325
5/14/64

NAA-SR-9233
COPY

QUICKIE –
A COMPUTER PROGRAM FOR SPATIALLY INDEPENDENT MULTIGROUP SLOWING-DOWN AND THERMALIZATION CALCULATIONS

AEC Research and Development Report

ATOMICS INTERNATIONAL
A DIVISION OF NORTH AMERICAN AVIATION, INC.

Price $1.25
Available from the Office of Technical Services
Department of Commerce
Washington 25, D. C.

NAA-SR-9233
MATHEMATICS AND
COMPUTERS
51 PAGES

QUICKIE –
A COMPUTER PROGRAM FOR SPATIALLY INDEPENDENT MULTIGROUP SLOWING-DOWN AND THERMALIZATION CALCULATIONS

By

M. BOLING
W. RHOADES

ATOMICS INTERNATIONAL

A DIVISION OF NORTH AMERICAN AVIATION, INC.
P.O. BOX 309 CANOGA PARK, CALIFORNIA

CONTRACT: AT(11-1)-GEN-8
ISSUED: APR 15 1964

DISTRIBUTION

This report has been distributed according to the category "Mathematics and Computers" as given in "Standard Distribution lists for Unclassified Scientific and Technical Reports" TID-4500 (26th Ed.) January 1, 1964. A total of 650 copies was printed.

ACKNOWLEDGMENT

The authors wish to acknowledge the assistance of W. J. Roberts, who suggested the method of solution for thermal fluxes; of G. A. Long, who assisted in programming and reporting an earlier version of QUICKIE; and of M. Hoffman, who developed the FLOPUN subroutine.

CONTENTS

ABSTRACT

In the case of an infinite homogeneous medium or a finite medium where buckling is known through calculation or physical considerations, the multigroup diffusion equations can be converted to spatially independent equations involving only integrated fluxes. These equations are quite simple to solve, giving the neutron energy spectrum, multiplication factor, and total reaction rates.

QUICKIE performs these calculations as well as calculating few-group cross sections and punching libraries for the spatially dependent diffusion and transport codes. Its advantages are simplicity of input data requirements and economy of operation.

I. INTRODUCTION

QUICKIE is a computer program designed to solve the multigroup neutron slowing down and thermalization equations without consideration of spatial dimensions. This calculation does not provide information about spatial distribution of neutron density, but it does provide an accurate neutron energy spectrum in the case of infinite homogeneous media, and also for a limited class of finite homogeneous media.

The effect of leakage, in the case of a finite medium, is included by means of an effective buckling. In the case of a bare core with flux vanishing at a single boundary, the geometric buckling is supplied as input data. If the flux is considered to vanish at the group dependent extrapolated boundaries, the core dimensions are supplied, and the bucklings are calculated internally. Homogeneous reactors reflected in one dimension are also satisfactorily treated using a reflector savings or effective buckling known from previous calculations.

The uses of QUICKIE include the study of thermal spectrum temperature coefficients, the effect of thermal poisons and fuel changes, and certain approximate criticality surveys involving size and composition. The program provides a neutron energy spectrum, fission, capture, and leakage spectra, few-group cross sections, and a few-group neutron economy table. In addition, energy and lethargy plots of the flux spectrum are provided. The cross-section data are read from tapes used with spatially dependent codes. The version of QUICKIE listed in the appendix uses the ULCER library tape,[1] and a version to use the DTF (FORTRAN DTK) tape is planned.

The advantages of QUICKIE are simplicity and economy. It may require as few as two cards of input data for a problem, as compared with 20 to 60 for a one-dimensional code. It requires no more than 6 sec of computer time for a calculation, as compared to 6 min for the corresponding one-dimensional code. For problems which can be treated satisfactorily by QUICKIE, these make it an attractive alternative to the spatial treatment.

II. METHOD

A. BASIC INTEGRAL FLUX EQUATIONS

The multigroup diffusion equations for a homogeneous reactor operating at steady state can be written:[2]

$$\Sigma_a^i \varphi^i(\bar{r}) - D^i \nabla^2 \varphi^i(\bar{r}) + \sum_j \varphi^i(\bar{r}) \Sigma_{ji} = \frac{1}{\lambda} \sum_N \chi_N^i \sum_j (\nu \Sigma_f)_N^j \varphi^j(\bar{r}) + \sum_j \varphi^j(\bar{r}) \Sigma_{ij} \quad , \qquad \ldots (1a)$$

$$\lambda = \int \sum_N \sum_j (\nu \Sigma_f)_N^j \varphi^j(\bar{r}) d\bar{r} \quad , \qquad \ldots (1b)$$

where

$\bar{r}$ = spatial variable

Σ_a^i = absorption cross section, ith energy group

D^i = diffusion coefficient, ith energy group

φ^i = neutron flux, ith energy group

$(\nu \Sigma_f)_N^i$ = fission cross section, ith energy group, N_{th} element

χ_N^i = fission source spectrum, ith energy group, N_{th} element, normalized to 1 neutron

Σ_{ji} = scattering cross section, group i to group j

λ = multiplication factor.

If we integrate this over the reactor volume, we obtain the reaction rate for the system:

$$\Sigma_a^i \int \varphi^i(\bar{r}) d\bar{r} - D^i \int \nabla^2 \varphi^i(\bar{r}) d\bar{r} + \sum_j \Sigma_{ji} \int \varphi^i(\bar{r}) d\bar{r} = \frac{1}{\lambda} \sum_N \chi_N^i \sum_j (\nu \Sigma_f)_N^j \int \varphi^j(\bar{r}) d\bar{r} + \sum_j \Sigma_{ij} \int \varphi^j(\bar{r}) d\bar{r} \quad . \qquad \ldots (2)$$

Defining:

$$\Phi^i \equiv \int \varphi^i(\bar{r})d\bar{r} \qquad \text{(integral flux)} \quad , \qquad \ldots(3)$$

and:

$$(B^2)^i = \frac{-\int \nabla^2 \varphi^i(\bar{r})d\bar{r}}{\int \varphi^i(\bar{r})d\bar{r}} \qquad \text{(effective buckling)} \quad , \qquad \ldots(4)$$

we have:

$$\Sigma_a^i \Phi^i + D^i(B^2)^i \Phi^i + \sum_j \Sigma_{ji} \Phi^i = \frac{1}{\lambda} \sum_N \chi_N^i \sum_j (\nu\Sigma_f)_N^j \Phi^j + \sum_j \Sigma_{ij} \Phi^j \quad . \qquad \ldots(5)$$

Now defining:

$$\Sigma_T^i \equiv \Sigma_a^i + D^i(B^2)^i + \sum_{j \neq i} \Sigma_{ji} \qquad \text{(total cross section)} \quad , \qquad \ldots(6)$$

$$S^i \equiv \frac{1}{\lambda} \sum_N \chi_N^i \sum_j (\nu\Sigma_f)_N^j \Phi^j \qquad \text{(group source)} \quad , \qquad \ldots(7)$$

the equations are:

$$\Sigma_T^i \Phi^i = S^i + \sum_{j \neq i} \Sigma_{ij} \Phi^j \qquad \begin{array}{l} i = 1, 2, \ldots N_g \\ j = 1, 2, \ldots N_g \end{array} \quad . \qquad \ldots(8)$$

These are a set of N_g simultaneous equations which completely determine the integral fluxes, providing $(B^2)^i$ is known. Since λ involves the flux spectrum, and is used in calculating the source, the set cannot easily be solved explicitly. An iterative process, starting with a guess of S^i, calculating Φ^i and λ, and then repeating the calculation of S^i is used. This converges in about three iterations, and is not objectionable.

B. SINGLE FISSIONING ISOTOPE

In the case where a single typical fission source is adequate, the fission source supplied as a starting guess (at the beginning of the ULCER tape) is used, and the flux solution is made using this starting guess for S^i, and calculating Φ^i directly. No iterations are required in this case.

C. SPECTRAL DATA

The differential flux, fission, absorption, and leakage spectra are calculated directly from the flux and the energy interval, $(\Delta E)^i$, for the ith energy group:

$$p^i \equiv \Phi^i/(\Delta E)^i \qquad \text{(flux spectrum)} \quad , \qquad \ldots (9a)$$

$$f^i = p^i (\nu\Sigma_f)^i \qquad \text{(fission spectrum)} \quad , \qquad \ldots (9b)$$

$$a^i = p^i \Sigma_a^i \qquad \text{(absorption spectrum)} \quad , \qquad \ldots (9c)$$

$$\ell^i = p^i D^i (B^2)^i \qquad \text{(leakage spectrum)} \quad . \qquad \ldots (9d)$$

D. FEW-GROUP NEUTRON ECONOMY

The few-group neutron economy data are calculated using a set of one to six-group breakpoints, M_1, M_2, $\cdots$, M_n which refer to the last multigroup to be included in each few group. The breakpoint M_o is understood to be O, and M_n is normally the last multigroup. The few-group flux, fission, absorption and leakage are calculated:

$$P^n = \sum_{i=M_{n-1}+1}^{M_n} \Phi^i \qquad \text{(few group flux)} \quad , \qquad \ldots (10a)$$

$$F^n = \sum_{i=M_{n-1}+1}^{M_n} \Phi^i (\nu\Sigma_f)^i \qquad \text{(few group fission)} \quad , \qquad \ldots (10b)$$

$$A^n = \sum_{i=M_{n-1}+1}^{M_n} \Phi^i \Sigma_a^i \qquad \text{(few group absorption)} \quad , \qquad \ldots (10c)$$

$$L^n = \sum_{i=M_{n-1}+1}^{M_n} \Phi^i D^i (B^2)^i \qquad \text{(few group leakage)} \quad . \qquad \ldots (10d)$$

E. FEW-GROUP CROSS SECTIONS

The few-group cross sections are then calculated by direct flux averaging:

$$(\overline{\nu\Sigma_f})^n = F^n / P^n \quad , \qquad \text{...(11a)}$$

$$(\overline{\Sigma}_a)^n = A^n / P^n \quad , \qquad \text{...(11b)}$$

$$(\overline{DB^2})^n = L^n / P^n \quad , \qquad \text{...(11c)}$$

$$(\overline{D})^n = L^n \Big/ \sum_{i=M_{n-1}+1}^{M_n} \Phi^i \left|\left(B^2\right)\right|^i \quad , \qquad \text{...(11d)}$$

$$(B^2)^n = (DB^2)^n \Big/ (\overline{D})^n \quad , \qquad \text{...(11e)}$$

$$\Sigma_{nn'} = \sum_{i=M_{n'-1}+1}^{M_{n'}} \Phi^i \sum_{j=M_{n-1}+1}^{M_n} \Sigma_{ji} \; / P^n \quad . \qquad \text{...(11f)}$$

III. METHOD OF SOLUTION

A. MACROSCOPIC CROSS-SECTION COMPILATION

The solution of the first problem starts by reading from the library tape the group structure information and an initial guess for the source spectrum. The first element on the library tape is read, and the list of elements required to form the cross sections is scanned to see if the element is needed. If so, the microscopic cross sections are multiplied by the number density to form macroscopic cross sections. If not, the next element is read and treated similarly. Each element on the tape is examined, its contribution to the macroscopic cross section, if any, being added to those previously found, until all the required elements have been located. Then the tape is immediately rewound for later use. This system is about as efficient as any, and in addition is quite simple.

The total cross section Σ_T^i and diffusion coefficient are constructed from the macroscopic data, and the scatter cross sections, Σ_{ji} are assembled.

The source, S^i, is set equal to the initial source guess.

B. FLUX SOLUTION

The solution for the fluxes (Equation 8) can now be represented in matrix form:

$$\begin{bmatrix} -\Sigma_T^1 & \Sigma_{1,2} & \cdots & \Sigma_{1,Ng} \\ \Sigma_{2,1} & -\Sigma_T^2 & & \vdots \\ \vdots & & \ddots & \vdots \\ \Sigma_{Ng,1} & \cdots & \cdots & \Sigma_{Ng,Ng} \end{bmatrix} \begin{bmatrix} \Phi^1 \\ \vdots \\ \vdots \\ \Phi^{Ng} \end{bmatrix} = - \begin{bmatrix} S^1 \\ \vdots \\ \vdots \\ S^{Ng} \end{bmatrix} \qquad \ldots (12)$$

In practice, upscatter to groups of index less than a limit N_{th} is prohibited by arranging the library such that:

$$\Sigma_{ji} = 0 \text{ for } \left\{ \begin{array}{r} j < N_{th} \\ \text{and } j < i \end{array} \right\}$$

The solution for T is thus reduced to:

$$\begin{bmatrix} -\Sigma_T^1 & 0 & \cdots & & 0 \\ \Sigma_{2,1} & \ddots & \ddots & 0 & \vdots \\ \vdots & & \ddots & & \\ \Sigma_{Nth,1} & \cdots & -\Sigma_T^{Nth} & \cdots & \Sigma_{Nth,Ng} \\ \vdots & & \vdots & \ddots & \vdots \\ \Sigma_{Ng,1} & \cdots & \Sigma_{Ng,Nth} & \cdots & -\Sigma_T^{Ng} \end{bmatrix} \begin{bmatrix} \Phi^1 \\ \vdots \\ \Phi^{Nth} \\ \vdots \\ \Phi^{Ng} \end{bmatrix} = - \begin{bmatrix} S^1 \\ \vdots \\ S^{Nth} \\ \vdots \\ S^{Ng} \end{bmatrix} , \qquad \ldots(13)$$

The solution for $\Phi^1 \cdots \Phi^{N_{th}-1}$ can now be made explicitly:

$$\Phi^1 = S^1/\Sigma_T^1 , \qquad \ldots(14a)$$

$$\Phi^i = \left[S^i + \sum_{j=1}^{i-1} \Phi^j \Sigma_{ij} \right] \Big/ \Sigma_T^i \quad 1 < i < N_{th} . \qquad \ldots(14b)$$

A thermal source is then constructed:

$$S_{th}^i = S^i + \sum_{j=1}^{N_{th}-1} \Phi^j \Sigma_{ij} \qquad N_{th} \le i \le N_g , \qquad \ldots(15)$$

and the thermal fluxes are found by:

$$\begin{bmatrix} -\Sigma_T^{Nth} & \Sigma_{Nth,\ Nth+1} & -- & \Sigma_{Nth,\ Ng} \\ \Sigma_{Nth+1,\ Nth} & -\Sigma_T^{Nth+1} & & | \\ | & & \ddots & | \\ \Sigma_{Ng,\ Nth} & -- & -- & -\Sigma_T^{Ng} \end{bmatrix} \begin{bmatrix} \Phi^{Nth} \\ | \\ | \\ \Phi^{Ng} \end{bmatrix} = - \begin{bmatrix} S_{Th}^{Nth} \\ | \\ | \\ S_{Th}^{Ng} \end{bmatrix} \qquad \ldots (16)$$

This is solved by a standard library routine for Φ^i.

This completes the flux solution, and λ is then calculated by Equation 1b. If the "chi-option" is not requested, the program performs the final edit using Equations 9, 10, and 11, and calls in the next problem. If the "chi-option" is requested, the source S^i is recomputed by Equation 7, and the flux solution and λ calculation repeated until convergence is reached on the ith iteration, according to:

$$\left| \frac{\lambda_i - \lambda_{i-1}}{\lambda_i} \right| < 0.00001 \quad .$$

When convergence is reached, the final edit is taken as usual.

C. SPECIAL CALCULATIONS

After the edit is completed, the source spectrum is replaced with the leakage spectrum. If the "reflector option" is requested on the next problem, this leakage spectrum will be used in lieu of the usual fission source spectrum. Thus, by supplying the composition of a reflector material for the next problem, a reflector spectrum calculation, useful to obtain reflector few-group cross sections, can be performed.

An option is provided to read in part or all of the fluxes in lieu of the usual flux calculation. This is useful for certain shielding cross-section calculations. Another option allows punching of multigroup or few-group libraries for various computer codes. The use of these options is covered in the Data Input Description section of this report.

IV. DATA INPUT DESCRIPTION

The QUICKIE program uses a library which gives microscopic cross sections for various elements used. Supplying the proper QUICKIE data along with this library, it is possible to perform a multigroup flux calculation for a one-region system. QUICKIE can also do a few-group cross-section calculation, a reflector spectrum calculation, an elemental fission spectrum calculation (chi-option), and gives CRT output.

The CRT output includes a histogram of $\Phi/\Delta u$ vs lethargy and a histogram of $\Phi/\Delta E$ vs energy for the thermal groups.

The following input is required.

A. CONTROL CARD

Col: 1-60 Title — Alphanumeric information

Col: 61-72 IC(1) through IC(6) in 6I2 format. This is control data and is used as follows:

IC(1) = -1: flux read from cards. A set of group-wise fluxes can be read from cards to enable the user to use his own flux spectrum. A blank or 0 flux for a group will cause the flux calculated in the previous problem to be used for that group.

= 0: the code will calculate its own flux

= 1: the flux from the previous case will be used for all groups

= 2: does a reflector spectrum calculation

= 3: the chi-option will be used

IC(2) = -1: punches the macroscopic cross sections onto cards which can be used to make an ULCER multigroup library

= 0: no punch

= +1: calculates few-group data (≤ 6) - reads new few-group structure

= +2: calculates few-group data (≤ 6) - uses previous few-group structure

IC(3) = -N: read over only those materials which are changing on the first N material cards, ignoring blank element names.

= 0: use previous material

= N: read N material cards. No element names from the previous case are retained.

IC(4) = 0: code will set $B^2 = 1.0 \times 10^{-20}$

= +1: read in a single B^2

= +2: read in groupwise B^2

= +3: use B^2 from previous problem

= -1: code will calculate B^2 for a plane geometry

= -2: code will calculate B^2 for a cylinder geometry

= -3: code will calculate B^2 for a spherical geometry

IC(5) = 0:D will be calculated as $1/(3\ \Sigma_{tr})$

= 1:D will be calculated as $1/3(1-0.4\ \Sigma_a/\Sigma_{tr})^2\ \Sigma_{tr}$

IC(6) = -1: plots Φ/Δ uvs lethargy (all groups) and Φ/ΔE vs energy (for the thermal groups only)

= 0: no plots

= +1: plots only Φ/ΔE vs energy (thermal groups)

B. MATERIAL CARDS

There will be N material cards which include the element names and concentrations. The format for these cards is 4(A6, F12.0). If IC(3) is negative, blank element names on the card will not change the element names previously appearing in those positions, and information in positions not specified will be retained. If IC(3) is positive, blanks will remove the previous element names, and information in positions not specified will be removed. If IC(3) = 0, no material cards are read and the previous element list is used.

C. BUCKLING CARDS

If IC(4) = 0 or 3 no B^2 card is read.

= +1: read in one B^2 card with group-independent B^2

= +2: read in one B^2 for each group in 4(6X, F12.0) format. Since there can be 4 per card, there will be $N_g/4$ cards read.

If IC(4) is negative, only one card is required. The first piece of data is the calculated buckling option. If this value is 0, the code will calculate the standard buckling.[2] If this value is 1, the code calculates a curvature-corrected buckling.[2] For a plane case, the X, Y, and Z dimensions are required. For a cylindrical case, the radius and height dimensions are required. For the spherical case, the radius is required. All of the data in this section is read on the 4(6X, F12.0) format.

D. FLUX CARDS

If the fluxes are to be read from cards they are entered here. A flux for each group is read and the format used is 4(6X, F12.0).

E. GROUP REDUCTION CARD

This card is required only if IC(2) = +1. Two to 6 few-groups may be specified. The last multigroup to be included in each few group is entered in columns 1-36 in 6I6 format, followed by the punch option in the I6 format.

This option enables the user to get few-group data punched on cards in the various library formats used by FAIM, ULCER, and FLOCO. In the case of ULCER, a value for N_{th} (number of first thermal group) is required. This is also in the I6 format.

Col. 1-6	7-12	13-18	19-24	25-30	31-36	37-42	43-48
← few group limits →						punch option = 0 no punch = 1 FAIM = 2 ULCER = 3 FLOCO	N_{th} for ULCER punch

F. LIBRARY TAPE

A cross section tape prepared by the ULCER library program[1] must be mounted as logical tape 9.

REFERENCES

1. D. Vargofcak, "The ULCER Code - A Multigroup, One-Dimensional, Diffusion Equation Code with Upscatter," NAA-SR-7146 (1962)

2. D.C. Baller, "The FAIM Code - A Multigroup One-Dimensional, Diffusion Equation Code," AMTD-118 (1962)

APPENDIX

This appendix contains:

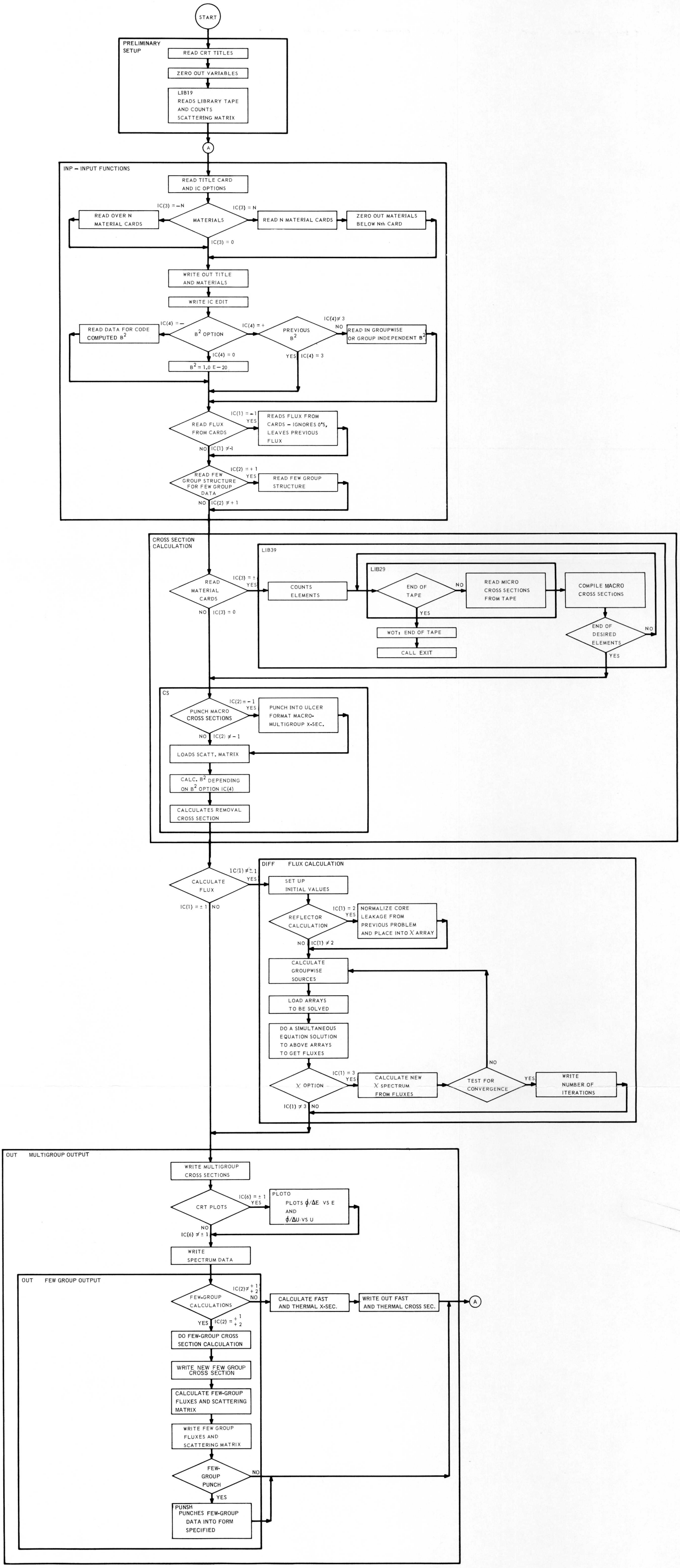
a. QUICKIE Block Diagram

b. Listing of QUICKIE

c. Sample Problem Input

d. Sample Problem Output

QUICKIE III BLOCK DIAGRAM



1-21-64

NAA-SR-9233

19

7569-01029

OSTI ID:0000 1949

```
*     LABEL(QKEIII)
C     QUICKIE III BY W.A.RHOADES AND M.A.BOLING                 6-12-63
      DIMENSION
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILO1(5),SCHI(40)
     7,AEXX(9),OEXX(9),ALXX(9),OLXX(9),ENGG(40),U(40),XXX(80),YYY(80)
      COMMON
     1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
     4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
     5,N5,N6,N9,I6M,SCHI,AEXX,OEXX,ALXX,OLXX,ENGG,U,XXX,YYY
      EQUIVALENCE
     1 (IBLANK,ABLANK)

      N5 = 5
      N6 = 6
      N9 = 9
      READ INPUT TAPE N5,20,AEXX,OEXX,ALXX,OLXX
   20 FORMAT(9A6)
      DO 10 I=1,40
   10 B2(I) = 0.0
      ABLANK = 1H
      DO 12 I=1,20
      DI(I) = 0.0
   12 NAME(I) = IBLANK

C     CALCULATE MACROSCOPIC CROSS SECTIONS

   11 CALL LIB19
      CALL  INP
      IF(IC(3))18,14,18
   18 CALL  LIB39
   14 CALL  CS

C     CALCULATE FLUXES

      IF(XABSF(IC(1)) -1 ) 16,15,16
   16 CALL  DIFF

C     CALCULATE OUTPUT DATA
```

```
   15 CALL OUT
      GO TO 11
      END
*     LABEL(LIB19 )
      SUBROUTINE LIB19
      DIMENSION
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILO1(5),SCHI(40)
      COMMON
     1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
     4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
     5,N5,N6,N9,I6M,SCHI

C     READS LIBRARY TAPE AND COUNTS SCATTERING MATRIX

      REWINDN9
      READ TAPEN9,NOG,NDSD,NDSU,NTH
      READ TAPEN9,(ENG(J),J=1,NOG)
      READ TAPE N9,(SCHI(J),J=1,NOG)
      JZD=0
      DO 20 J=2,NOG
      JM1=J-1
      DO 20 I=1,JM1
      IF(J-NDSD-I)30,30,20
   30 JZD = JZD + 1
   20 CONTINUE
      JZU=0
      JM1=NOG-1
      DO 40 J=NTH,JM1
      JP1 = J + 1
      DO 40 I=JP1,NOG
      IF(I - NDSU - J)140,140,40
  140 JZU = JZU + 1
   40 CONTINUE
      RETURN
      END
*     LABEL(INP    )
      SUBROUTINE INP
```

```
      DIMENSION
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILO1(5),SCHI(40)
      COMMON
     1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
     4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
     5,N5,N6,N9,I6M,SCHI
      DIMENSION NA(20),D(40)

C     READS INPUT AND STORES BUCKLING IN PROPER LOCATIONS

  150 READ INPUT TAPEN5 , 7 , TITLE , IC
    7 FORMAT (10A6,6I2)
      ICI =XABSF(IC(3)*4)
      IF(IC(3))70,50,60
   70 READ INPUT TAPE N5,6,(NA(I),D(I),I=1,ICI)
      DO 80 I=1,ICI
      IF(NA(I) - IBLANK)90,80,90
   90 NAME (I) = NA(I)
      DI(I) = D(I)
   80 CONTINUE
      GO TO 50
   60 DO 95 I=1,20
      NAME(I) = IBLANK
   95 DI(I) = 0.0
      READ INPUT TAPE N5,6,(NAME(I),DI(I),I=1,ICI)
    6 FORMAT (4(A6,F12.0))
   50 WRITE OUTPUT TAPEN6 , 1 ,TITLE ,(I,NAME(I),DI(I),I=1,20)
    1 FORMAT (1H1 10A6 /// (33X,I3,7X,A6,6X,1PE14.7 /   ))
      WRITE OUTPUT TAPEN6,551,(IC(I),I=1,6)
  551 FORMAT(//4X,10HIC EDIT---6I6)

 1150 IC4 = IC (4)
      IF(IC4)101,120,102
  101 N12 = - IC4
      GO TO ( 103, 104, 105 ), N12
  103 READ INPUT TAPE N5,8,BOP,X7,Y7,Z7
      GO TO 20
  104 READ INPUT TAPE N5,8,BOP,R8,Z8
```

```
      GØ TØ 20
  105 READ INPUT TAPE N5,8,BØP,R9
      GØ TØ 20

  120 DØ 122 I=1,NØG
      ØMEGA(I) = 0.0
  122 B2(I) = 1.0E-20
      GØ TØ 20
  102 GØ TØ (30,10,20),IC4
   30 READ INPUT TAPEN5 , 8 ,B
    8 FØRMAT (4(6X,F12.0))
      DØ 40 I = 1, NØG
   40 B2 (I) = B
      GØ TØ 20
   10 READ INPUT TAPEN5 ,8 ,(B2 (I), I = 1, NØG )

   20 IF(IC(1)+1)320,310,320
  310 READ INPUT TAPE N5, 8 ,(D(I),I=1,NØG)
      DØ 312 I=1,NØG
      IF(D(I))314,312,314
  314 PHI(I) = D(I)
  312 CØNTINUE

  320 IF(IC(2)-1)302,303,302
  303 READ INPUT TAPE N5,301,(IL(I),I=1,6),I6M,(ILØ1(I),I=1,5)
  301 FØRMAT(12I6)
  302 RETURN
      END
*     LABEL(LIB39 )
      SUBRØUTINE LIB39
      DIMENSIØN
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STØT(40),DNS(780),UPS(780),ØMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILØ1(5),SCHI(40)
      CØMMØN
     1 IBLANK,NØG,NDSD,NDSU,NTH,LEM,AW,VUZERØ,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STØT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BØP,X7,Y7,Z7,R8,Z8,R9,
     4 ØMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILØ1
     5,N5,N6,N9,I6M,SCHI
      DIMENSIØN NAMEX(20)

```

```
C     COUNTS ELEMENTS AND COMPILES MACROSCOPIC CROSS SECTIONS

      DO 30 I=1,40
      SF ( I ) = 0.0
      SA ( I ) = 0.0
      ST ( I ) = 0.0
      DO 30 J=1,40
      SIG(J,I) = 0.0
      AA( J,I) = 0.0
   30 CHI(J,I) = 0.0
      DO 34 I=1,780
   32 DNS ( I ) = 0.0
   34 UPS ( I ) = 0.0
      NSUM = 0
      DO 1 I = 1 , 20
      NAMEX(I) = NAME(I)
      IF(NAME(I) - IBLANK)3,1,3
    3 NSUM = NSUM + 1
    1 CONTINUE

   16 CALL LIB29
      IF( LEM - IBLANK )18,120,18
   18 DO 5 I = 1 , 20
      IF ( NAME (I) - LEM ) 5, 6, 5
    6 IF ( NAMEX(I) -IBLANK)7, 5, 7
    7 DII = DI ( I )
      NAMEX(I) = IBLANK
      NSUM = NSUM - 1

      DO 10 J=1,NOG
      SF(J) = SF(J) + SSF(J)*DII*VUZERO
      SA(J) = SA(J) + SSA(J)*DII
   10 ST(J) = ST(J) + SST(J)*DII
      DO 15 J=1,JZD
   15 DNS(J) = DNS(J) + DNSTR(J)*DII
      DO 20 J=1,JZU
   20 UPS(J) = UPS(J) + UPSTR(J)*DII
      IF(CHIIND)22,24,22
   22 DO 28 J=1, NOG
      DO 28 K=1, NOG
   28 CHI(K,J)= CHI(K,J)+VUZERO*SSF(K) * FCHI(J)* DII
      GO TO  29
   24 IF(VUZERO) 25,29,25
   25 DO 26 J=1,NOG
      DO 26 K=1,NOG
```

```
 26 CHI(K,J) = CHI(K,J) + VUZERO *SSF(K) * SCHI(J)*DII
 29 IF(NSUM) 14,14,5
  5 CONTINUE
    GO TO 16
 14 REWINDN9
114 RETURN
120 WRITE OUTPUT TAPE N6,121,NAMEX
121 FORMAT(//6X,55HTHE FOLLOWING ELEMENTS WERE NOT FOUND IN THE LIBRAR
   1Y  -  // (6(6X,A6))  )
    CALL EXIT
    STOP
    END
*   LABEL(LIB29 )
    SUBROUTINE LIB29
    DIMENSION
  1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
  2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
  3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
  4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
  5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
  6 C8(15),C9(15),ILO1(5),SCHI(40)
    COMMON
  1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
  2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
  3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
  4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
  5,N5,N6,N9,I6M,SCHI

C   READS MICRO CROSS SECTIONS FROM LIBRARY TAPE

    EQUIVALENCE (THEEND,IHEEND)
    THEEND = 6HTHEEND
    READ TAPEN9,LEM
    IF(LEM - IHEEND) 2,30,2
  2 READ TAPEN9,AW,VUZERO,B,CHIIND
    IF(VUZERO)5,10,5
  5 READ TAPEN9,(SSF(J),J=1,NOG)
 10 READ TAPEN9,(SSA(J),J=1,NOG)
    READ TAPEN9,(SST(J),J=1,NOG)
    READ TAPEN9,(DNSTR(J),J=1,JZD)
    READ TAPEN9,(UPSTR(J),J=1,JZU)
    IF(CHIIND)15,20,15
 15 READ TAPEN9,(FCHI(J),J=1,NOG)
 20 READ TAPEN9,AFN
    GO TO 25
```

```
   30 LEM = IBLANK
   25 RETURN
      END
      LABEL(CS     )
      SUBRØUTINE CS
      DIMENSIØN
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STØT(40),DNS(780),UPS(780),ØMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILØ1(5),SCHI(40)
      CØMMØN
     1 IBLANK,NØG,NDSD,NDSU,NTH,LEM,AW,VUZERØ,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STØT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BØP,X7,Y7,Z7,R8,Z8,R9,
     4 ØMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILØ1
     5,N5,N6,N9,I6M,SCHI

C     LØADS SCATTERING MATRICIES SIG AND AA WITH MACRØ CRØSS SECTIØNS
C     INCLUDES ØPTIØNAL PUNCH ØUTPUT ØFMACRØ CRØSS SECTIØNS FØR ULCER
C     AND ØPTIØN FØR CØDE-CALCULATED BUCKLING

      IF ( IC (2) + 1 ) 111 ,112,111
  112 PUNCH 7,( SA(I) , I = 1 , NØG )
      PUNCH 7,( ST(I) , I=  1 , NØG )
    7 FØRMAT( 6E12.5)
      PUNCH 7,( DNS(I) , I = 1, JZD   )
      PUNCH 7,( UPS(I) , I = 1, JZU   )
      DØ 9  I =1,5
    9 PUNCH 8
    8 FØRMAT (72H++++++++++++++++++++++++++++++++++++++++++++++++++++++
     1++++++++++++++++++)
  111 IF(IC(3)) 150,50,150
  150 DØ 10 I=1,40
      DØ 10 J=1,40
   10 SIG(I,J)=0.0

      KK  = 0
      DØ 20 J = 2 , NØG
      JM1 = J - 1
      DØ 20 I = 1 , JM1
      IF ( J - NDSD - I ) 30 , 30 , 20
   30 KK= KK+ 1
      SIG(I,J) = DNS(KK)
```

```
   20 CONTINUE
      KK = 0
      JM1 = NOG - 1
      DO 40 J = NTH , JM1
      JP1 = J + 1
      DO 40 I = JP1 , NOG
      IF ( I - NDSU - J ) 140,140, 40
  140 KK = KK + 1
      SIG(I,J) = UPS(KK)
   40 CONTINUE

      IF(IC(5) - 1)50,62,50
   62 DO 45  I = 1, NOG
   45 ST(I) = ST(I) * (1.0 - 0.4*SA(I)/ST(I))**2

   50 IF(IC(4))800,801,802
  800 PIE = 3.1415927
      NOP1 = BOP
      DO 501 I = 1, NOG
  501 OMEGA(I) = .710466 / ST(I)
      GO TO  ( 121, 129, 124 ),N12
  121 DO 122 I = 1,NOG
  122 B2(I) = (PIE / (X7 + 2.0 * OMEGA(I)))**2+(PIE / (Y7 + 2.0 * OMEGA(
     XI)))**2+(PIE / (Z7 + 2.0 * OMEGA(I)))**2
      GO TO 801
  124 DO 126 I = 1,NOG
  127 IF (NOP1) 126, 126, 125
  125 OMEGA(I) = OMEGA(I) / (1.0 + 0.710466/ST(I)/R9)
  126 B2(I) = (PIE / (R9 + OMEGA(I)))**2
      GO TO 801
  129 DO 132 I = 1,NOG
      B2(I) = (PIE / (Z8 + 2.0 * OMEGA(I)))**2
  123 IF (NOP1) 132, 132 , 130
  130 OMEGA(I) = OMEGA(I) / (1.0 + 0.355233/ST(I)/R8)
  132 B2(I) = B2(I) + (2.405 / (R8 + OMEGA(I)))**2
      GO TO 801
  802 DO 803 I = 1,NOG
  803 OMEGA(I) = 0.0
  801 CONTINUE

C     CALCULATE  REMOVAL  C. S.

      DO 60 I = 1 , NOG
      STOT(I) = SA(I) + B2(I)/(3.0*ST(I))
      DO 60 J=1,NOG
```

```
   60 STOT(I) = STOT(I) + SIG(I,J)



   72 RETURN
      END
*     LABEL(DIFF   )
      SUBROUTINE DIFF
      DIMENSION
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
     4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
     6 C8(15),C9(15),ILO1(5),SCHI(40)
     7,AEXX(9),OEXX(9),ALXX(9),OLXX(9),ENGG(40),U(40),XXX(80),YYY(80)
      COMMON
     1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
     4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
     5,N5,N6,N9,I6M,SCHI,AEXX,OEXX,ALXX,OLXX,ENGG,U,XXX,YYY

C     CALCULATES SOURCES AND FLUXES

      ITER = 0
      ITMAX = 100
      OLDK = 1.0E+32
    7 IF(IC(1) -2) 1,2,1
    2 SSUM = 0.0
      DO 3 I =1,NOG
    3 SSUM = SSUM + SS(I)
      DO 4 I = 1,NOG
    4 SCHI(I) = SS(I) /SSUM
    1 DO 5 I=1,NOG
    5 SS(I) = SCHI(I)
    6 NM1 = NTH - 1
      DO 10 I = 1 , NM1
      PHI(I) = SS(I)/STOT(I)
      NP1 = I + 1
      DO 10 J = NP1 , NOG
   10 SS(J) = SS(J) + PHI(I)*SIG(I,J)

C     LOAD AA AND BB
   98 DO 20 I=NTH,NOG
```

```
   KI = I-NTH+1
   DO 80 J= NTH,NOG
   KJ= J -NTH + 1
80 AA(KJ,KI) =SIG(I,J)
   BB(KI,1) = -SS(I)
20 AA(KI,KI) =-STOT(I)

   F = 1.0
   N = NOG - NTH + 1
   DO 30 I = 1,N
30 IA(I) = 0
   MX = XSIMEQF(40,N,1,AA,BB,F,IA)
31 DO 40 I = NTH,NOG
   KI = I - NTH + 1
40 PHI(I) = AA(KI,1)

   IF(IC(1) - 3) 42,32,42
32 ITER = ITER + 1
   XKEFF = 0.0
   DO 33 J=1,NOG
   SCHI(J) =0.0
   DO 34 K=1,NOG
34 SCHI(J) = SCHI(J) + CHI(K,J) * PHI(K)
33 XKEFF = XKEFF + SCHI(J)
   DO 35 I=1,NOG
   SCHI(I) = SCHI(I) /XKEFF
35 SS(I) = SCHI(I)

   EPS = OLDK / XKEFF -1.0
   IF( ABSF(EPS) - 0.00001) 41,41,36
36 IF(ITER - ITMAX) 37,43,43
37 OLDK = XKEFF
   GO TO 6

43 WRITE OUTPUT TAPE N6,44,EPS
44 FORMAT(//4X,44HWARNING, CONVERGENCE WAS NOT COMPLETE.  ESP= E18.6)
41 WRITE OUTPUT TAPE N6,45,ITER
45 FORMAT(//I12,3X,25HITERATIONS WERE PERFORMED )

42 GO TO (71,72,73),MX
71 WRITE OUTPUT TAPE N6,81,MX
81 FORMAT(//4X, 3HMX= I3 ,3X,26HSOLUTION WAS SUCCESSFUL    )
74 RETURN
72 WRITE OUTPUT TAPE N6,82,MX
82 FORMAT(//4X, 3HMX= I3,3X,27HSOLUTION INDICATES OVERFLOW )
```

```
      GØ TØ 77                                                          00004560
   73 WRITE ØUTPUT TAPE N6,83,MX                                        00004570
   83 FØRMAT(//4X, 3HMX= I3,3X,37HSØLUTIØN INDICATES A SINGULAR MATRIX )00004580
   77 CALL EXIT                                                         00004590
      STØP                                                              00004600
      END                                                               00004610
*     LABEL(ØUT    )                                                    00004615
      SUBRØUTINE ØUT                                                    00004620
      DIMENSIØN                                                         00004630
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),00004640
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),    00004650
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),     00004660
     4 STØT(40),DNS(780),UPS(780),ØMEGA(40),SIG1(40, 6),PH(6),IL(6),    00004670
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),00004680
     6 C8(15),C9(15),ILØ1(5),SCHI(40)                                   00004690
     7,AEXX(9),ØEXX(9),ALXX(9),ØLXX(9),ENGG(40),U(40),XXX(80),YYY(80)   00004695
      CØMMØN                                                            00004700
     1 IBLANK,NØG,NDSD,NDSU,NTH,LEM,AW,VUZERØ,B,CHIIND,AFN,JZD,JZU,MX,  00004710
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STØT,DNS,UPS,ENG,CHI,SSF,SSA,SST00004720
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BØP,X7,Y7,Z7,R8,Z8,R9,  00004730
     4 ØMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILØ100004740
     5,N5,N6,N9,I6M,SCHI,AEXX,ØEXX,ALXX,ØLXX,ENGG,U,XXX,YYY             00004750
                                                                        00004760
C     PRØBLEM RESULT EDIT                                               00004770
                                                                        00004780
      DØ 20 I = 1 , NØG                                                 00004790
      SIG1(I,1) = PHI(I)/ENG(I)                                         00004800
      SIG1(I,2) = 1.0/(3.0*ST(I))                                       00004810
   20 SIG1(I,3) = B2(I)*SIG1(I,2)                                       00004820
      WRITE ØUTPUT TAPEN6,3                                             00004830
    3 FØRMAT(1H0,1HI,7X,3HPHI,7X,13HPHI / DELTA E,8X,1HD,10X,10HNU SIGMA00004840
     X F,6X,7HSIGMA A,10X,3HDB2,8X,11HSIGMA TØTAL//)                    00004850
      WRITE ØUTPUT TAPEN6 , 2,(I,PHI(I),SIG1(I,1),SIG1(I,2),SF(I),SA(I),00004860
     XSIG1(I,3),STØT(I),I=1,NØG)                                        00004870
    2 FØRMAT(I3, 1P7E15.7 / )                                           00004880
                                                                        00004881
      IF(IC(6)) 2007,2003,2007                                          00004882
 2007 CALL PLØTC                                                        00004885
 2003 TF=0.0                                                            00004890
      DØ 24 I = 1 , NØG                                                 00004900
      SS(I) = PHI(I) * SIG1(I,3)                                        00004910
      SIG1(I,4) = SIG1(I,1) * SF (I)                                    00004920
      SIG1(I,5) = SIG1(I,1) * SA (I)                                    00004930
  110 SIG1(I,6) = SIG1(I,1) * SIG1(I,3)                                 00004940
   24 TF = TF + PHI (I) * SF (I)                                        00004950
```

```
   12 WRITE OUTPUT TAPEN6,4
    4 FORMAT (1H1,5X,1HI,8X,5HOMEGA,12X,2HB2,14X,3HCHI,11X,7HFISSION,7X,
     X10HABSORPTION,8X,7HLEAKAGE/1H ,61X,8HSPECTRUM,7X,8HSPECTRUM,9X,8HS
     XPECTRUM  )
      WRITE OUTPUT TAPE N6,5,(I,OMEGA(I),B2(I),SCHI(I),SIG1(I,4),SIG1(I,
     X5),SIG1(I,6),I=1,NOG)
    5 FORMAT (1H0,3X,I4,1P6E16.7 )
      WRITE OUTPUT TAPEN6 , 201, TF
  201 FORMAT (1H1,40X,7HK EFF =1PE15.7// )

C     CALCULATE FEW GROUP CROSS SECTIONS

 1115 J = 1
      IX = 1
      IF(IC(2))118,118,116
  116 DO 333 I=1,6
      IF(IL(I)) 333,334,333
  333 CONTINUE
      I=7
  334 I6 = I - 1
  117 J1 = IL(IX)
      GO TO 115
  118 I6=2
      I6M=0
      J1 = NTH - 1
  115 TF = 0.0
      TA = 0.0
      TS = 0.0
      TDB2 = 0.0
      TB2 = 0.0
      TPHI = 0.0
      DO 10 I = J , J1
      TF = TF + PHI (I) * SF (I)
      TA = TA + PHI (I) * SA (I)
      TS = TS + PHI(I) * ST(I)
      TDB2 = TDB2 + PHI (I) * SIG1(I,3)
      TB2 = TB2 + PHI(I) * B2(I)
   10 TPHI = TPHI + PHI (I)
      C1(IX) = TF
      C2(IX) = TA
      C3(IX) = TDB2
      CSA(IX) = TA/TPHI
      CVS6(IX) = TF/TPHI
      C4(IX) = TDB2/TB2
      C5(IX) = TS/TPHI
```

```
      C6(IX) = TDB2/TPHI
      C7(IX) = C6(IX)/C4(IX)
      CSTR(IX) = 1.0/(3.0*C4(IX))
  126 IF(J1 - NOG)26,28,28
   26 IF(IX - 6)27,28,28
   27 IX = IX + 1
      J = J1 + 1
      IF(IC(2))127,127,117
  127 J1 = NOG
      GO TO 115

   28 IF(IC(2))728,728,130
  130 WRITE OUTPUT TAPEN6,302,    (IL(I),I=1,I6)
  302 FORMAT(1H0,6X,16HFEW GROUP LIMITS,5X,14HMULTIGROUPS     6I4 )
  728 WRITE OUTPUT TAPEN6,1212,(I,I=1,I6)
 1212 FORMAT(////5X,12HGROUP- - - -/(I23,5I15)///)
      WRITE OUTPUT TAPEN6,1213,(C1(IX),IX=1,I6)
 1213 FORMAT(1H0,5X,7HFISSION 1P6E15.5)
      WRITE OUTPUT TAPEN6,1214,(C2(IX),IX=1,I6)
 1214 FORMAT(1H0,2X,10HABSORPTION 1P6E15.5)
      WRITE OUTPUT TAPEN6,1215,(C3(IX),IX=1,I6)
 1215 FORMAT(1H0,5X,7HLEAKAGE 1P6E15.5)
      WRITE OUTPUT TAPEN6,1216,(C4(IX),IX=1,I6)
 1216 FORMAT (1H0,11X,1HD 1P6E15.5)
      WRITE OUTPUT TAPEN6,1217,(CVS6(IX),IX=1,I6)
 1217 FORMAT(1H0,2X,10HNV SIGMA F 1P6E15.5)
      WRITE OUTPUT TAPEN6,1218,(CSA(IX),IX=1,I6)
 1218 FORMAT(1H0,5X,7HSIGMA A 1P6E15.5)
      WRITE OUTPUT TAPEN6,1219,(C5(IX),IX=1,I6)
 1219 FORMAT(1H0,4X, 8HSIGMA TR 1P6E15.5)
      WRITE OUTPUT TAPEN6,1220,(C6(IX),IX=1,I6)
 1220 FORMAT(1H0,8X,4HD B2 1P6E15.5)
      WRITE OUTPUT TAPEN6,1221,(C7(IX),IX=1,I6)
 1221 FORMAT(1H0,9X,3HB 2  1P6E15.5)
      WRITE OUTPUT TAPEN6,1222,(CSTR(IX),IX=1,I6)
 1222 FORMAT(1H0,5X,7H1/(3*D)  1P6E15.5)

C     CALCULATE FEW GROUP FLUXES AND SCATTERING MATRIX

      IF(IC(2))128,128,740
  740 IT = 1
      DO310 I = 1,I6
      IB = IL(I)
      PH(I) = 0.0
      DO 15  II = IT,IB
```

```
 15 PH(I) = PH(I) + PHI(II)
    JT = 1
    DO320 J = 1,I6
    JB = IL(J)
    SIG1(I,J) = 0.0
    DO 30 II = IT,IB
    DO 30 JJ = JT,JB
 30 SIG1(I,J) = SIG1(I,J) + PHI(II) * SIG(II,JJ)
    JT = JB + 1
320 SIG1(I,J)  = SIG1(I,J) /PH(I)
    SIG1(I,I)= 0.0
310 IT = IB + 1

    WRITE OUTPUT TAPEN6,350,(JIJ,JIJ=1,I6)
350 FORMAT(1H1////24X,14HFEW GROUP FLUX//I10,5(I14))
    WRITE OUTPUT TAPEN6,420,(PH(I),I=1,I6)
420 FORMAT(1PE16.7,1P5E15.7)
    WRITE OUTPUT TAPEN6,351,(JIJ,JIJ=1,I6)
351 FORMAT(///18X,27HFEW GROUP SCATTERING MATRIX//27X,10HFROM - - -//
   X I23,5I15)
    WRITE OUTPUT TAPEN6,411
411 FORMAT(//8H TO  - -)
    DO 353 J=1,I6
    WRITE OUTPUT TAPEN6,352, J,(SIG1(I,J),I=1,I6)
352 FORMAT(I11,2X,1P6E15.7)
353 CONTINUE
    IF(I6M ) 335,128,335
335 CALL PUNSH
128 RETURN
    END
*   LABEL(PUNSH )
    SUBROUTINE PUNSH
    DIMENSION
   1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),
   2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),
   3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),
   4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),
   5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),
   6 C8(15),C9(15),ILO1(5),SCHI(40)
    COMMON
   1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,
   2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST
   3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,
   4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO1
   5,N5,N6,N9,I6M,SCHI
```

```
      DIMENSION C11(6),SIL0(8),SILP(24)

      I6M1 = I6 - 1
      GO TO( 1,2,3           ),I6M

C     FAIM LIBRARY PUNCH

    1 PUNCH 1420,(CSA(I),I=1,I6),(CVS6(I),I=1,I6),(CSTR(I),I=1,I6),((SIG
     11(I,J),J=1,I6),I=1,I6M1)
 1420 FORMAT (6E12.5 )
      GO TO 100

C     ULCER LIBRARY PUNCH

    2 NTH = IL01(1)
      IF(CVS6(1)) 441,440,441
  441 PUNCH 1420,(CVS6(I),I=1,I6 )
  440 PUNCH 1420,(CSA(I),I=1,I6)
      PUNCH 1420,(CSTR(I),I=1,I6)
      NDSD = I6 - 1
      NDSU = I6-NTH
      KK=0
      DO 470 J=2,I6
      JM1 = J-1
      DO 470 I=1,JM1
      IF( J - NDSD - I) 473,473,470
  473 KK = KK + 1
      C8(KK) = SIG1(I,J)
  470 CONTINUE
      PUNCH 1420,(C8(I),I=1,KK )
      KK = 0
      JM1 = I6-1
      DO 474 J= NTH,JM1
      JP1 =  J+1
      DO 474 I= JP1,I6
      IF( I - NDSU - J ) 480, 480, 474
  480 KK = KK + 1
      C9(KK) = SIG1(I,J)
  474 CONTINUE
      PUNCH 1420,(C9(I),I=1,KK )
      GO TO 100

C     FLOCO PUNCH

    3 JJ = I6 - 1
```

```
      DO 10 I=1,JJ
   14 II = I + 1
      SUMS = 0.0
      DO 11 J = II,I6
   11 SUMS = SIG1(I,J) + SUMS
   10 C11(I) = C5(I) - (SUMS + CSA(I))
      C11(I6) = C5(I6) - CSA(I6)

      JK = I6 + 2
      JK3 = 3*JK
      WRITE OUTPUT TAPE N6, 5
    5 FORMAT( 52HOTHE FOLLOWING FLOCO CARDS HAVE BEEN PUNCHED - - - -//)
      DO 20 I = 1,I6
      DO 22 J=1,JJ
      II = I-J
      IF(II) 70,70,71
   71 SILO(J+3) = SIG1(II,I)
      GO TO 22
   70 SILO(J+3) = 0.0
   22 CONTINUE
    7 SILO(1) = CVS6(I)
      SILO(2) = C5(I)
      SILO(3) = C11(I)
      CALL FLOPUN(SILO(1),SILP(1),JK)
      WRITE OUTPUT TAPE N6,24,(SILP(K),K=1,JK3)
   24 FORMAT(6X,12A6)
   20 CONTINUE
  100 RETURN
      END
      SUBROUTINE FLOPUN (E,F,N)
      DIMENSION E(2),F(2)
      CALL FLOPN (E,F,N)
      NN=3*N
      DO 10  I=1,NN,12
      II=I+11
      IF(NN-II)1,2,2
    1 II=NN
    2 CONTINUE
      PUNCH 5,(F(J),J=I,II)
    5 FORMAT(12A6)
   10 CONTINUE
      RETURN
      END
*      FAP
       COUNT    100
```

```
*       THIS IS FLØPN SUBRØUTINE FØR W. RHØADES
*************************************************************************
*       THIS RØUTINE IS TØ MAKE AVAILABLE BCD FLØCØ CARD IMAGES
*
*       AND CHANGE FLØATING PØINT NUMBERS TØ BCD
*
*       EXECUTIØN TØ BE CØNTRØLLED BY A FØRTRAN STATEMENT
*       SIMILAR TØ
*       DIMENSIØNA(100),B(300)
*       CALL FLØPN (A(5),B(13),N)
*************************************************************************
        EJECT
        ENTRY   FLØPN
 KHALF  ØCT     000000606060
 RHALF  ØCT     777777000000
 TEN    ØCT     000010000000
 MINSE  ØCT     000040000025
 PLUSE  ØCT     000020000025
 PLUS   ØCT     002000000000
 MINUS  ØCT     004000000000
 K9DPT  ØCT     110033000000
 KBDPT  ØCT     600033000000
 K3ØCT  ØCT     000000000003
 CNTR   PZE
 K1ØCT  ØCT     000000000001
 K9DDD  ØCT     110033000000
 ZERØ   ØCT     000000000000
 TAG1   PZE     0,1,0
 TAG2   PZE     0,2,0
 CHAR   ØCT     233000000000
 LHALF  ØCT     606000000000
 ØNE    ØCT     000001000000
 B1     PZE
 B2     PZE
 B3     PZE
 B4     PZE
 B5     PZE
    D1  PZE
     D  PZE
    AK  PZE
    BK  PZE
        PZE
        PZE
        PZE
 FLØPN  SXD      *-2,2
```

```
      SXD    *-2,1
      CLA    1,4
      STO    AK
      CLA    2,4
      STO    BK
      CLA*   3,4
      SUB    ONE
      STD    TEST
      CLA    TAG2
      STT    AK
      CLA    TAG1
      STT    BK
      SXD    BK+1,4
      AXT    0,2
      AXT    0,1
      STZ    CNTR
      CLA    K9DDD
      STO    K9DPT
BEGIN CLA*   AK
      STO    D
      TSX    $BINBCD,4
      PZE    D
C     PZE    B1
K     PZE    B2
      CLA    ZERO
      XCA
      LDQ*   C
      LGL    18
      SSP
      LDQ    D
      TQP    *+3
      ADD    MINUS
      TRA    *+2
      ADD    PLUS
      ORA    K9DPT
      SLW*   BK
      CLA    CNTR
      SUB    K3OCT
      TZE    MID1
      TRA    MID2
MID1  STZ    CNTR
      CLA    K9DDD
      STO    K9DPT
      TRA    MID3
MID2  CLA    CNTR
```

```
        ADD     K1OCT                                       00008150
        STO     CNTR                                        00008160
        CLA     K8DPT                                       00008170
        STO     K9DPT                                       00008180
MID3    TXI     *+1,1,1                                     00008190
        LDQ*    C                                           00008200
        LGL     18                                          00008210
        CLA     ZERO                                        00008220
        LLS     35                                          00008230
        STO     B5                                          00008240
        CAL     B5                                          00008250
        ANA     RHALF                                       00008260
        ORA     KHALF                                       00008270
        SLW*    BK                                          00008280
        TXI     *+1,1,1                                     00008290
        CLA*    K                                           00008300
        ARS     18                                          00008310
        ORA     CHAR                                        00008320
        FAD     CHAR                                        00008330
        STO     D1                                          00008340
        TSX     BINBCD,4                                    00008350
        PZE     D1                                          00008360
C1      PZE     B3                                          00008370
K1      PZE     B4                                          00008380
        CLA     B2                                          00008390
        SSP                                                 00008400
        SUB     TEN                                         00008410
        TPL     ON1                                         00008420
        CLA     ZERO                                        00008430
        LDQ*    C1                                          00008440
        LGL     6                                           00008450
        TRA     ON2                                         00008460
ON1     CLA     ZERO                                        00008470
        LDQ*    C1                                          00008480
        LGL     12                                          00008490
ON2     ALS     6                                           00008500
        LDQ*    K                                           00008510
        TQP     *+3                                         00008520
        ADD     MINSE                                       00008530
        TRA     *+2                                         00008540
        ADD     PLUSE                                       00008550
        ORA     LHALF                                       00008560
        SLW*    BK                                          00008570
        TXI     *+1,1,1                                     00008580
        TXI     *+1,2,1                                     00008590
```

```
TEST   TXL      BEGIN,2,0                                                 00008600
       LXD      FLOPN-3,4                                                 00008610
       LXD      FLOPN-2,2                                                 00008620
       LXD      FLOPN-1,1                                                 00008630
       TRA      4,4                                                       00008640
       END                                                                00008650
*     LABEL(PLOTO )                                                       00008999
      SUBROUTINE PLOTO                                                    00009000
      DIMENSION                                                           00009010
     1 ENG(40),CHI(40,40),SSF(40),SSA(40),SST(40),DNSTR(780),UPSTR(780),00009020
     2 FCHI(40),IA(40),AA(40,40),BB(40,1),PHI(40),SIG(40,40),SS(40),      00009030
     3 IC(6),B2(40),NAME(20),DI(20),TITLE(10),SF(40),SA(40),ST(40),       00009040
     4 STOT(40),DNS(780),UPS(780),OMEGA(40),SIG1(40, 6),PH(6),IL(6),      00009050
     5 CSA(6),CVS6(6),CSTR(6),C1(6),C2(6),C3(6),C4(6),C5(6),C6(6),C7(6),00009060
     6 C8(15),C9(15),ILC1(5),SCHI(40)                                     00009070
     7,AEXX(9),OEXX(9),ALXX(9),OLXX(9),ENGG(40),U(40),XXX(80),YYY(80)     00009080
      COMMON                                                              00009090
     1 IBLANK,NOG,NDSD,NDSU,NTH,LEM,AW,VUZERO,B,CHIIND,AFN,JZD,JZU,MX,    00009100
     2 DII,IC,B2,NAME,DI,TITLE,SF,SA,ST,STOT,DNS,UPS,ENG,CHI,SSF,SSA,SST00009110
     3,DNSTR,UPSTR,FCHI,IA,AA,BB,PHI,SIG,SS,N12,BOP,X7,Y7,Z7,R8,Z8,R9,    00009120
     4 OMEGA,SIG1,PH,IL,CSA,CVS6,CSTR,C1,C2,C3,C4,C5,C6,C7,C8,C9,I6,ILO100009130
     5,N5,N6,N9,I6M,SCHI,AEXX,OEXX,ALXX,OLXX,ENGG,U,XXX,YYY               00009140
                                                                          00009145
C     PLOTS SPECTRUM GRAPHS                                               00009150
                                                                          00009160
      ETOP = 10.0E+06                                                     00009170
      ENGG(1)=ETOP -ENG(1)                                                00009180
      U(1)= LOGEF(ETOP/ENGG(1))                                           00009190
      SIG1(1,4) = PHI(1)/U(1)                                             00009195
      DO 2001 I=2,NOG                                                     00009200
 2000 ENGG(I) = ENGG(I-1)-ENG(I)                                          00009210
      U(I) = LOGEF(ETOP/ENGG(I))                                          00009240
 2001 SIG1(I,4)= PHI(I)/(U(I)-U(I-1))                                     00009250
      IF(IC(6)) 2004,2003,2005                                            00009260
 2004 J=0                                                                 00009270
      NUX = 2*NOG                                                         00009272
      NUXM = NUX-2                                                        00009276
      XXX(1) =0.0                                                         00009280
      XXX(NUX) = U(NOG)                                                   00009290
      DO 2444 I=2,NUXM,2                                                  00009300
      J=J+1                                                               00009310
      XXX(I) = U(J)                                                       00009320
 2444 XXX(I+1) =XXX(I)                                                    00009330
      J=0                                                                 00009340
      DO 2445 I=1,NUX ,2                                                  00009350
```

```
      J=J+1
      YYY(I) = SIG1(J,4)
 2445 YYY(I+1) = YYY(I)
      CALL AICRT 3(0,1,XXX(1),YYY(1),NUX,1,2,1,42,TITLE,ALXX,ØLXX,3,1,16
     1.0,16.0,1,XL,XU,1,YL,YU)

 2005 J=0
      NNP1 =(NØG-NTH +1) * 2
      NNPM = NNP1 - 2
      XXX(NNP1) = ENGG(NTH-1)
      XXX(1) =    ENGG(NØG)
      DØ 444 I=2,NNPM,2
      J=J+1
      NJNJ = NØG - J
      XXX(I) = ENGG(NJNJ)
  444 XXX(I+1) = XXX(I)
      J = -1
      DØ 11 I=1,NNP1,2
      J=J+1
      JJ = NØG - J
      YYY(I) = SIG1(JJ,1)
   11 YYY(I+1) = YYY(I)
      CALL AICRT 3(1,1,XXX(1),YYY(1),NNP1,1,2,1,42,TITLE,AEXX,ØEXX,3,1,
     116.0,16.0,1,XL,XU,1,YL,YU)
 2003 RETURN
      END
```

FORTRAN WORK SHEET

DECK NO. Data PROGRAMMER M. A. Boling DATE 10-31-63 PAGE 1 OF 1 JOB NO. 7000-00

FOR PROGRAM NO. 9W-005 TITLE QUICKIE Sample Problems – Input BY DEPT. 721 GROUP 131

1	10	20	30	40	50		60 / 70	IDENTIFICATION 73–80
SAMP	LE PROBLEM -	READ N	EW ELEMENTS	- COD	E CALC. B2		+0+1+1-1+0-1	10
HZR	0.05	ZR	0.03	U235	0.0010			20
	0.0		20.0		30.0		20.0	30
6	10 14	23						40
SECO	ND PROBLEM -	ADD NA	TO MATERIAL	S			+0+2-1+3+0+1	50
						NA	0.0005	60
THIR	D PROBLEM -	USE B2	=0.0 -NOTE F	EW GRO	UP D IS 0.0		+0+2+0+1+0+0	70
	0.0							80
FOUR	TH PROBLEM -	USE B2	=1.0E-20				+0+2+0+0+0+0	90
FIFT	H PROBLEM -	WEIGHT	B OVER OLD	FLUX			+1+2+1+3+0+0	100
B	1.0							110
SIXT	H PROBLEM -	BASE C	ASE FOR REFL	ECTOR	CALC.-CHI OP	TION	+3+1+2+2+1+0	120
U235	0.001166	U238	0.0008462	HZR	0.05454	ZR	0.02917	130
NA	0.000528	K	0.001142	PU241	0.0002			140
	0.0072		0.047		0.032		0.020	150
	0.0105		0.0049		0.004		0.0073	160
	0.0082		0.0070		0.0061		0.0067	170
	0.008		0.0067		0.0014		-0.0279	180
	-0.0279		-0.0279		-0.0279		-0.0279	190
	-0.0279		-0.0279		-0.0279			200
1	14 23							210
SEVE	NTH PROBLEM-	REFLEC	TOR CALC. -	ADD BE	AND DELETE	OTHERS	+2+2+1+0+1+0	220
BE	0.123							230

FORM 114-D-18

SAMPLE PRØBLEM -READ NEW ELEMENTS - CØDE CALC. B2

1	HZR	4.9999999E-02
2	ZR	3.0000000E-02
3	U235	9.9999999E-04
4		-0.
5		0.
6		0.
7		0.
8		0.
9		0.
10		0.
11		0.
12		0.
13		0.
14		0.
15		0.
16		0.
17		0.
18		0.
19		0.
20		0.

IC EDIT--- 0 1 1 -1 0 -1

MX= 1 SOLUTION WAS SUCCESSFUL

I	PHI	PHI / DELTA E	D	NU SIGMA F	SIGMA A	DB2	SIGMA TOTAL
1	1.1563397E 00	1.6519138E-07	2.9563932E 00	3.5569999E-03	1.3499999E-03	7.0978755E-02	1.7641875E-01
2	2.1280724E 00	1.3300452E-06	2.2988506E 00	3.1959999E-03	1.4500000E-03	6.5423764E-02	1.9165376E-01
3	1.2589950E 00	2.5179899E-06	1.7935612E 00	3.0869999E-03	1.5399999E-03	5.8813483E-02	2.3512347E-01
4	1.7593551E 00	3.5187101E-06	1.1799410E 00	2.9879999E-03	1.7100000E-03	4.6667318E-02	2.3660730E-01
5	1.9130787E 00	6.3769289E-06	8.9919972E-01	3.5179999E-03	2.1099999E-03	3.9001218E-02	2.4924121E-01
6	1.1130454E 00	1.3410185E-05	6.5397947E-01	6.1249999E-03	3.5999999E-03	3.0863339E-02	4.0252832E-01
7	8.3977650E-01	5.9984035E-05	5.9426181E-01	1.0290000E-02	5.9499999E-03	2.8644069E-02	4.8858405E-01
8	7.2839935E-01	2.9730586E-04	6.0500461E-01	1.9359999E-02	1.1600000E-02	2.9050724E-02	5.2492071E-01
9	6.6394476E-01	1.4754328E-03	5.9392298E-01	4.5814999E-02	2.7849999E-02	2.8631189E-02	5.3931117E-01
10	4.1650527E-01	5.9500752E-03	5.6858565E-01	9.3099998E-02	5.8849999E-02	2.7658620E-02	6.7210858E-01
11	3.3437777E-01	1.6718888E-02	5.3898187E-01	1.3475000E-01	9.1149998E-02	2.6498348E-02	7.2690330E-01
12	3.2250324E-01	4.6071891E-02	5.3806834E-01	1.0780000E-01	8.5699997E-02	2.6462128E-02	7.0749207E-01
13	2.8027039E-01	1.4013519E-01	5.7955896E-01	7.3499998E-02	4.1909999E-02	2.8082122E-02	6.9045208E-01
14	2.0864052E-01	3.4773420E-01	5.4024853E-01	1.7150000E-01	8.4579998E-02	2.6548529E-02	7.8135849E-01
15	1.3017261E-01	6.5086306E-01	3.2165048E-01	4.1265279E-01	2.1428909E-01	1.7127110E-02	9.6583511E-01
16	6.9521466E-02	1.1586911E 00	2.7225660E-01	4.6017539E-01	2.3475699E-01	1.4769360E-02	9.1365486E-01
17	1.4703961E-01	3.6759902E 00	2.4340050E-01	5.4618878E-01	2.7286479E-01	1.3349469E-02	3.3223744E-01
18	1.1813572E-01	3.9378574E 00	2.1469346E-01	6.8513489E-01	3.3986389E-01	1.1904836E-02	3.6125033E-01
19	1.0465957E-01	3.4886523E 00	1.8214837E-01	8.9524359E-01	4.4373028E-01	1.0227327E-02	4.6667832E-01
20	5.5212663E-02	2.7606332E 00	1.3996055E-01	1.2701462E 00	6.2883908E-01	7.9879987E-03	6.5408200E-01
21	1.9120094E-02	1.9120094E 00	9.9863039E-02	1.8741057E 00	9.2679697E-01	5.7894622E-03	9.5794371E-01
22	6.9695062E-03	1.1615844E 00	7.0426892E-02	2.8411503E 00	1.4050952E 00	4.1304821E-03	1.4474212E 00
23	1.6234376E-03	5.4114587E-01	4.5281229E-02	4.9825414E 00	2.4616641E 00	2.6822588E-03	2.5303246E 00

I	ØMEGA	B2	CHI	FISSIØN SPECTRUM	ABSØRPTIØN SPECTRUM	LEAKAGE SPECTRUM
1	6.3012506E 00	2.4008563E-02	2.0400000E-01	5.8758572E-10	2.2300836E-10	1.1725078E-08
2	4.8997656E 00	2.8459337E-02	3.4400000E-01	4.2508244E-09	1.9285655E-09	8.7016565E-08
3	3.8227926E 00	3.2791457E-02	1.6800000E-01	7.7730346E-09	3.8777043E-09	1.4809175E-07
4	2.5149239E 00	3.9550551E-02	1.8000000E-01	1.0513906E-08	6.0169942E-09	1.6420877E-07
5	1.9165525E 00	4.3373254E-02	8.9999999E-02	2.2434035E-08	1.3455320E-08	2.4870799E-07
6	1.3938905E 00	4.7193131E-02	1.4000000E-02	8.2137383E-08	4.8276665E-08	4.1388309E-07
7	1.2666084E 00	4.8201092E-02	-0.	6.1723571E-07	3.5690500E-07	1.7181869E-06
8	1.2895056E 00	4.8017359E-02	-0.	5.7558412E-06	3.4487478E-06	8.6369503E-06
9	1.2658863E 00	4.8206905E-02	-0.	6.7596951E-05	4.1090802E-05	4.2243395E-05
10	1.2118823E 00	4.8644597E-02	-0.	5.5395199E-04	3.5016192E-04	1.6457087E-04
11	1.1487849E 00	4.9163710E-02	-0.	2.2528701E-03	1.5239266E-03	4.4302293E-04
12	1.1468378E 00	4.9179865E-02	-0.	4.9665497E-03	3.9483609E-03	1.2191603E-03
13	1.2352708E 00	4.8454296E-02	-0.	1.0299936E-02	5.8730658E-03	3.9352935E-03
14	1.1514846E 00	4.9141326E-02	-0.	5.9636414E-02	2.9411358E-02	9.2318314E-03
15	6.8556518E-01	5.3247583E-02	-0.	2.6858045E-01	1.3947285E-01	1.1147403E-02
16	5.8028716E-01	5.4247941E-02	-0.	5.3320113E-01	2.7201084E-01	1.7113126E-02
17	5.1878334E-01	5.4845691E-02	-0.	2.0077846E 00	1.0030483E 00	4.9072517E-02
18	4.5759720E-01	5.5450390E-02	-0.	2.6979635E 00	1.3383356E 00	4.6879547E-02
19	3.8823067E-01	5.6148333E-02	-0.	3.1231936E 00	1.5480207E 00	3.5679589E-02
20	2.9831163E-01	5.7073218E-02	-0.	3.5064076E 00	1.7359940E 00	2.2051934E-02
21	2.1284788E-01	5.7974023E-02	-0.	3.5833077E 00	1.7720445E 00	1.1069506E-02
22	1.5010774E-01	5.8649218E-02	-0.	3.3002357E 00	1.6321367E 00	4.7979034E-03
23	9.6512319E-02	5.9235558E-02	-0.	2.6962817E 00	1.3321194E 00	1.4514933E-03

K EFF = 7.3625457E-01

FEW GRØUP LIMITS MULTIGRØUPS 6 10 14 23

GRØUP- - - -	1	2	3	4
FISSIØN	3.36055E-02	9.19384E-02	1.36205E-01	4.74506E-01
ABSØRPTIØN	1.76377E-02	5.64483E-02	8.75100E-02	2.37157E-01
LEAKAGE	4.86417E-01	7.57446E-02	3.08043E-02	8.28082E-03
D	1.45330E 00	5.93046E-01	5.48772E-01	2.30285E-01
NV SIGMA F	3.60231E-03	3.47117E-02	1.18874E-01	7.27262E-01
SIGMA A	1.89065E-03	2.13123E-02	7.63751E-02	3.63485E-01
SIGMA TR	2.62244E-01	5.62244E-01	6.07890E-01	1.58875E 00
D B2	5.21410E-02	2.85977E-02	2.68847E-02	1.26918E-02
B 2	3.58776E-02	4.82218E-02	4.89907E-02	5.51133E-02
1/(3*D)	2.29363E-01	5.62070E-01	6.07417E-01	1.44748E 00

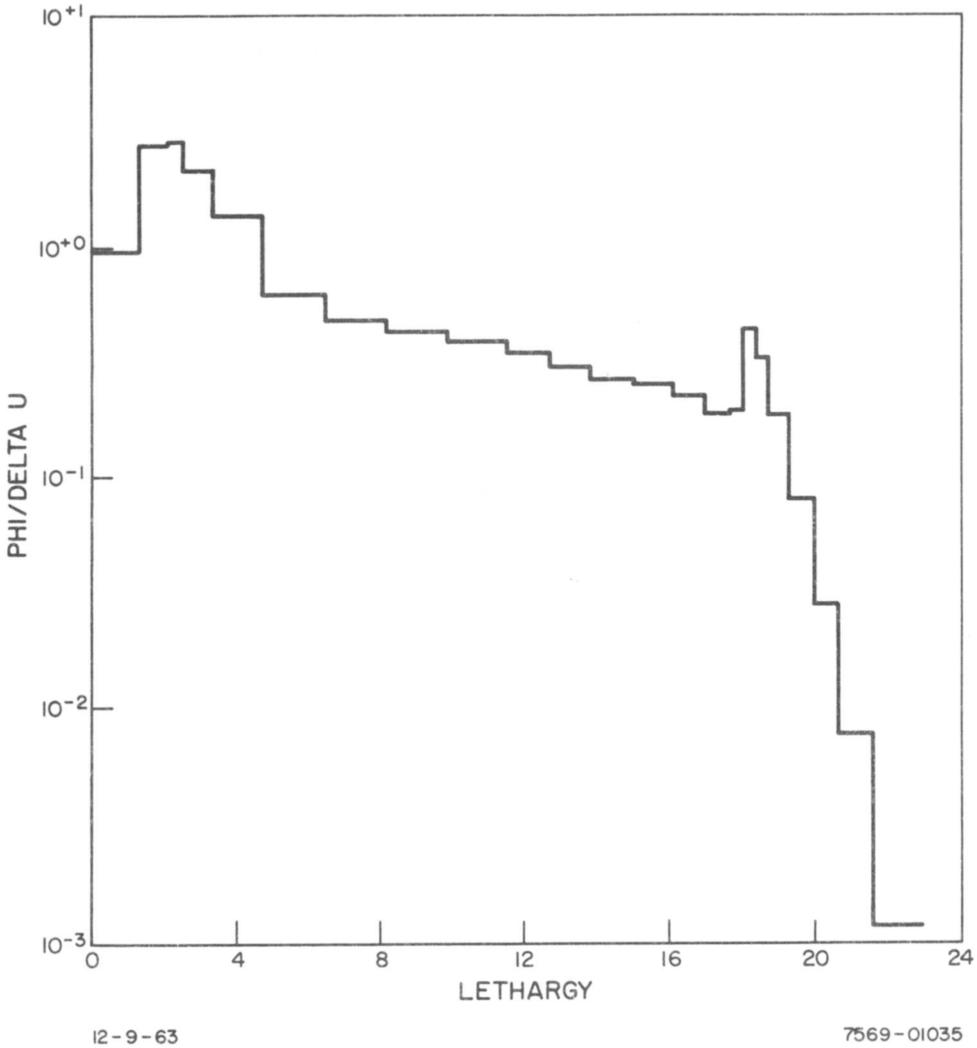
FEW GRØUP FLUX

1	2	3	4
9.3288860E 00	2.6486259E 00	1.1457919E 00	6.5245466E-01

FEW GRØUP SCATTERING MATRIX

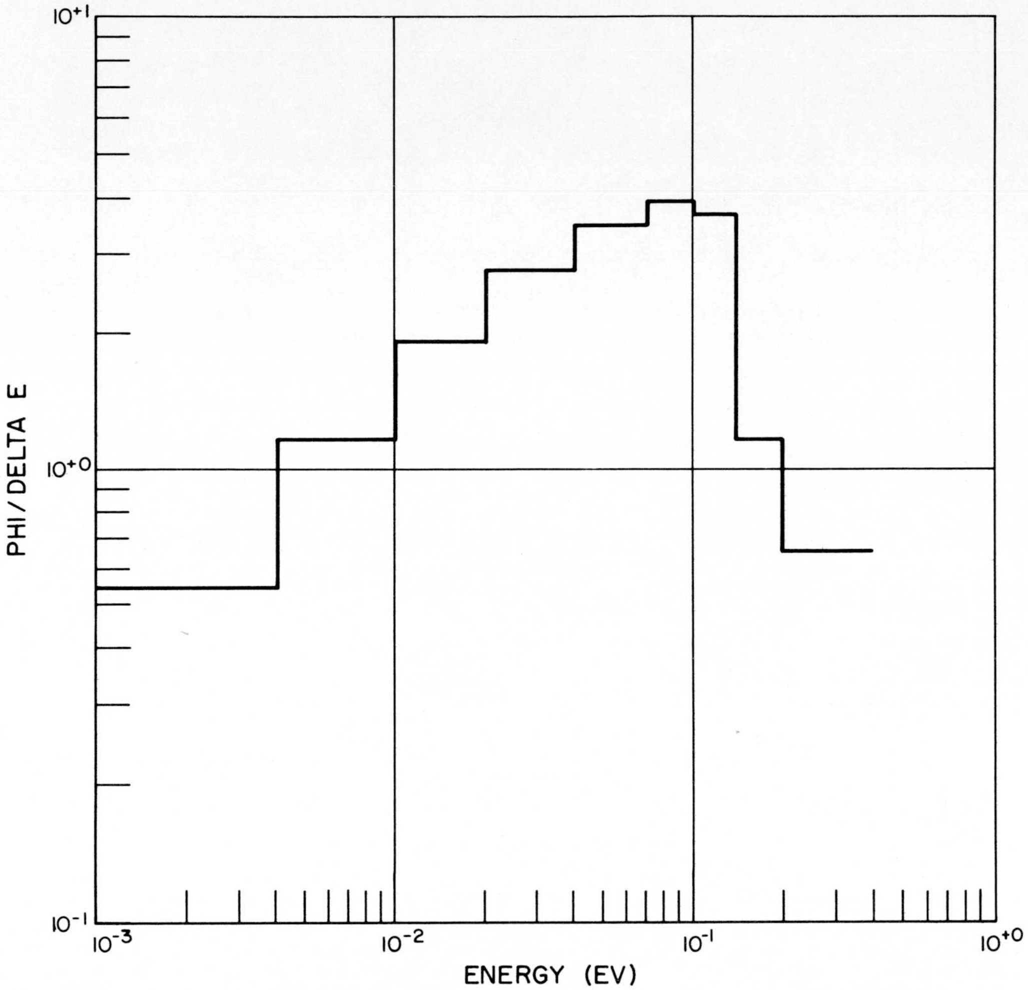
FRØM - - -

TØ - -	1	2	3	4
1	0.	0.	0.	0.
2	5.3087758E-02	0.	0.	0.
3	7.4569820E-05	1.3584783E-01	0.	0.
4	0.	1.2257897E-03	2.1137476E-01	0.



12-9-63 7569-01035

Sample Problem — Read New Elements — Code Calc.



12-9-63 7569-01047

Sample Problem — Read New Elements — Code Calc.