLUX BELOW
INTEGRATED FLUX BETWEEN GROUP BOUNDARIES IN INPUT

NEUTRON ENERGY GROUP NO.	LOWER ENERGY LIMIT (EV)	INCIDENT NEUTRON FLUX
1	4.13993 77E-01	2.0000013E 00
2	1.00000000E-03	7.9999986E 00

GAMMA ENERGY GROUP NO.	LOWER ENERGY LIMIT (MEV)
0	6.00000000E 00
1	3.00000000E 00
2	1.00000000E 00
3	1.00000000E-01

NAP SELF-SHIELDING CALCULATION

GROUP 1

MESH POINT	X (CM)	REGION	MACROSCOPIC CROSS SECTIONS SCATTERING	TOTAL	SLOWING DOWN
1	0.00000	0			
2	0.50000	1	0.529000E 00	0.536000E 00	0.648000E-02
3	1.00000	1			
4	1.50000	1			
5	2.00000	1			
6	2.50000	1			
7	3.00000	1			
8	3.50000	1			
9	4.00000	1			
10	4.50000	1			
11	5.00000	1			
12	5.50000	2	0.300000E 00	0.304000E 00	0.305000E-02
13	6.00000	2			
14	6.50000	2			
15	7.00000	2			
16	7.50000	2			
17	8.00000	2			
18	8.50000	2			
19	9.00000	2			
20	9.50000	2			
21	10.00000	2			
22	10.50000	2			
23	11.00000	2			
24	11.50000	2			
25	12.00000	2			
26	12.50000	2			
27	13.00000	2			
28	13.50000	2			
29	14.00000	2			
30	14.50000	2			
31	15.00000	2			

THE ANGULAR FLUX INCIDENT FROM THE LEFT IN GROUP 1 IS

MU	FLUX
0.861136310	0.10000006E 01
0.339981042	0.10000006E 01

THE GROUP 1 FLUX IS

POINT	X(CM)	
1	0.000000E-39	0.21220001E 01
2	5.000000E-01	0.21902178E 01
3	1.000000E 00	0.22644586E 01
4	1.500000E 00	0.23470885E 01
5	2.000000E 00	0.24405105E 01
6	2.500000E 00	0.25476697E 01
7	3.000000E 00	0.26727039E 01
8	3.500000E 00	0.28219744E 01
9	4.000000E 00	0.30054094E 01
10	4.500000E 00	0.32396620E 01
11	5.000000E 00	0.35520946E 01
12	5.500000E 00	0.39444424E 01
13	6.000000E 00	0.43676040E 01
14	6.500000E 00	0.47141789E 01
15	7.000000E 00	0.48880995E 01
16	7.500000E 00	0.46831160E 01
17	8.000000E 00	0.42945394E 01
18	8.500000E 00	0.38188319E 01
19	9.000000E 00	0.33533033E 01
20	9.500000E 00	0.29958440E 01
21	1.000000E 01	0.26449545E 01
22	1.050000E 01	0.23992160E 01
23	1.100000E 01	0.21575907E 01
24	1.150000E 01	0.19191408E 01
25	1.200000E 01	0.16830014E 01
26	1.250000E 01	0.14483191E 01
27	1.300000E 01	0.12141914E 01
28	1.350000E 01	0.97960227E 00
29	1.400000E 01	0.74334616E 00
30	1.450000E 01	0.50393501E 00
31	1.500000E 01	0.25947911E 00

GROUP 2

MESH POINT	X(CM)	REGION	MACROSCOPIC CROSS SECTIONS SCATTERING	TOTAL	SLOWING DOWN
1	0.00000	0			
2	0.50000	1	0.694000E 00	0.717000E 00	0.000000E-38
3	1.00000	1			
4	1.50000	1			
5	2.00000	1			
6	2.50000	1			
7	3.00000	1			
8	3.50000	1			
9	4.00000	1			
10	4.50000	1			
11	5.00000	1			
12	5.50000	2	0.370000E 00	0.387000E 00	0.000000E-38
13	6.00000	2			
14	6.50000	2			
15	7.00000	2			
16	7.50000	2			
17	8.00000	2			
18	8.50000	2			
19	9.00000	2			
20	9.50000	2			
21	10.00000	2			
22	10.50000	2			
23	11.00000	2			
24	11.50000	2			
25	12.00000	2			
26	12.50000	2			
27	13.00000	2			
28	13.50000	2			
29	14.00000	2			
30	14.50000	2			
31	15.00000	2			

THE ANGULAR FLUX INCIDENT FROM THE LEFT IN GROUP 2 IS

MU	FLUX
0.861136310	0.39999993E 01
0.339981042	0.39999993E 01

THE GROUP 2 FLUX IS

POINT	X(CM)	
1	0.000000E-39	0.11688144E 02
2	5.000000E-01	0.14380620E 02
3	1.000000E 00	0.17102189E 02
4	1.500000E 00	0.19986196E 02
5	2.000000E 00	0.23133880E 02
6	2.500000E 00	0.26645096E 02
7	3.000000E 00	0.30646292E 02
8	3.500000E 00	0.35330003E 02
9	4.000000E 00	0.41026051E 02
10	4.500000E 00	0.48341156E 02
11	5.000000E 00	0.58439283E 02
12	5.500000E 00	0.60354712E 02
13	6.000000E 00	0.54696312E 02
14	6.500000E 00	0.48586686E 02
15	7.000000E 00	0.43502668E 02
16	7.500000E 00	0.39173083E 02
17	8.000000E 00	0.35408970E 02
18	8.500000E 00	0.32077410E 02
19	9.000000E 00	0.29083729E 02
20	9.500000E 00	0.26359375E 02
21	1.000000E 01	0.23853636E 02
22	1.050000E 01	0.21527951E 02
23	1.100000E 01	0.19351944E 02
24	1.150000E 01	0.17300610E 02
25	1.200000E 01	0.15352215E 02
26	1.250000E 01	0.13486610E 02
27	1.300000E 01	0.11683700E 02
28	1.350000E 01	0.99218618E 01
29	1.400000E 01	0.81760587E 01
30	1.450000E 01	0.64153823E 01
31	1.500000E 01	0.45996534E 01

THE AVERAGE FLUX IN EACH REGION IS

GROUP	REGION	AVERAGE FLUX
1	1	0.26366691E 01
2	1	0.29165519E 02
1	2	0.19404540E 01
2	2	0.27391616E 02

UNIT A = 8.86200000E-01

Z	A	NON 1/V FACTOR
26	58	9.9000000E-01

ISOTOPIC CONCENTRATIONS ARE CALCULATED AT THE FOLLOWING TIMES AND NORMALIZED POWER LEVELS
THE POWER IS NORMALIZED TO A TOTAL FLUX OF 9.9999999E 13

TIME INTERVAL	TIME (HOURS)	POWER LEVEL
0	0.0000000E-39	
1	2.5000000E-01	1.0000000E 00
2	5.0000000E-01	1.0000000E 00
3	7.5000000E-01	1.0000000E 00
4	1.0000000E 00	1.0000000E 00
5	1.2500000E 00	0.0000000E-39
6	1.5000000E 00	0.0000000E-39
7	1.7500000E 00	0.0000000E-39
8	2.0000000E 00	0.0000000E-39
9	2.2500000E 00	2.0000000E 00
10	2.5000000E 00	2.0000000E 00
11	2.7500000E 00	2.0000000E 00
12	3.0000000E 00	2.0000000E 00
13	4.0000000E 00	0.0000000E-39
14	5.0000000E 00	0.0000000E-39
15	6.0000000E 00	0.0000000E-39
16	7.0000000E 00	0.0000000E-39
17	8.0000000E 00	0.0000000E-39
18	9.0000000E 00	0.0000000E-39

THERE ARE 1 ISOTOPES OR ELEMENTS IN REGION 1
THE REGION VOLUME IS 5.00000000E 02 CC
HALF THE MEAN CHORD LENGTH IS 5.00000000E 00 CM
THE TEMPERATURE IS 1.50000000E 02 DEG F.

THE INITIAL ATOM DENSITIES IN REGION 1 ARE

Z	A	ATOM DENSITY (E+24)	CROSS SECTION OPTION
11	23	2.54000000E-03	1

THE ATOM DENSITIES (E+24 ATOMS/CC) IN REGION 1 ARE

CHAIN NUMBER 1

TIME (HOURS)	Z	A	ATOM DENSITY
2.5000000E-01			
	11	23	2.5399997E-03
	11	24	3.1460293E-10
5.0000000E-01			
	11	23	2.5399993E-03
	11	24	6.2559228E-10
7.5000000E-01			
	11	23	2.5399990E-03
	11	24	9.3300956E-10
1.0000000E 00			
	11	23	2.5399987E-03
	11	24	1.2368958E-09
1.2500000E 00			
	11	23	2.5399987E-03
	11	24	1.2226888E-09
1.5000000E 00			
	11	23	2.5399987E-03
	11	24	1.2086449E-09
1.7500000E 00			
	11	23	2.5399987E-03
	11	24	1.1947624E-09
2.0000000E 00			
	11	23	2.5399987E-03
	11	24	1.1810393E-09

2.2500000E 00			
	11	23	2.5399987E-03
	11	24	1.7966792E-09
2.5000000E 00			
	11	23	2.5399987E-03
	11	24	2.4052477E-09
2.7500000E 00			
	11	23	2.5399987E-03
	11	24	3.0068263E-09
3.0000000E 00			
	11	23	2.5399987E-03
	11	24	3.6014950E-09
4.0000000E 00			
	11	23	2.5399987E-03
	11	24	3.4388560E-09
5.0000000E 00			
	11	23	2.5399987E-03
	11	24	3.2835628E-09
6.0000000E 00			
	11	23	2.5399987E-03
	11	24	3.1352816E-09
7.0000000E 00			
	11	23	2.5399987E-03
	11	24	2.9936969E-09
8.0000000E 00			
	11	23	2.5399987E-03
	11	24	2.8585058E-09
9.0000000E 00			
	11	23	2.5399987E-03
	11	24	2.7294198E-09

CHAIN NUMBER 2

TIME (HOURS)	Z	A	ATOM DENSITY
2.5000000E-01			
	11	23	2.5399997E-03
	11	22	7.6414830E-14
5.0000000E-01			
	11	23	2.5399993E-03
	11	22	1.5282907E-13
7.5000000E-01			
	11	23	2.5399990E-03
	11	22	2.2924272E-13
1.0000000E 00			
	11	23	2.5399987E-03
	11	22	3.0565578E-13
1.2500000E 00			
	11	23	2.5399987E-03
	11	22	3.0565346E-13
1.5000000E 00			
	11	23	2.5399987E-03
	11	22	3.0565114E-13
1.7500000E 00			
	11	23	2.5399987E-03
	11	22	3.0564882E-13
2.0000000E 00			
	11	23	2.5399987E-03
	11	22	3.0564650E-13
2.2500000E 00			
	11	23	2.5399987E-03
	11	22	4.5836059E-13

2.5000000E 00			
	11	23	2.5399987E-03
	11	22	6.1107352E-13
2.7500000E 00			
	11	23	2.5399987E-03
	11	22	7.4378530E-13
3.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1649592E-13
4.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1646810E-13
5.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1644028E-13
6.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1641245E-13
7.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1638463E-13
8.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1635681E-13
9.0000000E 00			
	11	23	2.5399987E-03
	11	22	9.1632898E-13

CHAIN NUMBER 3

TIME (HOURS)	7	A	ATOM DENSITY

2.5000000E-01

11	23	2.5399997E-03
10	23	1.2214802E-14

5.0000000E-01

11	23	2.5399993E-03
10	23	1.2214803E-14

7.5000000E-01

11	23	2.5399990E-03
10	23	1.2214802E-14

1.0000000E 00

11	23	2.5399987E-03
10	23	1.2214801E-14

1.2500000E 00

11	23	2.5399987E-03
10	23	2.7957088E-21

1.5000000E 00

11	23	2.5399987E-03
10	23	6.3987844E-28

1.7500000E 00

11	23	2.5399987E-03
10	23	1.4645460E-34

2.0000000E 00

11	23	2.5399987E-03
10	23	0.0000000E-39

2.2500000E 00

11	23	2.5399987E-03
10	23	2.4429595E-14

2.5000000E 00

11	23	2.5399987E-03
10	23	2.4429600E-14

2.75000000E 00			
	11	23	2.5399987E-03
	10	23	2.4429600E-14
3.00000000E 00			
	11	23	2.5399987E-03
	10	23	2.4429600E-14
4.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39
5.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39
6.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39
7.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39
8.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39
9.00000000E 00			
	11	23	2.5399987E-03
	10	23	0.0000000E-39

CHAIN NUMBER 4

TIME (HOURS)	Z	A	ATOM DENSITY
2.50000000E-01			
	11	23	2.5399997E-03
	9	20	6.0310599E-15

5.0000000E-01
11 23 2.5399993E-03
9 20 6.0310592E-15

7.5000000E-01
11 23 2.5399990E-03
9 20 6.0310584E-15

1.0000000E 00
11 23 2.5399987E-03
9 20 6.0310576E-15

1.2500000E 00
11 23 2.5399987E-03
9 20 4.3003575E-39

1.5000000E 00
11 23 2.5399987E-03
9 20 0.0000000E-39

1.7500000E 00
11 23 2.5399987E-03
9 20 0.0000000E-39

2.0000000E 00
11 23 2.5399987E-03
9 20 0.0000000E-39

2.2500000E 00
11 23 2.5399987E-03
9 20 1.2062115E-14

2.5000000E 00
11 23 2.5399987E-03
9 20 1.2062115E-14

2.7500000E 00
11 23 2.5399987E-03
9 20 1.2062115E-14

3.0000000E 00			
	11	23	2.5399987E-03
	9	20	1.2062115E-14
4.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39
5.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39
6.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39
7.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39
8.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39
9.0000000E 00			
	11	23	2.5399987E-03
	9	20	0.0000000E-39

THE PHOTON SOURCE STRENGTH IN REGION 1 IS

TIME(HOURS)	ENERGY GROUP	SOURCE STRENGTH DENSITY (PHOTONS/CC-SEC)	ENERGY DENSITY (PHOTONS-MEV/CC-SEC)	GAMMA SOURCE STRENGTH (PHOTONS/SEC)	ENERGY SOURCE STRENGTH (PHOTONS-MEV/SEC)
2.5000000E-01	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	8.4553695E 00	1.6910739E 10	4.2276847E 12	8.4553695E 12
	3	2.0129140E 08	1.1071027E 08	1.0064570E 11	5.5355135E 10
5.0000000E-01	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	1.6439157E 10	3.2878315E 10	8.2195787E 12	1.6439157E 13
	3	2.0129142E 08	1.1071028E 08	1.0064571E 11	5.5355140E 10
7.5000000E-01	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	2.4331242E 10	4.8662485E 10	1.2165621E 13	2.4331242E 13
	3	2.0129140E 08	1.1071027E 08	1.0064570E 11	5.5355134E 10
1.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	3.2132678E 10	6.4265356E 10	1.6066339E 13	3.2132678E 13
	3	2.0129137E 08	1.1071025E 08	1.0064569E 11	5.5355127E 10
1.2500000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	3.1389139E 10	6.2778279E 10	1.5694570E 13	3.1389139E 13
	3	4.6071326E 01	2.5339229E 01	2.3035663E 04	1.2669614E 04
1.5000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	3.1028603E 10	6.2057206E 10	1.5514301E 13	3.1028603E 13
	3	1.0544749E-05	5.7996120E-06	5.2723746E-03	2.8998060E-03

1.7500000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	3.0672208E 10	6.1344415E 10	1.5336104E 13	3.0672207E 13
	3	2.4134695E-12	1.3274082E-12	1.2067347E-09	6.6370410E-10
2.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	3.0319906E 10	6.0639811E 10	1.5159953E 13	3.0319905E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
2.2500000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	4.6882374E 10	9.3764747E 10	2.3441187E 13	4.6882374E 13
	3	4.0258264E 08	2.2142045E 08	2.0129132E 11	1.1071023E 11
2.5000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	6.2505682E 10	1.2501136E 11	3.1252841E 13	6.2505681E 13
	3	4.0258274E 08	2.2142050E 08	2.0129137E 11	1.1071025E 11
2.7500000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	7.7949541E 10	1.5589908E 11	3.8974770E 13	7.7949541E 13
	3	4.0258274E 08	2.2142050E 08	2.0129137E 11	1.1071025E 11
3.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	9.3216011E 10	1.8643202E 11	4.6608005E 13	9.3216010E 13
	3	4.0258274E 08	2.2142050E 08	2.0129137E 11	1.1071025E 11
4.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	8.8283096E 10	1.7656619E 11	4.4141548E 13	8.8283096E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
5.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	8.4296359E 10	1.6859272E 11	4.2148179E 13	8.4296358E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
6.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	8.0489655E 10	1.6097931E 11	4.0244827E 13	8.0489654E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
7.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	7.6854857E 10	1.5370971E 11	3.8427428E 13	7.6854856E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
8.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	7.3384201E 10	1.4676840E 11	3.6692100E 13	7.3384200E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
9.0000000E 00	1	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39
	2	7.0070276E 10	1.4014055E 11	3.5035138E 13	7.0070276E 13
	3	0.0000000E-39	0.0000000E-39	0.0000000E-39	0.0000000E-39

THE GAMMA DOSE RATE AT R = 1.1250000E 02 CM FROM REGION 1 IS

TIME(HOURS)	DOSE RATE (RAD/HR)
2.5000000E-01	7.8710556E 01
5.0000000E-01	1.5242622E 02
7.5000000E-01	2.2529517E 02
1.0000000E 00	2.9732714E 02
1.2500000E 00	2.8982123E 02
1.5000000E 00	2.8649233E 02
1.7500000E 00	2.8320168E 02
2.0000000E 00	2.7994881E 02
2.2500000E 00	4.3415428E 02
2.5000000E 00	5.7840692E 02
2.7500000E 00	7.2100769E 02
3.0000000E 00	8.6196059E 02
4.0000000E 00	8.1513274E 02
5.0000000E 00	7.7832253E 02
6.0000000E 00	7.4317458E 02
7.0000000E 00	7.0961388E 02
8.0000000E 00	6.7756873E 02
9.0000000E 00	6.4697070E 02

THE GAMMA DOSE AT R = 1.1250000E 02 CM FROM SOURCE REGION 1 IS

INITIAL TIME(HOURS)	FINIAL TIME(HOURS)	DOSE(RAD)
1.0000000E 00	2.0000000E 00	2.8703830E 02
3.0000000E 00	9.0000000E 00	4.4782780E 03

THERE ARE 1 ISOTOPES OR ELEMENTS IN REGION 2
THE REGION VOLUME IS 1.000000000E 03 CC
HALF THE MEAN CHORD LENGTH IS 1.000000000E 01 CM
THE TEMPERATURE IS 1.500000000E 02 DEG F.

THE INITIAL ATOM DENSITIES IN REGION 2 ARE

Z	A	ATOM DENSITY (E+24)	CROSS SECTION OPTION
26	54	4.9523199E-04	1
26	56	7.7744639E-03	1
26	57	1.8401600E-04	1
26	58	2.6288000E-05	1

THE ATOM DENSITIES (E+24 ATOMS/CC) IN REGION 2 ARE

CHAIN NUMBER 1

TIME (HOURS)	Z	A	ATOM DENSITY
2.5000000E-01	26	54	4.9523169E-04
	25	54	4.2817846E-13
5.0000000E-01	26	54	4.9523138E-04
	25	54	8.5634681E-13
7.5000000E-01	26	54	4.9523108E-04
	25	54	1.2845050E-12
1.0000000E 00	26	54	4.9523077E-04
	25	54	1.7126532E-12
1.2500000E 00	26	54	4.9523077E-04
	25	54	1.7126138E-12
1.5000000E 00	26	54	4.9523077E-04
	25	54	1.7125744E-12
1.7500000E 00	26	54	4.9523077E-04
	25	54	1.7125350E-12
2.0000000E 00	26	54	4.9523077E-04
	25	54	1.7124956E-12

2.2500000E 00			
	26	54	4.9523077E-04
	25	54	2.5686993E-12
2.5000000E 00			
	26	54	4.9523077E-04
	25	54	3.4248834E-12
2.7500000E 00			
	26	54	4.9523077E-04
	25	54	4.2810477E-12
3.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1371923E-12
4.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1367197E-12
5.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1362471E-12
6.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1357745E-12
7.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1353020E-12
8.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1348295E-12
9.0000000E 00			
	26	54	4.9523077E-04
	25	54	5.1343571E-12

CHAIN NUMBER 1

TIME (HOURS)	Z	A	ATOM DENSITY
2.5000000E-01			
	26	56	7.7744593E-03
	25	56	6.5276121E-13
5.0000000E-01			
	26	56	7.7744547E-03
	25	56	1.2629059E-12
7.5000000E-01			
	26	56	7.7744501E-03
	25	56	1.8332163E-12
1.0000000E 00			
	26	56	7.7744455E-03
	25	56	2.3662931E-12
1.2500000E 00			
	26	56	7.7744455E-03
	25	56	2.2118075E-12
1.5000000E 00			
	26	56	7.7744455E-03
	25	56	2.0674076E-12
1.7500000E 00			
	26	56	7.7744455E-03
	25	56	1.9324349E-12
2.0000000E 00			
	26	56	7.7744455E-03
	25	56	1.8062741E-12
2.2500000E 00			
	26	56	7.7744455E-03
	25	56	2.9938695E-12

TIME (HOURS)	Z	A	ATOM DENSITY
2.5000000E 00	26	56	7.7744455E-03
	25	56	4.1039316E-12
2.7500000E 00	26	56	7.7744455E-03
	25	56	5.1415222E-12
3.0000000E 00	26	56	7.7744455E-03
	25	56	6.1113727E-12
4.0000000E 00	26	56	7.7744455E-03
	25	56	4.6650240E-12
5.0000000E 00	26	56	7.7744455E-03
	25	56	3.5609756E-12
6.0000000E 00	26	56	7.7744455E-03
	25	56	2.7182169E-12
7.0000000E 00	26	56	7.7744455E-03
	25	56	2.0749098E-12
8.0000000E 00	26	56	7.7744455E-03
	25	56	1.5838510E-12
9.0000000E 00	26	56	7.7744455E-03
	25	56	1.2090088E-12

CHAIN NUMBER 1

2.5000000E-01			
	26	58	2.6287994E-05
	26	59	5.7550013E-12
5.0000000E-01			
	26	58	2.6287985E-05
	26	59	1.1509081E-11
7.5000000E-01			
	26	58	2.6287983E-05
	26	59	1.7262239E-11
1.0000000E 00			
	26	58	2.6287977E-05
	26	59	2.3014475E-11
1.2500000E 00			
	26	58	2.6287977E-05
	26	59	2.3010794E-11
1.5000000E 00			
	26	58	2.6287977E-05
	26	59	2.3007114E-11
1.7500000E 00			
	26	58	2.6287977E-05
	26	59	2.3003434E-11
2.0000000E 00			
	26	58	2.6287977E-05
	26	59	2.2999754E-11
2.2500000E 00			
	26	58	2.6287977E-05
	26	59	3.4505863E-11
2.5000000E 00			
	26	58	2.6287977E-05
	26	59	4.6010131E-11

2.7500000E 00			
	26	58	2.6287977E-05
	26	59	5.7512559E-11
3.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.9013147E-11
4.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8969004E-11
5.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8924884E-11
6.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8880803E-11
7.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8836744E-11
8.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8792715E-11
9.0000000E 00			
	26	58	2.6287977E-05
	26	59	6.8748712E-11

THE PHOTON SOURCE STRENGTH IN REGION 2 IS

TIME(HOURS)	ENERGY GROUP	SOURCE STRENGTH DENSITY (PHOTONS/CC-SEC)	ENERGY DENSITY (PHOTONS-MEV/CC-SEC)	GAMMA SOURCE STRENGTH (PHOTONS/SEC)	ENERGY SOURCE STRENGTH (PHOTONS-MEV/SEC)
2.5000000E-01	1	9.7935372E 04	4.4070917E 05	9.7935372E 07	4.4070917E 08
	2	2.4135588E 07	4.8271176E 07	2.4135588E 10	4.8271176E 10
	3	4.9009314E 07	2.6955123E 07	4.9009314E 10	2.6955123E 10
5.0000000E-01	1	1.8947688E 05	8.5264596E 05	1.8947688E 08	8.5264596E 08
	2	4.6762061E 07	9.3524123E 07	4.6762061E 10	9.3524123E 10
	3	9.4821693E 07	5.2151930E 07	9.4821692E 10	5.2151930E 10
7.5000000E-01	1	2.7504196E 05	1.2376888E 06	2.7504196E 08	1.2376888E 09
	2	6.7977932E 07	1.3595586E 08	6.7977932E 10	1.3595586E 11
	3	1.3764585E 08	7.5705217E 07	1.3764585E 11	7.5705216E 10
1.0000000E 00	1	3.5502079E 05	1.5975935E 06	3.5502079E 08	1.5975935E 09
	2	8.7875285E 07	1.7575057E 08	8.7875285E 10	1.7575057E 11
	3	1.7767688E 08	9.7722281E 07	1.7767688E 11	9.7722281E 10
1.2500000E 00	1	3.3184292E 05	1.4932931E 06	3.3184292E 08	1.4932931E 09
	2	8.2404655E 07	1.6480931E 08	8.2404654E 10	1.6480931E 11
	3	1.6608792E 08	9.1348357E 07	1.6608792E 11	9.1348356E 10
1.5000000E 00	1	3.1017825E 05	1.3958021E 06	3.1017824E 08	1.3958021E 09
	2	7.7291137E 07	1.5458227E 08	7.7291136E 10	1.5458227E 11
	3	1.5525557E 08	8.5390560E 07	1.5525556E 11	8.5390559E 10

1.7500000E 00	1	2.8992797E 05	1.3046759E 06	2.8992797E 08	1.3046759E 09
	2	7.2511418E 07	1.4502284E 08	7.2511417E 10	1.4502283E 11
	3	1.4513041E 08	7.9821723E 07	1.4513041E 11	7.9821723E 10
2.0000000E 00	1	2.7099975E 05	1.2194989E 06	2.7099975E 08	1.2194989E 09
	2	6.8043705E 07	1.3608741E 08	6.8043705E 10	1.3608741E 11
	3	1.3566628E 08	7.4616452E 07	1.3566628E 11	7.4616451E 10
2.2500000E 00	1	4.4917761E 05	2.0212992E 06	4.4917761E 08	2.0212992E 09
	2	1.1213867E 08	2.2427734E 08	1.1213867E 11	2.2427734E 11
	3	2.2483844E 08	1.2366114E 08	2.2483844E 11	1.2366114E 11
2.5000000E 00	1	6.1572296E 05	2.7707533E 06	6.1572296E 08	2.7707533E 09
	2	1.5348803E 08	3.0697606E 08	1.5348803E 11	3.0697606E 11
	3	3.0819433E 08	1.6950688E 08	3.0819433E 11	1.6950688E 11
2.7500000E 00	1	7.7139524E 05	3.4712786E 06	7.7139524E 08	3.4712786E 09
	2	1.9227103E 08	3.8454205E 08	1.9227102E 11	3.8454205E 11
	3	3.8611369E 08	2.1236253E 08	3.8611369E 11	2.1236252E 11
3.0000000E 00	1	9.1690431E 05	4.1260694E 06	9.1690431E 08	4.1260694E 09
	2	2.2865517E 08	4.5731035E 08	2.2865517E 11	4.5731035E 11
	3	4.5895142E 08	2.5242328E 08	4.5895142E 11	2.5242328E 11
4.0000000E 00	1	6.9990505E 05	3.1495727E 06	6.9990504E 08	3.1495727E 09
	2	1.7743550E 08	3.5487100E 08	1.7743550E 11	3.5487100E 11
	3	3.5045154E 08	1.9274835E 08	3.5045154E 11	1.9274835E 11
5.0000000E 00	1	5.3426194E 05	2.4041787E 06	5.3426194E 08	2.4041787E 09
	2	1.3833589E 08	2.7667178E 08	1.3833589E 11	2.7667178E 11
	3	2.6762975E 08	1.4719636E 08	2.6762974E 11	1.4719636E 11
6.0000000E 00	1	4.0782078E 05	1.8351935E 06	4.0782078E 08	1.8351935E 09
	2	1.0848794E 08	2.1697588E 08	1.0848794E 11	2.1697589E 11
	3	2.0440892E 08	1.1242490E 08	2.0440892E 11	1.1242490E 11
7.0000000E 00	1	3.1130383E 05	1.4008672E 06	3.1130383E 08	1.4008672E 09
	2	8.5702110E 07	1.7140422E 08	8.5702109E 10	1.7140422E 11
	3	1.5615020E 08	8.5882607E 07	1.5615019E 11	8.5882607E 10
8.0000000E 00	1	2.3762907E 05	1.0693308E 06	2.3762907E 08	1.0693308E 09
	2	6.8307042E 07	1.3661408E 08	6.8307042E 10	1.3661408E 11
	3	1.1931257E 08	6.5621914E 07	1.1931257E 11	6.5621914E 10
9.0000000E 00	1	1.8139056E 05	8.1625752E 05	1.8139056E 08	8.1625751E 08
	2	5.5026933E 07	1.1005387E 08	5.5026933E 10	1.1005387E 11
	3	9.1193069E 07	5.0156187E 07	9.1193069E 10	5.0156187E 10

THE GAMMA DOSE RATE AT R = 1.0500000E 02 CM FROM REGION 2 IS

TIME(HOURS)	DOSE RATE (RAD/HR)
2.5000000E-01	8.7344588E-01
5.0000000E-01	1.6912995E 00
7.5000000E-01	2.4571898E 00
1.0000000E 00	3.1745094E 00
1.2500000E 00	2.9729849E 00
1.5000000E 00	2.7846163E 00
1.7500000E 00	2.6085446E 00
2.0000000E 00	2.4439670E 00
2.2500000E 00	4.0370204E 00
2.5000000E 00	5.5289326E 00
2.7500000E 00	6.9263065E 00
3.0000000E 00	8.2353135E 00
4.0000000E 00	6.3485343E 00
5.0000000E 00	4.9082504E 00
6.0000000E 00	3.8087917E 00
7.0000000E 00	2.9694970E 00
8.0000000E 00	2.3287944E 00
9.0000000E 00	1.8396841E 00

THE GAMMA DOSE AT R = 1.0500000E 02 CM FROM SOURCE REGION 2 IS

INITIAL TIME(HOURS)	FINAL TIME(HOURS)	DOSE(RAD)
1.0000000E 00	2.0000000E 00	2.7938460E 00
3.0000000E 00	9.0000000E 00	2.5401366E 01